Routing and Bridging Guide, Cisco ACE Application Control Engine

for the Cisco ACE Application Control Engine Module and Cisco ACE 4700 Series Application Control Engine Appliance

Software Version A5(1.0)
September 2011

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-25327-01

Routing and Bridging Guide, Cisco ACE Application Control Engine

CONTENTS

Preface

This guide describes how to configure the routing and bridging features on the following products:

- Cisco ACE Application Control Engine Module (ACE module) in the Catalyst 6500 series switch or Cisco 7600 series router
- Cisco ACE 4700 Series Application Control Engine Appliance (ACE appliance)

The information in this guide applies to both the ACE module and the ACE appliance unless otherwise noted.

You configure the ACE by using the following interfaces:

- The command-line interface (CLI), a line-oriented user interface that provides commands for configuring, managing, and monitoring the ACE.
- (ACE appliance only) Device Manager graphic user interface (GUI), a Web browser-based GUI interface that provides a graphical user interface for configuring, managing, and monitoring the ACE appliance.
- Cisco Application Networking Manager (ANM), a networking management application for monitoring and configuring network devices, including the ACE.

This preface contains the following major sections:

- Symbols and Conventions
- Obtaining Documentation, Obtaining Support, and Security Guidelines

Audience

This guide is intended for the following trained and qualified service personnel who are responsible for configuring the ACE:

- Web master
- System administrator
- System operator

How to Use This Guide

This guide is organized as follows:

Chapter	Description
Chapter 1, Configuring ACE Appliance Ethernet Interfaces	(ACE appliance only) Describes how to configure the Ethernet ports on the ACE appliance.
Chapter 2, Overview of IPv6	Describes IPv6, including transitioning IPv4 networks to IPv6, IPv6 header format, IPv6 addressing, and suported protocols.
Chapter 3, Configuring VLAN Interfaces	Describes how to configure VLANs on the ACE.
Chapter 4, Configuring Routes on the ACE	Describes how to configure default and static routes.
Chapter 5, Bridging Traffic	Describes how to configure transparent (bridge) mode and a bridge-group virtual interface.

Chapter	Description
Chapter 6, Configuring Neighbor Discovery	Describes how to configure neighbor discovery (ND), including router discovery (RD) and duplicate address detectioon (DAD)
Chapter 7, Configuring ARP	Describes how to configure Address Resolution Protocol (ARP) parameters and enable ARP inspection.
Chapter 8, Configuring the DHCP Relay	Describes how to configure a Dynamic Host Configuration Protocol (DHCP) relay agent.
Appendix A, IPv4 Addresses, Protocols, and Ports Reference	Provides a reference for the following: • IP addresses and subnet masks • Protocols and applications • TCP and UDP ports • ICMP types

Related Documentation

In addition to this document, the ACE documentation set includes the following:

Document Title	Description
Administration Guide, Cisco ACE Application Control Engine	Describes how to perform the following administration tasks on the ACE: • Setting up the ACE • Establishing remote access • Managing software licenses • Configuring class maps and policy maps • Managing the ACE software • Configuring SNMP • Configuring redundancy • Configuring the XML interface • Upgrading the ACE software
Application Acceleration and Optimization Guide, Cisco ACE 4700 Series Application Control Engine Appliance	(ACE appliance only) Describes how to configure the web optimization features of the ACE appliance. This guide also provides an overview and description of those features.
Cisco Application Control Engine (ACE) Configuration Examples Wiki	Provides examples of common configurations for load balancing, security, SSL, routing and bridging, virtualization, and so on.
Cisco Application Control Engine (ACE) Troubleshooting Wiki	Describes the procedures and methodology in wiki format to troubleshoot the most common problems that you may encounter during the operation of your ACE.
Command Reference, Cisco ACE Application Control Engine	Provides an alphabetical list and descriptions of all CLI commands by mode, including syntax, options, and related commands.

Document Title	Description
CSM-to-ACE Conversion Tool Guide, Cisco ACE Application Control Engine Module	(ACE module only) Describes how to use the CSM-to-ACE module conversion tool to migrate Cisco Content Switching Module (CSM) running- or startup-configuration files to the ACE module.
CSS-to-ACE Conversion Tool Guide, Cisco ACE Application Control Engine	Describes how to use the CSS-to-ACE conversion tool to migrate Cisco Content Services Switches (CSS) running-configuration or startup-configuration files to the ACE.
Device Manager Guide, Cisco ACE 4700 Series Application Control Engine Appliance	(ACE appliance only) Describes how to use the Device Manager GUI, which resides in flash memory on the ACE appliance, to provide a browser-based interface for configuring and managing the appliance.
Getting Started Guide, Cisco ACE Application Control Engine Module	(ACE module only) Describes how to perform the initial setup and configuration tasks for the ACE module.
Getting Started Guide, Cisco ACE 4700 Series Application Control Engine Appliance	(ACE appliance only) Describes how to use the ACE appliance Device Manager GUI and CLI to perform the initial setup and configuration tasks.
Hardware Installation Guide, Cisco ACE 4710 Application Control Engine Appliance	(ACE appliance only) Provides information for installing the ACE appliance.
Installation Note, Cisco ACE Application Control Engine ACE30 Module	(ACE module only) Provides information for installing the ACE module into the Catalyst 6500 series switch or a Cisco 7600 series router.
Regulatory Compliance and Safety Information, Cisco ACE 4710 Application Control Engine Appliance	(ACE appliance only) Regulatory compliance and safety information for the ACE appliance.

Document Title	Description
Release Note, Cisco ACE 4700 Series Application Control Engine Appliance	(ACE appliance only) Provides information about operating considerations, caveats, and command-line interface (CLI) commands for the ACE appliance.
Release Note, Cisco ACE Application Control Engine Module	(ACE module only) Provides information about operating considerations, caveats, and command-line interface (CLI) commands for the ACE module.
Security Guide, Cisco ACE Application Control Engine	Describes how to perform the following ACE security configuration tasks: • Security access control lists (ACLs) • User authentication and accounting using a Terminal Access Controller Access Control System Plus (TACACS+), Remote Authentication Dial-In User Service (RADIUS), or Lightweight Directory Access Protocol (LDAP) server • Application protocol and HTTP deep packet inspection • TCP/IP normalization and termination parameters • Network Address Translation (NAT)
Server Load-Balancing Guide, Cisco ACE Application Control Engine	Describes how to configure the following server load-balancing features on the ACE: • Real servers and server farms • Class maps and policy maps to load balance traffic to real servers in server farms • Server health monitoring (probes) • Stickiness • Dynamic workload scaling (DWS) • Firewall load balancing • TCL scripts

Document Title	Description
SSL Guide, Cisco ACE Application Control Engine	Describes how to configure the following Secure Sockets Layer (SSL) features on the ACE: • SSL certificates and keys • SSL initiation • SSL termination • End-to-end SSL
System Message Guide, Cisco ACE Application Control Engine	Describes how to configure system message logging on the ACE. This guide also lists and describes the system log (syslog) messages generated by the ACE.
Upgrade/Downgrade Guide, Cisco ACE 4700 Series Application Control Engine Appliance	(ACE appliance only) Describes how to perform an ACE appliance software upgrade or downgrade.
User Guide, Cisco Application Networking Manager	Describes how to use Cisco Application Networking Manager (ANM), a networking management application for monitoring and configuring network devices, including the ACE.
Virtualization Guide, Cisco ACE Application Control Engine	Describes how to operate your ACE in a single context or in multiple contexts.

Symbols and Conventions

This publication uses the following conventions:

Convention	Description
boldface font	Commands, command options, and keywords are in **boldface**. Bold text also indicates a command in a paragraph.
italic font	Arguments for which you supply values are in *italics*. Italic text also indicates the first occurrence of a new term, book title, emphasized text.
[ ]	Elements in square brackets are optional.
{x | y | z}	Alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`screen` font	Terminal sessions and information the system displays are in `screen` font.
`boldface screen` font	Information you must enter in a command line is in **`boldface screen`** font.
italic screen font	Arguments for which you supply values are in *italic screen* font.
^	The symbol ^ represents the key labeled Control—for example, the key combination ^D in a screen display means hold down the Control key while you press the D key.
< >	Nonprinting characters, such as passwords are in angle brackets.

This document uses the following conventions:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

For additional information about CLI syntax formatting, see the corresponding *Command Reference, Cisco ACE Application Control Engine*.

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly *What's New* in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

CHAPTER 1

Configuring ACE Appliance Ethernet Interfaces

Note The information in this chapter applies to the ACE appliance only. The ACE appliance supports IPv6 or IPv4 for all the features described in this chapter unless otherwise noted.

The ACE appliance provides physical Ethernet ports that allow you to connect servers, PCs, routers, and other devices to the ACE appliance. The ACE appliance supports four Layer 2 Ethernet ports for Layer 2 switching.

You can configure the ACE appliance's four Ethernet ports to provide an interface for connecting to 10-Mbps, 100-Mbps, or 1000-Mbps networks. Each Layer 2 Ethernet port supports autonegotiate, full-duplex, or half-duplex operation on an Ethernet LAN and can have traffic within a designated VLAN.

A Layer 2 Ethernet port can be configured as follows:

- Member of Port-Channel Group—Associates a physical port on the ACE appliance to a logical port to create a port-channel logical interface. The VLAN association is derived from the port-channel configuration. The port is configured as a Layer 2 EtherChannel, where each EtherChannel bundles the individual physical Ethernet ports into a single logical link that provides the aggregate bandwidth of up to four physical links on the ACE appliance.
- Access VLAN—Provides a connection for end users or node devices, such as a router or server. The access VLAN port is assigned to a single VLAN.

- Trunk port—Allocates VLANs to ports and passes VLAN information (including VLAN identification) between switches for all Ethernet channels defined in a Layer 2 Ethernet port or a Layer 2 EtherChannel (port-channel) group on the ACE appliance. The port is associated with IEEE 802.1Q encapsulation-based VLAN trunking.

This chapter describes how to configure the Ethernet ports on the ACE appliance. It contains the following major sections:

After you configure the Ethernet ports on the ACE appliance and allocate VLANs to configured Ethernet ports, you create the corresponding VLAN interfaces on the ACE appliance as described in Chapter 3, Configuring VLAN Interfaces.

Ethernet Interface Configuration Quick Start

Table 1-1 provides a quick overview of the steps required to configure Ethernet interface ports on the ACE appliance. Each step includes the CLI command or a reference to the procedure required to complete the task. For a complete description of each feature and all the options associated with the CLI commands, see the sections following Table 1-1.

Table 1-1 Ethernet Interface Configuration Quick Start

Task and Command Example

1. Enter global configuration mode.
`host1/Admin# config`
`host1/Admin(config)#`

2. Configure a Layer 2 Ethernet port on the ACE appliance. You enter the interface mode.
`host1/Admin(config)# interface gigabitEthernet 1/3`
`host1/Admin(config-if)#`
Note Only users authenticated in the Admin context can use the **interface gigabitEthernet** command.

3. (Optional) Add a description about the Ethernet port to help you remember its function. `host1/Admin(config-if)# description Ethernet port 3 is configured for speeds of 1000 Mbps`

4. Configure the interface duplex and speed (the default is autonegotiate).
`host1/Admin(config-if)# speed 1000M`
`host1/Admin(config-if)# duplex full`

5. If you are using your ACE appliance in a redundancy configuration, configure one of the Ethernet ports on the ACE appliance for fault tolerance using a dedicated fault-tolerant (FT) VLAN for communication between the members of an FT group. `host1/Admin(config-if)# ft-port vlan 60` **Note** You may configure a port-channel interface on the ACE appliance for fault tolerance instead of an Ethernet port (see Table 1-2).
6. (Optional) Add a configurable delay at the physical port level to address any issues with transition time, based on the variety of peers. `host1/Admin(config-if)# carrier-delay 60`
7. (Optional) Map the physical Ethernet port to a port channel to automatically create a port-channel logical interface (see Table 1-2 for more information). `host1/Admin(config-if)# channel-group 255`

Table 1-1 Ethernet Interface Configuration Quick Start (continued)

Task and Command Example
8. (Optional) Enables VLAN Classes of Service (CoS) bits-based quality of service (QoS) to the configured physical Ethernet port. `host1/Admin(config-if)# qos trust cos`

9. Enable the Ethernet port to put the interface in the Up administrative state.
`host1/Admin(config-if)# no shutdown`
`host1/Admin(config-if)# exit`
`host1/Admin(config)#`

10. (Optional) Assign an access port to a specific VLAN for the Ethernet port. For example, to specify VLAN 101 as an access port for Ethernet port 3, enter:
`host1/Admin(config)# interface gigabitEthernet 1/3`
`host1/Admin(config-if)# switchport access vlan 101`
Note If you assign a VLAN as the access port for a specific Ethernet port, the VLAN is reserved and cannot be configured for a VLAN trunk.

11. Selectively allocate individual VLANs to a trunk link. For example, to add VLANs 200 and 266 to the defined list of VLANs currently set for Ethernet port 3, enter:
`host1/Admin(config)# interface gigabitEthernet 1/3`
`host1/Admin(config-if)# switchport trunk allowed vlan 200,266`
Note When allocating VLANs to ports, overlapping is not allowed. For example, if VLAN 200 is associated with Ethernet port 3, you cannot associate VLAN 200 with another Ethernet port or port channel.

12. (Optional) Set the 802.1Q native VLAN for a trunk. For example, to specify VLAN 266 as the 802.1Q native VLAN for the trunk, enter:
`host1/Admin(config)# interface gigabitEthernet 1/3`
`host1/Admin(config-if)# switchport trunk native vlan 266`

13. Enable VLAN trunking in a Layer 2 Ethernet port.
`host1/Admin(config-if)# no shutdown`
`host1/Admin(config-if)# exit`
`host1/Admin(config)#`

Table 1-1 Ethernet Interface Configuration Quick Start (continued)

Task and Command Example
14. Create the corresponding VLAN interfaces on the ACE appliance. For details, see Chapter 3, Configuring VLAN Interfaces.
15. (Optional) Save your configuration changes to Flash memory. `host1/Admin# copy running-config startup-config`

Table 1-2 provides a quick overview of the steps required to configure an Ethernet interface port on the ACE appliance as a Layer 2 EtherChannel (port channel). Each step includes the CLI command or a reference to the procedure required to complete the task. For a complete description of each feature and all the options associated with the CLI commands, see the sections following Table 1-2.

Table 1-2 EtherChannel (Port Channel) Configuration Quick Start

Task and Command Example

1. Enter global configuration mode.
`host1/Admin# config`
`host1/Admin(config)#`

2. (Optional) Create a port-channel interface to group physical ports together on the ACE appliance to form an EtherChannel.
`host1/Admin(config)# interface port-channel 255`
`host1/Admin(config-if)#`
Note Only users authenticated in the Admin context can use the **interface port-channel** command.

3. (Optional) Add a description about a port-channel interface to help you remember its function. `host1/Admin(config-if)# description A port-channel interface with a channel number of 255`

Table 1-2 EtherChannel (Port Channel) Configuration Quick Start (continued)

Task and Command Example
4. If you are using your ACE appliance in a redundancy configuration, configure a port-channel interface on the ACE appliance for fault tolerance using a dedicated fault-tolerant (FT) VLAN for communication between the members of an FT group. `host1/Admin(config-if)# ft-port vlan 60` **Note** You may configure an Ethernet interface on the ACE appliance for fault tolerance instead of a port-channel interface (see Table 1-1).
5. (Optional) Set the load-distribution method among the ports in the EtherChannel bundle. For example, to configure an EtherChannel to balance the traffic load across the links using source or destination IP addresses, enter: `host1/Admin(config-if)# port-channel load-balance src-dst-ip`

6. (Optional) Enable the port-channel interface to put the interface in the Up administrative state.
 	
`host1/Admin(config-if)# no shutdown`
`host1/Admin(config-if)# exit`
`host1/Admin(config)#`

7. (Optional) Assign an access port to a specific VLAN for the Layer 2 port-channel interface. For example, to specify VLAN 101 as an access port for port-channel interface 255, enter:
 	
`host1/Admin(config)# interface port-channel 255`
`host1/Admin(config-if)# switchport access vlan 101`
 	
Note If you assign a VLAN as the access port for a specific port-channel interface, the VLAN is reserved and cannot be configured for a VLAN trunk.

Table 1-2 EtherChannel (Port Channel) Configuration Quick Start (continued)

Task and Command Example

8. Selectively allocate individual VLANs to a trunk link. For example, to add VLANs 200 and 266 to the defined list of VLANs currently set for port-channel interface 255, enter:
`host1/Admin(config)# interface port-channel 255`
`host1/Admin(config-if)# switchport trunk allowed vlan 200,266`
Note When allocating VLANs to ports, overlapping is not allowed. For example, if VLAN 200 is associated with port-channel 255 you cannot associate VLAN 200 with another Ethernet port or port channel.

9. (Optional) Set the 802.1Q native VLAN for a trunk. For example, to specify VLAN 266 as the 802.1Q native VLAN for the trunk, enter:
`host1/Admin(config)# interface port-channel 255`
`host1/Admin(config-if)# switchport trunk native vlan 266`

10. Enable VLAN trunking in a Layer 2 port-channel interface.
`host1/Admin(config-if)# no shutdown`
`host1/Admin(config-if)# exit`
`host1/Admin(config)#`

11. Create the corresponding VLAN interfaces on the ACE appliance. For details, see Chapter 3, Configuring VLAN Interfaces.
12. (Optional) Save your configuration changes to the Flash memory. `host1/Admin# copy running-config startup-config`

Configuring a Layer 2 Ethernet Port

Four Ethernet ports allow you to connect servers, PCs, routers, and other devices to the ACE appliance. You can configure the four Ethernet ports to provide an interface for connecting to 10-Mbps, 100-Mbps, or 1000-Mbps networks. Each

Layer 2 Ethernet port supports autonegotiation (default), full-duplex, or half-duplex operation on an Ethernet LAN and can have traffic within a designated VLAN.

To configure a Layer 2 Ethernet port on the ACE appliance, use the **interface gigabitEthernet** command in configuration mode. The ACE appliance enters the interface configuration mode. Only users authenticated in the Admin context can use the **interface gigabitEthernet** command.

The syntax for the command is as follows:

interface gigabitEthernet *slot_number/port_number*

The keywords, arguments, and options are as follows:

- *slot_number*—Physical slot on the ACE appliance containing the Ethernet ports. This selection is always 1, which is the location of the daughter card in the ACE appliance. The daughter card includes the four Layer 2 Ethernet ports that allow you to perform Layer 2 switching.
- *port_number*—Physical Ethernet port on the ACE appliance. Valid selections are from 1 through 4, which allow you to specify one of the four Ethernet ports (1, 2, 3, or 4) associated with the slot 1 (daughter card) selection.

For example, to configure Ethernet port 3 and access the interface configuration mode, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)#
```

You can use the additional CLI commands in interface configuration mode to configure specific Ethernet port settings for the ACE appliance.

This section contains the following topics:

- Adding a Description for an Ethernet Port
- Configuring the Ethernet Interface Speed and Duplex Mode
- Designating an Ethernet Port as an FT VLAN Port
- Configuring a Delay at the Physical Port Level
- Configuring an Ethernet Port in a Port-Channel Group
- Enabling Quality of Service for a Port
- Enabling or Disabling the Ethernet Interface

You can also configure an Ethernet port using the following CLI commands in interface mode:

- Use the **interface port-channel** command to group physical ports on the ACE appliance to form an EtherChannel (or port-channel) interface. See the "Configuring Layer 2 EtherChannels" section.
- Use the **switchport access vlan** command to configure an access port to a specific VLAN for an Ethernet port. See the "Configuring a VLAN Access Port" section.
- Use the **switchport trunk allowed vlan** command to allocate VLANs to a Layer 2 Ethernet port. See the "Allocating an Ethernet Port or Port-Channel Interface to a VLAN Trunk" section.
- Use the **switchport trunk native vlan** command to set the 802.1Q native VLAN for a trunk. See the "Specifying the 802.1Q Native VLAN For a Trunk" section.

Adding a Description for an Ethernet Port

You can add a description about an Ethernet port to help you remember its function. The interface description appears in the output of the **show running-config** and **show interfaces** commands in Exec mode.

The syntax for the command is as follows:

description *text*

Use the *text* argument to enter an unquoted text string with a maximum of 240 alphanumeric characters.

For example, to add a description for Ethernet port 1, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# description Ethernet port 3 is configured for
speeds of 1000 Mbps
```

To remove the interface description, enter:

```
host1/Admin(config-if)# no description
```

Configuring the Ethernet Interface Speed and Duplex Mode

By default, the ACE appliance automatically uses the autonegotiate setting for Ethernet port speed and duplex mode parameters to allow the ACE appliance to negotiate the speed and duplex mode between ports. If you manually configure the port speed and duplex modes, follow these guidelines:

- The ACE appliance prevents you from making a duplex setting when you configure the speed of an Ethernet port to **auto**. You can configure the **speed** command with a setting of 10, 100, or 1000 Mbps to configure duplex mode for the Ethernet port.
- If you configure an Ethernet port speed to a value other than **auto** (for example, 10, 100, or 1000 Mbps), ensure that you configure the connecting port to match. Do not configure the connecting port to negotiate the speed through the **auto** keyword.
- The ports on both ends of a link must have the same setting. The link will not come up if the ports at each end of the connecting interface has a different setting. For example, if you configure the Ethernet port speed and duplex setting to a setting of 10, 100, or 1000 Mbps on one side of a link, you must configure the matching speed and duplex on the other side of the link to ensure proper communication.
- If you enter the **no speed** command, the ACE appliance automatically configures both the speed and duplex settings to auto.

The ACE appliance cannot automatically negotiate the interface speed and duplex mode if either connecting interface is configured to a value other than **auto**.

Caution Changing the Ethernet port speed and duplex mode configuration may shut down and reenable the interface during the reconfiguration.

This section contains the following topics:

Configuring the Ethernet Interface Speed

You can configure the Ethernet port speed for a setting of 10, 100, or 1000 Mbps. Use the **speed** command in interface configuration mode to configure the port speed. The default speed for an ACE appliance interface is autonegotiate.

The syntax for the command is as follows:

speed {**1000M** | **100M** | **10M** | **auto**}

The keywords, arguments, and options are as follows:

- **1000M**—Initiates 1000 Mbps operation.
- **100M**—Initiates 100 Mbps operation.
- **10M**—Initiates 10 Mbps operation.
- **auto**—Enables the ACE appliance to autonegotiate with other devices for speeds of 10, 100, or 1000 Mbps. If you set the Ethernet port speed to **auto**, the ACE appliance automatically sets the duplex mode to auto; **auto** is the default setting.

Note If you configure the Ethernet port speed to **auto**, the ACE appliance automatically sets the duplex mode to auto.

For example, to set the speed to 1000 Mbps on Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# speed 1000M
```

To restore the default setting of autonegotiate for an Ethernet port, enter:

```
host1/Admin(config-if)# no speed
```

Note If you enter the **no speed** command, the ACE appliance automatically configures both the speed and duplex settings to autonegotiate.

Setting the Interface Duplex Mode

To configure an Ethernet port for full or half duplex operation, use the **duplex** command in interface configuration mode. The default configuration for an ACE appliance interface is autonegotiate.

Note If you configure the Ethernet port speed to **auto** on a 10/100/1000-Mbps Ethernet port, both speed and duplex are autonegotiated. You cannot change the duplex mode of autonegotiation ports.

The syntax for the command is as follows:

duplex {**full** | **half**}

The keywords, arguments, and options are as follows:

- **full**—Configures the specified Ethernet port for full-duplex operation, which allows data to travel in both directions at the same time.
- **half**—Configures the specified Ethernet port for half-duplex operation. A half-duplex setting ensures that data only travels in one direction at any given time.

For example, to set the duplex mode to full on Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# duplex full
```

To restore the default setting of autonegotiate for an Ethernet port, enter:

```
host1/Admin(config-if)# no duplex
```

Designating an Ethernet Port as an FT VLAN Port

Peer ACE appliances can communicate with each other over a dedicated fault-tolerant (FT) VLAN. These redundant peers use an FT VLAN to transmit and receive heartbeat packets and state and configuration replication packets. To configure one of the Ethernet ports on the ACE appliance for fault tolerance using a dedicated FT VLAN for communication between the members of an FT group, use the **ft-port vlan** command in interface configuration mode.

Note When you specify the **ft-port vlan** command, the ACE appliance modifies the associated Ethernet port to a trunk port.

On both peer ACE appliances, you must configure the same Ethernet port as the FT VLAN port. For example, if you configure ACE appliance 1 to use Ethernet port 4 as the FT VLAN port, then be sure to configure ACE appliance 2 to use Ethernet port 4 as the FT VLAN port.

For details on configuring redundant ACE appliances, including an FT VLAN, see the *Administration Guide, Cisco ACE Application Control Engine.*

The syntax for this command is as follows:

ft-port vlan *number*

The *number* argument specifies a unique identifier for the FT VLAN. Valid values are from 2 to 4094.

Note You do not need to create an FT VLAN before you designate an Ethernet port as the FT VLAN port.

For example, to configure FT VLAN identifier 60 for Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# ft-port vlan 60
```

To remove the FT VLAN for the Ethernet port, enter:

```
host1/Admin(config-if)# no ft-port vlan 60
```

Configuring a Delay at the Physical Port Level

If you connect an ACE appliance to a Catalyst 6500 series switch, your configuration on the Catalyst may include the Spanning Tree Protocol (STP). However, the ACE appliance does not support STP. In this case, you may find that the Layer 2 convergence time is much longer than the physical port up time. For example, the physical port would normally be up within 3 seconds, but STP moving to the forward state may need approximately 30 seconds. During this transitional time, although the ACE appliance declares the port to be up, the traffic will not pass.

To add a configurable delay at the physical port level to address this transition time, based on the variety of peers, use the **carrier-delay** command.

The syntax for this command is as follows:

carrier-delay *seconds*

The *seconds* argument specifies the carrier transition delay in seconds. Valid values are 0 to 120 seconds. The default is 0 (no carrier delay).

For example, to add a configurable delay of 60 seconds at the physical port level for Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# carrier-delay 60
```

To remove the carrier delay for the Ethernet port, enter:

```
host1/Admin(config-if)# no carrier-delay 60
```

Configuring an Ethernet Port in a Port-Channel Group

You can group physical ports together on the ACE appliance to form an EtherChannel (or port channel). When configuring Layer 2 EtherChannels, you map the physical Ethernet port to a port channel using the **channel-group** command. This command configures the Ethernet port in a port-channel group and automatically creates the port-channel logical interface.

For details on creating a Layer 2 EtherChannel interface, see the "Configuring Layer 2 EtherChannels" section.

Note You do not need to configure a port-channel interface before you assign a physical Ethernet port to a channel group through the **channel-group** command. A port-channel interface is created automatically when the channel group receives its first physical interface, if it is not already created.

The syntax for the command is as follows:

channel-group *channel_number*

The *channel_number* argument specifies the channel number assigned to this channel group. Valid values are from 1 to 255.

For example, to create a channel group with a channel number of 255, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# channel-group 255
```

To remove the channel group assigned to the Ethernet port, enter:

```
host1/Admin(config-if)# no channel-group 255
```

Enabling Quality of Service for a Port

By default, Quality of Service (QoS) is disabled for each physical Ethernet port on the ACE appliance. You can enable QoS for a configured physical Ethernet port that is based on VLAN Class of Service (CoS) bits (priority bits that segment the traffic in eight different classes of service). If a VLAN header is present, the CoS bits are used by the ACE appliance to map frames into class queues. If the frame is untagged, it falls back to a default port QoS level for mapping.

Note QoS is configurable only for a physical Ethernet port and is not VLAN interface-based.

When you enable QoS on a port (a trusted port), traffic is mapped into different ingress queues based on their VLAN CoS bits. If there are no VLAN CoS bits, or QoS is not enabled on the port (untrusted port), the traffic is then mapped into the lowest priority queue.

You can enable QoS for an Ethernet port configured for fault tolerance (see the "Designating an Ethernet Port as an FT VLAN Port"). In this case, heartbeat packets are always tagged with CoS bits set to 7 (a weight of High).

Note We recommend that you enable QoS on the FT VLAN port to provide higher priority for FT traffic.

To enable QoS for a physical Ethernet port, use the **qos trust cos** command.

The syntax for this command is as follows:

qos trust cos

For example, to enable QoS at the physical port level for Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# qos trust cos
```

To disable QoS for the Ethernet port, enter:

```
host1/Admin(config-if)# no qos trust cos
```

To check if QoS is enabled or disabled for an Ethernet port, enter the **show interface gigabitEthernet** command in Exec mode. The **show interface gigabitEthernet** command display will identify whether QoS is enabled or disabled on the Ethernet port (see the "Displaying Ethernet Interface Configuration, Status, and Statistics" section).

Enabling or Disabling the Ethernet Interface

By default, when you configure an interface it remains in the shutdown state (administratively down) until you enable the interface.

- To enable an Ethernet port, use the **no shutdown** command in interface configuration mode. This action puts the interface in the Up administrative state.
- To disable an Ethernet port, use the **shutdown** command in interface configuration mode. This action puts the interface in the Down administrative state.

For example, to enable Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# no shutdown
```

To disable Ethernet port 3, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/3
host1/Admin(config-if)# shutdown
```

To check if an interface is disabled, enter the **show interface gigabitEthernet** command in Exec mode. An interface that has been shut down is shown as administratively down in the **show interface gigabitEthernet** command display. See the "Specifying the 802.1Q Native VLAN For a Trunk" section for details.

Configuring Layer 2 EtherChannels

An EtherChannel bundles individual Layer 2 Ethernet physical ports into a single logical link that provides the aggregate bandwidth of up to four physical links on the ACE appliance. The EtherChannel provides full-duplex bandwidth up to 4000-Mbps between the ACE appliance and another switch (for example, a Cisco

Catalyst 6500 series switch). Ports in an EtherChannel do not have to be contiguous; however, all ports in each EtherChannel must operate at the same speed.

Note The Catalyst 6500 series switch uses a proprietary protocol called Port Aggregation Protocol (PAgP). The IEEE later defined within 802.3ad, a new control protocol for link aggregation called Link Aggregate Control Protocol (LACP). The ACE appliance does not use either protocol. If you intend to configure Layer 2 EtherChannel bundles between an ACE appliance and a Catalyst 6500 series switch, all ports in the bundle must be statically assigned at both ends. See the "Example of a Port-Channel Configuration" section for details.

To create the EtherChannel interface, use the **interface port-channel** command in interface configuration mode. You can base the load-balance policy (frame distribution) on a MAC address (Layer 2), an IP address (Layer 3), or a port number (Layer 4).

Note Only users authenticated in the Admin context can use the **interface port-channel** command.

The EtherChannel interface (consisting of up to four Ethernet interfaces) is treated as a single interface, which is called a port channel. You configure an EtherChannel on the port-channel interface rather than on the individual member Ethernet interfaces. Each EtherChannel has a numbered port-channel interface, numbered from 1 to 255. After you configure an EtherChannel, the configuration that you apply to the assigned Ethernet ports in the port-channel group affects only those Ethernet ports.

Note You do not need to configure a port-channel interface before you assign a physical Ethernet port to a channel group through the **channel-group** command. A port-channel interface is created automatically when the channel group receives its first physical interface, if it is not already created.

To change the parameters of all ports in an EtherChannel, apply the configuration commands to the port-channel interface to configure a Layer 2 EtherChannel as a trunk.

In addition, you can configure EtherChannels as trunks (see Chapter 3, Configuring VLAN Interfaces). After a port channel is formed, configuring any port in the channel as a trunk applies the configuration to all ports in the EtherChannel.

Note If you disable a port in a channel, it is treated as a link failure and its traffic is transferred to one or more of the remaining ports in the channel.

You can also configure EtherChannels using the following CLI commands in interface mode:

- Use the **switchport access vlan** command to configure an access port to a specific VLAN for the Layer 2 EtherChannel interface. See the "Configuring a VLAN Access Port" section.
- Use the **switchport trunk allowed vlan** command to allocate VLANs for the Layer 2 EtherChannel interface. See the "Allocating an Ethernet Port or Port-Channel Interface to a VLAN Trunk" section.
- Use the **switchport trunk native vlan** command to set the 802.1Q native VLAN for a trunk. See the "Specifying the 802.1Q Native VLAN For a Trunk" section.

This section contains the following topics:

Configuring a Port-Channel Interface

You can group physical ports together on the ACE appliance to form an EtherChannel (or port channel). All the ports that belong to the same port channel must be configured with the same values; for example, port parameters, VLAN membership, or trunk configuration. Only one port channel in a channel group is allowed, and a physical port can belong to a single port-channel interface only.

Note If you use SNMP to query the ACE, be aware that the SNMP OID ifHighSpeed is not supported for an interface configured as a port channel. An SNMP request for ifHighSpeed on a port channel interface will return a value of zero (0).

To create a port-channel interface, use the **interface port-channel** command. Only users authenticated in the Admin context can use this command.

The syntax for the command is as follows:

interface port-channel *channel_number*

The *channel_number* argument specifies the channel number assigned to this port-channel interface. Valid values are from 1 to 255.

For example, to create a port-channel interface with a channel number of 255, enter:

```
host1/Admin(config)# interface port-channel 255
```

Adding a Description for a Port Channel

You can add a description about a port-channel interface to help you remember its function. The port-channel interface description appears in the output of the **show running-config** and **show interfaces** commands in Exec mode.

The syntax for the command is as follows:

description *text*

Use the *text* argument to enter an unquoted text string with a maximum of 240 alphanumeric characters.

For example, to add a description for port-channel interface 255, enter:

```
host1/Admin(config)# interface port-channel 255
host1/Admin(config-if)# description A port-channel interface with a
channel number of 255
```

To remove the port-channel description, enter:

```
host1/Admin(config-if)# no description
```

Designating a Port-Channel Interface as an FT VLAN Interface

Peer ACE appliances can communicate with each other over a dedicated fault-tolerant (FT) VLAN. These redundant peers use an FT VLAN to transmit and receive heartbeat packets and state and configuration replication packets. To configure a port-channel interface on the ACE appliance for fault tolerance using a dedicated FT VLAN for communication between the members of an FT group, use the **ft-port vlan** command in interface configuration mode.

Note When you specify the **ft-port vlan** command, the ACE appliance modifies the associated port-channel interface to a trunk port.

On both peer ACE appliances, you must configure the same port-channel interface as the FT VLAN. For example, if you configure ACE appliance 1 to use port-channel interface 255 as the FT VLAN port, you must configure ACE appliance 2 to use port-channel interface 255 as the FT VLAN.

For details on configuring redundant ACE appliances, including an FT VLAN, see the *Administration Guide, Cisco ACE Application Control Engine.*

The syntax for this command is as follows:

ft-port vlan *number*

The *number* argument specifies a unique identifier for the FT VLAN. Valid values are from 2 to 4094.

Note You do not need to create an FT VLAN before you designate a port-channel interface as the FT VLAN port.

For example, to configure FT VLAN identifier 60 for port-channel interface 255, enter:

```
host1/Admin(config)# interface port-channel 255
host1/Admin(config-if)# ft-port vlan 60
```

To remove the FT VLAN for the port-channel interface, enter:

```
host1/Admin(config-if)# no ft-port vlan 60
```

Configuring Port-Channel Load Balancing

An EtherChannel can balance the traffic load across the links in the designated port channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. Port-channel load balancing can use MAC addresses or IP addresses, Layer 4 port numbers, source addresses, destination addresses, or both source and destination addresses. Addresses can be either IPv4 or IPv6.

Use the option that provides the load-balance criteria with the greatest variety in your configuration. For example, if the traffic on an EtherChannel is going to a single MAC address only and you use the destination MAC address as the basis of EtherChannel load balancing, the EtherChannel always chooses the same link in the EtherChannel.

To set the load-distribution method among the ports in the EtherChannel bundle, use the **port-channel load-balance** command.

The syntax for this command is as follows:

port-channel load-balance {**dst-ip** | **dst-mac** | **dst-port** | **src-dst-ip** | **src-dst-mac** | **src-dst-port** | **src-ip** | **src-mac** | **src-port**}

The keywords, arguments, and options are as follows:

- **dst-ip**—Loads the distribution on the destination IP address
- **dst-mac**—Loads the distribution on the destination MAC address
- **dst-port**—Loads the distribution on the destination TCP or UDP port
- **src-dst-ip**—Loads the distribution on the source or destination IP address
- **src-dst-mac**—Loads the distribution on the source or destination MAC address
- **src-dst-port**—Loads the distribution on the source or destination port
- **src-ip**—Loads the distribution on the source IP address
- **src-mac**—Loads the distribution on the source MAC address
- **src-port**—Loads the distribution on the TCP or UDP source port

For example, to configure an EtherChannel to balance the traffic load across the links using source or destination IP addresses, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/1
host1/Admin(config-if)# port-channel load-balance src-dst-ip
```

Enabling or Disabling a Port-Channel Interface

By default, when you configure a port-channel interface it remains in the shutdown state (administratively down) until you enable the interface.

- To enable a port-channel interface, use the **no shutdown** command in interface configuration mode. This action puts the interface in the Up administrative state.
- To disable a port-channel interface, use the **shutdown** command in interface configuration mode. This action puts the interface in the Down administrative state.

For example, to enable port-channel interface 255, enter:

```
host1/Admin(config)# interface port-channel 255
host1/Admin(config-if)# no shutdown
```

For example, to disable port-channel interface 255, enter:

```
host1/Admin(config)# interface port-channel 255
host1/Admin(config-if)# shutdown
```

Example of a Port-Channel Configuration

The following configuration example shows the commands required on both the ACE appliance and the Catalyst 6500 series switch to configure a port channel.

ACE Appliance Configuration

```
interface g1/1
  channel-group 1
  no shutdown
interface g1/2
  channel-group 1
  no shutdown
interface po 1
  switchport allowed vlan 10-1000
  no shutdown
```

Catalyst 6500 Series Switch Configuration

```
interface g9/37
  channel-group 1 mode on
  no shutdown
interface g9/38
  channel-group 1 mode on
  no shutdown
interface po 1
  switchport
  switchport trunk encapsulation dot1q
  switchport trunk allowed vlan 10-1000
  switchport mode trunk
  no shutdown
```

Configuring a VLAN Access Port

On the ACE appliance, a port that is assigned to a single VLAN is referred to as a VLAN access port and provides a connection for end users or node devices, such as a router or server. By default, all devices are assigned to VLAN 1, known as the default VLAN. To configure an access port to a specific VLAN for either an Ethernet interface or a Layer 2 port-channel interface, use the **switchport access vlan** command in interface configuration mode.

Note You do not need to create a VLAN interface before you configure an access VLAN. To configure a VLAN interface and access its mode to configure its attributes, use the **interface vlan** command in configuration mode for the context. See Chapter 3, Configuring VLAN Interfaces, for details.

When you assign a VLAN for a specific Ethernet port or port-channel interface, the VLAN is reserved and cannot be configured for a VLAN trunk (see the "Configuring VLAN Trunks" section). A VLAN access port and a VLAN trunk cannot coexist for the same Ethernet port or port-channel interface. If you specify both configurations for the same Ethernet port or port-channel interface, the most recent configuration will overwrite the older configuration.

Note If you have QoS enabled for a physical Ethernet port (see the "Enabling Quality of Service for a Port" section) that has been designated as an FT VLAN port (see the "Designating an Ethernet Port as an FT VLAN Port" section), do not configure this Ethernet port as a VLAN access port. In this configuration, the QoS setting for redundancy traffic, such as heartbeat packets or TCP tracking probes, may not be handled properly by the ACE appliance and FT traffic may be dropped when there is network congestion.

The syntax is as follows:

switchport access vlan *number*

The *number* argument specifies the VLAN number that you want to configure as the 802.1Q native VLAN when operating in trunking mode. Valid values are from 1 to 4094. The default is VLAN 1.

For example, to configure VLAN 101 as an access port for Ethernet port 4, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/4
host1/Admin(config-if)# switchport access vlan 101
```

For example, to configure VLAN 101 as an access port for port-channel interface 255, enter:

```
host1/Admin(config)# interface port-channel 255
host1/Admin(config-if)# switchport access vlan 101
```

To reset the access mode to the default VLAN 1, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/4
host1/Admin(config-if)# no switchport access vlan 101
```

Configuring VLAN Trunks

You can use trunk links to pass VLAN information (including VLAN identification) between switches for all Ethernet channels defined in a Layer 2 Ethernet port or a Layer 2 EtherChannel (port-channel) group on the ACE appliance. By default, a trunk port is a member of all VLANs that exist on the ACE appliance and carries traffic for those VLANs as they pass between the switches. To distinguish between the traffic flows, a trunk port marks the frames with special tags.

You must enable trunking on both sides of a link. If two switches are connected together, you must configure both switch ports for trunking and with the same tagging mechanism.

The ACE appliance supports 802.1Q encapsulation-based VLAN trunking. The 802.1Q interconnects VLANs between multiple switches, routers, and servers. With 802.1Q, you can define a VLAN topology that spans multiple physical devices. In addition, the ACE appliance supports 802.1Q for Gigabit Ethernet interfaces. An 802.1Q trunk link provides VLAN identification by adding a 2-byte tag to an Ethernet Frame as it leaves a trunk port.

Ports configured in trunk mode can have traffic in more than one VLAN based on the trunk-allowed VLAN list configuration.

Note You can configure a trunk on a single Ethernet port or on an EtherChannel.

Follow these configuration guidelines and restrictions when you use VLAN trunks with the ACE appliance:

- If you configure a VLAN on a trunk, you cannot configure the VLAN as the access port for a specific Ethernet port or port-channel interface (see the "Configuring a VLAN Access Port" section). A VLAN access port and a VLAN trunk cannot coexist for the same Ethernet port or port-channel interface. If you specify both configurations for the same Ethernet port or port-channel interface, the most recent configuration will overwrite the older configuration.
- When allocating VLANs to ports, overlapping is not allowed. For example, if VLAN 10 is associated with Ethernet port 1 (or with port-channel interface 255), you cannot associate VLAN 10 with another Ethernet port or port channel.
- You do not need to create a VLAN interface before you allocate a VLAN to an Ethernet port or a port-channel interface. To configure a VLAN interface and access its mode to configure its attributes, use the **interface vlan** command in configuration mode for the context. See Chapter 3, Configuring VLAN Interfaces for details.
- When connecting a Cisco switch through an 802.1Q trunk, make sure that the native VLAN for an 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning tree loops might result.

- When you connect two Cisco switches through 802.1Q trunks, the switches exchange spanning tree Bridge Protocol Data Units (BPDUs) on each VLAN allowed on the trunks. The BPDUs on the native VLAN of the trunk are sent untagged to the reserved 802.1D spanning-tree multicast MAC address (01-80-C2-00-00-00). The BPDUs on all other VLANs on the trunk are sent tagged to the reserved Cisco Shared Spanning Tree (SSTP) multicast MAC address (01-00-0c-cc-cc-cd).
- Non-Cisco 802.1Q switches maintain only a single instance of spanning-tree (the Mono Spanning Tree, or MST) that defines the spanning-tree topology for all VLANs. When you connect a Cisco switch to a non-Cisco switch through an 802.1Q trunk, the MST of the non-Cisco switch and the native VLAN spanning tree of the Cisco switch combine to form a single spanning-tree topology known as the Common Spanning Tree (CST).
- Because Cisco switches transmit BPDUs to the SSTP multicast MAC address on VLANs other than the native VLAN of the trunk, non-Cisco switches do not recognize these frames as BPDUs and flood them on all ports in the corresponding VLAN. Other Cisco switches connected to the non-Cisco 802.1Q cloud receive these flooded BPDUs, which allows them to maintain a per-VLAN spanning-tree topology across a cloud of non-Cisco 802.1Q switches. The non-Cisco 802.1Q cloud that separates the Cisco switches is treated as a single broadcast segment between all switches connected to the non-Cisco 802.1Q cloud through 802.1Q trunks.
- Ensure that the native VLAN is the same on all of the 802.1Q trunks that connect the Cisco switches to the non-Cisco 802.1Q cloud.

This section contains the following topics:

Allocating an Ethernet Port or Port-Channel Interface to a VLAN Trunk

You can selectively allocate individual VLANs associated with an Ethernet port or a port-channel interface to a VLAN trunk link. Note that all added VLANs are active on a trunk link, and, as long as the VLAN is available for use, traffic for

that VLAN is carried across the trunk link. To specify which VLANs are to be allocated to a trunk link, use the **switchport trunk allowed vlan** command in interface configuration mode.

To remove a VLAN from the trunk link, use the **no** form of the command.

Note You do not need to create a VLAN interface before you allocate a VLAN to an Ethernet port or port-channel interface. To configure a VLAN interface and access its mode to configure its attributes, use the **interface vlan** command in configuration mode for the context. See Chapter 3, Configuring VLAN Interfaces for details.

The syntax is as follows:

switchport trunk allowed vlan *vlan_list*

The *vlan_list* argument specifies the allowed VLANs that transmit this Ethernet interface in tagged format when in trunking mode. The *vlan_list* argument can be one of the following:

- Single VLAN number
- Range of VLAN numbers separated by a hyphen
- Specific VLAN numbers separated by commas

Valid entries are from 1 through 4094. Do not enter any spaces between the dash-specified ranges or the comma-separated numbers in the *vlan_list* argument.

For example, to add VLANs 101, 201, and 250 through 260 to the defined list of VLANs currently set for Ethernet port 4, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/4
host1/Admin(config-if)# switchport trunk allowed vlan 101,201,250-260
```

To remove VLANs 101 through 499 from the defined list of VLANs currently set for Ethernet port 4, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/4
host1/Admin(config-if)# no switchport trunk allowed vlan 101-499
```

Completing the VLAN Trunking Configuration

By default, when you configure VLAN trunking, the interface is in the shutdown state (administratively down) until you enable it as follows:

- To enable VLAN trunking in a Layer 2 Ethernet port or port-channel interface, use the **no shutdown** command in interface configuration mode. This action puts the interface in the Up administrative state.
- To disable VLAN trunking, use the **shutdown** command in interface configuration mode. This action puts the interface in the Down administrative state.

For example, to enable VLAN trunking for Ethernet port 4, enter:

```
host1/Admin(config)# interface gigabitEthernet 1/4
host1/Admin(config-if)# switchport trunk allowed vlan 101,201,250-260
host1/Admin(config-if)# no shutdown
```

For example, to disable VLAN trunking for an interface, enter:

```
host1/Admin(config-if)# switchport trunk allowed vlan 101,201,250-260
host1/Admin(config-if)# shutdown
```

Specifying the 802.1Q Native VLAN For a Trunk

On an 802.1Q trunk port, the ACE appliance tags all transmitted and received frames except for those frames configured as the native VLAN for the trunk. Frames on the native VLAN are always transmitted untagged and are normally received untagged.

When configuring 802.1Q trunking, you must match the native VLAN across the link. Because the native VLAN is untagged, the native VLAN must match on both sides of the trunk link for 802.1Q; otherwise, the link will not work.

To set the 802.1Q native VLAN for a trunk, use the **switchport trunk native vlan** command in interface configuration mode.You can only have one assigned native VLAN.

Note If you have QoS enabled for a physical Ethernet port (see the "Enabling Quality of Service for a Port" section) that has been designated as an FT VLAN port (see the "Designating an Ethernet Port as an FT VLAN Port" section), do not configure the FT VLAN as an 802.1Q native VLAN. In this configuration, the QoS setting

for redundancy traffic, such as heartbeat packets or TCP tracking probes, may not be handled properly by the ACE appliance and FT traffic may be dropped when there is network congestion.

You do not need to create a VLAN interface to set the 802.1Q native VLAN for a trunk. To configure a VLAN interface and access its mode to configure its attributes, use the **interface vlan** command in configuration mode for the context. See Chapter 3, Configuring VLAN Interfaces for details.

The syntax is as follows:

switchport trunk native vlan *number*

The *number* argument specifies the VLAN number that you want to configure as the 802.1Q native VLAN when operating in trunking mode. Valid values are from 1 to 4094. The default is VLAN 1.

For example, to specify VLAN 3 as the 802.1Q native VLAN for the trunk, enter:

```
host1/Admin(config)# interface port-channel 255
host1/Admin(config-if)# switchport trunk native vlan 3
```

To revert to the default of VLAN 1, enter:

```
host1/Admin(config-if)# no switchport trunk native vlan
```

Displaying Ethernet Interface Configuration, Status, and Statistics

Use the **show interface** command in Exec mode to display the following:

- Configuration information and counter statistics for an Ethernet port
- Configuration information for a port-channel virtual interface

Use the **show interface** Exec command without a keyword to see a list of all interfaces that are programmed on the ACE appliance. A report is provided for each interface that the device supports.

Note You can display information for the VLAN or Bridged Virtual Interface (BVI) interface through the **show interface** command. See Chapter 3, Configuring VLAN Interfaces for details.

The syntax for the command is as follows:

show interface {**gigabitEthernet** *slot_number/port_number* [**counters**] | **port-channel** *channel_number*}

The keywords, arguments, and options are as follows:

- **gigabitEthernet—**Specifies an Ethernet port.
- *slot_number*—Physical slot on the ACE appliance that contains the Ethernet ports. This selection is always 1, the location of the daughter card in the ACE appliance. The daughter card includes the four Layer 2 Ethernet ports to perform Layer 2 switching.
- *port_number*—Physical Ethernet port on the ACE appliance. Valid selections are 1 through 4, which specifies one of the four Ethernet ports (1, 2, 3, or 4) associated with the slot 1 (daughter card) selection.
- **counters**—(Optional) Displays a summary of interface counters for the specified Ethernet port related to the receive and transmit queues.
- **port-channel** *channel_number*—Specifies the channel number assigned to a port-channel interface. Valid values are from 1 to 255.

For example, to view the configuration status for Ethernet port 1, enter:

```
host1/Admin# show interface gigabitEthernet 1/1
GigabitEthernet Port 1/1 is UP, line protocol is UP
Hardware is ACE Appliance 1000Mb 802.3, address is 00:01:02:03:04:06
Description:Ethernet port 3 is configured for speeds of 1000 Mbps
MTU 9216 bytes
Full-duplex, 1000Mb/s
COS bits based QoS is disabled
input flow-control is off, output flow-control is off GigabitEthernet
Port 1/4 is ADMIN DOWN, line protocol is UP
 Hardware is ACE Appliance 1000Mb 802.3, address is 00.00.00.00.20.62
 MTU 0 bytes
 Auto-duplex, Auto-speed
 input flow-control is off, output flow-control is off GigabitEthernet
   0 packets input, 0 bytes, 0 dropped
   Received 0 broadcasts (0 multicasts)
   0 runts , 0 giants
```

```
   0 FCS/Align errors , 0 runt FCS, 0 giant FCS
   0 packets output, 0 bytes
   0 broadcast, 0 multicast, 0 control output packets
   0 underflow, 0 single collision, 0 multiple collision output
packets
    0 excessive collision and dropped, 0 Excessive Deferral and
dropped
```

Note You can configure flow control on each Ethernet port of a Catalyst 6500 series switch. However, the ACE appliance does not support flow control. If you connect an ACE appliance to a Catalyst 6500 series switch, the flow control functionality is disabled on the ACE appliance. The output of the **show interface gigabitEthernet** command on the ACE appliance displays the "input flow-control is off, output flow control is off" flow-control status line as shown in the example above regardless of the state of flow control on the Catalyst 6500 series switch port to which the ACE appliance is connected.

For example, to view the configuration status for port-channel interface 23, enter:

```
switch/Admin# show interface port-channel 23
PortChannel 23:
----------------------------
Description:
mode: Access    access vlan: 201
status: (ADMIN DOWN), load-balance scheme: src-dst-mac
PortChannel 23 mapped phyport:
```

Table 1-3 describes the fields in the **show interface port-channel** command output.

Table 1-3 Field Descriptions for show interface port-channel Command

Field	Description
Description	Configured description for this interface.
mode	Interface switchport type: Access or Trunk.
access vlan	Assigned VLAN to the port-channel interface.
Status	State of the interface: UP or DOWN.

Table 1-3 Field Descriptions for show interface port-channel Command

Field	Description
load-balancing scheme	Configured load-balancing method. If you do not configure a load-balancing method, this field displays src-dst-mac, the default scheme on the source or destination MAC address.
PortChannel *number* mapped phyport	Physical port mapped to the port-channel interface.

For example, to view a summary of interface counters for Ethernet port 3, enter:

```
switch/Admin# show interface gigabitEthernet 1/3 counters
```

Table 1-4 describes the fields in the **show interface gigabitEthernet** command output.

Table 1-4 Field Descriptions for show interface gigabitEthernet counters Command

Field	Description
RX RGMII Packets	Total number of packets received on the Reduced Gigabit Media Independent Interface (RGMII).
RX RGMII Control Packets	Total number of octets transmitted on the RGMII.
RX RGMII DMAC filtered Packets	Number of destination MAC address-filtered packets received on the RGMII.
RX RGMII Dropped Packets	Total number of packets dropped on the RGMII. **Note** These packets will also be counted in the RX Packets field.
RX RGMII Bad Packets	Total number of bad packets received on the RGMII. **Note** These packets will also be counted in the RX Packets field.

Table 1-4 Field Descriptions for show interface gigabitEthernet counters Command (continued)

Field	Description
RX RGMII Octets	Total number of octets received on the RGMII. This statistic makes up a 64-bit counter that describes the number of good octets received.
RX RGMII Control Octets	Total number of control octets received on the RGMII.

Table 1-4 Field Descriptions for show interface gigabitEthernet counters Command (continued)

Field	Description
RX RGMII DMAC filtered Octets	Number of destination MAC address-filtered octets received on the RGMII.
RX RGMII Dropped Octets	Total number of octets dropped on the specified Ethernet port.
RX Packets	Total number of packets received on the specified Ethernet port.
RX Octets	Total number of octets received on the specified Ethernet port. This statistic makes up a 64-bit counter that describes the number of good octets received.
RX Dropped Packets	Total number of packets dropped by the specified Ethernet port. **Note** These packets will also be counted in the RX Packets field.
RX Broadcasts	Number of broadcast packets received on the specified Ethernet port.
RX Multicasts	Number of multicast packets received on the specified Ethernet port.
RX Runts	Number of packets that are discarded because they are smaller than the minimum packet size allowed by the ACE appliance.
RX Giants	Number of packets that are discarded because they exceed the maximum packet size allowed by the ACE appliance.
RX FCS/Align Errors	Total number of frame check sum (FCS) errors or nonintegral number of octets (alignment errors).
RX Runt FCS	Total number of runt FCS errors.
RX Giant FCS	Total number of giant FCS errors.
Total Inbound Packets	Total number of inbound packets received by the ACE appliance.

Table 1-4 Field Descriptions for show interface gigabitEthernet counters Command (continued)

Field	Description
Total Inbound Octets	Total number of inbound octets received by the ACE appliance.
Total Inbound Errors	Total number of inbound packets with errors.
TX Packets	Total number of packets transmitted from the specified Ethernet port.
TX Octets	Total number of octets transmitted from the specified Ethernet port. This statistic makes up a 64-bit counter that describes the number of good octets transmitted.
TX Broadcast Packets	Number of broadcast packets transmitted from the specified Ethernet port.
TX Multicast Packets	Number of multicast packets transmitted from the specified Ethernet port.
TX Control Packets	Number of control packets transmitted from the specified Ethernet port.
TX Underflow Packets	Number of underflow packets transmitted from the specified Ethernet port.
TX Single Collision Packets	Number of times that a transmitted packet encountered a single collision.
TX Multiple Collision Packets	Number of times that a transmitted packet encountered multiple collisions.
TX Excessive Collisions and Dropped Packets	Number of times that a transmitted packet encountered excessive collisions, which resulted in dropped packets.
TX Excessive Deferral and Dropped Packets	Number of times that a transmitted packet encountered excessive deferrals, which resulted in dropped packets.
TX Packets with Size 0-63 Octets	Number of packets transmitted that are from 0 to 63 octets.
TX Packets with Size 64 Octets	Number of packets transmitted that are 64 octets.

Table 1-4 Field Descriptions for show interface gigabitEthernet counters Command (continued)

Field	Description
TX Packets with Size 65-127 Octets	Number of packets transmitted that are from 65 to 127 octets.
TX Packets with Size 128-255 Octets	Number of packets transmitted that are from 128 to 255 octets.
TX Packets with Size 256-511 Octets	Number of packets transmitted that are from 256 to 511 octets.
TX Packets with Size 512-1023 Octets	Number of packets transmitted that are from 512 to 1023 octets.
TX Packets with Size 1024-1518 Octets	Number of packets transmitted that are from 1024 to 1518 octets.
TX Packets with Size > 1518 Octets	Number of packets transmitted that are greater than 1518 octets.

Clearing Ethernet Interface Configuration Information

You can clear the Ethernet port configuration information displayed through the **show interface** command, by using the **clear interface gigabitEthernet** command in Exec mode. The syntax for this command is as follows:

clear interface gigabitEthernet *slot_number/port_number*

The options and arguments are as follows:

- *slot_number*—Physical slot on the ACE appliance that contains the Ethernet ports. This selection is always 1, the location of the daughter card in the ACE appliance. The daughter card includes the four Layer 2 Ethernet ports that allow you to perform Layer 2 switching.
- *port_number*—Physical Ethernet port on the ACE appliance. Valid selections are from 1 through 4, which specifies one of the four Ethernet ports (1, 2, 3, or 4) associated with the slot 1 (daughter card) selection.

For example to clear the statistics for Ethernet port 3, enter:

```
host1/Admin# clear interface gigabitEthernet 1/3
```

CHAPTER 2

Overview of IPv6

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This chapter describes Internet Protocol Version 6 (IPv6), why it is needed, and how it works. It includes the following major sections:

- Introduction to IPv6
- Methods of Transitioning from IPv4 to IPv6
- IPv6 Header Format
- IPv6 Addressing
- IPv6 Protocols and Support

Introduction to IPv6

This section describes IPv6, including a brief history of the protocol, why it is needed now, and some of the advantages of using it. The section contains the following subsections:

- What is IPv6?
- Why is IPv6 Needed Now?
- Advantages of IPv6

What is IPv6?

IPv6 is the replacement Internet protocol for IPv4. It corrects some of the deficiencies of IPv4 and simplifies the way that addresses are configured and how they are handled by Internet hosts.

IPv4 has proven to be robust, easily implemented, and interoperable, and has stood the test of scaling an internetwork to a global utility the size of the Internet. However, the initial design did not anticipate the following conditions:

- Recent exponential growth of the Internet and the impending exhaustion of the IPv4 address space
- The ability of Internet backbone routers to maintain large routing tables
- Need for simpler autoconfiguration and renumbering
- Requirement for security at the IP level (IPSec)
- Need for better support for real-time delivery of data, known as quality of service (QoS)

Why is IPv6 Needed Now?

With its 32-bit address format, IPv4 can handle a maximum 4.3 billion unique IP addresses. While this number may seem very large, it is not enough to sustain and scale the rapidly rising growth of the Internet. Although improvements to IPv4, including the use of NAT, have allowed the extended use of the protocol, address exhaustion is inevitable and could happen as soon as 2012.

With its 128-bit address format, IPv6 can support 3.4×10^{38} or 340,282,366,920,938,463,463,374,607,431,768,211,456 unique IP addresses. This number of addresses is large enough to configure a unique address on every node in the Internet and still have plenty of addresses left over. It is also large enough to eliminate the need for NAT, which has its own inherent problems.

A few countries, governmental agencies, and multinational corporations have either already deployed or mandated deployment of IPv6 in their networks and software products. Some emerging nations have no choice but to deploy IPv6 because of the unavailability of new IPv4 addresses.

Advantages of IPv6

Besides providing an almost limitless number of unique IP addresses for global end-to-end reachability and scalability, IPv6 has the following additional advantages:

- Simplified header format for efficient packet handling
- Larger payload for increased throughput and transport efficiency
- Hierarchical network architecture for routing efficiency
- Support for widely deployed routing protocols (OSPF, BGP, etc.)
- Autoconfiguration and plug-and-play support
- Elimination of need for network address translation (NAT) and application layered gateway (ALG)
- Increased number of multicast addresses

Methods of Transitioning from IPv4 to IPv6

The transition from IPv4 to IPv6 will not happen quickly because of the scope of the change. The two protocols will likely need to coexist for many years before IPv6 replaces IPv4 completely. Many countries and corporations are currently using one or more of the methods described below to transition their networks to IPv6.

Dual Stack

A dual stack means that IPv4 and IPv6 addresses coexist on the same platform and support hosts of both types. This method is a way to transition from IPv4 to IPv6 with coexistence as a first step. The ACE supports a dual stack arrangement for IPv6.

VPN Tunneling

The ACE does not support tunneling for IPv6.

NAT

The ACE acts as a proxy device by terminating connections from clients and then establishing a back-end connection with servers. It then splices the two connections together to allow the clients and servers to communicate with each other.

For IPv6, the ACE supports the NATing of client or VIP IPv4 addresses to server IPv6 and the reverse for HTTP and HTTPS load balancing.

IPv6 Header Format

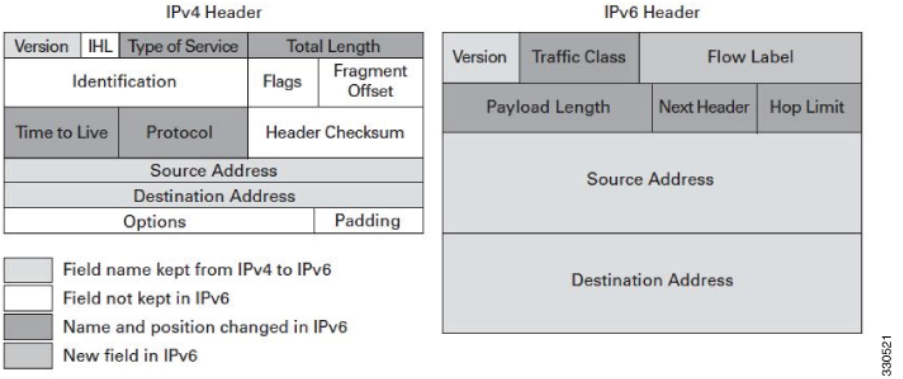
This section describes the IPv6 header format and how it differs from the IPv4 header format. It contains the following sections:

- IPv6 Header Format
- IPv6 Header Fields

IPv6 Header Format

A side-by-side comparison of the IPv4 header and the IPv6 header (Figure 2-1) shows that the IPv6 header is more streamlined and efficient than the IPv4 header.

Figure 2-1 IPv6 Header Format



IPv6 Header Fields

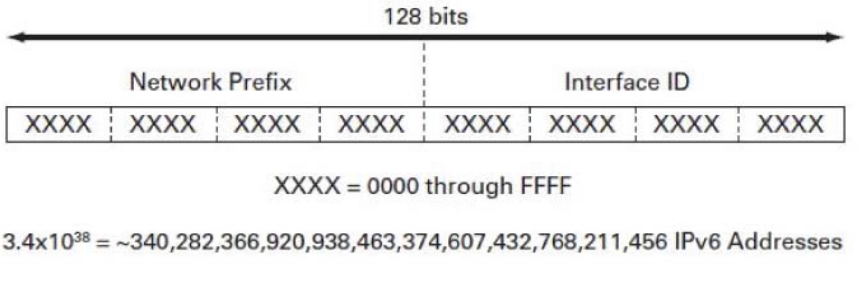
The IPv6 header contains the following fields:

- Version
- Traffic Class
- Flow Label
- Payload Length
- Next Header
- Hop Limit
- Source Address
- Destination Address

IPv6 Addressing

IPv6 addresses are 128 bits long. They are logically divided into a network prefix and a host identifier. The number of bits in the network prefix is represented by a prefix length (for example, /64). The remaining bits are used for the host identifier. If you do not specify a prefix length for an IPv6 address, the default prefix length is /64. See Figure 2-2.

Figure 2-2 IPv6 Address Format



Each IPv6 address type has a scope that describes the part of the network where the address is unique. Some IPv6 addresses are unique only in a subnet or a local network (link-local scope), others are unique in private networks or between organizations (unique-local scope), while still others are globally unique (global scope), that is, everywhere in the Internet.

Note that there is no concept of broadcast addresses in IPv6. For one to many addressing, use multicast addresses.

The ACE supports the compressed IPv6 address format where leading zeros in a 16-bit block are not shown and the longest string of 16-bit address blocks is compressed. All IPv6 addresses will display as compressed, but the CLI accepts both compressed and uncompressed addresses. For example, the following two addresses are equivalent:

```
2001:0000:ABCD:EF22:0000:1234:5678:0001
2001::ABCD:EF22:0:1234:5678:1
```

The double colon (::) in the second address indicates that a string of zeros has been omitted. You can use this compressed format only once in an address. If there are multiple contiguous strings of zeros in an address, you can replace them all with a double colon (::). Leading zeros can also be omitted, but not trailing zeros.

IPv6 supports the following types of addresses:

- Unicast Addresses
- Multicast Addresses

Unicast Addresses

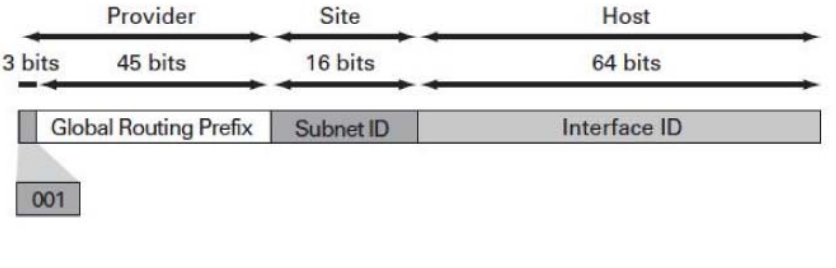
Use unicast for one-to-one communication between hosts. Unicast addresses work similarly for both IPv4 and IPv6. IPv6 supports several types of unicast addresses as described in the following sections.

- Global
- Link-Local
- Unique-Local
- Anycast Addresses

Global

A global IPv6 address is a unicast address with a predefined prefix of 2000::/3 (001). Cisco supports global IPv6 addresses in the range of 2000::/3 through 3000::/3. IPv6 addresses with a prefix of 2000::/3 (001) through E000::/3 (111), excluding the FF00::/8 (1111 1111) multicast addresses, are required to have 64-bit interface identifiers (VLAN IDs) in the IEEE 64-bit Extended Universal Identifier (EUI-64) format. The Internet Assigned Numbers Authority (IANA) allocates the IPv6 address space in the range of 2001::/16 to the registries. See Figure 2-3.

Figure 2-3 Global IPv6 Address Format



A global unicast address typically consists of a 48-bit global routing prefix and a 16-bit subnet ID. In the IPv6 aggregatable global unicast address format document (RFC 2374), the global routing prefix included two other hierarchically structured fields called Top-Level Aggregator and Next-Level Aggregator. Because these fields were policy-based, the IETF decided to remove the fields from the RFCs. However, some existing IPv6 networks deployed in the early days might still be using networks based on the older architecture.

A 16-bit subnet field called the Subnet ID can be used by individual organizations to create their own local addressing hierarchy and to identify subnets. This field allows an organization to use up to 65,535 individual subnets.

The IEEE 64-bit Extended Unique Identifier (EUI-64) is a global aggregatable address format that is used for generic IPv6 communication. All global addresses are required to include a 64-bit interface ID (VLAN ID) in the lowest 64 bits of the address. The lowest 64 bits of a global address can be assigned in one of several ways:

- Autoconfigured from EUI-64
- Expanded from a 48-bit MAC address (for example, an Ethernet address)
- Autogenerated pseudo-random number to address privacy concerns
- Assigned using DHCPv6
- Manually configured

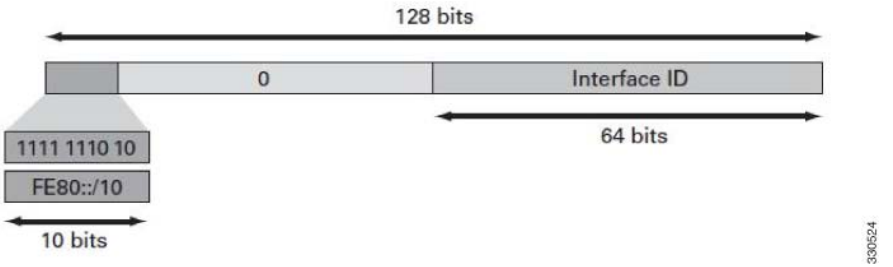
The EUI-64 format is used to perform stateless autoconfiguration. This format expands the 48-bit MAC address to 64 bits by inserting FFFE into the middle 16 bits between the upper three bytes (OUI field) and the lower 3 bytes (serial

number) of the link layer address. To better support the compression of addresses with a local scope, the seventh bit in the high-order byte of the MAC address (the universal/local "u" bit) is inverted.

Link-Local

A link-local address is a unicast address that has a scope of the local link only and one is required on every interface for IPv6 to work. The ACE automatically creates a link-local address for each IPv6-enabled interface using the EUI-64 format. Alternatively, the ACE accepts a user-configured link-local address. Each link-local address has a predefined prefix of FE80::/64. See Figure 2-4.

Figure 2-4 Link-Local Address Format



Link-local addresses have the following characteristics:

- Automatically assigned when you enable IPv6 using the **ipv6 enable** command
- Used for next-hop calculations in routing protocols
- The first ten bits of the prefix are always 1111 1110 10 (FE80::/64)
- The last 54 bits of the prefix can be zero or any configured value

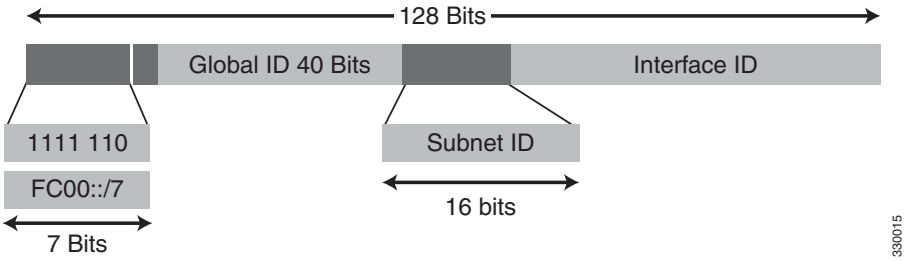
Unique-Local

A unique-local address is a unicast address that is valid only within a site or organization or between a limited number of sites. It is used for local communications and intersite VPNs. Unique-local addresses are similar to IPv4 private addresses and are not routable on the internet.

The first seven bits of the prefix are predefined as 1111 110 (FC00::/7). FC00::/7 is divided into two /8 blocks: FC00::/8 (eighth most significant bit set to 0) and FD00::/8 (eighth MSB set to 1). FC00::/8 is not defined yet. FD00::/8 is used with /48 prefixes by setting the 40 least significant bits (LSBs) to a randomly generated bit string. See Figure 2-5.

The ACE supports the EUI-64 format for both global and unique-local addresses. This format expands the 48-bit MAC address to 64 bits by inserting FFFE into the middle 16 bits between the upper three bytes (OUI field) and the lower 3 bytes (serial number) of the link layer address. To better support the compression of addresses with a local scope, the seventh bit in the high-order byte of the MAC address (the universal/local "u" bit) is inverted.

Figure 2-5 Unique-Local Address Format



Anycast Addresses

The ACE does not support anycast addressing for IPv6. Therefore, you cannot configure a single unicast address on multiple interfaces. If you attempt to do this, you will receive an error message.

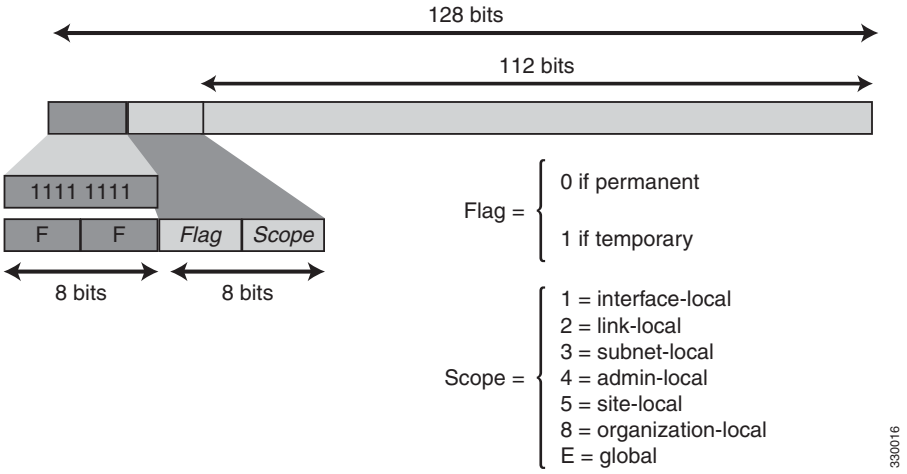
Multicast Addresses

An IPv6 multicast address has a predefined prefix of FF00::/8 (1111 1111). See Figure 2-6. The multicast address range uses 1/256 of the total IPv6 address space. An IPv6 multicast address is an identifier for a set of interfaces that typically belong to different nodes.

The ACE does not support the configuration of multicast addresses on interfaces. It does receive and process packets for the following addresses:

- All-nodes multicast group (FF02::1)
- Solicited node multicast group (FF02::1:ff00:0000/104)
- All-router multicast group (FF02::2)

Figure 2-6 Multicast Address Format



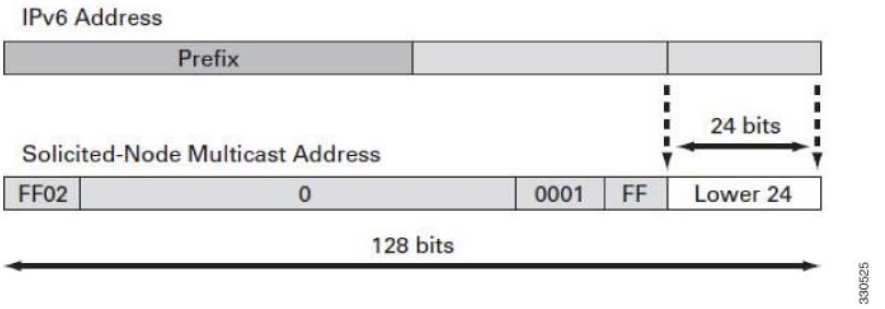
The second octet following the predefined prefix defines the lifetime and scope of the multicast address. A permanent (well-known) multicast address has a lifetime parameter equal to 0 and is assigned by IANA. A temporary multicast address has a lifetime parameter equal to 1 and is dynamically assigned. A multicast address that has the scope of an interface, link, subnet, admin, site, organization, or a global scope has a scope parameter of 1, 2, 3, 4, 5, 8, or E, respectively. The IPv6 addressing scheme is designed to support millions of multicast group addresses.

Solicited-Node Multicast Address

An IPv6 solicited-node address is used to send messages in the neighbor discovery (ND) protocol. ND is the IPv6 equivalent of ARP. For more information about ND, see Chapter 6, Configuring Neighbor Discovery.

A solicited-node multicast address is a multicast group address that corresponds to an IPv6 unicast address. An IPv6 node must join the associated solicited-node multicast group for every unicast address it has been assigned. A solicited-node multicast address has the prefix FF02:0:0:0:0:1:FF00:0000/104 concatenated with the 24 low-order bits of a corresponding IPv6 unicast or anycast address, as shown in Figure 2-7.

Figure 2-7 Solicited Node Multicast Address Format



For example, the solicited-node multicast address corresponding to the IPv6 global unicast address 2001::01:800:200E:8C6C is FF02::1:FF0E:8C6C.

IPv6 Protocols and Support

The ACE supports the following IPv6 protocols that are described in the following sections:

- Neighbor Discovery
- Router Discovery
- Duplicate Address Detection
- ICMPv6
- DHCPv6

Neighbor Discovery

Neighbor discovery (ND) is the protocol that the ACE uses to find other nodes on the same subnet. ND is always enabled when the interface is IPv6 enabled. A management policy on the interface is not necessary for ND to function on the interface. It will also not be possible to disable ND on the interface by configuring any kind of policy.

The ACE sends neighbor solicitation (NS) messages to resolve addresses, for neighbor unreachability detection (NUD), and duplicate address detection (DAD).

The ACE sends out both solicited and unsolicited neighbor advertisements (NAs). It responds to an NS when the target address of the NS is either configured on an enabled IPv6 interface or is configured as a VIP or NAT address on the interface.

The ACE supports statically defined neighbor link-local to Layer 2 address mappings on an interface. These mappings are not timed out or overwritten by the ND process and they can be removed only through configuration. If the address already exists in the cache at the time of configuration, the cached entry is overwritten by the configured one and becomes static.

Router Discovery

Router discovery (RD) is the process that the ACE uses to advertise its presence on the local subnet. The ACE sends out both solicited and unsolicited router advertisements (RAs). Although the ACE does not send out router solicitation (RS) messages, it does to respond to RS messages from its neighbors.

Duplicate Address Detection

duplicate address detection (DAD) is an IPv6 mechanism that detects duplicate addresses in a subnet. If a duplicate address is found on an interface, the duplicate address is disabled on the originating interface until the problem is resolved. If the link-local address is a duplicate, then all addresses on the IPv6 interface are disabled until the address problem is resolved.

ICMPv6

The ACE supports ICMPv6 management policies and ICMPv6 ACLs.

The ACE does not send out ICMPv6 redirects because the ACE does not support dynamic routing protocols and should not advise hosts about what is the best route.

DHCPv6

DHCP relay is an agent that resides between clients and DHCP servers, and forwards client requests to servers and server replies back to clients. For IPv6, DHCP is the stateful address autoconfiguration protocol, from which the clients can receive addressing information from DHCP servers. The ACE supports DHCPv6 relay only.

Clients can also query the DHCP servers for other configuration information (for example, DNS and NTP (appliance only) servers).

To identify the client interface to which a server reply must be forwarded, the ACE uses the interface ID (VLAN ID) DHCP option. The ACE will add this option to the Relay-Forward messages that it generates from the client request. The option value is the interface ID (VLAN ID) of the interface on which the client request is received. The server replies in a Relay-Reply message with this option value intact. The ACE knows which interface to use to send the reply on by reading back this option from the Relay-Reply message.

The DHCPv6 relay process configures ACLs so that it receives all the relevant DHCPv6 packets, and then processes the packets and sends them out according to the user configuration.

CHAPTER 3

Configuring VLAN Interfaces

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This chapter describes how to configure the VLAN interfaces on the ACE. The ACE appliance has external physical interfaces that you configure with VLANs to receive traffic from clients and servers.

(ACE module only) The ACE module does not have any external physical interfaces. Instead, it uses internal VLAN interfaces that you assign from the supervisor engine to the ACE module. After the VLANs are assigned to the ACE module, you can configure the corresponding VLAN interfaces on the module as either routed or bridged.

When you configure an IPv6 or an IPv4 address on an interface, the ACE automatically makes it a routed mode interface.

Similarly, when you configure a bridge group on an interface VLAN, the ACE automatically makes it a bridged interface. Then, you can associate a bridge-group virtual interface (BVI) with the bridge group. For more information on bridged groups and BVIs, see Chapter 5, “Bridging Traffic.”

The ACE also supports shared VLANs, which are multiple interfaces in different contexts on the same VLAN within the same subnet. Only routed interfaces can share VLANs. Note that there is no routing across contexts even when shared VLANs are configured.

Each ACE supports a maximum of 4093 VLANs and a maximum of 1024 shared VLANs.

Note Each ACE supports a maximum of 8192 interfaces that includes VLANs, shared VLANs, and BVI interfaces.

This chapter contains the following major sections:

ACE VLAN Interface Configuration Quick Start

Table 3-1 provides a quick overview of the steps required to configure VLAN interfaces on an ACE. Each step includes the CLI command or a reference to the procedure required to complete the task. For a complete description of each feature and all the options associated with the CLI commands, see the sections following Table 3-1.

Table 3-1 VLAN Interface Configuration Quick Start

Task and Command Example

1. If you are operating in multiple contexts, observe the CLI prompt to verify that you are operating in the desired context. If necessary, change to the correct context.
`host1/Admin# changeto C1`
`host1/C1#`
The rest of the examples in this table use the C1 user context for illustration purposes, unless otherwise specified. For details on creating contexts, see the *Virtualization Guide, Cisco ACE Application Control Engine*.

2. Enter global configuration mode.
`host1/Admin# config`
`host1/Admin(config)#`

3. (ACE appliance only, Optional) If you have not already done so, configure Ethernet ports and specify VLAN trunking on the ACE appliance (see Chapter 1, "Configuring ACE Appliance Ethernet Interfaces").

4. (ACE module only) Create VLAN groups on the Catalyst 6500 series switch or Cisco 7600 series router supervisor engine and then assign the groups to the ACE module.
`Router(config)# svclc vlan-group 50 55-57`
`Router(config)# svc module 5 vlan-group 50`

5. Configure a VLAN interface and access its mode to configure its attributes. For example, to create VLAN 200, enter the following command:
`host1/Admin(config)# interface vlan 200`
`host1/Admin(config-if)#`

6. Enable the interface to process IPv6 traffic. `host1/Admin(config-if)# ipv6 enable`

7. Assign an IPv6 or an IPv4 address to a VLAN interface for routing traffic. For example, enter the following command:
`host1/Admin(config-if)# ip address 2001:DB8:1::CAFE/64`
`host1/Admin(config-if)# ip address 192.168.1.1 255.255.255.0`

8. Enable the VLAN interface. `host1/Admin(config-if)# no shutdown`

Table 3-1 VLAN Interface Configuration Quick Start (continued)

Task and Command Example

9. (Optional) Specify the MTU for a VLAN interface.
`host1/Admin(config-if)# ipv6 mtu 1280`
or
`host1/Admin(config-if)# mtu 1000`

10. (Optional) In a redundant configuration, configure the IPv6 or IPv4 address for an interface on a standby ACE.
`host1/Admin(config-if)# peer ip address 2001:DB8:2::CAFE/64`
or
`host1/Admin(config-if)# peer ip address 192.168.1.20 255.255.255.0`

11. (Optional) Enable reverse-path forwarding (RPF) based on a source MAC address for a VLAN interface. `host1/Admin(config-if)# mac-sticky enable`
12. (Optional) Add a description about the interface to help you remember its function. `host1/Admin(config-if)# description FOR INBOUND AND OUTBOUND TRAFFIC`

13. Assign a policy map to an interface. For example, to assign the SLB_OPTIMIZE_POLICY policy map for inbound traffic to the VLAN 3, enter the following command:
`host1/Admin(config)# interface vlan 200`
`host1/Admin(config-if)# service-policy input SLB_OPTIMIZE_POLICY`

14. Apply an ACL to the inbound or outbound direction of an interface and make the ACL active. For example, enter the following command:
`host1/Admin(config-if)# access-group input INBOUNDv6`
`host1/Admin(config-if)# access-group input INBOUNDv4`
`host1/Admin(config-if)# exit`

15. Assign VLAN interfaces to a specific context. For example, to assign VLAN 200 to context C1, enter the following command:
`host1/Admin(config)# context C1`
`host1/C1(config-context)# allocate-interface vlan 200`

Table 3-1 VLAN Interface Configuration Quick Start (continued)

Task and Command Example
16. (Optional) Configure a specific bank of MAC addresses for an ACE. For example, to configure bank 2 of MAC addresses, enter the following command: `host1/Admin(config)# shared-vlan-hostid 2`
17. (Optional) If necessary, save your configuration changes to flash memory. `host1/Admin# copy running-config startup-config`

Configuring ACE Module VLAN Groups

To allow the ACE module to receive traffic from the supervisor engine in the Catalyst 6500 series switch or Cisco 7600 series router, you must create VLAN groups on the supervisor engine and then assign the groups to the ACE module. After the VLAN groups are assigned to the ACE module, you can configure the VLAN interfaces on the ACE module. By default, all VLANs are allocated to the Admin context on the ACE module.

This section contains the following topics:

- Creating VLAN Groups Using Cisco IOS Software
- Assigning VLAN Groups to the ACE Module through Cisco IOS Software
- Assigning a Switched Virtual Interface VLAN to the ACE Module

Creating VLAN Groups Using Cisco IOS Software

In Cisco IOS software, you can create one or more VLAN groups and then assign the groups to the ACE module. For example, you can assign all the VLANs to one group, create an inside group and an outside group, or create a group for each customer.

You cannot assign the same VLAN to multiple groups; however, you can assign up to a maximum of 16 groups to an ACE module. VLANs that you want to assign to multiple ACE modules, for example, can reside in a separate group from VLANs that are unique to each ACE module.

To assign VLANs to a group using Cisco IOS software on the supervisor engine, use the **svclc vlan-group** command. The syntax of this command is as follows:

svclc vlan-group *group_number vlan_range*

The arguments are as follows:

- *group_number*—Number of the VLAN group.
- *vlan_range*—One or more VLANs. The valid VLAN ranges are 2 to 1000 and 1025 to 4094 (VLANs 1 and 1001 to 1024 are reserved and cannot be used).

  VLANs are specified in one of the following ways:

  - A single number (*n*)
  - A range (*n-x*)

  Separate numbers or ranges by commas, as shown in this example:

  ```
  5,7-10,13,45-100
  ```

For example, to create three VLAN groups, 50 with a VLAN range of 55 to 57, 51 with a VLAN range of 75 to 86, and 52 with VLAN 100, enter:

```
Router(config)# svclc vlan-group 50 55-57
Router(config)# svclc vlan-group 51 70-85
Router(config)# svclc vlan-group 52 100
```

Assigning VLAN Groups to the ACE Module through Cisco IOS Software

The ACE module cannot receive traffic from the supervisor engine unless you assign VLAN groups to it. To assign the VLAN groups to the ACE module using Cisco IOS software on the supervisor engine, use the **svc module** command in configuration mode. The syntax of this command is as follows:

svc module *slot_number* **vlan-group** *group_number_range*

The arguments are as follows:

- *slot_number*—Slot number where the ACE module resides. To display slot numbers and the ACEs in the chassis, use the **show module** command in Exec mode. The ACE module appears as the Application Control Engine Module in the Card Type field.

- *group_number_range*—One or more group numbers that are identified in one of the following ways:
  - A single number (*n*)
  - A range (*n-x*)

  Separate numbers or ranges by commas, as shown in this example:

  ```
  5,7-10
  ```

For example, to assign VLAN groups 50 and 52 to the ACE module in slot 5, and VLAN groups 51 and 52 to the ACE in slot 8, enter:

```
Router(config)# svc module 5 vlan-group 50,52
Router(config)# svc module 8 vlan-group 51,52
```

To view the group configuration for the ACE module and the associated VLANs, use the **show svclc vlan-group** command. For example, enter:

```
Router(config)# exit
Router# show svclc vlan-group
```

To view VLAN group numbers for all ACEs, use the **show svc module** command. For example, enter:

```
Router# show svc module
```

Note Enter the **show vlans** command in Exec mode from the Admin context to display the ACE module VLANs that are downloaded from the supervisor engine.

Assigning a Switched Virtual Interface VLAN to the ACE Module

A VLAN defined on the Multilayer Switch Feature Card (MSFC) is called a switched virtual interface (SVI). If you assign the VLAN used for the SVI to the ACE module, then the MSFC routes between the ACE module and other Layer 3 VLANs. By default, only one SVI can exist between the MSFC and the ACE module. However, for multiple contexts, you may must configure multiple SVIs for unique VLANs on each context.

To add an SVI to the MSFC and configure it with a VLAN assigned to the ACE module, perform the following steps:

Step 1 (Optional) If you need to add more than one SVI to the ACE module, enter the following command:

```
Router(config)# svclc multiple-vlan-interfaces
```

Step 2 Add a VLAN interface to the MSFC. For example, to add VLAN 55, enter the following command:

```
Router(config)# interface vlan 55
```

Step 3 Set the IP address for this interface on the MSFC. For example, to set the address 10.1.1.1 255.255.255.0, enter the following command:

```
Router(config-if)# ip address 10.1.1.1 255.255.255.0
```

Step 4 Enable the interface. For example, enter the following command:

```
Router(config-if)# no shut
```

Note To monitor any VLAN that is associated with more than two trunk ports, physical ports, or trunk-physical ports on the supervisor engine, enable the autostate feature by using the **svclc autostate** command. When you associate at VLAN to these ports, autostate declares that the VLAN is up. When a VLAN state change occurs on the supervisor engine, autostate sends a notification to the ACE module to bring the interface up or down.

To view this SVI configuration, use the **show interface vlan** command. For example, enter:

```
Router# show int vlan 55
```

Allocating VLANs to a User Context

By default, all VLANs assigned to the ACE are available at the Admin context. At the Admin context, you can assign a VLAN to a user context. VLANs can be shared across multiple contexts.

Guidelines and Restrictions

- The ACE supports only 1024 shared VLANs per system.
- When a VLAN is shared in multiple contexts, the IP addresses across contexts must be unique and the interfaces must be on the same subnet. To classify traffic on multiple contexts, the same VLAN across contexts will have different MAC addresses. If you configure shared VLANs, no routing can occur across the contexts.
- (ACE module only) You can view the VLANs assigned from the supervisor engine to the ACE by using the **show vlans** command in Exec mode from the Admin context.
- (ACE module only) You can assign a VLAN number to a context even if the VLAN has not been assigned from the supervisor engine to the ACE module. You can configure the VLAN in the context, however the VLAN cannot receive traffic until it is assigned from the supervisor engine to the ACE module.
- You can configure one or more VLAN interfaces in any user context before you assign those VLAN interfaces to the associated user contexts through the **allocate-interface vlan** command in the Admin context.

Procedure

To assign VLAN interfaces to the context, access the context mode and use the **allocate-interface vlan** command in configuration mode. The syntax of this command is as follows:

allocate-interface vlan *vlan_number*

The *vlan_number* argument is the number of a VLAN or a range of VLANs assigned to the ACE.

For example, to assign VLAN 10 to context A, enter:

```
host1/Admin(config)# context A
host1/Admin(config-context)# allocate-interface vlan 10
```

To allocate an inclusive range of VLANs from VLAN 100 through VLAN 200 to a context, enter:

```
host1/Admin(config-context)# allocate-interface vlan 100-200
```

To remove a VLAN from a user context, use the **no allocate-interface vlan** command in context configuration mode. For example, enter:

```
host1/Admin(config)# context A
host1/Admin(config-context)# no allocate-interface vlan 10
```

Note You cannot deallocate a VLAN from a user context if the VLAN is currently in use on that context.

To remove a range of VLANs from a context, enter:

```
host1/Admin(config-context)# no allocate-interface vlan 100-200
```

Configuring a Bank of MAC Addresses for Shared VLANs

When contexts share a VLAN, the ACE assigns a different MAC address to the VLAN on each context. The MAC addresses reserved for shared VLANs are 0x001243dc6b00 to 0x001243dcaaff, inclusive. All ACEs derive these addresses from a global pool of 16,000 MAC addresses. This pool is divided into 16 banks, each containing 1024 addresses. Each subnet can have 16 ACEs.

Each ACE supports 1024 shared VLANs, and uses only one bank of MAC addresses out of the pool. A shared MAC address is associated with a shared VLAN interface.

By default, the bank of MAC addresses that the ACE uses is randomly selected at boot time. However, if you configure two ACEs in the same Layer 2 network and they are using shared VLANs, the ACEs may select the same address bank, which results in the use of the same MAC addresses. To avoid this conflict, you must configure the bank that the ACEs will use.

To configure a specific bank of MAC addresses for a local ACE or a peer ACE (in a redundant configuration), use the **shared-vlan-hostid** or the **peer shared-vlan-hostid** command, respectively, in configuration mode in the Admin context. The syntaxes of these commands are as follows:

shared-vlan-hostid *number*

peer shared-vlan-hostid *number*

The *number* argument indicates the bank of MAC addresses that the ACE uses. Enter a number from 1 to 16. Be sure to configure different bank numbers for multiple ACEs. For example, to configure bank 2 of MAC addresses for the local ACE and bank 3 for a peer ACE, enter:

```
host1/Admin(config)# shared-vlan-hostid 2
host1/Admin(config)# peer shared-vlan-hostid 3
```

To remove the configured bank of MAC addresses and allow the ACE to randomly select a bank, use the **no shared-vlan-hostid** command. For example, enter:

```
host1/Admin(config)# no shared-vlan-hostid
```

To remove the configured bank of MAC addresses from a peer ACE and allow it to randomly select a bank, use the **no peer shared-vlan-hostid** command. For example, enter:

```
host1/Admin(config)# no peer shared-vlan-hostid
```

Disabling the ACE Module Egress MAC Lookup

Normally, the ACE module performs a MAC address lookup when it receives a packet from the backplane and again when it forwards a packet out the egress interface. If you have multiple ACE modules installed in a Catalyst 6500 series switch or Cisco 7600 series router, you may experience lower performance than expected with very high rates of traffic. If you fail to achieve the advertised performance of the ACE module, you can disable the egress MAC address lookup using the **hw-module optimize-lookup** command in configuration mode. The syntax of this command is as follows:

hw-module optimize-lookup

Note Do not use this command if you have intelligent modules with distributed forwarding cards (DFCs) installed in the Catalyst 6500 series switch or Cisco 7600 series router. Using this command with such modules will cause the Encoded Address Recognition Logic (EARL) units on these modules and on the Supervisor to become unsynchronized.

For example, to disable all egress MAC address lookups in the ACE module, enter the following command:

```
Admin/host1(config)# hw-module optimize-lookup
```

To reenable egress MAC lookups, enter the following command:

```
Admin/host1(config)# no hw-module optimize-lookup
```

Configuring VLAN Interfaces on the ACE

You can configure a VLAN interface and access its mode to configure its attributes by using the **interface vlan** command in configuration mode for the context. The syntax of this command is as follows:

interface vlan *number*

The *number* argument is the VLAN number you want to assign to the interface. VLAN numbers are 2 to 4094 (VLAN 1 is reserved for internal use and cannot be used).

For example, to create VLAN 200, enter:

```
host1/Admin(config)# interface vlan 200
```

To remove a VLAN, use the **no interface vlan** command. For example, enter:

```
host1/Admin(config)# no interface vlan 200
```

Note For security reasons, the ACE does not allow pings from an interface on a VLAN on one side of the ACE through the ACE to an interface on a different VLAN on the other side of the ACE. For example, a host can ping the ACE address that is on the IP subnet using the same VLAN as the host, but cannot ping IP addresses configured on other VLANs on the ACE.

This section contains the following topics:

- Enabling IPv6 on an Interface
- Assigning IPv6 Addresses to an Interface for Routing Traffic
- Assigning IPv4 Addresses to Interfaces for Routing Traffic
- Disabling and Enabling Traffic on Interfaces
- Configuring the IPv4 MTU for an Interface
- Autogenerating a MAC Address for a VLAN Interface
- Enabling the Mac-Sticky Feature
- Providing an Interface Description
- Configuring the UDP Booster Feature
- Removing the ACE Ethernet IP Packet Trailing Byte
- Assigning a Policy Map to an Interface
- Applying an Access List to an Interface

Note The ACE requires a route back to the client before it can forward a request to a server. If the route back is not present, the ACE cannot establish a flow and drops the client request. Make sure that you configure the appropriate routing to the client network on the ACE VLAN where the client traffic enters the ACE.

Additional configurations and commands are available on a VLAN interface that are not documented in this chapter. These configurations are as follows:

- (ACE appliance only) Allocate individual VLANs to a trunk link—See Allocating an Ethernet Port or Port-Channel Interface to a VLAN Trunk in Chapter 1, “Configuring ACE Appliance Ethernet Interfaces.”

- (ACE appliance only) IEEE 802.1Q Native VLAN for a trunk—See "Specifying the 802.1Q Native VLAN For a Trunk" in Chapter 1, "Configuring ACE Appliance Ethernet Interfaces."
- (ACE appliance only) Access port to a specific VLAN—See "Configuring a VLAN Access Port" in Chapter 1, "Configuring ACE Appliance Ethernet Interfaces."
- Remote network management—See the *Administration Guide, Cisco ACE Application Control Engine*.
- Default and static routes—See Chapter 4, "Configuring Routes on the ACE."
- Bridge parameters including the **interface bvi** command—See Chapter 5, "Bridging Traffic."
- Address Resolution Protocol (ARP)—See Chapter 7, "Configuring ARP."
- Dynamic Host Configuration Protocol (DHCP)—See Chapter 8, "Configuring the DHCP Relay."
- Policy and class maps, and SNMP management for VLANs, and fault-tolerant VLANs—See the *Administration Guide, Cisco ACE Application Control Engine*.
- Load balancing traffic including stealth firewall load balancing—See the *Server Load-Balancing Guide, Cisco ACE Application Control Engine*.
- ACLs, Network Address Translation (NAT), IP fragment reassembly, and IP normalization—See the *Security Guide, Cisco ACE Application Control Engine*.

Enabling IPv6 on an Interface

To enable IPv6 on an interface, use the **ipv6 enable** command in interface configuration mode. By default, IPv6 is disabled on an interface. Note that the interface cannot be in bridged mode. The interface may or may not have IPv4 addresses configured on it. The syntax of this command is as follows;

ipv6 enable

When you enter this command, the ACE automatically creates a link-local address for the interface and performs duplicate address detection (DAD).

Note The active ACE in a FT pair performs DAD for the alias address. However, the ACE does not enable the interface for IPv6 processing until you configure the **ipv6 enable** command.

To disable IPv6 functionality on an interface, enter the following command:

```
host1/Admin(config-if)# no ipv6 enable
```

Assigning IPv6 Addresses to an Interface for Routing Traffic

The ACE supports several types of IPv6 addresses on an interface as follows:

- Link local
- Unique local
- Global
- Alias

For details about each address type, see Chapter 2, Overview of IPv6.

The following subsections describe how to configure the various types of IPv6 addresses:

- Configuring an IPv6 Link-Local Address
- Configuring an IPv6 Unique-Local Address
- Configuring an IPv6 Global Address
- Assigning IPv4 Addresses to Interfaces for Routing Traffic

Configuring an IPv6 Link-Local Address

A link-local address is an IPv6 unicast address that has a scope of the local link only and is required on every interface. You can configure a link-local address manually or you can instruct the ACE to generate one automatically. Every link-local address has a predefined prefix of FE80::/64. You can configure only one IPv6 link-local address on an interface. Any additional IPv6 link-local address that you configure will overwrite the existing one. For more information about IPv6 link-local addresses, see Chapter 2, Overview of IPv6.

Procedures

To automatically configure a link-local address on an interface, use the **ipv6 enable** command or configure a global IPv6 address on an interface. For information about the **ipv6 enable** command, see the "Enabling IPv6 on an Interface" section.

To manually configure a link-local address on an interface, use the **ip address** command in interface configuration mode. The syntax of this command is as follows:

ip address *ipv6_address/prefix_length* **link-local**

The keywords and arguments are as follows:

- *ipv6_address*—Complete IPv6 address with an FE80::/64 prefix
- */prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128.
- **link-local**—Specifies that the address is valid only for the current link

For example, to configure a link-local address on VLAN 100, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# ip address FE80:DB8:1::/64 link-local
```

To remove a link-local address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no ip address FE80:DB8:1::/64 link-local
```

Configuring an IPv6 Peer Link-Local Address

In a redundant configuration, you can configure an IPv6 peer link-local address for the standby ACE. You can configure only one peer link-local address on an interface. Any additional peer link-local address that you configure will overwrite the existing one.

Procedure

To configure a peer link-local address, use the **peer ip address** command in interface configuration mode. The syntax of this command is as follows:

peer ip address *ipv6_address*/*prefix_length* **link-local**

The keywords and arguments are as follows:

- *ipv6_address*—Complete IPv6 address with an FE80::/64 prefix
- */prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128.
- **link-local**—Specifies that the address is valid only for the current link.

Note The IPv6 peer link-local address must be unique across multiple contexts on a shared VLAN.

Caution Do not configure under a real server a peer IPv6 address that is calculated from EUI64. In a redundant configuration, if you configure a peer IPv6 address as EUI64 on an interface, the address will not be learned by the active member of an FT group because the address is calculated only on the peer. If you then configure the same calculated IPv6 address on the active under a real server, the CLI accepts it because it does not calculate it. This IPv6 address is not synced to the standby because it conflicts with the interface address. If you subsequently apply a probe to the real server, the state of the real server is PROBE-FAILED on the active and OUTOFSERVICE on the standby. This same check applies to VIPs, routes, interfaces, and probes.

For example, to configure a peer link-local address on VLAN 100, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# peer ip address FE80:DB8:1::/64 link-local
```

To remove a peer link-local address from VLAN 100, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no peer ip address FE80:DB8:1::/64 link-local
```

Configuring an IPv6 Unique-Local Address

A unique-local address is an optional IPv6 unicast address that is used for local communication within an organization and it is similar to a private IPv4 address (for example, 10.10.2.1). Unique local addresses have a global scope, but they are not routable on the internet, and they are assigned by a central authority. All unique-local addresses have a predefined prefix of FC00::/7. You can configure only one IPv6 unique-local address on an interface. Any additional unique-local address that you configure will overwrite the existing one. For more information about unique-local addresses, see Chapter 2, Overview of IPv6.

Procedure

To configure a unique-local address on an interface use the **ip address** command in interface configuration mode. The syntax of this command is as follows:

ip address *ipv6_address*/*prefix_length* **unique-local** [**eui64**]

The keywords and arguments are as follows:

- *ipv6_address*—Complete IPv6 address with an FC00::/7 prefix
- /*prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. If you use the optional **eui64** keyword, the prefix must be less than or equal to /64.
- **unique-local**—Specifies that this address is globally unique and used only for local communications within a site or organization
- **eui64**—(Optional) Specifies that the low order 64 bits are automatically generated in the IEEE 64-bit Extended Unique Identifier (EUI-64) format specified in RFC 2373. To use this keyword, you must enter a prefix length, the prefix must be less than or equal to /64, and the host segment must be all zeros. For more information about EUI-64, see Chapter 2, Overview of IPv6.

For example, to configure a unique-local address on VLAN 100, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# ip address FC00:DB8:1::/64 unique-local
```

To remove a unique-local address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
```

```
host1/Admin(config-if)# no ip address FC00:DB8:1::/64 unique-local
```

Configuring an IPv6 Peer Unique-Local Address

In a redundant configuration, you can configure an IPv6 peer unique-local address on the active that is synchronized to the standby ACE. You can configure only one peer unique-local IPv6 address on an interface. Any additional peer unique-local address that you configure will overwrite the existing one.

Procedure

To configure an IPv6 peer unique-local address, use the **peer ip address** command in interface configuration mode. The syntax of this command is as follows:

peer ip address *ipv6_address*/*prefix_length* **unique-local**

The keywords and arguments are:

- *ipv6_address*—Complete IPv6 address with an FC00::/7 prefix
- *prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier (for example, /64).
- **unique-local**—Specifies that this address is globally unique and used only for local communications within a site or organization

Note The IPv6 peer unique-local address must be unique across multiple contexts on a shared VLAN.

For example, to configure a peer unique-local IPv6 address on VLAN 100, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# peer ip address FC00:DB8:1::/64 unique-local
```

To remove an IPv6 peer unique-local IPv6 address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no peer ip address FC00:DB8:1::/64
unique-local
```

Configuring an IPv6 Global Address

A global address is an optional IPv6 unicast address that is used for general IPv6 communication. Each global address is unique across the entire Internet. Therefore, its scope is global. The low order 64 bits can be assigned in several ways, including autoconfiguration using the EUI-64 format. You can configure only one global IPv6 address on an interface. Any additional global address that you configure will overwrite the existing one. For more information about global addresses, see Chapter 2, Overview of IPv6.

Procedure

To configure an IPv6 global address on an interface, use the **ip address** command in interface configuration mode. The syntax of this command is as follows:

ip address *ipv6_address/prefix_length* [**eui64**]

The keywords and arguments are as follows:

- *ipv6_address*—Complete IPv6 address with a prefix of 2001::/3 to 2C00::/3.
- */prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. If you use the optional **eui64** keyword, the prefix must be less than or equal to 64.
- **eui64**—(Optional) Specifies that the low order 64 bits are automatically generated in the IEEE 64-bit Extended Unique Identifier (EUI-64) format specified in RFC 2373. To use this keyword, you must specify a prefix length, the prefix must be less than or equal to 64, and the host segment must be all zeros. For more information about EUI-64, see Chapter 2, Overview of IPv6.

For example, to configure an IPv6 global address on VLAN 100 using the EUI-64 format, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# ip address 2001:DB8:1::/64 eui64
```

To configure a global address without using EUI-64, enter the following command:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# ip address 2001:DB8:2::/64
```

To remove an IPv6 global address from an interface, enter the following commands:

```
host1/Admin(config-if)# no ip address 2001:DB8:1::/64 eui64
```

Configuring an IPv6 Peer Global Address

In a redundant configuration, you can configure an IPv6 peer global address that is synchronized to the standby ACE.

Procedure

To configure an IPv6 peer global address, use the **peer ip address** command in interface configuration mode. The syntax of this command is as follows:

peer ip address *ipv6_address/prefix_length* [**eui64**]

The arguments and option are as follows:

- *ipv6_address*—IPv6 address of the peer ACE.
- */prefix_length*— Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. If you use the optional **eui64** keyword, the prefix must be less than or equal to 64.
- **eui64**—(Optional) Specifies that the IPv6 address is in the EUI64 format and that the interface identifier (64 least significant bits (LSBs)) are randomly generated using the MAC address.

Note The IPv6 peer global address must be unique across multiple contexts on a shared VLAN.

For example, to configure an IPv6 peer global address and prefix for the peer ACE, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# peer ip address 2001:DB8:2::/64 eui64
```

To remove an IPv6 peer global address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no peer ip address 2001:DB8:2::/64 eui64
```

Configuring an IPv6 Alias Address

When you configure redundancy with active and standby ACEs, you can configure a VLAN interface that has an IPv6 alias address that is shared between the active and standby ACEs. The IPv6 alias address serves as a shared gateway for the two ACEs in a redundant configuration. The IPv6 alias address can be a unique-local or a global unicast address. You can configure only one IPv6 alias address on an interface. Any additional IPv6 alias address that you configure will overwrite the existing one.

Note You must configure redundancy (fault tolerance) on the ACE for the IPv6 alias address to work. For more information about redundancy, see the *Administration Guide, Cisco ACE Application Control Engine*.

Procedure

To configure an IPv6 alias, use the **alias** command in interface configuration mode. The syntax of this command is as follows:

alias *ipv6_address*

The arguments are as follows:

- *ipv6_address*— IPv6 alias unique-local or global unicast address of the ACE. The network portion of the alias address must match the network portion of the unique-local or global unicast address on that interface.
- */prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128.

For example, to configure an IPv6 alias unique-local address, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# alias FC00:DB8:1::/64
```

To remove an IPv6 alias unique-local address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no alias FC00:DB8:1::/64
```

For example, to configure an IPv6 alias global address, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# alias 2001:DB8:2::/64
```

To remove an IPv6 alias global address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no alias 2001:DB8:2::/64
```

Assigning IPv4 Addresses to Interfaces for Routing Traffic

The ACE supports only one primary IP address with a maximum of 15 secondary addresses per interface. It treats the secondary addresses the same as a primary address and handles IP broadcasts and ARP requests for the subnet that is assigned to the secondary address as well as the interface routes in the IP routing table.

The ACE accepts client, server, or remote access traffic on the primary and secondary addresses. When the destination for the control plane (CP)-originated packets is Layer 2 adjacent to either the primary subnet or one of the secondary subnets, the ACE uses the appropriate primary or secondary interface IP address for the destination subnet as the source IP address. For any destination that is not Layer 2 adjacent, the ACE uses the primary address as the source IP address. For packets destined to the secondary IP address, the ACE sends the response with the secondary IP address as the source address.

Note SSL probes always use the primary IP address as the source address for all destinations.

Guidelines and Restrictions

Observe the following requirements and restrictions when you assign an IP address to an interface:

- Assigning an IP address to a VLAN interface automatically makes it a routed mode interface.

- You must configure a primary IP address for the interface to allow a VLAN to become active. The primary address must be active before a secondary address can be active.
- You can configure only one primary address per VLAN.
- You can configure a maximum of 15 secondary addresses per VLAN. The ACE has a system limit of 1024 secondary addresses.
- In a single context, each interface address must be on a unique subnet and cannot overlap.
- In different contexts on a nonshared VLAN, the IP subnet can overlap an interface. However, on a shared VLAN, the IP address must be unique.
- Routed and bridged mode requires access control lists (ACLs) to allow traffic to pass. To apply an ACL to the inbound or outbound direction of an interface and make the ACL active, use the **access-group** command in interface configuration mode for the VLAN, as described in the "Applying an Access List to an Interface" section. For more information on configuring ACLs, see the *Security Guide, Cisco ACE Application Control Engine*.

  When you configure access to an interface, the ACE applies the access to all IP addresses configured on the interface.

  When you configure remote network management access on an interface, the interface does not require an ACL. However, it does require a management class map and management policy map configuration. For information on configuring remote access to the ACE, see the *Administration Guide, Cisco ACE Application Control Engine*.
- You cannot configure secondary IP addresses on FT VLANs. When you configure a query interface to assess the health of the active FT group member, it uses the primary IP address.

Procedure

To assign an IPv4 address to a VLAN interface, use the **ip address** command in interface configuration mode. The syntax of this command is as follows:

ip address *ip_address mask* [**secondary**]

The arguments and option are as follows:

- *ip_address mask*—IP address and mask for the VLAN interface.

If you do not include the **secondary** option, this address becomes the primary IP address. An interface can have only one primary IP address. To make the VLAN active, you must configure a primary IP address for the interface.

- **secondary**—(Optional) Configures the address as a secondary IP address that allows multiple subnets under the same VLAN. You can configure a maximum of 15 secondary addresses per VLAN. The ACE has a system limit of 1024 secondary addresses.

  The primary address must be active before the secondary address can be active.

Note The ACE has no counters specifically for traffic received or sent through secondary IP addresses. All counters are at the interface level or associated with the primary IP address.

For example, to assign the IP address and mask 192.168.1.1 255.255.255.0 to VLAN interface 200, enter:

```
host1/Admin(config)# interface vlan 200
host1/Admin(config-if)# ip address 192.168.1.1 255.255.255.0
```

If you make a mistake while entering this command, you can reenter the command with the correct information.

To assign a secondary IP address and mask 11.11.1.1 255.255.255.0 to VLAN interface 200, enter:

```
host1/Admin(config-if)# ip address 11.11.1.1 255.255.255.0 secondary
```

To remove the IP address for the VLAN, use the **no ip address** command. For example, enter:

```
host1/Admin(config-if)# no ip address
```

To remove a secondary IP address for the VLAN, enter:

```
host1/Admin(config-if)# no ip address 11.11.1.1 255.255.255.0
secondary
```

Configuring a Peer IP Address

When you configure redundancy, by default, configuration mode on the standby ACE is disabled and changes on an active ACE are automatically synchronized on the standby ACE. However, interface IP addresses on the active and standby ACEs must be unique. To ensure that the addresses on the interfaces are unique, the IP address of an interface on the active ACE is synchronized on the standby ACE as the peer IP address.

To configure the IP address for an interface on a standby ACE, use the **peer ip address** command in interface configuration mode. The peer IP address on the active ACE is synchronized on the standby ACE as the interface IP address. The syntax of this command is:

peer ip address *ip_address mask* [**secondary**]

The arguments and option are as follows:

- *ip_address mask*—IP address and mask for the peer ACE.
- **secondary**—(Optional) Configures the address as a secondary peer IP address. You can configure a maximum of 4 secondary peer addresses. The ACE has a system limit of 1024 secondary peer addresses.

Note The peer IP address must be unique across multiple contexts on a shared VLAN.

When the destination for the control plane (CP)-originated packets is Layer 2 adjacent to either the primary subnet or one of the secondary subnets, the ACE always uses the appropriate primary or secondary interface IP address that belongs to the destination subnet as the source IP address. For any destination that is not Layer 2 adjacent, the ACE uses the primary address as the source IP address.

For packets destined to the secondary IP address, the ACE sends the response with the secondary IP address as the source address.

SSL probes always uses the primary IP address as the source address for all destinations.

You cannot configure secondary IP addresses on FT VLANs.

For example, to configure an IPv4 address and netmask of the peer ACE, enter:

```
host1/Admin(config-if)# peer ip address 192.168.1.20 255.255.255.0
```

To configure a secondary IP address and mask for the peer ACE, enter:

```
host1/Admin(config-if)# peer ip address 10.10.1.2 255.255.255.0
secondary
```

To delete the IP address for the peer ACE, enter:

```
host1/Admin(config-if)# no peer ip address
```

To delete the secondary IP address for the peer ACE, enter:

```
host1/Admin(config-if)# no peer ip address 10.10.1.2 255.255.255.0
secondary
```

Configuring an Alias IP Address

When you configure redundancy with active and standby ACEs, you can configure a VLAN interface that has an alias IP address that is shared between the active and standby ACEs. The alias IP address serves as a shared gateway for the two ACEs in a redundant configuration.

Note You must configure redundancy (fault tolerance) on the ACE for the alias IP address to work. For more information on redundancy, see the *Administration Guide, Cisco ACE Application Control Engine*.

You cannot configure secondary IP addresses on FT VLANs.

The ACE also uses an alias IP address assigned to a VLAN to address a network device that you want to hide from the rest of the network. Typically, you assign alias IP addresses to VLANs with stealth firewalls so that the firewall remains invisible. An ACE uses the alias IP address configured on another ACE as the destination of the load-balancing process to direct flows through the firewalls. For details about configuring firewalls and firewall load balancing (FWLB) on the ACE, refer to the *Server Load-Balancing Guide, Cisco ACE Application Control Engine*.

To configure an alias IP address, use the **alias** command in interface configuration mode. The syntax of this command is as follows:

alias *ip_address netmask* [**secondary**]

The arguments and option are as follows:

- *ip_address mask*—Alias IP address and subnet mask. Enter the IP address and subnet mask in dotted-decimal notation (for example, 192.168.1.30 255.255.255.0).
- **secondary**—(Optional) Configures the address as a secondary alias IP address. You can configure a maximum of 15 secondary addresses. The ACE has a system limit of 1024 secondary alias addresses.

  The secondary alias address becomes active only when the corresponding secondary IP address on the same subnet is configured. If you remove the secondary IP address, the secondary alias address becomes inactive.

For example, to configure an alias IP address, enter:

```
host1/Admin(config-if)# alias 192.168.1.30 255.255.255.0
```

To configure a secondary alias IP address, enter:

```
host1/Admin(config-if)# alias 11.11.1.3 255.255.255.0 secondary
```

To remove an alias IP address, enter:

```
host1/Admin(config-if)# no alias 192.168.1.30 255.255.255.0
```

To remove a secondary alias IP address, enter:

```
host1/Admin(config-if)# no alias 11.11.1.3 255.255.255.0 secondary
```

Disabling and Enabling Traffic on Interfaces

When you configure an interface, the interface is in the shutdown state until you enable it. If you disable or reenable the interface within a context, only that context interface is affected.

Note When you enable the interface, all of its configured primary and secondary addresses are enabled. You must configure a primary IP address to enable an interface. The ACE does not enable an interface with only secondary addresses. When you disable an interface, all of its configured primary and secondary addresses are disabled.

To enable the interface, use the **no shutdown** command in interface configuration mode. For example, enter:

```
host1/Admin(config-if)# no shutdown
```

To disable a VLAN, use the **shutdown** command in interface configuration mode. The syntax of this command is as follows:

shutdown

For example, to disable VLAN 3, enter:

```
host1/Admin(config)# interface vlan 3
host1/Admin(config-if)# shutdown
```

Configuring the IPv6 MTU for a Layer 3 interface

For IPv6, the minimum MTU is 1280 and the default maximum transmission unit (MTU) is 1500 bytes. This value is sufficient for most applications, but you can pick a lower number if network conditions require this value (for example, to avoid fragmentation over IPSec tunnels) or a larger value (for example, for jumbo frames). Data that is larger than the MTU value is fragmented before being sent.

Caution If you configure a Layer 7 policy map and set the maximum transmit unit (MTU) of the ACE server-side VLAN lower than the client maximum segment size (MSS), ensure that the maximum value of the MSS that you set for the ACE using the **set tcp mss max** command is at least 40 bytes (size of the TCP header plus options) less than the MTU of the ACE server-side VLAN. Otherwise, the ACE may discard incoming packets from the server.

Note If you configure dynamic workload scaling (DWS) on the ACE, set the server-side VLAN MTU to 1430 bytes or less to accommodate a few bytes of overhead from the OTV encapsulation and the DF bit that is set by default within the OTV link.

Keep in mind the following configuration retrictions and guidelines when you configure an MTU for IPv6 traffic:

- You can configure the **ipv6 mtu** under a Layer 3 interface only (Layer 3 VLAN or BVI).
- The ACE will not recognize the **ipv6 mtu** command under a transparent VLAN (Layer 2 VLAN).

To specify the MTU for an IPv6 interface, use the **ipv6 mtu** command in interface configuration mode. This command allows you to set the data size that is sent on a connection. The syntax of this command is as follows:

ipv6 mtu *bytes*

The *bytes* argument is the number of bytes in the MTU. For IPv6, enter a number from 1280 to 9216 bytes. The default is 1500.

For example, to specify the MTU data size of 1360 for a Layer 3 interface, enter the following command:

```
host1/Admin(config-if)# ipv6 mtu 1360
```

To reset the MTU block size to 1500 bytes, use the **no ipv6 mtu** command. For example, enter the following command:

```
host1/Admin(config-if)# no ipv6 mtu
```

Configuring the IPv4 MTU for an Interface

For IPv4, the default MTU is a 1500-byte block for Ethernet interfaces. This value is sufficient for most applications, but you can pick a lower number if network conditions require this value (for example, to avoid fragmentation over IPSec tunnels) or a larger value (for example, for jumbo frames). Data that is larger than the MTU value is fragmented before being sent.

Caution If you configure a Layer 7 policy map and set the maximum transmit unit (MTU) of the ACE server-side VLAN lower than the client maximum segment size (MSS), ensure that the maximum value of the MSS that you set for the ACE using the **set tcp mss max** command is at least 40 bytes (size of the TCP header plus options) less than the MTU of the ACE server-side VLAN. Otherwise, the ACE may discard incoming packets from the server.

Note If you configure dynamic workload scaling (DWS) on the ACE, set the server-side VLAN MTU to 1430 bytes or less to accommodate a few bytes of overhead from the OTV encapsulation and the DF bit that is set by default within the OTV link.

To specify the MTU for an interface, use the **mtu** command in interface configuration mode. This command allows you to set the data size that is sent on a connection. The syntax of this command is as follows:

mtu *bytes*

The *bytes* argument is the number of bytes in the MTU. For IPv4, enter a number from 68 to 9216 bytes. The default is 1500.

For example, to specify the MTU data size of 1000 for an interface:

```
host1/Admin(config-if)# mtu 1000
```

To reset the MTU block size to 1500 bytes, use the **no mtu** command. For example, enter:

```
host1/Admin(config-if)# no mtu
```

Autogenerating a MAC Address for a VLAN Interface

By default, the ACE does not allow traffic from one context to another context over a transparent firewall. The ACE assumes that VLANs in different contexts are in different Layer 2 domains, unless it is a shared VLAN. The ACE allocates the same MAC address to the VLANs.

When you are using a firewall service module (FWSM) to bridge traffic between two contexts on the ACE, you must assign two Layer 3 VLANs to the same bridge domain. To support this configuration, these VLAN interfaces require different MAC addresses.

To enable the autogeneration of a MAC address on a VLAN interface, use the **mac-address autogenerate** command in interface configuration mode. The syntax of this command is as follows:

mac-address autogenerate

For example, enter:

```
host1/Admin(config-if)# mac-address autogenerate
```

To disable MAC address autogeneration on the VLAN, use the **no mac-address autogenerate** command. For example, enter:

```
host1/Admin(config-if)# no mac-address autogenerate
```

Note When you use the **mac-address autogenerate** command, the ACE assigns a MAC address from the bank of MAC address for shared VLANs. If you use the **no mac address autogenerate** command, the interface retains this address. To revert to a MAC address for an unshared VLAN, you must delete the interface and then add the interface again.

Enabling the Mac-Sticky Feature

The mac-sticky feature ensures that the ACE sends return traffic to the same upstream device through which the connection setup from the original client was received. When you enable this feature, the ACE uses the source MAC address from the first packet of a new connection to determine the device to send the

return traffic. This guarantees that the ACE sends the return traffic for load-balanced connections to the same device originating the connection. By default, the ACE performs a route lookup to select the next hop to reach the client.

This feature is useful when the ACE receives traffic from Layer 2 and Layer 3 adjacent stateful devices, like firewalls and transparent caches, guaranteeing that it sends return traffic to the correct stateful device that sourced the connection without any requirement for source NAT. For more information on firewall load balancing, see the *Security Guide, Cisco ACE Application Control Engine*.

To enable the mac-sticky feature for a VLAN interface, use the **mac-sticky enable** command in interface configuration mode. By default, the mac-sticky feature is disabled on the ACE. The syntax of this command is:

mac-sticky enable

Note You cannot use this command if you configure the **ip verify reverse-path** command. For information on the **ip verify reverse-path** command, see the *Security Guide, Cisco ACE Application Control Engine*.

For example, to enable the mac-sticky feature, enter:

```
host1/Admin(config-if)# mac-sticky enable
```

To disable the mac-sticky feature, use the **no mac-sticky enable** command. For example, enter:

```
host1/Admin(config-if)# no mac-sticky enable
```

Providing an Interface Description

You can provide a description for the interface by using the **description** command in interface configuration mode. The syntax of this command is as follows:

description *text*

The *text* argument is the description for the interface. Enter an unquoted text string that contains a maximum of 240 alphanumeric characters including spaces.

For example, to provide the description of POLICY MAP 3 FOR INBOUND AND OUTBOUND TRAFFIC, enter:

```
host1/Admin(config-if)# description POLICY MAP3 FOR INBOUND AND
OUTBOUND TRAFFIC
```

To remove the description for the interface, use the **no description** command. For example, enter:

```
host1/Admin(config-if)# no description
```

Configuring the UDP Booster Feature

When a network application requires very high UDP connection rates, configure the UDP booster feature. For detailed information concerning this feature and its configuration, see the *Server Load-Balancing Guide, Cisco ACE Application Control Engine*. To enable this feature, use the **udp** command in interface configuration mode. The syntax of this command is as follows:

udp {**ip-source-hash** | **ip-destination-hash**}

The keywords are as follows:

- **ip-source-hash**—Instructs the ACE to hash the source IP address of UDP packets that hit a source-hash VLAN interface prior to performing a connection match. Configure this keyword on a client-side interface.
- **ip-destination-hash**—Instructs the ACE to hash the destination IP address of UDP packets that hit a destination-hash VLAN interface prior to performing a connection match. Configure this keyword on a server-side interface.

For example, for a client-side interface, to enable the UDP hash forwarding on the source IP address of the UDP packets, enter:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# udp ip-source-hash
```

To disable this feature, enter:

```
host1/Admin(config-if)# no udp ip-source-hash
```

Removing the ACE Ethernet IP Packet Trailing Byte

By default, the ACE does not perform an internal length check on an Ethernet IP packet to determine whether there are any trailing bytes appended to it. If the packet has an appended byte, the ACE would ignore and forward the packet.

To enable an internal length check and remove the trailing byte appended to the end of an Ethernet IP packet coming into the ACE, use the **remove-eth-pad** command in interface configuration mode for the VLAN. The syntax of this command is as follows:

remove-eth-pad

For example, enter:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# remove-eth-pad
```

To disable an internal length check and the removal of the trailing byte, enter:

```
host1/Admin(config-if)# no remove-eth-pad
```

Assigning a Policy Map to an Interface

When you assign a policy map to a VLAN interface, the ACE can use the map to evaluate all network traffic on the interface. For more information on configuring policy maps, see the *Administration Guide, Cisco ACE Application Control Engine*.

You can apply one or more policy maps to a VLAN interface or globally to all VLAN interfaces in the same context. A policy map activated on an interface overwrites any specified global policy maps for overlapping classifications and actions.

You can assign multiple policy maps on an interface. However, the ACE allows only one policy map to be active on an interface at a given time. The order in which you configure the policy maps on the ACE is important.

To assign a policy map to an interface, use the **service-policy** command in interface configuration mode for an individual interface, or use the **service-policy** command in configuration mode for all interfaces in the same context.

The syntax of this command is as follows:

service-policy input *policy_name*

The keyword and argument are as follows:

- **input**—Specifies that the traffic policy is to be attached to the inbound direction of an interface. The traffic policy evaluates all traffic received by that interface.
- *policy_name*—Previously configured policy map that you want to apply to the interface.

For example, to specify a VLAN interface and apply multiple service policies to a VLAN, enter:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# service-policy input L4_SLB_POLICY
```

For example, to globally apply multiple service policies to all of the VLANs associated with a context, enter:

```
host1/Admin(config)# service-policy input L4_SLB_POLICY
```

To remove a traffic policy from a VLAN interface, enter:

```
host1/Admin(config-if)# no service-policy input L4_SLB_POLICY
```

To globally remove a traffic policy from all VLANs associated with a context, enter:

```
host1/Admin(config)# no service-policy input L4_SLB_POLICY
```

Applying an Access List to an Interface

To allow the traffic to pass on an interface, you must apply ACLs to a VLAN interface. You can apply one ACL of each type (extended, ICMP, or EtherType) to both directions of the interface. For more information about ACLs and ACL directions, see the *Security Guide, Cisco ACE Application Control Engine*.

For connectionless protocols, you must apply the ACL to the source and destination interfaces if you want traffic to pass in both directions. For example, to allow Border Gateway Protocol (BGP) in an ACL in transparent mode, you must apply the ACL to both interfaces.

To apply an ACL to the inbound or outbound direction of an interface and make the ACL active, use the **access-group** command in interface configuration mode.

The syntax of this command is as follows:

access-group {**input** | **output**} *acl_name*

The options and arguments are as follows:

- **input**—Specifies the inbound direction of the interface to apply the ACL.
- **output**—Specifies the outbound direction of the interface t o apply the ACL.
- *acl_name*—Identifier of an existing ACL to apply to an interface.

For example, enter:

```
host1/Admin(config)# interface vlan100
host1/Admin(config-if)# access-group input INBOUND
```

To remove an ACL from an interface, use the **no access-group** command. For example, enter:

```
host1/Admin(config-if)# no access-group input INBOUND
```

Displaying Interface Information

You can display information for the interfaces by using the **show interface** command. This section contains the following topics:

- Displaying IPv6 VLAN and BVI Information
- Displaying IPv6 Interface Summary Information
- Displaying IPv4 VLAN and BVI Information
- Displaying IPv4 VLAN and BVI Summary Statistics
- Displaying the ACE Module Interface Ethernet Out-of-Band Channel Information
- Displaying the Internal Interface Manager Tables
- Displaying ACE Module VLANs Downloaded from the Supervisor Engine
- Displaying ACE Module Private VLAN Information

(ACE appliance only) You can display information for an Ethernet data port, Ethernet management port, or a port-channel virtual interface by using the **show interface** command. See Chapter 1, "Configuring ACE Appliance Ethernet Interfaces" for details.

Displaying IPv6 VLAN and BVI Information

You can use the show ipv6 interface command in Exec mode to display the IPv6 statistics for all VLANs and BVIs or a specified VLAN or BVI interface. The syntax of this command is as follows:

show ipv6 interface [**bvi** *number* | **vlan** *number*]

The **bvi** | **vlan** *number* options display the IPv6 information for the specified VLAN or bridge-group virtual interface number.

If you enter the **show ipv6 interface** command with no options, the ACE displays all VLAN and BVI interfaces. For example, to display IPv6 interface information for BVI 23, enter:

```
host1/Admin# show ipv6 interface bvi 23
```

Table 3-2 describes the fields in the **show ipv6 interface** command output.

Table 3-2 Field Descriptions for the show ipv6 interface Command Output

Field	Description
VLAN_name/ *BVI_number* is	State of the specified VLAN or BVI (up, down, administratively up or down) and the reason for the transition to the state. **Note** (ACE module only) The command output can include VLANs 1006 to 1011, which are reserved VLANs used by the ACE module and supervisor engine.
IPv6	State of IPv6: enabled or disabled.

Table 3-2 Field Descriptions for the show ipv6 interface Command Output (continued)

Field	Description
Link-local address	Link-local IPv6 address assigned to the interface and the DAD status. Possible values for the DAD status are: • DUPLICATE—IPv6 address is already owned by another device. The address is inactive. • INACTIVE—Address is not installed. • N/A—Address is installed and DAD was not performed. DAD is not performed for multiple-address VIPs (those with prefix lengths less than128) or for out-of-subnet VIPs. In this case, the VIP immediately transitions to active. The interface addresses do not use this state. • PASSED—Address successfully passed DAD and is active. This state can occur only for in-subnet /128 VIPs. • TENTATIVE—address is installed and presently undergoing DAD. DAD is done only for single-address VIPS in the same subnet, and for interface addresses. The address is inactive when in this STATE.
Global unicast address	Global unicast IPv6 address assigned to the interface, subnet ID, and the DAD status.
Alias address	Alias IPv6 address, subnet ID, and DAD status.
Peer IP Addresses	
Peer global address	Peer global address, subnet ID, and DAD status.
Joined group addresses	List of joined multicast groups.
MTU	Configured MTU in bytes.
ICMP error message rate	Configured ICMP message rate.
ICMP redirects	Status of ICMP redirects: enabled or disabled.

Table 3-2 Field Descriptions for the show ipv6 interface Command Output (continued)

Field	Description
ICMP unreachables	Status of ICMP unreachable messages: sent or not sent.
ND DAD	Status: enable or disabled and the number of DAD attempts for neighbor discovery (ND).
ND reachable time	Configured ND reachable time in milliseconds.
ND advertised retransmit interval	Configured retransmit interval for NA messages in milliseconds.
ND Router Advertisement (RA) Information	
RA interval	Configured transmit interval for RA messages in seconds.
RA live	Configured time during which router advertisements are valid in seconds.
RA hop limit	Configured maximum number of hops for RA messages.
RA suppressed	The ACE is configured not to send RA messages to neighbors in response to RS messages.
Hosts use DHCP	Hosts should use DHCP to obtain routable addresses and other configurations
Hosts use stateless autoconfig	Hosts should use stateless autoconfig for addresses.
Neighbor Discovery (ND) Prefix Information	
IPv6 address	IPv6 prefix and length for ND.
No-advertise	ACE is configured not to advertise the ND prefix.
Valid	Actual length of time that neighboring nodes should consider the ACE as the default router before they send RS messages again.
Preferred lifetime	Preferred length of time that neighboring nodes should consider the ACE as the default router before they send RS messages again.

Displaying IPv6 Interface Summary Information

You can use the **show ipv6 interface brief** command in Exec mode to view summary information for all VLANs and BVIs or a specified VLAN or BVI. The syntax of this command is as follows;

show ipv6 interface brief [**bvi** *number* | **vlan** *number*]

The **bvi** | **vlan** *number* options display the IPv6 information for the specified VLAN or bridge-group virtual interface number.

If you enter the **show ipv6 interface** command with no options, the ACE displays all VLAN and BVI interfaces. For example, to display IPv6 interface summary information for VLAN 300, enter:

```
host1/Admin# show ipv6 interface brief vlan 300
```

Table 3-3 describes the fields in the **show ipv6 interface brief** command output.

Table 3-3 Field Descriptions for the show ipv6 interface brief Command Output

Field	Description
Interface	VLAN or bridge-group virtual interface number. (ACE appliance only) For FT interfaces, (ft) appears after the VLAN number. This command also displays the physical interfaces.
IP Address	Link-local, global, and alias addresses for the VLAN interface. The field displays unassigned for the following conditions: • The interface is not assigned. • (ACE appliance only) The interface is a physical interface.
Status	State of the interface: up, down, administratively up, administratively down.
Protocol	Status of the line protocol: either up or down.

Displaying IPv4 VLAN and BVI Information

You can use the **show interface** command in Exec mode to display the details, statistics, or IP information for all or a specified VLAN or BVI interface. The syntax of this command is as follows:

show interface [**bvi** *number* | **vlan** *number*]

The **bvi | vlan** *number* options display the information for the specified VLAN or bridge-group virtual interface number.

If you enter the **show interface** command with no options, the ACE displays all VLAN and BVI interfaces. For example, enter:

```
host1/Admin# show interface
```

Note The ACE has no counters specifically for traffic received or sent through secondary IP addresses. All counters are at the interface level or associated with the primary IP address.

Table 3-4 describes the fields in the **show interface** command output.

Table 3-4 Field Descriptions for the show interface Command Output

Field	Description
VLAN_name/ *BVI_number* is	State of the specified VLAN or BVI (up, down, administratively up or down) and the reason for the transition to the state. **Note** (ACE module only) The command output can include VLANs 1006 to 1011, which are reserved VLANs used by the ACE module and supervisor engine.
Hardware type is	Hardware type of the interface: either VLAN or BVI.
MAC address	MAC address of the system mapped to the IP address. Note that the BVI MAC address is the same address as an associated bridge-group VLAN address.

Table 3-4 Field Descriptions for the show interface Command Output (continued)

Field	Description
Mode	Mode associated with the VLAN or BVI. A bridge-group VLAN is displayed as transparent. A routed VLAN or BVI is displayed as routed. Otherwise, this field displays the value "unknown."
FT status	Status of whether the interface is redundant.
Description	Description for the VLAN or BVI.
MTU	Configured MTU in bytes.
Last cleared	Last time that the VLAN or BVI was cleared.
Last Changed	Timestamp when the last change occurred.
No. of transitions	Number of transitions that the interface experienced since it was created.
Alias IP address	Configured alias IP address.
Peer IP address/netmask	Configured peer IP address and netmask.
Virtual MAC address	MAC address used by the alias IP address and VIP address when the interface is in the redundant active state (displayed only if the interface is in this state).
[Not] Assigned - Supervisor	(ACE module only) Whether the VLAN or BVI is assigned from the supervisor engine and is up or down on the supervisor engine.
[Not] Assigned on physical port...	(ACE appliance only) Whether the interface is assigned on the physical port and is in the up or down state.
Previous State	Last three previous states including the timestamp and the reason for the up or down transitions.
# unicast packets input, # bytes	Total number of incoming unicast packets and number of bytes.
# multicast, # broadcast	Total number of incoming multicast and broadcast packets.

Table 3-4 Field Descriptions for the show interface Command Output (continued)

Field	Description
# input errors, # unknown, # ignored, # unicast RFP drops	Total number of errors for incoming packets, including numbers for packets that are unknown, ignored, and RFP drops.
# unicast packets output, # bytes	Total number of outgoing unicast packets and number of bytes.
# multicast, # broadcast	The total number of outgoing multicast and broadcast packets.
# output errors, # ignored	Number of errors for outgoing packets, including unknown packets.

Displaying IPv4 VLAN and BVI Summary Statistics

You can use the **show ip interface brief** command in Exec mode to display a brief configuration and status summary of all interfaces or a specified BVI or a VLAN display. The syntax of this command is as follows:

show ip interface brief [**bvi** *number* | **vlan** *number*]

The **bvi | vlan** *number* options display the information for the specified VLAN or bridge-group virtual interface number.

If you enter the **show ip interface brief** command with no options, the ACE displays all VLAN and BVI interfaces. For example, enter:

```
host1/Admin# show ip interface brief
```

Table 3-5 describes the fields in the **show ip interface brief** command output.

Table 3-5 Field Descriptions for the show ip interface brief Command Output

Field	Description
Interface	VLAN or bridge-group virtual interface number. (ACE appliance only) For FT interfaces, (ft) appears after the VLAN number. This command also displays the physical interfaces.
IP Address	IP address and mask for the VLAN interface. The field displays unassigned for the following conditions: • The interface is not assigned. • (ACE appliance only) The interface is a physical interface.
Status	State of the interface: up, down, administratively up, administratively down.
Protocol	Status of the line protocol: either up or down.

Displaying the ACE Module Interface Ethernet Out-of-Band Channel Information

You can display the ACE module Ethernet out-of-band channel (EOBC) information by using the **show interface eobc** command in Exec mode. This command is available in the Admin context only. For example, enter:

```
host1/Admin# show interface eobc
```

Table 3-6 describes the fields in the **show interface eobc** command output for the ACE module.

Table 3-6 Field Descriptions for the show interface eobc Command Output

Field	Description
Hardware type	Hardware type is EOBC.
MAC address	MAC address of the system mapped to the IP address.

Table 3-6 Field Descriptions for the show interface eobc Command Output (continued)

Field	Description
Description	Description for the VLAN.
MTU	MTU in bytes.
BW # bits/sec	Bits per second on the bus width.
IP address	Internal IP address.
# unicast packets input, # bytes	Total number of incoming unicast packets and number of bytes.
# input errors, # ignored	Number of errors for incoming packets, including numbers for packets that are ignored.
# unicast packets output, # bytes	Total number of outgoing unicast packets and number of bytes.
# output errors, # ignore	Number of errors for outgoing packets, including numbers for packets that are ignored.

Displaying the Internal Interface Manager Tables

You can display the internal interface manager tables and events by using the **show interface internal** command in Exec mode. The syntax of this command is as follows:

show interface internal {**event-history** {**dbg** | **mts**} |
iftable [*interface_name*] | **seciptable** | **vlantable** [*vlan_number*]}

The keywords and arguments are as follows:

- **event-history** {**dbg** | **mts**}—Displays the debug history (dbg) or message history (mts). This keyword is available in the Admin context only.
- **iftable** [*interface_name*]—Displays the master interface table. If you specify an interface name, the ACE displays the table information for that interface.
- **seciptable**—Displays the interface manager's (ifmgr) view of a logical interface and displays all the configured secondary IP addresses under an interface.

- **vlantable** [*vlan_number*]—Displays the VLAN table. If you specify an interface number, the ACE displays the table information for that interface.

Note The **show interface internal** command is used for debugging purposes. The output for this command is for use by trained Cisco personnel as an aid in debugging and troubleshooting the ACE. For information on the command syntax, see the *Command Reference, Cisco ACE Application Control Engine*.

For example, to display the interface internal debug event history starting with the most recent event, enter:

```
host1/Admin# show interface internal event-history dbg
```

To display the interface internal message event history starting with the most recent event, enter:

```
host1/Admin# show interface internal event-history mts
```

To display the master interface table, enter:

```
host1/Admin# show interface internal iftable
```

To display the master VLAN table, enter:

```
host1/Admin# show interface internal vlantable
```

Displaying ACE Module VLANs Downloaded from the Supervisor Engine

You can use the **show vlans** command in Exec mode for the Admin context to display the VLANs on the ACE module downloaded from the supervisor engine. For example, enter:

```
host1/Admin# show vlans
Vlans configured on SUP for this module
    vlan192-193 vlan333
```

Displaying ACE Module Private VLAN Information

The private VLAN feature on the Catalyst 6500 series switch or Cisco 7600 series router works with the ACE module. The Cisco IOS PVLAN configuration populates the PVLAN mapping database on the ACE module. See the documentation for the switch or router for detailed information.

To display the private VLANs on the ACE module that are downloaded from the supervisor engine, use the **show pvlans** command in Exec mode. For example, enter:

```
host1/Admin# show pvlans
```

Table 3-7 describes the fields in the **show pvlans** command output.

Table 3-7 Field Descriptions for the show pvlans Command Output

Field	Description
Primary	VLAN number for the primary private VLAN.
Secondary	VLAN number for the secondary private VLAN.
Type	One of the three ways that the private VLAN uses VLANs: primary, isolated, or community.

Clearing Interface Statistics

You can clear the statistics displayed through the **show interface** command by using the **clear interface** command in Exec mode. The syntax of this command is as follows:

clear interface [**vlan** *number* | **bvi** *number*]

If you do not enter an option and argument, the statistics for all VLANs and BVIs are set to zero. The options and arguments are as follows:

- **vlan** *number*—Clears the statistics for the specified VLAN.
- **bvi** *number*—Clears the statistics for the specified BVI. Statistics are not collected for BVI interfaces. The packets are counted against the underlying bridged (Layer 2) interfaces.

For example to clear the statistics for VLAN 10, enter:

```
host1/Admin# clear interface vlan 10
```

Note If you configure redundancy, you must explicitly clear the statistics (hit counts) on both the active and the standby ACEs. If you clear the statistics on the active ACE only, the standby ACE statistics remain at the old values.

CHAPTER 4

Configuring Routes on the ACE

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This chapter describes how the ACE is considered a router hop in the network when it is in routed mode. In the Admin or user contexts, the ACE supports static routes only. The ACE supports up to eight equal cost routes for load balancing.

This chapter describes how to configure a default or static route on the ACE and contains the following major sections:

Assigning an IP Address to Interfaces for Routing Traffic

When you assign an IP address on an interface, its mode automatically becomes routed. To assign an IP address to a VLAN interface, use the **ip address** command in interface VLAN configuration mode.

IPv6 Syntax and Example

The syntax of this command is as follows:

ip address *ipv6_address/prefix_length* [**eui64**]

The keywords and arguments are as follows;

- *ipv6_address*—Complete IPv6 address with a prefix (for example, 2001::CAFE).
- */prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. If you use the optional **eui64** keyword, the prefix length must be less than or equal to /64.
- **eui64**—(Optional) Specifies that the low order 64 bits are automatically generated in the IEEE 64-bit Extended Unique Identifier (EUI-64) format specified in RFC 2373. To use this keyword, the prefix length must be configured as less than or equal to 64 and the host segment must be all zeros. For more information about EUI64, see Chapter 2, Overview of IPv6.

To configure the IPv6 global address of 2001:DB8:1::CAFE on VLAN 200, enter the following commands:

```
host1/Admin(config)# interface VLAN 200
host1/Admin(config-if)# ip address 2001:DB8:1::CAFE/64
```

To remove this IPv6 global address from the interface, enter the following command:

```
host1/Admin(config-if)# no ip address 2001:DB8:1::CAFE/64
```

IPv4 Syntax and Example

The syntax of this command is as follows:

ip address *ip_address mask*

The *ip_address mask* arguments specify the IP address and mask of the VLAN interface.

For detailed information on configuring an IP address on an interface, see Chapter 3, "Configuring VLAN Interfaces."

To set the IP address of 192.168.1.1 255.255.255.0 on VLAN 200, enter:

```
host1/Admin(config)# interface vlan 200
host1/Admin(config-if)# ip address 192.168.1.1 255.255.255.0
```

To remove this IPv4 address from the interface, enter the following command:

```
host1/Admin(config-if)# no ip address 192.168.1.1 255.255.255.0
```

Note If you make a mistake while entering this command, you can reenter the command with the correct information.

Configuring a Default or Static Route

Admin and user contexts do not support dynamic routing. You must use static routes for any networks to which the ACE is not directly connected; for example, you must use a static route when there is a router between a network and the ACE.

For traffic that originates on or is routed through the ACE and is destined for a nondirectly connected network, configure either a default route or static routes so that the ACE knows where to send the traffic. Traffic that originates on the ACE might include communications to a syslog server, Websense or N2H2 server, or AAA server.

The simplest option is to configure a default route to send all traffic to an upstream router. The default route identifies the router IP address where the ACE sends all IP packets for which it does not have a route. You can configure a maximum of eight default ECMP routes or gateways in the ACE. For IPv6, one of these can be a link-local address.

Note Routes that identify a specific destination address take precedence over the default route.

To set a default or static route, use the **ip route** command in configuration mode.

IPv6 Syntax and Example

The syntax of this command is as follows:

ip route *ipv6_dest_address*/*prefix_length* {*global_nexthop_address* | {**bvi** *number* | **vlan** *number* {*link_local_address*}}}

The keywords and arguments are as follows:

- *ipv6_dest_address*—Destination IPv6 address for the route.
- */prefix_length*—Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128.
- *global_nexthop_address*—IP address of the gateway router (the next-hop address for this route). The gateway address must be in the same network as specified in the **ip address** command for a VLAN interface. For information on configuring the address, see the "Assigning an IP Address to Interfaces for Routing Traffic" section.

Note When you configure a default gateway, the MAC address of the gateway must not constantly change. We recommend to use a Hot Standby Router Protocol (HSRP) IP address or other virtual IP address which maintains a single MAC address for multiple interfaces.

- **bvi** *number*—Forward bridged VLAN interface for the link-local address
- *link_local_address*—Link-local address of the gateway
- **vlan** *number*—Forward VLAN interface for the link-local address

To configure a static route to send all traffic destined to 2001:DB8:1::/64 to the next-hop router at 2001:DB8:2::1, enter the following command:

```
host1/Admin(config)# ip route 2001:DB8:1::/64 2001:DB8:2::1
```

To configure a default route, set the IPv6 address for the route to ::/0, the IPv6 equivalent of "any." For example, if the ACE receives traffic that does not have a route and you want the ACE to send the traffic out the interface to the router at 2001:DB8:2::1, enter:

```
host1/Admin(config)# ip route ::/0 2001:DB8:2::1
```

To remove a default or static route, use the no form of the command as follows:

```
host1/Admin(config)# no ip route 2001:DB8:1::/64 2001:DB8:2::1
```

IPv4 Syntax and Example

The syntax of this command is as follows:

ip route *dest_ip_prefix netmask gateway_ip_address*

The keywords, arguments, and options are as follows:

- *dest_ip_prefix*—IP address for the route. Enter the address in dotted-decimal IP notation (for example, 192.168.20.1).
- *netmask*—Subnet mask for the route. Enter the subnet mask in dotted-decimal notation (for example, 255.255.255.0).
- *gateway_ip_address*—IP address of the gateway router (the next-hop address for this route). The gateway address must be in the same network as specified in the **ip address** command for a VLAN interface. For information on configuring the address, see the "Assigning an IP Address to Interfaces for Routing Traffic" section.

Note When you configure a default gateway, the MAC address of the gateway must not constantly change. We recommend to use a Hot Standby Router Protocol (HSRP) IP address or other virtual IP address which maintains a single MAC address for multiple interfaces.

Note Management traffic coming into the ACE is not affected by the **no normalization** command, which does not support asymmetric routes. For information about normalization, see the *Security Guide, Cisco ACE Application Control Engine*.

To configure a static route to send all traffic destined for 10.1.1.0/24 to the router (10.1.2.45), enter:

```
host1/Admin(config)# ip route 10.1.1.0 255.255.255.0 10.1.2.45
```

To configure a default route, set the IP address and the subnet mask for the route to 0.0.0.0. For example, if the ACE receives traffic that does not have a route and you want the ACE to send the traffic out the interface to the router at 192.168.4.8, enter:

```
host1/Admin(config)# ip route 0.0.0.0 0.0.0.0 192.168.4.8
```

To remove a default or static route, use the no form of the command as follows:

```
host1/Admin(config)# no ip route 192.168.42.0 255.255.255.0
192.168.1.5 1
```

Advertising an ACE Module VLAN for RHI (ACE module only)

To advertise an ACE module VLAN for route health injection (RHI) that is different from the VIP interface VLAN, use the **ip route inject vlan** command in interface configuration mode. By default, the ACE module advertises the VLAN of the VIP interface for RHI.

Note RHI for IPv6 routes is not supported at this time. However, RHI for IPv4 routes is fully functional.

Use this command when there is no directly shared VLAN between the ACE module and the Catalyst 6500 series supervisor engine. This topology can occur when there is an intervening device, for example, a Cisco Firewall Services Module (FWSM), configured between the ACE module and the supervisor engine.

Note Be sure to configure this command on the VIP interface of the ACE module.

The syntax of this command is as follows:

ip route inject vlan *vlan_id*

The *vlan_id* is the interface shared between the supervisor engine and the intervening device. Enter it as an integer from 2 to 4090.

For example, to advertise route 200 for RHI, enter:

```
host1/Admin(config-if)# ip route inject vlan 200
```

To restore the ACE module default behavior of advertising the VIP interface VLAN for RHI, enter:

```
host1/Admin(config-if)# no ip route inject vlan 200
```

Using the Supervisor Engine with RHI (ACE Module Only)

The Route Health Injection (RHI) feature allows the ACE module to inject (add) or withdraw (remove) static IPv4 routes in the supervisor engine. The ACE module maintains a hash table of VIP address-mask entries. The hash table includes the address-mask and a chain of interface entries. Each interface entry corresponds to an interface ID on which the VIP address, mask, and context is configured. Each interface entry has a chain of vserver IDs that correspond to the VIP address, mask, and context and the interface ID.

Note RHI for IPv6 routes is not supported at this time. However, RHI for IPv4 routes is fully functional.

The ACE module maintains the following two data structures for processing:

- A chain of vserver IDs per interface object. The ACE module uses this chain for processing if an interface's state changes.
- A chain of interface IDs per vserver object. The ACE module uses this chain for processing if a vserver's state changes.

When the following route-related changes occur, the ACE module performs the described actions:

- When the MSFC mapped VLAN on an interface changes. the ACE module readvertise the route with the updated VLAN number.
- When the IP address of an interface changes the ACE module advertises the route with the updated next hop.

- When the state of an interface changes, the ACE module examines the new state of the interface removes the route from the supervisor or adds a route to the supervisor.
- When the state of a vserver changes, the ACE module determines the vserver that has the best metric value because of this state change. If the vserver has changed, the ACE module advertises the route with the new vserver.
- When a vserver is removed from an interface, the ACE module deletes the VIP entry from the VIP hash table. The ACE module determines the best new vserver and advertises the route with the new vserver ID.
- When a vserver is added to an interface, the ACE module updates the VIP hash table with the new entry. The ACE module determines the best new best vserver and advertises the route that corresponds to the new vserver ID.

The ACE module and the supervisor engine use Switch-Module Configuration Protocol (SCP) messages to insert or withdraw all RHI routes. Only one route insertion or withdrawal is allowed per SCP message. The configuration manager sends all route information to the route manager in the ACE module. The route manager then forwards the route information to the supervisor engine through the SCP module.

Before it sends the route information to the SCP module, the ACE module caches all the routes that are to be sent to the supervisor in case a retransmission is necessary. The ACE module expects a acknowledgement from the supervisor for each request that it sends. If it receives an acknowledgement from the supervisor, the ACE module deletes the entries from the cache. If it does not receive an acknowledgement from the supervisor, the ACE module retransmits the request (both insertion and withdrawal of routes).

Verifying Connectivity of a Remote Host or Server

You can verify the connectivity of a remote host or server by using the **ping** command in Exec mode to send echo messages from the ACE.

The syntax of this command is as follows:

ping [**ip** | **ipv6** [*system_address* [**count** *count* [**size** *size* [**timeout** *time* [**extended commands y** [source] | **n**]]]]]]

The arguments and options are as follows:

- **ip | ipv6**—(Optional) Specifies the IPv4 or IPv6 protocol. If you do not specify the IP protocol, it is inferred from the address.
- *system_address*—(Optional) IP address of a remote host or server to ping. Enter an IPv4 or an IPv6 address depending on whether you specified the **ip** or the **ipv6** keyword. If you do not specify the IP address of the remote host, the CLI prompts you for the information. For information on additional prompts, see Table 4-1.
- **count** *count*—(Optional) Specifies the repeat count. Enter the repeat count as an integer from 1 to 65000. The default is 5.
- **size** *size*—(Optional) Specifies the datagram size. Enter the datagram size as an integer from 36 to 1440. The default is 100.
- **timeout** *time*—(Optional) Specifies the timeout in seconds. Enter the timeout value as an integer from 0 to 3600. The default is 2.
- **extended commands** [**y** | **n**]—The default is **n**. If you specify **y**, the following additional options are available:
  - source address or interface
  - hop count—The default is 255. Enter an integer from 1 to
  - output interface

IPv6 Example

To send a ping to the IPv6 loopback address 0:0:0:0:0:0:0:1, enter the following command:

```
host1/Admin# ping ::1
PING 0:0:0:0:0:0:0:1(::1) 56 data bytes
64 bytes from ::1: icmp_seq=1 ttl=255 time=0.039 ms
64 bytes from ::1: icmp_seq=2 ttl=255 time=0.000 ms
64 bytes from ::1: icmp_seq=3 ttl=255 time=0.000 ms
64 bytes from ::1: icmp_seq=4 ttl=255 time=0.108 ms
64 bytes from ::1: icmp_seq=5 ttl=255 time=0.126 ms

--- 0:0:0:0:0:0:0:1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 8002ms
rtt min/avg/max/mdev = 0.000/0.054/0.126/0.053 ms
```

To abnormally terminate a ping session, press **Ctrl-C**.

Note The first ping may fail because the ND table is not populated with the MAC address of the remote host or server.

IPv4 Example

The following example shows how to send a ping to a server located at IP address 192.168.219.140:

```
host1/Admin# ping 192.168.173.140
PING 192.168.173.140 with timeout = 2, count = 5, size = 100
Response from 192.168.173.140 :  seq 1 time 1.213 ms
Response from 192.168.173.140 :  seq 2 time 0.175 ms
Response from 192.168.173.140 :  seq 3 time 0.210 ms
Response from 192.168.173.140 :  seq 4 time 0.162 ms
Response from 11.1.11.4 :  seq 5 time 0.214 ms
5 packet sent, 5 responses received, 0% packet loss
```

To abnormally terminate a ping session, press **Ctrl-C**.

Note The first ping may fail because the ARP table is not populated with the MAC address for the remote host or server.

The **ping** command provides additional options to verify the connectivity of a remote host or server. To specify these additional parameters, type **ping** at the CLI ACE prompt and press enter.

Table 4-1 summarizes the options and the defaults for the **ping** command.

Table 4-1 Options and Defaults for the ping Command

Option	Description	Default
Target IP address	IP address or hostname of the destination node to ping.	Not applicable
Repeat count	Number of ping packets to be sent to the destination address. Enter an integer from 1 to 65000.	5 packets

Table 4-1 Options and Defaults for the ping Command (continued)

Option	Description	Default
Datagram size	Size of each ping packet in bytes. For IPv6, enter an integer from 48 to 1440. For IPv4, enter an integer from 36 to 1440.	100 bytes
Timeout	Timeout interval in seconds after which a ping request is considered a failure. The ping is not aborted and sends the next ping packet, if any. Enter an integer from 0 to 3600.	2 seconds
Extended commands	Provides additional commands for the ping command.	n(o)

To trace the routes taken for a specified IP address, use the **traceroute** command in Exec mode.

The syntax of this command is as follows:

traceroute [**ip** | **ipv6**] [*ip_address* [**size** *packet*]]]

The arguments and option are as follows:

- **ip** | **ipv6**—(Optional) Specifies the IPv4 or the IPv6 protocol. If you do not specify the IP protocol, it is inferred from the address.
- *ip_address*—(Optional) IP address for the route. Enter an IPv6 address in IPv6 format or an IPv4 address in dotted-decimal notation. This argument is optional. If you do not include it with the command, you are prompted for an IP address.
- **size** *packet*—(Optional) Specifies the packet size. Enter a number from 40 to 452. For IPv6, there is no default. For IPv4, the default is 40.

IPv6 Example

To trace the IPv6 address 2001:DB8:1::/64, enter the following command:

```
host1/Admin# traceroute ipv6 2001:DB8:1::/64
```

To terminate a traceroute session, press **Ctrl-C**.

IPv4 Example

To trace the IP address 192.168.173.140, enter:

```
host1/Admin# traceroute 192.168.173.140
traceroute to 192.168.173.140 (192.168.173.140), 30 hops max, 40 byte
packets
 1  192.86.215.2 (192.86.215.2)  0.558 ms  0.325 ms  0.297 ms
 2  * * *
 3  * * *
```

To terminate a traceroute session, press **Ctrl-C**.

Using Traceroute on the ACE-Configured IP Addresses

You can use traceroute on ACE-configured IP addresses, however there are certain restrictions. When you use traceroute to a configured ACE IP interface:

- ICMP traceroute works when you configure a management policy to permit ICMP traffic, similar to the following examples:

  IPv6 Example

  ```
  class-map type management match-any remote-access
      description ipv6-remote-access-traffic-match
      match protocol icmpv6 anyv6
  ```

  IPv4 Example

  ```
  class-map type management match-any remote-access
      description ipv4-remote-access-traffic-match
      match protocol icmp any
  ```

  Note Most traceroutes use the default protocol of UDP. Use a command line option to change traceroute to ICMP. For example, in Linux, use the -I option.

- UDP or TCP-based traceroute does not work. There is no method to permit UDP or TCP traffic to ephemeral ports going to the ACE.

When you use UDP, TCP, or ICMP-based traceroute to a host behind the ACE, it works as expected. However, the ACE does not appear in the traceroute as a hop. The ACE does not decrement the TTL of IP packets that it forwards.

When you use traceroute to a VIP address configured on the ACE, the ACE does not intercept traceroute packets sent to the configured VIP address. The ACE attempts to match the packet to the load-balance policies. If a protocol match occurs, the ACE sends the packet to the real server that responds to the traceroute accordingly.

Displaying IPv6 Route Information

To display IPv6 routes on the ACE, use the **show ipv6 route** command in Exec mode. The syntax of this command is as follows;

show ipv6 route

For example, enter:

```
host1/Admin# show ipv6 route
```

Table 4-2 describes the fields in the **show ipv6 route** command output.

Table 4-2 Field Description for the show ipv6 route Command

Field	Description
Destination	IPv6 destination address for the route.
Gateway	IPv6 gateway address for the route.
Interface	VLAN or BVI number for this entry.
Flag	Flag to identify the route type and state, as identified by one of the following codes displayed above the output information: • H indicates a host route. • I indicates an interface route. • S indicates a static route. • N indicates a NAT route. • A indicates that the route needs an ND resolve. • E indicates an ECMP route.
Total route entries	Total number of routes in the IPv6 routing table.

To display the route summary for the current context, use the **show ipv6 route summary** command. The syntax of this command is as follows:

show ipv6 route summary

For example, enter:

```
host1/Admin# show ipv6 route summary
```

Table 4-3 describes the fields in the **show ipv6 route summary** command output.

Table 4-3 Field Description for the show ipv6 route summary Command

Field	Description
Route Source	Source of the route. The possible value are as follows: • Connected for a route to hosts that are connected to the same network. • Static for a configured route.
Count	Number of routes that are connected or static.
Memory (bytes)	Memory consumed by the route entries.

To display IPv6 traffic information, use the **show ip traffic** command in Exec mode. For a description of the IPv4 output fields of this command, see the "Displaying IPv4 Route Information" section. The syntax of this command is as follows:

show ip traffic

For example, enter:

```
host1/Admin# show ip traffic
```

Table 4-4 describes the IPv6-specific fields in the **show ip traffic** command output.

Table 4-4 IPv6 Field Descriptions for the show ip traffic Command Output

Field	Description
IPv6 Statistics	
Rcvd	• total—Number of packets received by the ACE. • bytes—Number of bytes received by the ACE. • input errors—Number of receive errors. • no route—Number of packets with no route.
Frags	• reassembled—Number of fragments that the ACE reassembled. • couldn't reassemble—Number of fragments that the ACE could not reassemble. • fragmented—Number of packets that the ACE fragmented. • couldn't fragment—Number of packets that the ACE could not fragment.
Mcast	• received—Number of multicast packets received by the ACE. • sent—Number of multicast packets sent by the ACE.
Sent	• total—Total packets sent. • sent—Number of bytes sent. • no route—Number of packets sent with no route.
Drop	• no route—Number of packets discarded because they had no route. • out discarded—Number of packets discarded.

Table 4-4 IPv6 Field Descriptions for the show ip traffic Command Output (continued)

Field	Description
ICMPv6 Statistics	
Rcvd	• input—Number of packets received by the ACE. • errors—Number of received packet errors. • unreach—Number of ICMPv6 Unreachable messages received by the ACE. • parameter problem—Number of packets that were dropped by the ACE because of a problem with the IPv6 header or extension header fields. • hopcount expired—Number of packets whose hop counts went to zero that were received by the ACE. This message is the same as the Time Exceeded message in RFC4443. • too big—Number of packets received by the ACE that elicited a “packet too big” response because they were too long and could not be sent to their destination. • echo request—Number of ICMPv6 Echo Request packets received by the ACE. • echo reply—Number of ICMPv6 Echo Reply packets received by the ACE. • group query—Number of multicast group query messages received by the ACE. • group report—Number of group report messages received by the ACE. Group report messages are generated when a host joins a multicast group. • group reduce—Number of group reduce messages received by the ACE. Group reduce messages are sent by a member when it leaves a multicast group. • router solicit—Number of Router Solicitation messages received by the ACE.

Table 4-4 IPv6 Field Descriptions for the show ip traffic Command Output (continued)

Field	Description
ICMPv6 Statistics (cont.)	
Rcvd (cont.)	• router solicit drops—Number of Router Solicitation messages that were dropped by the ACE. • router advert—Number of Router Advertisement messages received by the ACE. • redirects—Number of Redirect messages received by the ACE. • neighbor solicit—Number of Neighbor Solicitation messages received by the ACE. • neighbor advert—Number of Neighbor Advertisements received by the ACE.
Sent	• output—Number of packets sent by the ACE • unreach—Number of Destination Unreachable messages sent by the ACE • parameter problem—Number of packets sent by the ACE that had a problem with the IPv6 header or extension header fields • hopcount expired—Number of packets whose hop counts went to zero that were sent by the ACE • too big—Number of packets sent by the ACE that elicited a “packet too big” response because they were too long and could not be sent to the destination • echo reply—Number of Echo Reply messages sent by the ACE • group report—Number of group report messages sent by the ACE. Group report messages are generated when a member joins a multicast group. • group reduce—Number of group reduce messages sent by the ACE. Group reduce messages are sent by a member when it leaves a multicast group.

Table 4-4 IPv6 Field Descriptions for the show ip traffic Command Output (continued)

Field	Description
Sent (cont.)	• router solicit—Number of Router Solicitation messages sent by the ACE. • router advert—Number of Router Advertisement messages sent by the ACE. • redirects—Number of Redirect messages sent by the ACE. • neighbor solicit—Number of Neighbor Solicitation messages sent by the ACE. • neighbor advert—Number of Neighbor Advertisements sent by the ACE.
TCP Statistics	
Rcvd	Total number of TCP segments and errors received by the ACE.
Sent	Total number of TCP segments sent by the ACE.
UDP Statistics	
Rcvd	Total number of UDP segments, UDP errors, and segments with no port number received by the ACE.
Sent	Total number of UDP segments sent by the ACE.
ND Statistics	
Rcvd	Number of ND packets, errors, requests, and responses received by the ACE.
Sent	Number of ND packets, errors, requests, and responses sent by the ACE.

The **show ipv6 route internal** command is used for debugging purposes. The output of this command is for use by trained Cisco personnel as an aid in debugging and troubleshooting the ACE. For information on the command syntax, see the *Command Reference, Cisco ACE Application Control Engine*.

Displaying the IPv6 FIB Table Information

The forwarding information base (FIB) table contains information that the forwarding processors require to make IP forwarding decisions. This table is derived from the route and ND tables. To display the FIB table for the context, use the **show ipv6 fib** command. The syntax of this command is as follows:

show ipv6 fib

For example, enter:

```
host1/Admin# show ipv6 fib
```

Table 4-10 describes the fields in the **show ipv6 fib** command output.

Table 4-5 Field Description for the show ipv6 fib Command

Field	Description
Destination	Destination address for the route.
Interface	VLAN interface number for this entry.
EncapID	Encapsulation identifier.
Flag	Flag to identify the route type and state, as identified by one of the following codes displayed above the output information: • H indicates a host route. • I indicates interface route. • S indicates a static route. • N indicates a NAT route. • A indicates that the route needs an ND resolve. • E indicates an ECMP route. • V indicates that the route destination matches a class map-defined virtual server.
Total route entries	Total number of route entries in the ND table.

To display a summary of the FIB table for the context, use the **show ip fib summary** command. For example, enter:

```
host1/Admin# show ipv6 fib summary
```

Table 4-11 describes the fields in the **show ip fib summary** command output.

Table 4-6 Field Description for the show ip fib summary Command

Field	Description
Resolved routes	Number of prefixes programmed in mtrie.
Leaves, bytes	Number of mtrie leaf nodes allocated and memory consumed in bytes.
Nodes, bytes	Number of mtrie internal nodes allocated and memory consumed in bytes.
ecmps, bytes	Number of ECMP nodes allocated and memory consumed in bytes.

The **show ipv6 fib** command is used for debugging purposes. The output of this command is for use by trained Cisco personnel as an aid in debugging and troubleshooting the ACE. For information on the command syntax, see the *Command Reference, Cisco ACE Application Control Engine*.

Displaying IPv4 Route Information

To display IPv4 routes on the ACE, use the **show ip route** command in Exec mode. The syntax of this command is as follows;

show ip route

For example, enter:

```
host1/Admin# show ip route
```

Table 4-7 describes the fields in the **show ip route** command output.

Table 4-7 Field Description for the show ip route Command

Field	Description
Destination	Destination address for the route.
Gateway	Gateway address for the route.
Interface	VLAN interface number for this entry.
Flag	Flag to identify the route type and state, as identified by one of the following codes displayed above the output information: • H indicates a host route. • I indicates an interface route. • S indicates a static route. • N indicates a NAT route. • A indicates that the route needs an ARP resolve. • E indicates an ECMP route.

To display the route summary for the current context, use the **show ip route summary** command. For example, enter:

```
host1/Admin# show ip route summary
```

Table 4-8 describes the fields in the **show ip route summary** command output.

Table 4-8 Field Description for the show ip route summary Command

Field	Description
Route Source	Source of the route. The possible value are as follows: • Connected for a route to hosts that are connected to the same network. • Static for a configured route.
Count	Number of routes that are connected or static.
Memory (bytes)	Memory consumed by the route entries.

To display IP traffic information, use the **show ip traffic** command in Exec mode. The syntax of this command is as follows:

show ip traffic

For example, enter:

```
host1/Admin# show ip traffic
```

Table 4-9 describes the fields in the **show ip traffic** command output.

Table 4-9 Field Descriptions for the show ip traffic Command Output

Field	Description
IP Statistics	
Rcvd	Total number of packets received by the ACE, number of bytes received by the ACE, number of input errors, number of packets received by the ACE with no route, and number of packets received by the ACE that had an unknown protocol.
Frags	Number of fragments that the ACE reassembled, number of fragments that the ACE could not reassemble, number of packets that the ACE fragmented, and number of packets that the ACE could not fragment.
Bcast	For IPv4, number of broadcast packets received and sent.
Mcast	Number of multicast packets received and sent.
Sent	Total packets sent, number of bytes sent, and number of packets sent with no route.
Drop	Number of packets discarded because they had no route and number of packets discarded.

Table 4-9 Field Descriptions for the show ip traffic Command Output (continued)

Field	Description
ICMP Statistics	
Rcvd	Reports statistics for the following ICMP messages received by the ACE: • Redirects • ICMP Unreachable • ICMP Echo • ICMP Echo Reply • Mask Requests • Mask Replies • Quench • Parameter • Timestamp
Sent	Reports statistics for the following ICMP messages sent by the ACE: • Redirects • ICMP Unreachable • ICMP Echo • ICMP Echo Reply • Mask Requests • Mask Replies • Quench • Timestamp • Parameter • Time Exceeded

Table 4-9 Field Descriptions for the show ip traffic Command Output (continued)

Field	Description
TCP Statistics	
Rcvd	Total number of TCP segments and errors received by the ACE.
Sent	Total number of TCP segments sent by the ACE.
UDP Statistics	
Rcvd	Total number of UDP segments, UDP errors, and segments with no port number received by the ACE.
Sent	Total number of UDP segments sent by the ACE.
ARP Statistics	
Rcvd	Number of ARP packets, errors, requests, and responses received by the ACE.
Sent	Number of ARP packets, errors, requests, and responses sent by the ACE.

The **show ip route internal** command is used for debugging purposes. The output of this command is for use by trained Cisco personnel as an aid in debugging and troubleshooting the ACE. For information on the command syntax, see the *Command Reference, Cisco ACE Application Control Engine*.

Displaying the IPv4 FIB Table Information

The forwarding information base (FIB) table contains information that the forwarding processors require to make IP forwarding decisions. This table is derived from the route and ARP tables. To display the IPv4 FIB table for the context, use the **show ip fib** command. For example, enter:

```
host1/Admin# show ip fib
```

Table 4-10 describes the fields in the **show ip fib** command output.

Table 4-10 Field Description for the show ip fib Command

Field	Description
Destination	Destination address for the route.
Interface	VLAN interface number for this entry.
EncapID	Encapsulation identifier.
Flag	Flag to identify the route type and state, as identified by one of the following codes displayed above the output information: • H indicates a host route. • I indicates interface route. • S indicates a static route. • N indicates a NAT route. • A indicates that the route needs an ARP resolve. • E indicates an ECMP route. • V indicates that the route destination matches a class map-defined virtual server.

To display a summary of the FIB table for the context, use the **show ip fib summary** command. For example, enter:

```
host1/Admin# show ip fib summary
```

Table 4-11 describes the fields in the **show ip fib summary** command output.

Table 4-11 Field Description for the show ip fib summary Command

Field	Description
Resolved routes	Number of prefixes programmed in mtrie.
Leaves, bytes	Number of mtrie leaf nodes allocated and memory consumed in bytes.

Table 4-11 Field Description for the show ip fib summary Command

Field	Description
Nodes, bytes	Number of mtrie internal nodes allocated and memory consumed in bytes.
ecmps, bytes	Number of ECMP nodes allocated and memory consumed in bytes.

The **show ip fib** command is used for debugging purposes. The output of this command is for use by trained Cisco personnel as an aid in debugging and troubleshooting the ACE. For information on the command syntax, see the *Command Reference, Cisco ACE Application Control Engine*.

CHAPTER 5

Bridging Traffic

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This chapter describes how clients and servers communicate through the ACE using either Layer 2 (L2) or Layer 3 (L3) in a VLAN configuration. When the client-side and server-side VLANs are on the same subnets, you can configure the ACE to bridge traffic on a single subnet mode.

When the client-side and server-side VLANs are on different subnets, you can configure the ACE to route the traffic. For more information, see Chapter 4, "Configuring Routes on the ACE."

In bridge mode, the ACE acts as a "bump in the wire" and is not a routed hop. No dynamic routing protocols are required.

This chapter contains the following major sections:

Guidelines and Restrictions

This topic includes the following guidelines and restrictions:

- When you configure a bridge group on an interface VLAN, the ACE automatically makes it a bridged interface. The ACE supports a maximum of two Layer 2 interface VLANs per bridge group.

  Note The ACE does not allow shared VLAN configurations on Layer 2 interfaces.

- Because L2 VLANs are not associated with an IP address, they require extended access control lists (ACLs) for controlling IP traffic. You can also optionally configure EtherType ACLs for the passing of non-IP traffic. For information on ACLs, see the *Security Guide, Cisco ACE Application Control Engine*.
- To enable the bridge-group VLANs, you must configure a bridge-group virtual interface (BVI) that is associated with a corresponding bridge group. You must configure an IP address on the BVI. This address is used as a source IP address for traffic from the ACE, for example, Address Resolution Protocol (ARP) requests or management traffic. The ACE supports 4094 BVIs per system.

  Note The ACE supports a maximum of 8192 interfaces per system that include VLANs, shared VLANs, and BVI interfaces.

- The ACE does not perform MAC address learning on a bridged interface. Instead learning is performed by ARP. Bridge lookup is based on the bridge-group identifier and destination MAC address. A bridged interface automatically sends multicast and broadcast bridged traffic to the other interface of the bridge group.
- ARP packets are always passed through an L2 interface after their verification and inspection. For information on configuring ARP on the ACE, see Chapter 7, “Configuring ARP.” Multicast and broadcast packets from the incoming interface are flooded to the other L2 interface in the bridge group.

- In bridge mode, a Layer 7 policy-map is required if source NAT is configured (see the "Example 3: Bridging Configuration with Source NAT" section on page 5-20.)
- (ACE module only) When two ACE modules are configured for redundancy to provide fault tolerance and you have **svlclc autostate** configured on the supervisor engine, the ACE can intermittently not bridge the spanning tree Bridge Protocol Data Units (BPDUs) bidirectionally after a failover. This issue can cause a delay in the failover. To avoid this issue, disable autostate on the supervisor engine using the **no svclc autostate** command.

Bridge Mode Configuration Quick Start

Table 5-1 provides a quick overview of the steps required to configure a bridge group for the ACE. Each step includes the CLI command required to complete the task.

Table 5-1 Bridge Mode Configuration Quick Start

Task and Command Example

1. If you are operating in multiple context mode, observe the CLI prompt to verify that you are operating in the desired context. Change to the correct context if necessary.
`host1/Admin# changeto C1`
`host1/C1#`
The rest of the examples in this table use the Admin context unless otherwise specified. For details about creating contexts, see the *Virtualization Guide, Cisco ACE Application Control Engine*.

2. Access configuration mode by entering the **config** command.
`host1/Admin# config`
`Enter configuration commands, one per line. End with CNTL/Z`
`host1/Admin(config)#`

Table 5-1 Bridge Mode Configuration Quick Start (continued)

Task and Command Example

3. Create a VLAN for the bridge group and access interface configuration mode by using the **interface vlan** command. For example, enter:
`host1/Admin(config)# interface vlan 2`
`host1/Admin(config-if)#`

4. Assign the VLAN to the bridge group by using the **bridge-group** command. For example, enter: `host1/Admin(config-if)# bridge-group 15`

5. Assign an ACL to the VLAN to permit traffic by using the **access-group** command. You must configure an ACL on an interface where you want to permit traffic. Otherwise, the ACE denies all traffic on the interface. For more information on extended ACLs for IP traffic or EtherType ACLs for non-IP traffic, see the *Security Guide, Cisco ACE Application Control Engine*.
The following example is an ACL that permits IP traffic:
`access-list ACL1 line 5 extended permit ip any any`
After you configure an ACL for the traffic, assign it to the VLAN. For example, to assign ACL1 for inbound traffic to the interface, enter:
`host1/Admin(config-if)# access-group input ACL1`

6. Enable the VLAN by using the **no shutdown** command. For example, enter:
`host1/Admin(config-if)# no shutdown`
`host1/Admin(config-if)# exit`

7. Configure a second VLAN for the bridge group. Repeat Steps 3 through 6.

8. Create a BVI for the bridge group and access interface configuration mode for the BVI by using the **interface bvi** command in configuration mode. For example, to create a BVI for bridge group 15, enter:
`host1/Admin(config)# interface bvi 15`
`host1/Admin(config-if)#`

Table 5-1 Bridge Mode Configuration Quick Start (continued)

Task and Command Example

9. Assign an IPv6 or an IPv4 address to a BVI by using the **ip address** command. For example, to configure an IP address and mask for a BVI, enter:
`host1/Admin(config-if)# ip address 2001:DB8:1::/64`
or
`host1/Admin(config-if)# ip address 10.0.0.81 255.0.0.0`

10. Enable a BVI by using the **no shutdown** command. For example, to enable a BVI, enter: `host1/Admin(config-if)# no shutdown`

Configuring a Bridge-Group VLAN

In bridge mode, you can configure two interface VLANs into a group and bridge packets between them. All interfaces are in one broadcast domain and packets from one VLAN are switched to the other VLAN. The ACE bridge mode supports only two L2 VLANs per bridge group. In this mode, L2 VLAN interfaces do not have configured IP addresses.

Before you create a bridge group, you must assign a VLAN to the context and access its mode to configure its attributes. Use the **interface vlan** command in configuration mode. The syntax of this command is as follows:

interface vlan *number*

The *number* argument is the VLAN number that you want to assign to the context. For example, enter:

```
host1/Admin(config)# interface vlan 2
```

To remove a VLAN, use the **no interface vlan** command. For example, enter:

```
host1/Admin(config)# no interface vlan 2
```

After you configure the VLAN, configure its attributes as described in the following topics:

Configuring a Bridge Group to the VLAN

When you configure a bridge group on the VLAN, the ACE automatically makes it bridged. To assign the VLAN to the bridge group, use the **bridge-group** command in interface configuration mode. The syntax of this command is as follows:

bridge-group *number*

The *number* argument is a number from 1 to 4094. For example, to assign bridge group 15 to the VLAN, enter:

```
host1/Admin(config-if)# bridge-group 15
```

To remove the bridge group from the VLAN, use the **no bridge-group** command. For example, enter:

```
host1/Admin(config-if)# no bridge-group
```

Assigning an ACL to the Bridge-Group VLAN

A bridge group VLAN supports extended ACLs for IP traffic and EtherType ACLs for non-IP traffic. The following is an example of an extended ACL that permits IP traffic:

```
host1/Admin(config)# access-list ACL1 line 5 extended permit ip any
any
```

When you configure access to an interface, the ACE applies it to all IP addresses configured on it.

For non-IP traffic, configure an EtherType ACL. EtherType ACLs support Ethernet V2 frames. You can configure the ACE to pass one or any of the following non-IP EtherTypes: Multiprotocol Label Switching (MPLS), Internet Protocol version 6 (IPv6), and bridge protocol data units (BDPUs).

You can permit or deny BPDUs. By default, all BPDUs are denied. The ACE receives trunk port (Cisco proprietary) BPDUs because ACE ports are trunk ports. Trunk BPDUs have VLAN information inside the payload, so the ACE modifies the payload with the outgoing VLAN if you permit BPDUs. BPDU packets are not subjected to bandwidth policing in a bridge-mode configuration.

Note If you configure failover on the ACE, you must permit BPDUs on both interfaces with an EtherType ACL to avoid bridging loops.

The following example shows an EtherType ACL that permits BDPUs:

```
host1/Admin(config)#  access-list NONIP ethertype permit bdpu
```

Note The ACE does not forward multiple spanning tree (MST) BPDUs.

For more detailed information on extended or EtherType ACLs, see the *Security Guide, Cisco ACE Application Control Engine*.

After you configure an ACL for permitting traffic, assign it to the bridge-group VLAN. To apply an ACL to the inbound or outbound direction of a VLAN, use the **access-group** command in interface configuration mode. The syntax of this command is as follows:

access-group {**input** | **output**} *acl_name*

The options and arguments are as follows:

- **input**—Specifies the inbound direction of the interface to apply the ACL.
- **output**—Specifies the outbound direction of the interface to apply the ACL. This option is not allowed for EtherType ACLs.
- *acl_name*—Identifier of an existing ACL to apply to an interface

For example, to assign ACL1 for inbound traffic to the interface, enter:

```
host1/Admin(config-if)# access-group input ACL1
```

To assign ACL1 for outbound traffic to the interface, enter:

```
host1/Admin(config-if)# access-group output ACL1
```

To remove an ACL from an interface, use the **no access-group** command. For example, enter:

```
host1/Admin(config-if)# no access-group output ACL1
```

Enabling the Interface

When you create an interface, the interface is in the shutdown state until you enable it. To enable an interface for use, use the **no shutdown** command. For example, enter:

```
host1/Admin (config-if)# no shutdown
```

To disable the VLAN, use the **shutdown** command. For example, enter:

```
host1/Admin(config-if)# shutdown
```

After you enable the bridge-group VLAN, configure a BVI to bring it into operation.

Configuring a Bridge-Group Virtual Interface

To initiate traffic, such as ARP requests, from the ACE or for management traffic, a bridge group requires an interface with an IP address on the same subnet. This interface is the BVI.

A BVI is associated with a corresponding bridge group to routed interfaces within the router but acts as a routed interface that does not support bridging. The BVI is assigned with the number of the associated bridge group. Only one BVI is supported for each bridge group. The MAC address of the BVI is the same as the addresses of the associated bridge-group interfaces. You must enable the BVI and the associated bridge-group interfaces to forward traffic.

To use a BVI to terminate management traffic, apply a management policy to the Layer 2 interface from which the management traffic is expected. To apply this policy, configure the service policy on the bridge-group interface VLAN, and then configure the management IP address to the BVI.

This section contains the following topics:

- Creating a Virtual Routed Interface for a Bridge Group
- Configuring a BVI IP Address

- Configuring an Alias IP Address
- Configuring a Peer IP Address
- Providing a BVI Description
- Enabling a BVI

Creating a Virtual Routed Interface for a Bridge Group

You can create a virtual routed interface for a bridge group by using the **interface bvi** command in configuration mode. The syntax of this command is as follows:

interface bvi *group_number*

The *group_number* argument is the bridge-group number configured on the Layer 2 VLAN interfaces.

For example, to create a BVI for bridge group 15, enter:

```
host1/Admin(config)# interface bvi 15
host1/Admin(config-if)#
```

To delete a BVI for bridge group 15, enter:

```
host1/Admin(config)# no interface bvi 15
```

Configuring a BVI IP Address

For IPv6, the ACE supports the following types of IPv6 addresses:

- Link local
- Unique local
- Global
- Multicast

For details about these IPv6 addresses, see Chapter 3, Configuring VLAN Interfaces.

For IPv4, the ACE supports only one primary IP address with a maximum of four secondary addresses per interface. It treats the secondary addresses the same as a primary address and handles IP broadcasts and ARP requests for the subnet that is assigned to the secondary address as well as the interface routes in the IP routing table.

The ACE accepts client, server, or remote access traffic on the primary and secondary addresses. When the destination for the control plane (CP)-originated packets is Layer 2 adjacent to either the primary subnet or one of the secondary subnets, the ACE uses the appropriate primary or secondary interface IP address for the destination subnet as the source IP address. For any destination that is not Layer 2 adjacent, the ACE uses the primary address as the source IP address. For packets destined to the secondary IP address, the ACE sends the response with the secondary IP address as the source address.

Note SSL probes use the primary IP address as the source address for all the destinations.

Observe the following requirements and restrictions when you assign an IP address to a BVI:

- For IPv4, you must configure a primary IP address before the interface can become active. The primary address must be active for a secondary address to be active. You can configure only one primary address per interface.
- You can configure a maximum of 10 secondary addresses per interface. The ACE has a system limit of 1024 secondary addresses.
- When you configure access to an interface, the ACE applies all IP addresses configured on the interface.

You can assign an IP address to a BVI by using the **ip address** command in interface configuration mode for the BVI.

IPv6 Syntax and Example

The syntax of this command is as follows:

ip address *ipv6_address* [*/prefix_length*] [**eui64 | link-local | unique-local** [**eui64**]]

The arguments and option are as follows:

- *ip_address*—IPv6 address and mask for the interface.

- */prefix_length*—(Optional) Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. The default is /128. If you use the optional **eui64** keyword, the prefix must be less than or equal to 64.
- **eui64**—(Optional) Specifies that the IPv6 address is in the EUI64 format and that the interface identifier (64 least significant bits (LSBs)) are randomly generated using the MAC address.
- **link-local**—(Optional) Specifies that the IPv6 address is valid only for the current link. All link local addresses have a predefined prefix of FE80::/10.
- **unique-local**—Specifies that this address is globally unique and used only for local communications within a site or organization. All unique local addresses have a predefined prefix of FC00::/7.

For example, to configure a IPv6 link local address on a BVI, enter:

```
host1/Admin(config-if)# ip address FE80:1234:ABCD::10/64 link-local
```

To delete the IP address from a BVI, enter:

```
host1/Admin(config-if)# no ip address
```

IPv4 Syntax and Example

The syntax of this command is as follows:

ip address *ip_address mask* [**secondary**]

The arguments and option are as follows:

- *ip_address mask*—IP address and mask for the interface. Enter the IP address and subnet mask in dotted-decimal notation (for example, 192.168.1.1 255.255.255.0).

  If you do not include the **secondary** option, this address becomes the primary IP address. For the BVI to be active, you must configure a primary IP address for the interface.
- **secondary**—(Optional) Configures the address as a secondary IP address allowing multiple subnets under the same interface. You can configure a maximum of four secondary addresses per BVI. The ACE has a system limit of 1024 secondary addresses.

Note The ACE has no counters specifically for traffic received or sent through secondary IP addresses. All counters are at the interface level or associated with the primary IP address.

For example, to configure an IP address and mask for a BVI, enter:

```
host1/Admin(config-if)# ip address 10.0.0.10 255.255.255.0
```

To assign a secondary IP address and mask 20.20.20.1 255.255.255.0 to a BVI, enter:

```
host1/Admin(config-if)# ip address 20.20.20.1 255.255.255.0 secondary
```

To delete the IP address from a BVI, enter:

```
host1/Admin(config-if)# no ip address
```

To remove a secondary IP address for the BVI, enter:

```
host1/Admin(config-if)# no ip address 20.20.20.1 255.255.255.0
secondary
```

Configuring an Alias IP Address

When you configure a redundant configuration with active and standby ACEs, you can configure a VLAN interface that has an IP address that is shared between the active and standby ACEs. To configure a shared address for the BVI, use the **alias** command in interface configuration mode.

IPv6 Syntax and Examples

The syntax of this command is as follows:

alias *ipv6_address* [*/prefix_length*]

The arguments are as follows:

- *ipv6_address*—IPv6 alias global-unique or alias unique-local address of the ACE. It must be in the same subnet as either the global or unique-local address. The corresponding global or unique-local address must be configured and passed DAD for the alias to be activated.

- */prefix_length*—(Optional) Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. The default is /128.

For example, to configure an IPv6 alias global address, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# alias 2001:DB8:2::/64
```

To remove an IPv6 alias global address from an interface, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no alias 2001:DB8:2::/64
```

IPv4 Syntax and Examples

The syntax of this command is as follows:

alias *ip_address mask* [**secondary**]

The arguments and option are as follows:

- *ip_address mask*—Alias IP address and subnet mask. Enter the IP address and subnet mask in dotted-decimal notation (for example, 192.168.1.1 255.255.255.0).
- **secondary**—(Optional) Configures the address as a secondary alias IP address. You can configure a maximum of four secondary addresses. The ACE has a system limit of 1024 secondary alias addresses.

  The secondary alias address becomes active only when the corresponding secondary IP address on the same subnet is configured. If you remove the secondary IP address, the secondary alias address becomes inactive.

For example, to configure an IP address and mask for a BVI, enter:

```
host1/Admin(config-if)# alias 10.0.0.11 255.255.255.0
```

To configure a secondary alias IP address, enter:

```
host1/Admin(config-if)# alias 20.20.20.2 255.255.255.0 secondary
```

To delete the alias IP address from a BVI, enter:

```
host1/Admin(config-if)# no alias 10.0.0.11 255.255.255.0
```

To delete a secondary alias IP address, enter:

```
host1/Admin(config-if)# no alias 20.20.20.2 255.255.255.0 secondary
```

Configuring a Peer IP Address

When you configure redundancy, by default, configuration mode on the standby ACE is disabled and changes on an active ACE are automatically synchronized on the standby ACE. However, interface IP addresses on the active and standby ACEs must be unique. To ensure that the addresses on the interfaces are unique, the IP address of an interface on the active ACE is automatically synchronized on the standby ACE as the peer IP address.

To configure an IP address for the interface on the standby ACE, use the **peer ip address** command in interface configuration mode. The peer IP address on the active ACE is synchronized on the standby ACE as the interface IP address.

IPv6 Syntax and Examples

The syntax of this command is as follows:

peer ip address ipv6_address [*/prefix_length*] [**eui64** | **link-local** | **unique-local** [eui64]]

The arguments and option are as follows:

- *ip_address*—IPv6 peer address and mask for the standby interface.
- */prefix_length*—(Optional) Specifies how many of the most significant bits (MSBs) of the IPv6 address are used for the network identifier. Enter a a forward slash character (/) followed by an integer from 1 to 128. The default is /128. If you use the optional **eui64** keyword, the prefix must be less than or equal to 64.
- **eui64**—(Optional) Specifies that the IPv6 address is in the EUI64 format and that the interface identifier (64 least significant bits (LSBs)) are randomly generated using the MAC address.
- **link-local**—(Optional) Specifies that the IPv6 address is valid only for the current link. All link local addresses have a predefined prefix of FE80::/10.

- **unique-local**—Specifies that this address is globally unique and used only for local communications within a site or organization. All unique local addresses have a predefined prefix of FC00::/7.

For example, to configure a IPv6 peer link local address on a BVI, enter:

```
host1/Admin(config-if)# peer ip address FE80:1234:EFGH::10/64
link-local
```

To delete the IPv6 peer link local address from a BVI, enter:

```
host1/Admin(config-if)# no peer ip address
```

IPv4 Syntax and Examples

The syntax of this command is as follows:

peer ip address *ip_address mask* [**secondary**]

The arguments and option are as follows:

- *ip_address mask*—IP address and mask for the peer ACE. Enter the IP address and subnet mask in dotted-decimal notation (for example, 192.168.1.1 255.255.255.0).
- **secondary**—(Optional) Configures the address as a secondary peer IP address. You can configure a maximum of four secondary peer addresses. The ACE has a system limit of 1024 secondary peer addresses.

Note When the destination for the control plane (CP)-originated packets is Layer 2 adjacent to either the primary subnet or one of the secondary subnets, the ACE always uses the appropriate primary or secondary interface IP address that belong to the destination subnet as the source IP address. For any destination that is not Layer 2 adjacent, the ACE uses the primary address as the source IP address. For packets destined to the secondary IP address, the ACE sends the response with the secondary IP address as the source address.

SSL probes always uses the primary IP address as the source address for all destinations.

You cannot configure secondary IP addresses on FT VLANs.

For example, to configure an IP address and mask for the peer ACE, enter:

```
host1/Admin(config-if)# peer ip address 10.0.0.12 255.255.255.0
```

To configure a secondary IP address and mask, enter:

```
host1/Admin(config-if)# peer ip address 20.20.20.3 255.255.255.0
secondary
```

To delete the IP address for the peer ACE, enter:

```
host1/Admin(config-if)# no peer ip address
```

To delete the secondary IP address for the peer ACE, enter:

```
host1/Admin(config-if)# no peer ip address 20.20.20.3 255.255.255.0
secondary
```

Providing a BVI Description

You can provide a description for the BVI by using the **description** command in interface configuration mode. The syntax of this command is as follows:

description *text*

The *text* argument is a text string with a maximum of 240 alphanumeric characters including spaces.

For example, to provide a description for the BVI, enter:

```
host1/Admin(config-if)# description BVI for Bridge Group 15
```

To delete the description, enter:

```
host1/Admin(config-if)# no description
```

Enabling a BVI

You can enable a BVI by using the **no shutdown** command in interface configuration mode. The syntax of this command is as follows:

no shutdown

Note When you enable the interface, all of its configured primary and secondary addresses are enabled. You must configure a primary IP address before you can enable the interface. The ACE does not enable an interface with only secondary addresses. When you disable an interface, all of its configured primary and secondary addresses are disabled.

For example, to enable a BVI, enter:

```
host1/Admin(config-if)# no shutdown
```

To disable the BVI, enter:

```
host1/Admin(config-if)# shutdown
```

Displaying Bridge Group or BVI Information

You can display information about a bridge-group VLAN by using the **show interface vlan** command in Exec mode. For example, enter:

```
host1/Admin# show interface vlan 15
```

To display information about a BVI, use the **show interface bvi** command in Exec mode. For example, enter:

```
host1/Admin# show interface bvi 15
```

For information about the fields in the **show interface** command, see Table 3-4 in Chapter 3, "Configuring VLAN Interfaces."

Examples of Bridging Configurations

Example 1: IPv6 Bridging Configuration

The following configuration is an example of how to configure IPv4 bridging in the ACE.

```
login timeout 0

access-list ANYONE line 10 extended permit ip anyv6 anyv6
```

```
probe tcp TCP

rserver host SERVER_01
  ip address 2001:DB8:1::/64
  inservice
rserver host SERVER_02
  ip address 2001:DB8:2::/64
  inservice
rserver host SERVER_03
  ip address 2001:DB8:3::/64
  inservice

serverfarm host REAL_SERVERS
  probe TCP
  rserver SERVER_11
    inservice
  rserver SERVER_12
    inservice
  rserver SERVER_13
    inservice

class-map match-all VIP-10
  2 match virtual-address 2001:DB8:1::/64 tcp eq www
class-map type management match-any REMOTE_ACCESS
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any

policy-map type management first-match REMOTE_MGT
  class REMOTE_ACCESS
    permit
policy-map type loadbalance first-match SLB_LOGIC
  class class-default-v6
    serverfarm REAL_SERVERS
policy-map multi-match CLIENT_VIPS
  class VIP-10
    loadbalance vip inservice
    loadbalance policy SLB_LOGIC
    loadbalance vip icmp-reply active

interface vlan 201
  description Client VLAN
  ipv6 enable
  bridge-group 200
  access-group input ANYONE
  service-policy input REMOTE_MGT
```

```
  service-policy input CLIENT_VIPS
  no shutdown
interface vlan 202
  description Server VLAN
  ipv6 enable
  bridge-group 200
  no shutdown
interface bvi 200
  description BVI interface for mgmt
  ipv6 enable
  ip address 2001:DB8:1::/64
  no shutdown

ip route ::/0 2001:DB8:1::/64
```

Example 2: IPv4 Bridging Configuration

The following configuration is an example of how to configure IPv4 bridging in the ACE.

```
login timeout 0

access-list ANYONE line 10 extended permit ip any any

probe tcp TCP

rserver host SERVER_01
  ip address 192.168.1.11
  inservice
rserver host SERVER_02
  ip address 192.168.1.12
  inservice
rserver host SERVER_03
  ip address 192.168.1.13
  inservice

serverfarm host REAL_SERVERS
  probe TCP
  rserver SERVER_11
    inservice
  rserver SERVER_12
    inservice
  rserver SERVER_13
    inservice

class-map match-all VIP-10
  2 match virtual-address 192.168.1.10 tcp eq www
class-map type management match-any REMOTE_ACCESS
```

```
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any

policy-map type management first-match REMOTE_MGT
  class REMOTE_ACCESS
    permit
policy-map type loadbalance first-match SLB_LOGIC
  class class-default
    serverfarm REAL_SERVERS
policy-map multi-match CLIENT_VIPS
  class VIP-10
    loadbalance vip inservice
    loadbalance policy SLB_LOGIC
    loadbalance vip icmp-reply active

interface vlan 201
  description Client VLAN
  bridge-group 200
  access-group input ANYONE
  service-policy input REMOTE_MGT
  service-policy input CLIENT_VIPS
  no shutdown
interface vlan 202
  description Server VLAN
  bridge-group 200
  no shutdown
interface bvi 200
  description BVI interface for mgmt
  ip address 192.168.1.2 255.255.255.0
  no shutdown

ip route 0.0.0.0 0.0.0.0 192.168.1.1
```

Example 3: Bridging Configuration with Source NAT

The following example shows why a Layer 7 policy map is required when a source NAT is configured and how to add it to a configuration.

This first example shows a configuration that lacks the required Layer 7 policy, which results in the service policy not being applied:

```
policy-map multi-match slb-vip-allow-policy-ss
    class cm-testnat2
        nat dynamic 5 vlan 105
```

In this next example, the Layer 7 policy map is added:

```
policy-map multi-match slb-vip-allow-policy-ss
    class cm-testnat2
        loadbalance vip inservice
        loadbalance policy pm007-cs04186
        nat dynamic 5 vlan 105
```

In both examples, the service policy slb-vip-allow-policy-ss is applied to one interface of a BVI as follows:

```
interface vlan 101
    bridge-group 1
    access-group input PERMIT-101
    service-policy input slb-vip-allow-policy-ss
    no shutdown
interface vlan 105
    bridge-group 1
    access-group input PERMIT-105
    nat-pool 5 192.168.219.245 192.168.219.245 netmask 255.255.255.255
pat
    no shutdown

interface bvi 1
    ip address 192.168.219.242 255.255.255.0
    alias 192.168.219.244 255.255.255.0
    peer ip address 192.168.219.243 255.255.255.0
    no shutdown
```

The L7 policy-map simply has a transparent server farm with a single real server (the next-hop gateway):

```
serverfarm host sf007-cs04186
    transparent
    rserver rs-csvlan506hsrp
        inservice
policy-map type loadbalance first-match pm007-cs04186
    class class-default
        serverfarm sf007-cs04186
```

If the configuration is changed from a BVI to a routed configuration, then the Layer 7 policy-map is not needed.

CHAPTER 6

Configuring Neighbor Discovery

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This chapter describes how the ACE uses the Neighbor Discovery (ND) protocol to manage and learn the mapping of IPv6 to Media Access Control (MAC) addresses of nodes attached to the local link. The ACE uses this information to forward and transmit IPv6 packets.

This chapter describes how to configure ND parameters and it contains the following major sections:

Overview of Neighbor Discovery

The neighbor discovery (ND) protocol enables IPv6 nodes and routers to:

- Determine the link-layer address of a neighbor on the same link
- Find neighboring routers
- Keep track of neighbors

The IPv6 ND process uses IPv6 ICMP (ICMPv6) messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), verify the reachability of a neighbor, and keep track of neighbor routers. Every IPv6 node is required to join the multicast groups corresponding to its unicast and anycast addresses.

The ACE creates an ND cache entry when it receives an ND packet or when you configure an IPv6 address on the ACE (for example, an IPv6 address for a real server, gateway, or an interface VLAN, an alias address, VIPs, and NAT pool addresses).

You can also configure static ND entries for IP to MAC translations. For details about the neighbor discovery protocol, see RFC 4861.

The IPv6 ND protocol uses the following mechanisms for its operation:

- Neighbor Solicitation
- Neighbor Advertisement
- Router Advertisement
- Duplicate Address Detection

Neighbor Solicitation

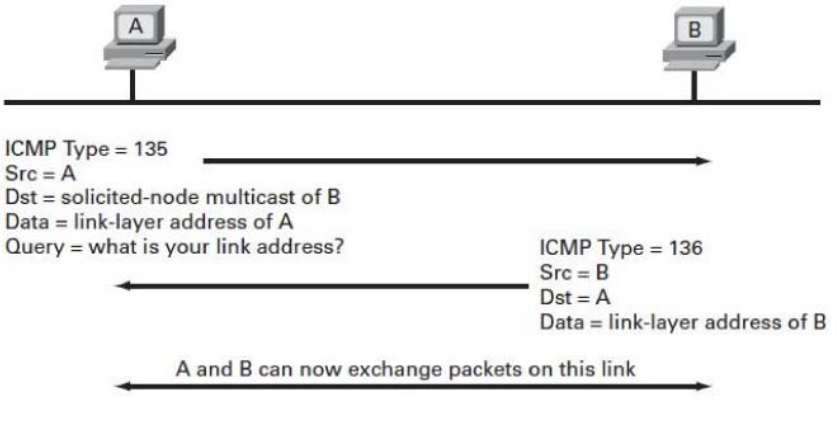
The ACE sends neighbor solicitation (NS) messages on the local link when it wants to determine the link-layer address of another node on the same local link. This function is similar to the ARP in IPv4, but avoids broadcasts used in IPv4 ARP messages, where all nodes receive unnecessary broadcast requests that do not concern them.

The ACE sends an ICMPv6 type 135 (neighbor solicitation) message to learn the corresponding link-layer address for a desired IPv6 unicast address. This request is sent to the solicited-node multicast address corresponding to the requested IPv6 unicast address.

It may be necessary for ACE to first convert a URI to an IPv6 address. If so, a naming service mechanism such as DNS must be used.

NS messages are also used to verify the reachability of a neighbor after the link-layer address of a neighbor is identified. Figure 6-1 shows how the NS message is used to determine the link-layer address of a neighbor.

Figure 6-1 Figure 14: Neighbor Discovery Message Exchange



Neighbor Advertisement

The IPv6 neighbor advertisement (NA) message is a response to the IPv6 NS message. After receiving the NS message, the destination node replies by sending an NA message on the local link with a value of 136 in the Type field of the ICMPv6 packet header. After receiving the NA, the source node and destination node can communicate.

Gratuitous NA messages are sent whenever an IPv6 address becomes active. This happens upon bootup, configuration changes, and for virtual IP (VIP) addresses, alias IP addresses, and NAT addresses following a fault-tolerant switchover.

Router Advertisement

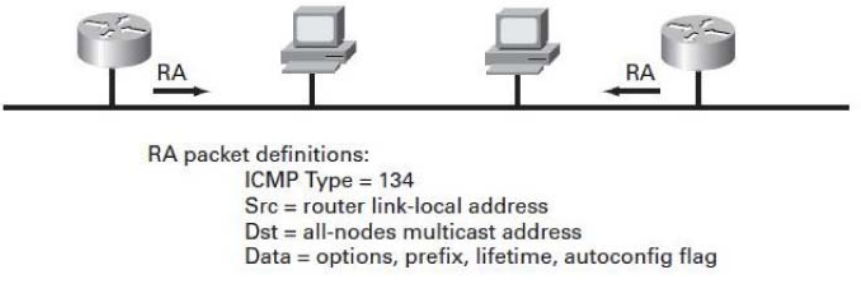
Router advertisement (RA) messages are periodically sent out on each configured interface of an IPv6 router. RAs are also sent out in response to RS messages from IPv6 nodes on the link. The RAs are sent to the all-nodes link-local multicast address (FF02::1) or the unicast IPv6 address of a node that sent the RS messages.

An RA has a value of 134 in the Type field of the ICMP packet header and contains the following information in the message:

- Whether nodes could use address autoconfiguration
- Flags to indicate the type of autoconfiguration (stateless or stateful) that can be completed
- One or more on-link IPv6 prefixes that nodes on the local link could use to automatically configure their IPv6 addresses
- Lifetime information for each prefix included in the advertisement
- Whether the router sending the advertisement should be used as a default router and, if so, the amount of time (in seconds) the router should be used as a default router
- Additional information for hosts, such as the hop limit and maximum transmission unit (MTU) a host should use in packets that it originates

The IPv6 nodes on the local link receive the RA messages and use the information to keep the information about default router and prefix lists and other configuration parameters updated. Figure 15 shows an example of an RA.

Figure 6-2 Router Advertisement



Duplicate Address Detection

Duplicate Address Detection (DAD) is an IPv6 process that a host uses to determine whether another host has the same IP address to avoid address collisions in a subnet. The originating host uses NS messages to query other nodes on the local link for their IP addresses. If the originating host receives an NA response message from another node with the same address, the originating node logs an error and cannot use the address on its interface. If the failing address is a link local address, the originating node's IPv6 interface is disabled.

Per RFC-2461 and RFC-2462, a configured IPv6 address becomes active only after the DAD process has determined that the IPv6 address is not already owned by another host. If the address is determined to be a duplicate, then it is not activated.

IPv6 Address Hierarchy

On the ACE, an IPv6 interface is considered to be operationally up after the link-local address has passed DAD. Once that happens, the ACE performs DAD on the global-unique and unique-local interface addresses, and the VIP addresses. Because the alias interface address may be in the same subnet as either the global-unique interface address or the unique-local interface address, the ACE performs DAD for the alias after DAD passes for the corresponding in-subnet global-unique or unique-local address. If a prerequisite address does not pass DAD, the ACE displays the status of that address as INACTIVE.

When you configure two ACEs as a fault-tolerant pair, only the active ACE performs DAD for virtual addresses. Therefore, the DAD status for interface alias addresses, VIP addresses, and NAT pool addresses always reflects the status on the active ACE.

If the global or unique-local address is a duplicate and the ACE is active in an FT pair, then the alias address is also disabled.

Configuring Neighbor Discovery Parameters

The ACE uses the ND protocol to find and learn the MAC addresses of other nodes that are connected to the local link. For mode details about neighbor discovery, see the "Overview of Neighbor Discovery" section.

This section contains the following subsections:

Configuring the Neighbor Solicitation Message Rate

The ACE sends neighbor solicitation messages via ICMPv6 on the local link to determine the IPv6 addresses of nearby nodes (hosts or routers). You can configure the rate at which the ACE sends these neighbor solicitation messages.

Procedure

To configure the rate at which the ACE sends NS messages for duplicate address detection (DAD) attempts, use the **ipv6 nd ns-interval** command in interface VLAN or interface BVI configuration mode. For information about configuring the number of DAD attempts, see the "Configuring the Number of Duplicate Address Detection Attempts" section.

The syntax of this command is as follows:

ipv6 nd ns-interval *interval*

The keywords and arguments are as follows:

- **ns-interval**—Indicates the frequency of the neighbor solicitation (NS) messages that are sent by the ACE
- *interval*—Specifies the frequency in milliseconds (msecs) of the NS messages that are sent by the ACE. Enter an integer from 1000 to 2147483647. The default is 1000 msecs.

For example, to configure an NS frequency of 36000 msecs, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
```

```
host1/Admin(config-if)# ipv6 nd ns-interval 36000
```

To reset the NS interval to the default value of 1000 msecs, enter the following commands:

```
host1/Admin(config)# interface VLAN 100
host1/Admin(config-if)# no ipv6 nd ns-interval 36000
```

Configuring a Static Neighbor Entry

You can configure a static ND entry that maps an IPv6 address to a Layer 2 address. The ACE stores this entry in the ND cache. To configure a static ND entry, use the **ipv6 neighbor** command in configuration mode. The syntax of this command is as follows:

ipv6 neighbor *ipv6_address mac_address*

The arguments are as follows:

- *ipv6_address*—IPv6 address of the host
- *mac_address*—Layer 2 media access control (MAC) address

For example, to configure a static ND entry, enter the following commands:

```
host1/Admin(config)# ipv6 neighbor 2001:DB8:1::80 00-0c-f1-56-98-ad
```

To remove the static ND entry, enter the following command:

```
host1/Admin(config)# no ipv6 neighbor
```

Configuring the ND Refresh Interval for Configured Host Entries

By default, the refresh interval for existing ND entries of configured hosts is 300 seconds. To configure this interval, use the **ipv6 nd interval** command in configuration mode. You configure this command for each context. The syntax of this command is as follows;

ipv6 nd interval *number*

The *number* argument specifies the time interval in seconds between NS messages for configured hosts. Enter an integer from 15 to 2073600. The default is 300 seconds (5 minutes).

For example, to configure an NS message interval of 600 seconds (10 minutes), enter the following command:

```
host1/Admin(config)# ipv6 nd interval 600
```

To reset the NS message interval to the default of 300 seconds, enter the following command;

```
host1/Admin(config)# no ipv6 nd interval 600
```

Configuring the ND Refresh Interval for Learned Host Entries

By default, the refresh interval for ND entries of learned hosts is 300 seconds. To configure this interval, use the **ipv6 nd learned-interval** command in configuration mode. You configure this command for each context. The syntax of this command is as follows:

ipv6 nd learned-interval *number*

The *number* argument specifies the time interval in seconds between NS messages for learned neighbor entries. Enter an integer from 60 to 2073600. The default is 14400 seconds (240 minutes or 4 hours).

For example, to configure a learned neighbor interval of 600 seconds (10 minutes), enter the following command:

```
host1/Admin(config)# ipv6 nd learned-interval 600
```

To reset the learned neighbor interval to the default of 300 seconds, enter the following command;

```
host1/Admin(config)# no ipv6 nd learned-interval 600
```

Configuring the Number of NS Retries

To configure the number of NS attempts before the ACE considers a host as down, use the **ipv6 nd retries** command in configuration mode. The syntax of this command is as follows:

ipv6 nd retries *number*

The *number* argument specifies the number of times that the ACE resends the NS messages before considering a host as down. Enter an integer from 1 to 15. The default is 3.

For example, to configure the ACE to resend NS messages five times before marking the host as down, enter the following command:

```
host1/Admin(config)# ipv6 nd retries 5
```

To reset the number of retries to the default value of 3, enter the following command;

```
host1/Admin(config)# no ipv6 nd retries 5
```

Disabling the Replication of Neighbor Discovery Entries

By default, the active ACE replicates ND entries to the standby in a redundant configuration. To disable the replication of ND entries, use the **ipv6 nd sync disable** command in configuration mode. The syntax of this command is as follows:

ipv6 nd sync disable

For example, to disable ND entry replication for the current context, enter the following command:

```
host1/Admin(config)# ipv6 nd sync disable
```

To reenable the replication of ND entries, enter the following command;

```
host1/Admin(config)# no ipv6 nd sync disable
```

Configuring the Neighbor Discovery Entry Replication Interval

By default, the time interval between ND synchronization messages for learned hosts is 5 seconds. To configure this time interval, use the **ipv6 nd sync-interval** command in configuration mode. The syntax of this command is as follows:

ipv6 nd sync-interval *number*

The *number* argument specifies the time interval between ND synchronization messages. Enter an integer from 1 to 3600 seconds (1 hour). The default is 5 seconds.

For example, to specify a time interval of 100 seconds, enter:

```
host1/Admin(config)# ipv6 nd sync-interval 100
```

To restore the default value of 5 seconds, enter the following command:

```
host1/Admin(config)# no ipv6 nd sync-interval
```

Configuring Router Advertisement Parameters

IPv6 routers periodically send router advertisement (RA) messages on each configured interface. Routers also send RA messages in response to router solicitation (RS) messages from hosts on the local link. RA messages use ICMPv6 type 134 in the ICMP packet header. When a host sends a router solicitation message to the ACE, it sends back an RA message to the host. For more information about router advertisement, see the "Router Advertisement" section. You can configure several RA message parameters in the ACE that affect how the ACE responds to RS messages, as described in the following sections:

- Configuring the Hop Limit in the Router Advertisement
- Configuring the Router Advertisement Interval
- Configuring the Router Advertisement Lifetime
- Suppressing Router Advertisements
- Configuring the Neighbor Reachable Time
- Configuring the Neighbor Discovery Retransmission Time
- Configuring the Managed Configuration Flag

- Configuring the Other Configuration Flag
- Configuring the Prefixes that the ACE Advertises in RA Messages

Configuring the Hop Limit in the Router Advertisement

You can specify the hop limit that the ACE's neighbors should use when originating IPv6 packets. To configure the hop limit in the IPv6 header, use the **ipv6 nd ra hop-limit** command in interface or BVI configuration mode. The syntax of this command is as follows;

ipv6 nd ra hop-limit *number*

The *number* argument specifies the number of hops that neighbors should use when they originate IPv6 packets. Enter an integer from 0 to 255. The default is 64.

For example, to configure the number of hops that neighbors should use, enter the following command:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd ra hop-limit 32
```

To reset the hop limit to the default of 64, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd ra hop-limit 32
```

Configuring the Router Advertisement Interval

To configure the rate at which the ACE sends RA messages, use the **ipv6 nd ra interval** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows:

ipv6 nd ra interval *number*

The *number* argument specifies the rate in seconds at which the ACE sends RA messages to other nodes on the local link. Enter an integer from 4 to 1800. The default is 600.

For example, to configure the ACE to send RA messages every 900 seconds (15 minutes), enter the following command:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd ra interval 900
```

To reset the interval to the default of 600 seconds (10 minutes), enter the following command:

```
host1/Admin(config-if)# no ipv6 nd ra interval
```

Configuring the Router Advertisement Lifetime

The RA lifetime is the length of time that neighboring nodes should consider the ACE as the default router before they send RS messages again. To configure the RA lifetime, use the **ipv6 nd ra lifetime** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows;

ipv6 nd ra lifetime *number*

The *number* argument specifies the length of time in seconds that the neighboring nodes should consider the ACE as the default router. Enter an integer from 0 to 9000. The default is 1800.

Note The RA lifetime should be less than or equal to the RA interval. The valid lifetime should be greater than or equal to the preferred lifetime.

For example, to configure an RA lifetime of 2400 seconds (40 minutes), enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd ra lifetime 2400
```

To reset the RA lifetime to the default of 1800 seconds (30 minutes), enter the following command:

```
host1/Admin(config-if)# no ipv6 nd ra lifetime
```

Suppressing Router Advertisements

By default, the ACE automatically responds to RS messages that it receives from neighbors with RA messages that include, for example, the network prefix. You can instruct the ACE to not respond to RS messages by using the **ipv6 nd ra suppress** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows:

ipv6 nd ra suppress

For example, to configure the ACE to not send RA messages to neighbors in response to RS messages, enter the following commands;

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd ra suppress
```

To reset the ACE behavior to the default of always sending RA messages in response to RS messages, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# no ipv6 nd ra suppress
```

Configuring the Neighbor Reachable Time

The reachable time parameter specifies the time in milliseconds during which a host considers a peer as reachable following the host's receipt of a reachability confirmation from the peer. A reachability confirmation can be an NA or NS message or any upper protocol traffic. The ACE sends the reachable time value in RA messages in response to RS messages. To configure the neighbor reachable time, use the **ipv6 nd reachable-time** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows:

ipv6 nd reachable-time *number*

The *number* argument specifies the length of time after which a node is considered reachable. Enter an integer from 0 to 3600000. The default is 0.

For example, to configure the ACE to send a reachable time value of 2000 msecs, enter the following commands;

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd reachable-time 2000
```

To restore the reachable time value to the default of 0 msecs, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd reachable-time
```

Configuring the Neighbor Discovery Retransmission Time

You can configure the time during which NS messages (including DAD) are retransmitted.The ND retransmission time is related to RA and applies to hosts. To configure the NS retransmission time, use the **ipv6 nd retransmission-time** command in interface VLAN or interface BVI configuration mode. The syntax of this command is:

ipv6 nd retransmission-time *number*

The *number* argument specifies the time in seconds during which NS messages are retransmitted. Enter an integer from 0 to 3600000. The default is 0.

For example, to configure the NS retransmission time for hosts, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd retransmission-time 1000
```

To restore the NS retransmission time value to the default of 0 msecs, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd retransmission-time
```

Configuring the Managed Configuration Flag

To instruct the ACE to notify hosts that they should use Dynamic Host Configuration Protocol (DHCP) for address configuration, use the **ipv6 nd managed-config-flag** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows:

ipv6 nd managed-config-flag

For example, to instruct the ACE to notify hosts to us e DHCP, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd managed-config-flag
```

To reset the ACE behavior to the default of not notifying hosts to use DHCP, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd managed-config-flag
```

Configuring the Other Configuration Flag

To notify hosts that they should use DHCP for nonaddress configurations, use the **ipv6 nd other-config-flag** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows:

ipv6 nd other-config-flag

For example, to instruct hosts to use DHCP for non-address configurations, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd other-config-flag
```

To reset the ACE behavior to the default of not notifying hosts to use DHCP for nonaddress configurations, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd other-config-flag
```

Configuring the Prefixes that the ACE Advertises in RA Messages

You can configure the IPv6 prefixes that the ACE advertises in router advertisement (RA) messages on the local link. You can configure a maximum of two prefixes for RA. To configure the prefixes that the ACE advertises, use the **ipv6 nd prefix** command in interface VLAN or interface BVI configuration mode. The syntax of this command is as follows:

ipv6 nd prefix *ipv6_address/prefix_length* [**at** *date month year time date month year time* | **no-advertise** | **no-autoconfig** | **off-link** | [**pref-lt** | **valid-lt** {*number* | **infinite**}]]

The keywords and arguments are as follows:

- *ipv6_address/prefix_length*—Specifies the prefix that the ACE advertises in RA messages.
- **at**—(Optional) Specifies that the IPv6 prefix expires on the date and time that follows.
- *date*—Valid lifetime expiration date. Enter an integer from 1 to 31.
- *month*—Valid lifetime expiration month. Enter the full name of the month or the three letter case-sensitive month abbreviation. For example, for January, enter January or Jan.
- *year*—Valid lifetime year of expiration. Enter the year as a four-digit integer.
- *time*—Valid lifetime expiration time. Enter the time in the hh:mm format.
- *date*—Preferred lifetime expiration date. Enter an integer from 1 to 31.
- *month*—Preferred lifetime expiration month. Enter the full name of the month or the three letter case-sensitive month abbreviation. For example, for January, enter January or Jan.
- *year*—Preferred lifetime year of expiration. Enter the year as a four-digit integer.
- *time*—Preferred lifetime expiration time. Enter the time in the hh:mm format.
- **no-advertise**—(Optional) Instructs the ACE to not advertise the prefix.
- **no-autoconfig**—(Optional) Specifies that the prefix should not be used for autoconfiguration.
- **off-link**—(Optional) Flag related to the L-bit as defined in RFC 2461. When you specify the optional **off-link** keyword, the L-bit flag is turned off, which indicates that the specified prefix should not be used for onlink determination. However, when the L-bit is enabled (the default setting), it indicates in the router advertisement messages that the specified prefix is assigned to the local link. Therefore, nodes sending traffic to addresses that contain the specified prefix consider the destination to be locally reachable on the link.

- **valid-lt** *number*—(Optional) Length of time in seconds that the prefix is valid. For the *number* argument, enter an integer from 0 to 2147483647. The default is 2592000 seconds (30 days).
- **pref-lt** *number*—(Optional) Length of time in seconds that prefix is preferred. For the *number* argument, enter an integer from 0 to 2147483647 The default is 604800 (seven days).
- **infinite**—(Optional) Specifies prefix never expires.

For example, to configure the prefixes that the ACE advertises in RA messages, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd prefix 2001:DB9:1::/64 valid-lt
3000000
```

To specify the valid expiration time and the preferred expiration time of the prefix, enter the following command:

```
host1/Admin(config-if)# ipv6 nd prefix 2001:DB9:1::/64 at 21 Jan 2019
12:12 21 Jan 2019 12:12
```

To remove the prefix from RA messages, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd prefix 2001:DB9:1::/64 valid-lt
3000000
```

Configuring Duplicate Address Detection Parameters

To prevent IPv6 address duplication, the ACE uses Duplicate Address Detection (DAD). When you configure a node interface with an IP address for the first time, the node solicits its neighbors to determine whether any other node uses the same IP address. If the address is already in use, the originating node logs an error. If the failed address is the link local address, the interface remains operationally down until the duplicate address is resolved by reconfiguring the address on the interface. For detailed information about DAD, see Chapter 2, Overview of IPv6.

Restrictions and Configuration Considerations

The ACE does not perform DAD on the following types of addresses:

- Aggregate VIPs, which are defined as those with network prefix lengths that are less than 128
- NAT pool addresses because of the potentially large number of addresses that would be involved

If redundancy is configured, DAD is performed on both the active and the standby ACEs for the link-local, global-unique, and unique-local addresses. However, only the active ACE performs DAD for the virtual addresses, which are the interface alias addresses and the VIPs. After DAD passes on the active ACE for a virtual address, the active ACE communicates the DAD status to the standby ACE. Therefore, the DAD status of PASSED or DUPLICATE for virtual addresses reflects the status on the active ACE, regardless of which ACE is queried for the address. The TENTATIVE state is not communicated from the active to the standby. Therefore, on the standby, the DAD status transitions immediately from INACTIVE to PASSED or DUPLICATE.

Note If you remove the global IPv6 address that is in the same subnet as the VIP from an interface of the active ACE in a redundant configuration where all the addresses have passed DAD, the DAD status of the VIP on the active changes from in-subnet (INACTIVE) to out-of-subnet(NA), but the VIP status on the standby remains in the PASSED state.

The ACE performs DAD only for VIPs that belong to a class map that is associated with a service policy that is active on an in-subnet interface. If a VIP is associated with an out-of-subnet interface, the VIP is installed without performing DAD. This is because the neighbor solicitation messages are multicast on the solicited-node multicast address of the target address and are not forwarded beyond the local subnet.

Configuring the Number of Duplicate Address Detection Attempts

You can configure the number of times that the ACE solicits its neighbors for duplicate address information. To set the number of duplicate address attempts, use the **ipv6 nd dad-attempts** command in interface VLAN configuration mode. The syntax of this command is as follows:

ipv6 nd dad-attempts *number*

The *number* argument specifies the number of times that the ACE sends NS messages to its neighbors on the local link for DAD. Enter an integer from 0 to 255. The default is 1.

Note The ACE uses the **ipv6 nd dad-attempts** command with the **ipv6 nd ns-interval** command to determine the number and frequency, respectively, of DAD attempts.

For example, to configure the ACE to send NS messages three times for DAD, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 nd dad-attempts 4
```

To reset the ACE behavior to the default of sending NS messages for DAD once, enter the following command:

```
host1/Admin(config-if)# no ipv6 nd dad-attempts
```

Displaying Neighbor Discovery Information

This section describes the **show** commands and output fields that are available to display neighbor discovery (ND) information and statistics. It contains the following topics:

- Displaying IPv6 Neighbors
- Displaying Additional Neighbor Discovery Information

Displaying IPv6 Neighbors

To display IPv6 neighbors, use the **show ipv6 neighbors** command in Exec mode. The syntax of this commands is as follows:

show ipv6 neighbors

For example, to display all IPv6 neighbors, enter the following command:

```
host1/Admin# show ipv6 neighbors
```

Table 6-1 describes the fields in the **show ipv6 neighbors** command output.

Table 6-1 Field Descriptions for the show ipv6 neighbors Command Output

Field	Description
Context	Name of the current context.
IPv6 Address	IPv6 address of the neighbor.
MAC Address	MAC address of the neighbor. This address is the interface MAC address if the entry is the local address of the interface. It is the VMAC address if the entry is a vserver.
NextNS(s)	Time in seconds until the next ND entry refresh.
Status	State of the neighbor: up or dn.
Use Count	Number of references to the IPv6 address from an upper layer. For example, for a VIP, if multiple **match virtual-address** statements are configured with different port numbers but the same IP address, the use count is incremented.
Interface	Name of the VLAN interface or BVI to which the neighbor is connected.

Table 6-1 Field Descriptions for the show ipv6 neighbors Command Output (continued)

Field	Description
Context	Name of the current context.
Type	Type of the neighbor entry. Possible values are: • VSERVER (if it is a VIP) • RSERVER • ALIAS (alias IPv6 address) • INTERFACE (interface IPv6 address) • GATEWAY • NAT
Encap	The encap ID is stored in the connection table and is used to fetch Layer 2 information in the data plane. Possible values are integers in the range 1 to (32K - 1).

Displaying the Duplicate Address Detection Status of VIPs

To display the operational status of VIPs with respect to duplicate address detection (DAD), use the **show service-policy** *name* **detail dad** command in Exec mode. The syntax of this command is as follows:

show service-policy *name* **detail dad**

For the *name* argument, enter the name of an existing service policy as an unquoted text string and a maximum of 64 alphanumeric characters. For details about the **show service-policy** command, see the *Server Load-Balancing Guide, Cisco ACE Application Control Engine*.

For example, to display the DAD status of IPv6 VIPs associated with the SERVICE_POLICY1 service policy, enter the following command:

```
host1/Admin# show service-policy SERVICE_POLICY1 detail dad
```

Table 6-2 describes the DAD-related fields in the **show service-policy** *name* **detail dad** command output. For a complete description of the other fields in the **show service-policy** command, see the *Server Load-Balancing Guide, Cisco ACE Application Control Engine*.

Table 6-2 Field Descriptions for the show service-policy name detail dad Command Output

Fields	Description
Policy-map	Name of the Layer 4 multimatch policy map.
Status	Current operational state of the service policy. Possible states are ACTIVE or INACTIVE.
Description	User-entered description of the policy map if any.
Interface	VLAN ID of the interface to which the policy map has been applied.
Service-policy	Unique identifier of the policy map.
class	Name of the class map associated with the service policy.
VIP address	Virtual IP address specified in the class map.
Protocol	Protocol specified in the class map.
Port	Port specified in the class map.

Table 6-2 Field Descriptions for the show service-policy name detail dad Command Output (continued)

Fields	Description
Subnet: Subnet DAD Status	
VLAN	VLAN ID of the interface where the VIP resides
In	In-subnet DAD status of the VIP. Possible values for the DAD status are: • DUPLICATE—IPv6 address is already owned by another device. The address is inactive. • INACTIVE—Address is not installed. • N/A—Address is installed and DAD was not done. DAD is not done for multiple-address VIPs (those with prefix lengths less than 128) or for out-of-subnet VIPs. In this case, the VIP immediately transitions to active. The interface addresses do not use this state. • PASSED—Address successfully passed DAD and is active. This state can occur only for in-subnet /128 VIPs. • TENTATIVE—address is installed and presently undergoing DAD. DAD is done only for single-address VIPS in the same subnet, and for interface addresses. The address is inactive when in this STATE.
Out	Out-of-subnet DAD status of the VIP. Values can be either INACTIVE or N/A. See In-subnet description.
Loadbalance	
VIP DAD state	Final VIP DAD status when there are multiple VIPs.

Displaying Additional Neighbor Discovery Information

To display additional ND information, use the **show ipv6 interface** command. For details about the syntax and output fields of this command, see the "Displaying IPv4 VLAN and BVI Information" section in Chapter 3, Configuring VLAN Interfaces.

Clearing Neighbor Discovery Learned Entries

To clear neighbor discovery learned entries, use the **clear ipv6 neighbors** command in Exec mode. The syntax of this command is as follows;

clear ipv6 neighbors [**no refresh** | *ipv6_address* [**no refresh**] | **vlan** *vlan_ID*]

The keywords and arguments are as follows:

- **no-refresh**—(Optional) Clears the ND cache entries without performing a new ND for learned entries. A new ND is performed for real server entries.
- *ipv6_address* [**no-refresh**]—(Optional) Clears the ND cache entry specified by the IPv6 address with or without a new ND.
- **vlan** *vlan_ID*—(Optional) Clears the ND cache entries associated with the specified VLAN. This option is required for link-local addresses.

If you do not specify one of the optional keywords or arguments, the ACE clears the ND cache entries and then performs a new ND on the entries.

For example, to clear all the ND cache entries without a new ND, enter the following command:

```
host1/Admin# clear ipv6 neighbors no-refresh
```

CHAPTER 7

Configuring ARP

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This chapter describes how the Address Resolution Protocol (ARP) on the ACE can manage and learn the mapping of IP to Media Access Control (MAC) information to forward and transmit packets. The ACE creates an ARP cache entry when it receives an ARP packet or you configure an IP address on the ACE (for example, an IP address for a real server, gateway, or an interface VLAN).

You can also configure static ARP entries for IP to Media Access Control (MAC) translations and ARP inspection to prevent ARP spoofing. ARP inspection ensures that an attacker cannot send an ARP response with the attacker MAC address if the correct MAC address and the associated IP address are in the static ARP table.

This chapter describes how to configure ARP parameters and enable ARP inspection, and contains the following major sections:

- Adding a Static ARP Entry
- Enabling ARP Inspection
- Configuring the ARP Retry Attempts
- Configuring the ARP Retry Interval
- Configuring the ARP Request Interval
- Enabling the Learning of MAC Addresses
- Enabling Source MAC Validation

Adding a Static ARP Entry

To add a static ARP entry in the ARP table, use the **arp** command in configuration mode or in interface configuration mode. You can create a static ARP entry at the context level. For bridged interfaces, you must configure static ARP entries in interface configuration mode.

Guidelines and Restrictions:

This topic includes the following guidelines and restrictions:

- When you enable ARP inspection, the ACE compares ARP packets with static ARP entries in the ARP table to determine what action to take. For more information, see the "Enabling ARP Inspection" section.
- The **arp** command in configuration mode allows the configuration of the multicast MAC address for a host. The ACE uses this multicast MAC address while sending packets to the host. However, the ACE does not learn the multicast MAC addresses for a host.
- The ACE supports multicast traffic that passes through it but it does not support the creation of multicast traffic.

The syntax of this command is as follows:

arp *ip_address mac_address*

The arguments are as follows:

- *ip_address*—IP address for an ARP table entry. Enter the IP address in dotted-decimal notation (for example, 172.16.56.76).
- *mac_address*—Hardware MAC address for the ARP table entry. Enter the MAC address in dotted-hexadecimal notation (for example, 00.60.97.d5.26.ab).

For example, to allow ARP responses from the router at 10.1.1.1 with the MAC address 00.02.9a.3b.94.d9, enter the following command:

```
host1/Admin(config)# arp 10.1.1.1 00.02.9a.3b.94.d9
```

To remove a static ARP entry, use the **no arp** command. For example, enter:

```
host1/Admin(config)# no arp 10.1.1.1 00.02.9a.3b.94.d9
```

Enabling ARP Inspection

ARP inspection prevents malicious users from impersonating other hosts or routers, known as ARP spoofing. ARP spoofing can enable a "man-in-the-middle" attack. For example, a host sends an ARP request to the gateway router. The gateway router responds with the gateway router MAC address.

However, the attacker sends another ARP response to the host with the attacker MAC address instead of the router MAC address. The attacker can now intercept all the host traffic before forwarding it on to the router. ARP inspection ensures that an attacker cannot send an ARP response with the attacker MAC address if the correct MAC address and the associated IP address are in the static ARP table.

ARP inspection operates only on ingress bridged interfaces. By default, ARP inspection is disabled on all interfaces, allowing all ARP packets through the ACE. When you enable ARP inspection, the ACE uses the IP address and

interface ID (ifID) of an incoming ARP packet as an index into the ARP table. The ACE then compares the MAC address of the ARP packet with the MAC address in the indexed static ARP entry in the ARP table and takes the following actions:

- If the IP address, source ifID, and MAC address match a static ARP entry, the inspection succeeds and the ACE allows the packet to pass.
- If the IP address and interface of the incoming ARP packet match a static ARP entry, but the MAC address of the packet does not match the MAC address that you configured in that static ARP entry, ARP inspection fails, the ACE drops the packet, and it increments the Inspect Failed counter regardless of whether the **flood** or **no-flood** option is configured.
- If the ARP packet does not match any static entries in the ARP table or there are no static entries in the table, then you can set the ACE to either forward the packet out all interfaces (**flood**) or to drop the packet (**no-flood**). In this case, the source IP address to MAC address mapping is new to the ACE. If you enter the **flood** option, the ACE creates a new ARP entry and marks it as LEARNED. If you enter the **no-flood** option, the ACE drops the ARP packet.

To enable ARP inspection, use the **arp inspection enable** command in configuration mode. The syntax of this command is as follows:

arp inspection enable [**flood | no-flood**]

The options are as follows:

- **flood**—Enables ARP forwarding of nonmatching ARP packets. The ACE forwards all ARP packets to all interfaces in the bridge group. This is the default setting. In the absence of a static ARP entry, this option bridges all packets. With this option, the ACE does not increment the Inspect Failed counter of the **show arp statistics** command.
- **no-flood**—Disables ARP forwarding for the interface and drops nonmatching ARP packets. In the absence of a static ARP entry, this option does not bridge any packets. With this option, the ACE does increment the Inspect Failed counter of the **show arp statistics** command.

For example, to enable ARP inspection and to drop all nonmatching ARP packets, enter:

```
host1/Admin(config)# arp inspection enable no-flood
```

To disable ARP inspection, use the **no arp inspection enable** command. For example, enter:

```
host1/Admin(config)# no arp inspection enable
```

Configuring the ARP Retry Attempts

By default, the number of ARP attempts before the ACE flags any learned and configured hosts as down is 3. To configure the number of ARP retry attempts, use the **arp retries** command in configuration mode. You configure this command per context. The syntax of this command is as follows:

arp retries *number*

The *number* argument is the number of ARP retry attempts. Enter a number from 1 to 15. The default is 3.

For example, to configure a retry attempts at 6, enter:

```
host1/Admin(config)# arp retries 6
```

To reset the number of ARP retry attempts to the default of 3, use the **no arp retries** command. For example, enter:

```
host1/Admin(config)# no arp retries
```

Configuring the ARP Retry Interval

By default, the interval when the ACE sends ARP retry attempts to any learned or configured hosts is 10 seconds. To configure this interval, use the **arp rate** command in configuration mode. You configure this command per context. The syntax of this command is as follows:

arp rate *seconds*

The *seconds* argument is the number of seconds between ARP retry attempts to hosts. Enter a number from 1 to 60. The default is 10.

For example, to configure the retry attempt interval of 15 seconds, enter:

```
host1/Admin(config)# arp rate 15
```

To reset the retry attempt interval to the default of 10 seconds, use the **no arp rate** command. For example, enter:

```
host1/Admin(config)# no arp rate
```

Configuring the ARP Request Interval

By default, the refresh interval for existing ARP entries of configured host addresses is 300 seconds. To configure this interval, use the **arp interval** command in configuration mode. You configure this command per context. The syntax of this command is as follows:

arp interval *seconds*

The *seconds* argument is the number of seconds between each ARP request sent to the host. Enter a number from 15 to 31536000. The default is 300.

Note When you change the ARP request internal for learned hosts and configured hosts, the new timeout does not take effect until the existing time is reached. If you want the new timeout to take effect immediately, enter the **clear arp** command to apply the new ARP interval (see the "Clearing ARP Learned Entries from the ARP Table" section).

For example, to configure a request period of 15 seconds, enter:

```
host1/Admin(config)# arp interval 15
```

To reset the ARP request interval to the default of 300 seconds, use the **no arp interval** command. For example, enter:

```
host1/Admin(config)# no arp interval
```

Enabling the Learning of MAC Addresses

By default, for bridged traffic, the ACE learns MAC addresses from all traffic. For routed traffic, the ACE learns MAC addresses only from ARP response packets or from packets that are destined to the ACE (for example, a ping to a VIP or a ping to a VLAN interface). To enable the ACE to learn MAC addresses from traffic

after the command has been disabled, use the **arp learned-mode enable** command in configuration mode. You configure this command per context. This command is enabled by default.

The syntax of this command is as follows:

arp learned-mode enable

For example, to enable the ACE to learn MAC addresses from traffic after the command has been disabled, enter:

```
host1/Admin(config)# arp learned-mode enable
```

To instruct the ACE to forward packets without learning the ARP information, use the **no arp learned-mode enable** command. For example, enter:

```
host1/Admin(config)# no arp learned-mode enable
```

Enabling Source MAC Validation

Source MAC validation allows you to instruct the ACE to check the source MAC address in an Ethernet header against the sender's MAC address in an ARP payload for every ARP packet received by the ACE on the specified interface. The ACE does not learn or update the ARP or MAC tables for packets with different MAC addresses. By default, source MAC validation is disabled.

Note If ARP inspection fails, then the ACE does not perform source MAC validation. For details about ARP inspection, see the "Enabling ARP Inspection" section.

To configure source MAC validation, use the **arp inspection** command in interface configuration mode. The syntax of this command is:

arp inspection validate src-mac [**flood** | **no-flood**]

The options are as follows:

- **flood**—Enables ARP forwarding for the interface and forwards ARP packets with nonmatching source MAC addresses to all interfaces in the bridge group. This is the default option when you enable source MAC validation.
- **no-flood**—Disables ARP forwarding for the interface and drops ARP packets with nonmatching source MAC addresses.

Note Regardless of whether you enter the **flood** or the **no-flood** option, if the source MAC address of the ARP packet does not match the MAC address of the Ethernet header, then the source MAC validation fails and the ACE increments the Smac-validation Failed counter of the **show arp statistics** command.

For example, to enable source MAC validation and instruct the ACE to drop ARP packets with nonmatching source MAC addresses, enter the following command:

```
host1/Admin(config-if)# arp inspection validate src-mac no-flood
```

To disable source MAC validation, enter the following command:

```
host1/Admin(config-if)# no arp inspection validate src-mac no-flood
```

Configuring the ARP Learned Interval

By default, the refresh interval for existing ARP entries for learned host addresses is 14400 seconds. To configure this interval, use the **arp learned-interval** command in configuration mode. You configure this command per context. The syntax of this command is as follows:

arp learned-interval *seconds*

The *seconds* argument is the number of seconds between ARP requests for learned addresses. Enter a number from 60 to 31536000. The default is 14400.

For example, to configure a learned interval of 800 seconds, enter:

```
host1/Admin(config)# arp learned-interval 800
```

To reset the learned interval to the default of 14,400 seconds, use the **no arp learned-interval** command. For example, enter:

```
host1/Admin(config)# no arp learned-interval
```

Disabling the Replication of ARP Entries

By default, ARP entry replication is enabled. To disable the replication of ARP entries, use the **arp sync disable** command in configuration mode.

The syntax of this command is as follows:

arp sync disable

For example, to disable the replication of ARP entries, enter:

```
host1/Admin(config)# arp sync disable
```

To reenable ARP entry replication, use the **no arp sync disable** command. For example, enter:

```
host1/Admin(config)# no arp sync disable
```

Specifying a Time Interval Between ARP Sync Messages

By default, the time interval between ARP synchronization messages for learned hosts is 5 seconds. To specify this time interval, use the **arp sync-interval** command in configuration mode.

The syntax of this command is as follows:

arp sync-interval *number*

The *number* argument defines the time interval. Enter an integer from 1 to 3600 seconds (1 hour). The default is 5 seconds.

For example, to specify a time interval of 100 seconds, enter:

```
host1/Admin(config)# arp sync-interval 100
```

To restore the default value of 5 seconds, use the **no arp sync-interval** command. For example, enter:

```
host1/Admin(config)# no arp sync-interval
```

Configuring the Rate Limit for Gratuitous ARP Packets

By default, the rate limit for gratuitous ARPs sent by the ACE is 512 packets per second. To configure this rate limit, use the **arp ratelimit** command in configuration mode. This command is available only in the Admin context. This rate limit applies to the ACE and not per context.

The syntax of this command is as follows:

arp ratelimit *number*

The *number* argument defines the rate limit as packets per second. Enter an integer from 100 to 8192. The default is 512.

Note The rate limit applies to all gratuitous ARPs sent for local addresses on new configurations, ACE reboot, and on MAC address changes.

For example, to specify a rate limit of 1000 packets per second, enter:

```
host1/Admin(config)# arp ratelimit 1000
```

To restore the default value of 512 packets per second, use the **no arp ratelimit** command. For example, enter:

```
host1/Admin(config)# no arp ratelimit
```

Displaying ARP Information

You can display ARP address mapping, statistics, and timeout intervals. For more information, see the following topics:

- Displaying IP Address-to-MAC Address Mapping
- Displaying ARP Statistics
- Displaying ARP Inspection Configuration
- Displaying ARP Timeout Values

Note The **show arp internal** command is used for debugging purposes. The output for this command is for use by trained Cisco personnel as an aid in debugging and troubleshooting the ACE. For information on the command syntax, see the *Command Reference, Cisco ACE Application Control Engine*.

Displaying IP Address-to-MAC Address Mapping

To display the current active IP address-to-MAC address mapping in the ARP table, use the **show arp** command in Exec mode. The syntax of this command is as follows:

show arp

Table 7-1 describes the fields in the **show arp** command output.

Table 7-1 Field Descriptions for the show arp Command

Field	Description
Context	Current context.
IP ADDRESS	IP address of the system for ARP mapping.
MAC-ADDRESS	MAC address of the system mapped to the IP address.
Interface	Interface name for this entry.
Type	Type of ARP entry. The possible types are LEARNED, GATEWAY, INTERFACE, VSERVER, RSERVER, and NAT.
Encap	Pointer to the adjacency entry, if any, for this host; Layer 2 and switch header rewrite information.
Next ARP(s)	Time in seconds that this dynamic ARP entry is valid.
Status	State of the system. The possible values are up or down.

For example, enter:

```
host1/admin# show arp
```

Displaying ARP Statistics

To display the ARP statistics globally or for a specified VLAN, use the **show arp statistics** command in Exec mode. The syntax of this command is as follows:

show arp statistics [**vlan** *vlan_number*]

The optional *vlan_number* argument displays the ARP statistics for the specified VLAN. Without this option, this command displays the ARP statistics for all VLAN interfaces.

Table 7-2 describes the fields in the **show arp statistics** command output.

Table 7-2 Field Descriptions for the show arp statistics Command Output

Field	Description
RX Packets	ARP packets received.
RX Errors	Number of errors on received ARP packets.
TX Packets	ARP packets transmitted.
TX Errors	Number of errors on transmitted ARP packets.
Bridged Packets	Number of bridged ARP packets.
Bridged Errors	Number of bridged errors.
Requests Recvd	ARP requests received.
Requests Sent	Number of ARP requests sent.
Response Recvd	ARP responses received.
Response Sent	Number of ARP responses sent.
Packets Dropped	Number of dropped ARP packets.
Inspect Failed	Number of packets failing ARP inspection.
Collision Detected	Number of detected collisions.
Gratuitous ARP sent	Number of gratuitous ARP packets sent.
Hosts learned	Number of hosts learned.

Table 7-2 Field Descriptions for the show arp statistics Command Output (continued)

Field	Description
Smac-validation failed	Number of times that the ACE detected a mismatch between the source MAC address in an Ethernet header and the sender's MAC address in an ARP payload of a received ARP packet.
Resolution requests	Number of resolution requests.
Encap-miss msg	Number of packets that contain no matching ARP entry; each learned ARP entry should correspond to an Encap. When a packet does not have a matching entry, the ACE considers it an Encap miss.
Pings attempted for Encap-miss msg	Number of times that the ACE recognizes that a ping attempt needs to occur when an Encap miss for a destination packet IP address not on an existing bridge-group subnet occurs.
Pings quenched for Encap-miss msg	Number of times that the ACE suppresses an effort to ping for the same destination packet IP address if the Encap miss for that address occurs repeatedly and too fast.
Pings rejected for Encap-miss msg	Number of times that the ACE rejects ping attempts for destination IP addresses when the Encap misses for that address are too many to handle. Similar to the quenched pings, these misses are unique.
Pings Encap-miss responded to	Number of actual pings sent for a missed IP address. The number of this counter should match the number of pings that were attempted for the Encap-miss msg counter.
Replication Counters	
Msg Received	Number of ARP replication messages that were received by the standby ACE.

Table 7-2 Field Descriptions for the show arp statistics Command Output (continued)

Field	Description
Hosts Replicated	Number of hosts for which ARP replication succeeded and entries were created on the standby.
Replication Failed	Number of hosts for which replication failed on the standby ACE.
Replication Ignored	Number of hosts for which replication messages were ignored on the standby, possibly because the entries are already present.

For example, enter:

```
host1/admin# show arp statistics
```

You can also display ARP traffic statistics by using the **show ip traffic** command. This command displays the number of received and sent packets, and associated errors, requests, and responses.

Displaying ARP Inspection Configuration

To display the ARP inspection configuration, use the **show arp inspection** command in Exec mode. The syntax of this command is as follows:

show arp inspection

Table 7-3 describes the fields in the **show arp inspection** command output.

Table 7-3 Field Descriptions for the show arp inspection Command

Field	Description
Context	Name of the current context.
ARP Inspection	Status of whether ARP inspection is enabled.
Flooding	Status of whether flooding is enabled.

Displaying ARP Timeout Values

To display the ARP timeout values, use the **show arp timeout** command in Exec mode. The syntax of this command is as follows:

show arp timeout

Table 7-4 describes the fields in the **show arp timeout** command output.

Table 7-4 Field Descriptions for the show arp timeout Command

Field	Description
Refresh Time	Interval in seconds between ARP requests sent to the ACE to validate the cache entry.
Learned Address	Interval in seconds when the ACE sends ARP requests for learned hosts.
Configured Address	Interval in seconds that the ACE sends ARP refresh requests for configured hosts. By default, the interval is 300 seconds.
Retry Rate	Interval in seconds when the ACE sends ARP retry attempts to hosts.
Max Retries per Host	Number of ARP attempts before the ACE flags the host as down.

Clearing ARP Learned Entries from the ARP Table

To clear the ARP learned entries from the ARP cache table, use the **clear arp** command. The syntax of this command is as follows:

clear arp [**no-refresh**]

The optional **no-refresh** keyword clears the learned ARP entries in the cache table without performing an ARP on the entries. Without this option, this command performs an ARP on the entries.

For example, to clear the ARP learned entries with a re-ARP on the entries, enter:

```
host1/Admin# clear arp
```

Clearing ARP Statistics

To clear the ARP statistics counters, use the **clear arp statistics** command. The syntax of this command is as follows:

clear arp statistics [**vlan** *number*]

The optional **vlan** *number* argument clears the statistic counters for the specified interface. Without this option, this command clears all counters for all interfaces.

For example, to clear the ARP statistics counters globally, enter:

```
host1/Admin# clear arp statistics
```

CHAPTER 8

Configuring the DHCP Relay

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted. The features described in this chapter apply to IPv4 and IPv6 unless otherwise noted.

This chapter describes how Dynamic Host Configuration Protocol (DHCP) servers provide configuration parameters to DHCP clients. DHCP supplies network settings, including the host IP address, the default gateway, and a DNS server. When DHCP clients and associated servers do not reside on the same IP network or subnet, a DHCP relay agent can transfer DHCP messages between them. The DHCP relay agent operates as the interface between DHCP clients and the server. It listens for client requests and adds vital configuration data, such as the client's link information, which is needed by the server to allocate the address for the client. When the DHCP server responds, the DHCP relay agent forwards the reply back to the DHCP client.

Note The ACE does not support DHCP relay for DHCP packets received on shared VLANs between contexts or on bridged interfaces.

This chapter contains the following major sections:

DHCP Server and Client Overview

DHCP provides a framework for passing configuration information dynamically to hosts on a TCP/IP network. A DHCP client is an Internet host using DHCP to obtain configuration parameters such as an IP address.

A DHCP relay agent is any host that forwards DHCP packets between clients and servers. Relay agents are used to forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is distinct from the normal forwarding of an IP router, where IP datagrams are switched between networks somewhat transparently. By contrast, relay agents receive DHCP messages and then generate a new DHCP message to send on another interface.

Figure 8-1 shows the basic steps that occur when a DHCPv6 client requests an IP address from a DHCPv6 server. The client, Host A, sends a DHCPSOLICIT multicast message to locate a DHCPv6 server. A relay agent forwards the packets between the DHCPv6 client and server. A DHCPv6 server offers configuration parameters (for example, an IPv6 address, a MAC address, a domain name, NTP server, and a lease for the IPv6 address) to the client in a DHCPRELAY-FORW unicast message.

Figure 8-1 DHCPv6 Request for an IPv6 Address from a DHCP Server

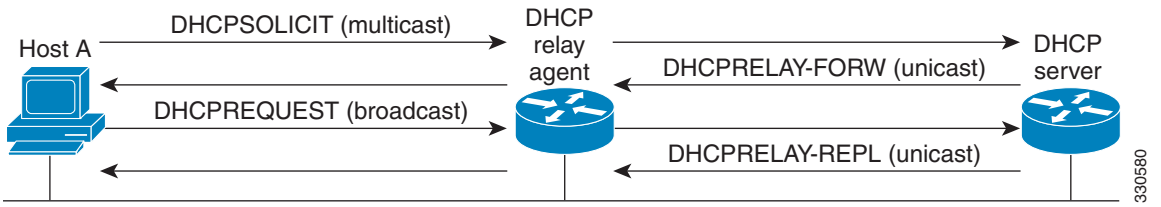
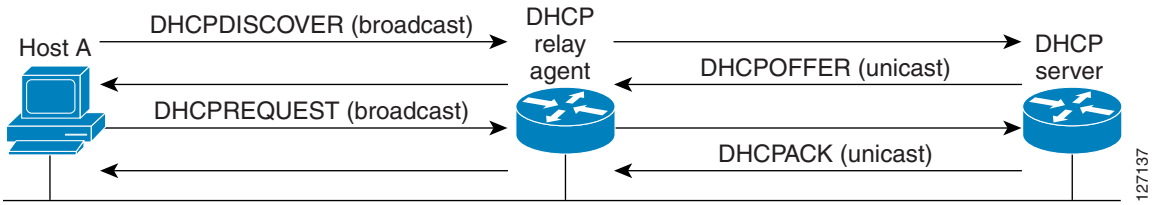


Figure 8-2 shows the basic steps that occur when a DHCPv4 client requests an IPv4 address from a DHCPv4 server. The client, Host A, sends a DHCPDISCOVER broadcast message to locate a DHCPv4 server. A relay agent forwards the packets between the DHCPv4 client and server. A DHCPv4 server offers configuration parameters (for example, an IPv4 address, a MAC address, a domain name, NTP server, and a lease for the IPv4 address) to the client in a DHCPOFFER unicast message.

Figure 8-2 DHCPv4 Request for an IPv4 Address from a DHCP Server



DHCP Relay Configuration Quick Start

Table 8-1 provides a quick overview of the steps required to configure the DHCP relay function on the ACE. Each step includes the CLI command required to complete the task. For a complete description of each feature and all the options associated with the CLI command, see the sections following Table 8-1.

Table 8-1 DHCP Relay Configuration Quick Start

Task and Command Example

1. If you are operating in multiple contexts, observe the CLI prompt to verify that you are operating in the desired context. If necessary, log directly in to, or change to, the correct context.
`host1/Admin# changeto C1`
`host1/C1#`
The rest of the examples in this table use the Admin context unless otherwise specified. For details on creating contexts, see the *Virtualization Guide, Cisco ACE Application Control Engine*.

2. Enter configuration mode by entering **config**.
`host1/Admin# config`
`Enter configuration commands, one per line. End with CNTL/Z`
`host1/Admin(config)#`

3. Enable the DHCP relay agent to accept DHCP requests from clients for the current context for IPv6 or IPv4. The default is disabled.
`host1/Admin(config)# ipv6 dhcp relay enable`
`host1/Admin(config)# ip dhcp relay enable`

Table 8-1 DHCP Relay Configuration Quick Start

Task and Command Example

4. Specify the IPv6 or IPv4 address of a DHCP server to which the DHCP relay agent forwards client requests for the current context.
`host1/Admin(config)# ipv6 dhcp relay server 2001:DB8:1::/64`
`host1/Admin(config)# ip dhcp relay server 192.168.20.1`

5. (Optional) For DHCPv6, specify a vlan interface on which the client requests are forwarded to the All_DHCP_Relay_Agents_and_Servers multicast address (FF02::1:2). `host1/Admin(config)# ipv6 dhcp relay fwd-interface vlan 100`
6. Specify a DHCPv6 server that is reachable on its link-local address on a particular VLAN. `host1/Admin(config)# ipv6 dhcp relay server fe80::250:56ff:fe90:2c fwd-interface vlan 100`
7. (Optional) For IPv4, configure a relay agent information reforwarding policy on the DHCP server to identify what the DHCP server should do if a forwarded message already contains relay information. `host1/Admin(config)# ip dhcp relay information policy replace`

8. (Optional) Save your configuration changes to flash memory.
`host1/Admin(config)# exit`
`host1/Admin# copy running-config startup-config`

Configuring the DHCP Relay Agent

This section describes how to configure the DHCP relay agent on the ACE. When you configure the ACE as a DHCP relay agent, it is responsible for forwarding the requests and responses that are negotiated between the DHCP clients and the server. By default, the DHCP relay agent is disabled. You must configure a DHCP server when you enable the DHCP relay agent.

You can configure the DHCP relay agent at both the context and VLAN interface levels of the ACE as follows:

- If you configure the DHCP relay agent at the context level, the configuration applies to all interfaces associated with the context.

- If you configure the DHCP relay agent at the VLAN interface level, the configuration applies to that particular interface only; the remaining interfaces revert to the context level configuration.

This section contains the following topics:

Enabling the DHCP Relay

You can accept DHCP requests from clients on the associated context or VLAN interface and enable the DHCP relay agent by using the **ipv6 dhcp relay enable** command (for IPv6) or the **ip dhcp relay enable** command (for IPv4). The DHCP relay starts forwarding packets to the DHCP server address specified in the **ipv6 dhcp relay server** command or the **ip dhcp relay server** command for the associated context or VLAN interface.

IPv6 Syntax

The syntax of this command is as follows:

ipv6 dhcp relay enable

IPv6 Examples

To enable the DHCP relay agent globally for all VLAN interfaces associated with a context, enter the following command:

```
host1/Admin(config)# ipv6 dhcp relay enable
```

To enable the DHCP relay agent at the VLAN interface level, enter the following command:

```
host1/Admin(config)# ipv6 dhcp relay enable
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# ipv6 dhcp relay enable
```

To disable the DHCP relay agent globally for VLAN interfaces in a context where DHCP relay is not explicitly configured, enter the following command:

```
host1/Admin(config)# no ipv6 dhcp relay enable
```

To disable the DHCP relay agent on a VLAN interface, enter the following commands:

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# no ipv6 dhcp relay enable
```

IPv4 Syntax

The syntax of this command is as follows:

ip dhcp relay enable

IPv4 Examples

To enable the DHCP relay agent for all interfaces associated with a context, enter the following command:

```
host1/Admin(config)# ip dhcp relay enable
```

For example, to enable the DHCP relay agent at the VLAN interface level, enter the following command:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# ip dhcp relay enable
```

To disable the DHCP relay agent for interfaces in a context where DHCP relay is not configured explicitly, enter the following command:

```
host1/Admin(config)# no ip dhcp relay enable
```

To disable the DHCP relay agent on a VLAN interface, enter the following commands:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# no ip dhcp relay enable
```

Specifying the DHCP Server IP Address

You can set the IP address of a DHCP server to which the DHCP relay agent forwards client requests by using either the **ipv6 dhcp relay server** command or the **ip dhcp relay server** command.

IPv6 Syntax

The syntax of this command is as follows:

ipv6 dhcp relay server *ipv6_address* [**fwd-interface vlan** *vlan_id*]

The keywords and arguments are as follows:

- *ipv6_address*—Specifies the IPv6 address of the destination DHCPv6 server
- **fwd-interface vlan** *vlan_id*—(Optional) Specifies the outgoing forwarding interface if the DHCP server address is a link-local address

IPv6 Examples

To set the IPv6 address of a DHCPv6 relay server globally for all interfaces associated with a context, enter:

```
host1/Admin(config)# ipv6 dhcp relay enable
host1/Admin(config)# ipv6 dhcp relay server 2001:DB8:1::123
```

To set the IPv6 address of a DHCP relay server at the VLAN interface level, enter:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# ipv6 dhcp relay enable
host1/Admin(config-if)# ipv6 dhcp relay server 2001:DB8:1::/64
```

To set the IPv6 address of a DHCPv6 server that is reachable on its link-local address, enter the following commands:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# ipv6 dhcp relay enable
host1/Admin(config-if)# ipv6 dhcp relay server fe80::250:56ff:fe90:2c
fwd-interface vlan 100
```

Note The ACE does not check if its EUI-64 autogenerated interface address is the same as the manually configured link-local, global unicast, or unique local address. DAD is performed only on manually configured IPv6 addresses. Therefore, do not us an EUI-64 autogenerated address for the DHCP server address.

To remove the IP address of a DHCP server from a VLAN interface, enter:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# no ipv6 dhcp relay server 2001:DB8:1::/64
```

IPv4 Syntax

The syntax of this command is as follows:

ip dhcp relay server *ip_address*

The *ip_address* argument specifies the IPv4 address of the DHCP server.

IPv4 Examples

To set the IPv4 address of a DHCP relay server on all interfaces associated with a context, enter:

```
host1/Admin(config)# ip dhcp relay enable
host1/Admin(config)# ip dhcp relay server 192.168.20.1
```

To set the IPv4 address of a DHCP relay server at the VLAN interface level, enter:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# ip dhcp relay enable
host1/Admin(config-if)# ip dhcp relay server 192.168.20.1
```

To remove the IPv4 address of a DHCP server from an interface, enter:

```
host1/Admin(config)# interface vlan 50
host1/Admin(config-if)# no ip dhcp relay server 192.168.20.1
```

Configuring a Forwarding Interface for DHCPv6 Relay

For IPv6, you can configure a forwarding VLAN interface that the ACE uses to forward all client requests on the specified VLAN interface to the All_DHCP_Relay_Agents_and_Servers address of FF02::1:2. To configure the DHCP forwarding VLAN interface, use the **ipv6 dhcp relay fwd-interface** command in either configuration mode or interface configuration mode. The syntax of this command is as follows:

ipv6 dhcp relay fwd-interface vlan *vlan_id*

The *vlan_id* argument specifies the VLAN interface number that the ACE uses to forward DHCP requests. Enter the VLAN interface number of an existing VLAN interface as an integer from 2 to 4094.

For example, to configure VLAN200 as the DHCP forwarding VLAN interface, enter the following command:

```
host1/Admin(config)# ipv6 dhcp relay fwd-interface vlan 200
```

To remove the forwarding VLAN interface from the configuration, enter the following command:

```
host1/Admin(config)# no ipv6 dhcp relay fwd-interface vlan 200
```

Configuring a Relay Agent Information Reforwarding Policy

For IPv4, you can configure the DHCP relay agent to identify the action to perform if a forwarded message already contains relay information by using the **ip dhcp relay information policy** command in configuration mode. By default, the reforwarding policy is to drop the DHCP relay packet.

Note You cannot set the relay agent information reforwarding policy at the VLAN interface level; you can only globally set this function for all interfaces associated with a context.

The syntax of this command is as follows:

ip dhcp relay information policy {**keep** | **replace**}

The keywords are as follows:

- **keep**—Indicates that existing information is left unchanged on the DHCP relay agent.
- **replace**—Indicates that existing information is overwritten on the DHCP relay agent.

For example, to set the relay agent information reforwarding policy to replace existing information for all interfaces associated with a context, enter:

```
host1/Admin(config)# ip dhcp relay information policy replace
```

To restore the default relay information policy to drop the DHCP relay packet, enter:

```
host1/Admin(config)# no ip dhcp relay information policy replace
```

Viewing DHCP Relay Configuration and Statistics

You can view configuration information and statistics collected for the DHCP relay agent by using the **show ipv6 dhcp relay** command or the **show ip dhcp relay** command for IPv4.

IPv6 DHCP Relay Show Commands

There are two show commands for IPv6 DHCP relay as follows:

- **show ipv6 dhcp relay**—Displays the state (enabled or disabled) of DHCP relay globally and at the interface level, and the DHCP server configuration
- **show ipv6 dhcp relay statistics**—Displays the count of various relayed DHCP packets (Solicit, Advertise, request, reply, and so on) and errors. The output of this command increments until you enter the **clear ip dhcp relay statistics** command.

For example, to display the configured status of the DHCP relay agent and the DHCP server configuration, enter the following command:

```
host/Admin# show ipv6 dhcp relay
```

To clear all the DHCP relay statistics, use the **clear ipv6 dhcp relay statistics** command. For example, enter the following command:

```
host1/Admin# clear ipv6 dhcp relay statistics
```

Table 8-4 describes the fields in the **show ipv6 dhcp relay** command output.

Table 8-2 Field Descriptions for the show ipv6 dhcp relay Command Output

Field	Description
Context level configuration	Configuration information of the DHCP relay agent at the context level.
Status	Operating status of the DHCP relay agent at the context level: Enabled or Disabled.
Server	IPv6 address of the DHCP server at the context level.

Table 8-2 Field Descriptions for the show ipv6 dhcp relay Command Output (continued)

Field	Description
Interface level configuration	Configuration information of the DHCP relay agent at the VLAN interface level.
VLAN	Assigned interface VLAN number.
Interface ID	Interface ID of the VLAN.
Status	Operating status of the DHCP relay agent at the VLAN interface level: Enabled or Disabled.
Server	IPv6 address of the DHCP server at the VLAN interface level.

Table 8-5 describes the fields in the **show ipv6 dhcp relay statistics** command output.

Table 8-3 Field Descriptions for the show ipv6 dhcp relay statistics Command Output

Field	Description
Context level configuration	Statistics for the DHCP relay agent at the context level.
Number of SOLICIT packets relayed	Number of SOLICIT packets forwarded to a DHCP server.
Number of REQUEST packets relayed	Number of REQUEST packets forwarded to a DHCP server.
Number of CONFIRM packets relayed	Number of CONFIRM packets forwarded to a DHCP server.
Number of RENEW packets relayed	Number of RENEW packets forwarded to a DHCP server.
Number of REBIND packets relayed	Number of REBIND packets forwarded to a DHCP server.
Number of RELEASE packets relayed	Number of RELEASE packets forwarded to a DHCP server.

Table 8-3 Field Descriptions for the show ipv6 dhcp relay statistics Command Output (continued)

Field	Description
Number of DECLINE packets relayed	Number of DECLINE packets forwarded to a DHCP server.
Number of INFO_REQUEST packets relayed	Number of INFO_REQUEST packets forwarded to a DHCP server.
Number of RELAY_FORM packets relayed	Number of RELAY_FORM packets forwarded to a DHCP server.
Number of LEASEQUERY packets relayed	Number of LEASEQUERY packets forwarded to a DHCP server.
Number of ADVERTISE packets relayed	Number of ADVERTISE packets forwarded to a DHCP server.
Number of REPLY packets relayed	Number of REPLY packets forwarded to a DHCP server.
Number of RECONFIGURE packets relayed	Number of RECONFIGURE packets forwarded to a DHCP server.
Number of RELAY_REPLY packets relayed	Number of RELAY_REPLY packets forwarded to a DHCP server.
Number of LEASEQUERY_REPLY packets relayed	Number of LEASEQUERY_REPLY packets forwarded to a DHCP server.
Number of failures while relaying	Number of failures that occurred while the DHCP relay agent forwarded packets to a DHCP server.
Interface level configuration	Statistics for the DHCP relay agent at the VLAN interface level.

IPv4 DHCP Relay Show Commands

There are three **show** commands for IPv4 DHCP relay as follows:

- **show ip dhcp relay conf**—Displays the DHCP configuration information.
- **show ip dhcp relay information policy**—Displays the relay agent information reforwarding policy status.
- **show ip dhcp relay statistics**—Displays the DHCP relay statistics. The output of this command increments until you enter the **clear ip dhcp relay statistics** command.

For example, to display the configured status of the DHCP relay information reforwarding policy, enter:

```
host/Admin# show ip dhcp relay information policy
DHCP Relay reforwarding policy configured = REPLACE
```

To clear all the DHCP relay statistics information, use the **clear ip dhcp relay statistics** command. For example, enter:

```
host1/Admin# clear ip dhcp relay statistics
```

Table 8-4 describes the fields in the **show ip dhcp relay conf** command output.

Table 8-4 Field Descriptions for the show ip dhcp relay conf Command Output

Field	Description
Context level configuration	Configuration information of the DHCP relay agent at the context level.
Status	Operating status of the DHCP relay agent at the context level: Enabled or Disabled.
Server	IP address of the DHCP server at the context level.

Table 8-4 Field Descriptions for the show ip dhcp relay conf Command Output (continued)

Field	Description
Interface level configuration	Configuration information of the DHCP relay agent at the VLAN interface level.
VLAN	Assigned interface VLAN number.
Interface ID	Interface ID for the VLAN.
Status	Operating status of the DHCP relay agent at the VLAN interface level: Enabled or Disabled.
Server	IP address of the DHCP server at the VLAN interface level.

Table 8-5 describes the fields in the **show ip dhcp relay statistics** command output.

Table 8-5 Field Descriptions for the show ip dhcp relay statistics Command Output

Field	Description
Context level configuration	Statistics for the DHCP relay agent at the context level.
Number of BOOTREQUEST packets relayed	Number of BOOTREQUEST packets forwarded to a DHCP server.
Number of DHCPDISCOVER packets relayed	Number of DHCPDISCOVER packets forwarded to a DHCP server.
Number of DHCPREQUEST packets relayed	Number of DHCPREQUEST packets forwarded to a DHCP server.
Number of DHCPDECLINE packets relayed	Number of DHCPDECLINE packets forwarded to a DHCP server.

Table 8-5 Field Descriptions for the show ip dhcp relay statistics Command Output (continued)

Field	Description
Number of DHCPRELEASE packets relayed	Number of DHCPRELEASE packets forwarded to a DHCP server.
Number of DHCPINFORM packets relayed	Number of DHCPINFORM packets forwarded to a DHCP server.
Number of BOOTREPLY packets relayed	Number of BOOTREPLY packets forwarded to a DHCP server.
Number of DHCPOFFER packets relayed	Number of DHCPOFFER packets forwarded to a DHCP server.
Number of DHCPACK packets relayed	Number of DHCPACK packets forwarded to a DHCP server.
Number of DHCPNAK packets relayed	Number of DHCPNAK packets forwarded to a DHCP server.
Number of failures while relaying	Number of failures that occurred while the DHCP relay agent forwarded packets to a DHCP server.
Interface level configuration	Statistics for the DHCP relay agent at the VLAN interface level.

APPENDIX A

IPv4 Addresses, Protocols, and Ports Reference

Note The information in this chapter applies to both the ACE module and the ACE appliance unless otherwise noted.

This appendix provides a quick reference for the following topics:

IP Addresses and Subnet Masks

This section describes how to use IP addresses in the ACE. An IP address is a 32-bit number written in dotted-decimal notation: four 8-bit fields (octets) converted from binary to decimal numbers, separated by dots. The first part of an IP address identifies the network on which the host resides, while the second part identifies the particular host on the given network. The network number field is called the network prefix. All hosts on a given network share the same network prefix but must have a unique host number. In classful IP, the class of the address determines the boundary between the network prefix and the host number.

This section contains the following topics:

Classes

IP host addresses are divided into three different address classes: Class A, Class B, and Class C. Each class fixes the boundary between the network prefix and the host number at a different point within the 32-bit address. Class D addresses are reserved for multicast IP. The class descriptions are as follows:

- Class A addresses (1.xxx.xxx.xxx through 126.xxx.xxx.xxx) use only the first octet as the network prefix.
- Class B addresses (128.0.xxx.xxx through 191.255.xxx.xxx) use the first two octets as the network prefix.
- Class C addresses (192.0.0.xxx through 223.255.255.xxx) use the first three octets as the network prefix.

Because Class A addresses have 16,777,214 host addresses and Class B addresses have 65,534 hosts, you can use subnet masking to break these huge networks into smaller subnets.

Private Networks

If you need large numbers of addresses on your network, and they do not need to be routed on the Internet, you can use private IP addresses that the Internet Assigned Numbers Authority (IANA) recommends (see RFC 1918). The following address ranges are designated as private networks that should not be advertised:

- 10.0.0.0 through 10.255.255.255
- 172.16.0.0 through 172.31.255.255
- 192.168.0.0 through 192.168.255.255

Subnet Masks

A subnet mask allows you to convert a single Class A, B, or C network into multiple networks. With a subnet mask, you can create an extended network prefix that adds bits from the host number to the network prefix. For example, a Class C network prefix always consists of the first three octets of the IP address. But a Class C extended network prefix uses part of the fourth octet as well.

Subnet masking is easy to understand if you use binary notation instead of dotted-decimal notation. The bits in the subnet mask have a one-to-one correspondence with the Internet address:

- The bits are set to 1 if the corresponding bit in the IP address is part of the extended network prefix.
- The bits are set to 0 if the bit is part of the host number.

Example 1—If you have the Class B address 129.10.0.0 and you want to use the entire third octet as part of the extended network prefix instead of the host number, you must specify a subnet mask of 11111111.11111111.11111111.00000000. This subnet mask converts the Class B address into the equivalent of a Class C address, where the host number consists of the last octet only.

Example 2—If you want to use only part of the third octet for the extended network prefix, then you must specify a subnet mask like 11111111.11111111.11111000.00000000, which uses only 5 bits of the third octet for the extended network prefix.

You can write a subnet mask as a dotted-decimal mask or as a */bits* (“slash *bits*”) mask. In Example 1, for a dotted-decimal mask, you convert each binary octet into a decimal number: 255.255.255.0. For a */bits* mask, you add the number of 1s: /24. In Example 2, the decimal number is 255.255.248.0 and the /bits is /21.

You can also combine multiple Class C networks into a larger network—or *supernet*—by using part of the third octet for the extended network prefix. An example is 192.168.0.0/20.

This section contains the following topics:

Determining the Subnet Mask

To determine the subnet mask based on the number of hosts that you want, see Table A-1.

Table A-1 Hosts, Bits, and Dotted-Decimal Masks

Hosts[1]	/Bits Mask	Dotted-Decimal Mask
16,777,216	/8	255.0.0.0 Class A Network
65,536	/16	255.255.0.0 Class B Network
32,768	/17	255.255.128.0
16,384	/18	255.255.192.0
8,192	/19	255.255.224.0
4,096	/20	255.255.240.0
2,048	/21	255.255.248.0
1,024	/22	255.255.252.0
512	/23	255.255.254.0
256	/24	255.255.255.0 Class C Network
128	/25	255.255.255.128
64	/26	255.255.255.192
32	/27	255.255.255.224
16	/28	255.255.255.240
8	/29	255.255.255.248
4	/30	255.255.255.252
Do not use	/31	255.255.255.254
1	/32	255.255.255.255 Single Host Address

1. The first and last number of a subnet are reserved, except for /32, which identifies a single host.

Determining the Address to Use with the Subnet Mask

The following sections describe how to determine the network address to use with a subnet mask for a Class C-size and a Class B-size network:

Class C-Size Network Address

For a network between 2 and 254 hosts, the fourth octet falls on a multiple of the number of host addresses, starting with 0. For example, the 8-host subnets (/29) of 192.168.0.x are as follows:

Subnet with Mask /29 (255.255.255.248)	Address Range[1]
192.168.0.0	192.168.0.0 to 192.168.0.7
192.168.0.8	192.168.0.8 to 192.168.0.15
192.168.0.16	192.168.0.16 to 192.168.0.31
…	…
192.168.0.248	192.168.0.248 to 192.168.0.255

1. The first and last address of a subnet are reserved. In the first subnet example, you cannot use 192.168.0.0 or 192.168.0.7.

Class B-Size Network Address

To determine the network address to use with the subnet mask for a network that has between 254 and 65,534 hosts, you must determine the value of the third octet for each possible extended network prefix. For example, you might want to subnet an address such as 10.1.*x*.0, where the first two octets are fixed because they are used in the extended network prefix, and the fourth octet is 0 because all bits are used for the host number.

To determine the value of the third octet, follow these steps:

Step 1 Calculate how many subnets you can make from the network by dividing 65,536 (the total number of addresses using the third and fourth octet) by the number of host addresses you want.

For example, 65,536 divided by 4096 hosts equals 16 subnets.

Therefore, there are 16 subnets of 4096 addresses each in a Class B-size network.

Step 2 Determine the multiple of the third octet value by dividing 256 (the number of values for the third octet) by the number of subnets.

In this example, 256/16 = 16.

The third octet falls on a multiple of 16, starting with 0.

Therefore, the 16 subnets of the network 10.1 are as follows:

Subnet with Mask /20 (255.255.240.0)	Address Range[1]
10.1.0.0	10.1.0.0 to 10.1.15.255
10.1.16.0	10.1.16.0 to 10.1.31.255
10.1.32.0	10.1.32.0 to 10.1.47.255
…	…
10.1.240.0	10.1.240.0 to 10.1.255.255

1. The first and last address of a subnet are reserved. In the first subnet example, you cannot use 10.1.0.0 or 10.1.15.255.

Protocols and Applications

This section describes the protocols and applications to help you configure the ACE. The ACE does not pass multicast or routing protocols in routed mode.

Possible literal values are **ah**, **eigrp**, **esp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **pcp**, **snp**, **tcp**, and **udp**. You can also specify any protocol by number.

Table A-2 lists the numeric values for the protocol literals.

Table A-2 Protocol Literal Values

Literal	Value	Description
ah	51	Authentication Header for IPv6, RFC 1826
eigrp	88	Enhanced Interior Gateway Routing Protocol
esp	50	Encapsulated Security Payload for IPv6, RFC 1827
gre	47	Generic routing encapsulation
icmp	1	Internet Control Message Protocol, RFC 792

Table A-2 Protocol Literal Values (continued)

Literal	Value	Description
igmp	2	Internet Group Management Protocol, RFC 1112
igrp	9	Interior Gateway Routing Protocol
ip	0	Internet Protocol
ipinip	4	IP-in-IP encapsulation
nos	94	Network Operating System (Novell's NetWare)
pcp	108	Payload Compression Protocol
snp	109	Sitara Networks Protocol
tcp	6	Transmission Control Protocol, RFC 793
udp	17	User Datagram Protocol, RFC 768

Protocol numbers can be viewed online at the IANA website:

http://www.iana.org/assignments/protocol-numbers

TCP and UDP Ports

Table A-3 lists the literal values and port numbers; either can be entered in ACE commands. See the following caveats:

- The ACE uses port 1521 for SQL*Net. This is the default port used by Oracle for SQL*Net. This value, however, does not agree with IANA port assignments.
- The ACE listens for Remote Authentication Dial-In User Service (RADIUS) on ports 1645 and 1646. If your RADIUS server uses the standard ports 1812 and 1813, you can configure the ACE to listen to those ports using the **aaa-server**, **radius-authport**, and **aaa-server radius-acctport** commands.
- To assign a port for Domain Name System (DNS) access, use **domain**, not **dns**. The **dns** keyword translates into the port value for **dnsix**.

Port numbers can be viewed online at the IANA website:

http://www.iana.org/assignments/port-numbers

Table A-3 Port Literal Values

Literal	Protocol	Value	Description
aol	TCP	5190	America Online
bgp	TCP	179	Border Gateway Protocol, RFC 1163
biff	UDP	512	Used by mail system to notify users that new mail is received
bootpc	UDP	68	Bootstrap Protocol Client
bootps	UDP	67	Bootstrap Protocol Server
chargen	TCP	19	Character Generator
citrix-ica	TCP	1494	Citrix Independent Computing Architecture (ICA) protocol
cmd	TCP	514	Similar to **exec** except that **cmd** has automatic authentication
ctiqbe	TCP	2748	Computer Telephony Interface Quick Buffer Encoding
daytime	TCP	13	Day time, RFC 867
discard	TCP, UDP	9	Discard
domain	TCP, UDP	53	DNS (Domain Name System)
dnsix	UDP	195	DNSIX Session Management Module Audit Redirector
echo	TCP, UDP	7	Echo
exec	TCP	512	Remote process execution
finger	TCP	79	Finger
ftp	TCP	21	File Transfer Protocol (control port)
ftp-data	TCP	20	File Transfer Protocol (data port)
gopher	TCP	70	Gopher

Table A-3 Port Literal Values (continued)

Literal	Protocol	Value	Description
https	TCP	443	Hypertext Transfer Protocol (SSL)
hostname	TCP	101	NIC Host Name Server
ident	TCP	113	Ident authentication service
imap4	TCP	143	Internet Message Access Protocol, version 4
irc	TCP	194	Internet Relay Chat protocol
isakmp	UDP	500	Internet Security Association and Key Management Protocol
kerberos	TCP, UDP	750	Kerberos
klogin	TCP	543	KLOGIN
kshell	TCP	544	Korn Shell
ldap	TCP	389	Lightweight Directory Access Protocol
ldaps	TCP	636	Lightweight Directory Access Protocol (SSL)
lpd	TCP	515	Line Printer Daemon—printer spooler
login	TCP	513	Remote login
lotusnotes	TCP	1352	IBM Lotus Notes
mobile-ip	UDP	434	MobileIP-Agent
nameserver	UDP	42	Host Name Server
netbios-ns	UDP	137	NetBIOS Name Service
netbios-dgm	UDP	138	NetBIOS Datagram Service
netbios-ssn	TCP	139	NetBIOS Session Service
nntp	TCP	119	Network News Transfer Protocol
ntp	UDP	123	Network Time Protocol
pcanywhere-status	UDP	5632	pcAnywhere status

Table A-3 Port Literal Values (continued)

Literal	Protocol	Value	Description
pcanywhere-data	TCP	5631	pcAnywhere data
pim-auto-rp	TCP, UDP	496	Protocol Independent Multicast, reverse path flooding, dense mode
pop2	TCP	109	Post Office Protocol—Version 2
pop3	TCP	110	Post Office Protocol—Version 3
pptp	TCP	1723	Point-to-Point Tunneling Protocol
radius	UDP	1645	Remote Authentication Dial-In User Service
radius-acct	UDP	1646	Remote Authentication Dial-In User Service (accounting)
rip	UDP	520	Routing Information Protocol
secureid-udp	UDP	5510	SecureID over UDP
smtp	TCP	25	Simple Mail Transport Protocol
snmp	UDP	161	Simple Network Management Protocol
snmptrap	UDP	162	Simple Network Management Protocol—Trap
sqlnet	TCP	1521	Structured Query Language Network
ssh	TCP	22	Secure Shell
sunrpc (rpc)	TCP, UDP	111	Sun Remote Procedure Call
syslog	UDP	514	System Log
tacacs	TCP, UDP	49	Terminal Access Controller Access Control System Plus

Table A-3 Port Literal Values (continued)

Literal	Protocol	Value	Description
talk	TCP, UDP	517	Talk
telnet	TCP	23	RFC 854 Telnet
tftp	UDP	69	Trivial File Transfer Protocol
time	UDP	37	Time
uucp	TCP	540	UNIX-to-UNIX Copy Program
who	UDP	513	Who
whois	TCP	43	Who Is
www	TCP	80	World Wide Web
xdmcp	UDP	177	X Display Manager Control Protocol

ICMP Types

Table A-4 lists the ICMP type numbers and names that you can enter in ACE commands.

Table A-4 ICMP Types

ICMP Number	ICMP Name
0	echo-reply
3	unreachable
4	source-quench
5	redirect
6	alternate-address
8	echo
9	router-advertisement
10	router-solicitation
11	time-exceeded

Table A-4 ICMP Types (continued)

ICMP Number	ICMP Name
12	parameter-problem
13	timestamp-request
14	timestamp-reply
15	information-request
16	information-reply
17	mask-request
18	mask-reply
31	conversion-error
32	mobile-redirect

INDEX

A

B

C

D

E

F

G

H

I

L

M

N

P

Q

R

S

T

U

V