Send document comments to nexus1k-docfeedback@cisco.com.

Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(2)

October 14, 2010

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-20451-01

Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(2)

New and Changed Information

This chapter describes the information that is either new and or was changed in this document in Release 4.0(4)SV1(2).

To find additional information, go to the following locations on Cisco.com.

- Command References
- Release Notes

The following table lists new and changed information, and where it is documented.

Feature	Description	Changed in Release	Where Documented
Layer 2 Configuration Limits	Added configuration limits for active VLANs across all VEMS, MACs over VLANs within a VEM, PVLANs across all VEMs, and physical trunks per VSM.	4.0(4)SV1(2)	Chapter 6, "Layer 2 Switching Configuration Limits"

CONTENTS

Preface

This section describes the *Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(2),* and includes the following topics.

Audience

This guide is for network administrators with the following experience and knowledge:

- An understanding of virtualization
- Using VMware tools to configure a vswitch

Note Note: Knowledge of VMware vNetwork Distributed Switch is not a required.

Recommended Reading

Before configuring the Cisco Nexus 1000V, Cisco recommends that you read and become familiar with the following documentation:

- *Cisco Nexus 1000V Getting Started Guide, Release 4.0(4)SV1(2)*
- *Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(2)*
- *Cisco VN-Link: Virtualization-Aware Networking* white paper

Send document comments to nexus1k-docfeedback@cisco.com.

Document Organization

This document is organized into the following chapters:

Chapter	Description
Chapter 1, "Overview"	Describes Layer 2 features.
Chapter 2, "Configuring the MAC Address Table"	Describes MAC address table configuration.
Chapter 3, "Configuring VLANs"	Describes how to configure a VLAN.
Chapter 4, "Configuring a Private VLAN"	Describes how to configure a private VLAN.
Chapter 5, "Configuring IGMP Snooping"	Describes how to configure IGMP Snooping.
Chapter 6, "Layer 2 Switching Configuration Limits"	Lists the layer 2 switching configuration limits.

Document Conventions

Command descriptions use these conventions:

boldface font	Commands and keywords are in boldface.
italic font	Arguments for which you supply values are in italics.
{ }	Elements in braces are required choices.
[ ]	Elements in square brackets are optional.
x | y | z	Alternative, mutually exclusive elements are 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 examples use these conventions:

`screen font`	Terminal sessions and information the device displays are in screen font.
`boldface screen font`	Information you must enter is in boldface screen font.
`italic screen font`	Arguments for which you supply values are in italic screen font.
< >	Nonprinting characters, such as passwords, are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

This document uses the following additional conventions:

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

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

Related Documentation

Cisco Nexus 1000V includes the following documents available on Cisco.com:

General Information

Cisco Nexus 1000V Release Notes, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Compatibility Information, Release 4.0(4)SV1(2)

Install and Upgrade

Cisco Nexus 1000V Software Installation Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Virtual Ethernet Module Software Installation Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Software Upgrade Guide, Release 4.0(4)SV1(2)

Configuration Guides

Cisco Nexus 1000V License Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Getting Started Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Quality of Service Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Security Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V High Availability and Redundancy Configuration Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V XML API User Guide, Release 4.0(4)SV1(2)

Programming Guide

Cisco Nexus 1000V XML API User Guide, Release 4.0(4)SV1(2)

Reference Guides

Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(2)

Cisco Nexus 1000V MIB Quick Reference

Troubleshooting and Alerts

Cisco Nexus 1000V Troubleshooting Guide, Release 4.0(4)SV1(2)

Cisco Nexus 1000V Password Recovery Guide

Cisco NX-OS System Messages Reference

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, 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

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

CHAPTER 1

Overview

The *Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(2)* provides an overview of the available Layer 2 features and how to configure them.

This chapter includes the following sections:

Information about Layer 2 Switching

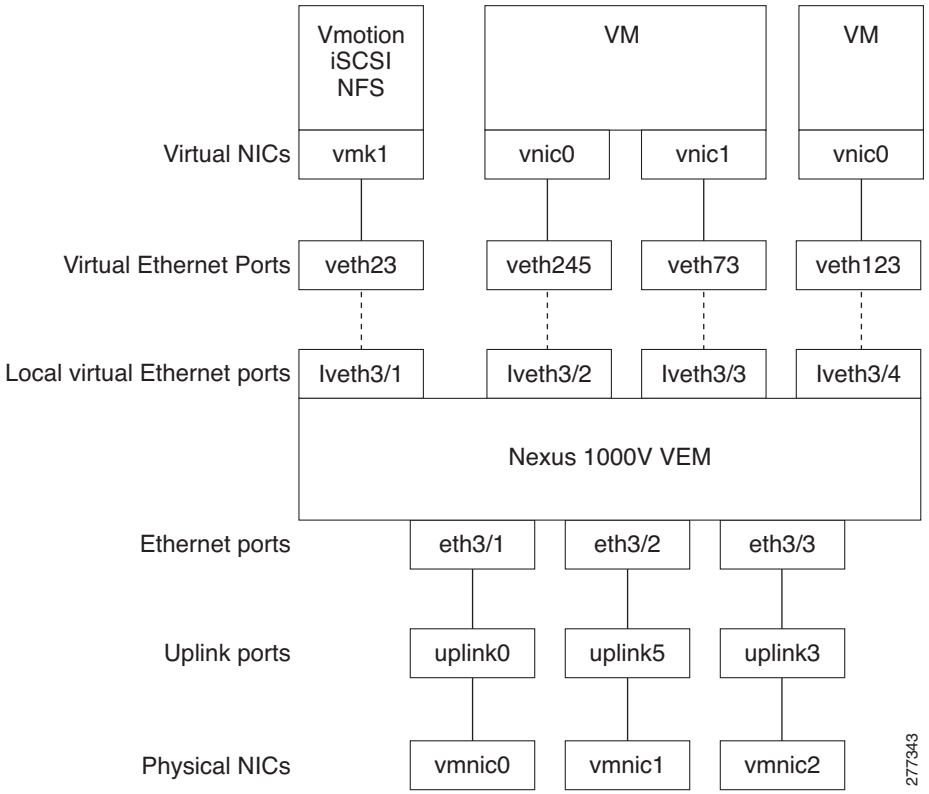
This section includes the following topics:

VEM Port Model

The Cisco Nexus 1000V differentiates the following Virtual Ethernet Module (VEM) ports:

Figure 1-1 shows how VEM ports are bound to physical and virtual VMware ports.

Figure 1-1 VEM Port View



VEM Virtual Ports

The virtual side of the VEM maps together the following three layers of ports:

- Virtual NICs, page 1-2
- Virtual Ethernet Ports, page 1-2
- Local Virtual Ethernet Ports, page 1-3

Virtual NICs

There are three types of Virtual NICs in VMware. The virtual NIC (vnic) is part of the VM, and represents the physical port of the host which is plugged into the switch. The virtual kernel NIC (vmknic) is used by the hypervisor for management, VMotion, iSCSI, NFS and other network access needed by the kernel. This interface would carry the IP address of the hypervisor itself, and is also bound to a virtual Ethernet port. The vswif (not shown) appears only in COS-based systems, and is used as the VMware management port. Each of these types maps to a veth port within Nexus1000V.

Virtual Ethernet Ports

A virtual Ethernet port (vEth) represents a port on the Cisco Nexus 1000V Distributed Virtual Switch. Cisco Nexus 1000V has a flat space of vEth ports, 0...*n*. These vEth ports are what the virtual “cable” plugs into, and are moved to the host that the VM is running on.

Virtual Ethernet ports are assigned to port groups.

Local Virtual Ethernet Ports

Each host has a number of local vEth (lvEth) ports. These ports are dynamically selected for vEth ports needed on the host.

Local vEths do not move, and are addressable by the convention, module/port number.

VEM Physical Ports

The physical side of the VEM includes the following from top to bottom:

- VMware NIC, page 1-3
- Uplink Ports, page 1-3
- Ethernet Ports, page 1-3

VMware NIC

Each physical NIC in VMware is represented by an interface called a VMNIC. The VMNIC number is allocated during VMware installation, or when a new physical NIC is installed, and remains the same for the life of the host.

Uplink Ports

Each uplink port on the host represents a physical interface. It acts a lot like an lvEth port, but since physical ports do not move between hosts, the mapping is 1:1 between an uplink port and a VMNIC.

Ethernet Ports

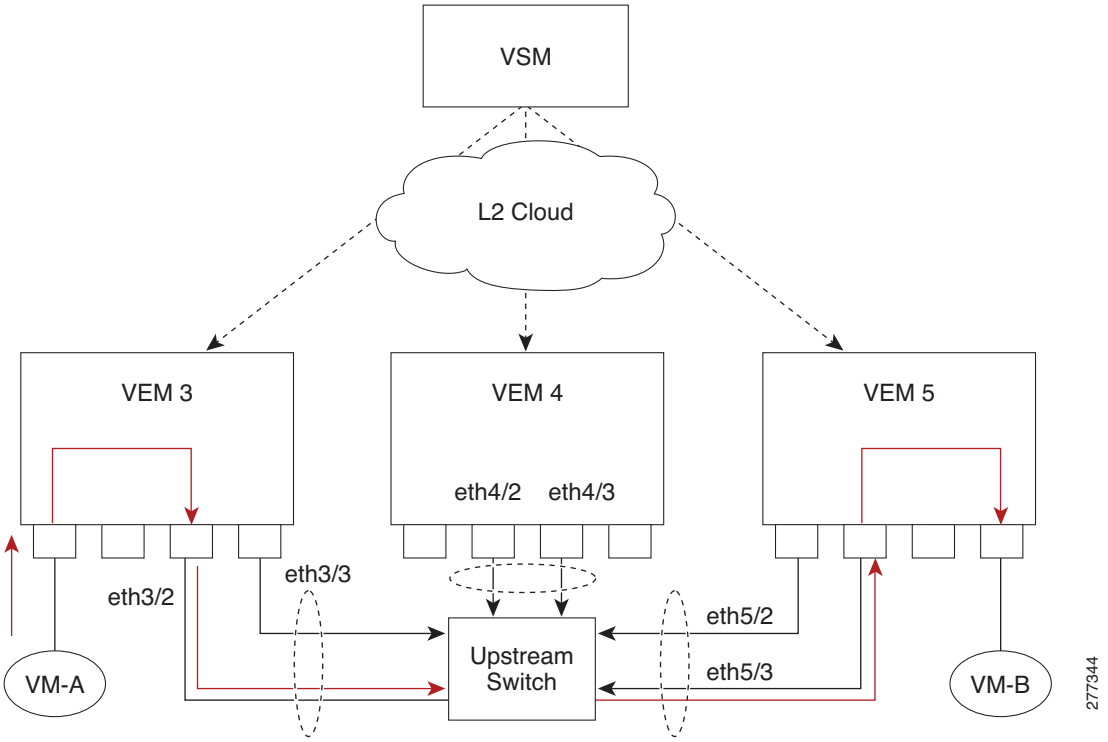
Each physical port added to Cisco Nexus 1000V appears as a physical Ethernet port, just as it would on a hardware-based switch.

Note The uplink ports are handled entirely by VMware, and are used to associate port configuration with VMNICs. There is no fixed relationship between the uplink number and VMNIC number, and these can be different on different hosts, and can change throughout the life of the host. On the VSM, the ethernet interface number, for example, ethernet 2/4, is derived from the VMNIC number, not the uplink number.

VSM Port Model

Figure 1-2 shows the VSM view of the network.

Figure 1-2 VSM View



The Virtual Supervisor Module (VEM) has the following ports or interfaces:

- Virtual Ethernet Interfaces, page 1-4
- Physical Ethernet Interfaces, page 1-4
- Port Channel Interfaces, page 1-5

Virtual Ethernet Interfaces

Virtual Ethernet interfaces (vEths) can be associated with any of the following:

- A virtual machine VNIC on the ESX host
- A virtual machine kernel NIC on the ESX host
- A virtual switch interface on an ESX COS host

Physical Ethernet Interfaces

Physical Ethernet interfaces (Eths) correspond to the physical NICs on the ESX host.

Port Channel Interfaces

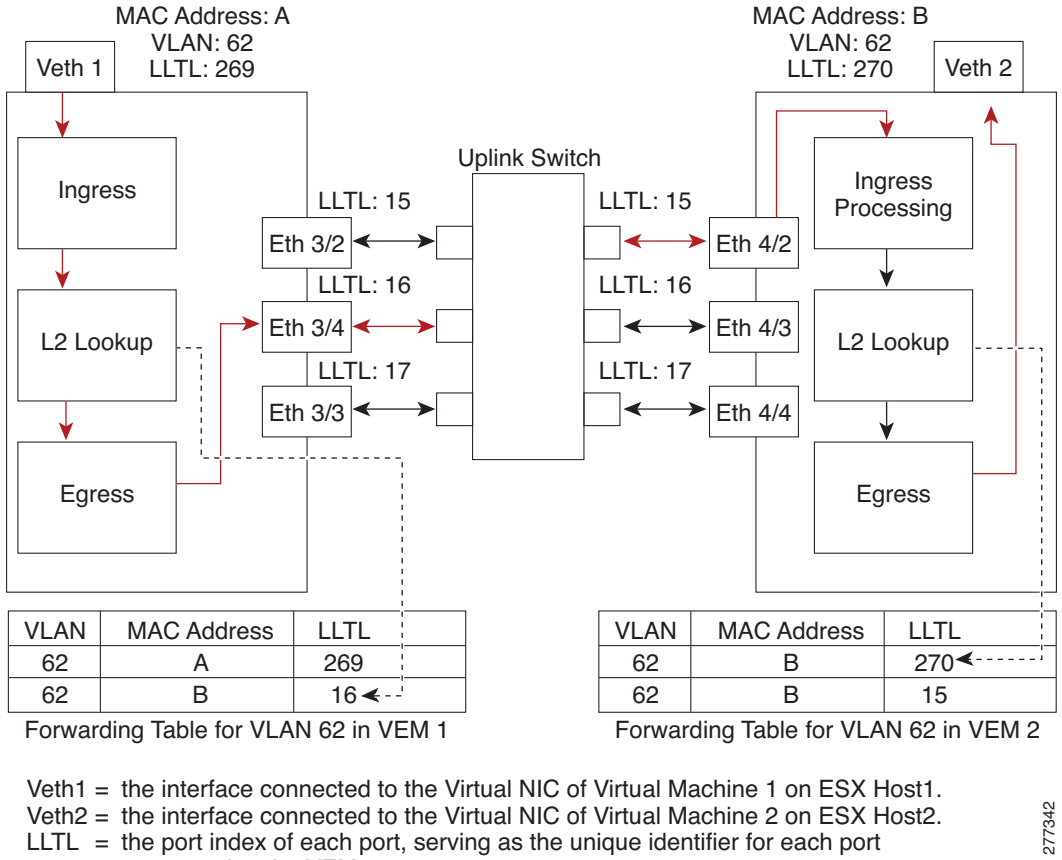
The physical NICs of an ESX host can be bundled into a logical interface called a port channel interface.

Switching Traffic Between VEMs

Each VEM attached to the VSM forwards traffic to and from the ESX server as an independent and intelligent line card. Each VLAN uses its forwarding table to learn and store MAC addresses for ports connected to the VEM.

Figure 1-3 shows the traffic flow between two VMs on different VEMs.

Figure 1-3 Traffic Flow Between VEMs



Layer 2 Ethernet Switching

The congestion related to high bandwidth and large numbers of users can be solved by assigning each device (for example, a server) to its own 10-, 100-, 1000-Mbps, or 10-Gigabit collision domain. Because each LAN port connects to a separate Ethernet collision domain, servers in a switched environment realize full bandwidth access.

Full duplex allows two stations to transmit and receive at the same time. This is unlike 10/100-Mbps Ethernet, which usually operates in half-duplex mode, so that stations can either receive or transmit but not both. When packets can flow in both directions simultaneously, the effective Ethernet bandwidth doubles. 1/10-Gigabit Ethernet operates in full-duplex only.

Each LAN port can connect to a single workstation or server or to another device through which workstations or servers connect to the network.

To reduce signal degradation, each LAN port is considered to be an individual segment. When stations connected to different LAN ports need to communicate, frames are forwarded from one LAN port to the other at wire speed to ensure full bandwidth for each session.

MAC Address Tables

To switch frames between LAN ports efficiently, a MAC addres table is maintained. The MAC address of the sending network is associated with the LAN port on which it was received. For more information about MAC address tables, see Chapter 2, “Configuring the MAC Address Table.”

VLANs

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes of physical LANs, but you can group end stations even if they are not physically located on the same LAN segment.

Any switchport can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in that VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a bridge or a router.

All ports, including the management port, are assigned to the default VLAN (VLAN1) when the device first comes up.

The devices support 4094 VLANs in accordance with the IEEE 802.1Q standard. These VLANs are organized into several ranges for different uses. Some of these VLANs are reserved for internal use by the device and are not available for configuration

Note Inter-Switch Link (ISL) trunking is not supported by the Cisco Nexus 1000V.

See Chapter 3, “Configuring VLANs” for complete information on configuring VLANs.

Private VLANs

Private VLANs (PVLANs) are used to segregate Layer 2 ISP traffic and convey it to a single router interface. PVLANs achieve device isolation by applying Layer 2 forwarding constraints that allow end devices to share the same IP subnet while being Layer 2 isolated. In turn, the use of larger subnets reduces address management overhead. For more information, see Chapter 4, “Configuring a Private VLAN.”

IGMP Snooping

The Internet Group Management Protocol (IGMP) snooping software examines Layer 2 IP multicast traffic within a VLAN to discover the ports where interested receivers reside. Using the port information, IGMP snooping can reduce bandwidth consumption in a multi-access LAN environment to avoid flooding the entire VLAN. The IGMP snooping feature tracks which ports are attached to multicast-capable routers to help the routers forward IGMP membership reports. The IGMP snooping software responds to topology change notifications. By default, IGMP snooping is enabled on the device. For more information, see Chapter 5, "Configuring IGMP Snooping."

Related Topics

The following documents contain related information:

- *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)*
- *Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(2)*
- *Cisco Nexus 1000V Security Configuration Guide, Release 4.0(4)SV1(2)*
- *Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(2)*

CHAPTER 2

Configuring the MAC Address Table

Note For information about creating interfaces, see the document, *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2).*

This chapter includes the following topics:

Information About the MAC Address Table

Layer 2 ports correlate the MAC address on a packet with the Layer 2 port information for that packet using the MAC address table. A MAC address table is built using the MAC source addresses of the frames received. When a frame is received for a MAC destination address not listed in the address table, the frame is flooded to all LAN ports of the same VLAN with the exception of the port that received the frame. When the destination station replies, the relevant MAC source addresses and port IDs are added to the address table. Then subsequent frames are forwarded to a single LAN port without flooding all LAN ports.

You can configure MAC addresses, which are called static MAC addresses, to statically point to specified interfaces on the device. These static MAC addresses override any dynamically learned MAC addresses on those interfaces. You cannot configure broadcast or multicast addresses as static MAC addresses. The static MAC entries are retained across reboots.

The address table can store up to 100 address entries. An aging timer triggers removal of addresses from the table when they remain inactive for 300 seconds. The aging timer can be configured on a global basis but not per VLAN. The aging timer is configurable on a global basis but not on a per VLAN basis.

You can configure the length of time an entry remains in the MAC address table, clearing the table, and so forth.

Guidelines and Limitations

Keep in mind the following guidelines for configuring MAC addresses:

- The forwarding table for each VLAN in a VEM can store up to 1024 MAC addresses.
- Static MAC address entries cannot be configured on an interfaces where port security is enabled. Instead, use the following command in Interface Configuration mode:

  ```
  n1000v(config-if)# switchport port-security mac address xxxx.yyyy.zzzz [vlan vlan id].
  ```

  For more information, see the *Cisco Nexus 1000V Security Configuration Guide, Release 4.0(4)SV1(2)*.
- The NX-OS software explicitly prohibits configuring port security on ports that have a static MAC addresses configured on the following VLANs:
  - the access VLAN of an access port
  - the native VLAN of a trunk port

Default Settings

Table 2-1 lists the default setting for the MAC address aging time.

Table 2-1 Default MAC Address Aging Time

Parameters	Default
Aging time	300 seconds

Configuring the MAC Address Table

This section includes the following procedures for configuring the MAC address table:

Configuring a Static MAC Address

Use this procedure to configure a MAC address to statically point to a specific interface.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.

- You cannot configure broadcast or multicast addresses as static MAC addresses.
- Static MAC addresses override dynamically-learned MAC addresses on an interface.

Note Be aware that the NX-OS commands may differ from those used in Cisco IOS.

SUMMARY STEPS

1. **config t**
2. **mac address-table static** *mac address* **vlan** *vlan-id* {[**drop** | **interface** {*type number* | | **port-channel** *number*]}
3. **exit**
4. **show mac address-table static**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you into CLI Global Configuration mode.
Step 2	**mac address-table static** *mac_address* **vlan** *vlan-id* {[**drop** | **interface** {*type number*} | **port-channel** *number*]} **Example:** n1000v(config)# mac address-table static 12ab.47dd.ff89 vlan 3 interface ethernet 2/1	Adds a static MAC address in the Layer 2 MAC address table and saves it in the running configuration. Interface can be specified as either of the following: • ethernet *slot/port* • veth *number*
Step 3	**exit** **Example:** n1000v(config)# exit n1000v#	Exits Global Configuration mode and returns you to EXEC mode.
Step 4	**show mac address static** **Example:** n1000v# show mac address static	(Optional) Displays the static MAC addresses.
Step 5	**copy running-config startup-config** **Example:** n1000v# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example:
```
n1000v# config t
n1000v(config)# mac address static 12ab.47dd.ff89 vlan 3 interface ethernet 2/1
n1000v# show mac address static
```

```
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC
        age - seconds since last seen
   VLAN     MAC Address      Type      age     Secure  NTFY   Ports
---------+-----------------+--------+---------+------+------+----------------
G     -    12ab.47dd.ff89    static       -     False  False  eth2/1

n1000v#
```

Note The Cisco Nexus 1000VMAC address table does not display multicast MAC addresses.

Configuring the Aging Time

Use this procedure to configure the amount of time that a packet source MAC address and port on which it was learned remain in the MAC table containing the Layer 2 information.

Note You can also configure MAC aging time in interface configuration mode or VLAN configuration mode.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.

Note Be aware that the Cisco NX-OS commands may differ from those used in Cisco IOS.

SUMMARY STEPS

1. **config t**
2. **mac address-table aging-time** *seconds*
3. **exit**
4. **show mac address-table**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you into CLI Global Configuration mode.
Step 2	**mac address-table aging-time** *seconds* **Example:** n1000v(config)# mac address-table aging-time 600	Specifies and saves in the running configuration the amount of time that will elapse before an entry in the Layer 2 MAC address table is discarded. Allowable entries: • 120 to 918000 seconds (default is 300) • If you specify zero (0), MAC aging is disabled.
Step 3	**exit** **Example:** n1000v(config)# exit n1000v#	Exits Global Configuration mode and returns you to EXEC mode.
Step 4	**show mac address-table aging-time** **Example:** n1000v# show mac address-table aging-time	(Optional) Displays the aging time in the MAC address table.
Step 5	**copy running-config startup-config** **Example:** n1000v# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example:
```
n1000v# config t
n1000v(config)# mac address-table aging-time 600
n1000v(config)#
```

Clearing Dynamic Addresses from the MAC Address Table

Use this procedure to clear all dynamic Layer 2 entries from the MAC address table.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.

Note Be aware that the NX-OS commands may differ from those used in Cisco IOS.

SUMMARY STEPS

1. **clear mac address-table dynamic** {**address** *mac_addr*} {**interface** {*type number]* | **port-channel** *number* | **vlan** *vlan_id*}
2. **show mac address-table**

DETAILED STEPS

	Command	Purpose
Step 1	**clear mac address-table dynamic** {**address** *mac_addr*} {**interface [***type number***]** | **port-channel** *number*} {**vlan** *vlan_id*}	Clears the dynamic address entries from the Layer 2 MAC address table. Interface can be specified as either of the following: • ethernet *slot/port* • veth *number*

	Example:
	`n1000v# clear mac address-table dynamic`
	`n1000v#`
		This example clears the entire MAC address table of all dynamic entries.

	Example:
	`n1000v# clear mac address-table dynamic vlan 5`
	`n1000v#`
		This example clears only those dynamic MAC addresses learned on VLAN 5 from the MAC address table.

Step 2	**show mac address-table** **Example:** `n1000v# show mac address-table`	(Optional) Displays the MAC address table.

Verifying the Configuration

Use the following commands to display and verify the Layer 2 MAC address configuration.

Command	Purpose
show mac address-table	Displays the MAC address table.
show mac address-table static	Displays information about the MAC address table static entries.
show mac address-table aging-time	Displays the aging time in the MAC address table.
show interface [<interface>] mac	Displays the MAC addresses and the burn-in MAC address for an interface.

Note The Cisco Nexus 1000VMAC address table does not display multicast MAC addresses.

Example Configuration for the MAC Address Table

The following example shows how to add a static MAC address:

```
n1000v# configure terminal
n1000v(config)# mac address-table static 0000.0000.1234 vlan 10 interface ethernet 2/15
n1000v(config)# mac address-table aging-time 120
```

Additional References

For additional information related to implementing Layer 2 switching, see the following sections:

- Related Documents, page 2-7
- Standards, page 2-7

Related Documents

Related Topic	Document Title
Interfaces	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)*
Port-Profiles	*Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(2)*
System management	*Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(2)*
Release Notes	*Cisco Nexus 1000V Release Notes, Release 4.0(4)SV1(2)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(2)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for the MAC Address Table

This section provides the MAC address table release history.

Table 2-2

Feature Name	Releases	Feature Information
MAC Address Tables	4.0(4)SV1(1)	This feature was introduced.

CHAPTER 3

Configuring VLANs

This section describes how to configure a VLAN, and includes the following topics:

Note For information about private VLANs, see Chapter 4, "Configuring a Private VLAN."

Information About VLANs

Physical NICs are always assigned as trunk ports, which transmit either VLAN tagged or untagged packets. A vswitch can have the following VLAN configurations:

Configuration	Description
External switch tagging (EST)	Physical NICs are untagged and all VNICs are access ports. EST is enabled by default and is used when the VLAN for the VNIC is set to 0 or left blank.
Virtual switch tagging (VST)	All physical NIC ports are tagged and VNICs are access ports. VST is enabled whenever the VNIC's VLAN is set to any value between 1 and 4094 inclusive.
Virtual machine guest tagging (VGT)	All physical NIC ports are tagged. VNICs are trunk ports. To configure VGT, the VLAN is set to 4095 on the VNIC connected to the virtual machine.

Physical ports are always trunk ports by default. The virtual machine interfaces can be either access ports or trunk ports. If a VEthernet interface is set as a trunk port, the VLAN is 4095.

VEthernet interfaces assigned to specific VLANs are tagged with the VLAN when transmitted. A VEthernet interface that is not assigned to a specific VLAN, or assigned to VLAN 0, are transmitted as untagged on the physical NIC interfaces. On the transmit side, this is equivalent to the native VLAN available in Cisco switches. When the VLAN is not specified, it is assumed to be 0.

Table 3-1 summarizes the actions taken on packets recieved by the virtual ethernet module (VEM) based on VLAN tagging.

Table 3-1 VEM Action on VLAN Tagging

Port Type	Packet received	Action
Access	Tagged	The packet is dropped.
Access	Untagged	VEM adds access VLAN to the packet.
Trunk	Tagged	No action is taken on the packet.
Trunk	Untagged	VEM adds native VLAN tag to packet.

Guidelines and Limitations

This section includes the following topics:

Maximum Allowed VLANs and MAC Addresses per VLAN

Table 3-2 lists the Cisco Nexus 1000V VLAN and MAC address limitations.

Table 3-2 Allowed VLANs and MAC Addresses per VLAN

Feature	Maximum Limit
Number of active VLANs	512
MAC addresses per VLAN within a VEM	1024

VLAN Numbering

In accordance with the IEEE 802.1Q standard, up to 4094 VLANs (numbered 1-4094) are supported in Cisco Nexus 1000V, and are organized as shown in Table 3-3.

Table 3-3 Cisco Nexus 1000V VLAN Numbering

VLANs Numbers	Range	Usage
1	Normal	Cisco Nexus 1000V default. You can use this VLAN, but you cannot modify or delete it.
2–1005	Normal	You can create, use, modify, and delete these VLANs.

Table 3-3 Cisco Nexus 1000V VLAN Numbering (continued)

VLANs Numbers	Range	Usage
1006-4094	Extended	You can create, name, and use these VLANs. You cannot change the following parameters: • State is always active. • VLAN is always enabled. You cannot shut down these VLANs. **Note** The extended system ID is always automatically enabled.
3968-4047 and 4094	Internally allocated	You cannot use, create, delete, or modify these VLANs. You can display these VLANs. Cisco Nexus 1000V allocates these 80 VLANs, plus VLAN 4094, for features, like diagnostics, that use internal VLANs for their operation.

Note For information about diagnostics, see the document, *Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(2).*

Default Settings

Table 3-4 lists the VLAN default settings.

Table 3-4 Default Private VLAN Setting

Parameters	Default
VLAN assignment for all interfaces and all ports configured as switchports	VLAN 1
VLAN name	VLANxxxx where xxxx represent four numeric digits (including leading zeroes) equal to the VLAN ID number.
Shut state	No shutdown
Operational state	Active
External switch tagging (EST)	Enabled
Physical ports	Trunk ports

Configuring a VLAN

This section includes the following procedures for configuring a VLAN:

- Creating a VLAN, page 3-4
- Configuring VLAN Characteristics, page 3-6

Creating a VLAN

Use this procedure to do one of the following:

- Create a single VLAN that does not already exist.
- Create a range of VLANs that do not already exist.
- Delete an existing VLAN.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

Note All interfaces and all ports configured as switchports are in VLAN 1 by default.

Note Be aware that the NX-OS commands may differ from those used in Cisco IOS.

- You are logged in to the CLI in EXEC mode.
- VLAN characteristics are configured in the VLAN submode. To configure a VLAN that is already created, see the procedure, Configuring VLAN Characteristics, page 3-6.
- You are familiar with the section, VLAN Numbering, page 3-2.
- Newly-created VLANs remain unused until Layer 2 ports are assigned to them.
- When you delete a specified VLAN, the ports associated to that VLAN are shut down and no traffic flows. When you delete a specified VLAN from a trunk port, only that VLAN is shut down and traffic continues to flow on all the other VLANs through the trunk port.

  However, the system retains all the VLAN-to-port mapping for that VLAN, and when you reenable, or re-create, that specified VLAN, the system automatically reinstates all the *original* ports to that VLAN. Note that the static MAC addresses and aging time for that VLAN are not restored when the VLAN is reenabled.
- For information about the following, see the document, *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)*.
  - Assigning Layer 2 interfaces to VLANs (access or trunk ports).
  - Configuring ports as VLAN access or trunk ports and assigning ports to VLANs.

SUMMARY STEPS

1. **config t**
2. show vlan
3. {*no*}**vlan** {*vlan-id* | *vlan-range*}
4. **exit**
5. **show vlan id** <*vlan-id*>
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you in CLI Global Configuration mode.
Step 2	show vlan **Example:** n1000v(config)# show vlan	Displays the VLANs that already exist.
Step 3	{***no***} **vlan** {*vlan-id* | *vlan-range*}	Creates or deletes, and saves in the running configuration, a VLAN or a range or VLANs. To configure the VLAN, see the procedure, Configuring VLAN Characteristics, page 3-6. **Note** If you enter a VLAN ID that is already assigned, you are placed into the VLAN Configuration mode for that VLAN. **Note** If you enter a VLAN ID that is assigned to an internally allocated VLAN, the system returns an error message. **Note** From the VLAN Configuration mode, you can also create and delete VLANs.
	Example: n1000v(config)# vlan 5 n1000v(config-vlan)#	This example shows VLAN 5 being created. The VLAN is activated and you are automatically placed into a submode for configuring VLAN 5.
	Example: n1000v# config t n1000v(config)# **vlan 15-20** n1000v(config-vlan)#	This example shows the range, VLAN 15-20, being created. The VLANs in the range are activated, and you are automatically placed into a submode for configuring VLAN 15-20. **Note** If you create a range of VLANs that includes an unusable VLAN, all VLANs in the range are created except those that are unusable; and Cisco Nexus 1000V returns a message listing the failed VLANs.
	Example: n1000v(config)# no vlan 3967 n1000v(config)#	This example shows VLAN 3967 being deleted, using the **no** form of the command.
Step 4	**exit** **Example:** n1000v(config-vlan)# exit n1000v(config)#	Exits the VLAN mode and returns you to CLI GLobal Configuration mode.

	Command	Purpose
Step 5	**show vlan id 5** **Example:** n1000v(config)# show vlan id 5	(Optional) Displays the VLAN configuration.
Step 6	**copy running-config startup-config** **Example:** n1000v(config)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

```
Example:
n1000v# config t
n1000v(config)# vlan 5
n1000v(config-vlan)# exit
n1000v(config)# show vlan id 5

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
5    VLAN0005                         active

VLAN Type
---- -----
5    enet

Remote SPAN VLAN
----------------
Disabled

Primary  Secondary  Type              Ports
-------  ---------  ---------------   ------------------------------------------

n1000v(config)# copy run start
[########################################] 100%
n1000v(config)#
```

Configuring VLAN Characteristics

Use this procedure to configure the following for a VLAN that has already been created:

- Name the VLAN.
- The operational state (active, suspend) of the VLAN.
- The VLAN media type (Ethernet).
- Shut down switching on the VLAN.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- Some characteristics cannot be modified on some VLANs. For more information, see the "VLAN Numbering" section on page 3-2.

Note Commands entered in the VLAN configuration submode are immediately saved to the running configuration.

SUMMARY STEPS

1. **config t**
2. **vlan** {*vlan-id* | *vlan-range*}
3. **name** *vlan-name*
4. **state** {**active** | **suspend**}
5. **no shutdown**
6. **exit**
7. **show vlan id**
8. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	**vlan** {*vlan-id* | *vlan-range*}
	
	Example:
	`n1000v(config)# vlan 5`
	`n1000v(config-vlan)#`
		Places you into the VLAN Configuration mode for the specified VLAN.
		Note If the VLAN does not already exist, the system creates it and then places you in the VLAN Configuration mode for that VLAN.

Step 3	**name** *vlan-name* **Example:** `n1000v(config-vlan)# name accounting`	Names the VLAN and saves it in the running configuration. • Up to 32 alphanumeric characters • You cannot change the name of VLAN1 nor the VLANs reserved for internal use. • The default name is VLANxxxx where xxxx represent four numeric digits (including leading zeroes) equal to the VLAN ID number.
Step 4	**state** {**active** | **suspend**} **Example:** `n1000v(config-vlan)# state active`	Changes the operational state of the VLAN and saves it in the running configuration. Allowable entries are: • Active (default) • Suspend While the VLAN state is suspended, the ports associated with this VLAN are shut down, and that VLAN does not pass any traffic. **Note** You cannot suspend the state for the default VLAN or VLANs 1006 to 4094.

	Command	Purpose
Step 5	**no shutdown** **Example:** n1000v(config-vlan)# no shutdown	Enables VLAN switching in the running configuration. Allowable entries are: • no shutdown (default) • shutdown **Note** You cannot shut down the default VLAN, VLAN1, or VLANs 1006 to 4094.
Step 6	**exit** **Example:** n1000v(config-vlan)# exit n1000v(config)#	Returns you to CLI Global Configuration mode.
Step 7	**show vlan [id <*vlan-id*>]** **Example:** n1000v(config)# show vlan id 5	(Optional) Displays the VLAN configuration.
Step 8	**copy running-config startup-config** **Example:** n1000v(config)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example:

```
n1000v# config t
n1000v(config)# vlan 5
n1000v(config-vlan)# name accounting
n1000v(config-vlan)# state active
n1000v(config-vlan)# no shutdown
n1000v(config-vlan)# exit
n1000v(config)# show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Eth2/1, Eth2/2, Eth2/3, Eth2/5
                                                Eth2/7, Eth2/8, Eth2/9, Eth2/10
                                                Eth2/15, Eth2/21, Eth2/22
                                                Eth2/23, Eth2/24, Eth2/25
                                                Eth2/46, Eth2/47, Eth2/48
5    accounting                       active
6    VLAN0006                         active
7    VLAN0007                         active
8    test                             active
9    VLAN0009                         active
10   VLAN0010                         active
50   VLAN0050                         active    Eth2/6
100  trunked                          active
200  VLAN0200                         active
201  VLAN0201                         active
202  VLAN0202                         active
3966 VLAN3966                         active
n1000v(config)#
```

Verifying a VLAN Configuration

Use the following commands to display and verify a VLAN configuration.

Command	Purpose
show running-config vlan <vlan-id>	Displays VLAN information in the running configuration.
show vlan [**all-ports** | **brief** | **id** <*vlan-id*> | **name** <*name*> | **dot1q tag native**]	Displays VLAN information as specified.
show vlan summary	Displays a summary of VLAN information.

Example 3-1 show vlan summary

```
n1000v(config)# show vlan summary

Number of existing VLANs           : 13
 Number of existing user VLANs     : 12
 Number of existing extended VLANs : 1

n1000v(config)#
```

Example 3-2 show vlan brief

```
n1000v(config)# show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Eth2/1, Eth2/2, Eth2/3, Eth2/5
                                                Eth2/7, Eth2/8, Eth2/9, Eth2/10
                                                Eth2/15, Eth2/21, Eth2/22
                                                Eth2/23, Eth2/24, Eth2/25
                                                Eth2/46, Eth2/47, Eth2/48
5    accounting                       active
6    VLAN0006                         active
7    VLAN0007                         active
8    test                             active
9    VLAN0009                         active
10   VLAN0010                         active
50   VLAN0050                         active    Eth2/6
100  trunked                          active
200  VLAN0200                         active
201  VLAN0201                         active
202  VLAN0202                         active
3966 VLAN3966                         active
n1000v(config)#
```

Additional References

For additional information related to implementing VLANs, see the following sections:

- Related Documents, page 3-10
- Standards, page 3-10

Related Documents

Related Topic	Document Title
Private VLANs	Chapter 4, "Configuring a Private VLAN."
Interfaces, VLAN interfaces (SVIs), IP addressing and port channels	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)*
Getting Started with Cisco Nexus 1000V and the CLI	*Cisco Nexus 1000V Getting Started Guide, Release 4.0(4)SV1(2)*
System management	*Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(2)*
Release notes	*Cisco Nexus 1000V Release Notes, Release 4.0(4)SV1(2)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(2)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for VLANs

This section provides the release history for the VLAN feature.

Table 3-5

Feature Name	Releases	Feature Information
VLANs	4.0(4)SV1(1)	This feature was introduced.

CHAPTER 4

Configuring a Private VLAN

Use this chapter to configure private VLAN (PVLAN) to divide a normal VLAN into isolated Layer 2 partitions.

The chapter contains the following topics:

Information About Private VLANs

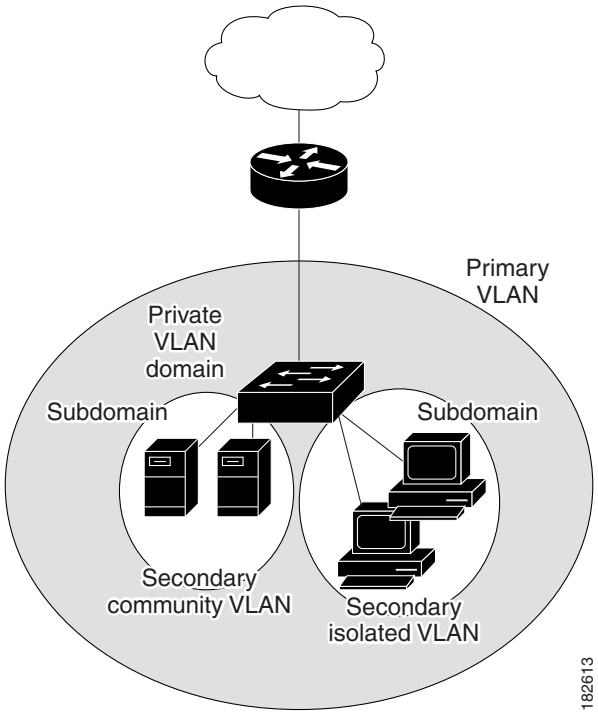
PVLANs achieve device isolation through the use of three separate port designations, each having its own unique set of rules regulating each connected endpoint's ability to communicate with other connected endpoints within the same private VLAN domain.

This section includes the following topics:

Private VLAN Domains

A private VLAN domain consists of one or more pairs of VLANs. The primary VLAN makes up the domain; and each VLAN pair makes up a subdomain. The VLANs in a pair are called the primary VLAN and the secondary VLAN. All VLAN pairs within a private VLAN have the same primary VLAN. The secondary VLAN ID is what differentiates one subdomain from another (see Figure 4-1).

Figure 4-1 Private VLAN Domain



Spanning Multiple Switches

Private VLANs can span multiple switches, just like regular VLANs. Inter-switch link ports need not be aware of the special VLAN type and carry frames tagged with these VLANs just like they do any other frames. Private VLANs ensure that traffic from an isolated port in one switch does not reach another isolated or community port in a different switch even after traversing an inter-switch link. By embedding the isolation information at the VLAN level and by transporting it along with the packet, it is possible to maintain consistent behavior throughout the network. Therefore, the mechanism which restricts Layer 2 communication between two isolated ports in the same switch, also restricts Layer 2 communication between two isolated ports in two different switches.

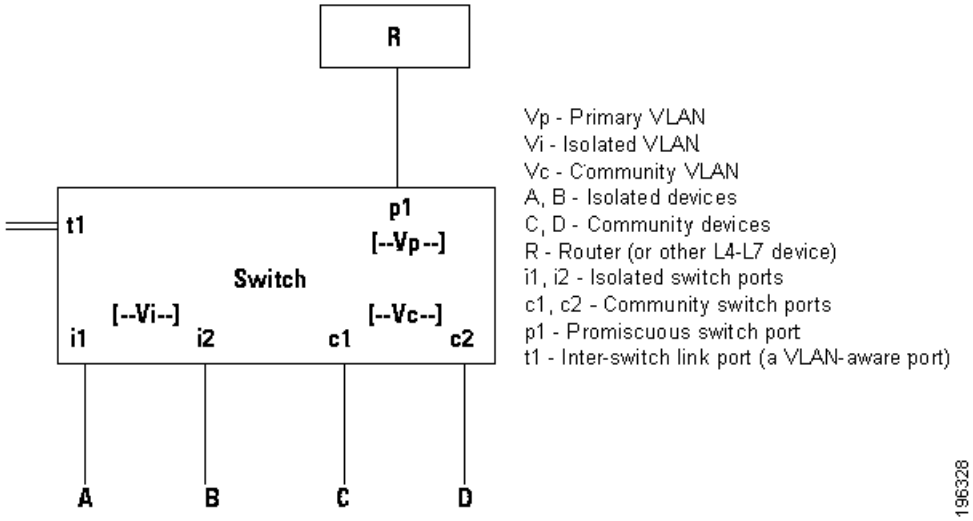
Private VLAN Ports

Within a private VLAN domain, there are three separate port designations. Each port designation has its own unique set of rules which regulate the ability of one endpoint to communicate with other connected endpoints within the same private VLAN domain. The following are the three port designations:

- promiscuous
- isolated
- community

Figure 4-2 shows the private VLAN ports

Figure 4-2 Private VLAN Ports



Primary VLANs and Promiscuous Ports

The primary VLAN encompasses the entire private VLAN domain. It is a part of each subdomain and provides the Layer 3 gateway out of the VLAN. A private VLAN domain has only one primary VLAN. Every port in a private VLAN domain is a member of the primary VLAN. In other words, the primary VLAN is the entire private VLAN domain.

As the name suggests, a promiscuous port (p1 in Figure 4-2) can talk to all other types of ports. A promiscuous port can talk to isolated ports as well as community ports and vice versa. Layer 3 gateways, DHCP servers and other trusted devices that need to communicate with the customer endpoints are typically connected with a promiscuous port. A promiscuous port can be either an access port or a hybrid/trunk port according to the terminology presented in Annex D of the IEEE 802.1Q specification.

Secondary VLANs and Host Ports

Secondary VLANs provide Layer 2 isolation between ports in a private VLAN domain. A private VLAN domain can have one or more subdomains. A subdomain is made up of a VLAN pair consisting of the primary VLAN and a secondary VLAN. Since the primary VLAN is a part of every subdomain, secondary VLANs differentiate the VLAN subdomains.

In order to communicate to the Layer 3 interface, a secondary VLAN must be associated with at least one of the promiscuous ports in the primary VLAN. You can associate a secondary VLAN to more than one promiscuous port within the same private VLAN domain, for example, if needed for load-balancing or redundancy. A secondary VLAN that is not associated with any promiscuous port cannot communicate with the Layer 3 interface.

A secondary VLAN can be one of the following types:

- Isolated VLANs— Isolated VLANs use isolated host ports. An isolated port (i1 or i2 in Figure 4-2) cannot talk to any other port in that private VLAN domain except for promiscuous ports. If a device needs to have access only to a gateway router, then it should be attached to an isolated port. An isolated port is typically an access port, but in certain applications it can also be a hybrid or trunk port.

The distinct characteristic of an isolated VLAN is that it allows all its ports to have the same degree of segregation that could be obtained from using one separate dedicated VLAN per port. Only two VLAN identifiers are consumed in providing this port isolation.

Note While there can be multiple community VLANs in a private VLAN domain, one isolated VLAN is sufficient to serve multiple customers. All endpoints connected to its ports are isolated at Layer 2. Service providers can assign multiple customers to the same isolated VLAN, and be assured that their Layer 2 traffic cannot be sniffed by other customers sharing the same isolated VLAN.

- Community VLANs—Community VLANs use community host ports. A community port (c1 or c2 in Figure 4-2) is part of a group of ports. The ports within a community can have Layer 2 communications with one another and can also talk to any promiscuous port. If an ISP customer has, for example, 4 devices and wants them isolated from those of other customers but still be able to communicate among themselves, then community ports should be used.

Note Because trunks can support a VLAN carrying traffic between its ports, it is possible for VLAN traffic to enter or leave the device through a trunk interface.

Communication Between Private VLAN Ports

The following table shows how access is permitted or denied between private VLAN port types.

Table 4-1 Communication Between Private VLAN Ports

	Isolated	Promiscuous	Community 1	Community 2	Interswitch Link Port [1]
Isolated	Deny	Permit	Deny	Deny	Permit
Promiscuous	Permit	Permit	Permit	Permit	Permit
Community 1	Deny	Permit	Permit	Deny	Permit
Community 2	Deny	Permit	Deny	Permit	Permit
Interswitch Link Port	Deny [2]	Permit	Permit	Permit	Permit

1. An interswitch link port is a regular port that connects two switches and that happens to carry two or more VLANs.
2. This behavior applies to traffic traversing inter-switch link ports over an isolated VLAN only. Traffic from an inter-switch link port to an isolated port will be denied if it is in the isolated VLAN. Traffic from an inter-switch link port to an isolated port will be permitted if it is in the primary VLAN.

Default Settings

Table 4-2 lists the default setting for a private VLAN.

Table 4-2 Default Private VLAN Setting

Parameters	Default
Private VLANs	Disabled

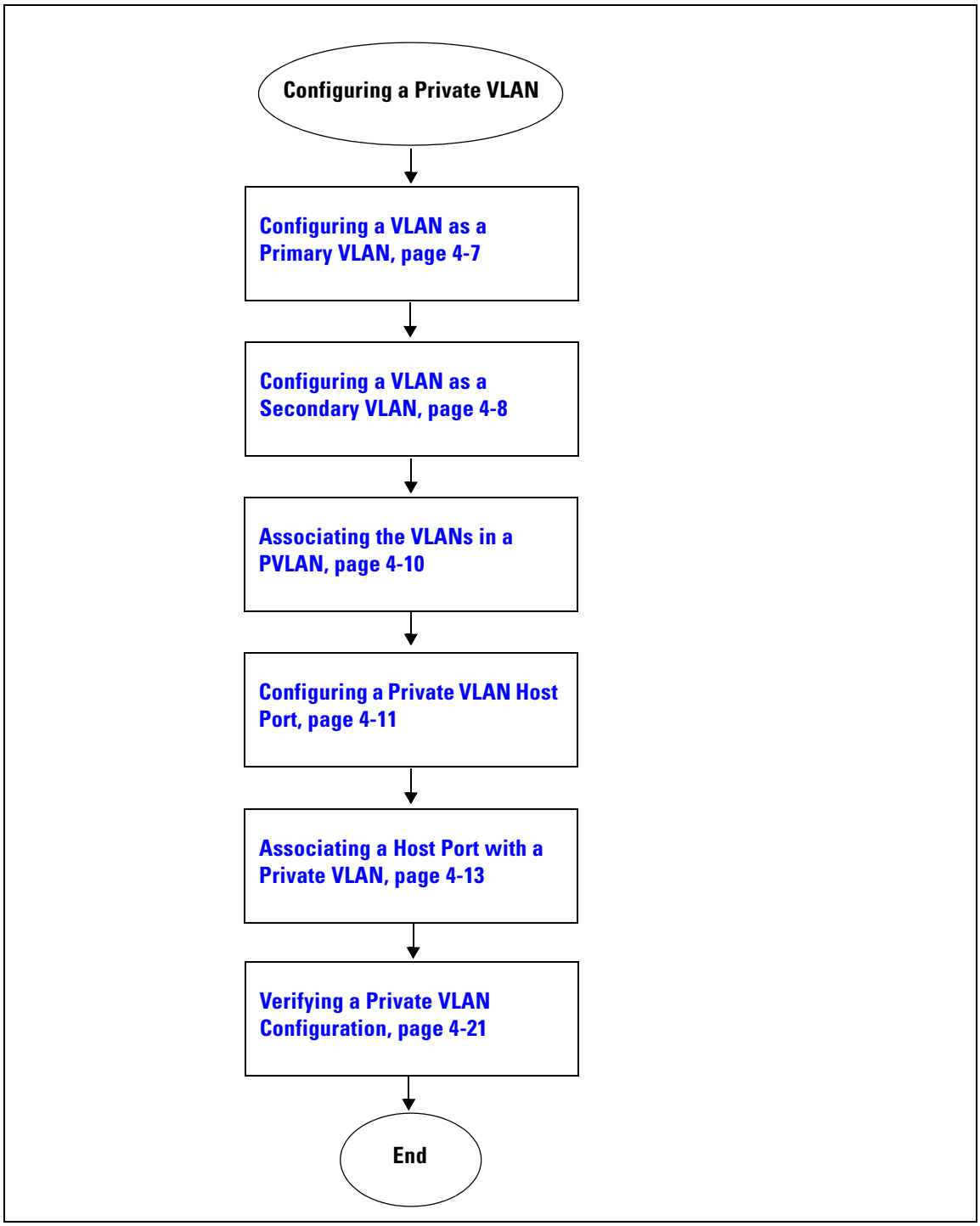
Configuring a Private VLAN

Use the following procedures in this section to configure a private VLAN.

Flow Chart: Configuring a Private VLAN

The following flow chart will guide you through this process. After completing each procedure, return to this section to make sure you complete all required procedures in the correct sequence.

Figure 4-3 Flow Chart: Configuring a Private VLAN



Configuring a VLAN as a Primary VLAN

Use this procedure to configure a VLAN to function as the primary VLAN in a PVLAN.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- The VLAN you are configuring as a primary VLAN already exists in the system as a normal VLAN, and you know the VLAN ID.

Note If the VLAN does not already exist, you are prompted to create it when you create the primary VLAN. For information about creating a VLAN, see the "Creating a VLAN" procedure on page 3-4.

SUMMARY STEPS

1. **config t**
2. **vlan** *primary-vlan-id*
3. **private-vlan primary**
4. **exit**
5. **show vlan private-vlan**
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**vlan** *primary-vlan-id*
	
	Example:
	`n1000v(config)# vlan 202`
	`n1000v(config-vlan)#`
		Places you into VLAN Configuration mode for the specified VLAN and configures the primary VLAN ID in the running configuration.

Step 3	**private-vlan primary** **Example:** `n1000v(config-vlan)# private-vlan primary`	Designates the primary VLAN as a private VLAN in the running configuration.

Step 4	**exit**
	
	Example:
	`n1000v(config-vlan)# exit`
	`n1000v(config)#`
		Exits the VLAN Configuration mode.

Step 5	**show vlan private-vlan** **Example:** `n1000v(config)# show vlan private-vlan`	(Optional) Displays the PVLAN configuration.

	Command	Description
Step 6	**copy running-config startup-config** **Example:** n1000v(config)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 7	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

```
Example:
n1000v(config)# vlan 202
n1000v(config-vlan)# private-vlan primary
n1000v(config-vlan)# exit
n1000v(config-vlan)# show vlan private-vlan
Primary  Secondary  Type             Ports
-------  ---------  ---------------  -------------------------------------------
202                 primary
n1000v(config-vlan)#
```

Configuring a VLAN as a Secondary VLAN

Use this procedure to configure VLANs to function as secondary VLANs in a PVLAN.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- The VLANs you are configuring as secondary VLANs already exists in the system as normal VLANs, and you know their VLAN IDs.

Note If the VLAN does not already exist, you are prompted to create it when you create the secondary VLAN. To create a VLAN, see the section, Creating a VLAN, page 3-4.

- You know whether you want the secondary VLANs to be community VLANs or isolated VLANs, and the VLAN IDs for each.
- For information about private VLANs, see the section, Private VLANs, page 1-6.

SUMMARY STEPS

1. **config t**
2. **vlan** *secondary-vlan-id*
3. **private-vlan {community | isolated}**
4. **exit**
5. **show vlan private-vlan**
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**vlan** *secondary-vlan-id*
	
	Example:
	`n1000v(config)# vlan 303`
	`n1000v(config-vlan)#`
		Places you in VLAN Configuration mode for the specified VLAN; and configures the secondary VLAN ID in the running configuration.

Step 3	**private-vlan {community | isolated}**
	
	Example:
	`n1000v(config-vlan)# private-vlan community`
	`n1000v(config-vlan)#`
	
	Example:
	`n1000v(config-vlan)# private-vlan isolated`
	`n1000v(config-vlan)#`
		Designates the VLAN as either a community or isolated private VLAN in the running configuration.

Step 4	Do one of the following: • If you are configuring additional secondary VLANs for your PVLAN, repeat Step 2 and Step 3. • Otherwise, continue with Step 5.	

Step 5	**exit**
	
	Example:
	`n1000v(config-vlan)# exit`
	`n1000v(config)#`
		Exits the VLAN Configuration mode.

Step 6	**show vlan private-vlan** **Example:** `n1000v(config)# show vlan private-vlan`	(Optional) Displays the PVLAN configuration.
Step 7	**copy running-config startup-config** **Example:** `n1000v(config)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 8	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

Example:

```
n1000v(config)# vlan 303
n1000v(config-vlan)# private-vlan community
n1000v(config-vlan)# exit
n1000v(config)# show vlan private-vlan
Primary  Secondary  Type              Ports
-------  ---------  ----------------  ------------------------------------------
202                 primary
303                 community
n1000v(config)#
```

Associating the VLANs in a PVLAN

Use this procedure to associate the primary VLANs in a PVLAN with the secondary VLANs.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- The primary VLAN for this PVLAN is already configured as a PVLAN.
- The secondary VLANs for this PVLAN are already configured as PVLANs.
- You know the VLAN IDs for each VLAN that is a part of the PVLAN.
- For information about private VLANs, see the "Private VLANs" section on page 1-6.

SUMMARY STEPS

1. **config t**
2. **vlan** *primary-vlan-id*
3. **private-vlan association** {**add** | **remove**} *secondary vlan-id*
4. **exit**
5. **show vlan private-vlan**
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**vlan** *primary-vlan-id*
	
	Example:
	`n1000v(config)# vlan 202`
	`n1000v(config-vlan)#`
		Places you in VLAN Configuration mode and associates the VLANs to function as a PVLAN in the running configuration.

Step 3	**private-vlan association** {**add** | **remove**} *secondary vlan-id*
	
	Example:
	`n1000v(config-vlan)# private-vlan association add 303`
	`n1000v(config-vlan)#`
		Associates a specified secondary VLAN with the primary VLAN to function as a PVLAN in the running configuration.

Step 4	Do one of the following: • If you are associating additional secondary VLANs, repeat Step 3. • Otherwise, continue with Step 5.	

	Command	Description

Step 5	**exit**
	
	Example:
	`n1000v(config-vlan)# exit`
	`n1000v(config)#`
		Exits the VLAN Configuration mode and returns you to CLI Global Configuration mode.

Step 6	**show vlan private-vlan** **Example:** `n1000v(config)# show vlan private-vlan`	(Optional) Displays the PVLAN configuration.
Step 7	**copy running-config startup-config** **Example:** `n1000v(config)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 8	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

```
Example:
n1000v(config)# vlan 202
n1000v(config-vlan)# private-vlan association add 303
n1000v(config-vlan)# exit
n1000v(config)# show vlan private-vlan
Primary  Secondary  Type             Ports
-------  ---------  ---------------  ------------------------------------------
202      303        community        Veth1
n1000v(config)#
```

Configuring a Private VLAN Host Port

Use this procedure to configure an interface as a host port to function with a PVLAN.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- The primary VLAN for this PVLAN is already configured as a PVLAN.
- The secondary VLANs for this PVLAN are already configured as PVLANs.
- The secondary VLANs are already associated with the primary VLAN.
- You know the name of the interface to be used with the PVLAN as a host port.
- For information about private VLANs, see the section, Private VLANs, page 1-6.

SUMMARY STEPS

1. **config t**
2. **interface** *interface name*
3. **switchport mode private-vlan host**
4. **exit**
5. **show interface** *interface name*
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description
Step 1	**config t** **Example:** n1000v# configure t n1000v(config)#	Places you in CLI Global Configuration mode.
Step 2	**interface** *interface name* **Example:** n1000v(config)# interface veth1 n1000v(config-if)#	Places you in Interface Configuration mode and configures a name for the specified interface in the running configuration.
Step 3	Do one of the following: • If you are configuring a physical interface, continue with the next step. • Otherwise, go to Step 4.	
Step 4	**switchport mode private-vlan host** **Example:** n1000v(config-if)# switchport mode private-vlan host n1000v(config-if)#	Designates that the physical interface is to function as a PVLAN host port in the running configuration.
Step 5	**exit** **Example:** n1000v(config-if)# exit n1000v(config)#	Exits the interface configuration submode and returns you to CLI Global Configuration mode.
Step 6	**show interface** *interface name* **Example:** n1000v(config)# show interface veth1	(Optional) Displays the interface configuration.
Step 7	**copy running-config startup-config** **Example:** n1000v(config)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 8	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

```
Example:
n1000v# config t
n1000v(config)# interface veth1
n1000v(config-if)# switchport mode private-vlan host
n1000v(config-if)# exit
n1000v(config)# show interface veth1
Vethernet1 is up
    Hardware is Virtual, address is 0050.56b0.34c8
    Owner is VM "HAM61-RH5-32bit-ENVM-7.60.1.3"
    Active on module 2, host VISOR-HAM61.localdomain 0
    VMware DVS port 16777215
    Port-Profile is vlan631
    Port mode is Private-vlan host
    Rx
    48600 Input Packets 34419 Unicast Packets
    0 Multicast Packets 14181 Broadcast Packets
    4223732 Bytes
    Tx
```

```
    34381 Output Packets 34359 Unicast Packets
    22 Multicast Packets 0 Broadcast Packets 0 Flood Packets
    3368196 Bytes
    5 Input Packet Drops 11 Output Packet Drops

n1000v(config)#
```

Associating a Host Port with a Private VLAN

Use this procedure to associate the host port with the primary and secondary VLANs in a PVLAN.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You know the VLAN IDs of the primary and secondary VLANs in the PVLAN.
- The primary and secondary VLANs are already configured as PVLAN.
- You know the name of the interface functioning in the PVLAN as a host port.
- For information about private VLANs, see the section, Private VLANs, page 1-6.

SUMMARY STEPS

1. **config t**
2. **interface** *interface name*
3. **switchport private-vlan host-association** *primaryvlan-id secondary vlan-id(s)*
4. **exit**
5. **show interface** *interface name*
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**interface** *interface name*
	
	Example:
	`n1000v(config-if)# interface veth1`
	`n1000v(config-if)#`
		Places you in Interface Configuration mode and configures a name for the specified interface in the running configuration.

Step 3	**switchport private-vlan host-association** *primaryvlan-id secondary vlan-id(s)*
	
	Example:
	`n1000v(config-if)# switchport private-vlan host-association 202 303`
	`n1000v(config-if)#`
		Associates the host port with the primary and secondary VLAN IDs for the PVLAN in the running configuration.
		
		The interface is associated with the VLANs in the PVLAN.

	Command	Description
Step 4	**exit** **Example:** n1000v(config-if)# exit n1000v(config)#	Exits the interface configuration submode and returns you to CLI Global Configuration mode.
Step 5	**show interface** *interface name* **Example:** n1000v(config)# show interface veth1	(Optional) Displays the interface configuration.
Step 6	**copy running-config startup-config** **Example:** n1000v# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 7	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

Example:

```
n1000v# config t
n1000v(config)# interface veth1
n1000v(config-if)# switchport mode private-vlan host
n1000v(config-if)# exit
n1000v(config)# show interface veth1
Vethernet1 is up
    Hardware is Virtual, address is 0050.56b0.34c8
    Owner is VM "HAM61-RH5-32bit-ENVM-7.60.1.3"
    Active on module 2, host VISOR-HAM61.localdomain 0
    VMware DVS port 16777215
    Port-Profile is vlan631
    Port mode is Private-vlan host
    Rx
    48600 Input Packets 34419 Unicast Packets
    0 Multicast Packets 14181 Broadcast Packets
    4223732 Bytes
    Tx
    34381 Output Packets 34359 Unicast Packets
    22 Multicast Packets 0 Broadcast Packets 0 Flood Packets
    3368196 Bytes
    5 Input Packet Drops 11 Output Packet Drops

n1000v(config)#
```

Configuring a Layer 2 Interface as a Promiscuous Trunk Port

Use this procedure to configure a Layer 2 interface as a promiscuous trunk port that does the following:

- Combines multiple promiscuous ports into a single trunk port.
- Carries all normal VLANs.
- Carries multiple PVLAN primary VLANs each with selected secondary VLANs.

Note A promiscuous port can be either access or trunk. If you have one primary vlan you can use a promiscuous access port. If you have multiple primary vlans you can use a promiscuous trunk port.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- The **private-vlan mapping trunk** command does not decide or override the trunk configuration of a port.
- The port is already configured in a regular trunk mode before adding the private-vlan trunk configurations.
- Primary VLANs must be added to the list of allowed VLAN for the promiscuous trunk port.
- Secondary VLANs are not configured in the allowed VLAN list.
- The trunk port can carry normal VLANs in addition to primary VLANs.

SUMMARY STEPS

1. **config t**
2. **interface** *type slot/port*
3. **switchport mode private-vlan trunk promiscuous**
4. **switchport private-vlan trunk allowed vlan all**
5. **switchport private-vlan mapping trunk** *primary_vlan_ID* {*secondary_vlan_list* | **add** *secondary_vlan_list* | **remove** *secondary_vlan_list*}
6. **exit**
7. **show interfaces** [*type slot/port*] **switchport**
8. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**interface** *type slot/port*
	
	Example:
	`n1000v(config)# interface eth2/6`
	`n1000v(config-if)#`
		Places you in Interface Configuration mode for the specified interface.

Step 3	**switchport mode private-vlan trunk promiscuous**
	
	Example:
	`n1000v(config-if)# switchport mode private-vlan trunk promiscuous`
	`n1000v(config-if)#`
		In the running configuration, designates the interface as a promiscuous private-vlan trunk port.

Step 4	**switchport private-vlan trunk allowed vlan all**
	
	Example:
	`n1000v(config-if)# switchport private-vlan trunk allowed vlan all`
	`n1000v(config-if)#`
		In the running configuration, designates that the private-vlan trunk port will carry all normal VLANs.

Step 5	**switchport private-vlan mapping trunk** *primary_vlan_ID* **{***secondary_vlan_list* **| add** *secondary_vlan_list* **| remove** *secondary_vlan_list***}**
	
	Example:
	`n1000v(config-if)# switchport private-vlan mapping trunk 202 303, 440`
	`n1000v(config-if)# switchport private-vlan mapping trunk 210 310, 450`
		Maps the private-vlan trunk port to a primary VLAN and to selected secondary VLANs in the running configuration.
		
		Multiple private-vlan pairs can be specified so that a promiscuous trunk port can carry multiple primary VLANs.

Step 6	**exit**	Exits Interface Configuration mode and returns you to CLI Global Configuration mode.
Step 7	**show interface** [*type slot/port*] **switchport** **Example:** `n1000v(config-if)# show int switchport`	Displays the configuration for verification.
Step 8	**copy running-config startup-config** **Example:** `n1000v# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example:

```
n1000v# config t
n1000v(config)# int eth2/6
n1000v(config-if)# switchport mode private-vlan trunk promiscuous
n1000v(config-if)# switchport private-vlan trunk allowed vlan all
n1000v(config-if)# switchport private-vlan mapping trunk 202 303, 440
n1000v(config-if)# switchport private-vlan mapping trunk 210 310, 450
n1000v(config-if)# show int switchport
Name: Vethernet1
  Switchport: Enabled
  Operational Mode: trunk
  Access Mode VLAN: 156(VLAN0156)
  Trunking Native Mode VLAN: 1 (default)
  Trunking VLANs Enabled: 1-3967,4048-4093
  Administrative private-vlan host-association: not available
  Administrative private-vlan mapping: not available
  Administrative private-vlan trunk native VLAN: none
  Administrative private-vlan trunk encapsulation: dot1q
  Administrative private-vlan trunk normal VLANs: none
  Administrative private-vlan trunk private VLANs: none
  Operational private-vlan: not available

n1000v(config-if)#
```

Configuring a Private VLAN Promiscuous Access Port

Use this procedure to configure a port to be used as a promiscuous access port in a PVLAN.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You know the name of the interface that will function as a promiscuous access port.
- For information about private VLANs, see the section, Private VLANs, page 1-6.

SUMMARY STEPS

1. **config t**
2. **interface** *type* [*slot/port* | *number*]
3. **switchport mode private-vlan promiscuous**
4. **exit**
5. **show interface** *type* [*slot/port* | *number*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**interface** *type* [*slot/port*	*number*]
		
		Example:
		`n1000v(config-if)# interface veth1`
		`n1000v(config-if)#`
		
		Example:
		`n1000v(config-if)# interface eth3/2`
		`n1000v(config-if)#`
			Places you in Interface Configuration mode for a specified interface.

Step 3	**switchport mode private-vlan promiscuous**
	
	Example:
	`n1000v(config-if)# switchport mode private-vlan promiscuous`
	`n1000v(config-if)#`
		Designates that the interface is to function as a promiscuous access port for a PVLAN in the running configurtion.

Step 4	**exit**
	
	Example:
	`n1000v(config-if)# exit`
	`n1000v(config)#`
		Returns you to Global Configuration mode.

Step 5	**show interface** *type* [*slot/port*	*number*] **Example:** `n1000v(config)# show interface eth3/2`	(Optional) Displays the interface configuration.

	Command	Description
Step 6	**copy running-config startup-config** **Example:** n1000v# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 7	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

```
Example:
n1000v# config t
n1000v(config)# interface eth3/2
n1000v(config-if)# switchport mode private-vlan promiscuous
n1000v(config-if)# exit
n1000v(config)# show int eth3/2
Ethernet3/2 is up
    Hardware is Ethernet, address is 0050.5655.2e85 (bia 0050.5655.2e85)
    MTU 1500 bytes, BW -1942729464 Kbit, DLY 10 usec,
       reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA
    Port mode is promiscuous
    full-duplex, 1000 Mb/s
    Beacon is turned off
    Auto-Negotiation is turned on
    Input flow-control is off, output flow-control is off
    Rx
    276842 Input Packets 100419 Unicast Packets
    138567 Multicast Packets 37856 Broadcast Packets
    25812138 Bytes
    Tx
    128154 Output Packets 100586 Unicast Packets
    1023 Multicast Packets 26545 Broadcast Packets 26582 Flood Packets
    11630220 Bytes
    173005 Input Packet Drops 37 Output Packet Drops

n1000v(config)#
```

Associating a Promiscuous Access Port with a Private VLAN

Use this procedure to associate the promiscuous access port with the primary and secondary VLANs in a PVLAN.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You know the VLAN IDs of the primary and secondary VLANs in the PVLAN.
- The primary and secondary VLANs are already configured as PVLAN.
- You know the name of the interface functioning in the PVLAN as a promiscuous access port.
- For information about private VLANs, see the section, Private VLANs, page 1-6.

SUMMARY STEPS

1. **config t**

2. **interface** *type* [*slot/port* | *number*]
3. **switchport private-vlan mapping** *primary vlan-id secondary vlan-id(s)*
4. **exit**
5. **show interface** *type* [*slot/port* | *number*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**interface** *type* [*slot/port*	*number*]
		
		Example:
		`n1000v(config)# interface eth3/2`
		`n1000v(config-if)#`
			Places you in Interface Configuration mode for the specified interface in the running configuration.

Step 3	**switchport private-vlan mapping** *primary vlan-id secondary vlan-id(s)*
	
	Example:
	`n1000v(config-if)# switchport private-vlan mapping 202 303`
	`n1000v(config-if)#`
		Associates the promiscuous access port with the VLAN IDs in the PVLAN in the running configuration.

Step 4	**exit**
	
	Example:
	`n1000v(config-if)# exit`
	`n1000v(config)#`
		Returns you to EXEC mode.

Step 5	**show interface** *type* [*slot/port*	*number*] **Example:** `n1000v(config)# show vlan private-vlan`	(Optional) Displays the interface configuration.
Step 6	**copy running-config startup-config** **Example:** `n1000v# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 7	You have completed this procedure. If using the flow chart, return to the Figure 4-3, Flow Chart: Configuring a Private VLAN, on page 4-6	

```
Example:
n1000v(config)# int eth3/2
n1000v(config-if)# switchport private-vlan mapping 202 303
n1000v(config-if)# exit

n1000v(config)# show vlan private-vlan
Primary  Secondary  Type              Ports
-------  ---------  ----------------  ------------------------------------------
202      303        community         Eth3/2, Veth1
n1000v(config)#
```

Removing a Private VLAN Configuration

Use this procedure to remove a private VLAN configuration and return the VLAN to normal VLAN mode.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- The VLAN is configured as a private VLAN, and you know the VLAN ID.
- When you remove a PVLAN configuration, the ports associated with it become inactive.
- For information about private VLANs, see the section, Private VLANs, page 1-6.

SUMMARY STEPS

1. **config t**
2. **vlan** *private vlan-id*
3. **no private-vlan {community | isolated | primary}**
4. **exit**
5. **show vlan private-vlan**
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# configure t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**vlan** *private vlan-id*
	
	Example:
	`n1000v(config)# vlan 5`
	`n1000v(config-vlan)#`
		Places you in the VLAN configuration mode for the specified VLAN.

Step 3	**no private-vlan {community | isolated | primary}**
	
	Example:
	`n1000v(config-vlan)# no private-vlan primary`
	`n1000v(config-vlan)#`
		Removes the specified VLAN from a PVLAN in the running configuration.
		
		The private VLAN configuration is removed from the specified VLAN(s). The VLAN is returned to normal VLAN mode. The ports associated with the VLAN are inactive.

Step 4	**exit**
	
	Example:
	`n1000v(config-vlan)# exit`
	`n1000v(config)#`
		Exits the VLAN configuration submode.

	Command	Purpose
Step 5	**show vlan private-vlan**	(Optional) Displays the PVLAN configuration.
Step 6	**copy running-config startup-config** **Example:** n1000v# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

```
Example:
n1000v# configure t
n1000v(config)# vlan 5
n1000v(config-vlan)# no private-vlan primary
n1000v(config-vlan)# exit
n1000v(config)# show vlan private-vlan
Primary  Secondary  Type             Ports
-------  ---------  ---------------  -------------------------------------------

n1000v(config)#
```

Verifying a Private VLAN Configuration

Use the following commands to displayand verify a private VLAN configuration.

Command	Purpose
show running-config vlan <**vlan-id**>	Displays VLAN information.
show vlan private-vlan [*type*]	Displays information about private VLANs
show interface private-vlan mapping	Displays interface private VLAN information.
show interface switchport	Displays information about all interfaces configured as switchports.

Example Configurations for Private VLAN

This section includes the following example configurations:

- PVLAN Trunk Port, page 4-21
- PVLAN Using Port Profiles, page 4-22

PVLAN Trunk Port

The following example shows how to configure interface Ethernet 2/6 as the following:

- private VLAN trunk port
- mapped to primary private VLAN 202 which is associated with secondary VLANs 303 and 440
- mapped to primary private VLAN 210 which is associated with secondary VLANs 310 and 450

```
Example:
n1000v# config t
n1000v(config)# int eth2/6
```

```
n1000v(config-if)# switchport mode private-vlan trunk promiscuous
n1000v(config-if)# switchport private-vlan trunk allowed vlan all
n1000v(config-if)# switchport private-vlan mapping trunk 202 303, 440
n1000v(config-if)# switchport private-vlan mapping trunk 210 310, 450
n1000v(config-if)# show int switchport
Name: Vethernet1
  Switchport: Enabled
  Operational Mode: trunk
  Access Mode VLAN: 156(VLAN0156)
  Trunking Native Mode VLAN: 1 (default)
  Trunking VLANs Enabled: 1-3967,4048-4093
  Administrative private-vlan host-association: not available
  Administrative private-vlan mapping: not available
  Administrative private-vlan trunk native VLAN: none
  Administrative private-vlan trunk encapsulation: dot1q
  Administrative private-vlan trunk normal VLANs: none
  Administrative private-vlan trunk private VLANs: none
  Operational private-vlan: not available

n1000v(config-if)#
```

PVLAN Using Port Profiles

The following example configuration shows how to configure interface eth2/6 using port-profile, uppvlanpromtrunk156.

In this configuration, packets from secondary interfaces 153, 154, and 155 are translated into the primary VLAN 156 as a result of the command, **switchport private-vlan mapping trunk 156 153-155**.

Example:

```
vlan 153-154
  private-vlan community
vlan 155
  private-vlan isolated
vlan 156
  private-vlan association 153-155,157-158
  private-vlan primary
vlan 157
  private-vlan community
vlan 158
  private-vlan isolated


n1000v# show run int eth2/6
version 4.0(1)

interface Ethernet2/6
  switchport
  inherit port-profile uppvlanpromtrunk156

n1000v# show port-profile name uppvlanpromtrunk156
port-profile uppvlanpromtrunk156
  description:
  status: enabled
  capability privileged: no
  capability uplink: yes
  port-group: uppvlanpromtrunk156
  config attributes:
    switchport mode private-vlan trunk promiscuous
    switchport private-vlan trunk allowed vlan all
    switchport private-vlan mapping trunk 156 153-155
    no shutdown
```

```
  evaluated config attributes:
    switchport mode trunk
    switchport trunk allowed vlan all
    switchport private-vlan mapping trunk 156 153-155
    no shutdown
  assigned interfaces:
    Ethernet2/6
    Ethernet3/3

n1000v# show int
mgmt0 is up
    Hardware is GigabitEthernet, address is 0000.0000.0000 (bia 0050.56b8.6790)
    Internet Address is 172.28.15.94/24
    MTU 1500 bytes, BW 0 Kbit, DLY 0 usec,
       reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA
    full-duplex, 1000 Mb/s
    Beacon is turned off
    Auto-Negotiation is turned on
    Input flow-control is off, output flow-control is off
    61570 packets input, 15391960 bytes
    0 multicast frames, 0 compressed
    0 input errors, 0 frame, 0 overrun, 0 fifo
    4062 packets output, 967297 bytes
    0 underrun, 0 output errors, 0 collisions
    0 fifo, 0 carrier errors

Ethernet2/2 is up
    Hardware is Ethernet, address is 0050.565e.4c39 (bia 0050.565e.4c39)
    MTU 1500 bytes, BW 1826767368 Kbit, DLY 10 usec,
       reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA
    Port mode is trunk
    full-duplex, 1000 Mb/s
    Beacon is turned off
    Auto-Negotiation is turned on
    Input flow-control is off, output flow-control is off
    Rx
    681251 Input Packets 161488 Unicast Packets
    365259 Multicast Packets 154504 Broadcast Packets
    54980953 Bytes
    Tx
    176537 Output Packets 172242 Unicast Packets
    4100 Multicast Packets 195 Broadcast Packets 5 Flood Packets
    19753822 Bytes
    98004 Input Packet Drops 9 Output Packet Drops

Ethernet2/6 is up
    Hardware is Ethernet, address is 0050.565b.b9db (bia 0050.565b.b9db)
    MTU 1500 bytes, BW 1689405960 Kbit, DLY 10 usec,
       reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA
    Port mode is trunk
    full-duplex, 1000 Mb/s
    Beacon is turned off
    Auto-Negotiation is turned on
    Input flow-control is off, output flow-control is off
    Rx
    702 Input Packets 0 Unicast Packets
    702 Multicast Packets 0 Broadcast Packets
    46133 Bytes
    Tx
    95 Output Packets 0 Unicast Packets
    95 Multicast Packets 0 Broadcast Packets 0 Flood Packets
```

```
    6487 Bytes
    44 Input Packet Drops 10 Output Packet Drops

Ethernet3/2 is up
    Hardware is Ethernet, address is 0050.5653.98ac (bia 0050.5653.98ac)
    MTU 1500 bytes, BW 348735240 Kbit, DLY 10 usec,
       reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA
    Port mode is access
    full-duplex, 1000 Mb/s
    Beacon is turned off
    Auto-Negotiation is turned on
    Input flow-control is off, output flow-control is off
    Rx
    579977 Input Packets 138070 Unicast Packets
    308931 Multicast Packets 132976 Broadcast Packets
    45973630 Bytes
    Tx
    165416 Output Packets 163449 Unicast Packets
    1830 Multicast Packets 137 Broadcast Packets 3 Flood Packets
    22602711 Bytes
    1975345 Input Packet Drops 0 Output Packet Drops

n1000v# show port-profile name uppvlanpromtrunk156
port-profile uppvlanpromtrunk156
  description:
  status: enabled
  capability privileged: no
  capability uplink: yes
  port-group: uppvlanpromtrunk156
  config attributes:
    switchport mode private-vlan trunk promiscuous
    switchport private-vlan trunk allowed vlan all
    switchport private-vlan mapping trunk 156 153-155
    no shutdown
  evaluated config attributes:
    switchport mode trunk
    switchport trunk allowed vlan all
    switchport private-vlan mapping trunk 156 153-155
    no shutdown
  assigned interfaces:
    Ethernet2/6
    Ethernet3/3

n1000v# show port-profile expand-interface

port-profile uplinkportprofile1
Ethernet2/2
    switchport mode private-vlan trunk promiscuous
    switchport private-vlan trunk allowed vlan all
    switchport trunk allowed vlan 150-152
    no shutdown

port-profile upaccess152

port-profile uppvlanpromaccess156
Ethernet3/2
    switchport mode private-vlan promiscuous
    switchport private-vlan mapping 156 153-155
    no shutdown

port-profile pvlancomm154
Vethernet49
    switchport mode private-vlan host
```

```
    switchport private-vlan host-association 156 154
    no shutdown
Vethernet81
    switchport mode private-vlan host
    switchport private-vlan host-association 156 154
    no shutdown

port-profile pvlaniso155

port-profile pvlancomm157
n1000v# show port-profile expand-interface ?
  <CR>
  >      Redirect it to a file
  name   Select a port profile by name
  |      Pipe command output to filter

n1000v# show port-profile expand-interface name uppvlanpromtrunk156

port-profile uppvlanpromtrunk156
Ethernet2/6
    switchport mode trunk
    switchport trunk allowed vlan 1-3967,4048-4093
    switchport private-vlan mapping trunk 156 153-155
        no shutdown
Ethernet3/3
    switchport trunk allowed vlan 1-3967,4048-4093
n1000v# show int eth2/6
Ethernet2/6 is up
    Hardware is Ethernet, address is 0050.565b.b9db (bia 0050.565b.b9db)
    MTU 1500 bytes, BW 1689405960 Kbit, DLY 10 usec,
       reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA
    Port mode is trunk
    full-duplex, 1000 Mb/s
    Beacon is turned off
    Auto-Negotiation is turned on
    Input flow-control is off, output flow-control is off
    Rx
    953 Input Packets 0 Unicast Packets
    953 Multicast Packets 0 Broadcast Packets
    62600 Bytes
    Tx
    96 Output Packets 0 Unicast Packets
    96 Multicast Packets 0 Broadcast Packets 0 Flood Packets
    6683 Bytes
    44 Input Packet Drops 10 Output Packet Drops
```

Additional References

For additional information related to implementing private VLANs, see the following sections:

- Related Documents, page 4-26
- Standards, page 4-26

Related Documents

Related Topic	Document or Chapter Title
VLANs	Chapter 3, "Configuring VLANs"
PVLAN	Chapter 4, "Configuring a Private VLAN"
Layer 2 MAC addresses	Chapter 2, "Configuring the MAC Address Table"
Port Profiles	*Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(2)*
VLAN interfaces, IP addressing	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)*
Static MAC addresses, security	*Cisco Nexus 1000V Security Configuration Guide, Release 4.0(4)SV1(2)*
Cisco Nexus 1000V and CLI configuration basics	*Cisco Nexus 1000V Getting Started Guide, Release 4.0(4)SV1(2)*
System management	*Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(2)*
Release notes	*Cisco Nexus 1000V Release Notes, Release 4.0(4)SV1(2)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(2)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for Private VLAN

This section provides the private VLAN release history.

Feature Name	Releases	Feature Information
Private VLAN	4.0(4)SV1(1)	This feature was introduced.

CHAPTER 5

Configuring IGMP Snooping

This chapter describes how to configure Internet Group Management Protocol (IGMP) snooping.

This chapter includes the following topics:

Information about IGMP Snooping

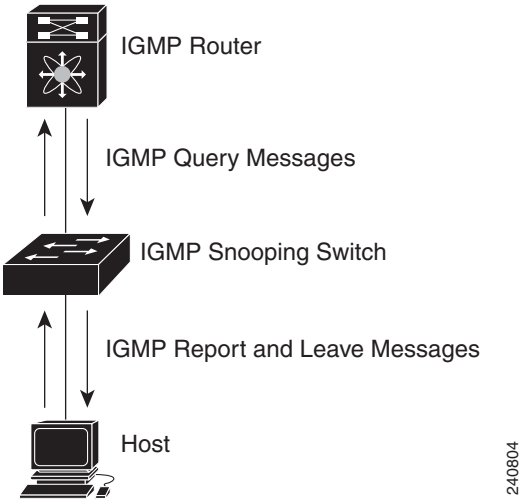
This section includes the following topics:

IGMP Snooping

The Internet Group Management Protocol (IGMP) snooping software examines Layer 2 IP multicast traffic within a VLAN to discover the ports where interested receivers reside. Using the port information, IGMP snooping can reduce bandwidth consumption in a multi-access LAN environment to avoid flooding the entire VLAN. The IGMP snooping feature tracks which ports are attached to multicast-capable routers to help the routers forward IGMP membership reports. The IGMP snooping software responds to topology change notifications. By default, IGMP snooping is enabled on the device.

Figure 5-1 shows an IGMP snooping switch that sits between the host and the IGMP router. The IGMP snooping switch snoops the IGMP membership reports and Leave messages and forwards them only when necessary to the connected IGMP routers.

Figure 5-1 IGMP Snooping Switch



The IGMP snooping software operates upon IGMPv1, IGMPv2, and IGMPv3 control plane packets where Layer 3 control plane packets are intercepted and influence the Layer 2 forwarding behavior.

The Cisco Nexus 1000V IGMP snooping implementation has the following proprietary features:

- Source filtering that allows forwarding of multicast packets based on destination and source IP.
- Multicast forwarding based on IP address rather than MAC address.
- Optimized multicast flooding (OMF) that forwards unknown traffic to routers only and performs no data driven state creation.

For more information about IGMP snooping, see RFC 4541.

IGMPv1 and IGMPv2

Both IGMPv1 and IGMPv2 support membership report suppression, which means that if two hosts on the same subnet want to receive multicast data for the same group, then the host that receives a member report from the other host suppresses sending its report. Membership report suppression occurs for hosts that share a port.

If no more than one host is attached to each VLAN switch port, then you can configure the fast leave feature in IGMPv2. The fast leave feature does not send last member query messages to hosts. As soon as the software receives an IGMP leave message, the software stops forwarding multicast data to that port.

IGMPv1 does not provide an explicit IGMP leave message, so the software must rely on the membership message time-out to indicate that no hosts remain that want to receive multicast data for a particular group.

Note The software ignores the configuration of the last member query interval when you enable the fast leave feature because it does not check for remaining hosts.

IGMPv3

The IGMPv3 snooping implementation on Cisco Nexus 1000V supports full IGMPv3 snooping, which provides constrained flooding based on the (S, G) information in the IGMPv3 reports. This source-based filtering enables the switch to constrain multicast traffic to a set of ports based on the source that sends traffic to the multicast group.

By default, the software tracks hosts on each VLAN port. The explicit tracking feature provides a fast leave mechanism. Because every IGMPv3 host sends membership reports, report suppression limits the amount of traffic that the switch sends to other multicast capable routers. When report suppression is enabled, and no IGMPv1 or IGMPv2 hosts requested the same group, the software provides proxy reporting. The proxy feature builds the group state from membership reports from the downstream hosts and generates membership reports in response to queries from upstream queries.

Even though the IGMPv3 membership reports provide a full accounting of group members on a LAN segment, when the last host leaves, the software sends a membership query. You can configure the parameter last member query interval. If no host responds before the time-out, the software removes the group state.

IGMP Snooping Query Feature

When the multicast traffic does not need to be routed, you must configure an external switch to query membership. On the external switch, define the query feature in a VLAN that contains multicast sources and receivers but no other active query feature.

When an IGMP snooping query feature is enabled, it sends out periodic IGMP queries that trigger IGMP report messages from hosts wanting to receive IP multicast traffic. IGMP snooping listens to these IGMP reports to identify accurate forwarding.

Prerequisites for IGMP Snooping

IGMP snooping has the following prerequisites:

- You are logged in to the switch.
- A querier must be running on the uplink switches on the VLANs that contain multicast sources and receivers.

Default Settings

Table 5-1 lists the default settings for IGMP snooping parameters.

Table 5-1 Default IGMP Snooping Parameters

Parameters	Default
IGMP snooping	Enabled
IGMPv3 Explicit tracking	Enabled
IGMPv2 Fast leave	Disabled
Last member query interval	1 second

Table 5-1 Default IGMP Snooping Parameters (continued)

Parameters	Default
Snooping querier	Disabled
IGMPv1/v2 Report suppression	Enabled
IGMPv3 Report suppression	Disabled

Configuring IGMP Snooping

Use this procedure to configure IGMP snooping.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.

Note Be aware that the NX-OS commands may differ from those used in Cisco IOS.

- Table 5-2 lists and describes the configurable IGMP snooping parameters.

Table 5-2 IGMP Snooping Parameters

Parameter	Description
IGMP snooping	Enables IGMP snooping globally or on a per-VLAN basis. The default is enabled. **Note** If the global setting is disabled, then all VLANs are treated as disabled, whether they are enabled or not.
Explicit tracking	Tracks IGMPv3 membership reports from individual hosts for each port on a per-VLAN basis. The default is enabled.
Fast leave	Enables the software to remove the group state when it receives an IGMP Leave report without sending an IGMP query message. This parameter is used for IGMPv2 hosts when no more than one host is present on each VLAN port. The default is disabled.
Last member query interval	Sets the interval the software waits after sending an IGMP query to verify that no hosts that want to receive a particular multicast group remain on a network segment. If no hosts respond before the last member query interval expires, the software removes the group from the associated VLAN port. Values range from 1 to 25 seconds. The default is 1 second.
Report suppression	Limits the membership report traffic sent to multicast-capable routers. When you disable report suppression, all IGMP reports are sent as is to multicast-capable routers. The default is enabled.

Table 5-2 IGMP Snooping Parameters (continued)

Parameter	Description
Multicast router	Configures a static connection to a multicast router. The interface to the router must be in the selected VLAN.
Static group	Configures a Layer 2 port of a VLAN as a static member of a multicast group.

Note Be aware that the NX-OS commands may differ from those used in Cisco IOS.

SUMMARY STEPS

1. **config t**
2. **ip igmp snooping**
3. **vlan** *vlan-id*
4. **ip igmp snooping**

   ip igmp snooping explicit-tracking

   ip igmp snooping fast-leave

   ip igmp snooping last-member-query-interval *seconds*

   ip igmp snooping report-suppression

   ip igmp snooping mrouter interface *interface*

   ip igmp snooping static-group *group-ip-addr* **interface** *interface*
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**ip igmp snooping**
	
	Example:
	`n1000v(config)# ip igmp snooping`
	`n1000v(config)#`
		Enables IGMP snooping in the running configuration. The default is enabled.
		Note If disabled, then IGMP snooping on all VLANs is disabled, whether IGMP snooping is enabled on a VLAN or not.

Step 3	**vlan** *vlan-id*
	
	Example:
	`n1000v(config)# vlan 2`
	`n1000v(config-vlan)#`
		Places you in CLI Global Configuration mode for the specified VLAN.

	Command	Purpose
Step 4	**ip igmp snooping** **Example:** `n1000v(config-vlan)# ip igmp snooping`	Enables IGMP snooping for the specific VLAN in the running configuration. The default is disabled.

	ip igmp snooping explicit-tracking
	
	Example:
	`n1000v(config-vlan)# ip igmp snooping explicit-tracking`
	`n1000v(config-vlan)#`
		Tracks IGMPv3 membership reports from individual hosts for each port on a per-VLAN basis in the running configuraiton.
		The default is enabled on all VLANs.

	ip igmp snooping fast-leave
	
	Example:
	`n1000v(config-vlan)# ip igmp snooping fast-leave`
	`n1000v(config-vlan)#`
		Enables fast-leave for the specified VLAN in the running configuration.
		Fast-leave supports IGMPv2 hosts that cannot be explicitly tracked because of the host report suppression mechanism of the IGMPv2 protocol.
		When you enable fast leave, the IGMP software assumes that no more than one host is present on each VLAN port.
		The default is disabled for all VLANs.

	`ip igmp snooping last-member-query-interval` *seconds*
	
	Example:
	`n1000v(config-vlan)# ip igmp snooping last-member-query-interval 3`
	`n1000v(config-vlan)#`
		Establishes a time interval in seconds after which the group is removed from the associated VLAN port if no hosts respond to an IGMP query message. This interval is saved in the running configuration
		Allowable intervals are from 1 (default) to 25 seconds.

	ip igmp snooping report-suppression
	
	Example:
	`n1000v(config-vlan)# ip igmp snooping report-suppression`
	`n1000v(config-vlan)#`
		Limits the membership report traffic sent to multicast-capable routers in the running configuration.
		When you disable report suppression, all IGMP reports are sent as is to multicast-capable routers. The default is enabled.

	ip igmp snooping mrouter interface *interface*
	
	Example:
	`n1000v(config-vlan)# ip igmp snooping mrouter interface ethernet 2/1`
	`n1000v(config-vlan)#`
		Configures a static connection to a multicast router in the running configuraiton.
		The interface to the router must be in the specified VLAN. You can specify the interface by the type and the number, such as **ethernet** *slot/port*.

	ip igmp snooping static-group *group-ip-addr* **interface** *interface*
	
	Example:
	`n1000v(config-vlan)# ip igmp snooping static-group 230.0.0.1 interface ethernet 2/1`
	`n1000v(config-vlan)#`
		Configures a Layer 2 port of a VLAN as a static member of a multicast group in the running configuration.
		You can specify the interface by the type and the number, such as **ethernet** *slot/port*.

Step 5	**copy running-config startup-config** **Example:** `n1000v# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Verifying the IGMP Snooping Configuration

To display the IGMP snooping configuration information, perform one of the following tasks:

Command	Purpose
show ip igmp snooping [**vlan** *vlan-id*]	Displays IGMP snooping configuration by VLAN.
show ip igmp snooping groups [**vlan** *vlan-id*] [**detail**]	Displays IGMP snooping information about groups by VLAN.
show ip igmp snooping querier [**vlan** *vlan-id*]	Displays IGMP snooping queriers by VLAN.
show ip igmp snooping mroute [**vlan** *vlan-id*]	Displays multicast router ports by VLAN.
show ip igmp snooping explicit-tracking [**vlan** *vlan-id*]	Displays IGMP snooping explicit tracking information by VLAN.

For detailed information about commands and their output, see the *Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(2)*.

Example Configuration for IGMP Snooping

The following example shows how to configure the IGMP snooping parameters:

```
n1000v# config t
n1000v(config)# ip igmp snooping
n1000v(config)# vlan 2
n1000v(config-vlan)# ip igmp snooping
n1000v(config-vlan)# ip igmp snooping explicit-tracking
n1000v(config-vlan)# ip igmp snooping report-suppression
n1000v(config-vlan)# ip igmp snooping mrouter interface ethernet 2/1
n1000v(config-vlan)# ip igmp snooping static-group 230.0.0.1 interface ethernet 2/1
n1000v(config-vlan)# copy run start
[########################################] 100%
n1000v(config-vlan)# exit
n1000v(config)# exit
n1000v#
```

Additional References

For additional information related to implementing IGMP snooping, see the following sections:

- Related Documents, page 5-8
- Standards, page 5-8

Related Documents

Related Topic	Document Title
Port Profiles	*Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(2)*
Interfaces	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(2)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(2)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for IGMP Snooping

This section provides the release history for the IGMP snooping feature.

Table 5-3

Feature Name	Releases	Feature Information
IGMP Snooping	4.0(4)SV1(1)	This feature was introduced.

CHAPTER 6

Layer 2 Switching Configuration Limits

This section lists the Cisco Nexus 1000V Layer 2 Switching configuration limits.

Table 6-1 Layer 2 Switching Configuration Limits

Layer 2 Feature	Maximum Limit
Active VLANs across all VEMs	512
MACs over VLAN within a VEM	1024 (1K)
PVLANs across all VEMs	512
Physical Trunks per DVS	512

INDEX

N

P

R

S

V