



Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.2(1)SV2(2.1)

First Published: June 21, 2013

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-28786-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <http://www.cisco.com/go/trademarks>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2009-2013 Cisco Systems, Inc. All rights reserved.



CONTENTS

Preface

Preface vii

Audience vii

Document Conventions vii

Related Documentation for Nexus 1000V Series NX-OS Software for VMware vSphere ix

Documentation Feedback x

Obtaining Documentation and Submitting a Service Request x

CHAPTER 1

New and Changed Information 1

New and Changed Information 1

CHAPTER 2

Overview 3

Information about Layer 2 Switching 3

VEM Port Model 3

VEM Virtual Ports 4

VEM Physical Ports 5

VSM Port Model 6

Switching Traffic Between VEMs 7

Layer 2 Ethernet Switching 7

MAC Address Tables 8

VLANs 8

Private VLANs 8

IGMP Snooping 8

CHAPTER 3

Configuring MAC Address Tables 11

Information About MAC Address Tables 11

Guidelines and Limitations 12

Default Settings 12

| | |
|---|----|
| Configuring the MAC Address Table | 12 |
| Configuring a Static MAC Address | 12 |
| Configuring the Aging Time | 13 |
| Clearing Dynamic Addresses from the MAC Address Table | 14 |
| Verifying the MAC Address Table Configuration | 15 |
| Configuration Example for MAC Address Tables | 15 |
| Feature History for MAC Address Tables | 16 |

CHAPTER 4**Configuring VLANs 17**

| | |
|----------------------------------|----|
| Information About VLANs | 17 |
| Guidelines and Limitations | 18 |
| Default Settings | 19 |
| Configuring a VLAN | 20 |
| Creating a VLAN | 20 |
| Configuring VLAN Characteristics | 22 |
| Verifying the Configuration | 24 |
| Feature History for VLANs | 24 |

CHAPTER 5**Configuring Private VLANs 25**

| | |
|--|----|
| Information About Private VLANs | 25 |
| Private VLAN Ports | 26 |
| Communication Between Private VLAN Ports | 28 |
| Guidelines and Limitations | 28 |
| Default Settings | 28 |
| Configuring a Private VLAN | 28 |
| Enabling or Disabling the Private VLAN Feature Globally | 29 |
| Configuring a VLAN as a Primary VLAN | 30 |
| Configuring a VLAN as a Secondary VLAN | 31 |
| Associating the VLANs in a PVLAN | 32 |
| Configuring a Private VLAN Host Port | 33 |
| Associating a vEthernet Port Profile with a Private VLAN | 34 |
| Configuring a Layer 2 Port Profile as a Promiscuous Trunk Port | 35 |
| Configuring a Private VLAN Promiscuous Access Port | 36 |
| Associating a Promiscuous Access Port with a Private VLAN | 38 |
| Removing a Private VLAN Configuration | 39 |

| | |
|--|----|
| Verifying a Private VLAN Configuration | 40 |
| Configuration Examples for Private VLANs | 40 |
| Feature History for Private VLANs | 42 |

CHAPTER 6**Configuring IGMP Snooping 43**

| | |
|--|----|
| Information About IGMP Snooping | 43 |
| Introduction | 43 |
| IGMPv1 and IGMPv2 | 44 |
| IGMPv3 | 45 |
| Prerequisites for IGMP Snooping | 45 |
| Default Settings | 45 |
| Configuring IGMP Snooping | 46 |
| Enabling or Disabling IGMP Snooping Globally for the VSM | 46 |
| Configuring IGMP Snooping on a VLAN | 47 |
| Verifying the IGMP Snooping Configuration | 49 |
| Example Configuration IGMP Snooping | 49 |
| Feature History for IGMP Snooping | 50 |

CHAPTER 7**Configuring Network Load Balancing for vEthernet 51**

| | |
|--|----|
| Information About Microsoft Network Load Balancing | 51 |
| Guidelines and Limitations | 51 |
| Configuring Microsoft Network Load Balancing Support in Interface Configuration Mode | 52 |
| Configuring Microsoft Network Load Balancing in Port Profile Configuration Mode | 53 |
| Feature History for Microsoft Network Load Balancing for vEthernet | 55 |

CHAPTER 8**Supporting Redundant Routing Protocols 57**

| | |
|--|----|
| Information About Redundant Routing Protocols | 57 |
| Guidelines and Limitations | 57 |
| Supporting Redundant Routing Protocols | 58 |
| Configuring a vEthernet Interface to Support Redundant Routing Protocols | 58 |
| Configuring a Port Profile to Support Redundant Routing Protocols | 59 |
| Feature History for Supporting Redundant Routing Protocol | 62 |

CHAPTER 9**Layer 2 Switching Configuration Limits 63**

| | |
|--|----|
| Layer 2 Switching Configuration Limits | 63 |
|--|----|



Preface

This preface contains the following sections:

- [Audience, page vii](#)
- [Document Conventions, page vii](#)
- [Related Documentation for Nexus 1000V Series NX-OS Software for VMware vSphere, page ix](#)
- [Documentation Feedback , page x](#)
- [Obtaining Documentation and Submitting a Service Request, page x](#)

Audience

This publication is for experienced network administrators who configure and maintain Cisco Nexus devices. This guide is for network and server administrators with the following experience and knowledge:

- An understanding of virtualization
- Using VMware software to create a virtual machine and configure a VMware vSwitch



Note

Knowledge of VMware vNetwork Distributed Switch is not required.

Document Conventions

Command descriptions use the following conventions:

| Convention | Description |
|---------------|--|
| bold | Bold text indicates the commands and keywords that you enter literally as shown. |
| <i>Italic</i> | Italic text indicates arguments for which the user supplies the values. |

| Convention | Description |
|-----------------|---|
| [x] | Square brackets enclose an optional element (keyword or argument). |
| [x y] | Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice. |
| {x y} | Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice. |
| [x {y z}] | Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element. |
| <i>variable</i> | Indicates a variable for which you supply values, in context where italics cannot be used. |
| string | A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks. |

Examples use the following conventions:

| Convention | Description |
|--|---|
| <code>screen font</code> | Terminal sessions and information the switch displays are in screen font. |
| <code>boldface screen font</code> | Information you must enter is in boldface screen font. |
| <i><code>italic screen font</code></i> | 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 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 for Nexus 1000V Series NX-OS Software for VMware vSphere

This section lists the documents used with the Cisco Nexus 1000V and available on Cisco.com at the following URL:

http://www.cisco.com/en/US/products/ps9902/tsd_products_support_series_home.html

General Information

Cisco Nexus 1000V Documentation Roadmap

Cisco Nexus 1000V Release Notes

Cisco Nexus 1000V and VMware Compatibility Information

Install and Upgrade

Cisco Nexus 1000V Installation and Upgrade Guide

Configuration Guides

Cisco Nexus 1000V High Availability and Redundancy Configuration Guide

Cisco Nexus 1000V Interface Configuration Guide

Cisco Nexus 1000V Layer 2 Switching Configuration Guide

Cisco Nexus 1000V License Configuration Guide

Cisco Nexus 1000V Network Segmentation Manager Configuration Guide

Cisco Nexus 1000V Port Profile Configuration Guide

Cisco Nexus 1000V Quality of Service Configuration Guide

Cisco Nexus 1000V REST API Plug-in Configuration Guide

Cisco Nexus 1000V Security Configuration Guide

Cisco Nexus 1000V System Management Configuration Guide

Cisco Nexus 1000V vCenter Plugin Configuration Guide

Cisco Nexus 1000V VXLAN Configuration Guide

Programming Guide

Cisco Nexus 1000V XML API Configuration Guide

Reference Guides

Cisco Nexus 1000V Command Reference

Cisco Nexus 1000V Resource Availability Reference

Troubleshooting and Alerts

Cisco Nexus 1000V Troubleshooting Guide

Cisco Nexus 1000V Password Recovery Procedure

Cisco NX-OS System Messages Reference

Cloud Services Platform Documentation

The *Cisco Cloud Services Platform* documentation is available at http://www.cisco.com/en/US/products/ps12752/tsd_products_support_series_home.html.

Virtual Security Gateway Documentation

The *Cisco Virtual Security Gateway for Nexus 1000V Series Switch* documentation is available at http://www.cisco.com/en/US/products/ps13095/tsd_products_support_series_home.html.

Virtual Wide Area Application Services (vWAAS) Documentation

The *Virtual Wide Area Application Services* documentation is available at http://www.cisco.com/en/US/products/ps6870/tsd_products_support_series_home.html.

ASA 1000V Cloud Firewall Documentation

The *ASA 1000V Cloud Firewall* documentation is available at http://www.cisco.com/en/US/products/ps12233/tsd_products_support_series_home.html.

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus1k-docfeedback@cisco.com. We appreciate your feedback.

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

New and Changed Information

This chapter contains the following sections:

- [New and Changed Information, page 1](#)

New and Changed Information

This section lists new and changed content in this document by software release.

To find additional information about new features or command changes, see the *Cisco Nexus 1000V Release Notes* and *Cisco Nexus 1000V Command Reference*.

Table 1: New and Changed Features for the Cisco Nexus 1000V Layer 2 Switching Configuration Guide

| Feature | Description | Changed in Release | Where Documented |
|--|--|--------------------|--|
| Supporting Redundant Routing Protocols | Added support for redundant routing protocols. | 4.2(1)SV1(5.1) | Supporting Redundant Routing Protocols, on page 57 |
| Network Load Balancing | Added the ability to configure network load balancing for vEthernet. | 4.2(1)SV1(5.1) | Configuring Network Load Balancing for vEthernet, on page 51 |

| Feature | Description | Changed in Release | Where Documented |
|--|--|--------------------|--|
| Layer 2 Configuration Limits | Increased configuration limits for active VLANs across all VEMs, MAC addresses over VLANs within a VEM, and MAC addresses per VLAN within a VEM. | 4.2(1)SV1(4) | Layer 2 Switching Configuration Limits, on page 63 |
| IGMP link-local group suppression | Added support to enable or disable link-local group suppression. | 4.2(1)SV1(4) | Configuring IGMP Snooping, on page 43 |
| clear mac address-table command | Removed address, interface, and port channel options. | 4.2(1)SV1(4) | Configuring MAC Address Tables, on page 11 |
| show mac address-table command | Updated show command output. | 4.2(1)SV1(4) | Configuring MAC Address Tables, on page 11 |
| feature private-vlan command | Added the ability to globally enable the private VLAN feature. | 4.2(1)SV1(4) | Configuring Private VLANs, on page 25 |
| 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) | Layer 2 Switching Configuration Limits, on page 63 |



CHAPTER 2

Overview

This chapter contains the following sections:

- [Information about Layer 2 Switching, page 3](#)
- [Layer 2 Ethernet Switching, page 7](#)
- [MAC Address Tables, page 8](#)
- [VLANs, page 8](#)
- [Private VLANs, page 8](#)
- [IGMP Snooping, page 8](#)

Information about Layer 2 Switching

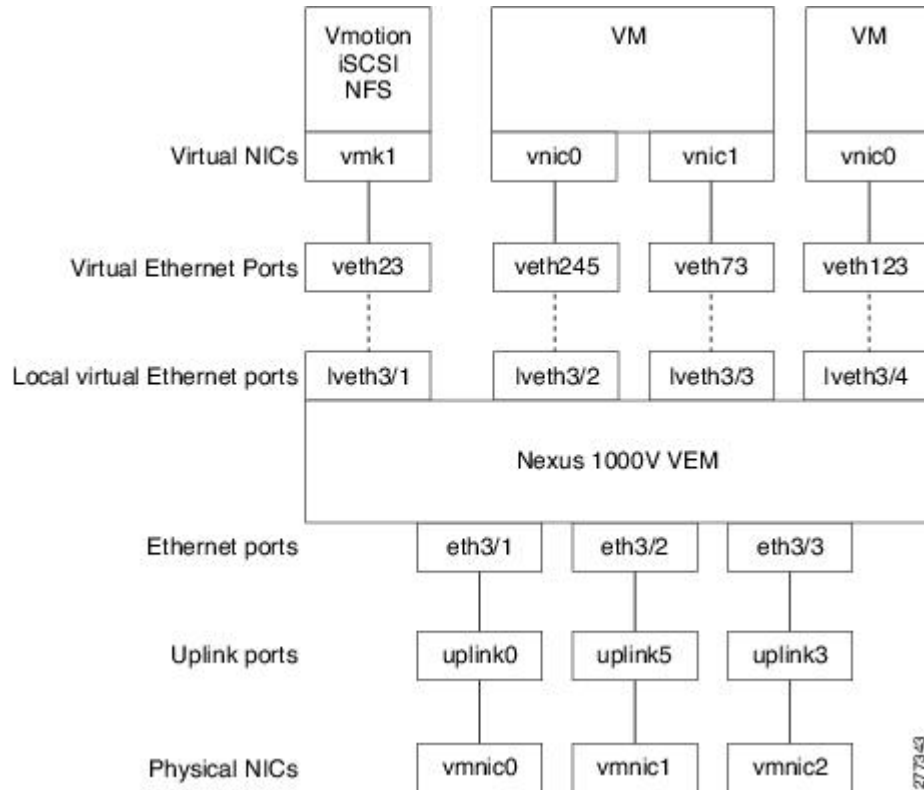
VEM Port Model

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

- VEM Virtual Ports

- VEM Physical Ports

Figure 1: VEM Port View



VEM Virtual Ports

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

Virtual NICs

There are three types of virtual NICs (vNICs). The vNIC is part of the Virtual Machine (VM), and represents the physical port of the host that is plugged into the switch. The internal NIC is used by the hypervisor for management, VMotion, NFS, iSCSI, and other network access. 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 Virtual Ethernet port within the Cisco Nexus 1000V.

Virtual Ethernet Ports

A virtual Ethernet port (vEth) represents a port on the Cisco Nexus 1000V Distributed Virtual Switch. The Cisco Nexus 1000V has a flat space of vEth ports, 1...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

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



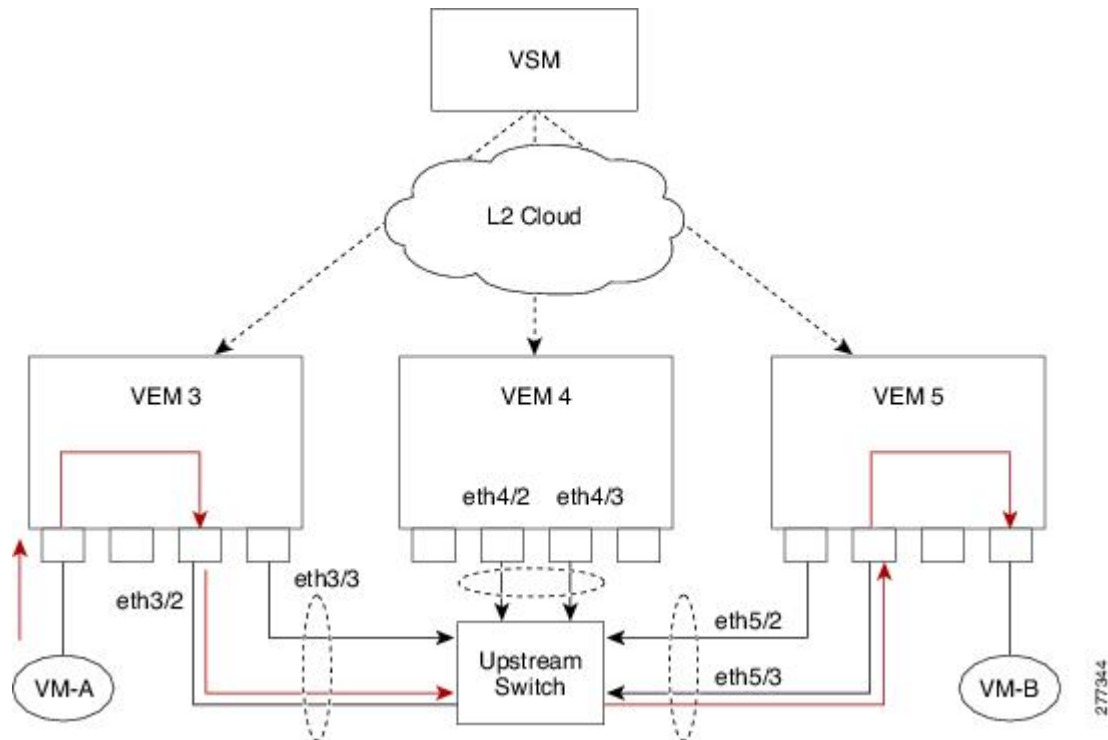
Note

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. These numbers 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

The following figure shows the VSM view of the network.

Figure 2: VSM View



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

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.

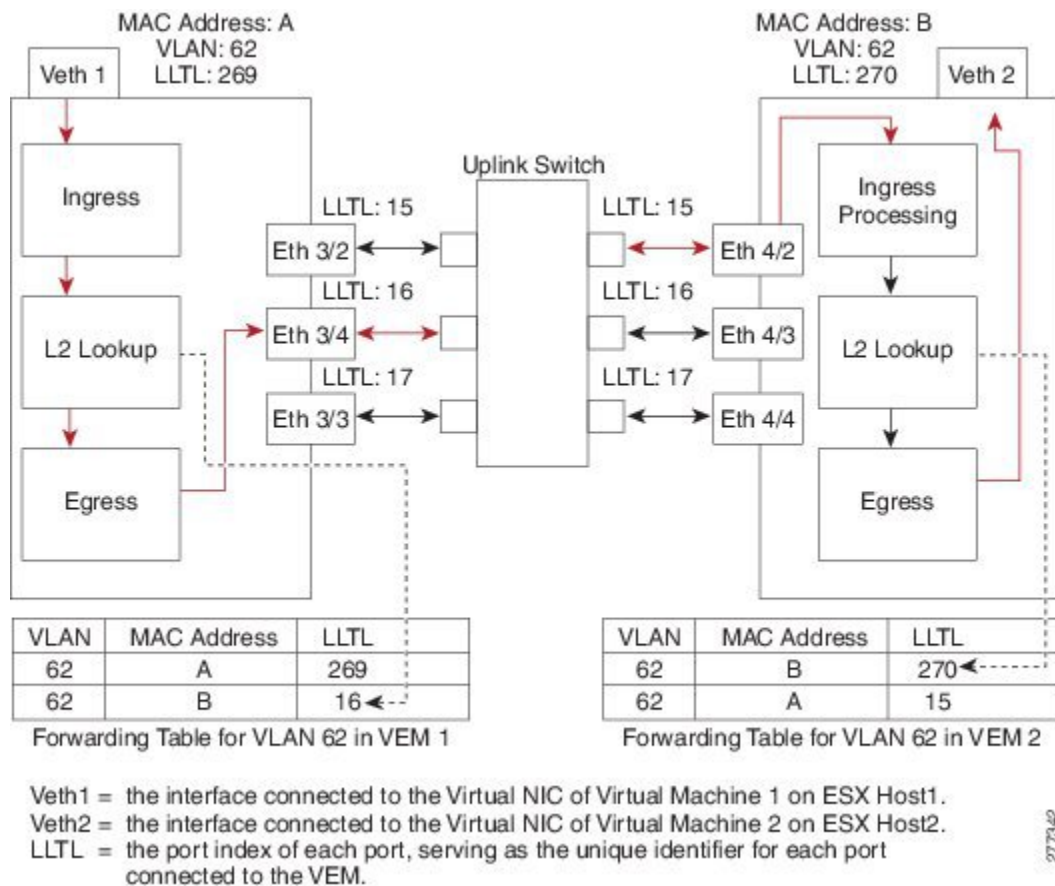
Port Channel Interfaces

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 that is 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 that are connected to the VEM.

Figure 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. 10/100-Mbps Ethernet 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 mode 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 address table is maintained. The MAC address of the sending network is associated with the LAN port on which it was received.

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.

Up to 4094 VLANs are supported 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 on the Cisco Nexus 1000V.

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. The use of larger subnets reduces address management overhead.

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.



Configuring MAC Address Tables

This chapter contains the following sections:

- [Information About MAC Address Tables, page 11](#)
- [Guidelines and Limitations, page 12](#)
- [Default Settings, page 12](#)
- [Configuring the MAC Address Table, page 12](#)
- [Verifying the MAC Address Table Configuration, page 15](#)
- [Configuration Example for MAC Address Tables, page 15](#)
- [Feature History for MAC Address Tables, page 16](#)

Information About MAC Address Tables

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. 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 if you copy the static MAC addresses configuration to the startup configuration by using the `copy running-config startup-config` command.

The address table per VEM can store up to 32,000 MAC entries. An aging timer triggers removal of addresses from the table when they remain inactive for the default time of 300 seconds. The aging timer can be configured on a global basis but not per VLAN.

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

Guidelines and Limitations

- The forwarding table for each VLAN in a VEM can store up to 4096 MAC addresses.
- You can configure only 32 static MAC addresses on a single interface and 1024 static MAC addresses DVS wide.
- Cisco Nexus 1000V supports a maximum of 2000 private VLAN MAC addresses on a VSM.

Default Settings

Table 2: Default MAC Address Aging Time

| Parameters | Default |
|------------|--------------|
| Aging time | 1800 seconds |

Configuring the MAC Address Table

Configuring a Static MAC Address

You can configure a MAC address to statically point to a specific interface.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know that you cannot configure broadcast or multicast addresses as static MAC addresses.
- Know that static MAC addresses override dynamically learned MAC addresses on an interface.



Note

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

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# mac address-table static mac_address vlan vlan-id {[drop interface {typeif_id} port-channel number]} | Adds a static MAC address in the Layer 2 MAC address table and saves it in the running configuration. |

| | Command or Action | Purpose |
|---------------|--|--|
| | | The interface can be specified as either of the following: <ul style="list-style-type: none"> • ethernet <i>slot/port</i> • veth <i>number</i> |
| Step 3 | switch(config)# show mac address static interface [type if_id] | (Optional) Displays static MAC addresses. |
| Step 4 | switch(config)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure a static MAC address:

```
switch# configure terminal
switch(config)# mac address static
switch(config)# show mac address static 12ab.47dd.ff89 vlan 3
interface ethernet 3/3
VLAN      MAC Address      Type    Age    Port                               Mod
-----+-----+-----+-----+-----+-----+-----
1         0002.3d11.5502   static  0      N1KV Internal Port                3
1         0002.3d21.5500   static  0      N1KV Internal Port                3
1         0002.3d21.5502   static  0      N1KV Internal Port                3
1         0002.3d31.5502   static  0      N1KV Internal Port                3
1         0002.3d41.5502   static  0      N1KV Internal Port                3
1         0002.3d61.5500   static  0      N1KV Internal Port                3
1         0002.3d61.5502   static  0      N1KV Internal Port                3
1         0002.3d81.5502   static  0      N1KV Internal Port                3
3         12ab.47dd.ff89   static  0      Eth3/3                             3
342      0002.3d41.5502   static  0      N1KV Internal Port                3
343      0002.3d21.5502   static  0      N1KV Internal Port                3
Total MAC Addresses: 11
n1000v(config)# show mac address static interface Ethernet 3/3
VLAN      MAC Address      Type    Age    Port                               Module
-----+-----+-----+-----+-----+-----
3         12ab.47dd.ff89   static  0      Eth3/3                             3
Total MAC Addresses: 1
switch(config)#
```

Configuring the Aging Time

You can configure the amount of time that packet source MAC addresses, and the ports on which they are learned, remain in the MAC table.



Note

The aging time is a global setting that cannot be configured per VLAN. Although it is a global setting, you can also configure the MAC aging time in interface configuration mode or VLAN configuration mode.

Before You Begin

Log in to the CLI in EXEC mode.

**Note**

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

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch# mac address-table aging-time seconds | 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 are as follows: <ul style="list-style-type: none"> • 120 to 918000 seconds (default is 300) • If you specify zero (0), MAC aging is disabled. |

This example shows how to configure the aging time:

```
switch# configure terminal
switch(config)# mac address-table aging-time 600
switch(config)# show mac address-table aging-time
Vlan Aging Time
-----
101    300
100    300
1      300
switch#
```

Clearing Dynamic Addresses from the MAC Address Table

Before You Begin

Log in to the CLI in EXEC mode.

**Note**

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

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | switch# clear mac address-table dynamic [vlan vlan_id] | Clears the dynamic address entries from the Layer 2 MAC address table. |
| Step 2 | switch# show mac address-table | (Optional) Displays the MAC address table. |

This example shows how to clear the entire MAC address table of all dynamic entries:

```
switch# clear mac address-table dynamic
switch#
```

This example shows how to clear the MAC address table of only those dynamic MAC addresses learned on VLAN 5:

```
switch# clear mac address-table dynamic vlan 5
switch#
```

Verifying the MAC Address Table Configuration

Use one of the following commands to verify the configuration:

| Command | Purpose |
|--|---|
| show mac address-table | Displays the MAC address table. |
| show mac address-table module | Use the show mac address-table module <i>mod id</i> command to debug a specific module. |
| show mac address-table static | Displays information about the MAC address table static entries. |
| show mac address-table static inc veth | Displays the static MAC address of vEthernet interfaces in case a VEM physical port learns a dynamic MAC and the packet source is in another VEM on the same VSM. |
| show mac address static interface [type <i>if_id</i>] | Displays all static MAC addresses. |
| show mac address-table aging-time | Displays the aging time in the MAC address table. |
| show mac address-table count | Displays a count of MAC address entries. |
| show interface <i>interface_id</i> mac | Displays the MAC addresses and the burned-in MAC address for an interface. |

Configuration Example for MAC Address Tables

This example shows how to add a static MAC address and establish a global aging time:

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

Feature History for MAC Address Tables

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



CHAPTER 4

Configuring VLANs

This chapter contains the following sections:

- [Information About VLANs, page 17](#)
- [Guidelines and Limitations, page 18](#)
- [Default Settings, page 19](#)
- [Configuring a VLAN, page 20](#)
- [Verifying the Configuration, page 24](#)
- [Feature History for VLANs, page 24](#)

Information About VLANs

Physical NICs are always assigned as trunk ports that 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 that is connected to the Virtual Machine (VM). |

Physical ports are always trunk ports by default. The VM interfaces can be either access ports or trunk ports.

vEthernet interfaces that are 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, is transmitted as untagged on the physical NIC interfaces. When the VLAN is not specified, it is assumed to be 1.

The following table summarizes the actions taken on packets that are received by the Virtual Ethernet Module (VEM) based on VLAN tagging.

Table 3: VEM Action on VLAN Tagging

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

Guidelines and Limitations

In accordance with the IEEE 802.1Q standard, up to 4094 VLANs (from 1 to 4094) are supported in the Cisco Nexus 1000V, and are listed in the following table.



Note

For VLAN configuration limits, see [Layer 2 Switching Configuration Limits](#), on page 63.

Table 4: Cisco Nexus 1000V VLAN Numbering

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

| VLAN Numbers | Range | Usage |
|-----------------------|----------------------|--|
| 1006 to 4094 | Extended | <p>You can create, name, or use these VLANs. You cannot change the following parameters:</p> <ul style="list-style-type: none"> • The state is always active. • These VLANs are always enabled. You cannot shut down these VLANs. <p>The extended system ID is always automatically enabled.</p> |
| 3968 to 4047 and 4094 | Internally allocated | <p>You cannot use, create, delete, or modify these VLANs. You can display these VLANs.</p> <p>The Cisco Nexus 1000V allocates these 80 VLANs, plus VLAN 4094, for features, like diagnostics, that use internal VLANs for their operation.</p> |

Default Settings

Table 5: Default VLAN Settings

| 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 |
| IGMP snooping | Enabled |

Configuring a VLAN

Creating a VLAN

You can do one of the following:

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



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

Before You Begin

- Log in to the CLI in EXEC mode.
- Know that VLAN characteristics are configured in the VLAN configuration mode. To configure a VLAN that is already created, see [Configuring VLAN Characteristics, on page 22](#).
- Be familiar with the VLAN numbering in the [Guidelines and Limitations, on page 18](#).
- Know that newly created VLANs remain unused until Layer 2 ports are assigned to them.
- Know that 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 reenables, or re-creates, 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 reenables.



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

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# show vlan | Displays the VLANs that already exist. |
| Step 3 | switch(config)# [no] vlan { <i>vlan-id</i> <i>vlan-range</i> } | Creates or deletes, and saves in the running configuration, a VLAN or a range of VLANs. To configure the VLAN, see Configuring VLAN Characteristics, on page 22 . |

| | Command or Action | Purpose |
|---------------|--|---|
| | | <p>Note If you enter a VLAN ID that is assigned to an internally allocated VLAN, the system returns an error message. From the VLAN configuration mode, you can also create and delete VLANs.</p> <p>For information about Assigning Layer 2 interfaces to VLANs (access or trunk ports), see the <i>Cisco Nexus 1000V Interface Configuration Guide</i>.</p> <p>For information about configuring ports as VLAN access or trunk ports and assigning ports to VLANs, see the <i>Cisco Nexus 1000V Interface Configuration Guide</i>.</p> |
| Step 4 | switch(config-vlan)# show vlan id <i>vlan-id</i> | (Optional) Displays the VLAN configuration. |
| Step 5 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

In this example, VLAN 5 is created and you are automatically placed into the VLAN configuration mode for VLAN 5:

```
switch# configure terminal
switch(config)# vlan 5
switch(config-vlan)#
```

This example shows the range, VLAN 15 to 20, being created. The VLANs in the range are activated, and you are automatically placed into VLAN configuration mode for VLANs 15 to 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.

```
switch# configure terminal
switch(config)# vlan 15-20
switch(config-vlan)#
```

This example shows how to delete VLAN 3967:

```
switch# configure terminal
switch(config)# no vlan 3967
switch(config)#
```

This example shows how to display the VLAN 5 configuration:

```
switch# configure terminal
switch(config)# vlan 5
switch(config-vlan)# 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-vlan)# copy run start
[#####] 100%
n1000v(config)#

```

Configuring VLAN Characteristics

You can do the following for a VLAN that has already been created:



Note

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

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

Before You Begin

Log in to the CLI in EXEC mode.



Note

Some characteristics cannot be modified on some VLANs. For more information, see the VLAN numbering described in the [Guidelines and Limitations, on page 18](#).

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# vlan {vlan-id vlan-range} | Enters VLAN configuration mode for the specified VLAN. Note If the VLAN does not already exist, the system creates it and then enters the VLAN configuration mode for that VLAN. |
| Step 3 | switch(config-vlan)# name vlan-name | Adds a name to the VLAN of up to 32 alphanumeric characters. <ul style="list-style-type: none"> • You cannot change the name of VLAN1 or the VLANs that are reserved for internal use. • The default name is VLANxxxx where xxxx represent four numeric digits (including leading zeroes) equal to the VLAN ID number. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 4 | switch(config-vlan)# state { active suspend } | <p>Changes the operational state of the VLAN and saves it in the running configuration.</p> <p>Allowable entries are as follows:</p> <ul style="list-style-type: none"> • active (default) • suspend <p>While the VLAN state is suspended, the ports associated with this VLAN are shut down, and that VLAN does not pass any traffic.</p> <p>Note You cannot suspend the state for the default VLAN or VLANs 1006 to 4094.</p> |
| Step 5 | switch(config-vlan)# no shutdown | <p>Enables VLAN switching in the running configuration.</p> <p>Allowable entries are as follows:</p> <ul style="list-style-type: none"> • no shutdown (default) • shutdown <p>Note You cannot shut down the default VLAN, VLAN1, or VLANs 1006 to 4094.</p> |
| Step 6 | switch(config-vlan)# show vlan [id <i>vlan-id</i>] | (Optional) Displays the VLAN configuration. |
| Step 7 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure VLAN characteristics:

```
n1000v# configure terminal
n1000v(config)# vlan 5
n1000v(config-vlan)# name accounting
n1000v(config-vlan)# state active
n1000v(config-vlan)# no shutdown
n1000v(config-vlan)# 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 the Configuration

Use one of the following commands to verify the configuration:

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

Feature History for VLANs

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



CHAPTER 5

Configuring Private VLANs

This chapter contains the following sections:

- [Information About Private VLANs, page 25](#)
- [Private VLAN Ports, page 26](#)
- [Communication Between Private VLAN Ports, page 28](#)
- [Guidelines and Limitations, page 28](#)
- [Default Settings, page 28](#)
- [Configuring a Private VLAN, page 28](#)
- [Verifying a Private VLAN Configuration, page 40](#)
- [Configuration Examples for Private VLANs, page 40](#)
- [Feature History for Private VLANs, page 42](#)

Information About Private VLANs

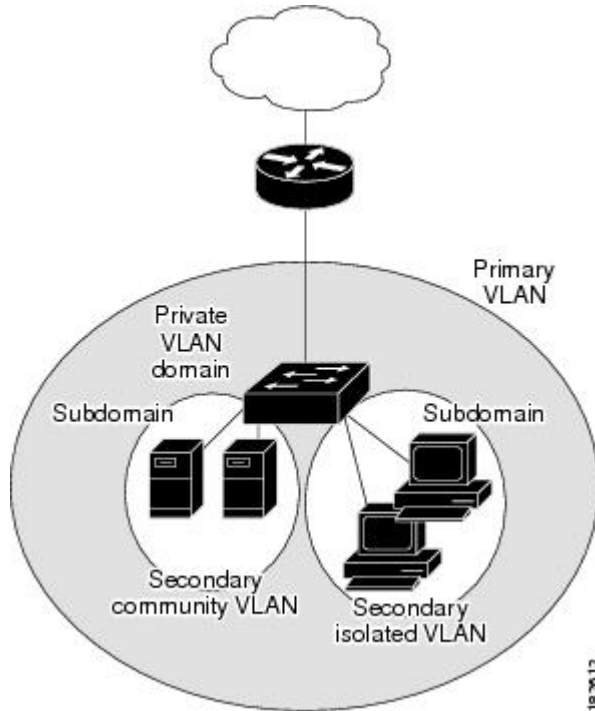
PVLANs achieve device isolation through the use of three separate port designations, each having its own unique set of rules that regulate each connected endpoint's ability to communicate with other connected endpoints within the same private VLAN domain.

Private VLAN Domains

A PVLAN 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 the following figure.

Figure 4: Private VLAN Domain



Spanning Multiple Switches

PVLANS can span multiple switches, just like regular VLANs. Inter-switch link ports do not need to be aware of the special VLAN type and carry frames tagged with these VLANs just like they do any other frames. PVLANS 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 with the packet, it is possible to maintain consistent behavior throughout the network. The mechanism that 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 PVLAN domain, there are three separate port designations. Each port designation has its own unique set of rules that regulate the ability of one endpoint to communicate with other connected endpoints within the same private VLAN domain. The three port designations are as follows:

- promiscuous
- isolated
- community

Primary VLANs and Promiscuous Ports

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

A promiscuous port can talk to all other types of ports; it 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 PVLAN domain. A PVLAN domain can have one or more subdomains. A subdomain is made up of a VLAN pair that consists of the primary VLAN and a secondary VLAN. Because the primary VLAN is a part of every subdomain, secondary VLANs differentiate the VLAN subdomains.

To communicate to the Layer 3 interface, you must associate a secondary VLAN 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 PVLAN 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 the above figure) 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, 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.

An isolated VLAN 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 used to provide this port isolation.



Note While multiple community VLANs can be in a private VLAN domain, one isolated VLAN can serve multiple customers. All endpoints that are 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 that share the same isolated VLAN.

- **Community VLANs**—Community VLANs use community host ports. A community port (c1 or c2 in the above figure) is part of a group of ports. The ports within a community can communicate at Layer 2 with one another and can also talk to any promiscuous port. For example, if an ISP customer has four devices and wants them isolated from those devices of other customers but still be able to communicate among themselves, community ports should be used.



Note Because trunks can support a VLAN that carries traffic between its ports, VLAN traffic can 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 PVLAN port types.

Table 6: Communication Between PVLAN 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 that traverses 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.

Guidelines and Limitations

PVLANS have the following configuration guidelines and limitations:

Control VLANs, packet VLANs, and management VLANs must be configured as regular VLANs and not as private VLANs.

Default Settings

Table 7: Default PVLAN Settings

| Parameters | Default |
|------------|----------|
| PVLANS | Disabled |

Configuring a Private VLAN

The following section guides you through the private VLAN configuration process. After completing each procedure, return to this section to make sure that you have completed all required procedures in the correct sequence.

Procedure

-
- Step 1** Enable or disable the PVLAN feature globally. See [Enabling or Disabling the Private VLAN Feature Globally, on page 29](#).
 - Step 2** Configure a VLAN as a primary VLAN. See [Configuring a VLAN as a Primary VLAN, on page 30](#).
 - Step 3** Configure a VLAN as a secondary VLAN. See [Configuring a VLAN as a Secondary VLAN, on page 31](#).
 - Step 4** Associate the VLANs in a PVLAN. See [Associating the VLANs in a PVLAN, on page 32](#).
 - Step 5** Configure a PVLAN host port. See [Configuring a Private VLAN Host Port, on page 33](#).
 - Step 6** Associate a host port with a PVLAN. See [Associating a vEthernet Port Profile with a Private VLAN, on page 34](#).
 - Step 7** Verify a PVLAN configuration. See [Verifying a Private VLAN Configuration, on page 40](#).
-

Enabling or Disabling the Private VLAN Feature Globally

You can globally enable or disable the PVLAN feature.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# [no] feature private-vlan | Globally enables or disables the PVLAN feature. |
| Step 3 | switch(config-vlan)# show feature | (Optional) Displays features available and whether they are enabled globally. |
| Step 4 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to enable or disable the PVLAN feature globally:

```
switch# configure terminal
switch(config)# feature private-vlan
switch(config-vlan)# show feature
Feature Name      Instance  State
-----
dhcp-snooping    1        enabled
http-server      1        enabled
ippool           1        enabled
lacp              1        enabled
lisp              1        enabled
lisphelper       1        enabled
netflow          1        disabled
port-profile-roles 1        enabled
```

```

private-vlan      1      enabled
sshServer        1      enabled
tacacs           1      enabled
telnetServer     1      enabled
switch(config-vlan)#

```

Configuring a VLAN as a Primary VLAN

You can configure a VLAN to function as the primary VLAN in a PVLAN.

Before You Begin

- Log in to the CLI in EXEC mode.
- You have already enabled the private VLAN feature using the [Enabling or Disabling the Private VLAN Feature Globally](#), on page 29.
- Know that the VLAN that 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 [Creating a VLAN](#), on page 20.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# vlan <i>primary-vlan-id</i> | Enters VLAN configuration mode for the specified VLAN and configures the primary VLAN ID in the running configuration. |
| Step 3 | switch(config-vlan)# private-vlan primary | Designates the primary VLAN as a private VLAN in the running configuration. |
| Step 4 | switch(config-vlan)# show vlan private-vlan | (Optional) Displays the PVLAN configuration. |
| Step 5 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure a VLAN as a primary VLAN:

```

switch# configure terminal
switch(config)# vlan 202
switch(config-vlan)# private-vlan primary
switch(config-vlan)# show vlan private-vlan
Primary Secondary Type          Ports

```



```
-----
202                primary
switch(config-vlan)#
```

Configuring a VLAN as a Secondary VLAN

You can configure a VLAN to function as the primary VLAN in a PVLAN.

Before You Begin

- Log in to the CLI in EXEC mode.
- You have already enabled the private VLAN feature using the [Enabling or Disabling the Private VLAN Feature Globally](#), on page 29.
- Know that the VLAN that you are configuring as a secondary 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 secondary VLAN. For information about creating a VLAN, see [Creating a VLAN](#), on page 20.

- Know whether you want the secondary VLANs to be community VLANs or isolated VLANs, and the VLAN IDs for each.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# vlan secondary-vlan-id | Enters VLAN configuration mode for the specified VLAN and configures the secondary VLAN ID in the running configuration. |
| Step 3 | switch(config-vlan)# private-vlan {community isolated} | Designates the VLAN as either a community or isolated private VLAN in the running configuration. |
| Step 4 | switch(config-vlan)# show vlan private-vlan | (Optional) Displays the PVLAN configuration. |
| Step 5 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure a VLAN as a secondary VLAN:

```
switch# configure terminal
switch(config)# vlan 202
```

```

switch(config-vlan)# private-vlan community
switch(config-vlan)# show vlan private-vlan
-----
Primary Secondary Type Ports
-----
202 community
switch(config-vlan)#

```

Associating the VLANs in a PVLAN

You can associate the primary VLANs in a PVLAN with the secondary VLANs.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know that the primary VLAN for this PVLAN is already configured as a PVLAN.
- Know that the secondary VLANs for this PVLAN are already configured as PVLANS.
- Know the VLAN IDs for each VLAN that is a part of the PVLAN.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# vlan primary-vlan-id | Enters VLAN configuration mode and associates the VLANs to function as a PVLAN in the running configuration. |
| Step 3 | switch(config-vlan)# private-vlan association {add remove} secondary vlan-id | Associates a specified secondary VLAN with the primary VLAN to function as a PVLAN in the running configuration. To associate additional secondary VLANs, repeat this step. |
| Step 4 | switch(config-vlan)# show vlan private-vlan | (Optional) Displays the PVLAN configuration. |
| Step 5 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to associate VLANs in a PVLAN:

```

switch# configure terminal
switch(config)# vlan 202
switch(config-vlan)# private-vlan association add 303
switch(config-vlan)# show vlan private-vlan
-----
Primary Secondary Type Ports
-----
202 303 community Veth1
n1000v(config-vlan)#

```

Configuring a Private VLAN Host Port

You can configure an interface as a host port to function with a PVLAN.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know that the primary VLAN for this PVLAN is already configured as a PVLAN.
- Know that the secondary VLANs for this PVLAN are already configured as PVLANs.
- Know that the secondary VLANs are already associated with the primary VLAN.
- Know the name of the interface to be used with the PVLAN as a host port.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# interface type if_id | Enters interface configuration mode and creates a the named interface if it does not exist. |
| Step 3 | switch(config-if)# switchport mode private-vlan host | Designates that the physical interface is to function as a PVLAN host port in the running configuration. |
| Step 4 | switch(config-if)# show interface type if_id | (Optional) Displays the interface configuration. |
| Step 5 | switch(config-if)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure a PVLAN host port:

```
switch# configure terminal
switch(config)# interface veth1
switch(config-if)# switchport mode private-vlan host
switch(config-if)# 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
```

```
switch(config-if)#
```

Associating a vEthernet Port Profile with a Private VLAN

You can associate the vEthernet port profile with the primary and secondary VLANs in a PVLAN.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know the VLAN IDs of the primary and secondary VLANs in the PVLAN.
- Know that the primary VLAN for this PVLAN is already configured as a PVLAN.
- Know that the secondary VLANs for this PVLAN are already configured as PVLANS.
- Know the name of the interface functioning in the PVLAN as a host port.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# port-profile type vethernet name | Enters port profile configuration mode for the specified port profile. |
| Step 3 | switch(config-port-profile)# switchport mode private-vlan host | Associates the vEthernet port with the PVLAN configuration. The port profile is associated with the VLANs in the PVLAN. |
| Step 4 | switch(config-port-profile)# switchport private-vlan host-association vlan_ids | Assigns the primary and secondary VLAN IDs to the port profile and saves this association in the running configuration. |
| Step 5 | switch(config-port-profile)# no shut | Enables the port profile. |
| Step 6 | switch(config-port-profile)# vmware port-group | Designates the port profile as a VMware port group. The port profile is mapped to a VMware port group of the same name unless you specify a name here. When you connect the VSM to vCenter Server, the port group is distributed to the virtual switch on the vCenter Server. |
| Step 7 | switch(config-port-profile)# state enabled | Enables the port profile and applies its configuration to the assigned ports. |
| Step 8 | switch(config-if)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to associate a vEthernet port with a PVLAN:

```
switch # configure terminal
switch(config)# port-profile type vethernet vlan_private_isolated_127
switch(config-port-prof)# switchport mode private-vlan host
switch(config-port-prof)# switchport private-vlan host-association 126 127
switch(config-port-prof)# no shut
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# state enabled
```

Configuring a Layer 2 Port Profile as a Promiscuous Trunk Port

You can 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

- Log in to the CLI in EXEC mode.
- Know that the **private-vlan mapping trunk** command does not decide or override the trunk configuration of a port.
- Know that the port is already configured in a regular trunk mode before adding the PVLAN trunk configurations.
- Know that primary VLANs must be added to the list of allowed VLAN for the promiscuous trunk port.
- Know that secondary VLANs are not configured in the allowed VLAN list.
- Know that the trunk port can carry normal VLANs in addition to primary VLANs.
- Know that you can map up to 64 primary VLANs to their secondary VLANs in one promiscuous trunk port.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# port-profile type ethernet name | Places you in the port-profile mode. |
| Step 3 | switch(config-port-prof)# switchport mode trunk | Designates that the interfaces are to be used as trunking ports. |

| | Command or Action | Purpose |
|----------------|---|--|
| Step 4 | <code>switch(config-port-prof)# switchport mode private-vlan trunk promiscuous</code> | In the running configuration, designates the interface as a promiscuous PVLAN trunk port. |
| Step 5 | <code>switch(config-port-prof)# switchport private-vlan trunk allowed vlan <i>vlan_range</i></code> | Sets the allowed VLANs and VLAN IDs when the interface is in PVLAN trunking mode. |
| Step 6 | <code>switch(config-port-prof)# switchport private-vlan mapping trunk <i>primary_vlan_ID {secondary_vlan_list add secondary_vlan_list remove secondary_vlan_list}</i></code> | Maps the PVLAN trunk port to a primary VLAN and to selected secondary VLANs in the running configuration. Multiple PVLAN pairs can be specified so that a promiscuous trunk port can carry multiple primary VLANs. |
| Step 7 | <code>switch(config-port-prof)# no shut</code> | Enables the port profile. |
| Step 8 | <code>switch(config-port-profile)# vmware port-group</code> | Designates the port profile as a VMware port group. The port profile is mapped to a VMware port group of the same name unless you specify a name here. When you connect the VSM to vCenter Server, the port group is distributed to the virtual switch on the vCenter Server. |
| Step 9 | <code>switch(config-port-profile)# state enabled</code> | Enables the port profile and applies its configuration to the assigned ports. |
| Step 10 | <code>switch(config-if)# copy running-config startup-config</code> | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure a Layer 2 port profile as a promiscuous trunk port:

```
switch # configure terminal
switch(config)# port-profile type eth allaccess1
switch(config-port-prof)# switchport mode trunk
switch(config-port-prof)# switchport mode private-vlan trunk promiscuous
switch(config-port-prof)# switchport private-vlan trunk allowed vlan 2,126-128,150-155
switch(config-port-prof)# switchport private-vlan mapping trunk 126 127,128
switch(config-port-prof)# no shut
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# state enabled
```

Configuring a Private VLAN Promiscuous Access Port

You can configure a port to be used as a promiscuous access port in a PVLAN.

Before You Begin

- Log in to the CLI in EXEC mode.

- Know the name of the interface that will function as a promiscuous access port.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# interface type [slot/port number] | Enters interface configuration mode for the specified interface. |
| Step 3 | switch(config-if)# switchport mode private-vlan promiscuous | Designates that the interface is to function as a promiscuous access port for a PVLAN in the running configuration. |
| Step 4 | switch(config-if)# show interface type [slot/port number] | (Optional) Displays the interface configuration. |
| Step 5 | switch(config-if)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure a PVLAN promiscuous access port:

```

switch# configure terminal
switch(config)# interface eth3/2
switch(config-if)# switchport mode private-vlan promiscuous
switch(config-if)# show interface 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

switch(config-if)#
switch# configure terminal
switch(config)# interface vethernet1
n1000v(config-if)# switchport mode private-vlan promiscuous
switch(config-if)# show interface vethernet 1
Vethernet1 is up
  Port description is VM-1, Network Adapter 7
  Hardware: Virtual, address: 0050.569e.009f (bia 0050.569e.009f)
  Owner is VM "VM-1", adapter is Network Adapter 7
  Active on module 5
  VMware DVS port 5404
  Port-Profile is pri_25
    
```

```

Port mode is Private-vlan promiscuous
5 minute input rate 0 bits/second, 0 packets/second
5 minute output rate 7048 bits/second, 2 packets/second
Rx
 20276 Input Packets 379239 Unicast Packets
 24 Multicast Packets 1395 Broadcast Packets
1428168 Bytes
Tx
 256229 Output Packets 74946 Unicast Packets
16247 Multicast Packets 2028117 Broadcast Packets 190123 Flood Packets
44432239 Bytes
162 Input Packet Drops 159 Output Packet Drops

switch(config-if)#

```

Associating a Promiscuous Access Port with a Private VLAN

You can associate the promiscuous access port with the primary and secondary VLANs in a PVLAN.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know the VLAN IDs of the primary and secondary VLANs in the PVLAN.
- Know the primary and secondary VLANs that are already configured as PVLAN.
- Know the name of the interface functioning in the PVLAN as a promiscuous access port.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# interface <i>type</i> [<i>slot/port</i> <i>number</i>] | Enters interface configuration mode for the specified interface. |
| Step 3 | switch(config-if)# switchport private-vlan mapping <i>primary_vlan_ID</i> { <i>secondary_vlan_list</i> add <i>secondary_vlan_list</i> remove <i>secondary_vlan_list</i> } | Associates the promiscuous access port with the VLAN IDs in the PVLAN in the running configuration. |
| Step 4 | switch(config-if)# show interface <i>type</i> [<i>slot/port</i> <i>number</i>] | (Optional) Displays the interface configuration. |
| Step 5 | switch(config-if)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to associate a promiscuous access port with a PVLAN:

```

switch# configure terminal
switch(config)# interface eth3/2
switch(config-if)# switchport private-vlan mapping 202 303

```



```

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

switch(config-if)#

```

Removing a Private VLAN Configuration

You can remove a PVLAN configuration and return the VLAN to normal VLAN mode.

Before You Begin

- Log 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.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# vlan private vlan-id | Enters the VLAN configuration mode for the specified VLAN. |
| Step 3 | switch(config-vlan)# no private-vlan {community isolated primary} | 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 | switch(config-vlan)# show vlan private-vlan | (Optional) Displays the PVLAN configuration. |
| Step 5 | switch(config-vlan)# copy running-config startup-config | (Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to remove a PVLAN configuration:

```

switch# configure terminal
switch(config)# vlan 5
switch(config-vlan)# no private-vlan primary
switch(config-vlan)# show vlan private-vlan
-----
Primary  Secondary  Type           Ports
-----
switch(config-vlan)#

```

Verifying a Private VLAN Configuration

Use the following commands to verify a private VLAN configuration:

| Command | Purpose |
|--|--|
| show feature | Displays features available and whether they are enabled globally. |
| show running-config vlan <i>vlan-id</i> | Displays VLAN information. |
| show vlan private-vlan [<i>type</i>] | Displays information about PVLANS. |
| show interface switchport | Displays information about all interfaces configured as switchports. |

Configuration Examples for Private VLANs

Example: PVLAN Trunk Port

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

- PVLAN trunk port
- Mapped to primary PVLAN 202 which is associated with secondary VLANs 303 and 440
- Mapped to primary PVLAN 210 which is associated with secondary VLANs 310 and 450

```
switch# configure terminal
switch(config)# vlan 303,310
switch(config-vlan)# private-vlan community
switch(config)# vlan 440,450
switch(config-vlan)# private-vlan isolated

switch(config)# vlan 202
switch(config-vlan)# private-vlan primary
switch(config-vlan)# private-vlan association 303,440

switch(config)# vlan 210
switch(config-vlan)# private-vlan primary
switch(config-vlan)# private-vlan association 310,450

switch# configure terminal
switch(config)# int eth2/6
switch(config-if)# switchport mode private-vlan trunk promiscuous
switch(config-if)# switchport private-vlan trunk allowed vlan all
switch(config-if)# switchport private-vlan mapping trunk 202 303, 440
switch(config-if)# switchport private-vlan mapping trunk 210 310, 450
switch(config-if)# show interface switchport
Name: Ethernet2/6
Switchport: Enabled
Operational Mode: Private-vlan trunk promiscuous
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: 1-3967,4048-4093
Administrative private-vlan primary host-association: none
```

```

Administrative private-vlan secondary host-association: none
Administrative private-vlan primary mapping: none
Administrative private-vlan secondary mapping: none
Administrative private-vlan trunk native VLAN: 1
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: 1-3967, 4048-4093
Administrative private-vlan trunk private VLANs: (202,303) (202,440) (210,310) (210,450)
Operational private-vlan: 202,210,303,310,440,450
switch(config-if)#

```

Example: PVLAN Using Port Profiles

This 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 PVLAN 156:

```

vlan 153-154
  private-vlan community
vlan 155
  private-vlan isolated
vlan 156
  private-vlan association 153-155
  private-vlan primary

switch# show run int eth2/6

version 4.0(1)
interface Ethernet2/6
switchport
inherit port-profile uppvlanpromtrunk156

switch# 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

switch# show interface eth 2/6 switchport
Name: Ethernet2/6
Switchport: Enabled
Switchport Monitor: Not enabled
Operational Mode: Private-vlan trunk promiscuous
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: 1-3967,4048-4093
Administrative private-vlan primary host-association: none
Administrative private-vlan secondary host-association: none
Administrative private-vlan primary mapping: none
Administrative private-vlan secondary mapping: none
Administrative private-vlan trunk native VLAN: 1
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: 1-155,157-3967,4048-4093
Administrative private-vlan trunk private VLANs: (156,153) (156,155)
Operational private-vlan: 156,153,155 inherit port-profile uppvlanpromtrunk156
switch#

```

Feature History for Private VLANs

| Feature Name | Feature Name | Releases |
|--|--------------|---|
| <code>feature private-vlan</code> command | 4.2(1)SV1(4) | Added the ability to globally enable the PVLAN feature. |
| Private VLAN | 4.0(4)SV1(1) | This feature was introduced. |



Configuring IGMP Snooping

This chapter contains the following sections:

- [Information About IGMP Snooping, page 43](#)
- [Prerequisites for IGMP Snooping, page 45](#)
- [Default Settings, page 45](#)
- [Configuring IGMP Snooping, page 46](#)
- [Verifying the IGMP Snooping Configuration, page 49](#)
- [Example Configuration IGMP Snooping, page 49](#)
- [Feature History for IGMP Snooping, page 50](#)

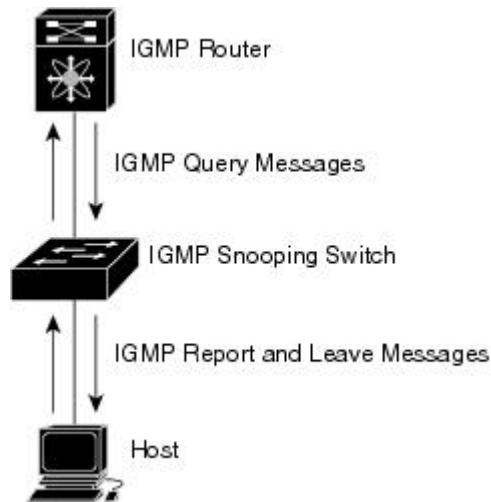
Information About IGMP Snooping

Introduction

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.

The following figure 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: 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:

- Multicast forwarding based on an IP address rather than a 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

If no more than one host is attached to each VLAN switch port, 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 timeout to indicate that no hosts remain that want to receive multicast data for a particular group.

Report suppression is not supported and is disabled by default.



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

IGMPv3 snooping provides constrained flooding based on the group IP information in the IGMPv3 reports. 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.

Even though the IGMPv3 membership reports provide a full accounting of group members on a LAN segment, when the last host leaves, the querier 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. If the querier specifies a mean-response-time (MRT) value in the queries, it overrides the last member query interval configuration.

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.

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. In the Cisco Nexus 1000V, report suppression is not supported and is disabled by default.

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.

Default Settings

Table 8: Default IGMP Snooping Settings

| Parameters | Default |
|-------------------------------|----------|
| IGMP snooping | Enabled |
| IGMPv3 Explicit tracking | Enabled |
| IGMPv2 Fast leave | Disabled |
| Last member query interval | 1 second |
| Link-local groups suppression | Enabled |
| Snooping querier | Disabled |

| Parameters | Default |
|------------------------------|----------|
| IGMPv1/v2 Report suppression | Disabled |
| IGMPv3 Report suppression | Disabled |

Configuring IGMP Snooping

Enabling or Disabling IGMP Snooping Globally for the VSM

You can enable or disable IGMP snooping globally for the VSM. IGMP snooping is enabled globally on the VSM (the default). If enabled globally, you can turn it on or off per VLAN.

Before You Begin

Log in to the CLI in EXEC mode.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# [no] ip igmp snooping | Enables or disables IGMP snooping in the running configuration for all VLANs. The default is enabled. If you have previously disabled the feature then you can enable it with this command. |
| Step 3 | switch(config)# show ip igmp snooping [vlan vlan-id] | (Optional) Displays the configuration for verification. Note If disabled, IGMP snooping on all VLANs is disabled. |
| Step 4 | switch(config)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to disable IGMP snooping:

```
switch# configure terminal
switch(config)# no ip igmp snooping
switch(config)# show ip igmp snooping
Global IGMP Snooping Information:
  IGMP Snooping enabled
  IGMPv1/v2 Report Suppression disabled
  IGMPv3 Report Suppression disabled
  Link Local Groups Suppression enabled

IGMP Snooping information for vlan 1
```



```

IGMP snooping enabled
IGMP querier none
Switch-querier disabled
IGMPv3 Explicit tracking enabled
IGMPv2 Fast leave disabled
IGMPv1/v2 Report suppression disabled
IGMPv3 Report suppression disabled
Link Local Groups suppression enabled
Router port detection using PIM Hellos, IGMP Queries
Number of router-ports: 0
Number of groups: 0
Active ports:

--More--
switch(config)#

```

Configuring IGMP Snooping on a VLAN

You can configure IGMP snooping on a VLAN. IGMP snooping is enabled by default for all VLANs in the VSM.

Before You Begin

Log in to the CLI in EXEC mode.



Note If IGMP snooping is disabled globally, it takes precedence over the VLAN state.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# vlan vlan-id | Enters configuration mode for the specified VLAN. |
| Step 3 | switch(config-vlan)# [no] ip igmp snooping | Enables or disables IGMP snooping in the running configuration for the specific VLAN. If IGMP snooping is enabled for the VSM, IGMP snooping is enabled for the VLAN by default. Note IGMP snooping must be enabled globally (the default) in order to toggle it on or off per VLAN. If IGMP snooping is disabled globally, then it cannot be enabled per VLAN. |
| Step 4 | switch(config-vlan)# [no] ip igmp snooping explicit-tracking | (Optional) Tracks IGMPv3 membership reports from individual hosts for each port on a per-VLAN basis in the running configuration. The default is enabled. |
| Step 5 | switch(config-vlan)# [no] ip igmp snooping fast-leave | (Optional) Enables fast-leave for the specified VLAN in the running configuration. |

| | Command or Action | Purpose |
|----------------|--|--|
| | | <p>Fast-leave supports IGMPv2 hosts that cannot be explicitly tracked because of the host report suppression mechanism of the IGMPv2 protocol.</p> <p>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.</p> <p>When you enable fast leave, the IGMP software assumes that no more than one host is present on each VLAN port.</p> <p>The default is disabled.</p> |
| Step 6 | switch(config-vlan)# [no] ip igmp snooping last-member-query-interval seconds | <p>(Optional)</p> <p>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.</p> <p>Values range from 1 to 25 seconds. The default is 1 second.</p> |
| Step 7 | switch(config-vlan)# [no] ip igmp snooping mrouter interface type if_id | <p>(Optional)</p> <p>Configures a static connection for the VLAN to a multicast router in the running configuration.</p> <p>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.</p> <p>vEths are not supported as router ports.</p> |
| Step 8 | switch(config-vlan)# [no] ip igmp snooping static-group group-ip-addr interface type if_id | <p>(Optional)</p> <p>Configures a VLAN Layer 2 port as a static member of a multicast group in the running configuration.</p> <p>You can specify the interface by the type and the number, such as ethernet slot/port.</p> |
| Step 9 | switch(config-vlan)# [no] ip igmp snooping link-local-groups-suppression | <p>(Optional)</p> <p>Configures link-local groups suppression. The default is enabled.</p> <p>Note You can apply link-local groups suppression to all interfaces in the VSM by entering this command in global configuration mode.</p> |
| Step 10 | switch(config-vlan)# show ip igmp snooping [vlan vlan-id] | <p>(Optional)</p> <p>Displays the configuration for verification.</p> |
| Step 11 | switch(config-vlan)# copy running-config startup-config | <p>(Optional)</p> <p>(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.</p> |

This example shows how to configure IGMP snooping on a VLAN:

```
switch# configure terminal
switch(config)# vlan 2
switch(config-vlan)# ip igmp snooping
switch(config-vlan)# ip igmp snooping explicit-tracking
switch(config-vlan)# ip igmp snooping fast-leave
switch(config-vlan)# ip igmp snooping last-member-query-interval 3
switch(config-vlan)# ip igmp snooping mrouter interface ethernet 2/1
switch(config-vlan)# ip igmp snooping static-group 230.0.0.1 interface ethernet 2/1
switch(config-vlan)# ip igmp snooping link-local-groups-suppression
switch(config-vlan)# show ip igmp snooping vlan 2

IGMP Snooping information for vlan 5
  IGMP snooping enabled
  IGMP querier none
  Switch-querier disabled
  IGMPv3 Explicit tracking enabled
  IGMPv2 Fast leave enabled
  IGMPv1/v2 Report suppression disabled
  IGMPv3 Report suppression disabled
  Link Local Groups suppression enabled
  Router port detection using PIM Hellos, IGMP Queries
  Number of router-ports: 0
  Number of groups: 0
  Active ports:
switch(config-vlan)#
```

Verifying the IGMP Snooping Configuration

Use the following commands to verify the IGMP snooping configuration information.

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

For detailed information about commands and their output, see the *Cisco Nexus 1000V Command Reference*.

Example Configuration IGMP Snooping

This example shows how to enable IP IGMP snooping for the VSM and make the following optional configurations for VLAN 2:

- Tracking of IGMPv3 membership reports from individual hosts for each port.
- A static connection to a multicast router through Ethernet 2/1.

- Static membership in multicast group 230.0.0.1.

```
switch# configure terminal
switch# ip igmp snooping
switch# vlan 2
switch# ip igmp snooping
switch# ip igmp snooping explicit-tracking
switch# ip igmp snooping mrouter interface ethernet 2/1
switch# ip igmp snooping static-group 230.0.0.1 interface ethernet 2/1
switch# show ip igmp snooping vlan 2
switch# copy running-config startup-config
switch#
```

Feature History for IGMP Snooping

| Feature Name | Releases | Description |
|------------------------|--------------|--|
| Link-local suppression | 4.2(1)SV1(4) | Added support to enable or disable link-local group suppression. |
| Report suppression | 4.0(4)SV1(3) | Removed support for report suppression. |
| IGMP Snooping | 4.0(4)SV1(1) | This feature was introduced. |



Configuring Network Load Balancing for vEthernet

This chapter contains the following sections:

- [Information About Microsoft Network Load Balancing](#), page 51
- [Guidelines and Limitations](#), page 51
- [Configuring Microsoft Network Load Balancing Support in Interface Configuration Mode](#), page 52
- [Configuring Microsoft Network Load Balancing in Port Profile Configuration Mode](#), page 53
- [Feature History for Microsoft Network Load Balancing for vEthernet](#), page 55

Information About Microsoft Network Load Balancing

Microsoft Network Load Balancing (NLB) is a clustering technology offered by Microsoft as part of the Windows server operating systems. Clustering enables a group of independent servers to be managed as a single system for higher availability, easier manageability, and greater scalability.

For more information about Microsoft NLB, see <http://technet.microsoft.com/en-us/library/bb742455.aspx>



Note

Access to third-party websites identified in this document is provided solely as a courtesy to customers and others. Cisco Systems, Inc. and its affiliates are not in any way responsible or liable for the functioning of any third-party website, or the download, performance, quality, functioning or support of any software program or other item accessed through the website, or any damages, repairs, corrections or costs arising out of any use of the website or any software program or other item accessed through the website. Cisco's End User License Agreement does not apply to the terms and conditions of use of a third-party website or any software program or other item accessed through the website.

Guidelines and Limitations

NLB has the following configuration guidelines and limitations:

- The **no mac auto-static-learn** command is not supported on PVLAN ports.
- The **no mac auto-static-learn** command is not supported on the ports that are configured with **switchport port-security mac-address sticky**.
- Unknown unicast flood blocking (UUFB) does not block Microsoft-Network Load Balancing (MS-NLB) packets on MS-NLB vEthernet interfaces. UUFB can be used to limit flooding of MS-NLB packets to non-MS-NLB ports within a VLAN.

Configuring Microsoft Network Load Balancing Support in Interface Configuration Mode

You can configure Microsoft NLB in the interface configuration mode.

Before You Begin



Note

Make sure that the Cisco Nexus 1000V is configured before you configure Microsoft NLB on Windows virtual machines (VMs).

- Log in to the CLI in EXEC mode.
- Know that unicast is the default Microsoft Network Load Balancing mode of operation.
- Know that Microsoft NLB replaces the MAC address of each server in the cluster to a common Microsoft NLB MAC address.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# show running-config interface veth number | Displays the vEthernet configuration to determine if no mac auto-static-learn is configured or not. |
| Step 3 | switc(config)# interface veth | Sets interface configuration mode on vEthernet modules. |
| Step 4 | switch(config-if)# [no] mac auto-static-learn | Toggles the auto-mac-learning on vEthernet modules. |
| Step 5 | switch(config-if)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure Microsoft NLB directly on vEthernet:

```
switch# configure terminal
switch(config)# interface vethernet 1
switch(config-if)# no mac auto-static-learn
switch(config-if)# show running-config interface vethernet 1
!Command: show running-config interface Vethernet1
!Time: Tue Nov 15 19:01:36 2011

version 4.2(1)SV1(5.1)

interface Vethernet1
 inherit port-profile vm59
 description stc3, Network Adapter 2
 no mac auto-static-learn
 vmware dvport 34 dvswitch uuid "ea 5c 3b 50 cd 00 9f 55-41 a3 2d 61 84 9e 0e c4"
 vmware vm mac 0050.56B3.0071

switch(config)#
```

This example shows how to unconfigure Microsoft NLB directly from a vEthernet interface:

```
switch# configure terminal
switch(config)# interface vethernet 1
switch(config-if)# mac auto-static-learn
switch(config-if)# show running-config interface vethernet 1
!Command: show running-config interface Vethernet1
!Time: Tue Nov 15 19:01:52 2011

version 4.2(1)SV1(5.1)

interface Vethernet1
 inherit port-profile vm59
 description stc3, Network Adapter 2
 mac auto-static-learn
 vmware dvport 34 dvswitch uuid "ea 5c 3b 50 cd 00 9f 55-41 a3 2d 61 84 9e 0e c4"
 vmware vm mac 0050.56B3.0071

switch(config)#
```

Configuring Microsoft Network Load Balancing in Port Profile Configuration Mode

You can configure Microsoft NLB in the port profile configuration mode.

Before You Begin



Note

Make sure that the Cisco Nexus 1000V is configured before you configure Microsoft NLB on Windows Virtual Machines (VMs).

- Log in to the CLI in EXEC mode.
- Know that unicast is the default Microsoft Network Load Balancing mode of operation.
- Know that Microsoft NLB replaces the MAC address of each server in the cluster to a common Microsoft NLB MAC address.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# show running config port-profile profile name | Displays the port profile configuration to determine if no mac auto-static-learn is configured or not. |
| Step 3 | switch(config)# port profile type vethernet ms-nlb | Sets port profile configuration mode on vEthernet modules. |
| Step 4 | switch(config-port-prof)# [no] mac auto-static-learn | Toggles the auto-mac-learning on vEthernet modules. |
| Step 5 | switch(config-port-prof)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure Microsoft NLB in port profile mode:

```
switch# configure terminal
switch(config)# port-profile type vethernet ms-nlb
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# switchport mode access
switch(config-port-prof)# switchport access vlan 59
switch(config-port-prof)# no mac auto-static-learn
switch(config-port-prof)# no shutdown
switch(config-port-prof)# state enabled
switch(config-port-prof)# show running-config port-profile ms-nlb
!Command: show running-config port-profile ms-nlb
!Time: Tue Nov 15 19:00:40 2011

version 4.2(1)SV1(5.1)
port-profile type vethernet ms-nlb
  vmware port-group
  switchport mode access
  switchport access vlan 59
  no mac auto-static-learn
  no shutdown
  state enabled
switch(config-port-prof)#
```

This example shows how to unconfigure Microsoft NLB on a vEthernet interface in port profile mode:

```
switch# configure terminal
switch(config)# port-profile type vethernet ms-nlb
switch(config-port-prof)# mac auto-static-learn
switch(config-port-prof)# show running-config port-profile ms-nlb
!Command: show running-config port-profile ms-nlb
!Time: Tue Nov 15 19:01:05 2011

version 4.2(1)SV1(5.1)
port-profile type vethernet ms-nlb
  vmware port-group
  switchport mode access
  switchport access vlan 59
  mac auto-static-learn
  no shutdown
  state enabled
switch(config-port-prof)#
```


Feature History for Microsoft Network Load Balancing for vEthernet

| Feature Name | Feature Name | Releases |
|------------------------|----------------|------------------------------|
| Network Load Balancing | 4.2(1)SV1(5.1) | This feature was introduced. |



Supporting Redundant Routing Protocols

This chapter contains the following sections:

- [Information About Redundant Routing Protocols, page 57](#)
- [Guidelines and Limitations, page 57](#)
- [Supporting Redundant Routing Protocols, page 58](#)
- [Feature History for Supporting Redundant Routing Protocol, page 62](#)

Information About Redundant Routing Protocols

The Cisco Nexus 1000V implements a loop detection mechanism that is based on source and destination MAC addresses and drops packets that are coming in on uplink ports if the source MAC address is already present on a local vEthernet interface. As a result, such protocols as the Virtual Router Redundancy Protocol (VRRP), the Common Address Redundancy Protocol (CARP), the Hot Standby Router Protocol (HSRP), and other similar protocols fail on Virtual Machines (VMs) that are associated to the Cisco Nexus 1000V.

Disabling loop detection provides a flexible way of supporting these protocols on VMs that are associated to the Cisco Nexus 1000V. By disabling the loop detection mechanism, you can configure any combination of the above mentioned protocols on a port profile or a vEthernet interface. As a result, you can run multiple protocols on the same VM.

Guidelines and Limitations

Supporting the redundant routing protocols feature has the following configuration guidelines and limitations:

- Disable IGMP Snooping on both Cisco Nexus 1000V and upstream switches between the servers to support most redundant routing protocols. See [Enabling or Disabling IGMP Snooping Globally for the VSM, on page 46](#).
- A disabled loop detection configuration is not supported on PVLAN ports.
- A disabled loop detection configuration is not supported on the port security ports.

Supporting Redundant Routing Protocols

Configuring a vEthernet Interface to Support Redundant Routing Protocols

You can configure a vEthernet interface to support redundant routing protocols.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know which redundant routing protocol that you want to disable.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# interface vethernet <i>interface-number</i> | Enters interface configuration mode for the specified vEthernet interface (from 1 to 1048575). |
| Step 3 | switch(config-if)# disable-loop-detection { carp hsrp vrrp custom-rp [src-mac-range <i>s_mac end_mac</i>] [dest-ip <i>ip_address</i>] [ip-proto <i>no</i>] [port <i>port</i>]} | <p>Enables or disables the loop detection mechanism to support a redundant routing protocol on a vEthernet interface.</p> <ul style="list-style-type: none"> • disable-loop-detection—Disables the loop detection mechanism. • no disable-loop-detection—Enables the loop detection mechanism. This is the default configuration. <p>The protocols supported on a vEthernet interface are as follows:</p> <ul style="list-style-type: none"> • carp—Common Address Redundancy Protocol • custom-rp—User-defined protocol • hsrp—Hot Standby Router Protocol • vrrp—Virtual Router Redundancy Protocol <p>The parameters for custom defined protocols are as follows:</p> <ul style="list-style-type: none"> • src-mac-range—Source MAC address range for the user-defined protocol. • dest-ip—Destination IP address for the user-defined protocol. • ip-proto—IP protocol number for the user-defined protocol. • port—UDP or TCP destination port number for the user-defined protocol. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 4 | switch(config-if)# show running-config interface vethernet interface-number | (Optional) Displays the interface status and information. |
| Step 5 | switch(config-if)# copy running-config startup-config | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to configure a vEthernet interface to support VRRP, CERP, HSRP, and user-defined protocols on a VM:

```
switch# configure terminal
switch(config)# interface veth5
switch(config-if)# disable-loop-detection carp
switch(config-if)# disable-loop-detection vrrp
switch(config-if)# disable-loop-detection hsrp
switch(config-if)# disable-loop-detection custom-rp dest-ip 224.0.0.12 port 2234
switch(config-if)# end
switch# show running-config interface vethernet 5
!Command: show running-config interface Vethernet5
!Time: Fri Nov 4 02:21:24 2011

version 4.2(1)SV1(5.1)

interface Vethernet5
inherit port-profile vm59
description Fedorall17, Network Adapter 2
disable-loop-detection carp
disable-loop-detection custom-rp dest-ip 224.0.0.12 port 2234
disable-loop-detection hsrp
disable-loop-detection vrrp
vmware dvport 32 dvswitch uuid "ea 5c 3b 50 cd 00 9f 55-41 a3 2d 61 84 9e 0e c4"
vmware vm mac 0050.56B3.00B2

switch#
```

Configuring a Port Profile to Support Redundant Routing Protocols

You can configure a port profile to support redundant routing protocols. Use this procedure when the master in a master/slave relationship has lost connectivity, the slave has taken over the master role, or the original master is attempting to overtake the master role.



Note

If you configure a vEthernet interface and a port profile to run multiple protocols on the same VM, the configuration on the vEthernet interface overrides the configuration on the port profile.

Before You Begin

- Log in to the CLI in EXEC mode.
- Know which redundant routing protocol that you want to disable.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | switch# configure terminal | Enters global configuration mode. |
| Step 2 | switch(config)# port-profile <i>name</i> | Enters port profile configuration mode for the named port profile. |
| Step 3 | switch(config-port-prof)# switchport mode { access trunk } | Designates that the interface is to be used as a trunking port. A trunk port transmits untagged packets for the native VLAN and transmits encapsulated, tagged packets for all other VLANs. |
| Step 4 | switch(config-port-prof)# no shutdown | Administratively enables all ports in the profile. |
| Step 5 | switch(config-port-prof)# disable-loop-detection { carp hsrp vrrp custom-rp [src-mac-range <i>s_mac end_mac</i>] [dest-ip <i>ip_address</i>] [ip-proto <i>no</i>] [port <i>port</i>]} | <p>Enables or disables the loop detection mechanism to support a redundant routing protocol on vEthernet interface.</p> <ul style="list-style-type: none"> • disable-loop-detection—Disables the loop detection mechanism. • no disable-loop-detection—Enables the loop detection mechanism. This is the default configuration. <p>The protocols supported on a vEthernet interface are as follows:</p> <ul style="list-style-type: none"> • carp—Common Address Redundancy Protocol • custom-rp—User defined protocol • hsrp—Hot Standby Router Protocol • vrrp—Virtual Router Redundancy Protocol <p>The parameters for custom defined protocols are as follows:</p> <ul style="list-style-type: none"> • src-mac-range—Source MAC address range for the user defined protocol. • dest-ip—Destination IP address for the user defined protocol. • ip-proto—IP protocol number for the user defined protocol. • port—UDP or TCP destination port number for the user defined protocol. |
| Step 6 | switch(config-port-prof)# show port-profile [brief expand-interface usage] [name <i>profile-name</i>] | (Optional) Displays the configuration for verification. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 7 | <code>switch(config-port-prof)# copy running-config startup-config</code> | (Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

This example shows how to disable loop detection for the Hot Standby Router Protocol:

```
switch# configure terminal
switch(config)# port-profile hsrp-1
switch(config-port-prof)# switchport mode trunk
switch(config-port-prof)# no shutdown
switch(config-port-prof)# disable-loop-detection hsrp
switch(config-port-prof)# state enabled
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# show port-profile name hsrp-1
port-profile hsrp-1
  type: Vethernet
  description:
  status: enabled
  max-ports: 32
  min-ports: 1
  inherit:
  config attributes:
    switchport mode trunk
    disable-loop-detection hsrp
    no shutdown
  evaluated config attributes:
    switchport mode trunk
    disable-loop-detection hsrp
    no shutdown
  assigned interfaces:
  port-group: hsrp-1
  system vlans: none
  capability l3control: no
  capability iscsi-multipath: no
  capability vxlan: no
  capability l3-vservice: no
  port-profile role: none
  port-binding: static
```

This example shows how to disable loop detection for the Virtual Router Redundancy Protocol:

```
n1000v# configure terminal
switch(config)# port-profile vrrp-1
switch(config-port-prof)# switchport mode trunk
switch(config-port-prof)# no shutdown
switch(config-port-prof)# disable-loop-detection vrrp
switch(config-port-prof)# state enabled
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# show port-profile name vrrp-1
port-profile vrrp-1
  type: Vethernet
  description:
  status: enabled
  max-ports: 32
  min-ports: 1
  inherit:
  config attributes:
    switchport mode trunk
    disable-loop-detection vrrp
    no shutdown
  evaluated config attributes:
    switchport mode trunk
    disable-loop-detection vrrp
```

```
no shutdown
assigned interfaces:
port-group: vrrp-1
system vlans: none
capability l3control: no
capability iscsi-multipath: no
capability vxlan: no
capability l3-vservice: no
port-profile role: none
port-binding: static
```

Feature History for Supporting Redundant Routing Protocol

| Feature Name | Feature Name | Releases |
|---------------------------------------|----------------|------------------------------|
| Supporting Redundant Routing Protocol | 4.2(1)SV1(5.1) | This feature was introduced. |



Layer 2 Switching Configuration Limits

This chapter contains the following sections:

- [Layer 2 Switching Configuration Limits](#), page 63

Layer 2 Switching Configuration Limits

The configuration limits are documented in the *Cisco Nexus 1000V Resource Availability Reference*.



INDEX

A

- associating [32, 34, 38](#)
 - host port with PVLAN [34](#)
 - promiscuous access port [38](#)
 - VLANs in a PVLAN [32](#)

C

- changed information [1](#)
 - description [1](#)
- clearing [14](#)
 - dynamic addresses from the MAC address table [14](#)
- configuring [12, 13, 28, 30, 33, 35, 36, 47, 52, 53, 59](#)
 - aging time [13](#)
 - IGMP snooping on VLAN [47](#)
 - Layer 2 interface as a promiscuous trunk port [35](#)
 - Microsoft network load balancing support [52, 53](#)
 - port profile to support redundant routing protocols [59](#)
 - private VLAN [28, 30](#)
 - private VLAN host port [33](#)
 - private VLAN promiscuous access port [36](#)
 - static MAC address [12](#)
- Configuring [58](#)
- configuring private VLAN [31](#)
- creating [20](#)
 - VLAN [20](#)

D

- default settings [12, 19, 28, 45](#)
 - IGMP Snooping [45](#)
 - MAC address table [12](#)
 - private VLAN [28](#)
 - VLAN [19](#)

E

- enabling [29, 46](#)
 - IGMP snooping [46](#)
 - private vlan [29](#)
- example [15, 40, 49](#)
 - IGMP snooping [49](#)
 - MAC address table [15](#)
 - private VLAN [40](#)

F

- feature history [16, 24, 42, 50, 55, 62](#)
 - IGMP snooping [50](#)
 - MAC address table [16](#)
 - network load balancing [55](#)
 - private VLAN [42](#)
 - supporting redundant routing protocol [62](#)
 - VLAN [24](#)

G

- guidelines and limitations [28, 51, 57](#)
 - network load balancing [51](#)
 - private VLAN [28](#)
 - supporting redundant routing protocols [57](#)

I

- IGMP snooping [8, 43](#)
- IGMPv1 [44](#)
- IGMPv2 [44](#)
- IGMPv3 [45](#)

L

- Layer 2 [63](#)
 - configuration limits [63](#)
- Layer 2 Ethernet switching [7](#)
- Layer 2 switching [1](#)
 - new and changed information [1](#)

M

- MAC address table [8, 11, 12](#)
 - guidelines [12](#)

N

- network load balancing [51](#)
- new information [1](#)
 - description [1](#)

P

- prerequisites [45](#)
 - IGMP snooping [45](#)
- private VLAN [25](#)
- private VLAN ports [26, 28](#)
- private VLANs [8](#)

R

- redundant routing protocols [57](#)
 - support [57](#)
- removing [39](#)
 - private vlan configuration [39](#)

V

- VEM [3, 4, 5, 7](#)
 - Port Model [3](#)
 - ports [5](#)
 - switching traffic [7](#)
 - virtual ports [4](#)
- verifying the configuration [15, 24, 40, 49](#)
 - IGMP snooping [49](#)
 - MAC address table [15](#)
 - private VLAN [40](#)
 - VLAN [24](#)
- vEthernet Interface to support redundant routing protocols [58](#)
- VLAN [8, 17, 18, 22](#)
 - configuring characteristics [22](#)
 - guidelines and limitations [18](#)
- VSM [6](#)
 - port model [6](#)