Cisco Nexus 1000VE for VMware vSphere Layer 2 Switching Configuration Guide, Release 5.2(1)SV5(1.1)

First Published: 2018-07-11

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

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.

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 NONINFRINGEMENT 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.

CONTENTS

CHAPTER 1

Overview

This chapter contains the following sections:

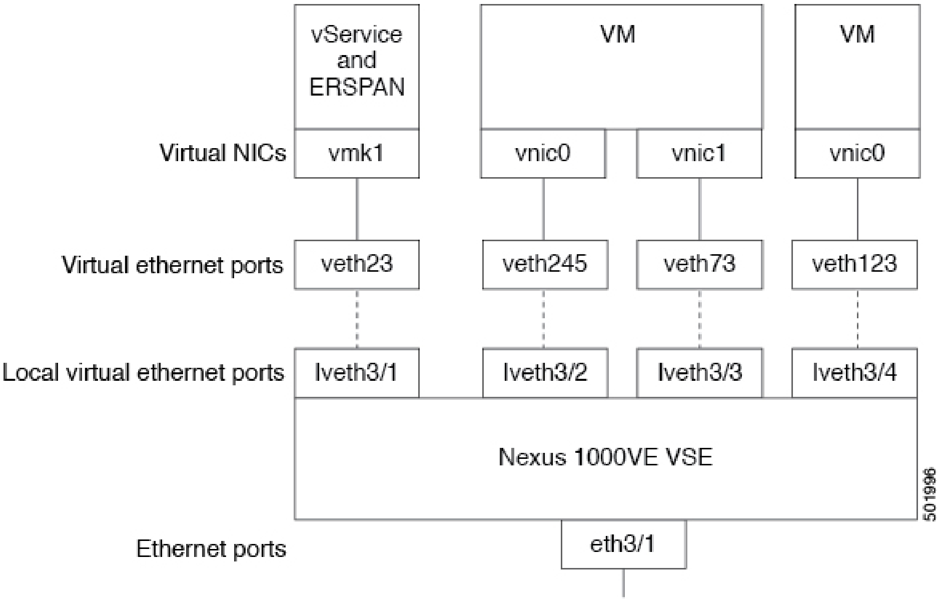
Information about Layer 2 Switching

VSE Port Model

The Virtual Service Engine (VSE) ports on Cisco Nexus 1000VE are referred to as VSE Virtual Ports.

The following figure shows the VSE view of the network.

Figure 1: VSE Port View



VSE Virtual Ports

The VSE maps together the following layers of ports:

Virtual NICs

There are two types of virtual NICs (vNICs). One vNIC represents a network interface on a Virtual Machine (VM), which emulates a physical port for the virtual host. The other vNIC is an internal port used by the hypervisor for management, iSCSI, and other network access. Each of these vNICs maps to a Virtual Ethernet port within the Cisco Nexus 1000VE.

Virtual Ethernet Ports

A Virtual Ethernet Ports represents a port on the Cisco Nexus 1000VE Distributed Virtual Switch. The Cisco Nexus 1000VE 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 profiles.

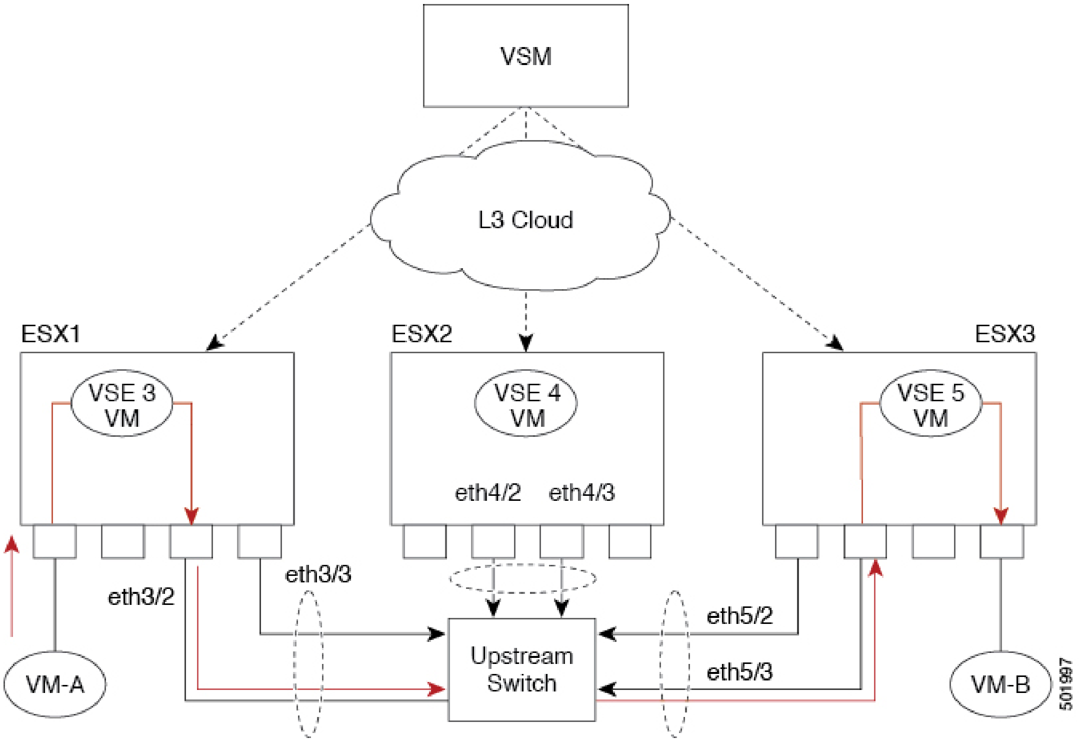
VSE Uplink Ports

The traffic egressing VSE Uplink port goes out through the Physical NICs added to the outside vDS.

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

Physical Ethernet Interfaces

Physical Ethernet interfaces (Eths) correspond to the outside trunk interface of the VSEs.

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.

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

Control VLANs

A control VLAN is used for communication between the VSM and the VSEs within a switch domain. The control interface is the first interface on the VSM.

A control VLAN is used for the following:

- VSM configuration commands to each VSE and their responses.
- VSE notifications to the VSM. For example, a VSE notifies the VSM of the attachment or detachment of ports to the Distributed Virtual Switch (DVS).
- VSE NetFlow exports that are sent to the VSM, where they are forwarded to a NetFlow Collector.
- VSM active to standby synchronization for high availability.

Management VLANs

A management VLAN, which is used for system login and configuration, corresponds to the mgmt0 interface. The mgmt0 interface appears as the mgmt0 port on a Cisco switch and is assigned an IP address (IPv4). When the mgmt0 interface (default) is used for Layer 3 connectivity on the VSM, the management interface communicates with the VSEs and the VMware vCenter Server.

The management interface is the second interface on the VSM.

Packet VLANs

Similar to the control VLAN, a packet VLAN is used for communication between the VSM and the VSEs within a switch domain.

A packet VLAN is used to tunnel network protocol packets between the VSM and the VSEs.

The packet interface is the third interface on the VSM.

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.

CHAPTER 2

Configuring MAC Address Tables

This chapter contains the following sections:

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 VSE 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 VSE can store up to 4096 MAC addresses.
- You can configure only 32 static MAC addresses on a single interface and 1024 static MAC addresses on a DVS.

- The Cisco Nexus 1000VE supports a maximum of 2000 private VLAN MAC addresses on a VSM.

Default Settings

Table 1: 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** {**type***if_id*} | **port-channel number**]}	Adds a static MAC address in the Layer 2 MAC address table and saves it in the running configuration. The interface can be specified as either of the following: • ethernet *slot/port* • veth *number*
Step 3	(Optional) switch(config)# **show mac address static interface** [**type** *if_id*]	Displays static MAC addresses.

	Command or Action	Purpose
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure a static MAC address:

```
switch# configure terminal
switch(config)# mac address-table static
switch(config)# show mac address static interface12ab.47dd.ff89 vlan 3
interface ethernet 3/1

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         Eth3/1                        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 3/1
VLAN      MAC Address       Type    Age       Port                          Module
---------+-----------------+-------+---------+------------------------------+---------
3         12ab.47dd.ff89    static  0         Eth3/1                        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: • 120 to 918000 seconds (default is 300) • If you specify zero (0), MAC aging is disabled.

Example

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	(Optional) switch# **show mac address-table**	Displays the MAC address table.

Example

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 the following commands to verify the configuration:

Command	Purpose
show mac address-table	Displays the MAC address table.
show mac address-table module	Displays information about specific module 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 VSE physical port learns a dynamic MAC and the packet source is in another VSE on the same VSM.
show mac address static interface [**type** *if_id*]	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 *interface_id* **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 aging-time 120
switch(config)#
```

CHAPTER 3

Configuring VLANs

This chapter contains the following sections:

Information About VLANs

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 Service Engine (VSE) based on VLAN tagging.

Table 2: VSE Action on VLAN Tagging

Port Type	Packet received	Action
Access	Tagged	The packet is dropped.
Access	Untagged	The VSE adds an access VLAN to the packet.
Trunk	Tagged	No action is taken on the packet.
Trunk	Untagged	The VSE adds a native VLAN tag to the 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 1000VE, and are listed in the following table.

Table 3: Cisco Nexus 1000VE VLAN Numbering

VLAN Numbers	Range	Usage
1	Normal	Cisco Nexus 1000VE default. You can use this VLAN, but you cannot modify or delete it.
2 to1005	Normal	You can create, use, modify, or delete these VLANs.
1006 to 4094	Extended	You can create, name, or use these VLANs. You cannot change the following parameters: • The state is always active. • These VLANs are always enabled. You cannot shut down these VLANs. The extended system ID is always automatically enabled.
3968 to 4047 and 4094	Internally allocated	You cannot use, create, delete, or modify these VLANs. You can display these VLANs. The Cisco Nexus 1000VE allocates these 80 VLANs, plus VLAN 4094, for features, like diagnostics, that use internal VLANs for their operation.

Default Settings

Table 4: 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

Parameters	Default
Physical ports	Trunk ports

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 15.
- Be familiar with the VLAN numbering in the Guidelines and Limitations, on page 11.
- 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 reenable, or re-create, that specified VLAN, the system automatically reinstates all the original ports to that VLAN. Note that the static MAC addresses and aging time for that VLAN are not restored when the VLAN is reenabled.

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.

	Command or Action	Purpose
Step 3	switch(config)# [**no**] **vlan** {*vlan-id* | *vlan-range*}	Creates or deletes, and saves in the running configuration, a VLAN or a range or VLANs. To configure the VLAN, see Configuring VLAN Characteristics, on page 15. **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. For information about Assigning Layer 2 interfaces to VLANs (access or trunk ports), see the *Cisco Nexus 1000VE Interface Configuration Guide*. For information about configuring ports as VLAN access or trunk ports and assigning ports to VLANs, see the *Cisco Nexus 1000VE Interface Configuration Guide*.
Step 4	(Optional) switch(config-vlan)# **show vlan id** *vlan-id*	Displays the VLAN configuration.
Step 5	(Optional) switch(config-vlan)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

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 1000VE 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 11.

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. • 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.
Step 4	switch(config-vlan)# **state** {**active** | **suspend**}	Changes the operational state of the VLAN and saves it in the running configuration. Allowable entries are as follows: • **active** (default) • **suspend** While the VLAN state is suspended, the ports associated with this VLAN are shut down, and that VLAN does not pass any traffic. **Note** You cannot suspend the state for the default VLAN or VLANs 1006 to 4094.
Step 5	switch(config-vlan)# **no shutdown**	Enables VLAN switching in the running configuration. Allowable entries are as follows: • **no shutdown** (default) • **shutdown** **Note** You cannot shut down the default VLAN, VLAN1, or VLANs 1006 to 4094.

	Command or Action	Purpose
Step 6	switch(config-vlan)# **exit**	Exits VLAN configuration mode. **Note** You must exit VLAN configuration mode for the configurations to take effect.
Step 7	(Optional) switch(config)# **show vlan** [**id** *vlan-id*]	Displays the VLAN configuration.
Step 8	(Optional) switch(config)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

This example shows how to configure VLAN characteristics:

```
switch# configure terminal
switch(config)# vlan 5
switch(config-vlan)# name accounting
switch(config-vlan)# state active
switch(config-vlan)# no shutdown
switch(config-vlan)# exit
switch(config)# show vlan brief

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

Verifying the Configuration

Use the following commands to verify the configuration:

Command	Purpose
show running-config vlan *vlan-id*	Displays VLAN information in the running configuration.

Command	Purpose
show vlan [**all-ports** | **brief** | **id** *vlan-id* | **name** *name* | **dot1q tag native**]	Displays the specified VLAN information.
show vlan summary	Displays a summary of VLAN information.

CHAPTER 4

Configuring Private VLANs

This chapter contains the following sections:

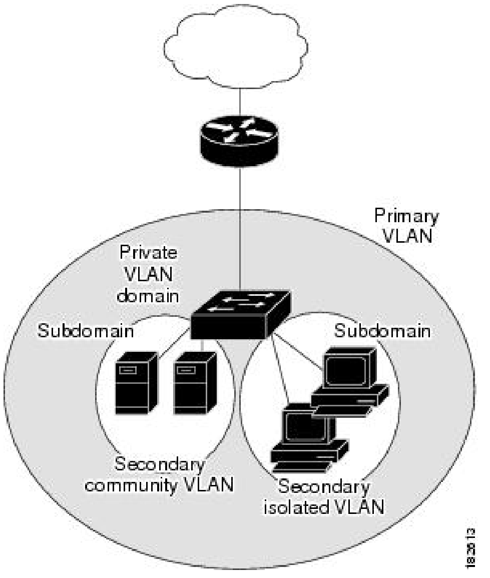
Information About Private VLANs

PVLANs achieve Layer 2 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 3: 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.

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 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 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 5: 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.

The following are configuration limits:

- Private VLANs per DVS: 512 maximum
- Primary VLANs per promiscuous trunk port: 64 maximum
- Private VLAN associations: 511 maximum
- Private VLAN ports per DVS : 4096 maximum

Default Settings

Table 6: 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.

Step 2 Configure a VLAN as a primary VLAN.

Step 3 Configure a VLAN as a secondary VLAN.

Step 4 Associate the VLANs in a PVLAN.

Step 5 Configure a PVLAN host port.

Step 6 Associate a host port with a PVLAN.

Step 7 Verify a PVLAN configuration.

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	(Optional) switch(config-vlan)# **show feature**	Displays features available and whether they are enabled globally.
Step 4	(Optional) switch(config-vlan)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

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 23.
- 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.

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 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)# **exit**	Exits VLAN configuration mode. **Note** You must exit VLAN configuration mode for the configurations to take effect.
Step 5	(Optional) switch(config)# **show vlan private-vlan**	Displays the PVLAN configuration.

	Command or Action	Purpose
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

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

switch(config)#
```

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.
- 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.

- 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.

	Command or Action	Purpose
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)# **exit**	Exits VLAN configuration mode. **Note** You must exit VLAN configuration mode for the configurations to take effect.
Step 5	(Optional) switch(config)# **show vlan private-vlan**	Displays the PVLAN configuration.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

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

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

switch(config)#
```

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.

	Command or Action	Purpose
Step 2	Required: 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)# **exit**	Exits VLAN configuration mode.
Step 5	(Optional) switch(config)# **show vlan private-vlan**	Displays the PVLAN configuration.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

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

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 vCenter Server.
Step 7	switch(config-port-profile)# **state enabled**	Enables the port profile and applies its configuration to the assigned ports.
Step 8	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

```
switch # configure terminal
switch(config)# port-profile type vethernet pvlan_community_303
switch(config-port-prof)# switchport mode private-vlan host
switch(config-port-prof)# switchport private-vlan host-association 202 303
switch(config-port-prof)# no shut
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# state enabled
```

Configuring a vEthernet Port Using PVLAN Port-Profile

You can associate a vEthernet port-profile with PVLAN configuration to a virtual machine adapter in the vCenter server.

Before you begin

- You should know the VMware vCenter login credentials.
- You should be logged into vCenter Server.
- You have information about the virtual machine adapeter to which por-profile will be attached.

Procedure

Step 1 Navigate to VMware vCenter Server.

Step 2 On the **Navigator** pane, choose the virtual machine to which you want to bind the port-profile

Step 3 Right click a virtual machine, and from the pop-up menu, select **Edit Settings**.

Step 4 In the **Edit Settings** dialog box, select a port-profile from the drop-down list for a network adapter.

Step 5 Click **Ok**

A vEthernet port with the selected port-profile configuration is brought up on Nexus 1000VE.

Step 6 (Optional) Use the **show interface brief** command to check whether the new interface is configured properly or not.

```
N1KV_140_NG# show interface brief

--------------------------------------------------------------------------------
Port     VRF          Status IP Address                              Speed    MTU
--------------------------------------------------------------------------------
mgmt0    --           up     10.197.128.234                          1000     1500

--------------------------------------------------------------------------------
Ethernet     VLAN    Type Mode    Status  Reason                   Speed    Port
Interface                                                                   Ch #
--------------------------------------------------------------------------------
Eth3/1       1       eth  trunk   up      none                     10G
Eth4/1       1       eth  trunk   up      none                     10G

--------------------------------------------------------------------------------
Vethernet    VLAN/    Type Mode    Status  Reason                   MTU  Module
             Segment
--------------------------------------------------------------------------------
Veth1        1500     virt access  up      none                     1500 3
```

Step 7 (Optional) Use the **show running-config interface vethernet** command to view the summary of the interface configuration.

```
N1KV_140_NG# show run interface vethernet 1

!Command: show running-config interface Vethernet1
!Time: Thu Jul  5 15:18:19 2018

version 5.2(1)SV5(1.1)

interface Vethernet1
  inherit port-profile pvlan_community_303
  description HPING_229_210, Net Adapter 1
```

```
vmware dvport 0 dvswitch uuid "50 37 b6 e5 fd 04 3f 61-9f f5 b0 e1 5b 00 db f3"
vmware vm mac 0050.56B7.C299
```

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 port-profile mode.
Step 3	switch(config-port-prof)# **switchport mode trunk**	Designates that the interfaces are to be used as trunking ports.
Step 4	switch(config-port-prof)# **switchport mode private-vlan trunk promiscuous**	In the running configuration, designates the interface as a promiscuous PVLAN trunk port.

	Command or Action	Purpose
Step 5	switch(config-port-prof)# **switchport private-vlan trunk allowed vlan** *vlan_range*	Sets the allowed VLANs and VLAN IDs when the interface is in PVLAN trunking mode.
Step 6	switch(config-port-prof)# **switchport private-vlan mapping trunk** *primary_vlan_ID* {*secondary_vlan_list* | **add** *secondary_vlan_list* | **remove** *secondary_vlan_list*}	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	switch(config-port-prof)# **no shut**	Enables the port profile.
Step 8	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 9	switch(config-port-profile)# **state enabled**	Enables the port profile and applies its configuration to the assigned ports.
Step 10	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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 Layer 2 Port Profile as a Promiscuous Access Port

You can configure a Layer 2 interface as a promiscuous access port.

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** command does not decide or override the access configuration of a port.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **port-profile type vethernet** *name*	Places you in port-profile mode.
Step 3	switch(config-port-prof)# **switchport mode private-vlan promiscuous**	In the running configuration, designates the interface as a promiscuous PVLAN port.
Step 4	switch(config-port-prof)# **switchport private-vlan mapping** *primary_vlan_ID* {*secondary_vlan_list* | **add** *secondary_vlan_list* | **remove** *secondary_vlan_list*}	Maps the PVLAN access port to a primary VLAN and to the selected secondary VLANs in the running configuration.
Step 5	switch(config-port-prof)# **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	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

```
switch # configure terminal
switch(config)# port-profile type vethernet pvlan-prom-pp
switch(config-port-prof)# switchport mode private-vlan promiscuous
switch(config-port-prof)# switchport private-vlan mapping 202 303
switch(config-port-prof)# no shut
switch(config-port-prof)# vmware port-group
switch(config-port-prof)# state enabled
```

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)# **exit**	Exits VLAN configuration mode.
Step 5	(Optional) switch(config)# **show vlan private-vlan**	Displays the PVLAN configuration.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Example

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

switch(config)#
```

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 *vlan-id*	Displays VLAN information.
show vlan private-vlan [*type*]	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)# exit
switch(config)# vlan 440,450
switch(config-vlan)# private-vlan isolated
switch(config-vlan)# exit
switch(config)# vlan 202
switch(config-vlan)# private-vlan primary
switch(config-vlan)# private-vlan association 303,440
switch(config-vlan)# exit
switch(config)# vlan 210
switch(config-vlan)# private-vlan primary
switch(config-vlan)# private-vlan association 310,450
switch(config-vlan)# exit
```

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 5.2(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#
```