VLAN Configuration Guide, Cisco IOS XE 17.16.x (Catalyst 9400 Switches)

First Published: 2024-12-11

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.

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 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.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

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: https://www.cisco.com/c/en/us/about/legal/trademarks.html. 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. (1721R)

© 2024 Cisco Systems, Inc. All rights reserved.

Preface

This preface describes the conventions of this document and information on how to obtain other documentation. It also provides information on what's new in Cisco product documentation.

- Document Conventions , on page iii
- Related Documentation, on page v
- Obtaining Documentation and Submitting a Service Request, on page v

Document Conventions

This document uses the following conventions:

Convention	Description
^ or Ctrl	Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination **^D** or **Ctrl-D** means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
bold font	Commands and keywords and user-entered text appear in **bold** font.
Italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
`Courier` font	Terminal sessions and information the system displays appear in `courier` font.
`Bold Courier` font	**`Bold Courier`** font indicates text that the user must enter.
[x]	Elements in square brackets are optional.
...	An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.
|	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
[x | y]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Convention	Description
{x | y}	Required alternative keywords are grouped in braces and separated by vertical bars.
[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.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	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.

Reader Alert Conventions

This document may use the following conventions for reader alerts:

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

Tip Means *the following information will help you solve a problem.*

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

Timesaver Means *the described action saves time.* You can save time by performing the action described in the paragraph.

Take note of the following general safety warnings:

Warning IMPORTANT SAFETY INSTRUCTIONS

Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Read the installation instructions before using, installing, or connecting the system to the power source. Use the statement number at the beginning of each warning statement to locate its translation in the translated safety warnings for this device.

SAVE THESE INSTRUCTIONS

Related Documentation

Note Before installing or upgrading the device, refer to the device release notes.

- Cisco Catalyst 9200 Series Switches documentation, located at:

  https://www.cisco.com/c/en/us/products/switches/catalyst-9200-series-switches/index.html

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/c/en/us/td/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.

CONTENTS

PREFACE

Preface iii

Document Conventions iii

Related Documentation v

Obtaining Documentation and Submitting a Service Request v

CHAPTER 1

Configuring VTP 1

Prerequisites for VTP 1

Restrictions for VTP 1

Information About VTP 2

VTP 2

VTP Domain 2

VTP Modes 3

VTP Advertisements 3

VTP Version 2 4

VTP Version 3 4

VTP Pruning 5

VTP and Device Stacks 6

VTP Configuration Guidelines 6

VTP Configuration Requirements 6

VTP Settings 7

Domain Names for Configuring VTP 7

Passwords for the VTP Domain 7

VTP Version 8

How to Configure VTP 8

Configuring VTP Mode 9

Configuring a VTP Version 3 Password 10

Configuring a VTP Version 3 Primary Server 12
Enabling the VTP Version 12
Enabling VTP Pruning 14
Configuring VTP on a Per-Port Basis 15
Adding a VTP Client to a VTP Domain 16
Monitoring VTP 18
Configuration Examples for VTP 18
Example: Configuring a Device as the Primary Server 18
Where to Go Next 19
Feature History for VTP 19

CHAPTER 2 **Configuring VLANs** 21
Prerequisites for VLANs 21
Restrictions for VLANs 21
Information About VLANs 22
Logical Networks 22
Supported VLANs 23
VLAN Port Membership Modes 23
VLAN Configuration Files 24
Normal-Range VLAN Configuration Guidelines 24
Extended-Range VLAN Configuration Guidelines 25
How to Configure VLANs 25
How to Configure Normal-Range VLANs 25
Creating or Modifying an Ethernet VLAN 26
Deleting a VLAN 28
Assigning Static-Access Ports to a VLAN 29
How to Configure Extended-Range VLANs 30
Creating an Extended-Range VLAN 31
Monitoring VLANs 32
Where to Go Next 33
Feature History for VLAN 33

CHAPTER 3 **Configuring Voice VLANs** 35
Prerequisites for Voice VLANs 35

Restrictions for Voice VLANs 35
Information About Voice VLAN 36
Voice VLANs 36
Cisco IP Phone Voice Traffic 36
Cisco IP Phone Data Traffic 36
Voice VLAN Configuration Guidelines 37
How to Configure Voice VLANs 38
Configuring Cisco IP Phone Voice Traffic 38
Configuring the Priority of Incoming Data Frames 40
Monitoring Voice VLAN 41
Where to Go Next 41
Feature History for Voice VLAN 41

CHAPTER 4 **Configuring VLAN Trunks** 43
Information About VLAN Trunks 43
Trunking Overview 43
Trunking Modes 43
Layer 2 Interface Modes 44
Allowed VLANs on a Trunk 44
Load Sharing on Trunk Ports 45
Network Load Sharing Using STP Priorities 45
Network Load Sharing Using STP Path Cost 45
Feature Interactions 45
Prerequisites for VLAN Trunks 46
Restrictions for VLAN Trunks 46
How to Configure VLAN Trunks 47
Configuring an Ethernet Interface as a Trunk Port 47
Configuring a Trunk Port 47
Defining the Allowed VLANs on a Trunk 49
Changing the Pruning-Eligible List 50
Configuring the Native VLAN for Untagged Traffic 52
Configuring Trunk Ports for Load Sharing 53
Configuring Load Sharing Using STP Port Priorities 53
Configuring Load Sharing Using STP Path Cost 56

Feature History for VLAN Tunks 58

CHAPTER 5 **Configuring Private VLANs 61**

Prerequisites for Private VLANs 61

Restrictions for Private VLANs 61

Information About Private VLANs 62

Private VLAN Domains 62

Secondary VLANs 63

Private VLANs Ports 63

Private VLANs in Networks 64

IP Addressing Scheme with Private VLANs 64

Private VLANs Across Multiple Switches 65

Standard Trunk Ports 65

Isolated Private VLAN Trunk Ports 66

Promiscuous Private VLAN Trunk Ports 67

Private-VLAN Interaction with Other Features 68

Private VLANs and Unicast, Broadcast, and Multicast Traffic 68

Private VLANs and SVIs 68

Private VLANs and Switch Stacks 69

Private VLAN with Dynamic MAC Address 69

Private VLAN with Static MAC Address 69

Private VLAN Interaction with VACL/QOS 70

Private VLANs and HA Support 70

Private-VLAN Configuration Guidelines 70

Default Private-VLAN Configurations 70

Secondary and Primary VLAN Configuration 70

Private VLAN Port Configuration 72

How to Configure Private VLANs 73

Configuring Private VLANs 73

Configuring and Associating VLANs in a Private VLAN 74

Configuring a Layer 2 Interface as a Private VLAN Host Port 77

Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port 78

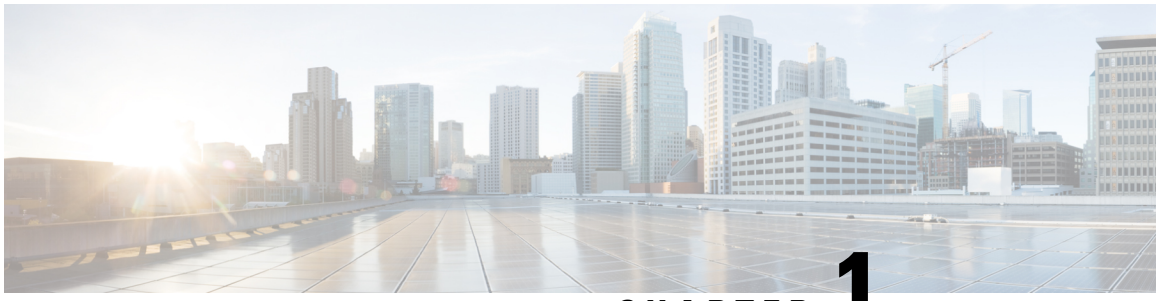
Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port 80

Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port 82

Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port on a Portchannel 83
Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port on a Portchannel 85
Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface 87
Monitoring Private VLANs 89
Configuration Examples for Private VLANs 89
Example: Configuring and Associating VLANs in a Private VLAN 89
Example: Configuring an Interface as a Host Port 90
Example: Configuring an Interface as a Private VLAN Promiscuous Port 90
Example: Mapping Secondary VLANs to a Primary VLAN Interface 90
Example: Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port 91
Example: Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port 92
Example: Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port on a Portchannel 92
Example: Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port on a Portchannel 93
Example: Monitoring Private VLANs 94
Where to Go Next 94
Additional References 94
Feature History for Private VLANs 95

CHAPTER 6 **Configuring Layer 3 Subinterfaces** 97
Restrictions for Configuring Layer 3 Subinterfaces 97
Information About Layer 3 Subinterfaces 98
How to Configure Layer 3 Subinterfaces 99
Example: Configuring Layer 3 Subinterfaces 100
Feature History for Layer 3 Subinterfaces 100

CHAPTER 7 **Configuring Wired Dynamic PVLAN** 103
Restrictions for Wired Dynamic PVLAN 103
Information About Wired Dynamic PVLAN 103
Configuring Wired Dynamic PVLAN 105
Feature History for Wired Dynamic PVLAN 110



CHAPTER 1

Configuring VTP

- Prerequisites for VTP, on page 1
- Restrictions for VTP, on page 1
- Information About VTP, on page 2
- How to Configure VTP, on page 8
- Monitoring VTP, on page 18
- Configuration Examples for VTP, on page 18
- Where to Go Next, on page 19
- Feature History for VTP, on page 19

Prerequisites for VTP

Before you create VLANs, you must decide whether to use the VLAN Trunking Protocol (VTP) in your network. Using VTP, you can make configuration changes centrally on one or more devices and have those changes automatically communicated to all the other devices in the network. Without VTP, you cannot send information about VLANs to other devices.

VTP is designed to work in an environment where updates are made on a single device and are sent through VTP to other devices in the domain. It does not work well in a situation where multiple updates to the VLAN database occur simultaneously on devices in the same domain, which would result in an inconsistency in the VLAN database.

You can enable or disable VTP per port by entering the [**no**] **vtp** interface configuration command. When you disable VTP on trunking ports, all VTP instances for that port are disabled. You cannot set VTP to *off* for the MST database and *on* for the VLAN database on the same port.

When you globally set VTP mode to off, it applies to all the trunking ports in the system. However, you can specify on or off on a per-VTP instance basis. For example, you can configure the device as a VTP server for the VLAN database but with VTP *off* for the MST database.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the device or device stack and that this trunk port is connected to the trunk port of another device. Otherwise, the device cannot receive any VTP advertisements.

Restrictions for VTP

The following are restrictions for a VTP:

Caution Before adding a VTP client device to a VTP domain, always verify that its VTP configuration revision number is lower than the configuration revision number of the other devices in the VTP domain. Devices in a VTP domain always use the VLAN configuration of the device with the highest VTP configuration revision number. If you add a device that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain.

Information About VTP

The following sections provide information about VTP and VTP configuration:

VTP

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

VTP functionality is supported across the stack, and all devices in the stack maintain the same VLAN and VTP configuration inherited from the active device. When a device learns of a new VLAN through VTP messages or when a new VLAN is configured by the user, the new VLAN information is communicated to all devices in the stack.

When a device joins the stack or when stacks merge, the new devices get VTP information from the active device.

VTP Domain

A VTP domain (also called a VLAN management domain) consists of one device or several interconnected devices under the same administrative responsibility sharing the same VTP domain name. A device can be in only one VTP domain. You make global VLAN configuration changes for the domain.

By default, the device is in the VTP no-management-domain state until it receives an advertisement for a domain over a trunk link (a link that carries the traffic of multiple VLANs) or until you configure a domain name. You can create or modify VLANs on a VTP server without specifying the domain name. However, when the management domain name is not specified VLAN information is not propagated over the network.

If the device receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The device then ignores advertisements with a different domain name or an earlier configuration revision number.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all devices in the VTP domain. VTP advertisements are sent over all IEEE trunk connections, including IEEE 802.1Q. VTP dynamically maps VLANs with unique names and internal index associates across multiple LAN types. Mapping eliminates excessive device administration required from network administrators.

If you configure a device for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other devices in the domain, and they affect only the individual device. However, configuration changes made when the device is in this mode are saved in the device running configuration and can be saved to the device startup configuration file.

VTP Modes

Table 1: VTP Modes

VTP Mode	Description
VTP server	In VTP server mode, you can create, modify, and delete VLANs, and specify other configuration (such as the VTP version) for the entire VTP domain. VTP servers advertise their VLAN configura devices in the same VTP domain and synchronize their VLAN configurations with other devices advertisements received over trunk links. VTP server is the default mode. In VTP server mode, VLAN configurations are saved in NVRAM. If the device detects a failure a configuration to NVRAM, VTP mode automatically changes from server mode to client mode. If the device cannot be returned to VTP server mode until the NVRAM is functioning.
VTP client	A VTP client functions like a VTP server and transmits and receives VTP updates on its trunks, b create, change, or delete VLANs on a VTP client. VLANs are configured on another device in th is in server mode. In VTP versions 1 and 2 in VTP client mode, VLAN configurations are not saved in NVRAM. In 3, VLAN configurations are saved in NVRAM in client mode.
VTP transparent	VTP transparent devices do not participate in VTP. A VTP transparent device does not advertise configuration and does not synchronize its VLAN configuration based on received advertisements VTP version 2 or version 3, transparent devices do forward VTP advertisements that they receive devices through their trunk interfaces. You can create, modify, and delete VLANs on a device in V mode. In VTP versions 1 and 2, the device must be in VTP transparent mode when you create private VL they are configured, you should not change the VTP mode from transparent to client or server mode 3 also supports private VLANs in client and server modes. When private VLANs are configured, the VTP mode from transparent to client or server mode. When the device is in VTP transparent mode, the VTP and VLAN configurations are saved in N they are not advertised to other devices. In this mode, VTP mode and domain name are saved in t running configuration, and you can save this information in the device startup configuration file b **copy running-config startup-config** privileged EXEC command. In a device stack, the running configuration and the saved configuration are the same for all devic
VTP off	A device in VTP off mode functions in the same manner as a VTP transparent device, except that forward VTP advertisements on trunks.

VTP Advertisements

Each device in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring devices receive these advertisements and update their VTP and VLAN configurations as necessary.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the switch stack and that this trunk port is connected to the trunk port of another switch. Otherwise, the switch cannot receive any VTP advertisements.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN
- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (including IEEE 802.1Q)
- VLAN name
- VLAN type
- VLAN state
- Additional VLAN configuration information specific to the VLAN type

In VTP version 3, VTP advertisements also include the primary server ID, an instance number, and a start index.

VTP Version 2

If you use VTP in your network, you must decide which version of VTP to use. By default, VTP operates in version 1.

VTP version 2 supports these features that are not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring Bridge Relay Function (TrBRF) and Token Ring Concentrator Relay Function (TrCRF) VLANs.
- Unrecognized Type-Length-Value (TLV) support—A VTP server or client propagates configuration changes to its other trunks, even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM when the device is operating in VTP server mode.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent device inspects VTP messages for the domain name and version and forwards a message only if the version and domain name match. Although VTP version 2 supports only one domain, a VTP version 2 transparent device forwards a message only when the domain name matches.
- Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are not performed when new information is obtained from a VTP message or when information is read from NVRAM. If the MD5 digest on a received VTP message is correct, its information is accepted.

VTP Version 3

VTP version 3 supports these features that are not supported in version 1 or version 2:

- Enhanced authentication—You can configure the authentication as **hidden** or **secret**. When **hidden**, the secret key from the password string is saved in the VLAN database file, but it does not appear in plain text in the configuration. Instead, the key associated with the password is saved in hexadecimal format in the running configuration. You must reenter the password if you enter a takeover command in the domain. When you enter the **secret** keyword, you can directly configure the password secret key.
- Support for extended range VLAN (VLANs 1006 to 4094) database propagation—VTP versions 1 and 2 propagate only VLANs 1 to 1005.

Note VTP pruning still applies only to VLANs 1 to 1005, and VLANs 1002 to 1005 are still reserved and cannot be modified.

- Cisco Catalyst 9400 Series Supervisor 1 Module (C9400-SUP-1) and Cisco Catalyst 9400 Series Supervisor 1XL Module (C9400-SUP-1XL) support VLAN IDs from 1 to 4094, however only 1000 VLANs can be active on the device. Cisco Catalyst 9400X Series Supervisor 2 Module (9400X-SUP-2) and Cisco Catalyst 9400X Series Supervisor 2XL Module (C9400X-SUP-2XL) Supervisor Modules support VLANs with IDs from 1-4094 are supported but only 4000 VLANs can be active on the device.
- Private VLAN support.
- Support for any database in a domain—In addition to propagating VTP information, version 3 can propagate Multiple Spanning Tree (MST) protocol database information. A separate instance of the VTP protocol runs for each application that uses VTP.
- VTP primary server and VTP secondary servers—A VTP primary server updates the database information and sends updates that are honored by all devices in the system. A VTP secondary server can only back up the updated VTP configurations received from the primary server to its NVRAM.

  By default, all devices come up as secondary servers. You can enter the **vtp primary** privileged EXEC command to specify a primary server. Primary server status is only needed for database updates when the administrator issues a takeover message in the domain. You can have a working VTP domain without any primary servers. Primary server status is lost if the device reloads, after a switchover, or domain parameters change, even when a password is configured on the device.
- With VTP version 3 in server mode the VLAN configuration is saved into vlan.dat file. VLAN configuration is not saved in NVRAM as is the case in the transparent mode. While taking a backup of the switch configuration, you also have to take a backup of the vlan.dat file.

Note VTP versions 1 and 2 are capable of publishing only standard VLANs (VLANs 1 to 1001) and extended VLANs (VLANs 1006 to 4094) are stored locally in the flash drive or the running configuration. VTP version 3 is capable of publishing extended VLANs to the entire VTP domain and extended VLANs are not stored locally.

VTP Pruning

VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a device floods broadcast, multicast,

and unknown unicast traffic across all trunk links within a VTP domain even though receiving devices might discard them. VTP pruning is disabled by default.

VTP pruning blocks unneeded flooded traffic to VLANs on trunk ports that are included in the pruning-eligible list. Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible device trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported in all VTP versions.

With VTP versions 1 and 2, when you enable pruning on the VTP server, it is enabled for the entire VTP domain. In VTP version 3, you must manually enable pruning on each device in the domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that trunk only (not on all devices in the VTP domain).

VTP pruning takes effect several seconds after you enable it. VTP pruning does not prune traffic from VLANs that are pruning-ineligible. VLAN 1 and VLANs 1002 to 1005 are always pruning-ineligible; traffic from these VLANs cannot be pruned. Extended-range VLANs (VLAN IDs higher than 1005) are also pruning-ineligible.

VTP and Device Stacks

VTP configuration is the same in all members of a device stack. When the device stack is in VTP server or client mode, all devices in the stack carry the same VTP configuration. When VTP mode is transparent, the stack is not taking part in VTP.

- When a device joins the stack, it inherits the VTP and VLAN properties of the active device.
- All VTP updates are carried across the stack.
- When VTP mode is changed in a device in the stack, the other devices in the stack also change VTP mode, and the device VLAN database remains consistent.

VTP version 3 functions the same on a standalone device or a stack except when the switch stack is the primary server for the VTP database. In this case, the MAC address of the active device is used as the primary server ID. If the active device reloads or is powered off, a new active device is elected.

- If you do not configure the persistent MAC address feature, when the new active device is elected, it sends a takeover message with the new active MAC address as the primary server.
- If a persistent MAC address is configured, the new active device waits for the configured timer value. If the previous active device does not rejoin the stack during this time, then the new active device issues the takeover message.

VTP Configuration Guidelines

This section provides information about VTP configuration guidelines:

VTP Configuration Requirements

When you configure VTP, you must configure a trunk port so that the device can send and receive VTP advertisements to and from other devices in the domain.

VTP versions 1 and 2 do not support private VLANs. VTP version 3 does support private VLANs. If you configure private VLANs, the device must be in VTP transparent mode. When private VLANs are configured on the device, do not change the VTP mode from transparent to client or server mode.

VTP Settings

The VTP information is saved in the VTP VLAN database. When VTP mode is transparent, the VTP domain name and mode are also saved in the device running configuration file, and you can save it in the device startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. You must use this command if you want to save VTP mode as transparent, even if the device resets.

When you save VTP information in the device startup configuration file and reboot the device, the device configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration do not match the VLAN database, the domain name and VTP mode and configuration for VLAN IDs 1 to 1005 use the VLAN database information.

Domain Names for Configuring VTP

When configuring VTP for the first time, you must always assign a domain name. You must configure all devices in the VTP domain with the same domain name. Devices in VTP transparent mode do not exchange VTP messages with other devices, and you do not need to configure a VTP domain name for them.

Note If the NVRAM and DRAM storage is sufficient, all devices in a VTP domain should be in VTP server mode.

Caution Do not configure a VTP domain if all devices are operating in VTP client mode. If you configure the domain, it is impossible to make changes to the VLAN configuration of that domain. Make sure that you configure at least one device in the VTP domain for VTP server mode.

Passwords for the VTP Domain

You can configure a password for the VTP domain, but it is not required. If you do configure a domain password, all domain devices must share the same password and you must configure the password on each device in the management domain. Devices without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a device that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the device accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new device to an existing network with VTP capability, the new device learns the domain name only after the applicable password has been configured on it.

Caution When you configure a VTP domain password, the management domain does not function properly if you do not assign a management domain password to each device in the domain.

VTP Version

Follow these guidelines when deciding which VTP version to implement:

- All devices in a VTP domain must have the same domain name, but they do not need to run the same VTP version.
- A VTP version 2-capable device can operate in the same VTP domain as a device running VTP version 1 if version 2 is disabled on the version 2-capable device (version 2 is disabled by default).
- If a device running VTP version 1, but capable of running VTP version 2, receives VTP version 3 advertisements, it automatically moves to VTP version 2.
- If a device running VTP version 3 is connected to a device running VTP version 1, the VTP version 1 device moves to VTP version 2, and the VTP version 3 device sends scaled-down versions of the VTP packets so that the VTP version 2 device can update its database.
- A device running VTP version 3 cannot move to version 1 or 2 if it has extended VLANs.
- Do not enable VTP version 2 on a device unless all of the devices in the same VTP domain are version-2-capable. When you enable version 2 on a device, all of the version-2-capable devices in the domain enable version 2. If there is a version 1-only device, it does not exchange VTP information with device that have version 2 enabled.
- Cisco recommends placing VTP version 1 and 2 device at the edge of the network because they do not forward VTP version 3 advertisements.
- If there are TrBRF and TrCRF Token Ring networks in your environment, you must enable VTP version 2 or version 3 for Token Ring VLAN switching to function properly. To run Token Ring and Token Ring-Net, disable VTP version 2.
- For VTP version 1 and version 2, the device must be in VTP transparent mode when you create extended-range VLANs. VTP version 3 also supports creating extended-range VLANs in client or server mode.
- When a VTP version 3 device trunk port receives messages from a VTP version 2 device, it sends a scaled-down version of the VLAN database on that particular trunk in VTP version 2 format. A VTP version 3 device does not send VTP version 2-formatted packets on a trunk unless it first receives VTP version 2 packets on that trunk port.
- When a VTP version 3 device detects a VTP version 2 device on a trunk port, it continues to send VTP version 3 packets, in addition to VTP version 2 packets, to allow both kinds of neighbors to coexist on the same trunk.
- A VTP version 3 device does not accept configuration information from a VTP version 2 or version 1 device.
- Two VTP version 3 regions can only communicate in transparent mode over a VTP version 1 or version 2 region.
- Devices that are only VTP version 1 capable cannot interoperate with VTP version 3 devices.

How to Configure VTP

The following sections provide information about Configuring VTP:

Configuring VTP Mode

You can configure VTP mode as one of these:

- VTP server mode—In VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.
- VTP client mode—In VTP client mode, you cannot change its VLAN configuration. The client device receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.
- VTP transparent mode—In VTP transparent mode, VTP is disabled on the device. The device does not send VTP updates and does not act on VTP updates received from other devices. However, a VTP transparent device running VTP version 2 does forward received VTP advertisements on its trunk links.
- VTP off mode—VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

When you configure a domain name, it cannot be removed; you can only reassign a device to a different domain.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**vtp domain** *domain-name* **Example:** `Device(config)# vtp domain eng_group`	Configures the VTP administrative-domain name. The name can be 1 to 32 characters. All devices operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name. This command is optional for modes other than server mode. VTP server mode requires a domain name. If the device has a trunk connection to a VTP domain, the device learns the domain name from the VTP server in the domain. You should configure the VTP domain before configuring other VTP parameters.

	Command or Action	Purpose
Step 4	**vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**} **Example:** `Device(config)# vtp mode server`	Configures the device for VTP mode (client, server, transparent, or off). • **vlan**—The VLAN database is the default if none are configured. • **mst**—The multiple spanning tree (MST) database. • **unknown**—An unknown database type.
Step 5	**vtp password** *password* **Example:** `Device(config)# vtp password mypassword`	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each device in the domain.
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 7	**show vtp status** **Example:** `Device# show vtp status`	Verifies your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves the configuration in the startup configuration file. Only VTP mode and domain name are saved in the device running configuration and can be copied to the startup configuration file.

Configuring a VTP Version 3 Password

You can configure a VTP version 3 password on the device.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	`Device> enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**vtp version 3** **Example:** `Device(config)# vtp version 3`	Enables VTP version 3 on the device. The default is VTP version 1.
Step 4	**vtp password** *password* [**hidden** | **secret**] **Example:** `Device(config)# vtp password mypassword hidden`	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. • (Optional) **hidden**: Saves the secret key generated from the password string in the `nvram:vlan.dat` file. If you configure a takeover by configuring a VTP primary server, you are prompted to reenter the password. • (Optional) **secret**: Directly configures the password. The secret password must contain 32 hexadecimal characters.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show vtp password** **Example:** `Device# show vtp password`	Verifies whether the VTP password is configured or not.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a VTP Version 3 Primary Server

When you configure a VTP server as a VTP primary server, the takeover operation starts.

Procedure

	Command or Action	Purpose
Step 1	**vtp version 3** **Example:** `Device(config)# vtp version 3`	Enables VTP version 3 on the device. The default is VTP version 1.
Step 2	**vtp primary** [**vlan** | **mst**] [**force**] **Example:** `Device# vtp primary vlan force`	Changes the operational state of a device from a secondary server (the default) to a primary server and advertises the configuration to the domain. If the device password is configured as **hidden**, you are prompted to reenter the password. • (Optional) **vlan**—Selects the VLAN database as the takeover feature. This is the default. • (Optional) **mst**—Selects the multiple spanning tree (MST) database as the takeover feature. • (Optional) **force**—Overwrites the configuration of any conflicting servers. If you do not enter **force**, you are prompted for confirmation before the takeover.

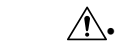
Enabling the VTP Version

VTP version 2 and version 3 are disabled by default.

- When you enable VTP version 2 on a device , every VTP version 2-capable device in the VTP domain enables version 2. To enable VTP version 3, you must manually configure it on each device.
- With VTP versions 1 and 2, you can configure the version only on devices in VTP server or transparent mode. If a device is running VTP version 3, you can change to version 2 when the device is in client mode if no extended VLANs exist, no private VLANs exist, and no hidden password was configured.

Caution VTP version 1 and VTP version 2 are not interoperable on devices in the same VTP domain. Do not enable VTP version 2 unless every device in the VTP domain supports version 2.

- In TrCRF and TrBRF Token Ring environments, you must enable VTP version 2 or VTP version 3 for Token Ring VLAN switching to function properly. For Token Ring and Token Ring-Net media, disable VTP version 2.



Caution In VTP version 3, both the primary and secondary servers can exist on an instance in the domain.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**vtp version** {**1** | **2** | **3**} **Example:** `Device(config)# vtp version 2`	Enables the VTP version on the device. The default is VTP version 1.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show vtp status** **Example:** `Device# show vtp status`	Verifies that the configured VTP version is enabled.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling VTP Pruning

Before you begin

VTP pruning is not designed to function in VTP transparent mode. If one or more devices in the network are in VTP transparent mode, you should do one of these actions:

- Turn off VTP pruning in the entire network.
- Turn off VTP pruning by making all VLANs on the trunk of the device upstream to the VTP transparent device pruning ineligible.

To configure VTP pruning on an interface, use the **switchport trunk pruning vlan** interface configuration command. VTP pruning operates when an interface is trunking. You can set VLAN pruning-eligibility, whether or not VTP pruning is enabled for the VTP domain, whether or not any given VLAN exists, and whether or not the interface is currently trunking.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**vtp pruning** **Example:** `Device(config)# vtp pruning`	Enables pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one device in VTP server mode.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show vtp status** **Example:** `Device# show vtp status`	Verifies your entries in the *VTP Pruning Mode* field of the display.

Configuring VTP on a Per-Port Basis

With VTP version 3, you can enable or disable VTP on a per-port basis. You can enable VTP only on ports that are in trunk mode. Incoming and outgoing VTP traffic are blocked, not forwarded.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet0/1`	Identifies an interface, and enters interface configuration mode.
Step 4	**vtp** **Example:** `Device(config-if)# vtp`	Enables VTP on the specified port.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id* **Example:** `Device# show running-config interface gigabitethernet 1/0/1`	Verifies the change to the port.
Step 7	**show vtp status** **Example:**	Verifies the configuration.

	Command or Action	Purpose
	`Device# show vtp status`	

Adding a VTP Client to a VTP Domain

Follow these steps to verify and reset the VTP configuration revision number on a device *before* adding it to a VTP domain.

Before you begin

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other devices in the VTP domain. Devices in a VTP domain always use the VLAN configuration of the device with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a device that has a revision number higher than the revision number in the VTP domain can erase all VLAN information from the VTP server and VTP domain. With VTP version 3, the VLAN information is not erased.

You can use the **vtp mode transparent** global configuration command to disable VTP on the device and then to change its VLAN information without affecting the other devices in the VTP domain.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**show vtp status** **Example:** `Device# show vtp status`	Checks the VTP configuration revision number. If the number is 0, add the device to the VTP domain. If the number is greater than 0, follow these substeps: • Write down the domain name. • Write down the configuration revision number. • Continue with the next steps to reset the device configuration revision number.
Step 3	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Device# configure terminal`	
Step 4	**vtp domain** *domain-name* **Example:** `Device(config)# vtp domain domain123`	Changes the domain name from the original one displayed in Step 1 to a new name.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. The VLAN information on the device is updated and the configuration revision number is reset to 0.
Step 6	**show vtp status** **Example:** `Device# show vtp status`	Verifies that the configuration revision number has been reset to 0.
Step 7	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 8	**vtp domain** *domain-name* **Example:** `Device(config)# vtp domain domain012`	Enters the original domain name on the device.
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. The VLAN information on the device is updated.
Step 10	**show vtp status** **Example:** `Device# show vtp status`	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.

Monitoring VTP

This section describes commands used to display and monitor the VTP configuration.

You monitor VTP by displaying VTP configuration information: the domain name, the current VTP revision, and the number of VLANs. You can also display statistics about the advertisements sent and received by the device.

Table 2: VTP Monitoring Commands

Command	Purpose
show vtp counters	Displays counters about VTP messages th
show vtp devices [**conflict**]	Displays information about all VTP versi version 3 devices with conflicting primar not display information when the device i
show vtp interface [*interface-id*]	Displays VTP status and configuration fo
show vtp password	Displays whether the VTP password is co
show vtp status	Displays the VTP device configuration in

Configuration Examples for VTP

The following section shows a VTP configuration example:

Example: Configuring a Device as the Primary Server

This example shows how to configure a device as the primary server for the VLAN database (the default) when a hidden or secret password was configured:

```
Device# vtp primary vlan
Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP  domain

VTP Database Conf Switch ID      Primary Server Revision System Name
------------ ---- --------------- -------------- -------- --------------------
VLANDB       Yes  00d0.00b8.1400=00d0.00b8.1400 1        stp7

Do you want to continue (y/n) [n]? y


Device# vtp primary vlan
This system is becoming primary server for feature vlan
Enter VTP Password: mypassword
No conflicting VTP3 devices found.
Do you want to continue? [confirm]
Switch#
*Sep 8 09:23:39.558: %SW_VLAN-4-VTP_PRIMARY_SERVER_CHG: 2c5a.0f1c.9240 has become the primary
 server for the VLAN VTP feature
This switch is becoming Primary server for vlan feature in the VTP domain
```

Where to Go Next

After configuring VTP, you can configure the following:

- VLANs
- VLAN trunking
- Voice VLANs
- Private VLANs

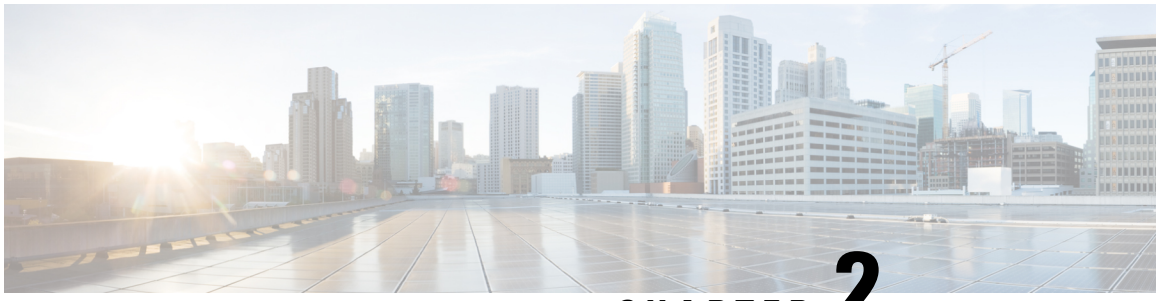
Feature History for VTP

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	VLAN Trunking Protocol (VTP)	VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.
Cisco IOS XE Gibraltar 16.12.4	VTP **show vtp password**command output modification	The **show vtp password**command output now displays whether the password is or is not configured.
Cisco IOS XE Cupertino 17.7.1	VLAN Trunking Protocol (VTP)	This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.



CHAPTER 2

Configuring VLANs

- Prerequisites for VLANs, on page 21
- Restrictions for VLANs, on page 21
- Information About VLANs, on page 22
- How to Configure VLANs, on page 25
- Monitoring VLANs, on page 32
- Where to Go Next, on page 33
- Feature History for VLAN, on page 33

Prerequisites for VLANs

The following are prerequisites and considerations for configuring VLANs:

- Before you create VLANs, you must decide whether to use VLAN Trunking Protocol (VTP) to maintain global VLAN configuration for your network.
- If you plan to configure many VLANs on the device and to not enable routing, you can set the Switch Database Management (SDM) feature to the Access template, which configures system resources to support the maximum number of unicast MAC addresses.
- A VLAN should be present in the device to be able to add it to the VLAN group.

Restrictions for VLANs

The following are restrictions for VLANs:

- The number of Spanning Tree Protocol (STP) virtual ports in the per-VLAN spanning-tree (PVST) or rapid PVST mode is based on the number of trunks, multiplied by the number of active VLANs, plus the number of access ports.

  STP virtual ports = trunks * active VLANs on trunk + number of non-trunk ports.

  Consider the following examples:

  - If a switch has 40 trunk ports (100 active VLANs on each trunk) and 8 access ports, the number of STP virtual ports on this switch would be: 40 * 100 + 8 = 4,008.

- If a switch has 8 trunk ports (200 active VLANs on each trunk) and 40 access ports, the number of STP virtual ports on this switch would be: 8 * 200 + 40 = 1,640

For information about the supported scalability of STP virtual ports, see the Cisco Catalyst 9400 Series Switch Data Sheet.

- The device supports IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.
- The interface VLAN already has an MAC address assigned by default. You can override the interface VLAN MAC address by using thee **mac-address** command. If this command is configured on a single SVI or router port that requires Layer 3 injected packets, all other SVIs or routed ports on the device also must be configured with the same first four most significant bits (4MSB) of the MAC address. For example, if you set the MAC address of any SVI to xxxx.yyyy.zzzz, set the MAC address of all other SVIs to start with xxxx.yyyy. If Layer 3 injected packets are not used, this restriction does not apply.

Note This applies to all Layer 3 ports, SVIs, and routed ports. This does not apply to GigabitEthernet0/0 port.

- When deploying Cisco StackWise Virtual, ensure that VLAN ID 4094 is not used anywhere on the network. All inter-chassis system control communication between stack members is carried over the reserved VLAN ID 4094 from the global range.
- Once a range of interfaces has been bundled, any VLAN interface configuration change must be done only on a port channel. Otherwise, the interfaces will get suspended.

Information About VLANs

The following sections provides information about VLANs:

Logical Networks

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 as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any device port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the 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 router or a device supporting fallback bridging. Because a VLAN is considered a separate logical network, it contains its own bridge Management Information Base (MIB) information and can support its own implementation of spanning tree.

VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the device is assigned manually on an interface-by-interface basis. When you assign device interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed.

The device can route traffic between VLANs by using device virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs.

Supported VLANs

The device supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. VLAN 1 is the default VLAN and is created during system initialization.

In the MSTP mode, you can configure 1000 active VLANs at any point in time.

VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. All of the VLANs except 1002 to 1005 are available for user configuration.

By default, VTP versions 1 and 2 support 20 characters for the VLAN name. VTP version 3 supports 128 characters.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic the port carries and the number of VLANs to which it can belong.

When a port belongs to a VLAN, the device learns and manages the addresses associated with the port on a per-VLAN basis.

Table 3: Port Membership Modes and Characteristics

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN.	VTP is not required. If you do not want VTP to globally propagate information, set the VTP mode to transparent. To participate in VTP, there must be at least one trunk port on the device or the device stack connected to a trunk port of a second device or device stack.
Trunk (IEEE 802.1Q) : • IEEE 802.1Q—Industry-standard trunking encapsulation.	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. You can also modify the pruning-eligible list to block flooded traffic to VLANs on trunk ports that are included in the list.	VTP is recommended but not required. VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP exchanges VLAN configuration messages with other devices over trunk links.
Voice VLAN	A voice VLAN port is an access port attached to a Cisco IP Phone, configured to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.	VTP is not required; it has no effect on a voice VLAN.

VLAN Configuration Files

Configurations for VLAN IDs 1 to 1005 are written to the vlan.dat file (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The vlan.dat file is stored in flash memory. If the VTP mode is transparent, they are also saved in the device running configuration file.

In a device stack, the whole stack uses the same vlan.dat file and running configuration. On some devices, the vlan.dat file is stored in flash memory on the active device.

You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the device, the device configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the VLAN IDs 1 to 1005 use the VLAN database information.
- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.

Note Ensure that you delete the vlan.dat file along with the configuration files before you reset the switch configuration using **write erase** command. This ensures that the switch reboots correctly on a reset.

Normal-Range VLAN Configuration Guidelines

Normal-range VLANs are VLANs with IDs from 1 to 1005.

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- Normal-range VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configurations are also saved in the device running configuration file.
- If the device is in VTP server or VTP transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)
- Before you can create a VLAN, the device must be in VTP server mode or VTP transparent mode. If the device is a VTP server, you must define a VTP domain or VTP will not function.
- The device does not support Token Ring or FDDI media. The device does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.

- A fixed number of spanning tree instances are supported on the device (See the Cisco Catalyst 9400 Series Switch Data Sheet for the latest information). If the device has more active VLANs than the supported number of spanning tree instances, spanning tree is still enabled only on the supported number of VLANs and disabled on all remaining VLANs.

  If you have already used all available spanning-tree instances on a device, adding another VLAN anywhere in the VTP domain creates a VLAN on that device that is not running spanning-tree. If you have the default allowed list on the trunk ports of that device (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent devices that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of devices that have used up their allocation of spanning-tree instances.

  If the number of VLANs on the device exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your device to map multiple VLANs to a single spanning-tree instance.

- When a device in a stack learns a new VLAN or deletes or modifies an existing VLAN (either through VTP over network ports or through the CLI), the VLAN information is communicated to all stack members.

- When a device joins a stack or when stacks merge, VTP information (the vlan.dat file) on the new device will be consistent with the active device.

Extended-Range VLAN Configuration Guidelines

Extended-range VLANs are VLANs with IDs from 1006 to 4094.

Follow these guidelines when creating extended-range VLANs:

- VLAN IDs in the extended range are not saved in the VLAN database and are not recognized by VTP unless the device is running VTP version 3.

- You cannot include extended-range VLANs in the pruning eligible range.

- For VTP version 1 or 2, you can set the VTP mode to transparent in global configuration mode. You should save this configuration to the startup configuration so that the device boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the device resets. If you create extended-range VLANs in VTP version 3, you cannot convert to VTP version 1 or 2.

How to Configure VLANs

The following sections provide information about configuring Normal-Range VLANs and Extended-Range VLANs:

How to Configure Normal-Range VLANs

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID

- VLAN name
- VLAN type
  - Ethernet
  - Fiber Distributed Data Interface [FDDI]
  - FDDI network entity title [NET]
  - TrBRF or TrCRF
  - Token Ring
  - Token Ring-Net
- VLAN state (active or suspended)
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

You can cause inconsistency in the VLAN database if you attempt to manually delete the vlan.dat file. If you want to modify the VLAN configuration, follow the procedures in this section.

Creating or Modifying an Ethernet VLAN

Before you begin

With VTP version 1 and 2, if the device is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database.

The device supports only Ethernet interfaces. Because FDDI and Token Ring VLANs are not locally supported, you only configure FDDI and Token Ring media-specific characteristics for VTP global advertisements to other devices.

Although the device does not support Token Ring connections, a remote device with Token Ring connections could be managed from one of the supported devices. Devices running VTP Version 2 advertise information about these Token Ring VLANs:

- Token Ring TrBRF VLANs
- Token Ring TrCRF VLANs

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**vlan** *vlan-id* **Example:** `Device(config)# vlan 20`	Enters a VLAN ID, and enters VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. **Note** The available VLAN ID range for this command is 1 to 4094.
Step 3	**name** *vlan-name* **Example:** `Device(config-vlan)# name test20`	(Optional) Enters a name for the VLAN. If no name is entered for the VLAN, the default is to append the *vlan-id* value with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4. **Note** By default, VTP versions 1 and 2 support 20 characters for the VLAN name. VTP version 3 supports 128 characters.
Step 4	**media** { **ethernet** | **fd-net** | **fddi** | **tokenring** | **trn-net** } **Example:** `Device(config-vlan)# media ethernet`	Configures the VLAN media type. Command options include: • **ethernet**—Sets the VLAN media type as Ethernet. • **fd-net**—Sets the VLAN media type as FDDI net. • **fddi**—Sets the VLAN media type as FDDI. • **tokenring**—Sets the VLAN media type as Token Ring. • **trn-net**—Sets the VLAN media type as Token Ring net.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Step 7	**show vlan** {**name** *vlan-name* | **id** *vlan-id*}
	Example:
	`Device# show vlan name test20`
	`or`
	`Device# show vlan id 20`
		Verifies your entries.

Deleting a VLAN

When you delete a VLAN from a device that is in VTP server mode, the VLAN is removed from the VLAN database for all devices in the VTP domain. When you delete a VLAN from a device that is in VTP transparent mode, the VLAN is deleted only on that specific device or a device stack.

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.

Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**no vlan** *vlan-id* **Example:** `Device(config)# no vlan 4`	Removes the VLAN by entering the VLAN ID.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show vlan brief** **Example:** `Device# show vlan brief`	Verifies the VLAN removal.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode).

If you assign an interface to a VLAN that does not exist, the new VLAN is created. In case of interface templates, make sure to create the VLAN explicitly before applying the command via templates.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet2/0/1`	Enters the interface to be added to the VLAN.

	Command or Action	Purpose
Step 4	**switchport mode access** **Example:** `Device(config-if)# switchport mode access`	Defines the VLAN membership mode for the port (Layer 2 access port).
Step 5	**switchport access vlan** *vlan-id* **Example:** `Device(config-if)# switchport access vlan 2`	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config interface** *interface-id* **Example:** `Device# show running-config interface gigabitethernet2/0/1`	Verifies the VLAN membership mode of the interface.
Step 8	**show interfaces** *interface-id* **switchport** **Example:** `Device# show interfaces gigabitethernet2/0/1 switchport`	Verifies your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.
Step 9	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

How to Configure Extended-Range VLANs

Extended-range VLANs enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any **switchport** commands that allow VLAN IDs.

With VTP version 1 or 2, extended-range VLAN configurations are not stored in the VLAN database, but because VTP mode is transparent, they are stored in the device running configuration file, and you can save

the configuration in the startup configuration file. Extended-range VLANs created in VTP version 3 are stored in the VLAN database.

You can change only the MTU size, private VLAN, and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.

Creating an Extended-Range VLAN

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**vlan** *vlan-id*
	Example:
	`Device(config)# vlan 2000`
	`Device(config-vlan)#`
		Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4094.

Step 4	**remote-span** **Example:** `Device(config-vlan)# remote-span`	(Optional) Configures the VLAN as the RSPAN VLAN.

Step 5	**exit**
	Example:
	`Device(config-vlan)# exit`
	`Device(config)#`
		Returns to configuration mode.

Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	**show vlan id** *vlan-id* **Example:** `Device# show vlan id 2000`	Verifies that the VLAN has been created.
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring VLANs

Table 4: Privileged EXEC show Commands

Command	Purpose
show interfaces [**vlan** *vlan-id*]	Displays characteristics for all interfaces or for the specified VLAN configured on the device .

Command	Purpose
show vlan [**access-map** *name* | **brief** | **dot1q** { **tag native** } | **filter** [**access-map** | **vlan**] | **group** [**group-name** *name*] | **id** *vlan-id* | **ifindex** | **mtu** | **name** *name* | **private-vlan remote-span** | **summary**]	Displays parameters for all VLANs or the specified VLAN on the device. The following command options are available: • **access-map**—Displays the VLAN access-maps. • **brief**—Displays VTP VLAN status in brief. • **dot1q**—Displays the dot1q parameters. • **filter**—Displays VLAN filter information. • **group**—Displays the VLAN group with its name and the connected VLANs that are available. • **id**—Displays VTP VLAN status by identification number. • **ifindex**—Displays SNMP ifIndex. • **mtu**—Displays VLAN MTU information. • **name**—Displays the VTP VLAN information by specified name. • **private-vlan**—Displays private VLAN information. • **remote-span**–Displays the remote SPAN VLANs. • **summary**—Displays a summary of VLAN information. **Note** The **private-vlan** command option that appears in the device CLI is not supported.

Where to Go Next

After configuring VLANs, you can configure the following:

- VLAN Trunking Protocol (VTP)
- VLAN trunks
- Private VLANs
- Voice VLANs

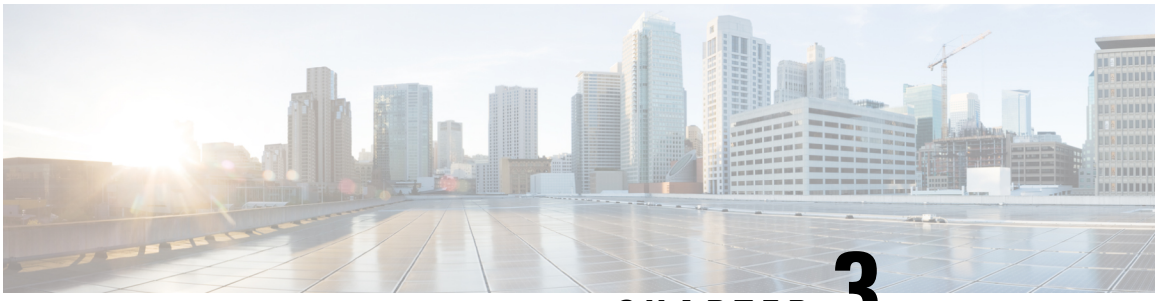
Feature History for VLAN

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	VLAN	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 as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment.
Cisco IOS XE Gibraltar 16.11.1	Support for increased number of spanning tree instances.	Starting with this release, in PVST+ or Rapid PVST+ mode, the device or device stack supports up to 256 spanning-tree instances.
Cisco IOS XE Cupertino 17.7.1	VLAN	This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.



CHAPTER 3

Configuring Voice VLANs

- Prerequisites for Voice VLANs, on page 35
- Restrictions for Voice VLANs, on page 35
- Information About Voice VLAN, on page 36
- How to Configure Voice VLANs, on page 38
- Monitoring Voice VLAN, on page 41
- Where to Go Next, on page 41
- Feature History for Voice VLAN, on page 41

Prerequisites for Voice VLANs

The following are the prerequisites for voice VLANs:

- Voice VLAN configuration is only supported on device access ports; voice VLAN configuration is not supported on trunk ports.

Note Trunk ports can carry any number of voice VLANs, similar to regular VLANs. The configuration of voice VLANs is not supported on trunk ports.

- Before you enable voice VLAN, enable QoS on the device by entering the **trust device cisco-phone** interface configuration command. If you use the auto QoS feature, these settings are automatically configured.
- You must enable CDP on the device port connected to the Cisco IP Phone to send the configuration to the phone. (CDP is globally enabled by default on all device interfaces.)

Restrictions for Voice VLANs

You cannot configure static secure MAC addresses in the voice VLAN.

Information About Voice VLAN

The following sections provide information about Voice VLAN:

Voice VLANs

The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. When the device is connected to a Cisco 7960 IP Phone, the phone sends voice traffic with Layer 3 IP precedence and Layer 2 class of service (CoS) values, which are both set to 5 by default. Because the sound quality of an IP phone call can deteriorate if the data is unevenly sent, the device supports quality of service (QoS) based on IEEE 802.1p CoS. QoS uses classification and scheduling to send network traffic from the device in a predictable manner.

The Cisco 7960 IP Phone is a configurable device, and you can configure it to forward traffic with an IEEE 802.1p priority. You can configure the device to trust or override the traffic priority assigned by a Cisco IP Phone.

Cisco IP Phone Voice Traffic

You can configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. You can configure access ports on the device to send Cisco Discovery Protocol (CDP) packets that instruct an attached phone to send voice traffic to the device in any of these ways:

- In the voice VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN, untagged (no Layer 2 CoS priority value)

Note In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5 for voice traffic and 3 for voice control traffic).

Cisco IP Phone Data Traffic

The device can also process tagged data traffic (traffic in IEEE 802.1Q or IEEE 802.1p frame types) from the device attached to the access port on the Cisco IP Phone. You can configure Layer 2 access ports on the device to send CDP packets that instruct the attached phone to configure the phone access port in one of these modes:

- In trusted mode, all traffic received through the access port on the Cisco IP Phone passes through the phone unchanged.
- In untrusted mode, all traffic in IEEE 802.1Q or IEEE 802.1p frames received through the access port on the Cisco IP Phone receive a configured Layer 2 CoS value. The default Layer 2 CoS value is 0. Untrusted mode is the default.

Note Untagged traffic from the device attached to the Cisco IP Phone passes through the phone unchanged, regardless of the trust state of the access port on the phone.

Voice VLAN Configuration Guidelines

- Because a Cisco 7960 IP Phone also supports a connection to a PC or other device, a port connecting the device to a Cisco IP Phone can carry mixed traffic. You can configure a port to decide how the Cisco IP Phone carries voice traffic and data traffic.
- The voice VLAN should be present and active on the device for the IP phone to correctly communicate on the voice VLAN. Use the **show vlan** privileged EXEC command to see if the VLAN is present (listed in the display). If the VLAN is not listed, create the voice VLAN.
- The Power over Ethernet (PoE) devices are capable of automatically providing power to Cisco pre-standard and IEEE 802.3af-compliant powered devices if they are not being powered by an AC power source.
- The Port Fast feature is automatically enabled when voice VLAN is configured. When you disable voice VLAN, the Port Fast feature is not automatically disabled.
- If the Cisco IP Phone and a device attached to the phone are in the same VLAN, they must be in the same IP subnet. These conditions indicate that they are in the same VLAN:
  - They both use IEEE 802.1p or untagged frames.
  - The Cisco IP Phone uses IEEE 802.1p frames, and the device uses untagged frames.
  - The Cisco IP Phone uses untagged frames, and the device uses IEEE 802.1p frames.
  - The Cisco IP Phone uses IEEE 802.1Q frames, and the voice VLAN is the same as the access VLAN.
- The Cisco IP Phone and a device attached to the phone cannot communicate if they are in the same VLAN and subnet but use different frame types because traffic in the same subnet is not routed (routing would eliminate the frame type difference).
- Voice VLAN ports can also be these port types:
  - Dynamic access port.
  - IEEE 802.1x authenticated port.

Note If you enable IEEE 802.1x on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the phone loses connectivity to the device for up to 30 seconds.

  - Protected port.
  - A source or destination port for a SPAN or RSPAN session.
  - Secure port.

Note When you enable port security on an interface that is also configured with a voice VLAN, you must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN. When the port is connected to a Cisco IP Phone, the phone requires up to two MAC addresses. The phone address is learned on the voice VLAN and might also be learned on the access VLAN. Connecting a PC to the phone requires additional MAC addresses.

How to Configure Voice VLANs

The following sections provide information about configuring Voice VLANs:

Configuring Cisco IP Phone Voice Traffic

You can configure a port connected to the Cisco IP Phone to send CDP packets to the phone to configure the way in which the phone sends voice traffic. The phone can carry voice traffic in IEEE 802.1Q frames for a specified voice VLAN with a Layer 2 CoS value. It can use IEEE 802.1p priority tagging to give voice traffic a higher priority and forward all voice traffic through the native (access) VLAN. The Cisco IP Phone can also send untagged voice traffic or use its own configuration to send voice traffic in the access VLAN. In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5).

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/1`	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 3	**trust device cisco-phone** **Example:** `Device(config-if)# trust device cisco-phone`	Configures the interface to trust incoming traffic packets for the Cisco IP phone.

	Command or Action	Purpose
Step 4	**switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged**} **Example:** `Device(config-if)# switchport voice vlan dot1p`	Configures the voice VLAN. • *vlan-id*—Configures the phone to forward all voice traffic through the specified VLAN. By default, the Cisco IP Phone forwards the voice traffic with an IEEE 802.1Q priority of 5. Valid VLAN IDs are 1 to 4094. • **dot1p**—Configures the device to accept voice and data IEEE 802.1p priority frames tagged with VLAN ID 0 (the native VLAN). By default, the device drops all voice and data traffic tagged with VLAN 0. If configured for 802.1p the Cisco IP Phone forwards the traffic with an IEEE 802.1p priority of 5. • **none**—Allows the phone to use its own configuration to send untagged voice traffic. • **untagged**—Configures the phone to send untagged voice traffic.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

Step 6	Use one of the following:
	• **show interfaces** *interface-id* **switchport**
	• **show running-config interface** *interface-id*
	Example:
	`Device# show interfaces gigabitethernet1/0/1 switchport`
	or
	`Device# show running-config interface gigabitethernet1/0/1`
		Verifies your voice VLAN entries or your QoS and voice VLAN entries.

Step 7	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	`Device# copy running-config startup-config`	

Configuring the Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco IP Phone port. To process tagged data traffic (in IEEE 802.1Q or IEEE 802.1p frames), you can configure the device to send CDP packets to instruct the phone how to send data packets from the device attached to the access port on the Cisco IP Phone. The PC can generate packets with an assigned CoS value. You can configure the phone to not change (trust) or to override (not trust) the priority of frames arriving on the phone port from connected devices.

Follow these steps to set the priority of data traffic received from the non-voice port on the Cisco IP Phone:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/1`	Specifies the interface connected to the Cisco IP Phone, and enters interface configuration mode.
Step 4	**switchport priority extend** {**cos** *value* | **trust**} **Example:** `Device(config-if)# switchport priority extend trust`	Sets the priority of data traffic received from the Cisco IP Phone access port: • **cos** *value*—Configures the phone to override the priority received from the PC or the attached device with the specified CoS value. The value is a number from 0 to 7, with 7 as the highest priority. The default priority is **cos** 0.

	Command or Action	Purpose
		• **trust**—Configures the phone access port to trust the priority received from the PC or the attached device.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Device# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Voice VLAN

To display voice VLAN configuration for an interface, use the **show interfaces** *interface-id* **switchport** privileged EXEC command.

Where to Go Next

After configuring voice VLANs, you can configure the following:

- VLANs
- VLAN Trunking
- VTP
- Private VLANs

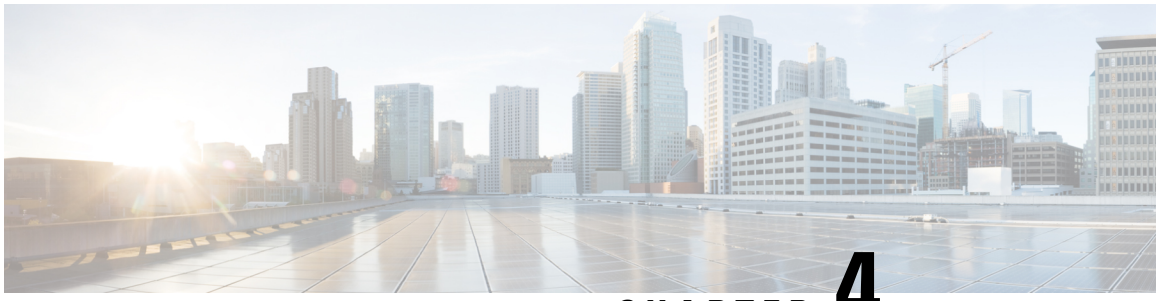
Feature History for Voice VLAN

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	Voice VLAN	The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. You can configure a port connected to the Cisco IP Phone to send CDP packets to the phone to configure the way in which the phone sends voice traffic. The phone can carry voice traffic in IEEE 802.1Q frames for a specified voice VLAN with a Layer 2 CoS value.
Cisco IOS XE Cupertino 17.7.1	Voice VLAN	TThis feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.



CHAPTER 4

Configuring VLAN Trunks

- Information About VLAN Trunks, on page 43
- Prerequisites for VLAN Trunks, on page 46
- Restrictions for VLAN Trunks, on page 46
- How to Configure VLAN Trunks, on page 47
- Feature History for VLAN Tunks, on page 58

Information About VLAN Trunks

The following sections provide information about VLAN Trunks:

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet interfaces and another networking device such as a router or a controller. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network.

IEEE 802.1Q— Industry-standard trunking encapsulation is available on all Ethernet interfaces.

Trunking Modes

Ethernet trunk interfaces support different trunking modes. You can set an interface as trunking or nontrunking or to negotiate trunking with the neighboring interface. To autonegotiate trunking, the interfaces must be in the same VTP domain.

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which is a Point-to-Point Protocol (PPP). However, some internetworking devices might forward DTP frames improperly, which could cause misconfigurations.

Layer 2 Interface Modes

Table 5: Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface.
switchport mode dynamic auto	Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to **trunk** or **desirable** mode. The default switchport mode for all Ethernet interfaces is **dynamic auto**.
switchport mode dynamic desirable	Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to **trunk**, **desirable**, or **auto** mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport nonegotiate	Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is **access** or **trunk**. You must manually configure the neighboring interface as a trunk interface to establish a trunk link.
switchport mode private-vlan	Configures the private VLAN mode.

Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), DTP, and VTP in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port will be added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled, if VTP knows of the VLAN, and if the VLAN is in the allowed list for the port. When VTP detects a newly enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of the enabled VLAN. When VTP detects a new VLAN and the VLAN is not in the allowed list for a trunk port, the trunk port does not become a member of the new VLAN.

Load Sharing on Trunk Ports

Load sharing divides the bandwidth supplied by parallel trunks connecting devices. To avoid loops, STP normally blocks all but one parallel link between the devices. Using load sharing, you divide the traffic between the links according to which VLAN the traffic belongs.

You configure load sharing on trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same device. For load sharing using STP path costs, each load-sharing link can be connected to the same device or to two different devices.

Network Load Sharing Using STP Priorities

When two ports on the same device form a loop, the device uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

Network Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

Feature Interactions

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the device propagates the setting that you entered to all ports in the group:
  - Allowed-VLAN list.
  - STP port priority for each VLAN.
  - STP Port Fast setting.
  - Trunk status:

    If one port in a port group ceases to be a trunk, all ports cease to be trunks.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

Prerequisites for VLAN Trunks

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco devices connected through IEEE 802.1Q trunks, the devices maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

  When you connect a Cisco device to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco device combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q device. However, spanning-tree information for each VLAN is maintained by Cisco devices separated by a cloud of non-Cisco IEEE 802.1Q devices. The non-Cisco IEEE 802.1Q cloud separating the Cisco device is treated as a single trunk link between the devices.

- Make sure the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.

- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure your network is loop-free before disabling spanning tree.

Restrictions for VLAN Trunks

The following are restrictions for VLAN trunks:

- A trunk port cannot be a secure port.

- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the device propagates the setting that you entered to all ports in the group:

  - Allowed-VLAN list.
  - STP port priority for each VLAN.
  - STP Port Fast setting.
  - Trunk status:

    If one port in a port group ceases to be a trunk, all ports cease to be trunks.

- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.

- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

- Dynamic Trunking Protocol (DTP) is not supported on tunnel ports.

- When native VLAN and management VLAN is configured with the same VLAN ID and a new VLAN is added as trunk port, both the new VLAN and native VLAN shifts between active and suspend state for a duration of 15 seconds. This duration is the time taken for STP to resolve all inconsistencies.

How to Configure VLAN Trunks

To avoid trunking misconfigurations, configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.
- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.

Configuring an Ethernet Interface as a Trunk Port

This section provides information about configuring an Ethernet Interface as a trunk port:

Configuring a Trunk Port

Because trunk ports send and receive VTP advertisements, to use VTP you must ensure that at least one trunk port is configured on the device and that this trunk port is connected to the trunk port of a second device. Otherwise, the device cannot receive any VTP advertisements.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/2`	Specifies the port to be configured for trunking, and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**} **Example:** `Device(config-if)# switchport mode dynamic desirable`	Configures the interface as a Layer 2 trunk (required only if the interface is a Layer 2 access port or tunnel port or to specify the trunking mode). • **dynamic auto**—Sets the interface to a trunk link if the neighboring interface is set to trunk or desirable mode. This is the default. • **dynamic desirable**—Sets the interface to a trunk link if the neighboring interface is set to trunk, desirable, or auto mode. • **trunk**—Sets the interface in permanent trunking mode and negotiate to convert the link to a trunk link even if the neighboring interface is not a trunk interface.
Step 5	**switchport access vlan** *vlan-id* **Example:** `Device(config-if)# switchport access vlan 200`	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
Step 6	**switchport trunk native vlan** *vlan-id* **Example:** `Device(config-if)# switchport trunk native vlan 200`	Specifies the native VLAN for IEEE 802.1Q trunks.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **switchport** **Example:** `Device# show interfaces gigabitethernet 1/0/2 switchport`	Displays the switch port configuration of the interface in the *Administrative Mode* and the *Administrative Trunking Encapsulation* fields of the display.
Step 9	**show interfaces** *interface-id* **trunk** **Example:**	Displays the trunk configuration of the interface.

	Command or Action	Purpose
	`Device# show interfaces gigabitethernet 1/0/2 trunk`	
Step 10	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Defining the Allowed VLANs on a Trunk

VLAN 1 is the default VLAN on all trunk ports in all Cisco devices, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. You can use the VLAN 1 minimization feature to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Specifies the port to be configured, and enters interface configuration mode.
Step 4	**switchport mode trunk** **Example:** `Device(config-if)# switchport mode trunk`	Configures the interface as a VLAN trunk port.

	Command or Action	Purpose
Step 5	**switchport trunk allowed vlan** { *word* | **add** | **all** | **except** | **none** | **remove**} *vlan-list* **Example:** `Device(config-if)# switchport trunk allowed vlan remove 2`	(Optional) Configures the list of VLANs allowed on the trunk. The *vlan-list* parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport** **Example:** `Device# show interfaces gigabitethernet 1/0/1 switchport`	Verifies your entries in the *Trunking VLANs Enabled* field of the display.
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing the Pruning-Eligible List

The pruning-eligible list applies only to trunk ports. Each trunk port has its own eligibility list. VTP pruning must be enabled for this procedure to take effect.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet0/1`	Selects the trunk port for which VLANs should be pruned, and enters interface configuration mode.
Step 4	**switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]	Configures the list of VLANs allowed to be pruned from the trunk. For explanations about using the **add**, **except**, **none**, and **remove** keywords, see the command reference for this release. Separate non-consecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs. Valid IDs are 2 to 1001. Extended-range VLANs (VLAN IDs 1006 to 4094) cannot be pruned. VLANs that are pruning-ineligible receive flooded traffic. The default list of VLANs allowed to be pruned contains VLANs 2 to 1001.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Device# show interfaces gigabitethernet 1/0/1 switchport`	Verifies your entries in the *Pruning VLANs Enabled* field of the display.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the device forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

The native VLAN can be assigned any VLAN ID.

If a packet has a VLAN ID that is the same as the outgoing port native VLAN ID, the packet is sent untagged; otherwise, the device sends the packet with a tag.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/2`	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.
Step 4	**switchport trunk native vlan** *vlan-id* **Example:** `Device(config-if)# switchport trunk native vlan 12`	Configures the VLAN that is sending and receiving untagged traffic on the trunk port. For *vlan-id*, the range is 1 to 4094.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Device# show interfaces gigabitethernet`	Verifies your entries in the *Trunking Native Mode VLAN* field.

	Command or Action	Purpose
	`1/0/2 switchport`	
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Trunk Ports for Load Sharing

The following sections provide information about configuring trunk ports for load sharing:

Configuring Load Sharing Using STP Port Priorities

If your switch is a member of a switch stack, you must use the **spanning-tree** [**vlan** *vlan-id*] **cost** *cost* interface configuration command instead of the **spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority* interface configuration command to select an interface to put in the forwarding state. Assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last.

These steps describe how to configure a network with load sharing using STP port priorities.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode on Device A.
Step 3	**vtp domain** *domain-name* **Example:** `Device(config)# vtp domain workdomain`	Configures a VTP administrative domain. The domain name can be 1 to 32 characters.
Step 4	**vtp mode server** **Example:**	Configures Device A as the VTP server.

	Command or Action	Purpose
	`Device(config)# vtp mode server`	
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show vtp status** **Example:** `Device# show vtp status`	Verifies the VTP configuration on both Device A and Device B. In the display, check the *VTP Operating Mode* and the *VTP Domain Name* fields.
Step 7	**show vlan** **Example:** `Device# show vlan`	Verifies that the VLANs exist in the database on Device A.
Step 8	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 9	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/1`	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 10	**switchport mode trunk** **Example:** `Device(config-if)# switchport mode trunk`	Configures the port as a trunk port.
Step 11	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 12	**show interfaces** *interface-id* **switchport** **Example:**	Verifies the VLAN configuration.

	Command or Action	Purpose
	`Device# show interfaces gigabitethernet 1/0/1 switchport`	
Step 13	Repeat the above steps on Device A for a second port in the device.	
Step 14	Repeat the above steps on Device B to configure the trunk ports that connect to the trunk ports configured on Device A.	
Step 15	**show vlan** **Example:** `Device# show vlan`	When the trunk links come up, VTP passes the VTP and VLAN information to Device B. This command verifies that Device B has learned the VLAN configuration.
Step 16	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode on Device A.
Step 17	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 18	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** `Device(config-if)# spanning-tree vlan 8-10 port-priority 16`	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 19	**exit** **Example:** `Device(config-if)# exit`	Returns to global configuration mode.
Step 20	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/2`	Defines the interface to set the STP port priority, and enters interface configuration mode.

	Command or Action	Purpose
Step 21	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** `Device(config-if)# spanning-tree vlan 3-6 port-priority 16`	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 22	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 23	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 24	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Load Sharing Using STP Path Cost

These steps describe how to configure a network with load sharing using STP path costs.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode on Device A.

	Command or Action	Purpose
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 4	**switchport mode trunk** **Example:** `Device(config-if)# switchport mode trunk`	Configures the port as a trunk port.
Step 5	**exit** **Example:** `Device(config-if)# exit`	Returns to global configuration mode.
Step 6	Repeat Steps 2 through 4 on a second interface in Device A or in Device A stack.	
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** `Device# show running-config`	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.
Step 9	**show vlan** **Example:** `Device# show vlan`	When the trunk links come up, Device A receives the VTP information from the other devices. This command verifies that Device A has learned the VLAN configuration.
Step 10	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 11	**interface** *interface-id* **Example:**	Defines the interface on which to set the STP cost, and enters interface configuration mode.

	Command or Action	Purpose
	`Device(config)# interface gigabitethernet 1/0/1`	
Step 12	**spanning-tree vlan** *vlan-range* **cost** *cost-value* **Example:** `Device(config-if)# spanning-tree vlan 2-4 cost 30`	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 13	**end** **Example:** `Device(config-if)# end`	Returns to global configuration mode.
Step 14	Repeat Steps 9 through 13 on the other configured trunk interface on Device A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
Step 15	**exit** **Example:** `Device(config)# exit`	Returns to privileged EXEC mode.
Step 16	**show running-config** **Example:** `Device# show running-config`	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 17	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

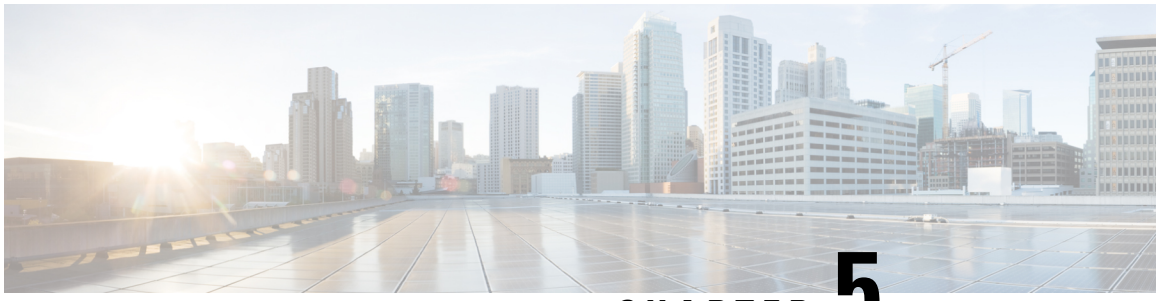
Feature History for VLAN Tunks

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	VLAN Trunks	A trunk is a point-to-point link between one or more Ethernet interfaces and another networking device such as a router or a controller. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network.
Cisco IOS XE Cupertino 17.7.1	VLAN Trunks	This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL.
Cisco IOS XE Dublin 17.10.1b	VLAN Trunks	This feature was implemented on the C9500X-60L4D model.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.



CHAPTER 5

Configuring Private VLANs

- Prerequisites for Private VLANs, on page 61
- Restrictions for Private VLANs, on page 61
- Information About Private VLANs, on page 62
- How to Configure Private VLANs, on page 73
- Monitoring Private VLANs, on page 89
- Configuration Examples for Private VLANs, on page 89
- Where to Go Next, on page 94
- Additional References, on page 94
- Feature History for Private VLANs, on page 95

Prerequisites for Private VLANs

When configuring private VLANs on the device, always use the default Switch Database Management (SDM) template to balance system resources between unicast routes and Layer 2 entries. If another SDM template is configured, use the **sdm prefer default** global configuration command to set the default template.

Note Private VLANs are supported in transparent mode for VTP 1, 2 and 3. Private VLANS are also supported on server mode with VTP 3.

Restrictions for Private VLANs

Note In some cases, the configuration is accepted with no error messages, but the commands have no effect.

- Do not configure fallback bridging on the device with private VLANs.
- Do not configure a remote SPAN (RSPAN) VLAN as a primary or a secondary VLAN of a private-VLAN.
- Do not configure private VLAN ports on interfaces configured for these other features:
  - Dynamic-access port VLAN membership

- Dynamic Trunking Protocol (DTP)
- IP Source Guard
- IPv6 First Hop Security (FHS)
- IPv6 Security Group (SG)
- Multicast VLAN Registration (MVR)
- Voice VLAN
- Web Cache Communication Protocol (WCCP)

- Port Aggregation Protocol (PAgP) and Link Aggregation Control Protocol (LACP) are supported only for Private VLAN promiscuous trunk ports and Private VLAN isolated trunk ports.
- You can configure IEEE 802.1x port-based authentication on a private-VLAN port, but do not configure 802.1x with port security, voice VLAN, or per-user ACL on private-VLAN ports.
- A private-VLAN host or promiscuous port cannot be a SPAN destination port. If you configure a SPAN destination port as a private-VLAN port, the port becomes inactive.
- If you configure a static MAC address on a promiscuous port in the primary VLAN, you need not add the same static address to all associated secondary VLANs. Similarly, if you configure a static MAC address on a host port in a secondary VLAN, you need not add the same static MAC address to the associated primary VLAN. Also, when you delete a static MAC address from a private-VLAN port, you do not have to remove all instances of the configured MAC address from the private VLAN.

Note Dynamic MAC addresses learned in the secondary VLAN of a private VLAN are replicated to the primary VLANs. All MAC entries are learnt on secondary VLANs, even if the traffic ingresses from primary VLAN. If a MAC address is dynamically learnt in the primary VLAN, it is not replicated in the associated secondary VLANs.

- Configure Layer 3 VLAN interfaces (switch value interfaces) only for primary VLANs.
- Private VLAN configured with MACsec or Virtual Private LAN Services (VPLS) or Cisco Software-Defined Access solution on the same VLAN does not work.

Information About Private VLANs

The following sections provide information about Private VLANs:

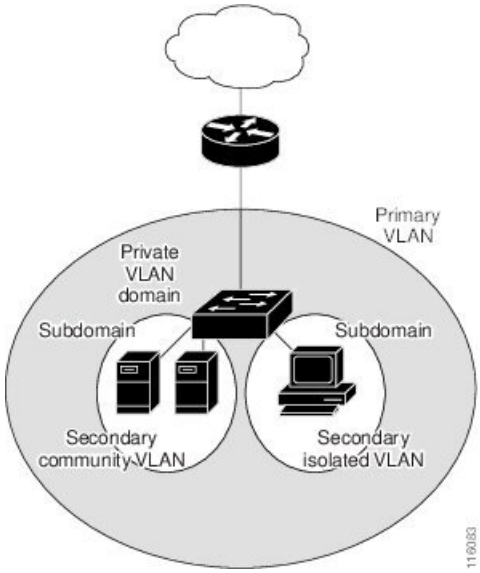
Private VLAN Domains

The private VLAN feature addresses two problems that service providers face when using VLANs:

- To enable IP routing, each VLAN is assigned a subnet address space or a block of addresses, which can result in wasting the unused IP addresses, and cause IP address management problems.

Figure 1: Private VLAN Domain

Using private VLANs addresses the scalability problem and provides IP address management benefits for service providers and Layer 2 security for customers. Private VLANs partition a regular VLAN domain into subdomains. A subdomain is represented by a pair of VLANs: a *primary* VLAN and a *secondary* VLAN. A private VLAN can have multiple VLAN pairs, one pair for each subdomain. All VLAN pairs in a private VLAN share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another.



Secondary VLANs

There are two types of secondary VLANs:

- Isolated VLANs—Ports within an isolated VLAN cannot communicate with each other at the Layer 2 level.
- Community VLANs—Ports within a community VLAN can communicate with each other but cannot communicate with ports in other communities at the Layer 2 level.

Private VLANs Ports

Private VLANs provide Layer 2 isolation between ports within the same private VLAN. Private VLAN ports are access ports that are one of these types:

- Promiscuous—A promiscuous port belongs to the primary VLAN and can communicate with all interfaces, including the community and isolated host ports that belong to the secondary VLANs associated with the primary VLAN.
- Isolated—An isolated port is a host port that belongs to an isolated secondary VLAN. It has complete Layer 2 separation from other ports within the same private VLAN, except for the promiscuous ports. Private VLANs block all traffic to isolated ports except traffic from promiscuous ports. Traffic received from an isolated port is forwarded only to promiscuous ports.
- Community—A community port is a host port that belongs to a community secondary VLAN. Community ports communicate with other ports in the same community VLAN and with promiscuous ports. These

interfaces are isolated at Layer 2 from all other interfaces in other communities and from isolated ports within their private VLAN.

Note Trunk ports carry traffic from regular VLANs and also from primary, isolated, and community VLANs.

Primary and secondary VLANs have these characteristics:

- Primary VLAN—A private VLAN has only one primary VLAN. Every port in a private VLAN is a member of the primary VLAN. The primary VLAN carries unidirectional traffic downstream from the promiscuous ports to the (isolated and community) host ports and to other promiscuous ports.
- Isolated VLAN —A private VLAN has only one isolated VLAN. An isolated VLAN is a secondary VLAN that carries unidirectional traffic upstream from the hosts toward the promiscuous ports and the gateway.
- Community VLAN—A community VLAN is a secondary VLAN that carries upstream traffic from the community ports to the promiscuous port gateways and to other host ports in the same community. You can configure multiple community VLANs in a private VLAN.

A promiscuous port can serve only one primary VLAN, one isolated VLAN, and multiple community VLANs. Layer 3 gateways are typically connected to the device through a promiscuous port. With a promiscuous port, you can connect a wide range of devices as access points to a private VLAN. For example, you can use a promiscuous port to monitor or back up all the private VLAN servers from an administration workstation.

Private VLANs in Networks

In a switched environment, you can assign an individual private VLAN and associated IP subnet to each individual or common group of end stations. The end stations need to communicate only with a default gateway to communicate outside the private VLAN.

You can use private VLANs to control access to end stations in these ways:

- Configure selected interfaces connected to end stations as isolated ports to prevent any communication at Layer 2. For example, if the end stations are servers, this configuration prevents Layer 2 communication between the servers.
- Configure interfaces connected to default gateways and selected end stations (for example, backup servers) as promiscuous ports to allow all end stations access to a default gateway.

You can extend private VLANs across multiple devices by trunking the primary, isolated, and community VLANs to other devices that support private VLANs. To maintain the security of your private VLAN configuration and to avoid other use of the VLANs configured as private VLANs, configure private VLANs on all intermediate devices, including devices that have no private VLAN ports.

IP Addressing Scheme with Private VLANs

Assigning a separate VLAN to each customer creates an inefficient IP addressing scheme:

- Assigning a block of addresses to a customer VLAN can result in unused IP addresses.

- If the number of devices in the VLAN increases, the number of assigned address might not be large enough to accommodate them.

These problems are reduced by using private VLANs, where all members in the private VLAN share a common address space, which is allocated to the primary VLAN. Hosts are connected to secondary VLANs and the DHCP server assigns them IP addresses from the block of addresses allocated to the primary VLAN. Subsequent IP addresses can be assigned to customer devices in different secondary VLANs, but in the same primary VLAN. When new devices are added, the DHCP server assigns them the next available address from a large pool of subnet addresses.

Private VLANs Across Multiple Switches

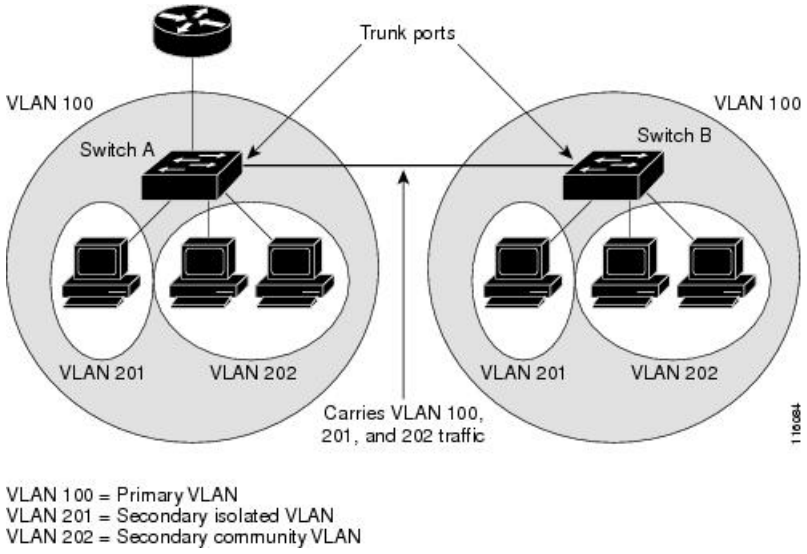
This section discusses the following topics:

- Standard Trunk Ports
- Isolated Private VLAN Trunk Ports
- Promiscuous Private VLAN Trunk Ports

Standard Trunk Ports

As with regular VLANs, private VLANs can span multiple switches. A trunk port carries the primary VLAN and secondary VLANs to a neighboring switch. The trunk port treats the private VLAN as any other VLAN. A feature of private VLANs across multiple switches is that traffic from an isolated port in the Switch A does not reach an isolated port on Switch B.

Figure 2: Private VLANs Across Switches

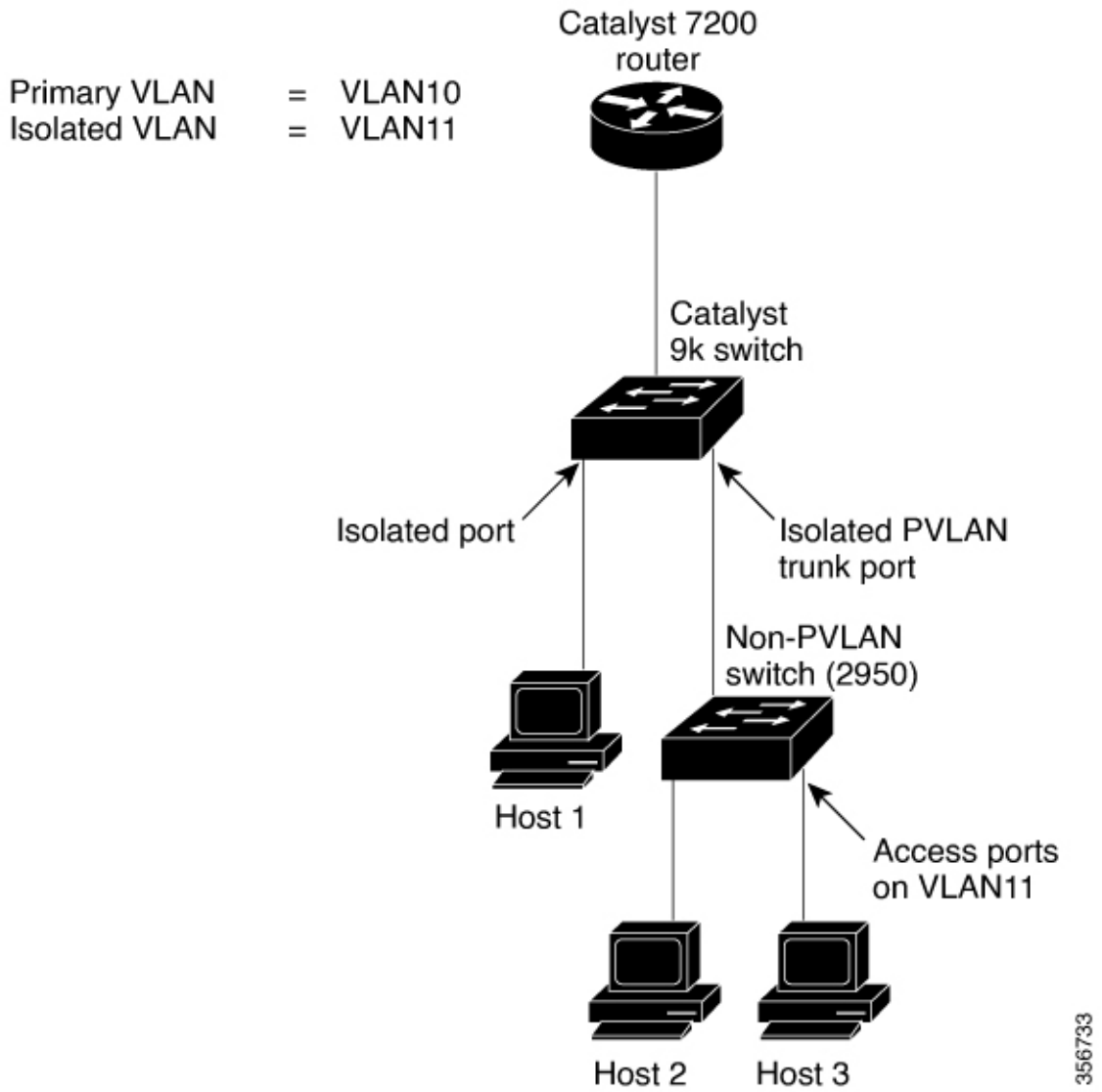


Private VLANs are supported in transparent mode for VTP 1, 2 and 3. Private VLAN is also supported on server mode for VTP 3. If you have a server client setup using VTP 3, private VLANs configured on the server should be reflected on the client.

Isolated Private VLAN Trunk Ports

You use isolated PVLAN trunk ports if you anticipate using PVLAN isolated host ports to carry multiple VLANs, either normal VLANs or multiple PVLAN domains. You can connect a downstream switch that does not support PVLANs, such as a Catalyst 2950.

Figure 3: Isolated PVLAN Trunk Ports



In this illustration, a Catalyst 9k switch connects to a downstream switch that does not support PVLANs.

Traffic being sent in the downstream direction towards host1 from the router is received by the

Catalyst 9k series switch on the promiscuous port and in the primary VLAN (VLAN 10). The packets are then switched out of the isolated PVLAN trunk; instead of being tagged with the primary VLAN (VLAN 10) they are transmitted with the isolated VLAN's tag (VLAN 11). When the packets arrive on the non-PVLAN switch, they can be bridged to the destination hosts' access port.

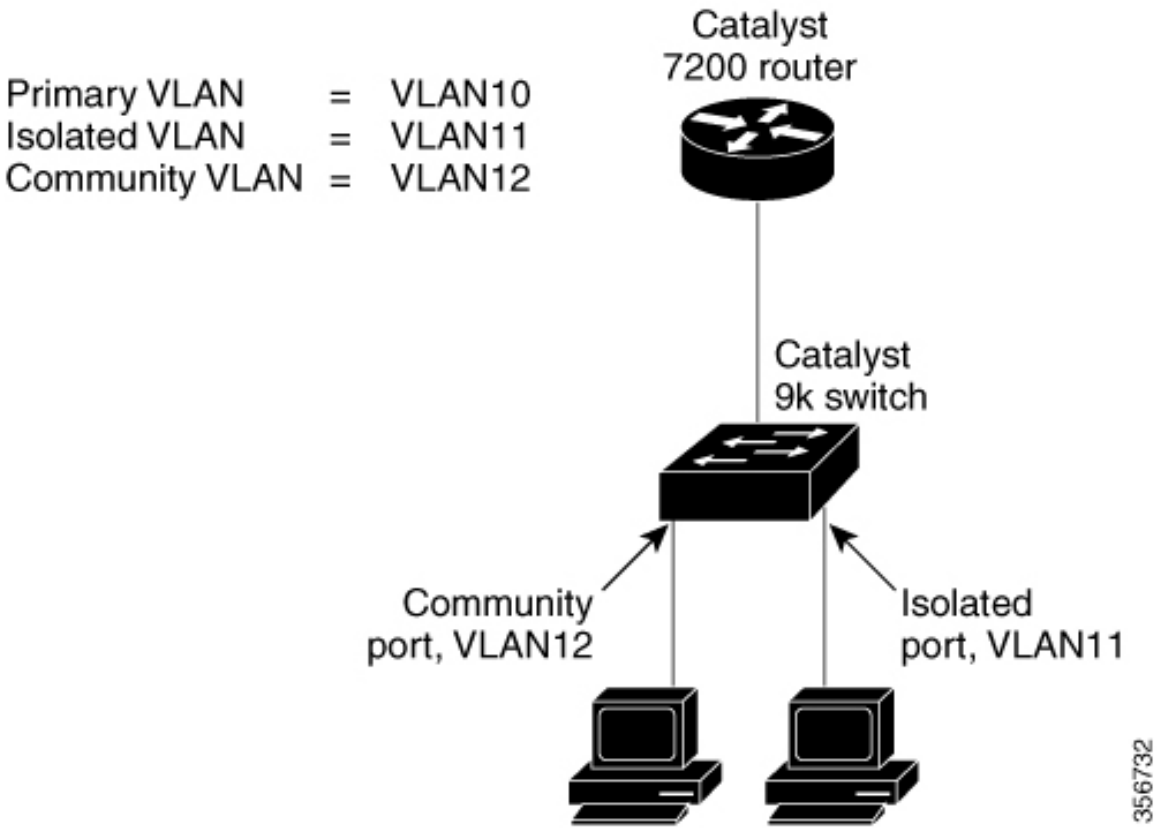
Traffic in the upstream direction is sent by host1 to the non-PVLAN switch, arriving in VLAN 11. The packets are then transmitted to the Catalyst 9k series switch tagged with that VLAN's tag (VLAN 11) over the trunk port. On the Catalyst 9k series switch, VLAN 11 is configured as the isolated VLAN, and the traffic is forwarded as if it came from an isolated host port.

Note The Catalyst 9k series switch provides isolation between the isolated trunk and directly connected hosts (such as host3), but not between hosts connected to the non-PVLAN switch (such as host1 and host2). Isolation between these hosts must be provided by the non-PVLAN switch, using a feature such as protected ports on a Catalyst 2950.

Promiscuous Private VLAN Trunk Ports

Promiscuous private VLAN trunk ports are used in situations where a PVLAN promiscuous host port is normally used, but where it is necessary to carry multiple VLANs, either normal VLANs or multiple PVLAN domains. You can connect to an upstream router that does not support PVLANs, such as a Cisco 7200 router.

Figure 4: Promiscuous PVLAN Trunk Ports



In this illustration, a Catalyst 9k series switch connects a PVLAN domain to an upstream router, which does not support PVLANs. Traffic being sent upstream by host1 arrives on the

Catalyst 9k series switch in the community VLAN (VLAN 12). When this traffic is bridged onto the promiscuous PVLAN trunk towards the router, it is tagged with the primary VLAN (VLAN 10), so that it can be routed by using the correct subinterface configured on the router.

Traffic in the downstream direction is received on the promiscuous PVLAN trunk port by the

Catalyst 9k series switch in the primary VLAN (VLAN 10), as if it had been received on a promiscuous host port. The traffic can then be bridged to the destination host as in any PVLAN domain.

PVLAN promiscuous trunks interact with VLAN QoS.

Private-VLAN Interaction with Other Features

The following sections provide information about Private-VLAN interaction with other features:

Private VLANs and Unicast, Broadcast, and Multicast Traffic

In regular VLANs, devices in the same VLAN can communicate with each other at the Layer 2 level, but devices connected to interfaces in different VLANs must communicate at the Layer 3 level. In private VLANs, the promiscuous ports are members of the primary VLAN, while the host ports belong to secondary VLANs. Because the secondary VLAN is associated with the primary VLAN, members of the these VLANs can communicate with each other at the Layer 2 level.

In a regular VLAN, broadcasts are forwarded to all ports in that VLAN. Private VLAN broadcast forwarding depends on the port sending the broadcast:

- An isolated port sends a broadcast only to the promiscuous ports or trunk ports.
- A community port sends a broadcast to all promiscuous ports, trunk ports, and ports in the same community VLAN.
- A promiscuous port sends a broadcast to all ports in the private VLAN (other promiscuous ports, trunk ports, isolated ports, and community ports).

Multicast traffic is routed or bridged across private VLAN boundaries and within a single community VLAN. Multicast traffic is not forwarded between ports in the same isolated VLAN or between ports in different secondary VLANs.

Private VLAN multicast forwarding supports the following:

- Sender can be outside the VLAN and the Receivers can be inside the VLAN domain.
- Sender can be inside the VLAN and the Receivers can be outside the VLAN domain.
- Sender and Receiver can both be in the same community VLAN.

Private VLANs and SVIs

A switch virtual interface (SVI) represents the Layer 3 interface of a VLAN. Layer 3 devices communicate with a private VLAN only through the primary VLAN and not through secondary VLANs. Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs. You cannot configure Layer 3 VLAN interfaces for secondary VLANs. SVIs for secondary VLANs are inactive while the VLAN is configured as a secondary VLAN.

- If you try to configure a VLAN with an active SVI as a secondary VLAN, the configuration is not allowed until you disable the SVI.
- If you try to create an SVI on a VLAN that is configured as a secondary VLAN and the secondary VLAN is already mapped at Layer 3, the SVI is not created, and an error is returned. If the SVI is not mapped at Layer 3, the SVI is created, but it is automatically shut down.

When the primary VLAN is associated with and mapped to the secondary VLAN, any configuration on the primary VLAN is propagated to the secondary VLAN SVIs. For example, if you assign an IP subnet to the primary VLAN SVI, this subnet is the IP subnet address of the entire private VLAN.

Private VLANs and Switch Stacks

Private VLANs can operate within the switch stack, and private-VLAN ports can reside on different member switches in the stack. However, the following changes to the stack can impact private-VLAN operation:

- If a stack contains only one private-VLAN promiscuous port and the member switch that contains that port is removed from the stack, host ports in that private VLAN lose connectivity outside the private VLAN.
- If an active switch that contains the only private-VLAN promiscuous port in the stack fails or leaves the stack and a new active switch is elected, host ports in a private VLAN that had its promiscuous port on the old active switch lose connectivity outside of the private VLAN.
- If two stacks merge, private VLANs on the winning stack are not affected, but private-VLAN configuration on the losing switch is lost when that switch reboots.

Private VLAN with Dynamic MAC Address

The MAC addresses learnt in the secondary VLAN are replicated to the primary VLAN and not vice-versa. This saves the hardware l2 cam space. The primary VLAN is always used for forwarding lookups in both directions.

Dynamic MAC addresses learned in Primary VLAN of a private VLAN are then, if required, replicated in the secondary VLANs. For example, if a MAC-address is dynamically received on the secondary VLAN, it will be learnt as part of primary VLAN. In case of isolated VLANs, a blocked entry for the same mac will be added to secondary VLAN in the mac address table. So, MAC learnt on host ports in secondary domain are installed as blocked type entries. All mac entries are learnt on secondary VLANs, even if the traffic ingresses from primary VLAN.

However, if a MAC-address is dynamically learnt in the primary VLAN it will not get replicated in the associated secondary VLANS.

Private VLAN with Static MAC Address

Users are not required to replicate the Static MAC Address CLI for private VLAN hosts as compare to legacy model.

Example:

- In the legacy model, if the user configures a static MAC address, they need to add the same static MAC address in the associated VLAN too. For example, if MAC address A is user configured on port 1/0/1 in VLAN 101, where VLAN 101 is a secondary VLAN and VLAN 100 is a primary VLAN, then the user has to configure

```
mac-address static A vlan 101 interface G1/0/1
mac-address static A vlan 100 interface G1/0/1
```

- In this device, the user does not need to replicate the mac address to the associated VLAN. For the above example, user has to configure only

```
mac-address static A vlan 101 interface G1/0/1
```

Private VLAN Interaction with VACL/QOS

Private VLANs are bidirectional in case of this device, as compared to “Unidirectional” in other platforms.

After layer-2 forward lookup, proper egress VLAN mapping happens and all the egress VLAN based feature processing happens in the egress VLAN context.

When a frame in Layer-2 is forwarded within a private VLAN, the VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side. Similarly, when the frame is routed from an external port to a Private VLAN, the private-VLAN is applied at the egress side. This is applicable to both bridged and routed traffic.

Bridging:

- For upstream traffic from secondary VLAN to primary VLAN, the MAP of the secondary VLAN is applied on the ingress side and the MAP of the primary VLAN is applied on the egress side.
- For downstream traffic from primary VLAN to secondary VLAN, the MAP of the primary VLAN is applied in the ingress direction and the MAP of the secondary VLAN is applied in the egress direction.

Routing

If we have two private VLAN domains - PV1 (sec1, prim1) and PV2 (sec2, prim2). For frames routed from PV1 to PV2:

- The MAP of sec1 and L3 ACL of prim1 is applied in the ingress port.
- The MAP of sec2 and L3 ACL of prim2 is applied in the egress port.

For packets going upstream or downstream from isolated host port to promiscuous port, the isolated VLAN’s VACL is applied in the ingress direction and primary VLAN’s VACL is applied in the egress direction. This allows user to configure different VACL for different secondary VLAN in a same primary VLAN domain.

Note 2-way community VLAN is now not required as the private VLANs on this device are always bi-directional.

Private VLANs and HA Support

PVLAN will work seamlessly with High Availability (HA) feature. The Private VLAN existing on the active switch before changeover should be the same after changeover (new active switch should have similar PVLAN configuration both on IOS side and FED side as that of the old active switch).

Private-VLAN Configuration Guidelines

The following sections provide information about Private-VLAN configuration guidelines:

Default Private-VLAN Configurations

No private VLANs are configured.

Secondary and Primary VLAN Configuration

Follow these guidelines when configuring private VLANs:

- Private VLANs are supported in transparent mode for VTP 1, 2 and 3. If the device is running VTP version 1 or 2, you must set VTP to transparent mode. After you configure a private VLAN, you should not change the VTP mode to client or server. VTP version 3 supports private VLANs in all modes.
- With VTP version 1 or 2, after you have configured private VLANs, use the **copy running-config startup config** privileged EXEC command to save the VTP transparent mode configuration and private-VLAN configuration in the device startup configuration file. Otherwise, if the device resets, it defaults to VTP server mode, which does not support private VLANs. VTP version 3 does support private VLANs.
- VTP version 1 and 2 do not propagate private-VLAN configuration. You must configure private VLANs on each device where you want private-VLAN ports unless the devices are running VTP version 3, as VTP3 propagate private vlans.
- You cannot configure VLAN 1 or VLANs 1002 to 1005 as primary or secondary VLANs. Extended VLANs (VLAN IDs 1006 to 4094) can belong to private VLANs.
- A primary VLAN can have one isolated VLAN and multiple community VLANs associated with it. An isolated or community VLAN can have only one primary VLAN associated with it.
- Although a private VLAN contains more than one VLAN, only one Spanning Tree Protocol (STP) instance runs for the entire private VLAN. When a secondary VLAN is associated with the primary VLAN, the STP parameters of the primary VLAN are propagated to the secondary VLAN.
- When copying a PVLAN configuration from a tftp server and applying it on a running-config, the PVLAN association will not be formed. You will need to check and ensure that the primary VLAN is associated to all the secondary VLANs.

  You can also use **configure replace flash:config_file force** instead of **copy flash:config_file running-config**.
- You can enable DHCP snooping on private VLANs. When you enable DHCP snooping on the primary VLAN, it is propagated to the secondary VLANs. If you configure DHCP on a secondary VLAN, the configuration does not take effect if the primary VLAN is already configured.
- When you enable IP source guard on private-VLAN ports, you must enable DHCP snooping on the primary VLAN.
- We recommend that you prune the private VLANs from the trunks on devices that carry no traffic in the private VLANs.
- You can apply different quality of service (QoS) configurations to primary, isolated, and community VLANs.
- Note the following considerations for sticky ARP:
  - Sticky ARP entries are those learned on SVIs and Layer 3 interfaces. These entries do not age out.
  - The **ip sticky-arp** global configuration command is supported only on SVIs belonging to private VLANs.
  - The **ip sticky-arp** interface configuration command is only supported on:
    - Layer 3 interfaces
    - SVIs belonging to normal VLANs
    - SVIs belonging to private VLANs

For more information about using the **ip sticky-arp** *global* configuration and the **ip sticky-arp interface** configuration commands, see the command reference for this release.

- You can configure VLAN maps on primary and secondary VLANs. However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.
- PVLANs are bidirectional. They can be applied at both the ingress and egress sides.

  When a frame inLayer-2 is forwarded within a private VLAN, the VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side. Similarly, when the frame is routed from an external port to a Private VLAN, the private-VLAN is applied at the egress side.

  Bridging

  - For upstream traffic from secondary VLAN to primary VLAN, the MAP of the secondary VLAN is applied on the ingress side and the MAP of the primary VLAN is applied on the egress side.
  - For downstream traffic from primary VLAN to secondary VLAN, the MAP of the primary VLAN is applied in the ingress direction and the MAP of the secondary VLAN is applied in the egress direction.

  Routing

  If we have two private VLAN domains - PV1 (sec1, prim1) and PV2 (sec2, prim2). For frames routed from PV1 to PV2:

  - The MAP of sec1 and L3 ACL of prim1 is applied in the ingress port .
  - The MAP of sec1 and L3 ACL of prim2 is applied in the egress port.
  - For packets going upstream or downstream from isolated host port to promiscuous port, the isolated VLAN's VACL is applied in the ingress direction and primary VLAN'S VACL is applied in the egress direction. This allows user to configure different VACL for different secondary VLAN in a same primary VLAN domain.

  To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs.
- You can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.
- Although private VLANs provide host isolation at Layer 2, hosts can communicate with each other at Layer 3.
- Private VLANs support these Switched Port Analyzer (SPAN) features:
  - You can configure a private-VLAN port as a SPAN source port.
  - You can use VLAN-based SPAN (VSPAN) on primary, isolated, and community VLANs or use SPAN on only one VLAN to separately monitor egress or ingress traffic.

Private VLAN Port Configuration

Follow these guidelines when configuring private VLAN ports:

- Use only the private VLAN configuration commands to assign ports to primary, isolated, or community VLANs. Layer 2 access ports assigned to the VLANs that you configure as primary, isolated, or community VLANs are inactive while the VLAN is part of the private VLAN configuration. Layer 2 trunk interfaces remain in the STP forwarding state.
- Do not configure ports that belong to a PAgP or LACP EtherChannel as private VLAN ports. While a port is part of the private VLAN configuration, any EtherChannel configuration for it is inactive.
- Enable Port Fast and BPDU guard on isolated and community host ports to prevent STP loops due to misconfigurations and to speed up STP convergence. When enabled, STP applies the BPDU guard feature to all Port Fast-configured Layer 2 LAN ports. Do not enable Port Fast and BPDU guard on promiscuous ports.
- If you delete a VLAN used in the private VLAN configuration, the private VLAN ports associated with the VLAN become inactive.
- Private VLAN ports can be on different network devices if the devices are trunk-connected and the primary and secondary VLANs have not been removed from the trunk.

How to Configure Private VLANs

The following sections provide information about configuring Private VLANs:

Configuring Private VLANs

To configure a private VLAN, perform these steps:

Note Private vlans are supported in transparent mode for VTP 1, 2 and 3. Private VLANS are also supported on server mode with VTP 3.

Procedure

Step 1 Set VTP mode to **transparent**

Note
Note: For VTP3, you can set mode to either server or transparent mode.

Step 2 Create the primary and secondary VLANs and associate them.

See Configuring and Associating VLANs in a Private VLAN

Note
If the VLAN is not created already, the private-VLAN configuration process creates it.

Step 3 Configure interfaces to be isolated or community host ports, and assign VLAN membership to the host port.

See Configuring a Layer 2 Interface as a Private VLAN Host Port

Step 4 Configure interfaces as promiscuous ports, and map the promiscuous ports to the primary-secondary VLAN pair.

See Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port

Step 5 If inter-VLAN routing will be used, configure the primary SVI, and map secondary VLANs to the primary. See Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface

Step 6 Verify private-VLAN configuration.

Configuring and Associating VLANs in a Private VLAN

The **private-vlan** commands do not take effect until you exit VLAN configuration mode.

To configure and associate VLANs in a Private VLAN, perform these steps:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**vtp mode transparent** **Example:** `Device(config)# vtp mode transparent`	Sets VTP mode to transparent (disable VTP). **Note** For VTP3, you can set mode to either server or transparent mode
Step 4	**vlan** *vlan-id* **Example:** `Device(config)# vlan 20`	Enters VLAN configuration mode and designates or creates a VLAN that will be the primary VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 5	**private-vlan primary** **Example:** `Device(config-vlan)# private-vlan primary`	Designates the VLAN as the primary VLAN.
Step 6	**exit** **Example:**	Returns to global configuration mode.

	Command or Action	Purpose
	`Device(config-vlan)# exit`	
Step 7	**vlan** *vlan-id* **Example:** `Device(config)# vlan 501`	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be an isolated VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 8	**private-vlan isolated** **Example:** `Device(config-vlan)# private-vlan isolated`	Designates the VLAN as an isolated VLAN.
Step 9	**exit** **Example:** `Device(config-vlan)# exit`	Returns to global configuration mode.
Step 10	**vlan** *vlan-id* **Example:** `Device(config)# vlan 502`	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 11	**private-vlan community** **Example:** `Device(config-vlan)# private-vlan community`	Designates the VLAN as a community VLAN.
Step 12	**exit** **Example:** `Device(config-vlan)# exit`	Returns to global configuration mode.
Step 13	**vlan** *vlan-id* **Example:** `Device(config)# vlan 503`	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 14	**private-vlan community** **Example:**	Designates the VLAN as a community VLAN.

	Command or Action	Purpose
	`Device(config-vlan)# private-vlan community`	
Step 15	**exit** **Example:** `Device(config-vlan)# exit`	Returns to global configuration mode.
Step 16	**vlan** *vlan-id* **Example:** `Device(config)# vlan 20`	Enters VLAN configuration mode for the primary VLAN designated in Step 4.
Step 17	**private-vlan association** [**add** | **remove**] *secondary_vlan_list* **Example:** `Device(config-vlan)# private-vlan association 501-503`	Associates the secondary VLANs with the primary VLAN. It can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The *secondary_vlan_list* parameter can contain multiple community VLAN IDs but only one isolated VLAN ID. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to associate secondary VLANs with a primary VLAN. • Use the **remove** keyword with a *secondary_vlan_list* to clear the association between secondary VLANs and a primary VLAN. • The command does not take effect until you exit VLAN configuration mode.
Step 18	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 19	**show vlan private-vlan** [**type**] or **show interfaces status** **Example:** `Device# show vlan private-vlan`	Verifies the configuration.
Step 20	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as a Private VLAN Host Port

Follow these steps to configure a Layer 2 interface as a private-VLAN host port and to associate it with primary and secondary VLANs:

Note Isolated and community VLANs are both secondary VLANs.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/22`	Enters interface configuration mode for the Layer 2 interface to be configured.

	Command or Action	Purpose
Step 4	**switchport mode private-vlan host** **Example:** `Device(config-if)# switchport mode private-vlan host`	Configures the Layer 2 port as a private-VLAN host port.
Step 5	**switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id* **Example:** `Device(config-if)# switchport private-vlan host-association 20 501`	Associates the Layer 2 port with a private VLAN. **Note** This is a required step to associate the PVLAN to a Layer 2 interface.
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **switchport** **Example:** `Device# show interfaces gigabitethernet1/0/22 switchport`	Verifies the configuration.
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port

Follow these steps to configure a Layer 2 interface as a private VLAN promiscuous port and map it to primary and secondary VLANs:

Note Isolated and community VLANs are both secondary VLANs.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	**switchport mode private-vlan promiscuous** **Example:** `Device(config-if)# switchport mode private-vlan promiscuous`	Configures the Layer 2 port as a private VLAN promiscuous port.
Step 5	**switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list* **Example:** `Device(config-if)# switchport private-vlan mapping 20 add 501-503`	Maps the private VLAN promiscuous port to a primary VLAN and to selected secondary VLANs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private VLAN ID or a hyphenated range of private VLAN IDs. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to the private VLAN promiscuous port. • Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and the private VLAN promiscuous port.

	Command or Action	Purpose
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **switchport** **Example:** `Device# show interfaces gigabitethernet1/0/2 switchport`	Verifies the configuration.
Step 8	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port

To configure a Layer 2 interface as an isolated PVLAN trunk port, perform this task:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 3	**switchport mode private-vlan** { **host** | **promiscuous** | **trunk promiscuous** | **trunk** [**secondary**]} **Example:** `Device(config-if)# switchport mode`	Configures the Layer 2 interface as an isolated private VLAN trunk port.

	Command or Action	Purpose
	`private-vlan trunk secondary`	
Step 4	**switchport private-vlan association trunk** *primary_vlan_id secondary_vlan_id* **Example:** `Device(config-if)# switchport private-vlan association trunk 3 301`	Maps the private VLAN trunk port to a primary VLAN and to the selected secondary VLAN.
Step 5	**switchport private-vlan trunk allowed vlan** { *word* | **add** |**all** | **except** |**none** | **remove** } *vlan_list* **Example:** `Device(config-if)# switchport private-vlan trunk allowed vlan 10. 3-4`	Configures a list of allowed VLANs on a PVLAN trunk port. The list must include primary VLANs. In addition, normal VLANs can be configured as well. You can use the no keyword to remove all allowed VLANs on a PVLAN trunk port.
Step 6	**switchport private-vlan trunk native vlan** *vlan_id* **Example:** `Device(config-if)# switchport private-vlan trunk native vlan 10`	Configures a VLAN to which untagged packets (as in IEEE 802.1Q tagging) are assigned on a PVLAN trunk port. You can use the no keyword to remove native VLAN configuration on an isolated PVLAN trunk port. The native VLAN is then set to default VLAN1.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*] **switchport** **Example:** `Device# show interfaces gigabitethernet1/0/2 switchport`	Verifies the configuration.
Step 9	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port

To configure a Layer 2 interface as a promiscuous private VLAN trunk port, perform this task:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 3	**switchport mode private-vlan** { **host** | **promiscuous** | **trunk promiscuous** | **trunk** [**secondary**]} **Example:** `Device(config-if)# switchport mode private-vlan trunk promiscuous`	Configures the Layer 2 port as a promiscuous private VLAN trunk port.
Step 4	**switchport private-vlan mapping trunk** *primary_vlan_id* [**add** | **remove**] *secondary_vlan_list* **Example:** `Device(config-if)# switchport private-vlan mapping trunk 20 add 501-503`	Maps the promiscuous private VLAN trunk port to a primary VLAN and to selected secondary VLANs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private VLAN ID or a hyphenated range of private VLAN IDs. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to the promiscuous private VLAN trunk port. • Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and the promiscuous private VLAN trunk port.

	Command or Action	Purpose
Step 5	**switchport private-vlan trunk allowed vlan** { *word* | **add** |**all** | **except** |**none** | **remove** } *vlan_list* **Example:** `Device(config-if)# switchport private-vlan trunk allowed vlan 10 3-4`	Configures a list of allowed VLANs on a PVLAN trunk port. The list must include primary VLANs. In addition, normal VLANs can be configured as well. You can use the no keyword to remove all allowed VLANs on a PVLAN promiscuous trunk port.
Step 6	**switchport private-vlan trunk native vlan** *vlan_id* **Example:** `Device(config-if)# switchport private-vlan trunk native vlan 10`	Configures a VLAN to which untagged packets (as in IEEE 802.1Q tagging) are assigned on a PVLAN trunk port. You can use the no keyword to remove native VLAN configuration on a PVLAN promiscuous trunk port. The native VLAN is then set to default VLAN1.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*] **switchport** **Example:** `Device# show interfaces gigabitethernet1/0/2 switchport`	Verifies the configuration.
Step 9	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port on a Portchannel

To configure a Layer 2 interface as an isolated private VLAN trunk port on a portchannel, perform this task:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface range* **Example:** `Device(config)# int range g5/0/17, g5/0/22, g6/0/12`	Enters interface configuration mode for the Layer 2 interface range to be configured.
Step 3	**switchport mode private-vlan** { **host** | **promiscuous** | **trunk promiscuous** | **trunk** [**secondary**]} **Example:** `Device(config-if)# switchport mode private-vlan trunk`	Configures the Layer 2 interface range as a private VLAN isolated trunk port.
Step 4	**switchport private-vlan association trunk** *primary_vlan_id secondary_vlan_id* **Example:** `Device(config-if)# switchport private-vlan association trunk 20 503`	Maps the private VLAN trunk port to a primary VLAN and to the selected secondary VLAN.
Step 5	**switchport private-vlan trunk allowed vlan** { *word* | **add** |**all** | **except** |**none** | **remove** } *vlan_list* **Example:** `Device(config-if)# switchport private-vlan trunk allowed vlan 20`	Configures a list of allowed VLANs on a PVLAN trunk port. The list must include primary VLANs. In addition, normal VLANs can be configured as well. You can use the no keyword to remove all allowed VLANs on an isolated private VLAN trunk port.
Step 6	**channel-group** *channel group number* **mode** { **active** | **auto** | **desirable** | **on** | **passive** } **Example:** `Device(config-if)# channel-group 1 mode active`	Configures the port in a channel group and sets the mode. The channel-number range is from 1 to 128. The port channel associated with this channel group is automatically created if the port channel does not already exist.

	Command or Action	Purpose
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	show etherchannel summary **Example:** `Device# show etherchannel summary`	Verifies the configuration.
Step 9	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port on a Portchannel

To configure a Layer 2 interface as a promiscuous private VLAN trunk port on a portchannel, perform this task:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface range* **Example:** `Device(config)# interface g5/0/17, g5/0/22, g6/0/12`	Enters interface configuration mode for the Layer 2 interface range to be configured.
Step 3	**switchport mode private-vlan** { **host** | **promiscuous** | **trunk promiscuous** | **trunk** [**secondary**]} **Example:**	Configures the Layer 2 ports as private VLAN promiscuous trunk port.

	Command or Action	Purpose
	`Device(config-if)# switchport mode private-vlan trunk promiscuous`	
Step 4	**switchport private-vlan mapping trunk** *primary_vlan_id* [**add** | **remove**] *secondary_vlan_list* **Example:** `Device(config-if)# switchport private-vlan mapping trunk 20 501-503`	Maps the private VLAN promiscuous trunk port to a primary VLAN and to selected secondary VLANs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private VLAN ID or a hyphenated range of private VLAN IDs. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to the private VLAN promiscuous port. • Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and the private VLAN promiscuous port.
Step 5	**switchport private-vlan trunk allowed vlan** { *word* | **add** |**all** | **except** |**none** | **remove** } *vlan_list* **Example:** `Device(config-if)# switchport private-vlan trunk allowed vlan 20`	Configures a list of allowed VLANs on a PVLAN trunk port. The list must include primary VLANs. In addition, normal VLANs can be configured as well. You can use the no keyword to remove all allowed VLANs on a promiscuous private VLAN trunk port.
Step 6	**channel-group** *channel group number* **mode** { **active** | **auto** | **desirable** | **on** | **passive** } **Example:** `Device(config-if)# channel-group 1 mode active`	Configures the port in a channel group and sets the mode. The channel-number range is from 1 to 128. The port channel associated with this channel group is automatically created if the port channel does not already exist.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	show etherchannel summary **Example:**	Verifies the configuration.

	Command or Action	Purpose
	`Device# show etherchannel summary`	
Step 9	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface

If the private VLAN will be used for inter-VLAN routing, you configure an SVI for the primary VLAN and map secondary VLANs to the SVI.

Note Isolated and community VLANs are both secondary VLANs.

Follow these steps to map secondary VLANs to the SVI of a primary VLAN to allow Layer 3 switching of private VLAN traffic:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface vlan** *primary_vlan_id* **Example:** `Device(config)# interface vlan 20`	Enters interface configuration mode for the primary VLAN, and configures the VLAN as an SVI. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 4	**private-vlan mapping** [**add** | **remove**] *secondary_vlan_list* **Example:**	Maps the secondary VLANs to the Layer 3 VLAN interface of a primary VLAN to allow Layer 3 switching of private VLAN ingress traffic.

	Command or Action	Purpose
	`Device(config-if)# private-vlan mapping 501-503`	**Note** The **private-vlan mapping** interface configuration command only affects private VLAN traffic that is Layer 3 switched. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to a primary VLAN. • Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and a primary VLAN.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces private-vlan mapping** **Example:** `Device# show interfaces private-vlan mapping`	Verifies the configuration.
Step 7	**copy running-config startup config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the device startup configuration file.

Monitoring Private VLANs

The following table displays the commands used to monitor private VLANs.

Table 6: Private VLAN Monitoring Commands

Command	Purpose
show interfaces status	Displays the status of interfaces, includ
show vlan private-vlan [**type**]	Displays the private VLAN information
show interface switchport	Displays private VLAN configuration c
show interface private-vlan mapping	Displays information about the private

Configuration Examples for Private VLANs

This following sections provide configuration examples for Private VLANSs:

Example: Configuring and Associating VLANs in a Private VLAN

This example shows how to configure VLAN 20 as a primary VLAN, VLAN 501 as an isolated VLAN, and VLANs 502 and 503 as community VLANs, to associate them in a private VLAN, and to verify the configuration:

```
Device# configure terminal
Device(config)# vlan 20
Device(config-vlan)# private-vlan primary
Device(config-vlan)# exit
Device(config)# vlan 501
Device(config-vlan)# private-vlan isolated
Device(config-vlan)# exit
Device(config)# vlan 502
Device(config-vlan)# private-vlan community
Device(config-vlan)# exit
Device(config)# vlan 503
Device(config-vlan)# private-vlan community
Device(config-vlan)# exit
Device(config)# vlan 20
Device(config-vlan)# private-vlan association 501-503
Device(config-vlan)# end
Device# show vlan private-vlan
Primary   Secondary      Type
--------- -------------- -----------------
20        501            isolated
20        502            community
20        503            community
```

Example: Configuring an Interface as a Host Port

This example shows how to configure an interface as a private VLAN host port, associate it with a private VLAN pair, and verify the configuration:

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/22
Device(config-if)# switchport mode private-vlan host
Device(config-if)# switchport private-vlan host-association 20 501
Device(config-if)# end
Device# show interfaces gigabitethernet1/0/22 switchport
Name: Gi1/0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: 20 501
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 501

<output truncated>
```

Example: Configuring an Interface as a Private VLAN Promiscuous Port

This example shows how to configure an interface as a private VLAN promiscuous port and map it to a private VLAN. The interface is a member of primary VLAN 20 and secondary VLANs 501 to 503 are mapped to it.

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# switchport mode private-vlan promiscuous
Device(config-if)# switchport private-vlan mapping 20 add 501-503
Device(config-if)# end
```

Use the **show vlan private-vlan** or the **show interface status** privileged EXEC command to display primary and secondary VLANs and private-VLAN ports on the device.

Example: Mapping Secondary VLANs to a Primary VLAN Interface

This example shows how to map the interfaces fo VLANs 501 and 502 to primary VLAN 10, which permits routing of secondary VLAN ingress traffic from private VLANs 501 and 502:

```
Device# configure terminal
Device(config)# interface vlan 20
```

```
Device(config-if)# private-vlan mapping 501-503
Device(config-if)# end
Device# show interfaces private-vlan mapping
Interface Secondary VLAN Type
--------- -------------- -----------------
vlan20    501            isolated
vlan20    502            community
vlan20    503            community
```

Example: Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port

This example shows how to configure an interface as an isolated private VLAN trunk port and to verify the configuration:

```
Device# configure terminal
Device(config)# interface GigabitEthernet5/0/1
Device(config-if)# switchport private-vlan trunk allowed vlan 20
Device(config-if)# switchport private-vlan association trunk 20 503
Device(config-if)# switchport mode private-vlan trunk
Device(config-if)# end

Device# show interface GigabitEthernet5/0/1 switchport
Name: GigabitEthernet5/0/1
Switchport: Enabled
Administrative Mode: private-vlan trunk secondary
Operational Mode: private-vlan trunk secondary
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: 20
Administrative private-vlan trunk associations:
20 (VLAN0020) 503 (VLAN0503)
Administrative private-vlan trunk mappings: none
Operational private-vlan:
20 (VLAN0020) 503 (VLAN0503)
Operational Normal VLANs: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Vepa Enabled: false
Appliance trust: none
```

Example: Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port

This example shows how to configure an interface as a promiscuous private VLAN trunk port and to verify the configuration:

```
Device# configure terminal
Device(config)# interface GigabitEthernet6/0/4
Device(config-if)# switchport private-vlan trunk native vlan 20
Device(config-if)# switchport private-vlan trunk allowed vlan 20
Device(config-if)# switchport private-vlan mapping trunk 20 501-503
Device(config-if)# switchport mode private-vlan trunk promiscuous
Device(config-if)# end

Device# show interface GigabitEthernet6/0/4 switchport
Name: Gi6/0/4
Switchport: Enabled
Administrative Mode: private-vlan trunk promiscuous
Operational Mode: private-vlan trunk promiscuous
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: 20
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: 20
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings:
20 (VLAN0020) 501 (VLAN0501)
502 (VLAN0502)
503 (VLAN0503)

Operational private-vlan:
20 (VLAN0020) 501 (VLAN0501) 502 (VLAN0502) 503 (VLAN0503)
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Vepa Enabled: false
Appliance trust: none
```

Example: Configuring a Layer 2 Interface as an Isolated Private VLAN Trunk Port on a Portchannel

This example shows how to configure an interface as an isolated private VLAN trunk port on a port channel and to verify the configuration:

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface range g5/0/17, g5/0/22, g6/0/12
Device(config-if-range)# switchport mode private-vlan trunk
Device(config-if-range)# switchport private-vlan trunk allowed vlan 20
Device(config-if-range)# switchport private-vlan association trunk 20 503
Device(config-if-range)# channel-group 1 mode active
Device(config-if-range)# end
```

```
Device# show etherchannel summary
Dec 10 13:51:28.423 PST: %SYS-5-CONFIG_I: Configured from console by consolesumm
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
M - not in use, minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
A - formed by Auto LAG
Number of channel-groups in use: 1
Number of aggregators: 1
Group Port-channel Protocol Ports
------+-------------+-----------+-----------------------------------------------
1 Po1(SU) LACP Gi5/0/17(P) Gi5/0/22(P) Gi6/0/12(P)
```

```
Device# show vlan private-vlan
Primary Secondary Type Ports
------- --------- ----------------- ------------------------------------------
20 501 community
20 502 community
20 503 isolated Po1
```

Example: Configuring a Layer 2 Interface as a Promiscuous Private VLAN Trunk Port on a Portchannel

This example shows how to configure an interface as a promiscuous private VLAN trunk port on a port channel and to verify the configuration:

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface range g5/0/17, g5/0/22, g6/0/12
Device(config-if-range)# switchport mode private-vlan trunk promiscuous
Device(config-if-range)# switchport private-vlan trunk allowed vlan 20
Device(config-if-range)# switchport private-vlan mapping trunk 20 501-503
Device(config-if-range)# channel-group 1 mode active
Device(config-if-range)# end
```

```
Device# show etherchannel summary
Dec 10 13:51:28.423 PST: %SYS-5-CONFIG_I: Configured from console by consolesumm
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
M - not in use, minimum links not met
```

```
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
A - formed by Auto LAG
Number of channel-groups in use: 1
Number of aggregators: 1
Group Port-channel Protocol Ports
------+-------------+-----------+-----------------------------------------------
1 Po1(SU) LACP Gi5/0/17(P) Gi5/0/22(P) Gi6/0/12(P)


Device# show vlan private-vlan
Primary Secondary Type Ports
------- --------- ----------------- ------------------------------------------
20 501 community Po1
20 502 community Po1
20 503 isolated Po1
```

Example: Monitoring Private VLANs

This example shows output from the **show vlan private-vlan** command:

```
Device# show vlan private-vlan
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
20      501       isolated          Gi1/0/22, Gi1/0/2
20      502       community         Gi1/0/2
20      503       community         Gi1/0/2
```

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN trunking
- VLAN Membership Policy Server (VMPS)
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Command Reference (Catalyst 9400 Series Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management Information Base
RFC 2021	Remote Network Monitoring Management Information Base Version 2 using SMIv2

MIBs

MIB	MIBs Link
All the supported MIBs for this release. • BRIDGE-MIB (RFC1493) • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB (RFC 1573) • RMON-MIB (RFC 1757) • RMON2-MIB (RFC 2021)	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

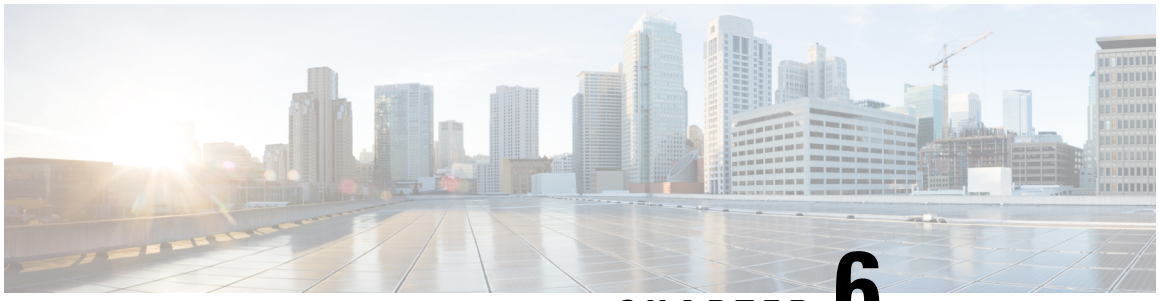
Feature History for Private VLANs

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	Private VLANs	The private VLAN feature addresses the problem that service providers face when using VLANs: • When running the Network Essentials or Network Advantage license, the device supports up to 4094 active VLANs. If a service provider assigns one VLAN per customer, this limits the numbers of customers the service provider can support. • To enable IP routing, each VLAN is assigned a subnet address space or a block of addresses, which can result in wasting the unused IP addresses, and cause IP address management problems.
Cisco IOS XE Amsterdam 17.3.1	Private VLAN for trunk ports and port channels	Private VLAN was introduced on isolated trunk ports, promiscuous trunk ports, and on port channels.
Cisco IOS XE Cupertino 17.7.1	Private VLANs Private VLAN for trunk ports and port channels	These features were implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.



CHAPTER 6

Configuring Layer 3 Subinterfaces

This module describes how to configure the dot1q VLAN subinterfaces on a Layer 3 interface, which forwards IPv4 and IPv6 packets to another device using static or dynamic routing protocols. You can use Layer 3 interfaces for IP routing and inter-VLAN routing of Layer 2 traffic.

- Restrictions for Configuring Layer 3 Subinterfaces, on page 97
- Information About Layer 3 Subinterfaces, on page 98
- How to Configure Layer 3 Subinterfaces, on page 99
- Example: Configuring Layer 3 Subinterfaces, on page 100
- Feature History for Layer 3 Subinterfaces, on page 100

Restrictions for Configuring Layer 3 Subinterfaces

- Subinterfaces are not supported on StackWise Virtual Link.
- Subinterfaces with Software-Defined Access (SD-Access) is not supported.
- Subinterfaces, physical routed interfaces, and VLAN interfaces share the same hardware resources. The combination of all three interfaces cannot exceed the platform limits.
- For a given physical interface, only one subinterface can have the encapsulation as native.
- For a given physical interface, each subinterface should have a unique encapsulation ID.
- A physical interface cannot be used for Layer 3 services (no IP address configurations are supported on the physical interface) if one of the subinterface is configured as native.
- Do not configure encapsulation on the native VLAN of an IEEE 802.1Q trunk without the **native** keyword. Always use the **native** keyword of the **dot1q vlan** command when the VLAN ID is the ID of the IEEE 802.1Q native VLAN.
- If you configure normal-range VLANs on subinterfaces, you cannot change the VLAN Trunking Protocol (VTP) mode from Transparent.
- If a Layer 3 port has a subinterface configured with dot1q as the native VLAN, Cisco recommends not to configure routing related configuration on the Layer 3 port as it will hinder the functionality of the native VLAN subinterface.

Information About Layer 3 Subinterfaces

A dot1q VLAN subinterface is a virtual Cisco IOS interface that is associated with a VLAN ID on a routed physical interface. A parent interface is a physical port. Subinterfaces can be created on Layer 3 physical interfaces and Layer 3 port channels. A subinterface can be associated with different functionalities such as IP addressing, forwarding policies, Quality of Service (QoS) policies, and security policies.

Subinterfaces divide the parent interface into two or more virtual interfaces on which you can assign unique Layer 3 parameters such as IP addresses and dynamic routing protocols. The IP address for each subinterface should be in a different subnet from any other subinterface on the parent interface.

You can create a subinterface with a name that consists of the parent interface name (for example, GigabitEthernet 1/0/33) followed by a period and then by a number that is unique for that subinterface. For example, you can create a subinterface for GigabitEthernet interface 1/0/33 named GigabitEthernet 1/0/33.1, where .1 indicates the subinterface.

One of the uses of subinterfaces is to provide unique Layer 3 interfaces to each VLAN that is supported by the parent interface. In this scenario, the parent interface connects to a Layer 2 trunking port on another device. You can configure a subinterface and associate the subinterface to a VLAN ID using 802.1Q trunking.

You can configure subinterfaces with any normal range or extended range VLAN ID in VLAN Trunking Protocol (VTP) transparent mode. Because VLAN IDs 1 to 1005 are global in the VTP domain and can be defined on other network devices in the VTP domain, you can use only extended range VLANs with subinterfaces in VTP client or server mode. In VTP client or server mode, normal-range VLANs are excluded from subinterfaces.

You can configure the same VLAN ID on a Layer 2 VLAN or Layer 3 VLAN interface and on a Layer 3 subinterface.

The following features and protocols are supported on Layer 3 subinterfaces:

- IPv4 addressing and routing.
- Unicast routing: Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and static routing.
- Multicast routing: Internet Group Management Protocol (IGMP), Protocol-Independent Multicast Sparse Mode (PIM-SM), Source Specific Multicast (SSM), and Multiprotocol Label Switching (MPLS).
- First-Hop Redundancy Protocol (FHRP) protocols: Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), and Gateway Load Balancing Protocol (GLBP).
- Unicast Reverse Path Forwarding (uRPF), and Equal-Cost Multipath (ECMP).
- Virtual routing and forwarding (VRF) lite.
- Router access control list and policy-based routing (PBR).
- Quality of Service (QoS): Marking and policing.
- Services: DHCP Server/Relay and NetFlow.
- Layer 3 EtherChannels.

How to Configure Layer 3 Subinterfaces

You can configure one or more subinterfaces on a routed interface. Configure the parent interface as a routed interface by using the **no switchport** command. The parent interface can have its own IP address, policies, and configurations attached to it. Untagged traffic and any tagged traffic or VLAN (not handled by the subinterface) that comes into the port are handled by the parent interface.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**interface** {*type switch* / *slot* / *port.subinterface*}
	Example:
	`Device(config)# interface GigabitEthernet 1/0/33.201`
	Or
	`Device(config)# interface range GigabitEthernet1/0/33.201- GigabitEthernet1/0/33.204`
		Selects an interface or interface range and enters subinterface configuration mode. (To remove an interface, use the **no** form of this command.)
		• You can also specify a range of interfaces along with the associated dot1q VLAN IDs as shown in this example.

Step 4	**encapsulation dot1q** *vlan-id* [**native**] **Example:** `Device(config-subif)# encapsulation dot1q 201 native`	Configures 802.1Q encapsulation for the subinterface. The range is from 1 to 4000. (To remove 802.1Q encapsulation for the subinterface, use the **no** form of this command.) • **native**: To make a subinterface the default handler of untagged packets coming to the port, use this keyword. If you configure this keyword on the subinterface, and IP and other configurations are also configured on the parent interface, this keyword overwrites the configurations on the parent interface. Configure either this keyword on the subinterface or the configurations on the parent interface at a time. **Note** **shutdown** and **no shutdown** commands can be used to shutdown or reverse the shutdown on a specific subinterface without affecting the

	Command or Action	Purpose
		traffic passing through the parent interface or other subinterfaces.
Step 5	**end** **Example:** `Device(config-subif)# end`	Exits subinterface mode and returns to privileged EXEC mode.

Example: Configuring Layer 3 Subinterfaces

The following example shows how to configure subinterfaces on layer 3 interfaces:

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/33
Device(config-if)# no switchport
Device(config-if)# no ip address
Device(config-if)# exit
Device(config)# interface GigabitEthernet 1/0/33.201
Device(config-subif)# encapsulation dot1q 201 native
Device(config-subif)# end
```

The following example shows how to configure subinterfaces on layer 3 port channels:

```
Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# no switchport
Device(config-if)# no ip address
Device(config-if)# exit
Device(config)# interface port-channel 2.10
Device(config-subif)# encapsulation dot1q 10
Decvice(config-subif)# ip address 10.10.10.11 255.255.255.0
Device(config-subif)# end
```

Feature History for Layer 3 Subinterfaces

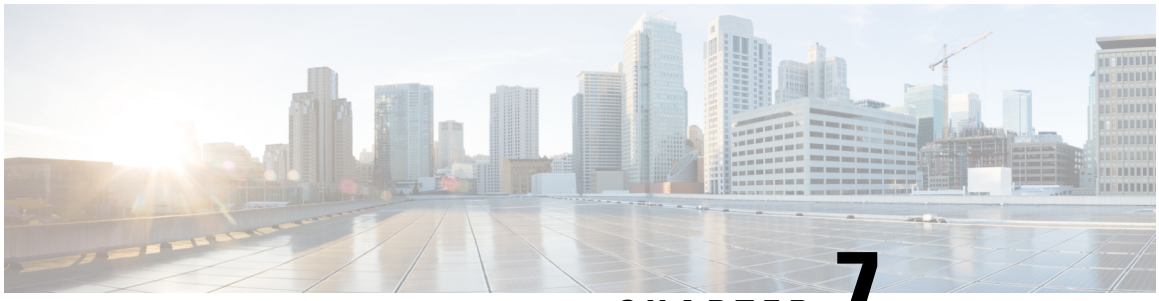
This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Table 7: Feature History for Layer 3 Subinterfaces

Release	Feature	Feature Information
Cisco IOS XE Gibraltar 16.10.1	Layer 3 Subinterfaces	Layer 3 interfaces forward IPv4 and IPv6 packets to another device using static or dynamic routing protocols. You can use Layer 3 interfaces for IP routing and inter-VLAN routing of Layer 2 traffic.
Cisco IOS XE Gibraltar 16.12.1	EtherChannel and Multiprotocol Label Switching	These features were introduced on Layer 3 subinterfaces.
Cisco IOS XE Cupertino 17.7.1	Layer 3 Subinterfaces	This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.



CHAPTER 7

Configuring Wired Dynamic PVLAN

The following sections provide information about configuring wired dynamic PVLAN:

- Restrictions for Wired Dynamic PVLAN, on page 103
- Information About Wired Dynamic PVLAN, on page 103
- Configuring Wired Dynamic PVLAN, on page 105
- Feature History for Wired Dynamic PVLAN, on page 110

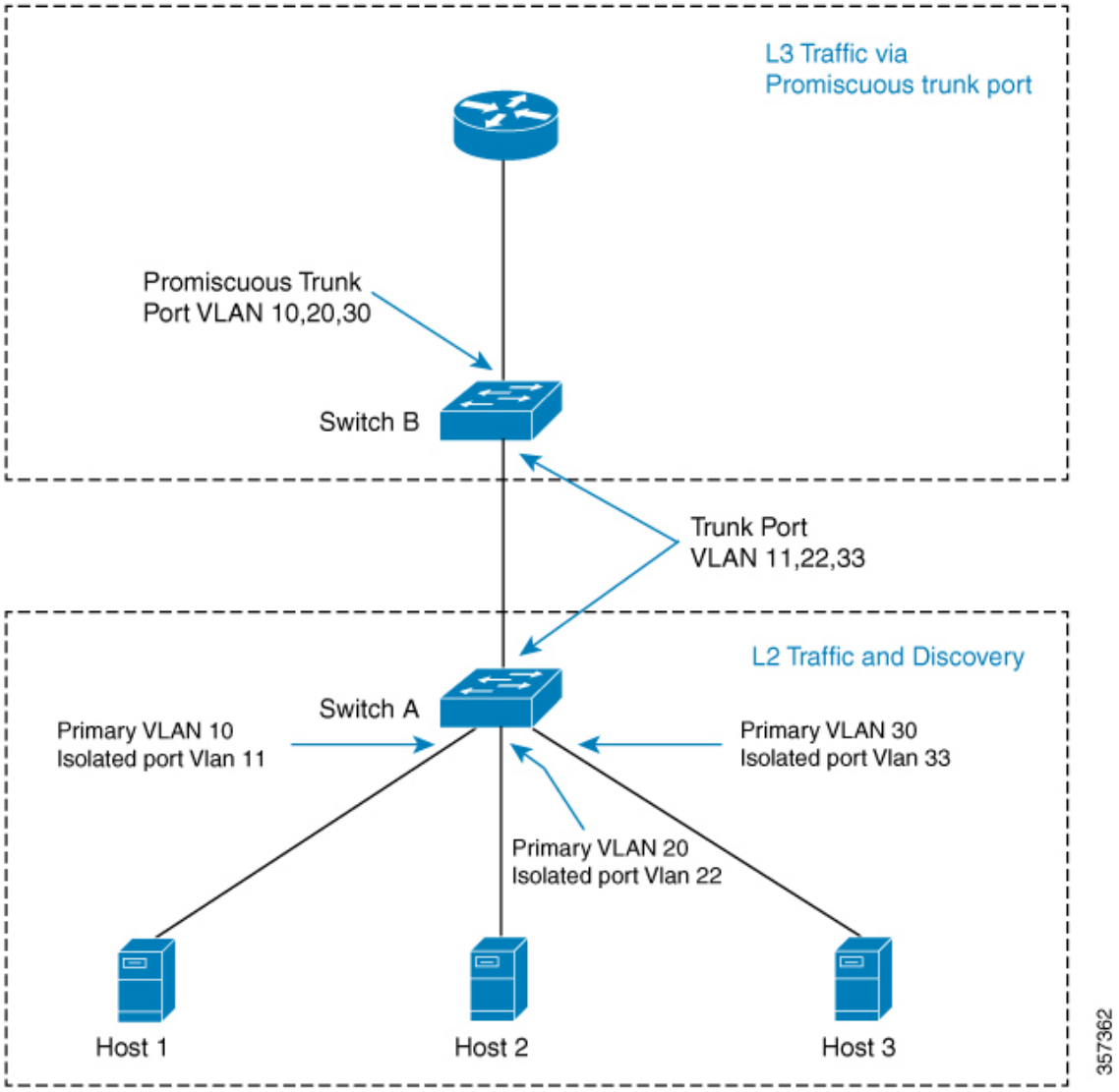
Restrictions for Wired Dynamic PVLAN

- High availability is not supported with Wired Dynamic PVLAN.
- Voice VLAN configuration cannot co-exist with this feature.
- Local Web Authentication (LWA) and Central Web Authentication (CWA) cannot be used with this feature.
- All wired clients using the dynamic PVLAN interface template will be programmed as data clients.
- Only interfaces with existing Access or PVLAN Host switchport mode support PVLAN template.
- Identity Based Networking Services 2.0 (IBNS 2.0) must be used for dynamic template support.

Information About Wired Dynamic PVLAN

Wired Dynamic PVLAN is a feature that uses a private VLAN with AAA authorization to isolate clients and provide Zero-Trust. It is a method to block peer to peer communications within a subnet/VLAN. Here, the client is assigned to a PVLAN which isolates a wired client connected on one port from all other ports on Layer 2 while the Layer 3 communication occurs via the promiscuous port. In this feature, a single wired data client is supported per port interface, to ensure point-to-point blocking.

Note Traffic from multiple clients on the same interface will not be blocked.



In this topology, the hosts are connected to Switch A and they can communicate only with the promiscuous trunk port on the switch. The PVLAN can be extended to span across multiple switches by adding intermediate switches. If there is a switch (Switch C) between Switch A and Switch B in the above topology, then layer 2 trunk ports need to be configured on the intermediate links. If case of a community VLAN, it allows packets to be seen on other hosts within the same community VLAN.

When a host is connected to a switch port with a cable, it is placed into an Isolated PVLAN where it cannot discover any other hosts. The host is then authenticated by the RADIUS server. Another scenario is when the port is placed in closed mode, and if the port is not authenticated, only Extensible Authentication Protocol over LAN (EAPoL) packets are allowed. Once the port is authenticated it is placed into an Isolated VLAN dynamically. As the host first authenticates with the RADIUS server, it sends the name of a dynamic interface template to be applied to the host's port. This interface template contains the configurations to enable the PVLAN Primary and Secondary VLANs on the port. With the template applied to the host, the switchport mode will be changed which will cause the port to flap from access mode to PVLAN mode.

Note The interface template with the same name as referred by AAA Authorization needs to be configured on the switch.

When the interface template is being applied, the port will physically go down for a time period set by the sticky timer and come up again. When the RADIUS server sends the interface template a second time, it is ignored as the conversion has been completed. The port is then assigned to a PVLAN which keeps it isolated. The host completes authorization and comes up to ready state.

Configure the keep time for which the interface template information is retained before it is removed from the port using the **access-session interface-template sticky timer** *time* command.

Configuring Wired Dynamic PVLAN

To configure Wired Dynamic PVLAN, perform these steps on the user device (Switch A in the above topology):

Before you begin

Ensure that the dot1x aaa is configured on the user device.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**vlan** *vlan-id* **Example:** `Device(config)# vlan 200`	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be an isolated VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 4	**private-vlan isolated** **Example:** `Device(config-vlan)# private-vlan isolated`	Designates the VLAN as an isolated VLAN.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Device(config-vlan)# exit`	Returns to global configuration mode.
Step 6	**vlan** *vlan-id* **Example:** `Device(config)# vlan 100`	Enters VLAN configuration mode and designates or creates a VLAN that will be the primary VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 7	**private-vlan primary** **Example:** `Device(config-vlan)# private-vlan primary`	Designates the VLAN as the primary VLAN.
Step 8	**private-vlan association** [**add** | **remove**] *secondary_vlan_list* **Example:** `Device(config-vlan)# private-vlan association 200`	Associates the secondary VLANs with the primary VLAN. It can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The *secondary_vlan_list* parameter can contain multiple community VLAN IDs but only one isolated VLAN ID. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to associate secondary VLANs with a primary VLAN. • Use the **remove** keyword with a *secondary_vlan_list* to clear the association between secondary VLANs and a primary VLAN. • The command does not take effect until you exit VLAN configuration mode.
Step 9	**exit** **Example:**	Returns to global configuration mode.

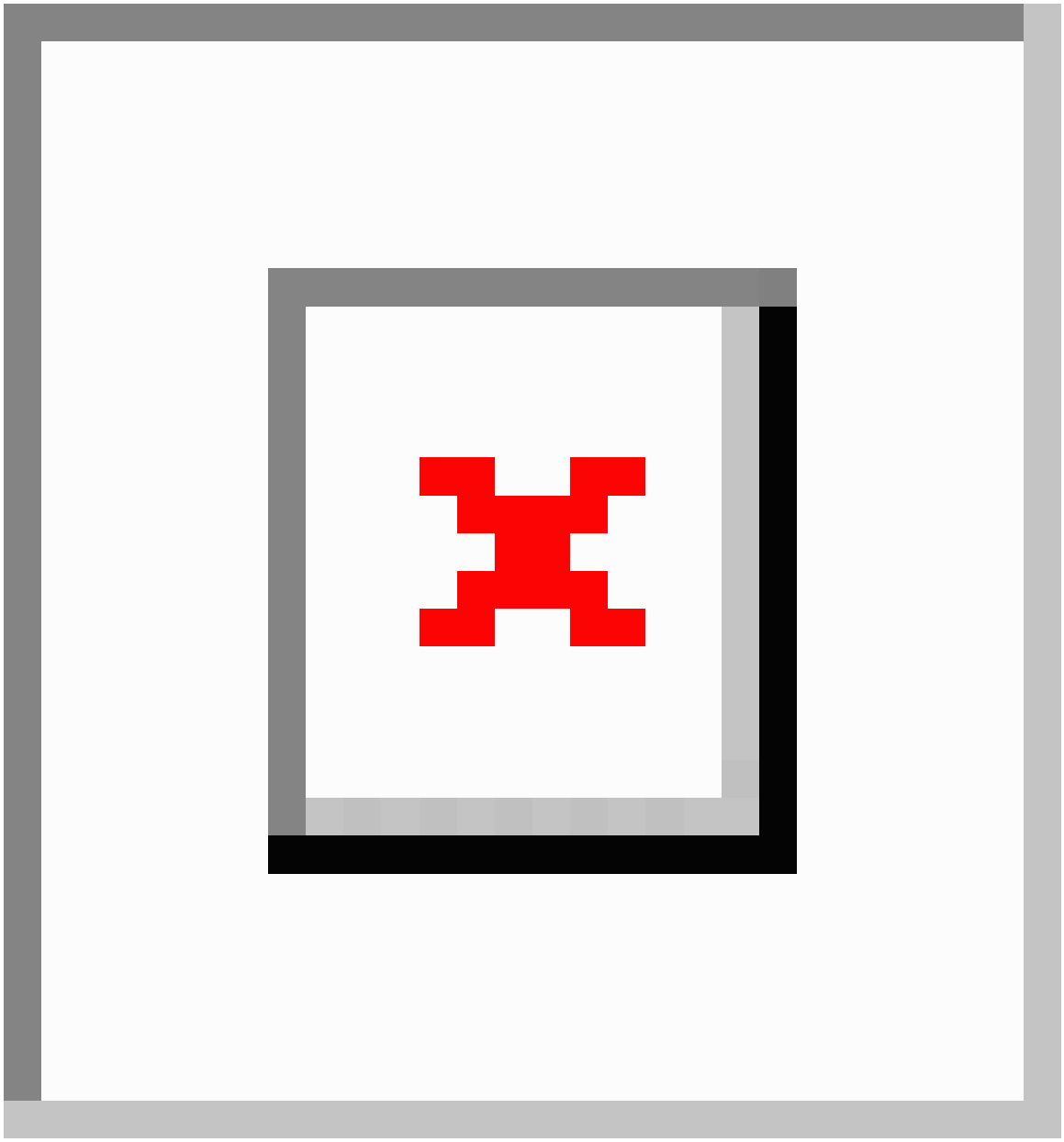
	Command or Action	Purpose
	`Device(config-vlan)# exit`	
Step 10	**template** *template-name* **Example:** `Device(config)# template PVLAN100_200_CFG`	Creates a user template and enters template configuration mode.
Step 11	**switchport mode private-vlan host** **Example:** `Device(config-template)# switchport mode private-vlan host`	Configures a Layer 2 port as a PVLAN host port on the template.
Step 12	**switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id* **Example:** `Device(config-template)# switchport private-vlan host-association 100 200`	Configures the association of a Layer 2 port with a PVLAN on the template.
Step 13	**exit** **Example:** `Device(config-template)# exit`	Returns to global configuration mode.
Step 14	**access-session interface-template sticky timer** *time* **Example:** `Device(config)# access-session interface-template sticky timer 60`	Configures the keep time of the template globally. Once the last client leaves, the template will be removed from the port after the configured keep time. **Note** It is recommended that you set the sticky timer to 60 seconds.
Step 15	**interface** *interface-id* **Example:** `Device(config)# interface GigabitEthernet1/0/1`	Enters interface configuration mode and specifies the interface.
Step 16	**access-session interface-template sticky timer** *time*	Configures the keep time of the template on the interface. Once the last client leaves, the

	Command or Action	Purpose
	Example: `Device(config-if)# access-session interface-template sticky timer 60`	template will be removed from the port after the configured keep time. **Note** It is recommended that you set the sticky timer to 60 seconds.
Step 17	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

What to do next

After the above steps, configure the Identity Services Engine (ISE) or any other RADIUS server to assign the template to the client's port interface after the client has been authenticated successfully.

Figure 5: Configuring the ISE to Assign the Interface Template



If you are using the ISE, go to the **Policy** > **Policy Elements** > **Authorization** > **Authorization Profile** page. Check the **Interface Template** check box and enter the name of the template to be assigned to the client interface.

If you are using a different RADIUS server, the attribute **Cisco-AVpair="interface:template=name"** must be pushed to the switch after the initial client authentication has been completed.

Feature History for Wired Dynamic PVLAN

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Bengaluru 17.5.1	Wired Dynamic PVLAN (Whitelist P2P Blocking)	Wired Dynamic PVLAN feature uses a private VLAN to isolate the clients and provide Zero-Trust. This methods blocks peer to peer communication within a subnet/VLAN. The client is assigned to a PVLAN which isolates a single wired client connected on a port from other ports.
Cisco IOS XE Cupertino 17.7.1	Wired Dynamic PVLAN	This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/.