Cisco Nexus 7000 Series NX-OS Layer 2 Switching Configuration Guide

First Published: 2011-09-27

Last Modified: 2017-11-29

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

Text Part Number: OL-25755-04

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.

CONTENTS

PREFACE

CHAPTER 1

CHAPTER 2

CHAPTER 3

Preface

The preface contains the following sections:

Preface

This preface describes the audience, organization, and conventions of the Book Title. It also provides information on how to obtain related documentation.

This chapter includes the following topics:

Audience

This publication is for experienced network administrators who configure and maintain Cisco NX-OS on Cisco Nexus 7000 Series Platform switches.

Document Conventions

Note

- As part of our constant endeavor to remodel our documents to meet our customers' requirements, we have modified the manner in which we document configuration tasks. As a result of this, you may find a deviation in the style used to describe these tasks, with the newly included sections of the document following the new format.
- The Guidelines and Limitations section contains general guidelines and limitations that are applicable to all the features, and the feature-specific guidelines and limitations that are applicable only to the corresponding feature.

Command descriptions use the following conventions:

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

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

Examples use the following conventions:

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

This document uses the following conventions:

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

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

Related Documentation

Documentation for Cisco Nexus 7000 Series Switches is available at:

- Configuration Guides

http://www.cisco.com/c/en/us/support/switches/nexus-7000-series-switches/products-installation-and-configuration-guides-list.html

- Command Reference Guides

  http://www.cisco.com/c/en/us/support/switches/nexus-7000-series-switches/products-command-reference-list.html

- Release Notes

  http://www.cisco.com/c/en/us/support/switches/nexus-7000-series-switches/products-release-notes-list.html

- Install and Upgrade Guides

  http://www.cisco.com/c/en/us/support/switches/nexus-7000-series-switches/products-installation-guides-list.html

- Licensing Guide

  http://www.cisco.com/c/en/us/support/switches/nexus-7000-series-switches/products-licensing-information-listing.html

Documentation for Cisco Nexus 7000 Series Switches and Cisco Nexus 2000 Series Fabric Extenders is available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-2000-series-fabric-extenders/products-installation-and-configuration-guides-list.html

Documentation Feedback

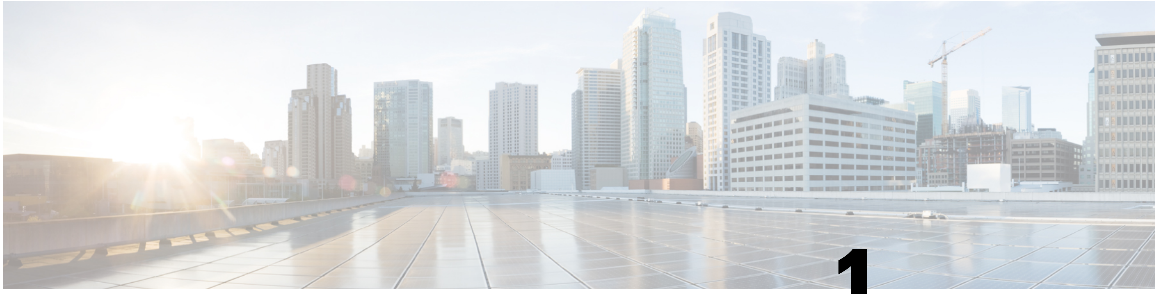
To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus7k-docfeedback@cisco.com. We appreciate your feedback.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER 1

New and Changed Information

New and Changed Information

Your software release might not support all the features in this document. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release.

This book does not contain any new feature for this release.

CHAPTER 2

Overview

This chapter provides an overview of the Cisco NX-OS devices that support Layer 2 features.

This chapter includes the following sections:

Licensing Requirements

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the *Cisco NX-OS Licensing Guide*.

Layer 2 Ethernet Switching Overview

The device supports simultaneous, parallel connections between Layer 2 Ethernet segments. Switched connections between Ethernet segments last only for the duration of the packet. New connections can be made between different segments for the next packet.

The device solves congestion problems caused by high-bandwidth devices and a large number of users by assigning each device (for example, a server) to its own domain. Because each LAN port connects to a separate Ethernet collision domain, servers in a switched environment achieve full access to the bandwidth.

Because collisions cause significant congestion in Ethernet networks, an effective solution is full-duplex communication. Typically, 10/100-Mbps Ethernet operates in half-duplex mode, which means that stations can either receive or transmit. In full-duplex mode, which is configurable on these interfaces, two stations can transmit and receive at the same time. When packets can flow in both directions simultaneously, the effective Ethernet bandwidth doubles. 1/10-Gigabit Ethernet operates in full duplex only.

VLANs

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

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

All ports are assigned to the default VLAN (VLAN1) when the device first comes up. A VLAN interface, or switched virtual interface (SVI), is a Layer 3 interface that is created to provide communication between VLANs.

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

Note Inter-Switch Link (ISL) trunking is not supported on the Cisco NX-OS.

Private VLANs

Private VLANs provide traffic separation and security at the Layer 2 level.

A private VLAN is one or more pairs of a primary VLAN and a secondary VLAN, all with the same primary VLAN. The two types of secondary VLANs are isolated and community VLANs. Hosts on isolated VLANs communicate only with hosts in the primary VLAN. Hosts in a community VLAN can communicate only among themselves and with hosts in the primary VLAN but not with hosts in isolated VLANs or in other community VLANs.

Regardless of the combination of isolated and community secondary VLANs, all interfaces within the primary VLAN comprise one Layer 2 domain, and therefore, require only one IP subnet.

Spanning Tree

This section discusses the implementation of the Spanning Tree Protocol (STP) on the software. Spanning tree is used to refer to IEEE 802.1w and IEEE 802.1s. When the IEEE 802.1D Spanning Tree Protocol is referred to in the publication, 802.1D is stated specifically.

STP Overview

STP provides a loop-free network at the Layer 2 level. Layer 2 LAN ports send and receive STP frames, which are called Bridge Protocol Data Units (BPDUs), at regular intervals. Network devices do not forward these frames but use the frames to construct a loop-free path.

802.1D is the original standard for STP, and many improvements have enhanced the basic loop-free STP. You can create a separate loop-free path for each VLAN, which is named Per VLAN Spanning Tree (PVST+).

Additionally, the entire standard was reworked to make the loop-free convergence process faster to keep up with the faster equipment. This STP standard with faster convergence is the 802.1w standard, which is known as Rapid Spanning Tree (RSTP). Now, these faster convergence times are available as you create STP for each VLAN, which is known as Per VLAN Rapid Spanning Tree (Rapid PVST+).

Finally, the 802.1s standard, Multiple Spanning Tree (MST), allows you to map multiple VLANs into a single spanning tree instance. Each instance runs an independent spanning tree topology.

Although the software can interoperate with legacy 802.1D systems, the system runs Rapid PVST+ and MST. You can use either Rapid PVST+ or MST in a given VDC; you cannot mix both in one VDC. Rapid PVST+ is the default STP protocol for Cisco NX-OS devices.

Note Cisco NX-OS uses the extended system ID and MAC address reduction; you cannot disable these features.

In addition, Cisco has created some proprietary features to enhance the spanning tree activities.

Rapid PVST+

Rapid PVST+ is the default spanning tree mode for the software and is enabled by default on the default VLAN and all newly created VLANs.

A single instance, or topology, of RSTP runs on each configured VLAN, and each Rapid PVST+ instance on a VLAN has a single root device. You can enable and disable STP on a per-VLAN basis when you are running Rapid PVST+.

MST

The software also supports MST. The multiple independent spanning tree topologies enabled by MST provide multiple forwarding paths for data traffic, enable load balancing, and reduce the number of STP instances required to support a large number of VLANs.

MST incorporates RSTP, so it also allows rapid convergence. MST improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).

Note Changing the spanning tree mode disrupts the traffic because all spanning tree instances are stopped for the previous mode and started for the new mode.

You can force specified interfaces to send prestandard, rather than standard, MST messages using the command-line interface.

STP Extensions

The software supports the following Cisco proprietary features:

- Spanning tree port types—The default spanning tree port type is normal. You can configure interfaces connected to Layer 2 hosts as edge ports and interfaces connected to Layer 2 switches or bridges as network ports.

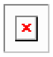
- Bridge Assurance—Once you configure a port as a network port, Bridge Assurance sends BPDUs on all ports and moves a port into the blocking state if it no longer receives BPDUs. This enhancement is available only when you are running Rapid PVST+ or MST.
- BPDU Guard—BPDU Guard shuts down the port if that port receives a BPDU.
- BPDU Filter—BPDU Filter suppresses sending and receiving BPDUs on the port.
- Loop Guard—Loop guard helps prevent bridging loops that could occur because of a unidirectional link failure on a point-to-point link.
- Root Guard—The root guard feature prevents a port from becoming root port or blocked port. If a port configured for root guard receives a superior BPDU, the port immediately goes to the root-inconsistent (blocked) state.

Virtualization

Cisco NX-OS devices introduce support for multiple virtual device contexts (VDCs) on a single switching device. Each VDC is treated as a standalone device with specific resources, such as physical interfaces, allocated to each VDC by the network admin role. An administrator is assigned to each VDC and that administrator has a limited view of the system within that specific VDC. Faults are also isolated to within the specific VDC.

This VDC concept applies to all features on Cisco NX-OS, including all Layer 2 switching features.

Figure 1: VDCs with Layer 2 Services

All processes work independently in each VDC. You can reuse the process identification numbers in different VDCs. This figure shows how to reuse the VLAN 100 identifier in each separate VDC. 

Each VDC acts as a standalone device with Layer 2 services available. VDCs allow you to share a physical device among several logical functions. You can provision and assign entirely separate Layer 2 resources to individual VDCs.

You can configure several VDCs, and each VDC is a group of physical device resources. You assign resources and user roles for each VDC. VDCs allows flexible resources as well as enhanced fault isolation.

VDCs allow the separation of processes and management environments, providing well-defined fault and administrative boundaries between logical devices. Each VDC can be considered as a separate device with its own configuration, resources, users, and management interface.

VDCs define different administrator levels, or roles, that can access and administer each VDC. Commands outside the scope of a given user role are either hidden from that user’s view or can return an error if the command is entered. This feature limits the number of users who can access the entire physical device and introduce traffic-disrupting misconfigurations.

Note See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on virtual device contexts (VDCs) and assigning resources.

See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for information on restartability and seamless transitions.

Related Topics

The following documents are related to the Layer 2 switching features:

- *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*
- *Cisco DCNM Layer 2 Switching Configuration Guide*
- *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
- *Cisco Nexus 7000 Series NX-OS Security Configuration Guide*
- *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
- *Cisco NX-OS Licensing Guide*
- *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
- *Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*

CHAPTER 3

Configuring Layer 2 Switching

This chapter describes how to configure Layer 2 switching using Cisco NX-OS.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About Layer 2 Switching

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on creating interfaces.

You can configure Layer 2 switching ports as access or trunk ports. Trunks carry the traffic of multiple VLANs over a single link and allow you to extend VLANs across an entire network. All Layer 2 switching ports maintain MAC address tables.

Note See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on high-availability features.

Layer 2 Ethernet Switching Overview

The device supports simultaneous, parallel connections between Layer 2 Ethernet segments. Switched connections between Ethernet segments last only for the duration of the packet. New connections can be made between different segments for the next packet.

The device solves congestion problems caused by high-bandwidth devices and a large number of users by assigning each device (for example, a server) to its own domain. Because each LAN port connects to a separate Ethernet collision domain, servers in a switched environment achieve full access to the bandwidth.

Because collisions cause significant congestion in Ethernet networks, an effective solution is full-duplex communication. Typically, 10/100-Mbps Ethernet operates in half-duplex mode, which means that stations can either receive or transmit. In full-duplex mode, which is configurable on these interfaces, two stations can transmit and receive at the same time. When packets can flow in both directions simultaneously, the effective Ethernet bandwidth doubles. 1/10-Gigabit Ethernet operates in full duplex only.

Switching Frames Between Segments

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

To reduce signal degradation, the device considers each LAN port to be an individual segment. When stations connected to different LAN ports need to communicate, the device forwards frames from one LAN port to the other at wire speed to ensure that each session receives full bandwidth.

To switch frames between LAN ports efficiently, the device maintains an address table. When a frame enters the device, it associates the media access control (MAC) address of the sending network device with the LAN port on which it was received.

Building the Address Table and Address Table Changes

The device dynamically builds the address table by using the MAC source address of the frames received. When the device receives a frame for a MAC destination address not listed in its address table, it floods the frame to all LAN ports of the same VLAN except the port that received the frame. When the destination station replies, the device adds its relevant MAC source address and port ID to the address table. The device then forwards subsequent frames to a single LAN port without flooding all LAN ports.

You can configure MAC addresses, which are called static MAC addresses, to statically point to specified interfaces on the device. These static MAC addresses override any dynamically learned MAC addresses on those interfaces. You cannot configure broadcast addresses as static MAC addresses. Beginning with Cisco NX-OS Release 5.2(1), multicast MAC addresses can be configured as static MAC addresses. For further information, see the “Configuring IGMP Snooping” of the *Cisco Nexus 7000 Series NX-OS Multicast Routing Configuration Guide*. The static MAC entries are retained across a reboot of the device.

Beginning with Cisco NX-OS Release 4.1(5), you must manually configure identical static MAC addresses on both devices connected by a virtual port channel (vPC) peer link. The MAC address table display is enhanced to display information on MAC addresses when you are using vPCs.

See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information about vPCs.

The address table can store a number of MAC address entries depending on the hardware I/O module. The device uses an aging mechanism, defined by a configurable aging timer, so if an address remains inactive for a specified number of seconds, it is removed from the address table.

See the *Cisco Nexus 7000 Series NX-OS Security Command Reference* for information on MAC port security.

Consistent MAC Address Tables on the Supervisor and on the Modules

Optimally, all the MAC address tables on each module exactly match the MAC address table on the supervisor. Beginning with Cisco NX-OS 4.1(2), when you enter the **show forwarding consistency l2** command, the device displays discrepant, missing, and extra MAC address entries.

Layer 3 Static MAC Addresses

Beginning with Release 4.2, you can configure a static MAC address for all Layer 3 interfaces. The default MAC address for the Layer 3 interfaces is the VDC MAC address.

You can configure a static MAC address for the following Layer 3 interfaces:

- Layer 3 interfaces
- Layer 3 subinterfaces
- Layer 3 port channels
- VLAN network interface

Note You cannot configure static MAC address on tunnel interfaces.

See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on configuring Layer 3 interfaces.

High Availability for Switching

You can upgrade or downgrade the software seamlessly, with respect to classical Ethernet switching. Beginning with Release 4.2(1), if you have configured static MAC addresses on Layer 3 interfaces, you must unconfigure those ports in order to downgrade the software.

Note See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on high availability features.

Virtualization Support for Layer 2 Switching

The device supports virtual device contexts (VDCs), and the configuration and operation of the MAC address table are local to the VDC.

Note — See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on VDCs and assigning resources.

MAC Address Movement

Rapid MAC address movement, caused by either Layer 2 loop or other system events (for example: misconfiguration, dual-active server cluster, and so on), if not limited could eventually overload the supervisor and potentially impact other processes. Such situation might lead to an overall instability of the control plane. To avoid this situation rapid MAC move protection has been implemented in the processes that handle MAC addresses learning.

MAC Move Protection

The following methods protect the SUP from excessive mac move:

- Software throttle: Using **mac address loop-detect flow-control-fe** command.
- Hardware throttle: Using **mac address loop-detect disable-learn-vlan** command.

Software throttling is enabled by default and this is the recommended method. You can use only one throttling method at a time. The throttling commands should be executed within a VDC.

Software Throttle

In software throttle, the mac-move notifications are throttled so the rate of mac-move notification is limited from the module to the supervisor.

This throttling is usually done per Forwarding Engine [FE] (per ASIC level) on a specific module. If necessary (for example: during rapid mac move across all modules in the system) global throttling is invoked that would throttle notification from all FEs on all modules in order protect the supervisor.

Hardware Throttle

In hardware throttle, mac-learning is disabled on a particular VLAN (for all FE and all modules) for specific time and then re-enabled. This throttling can be done per VLAN level (per VLAN throttle) or for all VLANs (global throttle).

Increasing the Throttle

In case the software throttle is found to be inadequate, in extreme cases, the mac-move information sent from the line card module is reduced.

This method is not a recommended option and should be exercised with caution.

Note — Increasing the threshold could make the system unstable if not set accordingly to the device scale.

The reduction in mac-move information sent is done in two ways:.

- Reduce number of notifications that can be batched.
- Change/increase the time-period after which this notification batch can be sent from the module to the supervisor module.

Use the **mac address throttle-buffer-intv** { **max** |**optimal**} command (to be executed within a VDC) to increase the throttle by tuning the throttle buffer and the scan duration on the line card module.

When the **max** keyword is used, the throttling is maximum. It means information sent from the line card module to the supervisor is reduced and are spaced out more.

When the **optimal** keyword is used, the throttling is medium.

When this command is not used, the throttling is minimum (which is the default).

Prerequisites for Configuring MAC Addresses

MAC addresses have the following prerequisites:

- You must be logged onto the device.

Guidelines and Limitations for Configuring MAC Addresses

MAC addresses have the following configuration guidelines and limitations:

MAC Address Table	Age Group
M1 Line Cards	128,000 entries
F1 Line Cards	16,000 to 256,000 entries
F2 and F2e Line Cards	16,000 to 192,000 entries

Note The F2 and F2e modules synchronize the MAC address tables for a VLAN across all Switch on Chips (SoCs) present in a virtual device context (VDC) when a switch virtual interface (SVI) for the VLAN is configured. Synchronizing the MAC address tables can reduce the number of MAC addresses supported in a VDC to 16,000.

Beginning with NX-OS Release 6.0.1, the learning mode feature is supported. Learning mode has the following configuration guidelines and limitations:

Line Cards	Classic Ethernet (CE) Nonconversational Learning Supported	Classic Ethernet (CE) Conversational Learning Supported	Fabric Path Conversational Learning	Fabric Path Nonconversational Learning
M1	Yes	NA	NA	NA
F1	Yes	Yes	Yes	No
F2 and F2e	Yes	Yes	Yes	Yes, if the switch virtual interface (SVI) is configured.

Note When you configure a static MAC address on a vPC switch, ensure to configure a corresponding static MAC address on the other vPC switch. If you configure the static MAC address only on one of the vPC switches, the other vPC switch will not learn the MAC address dynamically.

Default Settings for Layer 2 Switching

This table lists the default setting for Layer 2 switching parameters.

Table 1: Default Layer 2 Switching Parameters

Parameters	Default
Aging time	1800 seconds

Beginning with NX-OS Release 6.0.1, the learning mode feature is supported. This table lists the default learning mode parameters.

Table 2: Default Learning Mode Parameters

Parameters	Default
Classic Ethernet (CE) VLAN	Nonconversational
Fabric Path VLANs	Conversational

Configuring Layer 2 Switching

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Configuring a Static MAC Address

You can configure MAC addresses, which are called static MAC addresses, to statically point to specified interfaces on the device. These static MAC addresses override any dynamically learned MAC addresses on those interfaces. You cannot configure broadcast addresses as static MAC addresses. Beginning with Cisco NX-OS Release 5.2(1), multicast MAC addresses can be configured as static MAC addresses. For further information, see the "Configuring IGMP Snooping" of the *Cisco Nexus 7000 Series NX-OS Multicast Routing Configuration Guide*.

Before you begin

Before you configure static MAC addresses, ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**mac address-table static** *mac-address* **vlan** *vlan-id* {[**drop** | **interface** {*type slot/port*} | **port-channel** *number*]}
	Example:
	`switch(config)# mac address-table static 1.1.1 vlan 2 interface ethernet 1/2`
		Specifies a static MAC address to add to the Layer 2 MAC address table.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show mac address-table static** **Example:** `switch# show mac address-table static`	Displays the static MAC addresses.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to put a static entry in the Layer 2 MAC address table:

```
switch# config t
switch(config)# mac address-table static 1.1.1 vlan 2 interface ethernet 1/2
switch(config)#
```

Configuring a Static MAC Address on a Layer 3 Interface

Beginning with Release 4.2(1), you can configure static MAC addresses on Layer 3 interfaces. You cannot configure broadcast addresses as static MAC addresses. Beginning with Cisco NX-OS Release 5.2(1), multicast

MAC addresses can be configured as static MAC addresses. For further information, see the "Configuring IGMP Snooping" of the *Cisco Nexus 7000 Series NX-OS Multicast Routing Configuration Guide*.

Note You cannot configure static MAC addresses on tunnel interfaces.

See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on configuring Layer 3 interfaces.

Before you begin

Before you configure static MAC addresses, ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** [**ethernet** *slot/port* | **ethernet** *slot/port.number* | **port-channel** *number* | **vlan** *vlan-id*] **Example:** `switch(config)# interface ethernet 7/3`	Specifies the Layer 3 interface and enters interface configuration mode. **Note** You must create the Layer 3 interface before you can assign the static MAC address.

Step 3	**mac-address** *mac-address*
	Example:
	`switch(config-if)# mac-address`
	`22ab.47dd.ff89`
	`switch(config-if)#`
		Specified a static MAC address to add to the Layer 3 interface.

Step 4	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 5	(Optional) **show interface** [**ethernet** *slot/port* | **ethernet** *slot/port.number* | **port-channel** *number* | **vlan** *vlan-id*] **Example:** `switch# show interface ethernet 7/3`	Displays information about the Layer 3 interface.
Step 6	(Optional) **copy running-config startup-config** **Example:**	Copies the running configuration to the startup configuration.

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

Example

This example shows how to configure the Layer 3 interface on slot 7, port 3 with a static MAC address:

```
switch# config t
switch(config)# interface ethernet 7/3
switch(config-if)# mac-address 22ab.47dd.ff89
switch(config-if)#
```

Configuring the Aging Time for the MAC Address Table

You can configure the amount of time that a MAC address entry (the packet source MAC address and port on which that packet was learned) remains in the MAC address table, which contains the Layer 2 information.

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

Before you begin

Before you configure the aging time for the MAC address table, ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**mac address-table aging-time** *seconds* [**vlan** *vlan_id*] **Example:** `switch(config)# mac address-table aging-time 600`	Specifies the time before an entry ages out and is discarded from the Layer 2 MAC address table. The range is from 120 to 918000; the default is 1800 seconds. Entering the value 0 disables the MAC aging.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

	Command or Action	Purpose
Step 4	(Optional) **show mac address-table aging-time** **Example:** `switch# show mac address-table aging-time`	Displays the aging time configuration for MAC address retention.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to set the ageout time for entries in the Layer 2 MAC address table to 600 seconds (10 minutes):

```
switch# config t
switch(config)# mac address-table aging-time 600
switch(config)#
```

Configuring Learning Mode for VLANs

Beginning with NX-OS Release 6.0.1, configuring the learning mode for VLANs is supported. Based on the learning mode configured, the Cisco NX-OS software can install MAC addresses in hardware either conversationally or nonconversationally.

Before you begin

Before you configure the learning mode for VLANs, ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**mac address-table learning-mode conversational** *vlan-range of CE-vlans* **Example:** `switch(config)# mac address-table learning-mode conversational vlan1`	Specifies the learning mode for the Layer 2 MAC address table. The options are conversational learning and nonconversational learning.

	Command or Action	Purpose

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Example

This example shows how to set the learning mode to conversational for the VLANs:

```
switch# config t
switch(config)# mac address-table learning-mode conversational vlan1
switch(config)# end
switch(config)# show mac address-table learning-mode
```

Enabling MAC Move Protection

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**mac address loop-detect flow-control-fe global-thresh-time** *threshold-time* **global-thresh-count** *threshold-count* **Example:** `switch(config)# mac address loop-detect flow-control-fe global-thresh-time 5 global-thresh-count 500`	Enables FE-based flow control to turn on the software throttle for mac-move protection for all FEs on all line cards.
Step 3	**mac address loop-detect flow-control-fe threshold-time** *threshold-time* **threshold-count** *threshold-count* **Example:** `switch(config)# mac address-table loop-detect flow-control-fe threshold-time 5 threshold-count 500`	Enables FE-based flow control to turn on the software throttle for mac-move protection for a specific FE (per ASIC level).
Step 4	**mac address loop-detect disable-learn-vlan global-thresh-time** *threshold-time* **global-thresh-count** *threshold-count* **Example:**	Disables the mac-learning for all VLANs (global throttle).

	Command or Action	Purpose
	`switch(config)# mac address loop-detect disable-learn-vlan global-thresh-time 5 global-thresh-count 500`	
Step 5	**mac address loop-detect disable-learn-vlan threshold-time** *threshold-time* **threshold-count** *threshold-count* **Example:** `switch(config)# mac address loop-detect disable-learn-vlan-thresh-time 5 thresh-count 500`	Disables the mac-learning per VLAN (per VLAN throttle).
Step 6	**mac address throttle-buffer-intv max** **Example:** `switch(config)# mac address throttle-buffer-intv max`	Uses maximum scan interval and buffer size to increase/decrease the throttle; and to effect maximum throttle.
Step 7	**mac address throttle-buffer-intv optimal** **Example:** `switch(config)# mac address throttle-buffer-intv optimal`	Uses optimal scan interval and buffer size to throttle at a medium level..

Step 8	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Checking the Consistency of MAC Address Tables

Beginning with Release 4.1(2). you can check the match between the MAC address table on the supervisor and all the modules.

Procedure

	Command or Action	Purpose

Step 1	**show forwarding consistency l2** {*module_number*}
	Example:
	`switch# show forwarding consistency l2 7`
	`switch#`
		Displays the discrepant, missing, and extra MAC addresses between the supervisor and the specified module.

Example

This example shows how to display discrepant, missing, and extra entries in the MAC address tables between the supervisor and the specified module:

```
switch# show forwarding consistency l2 7
switch#
```

Clearing Dynamic Addresses from the MAC Address Table

You can clear all dynamic Layer 2 entries in the MAC address table.

Before you begin

Before you clear the dynamic MAC address table, ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**clear mac address-table dynamic** {**address** *mac_addr*} {**interface** [**ethernet** *slot/port* | **loopback** *number* | **port-channel** *channel-number*]} {**vlan** *vlan_id*}
	Example:
	`switch# clear mac address-table dynamic`
		Clears the dynamic address entries from the MAC address table in Layer 2.

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

Example

This example shows how to clear the dynamic entries in the Layer 2 MAC address table:

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

Verifying the Layer 2 Switching Configuration

To display Layer 2 switching configuration information, perform one of the following tasks:

Command	Purpose
show mac address-table	Displays information about the MAC address table.
show mac address-table aging-time	Displays information about the aging time set for the MAC address entries.
show mac address-table static	Displays information about the static entries on the MAC address table.

Command	Purpose
show interface [*interface*] **mac-address**	Displays the MAC addresses and the burned in MAC addresses for the interfaces.
show forwarding consistency l2 {*module*}	Displays discrepant, missing, and extra MAC addresses between the tables on the module and the supervisor.

For information on the output of these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Configuration Example for Layer 2 Switching

The following example shows how to add a static MAC address and how to modify the default global aging time for MAC addresses:

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

Additional References for Layer 2 Switching

Related Documents

Related Topic	Document Title
Port security, static MAC addresses	*Cisco Nexus 7000 Series NX-OS Security Configuration Guide*
Interfaces	*Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
Command reference	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*
High availability	*Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
System management	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*
Licensing	*Cisco NX-OS Licensing Guide*
Release Notes	*Cisco Nexus 7000 Series NX-OS Release Notes*

Standards

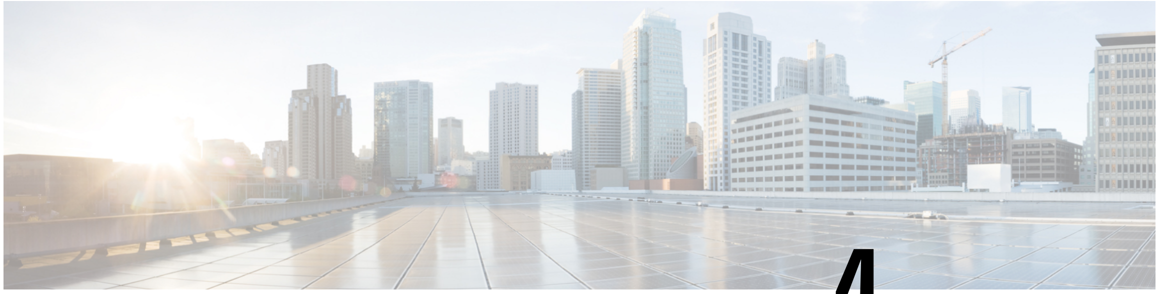
Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for Configuring Layer 2 Switching

This table lists the release history for this feature.

Table 3: Feature History for Configuring Layer 2 Switching

Feature Name	Releases	Feature Information	
MAC move protection	8.2(3)	MAC move protection using software throttle and hardware throttle is supported.	
Learning mode for VLANs	6.0(1)	You can configure conversational or nonconversational learning mode for VLANs.	
Layer 3 interface static MAC addresses	4.2(1)	You can configure a Layer 3 interface with a static MAC address.	
show mac address-table	4.1(2)	This display provides additional information when vPC is enabled and running.	
Layer 2 consistency	4.1(2)	The **show forwarding consistency l2** command displays inconsistent entries on the MAC address table between the modules.	



CHAPTER 4

Configuring VLANs

This chapter describes how to configure virtual LANs (VLANs) on Cisco NX-OS devices.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About VLANs

Note Beginning with Cisco Release 5.2(1) for Cisco Nexus 7000 Series devices, you can create Fibre Channel over Ethernet (FCoE) VLANs. For more information, see the *Cisco NX-OS FCoE Configuration Guide for Cisco Nexus 7000 and Cisco MDS 9500*.

You can use VLANs to divide the network into separate logical areas at the Layer 2 level. VLANs can also be considered as broadcast domains.

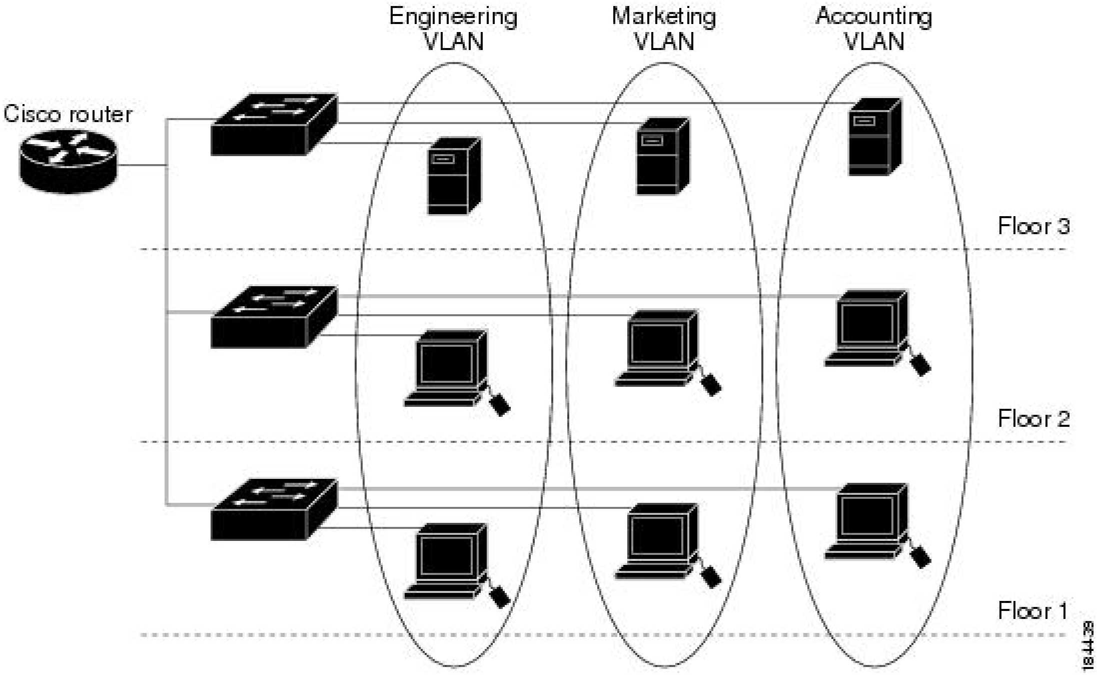
Any switch port can belong to a VLAN, and unicast broadcast and multicast packets are forwarded and flooded only to end stations in that VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router.

Understanding VLANs

A VLAN is a group of end stations in a switched network that is logically segmented by function 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 switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in that VLAN. Each VLAN is considered as a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router. The following figure shows VLANs as logical networks. The stations in the engineering department are assigned to one VLAN, the stations in the marketing department are assigned to another VLAN, and the stations in the accounting department are assigned to another VLAN.

Figure 2: VLANs as Logically Defined Networks



VLANs are usually associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. To communicate between VLANs, you must route the traffic.

By default, a newly created VLAN is operational; that is, the newly created VLAN is in the no shutdown condition. Additionally, you can configure VLANs to be in the active state, which is passing traffic, or the suspended state, in which the VLANs are not passing packets. By default, the VLANs are in the active state and pass traffic.

A VLAN interface, or switched virtual interface (SVI), is a Layer 3 interface that is created to provide communication between VLANs. In order to route traffic between VLANs, you must create and configure a VLAN interface for each VLAN. Each VLAN requires only one VLAN interface.

Note See the for complete information on configuring VLAN interfaces, and subinterfaces, as well as assigning IP addresses. This feature must be enabled before you can configure VLAN interfaces.

VLAN Ranges

Note The extended system ID is always automatically enabled in Cisco NX-OS devices.

The device supports up to 4094 VLANs in accordance with the IEEE 802.1Q standard in each VDC. The software organizes these VLANs into ranges, and you use each range slightly differently.

For information about configuration limits, see the verified scalability limits documentation for your switch.

This table describes the VLAN ranges.

Table 4: VLAN Ranges

VLANs Numbers	Range	Usage
1	Normal	Cisco default. You can use this VLAN, but you cannot modify or delete it.
2 to 1005	Normal	You can create, use, modify, and delete these VLANs.
1006 to 3967 and 4048 to 4093	Extended	You can create, name, and use these VLANs. You cannot change the following parameters: • The state is always active. • The VLAN is always enabled. You cannot shut down these VLANs.
3968 to 4047 and 4094	Internally allocated	These 80 VLANs and VLAN 4094 are allocated for internal device use. You cannot create, delete, or modify any VLANs within the block reserved for internal use.
3968 to 4095 **Note** 4095 is reserved and unused as per 802.1Q standard.	Internally allocated	Beginning with Cisco Release 5.2(1) for Cisco Nexus 7000 Series devices, VLANs 3968 to 4095 are reserved for internal use in each VDC by default. You can change the reserved VLANs to any other 128 contiguous VLAN range. When you reserve such a range, it frees up the range of VLANs that were allocated for internal use by default, and all of those VLANs are available for user configuration except for VLAN 4095. All VDCs inherit the new reserved range of VLANs. **Note** VLAN 0 is reserved for 802.1p trafffic.

The software allocates a group of VLAN numbers for features such as multicast and diagnostics that need to use internal VLANs for their operation. You cannot use, modify, or delete any of the VLANs in the reserved group. You can display the VLANs that are allocated internally and their associated use.

Beginning with Cisco NX-OS Release 5.2(1), the system allocates a block of 128 reserved VLANs (3968 to 4094) for these internal uses. You can change the block of 128 reserved VLANs to occupy another range of 128 adjacent VLANs. For example, you can change the reserved block of VLANs to be 400 to 528. You cannot assign a previously created VLAN as part of the 128 range of reserved VLANs. Anytime you change the reserved block of VLANs for the device, you must do the following:

- Enter the **copy running-configuration startup-configuration** command
- Reload the device

Note When you change the range of reserved VLANs, the existing configurations for the new range of VLANs get deleted. A warning note is displayed as in the following example:

```
switch(config)# system vlan 2000 reserve
This will delete all configs on vlans 2000-2127. Continue anyway? (y/n) [no] y
Note: After switch reload, VLANs 2000-2127 will be reserved for internal use.
      This requires copy running-config to startup-config before
      switch reload. Creating VLANs within this range is not allowed.
switch(config)#
```

To return to the default block of reserved VLANs (3968 to 4094), you must enter the **no system reserve vlan** command. The write-erase procedure does not restore the default reserved VLAN range to 3968 to 4094.

Creating, Deleting, and Modifying VLANs

Beginning with Cisco NX-OS Release 5.1(1) , you can configure a VLAN without actually creating the VLAN. This procedure is used for IGMP snooping, VTP, and other configurations.

Note By default, all Cisco NX-OS ports are Layer 3 ports.

VLANs are numbered from 1 to 4094 for each VDC. All ports that you have configured as switch ports belong to the default VLAN when you first bring up the switch as a Layer 2 device. The default VLAN (VLAN1) uses only default values, and you cannot create, delete, or suspend activity in the default VLAN.

You create a VLAN by assigning a number to it; you can delete VLANs and move them from the active operational state to the suspended operational state. If you attempt to create a VLAN with an existing VLAN ID, the device goes into the VLAN submode but does not create the same VLAN again.

Newly created VLANs remain unused until Layer 2 ports are assigned to the specific VLAN. All the ports are assigned to VLAN1 by default.

Depending on the range of the VLAN, you can configure the following parameters for VLANs (except the default VLAN):

- VLAN name
- VLAN state

- Shutdown or not shutdown

Beginning with Cisco NX-OS Release 6.1(1), you can configure VLAN long-names of up to 128 characters. To configure VLAN long-names, VTP must be in transparent or in off mode. If VTP is in client or server mode, the VLAN long-name feature cannot be enabled. For more details about VTP, see the Configuring VTP chapter.

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on configuring ports as VLAN access or trunk ports and assigning ports to VLANs.

When you delete a specified VLAN, the ports associated to that VLAN become inactive and no traffic flows. When you delete a specified VLAN from a trunk port, only that VLAN is shut down and traffic continues to flow on all the other VLANs through the trunk port.

However, the system retains all the VLAN-to-port mapping for that VLAN, and when you reenable or re-create, that specified VLAN, the system automatically reinstates all the original ports to that VLAN. The static MAC addresses and aging time for that VLAN are not restored when the VLAN is reenabled.

Note Before Cisco NX-OS Release 5.1, commands entered in the VLAN configuration submode are immediately executed. Beginning with Cisco Release NX-OS 5.1 for Nexus 7000 Series devices, you must exit the VLAN configuration submode for configuration changes to take effect.

High Availability for VLANs

The software supports high availability for both stateful and stateless restarts, as during a cold reboot, for VLANs. For the stateful restarts, the software supports a maximum of three retries. If you try more than 3 times within 10 seconds of a restart, the software reloads the supervisor module.

You can upgrade or downgrade the software seamlessly when you use VLANs.

Note See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on high availability features.

Virtualization Support for VLANs

The software supports virtual device contexts (VDCs), and VLAN configuration and operation are local to the VDC.

Note See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on VDCs and assigning resources.

Each VLAN must have all of its ports in the same VDC. If you do not have enough resources allocated to the VDC, the software returns an error message.

When you create a new VDC, the device automatically creates a new default VLAN, VLAN1, and internally reserves VLANs for device use.

You can reuse the same numbers for VLANs in different VDCs.

One or more VLANs can be associated with a role to either allow or disallow the user to configure it. When a VLAN is associated with a role, the corresponding interfaces will also be subjected to the same check. For instance, if a role is allowed to access VLAN1, that role also has access to the interfaces that have that VLAN. If an interface does not have the VLAN associated with a role, that interface is not accessible to that role.

Prerequisites for Configuring VLANs

VLANs have the following prerequisites:

- You must be logged onto the device.
- If necessary, install the Advanced Services license and enter the desired VDC. Ensure that you have allocated enough resources for that VDC. See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for information on creating VDCs and allocating resources.
- You must create the VLAN before you can do any modification of that VLAN.

Guidelines and Limitations for Configuring VLANs

VLANs have the following configuration guidelines and limitations:

- The maximum number of VLANs per VDC is 4094.
- You can configure a single VLAN or a range of VLANs.

  When you configure a large number of VLANs, first create the VLANs using the **vlan** command (for example, **vlan** *200 to 300*, *303 to 500*). After the VLANs have been successfully created, name or configure those VLANs sequentially.
- VLAN 4094 is a reserved VLAN.
- You cannot create, modify, or delete any VLANs that are within the group of VLANs reserved for internal use.
- VLAN1 is the default VLAN. You cannot create, modify, or delete this VLAN.
- VLANs 1006 to 4094 are always in the active state and are always enabled. You cannot suspend the state or shut down these VLANs.
- An interface policer and CoPP classification does not work for the Layer 2 control traffic in native VLAN in the following scenarios:
  - When the **native vlan** (ID other than 1) command is configured on the interface and the native VLAN ID is missing in the configuration.
  - If the **vlan dot1q tag native exclude control** command is configured.

VLAN translation has the following guidelines and limitations:

- A VLAN translation configuration is only applicable to Layer 2 trunks. It is inactive when applied to ports that are not Layer 2 trunks.
- Do not configure translation of ingress native VLAN traffic on an 802.1Q trunk. The 802.1Q native VLAN traffic is untagged and cannot be recognized for translation. However, you can translate traffic from other VLANs to the native VLAN of an 802.1Q trunk.
- The VLANs to which you are translating must be present in the trunk's allowed VLAN list. In addition, the VLANs that need to be forwarded on a trunk port, that are not involved in VLAN translation must also be included in the trunk ports allowed VLAN list. With per-port VLAN translation enabled, VLAN translation entries are consumed in hardware for all VLANs in the trunk ports allowed VLAN list.
- Do not change the VLAN on an access port or a trunk port it will flap the interface. However, if the port is part of a vPC, then first change the native VLAN on the secondary vPC, and then on the primary vPC.
- A VLAN translation must ensure that the original and translated VLANs are within the same MST instance.
- The VLAN translation configuration applies to all ports in a port group. VLAN translation is enabled by default on all ports.
- The number of supported VLAN translation maps is 4000. Layer 2 ports that have the same VLAN maps and the same trunk allowed VLAN list can benefit from sharing translation entries in hardware.
- The following limitations apply to the number of translation entries per port, based on the module type:
    - For F1 Series modules: Translation entries are limited to 512 entries on two ports, shared in the ingress and egress direction. The translation entries can be shared across the two ports for 256 entries per port.
    - For F2 Series modules: You can configure up to 2000 translations per port in each direction (ingress and egress).
    - For F3 Series modules: You can configure up to 2000 translations per port in each direction (ingress and egress).
    - For M1 Series modules: Translation entries are limited to eight per port.
    - For M1 Series modules: VLAN translations are supported only in the dedicated mode.
    - For M2 Series modules: You can configure up to 2000 translations per port.
    - For M3 Series modules: You can configure up to 2000 translations per port.

Default Settings for VLANs

This table lists the default settings for VLAN parameters.

Table 5: Default VLAN Parameters

Parameters	Default
VLANs	Enabled

Parameters	Default
VLAN	VLAN1—A port is placed in VLAN1 when you configure it as a switch port.
VLAN ID	1
VLAN name	• Default VLAN (VLAN1)—default • All other VLANs—VLAN *vlan-id*
VLAN state	Active
STP	Enabled; Rapid PVST+ is enabled
VTP	Disabled
VTP version	1

Configuring a VLAN

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on assigning Layer 2 interfaces to VLANs (access or trunk ports). All interfaces are in VLAN1 by default.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Creating and Deleting a VLAN

You can create or delete all VLANs except the default VLAN and those VLANs that are internally allocated for use by the device.

Once a VLAN is created, it is automatically in the active state.

Note When you delete a VLAN, ports associated to that VLAN become inactive. Therefore, no traffic flows and the packets are dropped. On trunk ports, the port remains open and the traffic from all other VLANs except the deleted VLAN continues to flow.

If you create a range of VLANs and some of these VLANs cannot be created, the software returns a message listing the failed VLANs, and all the other VLANs in the specified range are created.

Note You can also create and delete VLANs in the VLAN configuration submode.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vlan** {*vlan-id* | *vlan-range*}
	Example:
	`switch(config)# vlan 5`
	`switch(config-vlan)#`
		Creates a VLAN or a range or VLANs. If you enter a number that is already assigned to a VLAN, the device puts you into the VLAN configuration submode for that VLAN. If you enter a number that is assigned to an internally allocated VLAN, the system returns an error message. However, if you enter a range of VLANs and one or more of the specified VLANs is outside the range of internally allocated VLANs, the command takes effect on only those VLANs outside the range. The range is from 2 to 4094; VLAN1 is the default VLAN and cannot be created or deleted. You cannot create or delete those VLANs that are reserved for internal use.

Step 3	**exit**
	Example:
	`switch(config-vlan)# exit`
	`switch(config)#`
		Exits the VLAN mode.

Step 4	(Optional) **show vlan** **Example:** `switch# show vlan`	Displays information about the VLANs.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to create a range of VLANs from 15 to 20:

```
switch# config t
switch(config)# vlan 15-20
switch(config-vlan)# exit
switch(config)#
```

Entering the VLAN Configuration Submode

To configure or modify the VLAN for the following parameters, you must be in the VLAN configuration submode:

- Name
- State
- Shut down

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vlan** {*vlan-id* | *vlan-range*}
	Example:
	`switch(config)# vlan 5`
	`switch(config-vlan)#`
		Places you into VLAN configuration submode. This submode allows you to name, set the state, disable, and shut down the VLAN or range of VLANs.
		You cannot change any of these values for VLAN1 or the internally allocated VLANs.

Step 3	**exit**
	Example:
	`switch(config-vlan)# exit`
	`switch(config)#`
		Exits VLAN configuration mode.

Step 4	(Optional) **show vlan** **Example:** `switch# show vlan`	Displays information and status of VLANs.

	Command or Action	Purpose
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to enter and exit VLAN configuration submode:

```
switch# config t
switch(config)# vlan 15
switch(config-vlan)# exit
switch(config)#
```

Configuring a VLAN

To configure or modify a VLAN for the following parameters, you must be in VLAN configuration submode:

- Name
- State
- Shut down

Note You cannot create, delete, or modify the default VLAN or the internally allocated VLANs. Additionally, some of these parameters cannot be modified on some VLANs.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). VLAN names and IDs can be repeated in different VDCs, so you must confirm which VDC that you are working in.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vlan** {*vlan-id* | *vlan-range*}
	Example:
	`switch(config)# vlan 5`
	`switch(config-vlan)#`
		Places you into VLAN configuration submode. If the VLAN does not exist, the system creates the specified VLAN and then enters the VLAN configuration submode.

	Command or Action	Purpose
Step 3	**name** *vlan-name* **Example:** `switch(config-vlan)# name accounting`	Names the VLAN. You can enter up to 32 alphanumeric characters to name the VLAN. You cannot change the name of VLAN1 or the internally allocated VLANs. The default value is VLANxxxx where xxxx represents four numeric digits (including leading zeroes) equal to the VLAN ID number. The **system vlan long-name** command allows you to enable VLAN names that have up to 128 characters.

Step 4	**state** {**active** | **suspend**}
	Example:
	`switch(config-vlan)# state active`
		Sets the state of the VLAN to active or suspend. While the VLAN state is suspended, the ports associated with this VLAN become inactive, and that VLAN does not pass any traffic. The default state is active. You cannot suspend the state for the default VLAN or VLANs 1006 to 4094.

Step 5	**no shutdown** **Example:** `switch(config-vlan)# no shutdown`	Enables the VLAN. The default value is no shutdown (or enabled). You cannot shut down the default VLAN, VLAN1, or VLANs 1006 to 4094.

Step 6	**exit**
	Example:
	`switch(config-vlan)# exit`
	`switch(config)#`
		Exits VLAN configuration submode.

Step 7	(Optional) **show vlan** **Example:** `switch# show vlan`	Displays information about the VLANs.
Step 8	(Optional) **show vtp status** **Example:** `switch# show vtp status`	Displays information about the VLAN Trunking Protocol (VTP).
Step 9	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration. **Note** Commands entered in VLAN configuration submode are immediately executed. Beginning with Cisco Release 5.1 for Nexus 7000 series devices, you must exit the VLAN configuration submode for configuration changes to take effect.

Example

This example shows how to configure optional parameters for VLAN 5:

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

Changing the Range of Reserved VLANs

To change the range of reserved VLANs, you must be in global configuration mode. After entering this command, you must do the following tasks:

- Enter the **copy running-config startup-config** command
- Reload the device

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**system vlan** *start-vlan* **reserve** **Example:** `switch(config)# system vlan 3968 reserve`	Allows you to change the reserved VLAN range by specifying the starting VLAN ID for your desired range. You can change the reserved VLANs to any other 128 contiguous VLAN ranges. When you reserve such a range, it frees up the range of VLANs that were allocated for internal use by default, and all of those VLANs are available for user configuration except for VLAN 4094. All VDCs inherit the new reserved range of VLANs. **Note** To return to the default range of reserved VLANs (3968-4049 and 4094), you must enter the **no system vlan** *start-vlan* **reserve** command.
Step 3	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration. **Note** You must enter this command if you change the reserved block.

	Command or Action	Purpose
Step 4	**reload** **Example:** `switch(config)# reload`	Reloads the software, and modifications to VLAN ranges become effective. For more details about this command, see the *Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide, Release 6.x.*
Step 5	(Optional) **show system vlan reserved** **Example:** `switch(config)# show system vlan reserved`	Displays the configured changes to the VLAN range.

Example

This example shows how to change the range of reserved VLANs:

```
switch# configuration terminal
switch(config)# system vlan 2000 reserve
This will delete all configs on vlans 2000-2081. Continue anyway? (y/n) [no] y
Note: After switch reload, VLANs 2000-2081 will be reserved for internal use.
      This requires copy running-config to startup-config before
      switch reload. Creating VLANs within this range is not allowed.
switch(config)#
```

Note You must reload the device for this change to take effect.

Configuring a VLAN Before Creating the VLAN

Beginning with Cisco NX-OS Release 5.1(1), you can configure a VLAN before you create the VLAN. This procedure is used for IGMP snooping, VTP, and other configurations.

Note The **show vlan** command does not display these VLANs unless you create the VLANs using the **vlan** command.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vlan configuration** {*vlan-id*} **Example:**	Allows you to configure VLANs without actually creating them.

	Command or Action	Purpose
	switch(config)# vlan configuration 20 switch(config-vlan-config)#	

Example

This example shows how to configure a VLAN before creating it:

```
switch# config t
switch(config)# vlan configuration 20
switch(config-vlan-config)#
```

Configuring VLAN Long-Name

Note If VTP is enabled, it must be in transparent or in off mode. VTP cannot be in client or server mode. For more details about VTP, see the Configuring VTP chapter.

Procedure

Step 1 **configure terminal**

Example:

```
switch# configure terminal
```

Enters global configuration mode.

Step 2 **system vlan long-name**

Example:

```
switch(config)# system vlan long-name
```

Allows you to configure the length of VLAN names up to 128 characters.

Note Enabling or disabling the **system vlan long-name** command will trigger a system log message that will let you know if the VLAN long name is enabled or disabled.

If you try to enable or disable the **system vlan long-name** command, when it is already enabled or disabled, the system will throw error message. We recommend you view the status of the VLAN long-name knob before enabling or disabling this command.

Use the **no** form of this command to disable this feature.

Step 3 (Optional) **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Step 4 **show running-config | sec long-name**

Example:

```
switch(config)# show running-config | sec long-name
```

Displays the VLAN long-name status information.

Note When you configure a VLAN name of more than 32 characters, the **show vlan** commands will show the output in mulitple lines with each line containing a maximum of 32 characters.

Example

This example shows how to configure VLAN long-names of up to 128 characters.

```
switch# configure terminal
switch(config)# system vlan long-name
!2001 Sep 29 02:24:11 N72-3 %$ VDC-1 %$ %VLAN_MGR-2-CRITICAL_MSG: VLAN long name is Enabled!
switch(config)# copy running config startup config
switch(config)# show running-config | sec long-name
switch# configure terminal
switch(config)# vlan 2
switch(config-vlan)# name
VLAN128Char0000000000000000400000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000002

switch(config-vlan)# exit
switch# show vlan id 2

VLAN Name Status Ports
---- -------------------------------- --------- -------------------------------
2 VLAN128Char000000000000000040000 active
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000002
.
.
.
```

The following example displays the error output if you try to configure a VLAN long name of more than 128 characters.

```
switch# system vlan long-name
switch(config)# vlan 2
switch(config-vlan)# name
129Char123456789000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000987654321CiscoBangalore

!% String exceeded max length of (128) at '^' marker.!
Switch(config-vlan)# exit
```

The following example displays the error output if you try to configure VLAN name (more than 32 characters) without enabling the **system vlan long- name** command.

```
switch# configure terminal
switch(config)# vlan 2
switch(config-vlan)# name 33Char1234567890987CiscoBangalore
!ERROR: Long VLAN name is not enabled: Vlan name greater than 32 is not allowed!
Switch(config-vlan)# exit
```

Configuring VLAN Translation on a Trunk Port

You can configure VLAN translation between the ingress VLAN and a local VLAN on a port. The traffic arriving on the ingress VLAN maps to the local VLAN at the ingress of the trunk port and the traffic that is internally tagged with the translated VLAN ID is mapped back to the original VLAN ID before leaving the switch port.

Before you begin

- Ensure that the physical or port channel on which you want to implement VLAN translation is configured as a Layer 2 trunk port.
- Ensure that the translated VLANs are created on the switch and are also added to the Layer 2 trunk ports trunk-allowed VLAN vlan-list.
- For FEX port-channel trunk interfaces, the last VLAN in the allowed VLAN list must be associated with a translated VLAN in one of the VLAN maps configured on the FEX fabric interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type port*	Enters interface configuration mode.
Step 3	(Optional) switch(config-if)# [**no**] **switchport vlan mapping enable**	Enables VLAN translation on the switch port after VLAN translation is explicitly disabled. VLAN translation is enabled by default. **Note** Use the **no** form of this command to disable VLAN translation.
Step 4	switch(config-if)# [**no**] **switchport vlan mapping** *vlan-id translated-vlan-id*	Translates a VLAN to another VLAN. • The range for both the *vlan-id* and *translated-vlan-id* arguments is from 1 to 4094. • When you configure a VLAN mapping between a VLAN and a (local) VLAN on a port, traffic arriving on the VLAN gets mapped or translated to the local VLAN at the ingress of the switch port, and the traffic internally tagged with the translated VLAN ID gets mapped to the original VLAN ID before leaving the switch port. This method of VLAN mapping is a two-way mapping. **Note** Use the **no** form of this command to clear the mappings between a pair of VLANs.

	Command or Action	Purpose
Step 5	switch(config-if)# [**no**] **switchport vlan translation all**	Removes all VLAN translations configured on the interface.
Step 6	(Optional) switch(config-if)# **copy running-config startup-config**	Copies the running configuration to the startup configuration. **Note** The VLAN translation configuration does not become effective until the switch port becomes an operational trunk port
Step 7	(Optional) switch(config-if)# **show interface** [*if-identifier*] **vlan mapping**	Displays VLAN mapping information for all interfaces or for the specified interface.

Example

This example shows how to configure VLAN translation between (the ingress) VLAN 10 and (the local) VLAN 100:

```
switch# config t
switch(config)# interface ethernet1/1
switch(config-if)# switchport vlan mapping 10 100
switch(config-if)# show interface ethernet1/1 vlan mapping
```

```
Interface eth1/1:
Original VLAN          Translated VLAN
------------------     ---------------
10                          100
```

Verifying the VLAN Configuration

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

Command	Purpose
show running-config vlan *vlan-id*	Displays VLAN information.
show vlan [**all-ports** | **brief** | **id** *vlan-id* | **name** *name* | **dot1q tag native**]	Displays VLAN information.
show vlan summary	Displays a summary of VLAN information.
show vtp status	Displays VTP information.
show system vlan reserved	Displays system reserved VLAN range.

For information on the output of these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Displaying and Clearing VLAN Statistics

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

Command	Purpose
clear vlan [**id** *vlan-id*] **counters**	Clears counters for all VLANs or for a specified VLAN.
show vlan counters	Displays information on Layer 2 packets in each VLAN.

Configuration Example for VLANs

The following example shows how to create and name a VLAN as well as how to make the state active and administratively up:

```
switch# configure terminal
switch(config)# vlan 10
switch(config-vlan)# name test
switch(config-vlan)# state active
switch(config-vlan)# no shutdown
switch(config-vlan)# exit
switch(config)#
```

Additional References for VLANs

Related Documents

Related Topic	Document Title
Command reference	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*
NX-OS Layer 2 switching configuration	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Configuration Guide*
Interfaces, VLAN interfaces, IP addressing, and port channels	*Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
Multicast routing	*Cisco Nexus 7000 Series NX-OS Multicast Routing Configuration Guide*
NX-OS fundamentals	*Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide*

Related Topic	Document Title
High availability	*Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
System management	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
Licensing	*Cisco NX-OS Licensing Guide*
Release notes	*Cisco Nexus 7000 Series NX-OS Release Notes*

Standards

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

MIBs

MIBs	MIBs Link
CISCO-VLAN-MEMBERSHIP MIB: • vmMembership Table • MIBvmMembershipSummaryTable • MIBvmMembershipSummaryTable	To locate and download MIBs, go to the following URL: https://cfnng.cisco.com/mibs.

Feature History for Configuring VLANs

This table lists the release history for this feature.

Table 6: Feature History for Configuring VLANs

Feature Name	Releases	Feature Information
VLAN translation	6.2(6)	You can configure mapping between a pair of VLANs.
Configure VLAN long-name.	6.1(1)	You can configure VLAN long-names.

Feature Name	Releases	Feature Information
Dynamic system reserved VLAN range	5.2(1)	You can change the range of the system reserve VLANs.
Configure VLAN before creating the VLAN	5.1(1)	You can configure a VLAN before creating the VLAN.
No change	4.2(1)	--
VLAN Trunking Protocol	4.1(2)	The device now runs VTP in transparent mode.

CHAPTER 5

Configuring MVRP

This chapter describes how to configure Layer 2 switching using IEEE 802.1ak Multiple VLAN Registration Protocol (MVRP).

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About MVRP

Multiple VLAN Registration Protocol (MVRP) is an IEEE 802.1ak Multiple Registration Protocol (MRP) application that supports dynamic registration and deregistration of VLANs on ports in a VLAN-bridged network.

MRP allows participants in a MRP application to register attributes with other participants in a bridged local area network (LAN).

MVRP registers VLANs and enables a VLAN bridge to restrict unknown unicast, multicast, and broadcast traffic to those links that the traffic uses to access the appropriate network devices.

The IEEE 802.1ak MRP provides improved resource utilization and bandwidth conservation. With the 802.1ak MRP attribute encoding scheme, MVRP sends only one protocol data unit (PDU) that includes the state of all 4094 VLANs on a port.

Guidelines and Limitations for Configuring MVRP

- MVRP is supported only on Layer 2 ports. MVRP is not supported on sub interfaces.
- MVRP is supported only on IEEE.802.1Q (dot1q) port channel Layer 2 ports.
- MVRP must be enabled on both sides of the trunk. In vPC topologies, the MRVP configuration on the vPC legs must be identical for both peers.
- When MVRP is disabled on the Cisco Nexus device, all ingress MVRP PDUs remain unprocessed and are flooded to other ports like multicast data frames.
- MVRP and VLAN Trunk Protocol (VTP) can coexist on the same interface. You can use MVRP to perform VLAN pruning and VTP to manage VLANs, such as adding or deleting VLANs.
- MVRP pruning and VTP pruning are mutually exclusive. VTP pruning can run only on ports where MVRP is disabled. If you enable both MVRP and VTP pruning on the interfaces, MVRP pruning takes precedence.
- MVRP and Private VLANs (PVLANs) are mutually exclusive.
- Auto detection of MAC addresses is not supported.
- MVRP dynamic VLAN creation is not supported.
- FabricPath is not supported.
- Management Information Base (MIB) is not supported.
- Interaction between MVRP and the following features is not supported:
    - Fabric Extender (FEX)
    - Overlay Transport Virtualization (OTV)
    - VLAN translation

Default Settings for MVRP

The following table lists the default settings for MVRP on Cisco NX-OS devices.

Parameter	Default
MVRP	Disabled

Configuring MVRP

Enabling MVRP

You can enable MVRP on all trunk ports on an interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type number*	Configure an interface and enters interface configuration mode. The range for the *number* argument is from 1 to 253.
Step 3	switch(config-if)# **feature mvrp**	Enables MVRP on all trunk ports. If MVRP is not successfully enabled on the port, the port is put in the errdisabled state. Enter the **shutdown** and **no shutdown** commands to clear the errdisabled state. **Note** You must use the **no mrvp** command to explicitly disable MVRP on trunk ports that are connected to devices that do not support MVRP .

Modifying the MVRP Configuration on the Interface

You can perform this task to set the MVRP registrar state and configure MVRP timer values.

Before you begin

- Ensure that MVRP is enabled on the interface to be configured.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **interface** *type number*	Configure an interface and enters interface configuration mode. The range for the *number* argument is from 1 to 253.

	Command or Action	Purpose
Step 3	(Optional) switch(config-if)# **mvrp registration** {**normal** | **fixed** | **forbidden**}	Configure an interface and enters interface configuration mode. Sets the registrars in a Multiple Registration Protocol (MRP) Attribute Declaration (MAD) instance associated with an interface. • Use the **normal** keyword to specify that the registrar respond normally to incoming MVRP messages. Normal is the default registrar state. • Use the **fixed** keyword to specify that the registrar ignore all incoming MVRP messages (remain in the IN state). VLANs are not pruned. • Use the **forbidden** keyword to specify that the registrar ignore all incoming MVRP messages (remain in the EMPTY (MT) state) and prune VLANs. **Note** You can use the **no mvrp registration** command to return the registrar to the default value (normal).
Step 4	(Optional) switch(config-if)# **mvrp timer** {{**join** | **leave** | **join-leave**} *timer-value* | **periodic**}	Sets the period timers that are used in MVRP on a given interface. • Use the **join** keyword to specify the time interval between two transmit opportunities that are applied to the Applicant State Machine (ASMs). The range is from 20 to 1,000,000 centiseconds. The default value is 20. • Use the **leave** keyword to specify the duration time before a registrar is moved to EMPTY (MT) state from leave-all (LV) state. The range is from 60 to 1,000,000 centiseconds. The default is 60. • Use the **join-leave** keyword to specify the time it takes for a LeaveAll timer to expire . The range is from 10,000 to 1,000,000 centiseconds. The default is 10,000. • Use the **periodic** keyword to set the timer value to a fixed value of 100 centiseconds.

	Command or Action	Purpose
		Note You can use the **no mvrp timer** command to remove the configured timer values.

Example

Verifying the MVRP Configuration

To display MVRP information, perform one of the following tasks:

Command	Purpose
show mvrp interface	Displays MVRP interface details of the administrative and operational states for all trunk ports in a device.
show mvrp interface *type number*	Displays MVRP interface details of the administrative and operational states for the specified interface in a device.
show mvrp interface *type number* **statistic**	Displays MVRP statistics for the specified interface in a device.
show mvrp pruning interface *type number*	Displays the pruned VLANs for the specified interface.

For detailed information about the fields in the output from these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Clearing MVRP Statistics

You can clear collected statistics on one or all MVRP-enabled ports.

Procedure

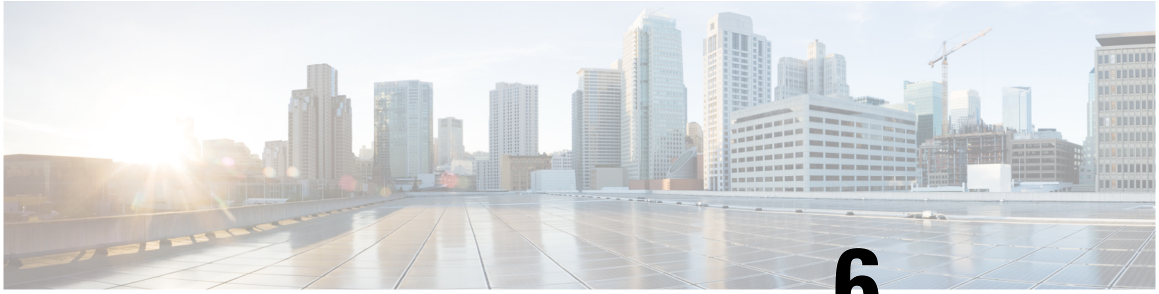
	Command or Action	Purpose
Step 1	switch# **clear mvrp statistics** [**interface** *type number*]	Clears collected statistics for MVRP-enabled devices or interfaces. If used without the interface keyword, the command clears all MVRP statistics on the device.

Feature History for Configuring MVRP

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Table 7: Feature History for Configuring MVRP

Feature Name	Releases	Feature Information
MVRP	6.2(6)	This feature was introduced.



CHAPTER 6

Configuring VTP

This chapter describes how to configure VLAN Trunking Protocol (VTP) and VTP pruning on Cisco NX-OS devices.

This chapter includes the following sections:

- Finding Feature Information, on page 53
- Information About VTP, on page 53
- Restrictions for Configuring the VLAN Trunking Protocol, on page 56
- Default Settings, on page 56
- Configuring VTP, on page 57
- Configuring VTP Pruning, on page 60

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About VTP

Beginning with Cisco NX-OS Release 5.1(1), VTP and VTP pruning are supported for VTP version 1 and 2. Before Release 5.1(1), only VTP transparent mode was supported.

Note Beginning with Cisco NX-OS Release 5.1(1), you can configure VLANs without actually creating the VLANs. For more details, see Configuring a VLAN Before Creating the VLAN, on page 38.

VTP

VTP is a Layer 2 messaging protocol that maintains VLAN consistency by managing the addition, deletion, and renaming of VLANs within a VTP domain. A VTP domain is made up of one or more network devices that share the same VTP domain name and that are connected with trunk interfaces. Each network device can be in only one VTP domain.

Layer 2 trunk interfaces, Layer 2 port channels, and virtual port channels (vPCs) support VTP functionality.

The VTP is disabled by default on the device. You can enable and configure VTP using the command-line interface (CLI). When VTP is disabled, the device does not relay any VTP protocol packets.

Note Before Release 5.1(1), VTP worked only in transparent mode in the Cisco Nexus 7000 Series devices, allowing you to extend a VTP domain across the device.

When the device is in the VTP transparent mode, the device relays all VTP protocol packets that it receives on a trunk port to all other trunk ports. When you create or modify a VLAN that is in VTP transparent mode, those VLAN changes affect only the local device. A VTP transparent network device does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements.

Note VLAN 1 is required on all trunk ports used for switch interconnects if VTP is supported in the network. Before 7.2(0)D1(1), disabling VLAN 1 from any of these ports prevents VTP from functioning properly.

If you enable VTP, you must configure either version 1 or version 2. If you are using VTP in a Token Ring environment, you must use version 2.

VTP Overview

VTP allows each router or LAN device to transmit advertisements in frames on its trunk ports. These frames are sent to a multicast address where they can be received by all neighboring devices. They are not forwarded by normal bridging procedures. An advertisement lists the sending device's VTP management domain, its configuration revision number, the VLANs which it knows about, and certain parameters for each known VLAN. By hearing these advertisements, all devices in the same management domain learn about any new VLANs that are configured in the transmitting device. This process allows you to create and configure a new VLAN only on one device in the management domain, and then that information is automatically learned by all the other devices in the same management domain.

Once a device learns about a VLAN, the device receives all frames on that VLAN from any trunk port by default, and if appropriate, forwards them to each of its other trunk ports, if any. This process prevents unnecessary VLAN traffic from being sent to a device. An extension of VTP called VTP pruning has been defined to limit the scope of broadcast traffic and save bandwidth. Beginning with Release 5.1(1), the Cisco NX-OS software supports VTP pruning.

VTP also publishes information about the domain and the mode in a shared local database that can be read by other processes such as Cisco Discovery Protocol (CDP).

VTP Modes

Beginning with Release 5.1(1), VTP is supported in these modes:

- Transparent—Allows you to relay all VTP protocol packets that it receives on a trunk port to all other trunk ports. When you create or modify a VLAN that is in VTP transparent mode, those VLAN changes affect only the local device. A VTP transparent network device does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. You cannot configure VLANs 1002 to 1005 in VTP client/server mode because these VLANs are reserved for Token Ring.

- Server— Allows you to create, remove, and modify VLANs over the entire network. You can set other configuration options like the VTP version and also turn on or off VTP pruning for the entire VTP domain. VTP servers advertise their VLAN configuration to other switches in the same VTP domain and synchronize their VLAN configuration with other switches based on messages received over trunk links. Beginning with Release 5.1(1), the server mode is the default mode. The VLAN information is stored on the bootflash and is not erased after a reboot.
- Client—VTP clients behave the same way as VTP servers, but you cannot create, change, or delete VLANs on a VTP client.
- Off— Behaves similarly to the transparent mode but does not forward any VTP packets. The off mode allows you to monitor VLANs by using the CISCO-VTP-MIB without having to run VTP. On Cisco Nexus 7000 Series devices, because VTP is a conditional service, its MIB is loaded only when the corresponding feature is enabled. The CISCO-VTP-MIB does not follow this convention. It is loaded by the VLAN manager and will always return the correct values whether the VTP process is enabled or disabled.

Beginning with Cisco NX-OS Release 6.1(1), if VTP is in transparent or in off mode, you can configure VLAN long names of up to 128 characters. If VTP is in client or server mode, you cannot enable VLAN long-names. For more details, see the Configuring VLANs chapter.

VTP Per Interface

VTP allows you to enable or disable the VTP protocol on a per-port basis to control the VTP traffic. When a trunk is connected to a switch or end device, it drops incoming VTP packets and prevents VTP advertisements on this particular trunk. By default, VTP is enabled on all the switch ports.

VTP Pruning

The VLAN architecture requires all flooded traffic for a VLAN to be sent across a trunk port even if it leads to switches that have no devices that are active in the VLAN. This method leads to wasted network bandwidth.

VTP pruning optimizes the usage of network bandwidth by restricting the flooded traffic to only those trunk ports that can reach all the active network devices. When this protocol is in use, a trunk port does not receive the flooded traffic that is meant for a certain VLAN unless an appropriate join message is received.

A join message is defined as a new message type in addition to the ones already supported by version 1 of VTP. A VTP implementation indicates that it supports this extension by appending a special TLV at the end of the summary advertisement messages that it generates. In VTP transparent mode, VTP relays all VTP packets, and pruning requires that the switch processes TLVs in the VTP summary packets. You cannot use pruning in VTP transparent mode.

VTP Pruning and Spanning Tree Protocol

VTP maintains a list of trunk ports in the Spanning Tree Protocol (STP) forwarding state by querying STP at bootup and listening to the notifications that are generated by STP.

VTP sets a trunk port into the pruned or joined state by interacting with STP. STP notifies VTP when a trunk port goes to the blocking or forwarding state. VTP notifies STP when a trunk port becomes pruned or joined.

VTPv3

VTP Version 3 (VTPv3) was introduced in Cisco NX-OS release 7.2(0) and has the following features:

- Provides interoperability with switches configured with VTP version 1 or 2.
- Allows only the primary server to make VTP configuration changes.
- Supports 4K VLANs.
- Permits feature-specific primary servers. A switch can be a primary server for a specific feature database like MST or for the entire VLAN database.
- Provides enhanced security with hidden and secret passwords.
- Provides interoperability with private VLANs (PVLAN). PVLANs and VTPs are no longer mutually exclusive.
- Resolves the issue of VTP bombing. VTP bombing occurs when a server with a higher revision number and a wrong VTP database is inserted into the VTP domain. This may occur when a new switch is plugged into a stable VTP domain. The incorrect database is propagated to the domain and the earlier stable database is overwritten.

Restrictions for Configuring the VLAN Trunking Protocol

- You cannot manually configure VLANs on a device configured as a VTP client.
- PVLAN forward referencing is not supported with VTP.
- Administered VLANs are not displayed in the show output for all the VTP versions in the VTP client and server mode.

VTP Version 3 has the following software restrictions:

- On the Cisco Nexus 7000 Series switches, by default, the VTP version 3 is configured under the server mode. Whereas on the Cisco Nexus 9000 Series switches, by default, the VTP version 3 is configured under the transparent mode.

Default Settings

This table lists the default settings for VTP parameters.

Table 8: Default VTP Parameters

Parameters	Default
VTP	Disabled
VTP Mode	Transparent
VTP Domain	blank

Parameters	Default
VTP Version	1
VTP Pruning	Disabled
VTP per Interface	Enabled

Configuring VTP

You can configure VTP on Cisco NX-OS devices.

Note VLAN 1 is required on all trunk ports used for switch interconnects if VTP is used in transparent mode in the network. Disabling VLAN 1 from any of these ports prevents VTP from functioning properly in transparent mode.

Note Before Cisco NX-OS Release 5.1(1), VTP worked only in transparent mode. With Cisco NX-OS Release 7.2(0), VTP version 3 was introduced.

Before you begin

Ensure that you are in the correct virtual device context (VDC) (or enter the **switchto vdc** command). VLAN names and IDs can be repeated in different VDCs, so you must confirm which VDC that you are working in.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **feature vtp**	Enables VTP on the device. The default is disabled.
Step 3	switch(config)# **vtp domain** *domain-name*	Specifies the name of the VTP domain that you want this device to join. The default is blank.
Step 4	switch(config)# **vtp version** {**1** | **2** | **3**}	Sets the VTP version that you want to use. The default is version 1. **Note** Version 3 is applicable for VTPv3 only and requires the configuration of VTP domain.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 5</b></td><td>Required: switch(config)# <b>vtp mode</b> {<b>client</b> | <b>server</b> | <b>transparent</b> | <b>off</b>} [<b>vlan</b> | <b>mst</b> | <b>unknown</b>]</td><td>Sets the VTP mode to client, server, transparent, or off. The default server mode is for vlan instance and transparent is for mst instance.</td></tr>
<tr><td><b>Step 6</b></td><td>switch(config)# <b>vtp interface</b> <i>interface-name</i> [<b>only</b>]</td><td>Configures the interface name used by the VTP updater for this device.</td></tr>
<tr><td><b>Step 7</b></td><td>switch(config)# <b>vtp file</b> <i>file-name</i></td><td>Specifies the ASCII filename of the IFS file system file where the VTP configuration is stored.</td></tr>
<tr><td><b>Step 8</b></td><td>switch(config)# <b>vtp password</b> <i>password-value</i> [<b>hidden</b> | <b>secret</b>]
<p><b>Example:</b></p>
<p>For Hidden:</p>
<pre>
Device(config)# vtp password helping
hidden

Generating the secret associated to the
 password.
Device# exit
Device# show vtp password
VTP Password:
89914640C8D90868B6A0D8103847A733
</pre>
<p><b>Example:</b></p>
<p>For Secret:</p>
<pre>
Device(config)# vtp password
89914640C8D90868B6A0D8103847A733 secret
Device# exit
Device# show vtp password
VTP Password:
89914640C8D90868B6A0D8103847A733
</pre></td><td>Specifies the password for the VTP administrative domain. Default value is taken from vlan.dat.
<p>The following options are applicable only on an image supporting VTP version 3:</p>
<ul>
<li>Hidden–Password is not saved as clear text in vlan.data file. Instead, a hexadecimal secret key generated from the password is saved. This is displayed as the output of the <b>show vtp password</b>.</li>
<li>Secret–Use this keyword to directly configure the 32-character hexadecimalsecret key. System administrators can distribute this secret key instead of the clear text password.</li>
</ul>
<p><b>Note</b> This command is applicable for VTP version 3 only.</p></td></tr>
<tr><td><b>Step 9</b></td><td>switch(config)# <b>exit</b></td><td>Exits the configuration submode.</td></tr>
<tr><td><b>Step 10</b></td><td>switch# <b>vtp primary</b> [<i>feature</i>] [<b>force</b>]
<p><b>Example:</b></p>
<pre>
Device# vtp primary vlan

Enter VTP password:
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
</pre></td><td>This command changes the operational state of a secondary server to primary and advertises the information to the entire VTP domain. If the password is configured as hidden, the user is prompted to re-enter the password after this command.
<p>Before the device takes over the role of primary, it attempts to discover servers that conflict this information and follows another primary server. If conflicting servers are discovered, the user must reconfirm the takeover of operational state and the subsequent overwriting of configuration.</p></td></tr>
</table>

	Command or Action	Purpose

	`stp7`
	
	`Do you want to continue (y/n) [n]? y`
		• feature–Configures the device as primary server for a specific feature database. For example, the MST database. Possible values are MST and VLAN. By default, the VLAN database is chosen.
		
		Note This command is applicable for VTPv3 only.

Step 11	(Optional) switch# **show vtp status**	Displays information about the VTP configuration on the device, such as the version, mode, and revision number.
Step 12	(Optional) switch# **show vtp counters**	Displays information about VTP advertisement statistics on the device.
Step 13	(Optional) switch# **show vtp interface**	Displays the list of VTP-enabled interfaces.
Step 14	(Optional) switch# **show vtp password**	Displays the password for the management VTP domain.

Step 15	(Optional) switch# **show vtp devices [conflict]**
	
	Example:
	`Device# show vtp devices`
	
	`Gathering information from the domain,`
	` please wait.`
	`VTP Database Conf switch ID      Primary`
	` Server Revision   System Name`
	`              lict`
	`------------ ---- --------------`
	`-------------- ----------`
	`----------------------`
	`VLAN         Yes  00b0.8e50.d000`
	`000c.0412.6300 12354      main.cisco.com`
	`MST          No   00b0.8e50.d000`
	`0004.AB45.6000 24        main.cisco.com`
	`VLAN         Yes`
	`000c.0412.6300=000c.0412.6300 67`
	`   qwerty.cisco.com`
		This is a VTP version 3 command that displays information about neighbor switches. The information is not learned from the summary packet used for regular VTP packets. This command sends out a separate packet to collect information regarding neighbor switches running VTP version 3.

Step 16	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to configure VTP in transparent mode for the device:

```
switch# configure terminal
switch(config)# feature vtp
switch(config)# vtp domain accounting
switch(config)# vtp version 2
switch(config)# vtp mode transparent
switch(config)# exit
switch#
```

Configuring VTP Pruning

You can configure VTP pruning on the Cisco NX-OS devices.

Before you begin

You must enable VTP on the device.

Procedure

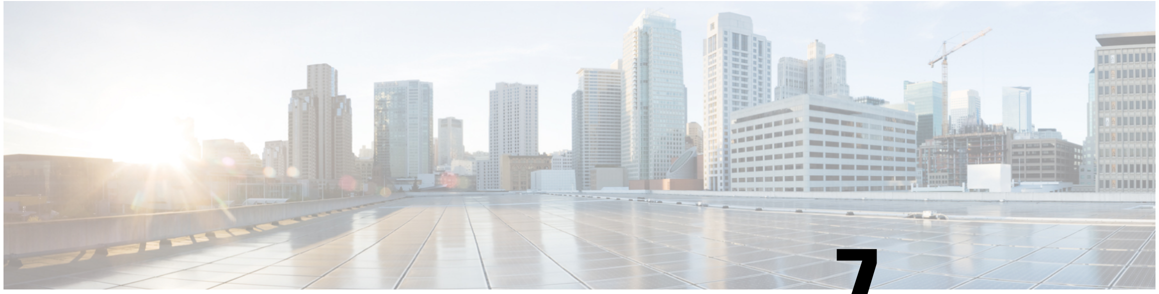
	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vtp pruning**	Enables VTP pruning on the device. The default is disabled.
Step 3	(Optional) switch(config)# **no vtp pruning**	Disables VTP pruning on the device. The default is disabled.
Step 4	(Optional) switch(config)# **show interface** *interface-identifier* **switchport**	Displays the VTP pruning eligibility of the trunk port. The default is that all the VLANs from 2 to 1001 are pruning eligible.
Step 5	switch(config)# **interface port-channel** *channel-number*	Creates a port-channel interface and enter interface configuration mode.
Step 6	Required: switch(config-if)# **switchport trunk pruning vlan** [**add** | **remove** | **except** | **none** | **all**] *VLAN-IDs*	Sets the specified VLANs to be VTP pruning eligible.
Step 7	switch(config-if)# **end**	returns to privileged EXEC mode.
Step 8	(Optional) switch# **show vtp counters**	Displays VTP pruning information and counters.
Step 9	(Optional) switch# **clear vtp counters**	Resets all the VTP pruning counter values.

Example

This example shows how to set VLANs 9 to 54 to be pruning eligible and set VLANs 2 to 8 and 55 to 1001 as not eligible for VTP pruning:

```
switch(config-if)# switchport trunk pruning vlan 9-54
```

VLAN 1 is never pruning eligible, because it is a factory-default VLAN. VLANs 1002 to 1005 are reserved for Token Ring networks. The VLANs 1006 is not pruning eligible.



CHAPTER 7

Configuring Private VLANs Using NX-OS

This chapter describes how to configure private VLANs on Cisco NX-OS devices. Private VLANs provide additional protection at the Layer 2 level.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About Private VLANs

Note A Layer 2 port can function as either a trunk port, an access port, or a private VLAN port.

Note Beginning with Cisco NX-OS Release 5.0(2), the system supports private VLAN promiscuous trunk ports and isolated trunk ports. Private VLAN community ports cannot be trunk ports.

Note You must enable the private VLAN feature before you can configure this feature.

In certain instances where similar systems do not need to interact directly, private VLANs provide additional protection at the Layer 2 level. Private VLANs are an association of primary and secondary VLANs.

A primary VLAN defines the broadcast domain with which the secondary VLANs are associated. The secondary VLANs may either be isolated VLANs or community VLANs. Hosts on isolated VLANs communicate only with associated promiscuous ports in primary VLANs, and hosts on community VLANs communicate only among themselves and with associated promiscuous ports but not with isolated ports or ports in other community VLANs.

In configurations that use integrated switching and routing functions, you can assign a single Layer 3 VLAN network interface to each private VLAN to provide routing. The VLAN network interface is created for the primary VLAN. In such configurations, all secondary VLANs communicate at Layer 3 only through a mapping with the VLAN network interface on the primary VLAN. Any VLAN network interfaces previously created on the secondary VLANs are put out-of-service.

Private VLAN Overview

You must enable private VLANs before the device can apply the private VLAN functionality.

You cannot disable private VLANs if the device has any operational ports in a private VLAN mode.

Note You must have already created the VLAN before you can convert the specified VLAN to a private VLAN, either primary or secondary.

Primary and Secondary VLANs in Private VLANs

The private VLAN feature addresses two problems that users encounter when using VLANs:

- Each VDC supports up to 4096 VLANs. If a user assigns one VLAN per customer, the number of customers that the service provider can support is limited.
- To enable IP routing, each VLAN is assigned with a subnet address space or a block of addresses, which can result in wasting the unused IP addresses and creating IP address management problems.

Using private VLANs solves the scalability problem and provides IP address management benefits and Layer 2 security for customers.

The private VLAN feature allows you to partition the Layer 2 broadcast domain of a VLAN into subdomains. A subdomain is represented by a pair of private VLANs: a primary VLAN and a secondary VLAN. A private VLAN domain can have multiple private VLAN pairs, one pair for each subdomain. All VLAN pairs in a private VLAN domain share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another.

Note A private VLAN domain has only one primary VLAN.

Secondary VLANs provide Layer 2 isolation between ports within the same private VLAN. The following two types are secondary VLANs within a primary VLAN:

- 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 community VLANs or in any isolated VLANs at the Layer 2 level.

Private VLAN Ports

Note Both community and isolated private VLAN ports are labeled as PVLAN host ports. A PVLAN host port is either a community PVLAN port or an isolated PVLAN port depending on the type of secondary VLAN with which it is associated.

The types of private VLAN ports are as follows:

- Promiscuous port—A promiscuous port belongs to the primary VLAN. The promiscuous port can communicate with all interfaces, including the community and isolated host ports, that belong to those secondary VLANs associated to the promiscuous port and associated with the primary VLAN. You can have several promiscuous ports in a primary VLAN. Each promiscuous port can have several secondary VLANs or no secondary VLANs, associated to that port. You can associate a secondary VLAN to more than one promiscuous port, as long as the promiscuous port and secondary VLANs are within the same primary VLAN. You may want to do this association for load balancing or redundancy purposes. You can also have secondary VLANs that are not associated to any promiscuous port, but these secondary VLANs cannot communicate to the Layer 3 interface.

  Beginning with Cisco NX-OS Release 5.0(2), the primary VLAN becomes inactive after you remove all the mapped secondary VLANs to that primary VLAN.

- Promiscuous trunk—Beginning with Cisco NX-OS Release 5.0(2) and Cisco DCNM Release 5.1(1), on the Cisco Nexus 7000 Series devices, you can configure a promiscuous trunk port to carry traffic for multiple primary VLANs. You map the private VLAN primary VLAN and either all or selected associated VLANs to the promiscuous trunk port. Each primary VLAN and one associated and secondary VLAN is a private VLAN pair, and you can configure a maximum of 16 private VLAN pairs on each promiscuous trunk port.

  Note Private VLAN promiscuous trunk ports carry traffic for normal VLANs as well as for primary private VLANs.

- Isolated port—An isolated port is a host port that belongs to an isolated secondary VLAN. This port has complete Layer 2 isolation from other ports within the same private VLAN domain, except that it can communicate with associated 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. You can have more than one isolated port in a specified isolated VLAN, and each port is completely isolated from all other ports in the isolated VLAN.

- Isolated or secondary trunk—Beginning with Cisco NX-OS Release 5.0(2) and Cisco DCNM Release 5.1(1) on the Cisco Nexus 7000 Series devices, you can configure an isolated trunk port to carry traffic for multiple isolated VLANs. Each secondary VLAN on an isolated trunk port must be associated with a different primary VLAN. You cannot put two secondary VLANs that are associated with the same primary VLAN on an isolated trunk port. Each primary VLAN and one associated secondary VLAN is a private VLAN pair, and you can configure a maximum of 16 private VLAN pairs on each isolated trunk port.

  Note Private VLAN isolated trunk ports carry traffic for normal VLANs as well as for secondary private VLANs.

- Community port—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 associated promiscuous ports. These interfaces are isolated at Layer 2 from all other interfaces in other communities and from all isolated ports within the private VLAN domain.

Note Because trunks can support the VLANs that carry traffic between promiscuous, isolated, and community ports, the isolated and community port traffic might enter or leave the device through a trunk interface.

Primary, Isolated, and Community Private VLANs

Because the primary VLAN has the Layer 3 gateway, you associate secondary VLANs with the primary VLAN in order to communicate outside the private VLAN. Primary VLANs and the two types of secondary VLANs, isolated VLANs and community VLANs, have these characteristics:

- Primary VLAN— The primary VLAN carries traffic from the promiscuous ports to the (isolated and community) host ports and to other promiscuous ports.
- Isolated VLAN —An isolated VLAN is a secondary VLAN that carries unidirectional traffic upstream from the hosts toward the promiscuous ports and the Layer 3 gateway. You can configure multiple isolated VLANs in a private VLAN domain, and all the traffic remains isolated within each one. In addition, each isolated VLAN can have several isolated ports, and the traffic from each isolated port also remains completely separate.
- 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 domain. The ports within one community can communicate, but these ports cannot communicate with ports in any other community or isolated VLAN in the private VLAN.

Figure 3: Private VLAN Layer 2 Traffic Flows

This figure shows the Layer 2 traffic flows within a primary, or private VLAN, along with the types of VLANs and types of ports.

Note The private VLAN traffic flows are unidirectional from the host ports to the promiscuous ports. Traffic that egresses the promiscuous port acts like the traffic in a normal VLAN, and there is no traffic separation among the associated secondary VLAN.

A promiscuous port can serve only one primary VLAN, but it can serve multiple isolated VLANs and multiple community VLANs. (Layer 3 gateways are 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.

Note Beginning with Cisco NX-OS Release 5.0(2) for the Nexus 7000 Series devices, you can configure private VLAN promiscuous and isolated trunk ports. These promiscuous and isolated trunk ports carry traffic for multiple primary and secondary VLANs as well as normal VLAN.

Although you can have several promiscuous ports in a primary VLAN, you can have only one Layer 3 gateway per primary VLAN.

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.

Note You must enable the VLAN interface feature before you can configure the Layer 3 gateway. See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for complete information on VLAN network interfaces and IP addressing.

Associating Primary and Secondary VLANs

To allow the host ports in secondary VLANs to communicate outside the private VLAN, you associate secondary VLANs to the primary VLAN. If the association is not operational, the host ports (isolated and community ports) in the secondary VLAN are brought down.

Note You can associate a secondary VLAN with only one primary VLAN.

For an association to be operational, the following conditions must be met:

- The primary VLAN must exist.
- The secondary VLAN must exist.
- The primary VLAN must be configured as a primary VLAN.
- The secondary VLAN must be configured as either an isolated or community VLAN.

Note See the **show** command display to verify that the association is operational. The device does not issue an error message when the association is nonoperational.

If you delete either the primary or secondary VLAN, the ports that are associated with the VLAN become inactive. When you reconvert the specified VLAN to private VLAN mode, the original associations are reinstated.

If the association is not operational on private VLAN trunk ports, only that VLAN goes down, not the entire port.

When you enter the **no private-vlan** command, the VLAN returns to the normal VLAN mode. All associations on that VLAN are suspended, but the interfaces remain in private VLAN mode.

If you enter the **no vlan** command for the primary VLAN, all private VLAN associations with that VLAN are lost. However, if you enter the **no vlan** command for a secondary VLAN, the private VLAN associations with that VLAN are suspended and return when you recreate the specified VLAN and configure it as the secondary VLAN.

Note This behavior is different from how Catalyst devices work.

In order to change the association between a secondary and primary VLAN, you must first remove the current association and then add the desired association.

Broadcast Traffic in Private VLANs

Broadcast traffic from ports in a private VLAN flows in the following ways:

- The broadcast traffic flows from all promiscuous ports to all ports in the primary VLAN. This broadcast traffic is distributed to all ports within the primary VLAN, including those ports that are not configured with private VLAN parameters.
- The broadcast traffic from all isolated ports is distributed only to those promiscuous ports in the primary VLAN that are associated to that isolated port.
- The broadcast traffic from community ports is distributed to all ports within the port's community and to all promiscuous ports that are associated to the community port. The broadcast packets are not distributed to any other communities within the primary VLAN or to any isolated ports.

Private VLAN Port Isolation

You can use private VLANs to control access to end stations as follows:

- 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.

Private VLANs and VLAN Interfaces

A VLAN interface to a Layer 2 VLAN is also called a switched virtual interface (SVI). Layer 3 devices communicate with a private VLAN only through the primary VLAN and not through secondary VLANs.

Configure VLAN network interfaces only for primary VLANs. Do not configure VLAN interfaces for secondary VLANs. VLAN network interfaces for secondary VLANs are inactive while the VLAN is configured as a secondary VLAN. You will see the following actions if you misconfigure the VLAN interfaces:

- If you try to configure a VLAN with an active VLAN network interface as a secondary VLAN, the configuration is not allowed until you disable the VLAN interface.
- If you try to create and enable a VLAN network interface on a VLAN that is configured as a secondary VLAN, that VLAN interface remains disabled and the system returns an error.

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

Note You must enable the VLAN interface feature before you configure VLAN interfaces. See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on VLAN interfaces and IP addressing.

Private VLANs Across Multiple Devices

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 uses of the VLANs configured to be private VLANs, configure private VLANs on all intermediate devices, including devices that have no private VLAN ports.

High Availability for Private VLANs

The software supports high availability for both stateful and stateless restarts, as during a cold reboot, for private VLANs. For the stateful restarts, the software supports a maximum of three retries. If you try more than 3 times within 10 seconds of a restart, the software reloads the supervisor module.

You can upgrade or downgrade the software seamlessly, with respect to private VLANs.

Beginning with Cisco NX-OS Release 5.0(2), if you configure private VLAN promiscuous or isolated trunk ports, you must unconfigure those ports in order to downgrade the software.

Note See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on high-availability features.

Virtualization Support for Private VLANs

The software supports virtual device contexts (VDCs).

Note See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on VDCs and assigning resources.

Each VLAN must have all of its private VLAN ports for both the primary VLAN and all secondary VLANs in the same VDC. Private VLANs cannot cross VDCs.

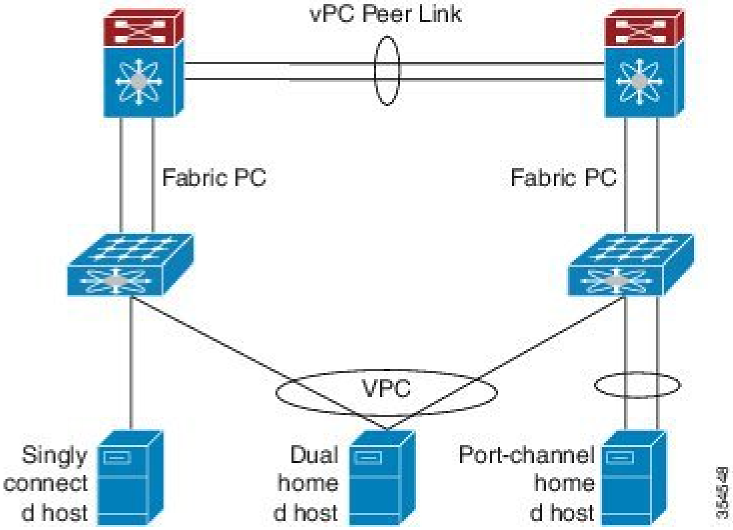
PVLAN (Isolated) on FEX HIF (Cisco Nexus 7000 Parent)

The isolated private VLAN (PVLAN) support on Fabric Extender (FEX) host interface (HIF) feature enables users to configure PVLAN isolated host and secondary trunk ports on FEX ports, where the parent switch is a Cisco Nexus 7000 series switch. With this feature, users can create end-to-end private VLAN (PVLAN) domain from trunk till host interface ports.

Supported Topologies for Isolated PVLAN on FEX HIF

Isolated PVLAN on FEX HIF is supported on single homed and dual homed vPC topologies.

Figure 4: Supported Topology for Isolated PVLAN on FEX HIF



Prerequisites for Private VLANs

Private VLANs have the following prerequisites:

- You must be logged onto the device.
- You must enable the private VLAN feature.

Guidelines and Limitations for Configuring Private VLANs

Private VLANs have the following configuration guidelines and limitations:

- You must enable private VLANs before the device can apply the private VLAN functionality.
- You must enable the VLAN interface feature before the device can apply this functionality.
- Shut down the VLAN network interface for all VLANs that you plan to configure as secondary VLANs before you configure these VLANs.
- Cisco NX-OS Release 6.0(x) does not support the PVLAN feature on F2 Series modules. .
- You cannot configure a shared interface to be part of a private VLAN. For more details, see the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*.
- M2, F2, F2e, and F3 fabric port modules only are supported for Isolated PVLAN on FEX HIF.
- A primary VLAN can be associated with only one isolated VLAN.
- Only isolated host and trunk secondary PVLAN port modes are supported on FEX HIF ports.
- If the **system private-vlan fex trunk disable** command is configured, then PVLAN are not carried on the HIF trunk ports on the FEX.
- PVLAN promiscuous, community and trunk promiscuous are not supported on FEX HIF ports.
- M1 and M1 XL fabric modules are not supported on FEX HIF ports.
- Isolated trunk configuration on any FEX port is not compatible with the global configuration **system private-vlan fex trunk**.
    - Before applying the global configuration **system private-vlan fex trunk**, you need to remove any isolated trunk configuration from FEX ports.
    - After applying the global configuration **system private-vlan fex trunk**, isolated trunk configuration on FEX ports is not supported.
    - To use isolated trunk on FEX ports, you must remove the global configuration **system private-vlan fex trunk**.

Secondary and Primary VLAN Configuration

Follow these guidelines when configuring secondary or primary VLANs in private VLANs:

- You cannot configure the default VLAN (VLAN1) or any of the internally allocated VLANs as primary or secondary VLANs.
- You must use VLAN configuration (config-vlan) mode to configure private VLANs.
- A primary VLAN can have multiple isolated and community VLANs associated with it. An isolated or community VLAN can be associated with only one primary VLAN.
- Although private VLANs provide host isolation at Layer 2, hosts can communicate with each other at Layer 3.

- When a secondary VLAN is associated with the primary VLAN, the STP parameters of the primary VLAN, such as bridge priorities, are propagated to the secondary VLAN. However, STP parameters do not necessarily propagate to other devices. You should manually check the STP configuration to ensure that the spanning tree topologies for the primary, isolated, and community VLANs match exactly so that the VLANs can properly share the same forwarding database.
- For normal trunk ports, note the following:
    - There is a separate instance of STP for each VLAN in the private VLAN.
    - STP parameters for the primary and all secondary VLANs must match.
    - The primary and all associated secondary VLANs should be in the same MST instance.
- For nontrunking ports, note the following:
    - STP is aware only of the primary VLAN for any private VLAN host port; STP does not run on secondary VLANs on a host port.

Note We recommend that you enable BPDU Guard on all ports that you configure as a host port; do not enable this feature on promiscuous ports.

- For private VLAN promiscuous trunk ports, note the following:
    - You can configure a maximum of 16 private VLAN primary and secondary VLAN pairs on each promiscuous trunk port.
    - The native VLAN must be either a normal VLAN or a private VLAN primary VLAN. You cannot configure a private VLAN secondary VLAN as the native VLAN for a private VLAN promiscuous trunk port.
    - To downgrade a system that has private VLAN promiscuous trunk ports configured, you must unconfigure these ports.
- For private VLAN isolated trunk ports, note the following:
    - You can configure a maximum of 16 private VLAN primary and secondary VLAN pairs on each isolated trunk port.
    - The native VLAN must be either a normal VLAN or a private VLAN secondary VLAN. You cannot configure a private VLAN primary port as the native VLAN for a private VLAN isolated trunk port.
    - To downgrade a system that has private VLAN isolated trunk ports configured, you must unconfigure these ports.
- You can apply different Quality of Service (QoS) configurations to primary, isolated, and community VLANs.
- To apply a VACL to all private VLAN traffic, map the secondary VLANs on the VLAN network interface of the primary VLAN, and then configure the VACLs on the VLAN network interface of the primary VLAN.

- The VACLs that you apply to the VLAN network interface of a primary VLAN automatically apply to the associated isolated and community VLANs only after you have configured the mapping.
- If you do not map the secondary VLAN to the VLAN network interface of the primary VLAN, you can have different VACLs for primary and secondary VLANs, which can cause problems.
- Because traffic in a private VLAN flows in different directions in different VLANs, you can have different VACLs for ingressing traffic and different VACLs for egressing traffic prior to configuring the mapping.

Note You must keep the same VACLs for the primary VLAN and all secondary VLANs in the private VLAN.

- You can enable DHCP snooping on private VLANs. When you enable DHCP snooping on the primary VLAN, the DHCP configuration 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.
- Before you configure a VLAN as a secondary VLAN, you must shut down the VLAN network interface for the secondary VLAN.
- To prevent interhost communication in isolated private VLANs with a promiscuous port, configure a role-based ACL (RBACL) that disallows hosts in that subnet from communicating with each other.

Private VLAN Port Configuration

Follow these guidelines when configuring private VLAN ports:

- Before Release 6.2(10), native VLANs are not supported for private VLAN configuration.
- Use only the private VLAN configuration commands to assign ports to primary, isolated, or community VLANs.
- The Layer 2 access ports that are 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, which may carry private VLANs, are active and remain part of the STP database.
- If you delete a VLAN used in the private VLAN configuration, the private VLAN ports that are associated with the VLAN become inactive.

Limitations with Other Features

Consider these configuration limitations with other features when configuring private VLANs:

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

- IGMP runs only on the primary VLAN and uses the configuration of the primary VLAN for all secondary VLANs.
- Any IGMP join request in the secondary VLAN is treated as if it is received in the primary VLAN.
- 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 host or promiscuous ports cannot be a SPAN destination port.
- A destination SPAN port cannot be an isolated port. (However, a source SPAN port can be an isolated port.)
- You can configure SPAN to span both primary and secondary VLANs or to span either one if the user is interested only in ingress or egress traffic.
- After you configure the association between the primary and secondary VLANs, the dynamic MAC addresses that learned the secondary VLANs are flushed.
- After you configure the association between the primary and secondary VLANs, all static MAC addresses that were created on the secondary VLANs are inserted into the primary VLAN. If you delete the association, the static MAC addresses revert to the secondary VLANs only.
- After you configure the association between the primary and secondary VLANs, you cannot create static MAC addresses for the secondary VLANs.
- After you configure the association between the primary and secondary VLANs, if you delete the association, all static MAC addresses that were created on the primary VLANs remain on the primary VLAN only.
- Port security features are not supported with private VLANs.
- In private VLANs, STP controls only the primary VLAN.
- Multicast or broadcast message,s such as ARP or HSRP hello, cannot be flooded through a private VLAN if you remove some of the secondary VLANs from a vPC trunk when a private VLAN, MST, or vPC is configured or if you delete some of the secondary VLANs. In this case, you should reconfigure the removed secondary VLANs as a trunk again, or reconfigure the deleted secondary VLANS again.

Note See the *Cisco Nexus 7000 Series NX-OS Security Configuration Guide* for information on configuring static MAC addresses.

Default Settings for Private VLANs

This table lists the default setting for private VLANs.

Table 9: Default Private VLAN Setting

Parameters	Default
Private VLANs	Disabled

Configuring a Private VLAN

You must have already created the VLAN before you can assign the specified VLAN as a private VLAN.

See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on assigning IP addresses to VLAN interfaces.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Enabling Private VLANs

You must enable private VLANs on the device to have the private VLAN functionality.

Note The private VLAN commands do not appear until you enable the private VLAN feature.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**feature private-vlan**
	Example:
	`switch(config)# feature private-vlan`
	`switch(config)#`
		Enables private VLAN functionality on the device.
		Note You cannot apply the **no feature private-vlan** command if there are operational ports on the device that are in private VLAN mode.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to enable private VLAN functionality on the device:

```
switch# config t
switch(config)# feature private-vlan
switch(config)#
```

Configuring a VLAN as a Private VLAN

Note Before you configure a VLAN as a secondary VLAN—that is, either a community or isolated VLAN—you must first shut down the VLAN network interface.

You can configure a VLAN as a private VLAN.

To create a private VLAN, you first create a VLAN and then configure that VLAN to be a private VLAN.

You create all VLANs that you want to use in the private VLAN as a primary VLAN, a community VLAN, or an isolated VLAN. You will later associate multiple isolated and multiple community VLANs to one primary VLAN. You can have many primary VLANs and associations, which means that you could have many private VLANs.

If you delete either the primary or secondary VLAN, the ports that are associated with the VLAN become inactive.

On private VLAN trunk ports, if you delete either the secondary or primary VLAN, only that specific VLAN becomes inactive; the trunk ports stay up.

Before you begin

Ensure that the private VLAN feature is enabled.

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vlan** {*vlan-id* | *vlan-range*}
	Example:
	`switch(config)# vlan 5`
	`switch(config-vlan)#`
		Places you into VLAN configuration submode.

Step 3	Enter one of the following commands:	

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td><table><tr><th>Option</th><th>Description</th></tr><tr><td><b>private-vlan</b> {<b>community</b> | <b>isolated</b> | <b>primary</b>}</td><td>Configures the VLAN as either a community, isolated, or primary private VLAN. In a private VLAN, you must have one primary VLAN. You can have multiple community and isolated VLANs.</td></tr><tr><td><b>no private-vlan</b> {<b>community</b> | <b>isolated</b> | <b>primary</b>}</td><td>Removes the private VLAN configuration from the specified VLAN(s) and returns it to normal VLAN mode. If you delete either the primary or secondary VLAN, the ports that are associated with the VLAN become inactive.</td></tr></table>
<b>Example:</b>
<code>switch(config-vlan)# private-vlan primary</code></td><td></td></tr>
<tr><td><b>Step 4</b></td><td><b>exit</b>
<b>Example:</b>
<code>switch(config-vlan)# exit</code>
<code>switch(config)#</code></td><td>Exits VLAN configuration submode.</td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) <b>show vlan private-vlan</b> [<i>type</i>]
<b>Example:</b>
<code>switch# show vlan private-vlan</code></td><td>Displays the private VLAN configuration.</td></tr>
<tr><td><b>Step 6</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
<code>switch(config)# copy running-config startup-config</code></td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to assign VLAN 5 to a private VLAN as the primary VLAN:

```
switch# config t
switch(config)# vlan 5
switch(config-vlan)# private-vlan primary
switch(config-vlan)# exit
switch(config)#
```

Associating Secondary VLANs with a Primary Private VLAN

Follow these guidelines when you associate secondary VLANs with a primary VLAN:

- The *secondary-vlan-list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single secondary VLAN ID or a hyphenated range of secondary VLAN IDs.
- The *secondary-vlan-list* parameter can contain multiple community and isolated VLAN IDs.
- Enter a *secondary-vlan-list* or enter the **add** keyword with a *secondary-vlan-list* to associate secondary VLANs with a primary VLAN.
- Enter the **remove** keyword with a *secondary-vlan-list* to clear the association between secondary VLANs and a primary VLAN.
- You change the association between a secondary and primary VLAN by removing the existing association and then adding the desired association.

If you delete either the primary or secondary VLAN, the ports that are associated with the VLAN become inactive.

When you enter the **no private-vlan** command, the VLAN returns to the normal VLAN mode. All associations on that VLAN are suspended, but the interfaces remain in private VLAN mode.

When you reconvert the specified VLAN to private VLAN mode, the original associations are reinstated.

If you enter the **no vlan** command for the primary VLAN, all private VLAN associations with that VLAN are lost. However, if you enter the **no vlan** command for a secondary VLAN, the private VLAN associations with that VLAN are suspended and return when you recreate the specified VLAN and configure it as the previous secondary VLAN.

Before you begin

Ensure that the private VLAN feature is enabled.

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vlan** *primary-vlan-id*
	Example:
	`switch(config)# vlan 5`
	`switch(config-vlan)#`
		Enters the number of the primary VLAN that you are working in for the private VLAN configuration.

Step 3	Enter one of the following commands:	

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>**private-vlan association** {[**add**] *secondary-vlan-list* | **remove** *secondary-vlan-list*}</td><td>Associates the secondary VLANs with the primary VLAN.</td></tr>
<tr><td>**no private-vlan association**</td><td>Removes all associations from the primary VLAN and returns it to normal VLAN mode.</td></tr>
</table>
Example:
`switch(config-vlan)# private-vlan association 100-105,109`</td><td></td></tr>
<tr><td>**Step 4**</td><td>**exit**
Example:
`switch(config-vlan)# exit`
`switch(config)#`</td><td>Exits VLAN configuration submode.</td></tr>
<tr><td>**Step 5**</td><td>(Optional) **show vlan private-vlan** [**type**]
Example:
`switch# show vlan private-vlan`</td><td>Displays the private VLAN configuration.</td></tr>
<tr><td>**Step 6**</td><td>(Optional) **copy running-config startup-config**
Example:
`switch(config)# copy running-config startup-config`</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to associate community VLANs 100 through 105 and isolated VLAN 109 with primary VLAN 5:

```
switch(config)# vlan 5
switch(config-vlan)# private-vlan association 100-105, 109
switch(config-vlan)# exit
switch(config)#
```

Mapping Secondary VLANs to the VLAN Interface of a Primary VLAN

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on assigning IP addresses to VLAN interfaces on primary VLANs of private VLANs.

You map secondary VLANs to the VLAN interface of a primary VLAN. Isolated and community VLANs are both called secondary VLANs. To allow Layer 3 processing of private VLAN ingress traffic, you map secondary VLANs to the VLAN network interface of a primary VLAN.

Note You must enable VLAN network interfaces before you configure the VLAN network interface. VLAN network interfaces on community or isolated VLANs that are associated with a primary VLAN will be out of service. Only the VLAN network interface on the primary VLAN is in service.

Before you begin

- Enable the private VLAN feature.
- Enable the VLAN interface feature.
- Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.
- Ensure that you are working on the correct primary VLAN Layer 3 interface to map the secondary VLANs.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface vlan** *primary-vlan-ID*
	Example:
	`switch(config)# interface vlan 5`
	`switch(config-if)#`
		Enters the number of the primary VLAN that you are working in for the private VLAN configuration and places you into the interface configuration mode for the primary VLAN.

Step 3	Enter one of the following commands: **Option**: **private-vlan mapping** {[**add**] *secondary-vlan-list* | **remove** *secondary-vlan-list*} **Description**: Maps the secondary VLANs to the SVI or Layer 3 interface of the primary VLAN. This action allows the Layer	

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td><table><tr><th>Option</th><th>Description</th></tr><tr><td></td><td>3 switching of private VLAN ingress traffic.</td></tr><tr><td><b>no private-vlan mapping</b></td><td>Clears the mapping to the Layer 3 interface between the secondary VLANs and the primary VLANs.</td></tr></table><b>Example:</b>
<code>switch(config-if)# private-vlan mapping 100-105, 109</code></td><td></td></tr>
<tr><td><b>Step 4</b></td><td><b>exit</b>
<b>Example:</b>
<code>switch(config-if)# exit</code>
<code>switch(config)#</code></td><td>Exits interface configuration mode.</td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) <b>show interface vlan</b> <i>primary-vlan-id</i> <b>private-vlan mapping</b>
<b>Example:</b>
<code>switch(config)# show interface vlan 101 private-vlan mapping</code></td><td>Displays the interface private VLAN information.</td></tr>
<tr><td><b>Step 6</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
<code>switch(config)# copy running-config startup-config</code></td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to map the secondary VLANs 100 through 105 and 109 on the Layer 3 interface of the primary VLAN 5:

```
switch # config t
switch(config)# interface vlan 5
switch(config-if)# private-vlan mapping 100-105, 109
switch(config-if)# exit
switch(config)#
```

Configuring a Layer 2 Interface as a Private VLAN Host Port

You can configure a Layer 2 interface as a private VLAN host port. In private VLANs, host ports are part of the secondary VLANs, which are either community VLANs or isolated VLANs.

Note We recommend that you enable BPDU Guard on all interfaces configured as a host port.

You then associate the host port with both the primary and secondary VLANs.

Before you begin

Ensure that the private VLAN feature is enabled.

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td><b>config t</b>
<b>Example:</b>
<code>switch# config t</code>
<code>switch(config)#</code></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td><b>interface</b> <i>type slot/port</i>
<b>Example:</b>
<code>switch(config)# interface ethernet 2/1</code>
<code>switch(config-if)#</code></td><td>Selects the Layer 2 port to configure as a private VLAN host port.</td></tr>
<tr><td><b>Step 3</b></td><td><b>switchport mode private-vlan host</b>
<b>Example:</b>
<code>switch(config-if)# switchport mode</code>
<code>private-vlan host</code>
<code>switch(config-if)#</code></td><td>Configures the Layer 2 port as a host port for a private VLAN.</td></tr>
<tr><td><b>Step 4</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>switchport private-vlan host-association</b> {<i>primary-vlan-id</i>} {<i>secondary-vlan-id</i>}</td><td>Associates the Layer 2 host port with the primary and secondary VLANs of a private VLAN. The secondary VLAN can be either an isolated or community VLAN.</td></tr>
<tr><td><b>no switchport private-vlan host-association</b></td><td>Removes the private VLAN association from the port.</td></tr>
</table>
<b>Example:</b>
<code>switch(config-if)# switchport</code>
<code>private-vlan host-association 10 50</code></td><td></td></tr>
</table>

	Command or Action	Purpose

Step 5	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits the interface configuration mode.

Step 6	(Optional) **show interface switchport** **Example:** `switch# show interface switchport`	Displays information on all interfaces configured as switch ports.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the Layer 2 port 2/1 as a host port for a private VLAN and associate it to primary VLAN 10 and secondary VLAN 50:

```
switch# config t
switch(config)# interface ethernet 2/1
switch(config-if)# switchport mode private-vlan host
switch(config-if)# switchport private-vlan host-association 10 50
switch(config-if)# exit
switch(config)#
```

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

Beginning with Cisco NX-OS Release 5.0(2), you can configure a Layer 2 interface as a private VLAN isolated trunk port. These isolated trunk ports carry traffic for multiple secondary VLANs as well as normal VLANs.

Note You must associate the primary and secondary VLANs before they become operational on the private VLAN isolated trunk port.

Before you begin

Ensure that the private VLAN feature is enabled.

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** {*type slot/port*}
	Example:
	`switch(config)# interface ethernet 2/11`
	`switch(config-if)#`
		Selects the Layer 2 port to configure as a private VLAN isolated trunk port.

Step 3	**switchport**
	Example:
	`switch(config-if)# switchport`
	`switch(config-if)#`
		Configures the Layer 2 port as a switch port.

Step 4	**switchport mode private-vlan trunk secondary**
	Example:
	`switch(config-if)# switchport mode private-vlan trunk secondary`
	`switch(config-if)#`
		Configures the Layer 2 port as an isolated trunk port to carry traffic for multiple isolated VLANs.
		Note You cannot put community VLANs into the isolated trunk port.

Step 5	(Optional) **switchport private-vlan trunk native vlan** *vlan-id* **Example:** `switch(config-if)# switchport private-vlan trunk native vlan 5`	Sets the native VLAN for the 802.1Q trunk. Valid values are from 1 to 3968 and 4048 to 4093. The default value is 1. **Note** If you are using a private VLAN as the native VLAN for the isolated trunk port, you must enter a value for a secondary VLAN or a normal VLAN; you cannot configure a primary VLAN as the native VLAN.

Step 6	**switchport private-vlan trunk allowed vlan** {**add** *vlan-list* | **all** | **except** *vlan-list* | **none** | **remove** *vlan-list*}
	Example:
	`switch(config-if)# switchport private-vlan trunk allowed vlan add 1`
	`switch(config-if)#`
		Sets the allowed VLANs for the private VLAN isolated trunk interface. Valid values are from 1 to 3968 and 4048 to 4093.
		When you map the private primary and secondary VLANs to the isolated trunk port, the system automatically puts all the primary VLANs into the allowed VLAN list for this port.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td></td><td><b>Note</b> Ensure that the native VLAN is part of the allowed VLAN list. The default for this command is to allow no VLANs on this interface, so you must configure the native VLAN as an allowed VLAN, unless it is already added as an associated VLAN, to pass native VLAN traffic.</td></tr>
<tr><td><b>Step 7</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>switchport private-vlan association trunk</b> {<i>primary-vlan-id</i>} {<i>secondary-vlan-id</i>}</td><td>Associates the Layer 2 isolated trunk port with the primary and secondary VLANs of private VLANs. The secondary VLAN must be an isolated VLAN. You can associate a maximum of 16 private VLAN primary and secondary pairs on each isolated trunk port. You must reenter the command for each pair of primary and secondary VLANs that you are working with.</td></tr>
</table></td><td></td></tr>
</table>

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td><table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td></td><td><b>Note</b> Each secondary VLAN on an isolated trunk port must be associated with a different primary VLAN. You cannot put two isolated VLANs that are associated with the same primary VLAN into a private VLAN isolated trunk port. If you do, the last entry overwrites the previous entry.</td></tr>
<tr><td><b>no switchport private-vlan association trunk</b> [<i>primary-vlan-id</i> [<i>secondary-vlan-id</i>]]</td><td>Removes the private VLAN association from the private VLAN isolated trunk port.</td></tr>
</table>
<b>Example:</b>
switch(config-if)# switchport
private-vlan association trunk 10 101
switch(config-if)#</td><td></td></tr>
<tr><td><b>Step 8</b></td><td><b>exit</b>
<b>Example:</b>
switch(config-if)# exit
switch(config)#</td><td>Exits interface configuration mode.</td></tr>
</table>

	Command or Action	Purpose
Step 9	(Optional) **show interface switchport** **Example:** `switch# show interface switchport`	Displays information on all interfaces configured as switch ports.
Step 10	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the Layer 2 port 2/1 as a private VLAN isolated trunk port associated with three different primary VLANs and an associated secondary VLAN:

```
switch# config t
switch(config)# interface ethernet 2/1
switch(config-if)# switchport mode private-vlan trunk secondary
switch(config-if)# switchport private-vlan trunk allowed vlan add 1
switch(config-if)# switchport private-vlan association trunk 10 101
switch(config-if)# switchport private-vlan association trunk 20 201
switch(config-if)# switchport private-vlan association trunk 30 102
switch(config-if)# exit
switch(config)#
```

Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port

You can configure a Layer 2 interface as a private VLAN promiscuous port and then associate that promiscuous port with the primary and secondary VLANs.

Before you begin

Ensure that the private VLAN feature is enabled.

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** {*type slot/port*} **Example:**	Selects the Layer 2 port to configure as a private VLAN promiscuous port.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>switch(config)# interface ethernet 2/1
switch(config-if)#</td><td></td></tr>
<tr><td>Step 3</td><td><b>switchport mode private-vlan promiscuous</b>
<b>Example:</b>
switch(config-if)# switchport mode
private-vlan promiscuous</td><td>Configures the Layer 2 port as a promiscuous port for a private VLAN.</td></tr>
<tr><td>Step 4</td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>switchport private-vlan mapping</b> {<i>primary-vlan-id</i>} {<i>secondary-vlan-list</i> | <b>add</b> <i>secondary-vlan-list</i> | <b>remove</b> <i>secondary-vlan-list</i>}</td><td>Configures the Layer 2 port as a promiscuous port and associates the specified port with a primary VLAN and a selected list of secondary VLANs. The secondary VLAN can be either an isolated or community VLAN.</td></tr>
<tr><td><b>no switchport private-vlan mapping</b></td><td>Clears the mapping from the private VLAN.</td></tr>
</table>
<b>Example:</b>
switch(config-if)# switchport
private-vlan mapping 10 50</td><td></td></tr>
<tr><td>Step 5</td><td><b>exit</b>
<b>Example:</b>
switch(config-if)# exit
switch(config)#</td><td>Exits interface configuration mode.</td></tr>
<tr><td>Step 6</td><td>(Optional) <b>show interface switchport</b>
<b>Example:</b>
switch# show interface switchport</td><td>Displays information on all interfaces configured as switch ports.</td></tr>
<tr><td>Step 7</td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
switch(config)# copy running-config
startup-config</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to configure the Layer 2 port 2/1 as a promiscuous port associated with the primary VLAN 10 and the secondary isolated VLAN 50:

```
switch# config t
switch(config)# interface ethernet 2/1
switch(config-if)# switchport mode private-vlan promiscuous
switch(config-if)# switchport private-vlan mapping 10 50
switch(config-if)# exit
switch(config)#
```

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

Beginning with Cisco NX-OS Release 5.0(2), you can configure a Layer 2 interface as a private VLAN promiscuous trunk port and then associate that promiscuous trunk port with multiple primary VLANs. These promiscuous trunk ports carry traffic for multiple primary VLANs as well as normal VLANs.

Note You must associate the primary and secondary VLANs before they become operational on the private VLAN promiscuous trunk port.

Before you begin

Ensure that the private VLAN feature is enabled.

Ensure that you are in the correct VDC (or enter the **switchto vdc** command). You can repeat VLAN names and IDs in different VDCs, so you must confirm that you are working in the correct VDC.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** {*type slot/port*}
	Example:
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		Selects the Layer 2 port to configure as a private VLAN promiscuous trunk port.

Step 3	**switchport**
	Example:
	`switch(config-if)# switchport`
	`switch(config-if)#`
		Configures the Layer 2 port as a switch port.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td>Step 4</td><td><b>switchport mode private-vlan trunk promiscuous</b>
<b>Example:</b>
switch(config-if)# switchport mode private-vlan trunk promiscuous
switch(config-if)#</td><td>Configures the Layer 2 port as a promiscuous trunk port to carry traffic for multiple private VLANs as well as normal VLANs.</td></tr>
<tr><td>Step 5</td><td>(Optional) <b>switchport private-vlan trunk native vlan</b> <i>vlan-id</i>
<b>Example:</b>
switch(config-if)# switchport private-vlan trunk native vlan 5</td><td>Sets the native VLAN for the 802.1Q trunk. Valid values are from 1 to 3968 and 4048 to 4093. The default value is 1.
<b>Note</b> If you are using a private VLAN as the native VLAN for the promiscuous trunk port, you must enter a value for a primary VLAN or a normal VLAN; you cannot configure a secondary VLAN as the native VLAN.</td></tr>
<tr><td>Step 6</td><td><b>switchport private-vlan trunk allowed vlan</b> {<b>add</b> <i>vlan-list</i> | <b>all</b> | <b>except</b> <i>vlan-list</i> | <b>none</b> | <b>remove</b> <i>vlan-list</i>}
<b>Example:</b>
switch(config-if)# switchport private-vlan trunk allowed vlan add 1
switch(config-if)#</td><td>Sets the allowed VLANs for the private VLAN promiscuous trunk interface. Valid values are from 1 to 3968 and 4048 to 4093.
When you map the private primary and secondary VLANs to the promiscuous trunk port, the system automatically puts all the primary VLANs into the allowed VLAN list for this port.
<b>Note</b> Ensure that the native VLAN is part of the allowed VLAN list. The default for this command is to allow no VLANs on this interface, so you must configure the native VLAN as an allowed VLAN, unless it is already added as an associated VLAN, to pass native VLAN traffic.</td></tr>
<tr><td>Step 7</td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>switchport private-vlan mapping trunk</b> <i>primary-vlan-id</i> {<b>add</b> <i>secondary-vlan-list</i> | <b>remove</b> <i>secondary-vlan-id</i>}</td><td>Maps or removes the mapping for the promiscuous trunk port with the primary VLAN and a selected list of associated secondary VLANs. The secondary VLAN can be either an</td></tr>
</table></td><td></td></tr>
</table>

	Command or Action	Purpose

	<table><tr><th>Option</th><th>Description</th></tr><tr><td></td><td>isolated or community VLAN. The private VLAN association between primary and secondary VLANs must be operational to pass traffic. You can map a maximum of 16 private VLAN primary and secondary pairs on each promiscuous trunk port. You must reenter the command for each primary VLAN that you are working with.</td></tr><tr><td>**no switchport private-vlan mapping trunk** [*primary-vlan-id* [*secondary-vlan-id*]]</td><td>Removes the private VLAN promiscuous trunk mappings from the interface.</td></tr></table>
	Example:
	`switch(config-if)# switchport private-vlan mapping trunk 4 add 5`
	`switch(config-if)#`	

Step 8	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 9	(Optional) **show interface switchport** **Example:** `switch# show interface switchport`	Displays information on all interfaces configured as switch ports.
Step 10	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the Layer 2 port 2/1 as a promiscuous trunk port associated with two primary VLANs and their associated secondary VLANs:

```
switch# config t
switch(config)# interface ethernet 2/1
switch(config-if)# switchport
switch(config-if)# switchport mode private-vlan trunk promiscuous
switch(config-if)# switchport private-vlan trunk allowed vlan add 1
switch(config-if)# switchport private-vlan mapping trunk 2 add 3
switch(config-if)# switchport private-vlan mapping trunk 4 add 5
switch(config-if)# switchport private-vlan mapping trunk 1 add 20
switch(config-if)# exit
switch(config)#
```

Configuring Isolated PVLANs on FEX HIF Ports

Disabling PVLAN on HIF Ports

Before you begin

- Ensure the **feature private-vlan** command is enabled.
- Ensure PVLANs are not configured or enabled on HIF FEX trunk ports.

Procedure

Bring down PVLAN from coming up on the HIF FEX trunk ports. HIF ports are non-PVLANs configured ports on FEX.

switch(config)# **system private-vlan fex trunk disable**

Note If the **system private-vlan fex trunk disable** command is not configured and you try to configure PVLANs on trunk ports, the system will display an error message.

Note The **system private-vlan fex trunk disable** command will disable PVLANs only on trunk ports. On access ports, the PVLANs are not operational on these ports.

What to do next

- Configuring PVLAN on FEX Isolated Host Port

Configuring PVLAN on FEX Isolated Ports

Procedure

Step 1 Configure interface and interface port for FEX isolated host port and enter interface configuration mode. FEX isolated host ports are ports with PVLAN configuration.

switch(config)# **interface** *interface-type/slot/port*

Step 2 Enable switchport for an interface.

switch(config-if)# **switchport**

Step 3 Configure the port mode to private-vlan host.

switch(config-if)# **switchport mode private-vlan host**

Step 4 Configure PVLAN association on the port.

switch(config-if)# **switchport private-vlan association** *primary-vlan-id* *secondary-vlan-id*

Example

```
switch(config)# interface Ethernet100/1/13
switch(config-if)# switchport
switch(config-if)# switchport mode private-vlan host
switch(config-if)# switchport private-vlan host-association 550 551
```

What to do next

- Configuring PVLAN on Trunk Secondary Port

Configuring PVLAN on Trunk Secondary Port

Procedure

Step 1 Configure interface and interface port for secondary trunk port and enter interface configuration mode.

switch(config)# **interface** *interface-type/slot/port*

Step 2 Enable switchport for an interface.

switch(config-if)# **switchport**

Step 3 Configure the port in trunk secondary mode.

switch(config-if)# **switchport mode private-vlan trunk secondary**

Step 4 Configure association PVLAN pair between primary and secondary VLAN.

switch(config-if)# **switchport private-vlan association trunk** *primary-vlan-id* *secondary-vlan-id*

Step 5 Configure association between primary PVLAN and isolated PVLAN.

switch(config-if)# **vlan** *primary-vlan-id*

switch(config-vlan)# **private-vlan association** *secondary-vlan-id*

switch(config-vlan)# **vlan** *secondary-vlan-id* switch(config-vlan)# **private-vlan isolated**

Example

```
switch(config)# interface Ethernet100/1/13
switch(config-if)# switchport
switch(config-if)# switchport mode private-vlan trunk secondary
switch(config-if)# switchport private-vlan association trunk 550 551
switch(config-if)# vlan 550
switch(config-vlan)# private-vlan association 551
switch(config-vlan)# vlan 551
switch(config-vlan)# private-vlan isolated
switch(config-vlan)# exit
```

What to do next

- (Optional) Verifying PVLAN (Isolated) Configurations on FEX HIF

Verifying PVLAN (Isolated) Configurations on FEX HIF

Procedure

Display the PVLAN operational status.

switch# **show vlan private-vlan**

Example

```
switch# show vlan private-vlan

Primary  Secondary     Type           Ports
         -------      ---------  ---------------
550       551         isolated         Eth100/1/13
```

Verifying the Private VLAN Configuration

To display private VLAN configuration information, perform one of the following tasks:

Command	Purpose
show running-config vlan *vlan-id*	Displays VLAN information.
show vlan private-vlan [*type*]	Displays information on private VLANs.

Command	Purpose
show interface private-vlan mapping	Displays information on interfaces for private VLAN mapping.
show interface vlan *primary-vlan-id* **private-vlan mapping**	Displays information on interfaces for private VLAN mapping.
show interface switchport	Displays information on all interfaces configured as switch ports.

For detailed information about the fields in the output from these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Displaying and Clearing Private VLAN Statistics

To display private VLAN configuration information, perform one of the following tasks:

Command	Purpose
clear vlan [**id** *vlan-id*] **counters**	Clears counters for all VLANs or for a specified VLAN.
show vlan counters	Displays information on Layer 2 packets in each VLAN.

Configuration Examples for Private VLANs

The following example shows how to create the three types of private VLANs, how to associate the secondary VLANs to the primary VLAN, how to create a private VLAN host and promiscuous port and assign them to the correct VLAN, and how to create a VLAN interface, or SVI, to allow the primary VLAN to communicate with the rest of the network:

```
switch# configure terminal
switch(config)# vlan 2
switch(config-vlan)# private-vlan primary
switch(config-vlan)# exit
switch(config)# vlan 3
switch(config-vlan)# private-vlan community
switch(config-vlan)# exit
switch(config)# vlan 4
switch(config-vlan)# private-vlan isolated
switch(config-vlan)# exit

switch(config)# vlan 2
switch(config-vlan)# private-vlan association 3,4
switch(config-vlan)# exit

switch(config)# interface ethernet 1/11
switch(config-if)# switchport
switch(config-if)# switchport mode private-vlan host
switch(config-if)# exit
switch(config)# interface ethernet 1/12
switch(config-if)# switchport
switch(config-if)# switchport mode private-vlan promiscuous
switch(config-if)# exit

switch(config)# interface ethernet 1/11
switch(config-if)# switchport private-vlan host-association 2 3
switch(config-if)# exit
switch(config)# interface ethernet 1/12
switch(config-if)# switchport private-vlan mapping 2 3,4
switch(config-if)# exit

switch(config)# interface vlan 2
switch(config-vlan)# private-vlan mapping 3,4
switch(config-vlan)# exit
switch(config)#
```

Additional References for Private VLANs

Related Documents

Related Topic	Document Title
Command reference	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*
VLAN interfaces, IP addressing	*Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
Static MAC addresses, security	*Cisco Nexus 7000 Series NX-OS Security Configuration Guide*
Cisco NX-OS fundamentals	*Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide*
High availability	*Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
System management	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
Licensing	*Cisco NX-OS Licensing Guide*
Release notes	*Cisco Nexus 7000 Series NX-OS Release Notes*

Standards

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

MIBs

MIBs	MIBs Link
• CISCO-PRIVATE-VLAN-MIB	To locate and download MIBs, go to the following URL: https://cfnng.cisco.com/mibs.

Feature History for Configuring Private VLANs

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Table 10: Feature History for Configuring Private VLANs

Feature Name	Releases	Feature Information	
PVLAN (Isolated) on FEX HIF (Cisco Nexus 7000 Parent)	7.3(0)D1(1)	This feature allows users to configure PVLAN isolated host and secondary trunk ports on FEX ports.	
Private VLAN promiscuous trunk ports and isolated trunk ports	5.0(2)	This feature allows promiscuous ports to carry traffic for multiple private VLANs and normal VLANs and allows isolated ports to carry traffic for multiple isolated VLANs and normal VLANs.	
No change	4.2(1)	–	
Display features enabled on the device	4.1(2)	You can display which features are enabled on the device by entering the following command: • **show feature**	

CHAPTER 8

Configuring Rapid PVST+ Using Cisco NX-OS

This chapter describes how to configure the Rapid per VLAN Spanning Tree (Rapid PVST+) protocol on Cisco NX-OS devices.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About Rapid PVST+

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on creating Layer 2 interfaces.

The Spanning Tree Protocol (STP) was implemented to provide a loop-free network at Layer 2 of the network. Rapid PVST+ is an updated implementation of STP that allows you to create one spanning tree topology for each VLAN. Rapid PVST+ is the default STP mode on the device.

Note Spanning tree is used to refer to IEEE 802.1w and IEEE 802.1s. If the IEEE 802.1D Spanning Tree Protocol is discussed in this publication, then 802.1D is stated specifically.

Note You can run either Rapid PVST+ or MST within each virtual device context (VDC). You cannot run both STP modes simultaneously in a VDC. Rapid PVST+ is the default STP mode.

The Rapid PVST+ protocol is the IEEE 802.1w standard, Rapid Spanning Tree Protocol (RSTP), implemented on a per VLAN basis. Rapid PVST+ interoperates with the IEEE 802.1Q VLAN standard, which mandates a single STP instance for all VLANs, rather than per VLAN.

Rapid PVST+ is enabled by default on the default VLAN (VLAN1) and on all newly created VLANs on the device. Rapid PVST+ interoperates with devices that run legacy IEEE 802.1D STP.

RSTP is an improvement on the original STP standard, 802.1D, which allows faster convergence.

Note The device supports full nondisruptive upgrades for Rapid PVST+. See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on nondisruptive upgrades.

The Cisco NX-OS release that is running on a managed device may not support all the features or settings described in this chapter. For the latest feature information and caveats, see the documentation and release notes for your platform and software release.

Note Beginning with Cisco NX-OS Release 5.x, when you are running virtual port channels (vPCs), you can configure STP for better performance. See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for more information on this feature.

STP

STP is a Layer 2 link-management protocol that provides path redundancy while preventing loops in the network.

Overview of STP

In order for a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. STP operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

When you create fault-tolerant internetworks, you must have a loop-free path between all nodes in a network. The STP algorithm calculates the best loop-free path throughout a switched Layer 2 network. Layer 2 LAN ports send and receive STP frames, which are called Bridge Protocol Data Units (BPDUs), at regular intervals. Network devices do not forward these frames but use the frames to construct a loop-free path.

Multiple active paths between end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages and network devices might learn end station MAC addresses on multiple Layer 2 LAN ports.

STP defines a tree with a root bridge and a loop-free path from the root to all network devices in the Layer 2 network. STP forces redundant data paths into a blocked state. If a network segment in the spanning tree fails and a redundant path exists, the STP algorithm recalculates the spanning tree topology and activates the blocked path.

When two Layer 2 LAN ports on a network device are part of a loop, the STP port priority and port path-cost setting determine which port on the device is put in the forwarding state and which port is put in the blocking state. The STP port priority value is the efficiency with which that location allows the port to pass traffic. The STP port path-cost value is derived from the media speed.

How a Topology is Created

All devices in a LAN that participate in a spanning tree gather information about other switches in the network by exchanging BPDUs. This exchange of BPDUs results in the following actions:

- The system elects a unique root switch for the spanning tree network topology.
- The system elects a designated switch for each LAN segment.
- The system eliminates any loops in the switched network by placing redundant switch ports in a backup state; all paths that are not needed to reach the root device from anywhere in the switched network are placed in an STP-blocked state.

The topology on an active switched network is determined by the following:

- The unique device identifier Media Access Control (MAC) address of the device that is associated with each device
- The path cost to the root that is associated with each switch port
- The port identifier that is associated with each switch port

In a switched network, the root switch is the logical center of the spanning tree topology. STP uses BPDUs to elect the root switch and root port for the switched network.

Note The **mac-address bpdu source version 2** command enables STP to use the new Cisco MAC address (00:26:0b:xx:xx:xx) as the source address of BPDUs generated on vPC ports.

To apply this command, you must have identical configurations for both vPC peer switches or peers.

We strongly recommend that you disable EtherChannel Guard on the edge devices before using this command to minimize traffic disruption from STP inconsistencies. Reenable the EtherChannel Guard after updating on both peers.

Bridge ID

Each VLAN on each network device has a unique 64-bit bridge ID that consists of a bridge priority value, an extended system ID (IEEE 802.1t), and an STP MAC address allocation.

Bridge Priority Value

The bridge priority is a 4-bit value when the extended system ID is enabled.

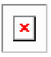
You can only specify a device bridge ID (used by the spanning tree algorithm to determine the identity of the root bridge; the lowest number is preferred) as a multiple of 4096.

Note In this device, the extended system ID is always enabled; you cannot disable the extended system ID.

Extended System ID

The device always uses the 12-bit extended system ID.

Figure 5: Bridge ID with Extended System ID

This figure shows the 12-bit extended system ID field that is part of the bridge ID.

This table shows how the system ID extension combined with the bridge ID functions as the unique identifier for a VLAN.

Table 11: Bridge Priority Value and Extended System ID with the Extended System ID Enabled

Bridge Priority Value				Extended System ID (Set Equal to the VLAN ID)											
Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

STP MAC Address Allocation

Note MAC address reduction is always enabled on the device.

Because MAC address reduction is always enabled on the device, you should also enable MAC address reduction on all other Layer 2 connected network devices to avoid undesirable root bridge election and spanning tree topology issues.

When MAC address reduction is enabled, the root bridge priority becomes a multiple of 4096 plus the VLAN ID. You can only specify a device bridge ID (used by the spanning tree algorithm to determine the identity of the root bridge; the lowest number is preferred) as a multiple of 4096. Only the following values are possible:

- 0
- 4096
- 8192

- 12288
- 16384
- 20480
- 24576
- 28672
- 32768
- 36864
- 40960
- 45056
- 49152
- 53248
- 57344
- 61440

STP uses the extended system ID plus a MAC address to make the bridge ID unique for each VLAN.

Note If another bridge in the same spanning tree domain does not run the MAC address reduction feature, it could win the root bridge ownership because of the finer granularity in the selection of its bridge ID.

BPDUs

Network devices transmit BPDUs throughout the STP instance. Each network device sends configuration BPDUs to communicate and compute the spanning tree topology. Each configuration BPDU contains the following minimal information:

- The unique bridge ID of the network device that the transmitting network device believes to be the root bridge
- The STP path cost to the root
- The bridge ID of the transmitting bridge
- The message age
- The identifier of the transmitting port
- Values for the hello, forward delay, and max-age protocol timer
- Additional information for STP extension protocols

When a network device transmits a Rapid PVST+ BPDU frame, all network devices connected to the VLAN on which the frame is transmitted receive the BPDU. When a network device receives a BPDU, it does not forward the frame but instead uses the information in the frame to calculate a BPDU. If the topology changes, the device initiates a BPDU exchange.

A BPDU exchange results in the following:

- One network device is elected as the root bridge.
- The shortest distance to the root bridge is calculated for each network device based on the path cost.
- A designated bridge for each LAN segment is selected. This network device is closest to the root bridge through which frames are forwarded to the root.
- A root port is elected. This port provides the best path from the bridge to the root bridge.
- Ports included in the spanning tree are selected.

Election of the Root Bridge

For each VLAN, the network device with the highest bridge ID (that is, the lowest numerical ID value) is elected as the root bridge. If all network devices are configured with the default priority (32768), the network device with the lowest MAC address in the VLAN becomes the root bridge. The bridge priority value occupies the most significant bits of the bridge ID.

When you change the bridge priority value, you change the probability that the device will be elected as the root bridge. Configuring a lower value increases the probability; a higher value decreases the probability.

The STP root bridge is the logical center of each spanning tree topology in a Layer 2 network. All paths that are not needed to reach the root bridge from anywhere in the Layer 2 network are placed in STP blocking mode.

BPDUs contain information about the transmitting bridge and its ports, including bridge and MAC addresses, bridge priority, port priority, and path cost. STP uses this information to elect the root bridge for the STP instance, to elect the root port that leads to the root bridge, and to determine the designated port for each Layer 2 segment.

Creating the Spanning Tree Topology

By increasing the priority (lowering the numerical value) of the ideal network device so that it becomes the root bridge, you force an STP recalculation to form a new spanning tree topology with the ideal network device as the root.

Figure 6: Spanning Tree Topology

In this figure, switch A is elected as the root bridge because the bridge priority of all the network devices is set to the default (32768) and switch A has the lowest MAC address. However, due to traffic patterns, the number of forwarding ports or link types, switch A might not be the ideal root bridge.

When the spanning tree topology is calculated based on default parameters, the path between the source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to a port that has a higher number than the current root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that one port on switch B is a fiber-optic link, and another port on switch B (an unshielded twisted-pair [UTP] link) is the root port. Network traffic might be more efficient over the high-speed fiber-optic link. By changing the STP port priority on the fiber-optic port to a higher priority (lower numerical value) than the root port, the fiber-optic port becomes the new root port.

Rapid PVST+

Rapid PVST+ is the default spanning tree mode for the software and is enabled by default on the default VLAN and all newly created VLANs.

A single instance, or topology, of RSTP runs on each configured VLAN, and each Rapid PVST+ instance on a VLAN has a single root device. You can enable and disable STP on a per-VLAN basis when you are running Rapid PVST+.

Overview of Rapid PVST+

Rapid PVST+ is the IEEE 802.1w (RSTP) standard implemented per VLAN. A single instance of STP runs on each configured VLAN (if you do not manually disable STP). Each Rapid PVST+ instance on a VLAN has a single root switch. You can enable and disable STP on a per-VLAN basis when you are running Rapid PVST+.

Note Rapid PVST+ is the default STP mode for the device.

Rapid PVST+ uses point-to-point wiring to provide rapid convergence of the spanning tree. The spanning tree reconfiguration can occur in less than 1 second with Rapid PVST+ (in contrast to 50 seconds with the default settings in the 802.1D STP). The device automatically checks the PVID.

Note Rapid PVST+ supports one STP instance for each VLAN.

Using Rapid PVST+, STP convergence occurs rapidly. By default, each designated port in the STP sends out a BPDU every 2 seconds. On a designated port in the topology, if hello messages are missed three consecutive times, or if the maximum age expires, the port immediately flushes all protocol information in the table. A port considers that it loses connectivity to its direct neighbor designated port if it misses three BPDUs or if the maximum age expires. This rapid aging of the protocol information allows quick failure detection.

Rapid PVST+ provides for rapid recovery of connectivity following the failure of a device, a device port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- Edge ports—When you configure a port as an edge port on an RSTP device, the edge port immediately transitions to the forwarding state. (This immediate transition was previously a Cisco-proprietary feature named PortFast.) You should only configure ports that connect to a single end station as edge ports. Edge ports do not generate topology changes when the link changes.

Enter the **spanning-tree port type** interface configuration command to configure a port as an STP edge port.

Note We recommend that you configure all ports connected to a Layer 2 host as edge ports.

- Root port—If Rapid PVST+ selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.

- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

Rapid PVST+ achieves rapid transition to the forwarding state only on edge ports and point-to-point links. Although the link type is configurable, the system automatically derives the link type information from the duplex setting of the port. Full-duplex ports are assumed to be point-to-point ports, while half-duplex ports are assumed to be shared ports.

Edge ports do not generate topology changes, but all other designated and root ports generate a topology change (TC) BPDU when they either fail to receive three consecutive BPDUs from the directly connected neighbor or the maximum age times out. At this point, the designated or root port sends a BPDU with the TC flag set. The BPDUs continue to set the TC flag as long as the TC While timer runs on that port. The value of the TC While timer is the value set for the hello time plus 1 second. The initial detector of the topology change immediately floods this information throughout the entire topology.

When Rapid PVST+ detects a topology change, the protocol does the following:

- Starts the TC While timer with a value equal to twice the hello time for all the nonedge root and designated ports, if necessary.
- Flushes the MAC addresses associated with all these ports.

The topology change notification floods quickly across the entire topology. The system flushes dynamic entries immediately on a per-port basis when it receives a topology change.

Note The TCA flag is used only when the device is interacting with devices that are running legacy 802.1D STP.

The proposal and agreement sequence then quickly propagates toward the edge of the network and quickly restores connectivity after a topology change.

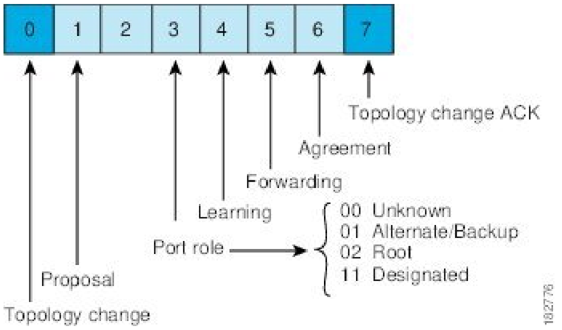
Rapid PVST+ BPDUs

Rapid PVST+ and 802.1w use all six bits of the flag byte to add the following:

- The role and state of the port that originates the BPDU
- The proposal and agreement handshake

Figure 7: Rapid PVST+ Flag Byte in BPDU

This figure shows the use of the BPDU flags in Rapid PVST+.



Another important change is that the Rapid PVST+ BPDU is type 2, version 2, which makes it possible for the device to detect connected legacy (802.1D) bridges. The BPDU for 802.1D is type 0, version 0.

Proposal and Agreement Handshake

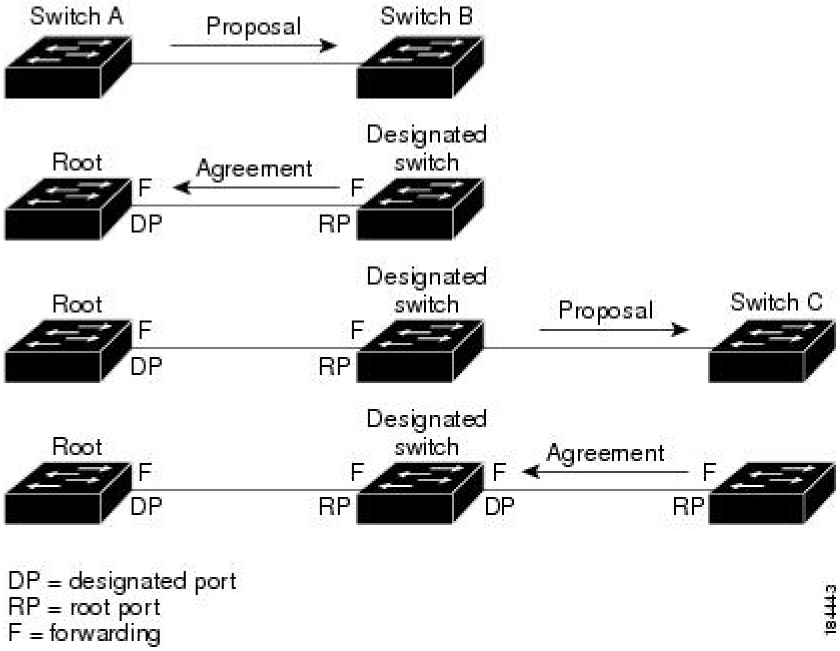
Figure 8: Proposal and Agreement Handshaking for Rapid Convergence

In this figure, switch A is connected to switch B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of switch A is a smaller numerical value than the priority of switch B. Switch A sends a proposal message (a configuration BPDU with the proposal flag set) to switch B, proposing itself as the designated switch.

After receiving the proposal message, switch B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

After receiving the agreement message from switch B, switch A also immediately transitions its designated port to the forwarding state. No loops in the network can form because switch B blocked all of its nonedge ports and because there is a point-to-point link between switches A and B.

When switch C connects to switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to switch B as its root port, and both ends of the link immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree as shown in this figure.



The switch learns the link type from the port duplex mode; a full-duplex port is considered to have a point-to-point connection and a half-duplex port is considered to have a shared connection. You can override the default setting that is controlled by the duplex setting by entering the **spanning-tree link-type** interface configuration command.

This proposal/agreement handshake is initiated only when a nonedge port moves from the blocking to the forwarding state. The handshaking process then proliferates step-by-step throughout the topology.

Protocol Timers

This table describes the protocol timers that affect the Rapid PVST+ performance.

Table 12: Rapid PVST+ Protocol Timers

Variable	Description
Hello timer	Determines how often each device broadcasts BPDUs to other network devices. The default is 2 seconds, and the range is from 1 to 10.
Forward delay timer	Determines how long each of the listening and learning states last before the port begins forwarding. This timer is generally not used by the protocol, but it is used when interoperating with the 802.1D spanning tree. The default is 15 seconds, and the range is from 4 to 30 seconds.
Maximum age timer	Determines the amount of time that protocol information received on a port is stored by the network device. This timer is generally not used by the protocol, but it is used when interoperating with the 802.1D spanning tree. The default is 20 seconds; the range is from 6 to 40 seconds.

Port Roles

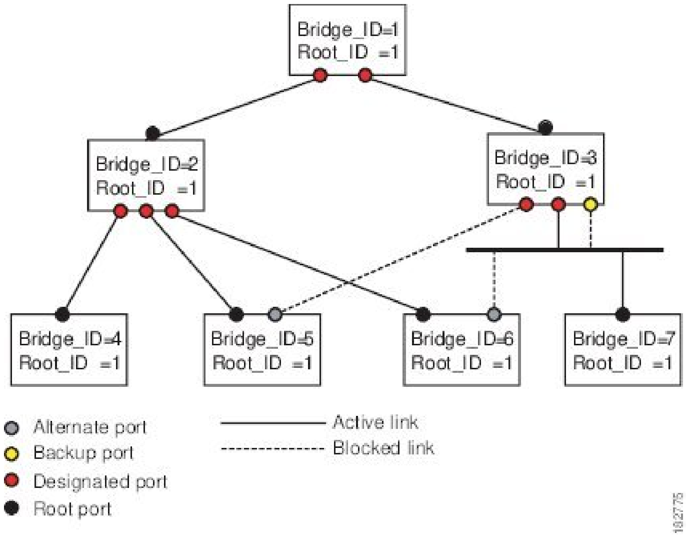
Rapid PVST+ provides rapid convergence of the spanning tree by assigning port roles and learning the active topology. Rapid PVST+ builds upon the 802.1D STP to select the device with the highest switch priority (lowest numerical priority value) as the root bridge. Rapid PVST+ assigns one of these port roles to individual ports:

- Root port—Provides the best path (lowest cost) when the device forwards packets to the root bridge.
- Designated port—Connects to the designated device that has the lowest path cost when forwarding packets from that LAN to the root bridge. The port through which the designated device is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root bridge to the path provided by the current root port. An alternate port provides a path to another device in the topology.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected in a loopback by a point-to-point link or when a device has two or more connections to a shared LAN segment. A backup port provides another path in the topology to the device.
- Disabled port—Has no role within the operation of the spanning tree.

In a stable topology with consistent port roles throughout the network, Rapid PVST+ ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the blocking state. Designated ports start in the blocking state. The port state controls the operation of the forwarding and learning processes.

Figure 9: Sample Topology Demonstrating Port Roles

This figure shows port roles. A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.



Rapid PVST+ Port State Overview

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When a Layer 2 LAN port transitions directly from nonparticipation in the spanning tree topology to the forwarding state, it can create temporary data loops. Ports must wait for new topology information to propagate through the switched LAN before starting to forward frames.

Each Layer 2 LAN port on the device that uses Rapid PVST+ or MST exists in one of the following four states:

- Blocking—The Layer 2 LAN port does not participate in frame forwarding.
- Learning—The Layer 2 LAN port prepares to participate in frame forwarding.
- Forwarding—The Layer 2 LAN port forwards frames.
- Disabled—The Layer 2 LAN port does not participate in STP and is not forwarding frames.

When you enable Rapid PVST+, every port in the device, VLAN, and network goes through the blocking state and the transitory states of learning at power up. If properly configured, each Layer 2 LAN port stabilizes to the forwarding or blocking state.

When the STP algorithm places a Layer 2 LAN port in the forwarding state, the following process occurs:

1. The Layer 2 LAN port is put into the blocking state while it waits for protocol information that suggests it should go to the learning state.
2. The Layer 2 LAN port waits for the forward delay timer to expire, moves the Layer 2 LAN port to the learning state, and restarts the forward delay timer.
3. In the learning state, the Layer 2 LAN port continues to block frame forwarding as it learns the end station location information for the forwarding database.
4. The Layer 2 LAN port waits for the forward delay timer to expire and then moves the Layer 2 LAN port to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 LAN port in the blocking state does not participate in frame forwarding.

A Layer 2 LAN port in the blocking state performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another port for forwarding.
- Does not incorporate the end station location into its address database. (There is no learning on a blocking Layer 2 LAN port, so there is no address database update.)
- Receives BPDUs and directs them to the system module.
- Receives, processes, and transmits BPDUs received from the system module.
- Receives and responds to control plane messages.

Learning State

A Layer 2 LAN port in the learning state prepares to participate in frame forwarding by learning the MAC addresses for the frames. The Layer 2 LAN port enters the learning state from the blocking state.

A Layer 2 LAN port in the learning state performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another port for forwarding.
- Incorporates the end station location into its address database.
- Receives BPDUs and directs them to the system module.
- Receives, processes, and transmits BPDUs received from the system module.
- Receives and responds to control plane messages.

Forwarding State

A Layer 2 LAN port in the forwarding state forwards frames. The Layer 2 LAN port enters the forwarding state from the learning state.

A Layer 2 LAN port in the forwarding state performs as follows:

- Forwards frames received from the attached segment.
- Forwards frames switched from another port for forwarding.
- Incorporates the end station location information into its address database.
- Receives BPDUs and directs them to the system module.
- Processes BPDUs received from the system module.
- Receives and responds to control plane messages.

Disabled State

A Layer 2 LAN port in the disabled state does not participate in frame forwarding or STP. A Layer 2 LAN port in the disabled state is virtually nonoperational.

A disabled Layer 2 LAN port performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another port for forwarding.
- Does not incorporate the end station location into its address database. (There is no learning, so there is no address database update.)
- Does not receive BPDUs from neighbors.
- Does not receive BPDUs for transmission from the system module.

Summary of Port States

This table lists the possible operational and Rapid PVST+ states for ports and whether the port is included in the active topology.

Table 13: Port State Active Topology

Operational Status	Port State	Is Port Included in the Active Topology?
Enabled	Blocking	No
Enabled	Learning	Yes
Enabled	Forwarding	Yes
Disabled	Disabled	No

Synchronization of Port Roles

When the device receives a proposal message on one of its ports and that port is selected as the new root port, Rapid PVST+ forces all other ports to synchronize with the new root information.

The device is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the device is synchronized if either of the following applies:

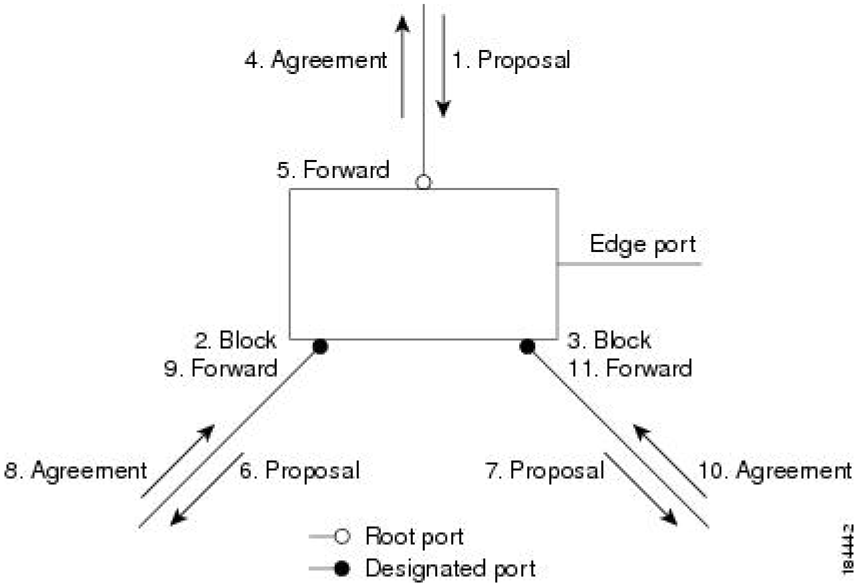
- That port is in the blocking state.
- It is an edge port (a port configured to be at the edge of the network).

If a designated port is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the Rapid PVST+ forces it to synchronize with new root information. In general, when the Rapid PVST+ forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

After ensuring that all of the ports are synchronized, the device sends an agreement message to the designated device that corresponds to its root port. When the devices connected by a point-to-point link are in agreement about their port roles, Rapid PVST+ immediately transitions the port states to the forwarding state.

Figure 10: Sequence of Events During Rapid Convergence

This figure shows the sequence of events during synchronization.



Processing Superior BPDU Information

A superior BPDU is a BPDU with root information (such as a lower switch ID or lower path cost) that is superior to what is currently stored for the port.

If a port receives a superior BPDU, Rapid PVST+ triggers a reconfiguration. If the port is proposed and is selected as the new root port, Rapid PVST+ forces all the designated, nonedge ports to synchronize.

If the received BPDU is a Rapid PVST+ BPDU with the proposal flag set, the device sends an agreement message after all of the other ports are synchronized. The new root port transitions to the forwarding state as soon as the previous port reaches the blocking state.

If the superior information received on the port causes the port to become a backup port or an alternate port, Rapid PVST+ sets the port to the blocking state and sends an agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires. At that time, the port transitions to the forwarding state.

Processing Inferior BPDU Information

An inferior BPDU is a BPDU with root information (such as a higher switch ID or higher path cost) that is inferior to what is currently stored for the port.

If a designated port receives an inferior BPDU, it immediately replies with its own information.

Detecting Unidirectional Link Failure:Rapid PVST+

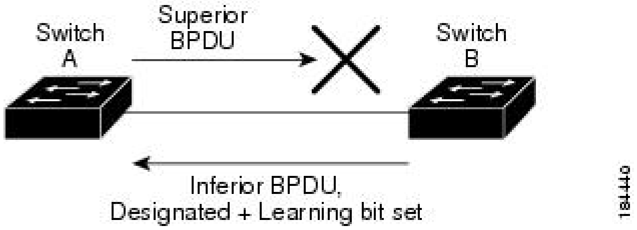
The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops using the Unidirectional Link Detection (UDLD) feature. This feature is based on the dispute mechanism.

See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on UDLD.

When a designated port detects a conflict, it keeps its role, but reverts to a discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 11: Detecting Unidirectional Link Failure

This figure illustrates a unidirectional link failure that typically creates a bridging loop. Switch A is the root bridge, and its BPDUs are lost on the link leading to switch B. The 802.1w-standard BPDUs include the role and state of the sending port. With this information, switch A can detect that switch B does not react to the superior BPDUs that it sends and that switch B is the designated, not the root port. As a result, switch A blocks (or keeps blocking) its port, which prevents the bridging loop.



Port Cost

Note Rapid PVST+ uses the short (16-bit) path-cost method to calculate the cost by default. With the short path-cost method, you can assign any value in the range of 1 to 65535. However, you can configure the device to use the long (32-bit) path-cost method, which allows you to assign any value in the range of 1 to 200,000,000. You configure the path-cost calculation method globally.

This table shows how the STP port path-cost default value is determined from the media speed and path-cost calculation method of a LAN interface.

Table 14: Default Port Cost

Bandwidth	Short Path-Cost Method of Port Cost	Long Path-Cost Method of Port Cost
10 Mbps	100	2,000,000
100 Mbps	19	200,000
1 Gigabit Ethernet	4	20,000
10 Gigabit Ethernet	2	2,000

If a loop occurs, STP considers the port cost when selecting a LAN interface to put into the forwarding state.

You can assign the lower cost values to LAN interfaces that you want STP to select first and higher cost values to LAN interfaces that you want STP to select last. If all LAN interfaces have the same cost value, STP puts the LAN interface with the lowest LAN interface number in the forwarding state and blocks other LAN interfaces.

On access ports, you assign the port cost by the port. On trunk ports, you assign the port cost by the VLAN; you can configure the same port cost to all the VLANs on a trunk port.

Port Priority

If a redundant path occurs and multiple ports have the same path cost, Rapid PVST+ considers the port priority when selecting which LAN port to put into the forwarding state. You can assign lower priority values to LAN ports that you want Rapid PVST+ to select first and higher priority values to LAN ports that you want Rapid PVST+ to select last.

If all LAN ports have the same priority value, Rapid PVST+ puts the LAN port with the lowest LAN port number in the forwarding state and blocks other LAN ports. The possible priority range is from 0 through 224 (the default is 128), configurable in increments of 32. The device uses the port priority value when the LAN port is configured as an access port and uses the VLAN port priority values when the LAN port is configured as a trunk port.

Rapid PVST+ and IEEE 802.1Q Trunks

The 802.1Q trunks impose some limitations on the STP strategy for a network. In a network of Cisco network devices connected through 802.1Q trunks, the network devices maintain one instance of STP for each VLAN allowed on the trunks. However, non-Cisco 802.1Q network devices maintain only one instance of STP for all VLANs allowed on the trunks, which is the Common Spanning Tree (CST).

When you connect a Cisco network device to a non-Cisco device through an 802.1Q trunk, the Cisco network device combines the STP instance of the 802.1Q VLAN of the trunk with the STP instance of the non-Cisco 802.1Q network device. However, all per-VLAN STP information that is maintained by Cisco network devices is separated by a cloud of non-Cisco 802.1Q network devices. The non-Cisco 802.1Q cloud that separates the Cisco network devices is treated as a single trunk link between the network devices.

For more information on 802.1Q trunks, see the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*.

Rapid PVST+ Interoperation with Legacy 802.1D STP

Rapid PVST+ can interoperate with devices that are running the legacy 802.1D protocol. The device knows that it is interoperating with equipment running 802.1D when it receives a BPDU version 0. The BPDUs for Rapid PVST+ are version 2. If the BPDU received is an 802.1w BPDU version 2 with the proposal flag set, the device sends an agreement message after all of the other ports are synchronized. If the BPDU is an 802.1D BPDU version 0, the device does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

The device interoperates with legacy 802.1D devices as follows:

- Notification—Unlike 802.1D BPDUs, 802.1w does not use TCN BPDUs. However, for interoperability with 802.1D devices, the device processes and generates TCN BPDUs.
- Acknowledgment—When an 802.1w device receives a TCN message on a designated port from an 802.1D device, it replies with an 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the TC timer in 802.1D) is active on a root port connected to an 802.1D device and a configuration BPDU with the TCA set is received, the TC-while timer is reset.

  This method of operation is required only for 802.1D devices. The 802.1w BPDUs do not have the TCA bit set.
- Protocol migration—For backward compatibility with 802.1D devices, 802.1w selectively sends 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which 802.1w BPDUs are sent), and 802.1w BPDUs are sent. While this timer is active, the device processes all BPDUs received on that port and ignores the protocol type.

If the device receives an 802.1D BPDU after the port migration-delay timer has expired, it assumes that it is connected to an 802.1D device and starts using only 802.1D BPDUs. However, if the 802.1w device is using 802.1D BPDUs on a port and receives an 802.1w BPDU after the timer has expired, it restarts the timer and starts using 802.1w BPDUs on that port.

Note If you want all devices on the same LAN segment to reinitialize the protocol on each interface, you must reinitialize Rapid PVST+.

Rapid PVST+ Interoperation with 802.1s MST

Rapid PVST+ interoperates seamlessly with the IEEE 802.1s Multiple Spanning Tree (MST) standard. No user configuration is needed. To disable this seamless interoperation, you can use PVST Simulation.

High Availability for Rapid PVST+

The software supports high availability for Rapid PVST+. However, the statistics and timers are not restored when Rapid PVST+ restarts. The timers start again and the statistics begin from 0.

Note See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on high-availability features.

Virtualization Support for Rapid PVST+

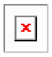
Note See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on virtual device contexts (VDCs) and assigning resources.

Figure 12: Separate STP in each VDC

This figure shows how the system provides support for VDCs. Using VDCs, you have a separate Layer 2 virtualization in each VDC, and each VDC runs a separate STP.

Each VDC will have its own Rapid PVST+. You cannot configure Rapid PVST+ across VDCs with Cisco NX-OS software. However, you can run Rapid PVST+ in one VDC and run MST in each VDC.

For example, VDC1 can run MST, VDC2 can run Rapid PVST+, and VDC3 can run MST.



Ensure that you are in the correct VDC before you begin configuring either Rapid PVST+ parameters.

Prerequisites for Configuring Rapid PVST+

Rapid PVST+ has the following prerequisites:

- You must be logged onto the device.
- If you are working in another VDC than the default VDC, that VDC must be created already.

Guidelines and Limitations for Configuring Rapid PVST+

Rapid PVST+ has the following configuration guidelines and limitations:

- There is a total of 4000 Rapid PVST+ for each VDC.
- The maximum number of VLANs and ports is 16,000.
- Only Rapid PVST+ or MST can be active at any time for each VDC.
- Port channeling—The port-channel bundle is considered as a single port. The port cost is the aggregation of all the configured port costs assigned to that channel.
- For private VLANs, on a normal VLAN trunk port, the primary and secondary private VLANs are two different logical ports and must have the exact STP topology. On access ports, STP sees only the primary VLAN.
- We recommend that you configure all ports connected to Layer 2 hosts as STP edge ports.
- Always leave STP enabled.
- Do not change timers because changing timers can adversely affect stability.
- Keep user traffic off the management VLAN; keep the management VLAN separate from the user data.
- Choose the distribution and core layers as the location of the primary and secondary root switches.
- When you connect two Cisco devices through 802.1Q trunks, the switches exchange spanning tree BPDUs on each VLAN allowed on the trunks. The BPDUs on the native VLAN of the trunk are sent untagged to the reserved 802.1D spanning tree multicast MAC address (01-80-C2-00-00-00). The BPDUs on all VLANs on the trunk are sent tagged to the reserved Cisco Shared Spanning Tree Protocol (SSTP) multicast MAC address (01-00-0c-cc-cc-cd).

Default Settings for Rapid PVST+

This table lists the default settings for Rapid PVST+ parameters.

Table 15: Default Rapid PVST+ Parameters

Parameters	Default
Spanning Tree	Enabled on all VLANs.

Parameters	Default
Spanning Tree mode	Rapid PVST+ **Caution** Changing the spanning tree mode disrupts the traffic because all spanning tree instances are stopped for the previous mode and started for the new mode.
VLAN	All ports assigned to VLAN1.
Extended system ID	Always enabled.
MAC address reduction	Always enabled.
Bridge ID priority	32769 (default bridge priority plus system ID extension of default VLAN1).
Port state	Blocking (changes immediately after convergence).
Port role	Designated (changes after convergence).
Port/VLAN priority	128.
Path-cost calculation method	Short.
Port/VLAN cost	Auto The default port cost is determined by the media speed and path-cost method calculation, as follows: • 10 Mbps: • short: 100 • long: 2,000,000 • 100 Mbps: • short: 19 • long: 200,000 • 1 Gigabit Ethernet: • short: 4 • long: 20,000 • 10 Gigabit Ethernet: • short: 2 • long: 2,000
Hello time	2 seconds.
Forward delay time	15 seconds.

Parameters	Default
Maximum aging time	20 seconds.
Link type	Auto The default link type is determined by the duplex, as follows: • Full duplex: point-to-point link • Half duplex: shared link

Configuring Rapid PVST+

Rapid PVST+, which has the 802.1 w standard applied to the PVST+ protocol, is the default STP setting in the device.

You enable Rapid PVST+ on a per-VLAN basis. The device maintains a separate instance of STP for each VLAN (except on those VLANS on which you disable STP). By default, Rapid PVST+ is enabled on the default VLAN and on each VLAN that you create.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Guidelines for Configuring Rapid PVST+

Ensure that you are in the correct VDC (see the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*).

Enabling Rapid PVST+—CLI Version

If you disable Rapid PVST+ on any VLANs, you must reenable Rapid PVRST+ on the specified VLANs. If you have enabled MST on the device and now want to use Rapid PVST+, you must enable Rapid PVST+ on the device.

Rapid PVST+ is the default STP mode. You cannot simultaneously run MST and Rapid PVST+ in the same VDC.

However, one VDC can run Rapid PVST+ and a different VDC can run MST.

Note When you change the spanning tree mode, traffic is disrupted because all spanning tree instances are stopped for the previous mode and started for the new mode.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mode rapid-pvst** **Example:** `switch(config)# spanning-tree mode rapid-pvst`	Enables Rapid PVST+ on the device. Rapid PVST+ is the default spanning tree mode. **Note** Changing the spanning tree mode disrupts traffic because all spanning tree instances are stopped for the previous mode and started for the new mode.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show running-config spanning-tree all** **Example:** `switch# show running-config spanning-tree all`	Displays information about the currently running STP configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to enable Rapid PVST+ on the device:

```
switch# config t
switch(config)# spanning-tree mode rapid-pvst
switch(config)# exit
switch#
```

Note Because Rapid PVST+ is enabled by default, entering the **show running** command to view the resulting configuration does not display the command that you entered to enable Rapid PVST+.

Disabling or Enabling Rapid PVST+ Per VLAN—CLI Version

You can enable or disable Rapid PVST+ on each VLAN.

Note Rapid PVST+ is enabled by default on the default VLAN and on all VLANs that you create.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td><b>config t</b>
<b>Example:</b>
<code>switch# config t</code>
<code>switch(config)#</code></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>spanning-tree vlan</b> <i>vlan-range</i></td><td>Enables Rapid PVST+ (default STP) on a per VLAN basis. The <i>vlan-range</i> value can be 2 through 4094 except for reserved VLAN values.</td></tr>
<tr><td><b>no spanning-tree vlan</b> <i>vlan-range</i></td><td>Disables Rapid PVST+ on the specified VLAN. See the Caution for information regarding this command.</td></tr>
</table>
<b>Example:</b>
<code>switch(config)# spanning-tree vlan 5</code></td><td></td></tr>
<tr><td><b>Step 3</b></td><td><b>exit</b>
<b>Example:</b>
<code>switch(config)# exit</code>
<code>switch#</code></td><td>Exits global configuration mode.</td></tr>
<tr><td><b>Step 4</b></td><td>(Optional) <b>show spanning-tree</b>
<b>Example:</b>
<code>switch# show spanning-tree</code></td><td>Displays the STP configuration.</td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b></td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

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

Example

This example shows how to enable STP on VLAN 5:

```
switch# config t
switch(config)# spanning-tree vlan 5
switch(config)# exit
switch#
```

Note Do not disable spanning tree on a VLAN unless all switches and bridges in the VLAN have spanning tree disabled. You cannot disable spanning tree on some switches and bridges in a VLAN and leave it enabled on other switches and bridges in the VLAN. This action can have unexpected results because switches and bridges with spanning tree enabled will have incomplete information regarding the physical topology of the network.

Caution We do not recommend disabling spanning tree even in a topology that is free of physical loops. Spanning tree serves as a safeguard against misconfigurations and cabling errors. Do not disable spanning tree in a VLAN without ensuring that no physical loops are present in the VLAN.

Note Because STP is enabled by default, entering the **show running** command to view the resulting configuration does not display the command that you entered to enable STP.

Configuring the Root Bridge ID

The device maintains a separate instance of STP for each active VLAN in Rapid PVST+. For each VLAN, the network device with the lowest bridge ID becomes the root bridge for that VLAN.

To configure a VLAN instance to become the root bridge, modify the bridge priority from the default value (32768) to a significantly lower value.

When you enter the **spanning-tree vlan** *vlan_ID* **primary root** command, the device checks the bridge priority of the current root bridges for each VLAN. The device sets the bridge priority for the specified VLANs to 24576 if this value will cause the device to become the root for the specified VLANs. If any root bridge for the specified VLAN has a bridge priority lower than 24576, the device sets the bridge priority for the specified VLANs to 4096 less than the lowest bridge priority.

Note The **spanning-tree vlan** *vlan_ID* **primary root** command fails if the value required to be the root bridge is less than 4096. If the software cannot lower the bridge priority any lower, the device returns the following message:

```
Error: Failed to set root bridge for VLAN 1
It may be possible to make the bridge root by setting the priority
for some (or all) of these instances to zero.
```

Caution The root bridge for each instance of STP should be a backbone or distribution device. Do not configure an access device as the STP primary root.

Enter the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of bridge hops between any two end stations in the Layer 2 network). When you specify the network diameter, the software automatically selects an optimal hello time, forward delay time, and maximum age time for a network of that diameter, which can significantly reduce the STP convergence time. You can enter the **hello-time** keyword to override the automatically calculated hello time.

Note With the device configured as the root bridge, do not manually configure the hello time, forward-delay time, and maximum-age time using the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and **spanning-tree mst max-age** global configuration commands.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree vlan** *vlan-range* **root primary** [**diameter** *dia* [**hello-time** *hello-time*]] **Example:** `switch(config)# spanning-tree vlan 5 root primary diameter 4`	Configures a device as the primary root bridge. The *vlan-range* value can be 2 through 4094 (except for reserved VLAN values.) The *dia* default is 7. The *hello-time* can be from 1 to 10 seconds, and the default value is 2 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

	Command or Action	Purpose
Step 4	(Optional) **show spanning-tree** **Example:** `switch# show spanning-tree`	Displays the STP configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the device as the root bridge for VLAN 5 with a network diameter of 4:

```
switch# config t
switch(config)# spanning-tree vlan 5 root primary
diameter 4
switch(config)# exit
switch#
```

Configuring a Secondary Root Bridge—CLI Version

When you configure a device as the secondary root, the STP bridge priority is modified from the default value (32768) so that the device is likely to become the root bridge for the specified VLANs if the primary root bridge fails (assuming the other network devices in the network use the default bridge priority of 32768). STP sets the bridge priority to 28672.

Enter the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of bridge hops between any two end stations in the Layer 2 network). When you specify the network diameter, the software automatically selects an optimal hello time, forward delay time, and maximum age time for a network of that diameter, which can significantly reduce the STP convergence time. You can enter the **hello-time** keyword to override the automatically calculated hello time.

You can configure more than one device in this manner to have multiple backup root bridges. Enter the same network diameter and hello time values that you used when configuring the primary root bridge.

Note With the device configured as the root bridge, do not manually configure the hello time, forward-delay time, and maximum-age time using the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and **spanning-tree mst max-age** global configuration commands.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree vlan** *vlan-range* **root secondary** [**diameter** *dia* [**hello-time** *hello-time*]] **Example:** `switch(config)# spanning-tree vlan 5 root secondary diameter 4`	Configures a device as the secondary root bridge. The *vlan-range* value can be 2 through 4094 (except for reserved VLAN values). The *dia* default is 7. The *hello-time* can be from 1 to 10 seconds, and the default value is 2 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree vlan** *vlan_id* **Example:** `switch# show spanning-tree vlan 5`	Displays the STP configuration for the specified VLANs.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the device as the secondary root bridge for VLAN 5 with a network diameter of 4:

```
switch# config t
switch(config)# spanning-tree vlan 5 root secondary diameter 4
switch(config)# exit
switch#
```

Configuring the Rapid PVST+ Bridge Priority of a VLAN

You can configure the Rapid PVST+ bridge priority of a VLAN. This is another method of configuring root bridges.

Note Be careful when using this configuration. We recommend that you configure the primary root and secondary root to modify the bridge priority.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree vlan** *vlan-range* **priority** *value* **Example:** `switch(config)# spanning-tree vlan 5 priority 8192`	Configures the bridge priority of a VLAN. Valid values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected. The default value is 32768.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree vlan** *vlan_id* **Example:** `switch# show spanning-tree vlan 5`	Displays the STP configuration for the specified VLANs.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the priority of VLAN 5 on Gigabit Ethernet port 1/4 to 8192:

```
switch# config t
switch(config)# spanning-tree vlan 5 priority 8192
switch(config)# exit
switch#
```

Configuring the Rapid PVST+ Port Priority—CLI Version

You can assign lower priority values to LAN ports that you want Rapid PVST+ to select first and higher priority values to LAN ports that you want Rapid PVST+ to select last. If all LAN ports have the same priority value, Rapid PVST+ puts the LAN port with the lowest LAN port number in the forwarding state and blocks other LAN ports.

The device uses the port priority value when the LAN port is configured as an access port and uses the VLAN port priority values when the LAN port is configured as a trunk port.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** *type slot/port*
	Example:
	`switch(config)# interface ethernet 1/4`
	`switch(config-if)#`
		Specifies the interface to configure and enters interface configuration mode.

Step 3	**spanning-tree** [**vlan** *vlan-list*] **port-priority** *priority* **Example:** `switch(config-if)# spanning-tree port-priority 160`	Configures the port priority for the LAN interface. The *priority* value can be from 0 to 224. A lower value indicates a higher priority. The priority values are 0, 32, 64, 96, 128, 160, 192, and 224. All other values are rejected. The default value is 128.

Step 4	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 5	(Optional) **show spanning-tree interface** {**ethernet** *slot/port* | *port channel channel-number*}
	Example:
	`switch# show spanning-tree interface ethernet 2/10`
		Displays the STP configuration for the specified interface.

Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the port priority of Ethernet access port 1/4 to 160:

```
switch# config t
switch (config)# interface ethernet 1/4
switch(config-if)# spanning-tree port-priority 160
switch(config-if)# exit
switch(config)#
```

Configuring the Rapid PVST+ Path-Cost Method and Port Cost—CLI Version

On access ports, you can assign the port cost for each port. On trunk ports, you can assign the port cost for each VLAN; you can configure all the VLANs on a trunk with the same port cost.

Note In Rapid PVST+ mode, you can use either the short or long path-cost method, and you can configure the method in either the interface or configuration submode. The default path-cost method is short.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree pathcost method** {**long** | **short**}
	Example:
	`switch(config)# spanning-tree pathcost method long`
		Selects the method used for Rapid PVST+ path-cost calculations. The default method is the short method.

Step 3	**interface** *type slot/port*
	Example:
	`switch(config)# interface ethernet 1/4`
	`switch(config-if)`
		Specifies the interface to configure and enters the interface configuration mode.

Step 4	**spanning-tree** [**vlan** *vlan-id*] **cost** [*value* | *auto*] **Example:** `switch(config-if)# spanning-tree cost 1000`	Configures the port cost for the LAN interface. The cost value, depending on the path-cost calculation method, can be as follows: • short—1 to 65535 • long—1 to 200000000 **Note** You configure this parameter per port on access ports and per VLAN on trunk ports.

	Command or Action	Purpose
		The default is **auto**, which sets the port cost on both the path-cost calculation method and the media speed.

Step 5	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 6	(Optional) **show spanning-tree pathcost method** **Example:** `switch# show spanning-tree pathcost method`	Displays the STP path-cost method.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the port cost of Ethernet access port 1/4 to 1000:

```
switch# config t
switch (config)# spanning-tree pathcost method long
switch (config)# interface ethernet 1/4
switch(config-if)# spanning-tree cost 1000
switch(config-if)# exit
switch(config)#
```

Configuring the Rapid PVST+ Hello Time for a VLAN—CLI Version

You can configure the Rapid-PVST+ hello time for a VLAN.

Note Be careful when using this configuration because you might disrupt the Spanning Tree. For most situations, we recommend that you configure the primary root and secondary root to modify the hello time.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree vlan** *vlan-range* **hello-time** *value*
	Example:
	`switch(config)# spanning-tree vlan 5`
	`hello-time 7`
		Configures the hello time of a VLAN. The hello time value can be from 1 to 10 seconds, and the default is 2 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree vlan** *vlan_id* **Example:** `switch# show spanning-tree vlan 5`	Displays the STP configuration per VLAN.

Step 5	(Optional) **copy running-config startup-config**
	Example:
	`switch# copy running-config`
	`startup-config`
		Copies the running configuration to the startup configuration.

Example

This example shows how to configure the hello time for VLAN 5 to 7 seconds:

```
switch# config t
switch(config)# spanning-tree vlan 5 hello-time 7
switch(config)# exit
switch#
```

Configuring the Rapid PVST+ Forward Delay Time for a VLAN—CLI Version

You can configure the forward delay time per VLAN when using Rapid PVST+.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree vlan** *vlan-range* **forward-time** *value* **Example:** `switch(config)# spanning-tree vlan 5 forward-time 21`	Configures the forward delay time of a VLAN. The forward delay time value can be from 4 to 30 seconds, and the default is 15 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree vlan** *vlan_id* **Example:** `switch# show spanning-tree vlan 5`	Displays the STP configuration per VLAN.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the forward delay time for VLAN 5 to 21 seconds:

```
switch# config t
switch(config)# spanning-tree vlan 5 forward-time 21
switch(config)# exit
switch#
```

Configuring the Rapid PVST+ Maximum Age Time for a VLAN—CLI Version

You can configure the maximum age time per VLAN when using Rapid PVST+.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree vlan** *vlan-range* **max-age** *value* **Example:** `switch(config)# spanning-tree vlan 5 max-age 36`	Configures the maximum aging time of a VLAN. The maximum aging time value can be from 6 to 40 seconds, and the default is 20 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree vlan** *vlan_id* **Example:** `switch# show spanning-tree vlan 5`	Displays the STP configuration per VLAN.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the maximum aging time for VLAN 5 to 36 seconds:

```
switch# config t
switch(config)# spanning-tree vlan 5 max-age 36
switch(config)# exit
switch#
```

Specifying the Link Type for Rapid PVST+—CLI Version

Rapid connectivity (802.1w standard) is established only on point-to-point links. By default, the link type is controlled from the duplex mode of the interface. A full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection.

If you have a half-duplex link physically connected point to point to a single port on a remote device, you can override the default setting on the link type and enable rapid transitions.

If you set the link to shared, STP falls back to 802.1D.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** *type slot/port*
	Example:
	`switch(config)# interface ethernet 1/4`
	`switch(config-if)#`
		Specifies the interface to configure and enters interface configuration mode.

Step 3	**spanning-tree link-type** {*auto* | *point-to-point* | *shared*}
	Example:
	`switch(config-if)# spanning-tree link-type point-to-point`
		Configures the link type to be either a point-to-point link or shared link. The system reads the default value from the device connection, as follows: half duplex links are shared and full-duplex links are point to point. If the link type is shared, the STP falls back to 802.1D. The default is auto, which sets the link type based on the duplex setting of the interface.

Step 4	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 5	(Optional) **show spanning-tree** **Example:** `switch# show spanning-tree`	Displays the STP configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the link type as a point-to-point link:

```
switch# config t
switch (config)# interface ethernet 1/4
switch(config-if)# spanning-tree link-type point-to-point
switch(config-if)# exit
switch(config)#
```

Reinitializing the Protocol for Rapid PVST+

A bridge that runs Rapid PVST+ can send 802.1D BPDUs on one of its ports when it is connected to a legacy bridge. However, the STP protocol migration cannot determine whether the legacy device has been removed from the link unless the legacy device is the designated switch. You can reinitialize the protocol negotiation (force the renegotiation with neighboring devices) on the entire device or on specified interfaces.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	**clear spanning-tree detected-protocol** [**interface** {**ethernet** *slot/port* | **port channel** *channel-number*}] **Example:** `switch# clear spanning-tree detected-protocol`	Reinitializes Rapid PVST+ on all interfaces on the device or specified interfaces.

Example

This example shows how to reinitialize Rapid PVST+ on the Ethernet interface on slot 2, port 8:

```
switch# clear spanning-tree detected-protocol interface ethernet 2/8
switch#
```

Verifying the Rapid PVST+ Configurations

To display Rapid PVST+ configuration information, perform one of the following tasks:

Command	Purpose
show running-config spanning-tree [**all**]	Displays STP information.
show spanning-tree summary	Displays summary STP information.
show spanning-tree detail	Displays detailed STP information.
show spanning-tree{**vlan***vlan-id* | **interface** {[**ethernet***slot/port*] | [**port-channel***channel-number*]}} [**detail**]	Displays STP information per VLAN and interface.
show spanning-tree vlan *vlan-id* **bridge**	Displays information on the STP bridge.

For information about the fields in the output from these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Displaying and Clearing Rapid PVST+ Statistics—CLI Version

To display Rapid PVRST+ configuration information, perform one of the following tasks:

Command	Purpose
clear spanning-tree counters [**interface** *type slot/port* | **vlan** *vlan-id*]	Clears the counters for STP.
show spanning-tree {**vlan** *vlan-id* | **interface** {[**ethernet** *slot/port*] | [**port-channel** *channel-number*]}} **detail**	Displays information about STP by interface or VLAN including BPDUs sent and received.

Rapid PVST+ Example Configurations

The following example shows how to configure Rapid PVST+:

```
switch# configure terminal
switch(config)# spanning-tree port type edge bpduguard default
switch(config)# spanning-tree port type edge bpdufilter default
switch(cnfig)# spanning-tree port type network default
switch(config)# spanning-tree vlan 1-10 priority 24576
switch(config)# spanning-tree vlan 1-10 hello-time 1
switch(config)# spanning-tree vlan 1-10 forward-time 9
switch(config)# spanning-tree vlan 1-10 max-age 13

switch(config)# interface Ethernet 3/1 switchport
switch(config-if)# spanning-tree port type edge
switch(config-if)# exit

switch(config)# spanning-tree port type edge
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# spanning-tree guard root
switch(config-if)# exit
switch(config)#
```

Additional References for Rapid PVST+—CLI Version

Related Documents

Related Topic	Document Title
Command reference	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*
Layer 2 interfaces	*Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
NX-OS fundamentals	*Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide*
High availability	*Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
System management	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*

Related Topic	Document Title
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
Licensing	*Cisco NX-OS Licensing Guide*
Release notes	*Cisco Nexus 7000 Series NX-OS Release Notes*

Standards

Standards	Title
IEEE 802.1Q-2006 (formerly known as IEEE 802.1s), IEEE 802.1D-2004 (formerly known as IEEE 802.1w), IEEE 802.1D, IEEE 802.1t	—

MIBs

MIBs	MIBs Link
• CISCO-STP-EXTENSION-MIB • BRIDGE-MIB	To locate and download MIBs, go to the following URL: https://cfnng.cisco.com/mibs.

Feature History for Configuring Rapid PVST+—CLI Version

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Table 16: Feature History for Configuring Rapid PVST+

Feature Name	Releases	Feature Information
No change	4.2(1)	--
No change	4.1(2)	--

CHAPTER 9

Configuring MST Using Cisco NX-OS

This chapter describes how to configure Multiple Spanning Tree (MST) on Cisco NX-OS devices.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About MST

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on creating Layer 2 interfaces.

MST, which is the IEEE 802.1s standard, allows you to assign two or more VLANs to a spanning tree instance. MST is not the default spanning tree mode; Rapid per VLAN Spanning Tree (Rapid PVST+) is the default mode. MST instances with the same name, revision number, and VLAN-to-instance mapping combine to form an MST region. The MST region appears as a single bridge to spanning tree configurations outside the region. MST forms a boundary to that interface when it receives an IEEE 802.1D Spanning Tree Protocol (STP) message from a neighboring device.

Note Spanning tree is used to refer to IEEE 802.1w and IEEE 802.1s. If the IEEE 802.1D Spanning Tree Protocol is discussed in this publication, 802.1D is stated specifically.

Note Beginning with Cisco NX-OS Release 5.x, when you are running virtual port channels (vPCs), you can configure STP for better performance. See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for more information on this feature.

MST Overview

Note You must enable MST; Rapid PVST+ is the default spanning tree mode.

MST maps multiple VLANs into a spanning tree instance, with each instance having a spanning tree topology independent of other spanning tree instances. This architecture provides multiple forwarding paths for data traffic, enables load balancing, and reduces the number of STP instances required to support a large number of VLANs. MST improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).

MST provides rapid convergence through explicit handshaking because each MST instance uses the IEEE 802.1w standard, which eliminates the 802.1D forwarding delay and quickly transitions root bridge ports and designated ports to the forwarding state.

MAC address reduction is always enabled on the device. You cannot disable this feature.

MST improves spanning tree operation and maintains backward compatibility with these STP versions:

- Original 802.1D spanning tree
- Rapid per-VLAN spanning tree (Rapid PVST+)

Note

- IEEE 802.1 was defined in the Rapid Spanning Tree Protocol (RSTP) and was incorporated into IEEE 802.1D.
- IEEE 802.1 was defined in MST and was incorporated into IEEE 802.1Q.

MST Regions

To allow devices to participate in MST instances, you must consistently configure the devices with the same MST configuration information.

A collection of interconnected devices that have the same MST configuration is an MST region. An MST region is a linked group of MST bridges with the same MST configuration.

The MST configuration controls the MST region to which each device belongs. The configuration includes the name of the region, the revision number, and the VLAN-to-MST instance assignment mapping.

A region can have one or multiple members with the same MST configuration. Each member must be capable of processing 802.1w bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network.

Each device can support up to 65 MST instances (MSTIs), including Instance 0, in a single MST region. Instances are identified by any number in the range from 1 to 4094. The system reserves Instance 0 for a special instance, which is the IST. You can assign a VLAN to only one MST instance at a time.

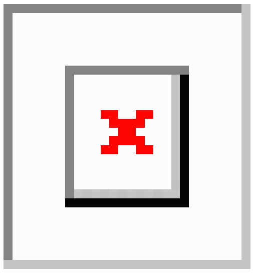
The MST region appears as a single bridge to adjacent MST regions and to other Rapid PVST+ regions and 802.1D spanning tree protocols.

Note We do not recommend that you partition the network into a large number of regions.

MST BPDUs

Each device has only one MST BPDU per interface, and that BPDU carries an M-record for each MSTI on the device. Only the IST sends BPDUs for the MST region; all M-records are encapsulated in that one BPDU that the IST sends. Because the MST BPDU carries information for all instances, the number of BPDUs that need to be processed to support MST is significantly reduced compared with Rapid PVST+.

Figure 13: MST BPDU with M-Records for MSTIs



MST Configuration Information

The MST configuration that must be identical on all devices within a single MST region is configured by the user.

You can configure the three parameters of the MST configuration as follows:

- Name—32-character string, null padded and null terminated, identifying the MST region
- Revision number—Unsigned 16-bit number that identifies the revision of the current MST configuration

Note You must set the revision number when required as part of the MST configuration. The revision number is not incremented automatically each time that the MST configuration is committed.

- VLAN-to-MST instance mapping—4096-element table that associates each of the potential 4094 VLANs supported in each virtual device context (VDC) to a given instance with the first (0) and last element (4095) set to 0. The value of element number X represents the instance to which VLAN X is mapped.

Note When you change the VLAN-to-MSTI mapping, the system reconverges MST.

MST BPDUs contain these three configuration parameters. An MST bridge accepts an MST BPDU into its own region only if these three configuration parameters match exactly. If one configuration attribute differs, the MST bridge considers the BPDU to be from another MST region.

IST, CIST, and CST

IST, CIST, and CST Overview

Unlike Rapid PVST+, in which all the STP instances are independent, MST establishes and maintains IST, CIST, and CST spanning trees, as follows:

- An IST is the spanning tree that runs in an MST region.

  MST establishes and maintains additional spanning trees within each MST region; these spanning trees are called multiple spanning tree instances (MSTIs).

  Instance 0 is a special instance for a region, known as the IST. The IST always exists on all ports; you cannot delete the IST, or Instance 0. By default, all VLANs are assigned to the IST. All other MST instances are numbered from 1 to 4094.

  The IST is the only STP instance that sends and receives BPDUs. All of the other MSTI information is contained in MST records (M-records), which are encapsulated within MST BPDUs.

  All MSTIs within the same region share the same protocol timers, but each MSTI has its own topology parameters, such as the root bridge ID, the root path cost, and so forth.

  An MSTI is local to the region; for example, MSTI 9 in region A is independent of MSTI 9 in region B, even if regions A and B are interconnected. Only CST information crosses region boundaries.

- The CST interconnects the MST regions and any instance of 802.1D and 802.1w STP that may be running on the network. The CST is the one STP instance for the entire bridged network and encompasses all MST regions and 802.1w and 802.1D instances.

- A CIST is a collection of the ISTs in each MST region. The CIST is the same as an IST inside an MST region, and the same as a CST outside an MST region.

The spanning tree computed in an MST region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed by the spanning tree algorithm running among devices that support the 802.1w, 802.1s, and 802.1D standards. The CIST inside an MST region is the same as the CST outside a region.

Spanning Tree Operation Within an MST Region

The IST connects all the MSTdevices in a region. When the IST converges, the root of the IST becomes the CIST regional root. The CIST regional root is also the CIST root if there is only one region in the network. If the CIST root is outside the region, the protocol selects one of the MST devices at the boundary of the region as the CIST regional root.

When an MST device initializes, it sends BPDUs that identify itself as the root of the CIST and the CIST regional root, with both the path costs to the CIST root and to the CIST regional root set to zero. The device also initializes all of its MSTIs and claims to be the root for all of them. If the device receives superior MSTI root information (lower switch ID, lower path cost, and so forth) than the information that is currently stored for the port, it relinquishes its claim as the CIST regional root.

During initialization, an MST region might have many subregions, each with its own CIST regional root. As devices receive superior IST information from a neighbor in the same region, they leave their old subregions and join the new subregion that contains the true CIST regional root. This action causes all subregions to shrink except for the subregion that contains the true CIST regional root.

All devices in the MST region must agree on the same CIST regional root. Any two devices in the region will only synchronize their port roles for an MSTI if they converge to a common CIST regional root.

Spanning Tree Operations Between MST Regions

If you have multiple regions or 802.1 w or 802.1D STP instances within a network, MST establishes and maintains the CST, which includes all MST regions and all 802.1w and 802.1D STP devices in the network. The MSTIs combine with the IST at the boundary of the region to become the CST.

The IST connects all the MST devices in the region and appears as a subtree in the CIST that encompasses the entire switched domain. The root of the subtree is the CIST regional root. The MST region appears as a virtual device to adjacent STP devices and MST regions.

Figure 14: MST Regions, CIST Regional Roots, and CST Root

This figure shows a network with three MST regions and an 802.1D device (D). The CIST regional root for region 1 (A) is also the CIST root. The CIST regional root for region 2 (B) and the CIST regional root for region 3 (C) are the roots for their respective subtrees within the CIST.

Only the CST instance sends and receives BPDUs. MSTIs add their spanning tree information into the BPDUs (as M-records) to interact with neighboring devices within the same MST region and compute the final spanning tree topology. The spanning tree parameters related to the BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MSTIs. You can configure the parameters related to the spanning tree topology (for example, the switch priority, the port VLAN cost, and the port VLAN priority) on both the CST instance and the MSTI.

MST devices use Version 3 BPDUs. If the MST device falls back to 802.1D STP, the device uses only 802.1D BPDUs to communicate with 802.1D-only devices. MST devices use MST BPDUs to communicate with MST devices.

MST Terminology

MST naming conventions include identification of some internal or regional parameters. These parameters are used only within an MST region, compared to external parameters that are used throughout the whole network. Because the CIST is the only spanning tree instance that spans the whole network, only the CIST parameters require the external qualifiers and not the internal or regional qualifiers. The MST terminology is as follows:

- The CIST root is the root bridge for the CIST, which is the unique instance that spans the whole network.
- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. An MST region looks like a single device to the CIST. The CIST external root path cost is the root path cost calculated between these virtual devices and devices that do not belong to any region.
- If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest device to the CIST root in the region. The CIST regional root acts as a root bridge for the IST.
- The CIST internal root path cost is the cost to the CIST regional root in a region. This cost is only relevant to the IST, instance 0.

Hop Count

MST does not use the message-age and maximum-age information in the configuration BPDU to compute the STP topology inside the MST region. Instead, the protocol uses the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region.

The hop count achieves the same result as the message-age information (triggers a reconfiguration). The root bridge of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the maximum value. When a device receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs that it generates. When the count reaches zero, the device discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the 802.1w portion of the BPDU remain the same throughout the region (only on the IST), and the same values are propagated by the region-designated ports at the boundary.

You configure a maximum aging time as the number of seconds that a device waits without receiving spanning tree configuration messages before attempting a reconfiguration.

Boundary Ports

A boundary port is a port that connects to a LAN, the designated bridge of which is either a bridge with a different MST configuration (and so, a separate MST region) or a Rapid PVST+ or 802.1D STP bridge. A designated port knows that it is on the boundary if it detects an STP bridge or receives an agreement proposal from an MST bridge with a different configuration or a Rapid PVST+ bridge. This definition allows two ports that are internal to a region to share a segment with a port that belongs to a different region, creating the possibility of receiving both internal and external messages on a port.

Figure 15: MST Boundary Ports

At the boundary, the roles of MST ports do not matter; the system forces their state to be the same as the IST port state. If the boundary flag is set for the port, the MST port-role selection process assigns a port role to the boundary and assigns the same state as the state of the IST port. The IST port at the boundary can take up any port role except a backup port role.

Detecting Unidirectional Link Failure

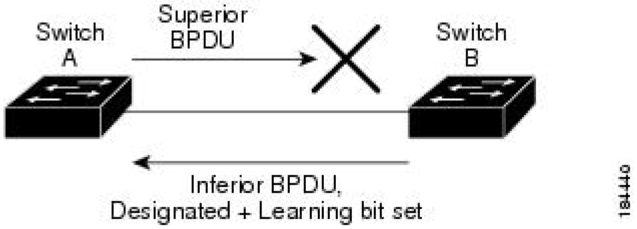
Currently, this feature is not present in the IEEE MST standard, but it is included in the standard-compliant implementation; it is based on the dispute mechanism. The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops. This feature is based on the dispute mechanism.

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on Unidirectional Link Detection (UDLD).

When a designated port detects a conflict, it keeps its role, but reverts to a discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 16: Detecting a Unidirectional Link Failure

This figure shows a unidirectional link failure that typically creates a bridging loop. Switch A is the root bridge, and its BPDUs are lost on the link leading to switch B. Rapid PVST+ (802.1w) and MST BPDUs include the role and state of the sending port. With this information, switch A can detect that switch B does not react to the superior BPDUs that it sends and that switch B is the designated, not root port. As a result, switch A blocks (or keeps blocking) its port, which prevents the bridging loop.



Port Cost and Port Priority

Spanning tree uses port costs to break a tie for the designated port. Lower values indicate lower port costs, and spanning tree chooses the least costly path. Default port costs are taken from the bandwidth of the interface, as follows:

- 10 Mbps—2,000,000
- 100 Mbps—200,000
- 1 Gigabit Ethernet—20,000
- 10 Gigabit Ethernet—2,000

You can configure the port costs in order to influence which port is chosen.

Note MST always uses the long path-cost calculation method, so the range of valid values is between 1 and 200,000,000.

The system uses port priorities to break ties among ports with the same cost. A lower number indicates a higher priority. The default port priority is 128. You can configure the priority to values between 0 and 224, in increments of 32.

Interoperability with IEEE 802.1D

A device that runs MST supports a built-in protocol migration feature that enables it to interoperate with 802.1D STP devices. If this device receives an 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only 802.1D BPDUs on that port. In addition, an MST device can detect that a port is at the boundary of a region when it receives an 802.1D BPDU, an MST BPDU (Version 3) associated with a different region, or an 802.1w BPDU (Version 2).

However, the device does not automatically revert to the MST mode if it no longer receives 802.1D BPDUs because it cannot detect whether the 802.1D device has been removed from the link unless the 802.1D device is the designated device. A device might also continue to assign a boundary role to a port when the device to which this device is connected has joined the region.

To restart the protocol migration process (force the renegotiation with neighboring devices), enter the **clear spanning-tree detected-protocols** command.

All Rapid PVST+ switches (and all 8021.D STP switches) on the link can process MST BPDUs as if they are 802.1w BPDUs. MST devices can send either Version 0 configuration and topology change notification (TCN) BPDUs or Version 3 MST BPDUs on a boundary port. A boundary port connects to a LAN, the designated device of which is either a single spanning tree device or a device with a different MST configuration.

MST interoperates with the Cisco prestandard MSTP whenever it receives prestandard MSTP on an MST port; no explicit configuration is necessary. In Cisco NX-OS Release 4.0(2) and later releases, you can configure specified interfaces to send prestandard MSTP messages all the time; it does not have to wait to receive a prestandard MST message to begin sending prestandard MST messages.

You can also configure the interface to proactively send prestandard MSTP messages.

High Availability for MST

The software supports high availability for MST. However, the statistics and timers are not restored when MST restarts. The timers start again and the statistics begin from 0.

The device supports full nondisruptive upgrades for MST. See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on nondisruptive upgrades and high-availability features.

Virtualization Support for MST

The system provides support for virtual device contexts (VDCs), and each VDC runs a separate STP.

Figure 17: Separate STP in Each VDC

You can run Rapid PVST+ in one VDC and run MST in another VDC as shown in this figure. Each VDC will have its own MST. Ensure that you are in the correct VDC.

For example, VDC1 can run MST, VDC2 can run Rapid PVST+, and VDC 3 can run MST.

Note MSTs in different VDCs are distinct, so you must perform MST region configuration for each VDC.

Note See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on VDCs and assigning resources.

Prerequisites for MST

MST has the following prerequisites:

- You must be logged onto the device.
- If you are working in another VDC than the default VDC, that VDC must be created.

Guidelines and Limitations for Configuring MST

Note When you change the VLAN-to-MSTI mapping, the system reconverges MST.

MST has the following configuration guidelines and limitations:

- You must enable MST; Rapid PVST+ is the default spanning tree mode.
- You can assign a VLAN to only one MST instance at a time.
- You cannot map VLANs 3968 to 4047 or 4094 to an MST instance. These VLANs are reserved for internal use by the device.
- You can have up to 65 MST instances on one device.
- The maximum number of VLANs and ports is 75,000.
- By default, all VLANs are mapped to MSTI 0 or the IST.
- When you are working with private VLANs on the system, ensure that all secondary VLANs are mapped to the same MSTI as the primary VLAN.
- You can load balance only within the MST region.
- Ensure that trunks carry all of the VLANs that are mapped to an MSTI or exclude all those VLANs that are mapped to an MSTI.
- Always leave STP enabled.
- Do not change timers because you can adversely affect your network stability.
- Keep user traffic off the management VLAN; keep the management VLAN separate from user data.

- Choose the distribution and core layers as the location of the primary and secondary root switches.
- Port channeling—The port channel bundle is considered as a single port. The port cost is the aggregation of all the configured port costs assigned to that channel.
- When you map a VLAN to an MSTI, the system automatically removes that VLAN from its previous MSTI.
- You can map any number of VLANs to an MSTI.
- All MST boundary ports must be forwarding for load balancing between Rapid PVST+ and an MST cloud or between a PVST+ and an MST cloud. The CIST regional root of the MST cloud must be the root of the CST. If the MST cloud consists of multiple MST regions, one of the MST regions must contain the CST root and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the Rapid PVST+ or PVST+ cloud.
- Do not partition the network into a large number of regions. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by non-Layer 2 devices.
- When you work with private VLANs, enter the **private-vlan synchronize** command to map the secondary VLANs to the same MST instance as the primary VLAN.
- When you are in the MST configuration submode, the following guidelines apply:
    - Each command reference line creates its pending regional configuration.
    - The pending region configuration starts with the current region configuration.
    - To leave the MST configuration submode without committing any changes, enter the **abort** command.
    - To leave the MST configuration submode and commit all the changes that you made before you left the submode, enter the **exit** or **end** commands, or press **Ctrl-Z**.

Note The software supports full nondisruptive upgrades for MST. See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information about nondisruptive upgrades.

Default Settings for MST

This table lists the default settings for MST parameters.

Table 17: Default MST Parameters

Parameters	Default
Spanning tree	Enabled

Parameters	Default
Spanning tree mode	Rapid PVST+ is enabled by default **Caution** Changing the spanning tree mode disrupts the traffic because all spanning tree instances are stopped for the previous mode and started for the new mode.
Name	Empty string
VLAN mapping	All VLANs mapped to a CIST instance
Revision	0
Instance ID	Instance 0; VLANs 1 to 4094 are mapped to Instance 0 by default
MSTIs per MST region	65
Bridge priority (configurable per CIST port)	32768
Spanning tree port priority (configurable per CIST port)	128
Spanning tree port cost (configurable per CIST port)	Auto The default port cost is determined by the port speed as follows: • 10 Mbps: 2,000,000 • 100 Mbps: 200,000 • 1 Gigabit Ethernet: 20,000 • 10 Gigabit Ethernet: 2,000
Hello time	2 seconds
Forward-delay time	15 seconds
Maximum-aging time	20 seconds
Maximum hop count	20 hops
Link type	Auto The default link type is determined by the duplex, as follows: • Full duplex: point-to-point link • Half duplex: shared link

Configuring MST

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco software commands for this feature might differ from the Cisco IOS commands that you would use.

Enabling MST—CLI Version

You can enable MST; Rapid PVST+ is the default.

You cannot simultaneously run MST and Rapid PVST+ on the same VDC.

Note When you change the spanning tree mode, traffic is disrupted because all spanning tree instances are stopped for the previous mode and started for the new mode.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td><b>config t</b>
<b>Example:</b>
<code>switch# config t</code>
<code>switch(config)#</code></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>spanning-tree mode mst</b></td><td>Enables MST on the device.</td></tr>
<tr><td><b>no spanning-tree mode mst</b></td><td>Disables MST on the device and returns you to Rapid PVST+.</td></tr>
</table>
<b>Example:</b>
<code>switch(config)# spanning-tree mode mst</code></td><td></td></tr>
<tr><td><b>Step 3</b></td><td><b>exit</b>
<b>Example:</b>
<code>switch(config)# exit</code>
<code>switch#</code></td><td>Exits global configuration mode.</td></tr>
</table>

	Command or Action	Purpose
Step 4	(Optional) **show running-config spanning-tree all** **Example:** `switch# show running-config spanning-tree all`	Displays the currently running STP configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to enable MST on the device:

```
switch# config t
switch(config)# spanning-tree mst configuration
switch(config-mst)# exit
switch(config)#
```

Entering MST Configuration Mode

You enter MST configuration mode to configure the MST name, VLAN-to-instance mapping, and MST revision number on the device.

If two or more devices are in the same MST region, they must have the identical MST name, VLAN-to-instance mapping, and MST revision number.

Note Each command reference line creates its pending regional configuration in MST configuration mode. In addition, the pending region configuration starts with the current region configuration.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	Enter one of the following commands:	

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>spanning-tree mst configuration</b></td><td>Enters MST configuration submode on the system. You must be in the MST configuration submode to assign the MST configuration parameters, as follows:
• MST name
• VLAN-to-MST instance mapping
• MST revision number
• Synchronize primary and secondary VLANs in private VLANs</td></tr>
<tr><td><b>no spanning-tree mst configuration</b></td><td>Returns the MST region configuration to the following default values:
• The region name is an empty string.
• No VLANs are mapped to any MST instance (all VLANs are mapped to the CIST instance).
• The revision number is 0.</td></tr>
</table>
<b>Example:</b>
switch(config)# spanning-tree mst configuration
switch(config-mst)#
</td><td></td></tr>
<tr><td><b>Step 3</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>exit</b></td><td>Commits all the changes and exits MST configuration submode.</td></tr>
<tr><td><b>abort</b></td><td>Exits the MST configuration submode without committing any of the changes.</td></tr>
</table>
<b>Example:</b>
switch(config-mst)# exit
switch(config)#
</td><td></td></tr>
</table>

	Command or Action	Purpose
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to enter MST configuration submode on the device:

```
switch# spanning-tree mst configuration
switch(config-mst)#
```

Specifying the MST Name

You can configure a region name on the bridge. If two or more bridges are in the same MST region, they must have the identical MST name, VLAN-to-instance mapping, and MST revision number.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst configuration**
	Example:
	`switch(config)# spanning-tree mst configuration`
	`switch(config-mst)#`
		Enters MST configuration submode.

Step 3	**name** *name* **Example:** `switch(config-mst)# name accounting`	Specifies the name for the MST region. The *name* string has a maximum length of 32 characters and is case sensitive. The default is an empty string.
Step 4	Enter one of the following commands: **Option** — **Description** **exit** — Commits all the changes and exits MST configuration submode.	

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td><table><tr><th>Option</th><th>Description</th></tr><tr><td><b>abort</b></td><td>Exits the MST configuration submode without committing any of the changes.</td></tr></table><b>Example:</b>
switch(config-mst)# exit
switch(config)#</td><td></td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) <b>show spanning-tree mst</b>
<b>Example:</b>
switch# show spanning-tree mst</td><td>Displays the MST configuration.</td></tr>
<tr><td><b>Step 6</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
switch(config)# copy running-config startup-config</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to set the name of the MST region:

```
switch# config t
switch(config)# spanning-tree mst configuration
switch(config-mst)# name accounting
switch(config-mst)#
```

Specifying the MST Configuration Revision Number

You configure the revision number on the bridge. If two or more bridges are in the same MST region, they must have the identical MST name, VLAN-to-instance mapping, and MST revision number.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	**config t** **Example:** switch# config t switch(config)#	Enters global configuration mode.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 2</b></td><td><b>spanning-tree mst configuration</b>
<b>Example:</b>
switch(config)# spanning-tree mst configuration
switch(config-mst)#</td><td>Enters MST configuration submode.</td></tr>
<tr><td><b>Step 3</b></td><td><b>revision</b> <i>version</i>
<b>Example:</b>
switch(config-mst)# revision 5</td><td>Specifies the revision number for the MST region. The range is from 0 to 65535, and the default value is 0.</td></tr>
<tr><td><b>Step 4</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>exit</b></td><td>Commits all the changes and exits MST configuration submode.</td></tr>
<tr><td><b>abort</b></td><td>Exits MST configuration submode without committing any of the changes.</td></tr>
</table>
<b>Example:</b>
switch(config-mst)# exit
switch(config)#</td><td></td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) <b>show spanning-tree mst</b>
<b>Example:</b>
switch# show spanning-tree mst</td><td>Displays the MST configuration.</td></tr>
<tr><td><b>Step 6</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
switch(config)# copy running-config startup-config</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to configure the revision number of the MSTI region to 5:

```
switch# config t
switch(config)# spanning-tree mst configuration
switch(config-mst)# revision 5
switch(config-mst)#
```

Specifying the Configuration on an MST Region

If two or more devices are to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same MST name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing IEEE 802.1w RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can support only up to 65 MST instances. You can assign a VLAN to only one MST instance at a time.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst configuration**
	Example:
	`switch(config)# spanning-tree mst`
	`configuration`
	`switch(config-mst)#`
		Enters MST configuration submode.

Step 3	**instance** *instance-id* **vlan** *vlan-range* **Example:** `switch(config-mst)# instance 1 vlan 10-20`	Maps VLANs to an MST instance as follows: • For *instance-id*, the range is from 1 to 4094. • For **vlan** *vlan-range*, the range is from 1 to 4094. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. To specify a VLAN range, enter a hyphen; for example, enter the **instance 1 vlan 1-63** command to map VLANs 1 through 63 to MST instance 1. To specify a VLAN series, enter a comma; for example, enter the **instance 1 vlan 10, 20, 30** command to map VLANs 10, 20, and 30 to MST instance 1.
Step 4	**name** *name* **Example:** `switch(config-mst)# name region1`	Specifies the instance name. The name string has a maximum length of 32 characters and is case sensitive.
Step 5	**revision** *version* **Example:**	Specifies the configuration revision number. The range is from 0 to 65535.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td><code>switch(config-mst)# revision 1</code></td><td></td></tr>
<tr><td><b>Step 6</b></td><td>Enter one of the following commands:<table><tr><th>Option</th><th>Description</th></tr><tr><td><b>exit</b></td><td>Commits all the changes and exits MST configuration submode.</td></tr><tr><td><b>abort</b></td><td>Exits MST configuration submode without committing any of the changes.</td></tr></table><b>Example:</b>
<code>switch(config-mst)# exit</code>
<code>switch(config)#</code></td><td></td></tr>
<tr><td><b>Step 7</b></td><td>(Optional) <b>show spanning-tree mst</b>
<b>Example:</b>
<code>switch# show spanning-tree mst</code></td><td>Displays the MST configuration.</td></tr>
<tr><td><b>Step 8</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
<code>switch(config)# copy running-config startup-config</code></td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to enter MST configuration mode, map VLANs 10 to 20 to MST instance 1, name the region *region1*, set the configuration revision to 1, display the pending configuration, apply the changes, and return to global configuration mode:

```
switch# config t
switch# spanning-tree mst configuration
switch(config-mst)# instance 1 vlan 10-20
switch(config-mst)# name region1
```

Mapping or Unmapping a VLAN to an MST Instance—CLI Version

If two or more bridges are to be in the same MST region, they must have the identical MST name, VLAN-to-instance mapping, and MST revision number.

You cannot map VLANs 3968 to 4047 or 4094 to any MST instance. These VLANs are reserved for internal use by the device.

Note When you change the VLAN-to-MSTI mapping, the system reconverges MST.

Note You cannot disable an MSTI.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td><b>config t</b>
<b>Example:</b><pre>
switch# config t
switch(config)#
</pre></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td><b>spanning-tree mst configuration</b>
<b>Example:</b><pre>
switch(config)# spanning-tree mst
configuration
switch(config-mst)#
</pre></td><td>Enters MST configuration submode.</td></tr>
<tr><td><b>Step 3</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>instance</b> <i>instance-id</i> <b>vlan</b> <i>vlan-range</i></td><td>Maps VLANs to an MST instance as follows:
• For <i>instance_id</i>, the range is from 1 to 4094. Instance 0 is reserved for the IST for each MST region.
• For <i>vlan-range</i>, the range is from 1 to 4094.
When you map VLANs to an MSTI, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped.</td></tr>
<tr><td><b>no instance</b> <i>instance-id</i> <b>vlan</b> <i>vlan-range</i></td><td>Deletes the specified instance and returns the VLANs to the default MSTI, which is the CIST.</td></tr>
</table>
<b>Example:</b><pre>
switch(config-mst)# instance 3 vlan 200
</pre></td><td></td></tr>
</table>

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 4</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>exit</b></td><td>Commits all the changes and exits MST configuration submode.</td></tr>
<tr><td><b>abort</b></td><td>Exits MST configuration submode without committing any of the changes.</td></tr>
</table>
<b>Example:</b>
switch(config-mst)# exit
switch(config)#</td><td></td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) <b>show spanning-tree mst</b>
<b>Example:</b>
switch# show spanning-tree mst</td><td>Displays the MST configuration.</td></tr>
<tr><td><b>Step 6</b></td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
switch(config)# copy running-config startup-config</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to map VLAN 200 to MSTI 3:

```
switch# config t
switch(config)# spanning-tree mst configuration
switch(config-mst)# instance 3 vlan 200
switch(config-mst)# exit
switch(config)#
```

Mapping Secondary VLANs to Same MSTI as Primary VLANs for Private VLANs

When you are working with private VLANs on the system, all secondary VLANs must be in the same MSTI as their associated primary VLAN. Enter the **private-vlan synchronize** command to accomplish this synchronization automatically.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td>Step 1</td><td><b>config t</b>
<b>Example:</b>
switch# config t
switch(config)#</td><td>Enters global configuration mode.</td></tr>
<tr><td>Step 2</td><td><b>spanning-tree mst configuration</b>
<b>Example:</b>
switch(config)# spanning-tree mst configuration
switch(config-mst)#</td><td>Enters MST configuration submode.</td></tr>
<tr><td>Step 3</td><td><b>private-vlan synchronize</b>
<b>Example:</b>
switch(config-mst)# private-vlan synchronize</td><td>Automatically maps all secondary VLANs to the same MSTI as their associated primary VLAN for all private VLANs.</td></tr>
<tr><td>Step 4</td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>exit</b></td><td>Commits all the changes and exits MST configuration submode.</td></tr>
<tr><td><b>abort</b></td><td>Exits MST configuration submode without committing any of the changes.</td></tr>
</table>
<b>Example:</b>
switch(config-mst)# exit
switch(config)#</td><td></td></tr>
<tr><td>Step 5</td><td>(Optional) <b>show spanning-tree mst</b>
<b>Example:</b>
switch# show spanning-tree mst</td><td>Displays the MST configuration.</td></tr>
<tr><td>Step 6</td><td>(Optional) <b>copy running-config startup-config</b>
<b>Example:</b>
switch(config)# copy running-config startup-config</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to automatically map all the secondary VLANs to the same MSTI as their associated primary VLANs in all private VLANs:

```
switch# config t
switch(config)# spanning-tree mst configuration
switch(config-mst)# private-vlan synchronize
switch(config-mst)# exit
switch(config)#
```

Configuring the Root Bridge

You can configure the device to become the MST root bridge.

The **spanning-tree vlan** *vlan_ID* **primary root** command fails if the value required to be the root bridge is less than 4096. If the software cannot lower the bridge priority any lower, the device returns the following message:

```
Error: Failed to set root bridge for VLAN 1
It may be possible to make the bridge root by setting the priority
for some (or all) of these instances to zero.
```

Note The root bridge for each MSTI should be a backbone or distribution device. Do not configure an access device as the spanning tree primary root bridge.

Enter the **diameter** keyword, which is available only for MSTI 0 (or the IST), to specify the Layer 2 network diameter (that is, the maximum number of Layer 2 hops between any two end stations in the Layer 2 network). When you specify the network diameter, the device automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can enter the **hello** keyword to override the automatically calculated hello time.

Note With the device configured as the root bridge, do not manually configure the hello time, forward-delay time, and maximum-age time using the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and **spanning-tree mst max-age** global configuration commands.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td><b>config t</b>
<b>Example:</b>
<code>switch# config t</code>
<code>switch(config)#</code></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>spanning-tree mst</b> <i>instance-id</i></td><td>Configures a device as the root bridge as follows:</td></tr>
</table></td><td></td></tr>
</table>

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>root</b> {<b>primary</b> | <b>secondary</b>} [<b>diameter</b> <i>dia</i> [<b>hello-time</b> <i>hello-time</i>]]</td><td>• For <i>instance-id</i>, specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is from 1 to 4094.
• For <b>diameter</b> <i>net-diameter</i>, specify the maximum number of Layer 2 hops between any two end stations. The default is 7. This keyword is available only for MST instance 0.
• For <b>hello-time</b> <i>seconds</i>, specify the interval in seconds between the generation of configuration messages by the root bridge. The range is from 1 to 10 seconds; the default is 2 seconds.</td></tr>
<tr><td><b>no spanning-tree mst</b> <i>instance-id</i> <b>root</b></td><td>Returns the switch priority, diameter, and hello time to default values.</td></tr>
</table>
<b>Example:</b>
<pre>switch(config)# spanning-tree mst 5 root
 primary</pre>
</td><td></td></tr>
<tr><td><b>Step 3</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>exit</b></td><td>Commits all the changes and exits MST configuration submode.</td></tr>
<tr><td><b>abort</b></td><td>Exits MST configuration submode without committing any of the changes.</td></tr>
</table>
<b>Example:</b>
</td><td></td></tr>
</table>

	Command or Action	Purpose

	`switch(config)# exit`
	`switch#`	

Step 4	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the device as the root switch for MSTI 5:

```
switch# config t
switch(config)# spanning-tree mst 5 root primary
switch(config)# exit
switch#
```

Configuring an MST Secondary Root Bridge

You use this command on more than one device to configure multiple backup root bridges. Enter the same network diameter and hello-time values that you used when you configured the primary root bridge with the **spanning-tree mst root primary** global configuration command.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td><b>config t</b>
<b>Example:</b>
<code>switch# config t</code>
<code>switch(config)#</code></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td>Enter one of the following commands:<table><tr><th>Option</th><th>Description</th></tr><tr><td><b>spanning-tree mst</b> <i>instance-id</i> <b>root</b></td><td>Configures a device as the secondary root bridge as follows:</td></tr></table></td><td></td></tr>
</table>

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>{<b>primary</b> | <b>secondary</b>} [<b>diameter</b> <i>dia</i>[<b>hello-time</b> <i>hello-time</i>]]</td><td>
<ul>
<li>For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is from 1 to 4094.</li>
<li>For <b>diameter</b> <i>net-diameter</i>, specify the maximum number of Layer 2 hops between any two end stations. The default is 7. This keyword is available only for MST instance 0.</li>
<li>For <b>hello-time</b> <i>seconds</i>, specify the interval in seconds between the generation of configuration messages by the root bridge. The range is from 1 to 10 seconds; the default is 2 seconds.</li>
</ul>
</td></tr>
<tr><td><b>no spanning-tree mst</b> <i>instance-id</i> <b>root</b></td><td>Returns the switch priority, diameter, and hello-time to default values.</td></tr>
</table>
<b>Example:</b>
<pre>switch(config)# spanning-tree mst 5 root
 secondary</pre>
</td><td></td></tr>
<tr><td><b>Step 3</b></td><td><b>exit</b>
<b>Example:</b>
<pre>switch# exit
switch(config)#</pre>
</td><td>Exits global configuration mode.</td></tr>
<tr><td><b>Step 4</b></td><td>(Optional) <b>show spanning-tree mst</b>
<b>Example:</b>
<pre>switch# show spanning-tree mst</pre>
</td><td>Displays the MST configuration.</td></tr>
</table>

	Command or Action	Purpose
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the device as the secondary root switch for MSTI 5:

```
switch# config t
switch(config)# spanning-tree mst 5 root secondary
switch(config)# exit
switch#
```

Configuring the MST Switch Priority

You can configure the switch priority for an MST instance so that it is more likely that the specified device is chosen as the root bridge.

Note Be careful when using the **spanning-tree mst priority** command. For most situations, we recommend that you enter the **spanning-tree mst root primary** and the **spanning-tree mst root secondary** global configuration commands to modify the switch priority.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst** *instance-id* **priority** *priority-value* **Example:** `switch(config)# spanning-tree mst 5 priority 4096`	Configures a device priority as follows: • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is from 1 to 4094. • For *priority-value* the range is from 0 to 61440 in increments of 4096; the default is 32768. A lower number indicates that

	Command or Action	Purpose
		the device will most likely be chosen as the root bridge. Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. The system rejects all other values.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the priority of the bridge to 4096 for MSTI 5:

```
switch# config t
switch(config)# spanning-tree mst 5 priority 4096
switch(config)# exit
switch#
```

Configuring the MST Port Priority

If a loop occurs, MST uses the port priority when selecting an interface to put into the forwarding state. You can assign lower priority values to interfaces that you want selected first and higher priority values to the interface that you want selected last. If all interfaces have the same priority value, MST puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** {{*type slot/port*} | {**port-channel** *number*}}
	Example:
	`switch(config)# interface ethernet 3/1`
	`switch(config-if)#`
		Specifies an interface to configure, and enters interface configuration mode.

Step 3	**spanning-tree mst** *instance-id* **port-priority** *priority* **Example:** `switch(config-if)# spanning-tree mst 3 port-priority 64`	Configures the port priority as follows: • For *instance-id*, you can specify a single MSTI, a range of MSTIs separated by a hyphen, or a series of MSTIs separated by a comma. The range is from 1 to 4094. • For *priority*, the range is from 0 to 224 in increments of 32. The default is 128. A lower number indicates a higher priority. The priority values are 0, 32, 64, 96, 128, 160, 192, and 224. The system rejects all other values.

Step 4	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 5	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to set the MST interface port priority for MSTI 3 on Ethernet port 3/1 to 64:

```
switch# config t
switch(config)# interface ethernet 3/1
switch(config-if)# spanning-tree mst 3 port-priority 64
switch(config-if)# exit
switch(config)#
```

Configuring the MST Port Cost

The MST port cost default value is derived from the media speed of an interface. If a loop occurs, MST uses the cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost to interfaces values that you want selected last. If all interfaces have the same cost value, MST puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Note MST uses the long path-cost calculation method.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** {{*type slot/port*} | {**port-channel** *number*}}
	Example:
	`switch# config t`
	`switch(config)# interface ethernet 3/1`
	`switch(config-if)#`
		Specifies an interface to configure, and enters interface configuration mode.

Step 3	**spanning-tree mst** *instance-id* **cost** {*cost* | *auto*}
	Example:
	`switch(config-if)# spanning-tree mst 4 cost 17031970`
		Configures the cost.
		If a loop occurs, MST uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission as follows:
		• For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is from 1 to 4094.

	Command or Action	Purpose
		• For *cost*, the range is from 1 to 200000000. The default value is **auto**, which is derived from the media speed of the interface.

Step 4	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 5	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to set the MST interface port cost on Ethernet 3/1 for MSTI 4:

```
switch# config t
switch(config)# interface ethernet 3/1
switch(config-if)# spanning-tree mst 4 cost 17031970
switch(config-if)# exit
switch(config)#
```

Configuring the MST Hello Time

You can configure the interval between the generation of configuration messages by the root bridge for all instances on the device by changing the hello time.

Note Be careful when using the **spanning-tree mst hello-time** command. For most situations, we recommend that you enter the **spanning-tree mst** *instance-id* **root primary** and the**spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the hello time.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst hello-time** *seconds* **Example:** `switch(config)# spanning-tree mst hello-time 1`	Configures the hello time for all MST instances. The hello time is the interval between the generation of configuration messages by the root bridge. These messages mean that the device is alive. For *seconds*, the range is from 1 to 10, and the default is 2 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the hello time of the device to 1 second:

```
switch# config t
switch(config)# spanning-tree mst hello-time 1
switch(config)# exit
switch#
```

Configuring the MST Forwarding-Delay Time

You can set the forward delay timer for all MST instances on the device with one command.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst forward-time** *seconds* **Example:** `switch(config)# spanning-tree mst forward-time 10`	Configures the forward time for all MST instances. The forward delay is the number of seconds that a port waits before changing from its spanning tree blocking and learning states to the forwarding state. For *seconds*, the range is from 4 to 30, and the default is 15 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the forward-delay time of the device to 10 seconds:

```
switch# config t
switch(config)# spanning-time mst forward-time 10
switch(config)# exit
switch#
```

Configuring the MST Maximum-Aging Time

You can set the maximum-aging timer for all MST instances on the device with one command (the maximum age time only applies to the IST).

The maximum-aging timer is the number of seconds that a device waits without receiving spanning tree configuration messages before attempting a reconfiguration.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst max-age** *seconds* **Example:** `switch(config)# spanning-tree mst max-age 40`	Configures the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds that a device waits without receiving spanning tree configuration messages before attempting a reconfiguration. For *seconds*, the range is from 6 to 40, and the default is 20 seconds.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the maximum-aging timer of the device to 40 seconds:

```
switch# config t
switch(config)# spanning-tree mst max-age 40
switch(config)# exit
switch#
```

Configuring the MST Maximum-Hop Count

You can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. MST uses the path cost to the IST regional root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism. The hop count achieves the same result as the message-age information (triggers a reconfiguration).

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**spanning-tree mst max-hops** *hop-count* **Example:** `switch(config)# spanning-tree mst max-hops 40`	Specifies the number of hops in a region before the BPDU is discarded and the information held for a port is aged. For *hop-count*, the range is from 1 to 255, and the default value is 20 hops.

Step 3	**exit**
	Example:
	`switch(config-mst)# exit`
	`switch#`
		Exits MST configuration submode.

Step 4	(Optional) **show spanning-tree mst** **Example:** `switch# show spanning-tree mst`	Displays the MST configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to set the maximum hops to 40:

```
switch# config t
switch(config)# spanning-tree mst max-hops 40
switch(config)# exit
switch#
```

Configuring an Interface to Proactively Send Prestandard MSTP Messages—CLI Version

By default, interfaces on a device running MST send prestandard, rather than standard, MSTP messages after they receive a prestandard MSTP message from another interface. In Cisco NX-OS Release 4.0(2) and later releases, you can configure the interface to proactively send prestandard MSTP messages. That is, the specified interface would not have to wait to receive a prestandard MSTP message; the interface with this configuration always sends prestandard MSTP messages.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to configure and enters interface configuration mode.
Step 3	switch(config-if)# **spanning-tree mst pre-standard**	Specifies that the interface always sends MSTP messages in the prestandard format, rather than in the MSTP standard format.
Step 4	switch(config-if)# **exit**	Exits interface configuration mode.
Step 5	(Optional) switch# **show spanning-tree mst**	Displays the MST configuration.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to set the MST interface so that it always sends MSTP messages in the prestandard format:

```
switch# config t
switch (config)# interface ethernet 1/4
switch(config-if)# spanning-tree mst pre-standard
switch(config-if)# exit
switch(config)#
```

Specifying the Link Type for MST—CLI Version

Rapid connectivity (802.1w standard) is established only on point-to-point links. By default, the link type is controlled from the duplex mode of the interface. A full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection.

If you have a half-duplex link physically connected point to point to a single port on a remote device, you can override the default setting on the link type and enable rapid transitions.

If you set the link to shared, STP falls back to 802.1D.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** *type slot/port*
	Example:
	`switch(config)# interface ethernet 1/4`
	`switch(config-if)#`
		Specifies the interface to configure and enters interface configuration mode.

Step 3	**spanning-tree link-type** {*auto* | *point-to-point* | *shared*}
	Example:
	`switch(config-if)# spanning-tree link-type point-to-point`
		Configures the link type to be either a point-to-point link or shared link. The system reads the default value from the device connection, as follows: half duplex links are shared and full-duplex links are point to point. If the link type is shared, the STP falls back to 802.1D. The default is auto, which sets the link type based on the duplex setting of the interface.

Step 4	**exit**
	Example:
	`switch(config-if)# exit`
	`switch(config)#`
		Exits interface configuration mode.

Step 5	(Optional) **show spanning-tree** **Example:** `switch# show spanning-tree`	Displays the STP configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the link type as a point-to-point link:

```
switch# config t
switch (config)# interface ethernet 1/4
switch(config-if)# spanning-tree link-type point-to-point
switch(config-if)# exit
switch(config)#
```

Reinitializing the Protocol for MST

An MST bridge can detect that a port is at the boundary of a region when it receives a legacy BPDU or an MST BPDU that is associated with a different region. However, the STP protocol migration cannot determine whether the legacy device, which is a device that runs only IEEE 802.1D, has been removed from the link unless the legacy device is the designated switch. Enter this command to reinitialize the protocol negotiation (force the renegotiation with neighboring devices) on the entire device or on specified interfaces.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	**clear spanning-tree detected-protocol** [**interface** *interface* [*interface-num* | *port-channel*]] **Example:** `switch# clear spanning-tree detected-protocol`	Reintializes MST on an entire device or specified interfaces.

Example

This example shows how to reinitialize MST on the Ethernet interface on slot 2, port 8:

```
switch# clear spanning-tree detected-protocol interface ethernet 2/8
```

Verifying the MST Configuration

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

Command	Purpose
show running-config spanning-tree [**all**]	Displays STP information.
show spanning-tree mst configuration	Displays MST information.
show spanning-tree mst [**detail**]	Displays information about MST instances.
show spanning-tree mst *instance-id* [**detail**]	Displays information about the specified MST instance.
show spanning-tree mst *instance-id* **interface** {**ethernet** *slot/port* | **port-channel** *channel-number*} [**detail**]	Displays MST information for the specified interface and instance.
show spanning-tree summary	Displays summary STP information.

Command	Purpose
show spanning-tree detail	Displays detailed STP information.
show spanning-tree {**vlan** *vlan-id* | **interface** {[**ethernet** *slot/port*] | [**port-channel** *channel-number*]}} [**detail**]	Displays STP information per VLAN and interface.
show spanning-tree vlan *vlan-id* **bridge**	Displays information on the STP bridge.

For information on the output of these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Displaying and Clearing MST Statistics—CLI Version

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

Command	Purpose
clear spanning-tree counters [**interface** *type slot/port* | **vlan***vlan-id*]	Clears the counters for STP.
show spanning-tree {**vlan** *vlan-id* | **interface** {[**ethernet** *slot/port*] | [**port-channel***channel-number*]}} **detail**	Displays information about STP by interface or VLAN including BPDUs sent and received.

MST Example Configuration

The following example shows how to configure MST:

```
switch# configure terminal
switch(config)# spanning-tree mode mst
switch(config)# spanning-tree port type edge bpduguard default
switch(config)# spanning-tree port type edge bpdufilter default
switch(config)# spanning-tree port type network default
switch(config)# spanning-tree mst 0-64 priority 24576
switch(config)# spanning-tree mst configuration
switch(config-mst)# name cisco_region_1
switch(config-mst)# revision 2
switch(config-mst)# instance 1 vlan 1-21
switch(config-mst)# instance 2 vlan 22-42
switch(config-mst)# instance 3 vlan 43-63
switch(config-mst)# instance 4 vlan 64-84
switch(config-mst)# instance 5 vlan 85-105
switch(config-mst)# instance 6 vlan 106-126
switch(config-mst)# instance 6 vlan 106-126
switch(config-mst)# instance 7 vlan 127-147
switch(config-mst)# instance 8 vlan 148-168
switch(config-mst)# instance 9 vlan 169-189
switch(config-mst)# instance 10 vlan 190-210
switch(config-mst)# instance 11 vlan 211-231
switch(config-mst)# instance 12 vlan 232-252
switch(config-mst)# instance 13 vlan 253-273
switch(config-mst)# instance 14 vlan 274-294
switch(config-mst)# instance 15 vlan 295-315
switch(config-mst)# instance 16 vlan 316-336
switch(config-mst)# instance 17 vlan 337-357
switch(config-mst)# instance 18 vlan 358-378
switch(config-mst)# instance 19 vlan 379-399
switch(config-mst)# instance 20 vlan 400-420
switch(config-mst)# instance 21 vlan 421-441
switch(config-mst)# instance 22 vlan 442-462
switch(config-mst)# instance 23 vlan 463-483
switch(config-mst)# instance 24 vlan 484-504
switch(config-mst)# instance 25 vlan 505-525
switch(config-mst)# instance 26 vlan 526-546
switch(config-mst)# instance 27 vlan 547-567
switch(config-mst)# instance 28 vlan 568-588
switch(config-mst)# instance 29 vlan 589-609
switch(config-mst)# instance 30 vlan 610-630
switch(config-mst)# instance 31 vlan 631-651
switch(config-mst)# instance 32 vlan 652-672
switch(config-mst)# instance 33 vlan 673-693
switch(config-mst)# instance 34 vlan 694-714
switch(config-mst)# instance 35 vlan 715-735
switch(config-mst)# instance 36 vlan 736-756
switch(config-mst)# instance 37 vlan 757-777
switch(config-mst)# instance 38 vlan 778-798
switch(config-mst)# instance 39 vlan 799-819
switch(config-mst)# instance 40 vlan 820-840
switch(config-mst)# instance 41 vlan 841-861
switch(config-mst)# instance 42 vlan 862-882
switch(config-mst)# instance 43 vlan 883-903
switch(config-mst)# instance 44 vlan 904-924
switch(config-mst)# instance 45 vlan 925-945
switch(config-mst)# instance 46 vlan 946-966
switch(config-mst)# instance 47 vlan 967-987
switch(config-mst)# instance 48 vlan 988-1008
switch(config-mst)# instance 49 vlan 1009-1029
switch(config-mst)# instance 50 vlan 1030-1050
switch(config-mst)# instance 51 vlan 1051-1071
switch(config-mst)# instance 52 vlan 1072-1092
switch(config-mst)# instance 53 vlan 1093-1113
switch(config-mst)# instance 54 vlan 1114-1134
switch(config-mst)# instance 55 vlan 1135-1155
switch(config-mst)# instance 56 vlan 1156-1176
switch(config-mst)# instance 57 vlan 1177-1197
switch(config-mst)# instance 58 vlan 1198-1218
switch(config-mst)# instance 59 vlan 1219-1239
switch(config-mst)# instance 60 vlan 1240-1260
switch(config-mst)# instance 61 vlan 1261-1281
switch(config-mst)# instance 62 vlan 1282-1302
switch(config-mst)# instance 63 vlan 1303-1323
switch(config-mst)# instance 64 vlan 1324-1344
switch(config-mst)# exit

switch(config)# interface ethernet 3/1
switch(config-if)# switchport
switch(config-if)# no shutdown
switch(config-if)# spanning-tree port type edge
switch(congig-if)# exit

switch(config)# interface ethernet 3/2
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# no shutdown
switch(config-if)# spanning-tree guard root
switch(config-if)# exit
switch(config)#
```

Additional References for MST—CLI Version

Related Documents

Related Topic	Document Title
Command reference	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*
Layer 2 interfaces	*Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
NX-OS fundamentals	*Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide*
High availability	*Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
System management	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
Licensing	*Cisco NX-OS Licensing Guide*
Release notes	*Cisco Nexus 7000 Series NX-OS Release Notes*

Standards

Standards	Title
IEEE 802.1Q-2006 (formerly known as IEEE 802.1s), IEEE 802.1D-2004 (formerly known as IEEE 802.1w), IEEE 802.1D, IEEE 802.1t	—

MIBs

MIBs	MIBs Link
• CISCO-STP-EXTENSION-MIB • BRIDGE-MIB	To locate and download MIBs, go to the following URL: https://cfnng.cisco.com/mibs.

Feature History for Configuring MST--CLI Version

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Table 18: Feature History for Configuring MSTs

Feature Name	Releases	Feature Information
No change.	4.2(1)	--
No change.	4.1(2)	--

CHAPTER 10

Configuring STP Extensions Using Cisco NX-OS

This chapter describes how to configure Spanning Tree Protocol (STP) extensions on Cisco NX-OS devices.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About STP Extensions

Note See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*, for information on creating Layer 2 interfaces.

Cisco has added extensions to STP that enhances loop prevention, protects against some possible user misconfigurations, and provides better control over the protocol parameters. Although, in some cases, similar functionality may be incorporated into the IEEE 802.1w Rapid Spanning Tree Protocol (RSTP) standard, we recommend using these extensions. All of these extensions, except PVST Simulation, can be used with both Rapid PVST+ and MST. You use PVST Simulation only with MST.

The available extensions are spanning tree edge ports (which supply the functionality previously known as PortFast), Bridge Assurance, BPDU Guard, BPDU Filtering, Loop Guard, Root Guard, and PVT Simulation. Many of these features can be applied either globally or on specified interfaces.

Note Spanning tree is used to refer to IEEE 802.1w and IEEE 802.1s. If the text is discussing the IEEE 802.1D Spanning Tree Protocol, 802.1D is stated specifically.

STP Port Types

You can configure a spanning tree port as an edge port, a network port, or a normal port. A port can be in only one of these states at a given time. The default spanning tree port type is normal.

Edge ports, which are connected to Layer 2 hosts, can be either an access port or a trunk port.

Note If you configure a port connected to a Layer 2 switch or bridge as an edge port, you might create a bridging loop.

Network ports are connected only to Layer 2 switches or bridges.

Note If you mistakenly configure ports that are connected to Layer 2 hosts, or edge devices, as spanning tree network ports, those ports will automatically move into the blocking state.

STP Edge Ports

You connect STP edge ports only to Layer 2 hosts. The edge port interface immediately transitions to the forwarding state, without moving through the blocking or learning states. (This immediate transition was previously configured as the Cisco-proprietary feature PortFast.)

Interfaces that are connected to Layer 2 hosts should not receive STP bridge protocol data units (BPDUs).

Bridge Assurance

You can use Bridge Assurance to protect against certain problems that can cause bridging loops in the network. Specifically, you use Bridge Assurance to protect against a unidirectional link failure or other software failure and a device that continues to forward data traffic when it is no longer running the spanning tree algorithm.

Note Bridge Assurance is supported only by Rapid PVST+ and MST.

Bridge Assurance is enabled by default and can only be disabled globally. Also, Bridge Assurance can be enabled only on spanning tree network ports that are point-to-point links. Finally, both ends of the link must have Bridge Assurance enabled. If the device on one side of the link has Bridge Assurance enabled and the device on the other side either does not support Bridge Assurance or does not have this feature enabled, the connecting port is blocked.

With Bridge Assurance enabled, BPDUs are sent out on all operational network ports, including alternate and backup ports, for each hello time period. If the port does not receive a BPDU for a specified period, the port moves into the blocking state and is not used in the root port calculation. Once that port receives a BPDU, it resumes the normal spanning tree transitions.

Figure 18: Network with Normal STP Topology

This figure shows a normal STP topology.

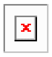


Figure 19: Network Problem without Running Bridge Assurance

This figure demonstrates a potential network problem when the device fails and you are not running Bridge Assurance.

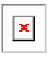


Figure 20: Network STP Topology Running Bridge Assurance

This figure shows the network with Bridge Assurance enabled, and the STP topology progressing normally with bidirectional BPDUs issuing from every STP network port.

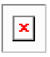
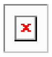


Figure 21: Network Problem Averted with Bridge Assurance Enabled

This figure shows how the potential network problem does not happen when you have Bridge Assurance enabled on your network.



BPDU Guard

BPDU Guard prevents a port from receiving BPDUs. You can configure BPDU Guard at the global or interface level.

When you configure BPDU Guard globally, it is effective only on operational spanning tree edge ports. In a valid configuration, Layer 2 LAN edge interfaces do not receive BPDUs. A BPDU that is received by an edge Layer 2 LAN interface signals an invalid configuration, such as the connection of an unauthorized device. BPDU Guard, when enabled globally, shuts down all spanning tree edge ports when they receive a BPDU.

Note When enabled globally, BPDU Guard applies to all operational spanning tree edge interfaces.

You can configure BPDU Guard at the interface level, using the following steps:

- BPDU Guard is configured in interface configuration mode using the **spanning-tree bpduguard enable** command.
- For a trunk port, specify an allowed VLAN list using the **switchport trunk allowed vlan** *vlan list* command.

BPDUs are dropped if they are not in the allowed VLAN list and BPDU Guard is enabled on the port.

In Cisco NX-OS Release 6.2(10) and later releases, the port will be error disabled when a BPDU is received on any VLAN and BPDU Guard is enabled on the port.

Note The native VLAN on the trunk port is an exception. BPDUs arriving on the native VLAN are passed on to the supervisor.

BPDU Guard provides a secure response to invalid configurations, because you must manually put the Layer 2 LAN interface back in service after an invalid configuration.

BPDU Filtering

You can use BPDU Filtering to prevent the device from sending or even receiving BPDUs on specified ports.

When configured globally, BPDU Filtering applies to all operational spanning tree edge ports. You should connect edge ports only to hosts, which typically drop BPDUs. If an operational spanning tree edge port receives a BPDU, it immediately returns to a normal spanning tree port type and moves through the regular transitions. In that case, BPDU Filtering is disabled on this port, and spanning tree resumes sending BPDUs on this port.

In addition, you can configure BPDU Filtering by the individual interface. When you explicitly configure BPDU Filtering on a port, that port does not send any BPDUs and drops all BPDUs that it receives. You can effectively override the global BPDU Filtering setting on individual ports by configuring the specific interface. This BPDU Filtering command on the interface applies to the entire interface, whether the interface is trunking or not.

Caution Use care when configuring BPDU Filtering per interface. If you explicitly configure BPDU Filtering on a port that is not connected to a host, it can result in bridging loops because the port will ignore any BPDU that it receives and go to forwarding.

This table lists all the BPDU Filtering combinations.

Table 19: BPDU Filtering Configurations

BPDU Filtering Per Port Configuration	BPDU Filtering Global Configuration	STP Edge Port Configuration	BPDU Filtering State
Default [1]	Enable	Enable	Enable [2]
Default	Enable	Disable	Disable
Default	Disable	Not applicable	Disable
Disable	Not applicable	Not applicable	Disable
Enable	Not applicable	Not applicable	Enable

[1] No explicit port configuration.

[2] The port transmits at least 10 BPDUs. If this port receives any BPDUs, the port returns to the spanning tree normal port state and BPDU filtering is disabled.

Loop Guard

Loop Guard helps prevent bridging loops that could occur because of a unidirectional link failure on a point-to-point link.

An STP loop occurs when a blocking port in a redundant topology erroneously transitions to the forwarding state. Transitions are usually caused by a port in a physically redundant topology (not necessarily the blocking port) that stops receiving BPDUs.

When you enable Loop Guard globally, it is useful only in switched networks where devices are connected by point-to-point links. On a point-to-point link, a designated bridge cannot disappear unless it sends an inferior BPDU or brings the link down. However, you can enable Loop Guard on shared links per interface,

You can use Loop Guard to determine if a root port or an alternate/backup root port receives BPDUs. If the port that was previously receiving BPDUs is no longer receiving BPDUs, Loop Guard puts the port into an inconsistent state (blocking) until the port starts to receive BPDUs again. If such a port receives BPDUs again, the port—and link—is deemed viable again. The protocol removes the loop-inconsistent condition from the port, and the STP determines the port state because the recovery is automatic.

Loop Guard isolates the failure and allows STP to converge to a stable topology without the failed link or bridge. Disabling Loop Guard moves all loop-inconsistent ports to the listening state.

You can enable Loop Guard on a per-port basis. When you enable Loop Guard on a port, it is automatically applied to all of the active instances or VLANs to which that port belongs. When you disable Loop Guard, it is disabled for the specified ports.

Enabling Loop Guard on a root device has no effect but provides protection when a root device becomes a nonroot device.

Root Guard

When you enable Root Guard on a port, Root Guard does not allow that port to become a root port. If a received BPDU triggers an STP convergence that makes that designated port become a root port, that port is put into a root-inconsistent (blocked) state. After the port stops receiving superior BPDUs, the port is unblocked again. Through STP, the port moves to the forwarding state. Recovery is automatic.

When you enable Root Guard on an interface, this functionality applies to all VLANs to which that interface belongs.

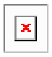
You can use Root Guard to enforce the root bridge placement in the network. Root Guard ensures that the port on which Root Guard is enabled is the designated port. Normally, root bridge ports are all designated ports, unless two or more of the ports of the root bridge are connected. If the bridge receives superior BPDUs on a Root Guard-enabled port, the bridge moves this port to a root-inconsistent STP state. In this way, Root Guard enforces the position of the root bridge.

You cannot configure Root Guard globally.

Applying STP Extension Features

Figure 22: Network with STP Extensions Correctly Deployed

We recommend that you configure the various STP extension features through your network as shown in this figure. Bridge Assurance is enabled on the entire network. You should enable either BPDU Guard or BPDU Filtering on the host interface.



PVST Simulation

MST interoperates with Rapid PVST+ with no need for user configuration. The PVST simulation feature enables this interoperability.

Note PVST simulation is enabled by default when you enable MST. By default, all interfaces on the device interoperate between MST and Rapid PVST+.

However, you may want to control the connection between MST and Rapid PVST+ to protect against accidentally connecting an MST-enabled port to a port enabled to run Rapid PVST+. Because Rapid PVST+ is the default STP mode, you may encounter many Rapid PVST+ connections.

Disabling Rapid PVST+ simulation, which can be done per port or globally for the entire device, moves the MST-enabled port to the blocking state once it detects it is connected to a Rapid PVST+-enabled port. This port remains in the inconsistent state until the port stops receiving Rapid PVST+/SSTP BPDUs, and then the port resumes the normal STP transition process.

The root bridge for all STP instances must all be in either the MST region or the Rapid PVST+ side. If the root bridge for all STP instances are not on one side or the other, the software moves the port into a PVST simulation-inconsistent state.

Note We recommend that you put the root bridge for all STP instances in the MST region.

High Availability for STP

The software supports high availability for STP. However, the statistics and timers are not restored when STP restarts. The timers start again and the statistics begin from 0.

Note See the *Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide* for complete information on high-availability features.

Virtualization Support for STP Extensions

The system provides support for virtual device contexts (VDCs), and each VDC runs a separate STP. You can run Rapid PVST+ in one VDC and MST in another VDC.

Note See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for complete information on VDCs and assigning resources.

Prerequisites for STP Extensions

STP has the following prerequisites:

- You must be logged onto the device.
- You must have STP configured already.

Guidelines and Limitations for Configuring STP Extensions

STP extensions have the following configuration guidelines and limitations:

- Connect STP network ports only to switches.
- You should configure host ports as STP edge ports and not as network ports.
- If you enable STP network port types globally, ensure that you manually configure all ports connected to hosts as STP edge ports.
- You should configure all access and trunk ports connected to Layer 2 hosts as edge ports.
- Bridge Assurance runs only on point-to-point spanning tree network ports. You must configure each side of the link for this feature.
- We recommend that you enable Bridge Assurance throughout your network.
- We recommend that you enable BPDU Guard on all edge ports.
- Enabling Loop Guard globally works only on point-to-point links.
- Enabling Loop Guard per interface works on both shared and point-to-point links.
- Root Guard forces a port to always be a designated port; it does not allow a port to become a root port. Loop Guard is effective only if the port is a root port or an alternate port. You cannot enable Loop Guard and Root Guard on a port at the same time.
- Loop Guard has no effect on a disabled spanning tree instance or a VLAN.
- Spanning tree always chooses the first operational port in the channel to send the BPDUs. If that link becomes unidirectional, Loop Guard blocks the channel, even if other links in the channel are functioning properly.

- If you group a set of ports that are already blocked by Loop Guard to form a channel, spanning tree loses all the state information for those ports and the new channel port may obtain the forwarding state with a designated role.
- If a channel is blocked by Loop Guard and the channel members go back to an individual link status, spanning tree loses all the state information. The individual physical ports may obtain the forwarding state with the designated role, even if one or more of the links that formed the channel are unidirectional.

Note You can enable UniDirectional Link Detection (UDLD) aggressive mode to isolate the link failure. A loop may occur until UDLD detects the failure, but Loop Guard will not be able to detect it. See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide* for information on UDLD.

- You should enable Loop Guard globally on a switch network with physical loops.
- You should enable Root Guard on ports that connect to network devices that are not under direct administrative control.
- If you configure the spanning-tree forward-delay time to below the default value, it can impact the system's ability to successfully perform a switchover supervisor at scale.

Default Settings for STP Extensions

This table lists the default settings for STP extensions.

Table 20: Default STP Extension Parameters

Parameters	Default
Port type	Normal
Bridge Assurance	Enabled (on STP network ports only)
Global BPDU Guard	Disabled
BPDU Guard per interface	Disabled
Global BPDU Filtering	Disabled
BPDU Filtering per interface	Disabled
Global Loop Guard	Disabled
Loop Guard per interface	Disabled
Root Guard per interface	Disabled
PVST simulation	Enabled

Configuring STP Extensions

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

You can enable Loop Guard per interface on either shared or point-to-point links.

Configuring Spanning Tree Port Types Globally

The spanning tree port type designation depends on the device the port is connected to, as follows:

- Edge—Edge ports are connected to Layer 2 hosts and are access ports.
- Network—Network ports are connected only to Layer 2 switches or bridges and can be either access or trunk ports.
- Normal—Normal ports are neither edge ports nor network ports; they are normal spanning tree ports. These ports can be connected to any device.

You can configure the port type either globally or per interface. By default, the spanning tree port type is normal.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you are configuring the ports correctly to the device to which the port is connected.

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td>switch# <b>config t</b></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td>Enter one of the following two commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>spanning-tree port type edge default</b></td><td>Configures all access ports connected to Layer 2 hosts as edge ports. Edge ports immediately transition to the forwarding state without passing through the blocking or learning state at linkup. By default, spanning tree ports are normal port types.</td></tr>
</table></td><td></td></tr>
</table>

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td><table><tr><th>Option</th><th>Description</th></tr><tr><td><b>spanning-tree port type network default</b></td><td>Configures all interfaces connected to Layer 2 switches and bridges as spanning tree network ports. If you enable Bridge Assurance, it automatically runs on network ports. By default, spanning tree ports are normal port types.
<b>Note</b> If you configure interfaces connected to Layer 2 hosts as network ports, those ports automatically move into the blocking state.</td></tr></table></td><td></td></tr>
<tr><td><b>Step 3</b></td><td>switch(config)# <b>exit</b></td><td>Exits global configuration mode.</td></tr>
<tr><td><b>Step 4</b></td><td>(Optional) switch# <b>show spanning-tree summary</b></td><td>Displays the STP configuration including STP port types if configured.</td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) switch# <b>copy running-config startup-config</b></td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to configure all access ports connected to Layer 2 hosts as spanning tree edge ports:

```
switch# config t
switch(config)# spanning-tree port type edge default
switch(config)# exit
switch#
```

This example shows how to configure all ports connected to Layer 2 switches or bridges as spanning tree network ports:

```
switch# config t
switch(config)# spanning-tree port type network default
switch(config)# exit
switch#
```

Configuring Spanning Tree Edge Ports on Specified Interfaces

You can configure spanning tree edge ports on specified interfaces. Interfaces configured as spanning tree edge ports immediately transition to the forwarding state, without passing through the blocking or learning states, on linkup.

This command has four states:

- **spanning-tree port type edge**—This command explicitly enables edge behavior on the access port.
- **spanning-tree port type edge trunk**—This command explicitly enables edge behavior on the trunk port.

Note If you enter the **spanning-tree port type edge trunk** command, the port is configured as an edge port even in the access mode.

- **spanning-tree port type normal**—This command explicitly configures the port as a normal spanning tree port and the immediate transition to the forwarding state is not enabled.
- **no spanning-tree port type**—This command implicitly enables edge behavior if you define the **spanning-tree port type edge default** command in global configuration mode. If you do not configure the edge ports globally, the **no spanning-tree port type**command is equivalent to the**spanning-tree port type normal** command.

Before you begin

- Ensure that you are in the correct VDC (or enter the**switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you are configuring the ports correctly to the device to which the port is connected.

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to configure and enters interface configuration mode.
Step 3	switch(config-if)# **spanning-tree port type edge**	Configures the specified access interfaces to be spanning edge ports. Edge ports immediately transition to the forwarding state without passing through the blocking or learning state at linkup. By default, spanning tree ports are normal port types.
Step 4	switch(config-if)# **exit**	Exits interface configuration mode.
Step 5	(Optional) switch# **show spanning-tree interface** *type slot/port*	Displays the STP configuration including the STP port type if configured.

	Command or Action	Purpose
Step 6	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the Ethernet access interface 1/4 to be a spanning tree edge port:

```
switch# config t
switch(config)# interface ethernet 1/4
switch(config-if)# spanning-tree port type edge
switch(config-if)# exit
switch(config)#
```

Configuring Spanning Tree Network Ports on Specified Interfaces

You can configure spanning tree network ports on specified interfaces.

Bridge Assurance runs only on spanning tree network ports.

This command has three states:

- **spanning-tree port type network**—This command explicitly configures the port as a network port. If you enable Bridge Assurance globally, it automatically runs on a spanning tree network port.
- **spanning-tree port type normal** —This command explicitly configures the port as a normal spanning tree port and Bridge Assurance cannot run on this interface.
- **no spanning-tree port type**—This command implicitly enables the port as a spanning tree network port if you define the **spanning-tree port type network default** command in global configuration mode. If you enable Bridge Assurance globally, it automatically runs on this port.

Note A port connected to a Layer 2 host that is configured as a network port automatically moves into the blocking state.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you are configuring the ports correctly to the device to which the port is connected.

Procedure

	Command or Action	Purpose
Step 1	switch(config)# **config t**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to configure and enters interface configuration mode.
Step 3	switch(config-if)# **spanning-tree port type network**	Configures the specified interfaces to be spanning network ports. If you enable Bridge Assurance, it automatically runs on network ports. By default, spanning tree ports are normal port types.
Step 4	switch(config-if)# **exit**	Exits interface configuration mode.
Step 5	(Optional) switch# **show spanning-tree interface** *type slot/port*	Displays the STP configuration including the STP port type if configured.
Step 6	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the Ethernet interface 1/4 to be a spanning tree network port:

```
switch# config t
switch(config)# interface ethernet 1/4
switch(config-if)# spanning-tree port type network
switch(config-if)# exit
switch(config)#
```

Enabling BPDU Guard Globally

You can enable BPDU Guard globally by default. In this condition, the system shuts down an edge port that receives a BPDU.

Note We recommend that you enable BPDU Guard on all edge ports.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you are configuring the ports correctly to the device to which the port is connected.

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **spanning-tree port type edge bpduguard default**	Enables BPDU Guard by default on all spanning tree edge ports. By default, global BPDU Guard is disabled.
Step 3	switch(config)# **exit**	Exits global configuration mode.
Step 4	(Optional) switch# **show spanning-tree summary**	Displays summary STP information.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to enable BPDU Guard on all spanning tree edge ports:

```
switch# config t
switch(confiig)# spanning-tree port type edge bpduguard default
switch(config)# exit
switch#
```

Enabling BPDU Guard on Specified Interfaces

Enabling BPDU Guard shuts down the port if it receives an invalid BPDU. You can configure BPDU Guard on specified interfaces as follows:

- **spanning-tree bpduguard enable** —Unconditionally enables BPDU Guard on the interface.
- **spanning-tree bpduguard disable** —Unconditionally disables BPDU Guard on the interface.
- **no spanning-tree bpduguard** —Enables BPDU Guard on the interface if it is an operational edge port and if the **spanning-tree port type edge bpduguard default** command is configured.

Note In Cisco NX-OS Release 6.2(10) and later releases, the port will be error disabled when a BPDU is received on any VLAN and BPDU Guard is enabled on the port.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- For a trunk port, configure an allowed VLAN list using the **switchport trunk allowed vlan** *vlan-list* command.

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td><b>Step 1</b></td><td>switch# <b>configure terminal</b></td><td>Enters global configuration mode.</td></tr>
<tr><td><b>Step 2</b></td><td>switch(config)# <b>interface</b> <i>type slot/port</i></td><td>Specifies the interface to configure, and enters the interface configuration mode.</td></tr>
<tr><td><b>Step 3</b></td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td><b>spanning-tree bpduguard</b> {<b>enable</b> | <b>disable</b>}</td><td>Enables or disables BPDU Guard for the specified spanning tree edge interface. By default, BPDU Guard is disabled on the interfaces.</td></tr>
<tr><td><b>no spanning-tree bpduguard</b></td><td>Falls back to the default BPDU Guard global setting that you set for the interfaces by entering the <b>spanning-tree port type edge bpduguard default</b> command.</td></tr>
</table></td><td></td></tr>
<tr><td><b>Step 4</b></td><td>switch(config-if)# <b>exit</b></td><td>Exits interface configuration mode.</td></tr>
<tr><td><b>Step 5</b></td><td>(Optional) switch(config)# <b>show spanning-tree summary interface</b> <i>type slot/port</i> <b>detail</b></td><td>Displays summary STP information.</td></tr>
<tr><td><b>Step 6</b></td><td>(Optional) switch(config)# <b>copy running-config startup-config</b></td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to explicitly enable BPDU Guard on the Ethernet edge port 1/4:

```
switch# config t
switch(config)# interface ethernet 1/4
switch(config-if)# spanning-tree bpduguard enable
switch(config-if)# exit
switch(config)#
```

Enabling BPDU Filtering Globally

You can enable BPDU Filtering globally by default on spanning tree edge ports.

If an edge port with BPDU Filtering enabled receives a BPDU, it loses its operation status as edge port and resumes the regular STP transitions. However, this port maintains its configuration as an edge port.

Caution Be careful when using this command. Using this command incorrectly can cause bridging loops.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you have configured some spanning tree edge ports.

Note When enabled globally, BPDU Filtering is applied only on ports that are operational edge ports. Ports send a few BPDUs at linkup before they effectively filter outbound BPDUs. If a BPDU is received on an edge port, it immediately loses its operational edge port status and BPDU Filtering is disabled.

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **spanning-tree port type edge bpdufilter default**	Enables BPDU Filtering by default on all operational spanning tree edge ports. Global BPDU Filtering is disabled by default.
Step 3	switch(config)# **exit**	Exits global configuration mode.
Step 4	(Optional) switch# **show spanning-tree summary**	Displays summary STP information.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to enable BPDU Filtering on all operational spanning tree edge ports:

```
switch# config t
switch(config)# spanning-tree port type edge bpdufilter default
switch(config)# exit
switch#
```

Enabling BPDU Filtering on Specified Interfaces

You can apply BPDU Filtering to specified interfaces. When enabled on an interface, that interface does not send any BPDUs and drops all BPDUs that it receives. This BPDU Filtering functionality applies to the entire interface, whether trunking or not.

Caution Be careful when you enter the **spanning-tree bpdufilter enable** command on specified interfaces. Explicitly configuring BPDU Filtering on a port that is not connected to a host can result in bridging loops because the port will ignore any BPDU that it receives and go to forwarding.

You can enter this command to override the port configuration on specified interfaces.

This command has three states:

- **spanning-tree bpdufilter enable**—Unconditionally enables BPDU Filtering on the interface.
- **spanning-tree bpdufilter disable**—Unconditionally disables BPDU Filtering on the interface.
- **no spanning-tree bpdufilter** —Enables BPDU Filtering on the interface if the interface is an operational edge port and if you configure the **spanning-tree port type edge bpdufilter default** command.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.

Note When you enable BPDU Filtering locally on a port, this feature prevents the device from receiving or sending BPDUs on this port.

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to configure and enters interface configuration mode.
Step 3	Enter one of the following commands: **Option**: **spanning-tree bpdufilter** {**enable** | **disable**} — **Description**: Enables or disables BPDU Filtering for the specified spanning tree edge interface. By default, BPDU Filtering is disabled. **Option**: **no spanning-tree bpdufilter** — **Description**: Enables BPDU Filtering on the interface if the interface is an operational spanning tree edge port and if you enter the **spanning-tree port type edge bpdufilter default** command.	

	Command or Action	Purpose
Step 4	switch(config-if)# **exit**	Exits interface configuration mode.
Step 5	(Optional) switch# **show spanning-tree summary interface** *type slot/port* **detail**	Displays summary STP information.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to explicitly enable BPDU Filtering on the Ethernet spanning tree edge port 1/4:

```
switch# config t
switch(config)# interface ethernet 1/4
switch(config-if)# spanning-tree bpdufilter enable
switch(config-if)# exit
switch(config)#
```

Enabling Loop Guard Globally

You can enable Loop Guard globally by default on all point-to-point spanning tree normal and network ports. Loop Guard does not run on edge ports.

Loop Guard provides additional security in the bridge network. Loop Guard prevents alternate or root ports from becoming the designated port because of a failure that could lead to a unidirectional link.

Note Entering the Loop Guard command for the specified interface overrides the global Loop Guard command.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you have spanning tree normal ports or have configured some network ports.

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **spanning-tree loopguard default**	Enables Loop Guard by default on all spanning tree normal and network ports. By default, global Loop Guard is disabled.
Step 3	switch(config)# **exit**	Exits global configuration mode.

	Command or Action	Purpose
Step 4	(Optional) switch# **show spanning-tree summary**	Displays summary STP information.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to enable Loop Guard on all spanning tree normal or network ports:

```
switch# config t
switch(config)# spanning-tree loopguard default
switch(config)# exit
switch#
```

Enabling Loop Guard or Root Guard on Specified Interfaces

Note You can run Loop Guard on spanning tree normal or network ports. You can run Root Guard on all spanning tree ports: normal, edge, or network.

You can enable either Loop Guard or Root Guard on specified interfaces.

Enabling Root Guard on a port means that port cannot become a root port, and Loop Guard prevents alternate or root ports from becoming the designated port because of a failure that could lead to a unidirectional link.

Both Loop Guard and Root Guard enabled on an interface apply to all VLANs to which that interface belongs.

Note Entering the Loop Guard command for the specified interface overrides the global Loop Guard command.

Before you begin

- Ensure that you are in the correct VDC (or enter the **switchto vdc** command).
- Ensure that STP is configured.
- Ensure that you are configuring Loop Guard on spanning tree normal or network ports.

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to configure and enters interface configuration mode.

	Command or Action	Purpose
Step 3	switch(config-if)# **spanning-tree guard** {**loop** | **root** | **none**}	Enables or disables either Loop Guard or Root Guard for the specified interface. By default, Root Guard is disabled by default, and Loop Guard on specified ports is also disabled. **Note** Loop Guard runs only on spanning tree normal and network interfaces. This example shows Loop Guard is enabled on the specified interface.
Step 4	switch(config-if)# **exit**	Exits interface configuration mode.
Step 5	switch(config)# **interface** *type slot/port*	Specifies the interface to configure and enters interface configuration mode.
Step 6	switch(config-if)# **spanning-tree guard** {**loop** | **root** | **none**}	Enables or disables either Loop Guard or Root Guard for the specified interface. By default, Root Guard is disabled by default, and Loop Guard on specified ports is also disabled. The example shows Root Guard is enabled on a different interface.
Step 7	switch(config-if)# **exit**	Exits interface mode.
Step 8	(Optional) switch# **show spanning-tree interface** *type slot/port* **detail**	Displays summary STP information.
Step 9	(Optional) switch(config)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to enable Root Guard on Ethernet port 1/4:

```
switch# config t
switch(config)# interface etherent 1/4
switch(config-if)# spanning-tree guard root
switch(config-if)# exit
switch(config)#
```

Configuring PVST Simulation Globally—CLI Version

Note PVST simulation is enabled by default. By default, all interfaces on the device interoperate between MST and Rapid PVST+.

MST interoperates with Rapid PVST+. However, to prevent an accidental connection to a device that does not run MST as the default STP mode, you may want to disable this automatic feature. If you disable PVST simulation, the MST-enabled port moves to the blocking state once it detects it is connected to a Rapid PVST+-enabled port. This port remains in the inconsistent state until the port stops receiving BPDUs, and then the port resumes the normal STP transition process.

You can block this automatic feature either globally or per port. You can enter the global command and change the PVST simulation setting for the entire device while you are in interface configuration command mode.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **config t**	Enters global configuration mode.
Step 2	switch(config)# **no spanning-tree mst simulate pvst global**	Disables all interfaces on the switch from automatically interoperating with a connected device that is running in Rapid PVST+ mode. The default for this feature is enabled; by default, all interfaces on the device operate between Rapid PVST+ and MST.
Step 3	switch(config)# **exit**	Exits global configuration mode.
Step 4	(Optional) switch# **show spanning-tree summary**	Displays detailed STP information.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to prevent the device from automatically interoperating with a connecting device that is running Rapid PVST+:

```
switch# config t
switch(config)# no spanning-tree mst simulate pvst global
switch(config)# exit
switch#
```

Configuring PVST Simulation Per Port

Note PVST simulation is enabled by default. By default, all interfaces on the device interoperate between MST and Rapid PVST+.

You can configure PVST simulation only when you are running MST on the device (Rapid PVST+ is the default STP mode). MST interoperates with Rapid PVST+. However, to prevent an accidental connection to a device that does not run MST as the default STP mode, you may want to disable this automatic feature. If you disable PVST simulation, the MST-enabled port moves to the blocking state once it detects that it is connected to a Rapid PVST+-enabled port. This port remains in the inconsistent state until the port stops receiving Rapid PVST+ BPDUs, and then the port resumes the normal STP transition process.

You can block this automatic feature either globally or per port.

Before you begin

Ensure that you are in the correct VDC (or enter the **switchto vdc** command).

Procedure

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td>Step 1</td><td>switch# **config t**</td><td>Enters global configuration mode.</td></tr>
<tr><td>Step 2</td><td>switch(config)# **interface** {{*type slot/port*} |{**port-channel** *number*}}</td><td>Specifies an interface to configure and enters interface configuration mode.</td></tr>
<tr><td>Step 3</td><td>Enter one of the following commands:
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>**spanning-tree mst simulate pvst disable**</td><td>Disables specified interfaces from automatically interoperating with a connected device that is running in Rapid PVST+ mode.
By default, all interfaces on the device operate between Rapid PVST+ and MST.</td></tr>
<tr><td>**spanning-tree mst simulate pvst**</td><td>Reenables seamless operation between MST and Rapid PVST+ on specified interfaces.</td></tr>
<tr><td>**no spanning-tree mst simulate pvst**</td><td>Sets the interface to the device-wide MST and Rapid PVST+ interoperation that you configured using the **spanning-tree mst simulate pvst global** command.</td></tr>
</table></td><td></td></tr>
<tr><td>Step 4</td><td>switch(config-if)# **exit**</td><td>Exits interface configuration mode.</td></tr>
<tr><td>Step 5</td><td>(Optional) switch# **show spanning-tree interface** *type slot/port* **detail**</td><td>Displays detailed STP information.</td></tr>
<tr><td>Step 6</td><td>(Optional) switch(config)# **copy running-config startup-config**</td><td>Copies the running configuration to the startup configuration.</td></tr>
</table>

Example

This example shows how to prevent the specified interfaces from automatically interoperating with a connecting device that is not running MST:

```
switch(config-if)# spanning-tree mst simulate pvst
switch(config-if)#
```

Verifying the STP Extension Configuration

To display the configuration information for the STP extensions, perform one of the following tasks:

Command	Purpose
show running-config spanning-tree [**all**]	Displays information about STP.
show spanning-tree summary	Displays summary information on STP.
show spanning-tree mst *instance-id* **interface** {**ethernet** *slot/port* | **port-channel** *channel-number*} [**detail**]	Displays MST information for the specified interface and instance.

For information on the output of these commands, see the *Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*.

Configuration Examples for STP Extension

The following example shows how to configure the STP extensions:

```
switch# configure terminal
switch(config)# spanning-tree port type network default
switch(config)# spanning-tree port type edge bpduguard default
switch(config)# spanning-tree port type edge bpdufilter default

switch(config)# interface ethernet 1/1
switch(config-if)# spanning-tree port type edge
switch(config-if)# exit

switch(config)# interface ethernet 1/2
switch(config-if)# spanning-tree port type edge
switch(config-if)# exit
switch(config)#
```

Additional References for STP Extensions—CLI Version

Related Documents

Related Topic	Document Title
Command reference	*Cisco Nexus 7000 Series NX-OS Layer 2 Switching Command Reference*

Related Topic	Document Title
Layer 2 interfaces	*Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*
NX-OS fundamentals	*Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide*
High availability	*Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide*
System management	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*
Licensing	*Cisco NX-OS Licensing Guide*
Release notes	*Cisco Nexus 7000 Series NX-OS Release Notes*

Standards

Standards	Title
IEEE 802.1Q-2006 (formerly known as IEEE 802.1s), IEEE 802.1D-2004 (formerly known as IEEE 802.1w), IEEE 802.1D, IEEE 802.1t	—

MIBs

MIBs	MIBs Link
• CISCO-STP-EXTENSION-MIB • BRIDGE-MIB	To locate and download MIBs, go to the following URL: https://cfnng.cisco.com/mibs.

Feature History for Configuring STP Extensions—CLI version

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Table 21: Feature History for Configuring STP Extensions

Feature Name	Releases	Feature Information	
BPDU Guard	6.2(10)	Added support for BPDU Guard error disable.	
No change.	4.2(1)	—	
No change.	4.1(2)	—	

APPENDIX A

Configuration Limits for Layer 2 Switching

Configuration Limits for Layer 2 Switching

The configuration limits are documented in the *Cisco Nexus 7000 Series NX-OS Verified Scalability Guide*.