



Layer 2/3 Configuration Guide, Cisco IOS XE Release 3.6E (Catalyst 3650 Switches)

First Published: June 27, 2014

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-32627-01



CONTENTS

Preface

Preface xi

Document Conventions xi

Related Documentation xiii

Obtaining Documentation and Submitting a Service Request xiii

CHAPTER 1

Configuring Spanning Tree Protocol 1

Finding Feature Information 1

Restrictions for STP 1

Information About Spanning Tree Protocol 2

Spanning Tree Protocol 2

Spanning-Tree Topology and BPDUs 3

Bridge ID, Device Priority, and Extended System ID 5

Port Priority Versus Path Cost 6

Spanning-Tree Interface States 6

Blocking State 7

Listening State 8

Learning State 8

Forwarding State 8

Disabled State 8

How a Switch or Port Becomes the Root Switch or Root Port 9

Spanning Tree and Redundant Connectivity 9

Spanning-Tree Address Management 10

Accelerated Aging to Retain Connectivity 10

Spanning-Tree Modes and Protocols 11

Supported Spanning-Tree Instances 11

Spanning-Tree Interoperability and Backward Compatibility 12

STP and IEEE 802.1Q Trunks 12

| | |
|--|----|
| VLAN-Bridge Spanning Tree | 12 |
| Spanning Tree and Switch Stacks | 13 |
| Default Spanning-Tree Configuration | 13 |
| How to Configure Spanning-Tree Features | 14 |
| Changing the Spanning-Tree Mode | 14 |
| Disabling Spanning Tree | 16 |
| Configuring the Root Switch | 17 |
| Configuring a Secondary Root Device | 18 |
| Configuring Port Priority | 19 |
| Configuring Path Cost | 21 |
| Configuring the Device Priority of a VLAN | 22 |
| Configuring the Hello Time | 24 |
| Configuring the Forwarding-Delay Time for a VLAN | 24 |
| Configuring the Maximum-Aging Time for a VLAN | 25 |
| Configuring the Transmit Hold-Count | 26 |
| Monitoring Spanning-Tree Status | 28 |
| Additional References for Spanning-Tree Protocol | 28 |
| Feature Information for STP | 29 |

CHAPTER 2**Configuring Multiple Spanning-Tree Protocol 31**

| | |
|---------------------------------|----|
| Finding Feature Information | 31 |
| Prerequisites for MSTP | 31 |
| Restrictions for MSTP | 32 |
| Information About MSTP | 33 |
| MSTP Configuration | 33 |
| MSTP Configuration Guidelines | 33 |
| Root Switch | 34 |
| Multiple Spanning-Tree Regions | 35 |
| IST, CIST, and CST | 35 |
| Operations Within an MST Region | 36 |
| Operations Between MST Regions | 36 |
| IEEE 802.1s Terminology | 37 |
| Illustration of MST Regions | 38 |
| Hop Count | 38 |
| Boundary Ports | 39 |

| | |
|---|----|
| IEEE 802.1s Implementation | 39 |
| Port Role Naming Change | 40 |
| Interoperation Between Legacy and Standard Switches | 40 |
| Detecting Unidirectional Link Failure | 41 |
| MSTP and Switch Stacks | 41 |
| Interoperability with IEEE 802.1D STP | 41 |
| RSTP Overview | 42 |
| Port Roles and the Active Topology | 42 |
| Rapid Convergence | 43 |
| Synchronization of Port Roles | 44 |
| Bridge Protocol Data Unit Format and Processing | 45 |
| Processing Superior BPDU Information | 46 |
| Processing Inferior BPDU Information | 46 |
| Topology Changes | 46 |
| Protocol Migration Process | 47 |
| Default MSTP Configuration | 47 |
| How to Configure MSTP Features | 48 |
| Specifying the MST Region Configuration and Enabling MSTP | 48 |
| Configuring the Root Switch | 51 |
| Configuring a Secondary Root Switch | 52 |
| Configuring Port Priority | 53 |
| Configuring Path Cost | 55 |
| Configuring the Switch Priority | 57 |
| Configuring the Hello Time | 58 |
| Configuring the Forwarding-Delay Time | 59 |
| Configuring the Maximum-Aging Time | 60 |
| Configuring the Maximum-Hop Count | 61 |
| Specifying the Link Type to Ensure Rapid Transitions | 62 |
| Designating the Neighbor Type | 64 |
| Restarting the Protocol Migration Process | 65 |
| Monitoring MST Configuration and Status | 66 |
| Additional References for MSTP | 67 |
| Feature Information for MSTP | 68 |

| | |
|---|----|
| Finding Feature Information | 69 |
| Restriction for Optional Spanning-Tree Features | 69 |
| Information About Optional Spanning-Tree Features | 70 |
| PortFast | 70 |
| BPDU Guard | 70 |
| BPDU Filtering | 71 |
| UplinkFast | 72 |
| Cross-Stack UplinkFast | 73 |
| How Cross-Stack UplinkFast Works | 74 |
| Events That Cause Fast Convergence | 76 |
| BackboneFast | 76 |
| EtherChannel Guard | 79 |
| Root Guard | 79 |
| Loop Guard | 80 |
| How to Configure Optional Spanning-Tree Features | 80 |
| Enabling PortFast | 80 |
| Enabling BPDU Guard | 82 |
| Enabling BPDU Filtering | 83 |
| Enabling UplinkFast for Use with Redundant Links | 85 |
| Disabling UplinkFast | 86 |
| Enabling BackboneFast | 87 |
| Enabling EtherChannel Guard | 89 |
| Enabling Root Guard | 90 |
| Enabling Loop Guard | 91 |
| Monitoring the Spanning-Tree Status | 93 |
| Additional References for Optional Spanning Tree Features | 93 |
| Feature Information for Optional Spanning-Tree Features | 94 |

CHAPTER 4

| | |
|----------------------------------|-----------|
| Configuring EtherChannels | 95 |
| Finding Feature Information | 95 |
| Restrictions for EtherChannels | 95 |
| Information About EtherChannels | 96 |
| EtherChannel Overview | 96 |
| EtherChannel Modes | 97 |
| EtherChannel on Switches | 98 |

| | |
|--|-----|
| EtherChannel Link Failover | 100 |
| Channel Groups and Port-Channel Interfaces | 100 |
| Port Aggregation Protocol | 102 |
| PAgP Modes | 102 |
| Silent Mode | 103 |
| PAgP Learn Method and Priority | 103 |
| PAgP Interaction with Other Features | 104 |
| Link Aggregation Control Protocol | 105 |
| LACP Modes | 105 |
| LACP and Link Redundancy | 106 |
| LACP Interaction with Other Features | 106 |
| EtherChannel On Mode | 106 |
| Load-Balancing and Forwarding Methods | 107 |
| MAC Address Forwarding | 107 |
| IP Address Forwarding | 108 |
| Load-Balancing Advantages | 108 |
| EtherChannel and Switch Stacks | 110 |
| Switch Stack and PAgP | 111 |
| Switch Stacks and LACP | 111 |
| Default EtherChannel Configuration | 111 |
| EtherChannel Configuration Guidelines | 112 |
| Layer 2 EtherChannel Configuration Guidelines | 114 |
| Layer 3 EtherChannel Configuration Guidelines | 115 |
| How to Configure EtherChannels | 115 |
| Configuring Layer 2 EtherChannels | 115 |
| Configuring Layer 3 EtherChannels | 117 |
| Configuring EtherChannel Load-Balancing | 119 |
| Configuring EtherChannel Extended Load-Balancing | 121 |
| Configuring the PAgP Learn Method and Priority | 122 |
| Configuring LACP Hot-Standby Ports | 124 |
| Configuring the LACP Max Bundle Feature | 124 |
| Configuring the Port Channel Min-Links Feature | 125 |
| Configuring the LACP System Priority | 126 |
| Configuring the LACP Port Priority | 127 |
| Monitoring EtherChannel, PAgP, and LACP Status | 129 |

| | |
|--|-----|
| Configuration Examples for Configuring EtherChannels | 130 |
| Configuring Layer 2 EtherChannels: Examples | 130 |
| Configuring Layer 3 EtherChannels: Examples | 131 |
| Configuring LACP Hot-Standby Ports: Example | 131 |
| Additional References for EtherChannels | 132 |
| Feature Information for EtherChannels | 133 |

CHAPTER 5

| | |
|--|------------|
| Configuring Flex Links and the MAC Address-Table Move Update Feature | 135 |
| Finding Feature Information | 135 |
| Restrictions for Configuring Flex Links and MAC Address-Table Move Update | 135 |
| Information About Flex Links and MAC Address-Table Move Update | 136 |
| Flex Links | 136 |
| Flex Links Configuration | 137 |
| VLAN Flex Links Load Balancing and Support | 138 |
| Multicast Fast Convergence with Flex Links Failover | 138 |
| Learning the Other Flex Links Port as the mrouter Port | 139 |
| Generating IGMP Reports | 139 |
| Leaking IGMP Reports | 139 |
| MAC Address-Table Move Update | 139 |
| Flex Links VLAN Load Balancing Configuration Guidelines | 142 |
| MAC Address-Table Move Update Configuration Guidelines | 142 |
| Default Flex Links and MAC Address-Table Move Update Configuration | 142 |
| How to Configure Flex Links and the MAC Address-Table Move Update Feature | 143 |
| Configuring Flex Links | 143 |
| Configuring a Preemption Scheme for a Pair of Flex Links | 144 |
| Configuring VLAN Load Balancing on Flex Links | 146 |
| Configuring MAC Address-Table Move Update | 147 |
| Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages | 148 |
| Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update | 149 |
| Configuration Examples for Flex Links | 150 |
| Configuring Flex Links: Examples | 150 |
| Configuring VLAN Load Balancing on Flex Links: Examples | 151 |
| Configuring the MAC Address-Table Move Update: Examples | 152 |

| | |
|---|-----|
| Configuring Multicast Fast Convergence with Flex Links Failover: Examples | 152 |
| Additional References for Flex Links and MAC Address-Table Move Update | 155 |
| Feature Information for Flex Links and MAC Address-Table Move Update | 156 |

CHAPTER 6**Configuring UniDirectional Link Detection 157**

| | |
|--|-----|
| Finding Feature Information | 157 |
| Restrictions for Configuring UDLD | 157 |
| Information About UDLD | 158 |
| Modes of Operation | 158 |
| Normal Mode | 158 |
| Aggressive Mode | 159 |
| Methods to Detect Unidirectional Links | 159 |
| Neighbor Database Maintenance | 159 |
| Event-Driven Detection and Echoing | 160 |
| UDLD Reset Options | 160 |
| Default UDLD Configuration | 160 |
| How to Configure UDLD | 161 |
| Enabling UDLD Globally | 161 |
| Enabling UDLD on an Interface | 162 |
| Monitoring and Maintaining UDLD | 164 |
| Additional References for UDLD | 164 |
| Feature Information for UDLD | 165 |



Preface

- [Document Conventions](#), page xi
- [Related Documentation](#), page xiii
- [Obtaining Documentation and Submitting a Service Request](#), page xiii

Document Conventions

This document uses the following conventions:

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

| Convention | Description |
|-------------|---|
| {x y} | Required alternative keywords are grouped in braces and separated by vertical bars. |
| [x {y z}] | Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element. |
| string | A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks. |
| <> | Nonprinting characters such as passwords are in angle brackets. |
| [] | Default responses to system prompts are in square brackets. |
| !, # | An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line. |

Reader Alert Conventions

This document may use the following conventions for reader alerts:



Note

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



Tip

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



Caution

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



Timesaver

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



Warning

IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device. Statement 1071

SAVE THESE INSTRUCTIONS

Related Documentation

**Note**

Before installing or upgrading the switch, refer to the switch release notes.

- Cisco Catalyst 3650 Switch documentation, located at:
http://www.cisco.com/go/cat3650_docs
- Cisco SFP and SFP+ modules documentation, including compatibility matrixes, located at:
http://www.cisco.com/en/US/products/hw/modules/ps5455/tsd_products_support_series_home.html
- Cisco Validated Designs documents, located at:
<http://www.cisco.com/go/designzone>
- Error Message Decoder, located at:
<https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi>

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>

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



CHAPTER

1

Configuring Spanning Tree Protocol

- [Finding Feature Information, page 1](#)
- [Restrictions for STP, page 1](#)
- [Information About Spanning Tree Protocol, page 2](#)
- [How to Configure Spanning-Tree Features, page 14](#)
- [Monitoring Spanning-Tree Status, page 28](#)
- [Additional References for Spanning-Tree Protocol, page 28](#)
- [Feature Information for STP, page 29](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for STP

- An attempt to configure a switch as the root switch fails if the value necessary to be the root switch is less than 1.
- If your network consists of switches that support and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.
- The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Related Topics

- [Configuring the Root Switch](#), on page 17
- [Bridge ID, Device Priority, and Extended System ID](#), on page 5
- [Spanning-Tree Topology and BPDUs](#), on page 3
- [Accelerated Aging to Retain Connectivity](#), on page 10

Information About Spanning Tree Protocol

Spanning Tree Protocol

Spanning Tree Protocol (STP) is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree 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.

The STP uses a spanning-tree algorithm to select one switch of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The switch that has *all* of its ports as the designated role or as the backup role is the root switch. The switch that has at least *one* of its ports in the designated role is called the designated switch.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a switch are part of a loop, the spanning-tree and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.

**Note**

By default, the switch sends keepalive messages (to ensure the connection is up) only on interfaces that do not have small form-factor pluggable (SFP) modules. You can change the default for an interface by entering the **[no] keepalive** interface configuration command with no keywords.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch. In a switch stack, all switches use the same bridge ID for a given spanning-tree instance.
- The spanning-tree path cost to the root switch.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface.

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the switch that the sending switch identifies as the root switch
- The spanning-tree path cost to the root
- The bridge ID of the sending switch
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One switch in the network is elected as the root switch (the logical center of the spanning-tree topology in a switched network). See the figure following the bullets.

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

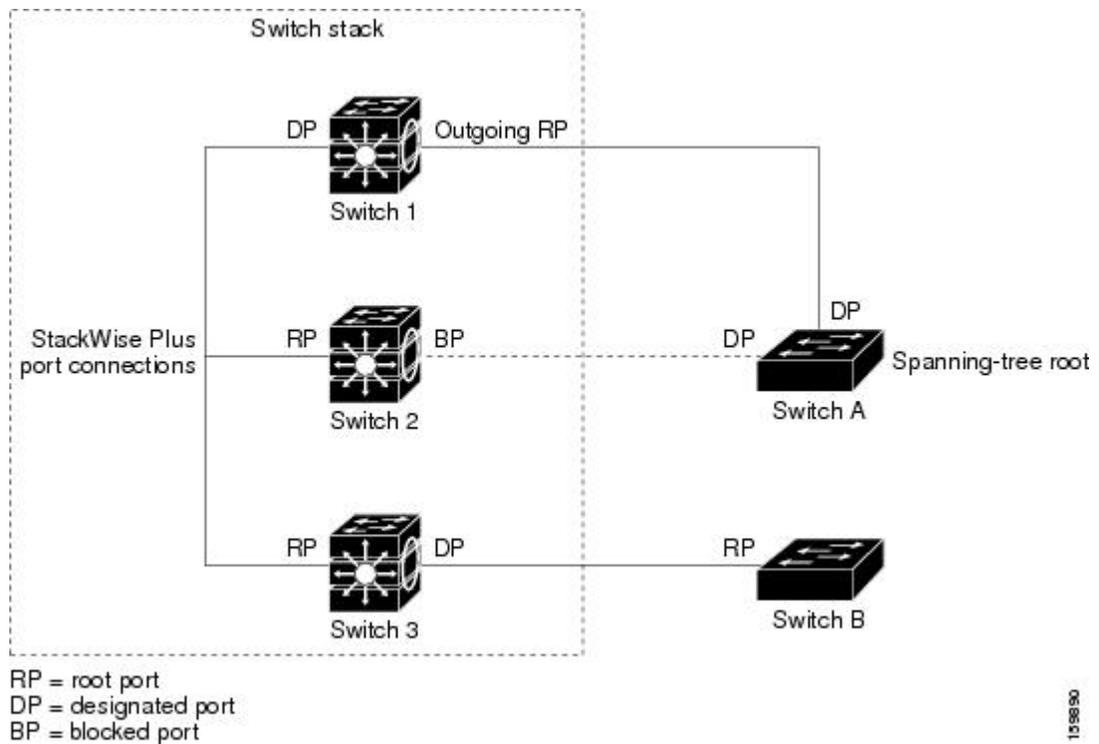
- A root port is selected for each switch (except the root switch). This port provides the best path (lowest cost) when the switch forwards packets to the root switch.

When selecting the root port on a switch stack, spanning tree follows this sequence:

- Selects the lowest root bridge ID
 - Selects the lowest path cost to the root switch
 - Selects the lowest designated bridge ID
 - Selects the lowest designated path cost
 - Selects the lowest port ID
- Only one outgoing port on the stack root switch is selected as the root port. The remaining switches in the stack become its designated switches (Switch 2 and Switch 3) as shown in the following figure.
 - The shortest distance to the root switch is calculated for each switch based on the path cost.
 - A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.

One stack member is elected as the stack root switch. The stack root switch contains the outgoing root port (Switch 1).

Figure 1: Spanning-Tree Port States in a Switch Stack



All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

Related Topics

[Configuring the Root Switch](#), on page 17

[Restrictions for STP, on page 1](#)

Bridge ID, Device Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has a unique bridge identifier (bridge ID), which controls the selection of the root switch. Because each VLAN is considered as a different *logical bridge* with PVST+ and Rapid PVST+, the same switch must have a different bridge ID for each configured VLAN. Each VLAN on the switch has a unique 8-byte bridge ID. The 2 most-significant bytes are used for the switch priority, and the remaining 6 bytes are derived from the switch MAC address.

The switch supports the IEEE 802.1t spanning-tree extensions, and some of the bits previously used for the switch priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the switch, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID.

The 2 bytes previously used for the switch priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID.

Table 1: Device Priority Value and Extended System ID

| 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 |

Spanning tree uses the extended system ID, the switch priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN. Because the switch stack appears as a single switch to the rest of the network, all switches in the stack use the same bridge ID for a given spanning tree. If the stack master fails, the stack members recalculate their bridge IDs of all running spanning trees based on the new MAC address of the new stack master.

Support for the extended system ID affects how you manually configure the root switch, the secondary root switch, and the switch priority of a VLAN. For example, when you change the switch priority value, you change the probability that the switch will be elected as the root switch. Configuring a higher value decreases the probability; a lower value increases the probability.

If any root switch for the specified VLAN has a switch priority lower than 24576, the switch sets its own priority for the specified VLAN to 4096 less than the lowest switch priority. 4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in the table.

Related Topics

[Configuring the Root Switch, on page 17](#)

[Restrictions for STP, on page 1](#)

[Configuring the Root Switch, on page 51](#)

[Root Switch, on page 34](#)

[Specifying the MST Region Configuration and Enabling MSTP, on page 48](#)

Port Priority Versus Path Cost

If a loop occurs, spanning tree uses port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

The spanning-tree path cost default value is derived from the media speed of an interface. If a loop occurs, spanning tree uses 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 values that you want selected last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

If your switch is a member of a switch stack, you must assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last instead of adjusting its port priority. For details, see Related Topics.

Related Topics

[Configuring Port Priority](#) , on page 19

[Configuring Path Cost](#) , on page 21

Spanning-Tree Interface States

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 an interface transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

Each Layer 2 interface on a switch using spanning tree exists in one of these states:

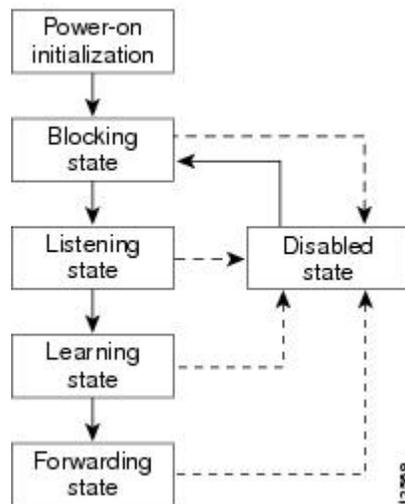
- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree decides that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

An interface moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

An interface moves through the states.

Figure 2: Spanning-Tree Interface States



When you power up the switch, spanning tree is enabled by default, and every interface in the switch, VLAN, or network goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 interface in the forwarding state, this process occurs:

- 1 The interface is in the listening state while spanning tree waits for protocol information to move the interface to the blocking state.
- 2 While spanning tree waits for the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
- 3 In the learning state, the interface continues to block frame forwarding as the switch learns end-station location information for the forwarding database.
- 4 When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each switch interface. A switch initially functions as the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root switch. If there is only one switch in the network, no exchange occurs, the forward-delay timer expires, and the interface moves to the listening state. An interface always enters the blocking state after switch initialization.

An interface in the blocking state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree decides that the interface should participate in frame forwarding.

An interface in the listening state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

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

An interface in the forwarding state performs these functions:

- Receives and forwards frames received on the interface
- Forwards frames switched from another interface
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

A disabled interface performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses

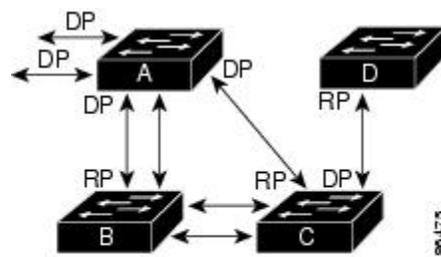
- Does not receive BPDUs

How a Switch or Port Becomes the Root Switch or Root Port

If all switches in a network are enabled with default spanning-tree settings, the switch with the lowest MAC address becomes the root switch.

Switch A is elected as the root switch because the switch priority of all the switches is set to the default (32768) and Switch A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical value) of the ideal switch so that it becomes the root switch, you force a spanning-tree recalculation to form a new topology with the ideal switch as the root.

Figure 3: Spanning-Tree Topology



RP = Root Port
DP = Designated Port

When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the 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 Gigabit Ethernet link and that another port on Switch B (a 10/100 link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet port to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet port becomes the new root port.

Related Topics

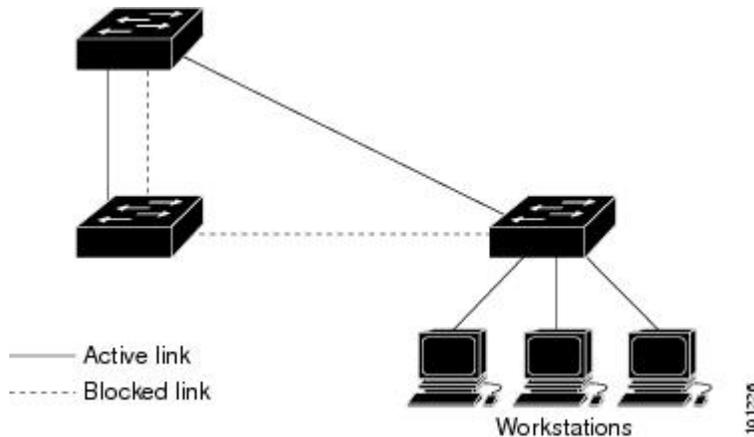
[Configuring Port Priority](#) , on page 19

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces to another device or to two different devices. Spanning tree automatically disables one interface but enables it if the other one fails. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds

are the same, the port priority and port ID are added together, and spanning tree disables the link with the lowest value.

Figure 4: Spanning Tree and Redundant Connectivity



You can also create redundant links between switches by using EtherChannel groups.

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C200000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, each switch in the stack receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the switch or on each switch in the stack receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the switch or each switch in the stack forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan *vlan-id* forward-time seconds** global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the switch accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the switch.

Related Topics

[Configuring the Root Switch](#), on page 17

[Restrictions for STP](#), on page 1

Spanning-Tree Modes and Protocols

The switch supports these spanning-tree modes and protocols:

- **PVST+**—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. It is the default spanning-tree mode used on all Ethernet port-based VLANs. The PVST+ runs on each VLAN on the switch up to the maximum supported, ensuring that each has a loop-free path through the network.

The PVST+ provides Layer 2 load-balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root switch. This root switch propagates the spanning-tree information associated with that VLAN to all other switches in the network. Because each switch has the same information about the network, this process ensures that the network topology is maintained.

- **Rapid PVST+**—This spanning-tree mode is the same as PVST+ except that it uses a rapid convergence based on the IEEE 802.1w standard. To provide rapid convergence, the Rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

Rapid PVST+ uses the same configuration as PVST+ (except where noted), and the switch needs only minimal extra configuration. The benefit of Rapid PVST+ is that you can migrate a large PVST+ install base to Rapid PVST+ without having to learn the complexities of the Multiple Spanning Tree Protocol (MSTP) configuration and without having to reprovision your network. In Rapid PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

- **MSTP**—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w), which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. In a switch stack, the cross-stack rapid transition (CSRT) feature performs the same function as RSTP. You cannot run MSTP without RSTP or CSRT.

Related Topics

[Changing the Spanning-Tree Mode](#), on page 14

Supported Spanning-Tree Instances

In PVST+ or Rapid PVST+ mode, the switch or switch stack supports up to 128 spanning-tree instances.

In MSTP mode, the switch or switch stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

Related Topics

[Disabling Spanning Tree](#), on page 16

[Default Spanning-Tree Configuration](#), on page 13

[Default MSTP Configuration](#), on page 47

Spanning-Tree Interoperability and Backward Compatibility

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ switch cannot connect to multiple MST regions.

When a network contains switches running Rapid PVST+ and switches running PVST+, we recommend that the Rapid PVST+ switches and PVST+ switches be configured for different spanning-tree instances. In the Rapid PVST+ spanning-tree instances, the root switch must be a Rapid PVST+ switch. In the PVST+ instances, the root switch must be a PVST+ switch. The PVST+ switches should be at the edge of the network.

All stack members run the same version of spanning tree (all PVST+, all Rapid PVST+, or all MSTP).

Table 2: PVST+, MSTP, and Rapid-PVST+ Interoperability and Compatibility

| | PVST+ | MSTP | Rapid PVST+ |
|-------------|-------------------------|-------------------------|------------------------|
| PVST+ | Yes | Yes (with restrictions) | Yes (reverts to PVST+) |
| MSTP | Yes (with restrictions) | Yes | Yes (reverts to PVST+) |
| Rapid PVST+ | Yes (reverts to PVST+) | Yes (reverts to PVST+) | Yes |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

[MSTP Configuration Guidelines](#), on page 33

[Multiple Spanning-Tree Regions](#), on page 35

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch uses PVST+ to provide spanning-tree interoperability. If Rapid PVST+ is enabled, the switch uses it instead of PVST+. The switch combines the spanning-tree instance of the IEEE 802.1Q VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch.

However, all PVST+ or Rapid PVST+ information is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

PVST+ is automatically enabled on IEEE 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports and Inter-Switch Link (ISL) trunk ports is not affected by PVST+.

VLAN-Bridge Spanning Tree

Cisco VLAN-bridge spanning tree is used with the fallback bridging feature (bridge groups), which forwards non-IP protocols such as DECnet between two or more VLAN bridge domains or routed ports. The

VLAN-bridge spanning tree allows the bridge groups to form a spanning tree on top of the individual VLAN spanning trees to prevent loops from forming if there are multiple connections among VLANs. It also prevents the individual spanning trees from the VLANs being bridged from collapsing into a single spanning tree.

To support VLAN-bridge spanning tree, some of the spanning-tree timers are increased. To use the fallback bridging feature, you must have the IP services feature set enabled on your switch.

Spanning Tree and Switch Stacks

When the switch stack is operating in PVST+ or Rapid PVST+ mode:

- A switch stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID for a given spanning tree. The bridge ID is derived from the MAC address of the active switch.
- When a new switch joins the stack, it sets its bridge ID to the active switch bridge ID. If the newly added switch has the lowest ID and if the root path cost is the same among all stack members, the newly added switch becomes the stack root.
- When a stack member leaves the stack, spanning-tree reconvergence occurs within the stack (and possibly outside the stack). The remaining stack member with the lowest stack port ID becomes the stack root.
- If the switch stack is the spanning-tree root and the active switch fails or leaves the stack, the standby switch becomes the new active switch, bridge IDs remain the same, and a spanning-tree reconvergence might occur.
- If a neighboring switch external to the switch stack fails or is powered down, normal spanning-tree processing occurs. Spanning-tree reconvergence might occur as a result of losing a switch in the active topology.
- If a new switch external to the switch stack is added to the network, normal spanning-tree processing occurs. Spanning-tree reconvergence might occur as a result of adding a switch in the network.

Default Spanning-Tree Configuration

Table 3: Default Spanning-Tree Configuration

| Feature | Default Setting |
|---|--|
| Enable state | Enabled on VLAN 1. |
| Spanning-tree mode | PVST+. (Rapid PVST+ and MSTP are disabled.) |
| Switch priority | 32768 |
| Spanning-tree port priority (configurable on a per-interface basis) | 128 |
| Spanning-tree port cost (configurable on a per-interface basis) | 1000 Mb/s: 4 100 Mb/s: 19 10 Mb/s: 100 |

| Feature | Default Setting |
|---|---|
| Spanning-tree VLAN port priority (configurable on a per-VLAN basis) | 128 |
| Spanning-tree VLAN port cost (configurable on a per-VLAN basis) | 1000 Mb/s: 4 100 Mb/s: 19 10 Mb/s: 100 |
| Spanning-tree timers | Hello time: 2 seconds Forward-delay time: 15 seconds Maximum-aging time: 20 seconds Transmit hold count: 6 BPDUs |

Related Topics

[Disabling Spanning Tree](#), on page 16

[Supported Spanning-Tree Instances](#), on page 11

How to Configure Spanning-Tree Features

Changing the Spanning-Tree Mode

The switch supports three spanning-tree modes: per-VLAN spanning tree plus (PVST+), Rapid PVST+, or multiple spanning tree protocol (MSTP). By default, the switch runs the PVST+ protocol.

If you want to enable a mode that is different from the default mode, this procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mode {pvst | mst | rapid-pvst}**
4. **interface *interface-id***
5. **spanning-tree link-type point-to-point**
6. **end**
7. **clear spanning-tree detected-protocols**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|-------------------|--|
| Step 1 | enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|---|
| | <p>Example:</p> <pre>Switch> enable</pre> | |
| Step 2 | <p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre> | Enters the global configuration mode. |
| Step 3 | <p>spanning-tree mode {pvst mst rapid-pvst}</p> <p>Example:</p> <pre>Switch(config)# spanning-tree mode pvst</pre> | <p>Configures a spanning-tree mode. All stack members run the same version of spanning tree.</p> <ul style="list-style-type: none"> • Select pvst to enable PVST+ (the default setting). • Select mst to enable MSTP (and RSTP). • Select rapid-pvst to enable rapid PVST+. |
| Step 4 | <p>interface interface-id</p> <p>Example:</p> <pre>Switch(config)# interface GigabitEthernet1/0/1</pre> | (Recommended for Rapid PVST+ mode only) Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port channels. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. |
| Step 5 | <p>spanning-tree link-type point-to-point</p> <p>Example:</p> <pre>Switch(config-if)# spanning-tree link-type point-to-point</pre> | <p>(Recommended for Rapid PVST+ mode only) Specifies that the link type for this port is point-to-point.</p> <p>If you connect this port (local port) to a remote port through a point-to-point link and the local port becomes a designated port, the switch negotiates with the remote port and rapidly changes the local port to the forwarding state.</p> |
| Step 6 | <p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre> | Returns to privileged EXEC mode. |
| Step 7 | <p>clear spanning-tree detected-protocols</p> <p>Example:</p> <pre>Switch# clear spanning-tree detected-protocols</pre> | <p>(Recommended for Rapid PVST+ mode only) If any port on the switch is connected to a port on a legacy IEEE 802.1D switch, this command restarts the protocol migration process on the entire switch.</p> <p>This step is optional if the designated switch detects that this switch is running rapid PVST+.</p> |

Related Topics

[Spanning-Tree Modes and Protocols, on page 11](#)

Disabling Spanning Tree

Spanning tree is enabled by default on VLAN 1 and on all newly created VLANs up to the spanning-tree limit. Disable spanning tree only if you are sure there are no loops in the network topology.



Caution

When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree vlan *vlan-id***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | no spanning-tree vlan <i>vlan-id</i> Example: Switch(config)# no spanning-tree vlan 300 | For <i>vlan-id</i> , the range is 1 to 4094. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Supported Spanning-Tree Instances, on page 11](#)

[Default Spanning-Tree Configuration, on page 13](#)

Configuring the Root Switch

To configure a switch as the root for the specified VLAN, use the **spanning-tree vlan *vlan-id* root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value. When you enter this command, the software checks the switch priority of the root switches for each VLAN. Because of the extended system ID support, the switch sets its own priority for the specified VLAN to 24576 if this value will cause this switch to become the root for the specified VLAN.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch 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 use the **hello** keyword to override the automatically calculated hello time.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* root primary [diameter *net-diameter*]**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree vlan <i>vlan-id</i> root primary [diameter <i>net-diameter</i>] Example: Switch(config)# spanning-tree vlan 20-24 root primary diameter 4 | Configures a switch to become the root for the specified VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. |

| | Command or Action | Purpose |
|---------------|---|---|
| | | <ul style="list-style-type: none"> (Optional) For diameter <i>net-diameter</i>, specify the maximum number of switches between any two end stations. The range is 2 to 7. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

What to Do Next

After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree vlan *vlan-id* hello-time**, **spanning-tree vlan *vlan-id* forward-time**, and the **spanning-tree vlan *vlan-id* max-age** global configuration commands.

Related Topics

- [Bridge ID, Device Priority, and Extended System ID, on page 5](#)
- [Spanning-Tree Topology and BPDUs, on page 3](#)
- [Accelerated Aging to Retain Connectivity, on page 10](#)
- [Restrictions for STP, on page 1](#)

Configuring a Secondary Root Device

When you configure a switch as the secondary root, the switch priority is modified from the default value (32768) to 28672. With this priority, the switch is likely to become the root switch for the specified VLAN if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768, and therefore, are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree vlan *vlan-id* root primary** global configuration command.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* root secondary [diameter *net-diameter*]**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree vlan <i>vlan-id</i> root secondary [diameter <i>net-diameter</i>] Example: Switch(config)# spanning-tree vlan 20-24 root secondary diameter 4 | Configures a switch to become the secondary root for the specified VLAN. <ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. (Optional) For diameter <i>net-diameter</i>, specify the maximum number of switches between any two end stations. The range is 2 to 7. Use the same network diameter value that you used when configuring the primary root switch. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Configuring Port Priority


Note

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

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree port-priority** *priority*
5. **spanning-tree vlan** *vlan-id* **port-priority** *priority*
6. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/2 | Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (port-channel <i>port-channel-number</i>). |
| Step 4 | spanning-tree port-priority <i>priority</i> Example: Switch(config-if)# spanning-tree port-priority 0 | Configures the port priority for an interface. For <i>priority</i> , the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority. |
| Step 5 | spanning-tree vlan <i>vlan-id</i> port-priority <i>priority</i> Example: Switch(config-if)# spanning-tree vlan 20-25 port-priority 0 | Configures the port priority for a VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>priority</i>, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority. |

| | Command or Action | Purpose |
|---------------|--|----------------------------------|
| Step 6 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[Port Priority Versus Path Cost, on page 6](#)

[How a Switch or Port Becomes the Root Switch or Root Port, on page 9](#)

Configuring Path Cost

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree cost** *cost*
5. **spanning-tree vlan** *vlan-id* **cost** *cost*
6. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |

| | Command or Action | Purpose |
|--------|--|--|
| Step 3 | interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/1</pre> | Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (port-channel <i>port-channel-number</i>). |
| Step 4 | spanning-tree cost <i>cost</i> Example: <pre>Switch(config-if)# spanning-tree cost 250</pre> | <p>Configures the cost for an interface.</p> <p>If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission.</p> <p>For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface.</p> |
| Step 5 | spanning-tree vlan <i>vlan-id</i> cost <i>cost</i> Example: <pre>Switch(config-if)# spanning-tree vlan 10,12-15,20 cost 300</pre> | <p>Configures the cost for a VLAN.</p> <p>If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission.</p> <ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface. |
| Step 6 | end Example: <pre>Switch(config-if)# end</pre> | Returns to privileged EXEC mode. |

The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Related Topics

[Port Priority Versus Path Cost, on page 6](#)

Configuring the Device Priority of a VLAN

You can configure the switch priority and make it more likely that a standalone switch or a switch in the stack will be chosen as the root switch.



Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan *vlan-id* root primary** and the **spanning-tree vlan *vlan-id* root secondary** global configuration commands to modify the switch priority.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* priority *priority***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree vlan <i>vlan-id</i> priority <i>priority</i> Example: Switch(config)# spanning-tree vlan 20 priority 8192 | Configures the switch priority of a VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>priority</i>, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected. |
| Step 4 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root switch. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **spanning-tree vlan *vlan-id* hello-time *seconds***
3. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | spanning-tree vlan <i>vlan-id</i> hello-time <i>seconds</i> Example: Switch(config)# spanning-tree vlan 20-24 hello-time 3 | Configures the hello time of a VLAN. The hello time is the time interval between configuration messages generated and sent by the root switch. These messages mean that the switch is alive. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 1 to 10; the default is 2. |
| Step 3 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Configuring the Forwarding-Delay Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* forward-time *seconds***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree vlan <i>vlan-id</i> forward-time <i>seconds</i> Example: Switch(config)# spanning-tree vlan 20,25 forward-time 18 | Configures the forward time of a VLAN. The forwarding delay is the number of seconds an interface waits before changing from its spanning-tree learning and listening states to the forwarding state. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 4 to 30; the default is 15. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Configuring the Maximum-Aging Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `spanning-tree vlan vlan-id max-age seconds`
4. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# <code>configure terminal</code> | Enters the global configuration mode. |
| Step 3 | spanning-tree vlan <i>vlan-id</i> max-age <i>seconds</i> Example: Switch(config)# <code>spanning-tree vlan 20 max-age 30</code> | Configures the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 6 to 40; the default is 20. |
| Step 4 | end Example: Switch(config-if)# <code>end</code> | Returns to privileged EXEC mode. |

Configuring the Transmit Hold-Count

You can configure the BPDU burst size by changing the transmit hold count value.

**Note**

Changing this parameter to a higher value can have a significant impact on CPU utilization, especially in Rapid PVST+ mode. Lowering this value can slow down convergence in certain scenarios. We recommend that you maintain the default setting.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree transmit hold-count** *value*
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree transmit hold-count <i>value</i> Example: Switch(config)# spanning-tree transmit hold-count 6 | Configures the number of BPDUs that can be sent before pausing for 1 second. For <i>value</i> , the range is 1 to 20; the default is 6. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Monitoring Spanning-Tree Status

Table 4: Commands for Displaying Spanning-Tree Status

| | |
|--|--|
| show spanning-tree active | Displays spanning-tree information on active interfaces only. |
| show spanning-tree detail | Displays a detailed summary of interface information. |
| show spanning-tree vlan <i>vlan-id</i> | Displays spanning-tree information for the specified VLAN. |
| show spanning-tree interface <i>interface-id</i> | Displays spanning-tree information for the specified interface. |
| show spanning-tree interface <i>interface-id</i> portfast | Displays spanning-tree portfast information for the specified interface. |
| show spanning-tree summary [totals] | Displays a summary of interface states or displays the total lines of the STP state section. |

To clear spanning-tree counters, use the **clear spanning-tree [interface *interface-id*]** privileged EXEC command.

Additional References for Spanning-Tree Protocol

Related Documents

| Related Topic | Document Title |
|---------------------------------|---|
| Spanning tree protocol commands | <i>LAN Switching Command Reference, Cisco IOS XE Release 3SE (Catalyst 3650 Switches)</i> |

Error Message Decoder

| Description | Link |
|---|---|
| To help you research and resolve system error messages in this release, use the Error Message Decoder tool. | https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi |

Standards and RFCs

| Standard/RFC | Title |
|--------------|-------|
| None | — |

MIBs

| MIB | MIBs Link |
|--------------------------------------|--|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|---|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | http://www.cisco.com/support |

Feature Information for STP

| Release | Modification |
|--|------------------------------|
| Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE | This feature was introduced. |



Configuring Multiple Spanning-Tree Protocol

- [Finding Feature Information, page 31](#)
- [Prerequisites for MSTP, page 31](#)
- [Restrictions for MSTP, page 32](#)
- [Information About MSTP, page 33](#)
- [How to Configure MSTP Features, page 48](#)
- [Monitoring MST Configuration and Status, page 66](#)
- [Additional References for MSTP, page 67](#)
- [Feature Information for MSTP, page 68](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for MSTP

- For two or more switches to be in the same multiple spanning tree (MST) region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- For two or more stacked switches to be in the same MST region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- For load-balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link. You can achieve load-balancing across a switch stack by manually configuring the path cost.

- For load-balancing between a per-VLAN spanning tree plus (PVST+) and an MST cloud or between a rapid-PVST+ and an MST cloud to work, all MST boundary ports must be forwarding. MST boundary ports are forwarding when the internal spanning tree (IST) master of the MST cloud is the root of the common spanning tree (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 PVST+ or rapid-PVST+ cloud. You might have to manually configure the switches in the clouds.

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#), on page 48

[MSTP Configuration Guidelines](#), on page 33

[Multiple Spanning-Tree Regions](#), on page 35

Restrictions for MSTP

- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.
- The switch stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.
- PVST+, Rapid PVST+, and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run Rapid PVST+, or all VLANs run MSTP.)
- VLAN Trunking Protocol (VTP) propagation of the MST configuration is not supported. However, you can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each switch within the MST region by using the command-line interface (CLI) or through the Simple Network Management Protocol (SNMP) support.
- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.
- A region can have one member or multiple members with the same MST configuration; each member must be capable of processing rapid spanning tree protocol (RSTP) Bridge Protocol Data Units (BPDUs). There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.
- After configuring a switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and the **spanning-tree mst max-age** global configuration commands.

Table 5: PVST+, MSTP, and Rapid PVST+ Interoperability and Compatibility

| | PVST+ | MSTP | Rapid PVST+ |
|-------------|-------------------------|-------------------------|------------------------|
| PVST+ | Yes | Yes (with restrictions) | Yes (reverts to PVST+) |
| MSTP | Yes (with restrictions) | Yes | Yes (reverts to PVST+) |
| Rapid PVST+ | Yes (reverts to PVST+) | Yes (reverts to PVST+) | Yes |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

[MSTP Configuration Guidelines](#), on page 33

[Multiple Spanning-Tree Regions](#), on page 35

[Configuring the Root Switch](#) , on page 51

[Root Switch](#), on page 34

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Information About MSTP

MSTP Configuration

MSTP, which uses RSTP for rapid convergence, enables multiple VLANs to be grouped into and mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).

**Note**

The multiple spanning-tree (MST) implementation is based on the IEEE 802.1s standard.

The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the switch is in the MST mode, the RSTP, which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP), and with existing Cisco PVST+ and rapid per-VLAN spanning-tree plus (Rapid PVST+).

A switch stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same switch ID.

MSTP Configuration Guidelines

- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.
- For configuration guidelines about UplinkFast, BackboneFast, and cross-stack UplinkFast, see the relevant sections in the Related Topics section.

- When the switch is in MST mode, it uses the long path-cost calculation method (32 bits) to compute the path cost values. With the long path-cost calculation method, the following path cost values are supported:

| Speed | Path Cost Value |
|----------|-----------------|
| 10 Mb/s | 2,000,000 |
| 100 Mb/s | 200,000 |
| 1 Gb/s | 20,000 |
| 10 Gb/s | 2,000 |
| 100 Gb/s | 200 |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

[Prerequisites for MSTP](#) , on page 31

[Restrictions for MSTP](#) , on page 32

[Spanning-Tree Interoperability and Backward Compatibility](#) , on page 12

[Optional Spanning-Tree Configuration Guidelines](#)

[BackboneFast](#) , on page 76

[UplinkFast](#) , on page 72

Root Switch

The switch maintains a spanning-tree instance for the group of VLANs mapped to it. A switch ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For a group of VLANs, the switch with the lowest switch ID becomes the root switch.

When you configure a switch as the root, you modify the switch priority from the default value (32768) to a significantly lower value so that the switch becomes the root switch for the specified spanning-tree instance. When you enter this command, the switch checks the switch priorities of the root switches. Because of the extended system ID support, the switch sets its own priority for the specified instance to 24576 if this value will cause this switches to become the root for the specified spanning-tree instance.

If any root switch for the specified instance has a switch priority lower than 24576, the switch sets its own priority to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value. For more information, select "Bridge ID, Switch Priority, and Extended System ID" link in Related Topics.

If your network consists of switches that support and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch 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 use the **hello** keyword to override the automatically calculated hello time.

Related Topics

[Configuring the Root Switch](#) , on page 51

[Restrictions for MSTP](#) , on page 32

[Bridge ID, Device Priority, and Extended System ID](#) , on page 5

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region.

The MST configuration controls to which MST region each switch belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the switch for a region by specifying the MST region configuration on it. You can map VLANs to an MST instance, specify the region name, and set the revision number. For instructions and an example, select the "Specifying the MST Region Configuration and Enabling MSTP" link in Related Topics.

A region can have one or multiple members with the same MST configuration. Each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can support up to 65 spanning-tree instances. Instances can be identified by any number in the range from 0 to 4094. You can assign a VLAN to only one spanning-tree instance at a time.

Related Topics

[Illustration of MST Regions](#) , on page 38

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

[Prerequisites for MSTP](#) , on page 31

[Restrictions for MSTP](#) , on page 32

[Spanning-Tree Interoperability and Backward Compatibility](#) , on page 12

[Optional Spanning-Tree Configuration Guidelines](#)

[BackboneFast](#) , on page 76

[UplinkFast](#) , on page 72

IST, CIST, and CST

Unlike PVST+ and Rapid PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 4094.

The IST is the only spanning-tree instance that sends and receives BPDUs. All of the other spanning-tree instance information is contained in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDU carries information for all instances, the number of BPDUs that need to be processed to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root switch ID, root path cost, and so forth. By default, all VLANs are assigned to the IST.

An MST instance is local to the region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

The spanning tree computed in a 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 switches that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D standards. The CIST inside an MST region is the same as the CST outside a region.

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the CIST regional root (called the *IST master* before the implementation of the IEEE 802.1s standard). It is the switch within the region with the lowest switch ID and path cost to the CIST 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, one of the MSTP switches at the boundary of the region is selected as the CIST regional root.

When an MSTP switch initializes, it sends BPDUs claiming itself as the root of the CIST and the CIST regional root, with both of the path costs to the CIST root and to the CIST regional root set to zero. The switch also initializes all of its MST instances and claims to be the root for all of them. If the switch receives superior MST root information (lower switch ID, lower path cost, and so forth) than currently stored for the port, it relinquishes its claim as the CIST regional root.

During initialization, a region might have many subregions, each with its own CIST regional root. As switches receive superior IST information, they leave their old subregions and join the new subregion that contains the true CIST regional root. All subregions shrink except for the one that contains the true CIST regional root.

For correct operation, all switches in the MST region must agree on the same CIST regional root. Therefore, any two switches in the region only synchronize their port roles for an MST instance if they converge to a common CIST regional root.

Related Topics

[Illustration of MST Regions, on page 38](#)

Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D switches within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP switches in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP switches 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 switch to adjacent STP switches and MST regions.

Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring switches and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, switch priority, port VLAN cost, and port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP switches use Version 3 RSTP BPDUs or IEEE 802.1D STP BPDUs to communicate with legacy IEEE 802.1D switches. MSTP switches use MSTP BPDUs to communicate with MSTP switches.

Related Topics

[Illustration of MST Regions, on page 38](#)

IEEE 802.1s Terminology

Some MST naming conventions used in Cisco's prestandard implementation have been changed to identify some *internal* or *regional* parameters. These parameters are significant only within an MST region, as opposed to external parameters that are relevant to the whole network. Because the CIST is the only spanning-tree instance that spans the whole network, only the CIST parameters require the external rather than the internal or regional qualifiers.

- The CIST root is the root switch for the unique instance that spans the whole network, the CIST.
- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. Remember that an MST region looks like a single switch for the CIST. The CIST external root path cost is the root path cost calculated between these virtual switches and switches that do not belong to any region.
- The CIST regional root was called the IST master in the prestandard implementation. If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest switch to the CIST root in the region. The CIST regional root acts as a root switch 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.

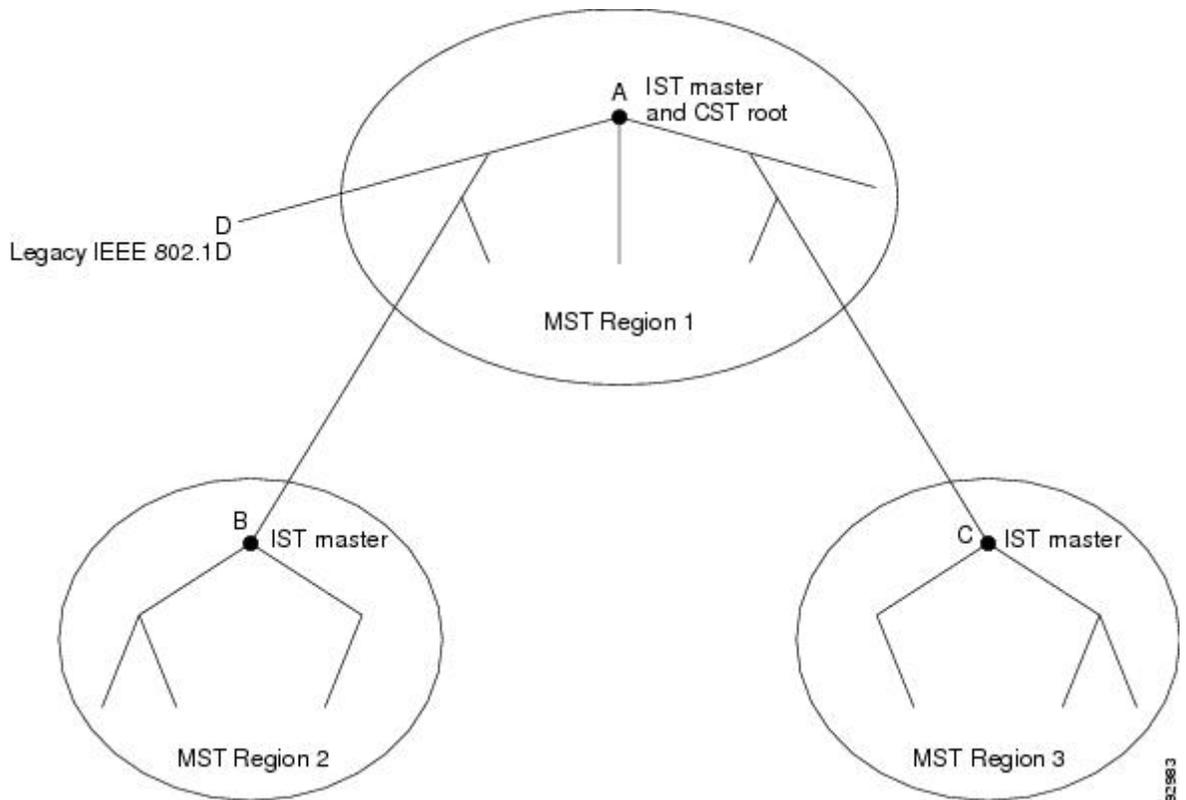
Table 6: Prestandard and Standard Terminology

| IEEE Standard | Cisco Prestandard | Cisco Standard |
|------------------------------|----------------------|-------------------------|
| CIST regional root | IST master | CIST regional root |
| CIST internal root path cost | IST master path cost | CIST internal path cost |
| CIST external root path cost | Root path cost | Root path cost |
| MSTI regional root | Instance root | Instance root |
| MSTI internal root path cost | Root path cost | Root path cost |

Illustration of MST Regions

This figure displays three MST regions and a legacy IEEE 802.1D switch (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. The RSTP runs in all regions.

Figure 5: MST Regions, CIST Masters, and CIST Root



Related Topics

- [Multiple Spanning-Tree Regions, on page 35](#)
- [Operations Within an MST Region, on page 36](#)
- [Operations Between MST Regions, on page 36](#)

Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use 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 switch 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 switch receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the switch discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region designated ports at the boundary.

Boundary Ports

In the Cisco prestandard implementation, a boundary port connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or rapid PVST+, or to another MST region with a different MST configuration. A boundary port also connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

There is no definition of a boundary port in the IEEE 802.1s standard. The IEEE 802.1Q-2002 standard identifies two kinds of messages that a port can receive:

- internal (coming from the same region)
- external (coming from another region)

When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record.

When a message is external, it is received only by the CIST. If the CIST role is root or alternate, or if the external BPDU is a topology change, it could have an impact on the MST instances.

An MST region includes both switches and LANs. A segment belongs to the region of its designated port. Therefore, a port in a different region than the designated port for a segment is a boundary port. This definition allows two ports internal to a region to share a segment with a port belonging to a different region, creating the possibility of a port receiving both internal and external messages.

The primary change from the Cisco prestandard implementation is that a designated port is not defined as boundary, unless it is running in an STP-compatible mode.

**Note**

If there is a legacy STP switch on the segment, messages are always considered external.

The other change from the Cisco prestandard implementation is that the CIST regional root switch ID field is now inserted where an RSTP or legacy IEEE 802.1Q switch has the sender switch ID. The whole region performs like a single virtual switch by sending a consistent sender switch ID to neighboring switches. In this example, switch C would receive a BPDU with the same consistent sender switch ID of root, whether or not A or B is designated for the segment.

IEEE 802.1s Implementation

The Cisco implementation of the IEEE MST standard includes features required to meet the standard, as well as some of the desirable prestandard functionality that is not yet incorporated into the published standard.

Port Role Naming Change

The boundary role is no longer in the final MST standard, but this boundary concept is maintained in Cisco's implementation. However, an MST instance port at a boundary of the region might not follow the state of the corresponding CIST port. Two boundary roles currently exist:

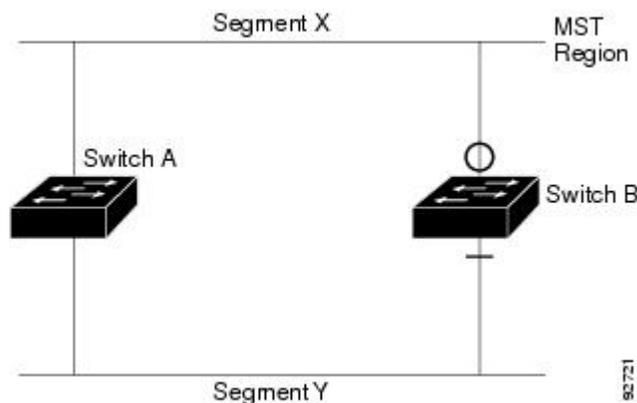
- The boundary port is the root port of the CIST regional root—When the CIST instance port is proposed and is in sync, it can send back an agreement and move to the forwarding state only after all the corresponding MSTI ports are in sync (and thus forwarding). The MSTI ports now have a special *master* role.
- The boundary port is not the root port of the CIST regional root—The MSTI ports follow the state and role of the CIST port. The standard provides less information, and it might be difficult to understand why an MSTI port can be alternately blocking when it receives no BPDUs (MRecords). In this case, although the boundary role no longer exists, the **show** commands identify a port as boundary in the *type* column of the output.

Interoperation Between Legacy and Standard Switches

Because automatic detection of prestandard switches can fail, you can use an interface configuration command to identify prestandard ports. A region cannot be formed between a standard and a prestandard switch, but they can interoperate by using the CIST. Only the capability of load-balancing over different instances is lost in that particular case. The CLI displays different flags depending on the port configuration when a port receives prestandard BPDUs. A syslog message also appears the first time a switch receives a prestandard BPDU on a port that has not been configured for prestandard BPDU transmission.

Assume that A is a standard switch and B a prestandard switch, both configured to be in the same region. A is the root switch for the CIST, and B has a root port (BX) on segment X and an alternate port (BY) on segment Y. If segment Y flaps, and the port on BY becomes the alternate before sending out a single prestandard BPDU, AY cannot detect that a prestandard switch is connected to Y and continues to send standard BPDUs. The port BY is fixed in a boundary, and no load balancing is possible between A and B. The same problem exists on segment X, but B might transmit topology changes.

Figure 6: Standard and Prestandard Switch Interoperation



**Note**

We recommend that you minimize the interaction between standard and prestandard MST implementations.

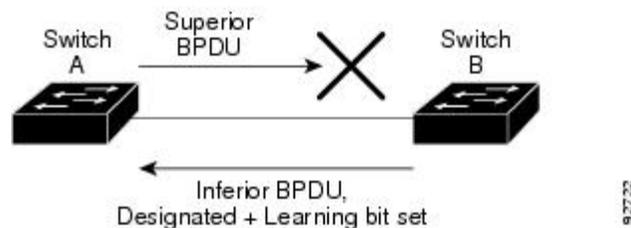
Detecting Unidirectional Link Failure

This feature is not yet present in the IEEE MST standard, but it is included in this Cisco IOS release. 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.

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

This figure illustrates a unidirectional link failure that typically creates a bridging loop. Switch A is the root switch, and its BPDUs are lost on the link leading to switch B. RSTP 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 it sends and that switch B is the designated, not root switch. As a result, switch A blocks (or keeps blocking) its port, which prevents the bridging loop.

Figure 7: Detecting Unidirectional Link Failure



MSTP and Switch Stacks

A switch stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID for a given spanning tree. The bridge ID is derived from the MAC address of the active switch.

The active switch is the stack root when the stack is the root of the network and no root selection has been made within the stack.

If the switch stack is the spanning-tree root and the active switch fails or leaves the stack, the standby switch becomes the new active switch, bridge IDs remain the same, and a spanning-tree reconvergence might occur.

If a switch that does not support MSTP is added to a switch stack that does support MSTP or the reverse, the switch is put into a version mismatch state. If possible, the switch is automatically upgraded or downgraded to the same version of software that is running on the switch stack.

Interoperability with IEEE 802.1D STP

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch also can

detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (Version 3) associated with a different region, or an RSTP BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch might also continue to assign a boundary role to a port when the switch to which this switch is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring switches), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy switches on the link are RSTP switches, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP switches send either a Version 0 configuration and TCN BPDUs or Version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

RSTP Overview

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree).

Port Roles and the Active Topology

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

- Root port—Provides the best path (lowest cost) when the switch forwards packets to the root switch.
- Designated port—Connects to the designated switch, which incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root switch to that provided by the current root port.
- 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 switch has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

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.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in IEEE 802.1D). The port state controls the operation of the forwarding and learning processes.

Table 7: Port State Comparison

| Operational Status | STP Port State (IEEE 802.1D) | RSTP Port State | Is Port Included in the Active Topology? |
|--------------------|------------------------------|-----------------|--|
| Enabled | Blocking | Discarding | No |
| Enabled | Listening | Discarding | No |
| Enabled | Learning | Learning | Yes |
| Enabled | Forwarding | Forwarding | Yes |
| Disabled | Disabled | Discarding | No |

To be consistent with Cisco STP implementations, this guide defines the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a switch, a switch 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**—If you configure a port as an edge port on an RSTP switch by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state. An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.
- **Root ports**—If the RSTP 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.

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 Switch B's agreement message, Switch A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Switch B blocked all of its nonedge ports and because there is a point-to-point link between Switches A and B.

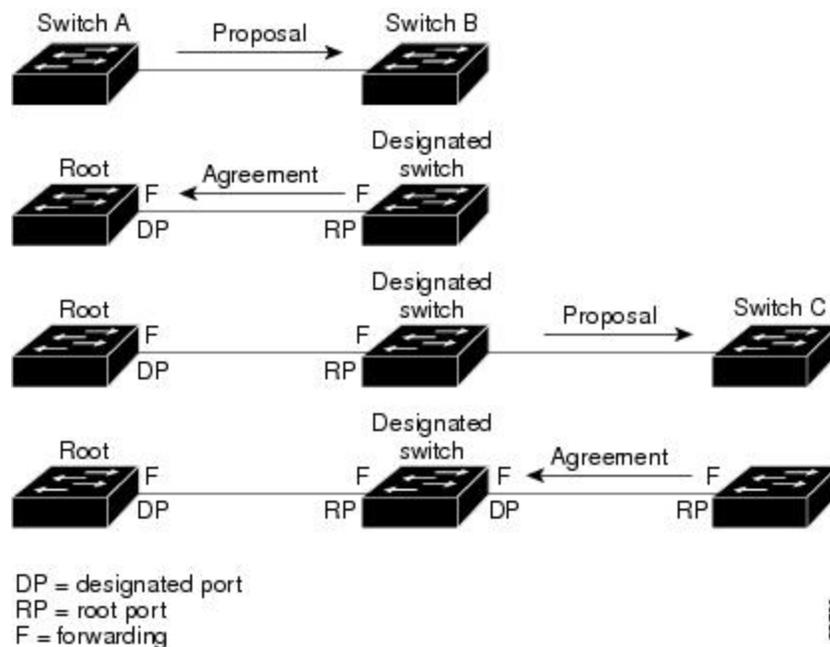
When Switch C is connected to Switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to Switch B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active

topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

In a switch stack, the cross-stack rapid transition (CSRT) feature ensures that a stack member receives acknowledgments from all stack members during the proposal-agreement handshaking before moving the port to the forwarding state. CSRT is automatically enabled when the switch is in MST mode.

The switch learns the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; 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 using the **spanning-tree link-type** interface configuration command.

Figure 8: Proposal and Agreement Handshaking for Rapid Convergence



Synchronization of Port Roles

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

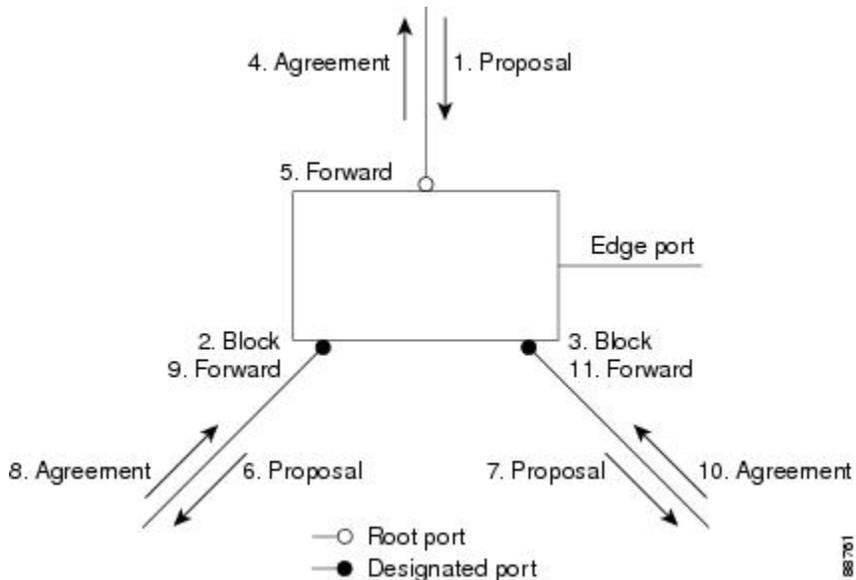
The switch is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the switch is synchronized if

- 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 RSTP forces it to synchronize with new root information. In general, when the RSTP 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 switch sends an agreement message to the designated switch corresponding to its root port. When the switches connected by a point-to-point link are in agreement about their port roles, the RSTP immediately transitions the port states to forwarding.

Figure 9: Sequence of Events During Rapid Convergence



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new 1-byte Version 1 Length field is set to zero, which means that no version 1 protocol information is present.

Table 8: RSTP BPDU Flags

| Bit | Function |
|------|----------------------|
| 0 | Topology change (TC) |
| 1 | Proposal |
| 2–3: | Port role: |
| 00 | Unknown |
| 01 | Alternate port |
| 10 | Root port |
| 11 | Designated port |
| 4 | Learning |
| 5 | Forwarding |

| Bit | Function |
|-----|---------------------------------------|
| 6 | Agreement |
| 7 | Topology change acknowledgement (TCA) |

The sending switch sets the proposal flag in the RSTP BPDU to propose itself as the designated switch on that LAN. The port role in the proposal message is always set to the designated port.

The sending switch sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with IEEE 802.1D switches, the RSTP switch processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower switch ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the switch sends an agreement message after all of the other ports are synchronized. If the BPDU is an IEEE 802.1D BPDU, the switch 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.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (such as a higher switch ID or a higher path cost than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- **Detection**—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP switch detects a topology change, it deletes the learned information on all of its nonedge ports except on those from which it received the TC notification.

- **Notification**—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 802.1D interoperability, an RSTP switch processes and generates TCN BPDUs.
- **Acknowledgement**—When an RSTP switch receives a TCN message on a designated port from an IEEE 802.1D switch, it replies with an IEEE 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D switch and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

This behavior is only required to support IEEE 802.1D switches. The RSTP BPDUs never have the TCA bit set.

- **Propagation**—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding the port on which it is received). The switch starts the TC-while timer for all such ports and flushes the information learned on them.
- **Protocol migration**—For backward compatibility with IEEE 802.1D switches, RSTP selectively sends IEEE 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 RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the switch processes all BPDUs received on that port and ignores the protocol type.

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

Protocol Migration Process

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (Version 3) associated with a different region, or an RST BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch also might continue to assign a boundary role to a port when the switch to which it is connected has joined the region.

Related Topics

[Restarting the Protocol Migration Process](#), on page 65

Default MSTP Configuration

Table 9: Default MSTP Configuration

| Feature | Default Setting |
|--------------------|-----------------|
| Spanning-tree mode | MSTP |

| Feature | Default Setting |
|---|--|
| Switch priority (configurable on a per-CIST port basis) | 32768 |
| Spanning-tree port priority (configurable on a per-CIST port basis) | 128 |
| Spanning-tree port cost (configurable on a per-CIST port basis) | 1000 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000 1000 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000 |
| Hello time | 3 seconds |
| Forward-delay time | 20 seconds |
| Maximum-aging time | 20 seconds |
| Maximum hop count | 20 hops |

Related Topics

[Supported Spanning-Tree Instances, on page 11](#)

[Specifying the MST Region Configuration and Enabling MSTP , on page 48](#)

How to Configure MSTP Features

Specifying the MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst configuration**
4. **instance** *instance-id* **vlan** *vlan-range*
5. **name** *name*
6. **revision** *version*
7. **show pending**
8. **exit**
9. **spanning-tree mode mst**
10. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst configuration Example: Switch(config)# spanning-tree mst configuration | Enters MST configuration mode. |
| Step 4 | instance <i>instance-id</i> vlan <i>vlan-range</i> Example: Switch(config-mst)# instance 1 vlan 10-20 | Maps VLANs to an MST instance. <ul style="list-style-type: none"> • For <i>instance-id</i>, the range is 0 to 4094. • For vlan <i>vlan-range</i>, the range is 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. <p>To specify a VLAN range, use a hyphen; for example, instance 1 vlan 1-63 maps VLANs 1 through 63 to MST instance 1.</p> <p>To specify a VLAN series, use a comma; for example, instance 1 vlan 10, 20, 30 maps VLANs 10, 20, and 30 to MST instance 1.</p> |

| | Command or Action | Purpose |
|----------------|---|--|
| Step 5 | name <i>name</i> Example: Switch(config-mst)# name region1 | Specifies the configuration name. The <i>name</i> string has a maximum length of 32 characters and is case sensitive. |
| Step 6 | revision <i>version</i> Example: Switch(config-mst)# revision 1 | Specifies the configuration revision number. The range is 0 to 65535. |
| Step 7 | show pending Example: Switch(config-mst)# show pending | Verifies your configuration by displaying the pending configuration. |
| Step 8 | exit Example: Switch(config-mst)# exit | Applies all changes, and returns to global configuration mode. |
| Step 9 | spanning-tree mode mst Example: Switch(config)# spanning-tree mode mst | <p>Enables MSTP. RSTP is also enabled.</p> <p>Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode.</p> <p>You cannot run both MSTP and PVST+ or both MSTP and Rapid PVST+ at the same time.</p> |
| Step 10 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[MSTP Configuration Guidelines, on page 33](#)

[Multiple Spanning-Tree Regions, on page 35](#)

[Prerequisites for MSTP, on page 31](#)

[Restrictions for MSTP, on page 32](#)

[Spanning-Tree Interoperability and Backward Compatibility, on page 12](#)

[Optional Spanning-Tree Configuration Guidelines](#)

[BackboneFast, on page 76](#)

[UplinkFast, on page 72](#)

- [Default MSTP Configuration, on page 47](#)
- [Configuring the Root Switch, on page 51](#)
- [Restrictions for MSTP, on page 32](#)
- [Bridge ID, Device Priority, and Extended System ID, on page 5](#)
- [Configuring a Secondary Root Switch, on page 52](#)
- [Configuring Port Priority, on page 53](#)
- [Configuring Path Cost, on page 55](#)
- [Configuring the Switch Priority, on page 57](#)
- [Configuring the Hello Time, on page 58](#)
- [Configuring the Forwarding-Delay Time, on page 59](#)
- [Configuring the Maximum-Aging Time, on page 60](#)
- [Configuring the Maximum-Hop Count, on page 61](#)
- [Specifying the Link Type to Ensure Rapid Transitions, on page 62](#)
- [Designating the Neighbor Type, on page 64](#)
- [Restarting the Protocol Migration Process, on page 65](#)

Configuring the Root Switch

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID. Step 2 in the example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* root primary**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst <i>instance-id</i> root primary Example: Switch(config)# spanning-tree mst 0 root primary | Configures a switch as the root switch. <ul style="list-style-type: none"> 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 0 to 4094. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Root Switch](#), on page 34

[Specifying the MST Region Configuration and Enabling MSTP](#), on page 48

[Restrictions for MSTP](#), on page 32

[Bridge ID, Device Priority, and Extended System ID](#), on page 5

[Configuring a Secondary Root Switch](#), on page 52

Configuring a Secondary Root Switch

When you configure a switch with the extended system ID support as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified instance if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree mst *instance-id* root primary** global configuration command.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* root secondary**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst <i>instance-id</i> root secondary Example: Switch(config)# spanning-tree mst 0 root secondary | Configures a switch as the secondary root switch. <ul style="list-style-type: none"> • 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 0 to 4094. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

- [Specifying the MST Region Configuration and Enabling MSTP , on page 48](#)
- [Configuring the Root Switch , on page 51](#)

Configuring Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same

priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.



Note If the switch is a member of a switch stack, you must use the **spanning-tree mst [instance-id] cost cost** interface configuration command instead of the **spanning-tree mst [instance-id] port-priority priority** interface configuration command to select a port to put in the forwarding state. Assign lower cost values to ports that you want selected first and higher cost values to ports that you want selected last. For more information, see the path costs topic listed under Related Topics.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **spanning-tree mst instance-id port-priority priority**
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface interface-id Example: Switch(config)# interface GigabitEthernet1/0/1 | Specifies an interface to configure, and enters interface configuration mode. |

| | Command or Action | Purpose |
|--------|---|---|
| Step 4 | <p>spanning-tree mst <i>instance-id</i> port-priority <i>priority</i></p> <p>Example:</p> <pre>Switch(config-if) # spanning-tree mst 0 port-priority 64</pre> | <p>Configures port priority.</p> <ul style="list-style-type: none"> 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 0 to 4094. For <i>priority</i>, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. The priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. |
| Step 5 | <p>end</p> <p>Example:</p> <pre>Switch(config-if) # end</pre> | <p>Returns to privileged EXEC mode.</p> |

The **show spanning-tree mst interface *interface-id*** privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48
[Configuring Path Cost](#) , on page 55

Configuring Path Cost

The MSTP path cost default value is derived from the media speed of an interface. If a loop occurs, the MSTP uses 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 values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **cost** *cost*
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1 | Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces. The port-channel range is 1 to 48. |
| Step 4 | spanning-tree mst <i>instance-id</i> cost <i>cost</i> Example: Switch(config-if)# spanning-tree mst 0 cost 17031970 | Configures the cost. If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. <ul style="list-style-type: none"> • 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 0 to 4094. • For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface. |
| Step 5 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Related Topics

[Configuring Port Priority](#) , on page 53

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Configuring the Switch Priority

Changing the priority of a switch makes it more likely to be chosen as the root switch whether it is a standalone switch or a switch in the stack.



Note

Exercise care when using this command. For normal network configurations, we recommend that you use the **spanning-tree mst instance-id root primary** and the **spanning-tree mst instance-id root secondary** global configuration commands to specify a switch as the root or secondary root switch. You should modify the switch priority only in circumstances where these commands do not work.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID used. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst instance-id priority priority**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst <i>instance-id</i> priority <i>priority</i> Example: Switch(config)# spanning-tree mst 0 priority 40960 | Configures the switch priority. <ul style="list-style-type: none"> 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 0 to 4094. For <i>priority</i>, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. These are the only acceptable values. |
| Step 4 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root switch.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst hello-time *seconds***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst hello-time <i>seconds</i> Example: Switch(config)# spanning-tree mst hello-time 4 | Configures the hello time for all MST instances. The hello time is the time interval between configuration messages generated and sent by the root switch. These messages indicate that the switch is alive. For <i>seconds</i> , the range is 1 to 10; the default is 3. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Configuring the Forwarding-Delay Time

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst forward-time *seconds***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst forward-time seconds Example: Switch(config)# spanning-tree mst forward-time 25 | Configures the forward time for all MST instances. The forwarding delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For <i>seconds</i> , the range is 4 to 30; the default is 20. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Configuring the Maximum-Aging Time

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-age seconds**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst max-age <i>seconds</i> 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 a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For <i>seconds</i> , the range is 6 to 40; the default is 20. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Configuring the Maximum-Hop Count

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-hops *hop-count***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree mst max-hops hop-count Example: Switch(config)# spanning-tree mst max-hops 25 | Specifies the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For <i>hop-count</i> , the range is 1 to 255; the default is 20. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Specifying the Link Type to Ensure Rapid Transitions

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

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 switch running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree link-type point-to-point**
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface GigabitEthernet1/0/1 | Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port-channel logical interfaces. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. |
| Step 4 | spanning-tree link-type point-to-point Example: Switch(config-if)# spanning-tree link-type point-to-point | Specifies that the link type of a port is point-to-point. |
| Step 5 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Designating the Neighbor Type

A topology could contain both prestandard and IEEE 802.1s standard compliant devices. By default, ports can automatically detect prestandard devices, but they can still receive both standard and prestandard BPDUs. When there is a mismatch between a device and its neighbor, only the CIST runs on the interface.

You can choose to set a port to send only prestandard BPDUs. The prestandard flag appears in all the **show** commands, even if the port is in STP compatibility mode.

This procedure is optional.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst pre-standard**
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface GigabitEthernet1/0/1 | Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports. |

| | Command or Action | Purpose |
|--------|---|--|
| Step 4 | spanning-tree mst pre-standard Example: Switch(config-if) # spanning-tree mst pre-standard | Specifies that the port can send only prestandard BPDUs. |
| Step 5 | end Example: Switch(config-if) # end | Returns to privileged EXEC mode. |

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48

Restarting the Protocol Migration Process

This procedure restarts the protocol migration process and forces renegotiation with neighboring switches. It reverts the switch to MST mode. It is needed when the switch no longer receives IEEE 802.1D BPDUs after it has been receiving them.

Follow these steps to restart the protocol migration process (force the renegotiation with neighboring switches) on the switch.

Before You Begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

If you want to use the interface version of the command, you must also know the MST interface used. This example uses GigabitEthernet1/0/1 as the interface because that was the interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. Enter one of the following commands:
 - **clear spanning-tree detected-protocols**
 - **clear spanning-tree detected-protocols interface *interface-id***

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|---|
| Step 1 | <p>enable</p> <p>Example:</p> <pre>Switch> enable</pre> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | <p>Enter one of the following commands:</p> <ul style="list-style-type: none"> • clear spanning-tree detected-protocols • clear spanning-tree detected-protocols interface <i>interface-id</i> <p>Example:</p> <pre>Switch# clear spanning-tree detected-protocols OR Switch# clear spanning-tree detected-protocols interface GigabitEthernet1/0/1</pre> | The switch reverts to the MSTP mode, and the protocol migration process restarts. |

What to Do Next

This procedure may need to be repeated if the switch receives more legacy IEEE 802.1D configuration BPDUs (BPDUs with the protocol version set to 0).

Related Topics

[Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48
[Protocol Migration Process](#), on page 47

Monitoring MST Configuration and Status

Table 10: Commands for Displaying MST Status

| | |
|--|---|
| show spanning-tree mst configuration | Displays the MST region configuration. |
| show spanning-tree mst configuration digest | Displays the MD5 digest included in the current MSTCI. |
| show spanning-tree mst | <p>Displays MST information for the all instances.</p> <p>Note show spanning-tree mst</p> <p>This command displays information for ports in a link-up operative state.</p> |

| | |
|---|---|
| show spanning-tree mst <i>instance-id</i> | Displays MST information for the specified instance. Note This command displays information only if the port is in a link-up operative state. |
| show spanning-tree mst interface <i>interface-id</i> | Displays MST information for the specified interface. |

Additional References for MSTP

Related Documents

| Related Topic | Document Title |
|---------------------------------|---|
| Spanning tree protocol commands | <i>LAN Switching Command Reference, Cisco IOS XE Release 3SE (Catalyst 3650 Switches)</i> |

Error Message Decoder

| Description | Link |
|---|---|
| To help you research and resolve system error messages in this release, use the Error Message Decoder tool. | https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi |

Standards and RFCs

| Standard/RFC | Title |
|--------------|-------|
| None | — |

MIBs

| MIB | MIBs Link |
|--------------------------------------|--|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|--|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p>http://www.cisco.com/support</p> |

Feature Information for MSTP

| Release | Modification |
|---------------------------------------|------------------------------|
| Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE | This feature was introduced. |



Configuring Optional Spanning-Tree Features

- [Finding Feature Information, page 69](#)
- [Restriction for Optional Spanning-Tree Features, page 69](#)
- [Information About Optional Spanning-Tree Features, page 70](#)
- [How to Configure Optional Spanning-Tree Features, page 80](#)
- [Monitoring the Spanning-Tree Status, page 93](#)
- [Additional References for Optional Spanning Tree Features, page 93](#)
- [Feature Information for Optional Spanning-Tree Features, page 94](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restriction for Optional Spanning-Tree Features

- PortFast minimizes the time that interfaces must wait for spanning tree to converge, so it is effective only when used on interfaces connected to end stations. If you enable PortFast on an interface connecting to another switch, you risk creating a spanning-tree loop.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Related Topics

[Enabling PortFast](#) , on page 80

[PortFast, on page 70](#)

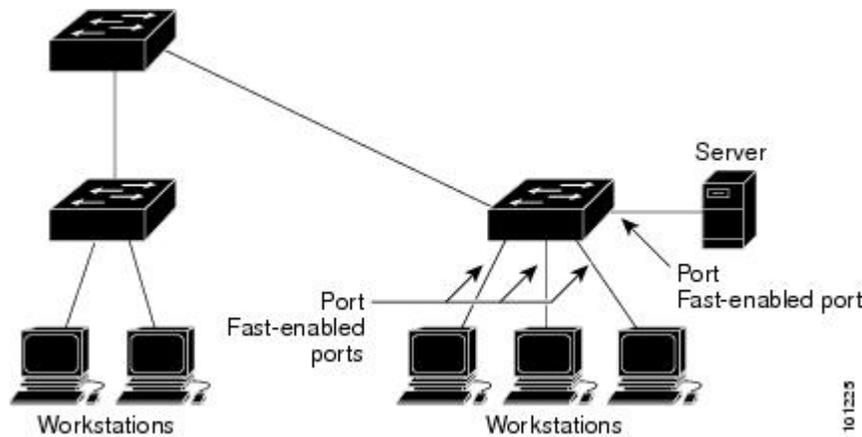
Information About Optional Spanning-Tree Features

PortFast

PortFast immediately brings an interface configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states.

You can use PortFast on interfaces connected to a single workstation or server to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

Figure 10: PortFast-Enabled Interfaces



Interfaces connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An interface with PortFast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

You can enable this feature by enabling it on either the interface or on all nontrunking ports.

Related Topics

[Enabling PortFast, on page 80](#)

[Restriction for Optional Spanning-Tree Features, on page 69](#)

BPDU Guard

The Bridge Protocol Data Unit (BPDU) guard feature can be globally enabled on the switch or can be enabled per port, but the feature operates with some differences.

When you enable BPDU guard at the global level on PortFast-enabled ports, spanning tree shuts down ports that are in a PortFast-operational state if any BPDU is received on them. In a valid configuration, PortFast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port

in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

When you enable BPDU guard at the interface level on any port without also enabling the PortFast feature, and the port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

Related Topics

[Enabling BPDU Guard](#) , on page 82

BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

Enabling BPDU filtering on PortFast-enabled interfaces at the global level keeps those interfaces that are in a PortFast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received on a PortFast-enabled interface, the interface loses its PortFast-operational status, and BPDU filtering is disabled.

Enabling BPDU filtering on an interface without also enabling the PortFast feature keeps the interface from sending or receiving BPDUs.



Caution

Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an interface.

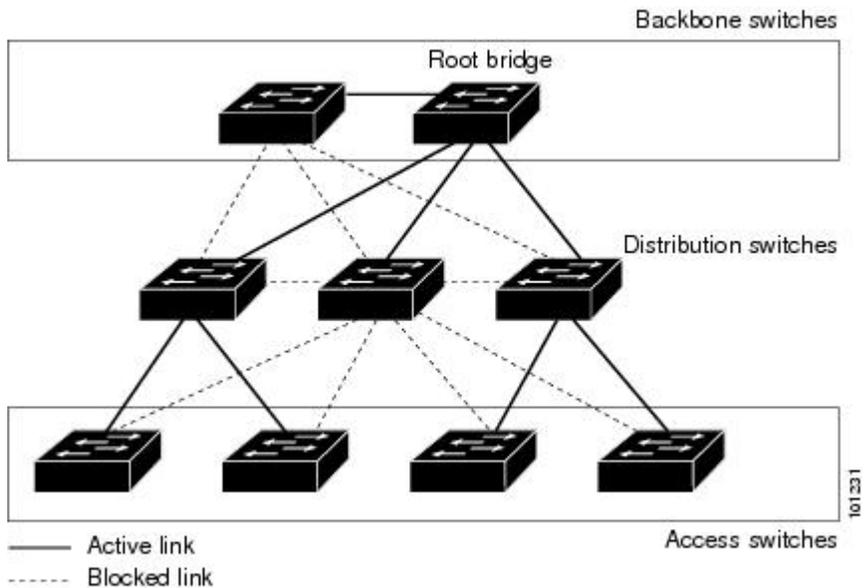
Related Topics

[Enabling BPDU Filtering](#) , on page 83

UplinkFast

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. This complex network has distribution switches and access switches that each have at least one redundant link that spanning tree blocks to prevent loops.

Figure 11: Switches in a Hierarchical Network



If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. You can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself by enabling UplinkFast. The root port transitions to the forwarding state immediately without going through the listening and learning states, as it would with the normal spanning-tree procedures.

When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets per second). However, if you enter zero, station-learning frames are not generated, so the spanning-tree topology converges more slowly after a loss of connectivity.



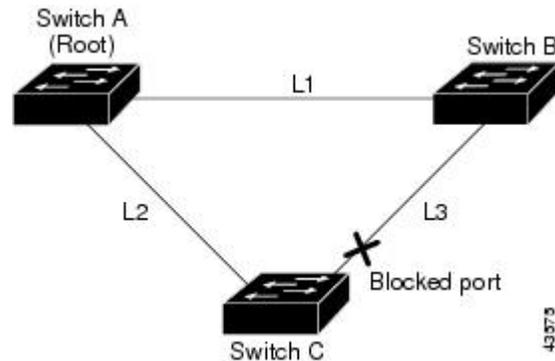
Note

UplinkFast is most useful in wiring-closet switches at the access or edge of the network. It is not appropriate for backbone devices. This feature might not be useful for other types of applications.

UplinkFast provides fast convergence after a direct link failure and achieves load-balancing between redundant Layer 2 links using uplink groups. An uplink group is a set of Layer 2 interfaces (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

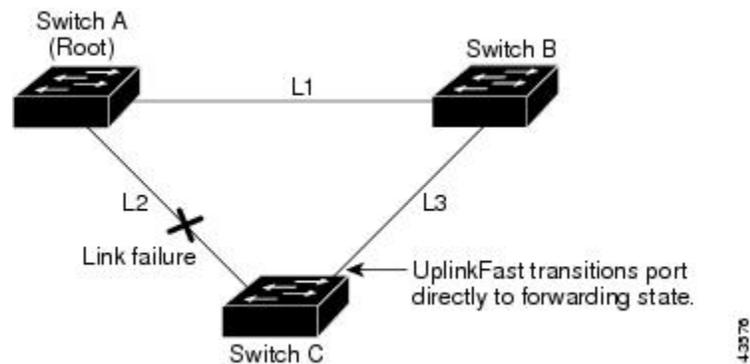
This topology has no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in a blocking state.

Figure 12: UplinkFast Example Before Direct Link Failure



If Switch C detects a link failure on the currently active link L2 on the root port (a direct link failure), UplinkFast unblocks the blocked interface on Switch C and transitions it to the forwarding state without going through the listening and learning states. This change takes approximately 1 to 5 seconds.

Figure 13: UplinkFast Example After Direct Link Failure



Related Topics

- [Specifying the MST Region Configuration and Enabling MSTP](#) , on page 48
- [MSTP Configuration Guidelines](#), on page 33
- [Multiple Spanning-Tree Regions](#), on page 35
- [Enabling UplinkFast for Use with Redundant Links](#) , on page 85
- [Events That Cause Fast Convergence](#), on page 76

Cross-Stack UplinkFast

Cross-Stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a switch stack. During the fast transition, an alternate redundant link on the switch stack is placed in the forwarding state without causing temporary spanning-tree loops or loss

of connectivity to the backbone. With this feature, you can have a redundant and resilient network in some configurations. CSUF is automatically enabled when you enable the UplinkFast feature.

CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see [Related Topics](#).

Related Topics

[Enabling UplinkFast for Use with Redundant Links](#) , on page 85

[Events That Cause Fast Convergence](#), on page 76

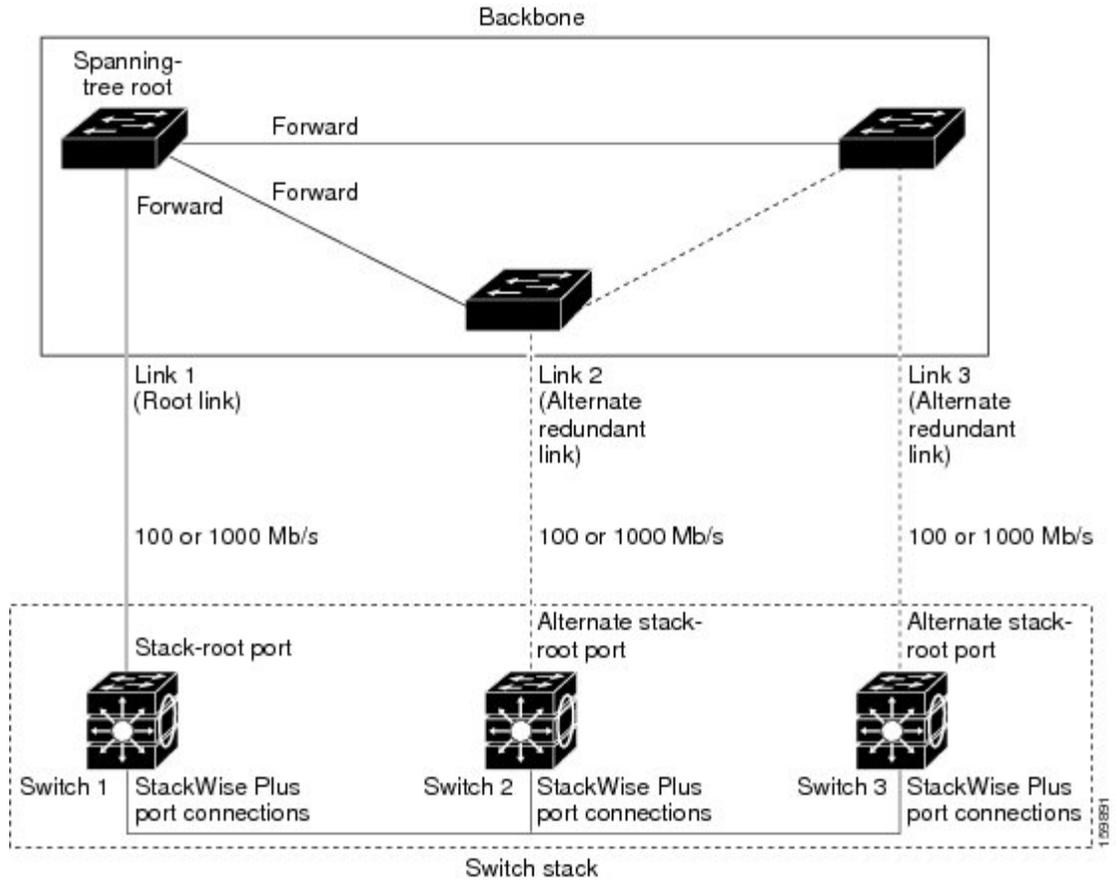
How Cross-Stack UplinkFast Works

Cross-Stack UplinkFast (CSUF) ensures that one link in the stack is elected as the path to the root.

The stack-root port on Switch 1 provides the path to the root of the spanning tree. The alternate stack-root ports on Switches 2 and 3 can provide an alternate path to the spanning-tree root if the current stack-root switch fails or if its link to the spanning-tree root fails.

Link 1, the root link, is in the spanning-tree forwarding state. Links 2 and 3 are alternate redundant links that are in the spanning-tree blocking state. If Switch 1 fails, if its stack-root port fails, or if Link 1 fails, CSUF selects either the alternate stack-root port on Switch 2 or Switch 3 and puts it into the forwarding state in less than 1 second.

Figure 14: Cross-Stack UplinkFast Topology



When certain link loss or spanning-tree events occur (described in the following topic), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests to stack members.

The switch sending the fast-transition request needs to do a fast transition to the forwarding state of a port that it has chosen as the root port, and it must obtain an acknowledgment from each stack switch before performing the fast transition.

Each switch in the stack decides if the sending switch is a better choice than itself to be the stack root of this spanning-tree instance by comparing the root, cost, and bridge ID. If the sending switch is the best choice as the stack root, each switch in the stack returns an acknowledgment; otherwise, it sends a fast-transition request. The sending switch then has not received acknowledgments from all stack switches.

When acknowledgments are received from all stack switches, the Fast Uplink Transition Protocol on the sending switch immediately transitions its alternate stack-root port to the forwarding state. If acknowledgments from all stack switches are not obtained by the sending switch, the normal spanning-tree transitions (blocking, listening, learning, and forwarding) take place, and the spanning-tree topology converges at its normal rate ($2 * \text{forward-delay time} + \text{max-age time}$).

The Fast Uplink Transition Protocol is implemented on a per-VLAN basis and affects only one spanning-tree instance at a time.

Related Topics

[Enabling UplinkFast for Use with Redundant Links](#) , on page 85

[Events That Cause Fast Convergence](#), on page 76

Events That Cause Fast Convergence

Depending on the network event or failure, the CSUF fast convergence might or might not occur.

Fast convergence (less than 1 second under normal network conditions) occurs under these circumstances:

- The stack-root port link fails.
If two switches in the stack have alternate paths to the root, only one of the switches performs the fast transition.
- The failed link, which connects the stack root to the spanning-tree root, recovers.
- A network reconfiguration causes a new stack-root switch to be selected.
- A network reconfiguration causes a new port on the current stack-root switch to be chosen as the stack-root port.



Note The fast transition might not occur if multiple events occur simultaneously. For example, if a stack member is powered off, and at the same time, the link connecting the stack root to the spanning-tree root comes back up, the normal spanning-tree convergence occurs.

Normal spanning-tree convergence (30 to 40 seconds) occurs under these conditions:

- The stack-root switch is powered off, or the software failed.
- The stack-root switch, which was powered off or failed, is powered on.
- A new switch, which might become the stack root, is added to the stack.

Related Topics

[Enabling UplinkFast for Use with Redundant Links](#), on page 85

[UplinkFast](#), on page 72

[Cross-Stack UplinkFast](#), on page 73

[How Cross-Stack UplinkFast Works](#), on page 74

BackboneFast

BackboneFast detects indirect failures in the core of the backbone. BackboneFast is a complementary technology to the UplinkFast feature, which responds to failures on links directly connected to access switches. BackboneFast optimizes the maximum-age timer, which controls the amount of time the switch stores protocol information received on an interface. When a switch receives an inferior BPDU from the designated port of another switch, the BPDU is a signal that the other switch might have lost its path to the root, and BackboneFast tries to find an alternate path to the root.

BackboneFast starts when a root port or blocked interface on a switch receives inferior BPDUs from its designated switch. An inferior BPDU identifies a switch that declares itself as both the root bridge and the designated switch. When a switch receives an inferior BPDU, it means that a link to which the switch is not

directly connected (an indirect link) has failed (that is, the designated switch has lost its connection to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the maximum aging time (default is 20 seconds).

The switch tries to find if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked interface, the root port and other blocked interfaces on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked interfaces become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked interfaces, the switch assumes that it has lost connectivity to the root switch, causes the maximum aging time on the root port to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root switch, it uses these alternate paths to send a root link query (RLQ) request. The switch sends the RLQ request on all alternate paths to learn if any stack member has an alternate root to the root switch and waits for an RLQ reply from other switches in the network and in the stack. The switch sends the RLQ request on all alternate paths and waits for an RLQ reply from other switches in the network.

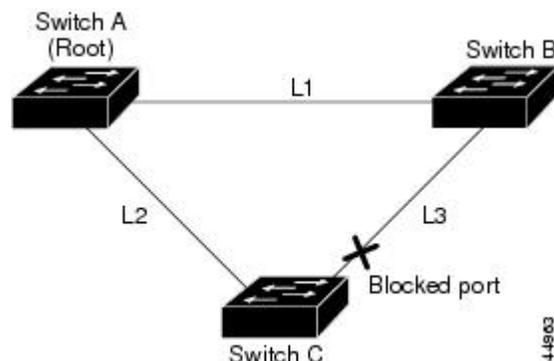
When a stack member receives an RLQ reply from a nonstack member on a blocked interface and the reply is destined for another nonstacked switch, it forwards the reply packet, regardless of the spanning-tree interface state.

When a stack member receives an RLQ reply from a nonstack member and the response is destined for the stack, the stack member forwards the reply so that all the other stack members receive it.

If the switch discovers that it still has an alternate path to the root, it expires the maximum aging time on the interface that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the interface that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all interfaces on which it received an inferior BPDU its designated ports and moves them from the blocking state (if they were in the blocking state), through the listening and learning states, and into the forwarding state.

This is an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B is in the blocking state.

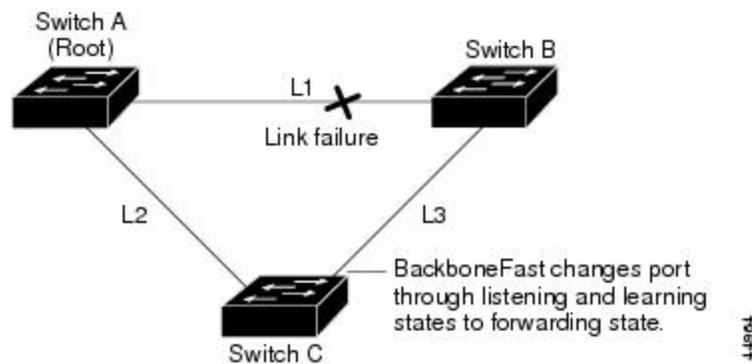
Figure 15: BackboneFast Example Before Indirect Link Failure



If link L1 fails, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying itself as the root. When Switch C receives the inferior

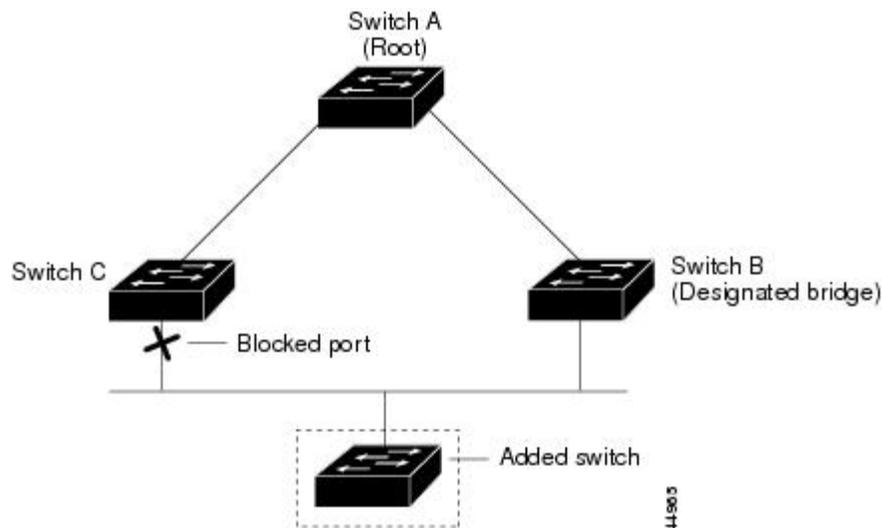
BPDUs from Switch B, Switch C assumes that an indirect failure has occurred. At that point, BackboneFast allows the blocked interface on Switch C to move immediately to the listening state without waiting for the maximum aging time for the interface to expire. BackboneFast then transitions the Layer 2 interface on Switch C to the forwarding state, providing a path from Switch B to Switch A. The root-switch election takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 16: BackboneFast Example After Indirect Link Failure



If a new switch is introduced into a shared-medium topology, BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated switch (Switch B). The new switch begins sending inferior BPDUs that indicate it is the root switch. However, the other switches ignore these inferior BPDUs, and the new switch learns that Switch B is the designated switch to Switch A, the root switch.

Figure 17: Adding a Switch in a Shared-Medium Topology



Related Topics

- [Specifying the MST Region Configuration and Enabling MSTP](#), on page 48
- [MSTP Configuration Guidelines](#), on page 33
- [Multiple Spanning-Tree Regions](#), on page 35
- [Enabling BackboneFast](#), on page 87

EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch interfaces are configured in an EtherChannel, but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and displays an error message.

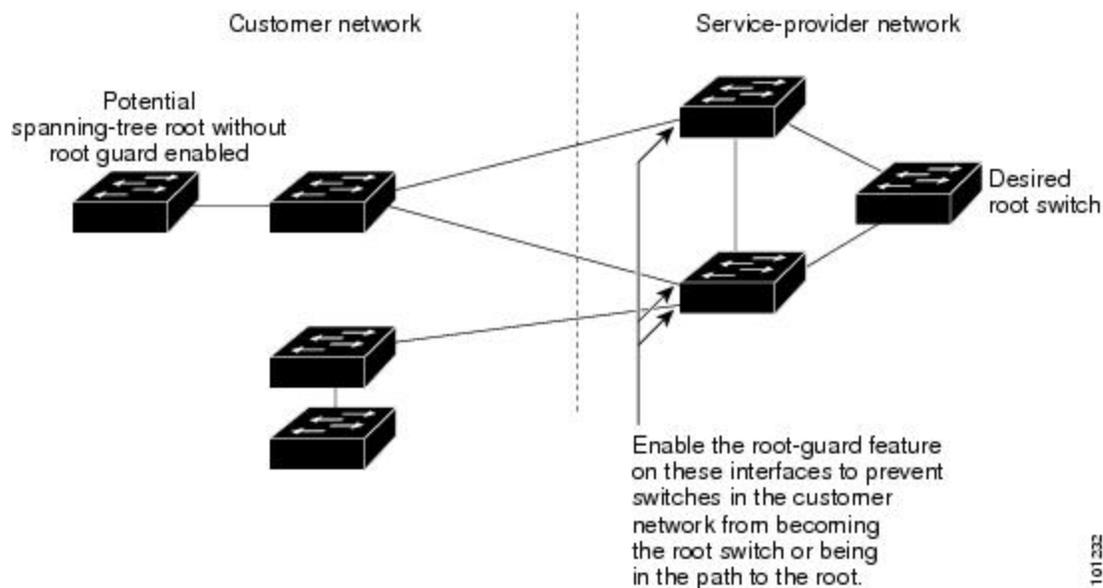
Related Topics

[Enabling EtherChannel Guard](#), on page 89

Root Guard

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a customer switch as the root switch. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.

Figure 18: Root Guard in a Service-Provider Network



If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root

guard, the interface also is blocked in all MST instances. A boundary port is an interface that connects to a LAN, the designated switch of which is either an IEEE 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.

**Caution**

Misuse of the root guard feature can cause a loss of connectivity.

Related Topics

[Enabling Root Guard](#) , on page 90

Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

Related Topics

[Enabling Loop Guard](#) , on page 91

How to Configure Optional Spanning-Tree Features

Enabling PortFast

An interface with the PortFast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

If you enable the voice VLAN feature, the PortFast feature is automatically enabled. When you disable voice VLAN, the PortFast feature is not automatically disabled.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

**Caution**

Use PortFast only when connecting a single end station to an access or trunk port. Enabling this feature on an interface connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast** [**trunk**]
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/2 | Specifies an interface to configure, and enters interface configuration mode. |
| Step 4 | spanning-tree portfast [trunk] Example: Switch(config-if)# spanning-tree portfast trunk | <p>Enables PortFast on an access port connected to a single workstation or server. By specifying the trunk keyword, you can enable PortFast on a trunk port.</p> <p>Note To enable PortFast on trunk ports, you must use the spanning-tree portfast trunk interface configuration command. The spanning-tree portfast command will not work on trunk ports.</p> <p>Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable PortFast on a trunk port.</p> <p>By default, PortFast is disabled on all interfaces.</p> |
| Step 5 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

What to Do Next

You can use the **spanning-tree portfast default** global configuration command to globally enable the PortFast feature on all nontrunking ports.

Related Topics

[PortFast, on page 70](#)

[Restriction for Optional Spanning-Tree Features, on page 69](#)

Enabling BPDU Guard

You can enable the BPDU guard feature if your switch is running PVST+, Rapid PVST+, or MSTP.

**Caution**

Configure PortFast only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast bpduguard default**
4. **interface *interface-id***
5. **spanning-tree portfast**
6. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree portfast bpduguard default | Globally enables BPDU guard. |

| | Command or Action | Purpose |
|---------------|--|---|
| | Example: Switch(config)# spanning-tree portfast bpduguard default | By default, BPDU guard is disabled. |
| Step 4 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/2 | Specifies the interface connected to an end station, and enters interface configuration mode. |
| Step 5 | spanning-tree portfast Example: Switch(config-if)# spanning-tree portfast | Enables the PortFast feature. |
| Step 6 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

What to Do Next

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the PortFast feature. When the port receives a BPDU, it is put in the error-disabled state.

Related Topics

[BPDU Guard, on page 70](#)

Enabling BPDU Filtering

You can also use the **spanning-tree bpdupfilter enable** interface configuration command to enable BPDU filtering on any interface without also enabling the PortFast feature. This command prevents the interface from sending or receiving BPDUs.



Caution

Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your switch is running PVST+, Rapid PVST+, or MSTP.

**Caution**

Configure PortFast only on interfaces that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast bpdufilter default**
4. **interface *interface-id***
5. **spanning-tree portfast**
6. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree portfast bpdufilter default Example: Switch(config)# spanning-tree portfast bpdufilter default | Globally enables BPDU filtering. By default, BPDU filtering is disabled. |
| Step 4 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/2 | Specifies the interface connected to an end station, and enters interface configuration mode. |
| Step 5 | spanning-tree portfast Example: Switch(config-if)# spanning-tree portfast | Enables the PortFast feature on the specified interface. |

| | Command or Action | Purpose |
|--------|---|----------------------------------|
| Step 6 | end Example: Switch(config-if) # end | Returns to privileged EXEC mode. |

Related Topics

[BPDU Filtering, on page 71](#)

Enabling UplinkFast for Use with Redundant Links



Note

When you enable UplinkFast, it affects all VLANs on the switch or switch stack. You cannot configure UplinkFast on an individual VLAN.

You can configure the UplinkFast or the Cross-Stack UplinkFast (CSUF) feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable UplinkFast and CSUF.

Before You Begin

UplinkFast cannot be enabled on VLANs that have been configured with a switch priority. To enable UplinkFast on a VLAN with switch priority configured, first restore the switch priority on the VLAN to the default value using the **no spanning-tree vlan *vlan-id* priority** global configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree uplinkfast** [*max-update-rate pkts-per-second*]
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree uplinkfast [max-update-rate pkts-per-second] Example: Switch(config)# spanning-tree uplinkfast max-update-rate 200 | Enables UplinkFast. (Optional) For <i>pkts-per-second</i> , the range is 0 to 32000 packets per second; the default is 150. If you set the rate to 0, station-learning frames are not generated, and the spanning-tree topology converges more slowly after a loss of connectivity. When you enter this command, CSUF also is enabled on all nonstack port interfaces. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

When UplinkFast is enabled, the switch priority of all VLANs is set to 49152. If you change the path cost to a value less than 3000 and you enable UplinkFast or UplinkFast is already enabled, the path cost of all interfaces and VLAN trunks is increased by 3000 (if you change the path cost to 3000 or above, the path cost is not altered). The changes to the switch priority and the path cost reduce the chance that a switch will become the root switch.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

When you enable the UplinkFast feature using these instructions, CSUF is automatically globally enabled on nonstack port interfaces.

Related Topics

[UplinkFast, on page 72](#)

[Cross-Stack UplinkFast, on page 73](#)

[How Cross-Stack UplinkFast Works, on page 74](#)

[Events That Cause Fast Convergence, on page 76](#)

Disabling UplinkFast

This procedure is optional.

Follow these steps to disable UplinkFast and Cross-Stack UplinkFast (CSUF).

Before You Begin

UplinkFast must be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree uplinkfast**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | no spanning-tree uplinkfast Example: Switch(config)# no spanning-tree uplinkfast | Disables UplinkFast and CSUF on the switch and all of its VLANs. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

When you disable the UplinkFast feature using these instructions, CSUF is automatically globally disabled on nonstack port interfaces.

Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.

You can configure the BackboneFast feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable BackboneFast on the switch.

Before You Begin

If you use BackboneFast, you must enable it on all switches in the network. BackboneFast is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree backbonefast**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree backbonefast Example: Switch(config)# spanning-tree backbonefast | Enables BackboneFast. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[BackboneFast, on page 76](#)

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable EtherChannel Guard on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree etherchannel guard misconfig**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | spanning-tree etherchannel guard misconfig Example: Switch(config)# spanning-tree etherchannel guard misconfig | Enables EtherChannel guard. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

What to Do Next

You can use the **show interfaces status err-disabled** privileged EXEC command to show which switch ports are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

Related Topics

[EtherChannel Guard, on page 79](#)

Enabling Root Guard

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. Do not enable the root guard on interfaces to be used by the UplinkFast feature. With UplinkFast, the backup interfaces (in the blocked state) replace the root port in the case of a failure. However, if root guard is also enabled, all the backup interfaces used by the UplinkFast feature are placed in the root-inconsistent state (blocked) and are prevented from reaching the forwarding state.

**Note**

You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable root guard on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree guard root**
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 2 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/2 | Specifies an interface to configure, and enters interface configuration mode. |
| Step 4 | spanning-tree guard root Example: Switch(config-if)# spanning-tree guard root | Enables root guard on the interface. By default, root guard is disabled on all interfaces. |
| Step 5 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[Root Guard, on page 79](#)

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on interfaces that are considered point-to-point by the spanning tree.



Note

You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional. Follow these steps to enable loop guard on the switch.

SUMMARY STEPS

1. Enter one of the following commands:
 - `show spanning-tree active`
 - `show spanning-tree mst`
2. `configure terminal`
3. `spanning-tree loopguard default`
4. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | Enter one of the following commands: <ul style="list-style-type: none"> • <code>show spanning-tree active</code> • <code>show spanning-tree mst</code> Example: <pre>Switch# show spanning-tree active OR Switch# show spanning-tree mst</pre> | Verifies which interfaces are alternate or root ports. |
| Step 2 | configure terminal Example: <pre>Switch# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | spanning-tree loopguard default Example: <pre>Switch(config)# spanning-tree loopguard default</pre> | Enables loop guard. By default, loop guard is disabled. |
| Step 4 | end Example: <pre>Switch(config)# end</pre> | Returns to privileged EXEC mode. |

Related Topics

[Loop Guard, on page 80](#)

Monitoring the Spanning-Tree Status

Table 11: Commands for Monitoring the Spanning-Tree Status

| Command | Purpose |
|---|--|
| show spanning-tree active | Displays spanning-tree information on active interfaces only. |
| show spanning-tree detail | Displays a detailed summary of interface information. |
| show spanning-tree interface <i>interface-id</i> | Displays spanning-tree information for the specified interface. |
| show spanning-tree mst interface <i>interface-id</i> | Displays MST information for the specified interface. |
| show spanning-tree summary [totals] | Displays a summary of interface states or displays the total lines of the spanning-tree state section. |

Additional References for Optional Spanning Tree Features

Related Documents

| Related Topic | Document Title |
|---------------------------------|---|
| Spanning tree protocol commands | <i>LAN Switching Command Reference, Cisco IOS XE Release 3SE (Catalyst 3650 Switches)</i> |

Error Message Decoder

| Description | Link |
|---|---|
| To help you research and resolve system error messages in this release, use the Error Message Decoder tool. | https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi |

Standards and RFCs

| Standard/RFC | Title |
|--------------|-------|
| None | — |

MIBs

| MIB | MIBs Link |
|--------------------------------------|--|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|---|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | http://www.cisco.com/support |

Feature Information for Optional Spanning-Tree Features

| Release | Modification |
|--|------------------------------|
| Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE | This feature was introduced. |



CHAPTER 4

Configuring EtherChannels

- [Finding Feature Information, page 95](#)
- [Restrictions for EtherChannels, page 95](#)
- [Information About EtherChannels, page 96](#)
- [How to Configure EtherChannels, page 115](#)
- [Monitoring EtherChannel, PAgP, and LACP Status, page 129](#)
- [Configuration Examples for Configuring EtherChannels, page 130](#)
- [Additional References for EtherChannels, page 132](#)
- [Feature Information for EtherChannels, page 133](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for EtherChannels

The following are restrictions for EtherChannels:

- All ports in an EtherChannel must be assigned to the same VLAN or they must be configured as trunk ports.
- Layer 3 EtherChannels are not supported if running the LAN Base license feature set.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

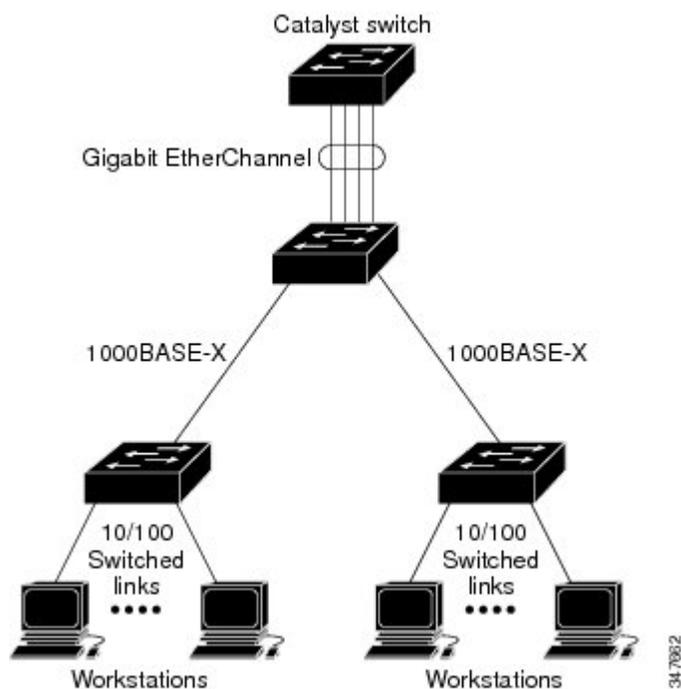
Information About EtherChannels

EtherChannel Overview

EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use the EtherChannel to increase the bandwidth between the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention.

An EtherChannel consists of individual Ethernet links bundled into a single logical link.

Figure 19: Typical EtherChannel Configuration



The EtherChannel provides full-duplex bandwidth up to 8 Gb/s (Gigabit EtherChannel) or 80 Gb/s (10-Gigabit EtherChannel) between your switch and another switch or host.

Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.

The number of EtherChannels is limited to 128.

All ports in each EtherChannel must be configured as either Layer 2 or Layer 3 ports. The EtherChannel Layer 3 ports are made up of routed ports. Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command. For more information, see the Configuring Interface Characteristics chapter.

Related Topics

[Configuring Layer 2 EtherChannels](#), on page 115

[EtherChannel Configuration Guidelines, on page 112](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

EtherChannel Modes

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. If the remote port cannot negotiate an EtherChannel, the local port is put into an independent state and continues to carry data traffic as would any other single link. The port configuration does not change, but the port does not participate in the EtherChannel.
- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

Related Topics

[Configuring Layer 2 EtherChannels , on page 115](#)

[EtherChannel Configuration Guidelines, on page 112](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

EtherChannel on Switches

You can create an EtherChannel on a switch, on a single switch in the stack, or on multiple switches in the stack (known as cross-stack EtherChannel).

Figure 20: Single-Switch EtherChannel

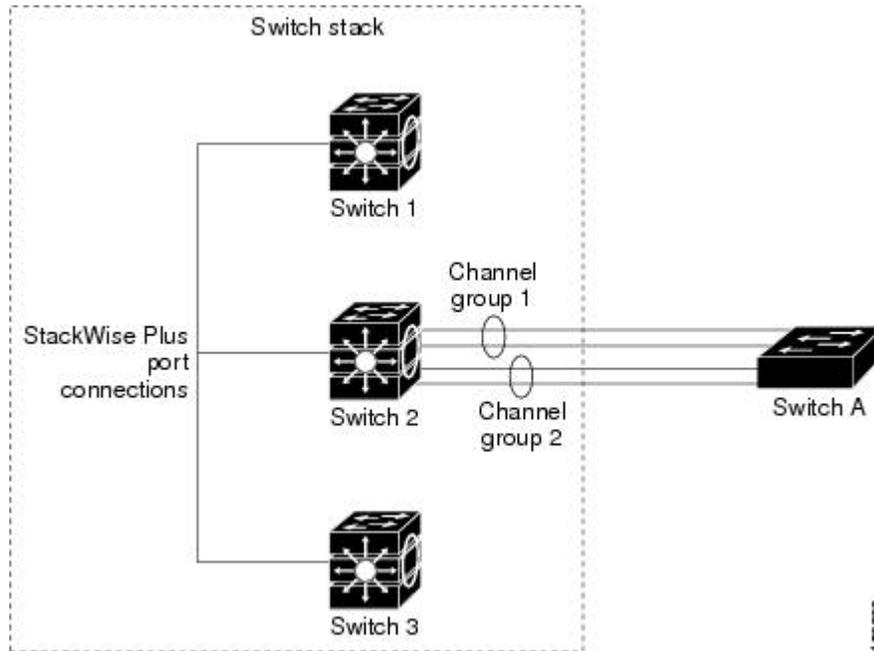


Figure 21: Cross-Stack EtherChannel

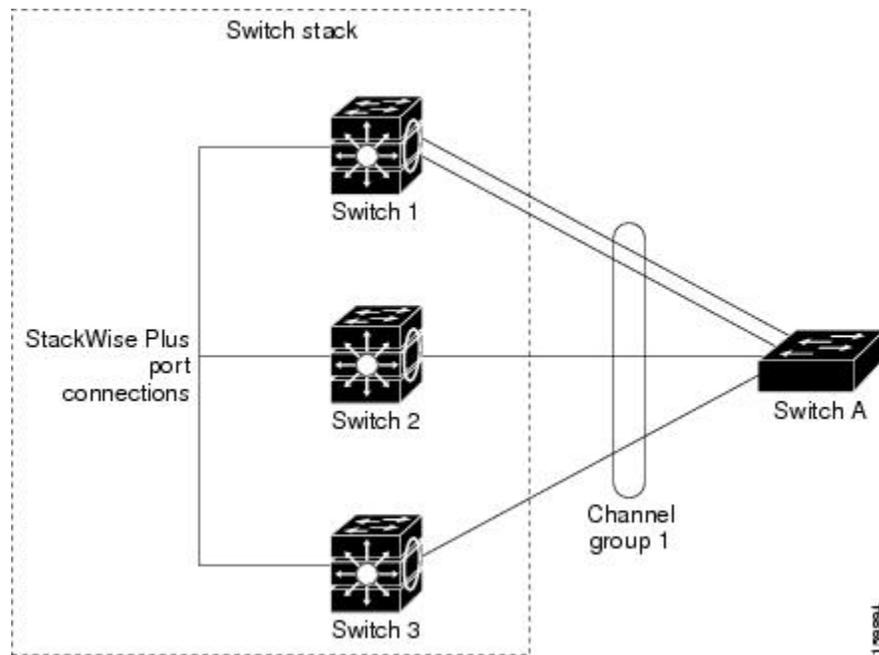
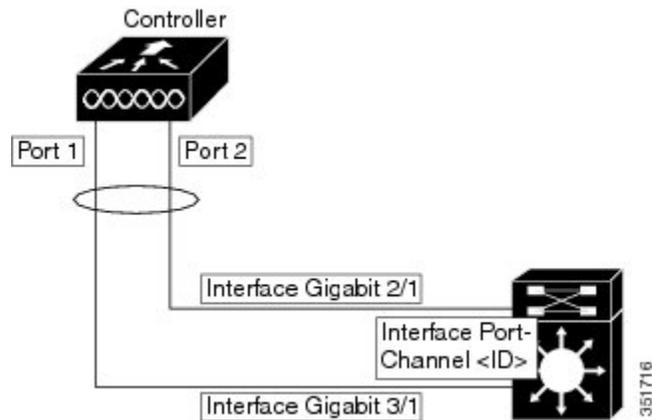
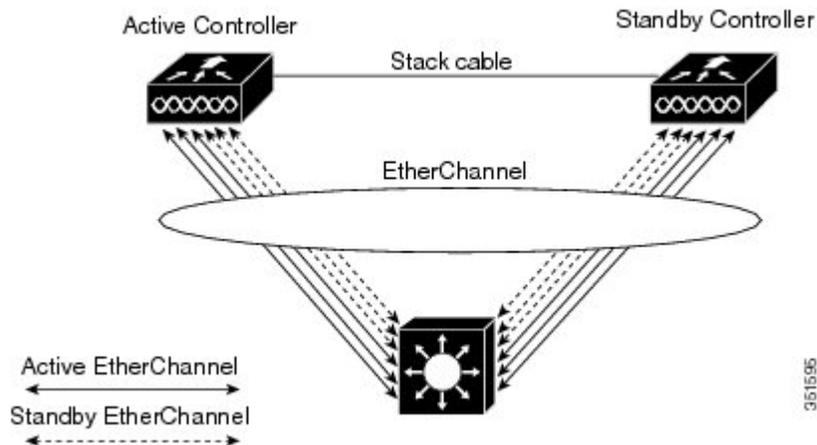


Figure 22: Single Controller EtherChannel



This figure shows the recommended cross-stack EtherChannel configuration with three active links and three standby links.

Figure 23: Cross-Stack EtherChannel



Related Topics

- [Configuring Layer 2 EtherChannels , on page 115](#)
- [EtherChannel Configuration Guidelines, on page 112](#)
- [Default EtherChannel Configuration, on page 111](#)
- [Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

EtherChannel Link Failover

If a link within an EtherChannel fails, traffic previously carried over that failed link moves to the remaining links within the EtherChannel. If traps are enabled on the switch, a trap is sent for a failure that identifies the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

Related Topics

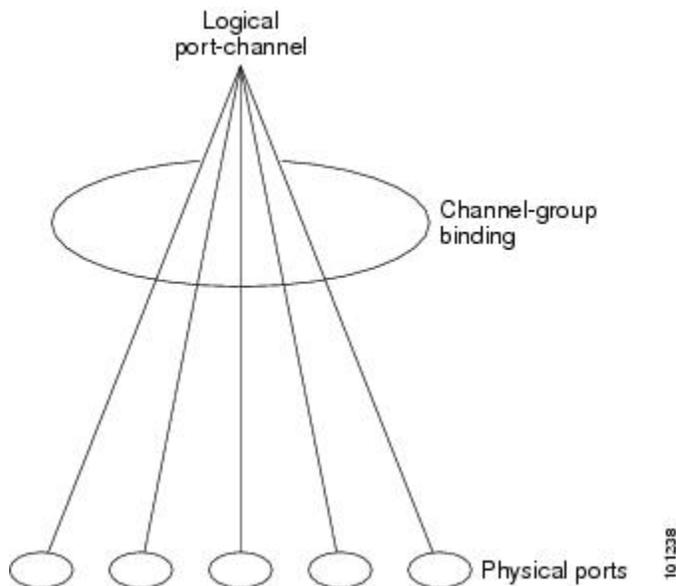
- [Configuring Layer 2 EtherChannels , on page 115](#)
- [EtherChannel Configuration Guidelines, on page 112](#)
- [Default EtherChannel Configuration, on page 111](#)
- [Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

Channel Groups and Port-Channel Interfaces

An EtherChannel comprises a channel group and a port-channel interface. The channel group binds physical ports to the port-channel interface. Configuration changes applied to the port-channel interface apply to all the physical ports bound together in the channel group.

The **channel-group** command binds the physical port and the port-channel interface together. Each EtherChannel has a port-channel logical interface numbered from 1 to 128. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.

Figure 24: Relationship of Physical Ports, Channel Group and Port-Channel Interface



- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel interface.

You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number*, or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.

- With Layer 3 ports, you should manually create the logical interface by using the **interface port-channel** global configuration command followed by the **no switchport** interface configuration command. You then manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command.
- With Layer 3 ports, use the **no switchport** interface command to configure the interface as a Layer 3 interface, and then use the **channel-group** interface configuration command to dynamically create the port-channel interface.

Related Topics

[Creating Port-Channel Logical Interfaces](#)

[EtherChannel Configuration Guidelines, on page 112](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

[Configuring the Physical Interfaces](#)

[EtherChannel Configuration Guidelines, on page 112](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

Port Aggregation Protocol

The Port Aggregation Protocol (PAgP) is a Cisco-proprietary protocol that can be run only on Cisco switches and on those switches licensed by vendors to support PAgP. PAgP facilitates the automatic creation of EtherChannels by exchanging PAgP packets between Ethernet ports.

By using PAgP, the switch or switch stack learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports (on a single switch in the stack) into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single switch port.

PAgP Modes

PAgP modes specify whether a port can send PAgP packets, which start PAgP negotiations, or only respond to PAgP packets received.

Table 12: EtherChannel PAgP Modes

| Mode | Description |
|------------------|--|
| auto | Places a port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets. |
| desirable | Places a port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This mode is not supported when the EtherChannel members are from different switches in the switch stack (cross-stack EtherChannel). |

Switch ports exchange PAgP packets only with partner ports configured in the **auto** or **desirable** modes. Ports configured in the **on** mode do not exchange PAgP packets.

Both the **auto** and **desirable** modes enable ports to negotiate with partner ports to form an EtherChannel based on criteria such as port speed. and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible. For example:

- A port in the **desirable** mode can form an EtherChannel with another port that is in the **desirable** or **auto** mode.
- A port in the **auto** mode can form an EtherChannel with another port in the **desirable** mode.

A port in the **auto** mode cannot form an EtherChannel with another port that is also in the **auto** mode because neither port starts PAgP negotiation.

Related Topics

[Configuring Layer 2 EtherChannels , on page 115](#)

[EtherChannel Configuration Guidelines](#), on page 112
[Default EtherChannel Configuration](#), on page 111
[Layer 2 EtherChannel Configuration Guidelines](#), on page 114
[Creating Port-Channel Logical Interfaces](#)
[EtherChannel Configuration Guidelines](#), on page 112
[Default EtherChannel Configuration](#), on page 111
[Layer 2 EtherChannel Configuration Guidelines](#), on page 114
[Configuring the Physical Interfaces](#)
[EtherChannel Configuration Guidelines](#), on page 112
[Default EtherChannel Configuration](#), on page 111
[Layer 2 EtherChannel Configuration Guidelines](#), on page 114

Silent Mode

If your switch is connected to a partner that is PAgP-capable, you can configure the switch port for nonsilent operation by using the **non-silent** keyword. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent mode is assumed.

Use the silent mode when the switch is connected to a device that is not PAgP-capable and seldom, if ever, sends packets. An example of a silent partner is a file server or a packet analyzer that is not generating traffic. In this case, running PAgP on a physical port connected to a silent partner prevents that switch port from ever becoming operational. However, the silent setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission.

Related Topics

[Configuring Layer 2 EtherChannels](#), on page 115
[EtherChannel Configuration Guidelines](#), on page 112
[Default EtherChannel Configuration](#), on page 111
[Layer 2 EtherChannel Configuration Guidelines](#), on page 114
[Creating Port-Channel Logical Interfaces](#)
[EtherChannel Configuration Guidelines](#), on page 112
[Default EtherChannel Configuration](#), on page 111
[Layer 2 EtherChannel Configuration Guidelines](#), on page 114
[Configuring the Physical Interfaces](#)
[EtherChannel Configuration Guidelines](#), on page 112
[Default EtherChannel Configuration](#), on page 111
[Layer 2 EtherChannel Configuration Guidelines](#), on page 114

PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports. The learn method must be configured the same at both ends of the link.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAGP cannot automatically detect when the partner device is a physical learner and when the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device to learn addresses by physical ports. You also must set the load-distribution method to source-based distribution, so that any given source MAC address is always sent on the same physical port.

You also can configure a single port within the group for all transmissions and use other ports for hot-standby. The unused ports in the group can be swapped into operation in just a few seconds if the selected single port loses hardware-signal detection. You can configure which port is always selected for packet transmission by changing its priority with the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.


Note

The switch supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the switch hardware, but they are required for PAGP interoperability with devices that only support address learning by physical ports, such as the Catalyst 1900 switch.

When the link partner of the switch is a physical learner, we recommend that you configure the switch as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command. Set the load-distribution method based on the source MAC address by using the **port-channel load-balance src-mac** global configuration command. The switch then sends packets to the physical learner using the same port in the EtherChannel from which it learned the source address. Only use the **pagp learn-method** command in this situation.

Related Topics

- [Configuring the PAGP Learn Method and Priority , on page 122](#)
- [EtherChannel Configuration Guidelines, on page 112](#)
- [Default EtherChannel Configuration, on page 111](#)
- [Monitoring EtherChannel, PAGP, and LACP Status, on page 129](#)
- [Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

PAGP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and the Cisco Discovery Protocol (CDP) send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive PAGP protocol data units (PDUs) on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel. For Layer 3 EtherChannels, the MAC address is allocated by the active switch as soon as the interface is created (through the **interface port-channel** global configuration command).

PAGP sends and receives PAGP PDUs only from ports that are up and have PAGP enabled for the auto or desirable mode.

Link Aggregation Control Protocol

The LACP is defined in IEEE 802.3ad and enables Cisco switches to manage Ethernet channels between switches that conform to the IEEE 802.3ad protocol. LACP facilitates the automatic creation of EtherChannels by exchanging LACP packets between Ethernet ports.

By using LACP, the switch or switch stack learns the identity of partners capable of supporting LACP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, LACP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, LACP adds the group to the spanning tree as a single switch port.

LACP Modes

LACP modes specify whether a port can send LACP packets or only receive LACP packets.

Table 13: EtherChannel LACP Modes

| Mode | Description |
|----------------|--|
| active | Places a port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. |
| passive | Places a port into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets. |

Both the **active** and **passive LACP** modes enable ports to negotiate with partner ports to an EtherChannel based on criteria such as port speed, and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A port in the **active** mode can form an EtherChannel with another port that is in the **active** or **passive** mode.
- A port in the **passive** mode cannot form an EtherChannel with another port that is also in the **passive** mode because neither port starts LACP negotiation.

Related Topics

- [Configuring Layer 2 EtherChannels](#), on page 115
- [EtherChannel Configuration Guidelines](#), on page 112
- [Default EtherChannel Configuration](#), on page 111
- [Layer 2 EtherChannel Configuration Guidelines](#), on page 114

LACP and Link Redundancy

LACP port-channel operation, bandwidth availability, and link redundancy can be further refined with the LACP port-channel min-links and the LACP max-bundle features.

The LACP port-channel min-links feature:

- Configures the minimum number of ports that must be linked up and bundled in the LACP port channel.
- Prevents a low-bandwidth LACP port channel from becoming active.
- Causes an LACP port channel to become inactive if there are too few active members ports to supply the required minimum bandwidth.

The LACP max-bundle feature:

- Defines an upper limit on the number of bundled ports in an LACP port channel.
- Allows hot-standby ports with fewer bundled ports. For example, in an LACP port channel with five ports, you can specify a max-bundle of three, and the two remaining ports are designated as hot-standby ports.

Related Topics

[Configuring the LACP Max Bundle Feature](#) , on page 124

[Configuring LACP Hot-Standby Ports: Example](#), on page 131

[Configuring the Port Channel Min-Links Feature](#) , on page 125

[Configuring LACP Hot-Standby Ports: Example](#), on page 131

LACP Interaction with Other Features

The DTP and the CDP send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive LACP PDUs on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel. For Layer 3 EtherChannels, the MAC address is allocated by the active switch as soon as the interface is created through the **interface port-channel** global configuration command.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. The **on** mode can be useful if the remote device does not support PAgP or LACP. In the **on** mode, a usable EtherChannel exists only when the switches at both ends of the link are configured in the **on** mode.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.

**Caution**

You should use care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load-Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. You can specify one of several different load-balancing modes, including load distribution based on MAC addresses, IP addresses, source addresses, destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the switch.

You configure the load-balancing and forwarding method by using the **port-channel load-balance** and the **port-channel load-balance extended** global configuration commands.

Related Topics

[Configuring EtherChannel Load-Balancing](#), on page 119

[EtherChannel Configuration Guidelines](#), on page 112

[Layer 2 EtherChannel Configuration Guidelines](#), on page 114

[Default EtherChannel Configuration](#), on page 111

[Layer 3 EtherChannel Configuration Guidelines](#), on page 115

MAC Address Forwarding

With source-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide load-balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel.

With destination-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the destination host's MAC address of the incoming packet. Therefore, packets to the same destination are forwarded over the same port, and packets to a different destination are sent on a different port in the channel.

With source-and-destination MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on both the source and destination MAC addresses. This forwarding method, a combination source-MAC and destination-MAC address forwarding methods of load distribution, can be used if it is not clear whether source-MAC or destination-MAC address forwarding is better suited on a particular switch. With source-and-destination MAC-address forwarding, packets sent from host A to host B, host A to host C, and host C to host B could all use different ports in the channel.

Related Topics

[Configuring EtherChannel Load-Balancing](#), on page 119

[EtherChannel Configuration Guidelines](#), on page 112

[Layer 2 EtherChannel Configuration Guidelines](#), on page 114

[Default EtherChannel Configuration](#), on page 111

[Layer 3 EtherChannel Configuration Guidelines, on page 115](#)

IP Address Forwarding

With source-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. To provide load balancing, packets from different IP addresses use different ports in the channel, and packets from the same IP address use the same port in the channel.

With destination-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. To provide load balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. Packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on both the source and destination IP addresses of the incoming packet. This forwarding method, a combination of source-IP and destination-IP address-based forwarding, can be used if it is not clear whether source-IP or destination-IP address-based forwarding is better suited on a particular switch. In this method, packets sent from the IP address A to IP address B, from IP address A to IP address C, and from IP address C to IP address B could all use different ports in the channel.

Related Topics

[Configuring EtherChannel Load-Balancing , on page 119](#)

[EtherChannel Configuration Guidelines, on page 112](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 3 EtherChannel Configuration Guidelines, on page 115](#)

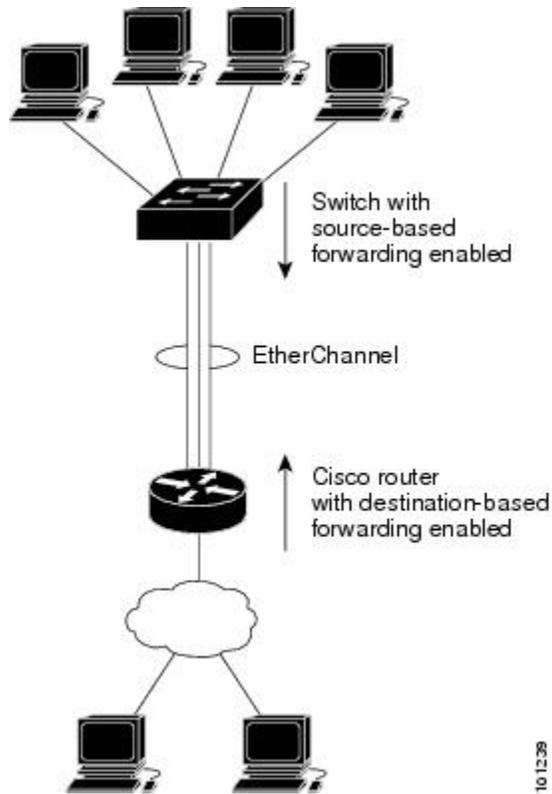
Load-Balancing Advantages

Different load-balancing methods have different advantages, and the choice of a particular load-balancing method should be based on the position of the switch in the network and the kind of traffic that needs to be load-distributed.

In the following figure, an EtherChannel of four workstations communicates with a router. Because the router is a single MAC-address device, source-based forwarding on the switch EtherChannel ensures that the switch

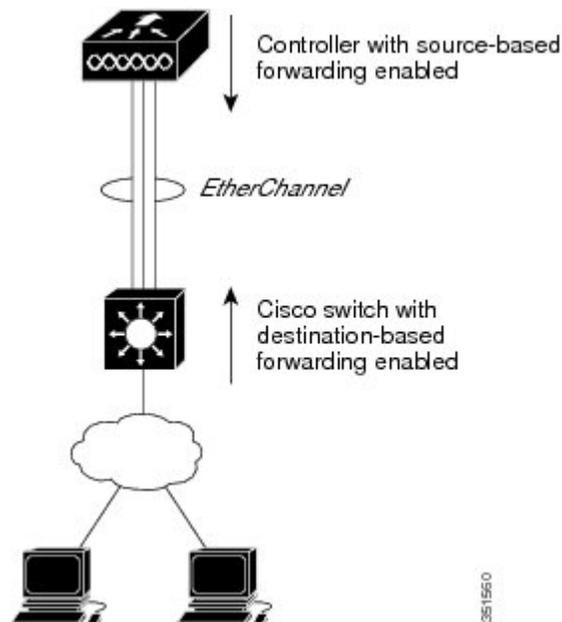
uses all available bandwidth to the router. The router is configured for destination-based forwarding because the large number of workstations ensures that the traffic is evenly distributed from the router EtherChannel.

Figure 25: Load Distribution and Forwarding Methods



In the figure below the controller does source MAC-based forwarding over the EtherChannel links and evenly distributes the traffic over multiple links of the EtherChannel.

Figure 26: Load Distribution and Forwarding



Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load-balancing.

Related Topics

[Configuring EtherChannel Load-Balancing](#), on page 119

[EtherChannel Configuration Guidelines](#), on page 112

[Layer 2 EtherChannel Configuration Guidelines](#), on page 114

[Default EtherChannel Configuration](#), on page 111

[Layer 3 EtherChannel Configuration Guidelines](#), on page 115

EtherChannel and Switch Stacks

If a stack member that has ports participating in an EtherChannel fails or leaves the stack, the active switch removes the failed stack member switch ports from the EtherChannel. The remaining ports of the EtherChannel, if any, continue to provide connectivity.

When a switch is added to an existing stack, the new switch receives the running configuration from the active switch and updates itself with the EtherChannel-related stack configuration. The stack member also receives the operational information (the list of ports that are up and are members of a channel).

When two stacks merge that have EtherChannels configured between them, self-looped ports result. Spanning tree detects this condition and acts accordingly. Any PAgP or LACP configuration on a winning switch stack is not affected, but the PAgP or LACP configuration on the losing switch stack is lost after the stack reboots.

Switch Stack and PAgP

With PAgP, if the active switch fails or leaves the stack, the standby switch becomes the new active switch. A spanning-tree reconvergence is not triggered unless there is a change in the EtherChannel bandwidth. The new active switch synchronizes the configuration of the stack members to that of the active switch. The PAgP configuration is not affected after an active switch change unless the EtherChannel has ports residing on the old active switch.

Switch Stacks and LACP

With LACP, the system ID uses the stack MAC address from the active switch. When an active switch fails or leaves the stack and the standby switch becomes the new active switch change, the LACP system ID is unchanged. By default, the LACP configuration is not affected after the active switch changes.

Default EtherChannel Configuration

The default EtherChannel configuration is described in this table.

Table 14: Default EtherChannel Configuration

| Feature | Default Setting |
|--------------------------------|--|
| Channel groups | None assigned. |
| Port-channel logical interface | None defined. |
| PAgP mode | No default. |
| PAgP learn method | Aggregate-port learning on all ports. |
| PAgP priority | 128 on all ports. |
| LACP mode | No default. |
| LACP learn method | Aggregate-port learning on all ports. |
| LACP port priority | 32768 on all ports. |
| LACP system priority | 32768. |
| LACP system ID | LACP system priority and the switch or stack MAC address. |
| Load-balancing | Load distribution on the switch is based on the source-MAC address of the incoming packet. |

Related Topics

- [Configuring Layer 2 EtherChannels , on page 115](#)
- [EtherChannel Overview, on page 96](#)
- [EtherChannel Modes, on page 97](#)
- [EtherChannel on Switches, on page 98](#)
- [EtherChannel Link Failover, on page 100](#)
- [LACP Modes, on page 105](#)
- [PAgP Modes , on page 102](#)
- [Silent Mode, on page 103](#)
- [Creating Port-Channel Logical Interfaces](#)
- [Channel Groups and Port-Channel Interfaces, on page 100](#)
- [PAgP Modes , on page 102](#)
- [Silent Mode, on page 103](#)
- [Configuring the Physical Interfaces](#)
- [Channel Groups and Port-Channel Interfaces, on page 100](#)
- [PAgP Modes , on page 102](#)
- [Silent Mode, on page 103](#)
- [Configuring EtherChannel Load-Balancing , on page 119](#)
- [Load-Balancing and Forwarding Methods, on page 107](#)
- [MAC Address Forwarding, on page 107](#)
- [IP Address Forwarding, on page 108](#)
- [Load-Balancing Advantages, on page 108](#)
- [Configuring the PAgP Learn Method and Priority , on page 122](#)
- [PAgP Learn Method and Priority, on page 103](#)
- [Configuring the LACP System Priority , on page 126](#)
- [Configuring the LACP Port Priority , on page 127](#)

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Do not try to configure more than 128 EtherChannels on the switch or switch stack.
- Configure a PAgP EtherChannel with up to eight Ethernet ports of the same type.
- Configure a LACP EtherChannel with up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- Enable all ports in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining ports in the EtherChannel.

- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
 - Allowed-VLAN list
 - Spanning-tree path cost for each VLAN
 - Spanning-tree port priority for each VLAN
 - Spanning-tree Port Fast setting
- Do not configure a port to be a member of more than one EtherChannel group.
- Do not configure an EtherChannel in both the PAgP and LACP modes. EtherChannel groups running PAgP and LACP can coexist on the same switch or on different switches in the stack. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.
- Do not configure a secure port as part of an EtherChannel or the reverse.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.
- If EtherChannels are configured on switch interfaces, remove the EtherChannel configuration from the interfaces before globally enabling IEEE 802.1x on a switch by using the **dot1x system-auth-control** global configuration command.
- If cross-stack EtherChannel is configured and the switch stack partitions, loops and forwarding issues can occur.

Related Topics

- [Configuring Layer 2 EtherChannels](#) , on page 115
- [EtherChannel Overview](#) , on page 96
- [EtherChannel Modes](#) , on page 97
- [EtherChannel on Switches](#) , on page 98
- [EtherChannel Link Failover](#) , on page 100
- [LACP Modes](#) , on page 105
- [PAgP Modes](#) , on page 102
- [Silent Mode](#) , on page 103
- [Creating Port-Channel Logical Interfaces](#)
- [Channel Groups and Port-Channel Interfaces](#) , on page 100
- [PAgP Modes](#) , on page 102
- [Silent Mode](#) , on page 103
- [Configuring the Physical Interfaces](#)
- [Channel Groups and Port-Channel Interfaces](#) , on page 100
- [PAgP Modes](#) , on page 102
- [Silent Mode](#) , on page 103
- [Configuring EtherChannel Load-Balancing](#) , on page 119
- [Load-Balancing and Forwarding Methods](#) , on page 107

- [MAC Address Forwarding, on page 107](#)
- [IP Address Forwarding, on page 108](#)
- [Load-Balancing Advantages, on page 108](#)
- [Configuring the PAgP Learn Method and Priority , on page 122](#)
- [PAgP Learn Method and Priority, on page 103](#)
- [Configuring the LACP System Priority , on page 126](#)
- [Configuring the LACP Port Priority , on page 127](#)

Layer 2 EtherChannel Configuration Guidelines

When configuring Layer 2 EtherChannels, follow these guidelines:

- Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
- An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
- Ports with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.

Related Topics

- [Configuring Layer 2 EtherChannels , on page 115](#)
- [EtherChannel Overview, on page 96](#)
- [EtherChannel Modes, on page 97](#)
- [EtherChannel on Switches, on page 98](#)
- [EtherChannel Link Failover, on page 100](#)
- [LACP Modes, on page 105](#)
- [PAgP Modes , on page 102](#)
- [Silent Mode, on page 103](#)
- [Creating Port-Channel Logical Interfaces](#)
- [Channel Groups and Port-Channel Interfaces, on page 100](#)
- [PAgP Modes , on page 102](#)
- [Silent Mode, on page 103](#)
- [Configuring the Physical Interfaces](#)
- [Channel Groups and Port-Channel Interfaces, on page 100](#)
- [PAgP Modes , on page 102](#)
- [Silent Mode, on page 103](#)
- [Configuring EtherChannel Load-Balancing , on page 119](#)
- [Load-Balancing and Forwarding Methods, on page 107](#)
- [MAC Address Forwarding, on page 107](#)
- [IP Address Forwarding, on page 108](#)

[Load-Balancing Advantages](#), on page 108

[Configuring the PAgP Learn Method and Priority](#), on page 122

[PAgP Learn Method and Priority](#), on page 103

[Configuring the LACP System Priority](#), on page 126

[Configuring the LACP Port Priority](#), on page 127

Layer 3 EtherChannel Configuration Guidelines

- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical ports in the channel.

Related Topics

[Configuring EtherChannel Load-Balancing](#), on page 119

[Load-Balancing and Forwarding Methods](#), on page 107

[MAC Address Forwarding](#), on page 107

[IP Address Forwarding](#), on page 108

[Load-Balancing Advantages](#), on page 108

How to Configure EtherChannels

After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface, and configuration changes applied to the physical port affect only the port where you apply the configuration.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by assigning ports to a channel group with the **channel-group** interface configuration command. This command automatically creates the port-channel logical interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {access | trunk}
4. **switchport access vlan** *vlan-id*
5. **channel-group** *channel-group-number* **mode** {auto [non-silent] | desirable [non-silent] | on } | { active | passive}
6. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters the global configuration mode. |
| Step 2 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1 | Specifies a physical port, and enters interface configuration mode. Valid interfaces are physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode. |
| Step 3 | switchport mode { access trunk } Example: Switch(config-if)# switchport mode access | Assigns all ports as static-access ports in the same VLAN, or configure them as trunks. If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094. |
| Step 4 | switchport access vlan <i>vlan-id</i> Example: Switch(config-if)# switchport access vlan 22 | (Optional) If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094. |
| Step 5 | channel-group <i>channel-group-number</i> mode { auto [non-silent] desirable [non-silent] } on } { active passive } Example: Switch(config-if)# channel-group 5 mode auto | Assigns the port to a channel group, and specifies the PAgP or the LACP mode. For <i>channel-group-number</i> , the range is 1 to 128. For mode , select one of these keywords: <ul style="list-style-type: none"> • auto —Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • desirable —Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • on —Forces the port to channel without PAgP or LACP. In the on mode, an EtherChannel exists only when a port group in the on mode is connected to another port group in the on mode. |

| | Command or Action | Purpose |
|---------------|--|---|
| | | <ul style="list-style-type: none"> • non-silent —(Optional) If your switch is connected to a partner that is PAgP-capable, configures the switch port for nonsilent operation when the port is in the auto or desirable mode. If you do not specify non-silent, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • active—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • passive —Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. |
| Step 6 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[EtherChannel Overview, on page 96](#)

[EtherChannel Modes, on page 97](#)

[EtherChannel on Switches, on page 98](#)

[EtherChannel Link Failover, on page 100](#)

[LACP Modes, on page 105](#)

[PAgP Modes, on page 102](#)

[Silent Mode, on page 103](#)

[EtherChannel Configuration Guidelines, on page 112](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

Configuring Layer 3 EtherChannels

Follow these steps to assign an Ethernet port to a Layer 3 EtherChannel. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no ip address**
5. **no switchport**
6. **channel-group** *channel-group-number* **mode** { **auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive** }
7. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2 | Specifies a physical port, and enters interface configuration mode. Valid interfaces include physical ports. For a PAGP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode. |
| Step 4 | no ip address Example: Switch(config-if)# no ip address | Ensures that there is no IP address assigned to the physical port. |
| Step 5 | no switchport Example: Switch(config-if)# no switchport | Puts the port into Layer 3 mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 6 | <p>channel-group <i>channel-group-number</i> mode { auto [non-silent] desirable [non-silent] on } { active passive }</p> <p>Example:</p> <pre>Switch(config-if)# channel-group 5 mode auto</pre> | <p>Assigns the port to a channel group, and specifies the PAgP or the LACP mode.</p> <p>For mode, select one of these keywords:</p> <ul style="list-style-type: none"> • auto—Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • desirable—Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • on—Forces the port to channel without PAgP or LACP. In the on mode, an EtherChannel exists only when a port group in the on mode is connected to another port group in the on mode. • non-silent—(Optional) If your switch is connected to a partner that is PAgP capable, configures the switch port for nonsilent operation when the port is in the auto or desirable mode. If you do not specify non-silent, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • active—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • passive —Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. |
| Step 7 | <p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre> | <p>Returns to privileged EXEC mode.</p> |

Configuring EtherChannel Load-Balancing

You can configure EtherChannel load-balancing to use one of several different forwarding methods.

This task is optional.

SUMMARY STEPS

1. `configure terminal`
2. `port-channel load-balance { dst-ip | dst-mac | dst-mixed-ip-port | dst-port | extended [dst-ip | dst-mac | dst-port | ipv6-label | l3-proto | src-ip | src-mac | src-port] | src-dst-ip | src-dst-mac | src-dst-mixed-ip-port | src-dst-portsrc-ip | src-mac | src-mixed-ip-port | src-port }`
3. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | <p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre> | Enters global configuration mode. |
| Step 2 | <p><code>port-channel load-balance { dst-ip dst-mac dst-mixed-ip-port dst-port extended [dst-ip dst-mac dst-port ipv6-label l3-proto src-ip src-mac src-port] src-dst-ip src-dst-mac src-dst-mixed-ip-port src-dst-portsrc-ip src-mac src-mixed-ip-port src-port }</code></p> <p>Example:</p> <pre>Switch(config)# port-channel load-balance src-mac</pre> | <p>Configures an EtherChannel load-balancing method.</p> <p>The default is <code>src-mac</code>.</p> <p>Select one of these load-distribution methods:</p> <ul style="list-style-type: none"> • dst-ip—Specifies destination-host IP address. • dst-mac—Specifies the destination-host MAC address of the incoming packet. • dst-mixed-ip-port—Specifies the host IP address and TCP/UDP port. • dst-port—Specifies the destination TCP/UDP port. • extended—Specifies extended load balance methods—combinations of source and destination methods beyond those available with the standard command. • ipv6-label—Specifies the IPv6 flow label. • l3-proto—Specifies the Layer 3 protocol. • src-dst-ip—Specifies the source and destination host IP address. • src-dst-mac—Specifies the source and destination host MAC address. • src-dst-mixed-ip-port—Specifies the source and destination host IP address and TCP/UDP port. • src-dst-port—Specifies the source and destination TCP/UDP port. • src-ip—Specifies the source host IP address. • src-mac—Specifies the source MAC address of the incoming packet. • src-mixed-ip-port—Specifies the source host IP address and TCP/UDP port. |

| | Command or Action | Purpose |
|---------------|---|--|
| | | <ul style="list-style-type: none"> • src-port—Specifies the source TCP/UDP port. |
| Step 3 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[Load-Balancing and Forwarding Methods](#), on page 107

[MAC Address Forwarding](#), on page 107

[IP Address Forwarding](#), on page 108

[Load-Balancing Advantages](#), on page 108

[EtherChannel Configuration Guidelines](#), on page 112

[Layer 2 EtherChannel Configuration Guidelines](#), on page 114

[Default EtherChannel Configuration](#), on page 111

[Layer 3 EtherChannel Configuration Guidelines](#), on page 115

Configuring EtherChannel Extended Load-Balancing

Configure EtherChannel extended load-balancing when you want to use a combination of load-balancing methods.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **port-channel load-balance extended [dst-ip | dst-mac dst-port | ipv6-label | l3-proto | src-ip | src-mac | src-port]**
3. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|-----------------------------------|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |

| | Command or Action | Purpose |
|--------|---|--|
| Step 2 | <p>port-channel load-balance extended [dst-ip dst-mac dst-port ipv6-label I3-proto src-ip src-mac src-port]</p> <p>Example:</p> <pre>Switch(config)# port-channel load-balance extended dst-ip dst-mac src-ip</pre> | <p>Configures an EtherChannel extended load-balancing method. The default is src-mac.</p> <p>Select one of these load-distribution methods:</p> <ul style="list-style-type: none"> • dst-ip—Specifies destination-host IP address. • dst-mac—Specifies the destination-host MAC address of the incoming packet. • dst-port—Specifies the destination TCP/UDP port. • ipv6-label—Specifies the IPv6 flow label. • I3-proto—Specifies the Layer 3 protocol. • src-ip—Specifies the source host IP address. • src-mac—Specifies the source MAC address of the incoming packet. • src-port—Specifies the source TCP/UDP port. |
| Step 3 | <p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre> | <p>Returns to privileged EXEC mode.</p> |

Configuring the PAgP Learn Method and Priority

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **pagp learn-method physical-port**
4. **pagp port-priority** *priority*
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2 | Specifies the port for transmission, and enters interface configuration mode. |
| Step 3 | pagp learn-method physical-port Example: Switch(config-if)# pagp learn-method physical port | <p>Selects the PAgP learning method.</p> <p>By default, aggregation-port learning is selected, which means the switch sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.</p> <p>Selects physical-port to connect with another switch that is a physical learner. Make sure to configure the port-channel load-balance global configuration command to src-mac.</p> <p>The learning method must be configured the same at both ends of the link.</p> |
| Step 4 | pagp port-priority <i>priority</i> Example: Switch(config-if)# pagp port-priority 200 | <p>Assigns a priority so that the selected port is chosen for packet transmission.</p> <p>For <i>priority</i>, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the port will be used for PAgP transmission.</p> |
| Step 5 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[PAgP Learn Method and Priority](#), on page 103

[EtherChannel Configuration Guidelines](#), on page 112

[Default EtherChannel Configuration](#), on page 111

[Monitoring EtherChannel, PAgP, and LACP Status](#), on page 129

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

Configuring LACP Hot-Standby Ports

When LACP is enabled, the software, by default, tries to configure the maximum number of LACP-compatible ports in a channel, up to a maximum of 16 ports. Only eight LACP links can be active at one time; the remaining eight links are placed in hot-standby mode. If one of the active links becomes inactive, a link that is in the hot-standby mode becomes active in its place.

You can override the default behavior by specifying the maximum number of active ports in a channel, in which case, the remaining ports become hot-standby ports. For example, if you specify a maximum of five ports in a channel, up to 11 ports become hot-standby ports.

If you configure more than eight links for an EtherChannel group, the software automatically decides which of the hot-standby ports to make active based on the LACP priority. To every link between systems that operate LACP, the software assigns a unique priority made up of these elements (in priority order):

- LACP system priority
- System ID (the switch MAC address)
- LACP port priority
- Port number

In priority comparisons, numerically lower values have higher priority. The priority decides which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

Determining which ports are active and which are hot standby is a two-step procedure. First the system with a numerically lower system priority and system ID is placed in charge of the decision. Next, that system decides which ports are active and which are hot standby, based on its values for port priority and port number. The port priority and port number values for the other system are not used.

You can change the default values of the LACP system priority and the LACP port priority to affect how the software selects active and standby links.

Configuring the LACP Max Bundle Feature

When you specify the maximum number of bundled LACP ports allowed in a port channel, the remaining ports in the port channel are designated as hot-standby ports.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of LACP ports in a port channel. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface port-channel** *channel-number*
3. **lACP max-bundle** *max-bundle-number*
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface port-channel <i>channel-number</i> Example: Switch(config)# interface port-channel 2 | Enters interface configuration mode for a port channel. The range is 1 to 128. |
| Step 3 | lACP max-bundle <i>max-bundle-number</i> Example: Switch(config-if)# lACP max-bundle 3 | Specifies the maximum number of LACP ports in the port-channel bundle. The range is 1 to 8. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[LACP and Link Redundancy](#) , on page 106

[Configuring LACP Hot-Standby Ports: Example](#), on page 131

Configuring the Port Channel Min-Links Feature

You can specify the minimum number of active ports that must be in the link-up state and bundled in an EtherChannel for the port channel interface to transition to the link-up state.

Beginning in privileged EXEC mode, follow these steps to configure the minimum number of links that are required for a port channel. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface port-channel *channel-number***
3. **port-channel min-links *min-links-number***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface port-channel <i>channel-number</i> Example: Switch(config)# interface port-channel 2 | Enters interface configuration mode for a port-channel. For <i>channel-number</i> , the range is 1 to 128. |
| Step 3 | port-channel min-links <i>min-links-number</i> Example: Switch(config-if)# port-channel min-links 3 | Specifies the minimum number of member ports that must be in the link-up state and bundled in the EtherChannel for the port channel interface to transition to the link-up state. For <i>min-links-number</i> , the range is 2 to 8. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

[LACP and Link Redundancy](#), on page 106

[Configuring LACP Hot-Standby Ports: Example](#), on page 131

Configuring the LACP System Priority

You can configure the system priority for all the EtherChannels that are enabled for LACP by using the **lacp system-priority** global configuration command. You cannot configure a system priority for each LACP-configured channel. By changing this value from the default, you can affect how the software selects active and standby links.

You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an H port-state flag).

Follow these steps to configure the LACP system priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lacp system-priority** *priority*
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 3 | lacp system-priority <i>priority</i> Example: Switch(config)# lacp system-priority 32000 | Configures the LACP system priority. The range is 1 to 65535. The default is 32768. The lower the value, the higher the system priority. |
| Step 4 | end Example: Switch(config)# end | Returns to privileged EXEC mode. |

Related Topics

- [EtherChannel Configuration Guidelines, on page 112](#)
- [Default EtherChannel Configuration, on page 111](#)
- [Layer 2 EtherChannel Configuration Guidelines, on page 114](#)
- [Monitoring EtherChannel, PAgP, and LACP Status, on page 129](#)

Configuring the LACP Port Priority

By default, all ports use the same port priority. If the local system has a lower value for the system priority and the system ID than the remote system, you can affect which of the hot-standby links become active first

by changing the port priority of LACP EtherChannel ports to a lower value than the default. The hot-standby ports that have lower port numbers become active in the channel first. You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an H port-state flag).



Note If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), all the ports that cannot be actively included in the EtherChannel are put in the hot-standby state and are used only if one of the channeled ports fails.

Follow these steps to configure the LACP port priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lacp port-priority** *priority*
5. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Switch> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2 | Specifies the port to be configured, and enters interface configuration mode. |
| Step 4 | lacp port-priority <i>priority</i> Example: Switch(config-if)# lacp port-priority 32000 | Configures the LACP port priority. The range is 1 to 65535. The default is 32768. The lower the value, the more likely that the port will be used for LACP transmission. |

| | Command or Action | Purpose |
|--------|--|----------------------------------|
| Step 5 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[EtherChannel Configuration Guidelines, on page 112](#)

[Default EtherChannel Configuration, on page 111](#)

[Layer 2 EtherChannel Configuration Guidelines, on page 114](#)

[Monitoring EtherChannel, PAgP, and LACP Status, on page 129](#)

Monitoring EtherChannel, PAgP, and LACP Status

You can display EtherChannel, PAgP, and LACP status using the commands listed in this table.

Table 15: Commands for Monitoring EtherChannel, PAgP, and LACP Status

| Command | Description |
|--|---|
| clear lacp { <i>channel-group-number</i> counters counters } | Clears LACP channel-group information and traffic counters. |
| clear pagp { <i>channel-group-number</i> counters counters } | Clears PAgP channel-group information and traffic counters. |
| show etherchannel [<i>channel-group-number</i> { detail port port-channel protocol summary }] [detail load-balance port port-channel protocol summary] | Displays EtherChannel information in a brief, detailed, and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, port-channel, and protocol information. |
| show pagp [<i>channel-group-number</i>] { counters internal neighbor } | Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information. |
| show pagp [<i>channel-group-number</i>] dual-active | Displays the dual-active detection status. |
| show lacp [<i>channel-group-number</i>] { counters internal neighbor sys-id } | Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information. |
| show running-config | Verifies your configuration entries. |

| Command | Description |
|---------------------------------------|---|
| show etherchannel load-balance | Displays the load balance or frame distribution scheme among ports in the port channel. |

Related Topics

[Configuring the PAgP Learn Method and Priority](#) , on page 122

[PAgP Learn Method and Priority](#) , on page 103

[Configuring the LACP System Priority](#) , on page 126

[Configuring the LACP Port Priority](#) , on page 127

Configuration Examples for Configuring EtherChannels

Configuring Layer 2 EtherChannels: Examples

This example shows how to configure an EtherChannel on a single switch in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode desirable non-silent
Switch(config-if-range)# end
```

This example shows how to configure an EtherChannel on a single switch in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It uses LACP passive mode and assigns two ports on stack member 1 and one port on stack member 2 as static-access ports in VLAN 10 to channel 5:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/4 -5
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode passive
Switch(config-if-range)# exit
Switch(config)# interface gigabitethernet3/0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# channel-group 5 mode passive
```

```
Switch(config-if) # exit
```

Configuring Layer 3 EtherChannels: Examples

This example shows how to configure a Layer 3 EtherChannel. It assigns two ports to channel 5 with the LACP mode active:

```
Switch# configure terminal
Switch(config) # interface range gigabitethernet2/0/1 -2
Switch(config-if-range) # no ip address
Switch(config-if-range) # no switchport
Switch(config-if-range) # channel-group 5 mode active
Switch(config-if-range) # end
```

This example shows how to configure a cross-stack Layer 3 EtherChannel. It assigns two ports on stack member 2 and one port on stack member 3 to channel 7 using LACP active mode:

```
Switch# configure terminal
Switch(config) # interface range gigabitethernet2/0/4 -5
Switch(config-if-range) # no ip address
Switch(config-if-range) # no switchport
Switch(config-if-range) # channel-group 7 mode active
Switch(config-if-range) # exit
Switch(config) # interface gigabitethernet3/0/3
Switch(config-if) # no ip address
Switch(config-if) # no switchport
Switch(config-if) # channel-group 7 mode active
Switch(config-if) # exit
```

Configuring LACP Hot-Standby Ports: Example

This example shows how to configure an Etherchannel (port channel 2) that will be active when there are at least three active ports, will comprise up to seven active ports and the remaining ports (up to nine) as hot-standby ports :

```
Switch# configure terminal
Switch(config) # interface port-channel 2
Switch(config-if) # port-channel min-links 3
Switch(config-if) # lacp max-bundle 7
```

Related Topics

[Configuring the LACP Max Bundle Feature](#) , on page 124

[LACP and Link Redundancy](#) , on page 106

[Configuring the Port Channel Min-Links Feature](#) , on page 125

[LACP and Link Redundancy](#) , on page 106

Additional References for EtherChannels

Related Documents

| Related Topic | Document Title |
|---------------------------|---|
| Layer 2 command reference | <i>Layer 2 Command Reference (Cisco WLC 5700 Series)</i> <i>Layer 2/3 Command Reference (Catalyst 3650 Switches)</i> |

Error Message Decoder

| Description | Link |
|---|---|
| To help you research and resolve system error messages in this release, use the Error Message Decoder tool. | https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi |

Standards and RFCs

| Standard/RFC | Title |
|--------------|-------|
| None | — |

MIBs

| MIB | MIBs Link |
|--------------------------------------|--|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|--|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p>http://www.cisco.com/support</p> |

Feature Information for EtherChannels

| Release | Modification |
|---------------------------------------|--|
| Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE | This feature was introduced. |
| Cisco IOS XE 3.3SE | Support for the LACP max-bundle feature and the port channel min-links features was added. |



Configuring Flex Links and the MAC Address-Table Move Update Feature

- [Finding Feature Information, page 135](#)
- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, page 135](#)
- [Information About Flex Links and MAC Address-Table Move Update, page 136](#)
- [How to Configure Flex Links and the MAC Address-Table Move Update Feature, page 143](#)
- [Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, page 149](#)
- [Configuration Examples for Flex Links, page 150](#)
- [Additional References for Flex Links and MAC Address-Table Move Update, page 155](#)
- [Feature Information for Flex Links and MAC Address-Table Move Update, page 156](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for Configuring Flex Links and MAC Address-Table Move Update

- Flex Links are supported only on Layer 2 ports and port channels.
- You can configure up to 16 backup links.

- You can configure only one Flex Links backup link for any active link, and it must be a different interface from the active interface.
- An interface can belong to only one Flex Links pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Links pair.
- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Gigabit Ethernet or port channel) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on Flex Links ports. A Flex Links port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Related Topics

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

[Configuring Flex Links: Examples](#), on page 150

[Configuring VLAN Load Balancing on Flex Links](#) , on page 146

[Configuring VLAN Load Balancing on Flex Links: Examples](#), on page 151

[Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages](#) , on page 148

[Configuring MAC Address-Table Move Update](#) , on page 147

[Configuring the MAC Address-Table Move Update: Examples](#), on page 152

Information About Flex Links and MAC Address-Table Move Update

Flex Links

Flex Links are a pair of a Layer 2 interfaces (switch ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. Flex Links are typically configured in service provider or enterprise networks where customers do not want to run STP on the switch. If the switch is running STP, Flex Links are not necessary because STP already provides link-level redundancy or backup.

You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Links or backup link. On switches, the Flex Links can be on the same switch or on another switch in the stack. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on Flex Links interfaces.

Related Topics

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

[Configuring Flex Links: Examples](#), on page 150

Flex Links Configuration

In the following figure, ports 1 and 2 on switch A are connected to uplink switches B and C. Because they are configured as Flex Links, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

You can also configure a preemption function, specifying the preferred port for forwarding traffic. For example, you can configure the Flex Links pair with preemption mode. In the scenario shown, when port 1 comes back up and has more bandwidth than port 2, port 1 begins forwarding traffic after 60 seconds. Port 2 becomes the standby port. You do this by entering the **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** interface configuration commands.

Figure 27: Flex Links Configuration Example

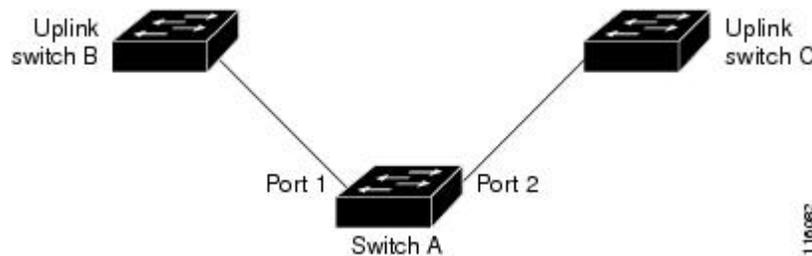
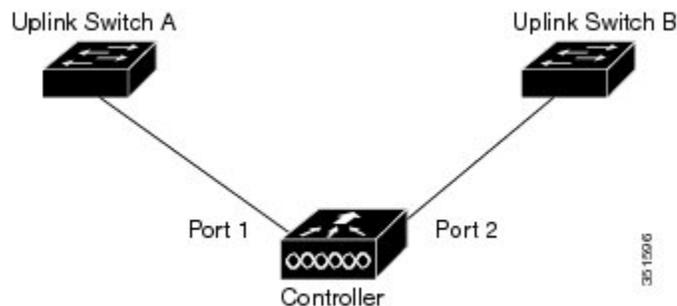


Figure 28: Flex Links Configuration Example



If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

Related Topics

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

VLAN Flex Links Load Balancing and Support

VLAN Flex Links load balancing allows users to configure a Flex Links pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if Flex Links ports are configured for 1 to 100 VLANs, the traffic of the first 50 VLANs can be forwarded on one port and the rest on the other port. If one of the ports fail, the other active port forwards all the traffic. When the failed port comes back up, it resumes forwarding traffic in the preferred VLANs. In addition to providing the redundancy, this Flex Links pair can be used for load balancing. Flex Links VLAN load balancing does not impose any restrictions on uplink switches.

The following figure displays a VLAN Flex Links load-balancing configuration.

Figure 29: VLAN Flex Links Load-Balancing Configuration Example

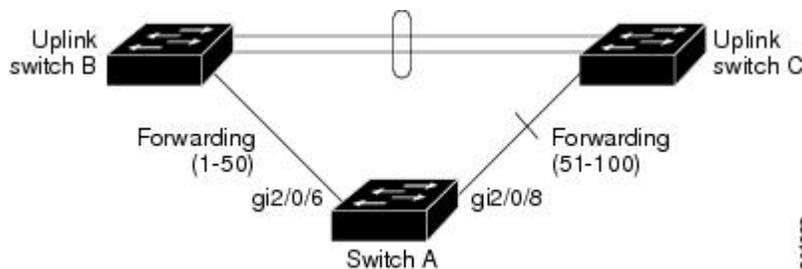
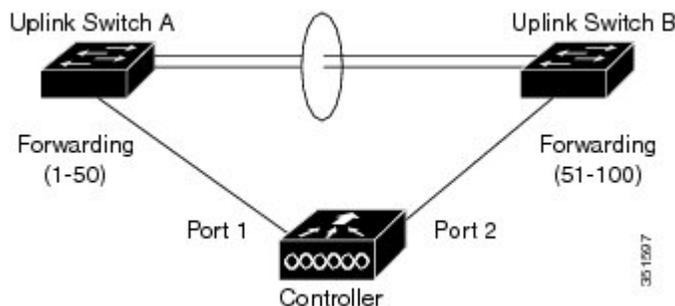


Figure 30: VLAN Flex Links Load-Balancing Configuration Example



Multicast Fast Convergence with Flex Links Failover

Multicast fast convergence reduces the multicast traffic convergence time after a Flex Links failure. Multicast fast convergence is implemented by a combination of learning the backup link as an mrouter port, generating IGMP reports, and leaking IGMP reports.

Related Topics

[Configuring Multicast Fast Convergence with Flex Links Failover: Examples](#) , on page 152

Learning the Other Flex Links Port as the mrouter Port

In a typical multicast network, there is a querier for each VLAN. A switch deployed at the edge of a network has one of its Flex Links ports receiving queries. Flex Links ports are also always forwarding at any given time.

A port that receives queries is added as an mrouter port on the switch. An mrouter port is part of all the multicast groups learned by the switch. After a changeover, queries are received by the other Flex Links port. The other Flex Links port is then learned as the mrouter port. After changeover, multicast traffic then flows through the other Flex Links port. To achieve faster convergence of traffic, both Flex Links ports are learned as mrouter ports whenever either Flex Links port is learned as the mrouter port. Both Flex Links ports are always part of multicast groups.

Although both Flex Links ports are part of the groups in normal operation mode, all traffic on the backup port is blocked. The normal multicast data flow is not affected by the addition of the backup port as an mrouter port. When the changeover happens, the backup port is unblocked, allowing the traffic to flow. In this case, the upstream multicast data flows as soon as the backup port is unblocked.

Generating IGMP Reports

When the backup link comes up after the changeover, the upstream new distribution switch does not start forwarding multicast data, because the port on the upstream router, which is connected to the blocked Flex Links port, is not part of any multicast group. The reports for the multicast groups were not forwarded by the downstream switch because the backup link is blocked. The data does not flow on this port, until it learns the multicast groups, which occurs only after it receives reports.

The reports are sent by hosts when a general query is received, and a general query is sent within 60 seconds in normal scenarios. When the backup link starts forwarding, to achieve faster convergence of multicast data, the downstream switch immediately sends proxy reports for all the learned groups on this port without waiting for a general query.

Leaking IGMP Reports

To achieve multicast traffic convergence with minimal loss, a redundant data path must be set up before the Flex Links active link goes down. This can be achieved by leaking only IGMP report packets on the Flex Links backup link. These leaked IGMP report messages are processed by upstream distribution routers, so multicast data traffic gets forwarded to the backup interface. Because all incoming traffic on the backup interface is dropped at the ingress of the access switch, no duplicate multicast traffic is received by the host. When the Flex Links active link fails, the access switch starts accepting traffic from the backup link immediately. The only disadvantage of this scheme is that it consumes bandwidth on the link between the distribution switches and on the backup link between the distribution and access switches. This feature is disabled by default and can be configured by using the **switchport backup interface *interface-id* multicast fast-convergence** command.

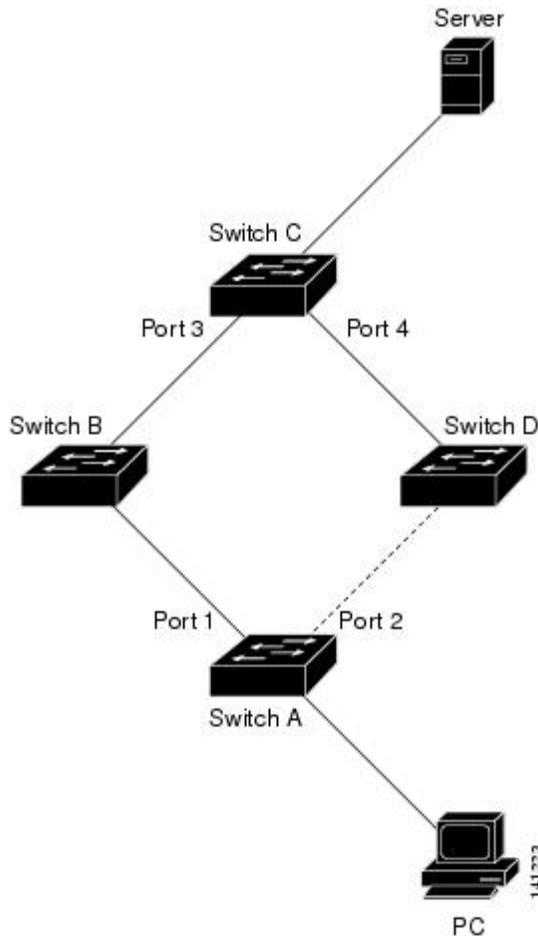
When this feature has been enabled at changeover, the switch does not generate the proxy reports on the backup port, which became the forwarding port.

MAC Address-Table Move Update

The MAC address-table move update feature allows the switch to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

In the following figure, switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink switches B and D through a Flex Links pair. Port 1 is forwarding traffic, and port 2 is in the backup state. Traffic from the PC to the server is forwarded from port 1 to port 3. The MAC address of the PC has been learned on port 3 of switch C. Traffic from the server to the PC is forwarded from port 3 to port 1.

Figure 31: MAC Address-Table Move Update Example



If the MAC address-table move update feature is not configured and port 1 goes down, port 2 starts forwarding traffic. However, for a short time, switch C keeps forwarding traffic from the server to the PC through port 3, and the PC does not get the traffic because port 1 is down. If switch C removes the MAC address of the PC on port 3 and relearns it on port 4, traffic can then be forwarded from the server to the PC through port 2.

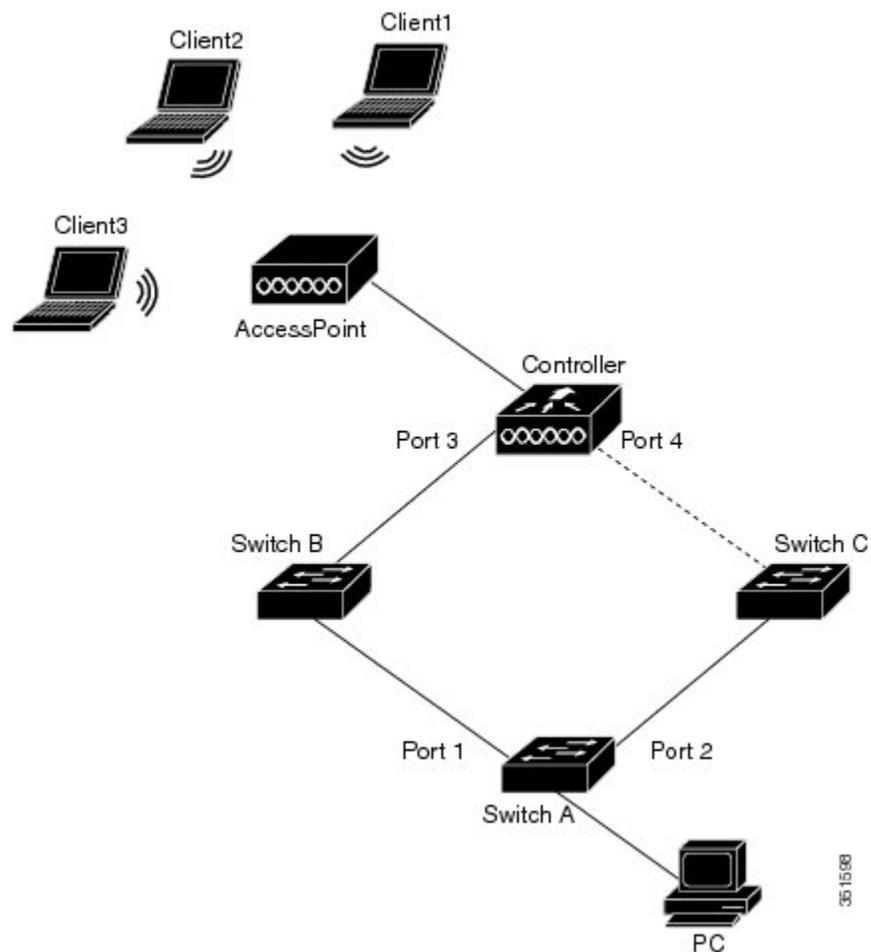
If the MAC address-table move update feature is configured and enabled on the switches, and port 1 goes down, port 2 starts forwarding traffic from the PC to the server. The switch sends a MAC address-table move update packet from port 2. Switch C gets this packet on port 4 and immediately learns the MAC address of the PC on port 4, which reduces the reconvergence time.

You can configure the access switch, switch A, to *send* MAC address-table move update messages. You can also configure the uplink switches B, C, and D to *get* and process the MAC address-table move update messages. When switch C gets a MAC address-table move update message from switch A, switch C learns the MAC address of the PC on port 4. Switch C updates the MAC address table, including the forwarding table entry for the PC.

Switch A does not need to wait for the MAC address-table update. The switch detects a failure on port 1 and immediately starts forwarding server traffic from port 2, the new forwarding port. This change occurs in less than 100 milliseconds (ms). The PC is directly connected to switch A, and the connection status does not change. Switch A does not need to update the PC entry in the MAC address table.

In the following figure, the controller and switches B and C form the Flexlink through a Flex Links pair. Port 3 is forwarding traffic, and port 4 is in the backup state. Switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink switches B and C. The MAC addresses of the wireless clients have been learnt on port 1 of the switch A.

Figure 32: MAC Address-Table Move Update Example



In the above figure, three wireless clients connect to an access point and communicate to the controller. The PC connected to switch A communicates with the wireless clients through the data path from port 3 to port 1. If the MAC address-table move update feature is not configured on the controller and port 3 goes down, port 4 starts forwarding traffic. However, for a short time the wireless clients will not be able to pass any traffic to the PC as port 3 is down.

If the MAC address-table move update feature is configured and enabled on the controller, and port 3 goes down, the controller sends a MAC address-table move update packet (MMU) from port 4. This MMU packet carries all the MAC addresses of the wireless clients. Switch C gets this packet on port 4 and immediately

learns the MAC addresses of the wireless clients, which reduces the reconvergence time. The PC transmits data to the wireless clients using the path from port 2 to port 4. Switch C also bridges the same MMU packet in the VLAN so that all the switches in the network will update the MAC address table to ensure that the next packet to any wireless client towards controller comes on the right path. The controller learns only the MAC addresses of the wireless clients.

Related Topics

[Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages](#) , on page 148

[Configuring MAC Address-Table Move Update](#) , on page 147

[Configuring the MAC Address-Table Move Update: Examples](#), on page 152

Flex Links VLAN Load Balancing Configuration Guidelines

- For Flex Links VLAN load balancing, you must choose the preferred VLANs on the backup interface.
- You cannot configure a preemption mechanism and VLAN load balancing for the same Flex Links pair.

Related Topics

[Configuring VLAN Load Balancing on Flex Links](#) , on page 146

[Configuring VLAN Load Balancing on Flex Links: Examples](#), on page 151

MAC Address-Table Move Update Configuration Guidelines

- You can enable and configure this feature on the access switch to *send* the MAC address-table move updates.
- You can enable and configure this feature on the uplink switches to *get* the MAC address-table move updates.

Default Flex Links and MAC Address-Table Move Update Configuration

- Flex Links is not configured, and there are no backup interfaces defined.
- The preemption mode is off.
- The preemption delay is 35 seconds.
- The MAC address-table move update feature is not configured on the switch.

Related Topics

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

[Configuring Flex Links: Examples](#), on page 150

How to Configure Flex Links and the MAC Address-Table Move Update Feature

Configuring Flex Links

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `switchport backup interface interface-id`
4. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | <p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre> | Enters global configuration mode. |
| Step 2 | <p><code>interface interface-id</code></p> <p>Example:</p> <pre>Switch(conf)# interface gigabitethernet1/0/1</pre> | Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128. |
| Step 3 | <p><code>switchport backup interface interface-id</code></p> <p>Example:</p> <pre>Switch(conf-if)# switchport backup interface gigabitethernet1/0/2</pre> | Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode. |
| Step 4 | <p><code>end</code></p> <p>Example:</p> <pre>Switch(conf-if)# end</pre> | Returns to privileged EXEC mode. |

Related Topics

[Flex Links](#), on page 136

[Default Flex Links and MAC Address-Table Move Update Configuration](#), on page 142

[Restrictions for Configuring Flex Links and MAC Address-Table Move Update](#), on page 135

[Configuring Flex Links: Examples](#), on page 150

[Flex Links Configuration](#), on page 137

[Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update](#), on page 149

[Configuring Flex Links: Examples](#), on page 150

Configuring a Preemption Scheme for a Pair of Flex Links**SUMMARY STEPS**

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]
5. **switchport backup interface** *interface-id* **preemption delay** *delay-time*
6. **end**
7. **show interface** [*interface-id*] **switchport backup**
8. **copy running-config startup config**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode |
| Step 2 | interface <i>interface-id</i> Example: Switch(conf)# interface gigabitethernet1/0/1 | Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128. |
| Step 3 | switchport backup interface <i>interface-id</i> Example: Switch(conf-if)# switchport backup interface gigabitethernet1/0/2 | Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 4 | <p>switchport backup interface <i>interface-id</i> preemption mode [forced bandwidth off]</p> <p>Example:</p> <pre>Switch(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption mode forced</pre> | <p>Configures a preemption mechanism and delay for a Flex Links interface pair. You can configure the preemption as:</p> <ul style="list-style-type: none"> • forced—(Optional) The active interface always preempts the backup. • bandwidth—(Optional) The interface with the higher bandwidth always acts as the active interface. • off—(Optional) No preemption occurs from active to backup. |
| Step 5 | <p>switchport backup interface <i>interface-id</i> preemption delay <i>delay-time</i></p> <p>Example:</p> <pre>Switch(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption delay 50</pre> | <p>Configures the time delay until a port preempts another port.</p> <p>Note Setting a delay time only works with forced and bandwidth modes.</p> |
| Step 6 | <p>end</p> <p>Example:</p> <pre>Switch(conf-if)# end</pre> | Returns to privileged EXEC mode. |
| Step 7 | <p>show interface [<i>interface-id</i>] switchport backup</p> <p>Example:</p> <pre>Switch# show interface gigabitethernet1/0/2 switchport backup</pre> | Verifies the configuration. |
| Step 8 | <p>copy running-config startup config</p> <p>Example:</p> <pre>Switch# copy running-config startup config</pre> | (Optional) Saves your entries in the switch startup configuration file. |

Related Topics

[Flex Links, on page 136](#)

[Default Flex Links and MAC Address-Table Move Update Configuration, on page 142](#)

[Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 135](#)

[Configuring Flex Links: Examples, on page 150](#)

[Flex Links Configuration, on page 137](#)

[Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update](#), on page 149

[Configuring Flex Links: Examples](#), on page 150

Configuring VLAN Load Balancing on Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **switchport backup interface *interface-id* prefer vlan *vlan-range***
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> Example: Switch (config)# interface gigabitethernet2/0/6 | Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128. |
| Step 3 | switchport backup interface <i>interface-id</i> prefer vlan <i>vlan-range</i> Example: Switch (config-if)# switchport backup interface gigabitethernet2/0/8 prefer vlan 2 | Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface and specifies the VLANs carried on the interface. The VLAN ID range is 1 to 4094. |
| Step 4 | end Example: Switch (config-if)# end | Returns to privileged EXEC mode. |

Related Topics

- [Flex Links VLAN Load Balancing Configuration Guidelines, on page 142](#)
- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 135](#)
- [Configuring VLAN Load Balancing on Flex Links: Examples, on page 151](#)
- [Configuring VLAN Load Balancing on Flex Links: Examples, on page 151](#)
- [Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 149](#)

Configuring MAC Address-Table Move Update

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. Use one of the following:
 - **switchport backup interface *interface-id***
 - **switchport backup interface *interface-id* mmu primary vlan *vlan-id***
4. **end**
5. **mac address-table move update transmit**
6. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> Example: Switch# interface gigabitethernet1/0/1 | Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128. |
| Step 3 | Use one of the following: <ul style="list-style-type: none"> • switchport backup interface <i>interface-id</i> • switchport backup interface <i>interface-id</i> mmu primary vlan <i>vlan-id</i> | Configures a physical Layer 2 interface (or port channel), as part of a Flex Links pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. Configure a physical Layer 2 interface (or port channel) and specifies the VLAN ID on the interface, which is used for sending the MAC address-table move update. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Example: <pre>Switch(config-if)# switchport backup interface gigabitethernet0/2 mmu primary vlan 2</pre> | When one link is forwarding traffic, the other interface is in standby mode. |
| Step 4 | end Example: <pre>Switch(config-if)# end</pre> | Returns to global configuration mode. |
| Step 5 | mac address-table move update transmit Example: <pre>Switch(config)# mac address-table move update transmit</pre> | Enables the access switch to send MAC address-table move updates to other switches in the network if the primary link goes down and the switch starts forwarding traffic through the standby link. |
| Step 6 | end Example: <pre>Switch(config)# end</pre> | Returns to privileged EXEC mode. |

Related Topics

- [Configuring the MAC Address-Table Move Update: Examples, on page 152](#)
- [Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 149](#)
- [MAC Address-Table Move Update, on page 139](#)
- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 135](#)
- [Configuring the MAC Address-Table Move Update: Examples, on page 152](#)

Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table move update receive**
3. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode |
| Step 2 | mac address-table move update receive Example: Switch (config)# mac address-table move update receive | Enables the switch to obtain and processes the MAC address-table move updates. |
| Step 3 | end Example: Switch (config)# end | Returns to privileged EXEC mode. |

Related Topics

[Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update](#), on page 149

[Configuring the MAC Address-Table Move Update: Examples](#), on page 152

[MAC Address-Table Move Update](#), on page 139

[Restrictions for Configuring Flex Links and MAC Address-Table Move Update](#), on page 135

[Configuring the MAC Address-Table Move Update: Examples](#), on page 152

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update

| Command | Purpose |
|---|---|
| show interface [<i>interface-id</i>] switchport backup | Displays the Flex Links backup interface configured for an interface or all the configured Flex Links and the state of each active and backup interface (up or standby mode). |
| show ip igmp profile address-table move update <i>profile-id</i> | Displays the specified IGMP profile or all the IGMP profiles defined on the switch. |

| Command | Purpose |
|---|---|
| <code>show mac address-table move update</code> | Displays the MAC address-table move update information on the switch. |

Related Topics

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

Configuration Examples for Flex Links

Configuring Flex Links: Examples

This example shows how to verify the configuration after you configure an interface with a backup interface:

```
Switch# show interface switchport backup

Switch Backup Interface Pairs:
Active Interface Backup Interface State
-----
GigabitEthernet1/0/1 GigabitEthernet1/0/2 Active Up/Backup Standby
```

This example shows how to verify the configuration after you configure the preemption mode as forced for a backup interface pair:

```
Switch# show interface switchport backup detail

Switch Backup Interface Pairs:
Active Interface Backup Interface State
-----
GigabitEthernet1/0/211 GigabitEthernet1/0/2 Active Up/Backup Standby
Interface Pair : Gi1/0/1, Gi1/0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi1/0/1), 100000 Kbit (Gi1/0/2)
Mac Address Move Update Vlan : auto
```

Related Topics

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

[Flex Links](#), on page 136

[Default Flex Links and MAC Address-Table Move Update Configuration](#), on page 142

[Restrictions for Configuring Flex Links and MAC Address-Table Move Update](#), on page 135

[Configuring a Preemption Scheme for a Pair of Flex Links](#) , on page 144

[Configuring Flex Links](#) , on page 143

Configuring VLAN Load Balancing on Flex Links: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the switch:

```
Switch(config)# interface gigabitethernet 2/0/6
Switch(config-if)# switchport backup interface gigabitethernet 2/0/8 prefer vlan 60,100-120
```

When both interfaces are up, Gi2/0/8 forwards traffic for VLANs 60 and 100 to 120 and Gi2/0/6 forwards traffic for VLANs 1 to 50.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the Flex Links pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the Flex Links pair.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface Gi2/0/6 comes up, VLANs preferred on this interface are blocked on the peer interface Gi2/0/8 and forwarded on Gi2/0/6.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

```
Switch# show interfaces switchport backup detail

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
FastEthernet1/0/3     FastEthernet1/0/4     Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4094
Vlans Preferred on Backup Interface: 3-4
Preemption Mode      : off
```

```
Bandwidth : 10000 Kbit (Fa1/0/3), 100000 Kbit (Fa1/0/4)
Mac Address Move Update Vlan : auto
```

Related Topics

- [Configuring VLAN Load Balancing on Flex Links , on page 146](#)
- [Flex Links VLAN Load Balancing Configuration Guidelines, on page 142](#)
- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 135](#)
- [Configuring VLAN Load Balancing on Flex Links , on page 146](#)

Configuring the MAC Address-Table Move Update: Examples

This example shows how to verify the configuration after you configure an access switch to send MAC address-table move updates:

```
Switch# show mac address-table move update

Switch-ID : 010b.4630.1780
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/On, Xmt Off/On
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 5
Rcv conforming packet count : 5
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : Po2
Rcv last src-mac-address : 000b.462d.c502
Rcv last switch-ID : 0403.fd6a.8700
Xmt packet count : 0
Xmt packet count this min : 0
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None
```

Related Topics

- [Configuring MAC Address-Table Move Update , on page 147](#)
- [Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages , on page 148](#)
- [Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages , on page 148](#)
- [Configuring MAC Address-Table Move Update , on page 147](#)
- [MAC Address-Table Move Update, on page 139](#)
- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 135](#)

Configuring Multicast Fast Convergence with Flex Links Failover: Examples

These are configuration examples for learning the other Flex Links port as the mrouter port when Flex Links is configured on GigabitEthernet1/0/11 and GigabitEthernet1/0/12, and output for the **show interfaces switchport backup** command:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Switch(config)# interface GigabitEthernet1/0/11
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport backup interface GigabitEthernet1/0/12
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet1/0/12
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# end
Switch# show interfaces switchport backup detail
Switch Backup Interface Pairs:
Active Interface Backup Interface State
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : Off
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLANs 1 and 401, with their queries reaching the switch through GigabitEthernet1/0/11:

```
Switch# show ip igmp snooping querier
```

| Vlan | IP Address | IGMP Version | Port |
|------|------------|--------------|----------|
| 1 | 1.1.1.1 | v2 | Gi1/0/11 |
| 401 | 41.41.41.1 | v2 | Gi1/0/11 |

This example is output for the **show ip igmp snooping mrouter** command for VLANs 1 and 401:

```
Switch# show ip igmp snooping mrouter
```

| Vlan | ports |
|------|--------------------------------------|
| 1 | Gi1/0/11(dynamic), Gi1/0/12(dynamic) |
| 401 | Gi1/0/11(dynamic), Gi1/0/12(dynamic) |

Similarly, both Flex Links ports are part of learned groups. In this example, GigabitEthernet2/0/11 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Switch# show ip igmp snooping groups
```

| Vlan | Group | Type | Version | Port List |
|------|-----------|------|---------|------------------------------|
| 1 | 228.1.5.1 | igmp | v2 | Gi1/0/11, Gi1/0/12, Gi2/0/11 |
| 1 | 228.1.5.2 | igmp | v2 | Gi1/0/11, Gi1/0/12, Gi2/0/11 |

When a host responds to the general query, the switch forwards this report on all the mrouter ports. In this example, when a host sends a report for the group 228.1.5.1, it is forwarded only on GigabitEthernet1/0/11, because the backup port GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding.

As soon as this port starts forwarding, the switch sends proxy reports for the groups 228.1.5.1 and 228.1.5.2 on behalf of the host. The upstream router learns the groups and starts forwarding multicast data. This is the default behavior of Flex Links. This behavior changes when the user configures fast convergence using the **switchport backup interface gigabitEthernet 1/0/12 multicast fast-convergence** command. This example shows turning on this feature:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitEthernet 1/0/11
Switch(config-if)# switchport backup interface gigabitEthernet 1/0/12 multicast
fast-convergence
Switch(config-if)# exit
```

```
Switch# show interfaces switchport backup detail
```

```
Switch Backup Interface Pairs:
Active      Interface      Backup Interface State
-----
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : On
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLAN 1 and 401 with their queries reaching the switch through GigabitEthernet1/0/11:

```
Switch# show ip igmp snooping querier
```

```
Vlan      IP Address      IGMP Version      Port
-----
1         1.1.1.1         v2                 Gi1/0/11
401      41.41.41.1     v2                 Gi1/0/11
```

This is output for the **show ip igmp snooping mrouter** command for VLAN 1 and 401:

```
Switch# show ip igmp snooping mrouter
```

```
Vlan      ports
-----
1         Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401      Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

Similarly, both the Flex Links ports are a part of the learned groups. In this example, GigabitEthernet2/0/11 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Switch# show ip igmp snooping groups
```

```
Vlan      Group      Type      Version      Port List
-----
1         228.1.5.1  igmp      v2           Gi1/0/11, Gi1/0/12, Gi2/0/11
1         228.1.5.2  igmp      v2           Gi1/0/11, Gi1/0/12, Gi2/0/11
```

Whenever a host responds to the general query, the switch forwards this report on all the mrouter ports. When you turn on this feature through the command-line port, and when a report is forwarded by the switch on GigabitEthernet1/0/11, it is also leaked to the backup port GigabitEthernet1/0/12. The upstream router learns the groups and starts forwarding multicast data, which is dropped at the ingress because GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding. You do not need to send any proxy reports as the multicast data is already being forwarded by the upstream router. By leaking reports to the backup port, a redundant multicast path has been set up, and the time taken for the multicast traffic convergence is very minimal.

Related Topics

[Multicast Fast Convergence with Flex Links Failover](#), on page 138

Additional References for Flex Links and MAC Address-Table Move Update

Related Documents

| Related Topic | Document Title |
|-------------------------------------|--|
| Layer 2 command reference | <i>Layer 2 Command Reference (Cisco WLC 5700 Series) Layer 2/3 Command Reference (Catalyst 3650 Switches)</i> |
| switchport backup interface command | <i>Interface Command Reference (Cisco WLC 5700 Series) Interface and Hardware Component Command Reference (Catalyst 3650 Switches)</i> |

Error Message Decoder

| Description | Link |
|---|---|
| To help you research and resolve system error messages in this release, use the Error Message Decoder tool. | https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi |

Standards and RFCs

| Standard/RFC | Title |
|--------------|-------|
| None | — |

MIBs

| MIB | MIBs Link |
|--------------------------------------|--|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|--|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p>http://www.cisco.com/support</p> |

Feature Information for Flex Links and MAC Address-Table Move Update

| Release | Modification |
|---------------------------------------|--|
| Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE | This feature was introduced. |
| Cisco IOS XE 3.3SE | Support for multicast fast convergence with Flex Links failover was added. |



Configuring UniDirectional Link Detection

- [Finding Feature Information, page 157](#)
- [Restrictions for Configuring UDLD, page 157](#)
- [Information About UDLD, page 158](#)
- [How to Configure UDLD, page 161](#)
- [Monitoring and Maintaining UDLD, page 164](#)
- [Additional References for UDLD, page 164](#)
- [Feature Information for UDLD, page 165](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for Configuring UDLD

The following are restrictions for configuring UniDirectional Link Detection (UDLD):

- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.

**Caution**

Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Information About UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it disables the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected ports on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected ports on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to learn the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

Normal Mode

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic port are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the ports are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In this case, the logical link is considered undetermined, and UDLD does not disable the port.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up because the Layer 1 mechanisms detects a physical problem with the link. In this case, UDLD does not take any action and the logical link is considered undetermined.

Related Topics

[Enabling UDLD Globally](#) , on page 161

[Enabling UDLD on an Interface](#) , on page 162

Aggressive Mode

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the ports cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the ports is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD disables the affected port.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to reestablish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode detects whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Related Topics

[Enabling UDLD Globally](#) , on page 161

[Enabling UDLD on an Interface](#) , on page 162

Methods to Detect Unidirectional Links

UDLD operates by using two methods:

- Neighbor database maintenance
- Event-driven detection and echoing

Related Topics

[Enabling UDLD Globally](#) , on page 161

[Enabling UDLD on an Interface](#) , on page 162

Neighbor Database Maintenance

UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active port to keep each device informed about its neighbors.

When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

Whenever a port is disabled and UDLD is running, whenever UDLD is disabled on a port, or whenever the switch is reset, UDLD clears all existing cache entries for the ports affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

Event-Driven Detection and Echoing

UDLD relies on echoing as its detection operation. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the port is disabled.

Related Topics

[Enabling UDLD Globally](#) , on page 161

[Enabling UDLD on an Interface](#) , on page 162

UDLD Reset Options

If an interface becomes disabled by UDLD, you can use one of the following options to reset UDLD:

- The **udld reset** interface configuration command.
- The **shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled port.
- The **no udld {aggressive | enable}** global configuration command followed by the **udld {aggressive | enable}** global configuration command reenables the disabled ports.
- The **no udld port** interface configuration command followed by the **udld port [aggressive]** interface configuration command reenables the disabled fiber-optic port.
- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval interval** global configuration command specifies the time to recover from the UDLD error-disabled state.

Related Topics

[Enabling UDLD Globally](#) , on page 161

[Enabling UDLD on an Interface](#) , on page 162

Default UDLD Configuration

Table 16: Default UDLD Configuration

| Feature | Default Setting |
|--|---|
| UDLD global enable state | Globally disabled |
| UDLD per-port enable state for fiber-optic media | Disabled on all Ethernet fiber-optic ports |
| UDLD per-port enable state for twisted-pair (copper) media | Disabled on all Ethernet 10/100 and 1000BASE-TX ports |

| Feature | Default Setting |
|----------------------|-----------------|
| UDLD aggressive mode | Disabled |

Related Topics

[Enabling UDLD Globally](#) , on page 161

[Enabling UDLD on an Interface](#) , on page 162

How to Configure UDLD

Enabling UDLD Globally

Follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic ports on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **udld {aggressive | enable | message time *message-timer-interval*}**
3. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | udld {aggressive enable message time <i>message-timer-interval</i>} Example: Switch(config)# udld enable message time 10 | Specifies the UDLD mode of operation: <ul style="list-style-type: none"> • aggressive—Enables UDLD in aggressive mode on all fiber-optic ports. • enable—Enables UDLD in normal mode on all fiber-optic ports on the switch. UDLD is disabled by default. An individual interface configuration overrides the setting of the udld enable global configuration command. • message time <i>message-timer-interval</i>—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are detected to be bidirectional. The range is from 1 to 90 seconds; the default value is 15. |

| | Command or Action | Purpose |
|---------------|---|---|
| | | <p>Note This command affects fiber-optic ports only. Use the udld interface configuration command to enable UDLD on other port types.</p> <p>Use the no form of this command, to disable UDLD.</p> |
| Step 3 | <p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre> | Returns to privileged EXEC mode. |

Related Topics

- [Monitoring and Maintaining UDLD Aggressive Mode, on page 159](#)
- [Normal Mode, on page 158](#)
- [Methods to Detect Unidirectional Links, on page 159](#)
- [Event-Driven Detection and Echoing, on page 160](#)
- [UDLD Reset Options, on page 160](#)
- [Default UDLD Configuration, on page 160](#)

Enabling UDLD on an Interface

Follow these steps either to enable UDLD in the aggressive or normal mode or to disable UDLD on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **udld port** [aggressive]
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1 | Specifies the port to be enabled for UDLD, and enters interface configuration mode. |
| Step 3 | udld port [aggressive] Example: Switch(config-if)# udld port aggressive | UDLD is disabled by default. <ul style="list-style-type: none"> • udld port—Enables UDLD in normal mode on the specified port. • udld port aggressive—(Optional) Enables UDLD in aggressive mode on the specified port. Note Use the no udld port interface configuration command to disable UDLD on a specified fiber-optic port. |
| Step 4 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |

Related Topics

[Monitoring and Maintaining UDLD](#)

[Aggressive Mode, on page 159](#)

[Normal Mode, on page 158](#)

[Methods to Detect Unidirectional Links, on page 159](#)

[Event-Driven Detection and Echoing, on page 160](#)

[UDLD Reset Options, on page 160](#)

[Default UDLD Configuration, on page 160](#)

Monitoring and Maintaining UDLD

| Command | Purpose |
|---|---|
| <code>show udld [interface-id neighbors]</code> | Displays the UDLD status for the specified port or for all ports. |

Additional References for UDLD

Related Documents

| Related Topic | Document Title |
|---------------------------|---|
| Layer 2 command reference | <i>Layer 2 Command Reference (Cisco WLC 5700 Series) Layer 2/3 Command Reference (Catalyst 3650 Switches)</i> |

Error Message Decoder

| Description | Link |
|---|---|
| To help you research and resolve system error messages in this release, use the Error Message Decoder tool. | https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi |

Standards and RFCs

| Standard/RFC | Title |
|--------------|-------|
| None | — |

MIBs

| MIB | MIBs Link |
|--------------------------------------|--|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|--|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p>http://www.cisco.com/support</p> |

Feature Information for UDLD

| Release | Modification |
|---------------------------------------|------------------------------|
| Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE | This feature was introduced. |



INDEX

A

- active link [139, 154](#)
- active links [136](#)
- addresses [10](#)
 - dynamic [10](#)
 - accelerated aging [10](#)
 - default aging [10](#)
 - multicast [10](#)
 - STP address management [10](#)
- aggregate-port learners [122](#)
- aging time [24, 59](#)
 - accelerated [24, 59](#)
 - for MSTP [59](#)
 - for STP [24](#)
- alternate [2](#)
 - port [2](#)
- automatic creation of [102, 105](#)

B

- BackboneFast [76, 87](#)
 - described [76](#)
 - enabling [87](#)
- backup [2](#)
 - port [2](#)
- backup interfaces [136](#)
 - See Flex Links [136](#)
- binding physical and logical interfaces [100](#)
- blocking [7](#)
 - state [7](#)
- BPDU [2, 3, 45, 71](#)
 - contents [3](#)
 - filtering [71](#)
 - RSTP format [45](#)
- bridge identifier (bridge ID) [5](#)
- bridge protocol data units [2](#)

C

- channel groups [100](#)
 - binding physical and logical interfaces [100](#)
 - numbering of [100](#)
- CIST regional root [36, 37](#)
 - See MSTP [36, 37](#)
- CIST root [37](#)
 - See MSTP [37](#)
- Configuration Examples for Configuring EtherChannels
- command [130](#)
- configuring [115](#)
 - Layer 2 interfaces [115](#)
 - on Layer 2 interfaces [115](#)
- Configuring LACP Hot-Standby Ports [131](#)
 - Example [131](#)
- Configuring Layer 2 EtherChannels [130](#)
 - Examples command [130](#)
- Configuring Layer 3 EtherChannels [131](#)
 - Examples command [131](#)
- cross-stack EtherChannel [96, 98, 112, 115, 117](#)
 - configuring [115](#)
 - on Layer 2 interfaces [115](#)
 - described [96](#)
 - illustration [96](#)
- cross-stack UplinkFast, STP [75, 76](#)
 - Fast Uplink Transition Protocol [75](#)
 - normal-convergence events [76](#)
- cross-stack UplinkFast, STP [73, 76](#)
 - described [73](#)
 - fast-convergence events [76](#)

D

- default configuration [13, 47, 111, 142, 160](#)
 - EtherChannel [111](#)
 - Flex Links [142](#)
 - MAC address-table move update [142](#)
 - MSTP [47](#)
 - STP [13](#)
 - UDLD [160](#)

- described [96, 102](#)
- designated [2](#)
 - port [2](#)
 - switch [2](#)
- destination-IP address-based forwarding [108](#)
- destination-IP address-based forwarding, EtherChannel [107](#)
- destination-MAC address forwarding [107](#)
- destination-MAC address forwarding, EtherChannel [107](#)
- detecting indirect link failures, STP [76](#)
- device [9](#)
 - root [9](#)
- device priority [22, 57](#)
 - MSTP [57](#)
 - STP [22](#)
- disabled [8](#)
 - state [8](#)
- dynamic addresses [10](#)
 - See addresses [10](#)

E

- EtherChannel [96, 100, 102, 103, 104, 105, 106, 107, 110, 111, 112, 115, 117, 119, 121, 122, 124, 125, 126, 127](#)
 - automatic creation of [102, 105](#)
 - channel groups [100](#)
 - binding physical and logical interfaces [100](#)
 - numbering of [100](#)
 - configuration guidelines [112](#)
 - configuring [115, 117](#)
 - Layer 2 interfaces [115](#)
 - Layer 3 [117](#)
 - default configuration [111](#)
 - extended load balancing [121](#)
 - forwarding methods [107, 119, 121](#)
 - IEEE 802.3ad, described [105](#)
 - interaction [112](#)
 - with STP [112](#)
 - LACP [105, 106, 124, 125, 126, 127](#)
 - hot-standby ports [124](#)
 - interaction with other features [106](#)
 - max bundles [124](#)
 - min links [125](#)
 - modes [105](#)
 - port priority [127](#)
 - system priority [126](#)
 - load balancing [107, 119](#)
 - logical interfaces, described [100](#)
 - PAgP [102, 103, 104, 122](#)
 - about aggregate-port learners [103](#)
 - about learn method and priority [103](#)
 - aggregate-port learners [122](#)
 - described [102](#)

- EtherChannel (*continued*)
 - PAgP (*continued*)
 - interaction with other features [104](#)
 - learn method and priority configuration [122](#)
 - modes [102](#)
 - port-channel interfaces [100](#)
 - numbering of [100](#)
 - stack changes, effects of [110](#)
- EtherChannel | interaction [112](#)
 - with VLANs [112](#)
- EtherChannel failover [100](#)
- EtherChannel guard [79, 89](#)
 - described [79](#)
 - enabling [89](#)
- EtherChannels [96, 115](#)
- extended load balancing [121](#)
- extended system ID [5, 17, 34](#)
 - MSTP [34](#)
 - STP [5, 17](#)

F

- fallback bridging [3, 12](#)
 - STP [3](#)
 - keepalive messages [3](#)
 - VLAN-bridge STP [12](#)
- Fast Uplink Transition Protocol [75](#)
- fiber-optic, detecting unidirectional links [158](#)
- Flex Links [136, 138, 142, 143, 144, 146, 149, 150, 151](#)
 - configuring [143, 144](#)
 - configuring VLAN load balancing [146](#)
 - default configuration [142](#)
 - description [136](#)
 - link load balancing [138](#)
 - monitoring [149](#)
 - preemption scheme [144](#)
 - preferred VLAN example [151](#)
 - switchport backup example [150](#)
 - forced preemption mode example [150](#)
 - VLAN load balancing examples [151](#)
- Flex Links failover [138](#)
- forward-delay time [24, 59](#)
 - MSTP [59](#)
 - STP [24](#)
- forwarding [8](#)
 - state [8](#)
- forwarding methods [107, 119, 121](#)

G

- general query [153](#)

Generating IGMP Reports [139](#)

H

hello time [24, 58](#)
 MSTP [58](#)
 STP [24](#)
 hot-standby ports [124](#)

I

IEEE 802.1s [33](#)
 See MSTP [33](#)
 IEEE 802.3ad [105](#)
 See EtherChannel [105](#)
 IEEE 802.3ad, described [105](#)
 interaction with other features [104, 106](#)

K

keepalive messages [3](#)

L

LACP [97, 105, 106, 115, 124, 125, 126, 127](#)
 hot-standby ports [124](#)
 interaction with other features [106](#)
 max bundles [124](#)
 min links [125](#)
 modes [105](#)
 port priority [127](#)
 system priority [126](#)
 Layer 2 EtherChannel configuration guidelines [114](#)
 Layer 2 interfaces [115](#)
 Layer 3 EtherChannel configuration guidelines [115](#)
 Leaking IGMP Reports [139](#)
 learn method and priority configuration [122](#)
 Link Failure, detecting unidirectional [41](#)
 link redundancy [136](#)
 See Flex Links [136](#)
 listening [8](#)
 state [8](#)
 load balancing [107, 119](#)
 load balancing advantages [108](#)
 logical interfaces, described [100](#)

M

MAC address-table move update [139, 142, 147, 148](#)
 configuration guidelines [142](#)
 configuring [147](#)
 default configuration [142](#)
 description [139](#)
 obtain and process messages [148](#)
 max bundles [124](#)
 maximum aging time [25, 60](#)
 MSTP [60](#)
 STP [25](#)
 maximum hop count, MSTP [61](#)
 min links [125](#)
 modes [102, 105](#)
 monitoring [149](#)
 Flex Links [149](#)
 mrouter Port [139](#)
 MSTP [11, 12, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 47, 48, 49, 51, 52, 53, 55, 57, 58, 59, 60, 61, 62, 64, 65, 66, 70, 71, 79, 80, 82, 83, 89, 90, 91](#)
 configuring [48, 51, 52, 53, 55, 57, 58, 59, 60, 61, 62, 64](#)
 neighbor type [64](#)
 boundary ports [32, 39](#)
 configuration guidelines [32](#)
 described [39](#)
 BPDU filtering [71, 83](#)
 described [71](#)
 enabling [83](#)
 BPDU guard [70, 82](#)
 described [70](#)
 enabling [82](#)
 CIST regional root [36, 37](#)
 CIST root [37](#)
 CIST, described [36](#)
 configuration guidelines [33](#)
 configuring [48, 51, 52, 53, 55, 57, 58, 59, 60, 61, 62, 64](#)
 device priority [57](#)
 forward-delay time [59](#)
 hello time [58](#)
 link type for rapid convergence [62](#)
 maximum aging time [60](#)
 maximum hop count [61](#)
 MST region [48](#)
 path cost [55](#)
 port priority [53](#)
 root device [51](#)
 secondary root device [52](#)
 CST [36](#)
 operations between regions [36](#)
 default configuration [47](#)
 displaying status [66](#)
 enabling the mode [48](#)

MSTP (*continued*)

- EtherChannel guard [79, 89](#)
 - described [79](#)
 - enabling [89](#)
 - extended system ID [34, 52](#)
 - effects on root device [34](#)
 - effects on secondary root device [52](#)
 - unexpected behavior [34](#)
 - IEEE 802.1s [37, 39, 40](#)
 - port role naming change [40](#)
 - implementation [39](#)
 - terminology [37](#)
 - instances supported [11](#)
 - interface state, blocking to forwarding [70](#)
 - interoperability and compatibility among modes [12, 32](#)
 - interoperability with IEEE 802.1D [41, 65](#)
 - described [41](#)
 - restarting migration process [65](#)
 - IST [36](#)
 - operations within a region [36](#)
 - loop guard [80, 91](#)
 - described [80](#)
 - enabling [91](#)
 - mapping VLANs to MST instance [49](#)
 - MST region [35, 36, 38, 48](#)
 - CIST [36](#)
 - configuring [48](#)
 - described [35](#)
 - hop-count mechanism [38](#)
 - IST [35](#)
 - supported spanning-tree instances [35](#)
 - PortFast [70, 80](#)
 - described [70](#)
 - enabling [80](#)
 - preventing root switch selection [79](#)
 - root device [34](#)
 - configuring [34](#)
 - effects of extended system ID [34](#)
 - unexpected behavior [34](#)
 - root guard [79, 90](#)
 - described [79](#)
 - enabling [90](#)
 - shutdown Port Fast-enabled port [70](#)
 - stack changes, effects of [41](#)
 - status, displaying [66](#)
- Multicast Fast Convergence [138, 152](#)

N

numbering of [100](#)

O

on Layer 2 interfaces [115](#)

P

- PaGP [97](#)
- PAgP [102, 104, 115, 122](#)
 - aggregate-port learners [122](#)
 - described [102](#)
 - interaction with other features [104](#)
 - learn method and priority configuration [122](#)
 - modes [102](#)
 - See EtherChannel [102](#)
- path cost [2, 21, 55](#)
 - MSTP [55](#)
 - STP [21](#)
- port [2, 9](#)
 - priority [2](#)
 - root [9](#)
- Port Aggregation Protocol [102](#)
 - See EtherChannel [102](#)
- port priority [19, 53, 127](#)
 - MSTP [53](#)
 - STP [19](#)
- port-channel interfaces [100](#)
 - numbering of [100](#)
- preemption delay, default configuration [142](#)
- preemption, default configuration [142](#)
- proxy reports [139](#)
- PVST+ [11, 12](#)
 - described [11](#)
 - IEEE 802.1Q trunking interoperability [12](#)
 - instances supported [11](#)

R

- rapid convergence [43](#)
- Rapid Spanning Tree Protocol [33](#)
 - See RSTP [33](#)
- redundancy [9, 73, 96](#)
 - EtherChannel [96](#)
 - STP [9, 73](#)
 - backbone [9](#)
 - multidrop backbone [73](#)
- redundant links and UplinkFast [85, 86](#)
- reference [40](#)
- restrictions [1, 32, 69](#)
 - MSTP [32](#)
 - Optional Spanning-Tree Features [69](#)
 - STP [1](#)

- role [2](#)
 - port [2](#)
 - root [2, 3](#)
 - port [2](#)
 - switch [2, 3](#)
 - root device [17, 51](#)
 - MSTP [51](#)
 - STP [17](#)
 - RSTP [41, 42, 43, 44, 45, 46, 62, 65](#)
 - active topology [42](#)
 - BPDU [45, 46](#)
 - format [45](#)
 - processing [46](#)
 - designated port, defined [42](#)
 - designated switch, defined [42](#)
 - interoperability with IEEE 802.1D [41, 46, 65](#)
 - described [41](#)
 - restarting migration process [65](#)
 - topology changes [46](#)
 - overview [42](#)
 - port roles [42, 44](#)
 - described [42](#)
 - synchronized [44](#)
 - rapid convergence [43, 44, 62](#)
 - cross-stack rapid convergence [44](#)
 - described [43](#)
 - edge ports and Port Fast [43](#)
 - point-to-point links [43, 62](#)
 - root ports [43](#)
 - root port, defined [42](#)
- ## S
- See EtherChannel [102, 105](#)
 - service-provider network, MSTP and RSTP [33](#)
 - show interfaces switchport [152](#)
 - single-switch EtherChannel [98](#)
 - source-and-destination MAC address forwarding, EtherChannel [107](#)
 - source-and-destination-IP address based forwarding, EtherChannel [107](#)
 - source-IP address based forwarding, EtherChannel [107](#)
 - source-IP address-based forwarding [108](#)
 - source-MAC address forwarding [107](#)
 - source-MAC address forwarding, EtherChannel [107](#)
 - Spanning Tree [6](#)
 - states [6](#)
 - spanning-tree [2](#)
 - port priority [2](#)
 - stack changes, effects of [110](#)
 - stack changes, effects on [13, 110, 112](#)
 - cross-stack EtherChannel [112](#)
 - stack changes, effects on (*continued*)
 - EtherChannel [110](#)
 - STP [13](#)
 - stack changes, effects on [41](#)
 - MSTP [41](#)
 - stacks, [3, 11](#)
 - MSTP instances supported [11](#)
 - STP [3](#)
 - bridge ID [3](#)
 - root port selection [3](#)
 - switch [3, 11](#)
 - STP [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 21, 22, 24, 25, 26, 28, 32, 72, 73, 76, 79, 85, 86, 87, 89](#)
 - accelerating root port selection [72](#)
 - BackboneFast [76, 87](#)
 - described [76](#)
 - enabling [87](#)
 - BPDU message exchange [3](#)
 - configuring [14, 17, 18, 19, 21, 22, 24, 25, 26](#)
 - device priority [22](#)
 - forward-delay time [24](#)
 - hello time [24](#)
 - maximum aging time [25](#)
 - path cost [21](#)
 - port priority [19](#)
 - root device [17](#)
 - secondary root device [18](#)
 - spanning-tree mode [14](#)
 - transmit hold-count [26](#)
 - cross-stack UplinkFast [73](#)
 - described [73](#)
 - default configuration [13](#)
 - designated, defined [4](#)
 - switch [4](#)
 - designated port, defined [4](#)
 - detecting indirect link failures [76](#)
 - disabling [16](#)
 - displaying status [28](#)
 - EtherChannel guard [79, 89](#)
 - described [79](#)
 - enabling [89](#)
 - extended system ID [1, 5, 17, 18](#)
 - effects on root device [17](#)
 - effects on the secondary root device [18](#)
 - overview [5](#)
 - unexpected behavior [1](#)
 - IEEE 802.1D and bridge ID [5](#)
 - IEEE 802.1D and multicast addresses [10](#)
 - IEEE 802.1t and VLAN identifier [5](#)
 - instances supported [11](#)
 - interface states [6, 7, 8](#)
 - blocking [7](#)
 - disabled [8](#)
 - forwarding [7, 8](#)

STP (*continued*)

- interface states (*continued*)
 - learning [8](#)
 - listening [8](#)
- interoperability and compatibility among modes [12, 32](#)
- keepalive messages [3](#)
- limitations with IEEE 802.1Q trunks [12](#)
- modes supported [11](#)
- overview [2](#)
- protocols supported [11](#)
- redundant connectivity [9](#)
- root [1, 3](#)
 - election [3](#)
 - switch [1, 3](#)
 - unexpected behavior [1](#)
- root device [5, 17](#)
 - configuring [5](#)
 - effects of extended system ID [5, 17](#)
- root port selection on a stack [3](#)
- root port, defined [3](#)
- stack changes, effects of [13](#)
- status, displaying [28](#)
- UplinkFast [72, 85, 86](#)
 - described [72](#)
 - disabling [86](#)
 - enabling [85](#)
- VLAN-bridge [12](#)
- switchport backup interface [153](#)
- system priority [126](#)

T

- twisted-pair, detecting unidirectional links [158](#)

U

- UDLD [157, 158, 159, 160, 161, 162](#)
 - aggressive [158, 159](#)

UDLD (*continued*)

- aggressive mode [161](#)
 - message time [161](#)
 - default configuration [160](#)
 - disabling [162](#)
 - per interface [162](#)
 - echoing detection mechanism [159, 160](#)
 - enabling [161, 162](#)
 - globally [161](#)
 - per interface [162](#)
 - fiber-optic links [159](#)
 - neighbor database [159](#)
 - neighbor database maintenance [159](#)
 - normal [158](#)
 - normal mode [158](#)
 - overview [158](#)
 - restrictions [157](#)
 - twisted-pair links [159](#)
- UplinkFast [72, 85, 86](#)
- described [72](#)
 - disabling [86](#)
 - enabling [85](#)

V

- VLAN load balancing on Flex Links [138, 142](#)
 - configuration guidelines [142](#)
 - described [138](#)
- VLANs [10, 12](#)
 - aging dynamic addresses [10](#)
 - STP and IEEE 802.1Q trunks [12](#)
 - VLAN-bridge STP [12](#)

W

- with STP [112](#)