



## Configuring Optional Spanning-Tree Features

This chapter describes how to configure optional spanning-tree features. You can configure all of these features when your Catalyst 2940 switch is running the per-VLAN spanning tree plus (PVST+) mode.

For information on configuring the Spanning Tree Protocol (STP), see [Chapter 11, “Configuring STP.”](#)



### Note

For complete syntax and usage information for the commands used in this chapter, refer to the command reference for this release.

This chapter consists of these sections:

- [Understanding Optional Spanning-Tree Features, page 12-1](#)
- [Configuring Optional Spanning-Tree Features, page 12-9](#)
- [Displaying the Spanning-Tree Status, page 12-16](#)

## Understanding Optional Spanning-Tree Features

These sections describe how the optional spanning-tree features work:

- [Understanding Port Fast, page 12-1](#)
- [Understanding BPDU Guard, page 12-2](#)
- [Understanding BPDU Filtering, page 12-3](#)
- [Understanding UplinkFast, page 12-3](#)
- [Understanding BackboneFast, page 12-5](#)
- [Understanding EtherChannel Guard, page 12-7](#)
- [Understanding Root Guard, page 12-8](#)
- [Understanding Loop Guard, page 12-9](#)

### Understanding Port Fast

Port Fast 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 Port Fast on ports connected to a single workstation or server, as shown in [Figure 12-1](#), to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

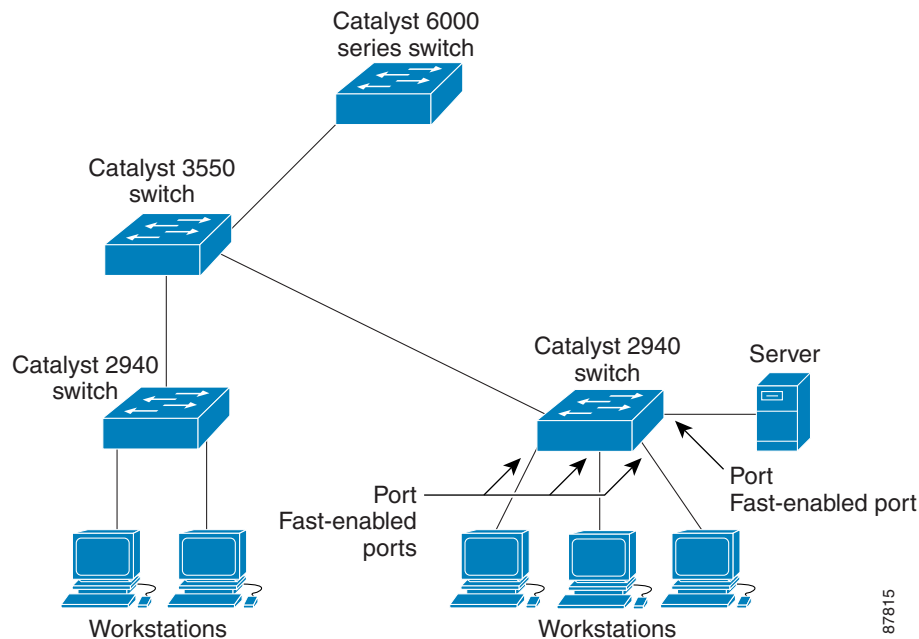
Ports connected to a single workstation or server should not receive bridge protocol data units (BPDUs). A port with Port Fast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

**Note**

Because the purpose of Port Fast is to minimize the time ports must wait for spanning-tree to converge, it is effective only when used on ports connected to end stations. If you enable Port Fast on a port connecting to another switch, you risk creating a spanning-tree loop.

If your switch is running PVST+, you can enable this feature by using the **spanning-tree portfast** interface configuration or the **spanning-tree portfast default** global configuration command.

**Figure 12-1 Port Fast-Enabled Ports**



## Understanding BPDU Guard

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

At the global level, you can enable BPDU guard on Port Fast-enabled ports by using the **spanning-tree portfast bpduguard default** global configuration command. Spanning tree shuts down ports that are in a Port Fast-operational state. In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state.

At the interface level, you can enable BPDU guard on any port by using the **spanning-tree bpduguard enable** interface configuration command without also enabling the Port Fast feature. When 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 port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

If your switch is running PVST+, you can enable the BPDU guard feature for the entire switch or for an interface.

## Understanding 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.

At the global level, you can enable BPDU filtering on Port Fast-enabled ports by using the **spanning-tree portfast bpdupfilter default** global configuration command. This command prevents ports that are in a Port Fast-operational state from sending or receiving BPDUs. The ports 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 ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled port, the port loses its Port Fast-operational status, and BPDU filtering is disabled.

At the interface level, you can enable BPDU filtering on any port without also enabling the Port Fast feature by using the **spanning-tree bpdupfilter enable** interface configuration command. This command prevents the port 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.

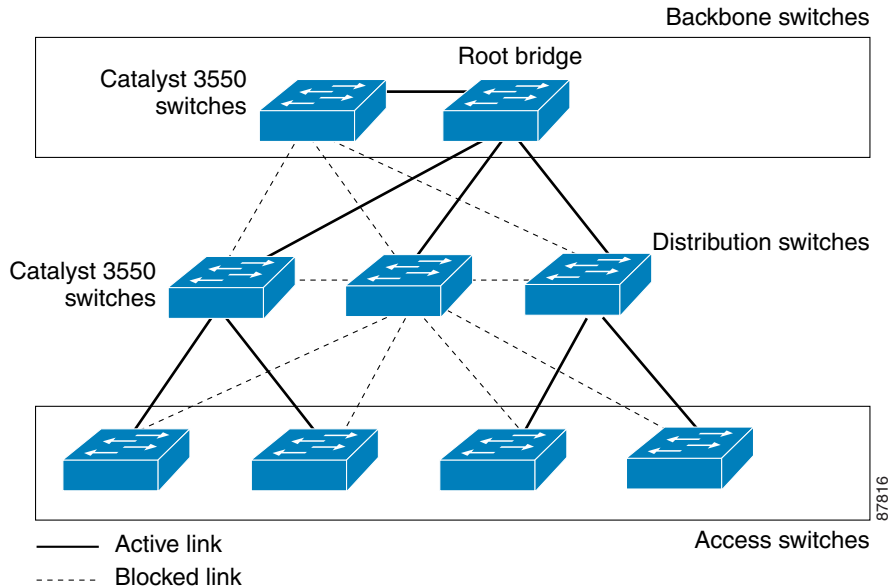
---

If your switch is running PVST+, you can enable the BPDU filtering feature for the entire switch or for an interface.

## Understanding UplinkFast

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. [Figure 12-2](#) shows a complex network where distribution switches and access switches each have at least one redundant link that spanning tree blocks to prevent loops.

Figure 12-2 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. By enabling UplinkFast with the **spanning-tree uplinkfast** global configuration command, you can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself. 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. The UplinkFast feature is supported only when the switch is running PVST+.

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.

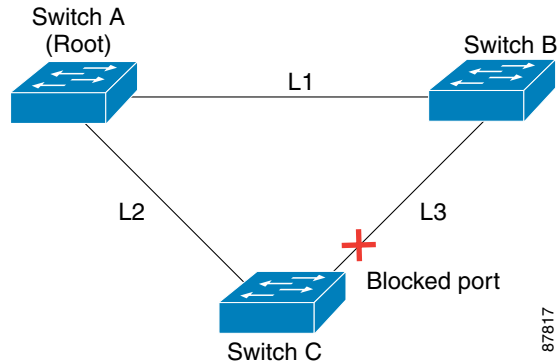
**Note**

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.

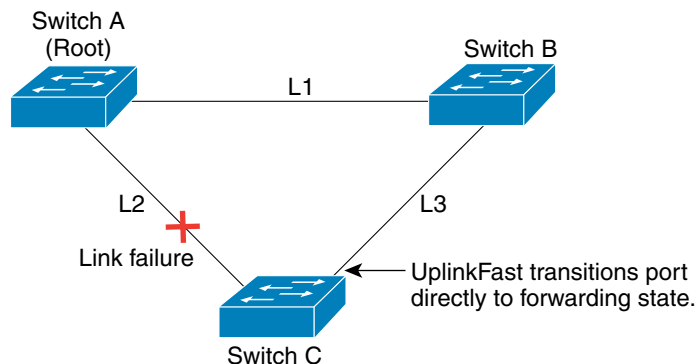
Figure 12-3 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in a blocking state.

Figure 12-3 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 port on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 12-4. This change takes approximately 1 to 5 seconds.

Figure 12-4 UplinkFast Example After Direct Link Failure



## Understanding 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 determines 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, which is enabled by using the **spanning-tree backbonefast** global configuration command, starts when a root port or blocked port 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 bridge has lost its connection to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time specified by the **vlan vlan-id** global configuration command.

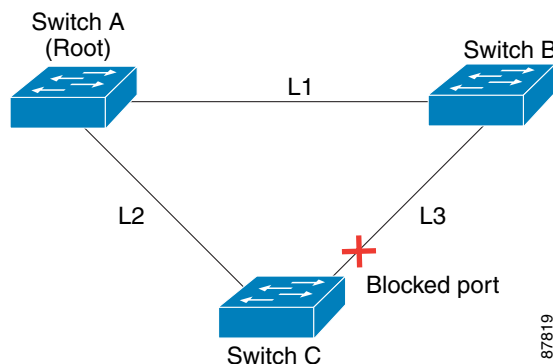
The switch tries to determine if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked port, the root port and other blocked ports on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked ports become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked ports, 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 RLS request on all alternate paths to the root switch and waits for an RLQ reply from other switches in the network.

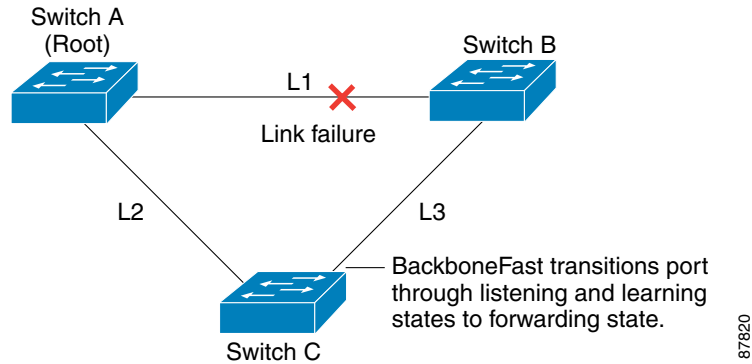
If the switch determines that it still has an alternate path to the root, it expires the maximum aging time on the port that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the port that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all ports 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.

Figure 12-5 shows 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.

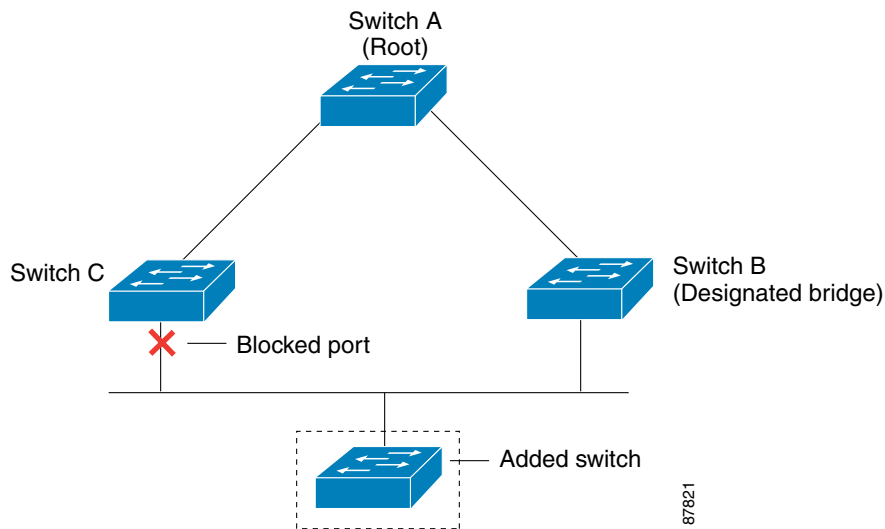
**Figure 12-5 BackboneFast Example Before Indirect Link Failure**



If link L1 fails as shown in Figure 12-6, 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 port on Switch C to move immediately to the listening state without waiting for the maximum aging time for the port 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. This switchover takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. Figure 12-6 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

**Figure 12-6 BackboneFast Example After Indirect Link Failure**

If a new switch is introduced into a shared-medium topology as shown in [Figure 12-7](#), BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated bridge (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 bridge to Switch A, the root switch.

**Figure 12-7 Adding a Switch in a Shared-Medium Topology**

## Understanding 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. For EtherChannel configuration guidelines, see the [“EtherChannel Configuration Guidelines”](#) section on page 25-8.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and this error message appears:

```
PM-4-ERR_DISABLE: Channel-misconfig error detected on [chars], putting [chars] in
err-disable state.
```

If your switch is running PVST+, you can enable this feature by using the **spanning-tree etherchannel guard misconfig** global configuration command.

## Understanding 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, as shown in [Figure 12-8](#). 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.

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.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs.

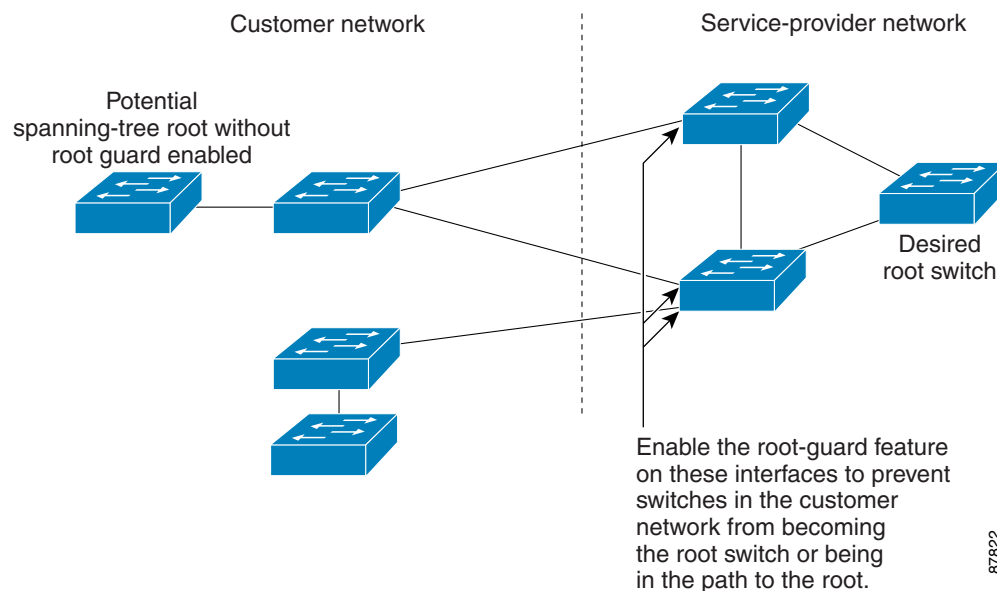
If your switch is running PVST+, you can enable this feature by using the **spanning-tree guard root** interface configuration command.



### Caution

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

**Figure 12-8 Root Guard in a Service-Provider Network**



87822

## Understanding 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.

If your switch is running PVST+, you can enable this feature by using the **spanning-tree loopguard default** global configuration command.

When the switch is operating in 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.

## Configuring Optional Spanning-Tree Features

These sections describe how to configure optional spanning-tree features:

- [Default Optional Spanning-Tree Configuration, page 12-9](#)
- [Optional Spanning-Tree Configuration Guidelines, page 12-10](#)
- [Enabling Port Fast \(Optional\), page 12-10](#)
- [Enabling BPDU Guard \(Optional\), page 12-11](#)
- [Enabling BPDU Filtering \(Optional\), page 12-11](#)
- [Enabling UplinkFast for Use with Redundant Links \(Optional\), page 12-12](#)
- [Enabling BackboneFast \(Optional\), page 12-13](#)
- [Enabling EtherChannel Guard \(Optional\), page 12-14](#)
- [Enabling Root Guard \(Optional\), page 12-14](#)
- [Enabling Loop Guard \(Optional\), page 12-15](#)

## Default Optional Spanning-Tree Configuration

[Table 12-1](#) shows the default optional spanning-tree configuration.

**Table 12-1** Default Optional Spanning-Tree Configuration

Feature	Default Setting
Port Fast, BPDU filtering, BPDU guard	Globally disabled (unless they are individually configured per interface)
UplinkFast	Globally disabled
BackboneFast	Globally disabled
EtherChannel guard	Globally enabled
Root guard	Disabled on all interfaces
Loop guard	Disabled on all interfaces

## Optional Spanning-Tree Configuration Guidelines

The UplinkFast, BackboneFast, and cross-stack UplinkFast features are not supported with the rapid PVST+ or the MSTP.

### Enabling Port Fast (Optional)

A port with the Port Fast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.




#### Caution

Use Port Fast *only* when connecting a single end station to an access or trunk port. Enabling this feature on a port 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.

If you enable the voice VLAN feature, the Port Fast feature is automatically enabled. When you disable voice VLAN, the Port Fast feature is not automatically disabled. For more information, see [Chapter 15, “Configuring Voice VLAN.”](#)

Beginning in privileged EXEC mode, follow these steps to enable Port Fast. This procedure is optional:

	Command	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>interface interface-id</code>	Enter interface configuration mode, and specify an interface to configure.
Step 3	<code>spanning-tree portfast [trunk]</code>	<p>Enable Port Fast on an access port connected to a single workstation or server. By specifying the <b>trunk</b> keyword, you can enable Port Fast on a trunk port.</p> <p> <b>Caution</b> Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable Port Fast on a trunk port.</p> <p>By default, Port Fast is disabled on all ports.</p>
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show spanning-tree interface interface-id portfast</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.



#### Note

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

To disable the Port Fast feature, use the **spanning-tree portfast disable** interface configuration command.

## Enabling BPDU Guard (Optional)

When you globally enable BPDU guard on ports that are Port Fast-enabled (the ports are in a Port Fast-operational state), spanning tree shuts down Port Fast-enabled ports that receive BPDUs.

In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.



### Caution

Configure Port Fast 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.

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

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU guard feature. This procedure is optional:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>spanning-tree portfast bpduguard default</b>	Globally enable BPDU guard. By default, BPDU guard is disabled.
Step 3	<b>interface <i>interface-id</i></b>	Enter interface configuration mode, and specify the interface connected to an end station.
Step 4	<b>spanning-tree portfast</b>	Enable the Port Fast feature.
Step 5	<b>end</b>	Return to privileged EXEC mode.
Step 6	<b>show running-config</b>	Verify your entries.
Step 7	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

To disable BPDU guard, use the **no spanning-tree portfast bpduguard default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpduguard default** global configuration command by using the **spanning-tree bpduguard enable** interface configuration command.

## Enabling BPDU Filtering (Optional)

When you globally enable BPDU filtering on Port Fast-enabled ports, it prevents ports that are in a Port Fast-operational state from sending or receiving BPDUs. The ports 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 ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled port, the port loses its Port Fast-operational status, and BPDU filtering is disabled.

**Caution**

Configure Port Fast 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.

You can also use the **spanning-tree bpdudfilter enable** interface configuration command to enable BPDU filtering on any port without also enabling the Port Fast feature. This command prevents the port 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.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU filtering feature. This procedure is optional:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>spanning-tree portfast bpdudfilter default</b>	Globally enable BPDU filtering. By default, BPDU filtering is disabled.
Step 3	<b>interface</b> <i>interface-id</i>	Enter interface configuration mode, and specify the interface connected to an end station.
Step 4	<b>spanning-tree portfast</b>	Enable the Port Fast feature.
Step 5	<b>end</b>	Return to privileged EXEC mode.
Step 6	<b>show running-config</b>	Verify your entries.
Step 7	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

To disable BPDU filtering, use the **no spanning-tree portfast bpdudfilter default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpdudfilter default** global configuration command by using the **spanning-tree bpdudfilter enable** interface configuration command.

## Enabling UplinkFast for Use with Redundant Links (Optional)

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

**Note**

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

Beginning in privileged EXEC mode, follow these steps to enable UplinkFast. This procedure is optional:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>spanning-tree uplinkfast</b> [ <b>max-update-rate</b> <i>pkts-per-second</i> ]	Enable 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.
Step 3	<b>end</b>	Return to privileged EXEC mode.
Step 4	<b>show spanning-tree summary</b>	Verify your entries.
Step 5	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

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 reduces the chance that the 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.

To return the update packet rate to the default setting, use the **no spanning-tree uplinkfast max-update-rate** global configuration command. To disable UplinkFast, use the **no spanning-tree uplinkfast** command.

## Enabling BackboneFast (Optional)

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



### Note

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.

Beginning in privileged EXEC mode, follow these steps to enable BackboneFast. This procedure is optional:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>spanning-tree backbonefast</b>	Enable BackboneFast.
Step 3	<b>end</b>	Return to privileged EXEC mode.
Step 4	<b>show spanning-tree summary</b>	Verify your entries.
Step 5	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

To disable the BackboneFast feature, use the **no spanning-tree backbonefast** global configuration command.

## Enabling EtherChannel Guard (Optional)

You can enable EtherChannel guard to detect an EtherChannel misconfiguration that causes a loop.

Beginning in privileged EXEC mode, follow these steps to enable EtherChannel guard. This procedure is optional:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>spanning-tree etherchannel guard misconfig</b>	Enable EtherChannel guard.
Step 3	<b>end</b>	Return to privileged EXEC mode.
Step 4	<b>show spanning-tree summary</b>	Verify your entries.
Step 5	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

To disable the EtherChannel guard feature, use the **no spanning-tree etherchannel guard misconfig** global configuration command.

You can use the **show interfaces status err-disabled** privileged EXEC command to determine 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** commands on the port-channel interfaces that were misconfigured.

## Enabling Root Guard (Optional)

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.

Beginning in privileged EXEC mode, follow these steps to enable root guard on an interface. This procedure is optional:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>interface <i>interface-id</i></b>	Enter interface configuration mode, and specify an interface to configure.

	Command	Purpose
Step 3	<code>spanning-tree guard root</code>	Enable root guard on the interface. By default, root guard is disabled on all interfaces.
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

To disable root guard, use the **no spanning-tree guard** interface configuration command.

## Enabling Loop Guard (Optional)

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 ports that are considered point-to-point by the spanning tree.



### Note

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

Beginning in privileged EXEC mode, follow these steps to enable loop guard. This procedure is optional:

	Command	Purpose
Step 1	<code>show spanning-tree active</code> or <code>show spanning-tree mst</code>	Determine which ports are alternate or root ports.
Step 2	<code>configure terminal</code>	Enter global configuration mode.
Step 3	<code>spanning-tree loopguard default</code>	Enable loop guard. By default, loop guard is disabled.
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

To globally disable loop guard, use the **no spanning-tree loopguard default** global configuration command. You can override the setting of the **no spanning-tree loopguard default** global configuration command by using the **spanning-tree guard loop** interface configuration command.

## Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in [Table 12-2](#):

**Table 12-2** *Commands for Displaying the Spanning-Tree Status*

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

You can clear spanning-tree counters by using the `clear spanning-tree [interface interface-id]` privileged EXEC command.

For information about other keywords for the `show spanning-tree` privileged EXEC command, refer to the command reference for this release.