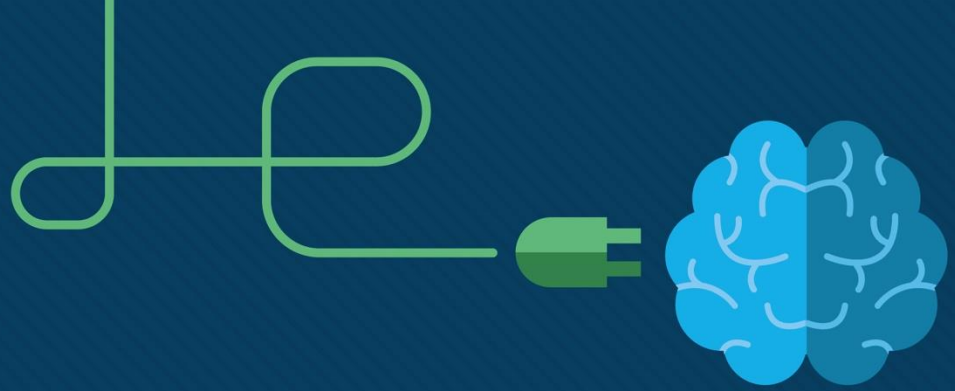

# Chapter 3: Advanced STP Tuning

## Instructor Materials

CCNP Enterprise: Core Networking

# Chapter 3 Content

**This chapter covers the following content:**

- **STP Topology Tuning -** This section explains some of the options for modifying the root bridge location or moving blocking ports to designated ports.
- **Additional STP Protection Mechanisms -** This section examines protection mechanisms such as root guard, BPDU guard, and STP loop guard.

# STP Topology Tuning

- In a properly designed network a switch is deliberately selected to become the root bridge and the designated and alternate ports are modified.
- Network design considerations factor in hardware platform, resiliency, and network topology.

# Root Bridge Placement

To ensure root bridge placement set the system priority on:

- The root bridge to the lowest value
- The secondary root bridge to a value slightly higher than that of the root bridge
- All other switches to a value higher than the secondary root bridge

| Command | Description |
|---|---|
| **spanning-tree vlan** *vlan-id* **priority** *priority* | The priority is a value between 0 and 61,440, in increments of 4,096. |
| **spanning-tree vlan** *vlan-id* **root {primary \| secondary} [diameter** *diameter***]** | The **primary** keyword sets the priority to 24,576, and the **secondary** keyword sets the priority to 28,672. The optional **diameter** command makes it possible to tune the Spanning Tree Protocol (STP) convergence and modifies the timers. |

# Configuring the Root Bridge

In the example:

- The initial priority for VLAN 1 on SW1 is verified, 32,769.
- SW1 is configured to be the primary root for VLAN 1
- The priority is verified again to ensure the change took place.

**Example 3-1** *Changing the STP System Priority on SW1*

```
! Verification of SW1 Priority before modifying the priority
SW1# show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol rstp
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              This bridge is the root
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0062.ec9d.c500
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  300 sec
```

```
! Configuring the SW1 priority as primary root for VLAN 1
SW1(config)# spanning-tree vlan 1 root primary
```

```
! Verification of SW1 Priority after modifying the priority
SW1# show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol rstp
  Root ID     Priority    24577
              Address     0062.ec9d.c500
              This bridge is the root
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    24577  (priority 24576 sys-id-ext 1)
              Address     0062.ec9d.c500
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  300 sec

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/2          Desg FWD 4         128.2    P2p
Gi1/0/3          Desg FWD 4         128.3    P2p
Gi1/0/14         Desg FWD 4         128.14   P2p
```

# Configuring the Backup Root Bridge

In the example:

- The initial priority for VLAN 1 on SW2 is verified, 32,769.
- SW2 is configured to be the secondary root for VLAN 1
- The priority is verified again to ensure the change took place.

**Example 3-2** *Changing the STP System Priority on SW2*

```
! Verification of SW2 Priority before modifying the priority
SW2# show spanning-tree vlan 1
! Output omitted for brevity

VLAN0001
  Spanning tree enabled protocol rstp
  Root ID     Priority    24577
              Address     0062.ec9d.c500
              Cost        4
              Port        1 (GigabitEthernet1/0/1)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0081.c4ff.8b00
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/1             Root FWD 4         128.1    P2p
Gi1/0/3             Desg FWD 4         128.3    P2p
Gi1/0/4             Desg FWD 4         128.4    P2p
```

```
! Configuring the SW2 priority as root secondary for VLAN 1
SW2(config)# spanning-tree vlan 1 root secondary
```

# Modifying STP Root Port & Blocked Switch Port Locations

Calculating total path cost to the root bridge:

- SW1 sends a BPDU to SW3 with the path cost of 0.
- SW3 receives the BPDU and adds its root port cost (4) to cost from the BPDU (0), resulting in the cost of 4.
- SW3 sends a BPDU to SW5 with the path cost of 4.
- SW5 receives the BPDU and adds its root port cost (4) to the cost from the BPDU (4), resulting in the cost of 8 for SW5 to reach the root bridge.

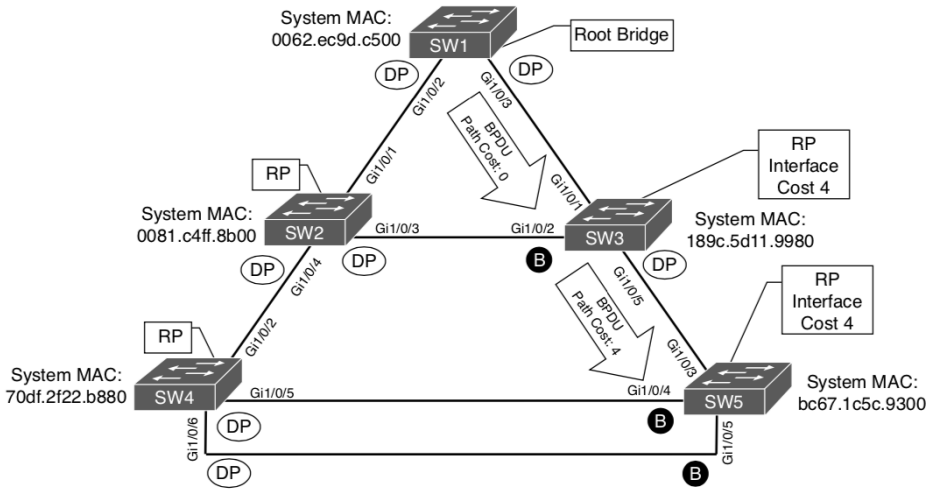


**Figure 3-2** *STP Path Cost Calculation*

# Verifying the Total Path Cost

The example highlights the total path cost to the root bridge from SW3 and SW5.

**Example 3-3** *Verifying the Total Path Cost*

```
SW1# show spanning-tree vlan 1
! Output omitted for brevity
VLAN0001
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              This bridge is the root
..
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/2             Desg FWD 4         128.2    P2p
Gi1/0/3             Desg FWD 4         128.3    P2p
```

```
SW3# show spanning-tree vlan 1
! Output omitted for brevity
VLAN0001
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              Cost        4
              Port        1 (GigabitEthernet1/0/1)
..
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/1             Root FWD 4         128.1    P2p
Gi1/0/2             Altn BLK 4         128.2    P2p
Gi1/0/5             Desg FWD 4         128.5    P2p
```

```
SW5# show spanning-tree vlan 1
! Output omitted for brevity
VLAN0001
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              Cost        8
              Port        3 (GigabitEthernet1/0/3)
..
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/3             Root FWD 4         128.3    P2p
Gi1/0/4             Altn BLK 4         128.4    P2p
Gi1/0/5             Altn BLK 4         128.5    P2p
```

**Note**: There is not a total path cost in SW1's output

# Modifying STP Port Cost

- The **spanning tree** [**vlan** *vlan-id]* **cost** *cost* command can be used to modify the STP forwarding path.
- Using the spanning tree command will modify the cost for all VLANs unless the optional **vlan** keyword is used.

**Example 3-4** *Modifying STP Port Cost*

```
SW3# conf t
SW3(config)# interface gi1/0/1
SW3(config-if)# spanning-tree cost 1
```

```
SW3# show spanning-tree vlan 1
! Output omitted for brevity
VLAN0001
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              Cost        1
              Port        1 (GigabitEthernet1/0/1)

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     189c.5d11.9980
..
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/1          Root FWD 1         128.1    P2p
Gi1/0/2          Desg FWD 4         128.2    P2p
Gi1/0/5          Desg FWD 4         128.5    P2p
```

```
SW2# show spanning-tree vlan 1
! Output omitted for brevity
VLAN0001
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              Cost        4
              Port        1 (GigabitEthernet1/0/1)

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0081.c4ff.8b00
..
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/1          Root FWD 4         128.1    P2p
Gi1/0/3          Altn BLK 4         128.3    P2p
Gi1/0/4          Desg FWD 4         128.4    P2p
```

# Modifying STP Port Priority

STP port priority influences which port becomes the alternate port when multiple links are used between switches. Use the command **spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority* to change the STP port priority on a switch's interface.

**Example 3-5** *Viewing STP Port Priority*

```
SW6# show spanning-tree vlan 1
! Output omitted for brevity
VLAN0001
  Spanning tree enabled protocol rstp
  Root ID     Priority    32769
              Address     0062.ec9d.c500
              Cost        12
              Port        4 (GigabitEthernet1/0/4)

  Bridge ID   Priority    32769  (priority 32768 sys-id-ext 1)
              Address     bc67.1c5c.9300
..
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/4          Root FWD 4         128.4    P2p
Gi1/0/5          Altn BLK 4         128.5    P2p
```

**Example 3-6** *Verifying Port Priority Impact on an STP Topology*

```
SW4# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW4(config)# interface gi1/0/6
SW4(config-if)# spanning-tree port-priority 64
```

**Example 3-7** *Determining the Impact of Port Priority on a Topology*

```
SW4# show spanning-tree vlan 1
! Output omitted for brevity
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/2          Root FWD 4         128.2    P2p
Gi1/0/5          Desg FWD 4         128.5    P2p
Gi1/0/6          Desg FWD 4         64.6     P2p

SW6# show spanning-tree vlan 1
! Output omitted for brevity
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/4          Altn BLK 4         128.4    P2p
Gi1/0/5          Root FWD 4         128.5    P2p
```

# Additional STP Protection Mechanisms

- A network forwarding loop occurs when there are multiple active paths between two devices. Broadcast and multicast traffic are forwarded out every switch port continuing the forwarding loop.
- The network's throughput is drastically effected as the switches are processing numerous frames. The switches CPU utilization will be high and memory space will be consumed. The switches might crash and users will likely notice the impact on the network.

# Additional STP Protection Mechanisms

Common issues for Layer 2 forwarding loops:

- STP is disabled on a switch.
- A load balancer is misconfigured and sends traffic out multiple ports with the same MAC address.
- A virtual switch that bridges two physical ports.
- End users using an unmanaged switch or hub.

# Root Guard

Root guard is an STP feature that prevents a configured port from becoming a root port.

- It does this by placing the port in an ErrDisabled state if a superior BDPU is received on that port.
- Root guard is placed on designated ports towards other switches that should never become root bridges.
- Root guard is enabled on a port-by-port basis.

Use the interface command **spanning-tree guard root** to enable root guard.

# STP Portfast

STP portfast disables the topology notification notification (TCN) generation and causes access ports that come up to bypass the learning and listening states and enter the forwarding state immediately. If a BPDU is received on a portfast-enabled port, the portfast functionality is removed from that port.

| Command | Description |
|---|---|
| **spanning-tree portfast** | Interface command to enable portfast on a specific access port |
| **spanning-tree portfast default** | Global command to enable portfast on all access ports |
| **spanning-tree portfast disable** | Disable portfast on a port |
| **spanning-tree portfast trunk** | Command used on trunk links to enable portfast *This command should only be used with ports connected to a single host. |

# STP Portfast Examples

The following shows how to enable STP portfast globally and on a specific interface.

**Example 3-9** *Enabling STP Portfast on Specific Interfaces*

```
SW1(config)# interface gigabitEthernet 1/0/13
SW1(config-if)# switchport mode access
SW1(config-if)# switchport access vlan 10
SW1(config-if)# spanning-tree portfast
```

```
SW1# show spanning-tree vlan 10
! Output omitted for brevity
VLAN0010
Interface            Role Sts Cost      Prio.Nbr Type
-------------------- ---- --- --------- -------- --------------------------------
Gi1/0/2              Desg FWD 4         128.2    P2p
Gi1/0/3              Desg FWD 4         128.3    P2p
Gi1/0/13             Desg FWD 4         128.13   P2p Edge
```

```
SW1# show spanning-tree interface gi1/0/13 detail
 Port 13 (GigabitEthernet1/0/13) of VLAN0010 is designated forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.7.
   Designated root has priority 32778, address 0062.ec9d.c500
   Designated bridge has priority 32778, address 0062.ec9d.c500
   Designated port id is 128.7, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   The port is in the portfast mode
   Link type is point-to-point by default
   BPDU: sent 23103, received 0
```

**Example 3-10** *Enabling STP Portfast Globally*

```
SW2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
SW2(config)# spanning-tree portfast default
%Warning: this command enables portfast by default on all interfaces. You
 should now disable portfast explicitly on switched ports leading to hubs,
 switches and bridges as they may create temporary bridging loops.

SW2(config)# interface gi1/0/8
SW2(config-if)# spanning-tree portfast disable
```

# BPDU Guard

BPDU guard is a safety mechanism that shuts down ports configured with STP portfast upon receiving a BPDU.

| Command | Description |
|---|---|
| **spanning-tree portfast bpduguard default** | Global command to enable BPDU guard on all STP portfast ports |
| **spanning-tree portfast bpduguard default** {**enable** \| **disable**} | Interface command to enables or disable BPDU guard on a specific interface |
| **show spanning-tree interface** *interface-id* **detail** | Displays whether BPDU guard is enabled for the specified interface |

**Note**: BPDU Guard is typically configured with all host-facing ports that are enabled with portfast.

# BPDU Guard Examples

The following shows how to configure BPDU guard and a BPDU guard-enabled port detecting a BPDU.

**Example 3-11** *Configuring BPDU Guard*

```
SW1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)# spanning-tree portfast bpduguard default
SW1(config)# interface gi1/0/8
SW1(config-if)# spanning-tree bpduguard disable
```

```
SW1# show spanning-tree interface gi1/0/7 detail
 Port 7 (GigabitEthernet1/0/7) of VLAN0010 is designated forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.7.
   Designated root has priority 32778, address 0062.ec9d.c500
   Designated bridge has priority 32778, address 0062.ec9d.c500
   Designated port id is 128.7, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   The port is in the portfast mode
   Link type is point-to-point by default
   Bpdu guard is enabled by default
   BPDU: sent 23386, received 0
SW1# show spanning-tree interface gi1/0/8 detail
 Port 8 (GigabitEthernet1/0/8) of VLAN0010 is designated forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.8.
   Designated root has priority 32778, address 0062.ec9d.c500
   Designated bridge has priority 32778, address 0062.ec9d.c500
   Designated port id is 128.8, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   The port is in the portfast mode by default
   Link type is point-to-point by default
   BPDU: sent 23388, received 0
```

**Example 3-12** *Detecting a BPDU on a BPDU Guard–Enabled Port*

```
12:47:02.069: %SPANTREE-2-BLOCK_BPDUGUARD: Received BPDU on port GigabitEthernet1/0/2 with BPDU Guard enabled. Disabling port.
12:47:02.076: %PM-4-ERR_DISABLE: bpduguard error detected on Gi1/0/2, putting Gi1/0/2 in err-disable state
12:47:03.079: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/2, changed state to down
12:47:04.082: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to down
```

```
SW1# show interfaces status

Port      Name               Status       Vlan     Duplex  Speed Type
Gi1/0/1                      notconnect   1          auto   auto 10/100/1000BaseTX
Gi1/0/2   SW2 Gi1/0/1        err-disabled 1          auto   auto 10/100/1000BaseTX
Gi1/0/3   SW3 Gi1/0/1        connected    trunk    a-full a-1000 10/100/1000BaseTX
```

The **show interfaces status** command shows the err-disabled status of the port that received the BPDU.

# BPDU Guard Error Recovery

The Error Recovery service can be used to reactivate ports that are shut down. Ports that are put into the ErrDisabled mode due to BPDU guard do not automatically restore themselves. Use the following commands to recover ports that were shutdown from BPDU guard:

| Command | Description |
|---|---|
| **errdisable recovery cause bpduguard** | Recovers ports shutdown by BPDU guard |
| **errdisable recovery interval** *time-seconds* | The period that Error Recovery checks for ports |

# BPDU Guard Error Recovery Example

The following example shows how to configure the Error Recovery service.

**Example 3-13** *Configuring Error Recovery Service*

```
SW# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW(config)# errdisable recovery cause bpduguard
```

```
SW# show errdisable recovery
! Output omitted for brevity
ErrDisable Reason            Timer Status
-----------------            --------------
arp-inspection               Disabled
bpduguard                    Enabled
..
Recovery command: "clear     Disabled

Timer interval: 300 seconds

Interfaces that will be enabled at the next timeout:

Interface        Errdisable reason        Time left(sec)
---------        -----------------        --------------
Gi1/0/2                bpduguard              295
```

```
! Syslog output from BPDU recovery. The port will be recovered, and then
! triggered again because the port is still receiving BPDUs.
SW#
01:02:08.122: %PM-4-ERR_RECOVER: Attempting to recover from bpduguard err-disable
   state on Gi1/0/2
01:02:10.699: %SPANTREE-2-BLOCK_BPDUGUARD: Received BPDU on port GigabitEthernet
   Ethernet1/0/2 with BPDU Guard enabled. Disabling port.
01:02:10.699: %PM-4-ERR_DISABLE: bpduguard error detected on Gi1/0/2, putting
   Gi1/0/2 in err-disable state
```

Note: The Error Recovery service operates every 300 seconds (5 minutes). This can be changed from 5 to 86,000 seconds with the global command **errdisable recovery interval** *time*

# BPDU Filter

BPDU filter blocks BPDUs from being transmitted out of a port. It can be enabled globally or on a specific interface.

Global BPDU filter command:

**spanning-tree portfast bpdufilter default**

With the global BPDU configuration the port sends a series of 10 – 12 BPDUs. If the switch receives any BPDUs, it checks to identify which switch is more preferred.

- The preferred switch doesn't process any BPDUs but still passes them along to inferior switches.
- A non-preferred switch processes the BPDUs that are received but doesn't transmit any BPDUs to superior switches.

Interface-specific BPDU filter command:

**Spanning-tree bpdufilter enable**

With the interface-specific BPDU configuration the port does not send any BPDUs on an ongoing basis. If the remote port has BPDU guard, that generally shuts down the port as a loop prevention mechanism.

# Verifying a BPDU Filter

The following shows using the **show spanning-tree interface** *interface-id* **detail** command to verify that BPDU filter is enabled.

**Example 3-14** *Verifying a BPDU Filter*

```
! SW1 was enabled with BPDU filter only on port Gi1/0/2
SW1# show spanning-tree interface gi1/0/2 detail | in BPDU|Bpdu|Ethernet
 Port 2 (GigabitEthernet1/0/2) of VLAN0001 is designated forwarding
   Bpdu filter is enabled
   BPDU: sent 113, received 84
SW1# show spanning-tree interface gi1/0/2 detail | in BPDU|Bpdu|Ethernet
 Port 2 (GigabitEthernet1/0/2) of VLAN0001 is designated forwarding
   Bpdu filter is enabled
   BPDU: sent 113, received 84
```

```
! SW1 was enabled with BPDU filter globally
SW2# show spanning-tree interface gi1/0/2 detail | in BPDU|Bpdu|Ethernet
 Port 1 (GigabitEthernet1/0/2) of VLAN0001 is designated forwarding
   BPDU: sent 56, received 5
SW2# show spanning-tree interface gi1/0/2 detail | in BPDU|Bpdu|Ethernet
 Port 1 (GigabitEthernet1/0/2) of VLAN0001 is designated forwarding
   BPDU: sent 58, received 5
```

# Problems with Unidirectional Links

Network devices that utilize fiber-optic cables for connectivity can encounter unidirectional traffic flows if one strand is broken. BPDUs will not able to be transmitted causing other switches on the network to eventually time out the existing root port and change root ports resulting in a forwarding loop.

Two solutions to problems with unidirectional links:

- STP Guard
- Unidirectional Link Detection

# STP Loop Guard

STP Loop guard prevents any alternative or root ports from becoming designated ports due to loss of BPDUs on the root port. Loop guard places the original port into an ErrDisabled state while BPDUs are not being received and transitions back through the STP states when it begins receiving BPDUs again.

| Command | Description |
|---|---|
| **spanning-tree loopguard default** | Global command to enable loop guard |
| **spanning-tree guard loop** | Interface command to enable loop guard |
| **show spanning-tree inconsistent-ports** | Shows ports in the inconsistent state due to the port not receiving BPDUs |

**Note**: Loop guard shouldn't be enabled on portfast-enabled ports because it directly conflicts with root/alternate port logic

# STP Loop Guard Examples

The following examples show configuring loop guard, triggering loop guard by blocking BPDUs and the port in an inconsistent state.

**Example 3-15** *Configuring Loop Guard*

```
SW2# config t
SW2(config)# interface gi1/0/1
SW2(config-if)# spanning-tree guard loop
! Placing BPDU filter on SW2's RP (Gi1/0/1) bridge) triggers loop guard.
SW2(config-if)# interface gi1/0/1
SW2(config-if)# spanning-tree bpdufilter enabled
01:42:35.051: %SPANTREE-2-LOOPGUARD_BLOCK: Loop guard blocking port GigabitEthernet1/0/1 on VLAN0001
```

```
SW2# show spanning-tree vlan 1 | b Interface
Interface            Role Sts Cost      Prio.Nbr Type
-------------------- ---- --- --------- -------- --------------------
Gi1/0/1              Root BKN*4         128.1    P2p *LOOP_Inc
Gi1/0/3              Root FWD 4         128.3    P2p
Gi1/0/4              Desg FWD 4         128.4    P2p
```

**Example 3-16** *Viewing the Inconsistent STP Ports*

```
SW2# show spanning-tree inconsistentports

Name                 Interface            Inconsistency
-------------------- -------------------- ------------------
VLAN0001             GigabitEthernet1/0/1 Loop Inconsistent
VLAN0010             GigabitEthernet1/0/1 Loop Inconsistent
VLAN0020             GigabitEthernet1/0/1 Loop Inconsistent
VLAN0099             GigabitEthernet1/0/1 Loop Inconsistent

Number of inconsistent ports (segments) in the system: 4
```

# Unidirectional Link Detection

Unidirectional Link Detection (UDLD) allows for the bidirectional monitoring of fiber-optic cables.

UDLD operates in two modes:

- **Normal** – If a frame is not acknowledged, the link is considered undetermined and the port remains active.
- **Aggressive** – If a frame is not acknowledged, the switch sends another 8 packets in 1 second intervals. If those packets aren't acknowledged, the port is placed into an error state.

# UDLD Commands

The following are commands for configuring and verifying UDLD:

| Command | Description |
|---|---|
| **udld enable** [**aggressive**] | Global command to enable UDLD. *Optional aggressive keyword sets the mode to aggressive. |
| **udld port** [**aggressive**] | Interface command to enable UDLD *Optional aggressive keyword sets the mode to aggressive. |
| **udld port disable** | Disable UDLD on a specific interface |
| **udld recovery** [**interval** *time*] | Enables UDLD recovery. The *time* default value is 5 minutes. |
| **show udld neighbors** | Displays the status of UDLD neighborship |
| **show udld** *interface-id* | Displays detailed information about UDLD |

# Configuring & Verifying UDLD Examples

The following are examples for configuring and verifying UDLD:

**Example 3-17** *Configuring UDLD*

```
SW1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)# udld enable
```

**Example 3-18** *Verifying UDLD Neighbors and Switch Port Status*

```
SW1# show udld neighbors

Current operational state: Advertisement - Single neighbor detected
Message interval: 15000 ms
Time out interval: 5000 ms

Port fast-hello configuration setting: Disabled
Port fast-hello interval: 0 ms
Port fast-hello operational state: Disabled
Neighbor fast-hello configuration setting: Disabled
Neighbor fast-hello interval: Unknown


    Entry 1
    ---
    Expiration time: 41300 ms
    Cache Device index: 1
    Current neighbor state: Bidirectional
    Device ID: 081C4FF8B0
    Port ID: Te1/1/3
    Neighbor echo 1 device: 062EC9DC50
    Neighbor echo 1 port: Te1/1/3

    TLV Message interval: 15 sec
    No TLV fast-hello interval
    TLV Time out interval: 5
    TLV CDP Device name: SW2
```

# Prepare for the Exam

Prepare for the Exam

# Key Topics for Chapter 3

| Description |
| --- |
| Root bridge placement |
| Root bridge values |
| Spanning tree port cost |
| Root guard |
| STP portfast |
| BPDU guard |
| BPDU filter |

Prepare for the Exam

# Key Terms for Chapter 3

| Terms |
| --- |
| BPDU filter |
| Root guard |
| STP loop guard |
| BPDU guard |
| STP portfast |
| Unidirectional Link Detection (UDLD) |

# Command Reference for Chapter 3

| Task | Command Syntax |
|---|---|
| Configure the STP priority for a switch so that it is a root bridge or a backup root bridge | **spanning-tree vlan** vlan-id **root** {**primary** \| **secondary**} [**diameter** diameter]<br>OR<br>**spanning-tree vlan** vlan-id **priority** priority |
| Configure the STP port cost | **spanning tree** [**vlan** vlan-id] **cost** cost |
| Configure the STP port priority on the downstream port | **spanning-tree** [**vlan** vlan-id] **port-priority** priority |
| Enable root guard on an interface | **spanning-tree guard root** |
| Enable STP portfast globally, for a specific port, or for a trunk port | **spanning-tree portfast default**<br>OR<br>**spanning-tree portfast**<br>OR<br>**spanning-tree portfast trunk** |
| Enable BPDU guard globally or for a specific switch port | **spanning-tree portfast bpduguard default**<br>OR<br>**spanning-tree bpduguard** {**enable** \| **disable**} |

# Command Reference for Chapter 3 (Cont.)

| Task | Command Syntax |
|---|---|
| Enable BPDU guard globally or for a specific interface | **spanning-tree portfast bpdufilter default**<br>OR<br>**spanning-tree bpdufilter enable** |
| Enable STP loop guard globally or for a specific interface | **spanning-tree loopguard default**<br>OR<br>**spanning-tree guard loop** |
| Enable automatic error recovery for BPDU guard. | **errdisable recovery cause bpduguard** |
| Enable BPDU guard globally or for a specific interface | **spanning-tree portfast bpdufilter default**<br>OR<br>**spanning-tree bpdufilter enable** |
| Enable STP loop guard globally or for a specific interface | **spanning-tree loopguard default**<br>OR<br>**spanning-tree guard loop** |
| Enable automatic error recovery for BPDU guard. | **errdisable recovery cause bpduguard** |

# Command Reference for Chapter 3 (Cont.)

| Task | Command Syntax |
|---|---|
| Change the automatic error recovery time | **errdisable recovery interval** time-seconds |
| Enable UDLD globally or for a specific port | **udld enable** [**aggressive**] OR<br>**udld port** [**aggressive**] |
| Display the list of STP ports in an inconsistent state | **show spanning-tree inconsistentports** |
| Display the list of neighbor devices running UDLD | **show udld neighbors** |


CISCO