

Errdisable Port State Recovery on the Cisco IOS Platforms

Document ID: 69980

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

Background Information

- Platforms That Use Errdisable

Errdisable

- Function of Errdisable
- Causes of Errdisable
- Determine If Ports Are in the Errdisabled State
- Determine the Reason for the Errdisabled State (Console Messages, Syslog, and the show errdisable recovery Command)
- Recover a Port from Errdisabled State

Related Information

Introduction

This document defines the errdisabled state, describes how to recover from it, and provides examples of errdisable recovery. This document uses the terms errdisable and error disable interchangeably. Customers often contact Cisco Technical Support when they notice that one or more of their switch ports have become error disabled, which means that the ports have a status of errdisabled. These customers want to know why the error disablement happened and how they can restore the ports to normal.

Note: The port status of `err-disabled` displays in the output of the **show interfaces *interface_number* status** command.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

In order to create the examples in this document, you need two Cisco Catalyst 4500/6500 Series Switches (or the equivalent) in a lab environment with cleared configurations. The switches should run Cisco IOS® Software and each switch should have two Fast Ethernet ports that are capable of EtherChannel and PortFast.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Background Information

Platforms That Use Errdisable

The errdisable feature is supported on these Catalyst switches:

- Catalyst switches that run Cisco IOS Software:
 - ◆ 2900XL / 3500XL
 - ◆ 2940 / 2950 / 2960 / 2970
 - ◆ 3550 / 3560 / 3560-E / 3750 / 3750-E
 - ◆ 4000 / 4500
 - ◆ 6000 / 6500
- Catalyst switches that run Catalyst OS (CatOS) software:
 - ◆ 2948G
 - ◆ 4500 / 4000
 - ◆ 5500 / 5000
 - ◆ 6500 / 6000

The way in which errdisable is implemented varies between software platforms. This document specifically focuses on errdisable for switches that run Cisco IOS Software.

Errdisable

Function of Errdisable

If the configuration shows a port to be enabled, but software on the switch detects an error situation on the port, the software shuts down that port. In other words, the port is automatically disabled by the switch operating system software because of an error condition that is encountered on the port.

When a port is error disabled, it is effectively shut down and no traffic is sent or received on that port. The port LED is set to the color orange and, when you issue the **show interfaces** command, the port status shows **err-disabled**. Here is an example of what an error-disabled port looks like from the command-line interface (CLI) of the switch:

```
cat6knative#show interfaces gigabitethernet 4/1 status

Port      Name      Status      Vlan      Duplex  Speed Type
Gi4/1                    err-disabled 100       full     1000 1000BaseSX
```

Or, if the interface has been disabled because of an error condition, you can see messages that are similar to these in both the console and the syslog:

```
%SPANTREE-SP-2-BLOCK_BPDUGUARD:
  Received BPDU on port GigabitEthernet4/1 with BPDU Guard enabled. Disabling port.
%PM-SP-4-ERR_DISABLE:
  bpduguard error detected on Gi4/1, putting Gi4/1 in err-disable state
```

This example message displays when a host port receives the bridge protocol data unit (BPDU). The actual message depends on the reason for the error condition.

The error disable function serves two purposes:

- It lets the administrator know when and where there is a port problem.
- It eliminates the possibility that this port can cause other ports on the module (or the entire module) to fail.

Such a failure can occur when a bad port monopolizes buffers or port error messages monopolize interprocess communications on the card, which can ultimately cause serious network issues. The error disable feature helps prevent these situations.

Causes of Errdisable

This feature was first implemented to handle special collision situations in which the switch detected excessive or late collisions on a port. Excessive collisions occur when a frame is dropped because the switch encounters 16 collisions in a row. Late collisions occur after every device on the wire should have recognized that the wire was in use. Possible causes of these types of errors include:

- A cable that is out of specification (either too long, the wrong type, or defective)
- A bad network interface card (NIC) card (with physical problems or driver problems)
- A port duplex misconfiguration

A port duplex misconfiguration is a common cause of the errors because of failures to negotiate the speed and duplex properly between two directly connected devices (for example, a NIC that connects to a switch). Only half-duplex connections should ever have collisions in a LAN. Because of the carrier sense multiple access (CSMA) nature of Ethernet, collisions are normal for half duplex, as long as the collisions do not exceed a small percentage of traffic.

There are various reasons for the interface to go into errdisable. The reason can be:

- Duplex mismatch
- Port channel misconfiguration
- BPDU guard violation
- UniDirectional Link Detection (UDLD) condition
- Late-collision detection
- Link-flap detection
- Security violation
- Port Aggregation Protocol (PAgP) flap
- Layer 2 Tunneling Protocol (L2TP) guard
- DHCP snooping rate-limit
- Incorrect GBIC / Small Form-Factor Pluggable (SFP) module or cable
- Address Resolution Protocol (ARP) inspection
- Inline power

Note: Error-disable detection is enabled for all of these reasons by default. In order to disable error-disable detection, use the **no errdisable detect cause** command. The **show errdisable detect** command displays the error-disable detection status.

Determine If Ports Are in the Errdisabled State

You can determine if your port has been error disabled if you issue the **show interfaces** command.

Here is an example of an active port:

```
cat6knative#show interfaces gigabitethernet 4/1 status

!--- Refer to show interfaces status for more information on the command.

Port      Name                Status      Vlan      Duplex  Speed Type
Gi4/1     Connected           100        full     1000   1000BaseSX
```

Here is an example of the same port in the error disabled state:

```
cat6knative#show interfaces gigabitethernet 4/1 status

!--- Refer to show interfaces status for more information on the command.

Port      Name                Status      Vlan      Duplex  Speed Type
Gi4/1     err-disabled       100        full     1000   1000BaseSX
```

Note: When a port is error disabled, the LED on the front panel that is associated with the port is off.

Determine the Reason for the Errdisabled State (Console Messages, Syslog, and the show errdisable recovery Command)

When the switch puts a port in the error-disabled state, the switch sends a message to the console that describes why it disabled the port. The example in this section provides two sample messages that show the reason for port disablement:

- One disablement is because of the PortFast BPDU guard feature.
- The other disablement is because of an EtherChannel configuration problem.

Note: You can also see these messages in the syslog if you issue the **show log** command.

Here are the sample messages:

```
%SPANTREE-SP-2-BLOCK_BPDUGUARD:
  Received BPDU on port GigabitEthernet4/1 with BPDU Guard enabled. Disabling port.

%PM-SP-4-ERR_DISABLE:
  bpduguard error detected on Gi4/1, putting Gi4/1 in err-disable state

%SPANTREE-2-CHNMISCFG: STP loop - channel 11/1-2 is disabled in vlan 1
```

If you have enabled **errdisable recovery**, you can determine the reason for the errdisable status if you issue the **show errdisable recovery** command. Here is an example:

```
cat6knative#show errdisable recovery
ErrDisable Reason      Timer Status
-----
udld                    Enabled
bpduguard              Enabled
security-violatio     Enabled
channel-misconfig     Enabled
pagp-flap              Enabled
dtp-flap               Enabled
link-flap              Enabled
l2ptguard              Enabled
psecure-violation     Enabled
gbic-invalid           Enabled
dhcp-rate-limit       Enabled
mac-limit              Enabled
```

```
unicast-flood      Enabled
arp-inspection     Enabled
```

```
Timer interval: 300 seconds
```

```
Interfaces that will be enabled at the next timeout:
```

Interface	Errdisable reason	Time left(sec)
-----	-----	-----
Fa2/4	bpduguard	273

Recover a Port from Errdisabled State

This section provides examples of how you can encounter an error-disabled port and how to fix it, as well as a brief discussion of a few additional reasons that a port can become error disabled. In order to recover a port from the errdisable state, first identify and correct the root problem, and then reenabling the port. If you reenabling the port before you fix the root problem, the ports just become error disabled again.

Correct the Root Problem

After you discover why the ports were disabled, fix the root problem. The fix depends on the triggering problem. There are numerous things that can trigger the shutdown. This section discusses some of the most noticeable and common causes:

- EtherChannel misconfiguration

In order for EtherChannel to work, the ports that are involved must have consistent configurations. The ports must have the same VLAN, the same trunk mode, the same speed, the same duplex, and so on. Most of the configuration differences within a switch are caught and reported when you create the channel. If one switch is configured for EtherChannel and the other switch is not configured for EtherChannel, the spanning tree process can shut down the channeled ports on the side that is configured for EtherChannel. The on mode of EtherChannel does not send PAgP packets to negotiate with the other side before channeling; it just assumes that the other side is channeling. In addition, this example does not turn on EtherChannel for the other switch, but leaves these ports as individual, unchanneled ports. If you leave the other switch in this state for a minute or so, Spanning Tree Protocol (STP) on the switch where the EtherChannel is turned on thinks that there is a loop. This puts the channeling ports in the errdisabled state.

In this example, a loop was detected and the ports were disabled. The output of the **show etherchannel summary** command shows that the Number of channel-groups in use is 0. When you look at one of the ports that are involved, you can see that the status is err-disabled:

```
%SPANTREE-2-CHNL_MISCFG: Detected loop due to etherchannel misconfiguration
of Gi4/1
```

```
cat6knative#show etherchannel summary
```

```
!--- Refer to show etherchannel for more information on the command.
```

```
Flags:  D - down          P - in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator
```

```
        u - unsuitable for bundling
Number of channel-groups in use: 0
Number of aggregators:          0
```

```
Group Port-channel Protocol Ports
```

The EtherChannel was torn down because the ports were placed in errdisabled on this switch.

```
cat6knative#show interfaces gigabitethernet 4/1 status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Gi4/1		err-disabled	100	full	1000	1000BaseSX

In order to determine what the problem was, look at the error message. The message indicates that the EtherChannel encountered a spanning tree loop. As this section explains, this problem can occur when one device (the switch, in this case) has EtherChannel turned on manually with use of the on mode (as opposed to desirable) and the other connected device (the other switch, in this case) does not have EtherChannel turned on at all. One way to fix the situation is to set the channel mode to desirable on both sides of the connection, and then reenables the ports. Then, each side forms a channel only if both sides agree to channel. If they do not agree to channel, both sides continue to function as normal ports.

```
cat6knative(config-terminal)#interface gigabitethernet 4/1
cat6knative(config-if)#channel-group 3 mode desirable non-silent
```

- Duplex mismatch

Duplex mismatches are common because of failures to autonegotiate speed and duplex properly. Unlike a half duplex device, which must wait until there are no other devices that transmit on the same LAN segment, a full-duplex device transmits whenever the device has something to send, regardless of other devices. If this transmission occurs while the half-duplex device transmits, the half-duplex device considers this either a collision (during the slot time) or a late collision (after the slot time). Because the full-duplex side never expects collisions, this side never realizes that it must retransmit that dropped packet. A low percentage rate of collisions is normal with half duplex but is not normal with full duplex. A switch port that receives many late collisions usually indicates a duplex mismatch problem. Be sure that the ports on both sides of the cable are set to the same speed and duplex. The **show interfaces interface_number** command tells you the speed and duplex for Catalyst switch ports. Later versions of Cisco Discovery Protocol (CDP) can warn you about a duplex mismatch before the port is put in the error-disabled state.

In addition, there are settings on a NIC, such as autopolarity features, that can cause the problem. If you are in doubt, turn these settings off. If you have multiple NICs from a vendor and the NICs all appear to have the same problem, check the manufacturer website for the release notes and be sure that you have the latest drivers.

Other causes of late collisions include:

- ◆ A bad NIC (with physical problems, not just configuration problems)
- ◆ A bad cable
- ◆ A cable segment that is too long
- BPDU port guard

A port that uses PortFast must only connect to an end station (such as a workstation or server) and not to devices that generate spanning tree BPDUs, such as switches, or bridges and routers that do bridging. If the switch receives a spanning tree BPDU on a port that has spanning tree PortFast and spanning tree BPDU guard enabled, the switch puts the port in errdisabled mode in order to guard against potential loops. PortFast assumes that a port on a switch cannot generate a physical loop. Therefore, PortFast skips the initial spanning tree checks for that port, which avoids the timeout of end stations at bootup. The network administrator must carefully implement PortFast. On ports that

have PortFast enabled, BPDU guard helps ensure that the LAN stays loop-free.

This example shows how to turn on this feature. This example was chosen because creation of an error-disable situation is easy in this case:

```
cat6knative(config-if)#spanning-tree bpduguard enable

!--- Refer to spanning-tree bpduguard for more information on the command.
```

In this example, a Catalyst 6509 switch is connected to another switch (a 6509). The 6509 sends BPDUs every 2 seconds (with use of the default spanning tree settings). When you enable PortFast on the 6509 switch port, the BPDU guard feature watches for BPDUs that come in on this port. When a BPDU comes into the port, which means that a device that is not an end device is detected on that port, the BPDU guard feature error disables the port in order to avoid the possibility of a spanning tree loop.

```
cat6knative(config-if)#spanning-tree portfast enable

!--- Refer to spanning-tree portfast (interface configuration mode)
!--- for more information on the command.
```

```
Warning: Spanntree port fast start should only be enabled on ports connected
to a single host. Connecting hubs, concentrators, switches, bridges, etc. to
a fast start port can cause temporary spanning tree loops.
```

```
%PM-SP-4-ERR_DISABLE: bpduguard error detected on Gi4/1, putting Gi4/1 in
err-disable state.
```

In this message, the switch indicates that it received a BPDU on a PortFast-enabled port, and so the switch shuts down port Gi4/1.

```
cat6knative#show interfaces gigabitethernet 4/1 status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Gi4/1		err-disabled	100	full	1000	1000BaseSX

You need to turn off the PortFast feature because this port is a port with an improper connection. The connection is improper because PortFast is enabled, and the switch connects to another switch. Remember that PortFast is only for use on ports that connect to end stations.

```
cat6knative(config-if)#spanning-tree portfast disable
```

• UDLD

The UDLD protocol allows devices that are connected through fiber-optic or copper Ethernet cables (for example, Category 5 cabling) to monitor the physical configuration of the cables and detect when a unidirectional link exists. When a unidirectional link is detected, UDLD shuts down the affected port and alerts the user. Unidirectional links can cause a variety of problems, which include spanning-tree topology loops.

Note: UDLD works by exchanging protocol packets between the neighboring devices. Both devices on the link must support UDLD and have UDLD enabled on the respective ports. If you have UDLD enabled on only one port of a link, it can also leave the end configured with UDLD to go to errdisable state.

Each switch port that is configured for UDLD sends UDLD protocol packets that contain the port device (or port ID) and the neighbor device (or port IDs) that are seen by UDLD on that port. The

neighboring ports must see their own device or port ID (echo) in the packets that are received from the other side. If the port does not see its own device or port ID in the incoming UDLD packets for a specific duration of time, the link is considered unidirectional. Therefore, the respective port is disabled and a message that is similar to this is printed on the console:

```
PM-SP-4-ERR_DISABLE: udld error detected on Gi4/1, putting Gi4/1 in
err-disable state.
```

For more information on UDLD operation, configuration, and commands, refer to the document [Configuring UniDirectional Link Detection \(UDLD\)](#).

- Link-flap error

Link flap means that the interface continually goes up and down. The interface is put into the errdisabled state if it flaps more than five times in 10 seconds. The common cause of link flap is a Layer 1 issue such as a bad cable, duplex mismatch, or bad Gigabit Interface Converter (GBIC) card. Look at the console messages or the messages that were sent to the syslog server that state the reason for the port shutdown.

```
%PM-4-ERR_DISABLE: link-flap error detected on Gi4/1, putting Gi4/
1 in err-disable state
```

Issue this command in order to view the flap values:

```
cat6knative#show errdisable flap-values

!--- Refer to show errdisable flap-values for more information on the command.

ErrDisable Reason      Flaps      Time (sec)
-----
pagp-flap              3           30
dtp-flap                3           30
link-flap               5           10
```

- Loopback error

A loopback error occurs when the keepalive packet is looped back to the port that sent the keepalive. The switch sends keepalives out all the interfaces by default. A device can loop the packets back to the source interface, which usually occurs because there is a logical loop in the network that the spanning tree has not blocked. The source interface receives the keepalive packet that it sent out, and the switch disables the interface (errdisable). This message occurs because the keepalive packet is looped back to the port that sent the keepalive:

```
%PM-4-ERR_DISABLE: loopback error detected on Gi4/1, putting Gi4/1 in
err-disable state
```

Keepalives are sent on all interfaces by default in Cisco IOS Software Release 12.1EA-based software. In Cisco IOS Software Release 12.2SE-based software and later, keepalives are not sent by default on fiber and uplink interfaces. For more information, refer to [Cisco bug ID CSCea46385](#) (registered customers only).

The suggested workaround is to disable keepalives and upgrade to Cisco IOS Software Release 12.2SE or later.

- Port security violation

You can use port security with dynamically learned and static MAC addresses in order to restrict the ingress traffic of a port. In order to restrict the traffic, you can limit the MAC addresses that are allowed to send traffic into the port. In order to configure the switch port to error disable if there is a security violation, issue this command:

```
cat6knative(config-if)#switchport port-security violation shutdown
```

A security violation occurs in either of these two situations:

- ◆ When the maximum number of secure MAC addresses is reached on a secure port and the source MAC address of the ingress traffic differs from any of the identified secure MAC addresses

In this case, port security applies the configured violation mode.

- ◆ If traffic with a secure MAC address that is configured or learned on one secure port attempts to access another secure port in the same VLAN

In this case, port security applies the shutdown violation mode.

For more information on port security, refer to *Configuring Port Security*.

- L2pt guard

When the Layer 2 PDUs enter the tunnel or access port on the inbound edge switch, the switch overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If 802.1Q tunneling is enabled, packets are also double-tagged. The outer tag is the customer metro tag and the inner tag is the customer VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel or access ports in the same metro VLAN. Therefore, the Layer 2 PDUs are kept intact and delivered across the service-provider infrastructure to the other side of the customer network.

```
Switch(config)#interface gigabitethernet 0/7  
l2protocol-tunnel {cdp | vtp | stp}
```

The interface goes to errdisabled state. If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually reenab the port (by issuing a **shutdown, no shutdown** command sequence) or if errdisable recovery is enabled, the operation is retried after a specified time interval.

The interface can be recovered from errdisable state by reenabling the port using the command **errdisable recovery cause l2ptguard**. This command is used to configure the recovery mechanism from a Layer 2 maximum rate error so that the interface can be brought out of the disabled state and allowed to try again. You can also set the time interval. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.

- Incorrect SFP cable

Ports go into errdisable state with the %PHY-4-SFP_NOT_SUPPORTED error message when you connect Catalyst 3560 and Catalyst 3750 Switches using an SFP Interconnect Cable.

The Cisco Catalyst 3560 SFP Interconnect Cable (CAB-SFP-50CM=) provides for a low-cost, point-to-point, Gigabit Ethernet connection between Catalyst 3560 Series Switches. The 50-centimeter (cm) cable is an alternative to using SFP transceivers when interconnecting Catalyst 3560 Series Switches through their SFP ports over a short distance. All Cisco Catalyst 3560 Series Switches support the SFP Interconnect Cable.

When a Catalyst 3560 Switch is connected to a Catalyst 3750 or any other type of Catalyst switch model, you **cannot** use the CAB-SFP-50CM= cable. You can connect both switches using a copper cable with SFP (GLC-T) on both devices instead of a CAB-SFP-50CM= cable.

- 802.1X Security Violation

```
DOT1X-SP-5-SECURITY_VIOLATION: Security violation on interface GigabitEthernet4/8,
New MAC address 0080.ad00.c2e4 is seen on the interface in Single host mode
%PM-SP-4-ERR_DISABLE: security-violation error detected on Gi4/8, putting Gi4/8 in
err-disable state
```

This message indicates that the port on the specified interface is configured in single-host mode. Any new host that is detected on the interface is treated as a security violation. The port has been error disabled.

Ensure that only one host is connected to the port. If you need to connect to an IP phone and a host behind it, configure Multidomain Authentication Mode on that switchport.

The Multidomain authentication (MDA) mode allows an IP phone and a single host behind the IP phone to authenticate independently, with 802.1X, MAC authentication bypass (MAB), or (for the host only) web-based authentication. In this application, Multidomain refers to two domains—data and voice—and only two MAC addresses are allowed per port. The switch can place the host in the data VLAN and the IP phone in the voice VLAN, though they appear to be on the same switch port. The data VLAN assignment can be obtained from the vendor-specific attributes (VSAs) received from the AAA server within authentication.

For more information, refer to the Multidomain Authentication Mode section of Configuring 802.1X Port-Based Authentication.

Reenable the Errdisabled Ports

After you fix the root problem, the ports are still disabled if you have not configured errdisable recovery on the switch. In this case, you must reenable the ports manually. Issue the **shutdown** command and then the **no shutdown** interface mode command on the associated interface in order to manually reenable the ports.

The **errdisable recovery** command allows you to choose the type of errors that automatically reenable the ports after a specified amount of time. The **show errdisable recovery** command shows the default error-disable recovery state for all the possible conditions.

```
cat6knative#show errdisable recovery
ErrDisable Reason      Timer Status
-----
udld                    Disabled
bpduguard               Disabled
security-violatio      Disabled
channel-misconfig      Disabled
pagp-flap              Disabled
dtp-flap               Disabled
link-flap              Disabled
l2ptguard              Disabled
psecure-violation      Disabled
gbic-invalid           Disabled
dhcp-rate-limit        Disabled
mac-limit              Disabled
unicast-flood          Disabled
arp-inspection         Disabled
```

```
Timer interval: 300 seconds
```

```
Interfaces that will be enabled at the next timeout:
```

Note: The default timeout interval is 300 seconds and, by default, the timeout feature is disabled.

In order to turn on **errdisable recovery** and choose the errdisable conditions, issue this command:

```
cat6knative#errdisable recovery cause ?
all          Enable timer to recover from all causes
arp-inspection  Enable timer to recover from arp inspection error disable
              state
bpduguard    Enable timer to recover from BPDU Guard error disable
              state
channel-misconfig  Enable timer to recover from channel misconfig disable
              state
dhcp-rate-limit  Enable timer to recover from dhcp-rate-limit error
              disable state
dtp-flap      Enable timer to recover from dtp-flap error disable state
gbic-invalid   Enable timer to recover from invalid GBIC error disable
              state
l2ptguard     Enable timer to recover from l2protocol-tunnel error
              disable state
link-flap     Enable timer to recover from link-flap error disable
              state
mac-limit     Enable timer to recover from mac limit disable state
pagp-flap     Enable timer to recover from pagp-flap error disable
              state
psecure-violation  Enable timer to recover from psecure violation disable
              state
security-violation  Enable timer to recover from 802.1x violation disable
              state
udld         Enable timer to recover from udld error disable state
unicast-flood  Enable timer to recover from unicast flood disable state
```

This example shows how to enable the BPDU guard errdisable recovery condition:

```
cat6knative(Config)#errdisable recovery cause bpduguard
```

A nice feature of this command is that, if you enable errdisable recovery, the command lists general reasons that the ports have been put into the error-disable state. In this example, notice that the BPDU guard feature was the reason for the shutdown of port 2/4:

```
cat6knative#show errdisable recovery
ErrDisable Reason      Timer Status
-----
udld                   Disabled
bpduguard              Enabled
security-violatio     Disabled
channel-misconfig     Disabled
pagp-flap              Disabled
dtp-flap               Disabled
link-flap              Disabled
l2ptguard              Disabled
psecure-violation     Disabled
gbic-invalid           Disabled
dhcp-rate-limit        Disabled
mac-limit              Disabled
unicast-flood          Disabled
arp-inspection         Disabled

Timer interval: 300 seconds

Interfaces that will be enabled at the next timeout:

Interface      Errdisable reason      Time left(sec)
-----
Fa2/4          bpduguard              290
```

If any one of the errdisable recovery conditions is enabled, the ports with this condition are reenabled after 300 seconds. You can also change this default of 300 seconds if you issue this command:

```
cat6knative(Config)#errdisable recovery interval timer_interval_in_seconds
```

This example changes the errdisable recovery interval from 300 to 400 seconds:

```
cat6knative(Config)#errdisable recovery interval 400
```

Verify

- **show version** Displays the version of the software that is used on the switch.
- **show interfaces *interface interface_number* status** Shows the current status of the switch port.
- **show errdisable detect** Displays the current settings of the errdisable timeout feature and, if any of the ports are currently error disabled, the reason that they are error disabled.

Troubleshoot

- **show interfaces status err-disabled** Shows which local ports are involved in the errdisabled state.
- **show etherchannel summary** Shows the current status of the EtherChannel.
- **show errdisable recovery** Shows the time period after which the interfaces are enabled for errdisable conditions.
- **show errdisable detect** Shows the reason for the errdisable status.

For more information on troubleshooting switchport issues, refer to [Troubleshooting Switch Port and Interface Problems](#).

Related Information

- [Recovering From errDisable Port State on the CatOS Platforms](#)
- [Interface Is in errdisable Status](#) section of [Troubleshooting Hardware and Common Issues on Catalyst 6500/6000 Series Switches Running Cisco IOS System Software](#)
- [Spanning Tree PortFast BPDU Guard Enhancement](#)
- [Understanding EtherChannel Inconsistency Detection](#)
- [Troubleshooting Switch Port and Interface Problems](#)
- [LAN Product Support](#)
- [LAN Switching Technology Support](#)
- [Technical Support & Documentation – Cisco Systems](#)

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Apr 17, 2009

Document ID: 69980
