Configure the UDLD Protocol Feature

Contents

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

Problem Definition

How Unidirectional Link Detection Protocol Works

UDLD Modes of Operation

- Normal Mode
- Aggressive Mode
  - UDLD Message and Hold Timers

Configuration and Monitoring

Related Information

Introduction

This document describes how the Unidirectional Link Detection (UDLD) protocol can help to prevent loops and traffic anomalies in switched networks.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document describes general UDLD operation. The configuration and verification examples in this document were validated on Cisco Catalyst 9300 Series Switches running Cisco IOS XE 17.X release.



Note: Command syntax, defaults, timer ranges, and output can vary by platform and software release.

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, ensure that you understand the potential impact of any command.

Conventions

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

Problem Definition

Spanning-Tree Protocol (STP) resolves redundant physical topology into a loop-free, tree-like forward topology. To do this, it blocks one or more ports. With one or more ports blocked, there are no loops in the forward topology. STP relies in its operation on reception and transmission of the Bridge Protocol Data Units (BPDUs). If a port in the STP blocking or discarding state stops receiving BPDUs from the designated bridge, STP eventually ages out the information associated with that port and moves it to a forwarding state.

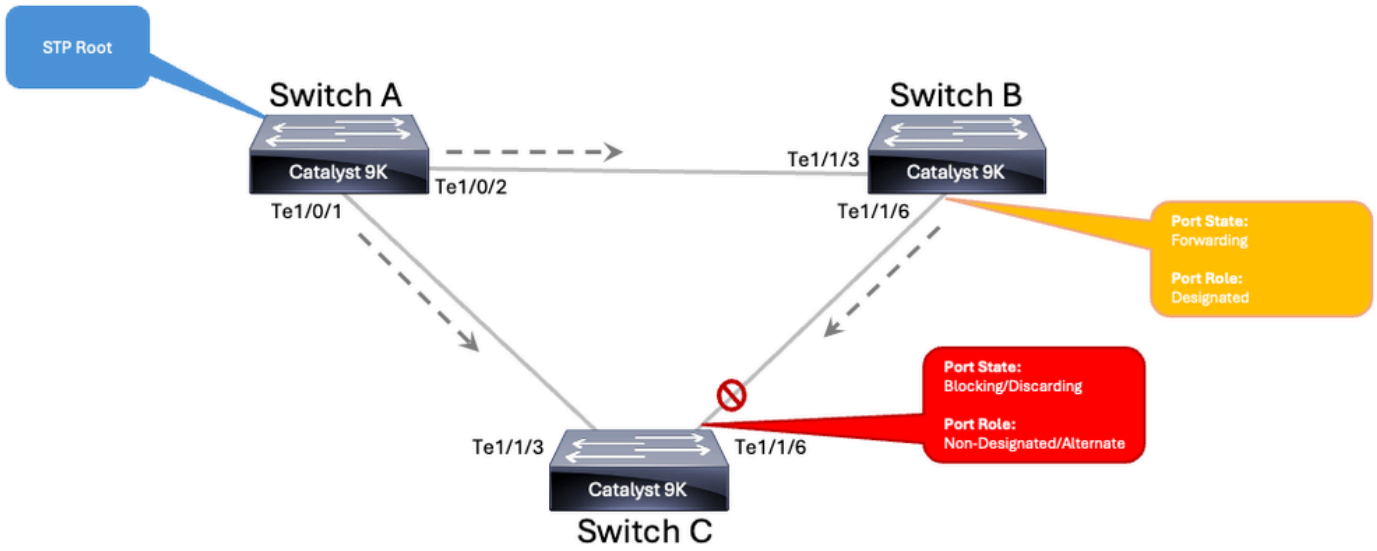
This can create a STP loop where packets start to cycle indefinitely along the looped path, and consumes more bandwidth and resources. This leads to a possible network outage.

How is it possible for the switch to not receive BPDUs while the port is up? This is due to a unidirectional link.

A link is considered unidirectional when:

- The link is up on both sides of the connection.
- The local side do not receive packets sent by the remote side, while the remote side receives packets sent by the local side.

Consider the next scenario, the arrows indicate the flow of STP BPDUs:



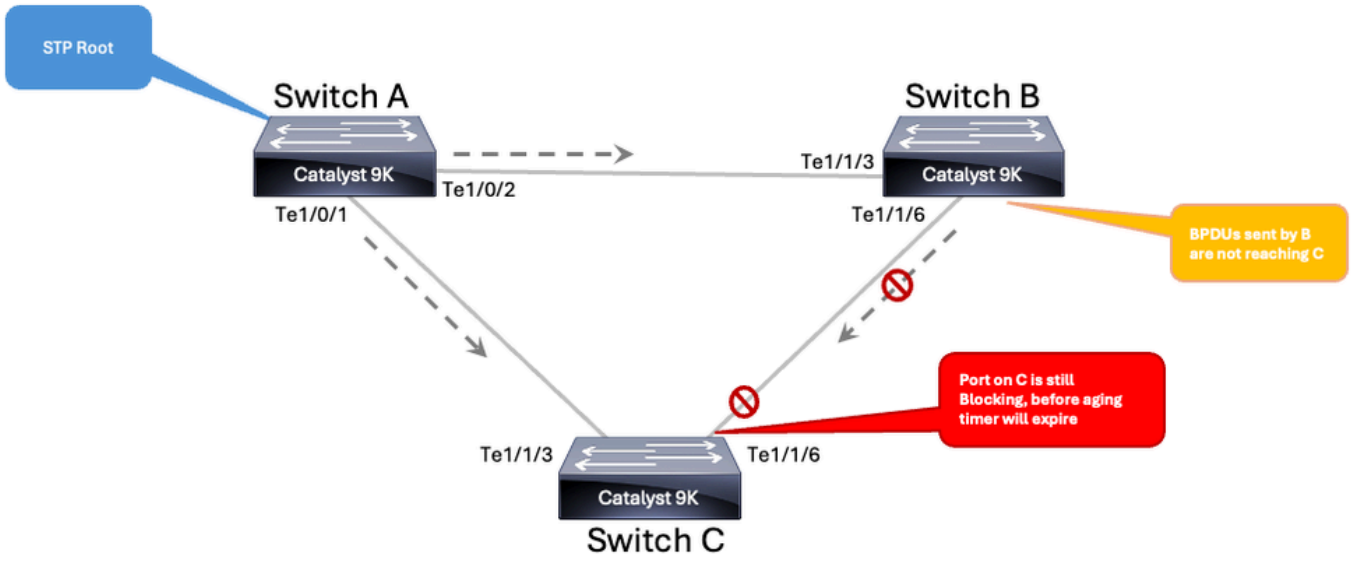
STP Port State Topology

During the condition shown in this topology, the interface on switch B toward switch C is the designated port for the B-C segment and transmits BPDUs toward switch C. The interface on switch C toward switch B is a non-designated port in classic IEEE 802.1D STP. Or an alternate port in RSTP and remains in the blocking/discarding state while it receives and processes BPDUs from switch B. Although the interface on switch C is in the blocking or discarding state, the port continues to receive and process BPDUs; these states prevent the port from forwarding data frames but do not prevent it from receiving BPDUs.

Consider what happens if the B-C link becomes unidirectional and the B → C direction fails. Switch C can no longer receive BPDUs from switch B, while switch B can receive BPDUs transmitted by switch C. Switch C retains the information learned from the last BPDU until that information ages out. With classic IEEE 802.1D STP and the default Max Age of 20 seconds, this can take up to 20 seconds. Once the STP information ages out, switch C no longer considers switch B superior on that segment, and the port can transition from Blocking to Listening, then Learning, and eventually Forwarding.

This creates a Layer 2 forwarding loop because there is no longer a blocked port in the A-B-C triangle. Broadcast, unknown-unicast, and multicast frames can circulate repeatedly around the loop, consuming bandwidth, CPU and potentially resulting in a broadcast storm.

This scenario can bring the network down, which is another possible issue caused by a unidirectional link like a traffic black hole.



STP BPDU Flow

How Unidirectional Link Detection Protocol Works

UDLD is a Layer 2 protocol that supplements Layer 1 link-detection mechanisms by identifying unidirectional communications and specific cabling inconsistencies between directly connected devices.

At Layer 1, auto-negotiation resolves physical signaling and fault detection. UDLD performs tasks that auto-negotiation cannot perform, such as detecting neighbor identities and shutting down misconnected ports. When enabling both auto-negotiation and UDLD; Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

UDLD works through the exchange of protocol packets between the neighboring devices. To establish a bidirectional UDLD neighbor relationship, both directly connected devices must support UDLD and have it enabled on the connected interfaces. Verify the bidirectional state on both devices.

Each switch port configured for UDLD sends UDLD protocol packets that contain the port device/port ID, and the neighbor device/port IDs seen by UDLD on that port.

Neighboring ports see their own device/port ID (echo) in the packets received from the other side. If the port does not see its own device/port ID in the incoming UDLD packets for a specific duration of time, the link is considered unidirectional.

This echo-algorithm allows detection of these issues:

- Link is up on both sides, however, packets are only received by one side.

- Connection (wire) mistakes when receive and transmit fibers are not connected to the same port on the remote side.

Once UDLD detects a unidirectional condition, it places the detecting local interface into the **err-disable** state. The remote interface state depends on remote UDLD detection and physical link behavior. A similar message is printed on the console:

```
UDLD-3-DISABLE: Unidirectional link detected on port 1/2. Port disabled
```

An interface disabled by UDLD remains in the err-disable state until it is manually recovered or an enabled errdisable recovery timer expires. Correct the fiber, transceiver, cabling, or remote-interface fault before recovery. Use shutdown and no shutdown for manual recovery. On platforms that support it, **udld reset** resets interfaces disabled/shutdown by UDLD. After recovery, run **show interfaces status err-disabled**, **show udld <interface-id>**, and **show udld neighbors** commands to confirm the interface is operational and the UDLD relationship is bidirectional. Review the system logs to confirm the fault does not recur.

UDLD Modes of Operation

UDLD can operate in two modes: Normal and Aggressive:

Normal Mode

When UDLD is enabled on a supported interface, normal mode is the default operating mode unless aggressive mode is explicitly configured. UDLD works with Layer 1 mechanisms, such as auto-negotiation, to validate a link. Layer 1 mechanisms detect physical signaling and link faults, while UDLD identifies neighboring devices and verifies fiber strands that connect to the correct ports.

In normal mode, UDLD detects a unidirectional condition when fiber strands are misconnected between ports and Layer 1 mechanisms do not detect the cabling error. If the fiber strands connect to the correct ports but traffic flows in only one direction, UDLD normal mode marks the logical link as **undetermined** and does not disable the port. This behavior relies on Layer 1 mechanisms to detect the physical failure. If one fiber strand is disconnected and auto-negotiation detects the physical fault, the link does not remain up. UDLD takes no shutdown action because Layer 1 has already detected the problem, and the UDLD logical-link state is **undetermined**.

Aggressive Mode

Aggressive mode includes the detection capabilities of normal mode and provides additional protection for

point-to-point fiber-optic and twisted-pair links. It detects misconnected fiber strands and conditions in which one endpoint cannot send or receive traffic, one port remains up while the other is down, or one fiber strand becomes disconnected.

UDLD hello packets act as a heartbeat for the point-to-point link. If UDLD stops receiving these packets after the link has been established as bidirectional, aggressive mode attempts to reestablish the bidirectional relationship. If UDLD cannot restore the relationship, it disables the affected local port to prevent continued use of a link whose bidirectional operation cannot be verified.

When both fiber strands appear operational to Layer 1, aggressive UDLD verifies that they connect to the correct neighboring ports and that traffic flows bidirectionally between the expected neighbors. Auto-negotiation cannot perform this neighbor and port-identity validation because it operates at Layer 1.

UDLD Message and Hold Timers

UDLD information ages out when a port running UDLD does not receive UDLD packets from the neighboring port for the duration of the hold time. These timers apply to UDLD neighbor-information maintenance in both normal and aggressive modes; the action after expiration depends on the configured mode and detected condition. The hold time for the port is dictated by the remote port and depends on the message interval at the remote side. The shorter the message interval, the shorter the hold time and faster the detection. Recent implementations of UDLD allow configuration of message intervals. UDLD information can age out due to the high error rate on the port caused by some physical issue or duplex mismatch. Such packet drop does not mean the link is unidirectional and UDLD in normal mode does not disable the link.

It is important to choose the right message interval to ensure proper detection time. The message interval must be fast enough to detect the unidirectional link before the forwarding loop is created, however, it must not overload the switch CPU. The default message interval in this example is 15 seconds. For the classic IEEE 802.1D STP timer scenario described, the estimated UDLD information-expiration time is less than the estimated time for the blocked port to reach the forwarding state.

This comparison does not guarantee the aggressive UDLD places the interface into the err-disable state before STP changes the forwarding topology. The approximate time for UDLD neighbor information to expire is three times the message interval. For example:

$$T_{expiration} \approx message_interval \times 3$$

With the default message interval **$T_{expiration} \approx 15 \times 3 = 45$ seconds**. For classic IEEE 802.1D STP operation, the approximate time for a blocked port to age out its stored STP information and transition through the listening and learning states to the forwarding state is:

$$T_{forward} = max_age + (2 \times forward_delay)$$

With the default STP timers: **Tforward = 20 + (2 × 15) = 50 seconds** -When comparing UDLD neighbor-information expiration with the STP port transition, select a message interval that maintains: **Texpiration < Tforward**

In aggressive mode, after the UDLD neighbor information expires, UDLD attempts to reestablish the bidirectional relationship by sending one message per second for eight seconds. If the bidirectional state cannot be reestablished, UDLD disables the local port.



Note: These calculations are approximate and apply to the UDLD behavior and classic IEEE 802.1D STP timer scenario described in this document. The deployed platform, software release, STP mode, configured timers, and the time the failure occurs can affect the actual timing.



Note: The classic STP calculation **Treconvergence = max_age + (2 × forward_delay)** does not apply to normal RSTP rapid convergence. RSTP protocol information can expire when BPDUs are not received for three consecutive Hello intervals or when the applicable Max Age condition is reached. An eligible alternate port can then transition rapidly to the forwarding state. The total transition time depends on the topology, port roles, link type, synchronization process, and failure condition. Therefore, a fixed timer calculation cannot guarantee UDLD detects or disables a unidirectional link before RSTP changes the forwarding topology. Validate the interaction between UDLD and RSTP on the deployed platform, software release, topology, and timer configuration.

Examples of additional conditions detected by aggressive mode include:

- The physical link is up on both sides, but the port on one side does not transmit or receive UDLD messages.
- The link is up on one side and down on the other. For example, on some fiber interfaces, disconnecting the transmit fiber from the local port can leave the local link up while the remote link transitions down. This behavior depends on the platform, interface, transceiver, and physical-layer mechanisms.

Some Ethernet PHY implementations provide remote-fault signaling or link-negotiation mechanisms that can cause one or both link endpoints to transition down after specific physical failures. Behavior varies by platform, interface type, transceiver, media, and negotiation mode. UDLD provides Layer 2 validation for unidirectional conditions that physical-layer signaling does not detect. When one endpoint cannot transmit/receive, or when one endpoint is up while the other is down, the failed link itself does not provide a complete forwarding path and does not form a forwarding loop. However, an interface that remains up, can continue forwarding traffic into a nonfunctional path resulting in a traffic black hole. Aggressive UDLD detects the loss of bidirectional communication and disables the affected local port if it cannot reestablish the UDLD relationship.

Configuration and Monitoring

Enable UDLD on both connected interfaces. Configure the same UDLD mode on both endpoints to provide consistent local fault-detection and err-disable behavior:

```
<#root>

9300-1#

configure terminal


Enter configuration commands, one per line.  End with CNTL/Z.
9300-1(config)#

interface TenGigabitEthernet1/1/6

9300-1(config-if)#

udld port


9300-1(config-if)#

end


9300-2#

configure terminal


Enter configuration commands, one per line.  End with CNTL/Z.
9300-2(config)#

interface TenGigabitEthernet1/1/6

9300-2(config-if)#

udld port


9300-2(config-if)#

end
```



Note: Global UDLD commands and their interface scope vary by platform; verify the target platform command reference before using a global command.

Run **show udld <interface-id>** and **show udld neighbors** commands on both link endpoints. Confirm each interface is operationally enabled, the current state is **Bidirectional**, and the expected neighbor device and port identifiers are displayed:

<#root>

9300-1#

show udld TenGigabitEthernet1/1/6

Interface Te1/1/6

Port enable administrative configuration setting: Enabled
Port enable operational state:

Enabled

Current bidirectional state:

Bidirectional

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: 37500 ms
    Cache Device index: 1
    Current neighbor state:

Bidirectional

    Device ID: F87A41A8CA00
    Port ID:

Te1/1/6

    Neighbor echo 1 device: F87A41A8C180
    Neighbor echo 1 port:

Te1/1/6

    TLV Message interval: 15 sec
    No TLV fast-hello interval
    TLV Time out interval: 5
    TLV CDP Device name: 9300-2

9300-1#

show udld neighbors

Port	Device Name	Device ID	Port ID	Neighbor State
Te1/1/6	**F87A41A8CA00**	**1**	**Te1/1/6**	**Bidirectional**

Total number of bidirectional entries displayed: 1

9300-2#

show udld TenGigabitEthernet1/1/6

Interface Te1/1/6

Port enable administrative configuration setting: Enabled
Port enable operational state:

Enabled

Current bidirectional state:

Bidirectional

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: 32500 ms
    Cache Device index: 1
    Current neighbor state:

Bidirectional

```
    Device ID: F87A41A8C180
    Port ID:

Te1/1/6



    Neighbor echo 1 device: F87A41A8CA00
    Neighbor echo 1 port:

Te1/1/6



    TLV Message interval: 15 sec
    No TLV fast-hello interval
    TLV Time out interval: 5
    TLV CDP Device name: 9300-1

9300-2#show udld neighbors
Port           Device Name     Device ID    Port ID         Neighbor State
----           -----------     ---------    -------         --------------

Te1/1/6 F87A41A8C180 1 Te1/1/6 Bidirectional

Total number of bidirectional entries displayed: 1
```

Aggressive UDLD can be configured on the interface with the **udld port aggressive** command:

```
<#root>

9300-1#

configure terminal



Enter configuration commands, one per line.  End with CNTL/Z.
9300-1(config)#

interface TenGigabitEthernet1/1/6

9300-1(config-if)#

udld port aggressive



9300-1(config-if)#

end



9300-2#
```

```
configure terminal


Enter configuration commands, one per line.  End with CNTL/Z.
9300-2(config)#

interface TenGigabitEthernet1/1/6

9300-2(config-if)#

udld port aggressive


9300-2(config-if)#

end
```

Run the **show udld <interface-id>** command on both endpoints to verify aggressive mode is operationally enabled:

<#root>

9300-1#

show udld TenGigabitEthernet1/1/6

```
Interface Te1/1/6
---
Port enable administrative configuration setting: Enabled / in aggressive mode
Port enable operational state:

Enabled / in aggressive mode

Current bidirectional state: Bidirectional
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: 31200 ms
    Cache Device index: 1
    Current neighbor state: Bidirectional
    Device ID: F87A41A8CA00
```

```
    Port ID: Te1/1/6
    Neighbor echo 1 device: F87A41A8C180
    Neighbor echo 1 port: Te1/1/6

    TLV Message interval: 15 sec
    No TLV fast-hello interval
    TLV Time out interval: 5
    TLV CDP Device name: 9300-2

9300-2#
```

show udld TenGigabitEthernet1/1/6

```
Interface Te1/1/6
---
Port enable administrative configuration setting: Enabled / in aggressive mode
Port enable operational state:
```

Enabled / in aggressive mode

```
Current bidirectional state: Bidirectional
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: 38600 ms
    Cache Device index: 1
    Current neighbor state: Bidirectional
    Device ID: F87A41A8C180
    Port ID: Te1/1/6
    Neighbor echo 1 device: F87A41A8CA00
    Neighbor echo 1 port: Te1/1/6

    TLV Message interval: 15 sec
    No TLV fast-hello interval
    TLV Time out interval: 5
    TLV CDP Device name: 9300-1
```

Run the **udld message** time command to change the message interval:

<#root>

9300-1#

configure terminal

```
Enter configuration commands, one per line.  End with CNTL/Z.
9300-1(config)#

udld message time ?

  <1-90>  Time in seconds between sending of messages in steady state
```

On Cisco Catalyst 3000 and 9000 Series Switches, the **udld message** time value ranges from 1 through 90 seconds, and the default is 15 seconds. Refer to the command reference guides to verify the accepted range and default on other systems.

Related Information

- Cisco Technical Support & Downloads
- For Catalyst 3560 switches, refer to Configuring UDLD
- For Catalyst 9300 switches, refer to How to Configure UDLD
- For Catalyst 9500 switches, refer to How to Configure UDLD