Configure the UDLD Protocol Feature

Contents

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 is not restricted to specific software and hardware versions.

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 the STP process that runs on the switch with a port in `blocking` state do not receive BPDUs from its upstream (designated) switch, STP eventually ages out the STP information for the port and moves it to the `forwarding` state.

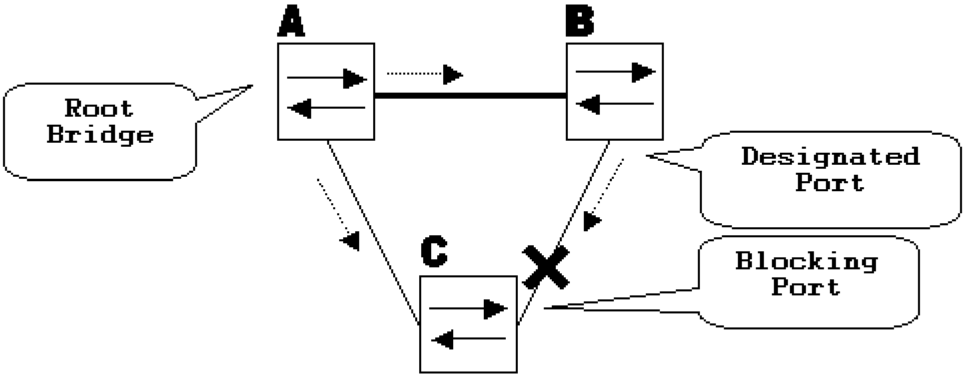
This can create a STP loop where packets start to cycle indefinitely along the looped path, and consumes more and more bandwidth and resources. This leads to a possible network outage.

How is it possible for the switch to not receive BPDUs while the port is `up` ? The reason is a unidirectional link.

A link is considered unidirectional when this occurs:

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

Consider this scenario. The arrows indicate the flow of STP BPDUs.



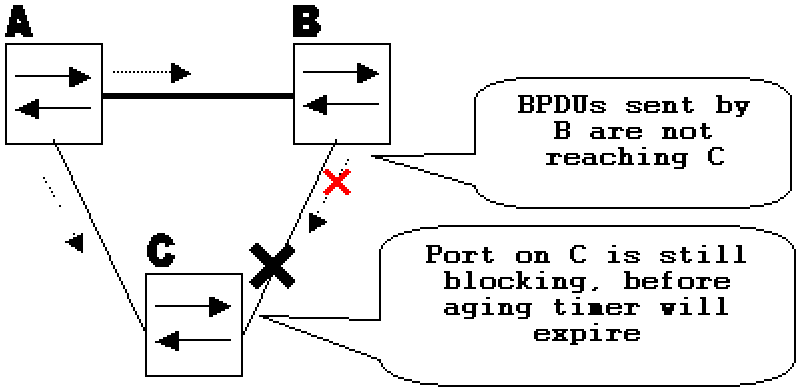
During normal operation, bridge B is a designated port on the link B-C. Bridge B sends BPDUs down to C, which is the blocking the port. The port is blocked while C sees BPDUs from B on that link.

Now, consider what happens if the link B-C fails in the direction of C. C ceases to receive traffic from B, however, B still receives traffic from C.

up

C does not receive BPDUs on the link B-C, and ages the information received with the last BPDU. This takes up to 20 seconds, which depends on the maxAge STP timer. Once the STP information is aged out on the port, that port transitions from the `blocking` state to the `listening` , `learning` , and eventually to the `forwarding` STP state. This creates a loop, as there is no blocked port in the triangle A-B-C. Packets cycle along the path (B still receives packets from C) which consumes additional bandwidth until the links are completely filled up.

This scenario can bring the network down. Another possible issue that can be caused by a unidirectional link is a traffic blackhole.


A
B
C
BPDUs sent by B are not reaching C
Port on C is still blocking, before aging timer will expire

How Unidirectional Link Detection Protocol Works

UDLD is a Layer 2 (L2) protocol that works with the Layer 1 (L1) mechanisms to determine the physical status of a link. At Layer 1, auto-negotiation takes care of physical signaling and fault detection. UDLD performs tasks that auto-negotiation cannot perform, such as the detection of the identities of neighbors and shut down misconnected ports. When you enable 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. In order for UDLD to work, both devices on the link must support UDLD and have it enabled on the respective ports.

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 the unidirectional link is detected by UDLD, the respective port is disabled and this message is printed on the console:

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

Port shutdown by UDLD remains disabled until it is manually enabled, or until`errdisable`timeout expires (if configured).

UDLD Modes of Operation

UDLD can operate in two modes: `normal` and `aggressive: .`

- In `normal` mode, if the link state of the port was determined to be bi-directional and the UDLD information times out, no action is taken by UDLD. The port state for UDLD is marked as

`undetermined` . The port behaves in accordance with its STP state.
- In `aggressive` mode, if the link state of the port is determined to be bi-directional and the UDLD information times out while the link on the port is still `up` , UDLD tries to re-establish the state of the port. If not successful, the port is put into the `errdisable` state.

Age out of UDLD information happens when the port that runs UDLD does not receive UDLD packets from the neighbor port for the duration of hold time. 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 interval. 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 that the link is unidirectional and UDLD in `normal` mode does not disable such link.

It is important to be able to choose the right message interval in order to ensure proper detection time. The message interval needs to be fast enough to detect the unidirectional link before the forward loop is created, however, it must not overload the switch CPU. The default message interval is 15 seconds, and is fast enough to detect the unidirectional link before the forward loop is created with default STP timers. The detection time is approximately equal to three times the message interval.

For example: $T_{detection}$~ message_interval x3

This is 45 seconds for the default message interval of 15 seconds.

It takes $T_{reconvergence}$=max_age + 2x forward_delay for the STP to reconverge in case of unidirectional link failure. With the default timers, it takes 20+2x15=50 seconds.

It is recommended to keep $T_{detection} < T_{reconvergence}$ and choose an appropriate message interval.

In `aggressive` mode, once the information is aged, UDLD makes an attempt to re-establish the link state and send packets every second for eight seconds. If the link state is still not determined, the link is disabled.

`Aggressive`mode adds additional detection of these situations:

- The port is stuck (on one side the port neither transmits nor receives, however, the link is `up` on both sides).

- The link is `up` on one side and `down` on the other side. This issue can seen on fiber ports when transmit fiber is unplugged on the local port, the link remains `up` on the local side. However, it is `down< /tt>on the remote side.`

Most recently, fiber FastEthernet hardware implementations have Far End Fault Indication (FEFI) functions in order to bring the link `down` on both sides in these situations. On GigabitEthernet, a similar function is provided by link negotiation. Copper ports are normally not susceptible to this type of issue, as they use Ethernet link pulses to monitor the link. It is important to mention that in both cases, no forward loop occurs because there is no connectivity between the ports. However, if the link is up on one side and down on the other traffic blackhole can occur. Aggressive UDLD is designed to prevent this.

Availability

UDLD is available in normal and aggressive mode from Cisco IOS® Software Release 12 and later.

Configuration and Monitoring

Run the command **show udld** to verify if UDLD is enabled on the interfaces:

```
<#root>

Switch#

show udld



Interface Gi1/0/1
---
Port enable administrative configuration setting: Disabled
Port enable operational state: Disabled
Current bidirectional state: Unknown

Interface Gi1/0/2
---
Port enable administrative configuration setting: Disabled
Port enable operational state: Disabled
Current bidirectional state: Unknown

Interface Gi1/0/3
---
Port enable administrative configuration setting: Disabled
Port enable operational state: Disabled
Current bidirectional state: Unknown
```

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

```
<#root>

Switch#

configure terminal


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

interface gigabitEthernet1/0/1


Switch(config-if)#

udld port aggressive


Switch(config-if)#

end


Switch#
```

Issue the**show udld <interface>** and **show udld neighbors** command to verify whether UDLD is enabled or disabled on the port and what the link and neighbor state is:

```
<#root>
```

```
Switch#

show udld GigabitEthernet1/0/1



Interface Gi1/0/1
---
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: 31600 ms
Cache Device index: 1
Current neighbor state:

Bidirectional


Device ID: 346288238580
Port ID: Gi4/0/1
Neighbor echo 1 device: 70B4F35F080
Neighbor echo 1 port: Gi1/0/1

TLV Message interval: 15 sec
No TLV fast-hello interval
TLV Time out interval: 5
TLV CDP Device name: MXC.TAC.M.02-3850-01
```

<#root>

```
Switch#

show udld neighbors


Port      Device Name    Device ID   Port ID     Neighbor State
----      -----------    ---------   -------     --------------
Gi1/0/1   346288238580   1           Gi4/0/1     Bidirectional

Total number of bidirectional entries displayed: 1
```

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

```
<#root>

Switch(config)#

udld message time 10

UDLD message interval set to 10 seconds
```

The interval can range from 1 to 90 seconds, with the default of 15 seconds.

Related Information

- Cisco Technical Support & Downloads
- For Catalyst 3560 switches, refer toConfiguring UDLD.
- For Catalyst 4500/4000 which run Cisco IOS, refer toConfiguring UDLD.
- For Catalyst 9300 switches, refer to How to Configure UDLD
- For Catalyst 9500 switches, refer to How to Configure UDLD