

Practical 6

AIM: Observe STP Topology Changes and Implement RSTP

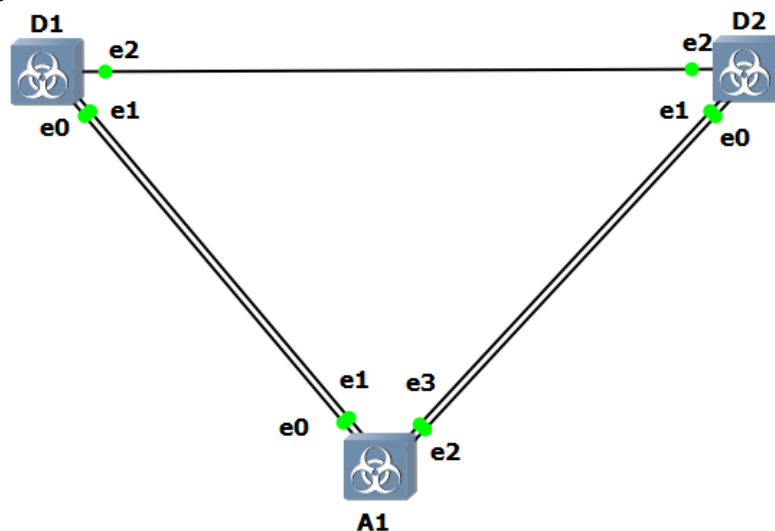
1. Implement Advanced STP Modifications and Mechanisms
2. Implement MST

Implement Advanced STP Modifications and Mechanisms

What is STP?

- Spanning Tree Protocol (STP) is used to make a loop free network by monitoring the network to track all the links and shut down the least redundant ones.
- Root bridge is a switch in a single VLAN or whole topology (according to the type of STP standard used) which is responsible for distributing BPDUs and block the least redundant port.
- Networks are often configured with redundant paths when connecting network segments.
- Although redundancy can help protect against disaster, it can also lead to bridge or switch looping.
- Looping occurs when data travels from a source to a destination along redundant paths and the data begins to circle around the same paths, becoming amplified and resulting in a broadcast storm.
- STP can help prevent bridge looping on LANs that include redundant links. Without STP, it would be difficult to implement that redundancy and still avoid network looping. STP monitors all network links, identifies redundant connections, and disables the ports that can lead to looping.

Step 1: Design the network



Step 2: Configure the devices:**D1:**

```

vIOS-L2-01>
vIOS-L2-01>enable
vIOS-L2-01#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vIOS-L2-01(config)#hostname D1
D1(config)#spanning-tree mode pvst
D1(config)#banner motd #D1, STP Topology Changes and RSTP Lab#
D1(config)#line con 0
D1(config-line)#exec-timeout 0 0
D1(config-line)#logging synchronous
D1(config-line)#exit
D1(config)#interface range gi0/0-2
D1(config-if-range)#switchport trunk encapsulation dot1q
D1(config-if-range)#switchport mode trunk
D1(config-if-range)#no shutdown
D1(config-if-range)#exit
D1(config)#vlan 2
D1(config-vlan)#name SecondVLAN
D1(config-vlan)#exit
D1(config)#interface vlan 1
D1(config-if)#ip address 10.0.0.1 255.0.0.0
D1(config-if)#no shutdown
D1(config-if)#exit
D1(config)#
D1(config)#

```

D2:

```

vIOS-L2-01>
vIOS-L2-01>enable
vIOS-L2-01#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vIOS-L2-01(config)#hostname D2
D2(config)#banner motd #D2, STP Topology Changes and RSTP Lab#
D2(config)#spanning-tree mode pvst
D2(config)#line con 0
D2(config-line)#exec-timeout 0 0
D2(config-line)#logging synchronous
D2(config-line)#exit
D2(config)#interface range gi0/0-2
D2(config-if-range)#switchport trunk encapsulation dot1q
D2(config-if-range)#switchport mode trunk
D2(config-if-range)#no shutdown
D2(config-if-range)#exit
D2(config)#vlan 2
D2(config-vlan)#name SecondVLAN
D2(config-vlan)#exit
D2(config)#interface vlan 1
D2(config-if)#ip address 10.0.0.2 255.0.0.0
D2(config-if)#no shutdown
D2(config-if)#exit
*Jan  6 14:59:11.825: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
D2(config-if)#exit
*Jan  6 14:59:14.756: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
*Jan  6 14:59:16.496: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
D2(config-if)#exit
D2(config)#
D2(config)#

```

A1:

```

vIOS-L2-01>
vIOS-L2-01>enable
vIOS-L2-01#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vIOS-L2-01(config)#hostname A1
A1(config)#banner motd #A1, STP Topology Changes and RSTP Lab#
A1(config)#spanning-tree mode pvst
A1(config)#line con 0
A1(config-line)#exec-timeout 0 0
A1(config-line)#logging synchronous
A1(config-line)#exit
A1(config)#interface range gi0/0-3
A1(config-if-range)#switchport trunk encapsulation dot1q
A1(config-if-range)#switchport mode trunk
A1(config-if-range)#no shutdown
A1(config-if-range)#exit
A1(config)#vlan 2
A1(config-vlan)#name SecondVLAN
A1(config-vlan)#exit
A1(config)#interface vlan 1
A1(config-if)#ip address 10.0.0.3 255.0.0.0
A1(config-if)#no shutdown
A1(config-if)#exit
*Jan 6 14:46:43.960: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
A1(config-if)#exit
*Jan 6 14:46:47.144: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
*Jan 6 14:46:48.742: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
A1(config-if)#exit
A1(config)#

```

Step 3: Discover the Default Spanning Tree.**1. Find the root bridge.**

- The root bridge is elected based upon which switch has the highest Bridge ID (BID).
- The BID is made up of a configurable priority value (which defaults to 32768) and the base MAC address for the switch.
- Use the command show spanning-tree root to gather that information from your switches to support the root bridge decision.

Enter the command ‘show spanning-tree root’ and observe what the output tells you about the root bridge.

D1:

```

D1#show spanning-tree root

```

| Vlan | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|----------|----------------------|-----------|------------|---------|---------|-----------|
| VLAN0001 | 32769 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0002 | 32770 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0100 | 32868 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0200 | 32968 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0300 | 33068 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |

```

D1#

```

D2:

```

Jan  6 14:55:42.924: %SYS-5-CONFIG-I: Configured from console by console
D2#show spanning-tree root

```

| Vlan | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|----------|----------------------|-----------|------------|---------|---------|-----------|
| VLAN0001 | 32769 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0002 | 32770 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0100 | 32868 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0200 | 32968 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0300 | 33068 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |

```

D2#
D2#

```

A1:

```

A1#show spanning-tree root

```

| Vlan | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|----------|----------------------|-----------|------------|---------|---------|-----------|
| VLAN0001 | 32769 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0002 | 32770 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0100 | 32868 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0200 | 32968 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0300 | 33068 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |

```

A1#
A1#
*Jan  6 14:47:21.369: %SYS-5-CONFIG-I: Configured from console by console
A1#

```

The root cost is the cost to go to root bridge and in D1 and A1 there is a cost of 4.

But in D2 there is no cost because D2 is our root bridge.

We will see it clearly in upcoming Screenshot

2. Find the Root Port for each switch

- As we saw in the previous output of show spanning-tree root on each switch, the Path Cost can be different amongst switches.
- In this case, the path cost from A1 to D2 is 4, reflecting connectivity via a Gigabit Ethernet port
- While the path cost from D1 to D2 is 4, reflecting connectivity via a Gigabit Ethernet port.
- While the path cost from D2 to D2 is 0 because it is root bridge.

```

D1#show spanning-tree root

```

| Vlan | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|----------|----------------------|-----------|------------|---------|---------|-----------|
| VLAN0001 | 32769 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0002 | 32770 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0100 | 32868 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0200 | 32968 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0300 | 33068 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |

```

D1#

```

```

*Jan  6 14:55:42.924: %SYS-5-CONFIG-I: Configured from console by console
D2#show spanning-tree root

```

| Vlan | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|----------|----------------------|-----------|------------|---------|---------|-----------|
| VLAN0001 | 32769 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0002 | 32770 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0100 | 32868 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0200 | 32968 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |
| VLAN0300 | 33068 0c9d.c7a4.0000 | 0 | 2 | 20 | 15 | |

```

D2#

```

```

A1#show spanning-tree root

```

| Vlan | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|----------|----------------------|-----------|------------|---------|---------|-----------|
| VLAN0001 | 32769 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0002 | 32770 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0100 | 32868 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0200 | 32968 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |
| VLAN0300 | 33068 0c9d.c7a4.0000 | 4 | 2 | 20 | 15 | Gi0/2 |

```

A1#

```

```

A1#

```

```

*Jan  6 14:47:21.369: %SYS-5-CONFIG-I: Configured from console by console

```

```

A1#

```

3. Changes in network:

A1: Issue 'show spanning-tree' in A1 and check the output

```

A1#show spanning-tree

```

```

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0c9d.c7a4.0000
             Cost        4
             Port        3 (GigabitEthernet0/2)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
             Address     0cbd.e307.0000
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

```

| Interface | Role | Sts | Cost | Prio.Nbr | Type |
|-----------|------|-----|------|----------|------|
| Gi0/0 | Desg | FWD | 4 | 128.1 | Shr |
| Gi0/1 | Desg | FWD | 4 | 128.2 | Shr |
| Gi0/2 | Root | FWD | 4 | 128.3 | Shr |
| Gi0/3 | Altn | BLK | 4 | 128.4 | Shr |

Our topology does not really illustrate the difference between port cost and path cost very well, so we will introduce a change in the network to achieve this. At D1, shutdown the g1/0/1 interface.

The result of this is that D2 will have to change the port it considers root, and we will then see the difference between port cost and path cost.

```
D1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D1(config)#interface gi0/2
D1(config-if)#shutdown
D1(config-if)#
*Jan 6 07:51:59.728: %LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to administratively down
*Jan 6 07:52:00.962: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to down
```

Now enter ‘show spanning-tree’ at root i.e. D2:

```

D2#
D2#show spanning-tree

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0c9d.c7a4.0000
             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     0c9d.c7a4.0000
             Hello Time  2 sec    Max Age 20 sec    Forward Delay 15 sec
             Aging Time  15 sec

Interface                Role Sts Cost           Prio.Nbr Type
-----
Gi0/0                    Desg FWD 4             128.1 Shr
Gi0/1                    Desg FWD 4             128.2 Shr
Gi0/2                    Desg FWD 4             128.3 Shr

```

4. Identify Designated Ports.

If you have not already done so, issue the no shutdown command for D1 interface g1/0/1.

This will restore our full topology and allow for the non-root attached segment to exist (the links between A1 and D2).

```
D1(config)#interface gi0/2
D1(config-if)#no shutdown
D1(config-if)#exit
D1(config)#
```

On D2, issue the show spanning-tree command, and you will see that there are two ports now identified as being in the Designated Port role.

```
D2#
D2#show spanning-tree

VLAN0001
Spanning tree enabled protocol ieee
Root ID    Priority    32769
           Address     0c9d.c7a4.0000
           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     0c9d.c7a4.0000
           Hello Time 2 sec    Max Age 20 sec    Forward Delay 15 sec
           Aging Time 15 sec

Interface                Role Sts Cost          Prio.Nbr Type
-----
Gi0/0                    Desg FWD 4             128.1 Shr
Gi0/1                    Desg FWD 4             128.2 Shr
Gi0/2                    Desg FWD 4             128.3 Shr
```


And now look at the segments from the A1 side. Issue the show spanning-tree command on A1.

```
A1#
A1#show spanning-tree
```

VLAN0001

```
Spanning tree enabled protocol ieee
Root ID      Priority      32769
              Address      0c9d.c7a4.0000
              Cost        4
              Port        3 (GigabitEthernet0/2)
              Hello Time   2 sec    Max Age 20 sec    Forward Delay 15 sec

Bridge ID     Priority      32769 (priority 32768 sys-id-ext 1)
              Address      0cbd.e307.0000
              Hello Time   2 sec    Max Age 20 sec    Forward Delay 15 sec
              Aging Time   15 sec
```

| Interface | Role | Sts | Cost | Prio.Nbr | Type |
|-----------|------|-----|------|----------|------|
| Gi0/0 | Desg | FWD | 4 | 128.1 | Shr |
| Gi0/1 | Desg | FWD | 4 | 128.2 | Shr |
| Gi0/2 | Root | FWD | 4 | 128.3 | Shr |
| Gi0/3 | Altn | BLK | 4 | 128.4 | Shr |

You may have noticed in the previous output that the two links from A1 to D1 were not being used.

Step 4: Implement and Observe Rapid Spanning Tree Protocol.

- We will implement Rapid Spanning Tree Protocol (RSTP) on all the switches. Using the same basic rules, RSTP speeds up convergence significantly.
- On D2, issue the debug spanning-tree events command, and then issue the shutdown command for interface g1/0/1 and observe the output.

```
D2#
D2#debug spanning-tree events
Spanning Tree event debugging is on
D2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D2(config)#interface gi0/2
D2(config-if)#shutdown
D2(config-if)#
*Jan 6 15:03:36.618: STP: VLAN0001 we are the spanning tree root
*Jan 6 15:03:36.620: STP[1]: Generating TC trap for port GigabitEthernet0/2
*Jan 6 15:03:36.621: STP: VLAN0002 we are the spanning tree root
*Jan 6 15:03:36.621: STP[2]: Generating TC trap for port GigabitEthernet0/2
*Jan 6 15:03:36.621: STP: VLAN0100 we are the spanning tree root
*Jan 6 15:03:36.621: STP[100]: Generating TC trap for port GigabitEthernet0/2
*Jan 6 15:03:36.621: STP: VLAN0200 we are the spanning tree root
*Jan 6 15:03:36.621: STP[200]: Generating TC trap for port GigabitEthernet0/2
*Jan 6 15:03:36.622: STP: VLAN0300 we are the spanning tree root
D2(config-if)#
*Jan 6 15:03:36.623: STP[300]: Generating TC trap for port GigabitEthernet0/2
D2(config-if)#
*Jan 6 15:03:39.387: %LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to administratively down
*Jan 6 15:03:41.182: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to down
D2(config-if)#
*Jan 6 15:04:09.755: %PLATFORM-5-SIGNATURE_VERIFIED: Image 'flash0:/vios_l2-adventerprisek9-m' passed code signing verification
D2(config-if)#
*Jan 6 15:05:58.490: STP: VLAN0002 Topology Change rcvd on Gi0/0
*Jan 6 15:05:58.495: STP: VLAN0100 Topology Change rcvd on Gi0/0
*Jan 6 15:05:58.495: STP: VLAN0200 Topology Change rcvd on Gi0/0
*Jan 6 15:05:58.500: STP: VLAN0300 Topology Change rcvd on Gi0/0
*Jan 6 15:05:58.502: STP: VLAN0001 Topology Change rcvd on Gi0/0
D2(config-if)#
```

On D1, change the spanning tree mode to rapid-pvst:

```
D1(config)#spanning-tree mode rapid-pvst
D1(config)#
```

On D1, issue the command show spanning-tree.

```
D1#show spanning-tree

VLAN0001
  Spanning tree enabled protocol rstp
  Root ID    Priority    32769
             Address     0c9d.c7a4.0000
             Cost        8
             Port        1 (GigabitEthernet0/0)
             Hello Time   2 sec   Max Age 20 sec   Forward Delay 15 sec

  Bridge ID   Priority    32769 (priority 32768 sys-id-ext 1)
             Address     0ceb.a4ba.0000
             Hello Time   2 sec   Max Age 20 sec   Forward Delay 15 sec
             Aging Time   300 sec

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gi0/0                    Root FWD 4         128.1   Shr Peer(STP)
Gi0/1                    Altn BLK 4         128.2   Shr Peer(STP)
Gi0/2                    Desg BLK 4         128.3   Shr
```

The type values tell the story. Rapid spanning tree is backwards compatible with common spanning tree.

We will not see the benefits of rapid spanning tree if only one switch is running it.

On D2 and A1, change the spanning tree mode to rapid spanning tree.

For D2:(Note that I have only enabled the debugging mode for D2, this is why the output for D2 is different from D1 and A1 when the spanning-tree mode is set to rstp.)

```
D2(config)#
*Jan 6 15:07:20.079: setting bridge id (which=3) prio 32769 prio cfg 32768 sysid 1 (on) id 8001.0c9d.c7a4.0000
*Jan 6 15:07:20.080: RSTP(1): initializing port Gi0/0
*Jan 6 15:07:20.081: RSTP(1): Gi0/0 is now designated
*Jan 6 15:07:20.082: RSTP(1): initializing port Gi0/1
*Jan 6 15:07:20.083: RSTP(1): Gi0/1 is now designated
*Jan 6 15:07:20.084: setting bridge id (which=3) prio 32770 prio cfg 32768 sysid 2 (on) id 8002.0c9d.c7a4.0000
*Jan 6 15:07:20.084: RSTP(2): initializing port Gi0/0
*Jan 6 15:07:20.084: RSTP(2): Gi0/0 is now designated
*Jan 6 15:07:20.084: RSTP(2): initializing port Gi0/1
*Jan 6 15:07:20.084: RSTP(2): Gi0/1 is now designated
*Jan 6 15:07:20.085: setting bridge id (which=3) prio 32868 prio cfg 32768 sysid 100 (on) id 8064.0c9d.c7a4.0000
*Jan 6 15:07:20.085: RSTP(100): initializing port Gi0/0
*Jan 6 15:07:20.085: RSTP(100): Gi0/0 is now designated
*Jan 6 15:07:20.085: RSTP(100): initializing port Gi0/1
*Jan 6 15:07:20.085: RSTP(100): Gi0/1 is now designated
*Jan 6 15:07:20.085: setting bridge id (which=3) prio 32968 prio cfg 32768 sysid 200 (on) id 80C8.0c9d.c7a4.0000
*Jan 6 15:07:20.086: RSTP(200): initializing port Gi0/0
*Jan 6 15:07:20.086: RSTP(200): Gi0/0 is now designated
*Jan 6 15:07:20.086: RSTP(200): initializing port Gi0/1
*Jan 6 15:07:20.086: RSTP(200): Gi0/1 is now designated
*Jan 6 15:07:20.086: setting bridge id (which=3) prio 33068 prio cfg 32768 sysid 300 (on) id 812C.0c9d.c7a4.0000
*Jan 6 15:07:20.086: RSTP(300): initializing port Gi0/0
*Jan 6 15:07:20.086: RSTP(300): Gi0/0 is now designated
*Jan 6 15:07:20.086: RSTP(300): initializing port Gi0/1
*Jan 6 15:07:20.086: RSTP(300): Gi0/1 is now designated
*Jan 6 15:07:20.099: RSTP(1): transmitting a proposal on Gi0/0
*Jan 6 15:07:20.099: RSTP(1): transmitting a proposal on Gi0/1
*Jan 6 15:07:20.101: RSTP(2): transmitting a proposal on Gi0/0
*Jan 6 15:07:20.104: RSTP(2): transmitting a proposal on Gi0/1
*Jan 6 15:07:20.105:
D2(config)#
```



```

Jan 6 15:07:25.099: RSTP(300): transmitting a proposal on Gi0/1
D2(config)#
*Jan 6 15:07:25.090: RSTP(1): transmitting a proposal on Gi0/0
*Jan 6 15:07:25.093: RSTP(1): transmitting a proposal on Gi0/1
*Jan 6 15:07:25.096: RSTP(2): transmitting a proposal on Gi0/0
*Jan 6 15:07:25.098: RSTP(2): transmitting a proposal on Gi0/1
*Jan 6 15:07:25.099: RSTP(100): transmitting a proposal on Gi0/0
*Jan 6 15:07:25.099: RSTP(100): transmitting a proposal on Gi0/1
*Jan 6 15:07:25.099: RSTP(200): transmitting a proposal on Gi0/0
*Jan 6 15:07:25.099: RSTP(200): transmitting a proposal on Gi0/1
*Jan 6 15:07:25.100: RSTP(300): transmitting a proposal on Gi0/0
*Jan 6 15:07:25.100: RSTP(300): transmitting a proposal on Gi0/1
D2(config)#
*Jan 6 15:07:26.547: RSTP(2): transmitting a proposal on Gi0/1
*Jan 6 15:07:26.551: RSTP(300): transmitting a proposal on Gi0/1
*Jan 6 15:07:26.551: RSTP(100): transmitting a proposal on Gi0/1
*Jan 6 15:07:26.552: RSTP(200): transmitting a proposal on Gi0/1
*Jan 6 15:07:27.095: RSTP(1): transmitting a proposal on Gi0/0
*Jan 6 15:07:27.096: RSTP(1): transmitting a proposal on Gi0/1
*Jan 6 15:07:27.101: RSTP(2): transmitting a proposal on Gi0/1
*Jan 6 15:07:27.101: RSTP(100): transmitting a proposal on Gi0/1
*Jan 6 15:07:27.101: RSTP(200): transmitting a proposal on Gi0/1
*Jan 6 15:07:27.101: RSTP(300): transmitting a proposal on Gi0/1
D2(config)#
*Jan 6 15:07:27.480: RSTP(1): transmitting a proposal on Gi0/1
D2(config)#
*Jan 6 15:07:29.100: RSTP(1): transmitting a proposal on Gi0/1
*Jan 6 15:07:29.102: RSTP(2): transmitting a proposal on Gi0/1
*Jan 6 15:07:29.104: RSTP(100): transmitting a proposal on Gi0/1
*Jan 6 15:07:29.105: RSTP(200): transmitting a proposal on Gi0/1
*Jan 6 15:07:29.105: RSTP(300): transmitting a proposal on Gi0/1
D2(config)#

```

For A1:

```

A1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
A1(config)#spanning-tree mode rapid-pvst
A1(config)#

A1#
A1#show spanning-tree

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0c9d.c7a4.0000
             Cost        4
             Port        3 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
             Address     0cbd.e307.0000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time   15 sec

Interface                Role Sts Cost        Prio.Nbr Type
-----
Gi0/0                    Desg FWD 4           128.1   Shr
Gi0/1                    Desg FWD 4           128.2   Shr
Gi0/2                    Root FWD 4           128.3   Shr
Gi0/3                    Altn BLK 4           128.4   Shr

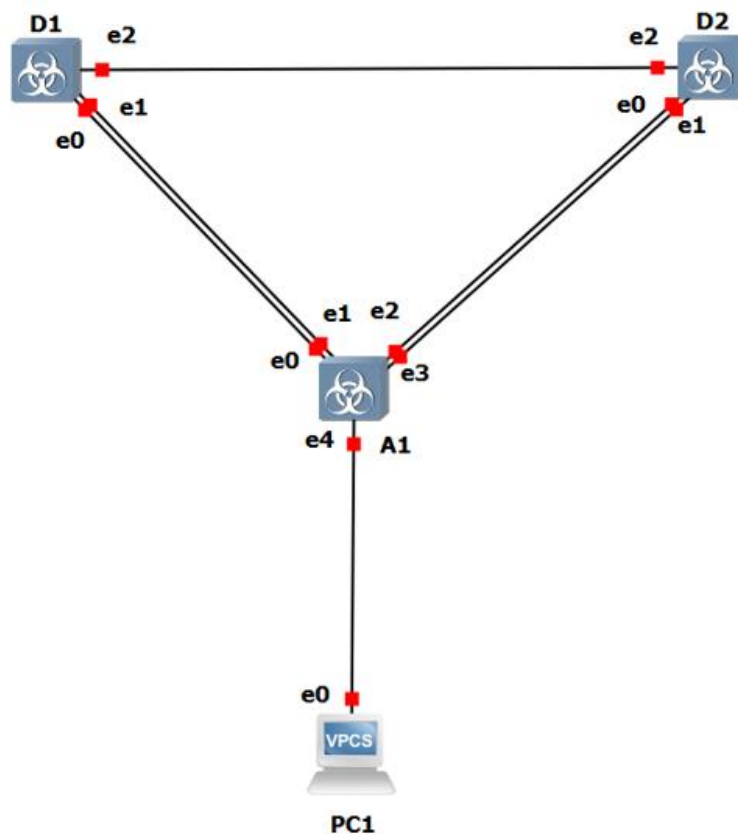
```

Implement MST

What is MST?

- Multiple Spanning Tree Protocol (MSTP), initially defined in IEEE 802.1s and later included in IEEE 802.1Q, supports mapping of multiple VLANs onto a single spanning-tree instance.
- This reduces the number of spanning-tree instances required in a switched network with many VLANs.
- Although RSTP provides faster convergence time than STP does, it still does not solve a problem inherent in STP: all VLANs within a LAN must share the same spanning tree.
- To solve this problem, the QFX Series products use Multiple Spanning Tree Protocol (MSTP) to create a loop-free topology in networks with multiple spanning-tree regions.
- An MSTP region allows a group of bridges to be modeled as a single bridge. An MSTP region contains multiple spanning-tree instances (MSTIs).
- MSTIs provide different paths for different VLANs. This functionality facilitates more efficient load sharing across redundant links.
- An MSTP region can support up to 64 MSTIs, and each instance can support from 1 through 4094 VLANs.

Step 1: Design the network:



Step 2: Configure the switches:**Switch D1:**

```

vIOS-L2-01>
vIOS-L2-01>en
vIOS-L2-01#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vIOS-L2-01(config)#hostname D1
D1(config)#spanning-tree mode rapid-pvst
D1(config)#line con 0
D1(config-line)#exec-timeout 0 0
D1(config-line)#logging synchronous
D1(config-line)#exit
D1(config)#int range gi0/0-2
D1(config-if-range)#switchport trunk encapsulation dot1q
D1(config-if-range)#switchport mode trunk
D1(config-if-range)#no shut
D1(config-if-range)#ex
D1(config)#vlan 2
D1(config-vlan)#name SecondVLAN
D1(config-vlan)#ex
D1(config)#vlan 3
D1(config-vlan)#name ThirdVLAN
D1(config-vlan)#ex
D1(config)#vlan 4
D1(config-vlan)#name FourthVLAN
D1(config-vlan)#ex
D1(config)#vlan 5
D1(config-vlan)#name FifthVLAN
D1(config-vlan)#ex
D1(config)#end
D1#wr
Building configuration...
Compressed configuration from 5147 bytes to 2003 bytes

```

Switch D2:

```

vIOS-L2-01>
vIOS-L2-01>en
vIOS-L2-01#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vIOS-L2-01(config)#hostname D2
D2(config)#spanning-tree mode rapid-pvst
D2(config)#line con 0
D2(config-line)#exec-timeout 0 0
D2(config-line)#logging synchronous
D2(config-line)#exit
D2(config)#int range gi0/0-2
D2(config-if-range)#switchport trunk encapsulation dot1q
D2(config-if-range)#switchport mode trunk
D2(config-if-range)#no shut
D2(config-if-range)#ex
D2(config)#vlan 2
D2(config-vlan)#name SecondVLAN
D2(config-vlan)#ex
D2(config)#vlan 3
D2(config-vlan)#name ThirdVLAN
D2(config-vlan)#ex
D2(config)#vlan 4
D2(config-vlan)#name FourthVLAN
D2(config-vlan)#ex
D2(config)#vlan 5
D2(config-vlan)#name FifthVLAN
D2(config-vlan)#ex
D2(config)#end'
^
% Invalid input detected at '^' marker.

D2(config)#end
D2#wr
Building configuration...
Compressed configuration from 5147 bytes to 2005 bytes[OK]

```

Switch A1:

```

vIOS-L2-01>
vIOS-L2-01>en
vIOS-L2-01#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vIOS-L2-01(config)#hostname A1
A1(config)#spanning-tree mode rapid pvst
^
% Invalid input detected at '^' marker.

A1(config)#spanning-tree mode rapid-pvst
A1(config)#line con 0
A1(config-line)#exec-timeout 0 0
A1(config-line)#logging synchronous
A1(config-line)#exit
A1(config)#int range gi0/0-3
A1(config-if-range)#switchport trunk encapsulation dot1q
A1(config-if-range)#switchport mode trunk
A1(config-if-range)#no shut
A1(config-if-range)#ex
A1(config)#int gi1/0
A1(config-if)#no shut
A1(config-if)#ex
A1(config)#vlan 2
A1(config-vlan)#name SecondVLAN
A1(config-vlan)#ex
A1(config)#vlan 3
A1(config-vlan)#name ThirdVLAN
A1(config-vlan)#ex
A1(config)#vlan 4
A1(config-vlan)#name FourthVLAN
A1(config-vlan)#ex
A1(config)#vlan 5
A1(config-vlan)#name FifthVLAN
A1(config-vlan)#ex
A1(config)#end
A1#wr
Building configuration...
Compressed configuration from 5340 bytes to 2081 bytes
*Oct 31 13:29:26.076: %SYS-5-CONFIG I: Configured from console by console[OK]
*Oct 31 13:29:26.081: %CONF-5-CONF5 WRITING: CONF configuration is being updated on disk.

```

Step 3: Implement and Observe MST.**Configure MST on D1 and D2.**

On D1 and D2, issue the command spanning-tree mode mst.

```

D1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D1(config)#spanning-tree mode mst

D2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D2(config)#spanning-tree mode mst
D2(config)#end
D2#show

```

- At this point, with no MST-specific configuration, MST Instance 0 is operational for all VLANs.
- Issue the command **show spanning-tree** and you will see in the output that the spanning tree information is about MST 0.
- Issue the command **show spanning-tree mst** and you will see the MST-specific STP information that is specific to MST 0 only.

- Take note of the information displayed for interfaces g1/0/5 and g1/0/6 because they are connected to a switch that is not running MST.

```
D1#show spanning-tree

MST0
  Spanning tree enabled protocol mstp
  Root ID    Priority    32768
             Address     0c76.bb6f.0000
             Cost        20000
             Port        3 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768 (priority 32768 sys-id-ext 0)
             Address     0c8f.fc0c.0000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gi0/0                    Altn BKN*20000    128.1 Shr Bound(PVST) *PVST_Inc
Gi0/1                    Altn BLK 20000    128.2 Shr Bound(PVST)
Gi0/2                    Root FWD 20000    128.3 Shr Bound(PVST)

D1#show spanning-tree mst

##### MST0    vlans mapped: 1-4094
Bridge         address 0c8f.fc0c.0000 priority 32768 (32768 sysid 0)
Root           address 0c76.bb6f.0000 priority 32768 (32768 sysid 0)
               port Gi0/2 path cost 20000
Regional Root  this switch
Operational    hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured     hello time 2 , forward delay 15, max age 20, max hops 20

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gi0/0                    Altn BKN*20000    128.1 Shr Bound(PVST) *PVST_Inc
Gi0/1                    Altn BLK 20000    128.2 Shr Bound(PVST)
Gi0/2                    Root FWD 20000    128.3 Shr Bound(PVST)
```

- The basic behaviour of MST is the same as spanning tree, where a root bridge must be selected, then root ports, and finally best paths to the root bridge from all non-root bridges.
- In the current network, we can see that D1 has been elected the root bridge. The bridge priority defaults to 32768, so the election is based on D1 having a lower base MAC address.
- The switch elected as root may be different but the rules for election remain the same.
- Issue the command `show spanning-tree root` on switch A1. Switch A1 is running five instances of spanning tree.

A1:

```
A1#show spanning-tree root

Vlan                Root ID                Root Cost    Hello Time    Max Age    Fwd Dly    Root Port
-----
VLAN0001            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0002            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0003            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0004            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0005            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0100            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0200            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
VLAN0300            32768 0c76.bb6f.0000        4           2           20        15        Gi0/2
```


Issue the show spanning-tree root command on D2 and the output will be different.

This is because with MST, only one instance of the spanning-tree algorithm runs, regardless of the number of VLANs mapped to it.

D2:

```
Oct 31 13:30:49.793: %SYS-5-CONFIG_1: Configured from console by consolespan
D2#show spanning-tree root
```

| MST Instance | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|--------------|----------------------|-----------|------------|---------|---------|-----------|
| MST0 | 32768 0c76.bb6f.0000 | 0 | 2 | 20 | 15 | |

```
D2#
```

Configure A1 to use MST.

```
A1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
A1(config)#spanning-tree mode mst
A1(config)#show spanning-tree mst
```

Step 4: Configure, Tune and Verify Basic MST Operation.

In the last part, you configured all three switches to run MST. In this part, you will further configure, tune, and verify MST to support the unique topological requirements.

Create and verify an MST configuration.

Enter MST configuration mode using the command spanning-tree mst configuration.

```
Oct 31 13:30:34.449: %SPANNTREE-2-PVSTSIM_OK: PVST Simulation Inco
D1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D1(config)#spanning-tree mst configuration
D1(config-mst)#name CCNPv8
```

Configure an MST region name. Our example will be CCNPv8.

```
D1(config-mst)#name CCNPv8
D1(config-mst)#
```

Configure an MST configuration revision number. Our example will be 1.

```
D1(config-mst)#
D1(config-mst)#revision 1
D1(config-mst)#
```

Configure instance 1 to include VLAN 2.

Configure instance 2 to include VLAN 4.

Commit the configuration by typing exit and returning to global configuration mode.

```
D1(config-mst)#instance 1 vlan 2
D1(config-mst)#
D1(config-mst)#instance 2 vlan 4
D1(config-mst)#
D1(config-mst)#ex
D1(config)#end
D1#
Oct 31 13:31:30.221: %SYS-5-CONFETG
```


Issue the command `show spanning-tree mst` to verify the configuration is in place.

```
D1#show spanning-tree mst

##### MST0      vlans mapped: 1,3,5-4094
Bridge          address 0c8f.fc0c.0000 priority      32768 (32768 sysid 0)
Root            address 0c28.f7bd.0000 priority      32768 (32768 sysid 0)
                 port    Gi0/0          path cost    20000
Regional Root   this switch
Operational     hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured      hello time 2 , forward delay 15, max age 20, max hops 20

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Root FWD 20000    128.1   Shr Bound(RSTP)
Gi0/1          Altn BLK 20000    128.2   Shr Bound(RSTP)
Gi0/2          Altn BLK 20000    128.3   Shr Bound(RSTP)

##### MST1      vlans mapped: 2
Bridge          address 0c8f.fc0c.0000 priority      32769 (32768 sysid 1)
Root            this switch for MST1

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Mstr FWD 20000    128.1   Shr Bound(RSTP)
Gi0/1          Altn BLK 20000    128.2   Shr Bound(RSTP)
Gi0/2          Altn BLK 20000    128.3   Shr Bound(RSTP)

##### MST2      vlans mapped: 4
Bridge          address 0c8f.fc0c.0000 priority      32770 (32768 sysid 2)
Root            this switch for MST2

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Mstr FWD 20000    128.1   Shr Bound(RSTP)
Gi0/1          Altn BLK 20000    128.2   Shr Bound(RSTP)
Gi0/2          Altn BLK 20000    128.3   Shr Bound(RSTP)

D1#
```

This configuration does not propagate to other switches. Each switch exchanges digest information summarizing the VLAN-to-Instance mappings it has configured. If a switch receives a BPDU with a different digest, it assumes that the sender is in a different MST region. The output below is what A1 shows in the topology used to create this lab.

```
A1#show spanning-tree mst

##### MST0      vlans mapped: 1-4094
Bridge          address 0c28.f7bd.0000 priority      32768 (32768 sysid 0)
Root            this switch for the CIST
Operational     hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured      hello time 2 , forward delay 15, max age 20, max hops 20

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Desg FWD 20000    128.1   Shr
Gi0/1          Desg FWD 20000    128.2   Shr
Gi0/2          Desg FWD 20000    128.3   Shr
Gi0/3          Desg FWD 20000    128.4   Shr
Gi1/0          Desg FWD 20000    128.5   Shr

A1#show spanning-tree mst configuration digest
A1#show spanning-tree mst configuration digest
Name           []
Revision 0     Instances configured 1
Digest         0xAC36177F50283CD4B83821D8AB26DE62
Pre-std Digest 0xBB3B6C15EF8D089BB55ED10D24DF44DE
A1#
*Oct 31 13:33:27.937: %PLATFORM-5-SIGNATURE VERIFIED: Image 'flash0:/vios l2-
```

Enter digest command in D1:

```

D1#
D1#show spanning-tree mst configuration digest
Name      [CCNPv8]
Revision  1      Instances configured 3
Digest     0x746D865FEAD726D8F401F9396B8862DA
Pre-std Digest  0xDE5D7C8B79A99142EBC0A1C265ED7B05
D1#conf t
Enter configuration commands, one per line. End with CNTL/Z

```

Configure MST on D1, change the revision number to 2 and add VLAN 3 to instance 1 and VLAN 5 to instance 2.

```

D1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D1(config)#spanning-tree mst configuration
D1(config-mst)#show current
Current MST configuration
Name      [CCNPv8]
Revision  1      Instances configured 3

Instance  Vlans mapped
-----
0          1,3,5-4094
1          2
2          4

-----
D1(config-mst)#revision 2
D1(config-mst)#instance 1 vlan 3
D1(config-mst)#instance 2 vlan 5
D1(config-mst)#show pending
Pending MST configuration
Name      [CCNPv8]
Revision  2      Instances configured 3

Instance  Vlans mapped
-----
0          1,6-4094
1          2-3
2          4-5

-----
D1(config-mst)#ex
D1(config)#end
D1#

```

Now configure D2 and A1 with the same configuration settings (name CCNPv8, revision2, instance 1 vlans 2-3, instance 2 vlans 4-5).

After completing the configuration on D2 and A1, the output of show spanning-tree mst on A1 should be similar to the following output.

D2:

```

D2#
D2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D2(config)#spanning-tree mst configuration
*Oct 31 13:33:36.649: %PLATFORM-5-SIGNATURE_VERIFIED: Image 'flash0:/vios_l2-adventerprise9-m' passed code signing verification
D2(config)#spanning-tree mst configuration
D2(config-mst)#name CCNPv8
D2(config-mst)#revision 2
D2(config-mst)#instance 1 vlan 2
D2(config-mst)#instance 2 vlan 4
D2(config-mst)#instance 1 vlan 3
D2(config-mst)#instance 2 vlan 5
D2(config-mst)#ex
D2(config)#end
D2#

```

A1:

```

A1#
A1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
A1(config)#spanning-tree mst configuration
A1(config-mst)#name CCNPv8
A1(config-mst)#revision 2
A1(config-mst)#instance 1 vlan 2
A1(config-mst)#instance 2 vlan 4
A1(config-mst)#instance 1 vlan 3
A1(config-mst)#instance 2 vlan 5
A1(config-mst)#ex
A1(config)#end
A1#
A1#
*Oct 31 13:33:53.551: %SYS-5-CONFIG-I: Configured from console

```

```

Oct 31 13:33:53.551: MST1 2 Conf 10_11 Configured from console by console
A1#show spanning-tree mst
##### MST0    vlans mapped: 1,6-4094
Bridge        address 0c28.f7bd.0000  priority      32768 (32768 sysid 0)
Root          this switch for the CIST
Operational   hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured    hello time 2 , forward delay 15, max age 20, max hops 20

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Desg BLK 20000 128.1  Shr
Gi0/1          Desg BLK 20000 128.2  Shr
Gi0/2          Desg BLK 20000 128.3  Shr
Gi0/3          Desg BLK 20000 128.4  Shr
Gi1/0          Desg BLK 20000 128.5  Shr

##### MST1    vlans mapped: 2-3
Bridge        address 0c28.f7bd.0000  priority      32769 (32768 sysid 1)
Root          this switch for MST1

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Desg BLK 20000 128.1  Shr
Gi0/1          Desg BLK 20000 128.2  Shr
Gi0/2          Desg BLK 20000 128.3  Shr
Gi0/3          Desg BLK 20000 128.4  Shr

##### MST2    vlans mapped: 4-5
Bridge        address 0c28.f7bd.0000  priority      32770 (32768 sysid 2)
Root          this switch for MST2

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Desg BLK 20000 128.1  Shr
Gi0/1          Desg BLK 20000 128.2  Shr
Gi0/2          Desg BLK 20000 128.3  Shr
Gi0/3          Desg BLK 20000 128.4  Shr

```

Controlling the Root Bridge.

- Just like with PVST+ and Rapid PVST+, the current root bridge was elected based on the lowest Bridge ID (consisting of the Priority, extended system ID equal to the VLAN ID, and base MAC address values).
- With PVST+ or Rapid PVST+, root bridge selection is done for each VLAN. With MST, the root bridge is based on instances.

- There are two basic ways to manipulate the configuration to control the location of the root bridge:
 - The spanning-tree mst instance-id priority value command can be used to manually set a priority value.
 - The spanning-tree mst instance-id root { primary | secondary } command can be used to automatically set a priority value.
- You will need to make configuration changes on both D1 and D2. The commands used at D1:

D1:

```
D1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D1(config)#spanning-tree mst root primary
      ^
% Invalid input detected at '^' marker.

D1(config)#spanning-tree mst 1 root primary
D1(config)#
D1(config)#spanning-tree mst 2 root secondary
D1(config)#
D1(config)#
```

After you have configured both D1 and D2, go to A1 and issue the command `show spanning-tree root`.

In this output, you will see the root bridges differentiated.

```
A1#show spanning-tree root
```

| MST Instance | Root ID | Root Cost | Hello Time | Max Age | Fwd Dly | Root Port |
|--------------|----------------------|-----------|------------|---------|---------|-----------|
| MST0 | 32768 0c28.f7bd.0000 | 0 | 2 | 20 | 15 | |
| MST1 | 24577 0c8f.fc0c.0000 | 20000 | 2 | 20 | 15 | Gi0/0 |
| MST2 | 28674 0c8f.fc0c.0000 | 20000 | 2 | 20 | 15 | Gi0/0 |

```
A1#show spanning-tree mst 1
```

On A1, issue the commands `show spanning-tree vlan 1` and `show spanning-tree blockedports`.

```
MST2      28674 0c8f.fc0c.0000      20000      2      20      15      Gi0/0
A1#show spanning-tree mst 1

#### MST1      vlans mapped: 2-3
Bridge      address 0c28.f7bd.0000      priority      32769 (32768 sysid 1)
Root        address 0c8f.fc0c.0000      priority      24577 (24576 sysid 1)
            port    Gi0/0              cost          20000      rem hops 19

Interface    Role Sts Cost      Prio.Nbr Type
-----
Gi0/0        Root FWD 20000     128.1    Shr
Gi0/1        Altn BLK 20000     128.2    Shr
Gi0/2        Desg FWD 20000     128.3    Shr
Gi0/3        Desg FWD 20000     128.4    Shr

A1#show spanning-tree blockedports
      ^
% Invalid input detected at '^' marker.

A1#show spanning-tree blockedports

Name          Blocked Interfaces List
-----
MST1          Gi0/1
MST2          Gi0/1

Number of blocked ports (segments) in the system : 2
```


On A1, shutdown interfaces F0/1 and F0/2, assign a new port cost of 1000 to F0/2 using the spanning-tree mst 1 cost value command, and then issue the no shutdown command on the ports.

```
A1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
A1(config)#int range gi0/0-1
A1(config-if-range)#shut
A1(config-if-range)#exit
A1(config)#int
*Oct 31 13:35:26.551: %LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to administratively down
*Oct 31 13:35:26.580: %LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to administratively down
*Oct 31 13:35:27.551: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to down
A1(config)#int
*Oct 31 13:35:27.580: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to down
A1(config)#int gi0/1
A1(config-if)#spanning-tree mst 1 cost 100
A1(config-if)#ex
A1(config)#int range gi0/0-1
A1(config-if-range)#no shut
A1(config-if-range)#ex
A1(config)#
*Oct 31 13:35:42.951: %LINK-3-UPDOWN: Interface GigabitEthernet0/0, changed state to up
*Oct 31 13:35:42.979: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
*Oct 31 13:35:43.951: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
A1(config)#
*Oct 31 13:35:43.979: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
A1(config)#end
```

Now verify that this impacts root port selection on A1 using the show spanning-tree vlan 1 and show spanning-tree blockedports commands.

```
*Oct 31 13:35:50.921: %SYS-5-CONFIG_1: Configured from console by console
A1#show spanning-tree mst 1

##### MST1    vlans mapped: 2-3
Bridge        address 0c28.f7bd.0000 priority 32769 (32768 sysid 1)
Root          address 0c8f.fc0c.0000 priority 24577 (24576 sysid 1)
              port    Gi0/1          cost    100      rem hops 19

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Altn BLK 20000    128.1   Shr
Gi0/1          Root FWD 100       128.2   Shr
Gi0/2          Desg BLK 20000    128.3   Shr
Gi0/3          Desg BLK 20000    128.4   Shr

A1#show spanning-tree blockedports

Name           Blocked Interfaces List
-----
MST0           Gi0/0, Gi0/1
MST1           Gi0/0, Gi0/2, Gi0/3
MST2           Gi0/1, Gi0/2, Gi0/3

Number of blocked ports (segments) in the system : 8
```

The port priority can be any value between 0 and 240, in increments of 16 (older switches may allow setting the priority in different increments).

On A1, issue the command show spanning-tree mst 2 and take note of the port ID values listed.

```
A1#show spanning-tree mst 2

##### MST2    vlans mapped: 4-5
Bridge        address 0c28.f7bd.0000 priority 32770 (32768 sysid 2)
Root          address 0c8f.fc0c.0000 priority 28674 (28672 sysid 2)
              port    Gi0/0          cost    20000    rem hops 19

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Root FWD 20000    128.1   Shr
Gi0/1          Altn BLK 20000    128.2   Shr
Gi0/2          Desg FWD 20000    128.3   Shr
Gi0/3          Desg FWD 20000    128.4   Shr
```

Modify the port priority of D2 interface G1/0/6 so that it becomes the preferred port by issuing the spanning-tree mst 2 port-priority value interface configuration command. Use a value of 64.

```

Oct 31 13:37:23.478: %LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to administratively down
D2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
D2(config)#int range gi0/0-1
D2(config-if-range)#shut
D2(config-if-range)#ex
D2(config)#int gi0/1
*Oct 31 13:37:23.511: %LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to administratively down
*Oct 31 13:37:24.478: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to down
D2(config)#int gi0/1
*Oct 31 13:37:24.511: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to down
D2(config)#int gi0/1
D2(config-if)#spanning-tree mst 2 port-priority 64
D2(config-if)#ex
D2(config)#int range gi0/0-1
D2(config-if-range)#no shut
D2(config-if-range)#ex
D2(config)#

```

On A1, issue the show spanning-tree mst 2 command and you will see that Fa0/4 is now the selected root port.

```

A1#show spanning-tree mst 2

##### MST2      vlans mapped: 4-5
Bridge          address 0c28.f7bd.0000  priority      32770 (32768 sysid 2)
Root            address 0c8f.fc0c.0000  priority      28674 (28672 sysid 2)
                port    Gi0/0          cost         20000      rem hops 19

Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi0/0          Root FWD 20000     128.1   Shr
Gi0/1          Altn BLK 20000     128.2   Shr
Gi0/2          Desg FWD 20000     128.3   Shr
Gi0/3          Desg FWD 20000     128.4   Shr
A1#

```