Contents

1 Configuring MSTP

1.1 Introduction

1.1.1 Overview

The Spanning Tree Protocol (STP) is an L2 management protocol used to eliminate L2 loops by blocking redundant links in the network. It enables a backup link in the case of a link failure.

With the network development and update, multiple STP versions become available. This device supports STP, Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP).

1.1.2 Background and Functions

1. 802.1D STP

To prevent single point of failures (SPOFs), L2 devices enhance line reliability by adding redundant lines. One of the most economical ways is to add loops. However, loops may cause broadcast storms and Media Access Control (MAC) address flapping. Therefore, the STP protocol arises.

STP can generate the best loop-free tree topology for a local area network (LAN) via computation. If detecting a fault, STP automatically eliminates the fault and updates the topology.

STP cannot enter forwarding state quickly but needs 50 seconds to complete topology convergence. All virtual local area networks (VLANs) in an LAN share one spanning tree. Packets of all the VLANs are forwarded along this spanning tree, and the bandwidth of redundant lines is wasted. VLAN data may be isolated at L2, and L2 loops do not exist in some circumstances. However, STP cannot recognize VLANs, and packets from interfaces in the same VLAN may be intercepted by blocking ports, resulting in a communication failure.

2. 802.1w RSTP

RSTP is fully compatible with STP backward. Apart from the conventional STP functions of eliminating loops and enabling backup links, RSTP solves the problem of entering forwarding state rapidly. If all bridges in a LAN support RSTP and are properly configured, once the network topology changes, RSTP takes only less than 1s to regenerate a topology tree.

Although RSTP can complete convergence quickly, it cannot identify VLANs just as STP does not.

3. 802.1s MSTP

MSTP resolves defects of STP and RSTP. It not only supports fast convergence but also forwards traffic of different VLANs along their respective paths, thereby providing a better load balancing mechanism for redundant links. STP/RSTP works based on ports while MSTP works based on instances. An instance is a collection of multiple VLANs. Binding multiple VLANs to one instance can reduce communication costs and improve resource usage. MSTP can implement the following functions:

- When lines are normal, MSTP can eliminate loops and prevent broadcast storms and MAC address flapping, to ensure normal L2 communication.
- When a line is faulty, MSTP enables a backup link to enhance the link reliability. It can complete

convergence rapidly to enhance the link availability.

- MSTP can produce multiple spanning trees by VLAN instance to implement traffic balancing.

1.1.3 STP

1. Overview

STP, defined by the Institute of Electrical and Electronics Engineers (IEEE) 802.1D standard, is used to break physical loops at the data link layer in a LAN and prevent broadcast storms, so as to implement link redundancy.

L2 devices connected using single lines have no redundant lines and devices. Any SPOF can cause device disconnection and the line reliability is low. When a redundant line is added, an L2 loop will form and the SPOF can be eliminated.

The adding of a redundant line brings two new problems:

- Once a broadcast frame appears in an L2 loop, it will be flooded continuously by L2 devices and a broadcast storm will be incurred, which will heavily consume device resources and network bandwidth.
- In a loop topology, an L2 device may receive packets from the same source MAC address through two ports of two paths, causing MAC address flapping and the failure of the device to work properly.

Therefore, for L2 Ethernet, only one valid channel is required to forward data between two LANs. Ideally, when an L2 loop exists on a network, a specific port can be blocked to invalidate the redundant link so as to break the loop; when a network fault occurs, the specific port can be enabled to restore a redundant link to the valid state. It is difficult to attain this ideal effect via manual control. Therefore, the STP protocol emerges to automatically complete this work:

- Devices with STP enabled exchange specific STP Protocol Packets, that is, bridge protocol data units (BPDUs), and elect Device Roles (root bridge and non-root bridge) and Port Roles (root port, designated port, and blocking port) based on Key Information of BPDUs. Different port roles are in different Port States and have different functions. They form a Generating a Spanning Tree Topology in a LAN.
- Protocol packets are used to monitor the line status to Maintaining a Spanning Tree Topology. Once a fault is found, STP rectifies the fault and automatically Updating a Spanning Tree Topology.

2. STP Protocol Packets

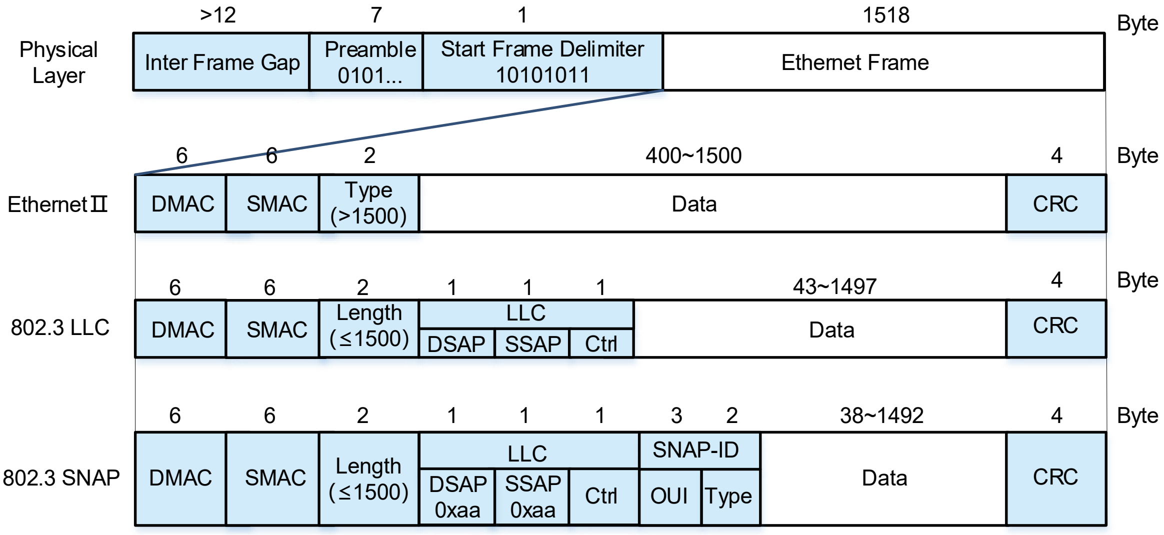
STP uses two types of protocol packets: configuration BPDU and topology change notification (TCN) BPDU.

- Configuration BPDU: A configuration BPDU contains 35 bytes, which provide information required for calculating the spanning tree topology (root bridge ID, root path cost, designated bridge ID, and designated port ID) and information required for maintaining the spanning tree (**Message Age**, **Max Age**, **Hello Time**, and **Forward Delay**). When STP is enabled on a designated port, the port sends configuration BPDUs at an interval of **Hello Time**. When a device receives a configuration BPDU from a root port, it sends the configuration BPDU to each designated port on the device, and the designated ports generate their own BPDUs. If the configuration BPDU received by a designated port from the line has a lower priority than the BPDU generated by the designated port itself (see Key Information of BPDUs), the designated port sends its BPDU to the downstream device.
- TCN BPDU: A TCN BPDU contains four bytes. On a bridge with at least one designated port, when a port on the bridge transitions to forwarding or blocking state, the network topology has changed. The bridge will send a TCN BPDU to inform the upstream device of the network topology change. When the upstream

device receives the TCN BPDU through a designated port, it sets TCA in the configuration BPDU to 1 and instructs the downstream device to stop sending the TCN BPDU. The upstream device also sends a copy of the TCN BPDU to its upstream device until the TCN BPDU reaches the root bridge. By setting the TC flag bit in the configuration BPDU to 1, the root bridge informs all bridges in the spanning tree of the network topology change, and asks them to clear dynamic MAC address entries of ports and relearn MAC addresses. A shorter MAC address aging time facilitates fast convergence of the topology.

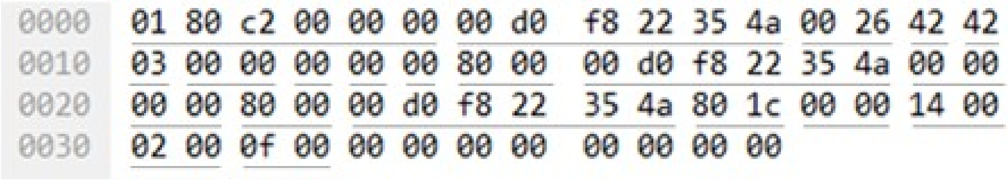
BPDUs are encapsulated in IEEE 802.3 logical link control (LLC) Ethernet frames. The format differences between BPDUs and other Ethernet frames are as follows.

Figure 1-1Formats of Ethernet Frames



An actual STP configuration BPDU is taken as an example for description.

Figure 1-2STP Configuration BPDU



```
0000  01 80 c2 00 00 00 00 d0  f8 22 35 4a 00 26 42 42
0010  03 00 00 00 00 00 80 00  00 d0 f8 22 35 4a 00 00
0020  00 00 80 00 00 d0 f8 22  35 4a 80 1c 00 00 14 00
0030  02 00 0f 00 00 00 00 00  00 00 00 00
```

The header of an IEEE 802.3 Ethernet frame contains 14 bytes and the fields in the header are described in Table 1-1.

Table 1-1Description of Fields in the Header of an IEEE 802.3 Ethernet Frame

Field	Bytes	Description
DMAC	6	Destination MAC address. According to the protocol, the destination address of the Ethernet frame used for encapsulating a BPDU is fixed to the multicast address 0180.c200.0000. A device receiving a frame with such a destination address will not forward the frame.

Field	Bytes	Description
SMAC	6	Source MAC address of the device sending the BPDU. The value in the example is 00-d0-f8-22-35-4a.
Length	2	The value is 0x0026, indicating that the packet length is 38 bytes (3 bytes of the LLC header and 35 bytes of the BPDU).

The LLC header occupies three bytes, which are described in Table 1-2.

Table 1-2Description of Fields in the LLC Header of an Ethernet Frame

Field	Bytes	Description
DSAP	1	Destination service access point. The value is 0x42.
SSAP	1	Source service access point. The value is 0x42.
Control	1	This field stores various control information and identifies the type of an LLC frame. The value is 0x03.

The bytes behind the LLC header is a BPDU. There are four types of BPDUs for STP, RSTP, and MSTP, which are differentiated using different PVIDs and BPDU type values.

A configuration BPDU of STP occupies 35 bytes, which are described in Table 1-3. The last part is an 8-byte filling field and a 4-byte CRC field (not shown in the figure). They form a 64-byte Ethernet frame.

Table 1-3Format of an STP Configuration BPDU

Field	Bytes	Description
PID	2	Protocol identifier. The value is 0x0000.
PVID	1	Protocol version identifier. 0x00 indicates an STP packet.
BPDU Type	1	BPDU packet type. 0x00 indicates a configuration BPDU of STP.
Flags	1	The value is 0x00 in the example. Configuration BPDUs of STP use only bit 7 and bit 0 and other bits are not used. ● Bit 7: Topology change acknowledgment (TCA). An upstream device sets bit 7 in a configuration BPDU to 1 and sends the BPDU to notify the downstream device of the topology change and requests the downstream device to stop sending the TCN BPDU. ● Bit 0: Topology change (TC). The root bridge sets bit 0 in a configuration BPDU to 1 and sends the BPDU to instruct the downstream device to directly delete MAC address entries of the bridges.
Root ID	8	Identifier of the root bridge recognized by the device. It consists of a 2-byte bridge priority and a 6-byte bridge MAC address.

Field	Bytes	Description
		In the example, the priority value is 0x8000, which indicates 32768, and the bridge MAC address is 00-d0-f8-22-35-4a.
RPC	4	Root path cost, which is the cumulative cost from the port to the root bridge. The value is 0x00-00-00-00 in the example.
Bridge ID	8	Bridge identifier of the device. It consists of a 2-byte bridge priority and a 6-byte bridge MAC address. In the example, the priority value is 0x8000, which indicates 32768, and the bridge MAC address is 00-d0-f8-22-35-4a.
Port ID	2	Identifier of the port that sends the BPDU. It consists of a 1-byte priority and 1-byte port ID. In the example, the priority value is 0x80, which indicates 128, and the port ID is 1c.
Message Age	2	Age of the BPDU, which represents the time that the BPDU has been alive in the network. The value is 0x0000 in the example.
Max Age	2	Aging time of the BPDU. If a non-root bridge fails to receive a BPDU from the root bridge within the time of **Max Age**, the root bridge or the link to the root bridge is deemed as faulty. The value in the example is 0x1400 (low-order bytes on the left), indicating 20 seconds.
Hello Time	2	Interval for sending two adjacent BPDUs. The root bridge sends configuration BPDUs at an interval of **Hello Timer**. A non-root switch sends a configuration BPDU only after receiving the configuration BPDU from the upstream device. The value in the example is 0x0200 (low-order bytes on the left), indicating 2 seconds.
FWD Delay	2	Forwarding delay. This time parameter controls the duration of listening and learning states. The value in the example is 0x0f00 (low-order bytes on the left), indicating 15 seconds.

A TCN BPDU consists of four bytes and contains only the **PID**, **PVID**, and **BPDU Type** fields. The packet format is described in Table 1-4.

Table 1-4Format of an STP TCN BPDU

Field	Bytes	Description
PID	2	Protocol identifier. The value is 0x0000.
PVID	1	Protocol version identifier. 0x00 indicates an STP packet.
BPDU Type	1	BPDU packet type.

Field	Bytes	Description
		0x80 indicates a TCN BPDU of STP. When the downstream network topology changes, it is used to notify the upstream device of the network topology change. After receiving a TCN BPDU, a port copies the BPDU and forwards it to upstream devices till the root device receives the BPDU.

3. Key Information of BPDUs

- Priority vector

  Bridges exchange BPDUs to obtain information and generate a stable optimal tree topology. The most basic elements in the information are listed as follows:

  - Root ID: ID of the root bridge recognized by this local bridge. It consists of the bridge priority and MAC address of the root. The root bridge ID is the bridge ID of the root.
  - Root Path Cost: Path cost from this bridge to the root bridge.
  - Bridge ID: Each bridge has a unique bridge ID consisting of the bridge priority and MAC address.
  - Port ID: ID of each port, which consists of the port priority and port ID.

  These key elements constitute a priority vector <**Root Identifier**, **Root Path Cost**, **Bridge ID**, **Port ID**>. Bridges compare elements in the priority vector in sequence to determine the priority of BPDUs. A smaller priority element value indicates a higher priority. If a port of a bridge receives a BPDU with the priority higher than the BPDU priority of the port, the port saves and transmits the BPDU. If it receives a BPDU with the priority lower than the BPDU priority of the port, in case of STP, it discards the BPDU, but in case of RSTP or MSTP, it will send a Proposal BPDU to request the peer to recognize the local device as the upstream device. This mechanism ensures that high-priority BPDUs are transmitted throughout the network.

- Bridge ID

  According to IEEE 802.1W, a root ID is used to identify a root bridge in a spanning tree while a bridge ID is used to identify each bridge. When a device just runs STP, it uses its bridge ID as the root ID and STP compares root IDs to elect the root bridge according to algorithms. A bridge ID consists of eight bytes, with the first two bytes of the bridge priority and the last six bytes of the bridge MAC address. A device compares the bridge priority first and then the MAC address during root bridge selection. A smaller value indicates a higher priority.

  Bridge priority: To ensure network reliability, you are advised to specify bridge priorities during network planning, to prevent devices from electing the root bridge by comparing MAC addresses. In the 2-byte bridge priority, bit 1 to bit 12 indicate the system ID used for protocol extension and the value is **0** in RSTP. Bit 13 to bit 16 indicate the priority, which can be configured. If the configured priority is 0001 0000 0000 0000 (**4096** in decimal notation), the bridge priority is a multiple of 4096. Bit 13 to bit 16 can indicate 15 numbers from 0000 to 1111. Therefore, the priority is 4096 × 0 to 4096 × 15 in decimal notation.

Table 1-1Bridge Priority Field (Two Bytes)

	Priority				System ID											
Bit	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Value	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

- Port ID

  Each port participating in the spanning tree calculation must have a port ID, which is the port ID of a device.

  Format: A port ID consists of two bytes, with the first byte of the port priority and the second byte of the port ID.

  The default port priority is **128** and the port priority range is from 0 to 255. The port ID range is from 0 to 255, for example, the ID (0) of port GigabitEthernet 0/0 is smaller than that (1) of port GigabitEthernet 0/1.

- Root path cost

  Root path cost is the cumulative path cost from the port to the root bridge. A link between two directly connected ports is called a "segment". A segment with a higher network bandwidth has a lower path cost. The total cost calculated based on the path costs of all ports in the necessary lines from a port to the root is the cumulative path cost from the port to the root bridge. Table 1-2 shows the cost values supported by different protocols. Path cost values defined for the device are provided in 1.6 Table 1-1.

Table 1-2Path Costs Supported by Protocols

Bandwidth	RPC Supported by IEEE 802.1t	RPC Supported by IEEE 802.1d
10 Mbps	2000000	100
100 Mbps	200000	19
1 Gbps	20000	4
10 Gbps	2000	2

4. Device Roles

Devices with STP enabled exchange BPDUs and compare root IDs to elect device roles.

- Root bridge: Each spanning tree has a unique root.
- Designated bridge: A device in the shortest path between the local device and the root bridge, that is, the upstream device of the local device.
- Bridge: Generally the local device, that is, downstream device of a designated bridge. Designated bridges and bridges can be collectively referred to as non-root bridges.

5. Port Roles

The root bridge and non-root bridges elect different port roles by exchanging BPDUs.

- Root port: A port with the shortest path from a non-root bridge to the root bridge.

- Designated port: Each port on the root bridge is a designated port. The root bridge connects to a root port of a non-root bridge through a designated port. The port that connects a designated bridge to a downstream device is also a designated port.
- Disable port: Also called a blocking port. A disable port is a port in blocked or down state on a non-root bridge.

6. Port States

Ports work in different states and perform different functions according to the protocol. Table 1-1 lists functions corresponding to port states.

STP defines five port states: disabled, blocking, listening, learning, and forwarding. RSTP reduces port states to only three states: discarding, learning, and forwarding. MSTP uses the port states of RSTP.

Table 1-1Port Roles, Port States, and Functions Supported by STP, RSTP, and MSTP

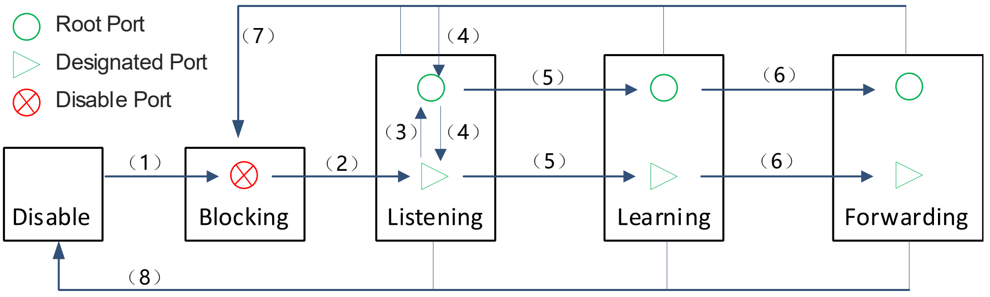
Note: In this table, R = root port, D = designated port, A = alternate port, B = backup port, BLK = blocking port.

STP		RSTP and MSTP		Function				
Port Role	**Port State**	**Port Role**	**Port State**	**Receiving BPDUs**	**Sending BPDUs**	**MAC Address Learning**	**Receiving Data**	**Forwarding Data**
Disable	Disabled	Disable	Discarding	No	No	No	No	No
Blk	Blocking	A, B	Discarding	Blk, A, B	No	No	No	No
R, D	Listening	R, D	Discarding	R, D	D	No	No	No
	Learning		Learning	R, D	D	R, D	No	No
	Forwarding		Forwarding	R, D	D	R, D	R, D	R, D

After a topology becomes stable, only root ports and designated ports are in forwarding state and can send and receive data. Other ports are in blocking state and can only receive BPDUs. Different port roles have different functions in the same port state. For example, a root port cannot send but can receive BPDUs in forwarding state, while a designated port can send and receive BPDUs in forwarding state.

The port states can be transitioned according to the rules shown in Figure 1-2.

Figure 1-2State Transition Rules of STP Ports



(1) After being initialized or enabled, a port enters blocking state.

(2) When STP is just running on a device, all the ports on the device enter listening state as designated ports, and send and receive BPDUs for election. After STP runs stably, if a blocking port fails to receive a BPDU within **MAX Age**, it becomes a designated port, transitions from blocking state to listening state, and receives and sends BPDUs for re-election.

(3) After election, if a port is elected as a root port, it only receives BPDUs but cannot send BPDUs.

(4) If a root port fails to receive a BPDU within **MAX Age**, the root is considered to be faulty, and the root port becomes a designated port and sends a BPDU.

(5) After a port is elected as a root port or designated port, it waits for one period of **Forward Delay**, transitions from listening state to learning state, and starts MAC address learning.

(6) Then, the port waits for another period of **Forward Delay**, transitions from learning state to forwarding state, and starts sending and receiving BPDUs.

(7) If the port is no longer a root port or designated port, it returns to blocking state.

(8) If a port is disabled or the link is faulty, the port returns to disable state.

7. Generating a Spanning Tree Topology

STP automatically calculates and generates a spanning tree topology for a LAN according to a set of bridge parameters configured by the administrator. Only appropriate configuration can ensure the fast generation of a stable topology. The steps for STP to generate a spanning tree topology are as follows:

(1) When a network starts to run STP, all switches consider themselves as root bridges and all ports on the switches are considered as designated ports. So, all ports enter listening state, broadcast configuration BPDUs, and receive BPDUs from the network. At this time, the root ID in the BPDU of each switch is the bridge ID of the switch.

(2) Each broadcast domain selects one root bridge. Each switch receives BPDUs from other switches and each BPDU contains the priority vector <**Root ID**, **Root Path Cost**, **Bridge ID**, **Port ID**>. According to the rule that a smaller value indicates a higher priority, each switch compares root IDs (bridge ID of each switch at first) in BPDUs. When a port of a switch receives a smaller root ID, the root ID is updated in the **Root ID** field in the configuration BPDUs stored in all ports of the switch and then sent to other switches. After interaction, the bridge ID of the device with the highest priority is recorded as the root ID, indicating that the device is elected as the root bridge.

(3) All ports on the root bridge are designated ports. After waiting for one period of **Forward Delay**, designated ports transition from listening state to learning state to learn MAC addresses. After waiting for another period of **Forward Delay**, the designated ports transition from learning state to forwarding state and start receiving and sending packets.

(4) Each non-root bridge compares the cumulative path cost from each port to the root and elects the port with the smallest cost as the root port. Root ports only receive configuration BPDUs from designated ports and do not send configuration BPDUs. After waiting for one period of **Forward Delay**, root ports transition from listening state to learning state to learn MAC addresses. After waiting for another period of **Forward Delay**, root ports transition from learning state to forwarding state and start sending and receiving packets.

(5) A link between two directly connected switches is called a "segment". The priority vector <**Root ID**, **Root Path Cost**, **Bridge ID**, **Port ID**> is compared in sequence to select a designated port for each segment. After waiting for one period of **Forward Delay**, designated ports transition from listening state to learning

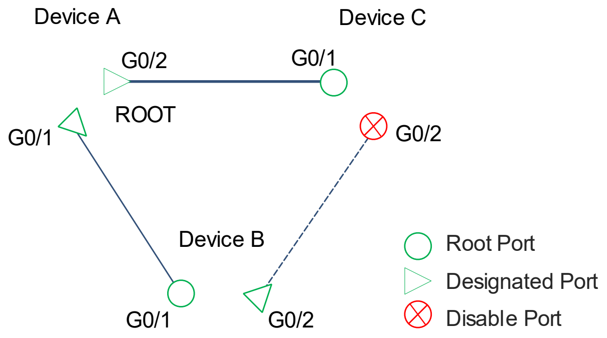
state to learn MAC addresses. After waiting for another period of **Forward Delay**, the designated ports transition from learning state to forwarding state and start receiving and sending packets.

(6) The other ports are non-designated ports and are blocked. Such ports return to blocking state and only receive BPDUs. If a blocking port receives a high-priority BPDU within **MAX Age**, the current line from the peer device to the root is better and the blocking port remains in blocking state. If a blocking port fails to receive a high-priority BPDU within **MAX Age**, the current line from the peer device to the root is worse or even faulty. The blocking port transitions to listening state and sends its configuration BPDU to inform the peer device that the communication line passing through this port is better.

Note

- In STP and RSTP, the same device has the same bridge ID and the same port has the same port ID.
- In MSTP, different bridge IDs can be configured for the same device and different port IDs can be configured for the same port by instance.

Figure 1-1Basic Topology of STP



8. Maintaining a Spanning Tree Topology

After a spanning tree topology becomes stable, only designated ports and root ports are in forwarding state while other ports are in blocking state. The root bridge continues to send configuration BPDUs at an interval of **Hello Time**. A downstream non-root bridge stores the optimal configuration BPDU received from the root port in the device, fills the bridge ID of the root bridge in the root ID, and then sends the configuration BPDU to the downstream device through a designated port. If a downstream device receives a higher-priority BPDU from the upstream device within one period of **Max Age**, the network topology remains unchanged.

9. Updating a Spanning Tree Topology

Taking the topology shown in Figure 1-1 as an example, this section describes the update process of the spanning tree topology. To simplify the description, the priority vector is <RID, Cost, BID, PID >, BIDs of Device A, Device B, and Device C are 1, 2, and 3 respectively. PIDs of ports GigabitEthernet 0/1 and GigabitEthernet 0/2 are 1 and 2 respectively. The lines are 1000M links and the path cost of each segment of line is 4.

- Device B and Device C consider themselves as root devices if the following situation occurs: Root bridge A malfunctions, Device B fails to receive BPDU<1,0,1,1> from the root bridge, and Device C fails to receive BPDU<1,0,1,2> from the root bridge within one period of **Max Age**. Then, port GigabitEthernet 0/2 of

Device B sends BPDU<2,0,2,2>. Port GigabitEthernet 0/2 of Device C becomes a designated port and sends BPDU<3,0,3,2>. <2,0,2,2> has a higher priority than <3,0,3,2>. Therefore, Device B is elected as the root bridge and port GigabitEthernet 0/2 of Device C becomes a root port.

- Device C deems that root bridge A is faulty and considers itself as the root bridge if the following situation occurs: Port GigabitEthernet 0/2 of root bridge A or the link from root bridge A to Device C is faulty and downstream Device C fails to receive BPDU<1,0,1,2> from root bridge A through the root port GigabitEthernet 0/1 within one period of **Max Age**. Blocking port GigabitEthernet 0/2 (that can only receive BPDUs but cannot send BPDUs) of Device C becomes a designated port, transitions from blocking state to listening port, and sends BPDU<3,0,3,2>. Port GigabitEthernet 0/2 of Device C receives BPDU<1,4,2,2> from designated port GigabitEthernet 0/2 of Device B and finds that the root ID has a higher priority than the root ID in Device C. Device C considers Device A as the root bridge again, updates the spanning tree topology, and sets port GigabitEthernet 0/2 as the root port. After waiting for one period of **Forward Delay**, port GigabitEthernet 0/2 transitions to learning state to learn MAC addresses. After waiting for another period of **Forward Delay**, port GigabitEthernet 0/2 transitions to forwarding state to forward data, implementing link auto-recovery. In this case, the link recovery needs **Max Age** + 2 × FWD time. The default link recovery time is 50s, that is, 20 + 2 × 15 = 50.
- Device B deems that root bridge A is faulty and considers itself as the root bridge if the following situation occurs: Port GigabitEthernet 0/1 of Device A or the link from root bridge A to Device B is faulty and downstream Device B fails to receive BPDU<1,0,1,1> from root bridge A through the root port GigabitEthernet 0/1 within one period of **Max Age**. The BPDU sent by port GigabitEthernet 0/2 of Device B changes from <1,4,2,2> (Device A functions properly) to <2,0,2,2> (Device A is faulty). Blocking port GigabitEthernet 0/2 of Device C finds that the BPDU sent from port GigabitEthernet 0/2 of Device B has a lower priority than the configuration BPDU<1,4,3,2> stored by the blocking port itself, and discards the BPDU sent from Device B. If blocking port GigabitEthernet 0/2 of Device C fails to receive a BPDU with the priority higher than that of the BPDU stored in the blocking port within one period of **Max Age**, the blocking port becomes a designated port and sends its stored BPDU<1,4,3,2> to Device B. Device B find that BPDU<1,4,3,2> has a higher priority than BPDU<2,0,2,2> stored in port GigabitEthernet 0/2 of Device B, deems that Device A is the root bridge, and sets port GigabitEthernet 0/2 as the root port. After two periods of **Forward Delay** elapse, port GigabitEthernet 0/2 of Device B transitions to forwarding state to forward data, implementing link auto-recovery in the STP network.
- If a port on a downstream device changes to down state, the device sends a TCN BPDU to the upstream device to notify the upstream device of the topology change. After receiving the TCN BPDU from the downstream device, the upstream device sets bit 7 (indicating TCA) in the **Flags** field of the configuration BPDU to **1**, and sends the configuration BPDU to the downstream device, indicating that the upstream device has known the topology change and requesting the downstream device to stop sending the TCN BPDU. Only the designated port of the upstream device processes TCN BPDUs. Non-designated ports discard TCN BPDUs directly once they receive such BPDUs. The upstream device copies the TCN BPDU and forwards it to its upstream device through the root port. The TCN BPDU is forwarded segment by segment until it reaches the root bridge. After receiving the TCN BPDU from a downstream device, the root bridge sets bit 7 (indicating TCA) in the **Flags** field of the configuration BPDU to **1** and sends the configuration BPDU to the downstream device, indicating that the upstream device has known the topology change and requesting the downstream device to stop sending the TCN BPDU. In addition, the root bridge sets bit 0 (indicating a TC) in the **Flags** field of the configuration BPDU to **1** to ask all downstream devices to clear dynamic MAC address entries of ports and relearn MAC addresses due to the topology change.

- If a new Device D is connected to Device B and the port status of Device B becomes up, Device B will also notify others of the topology change. Device D will send and receive BPDUs. If Device D has a higher priority than the original root (Device A), great changes may be incurred to the spanning tree. In conclusion, STP is not stable.

10. Drawbacks of STP

- Topology convergence is slow. After port role election is complete, STP waits for twice the period of **Forward Delay** (15s by default) before entering forwarding state, namely, STP needs to wait 30 seconds (that is, 2 × 15). Every time the topology changes, a bridge needs to re-elect the root port and designated ports after the period of **MAX Age** (20s by default), and then wait for twice the period of **Forward Delay** before entering forwarding state. Therefore, it takes about 50s (that is, 20 + 2 × 15 = 50 seconds) to generate a stable topology. STP requires the root bridge to actively send configuration BPDUs and other devices to forward the BPDUs. This mechanism leads to slow network convergence.
- If the network topology changes frequently, the network will be frequently disconnected, which is unacceptable to users.

1.1.4 RSTP

1. Overview

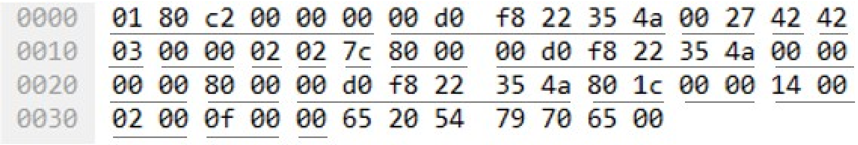
RSTP, defined by IEEE 802.1W, evolves from STP and is completely compatible with IEEE 802.1D STP downward. RSTP has all functions of the conventional STP protocol and is capable of preventing loops and providing redundant links. Compared with STP, RSTP shortens the time required for network topology convergence. If all the bridges in a LAN support RSTP and are configured properly, once the network topology changes, the time required by RSTP to re-generate the topology tree is less than 1s while STP needs about 50s.

2. RSTP Protocol Packets

RSTP needs only one type of protocol packet: configuration BPDU. The packet format is slightly different from that in STP. The configuration BPDU of RSTP is also called rapid spanning tree BPDU (RST BPDU).

When a topology changes, RSTP no longer uses a TCN BPDU to notify the topology change. Instead, RSTP sets the TC bit in the **Flags** field of the RST BPDU to **1** and notifies the entire network of the topology change via RST BPDU flooding.

Figure 1-1Configuration BPDU of RSTP



```
0000  01 80 c2 00 00 00 00 d0  f8 22 35 4a 00 27 42 42
0010  03 00 00 02 02 7c 80 00  00 d0 f8 22 35 4a 00 00
0020  00 00 80 00 00 d0 f8 22  35 4a 80 1c 00 00 14 00
0030  02 00 0f 00 00 65 20 54  79 70 65 00
```

As shown in Figure 1-1, the **Length** field is 0x0027, which indicates that the BPDU length is 39 bytes, including a 3-byte LLC header and a 36-byte RST BPDU. The BPDU starts from the 18th byte. The fields are described in Table 1-1.

Table 1-1Format of an RST BPDU

Field	Bytes	Description
PID	2	Protocol identifier. The value is **0x0000**.
PVID	1	Protocol version identifier. The value is **0x02**, indicating an RSTP packet. Devices running STP will discard RST BPDUs upon receiving.
BPDU Type	1	BPDU packet type. The value is **0x02**, indicating an RST BPDU.
Flags	1	An RST BPDU uses all the eight bits. In the example, the value is 0x7c, that is, 0111 1100. ● Bit 7: TCA. The value **1** indicates that a topology change is known. ● Bit 6: Agreement. The value **1** indicates consent to status switching. ● Bit 5: Forwarding. The value **1** indicates that forwarding is allowed. ● Bit 4: Learning. The value **1** indicates that learning is allowed. ● Bit 3 and bit 2: Port role. ○ **00**: Unknown. ○ **01**: Root port. ○ **10**: Alternate/Backup. ○ **11**: Designated port. ● Bit 1: Proposal. The value **1** indicates sending a proposal to request status switching. ● Bit 0: TC. The value **1** indicates notifying the topology change.
Root ID	8	Identifier of the root bridge recognized by the device. It consists of a 2-byte bridge priority and a 6-byte bridge MAC address.
RPC	4	Root path cost, which is the cumulative path cost from the port to the root bridge.
Bridge ID	8	Bridge identifier of the device. It consists of a 2-byte bridge priority and a 6-byte bridge MAC address.
Port ID	2	Identifier of the port that sends the BPDU. It consists of a 1-byte priority and 1-byte port ID.
Message Age	2	Age of the BPDU, which represents the time that the BPDU has been alive in the network.
Max Age	2	Aging time of the BPDU.
Hello Time	2	Interval for sending two adjacent BPDUs. After an RSTP topology becomes stable, a non-root switch automatically sends configuration BPDUs at an interval of **Hello Timer** regardless of whether it receives the configuration BPDU from the root bridge. If a non-root bridge fails to receive a BPDU from the root within the time of **Time-factor** × **Hello Time**, it considers the root bridge or the link to the root bridge to be faulty. **Time-factor** is a multiple of **Hello Time** and can be configured.

Field	Bytes	Description
FWD Delay	2	Forwarding delay. This time parameter controls the duration of listening and learning states.
Ver 1 Length	1	The value is 0x00, indicating that the BPDU does not contain content of Version 1. STP does not have this field.

3. Port Roles

RSTP uses alternate ports and backup ports to replace blocking ports, and adds the edge port auto-identification function. Therefore, RSTP supports six types of ports:

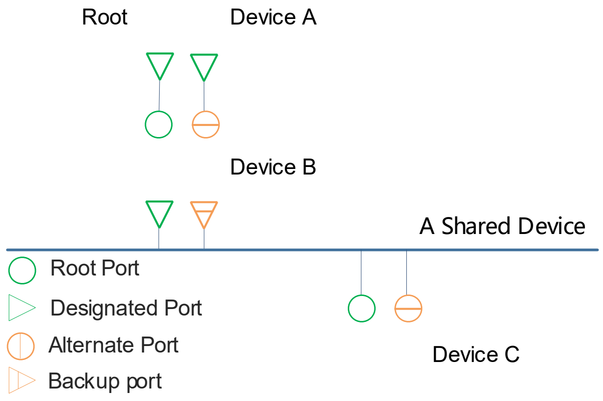
- Root port: A port with the shortest path to the root bridge.
- Alternate port: A backup port of a root port. Once the root port fails, the alternate port becomes the root port immediately.
- Designated port: A port that connects a root bridge or upstream bridge to a downstream device.
- Backup port: A backup port of a designated port. When a root bridge or upstream bridge has two interfaces to connect to the same downstream bridge, the interface with a higher priority is the designated port and the interface with a lower priority is the backup port. Once the designated port fails, the backup port becomes the designated port immediately.
- Edge port: A port connected to a terminal. Edge ports do not participate in spanning tree calculation. They can rapidly enter forwarding state with no need to wait for twice the period of **Forward Delay**. An edge port that receives a BPDU automatically becomes a non-edge port.
- Disable port: A port in down state.

Improvement effects:

- The work mechanism of the alternate port and backup port enables a device to restore the network connectivity within several milliseconds after discovering a topology change, with no need to transmit configurations.
- The status change of an edge port will neither affect the network connectivity nor cause loops. Edge ports can enter forwarding state without delay.

Figure 1-6 shows port roles. The port priority sequence is as follows unless otherwise specified: root port > designated port > alternate port > backup port.

Figure 1-1RSTP Port Roles



4. BPDU Processing of RSTP

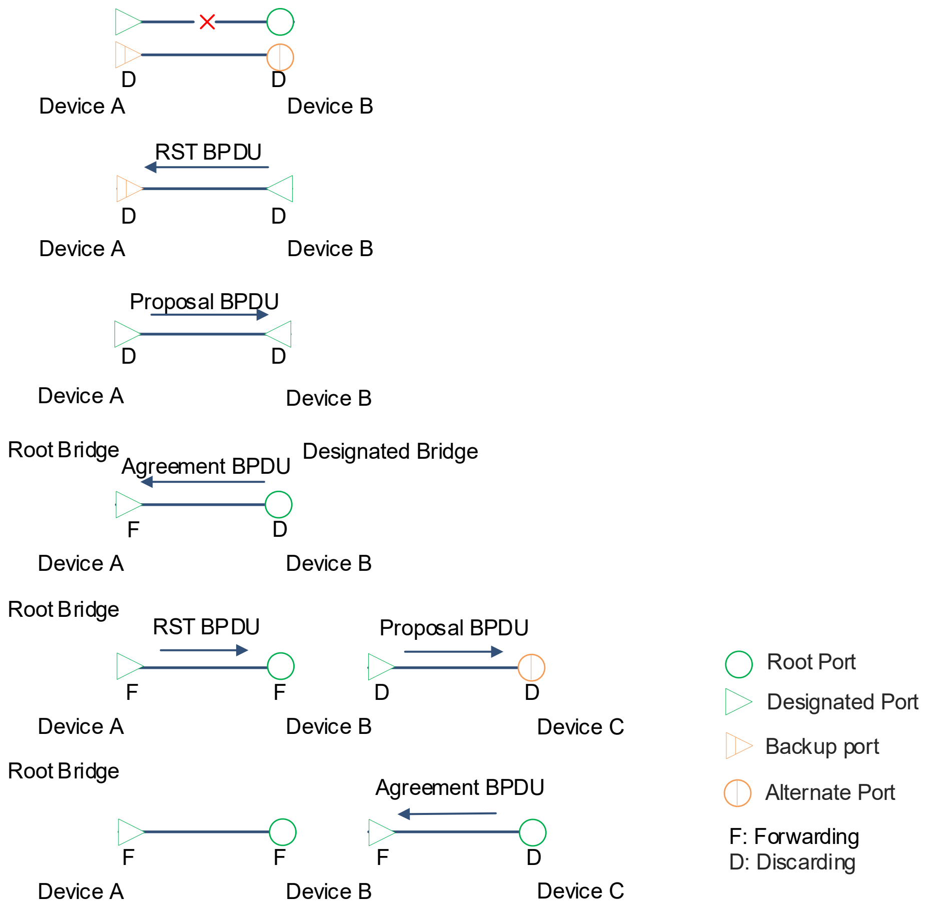
In RSTP, a device processes BPDUs in a different way.

- Independent Hello: After an STP topology becomes stable, the root bridge sends configuration BPDUs at an interval of **Hello Timer**. Non-root switches send configuration BPDUs only after receiving the configuration BPDU from the upstream device, resulting in long convergence time in the case of a topology change. After an RSTP topology is stable, non-root switches automatically send configuration BPDUs at an interval of **Hello Timer** regardless of whether they receive configuration BPDUs from the root bridge.
- Fast fault detection: In STP, if a non-root switch fails to receive a BPDU from the upstream device within one period of **MAX Age**, the switch considers the upstream device to be faulty. In RSTP, if a non-root switch fails to receive a BPDU from the upstream device within three periods of **Hello Time**, it considers the upstream device to be faulty. The value range of **Max Age** is from 6 to 40, in seconds, the default value is **20**, and it takes at least 6s to detect the upstream device fault. The value range of **Hello Time** is from 1 to 10, in seconds, the default value is **2**, and it takes at least 3s to detect the upstream device fault. Therefore, RSTP supports a shorter fault detection time.
- Fast convergence: In STP, all ports discard low-priority BPDUs upon receipt. If a blocking port fails to receive a high-priority BPDU within one period of **MAX Age**, the port changes to the designated port and sends its BPDU to the peer. In RSTP, when a downstream device receives a BPDU from the upstream device and finds that the priority of the BPDU is lower than that of the BPDU stored in the port, it sends its stored BPDU to the upstream device immediately. The upstream device compares the received BPDU with its own BPDU and updates the BPDU stored in the local port. This mechanism can accelerate topology convergence.

5. Fast Convergence of RSTP

In STP, to transition from listening state to learning state and then to forwarding state, a port needs to wait for two periods of **Forward Delay**. The value range of **Forward Delay** is from 4 to 30, in seconds, the default value is **15**, and therefore, the default waiting time is **30**s. In RSTP, a port enters discarding state after role election is complete, and then performs a Proposal/Agreement handshake with the connected bridge so as to quickly enter forwarding state. Figure 1-1 shows a fast convergence process of RSTP for example.

Figure 1-1Fast Convergence of RSTP



(1) Proposal: If the root port of Device B fails to receive a BPDU from the upstream device within three periods of **Hello Time**, it considers the upstream device to be faulty. Device B returns to the initial state, considers itself as the root, and sets all its ports as designated ports (in discarding state) to send RST BPDUs. The alternate port (if Device A is not the root) or backup port (if Device A is the root) of Device A receives an RST BPDU from Device B. If Device A finds that the RST BPDU from Device B has a lower priority than its RST BPDU, Device A sets its port as a designated port (in discarding state), sets the **Proposal** bit in the configuration BPDU of the port to **1**, and sends a Proposal BPDU to Device B to request Device B to recognize Device A as the root bridge.

(2) Agreement: Device B receives the Proposal BPDU from Device A. If the BPDU of Device A is superior to that of Device B, Device B considers Device A as the root bridge (Device A may be the root or an intermediate device between Device B and the root). The port receiving the Proposal BPDU on Device B becomes the root port, and the root port stops sending RST BPDUs. Before a port transitions to forwarding state, Device B needs to block non-edge designated ports to prevent loops. The time required by ports to transition to -discarding state may be different. Devices use a set of synchronization variables to indicate whether the port state transition is complete. Device B sets the **Sync** variable to **1** for all ports. After all the ports enter discarding state, the **Synced** variable of all the ports is set to **1**. Then, the **Synced** variable of the root port is set to **1**, indicating that all the ports have been transitioned to the correct state. The root port sets the **Proposal** bit to **0** and the **Agreement** bit to **1** in the Proposal BPDU received from Device A and keeps other content of the BPDU unchanged, to create an Agreement BPDU. Then, the root port sends the BPDU to Device A.

(2) Forwarding: The designated port of Device A enters forwarding state after receiving the Agreement BPDU, and sends the RST BPDU with the **Forwarding** bit set to **1** at an interval of **Hello Time**. After receiving the RST BPDU, the root port of Device B enters forwarding state.

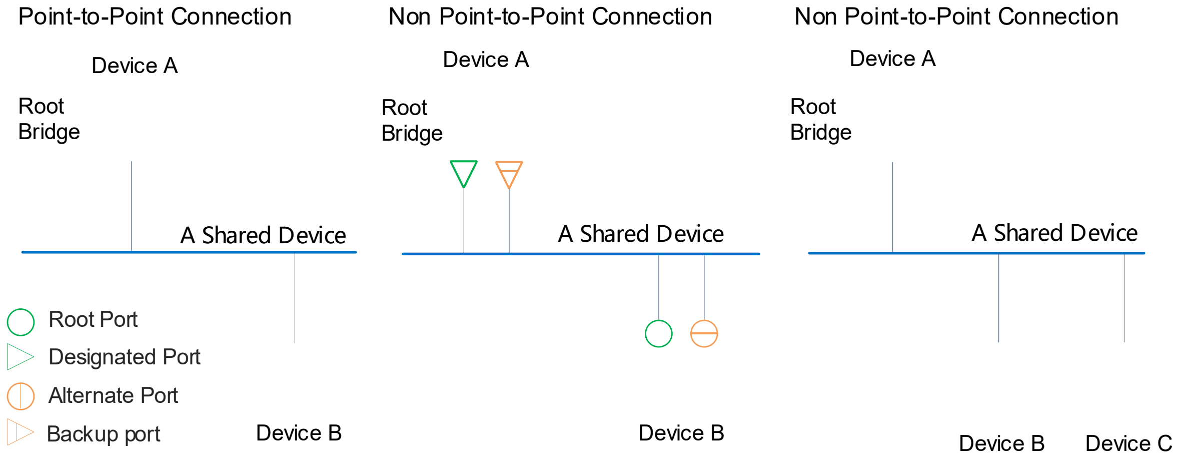
(3) Extension proposal: The designated port of Device B sends a Proposal BPDU with both **Proposal** and **Agreement** bits set to **1** to the downstream device, to request the downstream device to recognize Device B as the upstream device and extend the spanning tree. Theoretically, RSTP can recover the network tree topology immediately and achieve fast convergence once the network topology changes.

Improvement effect: Non-edge designated ports are blocked actively, without waiting for the period of **Forward Delay**. Hence, the convergence is accelerated. Network connectivity can be restored within the time of exchanging two handshake BPDUs, and the delay incurred by handshakes is less than 1s.

⚠ **Caution**

The above "handshake" process must meet one condition: Ports must be in a point-to-point connection state. Otherwise, the ports cannot shake hands and fast convergence cannot be achieved. To make RSTP-compliant devices function as they should, you are advised to connect devices in a point-to-point way.

Figure 1-1Examples of Point-to-Point Connection and Non Point-to-Point Connection



6. Drawbacks of RSTP

- RSTP produces only one spanning tree in the whole switching network, which cannot prevent performance degradation caused by the network expansion. Therefore, when using RSTP, it is not recommended that the number of devices between any two nodes running STP/RSTP in the network exceed 7.
- RSTP can only implement redundant backup but does not support load balancing based on VLAN traffic.
- STP and RSTP calculate the spanning tree topology based on devices (nodes), and the impact of VLANs is not taken into account. In a particular topology, ports in the same VLAN may be blocked and fail to communicate with each other. This problem and the solution will be detailed in the MSTP section.

1.1.5 MSTP

1. Overview

MSTP, defined by the IEEE 802.1s standard, is capable of ironing out drawbacks of STP, RSTP (only one spanning tree is generated for a device), and Per-VLAN Spanning Tree (PVST) (one spanning tree is generated for each VLAN).

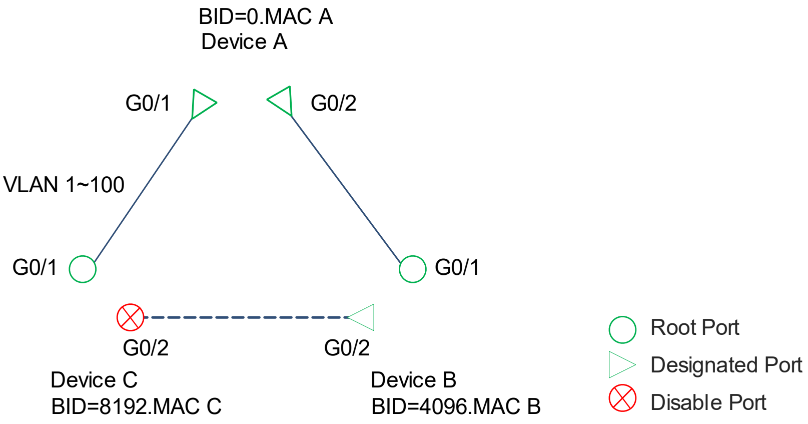
- After MSTP is deployed, if a loop arises in a network, MSTP performs topology calculation, and eliminates possible loops in the network by blocking redundant links.
- When an active path fails, MSTP activates the redundant backup link to restore the network connectivity.
- MSTP supports fast convergence.
- MSTP forwards the traffic of different VLANs along their respective spanning trees, which provides a better load balancing mechanism for redundant links.

2. Origin of MSTP

The conventional STP and RSTP protocols have the following two problems and MSTP emerges as a solution to them.

- Both STP and RSTP block redundant lines and concentrate traffic on the active link, which leads to a waste of bandwidth of the redundant link and heavy traffic on the active link, resulting in traffic imbalance.

Figure 1-1Traffic Sharing Failure in STP

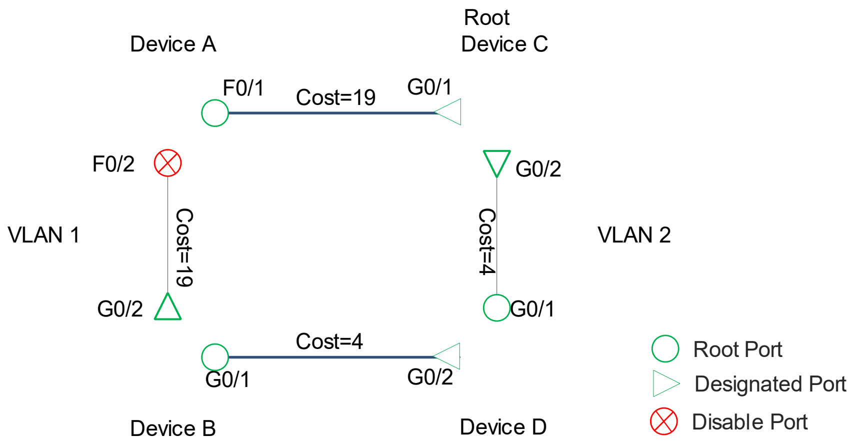


As shown in Figure 1-1, assume that the rate of each segment of link is 1 Gbps. If Devices A, B, and C run STP, the STP algorithm first elects Device A as the root based on the bridge ID. On non-root bridge B, the STP algorithm compares <**Root Path Cost**, **Bridge ID**, **Port ID**> in sequence, and selects port GigabitEthernet 0/1 as the designated port based on the root path cost. STP performs the same operation on Device C. Ports on the directly connected segment between Device C and Device B have the same root path cost. Port GigabitEthernet 0/2 of Device B is elected as the designated port based on the bridge ID, and port GigabitEthernet 0/2 of Device C is blocked. Port GigabitEthernet 0/2 of Device C cannot make communication due to STP blocking. VLANs 1–100 can be configured only on port GigabitEthernet 0/1 of Device C. It is impossible to implement traffic sharing by configuring port GigabitEthernet 0/1 of Device C to allow VLANs 1–50 and port GigabitEthernet 0/2 of Device C to allow VLANs 51–100.

- In special cases, STP and RSTP may block links in the same VLAN, causing interfaces within the same

VLAN to be unable to communicate with each other.

Figure 1-2Communication Failure in the Same VLAN Caused by STP



As shown in Figure 1-2, Devices A and B are in VLAN 1, and Devices C and D are in VLAN 2. Assume that the topology tree calculated by STP is shown in the figure. Port FastEthernet 0/2 of Device A is blocked, and the link between Devices A and B is blocked. VLAN 1 is not configured on Devices C and D, and Devices C and D cannot forward packets of VLAN 1. As a result, Devices A and B in VLAN 1 cannot communicate with each other.

3. Instance and Instance Mapping

The collection of one or more VLANs on one device is called a Multiple Spanning Tree Instance (MSTI) or instance for short. If a device supports a maximum of 65 instances, instance **0** exists by default and cannot be deleted, and instances 1–64 can be created or deleted. Users can allocate VLANs 1–4094 to different instances as required, and unallocated VLANs belong to instance **0** by default. The mapping between MSTIs and VLANs is called instance mapping, as shown in Table 1-1.

Table 1-1Mappings Between MSTIs and VLANs

Instance	VLAN Mapped
0	1–2, 4, 11–4094
1	3, 5–7
2	8–10

Note

You are advised to disable MSTP before configuring mappings between MSTIs and VLANs, and then enable MSTP after configuration, to ensure stable convergence of the network topology.

4. Multiple Spanning Tree Region

Devices that run the MSTP protocol, have the same configuration name, revision number, and instance mappings constitute a multiple spanning tree (MST) region. Configuration names, revision numbers, and instance mappings are recorded in the **MST CFG ID** field of MST BPDUs and they can be configured.

- Configuration name: A string of 32 bytes used to identify an MST region.
- Revision number: A 2-byte non-negative integer used to identify an MST region.
- Instance mapping: The mappings between MSTIs and VLANs are encrypted into a 16-byte digest. One MST region can contain multiple MSTIs.

MST regions are independent of each other. If a port on a device receives a BPDU with **MST CFG ID** same as that of the MST BPDU of the device, the device deems that the peer device and the device belong to the same MST region. Otherwise, the device deems that the peer device belongs to a different MST region. The load sharing advantage of MSTP can be reflected only after multiple devices are configured to the same MST region. Therefore, MST regions need to be properly divided and devices in the same MST region need to have the same **MST CFG ID**.

5. Instance Spanning Tree

One instance spanning tree can be generated for each instance. Each instance calculates and generates an instance spanning tree based on MSTI configuration messages in MST BPDUs. Information contained in MST configuration messages is described in Table 1-1.

If a device contains only one instance or MST configuration messages of multiple instances are the same, the same instance spanning tree will be generated, which is useless in traffic sharing. If VLANs are allocated to different instances to form multiple VLAN groups and different bridge priorities and port priorities are configured for different instances, multiple different instance spanning trees can be generated for the instances in the region to share VLAN traffic. The spanning tree of each instance is not affected by the spanning trees of other instances.

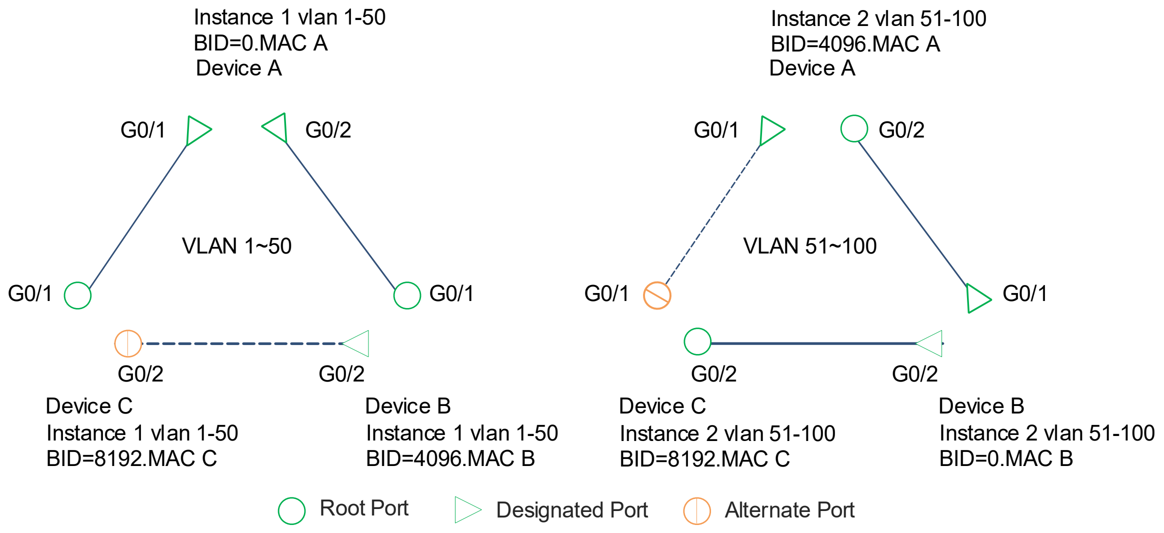
⚠ **Caution**

"MSTI configuration messages" of instances need to be manually modified so that spanning trees of instances are different, that is, the selected root, designated ports, blocking ports, and active links are different for different instances. In this way, VLAN traffic sharing can come into play.

MSTP calculates the topology for each instance in the same way as RSTP. In each instance, MSTP first elects the root bridge by comparing the root ID, and all ports on the root bridge are designated ports. Then, MSTP compares key elements in the priority vector <**Root ID**, **Root Path Cost**, **Bridge ID**, **Port ID**> on non-root bridges to elect root ports. Afterwards, MSTP compares the two ports of each directly connected network segment, and elects the port with a higher priority as the designated port and the one with a lower priority as the alternate port. Devices produce multiple instance spanning trees by means of MST BPDU interaction and calculation. Each instance provides a single loop-free network topology for VLAN groups in the instance, and the topologies of different VLAN groups are different.

For example, after MSTP runs in the topology as shown in Figure 1-1, the calculated topology is shown in Figure 1-1.

Figure 1-1Traffic Sharing Implemented by MSTP



In instance 1, devices elect Device A as the root by comparing root IDs (bridge IDs of the devices at this moment). Devices compare the root path cost and elect port GigabitEthernet 0/1 of Device B and port GigabitEthernet 0/1 of Device C as the root ports. In the directly connected network segment between Devices C and B, the two devices have the same root ID (0.MAC A). Bridge ID of Device B is smaller than that of Device C, and therefore, port GigabitEthernet 0/2 of Device B is elected as the designated port and port GigabitEthernet 0/2 of Device C is elected as the alternate port. When port GigabitEthernet 0/1 of Device C malfunctions, port GigabitEthernet 0/2 of Device C becomes the root port immediately.

In instance 2, devices elect Device B as the root by comparing root IDs (bridge IDs of the devices at this moment). Devices compare the root path cost and elect port GigabitEthernet 0/2 of Device A and port GigabitEthernet 0/2 of Device C as the root ports. In the directly connected network segment between Devices A and C, the two devices have the same root ID (0.MAC B). Bridge ID of Device A is smaller than that of Device C, and therefore, port GigabitEthernet 0/1 of Device A is elected as the designated port and port GigabitEthernet 0/1 of Device C is elected as the alternate port. When port GigabitEthernet 0/2 of Device C malfunctions, port GigabitEthernet 0/1 of Device C becomes the root port immediately.

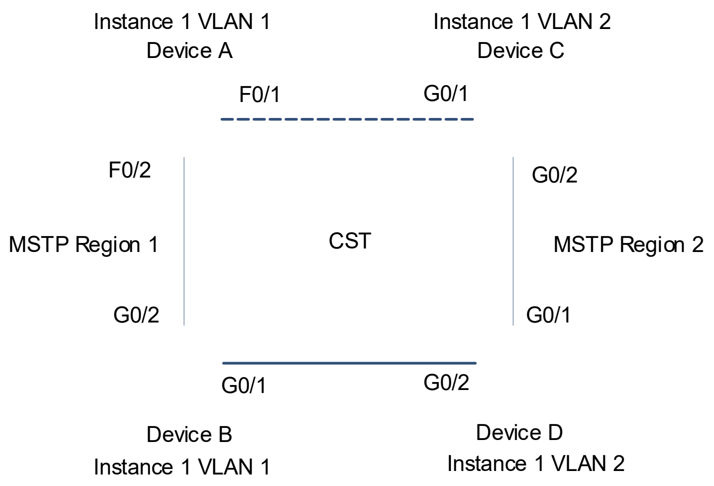
It can be seen that different spanning trees are calculated for instance 1 and instance 2 due to different bridge IDs. Consequently, packets of VLANs 1–50 are forwarded along the spanning tree of instance 1 and packets of VLANs 51–100 are forwarded along the spanning tree of instance 2, implementing VLAN traffic sharing.

Users can, as needed, configure different instances for different VLAN groups, Configuring the Bridge Priority and Port Priority by instance, Configuring the Port Path Cost, and plan VLAN groups to forward traffic through different paths.

6. Common Spanning Tree

Devices with different instances and VLAN mapping configurations do not belong to the same MST region and each MST region is a whole. Therefore, one MST region can be considered as one node. The inter-node spanning tree calculated via STP is called a common spanning tree (CST). Instances with the same ID can exist in different MST regions, for example, instance 1 exists in two MST regions. Though the instance IDs are the same, instance spanning trees are generated independently for the instances and the instances are unrelated to each other and unrelated to the CST. If two MST regions are configured in a topology shown in Figure 1-2, the calculated topology is shown in Figure 1-1 after MSTP runs.

Figure 1-1Preventing a Communication Failure of Ports in the Same VLAN by MSTP



Devices A and B (both instance 1 VLAN 1) have the same instance mapping. There is no loop in MST region 1, and therefore, no port is blocked in this region.

Devices C and D (both instance 1 VLAN 2) have the same instance mapping. There is no loop in MST region 2, and therefore, no port is blocked in this region.

MST region 1 and MST region 2 can be considered as two nodes and there are loops between the two nodes. Therefore, a link is blocked according to the configuration to prevent loops and ensure that the intra-VLAN communication is not affected.

7. Common and Internal Spanning Trees

In a topology as shown in Figure 1-1, there are three MST regions. In each MST region, one instance spanning tree is generated for each instance. As shown in Figure 1-2, in region 2, one spanning tree is generated for instance 0, instance 1, and instance 2 each. According to the MSTP protocol, the spanning tree of instance 0 represents the region and is called an internal spanning tree (IST).

Each MST region is considered as a node, and one spanning tree is calculated between nodes according to the STP protocol. This spanning tree is called a common spanning tree (CST). As shown in Figure 1-1, a spanning tree connected using blue lines indicates an IST in a region while a spanning tree connected using orange lines indicates an inter-region CST.

ISTs and a CST constitute a common and internal spanning tree (CIST).

CIST root: Root of a CIST. It is also the root of a CST, that is, CST root. In the CST calculation, the device with the smallest bridge ID is elected as the CIST root. As shown in Figure 1-1, assume that Device A has the smallest bridge ID. Device A is elected as the CIST root.

Region root: In an IST, the device with the smallest root path cost to the CIST root is elected as the region root of the IST, which is also called master bridge. CIST root A is also the region root of region 1. As shown in Figure 1-1, assume that Device B in region 2 has the smallest root path cost to Device A. Device B is elected as the region root of region 2. Device B may not be directly connected to Device A but may through other devices in region 1. Likewise, Device C is elected as the region root of region 3.

The region root of region 2 is Device B. Suppose that three spanning trees are calculated for the three instances in region 2, and the roots of the three spanning trees are all Device B. On Device B, the port with the smallest root path cost to the CIST root is the master port. Therefore, compared with RSTP, MSTP adds two port roles: master port and region edge port.

⚠ **Caution**

The **CIST Regional Root** field in MSTP packets indicates a region root rather than the CIST root.

A region root (represented by **CIST Regional Root**) may not be the device with the smallest bridge ID in a region but is the device with the smallest root path cost from the region to the CIST root.

To make the topology more stable, it is recommended that the "outlets" of different instance spanning trees in each region to the CIST root be configured on the same device of the region. As shown in Figure 1-14, the outlets of MSTI 1, MSTI 2, and MSTI 3 in region 2 to the CIST root are configured on Device B.

Figure 1-1CIST Composed of ISTs and the CST

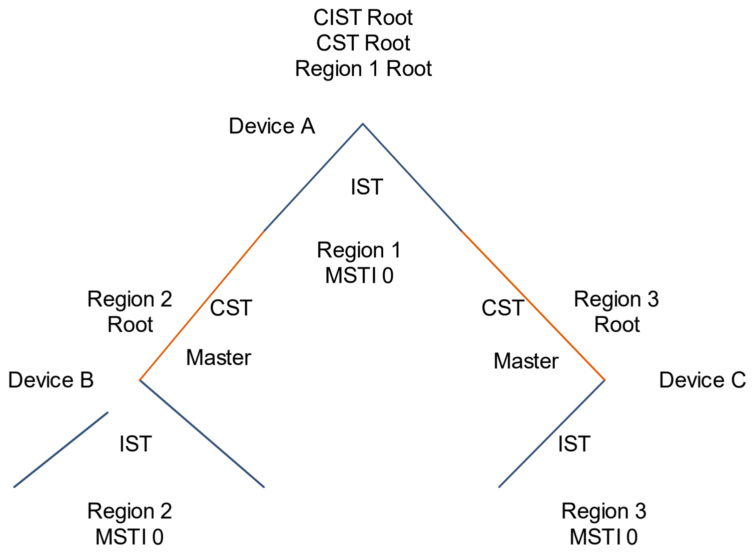
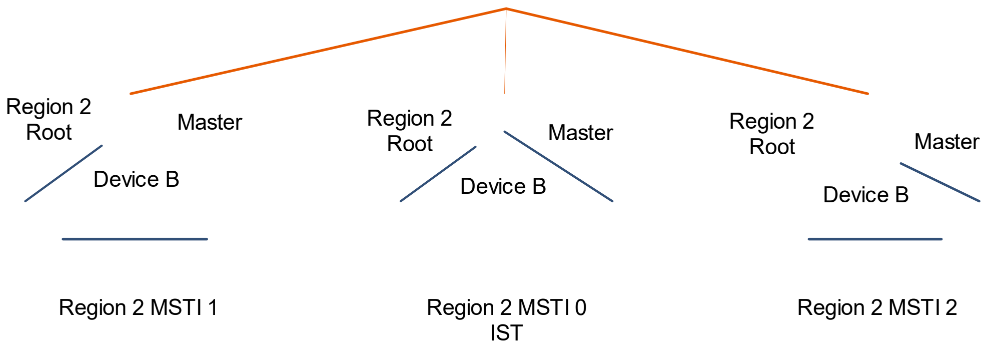


Figure 1-2Three Spanning Trees Generated for the Three Instances in Region 2



8. Port Roles

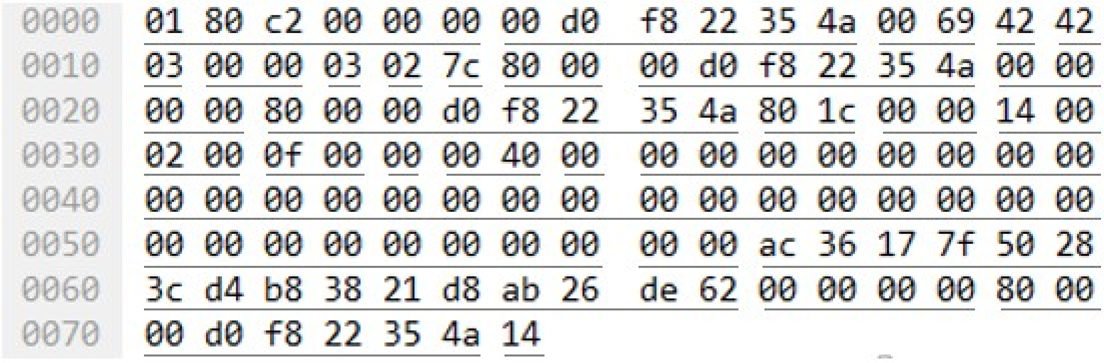
MSTP introduces a new port role to the region root: master port, which is the port with the smallest root path cost from a region to the CIST root. The master port is the "outlet" of all instances in a region, and all instances can forward data through the master port.

- Root port: A port with the shortest path to the root bridge.
- Designated port: A port that connects a bridge to the root bridge in each LAN.
- Alternate port: A backup of a root port. Once the root port fails, the alternate port becomes the root port immediately.
- Backup port: A backup of a designated port. When two ports on a bridge connect to the same LAN, the port with a higher priority is the designated port and the one with a lower priority is the backup port. Once the designated port fails, the backup port changes to the designated port immediately.
- Edge port: A port connected to a terminal. Edge ports do not participate in spanning tree calculation.
- Disable port: A port in down state.
- Master port: A port with the shortest path cost to the CIST root. It is the "outlet" of all MSTIs in an MST region.
- Region edge port: A port directly connected to other MST regions. A master port is a special region edge port.

9. MSTP Packets

MSTP uses MST BPDUs as the basis for spanning tree calculation. MST BPDUs can be used to calculate spanning tree topologies, maintain network topologies, and communicate topology change records.

Figure 1-1Configuration BPDU of MSTP



```
0000  01 80 c2 00 00 00 00 d0  f8 22 35 4a 00 69 42 42
0010  03 00 00 03 02 7c 80 00  00 d0 f8 22 35 4a 00 00
0020  00 00 80 00 00 d0 f8 22  35 4a 80 1c 00 00 14 00
0030  02 00 0f 00 00 00 40 00  00 00 00 00 00 00 00 00
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0050  00 00 00 00 00 00 00 00  00 00 ac 36 17 7f 50 28
0060  3c d4 b8 38 21 d8 ab 26  de 62 00 00 00 00 80 00
0070  00 d0 f8 22 35 4a 14
```

As shown in Figure 1-1, the **Length** field is 0x0069, which indicates that the packet length is 105 bytes in this example, including a 3-byte LLC header and a 102-byte MST BPDU. This length is the minimum length of an MST BPDU. For both intra-region and inter-region MST BPDUs, the first 36 bytes are the same as those of an RST BPDU. MSTP-specific fields start from the 37th byte, and the MSTI configuration message field at the end is composed of multiple MSTI configuration messages. The fields are described in Table 1-1.

Table 1-1Format of an MST BPDU

Field	Bytes	Description
PID	2	Protocol identifier. The value is 0x0000.
PVID	1	Protocol version identifier. 0x03 indicates that the packet protocol is MSTP.
BPDU Type	1	BPDU packet type. The value is 0x02 for MSTP. 0x02 indicates that the BPDU type is MST BPDU.
CIST Flags	1	Configuration BPDUs of MSTP use all the eight bits. In the example, the value is 0x7c, that is, 0111 1100. ● Bit 7: TCA. An upstream device sets bit 7 in a configuration BPDU to **1** and sends the BPDU to notify the downstream device of the topology change and request the downstream device to stop sending the TCN BPDU. ● Bit 6: Agreement. The value **1** indicates consent to status switching. ● Bit 5: Forwarding. The value **1** indicates forwarding state. ● Bit 4: Learning. The value **1** indicates learning state. ● Bit 3 and bit 2: Port role. ○ 00: Unknown. ○ 01: Root port. ○ 10: Alternate/Backup. ○ 11: Designated port. ● Bit 1: Proposal. The value **1** indicates sending a proposal to request status switching. ● Bit 0: TC. The root bridge sets bit 0 in a configuration BPDU to **1** and sends the BPDU to instruct the downstream device to delete MAC address entries.
CIST Root ID	8	Bridge ID of the CIST root, which consists of a 2-byte root priority and a 6-byte root MAC address. In the example, the priority value is 0x8000, which indicates 32768, and the bridge MAC address is 00-d0-f8-22-35-4a.
CIST External Path Cost	4	CIST external path cost, which refers to the cumulative path cost from the MST region, to which the device belongs, to the CIST root. The CIST external path cost is calculated based on the link bandwidth, which is 0x00 00 00 00 in the example.
CIST Regional Root ID	8	CIST regional root identifier, that is, bridge ID of the IST master bridge. If the CIST root is in the region, the region root ID is the CIST root ID. In the example, the priority value is 0x8000, which indicates 32768, and the bridge MAC address is 00-d0-f8-22-35-4a.
CIST Port ID	2	ID of a port in an IST. In the example, the priority value is 0x80, which indicates 128, and the port ID is 1c.
Message Age	2	Age of a BPDU, that is, keepalive time of the packet. The value is 0x0000 in the example.
Max Age	2	Maximum timeout time of a BPDU, after which the link to the root switch is

Field	Bytes	Description
		considered to be faulty. The value in the example is 0x1400 (low-order bytes on the left), indicating 20 seconds.
Hello Time	2	Hello interval for sending two adjacent BPDUs. The default value is **2**s. The value in the example is 0x0200 (low-order bytes on the left), indicating 2 seconds.
FWD Delay	2	**Forward Delay** timer. The default value is **15**s. This field controls the duration of listening and learning states. The value in the example is 0x0f00 (low-order bytes on the left), indicating 15 seconds.
Ver 1 Length	1	Length of a Version1 BPDU. The value is fixed to 0x00.
Ver 3 Length	2	Length of a Version3 BPDU. The value is 0x0040 in the example.
MST CFG ID	51	MST configuration identifier, that is, MST region label information, which contains four fields. Interconnected devices with the same four fields in **MST CFG ID** belong to the same region. ● (1-byte) Configuration Identifier Format Selector: The value is fixed to 0x00. ● (32-byte) Configuration Name: Region name. The default value is all 0's. ● (2-byte) Revision Level: Revision number, which is a non-negative integer. The value is 0x0000. ● (16-byte) Configuration Digest: MSTI-VLAN instance mapping table. The HMAC-MD5 algorithm is used to encrypt the mappings between instances and VLANs in a region into a 16-byte digest.
CIST Internal RPC	4	CIST internal root path cost, which is the cumulative path cost from the local port to the master switch of the IST. The internal root path cost of a CIST is calculated based on the link bandwidth. The value is 0x00-00-00-00 in the example.
CIST Bridge ID	8	ID of the designated switch in the CIST. It consists of a 2-byte bridge priority and a 6-byte bridge MAC address. In the example, the priority value is 0x8000, which indicates 32768, and the bridge MAC address is 00-d0-f8-22-35-4a.
CIST R Hops	1	Number of remaining hops of a BPDU in a CIST. The default value is 0x14, which indicates 20 hops.
MSTI Configuration Messages (may be absent)	16	MSTI configuration message. The configuration message of each MSTI occupies 16 bytes. If there are *n* MSTIs, they occupy *n* × 16 bytes. Fields in the configuration message of a single MSTI are described as follows: ● (1-byte) MSTI Flags: MSTI flag. ● (8-byte) MSTI Regional Root Identifier: MSTI region root ID. ● (4-byte) MSTI Internal Root Path Cost: Cumulative path cost from the local port to the MSTI region root switch. The internal root path cost of an MSTI is calculated based on the link bandwidth. ● (1-byte) MSTI Bridge Priority: Bridge priority of the device in the MSTI. ● (1-byte) MSTI Port Priority: Port priority of the device in the MSTI. ● (1-byte) MSTI Remaining Hops: Number of remaining hops of a BPDU in

Field	Bytes	Description
		the MSTI.

1.1.6 Protocols and Standards

- IEEE 802.1D: Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Bridges
- IEEE 802.1w: IEEE Standard for Information Technology -Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Common Specifications - Part 3: Media Access Control (MAC) Bridges:Amendment 2:Rapid Reconfiguration
- IEEE 802.1s: IEEE Standards for Local and Metropolitan Area Networks - Amendment to 802.1Q Virtual Bridged Local Area Networks: Amendment 3: Multiple Spanning Trees

1.2 Configuration Task Summary

1.2.1 Configuring STP

The STP basic configuration includes the following tasks:

(1) Configuring the Spanning Tree Mode

(2) (Optional) Configuring the Bridge Priority and Port Priority

(3) (Optional) Configuring the Port Path Cost. Select at least one of the following to configure.

- Configuring the Path Cost Calculation Method
- Configuring the Path Cost Value

(4) (Optional) Configuring Spanning Tree Time Parameters

(5) Enabling the STP Function

1.2.2 Configuring RSTP

The RSTP basic configuration includes the following tasks:

(1) Configuring the Spanning Tree Mode

(2) (Optional) Configuring the Bridge Priority and Port Priority

(3) (Optional) Configuring the Port Path Cost. Select at least one of the following to configure.

- Configuring the Path Cost Calculation Method
- Configuring the Path Cost Value

(4) (Optional) Configuring Spanning Tree Time Parameters

(5) (Optional) Configuring Fast Convergence of Spanning Trees

(6) (Optional) Enabling Protocol Migration

(7) Enabling the STP Function

1.2.3 Configuring MSTP

The MSTP basic configuration includes the following tasks:

(1) Configuring the Spanning Tree Mode

(2) Configuring an MST Region

(3) Configuring the Bridge Priority and Port Priority

(4) (Optional) Configuring the Port Path Cost. Select at least one of the following to configure.

- Configuring the Path Cost Calculation Method
- Configuring the Path Cost Value

(5) (Optional) Configuring Spanning Tree Time Parameters

(6) (Optional) Configuring the Maximum Hop Count for BPDUs

(7) (Optional) Configuring Fast Convergence of Spanning Trees

(1) (Optional) Configuring Spanning Tree Compatibility for Interfaces

(2) Enabling the STP Function

1.2.4 Configuring Advanced Functions for Spanning Trees

Configuring advanced functions for spanning trees includes the following tasks. In networking applications, these functions are used to enhance the stability, robustness, and anti-attack capability of protocols based on the network topology and application characteristics, so as to meet the application requirements of protocols in different user scenarios. The following configuration tasks are all optional. When performing the following configuration, be sure to complete the configuration and then enable spanning tree protocols.

- Configuring Port Protection
    - Configuring Root Guard
    - Configuring Loop Guard
- Configuring BPDU Source MAC Address Check
- Configuring an Edge Port
    - Configuring Autoedge
    - Configuring the Port Fast Attribute
- Configuring BPDU Guard or BPDU Filter
    - Configuring BPDU Guard
    - Configuring BPDU Filter
- Configuring TC Attack Defense Functions
    - Configuring TC Protection
    - Configuring TC Guard
    - Configuring TC Filter
- Configuring BPDU Tunnel

1.2.5 Configuring BPDU Transparent Transmission

When STP/RSTP/MSTP is not enabled on a device, if the spanning tree between the device and the interconnected device needs to be calculated properly, BPDU transparent transmission needs to be configured on the device to transparently transmit BPDU frames.

- Configuring BPDU Transparent Transmission

1.3 Configuring the Spanning Tree Mode

1.3.1 Overview

According to IEEE 802.1 protocol standards, STP, RSTP, and MSTP are mutually compatible, without any configuration by the administrator. Some vendors' devices may not implement STP/RSTP/MSTP according to protocol standards, causing incompatibility. Therefore, Orion_B26Q provides a spanning tree mode configuration command for the administrator to switch the spanning tree mode to a lower version if other vendors' devices are incompatible with a Orion_B26Q device.

1.3.2 Restrictions and Guidelines

- **mstp** indicates the Multiple Spanning Tree Protocol (IEEE 802.1s).
- **rstp** indicates the Rapid Spanning Tree Protocol (IEEE 802.1w).
- **stp** indicates the Spanning Tree Protocol (IEEE 802.1d).
- If the STP mode is switched to RSTP mode, the convergence time is the convergence time of STP at the first topology change and the convergence time of RSTP at the second topology change.

1.3.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the spanning tree mode.

spanning-tree mode { **mstp** | **rstp** | **stp** }

The default spanning tree mode is MSTP.

1.4 Configuring an MST Region

1.4.1 Overview

This section describes how to configure an MST region to change device members in the MST region, so as to affect the spanning tree topology.

1.4.2 Restrictions and Guidelines

- To enable multiple devices to belong to the same MST region, you need to configure the same configuration name, revision number, and instance mappings for the devices.

- One VLAN belongs to only one instance. All VLANs belong to instance **0** by default. In MST configuration mode, you can run the **instance vlan** command to move a VLAN group to an instance. For example, the **instance** 1 **vlan** 2-200 command moves VLANs 2–200 to instance **1**. The **instance** 1 **vlan** 2,20,200 command moves VLANs 2, 20, and 200 to instance **1**.
- You can run the **no instance vlan** command to move a VLAN from a custom instance to instance **0**.
- *instance-id* indicates an MST instance ID. The value range is from 0 to 64. *vlan-range* indicates a VLAN ID range. The value range is from 1 to 4094. It can be configured in MST configuration mode.
- *name* indicates an MST name, which can contain 32 bytes at most. The default value is an empty string.
- *version* indicates the MST revision number. The default value is **0** and the value range is from 0 to 65535.

1.4.3 Prerequisites

Only MSTP supports instance configuration. If the spanning tree mode is incorrect, run the **spanning-tree mode mstp** command to set the spanning tree mode to MSTP first.

You are advised to configure instance-VLAN mappings when STP is disabled, to ensure the stability and convergence of the network topology. If STP has been enabled, run the **no spanning-tree** command to disable STP. Enable STP only after completing the instance-VLAN mapping configuration.

1.4.4 Procedure

(1) Create VLANs.

a Enter the privileged EXEC mode.

enable

b Enter the global configuration mode.

configure terminal

c Create a group of VLANs.

vlan { *vlan-id* | **range** *vlan-range* }

Only VLAN 1 exists by default.

d Return to the global configuration mode.

exit

(2) Configure instance-VLAN mappings.

a Enter the MST configuration mode.

spanning-tree mst configuration

b Add the VLAN group to an MST instance.

instance *instance-id* **vlan** *vlan-range*

Only instance **0** exists and all VLANs belong to instance **0** by default.

(3) (Optional) Configure a name of an MST region.

name *name*

The default name of an MST region is empty.

(4) (Optional) Configure a revision number for the MST region.

revision *version*

The default revision number of an MST region is **0**.

1.5 Configuring the Bridge Priority and Port Priority

1.5.1 Overview

The bridge priority concerns the root bridge election as well as the topology of the entire network. Configure the bridge priority when the administrator needs to change the root or topology of a network.

The port priority concerns the port role election as well as ports entering forwarding state. Configure the port priority when the administrator needs to change ports entering forwarding state. If a loop arises in a region, the port with a higher priority is preferentially selected to transition to forwarding state. When ports share the same priority, the port with a smaller port ID is selected to transition to forwarding state.

1.5.2 Restrictions and Guidelines

- The value range of an instance ID is from 0 to 64 and the default value is **0**.
- The options of bridge priority indicated by **priority** include 16 values, which are multiples of 4096, that is, 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. The default value is **32768**. It is recommended that the administrator configure a smaller bridge priority for core devices to ensure the network stability. Different bridge priorities can be configured for different instances, and each instance runs an independent STP based on the configuration. Devices in different regions care about the priority of only instance 0.
- The options of **port-priority** include 16 values, which are multiples of 16, that is, 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. The default value is **128**. Different port priorities can be configured for different instances on the same port, and an independent spanning tree can be generated for each instance based on the configuration. The changed port priority takes effect only if the port priority is changed on designated ports.

1.5.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the bridge priority.

- ○ In STP/RSTP mode, configure the bridge priority.

  spanning-tree priority *priority*

  The default bridge priority is **32768**.

- ○ In MSTP mode, configure the bridge priority for a specified instance.

  spanning-tree mst *instance-id* **priority** *priority*

  The default bridge priority is **32768**.

(4) Enter the interface configuration mode.

interface *interface-type interface-number*

(5) Configure the port priority.

- In STP/RSTP mode, configure the port priority.

  spanning-tree port-priority *port-priority*

  The default port priority is **128**.

- In MSTP mode, configure the port priority for a specified instance.

  spanning-tree mst *instance-id* **port-priority** *port-priority*

  The default port priority is **128**.

1.6 Configuring the Port Path Cost

1.6.1 Overview

The port path cost determines the root port election result on a device. A device elects the port with the minimum root path cost as the root port. The root path cost is the sum of path costs of all ports in the path from a device to the root.

The port path cost also affects the BPDU priority as well as the overall network topology. When devices compare the BPDU priority, the root path cost is compared if BPDUs have the same root ID and bridge ID.

The administrator can configure the port path cost to determine the port or path, through which packets pass preferentially. You can manually configure the port path cost by modifying the method of calculating the default path cost or directly specifying the path cost.

1.6.2 Restrictions and Guidelines

By default, the port path cost is automatically calculated based on the interface link rate. A higher link rate indicates a lower path cost and vice versa. With this calculation method, devices can calculate the optimal spanning tree topology. You are not advised to modify the path cost unless otherwise indicated.

1.6.3 Configuration Tasks

The configuration includes the following tasks:

(1) Configuring the Path Cost Calculation Method

(2) Configuring the Path Cost Value

1.6.4 Configuring the Path Cost Calculation Method

1. Overview

If no specific path cost is configured, the device automatically calculates the port path cost based on the physical port rate by using the calculation method stipulated in the standard. The administrator can modify the method of automatically calculating the port path cost for a device so that the device calculates different port path cost values.

2. Restrictions and Guidelines

- Be sure to adopt a consistent port path cost standard for the entire network.
- IEEE 802.1d Short: The range of the port path cost is from 1 to 65535. Aggregate port cost = Physical port cost × 95%.

- IEEE 802.1t Long: The range of the port path cost is from 1 to 200000000. Aggregate port cost = Physical port cost × 95%.
- IEEE 802.1t Long Standard: The range of the port path cost is from 1 to 200000000. Aggregate port cost = Physical port cost/Linkupcnt. At this moment, the cost value of the aggregate port will change with the number of member ports, which will lead to the network topology change. For configurations of aggregate ports and the Link Aggregation Control Protocol (LACP), see *Configuring Aggregate Port*.
  - When an aggregate port is a static aggregate port, **Linkupcnt** refers to the number of member ports in Link Up state.
  - When an aggregate port is an LACP aggregate port, **Linkupcnt** refers to the number of member ports participating in the aggregate port data forwarding.
  - When no member port of an aggregate port is in Link Up state or forwarding data, the value of **Linkupcnt** is **1**.

Table 1-1Port Path Costs Calculated Based on the Link Rate

Port Rate	Port	IEEE 802.1d Short	IEEE 802.1t Long	IEEE 802.1t Long Standard
10 Mbps	Common port	100	2000000	2000000
	Aggregate port	95	1900000	2000000/Linkupcnt
100 Mbps	Common port	19	200000	200000
	Aggregate port	18	190000	200000/Linkupcnt
1000 Mbps	Common port	4	20000	20000
	Aggregate port	3	19000	20000/Linkupcnt
10000 Mbps	Common port	2	2000	2000
	Aggregate port	1	1900	20000/Linkupcnt

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Modify the automatic calculation method.

spanning-tree pathcost method { **long** | **long standard** | **short** }

The port path cost is calculated according to IEEE 802.1t Long by default.

1.6.5 Configuring the Path Cost Value

1. **Overview**

You can manually specify the port path cost. After configuration, the port path costs automatically calculated based on the physical port rate are not used.

2. **Restrictions and Guidelines**

- In MSTP application, you can configure different path costs for different instances on an interface so that different instance spanning trees are generated for the instances to implement load sharing.
- **mst** *instance-id*: Specifies the instance ID. The value range is from 0 to 64 and the default value is **0**.
- **cost** *cost*: Specifies the path cost value. The value range is from 1 to 200000000. The path cost value is automatically calculated based on the link rate of a port by default. A larger value of *cost* indicates a greater path cost.

3. **Procedure**

(1) Enter the privileged EXEC mode.

enable

(1) Enter the global configuration mode.

configure terminal

(2) Enter the interface configuration mode.

interface *interface-type interface-number*

(3) Configure the port path cost.

○ In STP/RSTP mode, configure the port path cost.

spanning-tree cost *cost*

The port path cost is calculated based on the port rate by default.

○ In MSTP mode, configure the port path cost for a specified instance.

spanning-tree mst *instance-id* **cost** *cost*

The port path cost is calculated based on the port rate by default.

1.7 Configuring Spanning Tree Time Parameters

1.7.1 Overview

The timer in STP affects the spanning tree election and recovery performance.

Multiple parameters cannot be configured in one command at the same time. Only one of the following time parameters can be carried in the **spanning-tree** command:

- Interval for entering forwarding state (**Forward Time**)
- BPDU transmission interval (**Hello Time**)
- BPDU expiration time (**Max Age**)
- Number of BPDUs sent per second (**TX Hold Count**)

- Spanning tree timeout factor (**Factor**)

1.7.2 Restrictions and Guidelines

- **forward-time** *forward*: Specifies the port status change interval, in seconds. It refers to the interval for a port to transition from listening state to learning state or from learning state to forwarding when STP runs on a device or RSTP runs in STP compatible mode. It indicates the delay in the port status change. The value range is from 4 to 30, and the default value is **15**. If **Forward Delay** is too small, a temporary loop may occur. If **Forward Delay** is too large, the network recovery time may be very long.
- **hello-time** *hello*: Indicates the interval for the device to send BPDUs, in seconds. The value range is from 1 to 10, and the default value is **2**. If **Hello Time** is too small, the resource consumption of the device and network may increase. If the value is too small, the downstream device may mistakenly consider the uplink to be faulty, especially when packet loss occurs occasionally.
- **max-age** *age*: Indicates the maximum expiration time of BPDUs, in seconds. When the time expires, BPDUs will be discarded. The value range is from 6 to 40, and the default value is **20**. If **Max Age** is too small, link congestion may be wrongly judged as a link fault and the spanning tree is calculated very frequently. If the value is too large, the link fault cannot be found in time, resulting in poor network auto-recovery capability.

  Each device running STP or RSTP is a separate region, and is not allocated to a region with any other devices. Though MSTP uses the hop count mechanism to represent the lifecycle of BPDUs, the **Message Age** and **Max Age** mechanisms are still retained in order to be compatible with STP and RSTP outside regions.
- The value ranges of **forward-time, hello-time, and max-age** are related. Changing one of them will affect the value ranges of the other two parameters. Their restrictive relationship is as follows: 2 × (Hello Time + 1s) ≤ Max Age ≤ 2 × (Forward Delay – 1s). The configured three parameters must meet this condition. Otherwise, the topology may be unstable. In addition, the three parameters need to match the network size. A larger network diameter needs larger values of the parameters. You are advised to adopt automatic calculation according to the protocol to avoid manual configuration.
- **tx-hold-count** *count*: Configures the maximum number of BPDUs that can be sent per second. The value range is from 1 to 10 and the default value is **3**.
- **timer-factor** *factor*: Configures the packet receiving timeout factor. Timeout time = Timeout factor (indicated by *factor*) × Hello Time. If a device fails to receive a BPDU from the upstream device within the timeout time, the spanning tree is re-calculated. The value range is from 1 to 30, and the default value is **3**.

1.7.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the interval for the status change of a spanning tree port.

spanning-tree forward-time *forward*

The default interval for transitioning to forwarding state is 15s.

(4) Configure the interval for a spanning tree port to send BPDUs.

spanning-tree hello-time *hello*

The default interval for sending BPDUs is 2s.

(5) Configure the maximum number of BPDUs that can be sent by a spanning tree port.

spanning-tree tx-hold-count *count*

Three BPDUs are sent per second by default.

(6) Configure the maximum timeout time of spanning tree BPDUs.

spanning-tree max-age *age*

The default expiration time of BPDUs is 20s.

(7) Configure the spanning tree timeout factor.

spanning-tree timer-factor *factor*

The default spanning tree timeout factor is 3.

1.8 Configuring the Maximum Hop Count for BPDUs

1.8.1 Overview

MST regions do not use the **Message Age** and **Max Age** mechanisms to calculate whether BPDUs time out but use the maximum hop count mechanism for calculation. The maximum hop count mechanism is similar to the Time to Live (TTL) of IP packets. The maximum hop count of BPDUs affects the BPDU lifetime, and thus affects the network topology.

In an MST region, the BPDU hop count decreases by 1 each time the BPDU passes through one device from the region root until the remaining hop count is 0, indicating that the BPDU times out. Devices discard received BPDUs with the hop count of 0.

The maximum hop count of BPDUs is 20 by default. The value does not need to be changed for a network with the scale less than 20 hops, but needs to be changed to match the actual network situation when the network scale is greater than 20 hops. Changing the maximum hop count will affect all instances.

1.8.2 Restrictions and Guidelines

Hop-count:-count: Indicates the number of devices that a BPDU can pass through before it is discarded. The value range is from 1 to 40 and the default value is **20**.

1.8.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the maximum hop count for BPDUs.

spanning-tree max-hops *hop-count*

The maximum hop count of BPDUs is 20 by default.

1.9 Configuring Fast Convergence of Spanning Trees

1.9.1 Overview

In a full-duplex point-to-point link, an interface supporting RSTP first enters discarding state after port role election is complete, and then performs a Proposal/Agreement handshake with the connected bridge to quickly enter forwarding state.

Only ports in point-to-point connection state support fast convergence. Ports in shared connection state do not support fast convergence. See Fast Convergence of RSTP.

1.9.2 Restrictions and Guidelines

- If the connection type is not configured, the device automatically sets the connection type of an interface based on the duplex state of the interface. When an interface works in full-duplex mode, the connection type of the interface is point-to-point. When an interface works in half-duplex mode, the connection type of the interface is shared (non point-to-point).
- You can forcibly configure the interface connection type. The parameter **point-to-point** indicates the point-to-point connection type and **shared** indicates the shared connection type.

1.9.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure fast convergence for spanning trees.

spanning-tree link-type { **point-to-point** | **shared** }

The default connection type of an interface is auto mode. If an interface works in full-duplex mode, the connection type is point-to-point. If an interface works in half-duplex mode, the connection type is shared.

1.10 Enabling Protocol Migration

1.10.1 Overview

1. Compatibility of Different Protocols

If a device running STP receives an RST BPDU or MST BPDU, it ignores or discards the BPDU, resulting in incompatibility between the device running STP and other devices running RSTP or MSTP.

However, a device running RSTP or MSTP can recognize and process STP protocol packets from other devices. RSTP and MSTP are capable of automatically determining the protocol type supported by the connected bridge based on the version of received BPDUs. If the peer supports STP, after port role election is complete, RSTP and MSTP wait for twice the period of **Forward Delay** (30s by default) before entering forwarding state according to the STP status transition mechanism.

RSTP can process the CIST part of MST BPDUs. Therefore, MSTP does not have to send RST BPDUs to ensure RSTP compatibility.

Each device running STP or RSTP is a separate region, and does not belong to the same region with any other devices.

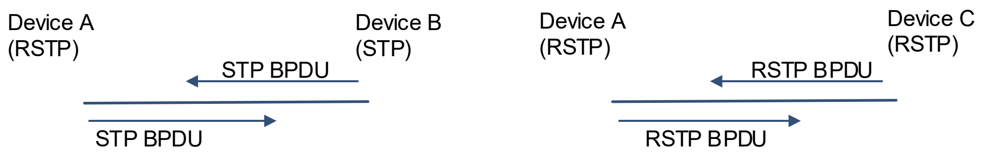
2. Protocol Migration Principle

When both RSTP and STP are used, a problem shown in Figure 1-1 may occur: Device A supports RSTP, Device B supports only STP, and the two devices are connected to each other. Device A judges that the peer device supports only STP, and sends STP BPDUs to ensure compatibility with the peer device. After Device B is replaced with Device C supporting RSTP, Device A still runs STP and sends STP BPDUs to the peer device. In this case, Device C mistakenly judges that Device A supports only STP. Finally, both devices run STP and the spanning tree convergence speed is greatly reduced. Hence, the protocol migration function emerges.

The protocol migration function enabled on Device C clears the original protocol and forces Device C to send RST BPDUs. Device A switches to the RSTP mode when finding that the connected bridge supports RSTP. Then, the two devices run RSTP.

When finding that the peer device supports RSTP, the administrator can enable the protocol migration function to force the local device to send RST BPDUs to achieve protocol migration of the local and peer devices.

Figure 1-1Forcing Protocol Migration to RSTP



1.10.2 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Clear the original protocol and force the device to migrate to the RSTP protocol.

clear spanning-tree detected-protocols [**interface** *interface-type interface-number*]

1.11 Configuring Spanning Tree Compatibility for Interfaces

1.11.1 Overview

After the spanning tree compatibility function is enabled on an interface, STP calculates whether the interface participates in the calculation of a specified instance based on the VLAN, to which the interface belongs, and the mapping between the VLAN and the instance. When the interface sends a BPDU, only the MSTI configuration message of the instance calculated by the interface, is carried to ensure compatibility with other devices.

For example, instances 1 and 2 exist on a device. Port GigabitEthernet 0/1 belongs only to VLAN 10, and VLAN 10 belongs to instance 1. If the spanning tree compatibility mode is enabled on port GigabitEthernet 0/1, the BPDU sent by port GigabitEthernet 0/1 carries only information of instance 0 (the port participates in calculation of this instance by default) and instance 1, with no information of instance 2.

1.11.2 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure spanning tree compatibility for an interface.

spanning-tree compatible enable

The spanning tree compatibility function of interfaces is disabled by default.

1.12 Enabling the STP Function

1.12.1 Overview

Unless otherwise specified, STP should be enabled on each device.

1.12.2 Restrictions and Guidelines

STP and the Transparent Interconnection of Lots of Links (TRILL) protocol of data centers are mutually exclusive.

1.12.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable the STP function.

spanning-tree

The STP function is disabled by default.

1.13 Configuring Port Protection

1.13.1 Overview

A root bridge may receive a configuration BPDU with a higher priority due to a misconfiguration or a malicious attack using illegitimate packets in the network, and lose the current root bridge role. This can cause an incorrect network topology change. To prevent this situation, you can configure the root guard function on designated ports of the root bridge.

If the root port, master port, and alternate port on a non-root bridge periodically receive BPDUs with a higher priority from the upstream bridge, the spanning tree topology will keep unchanged. If they fail to receive high-priority BPDUs from the upstream bridge, they judge that the upstream device is faulty. Then, the bridge considers itself as the root and sets all ports as designated ports, which cause loops. To prevent this situation, you can configure the loop guard function on the above ports to enhance the device stability.

1.13.2 Restrictions and Guidelines

Root guard and loop guard cannot take effect on a port at the same time.

1.13.3 Configuration Tasks

The configuration includes the following tasks: The configuration steps below are mutually exclusive. Select one of them for configuration.

- Configuring Root Guard
- Configuring Loop Guard

1.13.4 Configuring Root Guard

1. Overview

In the network design, the root bridge and backup root bridge are usually classified into the same region. Designated ports on a root bridge may receive configuration BPDUs with a higher priority due to a misconfiguration or a malicious attack, and the root bridge loses the current root bridge role. As a result, incorrect network topology change is incurred. To prevent this situation, you can configure the root guard function on designated ports of the root bridge.

After the root guard function is enabled, the device interfaces are designated ports in all instances. If a port receives a high-priority BPDU, the port enters blocking state due to root-inconsistent. If the port fails to receive a high-priority BPDU within a period of time, it returns to the normal state.

When a port enters blocking state due to root guard, you can manually restore the port to the normal state by using two methods:

- Run the **no spanning-tree guard root** command to disable the root guard function on the port.
- Run the **spanning-tree guard none** command to disable the guard function on the port.

⚠ **Caution**

- When the root guard function is configured on a non-designated port, the function forcibly sets the non-designated port as a designated port. The port will inevitably receive a high-priority BPDU according to the network design, and the port will enter blocking state due to root-inconsistent. Therefore, configuring root guard on a non-designated port can cause a communication failure.
- If a port receives a BPDU with a higher priority in an instance, it enters blocking state in this instance. If a port receives a BPDU with a higher priority in instance **0**, the port enters blocking state in all instances.

2. Restrictions and Guidelines

- In interface configuration mode, run the **spanning-tree guard root** command to enable the root guard function on an interface, or run the **no spanning-tree guard root** command to disable the root guard function of the interface.
- When a port is blocked due to root guard, you can run the **spanning-tree guard none** command in interface configuration mode to disable the root guard function and restore the port to the normal state.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface { *interface-type interface-number* | **range** *interface-type interface-range* }

(4) Enable the root guard function.

spanning-tree guard root

The root guard function is disabled on an interface by default.

1.13.5 Configuring Loop Guard

1. Overview

The root port or alternate port of a non-root bridge may fail to receive BPDUs due to a unidirectional link failure, and the port becomes a designated port and enters forwarding state. As a result, loops occur in the network. To prevent this situation, you can configure loop guard on a non-root bridge.

After loop guard is enabled, when the root port or alternate port fails to receive BPDUs and changes to a designated port, the port will remain in discarding state until it receives a BPDU for spanning tree calculation.

⚠ **Caution**

- You can enable loop guard globally or on an interface.
- Before the MSTP process is restarted, a port enters the loop guard blocking state. After the MSTP process is restarted, if the port still fails to receive BPDUs, the port changes to a designated port and enters forwarding state. Therefore, you are advised to figure out the cause that a port enters the loop guard blocking state and rectify the fault before the MSTP process is restarted. Otherwise, the spanning tree topology is still abnormal after the MSTP process is restarted.

2. Restrictions and Guidelines

- In global configuration mode, run the **spanning-tree loopguard default** command to enable the loop guard function on all interfaces, or run the **no spanning-tree loopguard default** command to disable the loop guard function of all interfaces.
- In interface configuration mode, run the **spanning-tree guard loop** command to enable the loop guard function on an interface or run the **no spanning-tree guard loop** command to disable the loop guard function of the interface.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable the loop guard function. Select one of the following to configure.

- In global configuration mode, enable the loop guard function on all interfaces.

  spanning-tree loopguard default

  The loop guard function is disabled on an interface by default.

- In interface configuration mode, enable the loop guard function on one or a group of interfaces. Run the following commands in sequence:

  interface { *interface-type interface-number* | **range** *interface-type interface-range* }

  spanning-tree guard loop

  The loop guard function is disabled on an interface by default.

1.14 Configuring BPDU Source MAC Address Check

1.14.1 Overview

This section describes how to enable the BPDU source MAC address check function to prevent BPDU attacks. After the function is enabled on a device, the device receives only BPDU frames from specified source MAC addresses and discards other BPDU frames.

1. BPDU Source MAC Address Check

If the peer device connects to the local device in a point-to-point manner and the MAC address of the peer device is certain, the BPDU source MAC address check function can be configured on the local device. After this function is enabled, the device receives only BPDU frames matching the designated source MAC address and discards all the other BPDU frames. Therefore, when the device encounters BPDU packet attacks, illegitimate BPDU packets can be identified and discarded to prevent the MSTP function failure due to the attacks.

2. Filtering of BPDUs with Invalid Length

If the Ethernet length field in a BPDU frame exceeds 1,500, this illegitimate BPDU frame will be discarded.

1.14.2 Restrictions and Guidelines

- In interface configuration mode, run the **bpdu src-mac-check** *H.H.H* command to enable the BPDU source MAC address check function on an interface, or run the **no bpdu src-mac-check** command to disable the BPDU source MAC address check function of the interface.
- *H.H.H* indicates that only BPDU frames with the source MAC address matching this address are received. Only one filter MAC address can be configured for one interface.
- If the configured MAC address is incorrect, required BPDU frames may be discarded.

1.14.3 Prerequisites

Configure an interface as an L2 switching port before configuring this function.

1.14.4 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface { *interface-type interface-number* | **range** *interface-type interface-range* }

(4) Enable BPDU source MAC address check so that only BPDUs with the source MAC address matching *H.H.H* are received.

bpdu src-mac-check *H.H.H*

The BPDU source MAC address check function is disabled on an interface by default.

1.15 Configuring an Edge Port

1.15.1 Overview

After the spanning tree role election is complete, a designated port needs to wait for twice the period of **Forward Delay** (2 × 15 = 30s) before entering forwarding state. Designated ports can be converted into edge ports via automatic recognition or manual configuration. Edge ports can rapidly enter forwarding state without waiting for twice the period of **Forward Delay**. If an edge port receives a BPDU, it transitions from forwarding state to disabled state, changes to a non-edge port, and participates in the spanning tree calculation.

- Autoedge

  If the autoedge function is enabled on a designated port and the port fails to receive a BPDU from the downstream device within a certain period of time (3s), the port judges that the connected peer device is a network terminal. Then, the designated port changes to an edge port and rapidly enters forwarding state.

- Port fast

  If you are sure that a designated port is directly connected to a network terminal, you can manually configure port fast rather than rely on the autoedge function. A manually configured edge port rapidly enters forwarding state without waiting for twice the period of **Forward Delay**.

1.15.2 Restrictions and Guidelines

When autoedge conflicts with the edge port manually configured via port fast, the manual configuration prevails.

Unless otherwise specified, do not disable the autoedge function.

1.15.3 Configuration Tasks

The configuration includes the following tasks:

- Configuring Autoedge
- Configuring the Port Fast Attribute

1.15.4 Configuring Autoedge

1. Overview

After the spanning tree role election is complete, a designated port needs to wait for twice the period of **Forward Delay** (2 × 15 = 30s) before entering forwarding state. If the autoedge function is enabled on a designated port and the port fails to receive a BPDU from the downstream device within a certain period of time (3s), the port judges that the connected peer device is a network terminal. Then, the designated port changes to an edge port and directly enters forwarding state. If an edge port receives a BPDU, it transitions from forwarding state to disabled state, changes to a non-edge port, and participates in the spanning tree calculation.

⚠ **Caution**

- STP does not support this function.
- This function applies only to designated ports, and takes effect only in the process of fast negotiation between a designated port and a downstream port and the designated port has not entered forwarding state in this process. If a designated port has been in forwarding state, the autoedge function does not take effect. For example, if the BPDU filter function is enabled on a port and the port has transitioned to forwarding state, the autoedge function does not take effect. It takes effect only when the designated port restarts fast negotiation (for example, the network cable is removed and then inserted).

2. Restrictions and Guidelines

The **disabled** parameter is used to disable the autoedge function.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure the autoedge function.

spanning-tree autoedge [**disabled**]

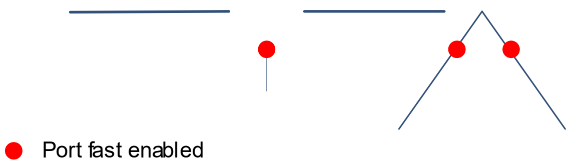
The autoedge function is enabled by default.

1.15.5 Configuring the Port Fast Attribute

1. Overview

After the spanning tree role election is complete, a designated port needs to wait for twice the period of **Forward Delay** (2 × 15 = 30s) before entering forwarding state. The autoedge function is enabled on a designated port by default. If the port fails to receive a BPDU from the downstream device within a period of time (3s), it is automatically recognized as an edge port. If a designated port fails to receive a BPDU within the timeout time due to poor network condition, packet loss, or packet transmission/receiving delay, the device mistakenly judges that the designated port connects to a network terminal and automatically identifies the designated port as an edge port. In this network condition, you are advised to disable the autoedge function, and manually configure the interface connected to a terminal as an edge port according to the network plan. When you are sure that a designated port is directly connected to a network terminal, configure the port fast attribute manually rather than rely on the autoedge function. A manually configured edge port enters forwarding state without waiting for twice the period of **Forward Delay**. If an edge port receives a BPDU, it transitions from forwarding state to disabled state, changes to a non-edge port, and participates in the spanning tree calculation.

Figure 1-1Positions of Edge Ports



2. Restrictions and Guidelines

- In global configuration mode, run the **no spanning-tree portfast default** command to restore an interface to a non-edge port.
- In interface configuration mode, run the **spanning-tree portfast disabled** command to configure an interface as a non-edge port.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure an interface as an edge port. Select one of the following to configure.

- In global configuration mode, configure all interfaces as edge ports.

  spanning-tree portfast default

  Interfaces are non-edge ports by default.

- In interface configuration mode, configure a designated port as an edge port. Run the following commands in sequence:

  interface { *interface-type interface-number* | **range** *interface-type interface-range* }

  spanning-tree portfast

  Interfaces are non-edge ports by default.

1.16 Configuring BPDU Guard or BPDU Filter

1.16.1 Overview

If a device interface is directly connected to a network terminal, BPDU guard or BPDU filter can be configured on the interface to prevent illegitimate access, BPDU attacks, or loops in downstream devices. BPDU guard and BPDU filter can be enabled globally and take effect only on edge ports, or enabled and take effect only on specific interfaces.

1.16.2 Restrictions and Guidelines

BPDU filter has a higher priority than BPDU guard. If both BPDU filter and BPDU guard are enabled on an interface, only BPDU filter takes effect.

1.16.3 Configuration Tasks

The configuration includes the following tasks:

- Configuring BPDU Guard
- Configuring BPDU Filter

1.16.4 Configuring BPDU Guard

1. Overview

Edge ports do not receive BPDUs in normal cases. If an edge port is attacked by forged BPDUs or an illegitimate device is added to the network, the edge port will receive BPDUs, change to a non-edge port, and participate in spanning tree calculation, resulting in network flapping.

The BPDU guard function aims to defend against BPDU packet attacks and illegitimate access. If this function is enabled, a port enters the error-disabled state when receiving a BPDU, indicating that a port exception occurs. Then, the device disables the port, which can be restored to the normal state only through the errdisable recovery function.

BPDU guard can be enabled globally or on specific interfaces. The global BPDU guard function takes effect only on edge ports. The BPDU guard function takes effect on interfaces regardless of whether the interfaces are edge ports.

2. Restrictions and Guidelines

- In the global BPDU guard configuration command, the **no** parameter is used to disable the global BPDU guard function.
- In the interface BPDU guard configuration command, the **disabled** parameter is used to disable the BPDU guard function on a specific interface.
- A port in error-disabled state can be restored automatically or manually. The **errdisable recovery** [**interval** *seconds*] command is used to configure the auto-recovery interval, in seconds. The value range is from 30 to 86400. The errdisable recovery function is disabled on a port by default and no default recovery interval is available. To restore a port in error-disabled state manually, run the **errdisable recovery** command. See *Configuring Ethernet Interface*.
- Run the **show spanning-tree** command to display the configuration.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the BPDU guard function. Select one of the following to configure.

- ○ Enable the global BPDU guard function. The function takes effect only on edge ports.

  spanning-tree portfast bpduguard default

  The global BPDU guard function is disabled by default.

- ○ Enable the BPDU guard function. Run the following commands in sequence:

interface { *interface-type interface-number* | **range** *interface-type interface-range* }

spanning-tree bpduguard enabled

The BPDU guard function is disabled by default.

1.16.5 Configuring BPDU Filter

1. **Overview**

BPDU filter is a method of preventing BPDU attacks. When BPDU filter is enabled, a port neither sends nor receives BPDUs, but directly enters forwarding state. If a port receives a BPDU, it transitions from forwarding state to disabled state, the BPDU filter function automatically fails, and the port participates in spanning tree calculation.

BPDU filter can be enabled globally or on specific interfaces. The global BPDU filter function takes effect only on edge ports. The BPDU guard function takes effect on interfaces regardless of whether the interfaces are edge ports.

2. **Restrictions and Guidelines**

- In the global BPDU filter configuration command, the **no** parameter is used to disable the global BPDU filter function.
- In the interface BPDU filter configuration command, the **enabled** parameter is used to enable the BPDU filter function and the **disabled** parameter is used to disable the BPDU filter function.

3. **Procedure**

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the BPDU filter function. Select one of the following to configure.

○ Enable the global BPDU filter function. The function takes effect only on edge ports.

spanning-tree portfast bpdufilter default

The global BPDU filter function is disabled by default.

○ Enable the BPDU filter function. Run the following commands in sequence:

interface { *interface-type interface-number* | **range** *interface-type interface-range* }

spanning-tree bpdufilter enabled

The BPDU filter function is disabled by default.

1.17 Configuring TC Attack Defense Functions

1.17.1 Overview

When the downstream network topology changes, a port generates a TC packet (that is, TCN BPDU; see 1.1 Table 1-4 for details) to notify the upstream device of the spanning tree change. After receiving the TCN BPDU, the port copies the BPDU and forwards it to upstream devices until the root bridge receives the BPDU. This is called TC diffusion. After receiving a TC packet, a device deletes dynamic MAC addresses and ARP entries that have been learned, and an L3 device also enables the fast forwarding module to change the port status of ARP entries.

If a device is attacked by forged TC packets, it frequently performs the deletion operation, which occupies excessive device resources and makes the device overburdened. After TC packet attacks diffuse to the whole network, the performance of devices throughout the network will deteriorate and the network stability will be affected. Therefore, TC protection, TC guard, and TC filter arise to solve this problem.

- TC protection

After TC protection is enabled on a device, the device performs only one deletion operation within a period of time (generally 4s) after receiving TC packets, and monitors whether any TC packet is received in this period. If the device receives TC packets in this period, it performs another deletion operation after the period expires, to prevent frequent deletion of MAC address entries and ARP entries.

- TC guard

TC protection can reduce the frequency of deleting dynamic MAC address entries and ARP entries, but many deletion operations still need to be performed in the case of TC packet attacks, which affects the normal operation of devices. In addition, TC packets still are diffused in the network and the entire network is affected. For this, the TC guard function emerges.

After TC guard is enabled on a port, the port will not diffuse TC packets to other ports of the device that participate in spanning tree calculation, so as to effectively control possible TC attacks in the network and retain the network stability. Especially on L3 devices, this function can effectively prevent interruption of core routes caused by access device flapping.

There are two situations of no diffusion: TC packets received by a port from other ports are not diffused; TC packets generated by a port are not diffused. TC packets generated by a port refer that, when the port status changes (for example, transition from blocking state to forwarding state), the port generates TC packets, indicating that the topology may have changed.

⚠ **Caution**

- The TC guard function will also limit the propagation of normal TC packets. When a spanning tree network topology changes, devices cannot obtain topology change information and the topology cannot be restored, resulting in a communication interruption. Therefore, it is recommended that the TC guard function be enabled only when the network is attacked by illegitimate TC packets.
- If the global TC guard function is enabled, all ports will not diffuse TC packets. The global TC guard function is applicable to desktop access devices.
- If the interface TC guard function is enabled, the port will not diffuse TC packets. The interface TC guard function is applicable to uplink ports, especially the ports that connect aggregation devices to core devices.

- TC filter

TC guard blocks the diffusion of all TC packets, including normal TC packets generated in the case of a topology change. As a result, devices in the network cannot perceive the topology change and fail to clear MAC address entries and ARP entries of relevant ports in time, and errors occur in data forwarding. Hence, the TC filter function emerges.

TC filter refers that a port does not process TC packets received from other ports but processes TC packets generated due to the normal topology change of the port.

This function solves the problems of MAC address entry clearing and core route interruption caused by frequent up/down state switching of ports with port fast not configured. It also ensures that core routing entries are updated in time when a topology change occurs.

1.17.2 Restrictions and Guidelines

It is recommended that the TC attack defense functions be enabled only when the network is attacked by TC packets.

1.17.3 Configuration Tasks

The configuration includes the following tasks:

- Configuring TC Protection
- Configuring TC Guard
- Configuring TC Filter

1.17.4 Configuring TC Protection

1. Overview

After TC protection is enabled, the device performs only one deletion operation within a period of time (generally 4s) after receiving TC packets. This can prevent frequent deletion of MAC address entries and ARP entries.

2. Restrictions and Guidelines

TC protection can be enabled or disabled only globally and takes effect on all interfaces. Run the **spanning-tree tc-protection** command to enable TC protection or run the **no spanning-tree tc-protection** command to disable TC protection.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable TC protection on all interfaces.

spanning-tree tc-protection

TC protection is disabled by default.

1.17.5 Configuring TC Guard

1. Overview

After TC guard is enabled on a port, the port shields received TC packets and the TC packets generated by the port so that the TC packets are not diffused to other ports. In this way, possible TC attacks in the network are effectively contained, ensuring network stability.

2. Restrictions and Guidelines

- In global configuration mode, the TC guard function configuration takes effect on all interfaces. Run the

spanning-tree tc-protection tc-guard command to enable the TC guard function or run the **no spanning-tree tc-protection tc-guard** command to disable the TC guard function.

- In interface configuration mode, the TC guard function configuration takes effect only on specific interfaces. Run the **spanning-tree tc-guard** command to enable the TC guard function or run the **no spanning-tree tc-guard** command to disable the TC guard function.
- If no TC attack exists in a network but TC guard is configured, the spread of normal TC packets is contained. When the topology changes, the device fails to clear learned MAC addresses and a packet forwarding error may occur.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the TC guard function. Select one of the following to configure.

- Enable TC guard on all interfaces.

  spanning-tree tc-protection tc-guard

  TC guard is disabled by default.

- Enable TC guard on a specific interface. Run the following commands in sequence:

  interface { *interface-type interface-number* | **range** *interface-type interface-range* }

  spanning-tree tc-guard

  TC guard is disabled by default.

1.17.6 Configuring TC Filter

1. Overview

TC filter enables a port not to process received TC packets but to process TC packets generated by the port due to topology changes.

2. Restrictions and Guidelines

TC filter can be enabled or disabled only on interfaces and takes effect on specific interfaces. In interface configuration mode, run the **spanning-tree ignore tc** command to enable the TC filter function or run the **no spanning-tree ignore tc** command to disable the TC filter function.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface { *interface-type interface-number* | **range** *interface-type interface-range* }

(4) Enable TC filter on a specific interface.

spanning-tree ignore tc

TC filter is disabled by default.

1.18 Configuring BPDU Tunnel

1.18.1 Overview

BPDU tunnel is an L2 tunnel function based on the Layer 2 Tunnel Protocol (L2TP).

A 802.1Q (QinQ) network is usually divided into the customer network and service provider network. The basic principle of QinQ is described as follows: When a user packet is sent to a provider edge (PE), the PE considers the VLAN tag in the user packet as data, and encapsulates the VLAN tag of the service provider network into the packet so that the user packet carries two VLAN tags to traverse the service provider network. In the service provider network, the packet is forwarded based on the outer VLAN tag. When the packet leaves the service provider network, the outer VLAN tag is removed.

STP BPDUs use the BPDU dedicated address 0180.c200.0000 as the destination MAC address. Therefore, the BPDUs of the customer network and service provider network have the same destination MAC address. When a tunnel interface of a PE receives a BPDU from the customer network, the PE recognizes the BPDU as an STP packet and does not forward it because the destination MAC address of the BPDU is the BPDU dedicated address (0180.c200.0000). In addition, the PE performs spanning tree calculation based on the BPDU content, which affects the topology of the service provider network. In order to prevent the customer network BPDUs from interfering with the spanning tree calculation of the service provider network, you need to differentiate customer network BPDUs from service provider network BPDUs, that is, modify the destination MAC address in customer network BPDUs.

After the BPDU tunnel function is enabled on a tunnel interface of a PE, if the interface receives a customer network BPDU, it changes the destination MAC address in the BPDU from the BPDU dedicated address (0180.c200.0000) to the tunnel address (01d0.f800.0005 by default) and forwards the BPDU in the service provider network. When the BPDU reaches a PE at the other end, the PE restores the destination MAC address of the BPDU from the tunnel address (01d0.f800.0005 by default) to the BPDU dedicated address (0180.c200.0000) and forwards the BPDU to the peer customer network. Customer network BPDUs are transmitted through BPDU tunnels in the service provider network. In this way, the STP calculations of the customer network and service provider network are performed separately without interfering with each other. For details, see *Configuring QinQ* and *Configuring VPDN*.

1.18.2 Restrictions and Guidelines

- This function takes effect only when it is enabled in both global configuration mode and interface configuration mode.
  - Run the **l2protocol-tunnel stp** command to enable the global BPDU tunnel function or run the **no l2protocol-tunnel stp** command to disable the global BPDU tunnel function.
  - Run the **l2protocol-tunnel stp enable** command to enable the BPDU tunnel function or run the **no l2protocol-tunnel stp enable** command to disable the BPDU tunnel function.
- Run the **l2protocol-tunnel stp tunnel-dmac** *mac-address* command to configure a BPDU tunnel address. The optional tunnel addresses of STP packets include 01d0.f800.0005 (default), 011a.a900.0005,

010f.e200.0003, 0100.0ccd.cdd0, 0100.0ccd.cdd1, and 0100.0ccd.cdd2.

1.18.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable the global BPDU tunnel function.

l2protocol-tunnel stp

The global BPDU tunnel function is disabled by default.

(4) (Optional) Configure a BPDU tunnel address.

l2protocol-tunnel stp tunnel-dmac *mac-address*

The default BPDU tunnel address is 01d0.f800.0005.

(5) Enter the interface configuration mode of the port that connects the customer network.

interface *interface-type interface-number*

(6) Enable the BPDU tunnel function on the interface.

l2protocol-tunnel stp enable

The BPDU tunnel function is disabled by default.

1.19 Configuring BPDU Transparent Transmission

1.19.1 Overview

According to the IEEE 802.1Q standard, BPDUs use 0180.c200.0000 as the destination MAC address. When a device supporting IEEE 802.1Q receives a frame with the destination address of 0180.c200.0000, it recognizes the frame as a BPDU and will not forward it.

However, in the actual network deployment, some BPDU frames need to be transparently transmitted by devices. For example, STP is disabled on Device A but enabled on Device B and Device C connected through Device A. In this case, Device A needs to transparently transmit BPDU frames so that Device B and Device C can normally calculate and generate spanning trees.

1.19.2 Restrictions and Guidelines

- BPDU transparent transmission takes effect only when STP is disabled. If STP is enabled on a device, the device does not transparently transmit BPDU frames. If STP is enabled on a service provider network device and the device is required to transmit customer network BPDUs without affecting the spanning tree of the service provider network, use the BPDU tunnel function.
- In global configuration mode, run the **bridge-frame forwarding protocol bpdu** command to enable the BPDU transparent transmission function, or run the **no bridge-frame forwarding protocol bpdu** command to disable the BPDU transparent transmission function.

1.19.3 Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable the BPDU transparent transmission function.

bridge-frame forwarding protocol bpdu

BPDU transparent transmission is disabled by default.

1.20 Monitoring

Run the **show** commands to check the running status of a configured function to verify the configuration effect.

Run the **debug** commands to output debugging information.

⚠ **Caution**

The output debugging information occupies system resources. Therefore, disable the debugging switch immediately after use.

Run the **clear** commands to clear information.

⚠ **Caution**

Running the **clear** commands may lose vital information and thus interrupt services.

Table 1-1Monitoring

Command	Purpose
show spanning-tree [**forward-time** | **hello-time** | **max-age** | **max-hops** | **mst** *instance-id* | **pathcost method** | **tx-hold-count**]	Displays the global spanning tree configuration.
show spanning-tree counters	Displays statistics on sent and received STP packets.
show spanning-tree inconsistentports	Displays ports blocked due to root guard or loop guard.
show spanning-tree summary	Displays the spanning tree topology and port forwarding status.
show spanning-tree [**mst** *instance-id*] **interface** *interface-type interface-number* [**bpdufilter** | **bpduguard** | **link-type** | **portfast**]	Displays the spanning tree configuration and status information of an interface by interface type and number.
show spanning-tree [**mst** *instance-id*] *port-index*	Displays the spanning tree configuration and status information of an interface by interface number.
show spanning-tree mst configuration	Displays the MST region configuration.
show spanning-tree mst *instance-id* **topochange record**	Displays the topology change information of an interface in a specified instance.
show l2protocol-tunnel stp	Displays BPDU tunnel information.

Command	Purpose
debug mstp all	Debugs STP, RSTP, and MSTP.
debug mstp gr	Debugs the spanning tree GR function.
debug mstp rx	Debugs the BPDU receiving.
debug mstp tx	Debugs BPDU transmission.
debug mstp event	Debugs spanning tree events.
debug mstp loopguard	Debugs spanning tree loop guard.
debug mstp rootguard	Debugs spanning tree root guard.
debug mstp bridgedetect	Debugs the bridge detect state machine.
debug mstp portinfo	Debugs the port information state machine.
debug mstp protomigrat	Debugs the port protocol migration state machine.
debug mstp topochange	Debugs the spanning tree topology changes.
debug mstp receive	Debugs the spanning tree receiving state machine.
debug mstp roletran	Debugs the port role transition state machine.
debug mstp statetran	Debugs the port state transition state machine.
debug mstp transmit	Debugs the MSTP transmission state machine.
clear spanning-tree counters [**interface** *interface-type interface-number*]	Clears statistics on packets sent and received by a port.
clear spanning-tree mst *instance-id* **topochange record**	Clears the STP topology change information.

1.21 Configuration Examples

1.21.1 Configuring MSTP

1. Requirements

The topology shown in Figure 1-1 includes two layers: Device A and Device B are in the core layer, Device C is in the access layer, and Device C connects to intranet terminals. There are four VLANs in the intranet. The configured MSTP needs to meet the following requirements:

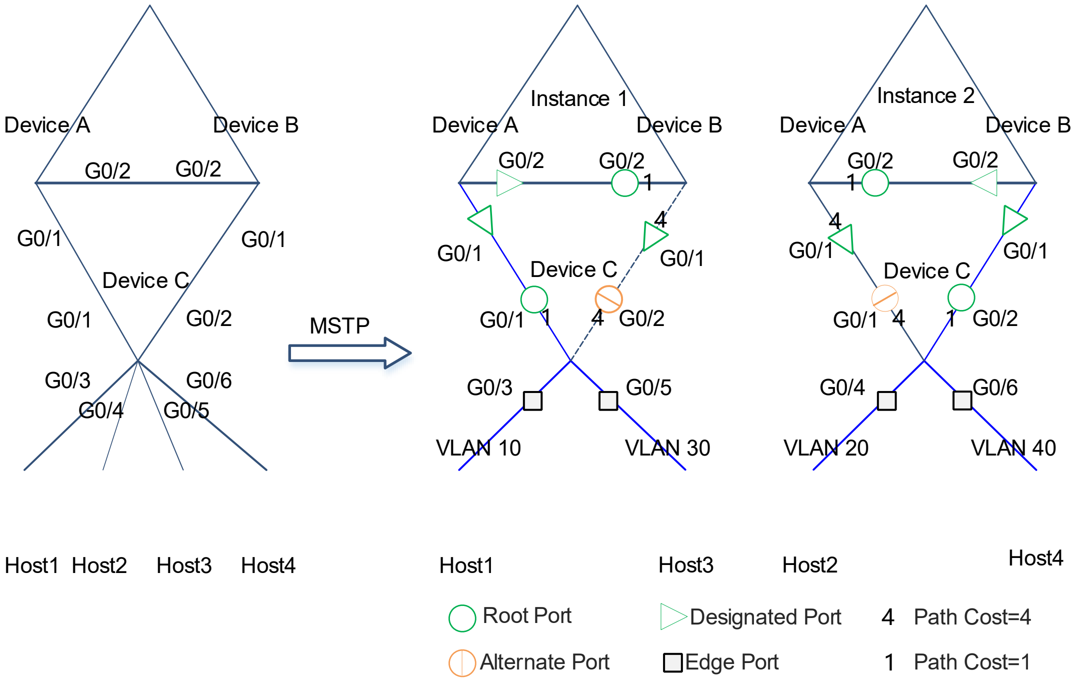
- The spanning tree root bridge of VLAN 10 and VLAN 30 is Device A and data is forwarded through port GigabitEthernet 0/1 of Device C.
- The spanning tree root bridge of VLAN 20 and VLAN 40 is Device B and data is forwarded through port

GigabitEthernet 0/2 of Device C.

- Enable loop guard on the root ports of Device C. Enable BPDU guard on the edge ports of Device C that connect to terminals.

2. Topology

Figure 1-1MSTP Basic Topology



3. Notes

- Create the same VLANs on Devices A, B, and C and configure the same instance mappings: Map instance 1 to VLANs 10 and 30 and map instance 2 to VLANs 20 and 40. Instance 0 contains other VLANs that have not been created.

- The devices elect the device roles and port roles in the spanning tree by comparing the priority vector <**Root Identifier**, **Root Path Cost**, **Bridge ID**, **Port ID**>. Configure the bridge priorities and port path costs for instances so that the required topology is calculated. For the ease of management, configure the same spanning tree for instance 0 and instance 1.

  Instance 0 and instance 1:

  - Set the bridge priority to 4096 for Device A and 8192 for Device B, and use the default bridge priority 32768 (no configuration is required) for Device C so that Device A becomes the root bridge.
  - On Device B, set the path cost to 1 for port GigabitEthernet 0/2 and 4 for port GigabitEthernet 0/1 so that port GigabitEthernet 0/2 becomes the root port of Device B.
  - On Device C, set the path cost to 1 for port GigabitEthernet 0/1 and 4 for port GigabitEthernet 0/2 so that port GigabitEthernet 0/1 becomes the root port of Device C.
  - The bridge priority of Device B is 8192, which is higher than that of Device C (32768). Therefore, port

GigabitEthernet 0/1 of Device B is elected as the designated port and port GigabitEthernet 0/2 of Device C is elected as the alternate port.

Instance 2:

  - Set the bridge priority to 4096 for Device B and 8192 for Device A, and use the default bridge priority 32768 (no configuration is required) for Device C so that Device B becomes the root bridge.
  - On Device A, set the path cost to 1 for port GigabitEthernet 0/2 and 4 for port GigabitEthernet 0/1 so that port GigabitEthernet 0/2 becomes the root port of Device A.
  - On Device C, set the path cost to 1 for port GigabitEthernet 0/2 and 4 for port GigabitEthernet 0/1 so that port GigabitEthernet 0/2 becomes the root port of Device C.
  - The bridge priority of Device A is 8192, which is higher than that of Device C (32768). Therefore, port GigabitEthernet 0/1 of Device A is elected as the designated port and port GigabitEthernet 0/1 of Device C is elected as the alternate port.

- Configure MSTP loop guard on ports GigabitEthernet 0/1–0/2 of Device C. When the root port or backup port fails to receive BPDUs and changes to a designated port, the port will remain in discarding state until it receives BPDUs for spanning tree calculation.
- On Device C, configure ports GigabitEthernet 0/3–0/6 that connect to terminals as edge ports. The autoedge function is enabled by default. If ports GigabitEthernet 0/3–0/6 fail to receive BPDUs within 3s after being elected as designated ports, they are automatically recognized as edge ports and immediately enters forwarding state. Packet loss or packet transmission/receiving delay in the network may affect the autoedge function. Therefore, disable the autoedge function, manually configure the ports as edge ports, and enable the BPDU guard function.
- Enable the STP function globally on Devices A, B, and C and use the default MSTP mode.
- Configure interconnected ports among Devices A, and B, and C as trunk ports and configure the ports to allow all VLAN traffic to pass. Add the interfaces of Device C that connect to terminals to their respective VLANs.

4. Procedure

(1) Configure Device A.

Create VLANs and configure instance mappings.

```
DeviceA>enable
DeviceA# configure terminal
DeviceA(config)# vlan range 10,20,30,40
DeviceA(config-vlan-range)# exit
DeviceA(config)# spanning-tree mst configuration
DeviceA(config-mst)# instance 1 vlan 10,30
DeviceA(config-mst)# instance 2 vlan 20,40
```

Set the bridge priority to 4094 for instances 0 and 1 and 8192 for instance 2.

```
DeviceA(config-mst)# spanning-tree mst 0 priority 4096
DeviceA(config)# spanning-tree mst configuration
DeviceA(config-mst)# spanning-tree mst 1 priority 4096
DeviceA(config)# spanning-tree mst configuration
DeviceA(config-mst)# spanning-tree mst 2 priority 8192
```

Configure port GigabitEthernet 0/2 as a trunk port. In instance 2, set the path cost of the port to 1.

```
DeviceA(config)# interface range gigabitethernet 0/2
DeviceA(config-if-GigabitEthernet 0/2)# switchport
DeviceA(config-if-GigabitEthernet 0/2)# switchport mode trunk
DeviceA(config-if-GigabitEthernet 0/2)# spanning-tree mst 2 cost 1
DeviceA(config-if-GigabitEthernet 0/2)# exit
```

Configure port GigabitEthernet 0/1 as a trunk port. In instance 2, set the path cost of the port to 4.

```
DeviceA(config)# interface range gigabitethernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)# switchport
DeviceA(config-if-GigabitEthernet 0/1)# switchport mode trunk
DeviceA(config-if-GigabitEthernet 0/1)# spanning-tree mst 2 cost 4
DeviceA(config-if-GigabitEthernet 0/1)# exit
```

Enable the STP function globally.

```
DeviceA(config)# spanning-tree
DeviceA(config)# end
DeviceA# write
```

(2) Configure Device B.

Create VLANs and configure instance mappings.

```
DeviceB> enable
DeviceB# configure terminal
DeviceB(config)# vlan range 10,20,30,40
DeviceB(config-vlan-range)# exit
DeviceB(config)# spanning-tree mst configuration
DeviceB(config-mst)# instance 1 vlan 10,30
DeviceB(config-mst)# instance 2 vlan 20,40
```

Set the bridge priority to 4094 for instance 2 and 8192 for instance 0 and instance 1.

```
DeviceB(config-mst)# spanning-tree mst 0 priority 8192
DeviceB(config)# spanning-tree mst configuration
DeviceB(config-mst)# spanning-tree mst 1 priority 8192
DeviceB(config)# spanning-tree mst configuration
DeviceB(config-mst)# spanning-tree mst 2 priority 4096
```

Configure port GigabitEthernet 0/2 as a trunk port. In instances 0 and 1, set the path cost of the port to 1.

```
DeviceB(config)# interface range gigabitethernet 0/2
DeviceB(config-if-GigabitEthernet 0/2)# switchport
DeviceB(config-if-GigabitEthernet 0/2)# switchport mode trunk
DeviceB(config-if-GigabitEthernet 0/2)# spanning-tree mst 0 cost 1
DeviceB(config-if-GigabitEthernet 0/2)# spanning-tree mst 1 cost 1
DeviceB(config-if-GigabitEthernet 0/2)# exit
```

Configure port GigabitEthernet 0/1 as a trunk port. In instances 0 and 1, set the path cost of the port to 4.

```
DeviceB(config)# interface range gigabitethernet 0/1
DeviceB(config-if-GigabitEthernet 0/1)# switchport
DeviceB(config-if-GigabitEthernet 0/1)# switchport mode trunk
DeviceB(config-if-GigabitEthernet 0/1)# spanning-tree mst 0 cost 4
DeviceB(config-if-GigabitEthernet 0/1)# spanning-tree mst 1 cost 4
DeviceB(config-if-GigabitEthernet 0/1)# exit
```

Enable the STP function globally.

```
DeviceB(config)# spanning-tree
DeviceB(config)# end
DeviceB# write
```

(3) Configure Device C.

Create VLANs and configure instance mappings.

```
DeviceC> enable
DeviceC# configure terminal
DeviceC(config)# vlan range 10,20,30,40
DeviceC(config-vlan-range)# exit
DeviceC(config)# spanning-tree mst configuration
DeviceC(config-mst)# instance 1 vlan 10,30
DeviceC(config-mst)# instance 2 vlan 20,40
DeviceC(config-mst)# exit
```

Configure uplink port GigabitEthernet 0/1 as a trunk port. In instances 0 and 1, set the path cost of the port to 1. In instance 2, set the path cost of the port to 4. Configure loop guard on the uplink port.

```
DeviceC(config)# interface gigabitethernet 0/1
DeviceC(config-if-GigabitEthernet 0/1)# switchport
DeviceC(config-if-GigabitEthernet 0/1)# switchport mode trunk
DeviceC(config-if-GigabitEthernet 0/1)# spanning-tree mst 0 cost 1
DeviceC(config-if-GigabitEthernet 0/1)# spanning-tree mst 1 cost 1
DeviceC(config-if-GigabitEthernet 0/1)# spanning-tree mst 2 cost 4
DeviceC(config-if-GigabitEthernet 0/1)# spanning-tree guard loop
DeviceC(config-if-GigabitEthernet 0/1)# exit
```

Configure uplink port GigabitEthernet 0/2 as a trunk port. In instance 2, set the path cost of the port to 1. In instances 0 and 1, set the port path cost to 4. Configure loop guard on the uplink port.

```
DeviceC(config)# interface gigabitethernet 0/2
DeviceC(config-if-GigabitEthernet 0/2)# switchport
DeviceC(config-if-GigabitEthernet 0/2)# switchport mode trunk
DeviceC(config-if-GigabitEthernet 0/2)# spanning-tree mst 0 cost 4
DeviceC(config-if-GigabitEthernet 0/2)# spanning-tree mst 1 cost 4
DeviceC(config-if-GigabitEthernet 0/2)# spanning-tree mst 2 cost 1
DeviceC(config-if-GigabitEthernet 0/2)# spanning-tree guard loop
DeviceC(config-if-GigabitEthernet 0/2)# exit
```

Add downlink ports GigabitEthernet 0/3–0/6 to VLANs. Configure the ports as edge ports and configure BPDU guard.

```
DeviceC(config)# interface gigabitethernet 0/3
DeviceC(config-if-GigabitEthernet 0/3)# switchport
DeviceC(config-if-GigabitEthernet 0/3)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/3)# switchport access vlan 10
DeviceC(config-if-GigabitEthernet 0/3)# spanning-tree autoedge disabled
DeviceC(config-if-GigabitEthernet 0/3)# spanning-tree portfast
DeviceC(config-if-GigabitEthernet 0/3)# spanning-tree bpduguard enable
DeviceC(config-if-GigabitEthernet 0/3)# exit
DeviceC(config)# interface gigabitethernet 0/4
DeviceC(config-if-GigabitEthernet 0/4)# switchport
DeviceC(config-if-GigabitEthernet 0/4)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/4)# switchport access vlan 20
DeviceC(config-if-GigabitEthernet 0/4)# spanning-tree autoedge disabled
DeviceC(config-if-GigabitEthernet 0/4)# spanning-tree portfast
DeviceC(config-if-GigabitEthernet 0/4)# spanning-tree bpduguard enable
DeviceC(config-if-GigabitEthernet 0/4)# exit
DeviceC(config)# interface gigabitethernet 0/5
DeviceC(config-if-GigabitEthernet 0/5)# switchport
DeviceC(config-if-GigabitEthernet 0/5)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/5)# switchport access vlan 30
DeviceC(config-if-GigabitEthernet 0/5)# spanning-tree autoedge disabled
DeviceC(config-if-GigabitEthernet 0/5)# spanning-tree portfast
DeviceC(config-if-GigabitEthernet 0/5)# spanning-tree bpduguard enable
DeviceC(config-if-GigabitEthernet 0/5)# exit
DeviceC(config)# interface gigabitethernet 0/6
DeviceC(config-if-GigabitEthernet 0/6)# switchport
DeviceC(config-if-GigabitEthernet 0/6)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/6)# switchport access vlan 40
DeviceC(config-if-GigabitEthernet 0/6)# spanning-tree autoedge disabled
DeviceC(config-if-GigabitEthernet 0/6)# spanning-tree portfast
DeviceC(config-if-GigabitEthernet 0/6)# spanning-tree bpduguard enable
DeviceC(config-if-GigabitEthernet 0/6)# exit
```

Enable the STP function globally.

```
DeviceC(config)# spanning-tree
DeviceC(config)# end
DeviceC# write
```

5. Verification

(1) Verify that instance mappings on the devices are the same.

Check instance mappings on Device A.

```
DeviceA# show spanning-tree mst configuration
Multi spanning tree protocol : Enable
Name     :
Revision : 0
Instance  Vlans Mapped
-------- --------------------------------------------
0        : 1-9, 11-19, 21-29, 31-39, 41-4094
1        : 10, 30
```

```
2        : 20, 40
-----------------------------------------------------
```

Check instance mappings on Device B.

```
DeviceB# show spanning-tree mst configuration
Multi spanning tree protocol : Enable
Name     :
Revision : 0
Instance  Vlans Mapped
-------- --------------------------------------------
0        : 1-9, 11-19, 21-29, 31-39, 41-4094
1        : 10, 30
2        : 20, 40
-----------------------------------------------------
```

Check instance mappings on Device C.

```
DeviceC# show spanning-tree mst configuration
Multi spanning tree protocol : Enable
Name     :
Revision : 0
Instance  Vlans Mapped
-------- --------------------------------------------
0        : 1-9, 11-19, 21-29, 31-39, 41-4094
1        : 10, 30
2        : 20, 40
-----------------------------------------------------
```

(2) Check the instance spanning tree topology and port forwarding status on Device A.

In instances 0 and 1, Device A (0074.9cee.f49e) is the root bridge. Ports GigabitEthernet 0/1–0/2 of Device A are designated ports. They are all in forwarding state.

```
DeviceA# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     0074.9cee.f49e
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    4096
             Address     0074.9cee.f49e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p
```

```
MST 1 vlans map : 10, 30
  Region Root Priority   4096
             Address     0074.9cee.f49e
             this bridge is region root

  Bridge ID  Priority    4096
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p
```

In instance 2, Device B (00d0.f8ee.8c1e) is the root bridge. Port GigabitEthernet 0/2 of Device A is the root port and the port path cost is 1. Port GigabitEthernet 0/1 of Device A is the designated port and the port path cost is 4. The two ports are in forwarding state.

```
MST 2 vlans map : 20, 40
  Region Root Priority   4096
             Address     00d0.f8ee.8c1e
             this bridge is region root

  Bridge ID  Priority    8192
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/1            Desg FWD 4          128      False    P2p
Gi0/2            Root FWD 1          128      False    P2p
```

(3) Check the instance spanning tree topology and port forwarding status on Device B.

In instances 0 and 1, Device A (0074.9cee.f49e) is the root bridge. Port GigabitEthernet 0/1 of Device B is the root port and the port path cost is 1. Port GigabitEthernet 0/2 of Device B is the designated port and the port path cost is 4. The two ports are in forwarding state.

```
DeviceB# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     0074.9cee.f49e
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    8192
             Address     00d0.f8ee.8c1e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
```

```
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 1          128      False    P2p
Gi0/2            Desg FWD 4          128      False    P2p

MST 1 vlans map : 10, 30
  Region Root Priority   4096
             Address     0074.9cee.f49e
             this bridge is region root

  Bridge ID  Priority    8192
             Address     00d0.f8ee.8c1e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 1          128      False    P2p
Gi0/2            Desg FWD 4          128      False    P2p
```

In instance 2, Device B (00d0.f8ee.8c1e) is the root bridge. Ports GigabitEthernet 0/1–0/2 of Device B are designated ports. They are all in forwarding state.

```
MST 2 vlans map : 20, 40
  Region Root Priority   4096
             Address     00d0.f8ee.8c1e
             this bridge is region root

  Bridge ID  Priority    4096
             Address     00d0.f8ee.8c1e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p
```

(4) Check the instance spanning tree topology and port forwarding status on Device C.

In instances 0 and 1, Device A (0074.9cee.f49e) is the root bridge. Port GigabitEthernet 0/1 of Device C is the root port and the port path cost is 1. The port is in forwarding state. Port GigabitEthernet 0/2 of Device C is the alternate port and the port path cost is 4. The port is in blocking state. Ports GigabitEthernet 0/3–0/6 of Device C are edge ports. They are all in forwarding state.

```
DeviceC# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     0074.9cee.f49e
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 1          128      False    P2p
Gi0/2            Altn BLK 4          128      False    P2p
Gi0/3            Desg FWD 20000      128      True     P2p
Gi0/4            Desg FWD 20000      128      True     P2p
Gi0/5            Desg FWD 20000      128      True     P2p
Gi0/6            Desg FWD 20000      128      True     P2p

MST 1 vlans map : 10, 30
  Region Root Priority   4096
             Address     0074.9cee.f49e
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 1          128      False    P2p
Gi0/2            Altn BLK 4          128      False    P2p
Gi0/3            Desg FWD 20000      128      True     P2p
Gi0/4            Desg FWD 20000      128      True     P2p
Gi0/5            Desg FWD 20000      128      True     P2p
Gi0/6            Desg FWD 20000      128      True     P2p
```

In instance 2, Device B (00d0.f8ee.8c1e) is the root bridge. Port GigabitEthernet 0/2 of Device C is the root port and the port path cost is 1. The port is in forwarding state. Port GigabitEthernet 0/1 of Device C is the alternate port and the port path cost is 4. The port is in blocking state. Ports GigabitEthernet 0/3–0/6 of Device C are edge ports. They are all in forwarding state.

```
MST 2 vlans map : 20, 40
  Region Root Priority   4096
             Address     00d0.f8ee.8c1e
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Altn BLK 4          128      False    P2p
Gi0/2            Root FWD 1          128      False    P2p
Gi0/3            Desg FWD 20000      128      True     P2p
Gi0/4            Desg FWD 20000      128      True     P2p
Gi0/5            Desg FWD 20000      128      True     P2p
```

```
Gi0/6            Desg FWD 20000      128      True      P2p
```

(5) Check "DesignatedRoot", "RootPort", and "RootCost" of each device in instance 1. Check the "DesignatedRoot", "RootPort", and "RootCost" of each device in instances 0 and 2.

Device A is the root and the root path cost is 0.

```
DeviceA# show spanning-tree mst 1
###### MST 1 vlans mapped : 10, 30
BridgeAddr : 0074.9cee.f49e
Priority: 4096
TimeSinceTopologyChange : 0d:0h:16m:29s
TopologyChanges : 7
DesignatedRoot : 4097.0074.9cee.f49e
RootCost : 0
RootPort : 0
```

The root path cost from Device B to the root (4097.0074.9cee.f49e) through root port GigabitEthernet 0/2 is 1.

```
DeviceB# show spanning-tree mst 1
###### MST 1 vlans mapped : 10, 30
BridgeAddr : 00d0.f8ee.8c1e
Priority: 8192
TimeSinceTopologyChange : 0d:0h:16m:27s
TopologyChanges : 7
DesignatedRoot : 4097.0074.9cee.f49e
RootCost : 1
RootPort : GigabitEthernet 0/2
```

The root path cost from Device C to the root (4097.0074.9cee.f49e) through root port GigabitEthernet 0/1 is 1.

```
DeviceC# show spanning-tree mst 1
###### MST 1 vlans mapped : 10, 30
BridgeAddr : 0074.9cee.53ca
Priority: 32768
TimeSinceTopologyChange : 0d:0h:19m:48s
TopologyChanges : 5
DesignatedRoot : 4097.0074.9cee.f49e
RootCost : 1
RootPort : GigabitEthernet 0/1
```

(6) Verify that the loop guard function ("PortGuardmode:Guard loop") on uplink ports GigabitEthernet 0/1–0/2 of Device C is enabled. The following uses GigabitEthernet 0/1 as an example.

```
DeviceC# show spanning-tree interface gigabitethernet 0/1
PortAdminPortFast : Disabled
PortOperPortFast : Disabled
PortAdminAutoEdge : Enabled
PortOperAutoEdge : Disabled
PortAdminLinkType : auto
PortOperLinkType : point-to-point
```

```
PortBPDUGuard : Disabled
PortBPDUFilter : Disabled
PortGuardmode  : Guard loop

###### MST 0 vlans mapped :1-9, 11-19, 21-29, 31-39, 41-4094
PortState : forwarding
PortPriority : 128
PortDesignatedRoot : 4096.0074.9cee.f49e
PortDesignatedCost : 0
PortDesignatedBridge :4096.0074.9cee.f49e
PortDesignatedPortPriority : 128
PortDesignatedPort : 1
PortForwardTransitions : 1
PortAdminPathCost : 1
PortOperPathCost : 1
Inconsistent states : normal
PortRole : rootPort

###### MST 1 vlans mapped :10, 30
PortState : forwarding
PortPriority : 128
PortDesignatedRoot : 4097.0074.9cee.f49e
PortDesignatedCost : 0
PortDesignatedBridge :4097.0074.9cee.f49e
PortDesignatedPortPriority : 128
PortDesignatedPort : 1
PortForwardTransitions : 1
PortAdminPathCost : 1
PortOperPathCost : 1
Inconsistent states : normal
PortRole : rootPort

###### MST 2 vlans mapped :20, 40
PortState : discarding
PortPriority : 128
PortDesignatedRoot : 4098.00d0.f8ee.8c1e
PortDesignatedCost : 0
PortDesignatedBridge :8194.0074.9cee.f49e
PortDesignatedPortPriority : 128
PortDesignatedPort : 1
PortForwardTransitions : 3
PortAdminPathCost : 4
PortOperPathCost : 4
Inconsistent states : normal
PortRole : alternatePort
```

(7) Verify that port fast is configured on ports GigabitEthernet 0/3–0/6 of Device C ("PortAdminPortFast : Enabled"), the ports work in port fast state ("PortOperPortFast : Enabled"), and BPDU guard is enabled on the ports (“PortBPDUGuard: Enabled”). The following uses port GigabitEthernet 0/3 as an example.

```
DeviceC# show spanning-tree interface gigabitethernet 0/3

PortAdminPortFast : Enabled
PortOperPortFast : Enabled
PortAdminAutoEdge : Disabled
PortOperAutoEdge : Enabled
PortAdminLinkType : auto
PortOperLinkType : point-to-point
PortBPDUGuard : Enabled
PortBPDUFilter : Disabled
PortGuardmode  : None

###### MST 0 vlans mapped :1-9, 11-19, 21-29, 31-39, 41-4094
PortState : forwarding
PortPriority : 128
PortDesignatedRoot : 4096.0074.9cee.f49e
PortDesignatedCost : 0
PortDesignatedBridge :32768.0074.9cee.53ca
PortDesignatedPortPriority : 128
PortDesignatedPort : 1
PortForwardTransitions : 1
PortAdminPathCost : 20000
PortOperPathCost : 20000
Inconsistent states : normal
PortRole : designatedPort

###### MST 1 vlans mapped :10, 30
PortState : forwarding
PortPriority : 128
PortDesignatedRoot : 4097.0074.9cee.f49e
PortDesignatedCost : 0
PortDesignatedBridge :32769.0074.9cee.53ca
PortDesignatedPortPriority : 128
PortDesignatedPort : 1
PortForwardTransitions : 1
PortAdminPathCost : 20000
PortOperPathCost : 20000
Inconsistent states : normal
PortRole : designatedPort

###### MST 2 vlans mapped :20, 40
PortState : forwarding
PortPriority : 128
```

```
PortDesignatedRoot : 4098.00d0.f8ee.8c1e
PortDesignatedCost : 0
PortDesignatedBridge :32770.0074.9cee.53ca
PortDesignatedPortPriority : 128
PortDesignatedPort : 1
PortForwardTransitions : 1
PortAdminPathCost : 20000
PortOperPathCost : 20000
Inconsistent states : normal
PortRole : designatedPort
```

6. Configuration Files

- Device A configuration file

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree mst 0 priority 4096
spanning-tree mst 1 priority 4096
spanning-tree mst 2 priority 8192
spanning-tree
!
sysmac 0074.9cee.f49e
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode trunk
 spanning-tree mst 2 cost 4
!
interface GigabitEthernet 0/2
 switchport mode trunk
 spanning-tree mst 2 cost 1
```

- Device B configuration file

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree mst 0 priority 8192
spanning-tree mst 1 priority 8192
spanning-tree mst 2 priority 4096
spanning-tree
!
```

```
sysmac 00d0.f8ee.8c1e
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode trunk
 spanning-tree mst 1 cost 4
 spanning-tree mst 0 cost 4
!
interface GigabitEthernet 0/2
 switchport mode trunk
 spanning-tree mst 1 cost 1
 spanning-tree mst 0 cost 1
```

- Device C configuration file

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree
!
sysmac 0074.9cee.53ca
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode trunk
 spanning-tree guard loop
 spanning-tree mst 2 cost 4
 spanning-tree mst 1 cost 1
 spanning-tree mst 0 cost 1
!
interface GigabitEthernet 0/2
 switchport mode trunk
 spanning-tree guard loop
 spanning-tree mst 2 cost 1
 spanning-tree mst 1 cost 4
 spanning-tree mst 0 cost 4
!
interface GigabitEthernet 0/3
 switchport access vlan 10
 spanning-tree bpduguard enable
 spanning-tree portfast
 spanning-tree autoedge disabled
!
interface GigabitEthernet 0/4
```

```
 switchport access vlan 20
 spanning-tree bpduguard enable
 spanning-tree portfast
 spanning-tree autoedge disabled
!
interface GigabitEthernet 0/5
 switchport access vlan 30
 spanning-tree bpduguard enable
 spanning-tree portfast
 spanning-tree autoedge disabled
!
interface GigabitEthernet 0/6
 switchport access vlan 40
 spanning-tree bpduguard enable
 spanning-tree portfast
 spanning-tree autoedge disabled
```

7. Common Errors

- When no bridge priority lower than that of the root bridge but higher than that of access devices is configured for non-root devices, the non-root core devices and access devices will determine upstream devices by comparing MAC addresses in the spanning tree calculation. As a result, the calculation result may not meet networking requirements.
- When no port path cost is configured for non-root devices, the spanning tree calculation result may not meet networking requirements due to different link conditions.
- The root guard function is configured on a root port, master port, or alternate port, and ports may be blocked incorrectly.

1.21.2 Configuring MSTP+VRRP

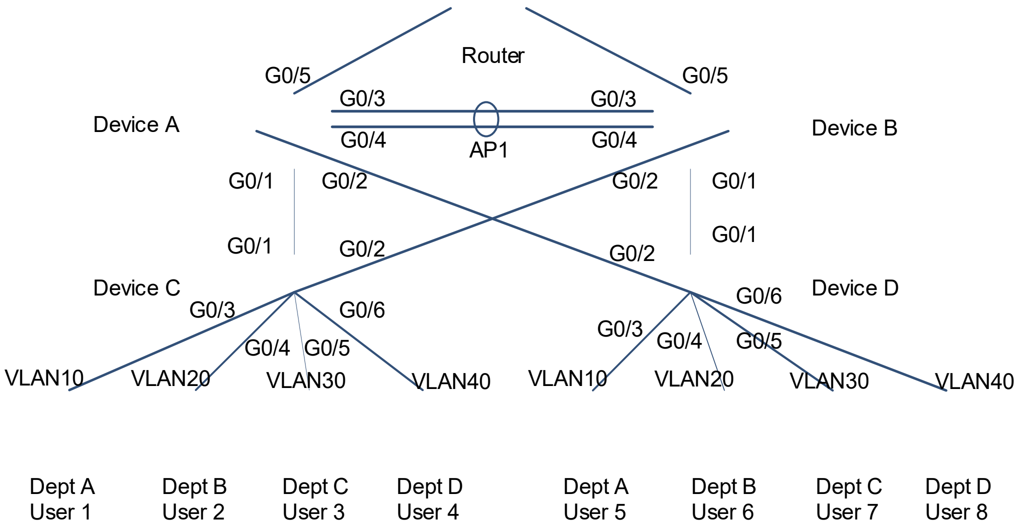
1. Requirements

The MSTP+VRRP dual-core solution is a typical application of MSTP. This solution uses a hierarchical network architecture and MSTP and Virtual Router Redundancy Protocol (VRRP) to implement redundant backup and VLAN load balancing, to enhance the network system availability. The highlight of this architecture is the hierarchical network structure. The capacity indicators, characteristics, and functions of network devices in each layer can be optimized based on their network locations and functions to boost the system stability and availability. This solution usually uses a three-layer (core layer, aggregation layer, and access layer) or two-layer (core layer and access layer) architecture. The topology shown in Figure 1-1 adopts a two-layer architecture. The networking requirements are as follows:

- Core layer: Configure multiple MSTP instances to achieve load balancing. Create instance 1 and instance 2. Map instance 1 to VLAN 10 and VLAN 30 and instance 2 to VLAN 20 and VLAN 40. Device A is the root bridge of instances 0 and instance 1 (instance 0 exists by default) as well as the VRRP master device of VLAN 10 and VLAN 30. Device B is the root bridge of instance 2 as well as the VRRP master device of VLANs 20 and 40.
- Access layer: Configure ports that are directly connected to terminals (such as PCs or servers) as edge ports and enable BPDU guard to prevent the access of unauthorized user devices.

2. Topology

Figure 1-1MSTP+VRRP Dual-Core Topology



3. Notes

- Configure MSTP.
  - Set the spanning tree mode to MSTP. The default spanning tree mode is MSTP and no configuration is required.
  - Configure an MST region. On Devices A, B, C, and D, map instance 1 to VLAN 10 and VLAN 30 and instance 2 to VLAN 20 and VLAN 40.
  - Set the bridge priority of instances 0 and 1 to 4096 on Device A and 8192 on Device B to make Device A become the root bridge of instances 0 and 1.
  - Set the bridge priority of instance 2 to 8192 on Device A and 4096 on Device B to make Device B become the root bridge of instance 2.
  - On access Devices C and D, instances 0, 1, and 2 use the default bridge priority (32768). Configure ports that connect to user terminals as edge ports so that they do not participate in spanning tree calculation. Enable BPDU guard on the ports.
- Configure monitoring ports for a VRRP group. Configure the uplink port of the master device as the VLAN monitoring port. Configure port GigabitEthernet 0/5 of Device A to monitor VLAN 10 and VLAN 30, and port GigabitEthernet 0/5 of Device B to monitor VLAN 20 and VLAN 40.
  - In interface configuration mode of the uplink port, run the **no switchport** command to configure the monitoring port as an L3 interface and configure an IP address for the port. L3 interfaces do not participate in spanning tree calculation.
  - In switch virtual interface (SVI) configuration mode of a VLAN, run the **vrrp** *group-id* **track** *interface-type interface-number* [*priority decrement*] command to configure a monitoring interface *interface-type interface-number* for the VRRP group *group-id* and VRRP priority change value *priority decrement*. *priority decrement* indicates the VRRP priority change value when the link status of the monitored interface or IP

route reachability status changes. When the link is interrupted, the priority is reduced. When the link recovers, the priority is restored. The value range is from 1 to 255 and the default value is **10**.

- Configure priority parameters: The VRRP priority change value needs to be taken into account together with the VRRP priority. In SVI configuration mode of a VLAN, run the **vrrp** *group-id* **priority** *priority* command to configure the VRRP priority. The value range of *priority* is from 1 to 254 and the default value is **100**. The default value of *priority decrement* is **10**. When the monitoring port is down, the VRRP priority is reduced to 90 (that is, 100 – 10).
  - In this example, raise the VRRP priority of VLANs 10 and 30 to 120 on the master device (Device A), use the default priority **100** on Device B, and set *priority decrement* to 30. When monitoring port GigabitEthernet 0/5 of Device A is down due to a fault, the VRRP priority of VLAN 10 and VLAN 30 is reduced by 30 to 90, which is lower than the default priority (100) on Device B. Therefore, data of VLAN 10 and VLAN 30 will be transmitted through Device B. When monitoring port GigabitEthernet 0/5 of Device A recovers, the VRRP priority of VLAN 10 and VLAN 30 is restored to 120, which is greater than the default priority on Device B. Therefore, data of VLAN 10 and VLAN 30 is transmitted through Device A. Then, VLAN 10 and VLAN 30 use Device A as the VRRP master device and Device B as the VRRP backup device.
  - Likewise, raise the VRRP priority of VLAN 20 and VLAN 40 to 120 on Device B and use the default priority (**100**) on Device A.
- Configure VRRP: Add SVIs of VLANs to a VRRP group and configure a virtual IP address for the VRRP group.
- Pay attention to the following when configuring SVI addresses: If the monitoring port is down and the VRRP priorities are the same on two devices after priority reduction, the SVI addresses of a VLAN on the two devices will be compared. A larger SVI address indicates a higher priority. Therefore, it is recommended that the SVI IP addresses of VLAN 10 and VLAN 30 on the master device (Device A) be greater than those of VLAN 10 and VLAN 30 on Device B. The same configuration applies to other VLANs.
- Configure an aggregate link between core devices. Configure the downlink ports of core devices and uplink ports of access devices as trunk ports. On access devices, configure ports that connect to user terminals as access ports and add them to VLANs.

4. Procedure

(1) Configure the MSTP function.

Configure the MSTP function for core Device A. Create VLANs, and map instance 1 to VLAN 10 and VLAN 30 and instance 2 to VLAN 20 and VLAN 40. Set the bridge priority to 4096 for instances 0 and 1 and 8192 for instance 2 to make Device A become the root bridge of instances 0 and 1. Enable MSTP.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA(config)# vlan range 10,20,30,40
DeviceA(config-vlan-range)# exit
DeviceA(config)# spanning-tree mode mstp
DeviceA(config)# spanning-tree mst configuration
DeviceA(config-mst)# instance 1 vlan 10,30
DeviceA(config-mst)# instance 2 vlan 20,40
DeviceA(config-mst)# exit
DeviceA(config)# spanning-tree mst 0 priority 4096
DeviceA(config)# spanning-tree mst 1 priority 4096
DeviceA(config)# spanning-tree mst 2 priority 8192
DeviceA(config)# spanning-tree
```

Configure the MSTP function for core Device B. Create VLANs, and map instance 1 to VLAN 10 and VLAN 30 and instance 2 to VLAN 20 and VLAN 40. Set the bridge priority to 8192 for instances 0 and 1 and 4096 for instance 2 to make Device B become the root bridge of instance 2. Enable MSTP.

```
DeviceB> enable
DeviceB# configure terminal
DeviceB(config)# vlan range 10,20,30,40
DeviceB(config-vlan-range)# exit
DeviceB(config)# spanning-tree mode mstp
DeviceB(config)# spanning-tree mst configuration
DeviceB(config-mst)# instance 1 vlan 10,30
DeviceB(config-mst)# instance 2 vlan 20,40
DeviceB(config-mst)# exit
DeviceB(config)# spanning-tree mst 0 priority 8192
DeviceB(config)# spanning-tree mst 1 priority 8192
DeviceB(config)# spanning-tree mst 2 priority 4096
DeviceB(config)# spanning-tree
```

Configure the MSTP function for access Devices C and D. The bridge priority does not need to be configured on access devices. Configure ports that are directly connected to user terminals as edge ports and enable BPDU guard on the ports. The configurations on Device D are similar to those on Device C. The following uses Device C as an example.

```
DeviceC> enable
DeviceC# configure terminal
DeviceC(config)# vlan range 10,20,30,40
DeviceC(config-vlan-range)# exit
DeviceC(config)# spanning-tree mode mstp
DeviceC(config)# spanning-tree mst configuration
DeviceC(config-mst)# instance 1 vlan 10,30
DeviceC(config-mst)# instance 2 vlan 20,40
DeviceC(config-mst)# exit
DeviceC(config)# spanning-tree
DeviceC(config)# interface range gigabitethernet 0/3-6
DeviceC(config-if-range)# spanning-tree portfast
DeviceC(config-if-range)# spanning-tree bpduguard enable
DeviceC(config-if-range)# exit
```

(2) Configure monitoring ports for the VRRP group.

Configure port GigabitEthernet 0/5 of Device A as a routing port and set the IP address to 10.10.1.1/24. Then, this port serves as the monitoring port of VLAN 10 and VLAN 30.

```
DeviceA(config)# interface gigabitethernet 0/5
DeviceA(config-if-GigabitEthernet 0/5)# no switchport
DeviceA(config-if-GigabitEthernet 0/5)# ip address 10.10.1.1 255.255.255.0
DeviceA(config-if-GigabitEthernet 0/5)# exit
```

Configure port GigabitEthernet 0/5 of Device B as a routing port and set the IP address to 10.10.2.1/24. Then, this port serves as the monitoring port of VLAN 20 and VLAN 40.

```
DeviceB(config)# interface gigabitethernet 0/5
DeviceB(config-if-GigabitEthernet 0/5)# no switchport
DeviceB(config-if-GigabitEthernet 0/5)# ip address 10.10.2.1 255.255.255.0
DeviceB(config-if-GigabitEthernet 0/5)# exit
```

(3) Configure VRRP for VLAN 10 and VLAN 30 on core devices.

Configure the master device (Device A). Enter an SVI and configure an SVI address. Add the SVI to the VRRP group and configure a virtual gateway IP address for the VRRP group. Raise the VRRP priority to 120, configure port GigabitEthernet 0/5 as a monitoring port, and set priority decrement to 30.

```
DeviceA(config)# interface vlan 10
DeviceA(config-if-VLAN 10)# ip address 192.168.10.3 255.255.255.0
DeviceA(config-if-VLAN 10)# vrrp 10 ip 192.168.10.1
DeviceA(config-if-VLAN 10)# vrrp 10 priority 120
DeviceA(config-if-VLAN 10)# vrrp 10 track gigabitethernet 0/5 30
DeviceA(config-if-VLAN 10)# exit
DeviceA(config)# interface vlan 30
DeviceA(config-if-VLAN 30)# ip address 192.168.30.3 255.255.255.0
DeviceA(config-if-VLAN 30)# vrrp 30 ip 192.168.30.1
DeviceA(config-if-VLAN 30)# vrrp 30 priority 120
DeviceA(config-if-VLAN 30)# vrrp 30 track gigabitethernet 0/5 30
DeviceA(config-if-VLAN 30)# exit
```

Configure the backup device (Device B). Enter an SVI and configure an SVI address. Add the SVI to the VRRP group and configure a virtual gateway IP address for the VRRP group. On the backup device, use the default VRRP priority (100). No monitoring port needs to be configured.

```
DeviceB(config)# interface vlan 10
DeviceB(config-if-VLAN 10)# ip address 192.168.10.2 255.255.255.0
DeviceB(config-if-VLAN 10)# vrrp 10 ip 192.168.10.1
DeviceB(config-if-VLAN 10)# exit
DeviceB(config)# interface vlan 30
DeviceB(config-if-VLAN 30)# ip address 192.168.30.2 255.255.255.0
DeviceB(config-if-VLAN 30)# vrrp 30 ip 192.168.30.1
DeviceB(config-if-VLAN 30)# exit
```

(4) Configure VRRP for VLAN 20 and VLAN 40 on core devices.

Configure the master device (Device B). Enter an SVI and configure an SVI address. Add the SVI to the VRRP group and configure a virtual gateway IP address for the VRRP group. Raise the VRRP priority to 120, configure port GigabitEthernet 0/5 as a monitoring port, and set priority decrement to 30.

```
DeviceB(config)# interface vlan 20
DeviceB(config-if-VLAN 20)# ip address 192.168.20.3 255.255.255.0
DeviceB(config-if-VLAN 20)# vrrp 20 ip 192.168.20.1
DeviceB(config-if-VLAN 20)# vrrp 20 priority 120
DeviceB(config-if-VLAN 20)# vrrp 20 track gigabitethernet 0/5 30
DeviceB(config-if-VLAN 20)# exit
DeviceB(config)# interface vlan 40
DeviceB(config-if-VLAN 40)# ip address 192.168.40.3 255.255.255.0
DeviceB(config-if-VLAN 40)# vrrp 40 ip 192.168.40.1
DeviceB(config-if-VLAN 40)# vrrp 40 priority 120
DeviceB(config-if-VLAN 40)# vrrp 40 track gigabitethernet 0/5 30
DeviceB(config-if-VLAN 40)# exit
```

Configure the backup device (Device A). Enter an SVI and configure an SVI address. Add the SVI to the VRRP group and configure a virtual gateway IP address for the VRRP group. On the backup device, use the default VRRP priority (100). No monitoring port needs to be configured.

```
DeviceA(config)# interface vlan 20
DeviceA(config-if-VLAN 20)# ip address 192.168.20.2 255.255.255.0
DeviceA(config-if-VLAN 20)# vrrp 20 ip 192.168.20.1
DeviceA(config-if-VLAN 20)# exit
DeviceA(config)# interface vlan 40
DeviceA(config-if-VLAN 40)# ip address 192.168.40.2 255.255.255.0
DeviceA(config-if-VLAN 40)# vrrp 40 ip 192.168.40.1
DeviceA(config-if-VLAN 40)# exit
```

(5) Configure an aggregate link between VRRP core devices.

On Device A, configure ports GigabitEthernet 0/3 and 0/4 as AggregatePort 1 and set the aggregate port to work in trunk mode.

```
DeviceA(config)# interface range gigabitethernet 0/3-4
DeviceA(config-if-range)# port-group 1
DeviceA(config-if-range)# exit
DeviceA(config-if-range)# interface aggregateport 1
DeviceA(config-if-AggregatePort 1)# switchport mode trunk
DeviceA(config-if-AggregatePort 1)# exit
```

On Device B, configure ports GigabitEthernet 0/3 and 0/4 as AggregatePort 1 and set the aggregate port to work in trunk mode.

```
DeviceB(config)# interface range gigabitethernet 0/3-4
DeviceB(config-if-range)# port-group 1
DeviceB(config-if-range)# exit
DeviceB(config-if-range)# interface aggregateport 1
DeviceB(config-if-AggregatePort 1)# switchport mode trunk
DeviceB(config-if-AggregatePort 1)# exit
```

(6) Configure the downlink ports of core devices and the uplink ports of access devices as trunk ports.

Configure downlink ports GigabitEthernet 0/1–0/2 of Device A as trunk ports.

```
DeviceA(config)# interface range gigabitethernet 0/1-2
DeviceA(config-if-range)# switchport mode trunk
DeviceA(config-if-range)# end
DeviceA# write
```

Configure downlink ports GigabitEthernet 0/1–0/2 of Device B as trunk ports.

```
DeviceB(config)# interface range gigabitethernet 0/1-2
DeviceB(config-if-range)# switchport mode trunk
DeviceB(config-if-range)# end
DeviceB# write
```

Configure uplink ports GigabitEthernet 0/1–0/2 of Device C as trunk ports.

```
DeviceC(config)# interface range gigabitethernet 0/1-2
DeviceC(config-if-range)# switchport mode trunk
DeviceC(config-if-range)# exit
```

Configure uplink ports GigabitEthernet 0/1–0/2 of Device D as trunk ports.

```
DeviceD(config)# interface range gigabitethernet 0/1-2
DeviceD(config-if-range)# switchport mode trunk
DeviceD(config-if-range)# exit
```

(7) Set user ports on access devices to work in access mode and add them to VLANs..

Configure Device C.

```
DeviceC(config)# interface gigabitethernet 0/3
DeviceC(config-if-GigabitEthernet 0/3)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/3)# switchport access vlan 10
DeviceC(config-if-GigabitEthernet 0/3)# exit
DeviceC(config)# interface gigabitethernet 0/4
DeviceC(config-if-GigabitEthernet 0/4)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/4)# switchport access vlan 20
DeviceC(config-if-GigabitEthernet 0/4)# exit
DeviceC(config)# interface gigabitethernet 0/5
DeviceC(config-if-GigabitEthernet 0/5)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/5)# switchport access vlan 30
DeviceC(config-if-GigabitEthernet 0/5)# exit
DeviceC(config)# interface gigabitethernet 0/6
DeviceC(config-if-GigabitEthernet 0/6)# switchport mode access
DeviceC(config-if-GigabitEthernet 0/6)# switchport access vlan 40
DeviceC(config-if-GigabitEthernet 0/6)# end
DeviceC# write
```

Configure Device D.

```
DeviceD(config)# interface gigabitethernet 0/3
DeviceD(config-if-GigabitEthernet 0/3)# switchport mode access
DeviceD(config-if-GigabitEthernet 0/3)# switchport access vlan 10
DeviceD(config-if-GigabitEthernet 0/3)# exit
DeviceD(config)# interface gigabitethernet 0/4
DeviceD(config-if-GigabitEthernet 0/4)# switchport mode access
DeviceD(config-if-GigabitEthernet 0/4)# switchport access vlan 20
DeviceD(config-if-GigabitEthernet 0/4)# exit
DeviceD(config)# interface gigabitethernet 0/5
DeviceD(config-if-GigabitEthernet 0/5)# switchport mode access
DeviceD(config-if-GigabitEthernet 0/5)# switchport access vlan 30
DeviceD(config-if-GigabitEthernet 0/5)# exit
DeviceD(config)# interface gigabitethernet 0/6
DeviceD(config-if-GigabitEthernet 0/6)# switchport mode access
DeviceD(config-if-GigabitEthernet 0/6)# switchport access vlan 40
DeviceD(config-if-GigabitEthernet 0/6)# end
DeviceD# write
```

5. Verification

On Device A, run the **show spanning-tree summary** command to display the spanning tree topology.

```
DeviceA# show spanning-tree summary
Spanning tree enabled protocol mstp

MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     00d0.f822.3344
             this bridge is root
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec

  Bridge ID  Priority    4096
             Address     00d0.f822.3344
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Ag1              Desg FWD 19000      128      False    P2p
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 200000     128      False    P2p

MST 1 vlans map : 10,30
  Region Root Priority   4096
             Address     00d0.f822.3344
             this bridge is region root

  Bridge ID  Priority    4096
             Address     00d0.f822.3344

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Ag1              Desg FWD 19000      128      False    P2p
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 200000     128      False    P2p

MST 2 vlans map : 20,40
  Region Root Priority   4096
             Address     001a.a917.78cc
             this bridge is region root

  Bridge ID  Priority    8192
```

```
             Address     00d0.f822.3344

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Ag1              Root FWD 19000      128      False    P2p
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 200000     128      False    P2p
```

On Device B, run the **show spanning-tree summary** command to display the spanning tree topology.

```
DeviceB# show spanning-tree summary

Spanning tree enabled protocol mstp

MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     00d0.f822.3344
             this bridge is root
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec

  Bridge ID  Priority    8192
             Address     001a.a917.78cc
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Ag1              Root FWD 19000      128      False    P2p
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 200000     128      False    P2p

MST 1 vlans map : 10,30
  Region Root Priority   4096
             Address     00d0.f822.3344
             this bridge is region root

  Bridge ID  Priority    8192
             Address     001a.a917.78cc

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Ag1              Root FWD 19000      128      False    P2p
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 200000     128      False    P2p

MST 2 vlans map : 20,40
  Region Root Priority   4096
             Address     001a.a917.78cc
             this bridge is region root
  Bridge ID  Priority    4096
             Address     001a.a917.78cc

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Ag1              Desg FWD 19000      128      False    P2p
Gi0/1            Desg FWD 200000     128      False    P2p
Gi0/2            Desg FWD 200000     128      False    P2p
```

On Devices C and D, run the **show spanning-tree summary** command to check whether the spanning tree topology is correctly calculated. The following uses Device C as an example.

```
DeviceC# show spanning-tree summary

Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     00d0.f822.3344
             this bridge is root
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec

  Bridge ID  Priority    32768
             Address     001a.a979.00ea
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     Type  OperEdge
---------------- ---- --- ---------- -------- ----- ---------------
Gi0/2            Altn BLK 200000     128      P2p   False
Gi0/1            Root FWD 200000     128      P2p   False

MST 1 vlans map : 10,30
  Region Root Priority   4096
             Address     00d0.f822.3344
             this bridge is region root

  Bridge ID  Priority    32768
             Address     001a.a979.00ea

Interface        Role Sts Cost       Prio     Type  OperEdge
---------------- ---- --- ---------- -------- ----- ---------------
Gi0/2            Altn BLK 200000     128      P2p   False
Gi0/1            Root FWD 200000     128      P2p   False

MST 2 vlans map : 20,40
  Region Root Priority   4096
             Address     001a.a917.78cc
             this bridge is region root
```

```
 Bridge ID  Priority    32768
            Address     001a.a979.00ea

Interface        Role Sts Cost       Prio     Type  OperEdge
---------------- ---- --- ---------- -------- ----- ----------------
Gi0/2            Root FWD 200000     128      P2p   False
Gi0/1            Altn BLK 200000     128      P2p   False
```

On Devices A and B, run the **show vrrp brief** command to check whether VRRP master/slave relationship is successfully established.

```
DeviceA# show vrrp brief
Interface   Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
VLAN 10     10   120  3      -    P    Master  192.168.10.3  192.168.10.1
VLAN 20     20   100  3      -    P    Backup  192.168.20.2  192.168.20.1
VLAN 30     30   120  3      -    P    Master  192.168.30.3  192.168.30.1
VLAN 40     40   100  3      -    P    Backup  192.168.40.2  192.168.40.1

DeviceB# show vrrp brief
Interface   Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
VLAN 10     10   100  3      -    P    Backup  192.168.10.2  192.168.10.1
VLAN 20     20   120  3      -    P    Master  192.168.20.3  192.168.20.1
VLAN 30     30   100  3      -    P    Backup  192.168.30.2  192.168.30.1
VLAN 40     40   120  3      -    P    Master  192.168.40.3  192.168.40.1
```

Disconnect uplink GigabitEthernet 0/5 of Device A and check the VRRP status change on Device A and Device B.

```
DeviceA#show vrrp brief
Interface   Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
VLAN 10     10   90    3       -   P    Backup  192.168.10.3  192.168.10.1
VLAN 20     20   100   3       -   P    Backup  192.168.20.2  192.168.20.1
VLAN 30     30   90   3      -   P    Backup  192.168.30.3  192.168.30.1
VLAN 40     40   100  3      -   P    Backup  192.168.40.2  192.168.40.1

DeviceB# show vrrp brief
Interface   Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
VLAN 10     10   100  3      -    P    Master  192.168.10.2  192.168.10.1
VLAN 20     20   120  3      -    P    Master  192.168.20.3  192.168.20.1
VLAN 30     30   100  3      -    P    Master  192.168.30.2  192.168.30.1
VLAN 40     40   120  3      -    P    Master  192.168.40.3  192.168.40.1
```

Disconnect uplink GigabitEthernet 0/5 of Device B and check the VRRP status change on Device A and Device B.

```
DeviceA# show vrrp brief
Interface   Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
VLAN 10     10   120  3      -    P    Master  192.168.10.3  192.168.10.1
VLAN 20     20   100  3      -    P    Master  192.168.20.2  192.168.20.1
VLAN 30     30   120  3      -    P    Master  192.168.30.3  192.168.30.1
VLAN 40    40   100  3      -    P    Master  192.168.40.2  192.168.40.1

DeviceB# show vrrp brief
Interface  Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
VLAN 10    10   100  3      -    P    Backup  192.168.10.2  192.168.10.1
VLAN 20    20   90   3      -    P    Backup  192.168.20.3  192.168.20.1
VLAN 30    30   100  3      -    P    Backup  192.168.30.2  192.168.30.1
VLAN 40    40   90   3      -    P    Backup  192.168.40.3  192.168.40.1
```

6. Configuration Files

- Device A configuration file

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree mst 0 priority 4096
spanning-tree mst 1 priority 4096
spanning-tree mst 2 priority 8192
spanning-tree
!
sysmac 00d0.f822.3344
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
!
interface GigabitEthernet 0/3
 port-group 1
!
interface GigabitEthernet 0/4
 port-group 1
!
interface AggregatePort 1
 switchport mode trunk
!
interface gigabitethernet 0/5
 no switchport
 ip address 10.10.1.1 255.255.255.0
!
interface vlan 10
 ip address 192.168.10.3 255.255.255.0
 vrrp 10 ip 192.168.10.1
 vrrp 10 priority 120
 vrrp 10 track gigabitethernet 0/5 30
!
interface vlan 20
 ip address 192.168.20.2 255.255.255.0
 vrrp 20 ip 192.168.20.1
!
interface vlan 30
 ip address 192.168.30.3 255.255.255.0
 vrrp 30 ip 192.168.30.1
 vrrp 30 priority 120
 vrrp 30 track gigabitethernet 0/5 30
!
interface vlan 40
 ip address 192.168.40.2 255.255.255.0
 vrrp 40 ip 192.168.40.1
```

- Device B configuration file

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree mst 0 priority 8192
spanning-tree mst 1 priority 8192
spanning-tree mst 2 priority 4096
spanning-tree
!
sysmac 001a.a917.78cc
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
!
interface GigabitEthernet 0/3
 port-group 1
!
interface GigabitEthernet 0/4
 port-group 1
!
interface AggregatePort 1
 switchport mode trunk
```

```
!
interface gigabitethernet 0/5
 no switchport
 ip address 10.10.2.1 255.255.255.0
!
interface vlan 10
 ip address 192.168.10.2 255.255.255.0
 vrrp 10 ip 192.168.10.1
!
interface vlan 20
 ip address 192.168.20.3 255.255.255.0
 vrrp 20 ip 192.168.20.1
 vrrp 20 priority 120
 vrrp 20 track gigabitethernet 0/5 30
!
interface vlan 30
 ip address 192.168.30.2 255.255.255.0
 vrrp 30 ip 192.168.30.1
!
interface vlan 40
 ip address 192.168.40.3 255.255.255.0
 vrrp 40 ip 192.168.40.1
 vrrp 40 priority 120
 vrrp 40 track gigabitethernet 0/5 30
```

- Configuration files of Devices C and D

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
!
interface GigabitEthernet 0/3
 switchport access vlan 10
 spanning-tree bpduguard enable
 spanning-tree portfast
!
interface GigabitEthernet 0/4
```

```
 switchport access vlan 20
 spanning-tree bpduguard enable
 spanning-tree portfast
!
interface GigabitEthernet 0/5
 switchport access vlan 30
 spanning-tree bpduguard enable
 spanning-tree portfast
!
interface GigabitEthernet 0/6
 switchport access vlan 40
 spanning-tree bpduguard enable
 spanning-tree portfast
```

7. Common Errors

- In the MSTP+VRRP topology, the MST region configurations on different devices are different.
- VLANs are not created before the mappings between instances and VLANs are configured.
- In the MSTP+VRRP topology, if some devices run STP or RSTP, the devices will be considered as different MST regions for spanning tree calculation. In this case, the calculated spanning tree will be different from expectations.

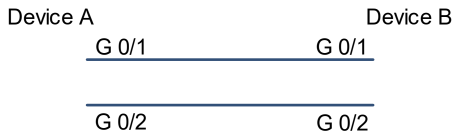
1.21.3 Configuring Spanning Tree Compatibility for MSTP Interfaces

1. Requirements

In the topology as shown in Figure 1-1, enable MSTP on Device A and Device B and configure the same instance mappings on the devices: Associate instance 1 with VLAN 10 and add port GigabitEthernet 0/1 to VLAN 10; associate instance 2 with VLAN 20 and add port GigabitEthernet 0/2 to VLAN 20. Configure spanning tree compatibility for interfaces so that BPDUs sent by interfaces carry only information about instances corresponding to the VLANs, to which the interfaces belong.

2. Topology

Figure 1-1Basic Configuration



3. Notes

- Adopt the same configurations on Device A and Device B. The following uses Device A as an example.
- Create VLAN 10, VLAN 20, instance 1, and instance 2. Associate instance 1 with VLAN 10 and instance 2 with VLAN 20. Add port GigabitEthernet 0/1 to VLAN 10 and port GigabitEthernet 0/2 to VLAN 20, and configure spanning tree compatibility for the ports. Enable the STP function.

4. Procedure

(1) Create VLAN 10, VLAN 20, instance 1, and instance 2. Associate instance 1 with VLAN 10 and instance 2 with VLAN 20.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA(config)# vlan range 10,20
DeviceA(config-vlan-range)# exit
DeviceA(config)# spanning-tree mst configuration
DeviceA(config-mst)# instance 1 vlan 10
DeviceA(config-mst)# instance 2 vlan 20
DeviceA(config-mst)# exit
```

(2) Add port GigabitEthernet 0/1 to VLAN 10 and port GigabitEthernet 0/2 to VLAN 20, and configure spanning tree compatibility for the ports.

```
DeviceA(config)# interface gigabitethernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)# switchport access vlan 10
DeviceA(config-if-GigabitEthernet 0/1)# spanning-tree compatible enable
DeviceA(config-if-GigabitEthernet 0/1)# exit
DeviceA(config)# interface gigabitethernet 0/2
DeviceA(config-if-GigabitEthernet 0/2)# switchport access vlan 20
DeviceA(config-if-GigabitEthernet 0/2)# spanning-tree compatible enable
DeviceA(config-if-GigabitEthernet 0/2)# exit
```

(3) Enable the STP function.

```
DeviceA(config)# spanning-tree
DeviceA(config)# end
DeviceA# write
```

5. Verification

(1) If no spanning tree compatibility is configured for interfaces, check the spanning tree configuration as follows.

Device A becomes the root of Device B because its bridge ID is smaller. Ports GigabitEthernet 0/1 and GigabitEthernet 0/2 of Device A are designated ports in their respective instances.

```
DeviceA# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-4094
  Root ID    Priority    32768
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
```

```
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p

MST 1 vlans map : 10
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p

MST 2 vlans map : 20
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000     128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p
```

Check Device B. In the spanning tree of instance 1, port GigabitEthernet 0/1 belonging to VLAN 10 is the root port and port GigabitEthernet 0/2 belonging to VLAN 20 is the alternate port. Port GigabitEthernet 0/2 rejects traffic of VLAN 10 to pass. Therefore, the alternate port cannot serve as an alternative to the root port. In the spanning tree of instance 2, port GigabitEthernet 0/2 belonging to VLAN 20 is the root port and port GigabitEthernet 0/1 belonging to VLAN 10 is the alternate port. Port GigabitEthernet 0/1 rejects traffic of VLAN 20 to pass. Therefore, the alternate port cannot serve as an alternative to the root port.

```
DeviceB# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-4094
  Root ID    Priority    32768
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    32768
             Address     0074.9cee.f49e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec
```

```

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/1            Root FWD 20000      128      False    P2p
Gi0/2            Altn BLK 20000      128      False    P2p

MST 1 vlans map : 10
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/1            Root FWD 20000      128      False    P2p
Gi0/2            Altn BLK 20000      128      False    P2p

MST 2 vlans map : 20
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/1            Altn BLK 20000      128      False    P2p
Gi0/2            Root FWD 20000      128      False    P2p
```

(2) After spanning tree compatibility is configured for interfaces, check the spanning tree topology on Device A. Device A becomes the root of Device B because its bridge ID is smaller.

The spanning tree is not pruned in instance 0. Ports GigabitEthernet 0/1 and GigabitEthernet 0/2 are designated ports.

```
DeviceA# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-4094
  Root ID    Priority    32768
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p
```

The spanning tree is pruned in instance 1. Only port GigabitEthernet 0/1 belonging to VLAN 10 is the designated port.

```
MST 1 vlans map : 10
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000      128      False    P2p
```

The spanning tree is pruned in instance 2. Only port GigabitEthernet 0/2 belonging to VLAN 20 is the designated port.

```
MST 2 vlans map : 20
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.53ca

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/2            Desg FWD 20000      128      False    P2p
```

(3) Check the spanning tree topology on Device B. Device A becomes the root of Device B because its bridge ID is smaller.

The spanning tree is not pruned in instance 0. Port GigabitEthernet 0/1 is the root port and port GigabitEthernet 0/2 is the alternate port.

```
DeviceB# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-4094
  Root ID    Priority    32768
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    32768
             Address     0074.9cee.f49e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 20000      128      False    P2p
Gi0/2            Altn BLK 20000      128      False    P2p
```

The spanning tree is pruned in instance 1. Only port GigabitEthernet 0/1 belonging to VLAN 10 is the root port and there is no alternate port.

```
MST 1 vlans map : 10
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 20000      128      False    P2p
```

The spanning tree is pruned in instance 2. Only port GigabitEthernet 0/2 belonging to VLAN 20 is the root port and there is no alternate port.

```
MST 2 vlans map : 20
  Region Root Priority   32768
             Address     0074.9cee.53ca
             this bridge is region root

  Bridge ID  Priority    32768
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/2            Root FWD 20000      128      False    P2p
```

6. Configuration Files

Configuration files of Devices A and B

```
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-4094
 instance 1 vlan 10
 instance 2 vlan 20
!
spanning-tree
!
vlan range 1,10,20
```

```
!
interface GigabitEthernet 0/1
 switchport access vlan 10
 spanning-tree compatible enable
!
interface GigabitEthernet 0/2
 switchport access vlan 20
 spanning-tree compatible enable
```

7. Common Errors

- If the VLAN lists configured on interfaces at both ends of a link are inconsistent, communication may fail after VLANs are pruned by the spanning tree compatibility function of the interfaces.
- If different instances are not configured or interfaces permit the traffic of all VLANs to pass, the spanning tree compatibility function cannot perform spanning tree pruning based on instances. In this case, there is no necessary to configure the spanning tree compatibility function.

1.21.4 Configuring BPDU Tunnel

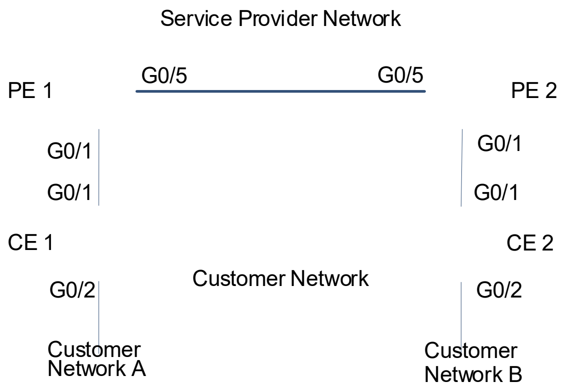
1. Requirements

In the typical QinQ network topology shown in Figure 1-1, the upper part is the service provider network while the lower part is the customer network. The service provider network consists of two PEs: PE 1 and PE 2. Customer Network A and Customer Network B are two sites of the same customer in different regions. CE 1 and CE 2 are access devices for connecting the customer networks to the service provider network, and they connect to the service provider network respectively through PE 1 and PE 2.

Configure an L2 protocol tunnel to transparently transmit L2 protocol packets from the customer network at one end to the customer network at the other end so that the spanning trees of the customer network and service provider network are separately calculated without interference. According to the needs of the customer, configure the STP tunnel function to transmit BPDUs from the customer network through a tunnel in the service provider network so that the spanning tree is calculated in a unified manner for the cross-region customer network, independent of the service provider network.

2. Topology

Figure 1-1BPDU Tunnel Application Topology



3. Notes

- Configure MSTP on PE 1 and PE 2. Configure the bridge priority to make PE 1 become the root of instances 0 and 1, and PE 2 the root of instance 2. Enable STP.
- Configure the QinQ encapsulation mode (for example, basic QinQ) on PE 1 and PE 2. Configure VLAN 10 as the native VLAN of tunnel ports and add VLAN 10 to the permitted untagged VLAN list of the tunnel ports. Configure VLAN 10 as the public network transmission channel. VLAN 10 is encapsulated into the user data packets before they are forwarded in the public network. Configure port GigabitEthernet 0/1 as a tunnel port and enable the STP tunnel function. Enable the global STP tunnel function.
- Configure the uplink ports of PE 1 and PE 2 as the uplink ports.
- Configure STP on CE 1 and CE 2. Set the bridge priority to 0, configure timer parameters, and enable STP on CE 1.
- Configure the uplink and downlink ports of CE 1 and CE 2 as trunk ports.

4. Procedure

(1) Configure MSTP on PE 1 and PE 2.

Set the bridge priority to 4094 for instances 0 and 1 and 8192 for instance 2 on PE 1.

```
PE1> enable
PE1# configure terminal
PE1(config)# vlan range 10,20,30,40
PE1(config-vlan-range)# exit
PE1(config)# spanning-tree mst configuration
PE1(config-mst)# instance 1 vlan 10,30
PE1(config-mst)# instance 2 vlan 20,40
PE1(config-mst)# spanning-tree mst 0 priority 4096
PE1(config)# spanning-tree mst configuration
PE1(config-mst)# spanning-tree mst 1 priority 4096
PE1(config)# spanning-tree mst configuration
PE1(config-mst)# spanning-tree mst 2 priority 8192
PE1(config)# spanning-tree
```

Set the bridge priority to 4094 for instance 2 and 8192 for instances 0 and 1 on PE 2.

```
PE2> enable
PE2# configure terminal
PE2(config)# vlan range 10,20,30,40
PE2(config-vlan-range)# exit
PE2(config)# spanning-tree mst configuration
PE2(config-mst)# instance 1 vlan 10,30
PE2(config-mst)# instance 2 vlan 20,40
PE2(config-mst)# spanning-tree mst 0 priority 8192
PE2(config)# spanning-tree mst configuration
PE2(config-mst)# spanning-tree mst 1 priority 8192
PE2(config)# spanning-tree mst configuration
PE2(config-mst)# spanning-tree mst 2 priority 4096
PE2(config)# spanning-tree
```

(2) Configure the basic QinQ transmission channel VLAN 10 on PE 1 and PE 2. Enable the interface STP tunnel function and global STP tunnel function. The configurations on PE 1 are the same as those on PE 2. The following uses PE 1 as an example.

```
PE1(config)# interface gigabitethernet 0/1
PE1(config-if-GigabitEthernet 0/1)# switchport
PE1(config-if-GigabitEthernet 0/1)# switchport mode dot1q-tunnel
PE1(config-if-GigabitEthernet 0/1)# switchport dot1q-tunnel native vlan 10
PE1(config-if-GigabitEthernet 0/1)# switchport dot1q-tunnel allowed vlan add
untagged 10
PE1(config-if-GigabitEthernet 0/1)# l2protocol-tunnel stp enable
PE1(config-if-GigabitEthernet 0/1)# exit
PE1(config)# l2protocol-tunnel stp
```

(3) Configure the uplink ports of PE 1 and PE 2 as the uplink ports. The configurations on PE 1 are the same as those on PE 2. The following uses PE 1 as an example.

```
PE1(config)# interface gigabitethernet 0/5
PE1(config-if-GigabitEthernet 0/5)# switchport
PE1(config-if-GigabitEthernet 0/5)# switchport mode uplink
PE1(config-if-GigabitEthernet 0/5)# end
PE1# write
```

(4) Configure STP on CE 1 and CE 2.

Set the spanning tree mode to STP and bridge priority to 0, configure timer parameters, and enable STP on CE 1.

```
CE1> enable
CE1# configure terminal
CE1(config)# spanning-tree mode stp
CE1(config)# spanning-tree priority 0
CE1(config)# spanning-tree hello-time 4
CE1(config)# spanning-tree max-age 25
CE1(config)# spanning-tree forward-time 18
CE1(config)# spanning-tree
```

Set the spanning tree mode to STP and enable STP on CE 2.

```
CE2> enable
CE2# configure terminal
CE2(config)# spanning-tree mode stp
CE2(config)# spanning-tree
```

(5) Configure the uplink and downlink ports of CE 1 and CE 2 as trunk ports. The configurations on CE 1 are the same as those on CE 2. The following uses CE 1 as an example.

```
CE1(config)# interface range gigabitethernet 0/1-2
CE1(config-if-range)# switchport
CE1(config-if-range)# switchport mode trunk
CE1(config-if-range)# end
CE1# write
```

5. Verification

(1) Check tunnel port configurations on PE 1 and PE 2. The configurations on PE 1 are the same as those on PE 2. The following uses PE 1 as an example.

```
PE1# show interfaces dot1q-tunnel
========Interface Gi0/1========
Native vlan: 10
Allowed vlan list:1,10,
Tagged vlan list:
```

(2) Verify that the STP tunnel function is enabled in both global configuration mode and interface configuration mode on PE 1 and PE 2. The configurations on PE 1 are the same as those on PE 2. The following uses PE 1 as an example.

```
PE1# show l2protocol-tunnel stp
L2protocol-tunnel: Stp Enable
L2protocol-tunnel destination mac address: 01d0.f800.0005
GigabitEthernet 0/1 l2protocol-tunnel stp enable
```

(3) Check that PE 1 and PE 2 have independent spanning tree topologies.

Check the spanning tree topology on PE 1. For instances 0 and 1, PE 1 is the root of PE 2 and port GigabitEthernet 0/5 is the designated port. For instance 2, PE 2 is the root of PE 1 and port GigabitEthernet 0/5 is the root port. Port GigabitEthernet 0/1 that connects to CE 1 of the customer network is not included in the spanning tree topologies of PE 1 and PE 2. The spanning tree time parameters (**Hello Time**: **2 sec**; **Forward Delay**: **15 sec**; **Max Age**: **20 sec**) of PE 1 use default values and are not affected by the configurations of CE 1.

```
PE1# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     0074.9cee.f49e
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    4096
             Address     0074.9cee.f49e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/5            Desg FWD 20000      128      False    P2p

MST 1 vlans map : 10, 30
  Region Root Priority   4096
             Address     0074.9cee.f49e
             this bridge is region root

  Bridge ID  Priority    4096
```

```
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/5            Desg FWD 20000      128      False    P2p


MST 2 vlans map : 20, 40
  Region Root Priority   4096
             Address     00d0.f8ee.8c1e
             this bridge is region root

  Bridge ID  Priority    8192
             Address     0074.9cee.f49e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/5            Root FWD 20000          128      False    P2p
```

Check the spanning tree topology on PE 2. For instances 0 and 1, PE 1 is the root of PE 2 and port GigabitEthernet 0/5 is the root port. For instance 2, PE 2 is the root of PE 1 and port GigabitEthernet 0/5 is the designated port. Port GigabitEthernet 0/1 that connects to CE 2 of the customer network is not included in the spanning tree topologies of PE 1 and PE 2. The spanning tree time parameters (**Hello Time**: **2 sec**; **Forward Delay**: **15 sec**; **Max Age**: **20 sec**) of PE 2 use default values and are not affected by the configurations of CE 1.

```
PE2# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : 1-9, 11-19, 21-29, 31-39, 41-4094
  Root ID    Priority    4096
             Address     0074.9cee.f49e
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    8192
             Address     00d0.f8ee.8c1e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/5            Root FWD 20000          128      False    P2p


MST 1 vlans map : 10, 30
  Region Root Priority   4096
             Address     0074.9cee.f49e
             this bridge is region root

  Bridge ID  Priority    8192
```

```
             Address     00d0.f8ee.8c1e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/5            Root FWD 20000          128      False    P2p


MST 2 vlans map : 20, 40
  Region Root Priority   4096
             Address     00d0.f8ee.8c1e
             this bridge is region root

  Bridge ID  Priority    4096
             Address     00d0.f8ee.8c1e

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/5            Desg FWD 20000      128      False    P2p
```

(4) Check that CE 1 and CE 2 have independent spanning tree topologies.

Check the spanning tree topology on CE 1. CE 1 is the root of CE 2 and port GigabitEthernet 0/1 of CE 1 is the designated port. The spanning tree time parameters (**Hello Time**: **4 sec**; **Forward Delay**: **18 sec**; **Max Age**: **25 sec**) of CE 1 use configured values.

```
CE1# show spanning-tree summary
Spanning tree enabled protocol stp
  Root ID    Priority    0
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec

  Bridge ID  Priority    0
             Address     0074.9cee.53ca
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- -----------------
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      128      False    P2p
```

Check the spanning tree topology on CE 2. CE 1 is the root of CE 2 and port GigabitEthernet 0/1 of CE 2 is the root port. The spanning tree time parameters (**Hello Time**: **4 sec**; **Forward Delay**: **18 sec**; **Max Age**: **25 sec**) of CE 2 use configured values of the root CE 1.

```
CE2# show spanning-tree summary
Spanning tree enabled protocol stp
  Root ID    Priority    0
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   4 sec  Forward Delay 18 sec  Max Age 25 sec
  Bridge ID  Priority    32768
             Address     0074.9c11.c9e6
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 20000      128      False    P2p Bound(STP)
Gi0/2            Desg FWD 20000      128      False    P2p
```

6. Configuration Files

- PE 1 configuration file

```
l2protocol-tunnel stp
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree mst 0 priority 4096
spanning-tree mst 1 priority 4096
spanning-tree mst 2 priority 8192
spanning-tree
!
sysmac 0074.9cee.f49e
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode dot1q-tunnel
 switchport dot1q-tunnel allowed vlan add untagged 10
 switchport dot1q-tunnel native vlan 10
 l2protocol-tunnel stp enable
!
interface GigabitEthernet 0/5
 switchport mode uplink
```

- PE 2 configuration file

```
l2protocol-tunnel stp
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 30
 instance 2 vlan 20, 40
!
spanning-tree mst 0 priority 8192
spanning-tree mst 1 priority 8192
spanning-tree mst 2 priority 4096
```

```
spanning-tree
!
sysmac 00d0.f8ee.8c1e
!
vlan range 1,10,20,30,40
!
interface GigabitEthernet 0/1
 switchport mode dot1q-tunnel
 switchport dot1q-tunnel allowed vlan add untagged 10
 switchport dot1q-tunnel native vlan 10
 l2protocol-tunnel stp enable
!
interface GigabitEthernet 0/5
 switchport mode uplink
```

- CE 1 configuration file

```
spanning-tree max-age 25
spanning-tree forward-time 18
spanning-tree hello-time 4
spanning-tree mode stp
spanning-tree mst 0 priority 0
spanning-tree
!
sysmac 0074.9cee.53ca
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
```

- CE 2 configuration file

```
spanning-tree mode stp
spanning-tree
!
sysmac 0074.9c11.c9e6
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
```

7. Common Errors

- The L2 protocol tunnel function is not enabled in both global configuration mode and interface configuration mode. As a result, the BPDU tunnel function does not take effect.
- The BPDU tunnel addresses configured in the service provider network are inconsistent. As a result, BPDU

frames from the customer network cannot be correctly transmitted.

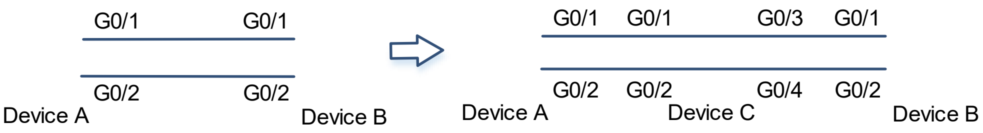
1.21.5 Configuring BPDU Transparent Transmission

1. Requirements

In the topology as shown in Figure 1-1, Device A is connected to Device B and MSTP is enabled on them. Device A is the root of Device B. One Device C with STP disabled is added between Device A and Device B. Configure the BPDU transparent transmission function on Device C to transmit STP packets of Device A and Device B so that Device A and Device B can still conduct calculation together and generate spanning trees.

2. Topology

Figure 1-1Configuring BPDU Transparent Transmission



3. Notes

- On Device A, configure STP, set the bridge priority to 0 (smaller than the default value **32768**), and set the priority of port GigabitEthernet 0/2 to 16 (smaller than the default value **128**). In this way, Device A becomes the root bridge of Device B and port GigabitEthernet 0/2 of Device B becomes the root port.
- Enable STP on Device B.
- Configure BPDU transparent transmission on Device C.
- Configure interconnected ports as trunk ports.

4. Procedure

(1) On Device A, configure the interconnected ports as trunk ports. Configure the device priority and port priority and enable the STP function.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA(config)# spanning-tree mst 0 priority 0
DeviceA(config)# interface gigabitethernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)# switchport
DeviceA(config-if-GigabitEthernet 0/1)# switchport mode trunk
DeviceA(config-if-GigabitEthernet 0/1)# exit
DeviceA(config)# interface gigabitethernet 0/2
DeviceA(config-if-GigabitEthernet 0/2)# switchport
DeviceA(config-if-GigabitEthernet 0/2)# switchport mode trunk
DeviceA(config-if-GigabitEthernet 0/2)# spanning-tree mst 0 port-priority 16
DeviceA(config)# spanning-tree
```

(2) On Device B, enable the STP function and configure the interconnected ports as trunk ports.

```
DeviceB> enable
DeviceB# configure terminal
```

```
DeviceB(config)# spanning-tree
DeviceB(config)# interface gigabitethernet 0/1
DeviceB(config-if-GigabitEthernet 0/1)# switchport
DeviceB(config-if-GigabitEthernet 0/1)# switchport mode trunk
DeviceB(config-if-GigabitEthernet 0/1)# exit
DeviceB(config)# interface gigabitethernet 0/2
DeviceB(config-if-GigabitEthernet 0/2)# switchport
DeviceB(config-if-GigabitEthernet 0/2)# switchport mode trunk
```

(3) On Device C, configure BPDU transparent transmission and configure the interconnected ports as trunk ports.

```
DeviceC> enable
DeviceC# configure terminal
DeviceC(config)# no spanning-tree
DeviceC(config)# bridge-frame forwarding protocol bpdu
DeviceC(config)# interface range gigabitethernet 0/1-4
DeviceC(config-if-rang)# switchport
DeviceC(config-if-rang)# switchport mode trunk
```

5. Verification

(1) When Device C is not connected, check the spanning tree topology.

Check the spanning tree topology of Device A. Device A is the root.

```
DeviceA# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : ALL
  Root ID    Priority    0
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    0
             Address     0074.9cee.53ca
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      16       False    P2p
```

Check the spanning tree topology of Device B. Device A is the root of Device B. The port priority of Device A is changed. Therefore, port GigabitEthernet 0/2 of Device B becomes the root port and port GigabitEthernet 0/1 becomes an alternate port.

```
DeviceB# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : ALL
  Root ID    Priority    0
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    8192
             Address     00d0.f8ee.8c1e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Altn BLK 20000      128      False    P2p
Gi0/2            Root FWD 20000      128      False    P2p
```

(2) After Device C is connected to the topology, check the spanning tree topology.

Check the spanning tree topology on Device A. The spanning tree topology of Device A keeps unchanged.

```
DeviceA# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : ALL
  Root ID    Priority    0
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    0
             Address     0074.9cee.53ca
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Desg FWD 20000      128      False    P2p
Gi0/2            Desg FWD 20000      16       False    P2p
```

Check the spanning tree topology on Device B. Device A is still the root of Device B, but port GigabitEthernet 0/1 of Device B becomes the root port, and port GigabitEthernet 0/2 becomes an alternate port. After Device C is configured to transparently transmit STP packets, ports GigabitEthernet 0/1 and 0/2 of Device B can receive BPDUs from port GigabitEthernet 0/2 of Device A. Therefore, the root port cannot be elected based on the port priority and port GigabitEthernet 0/1 of Device B is elected as the root port because of its smaller port ID.

```
DeviceB# show spanning-tree summary
Spanning tree enabled protocol mstp
MST 0 vlans map : ALL
  Root ID    Priority    0
             Address     0074.9cee.53ca
             this bridge is root
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec

  Bridge ID  Priority    8192
             Address     00d0.f8ee.8c1e
             Hello Time   2 sec  Forward Delay 15 sec  Max Age 20 sec
```

```
Interface        Role Sts Cost       Prio     OperEdge Type
---------------- ---- --- ---------- -------- -------- ----------------
Gi0/1            Root FWD 20000      128      False    P2p
Gi0/2            Altn BLK 20000      128      False    P2p
```

6. Configuration Files

- Device A configuration file

```
spanning-tree mst 0 priority 0
spanning-tree
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
 spanning-tree mst 0 port-priority 16
!
```

- Device B configuration file

```
spanning-tree

interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
```

- Device C configuration file

```
bridge-frame forwarding protocol bpdu
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
!
interface GigabitEthernet 0/3
 switchport mode trunk
!
interface GigabitEthernet 0/4
 switchport mode trunk
```

-