Руководство пользователя

QSR-2830

Оглавление

1 CONFIGURING RLDP

1.1 Overview

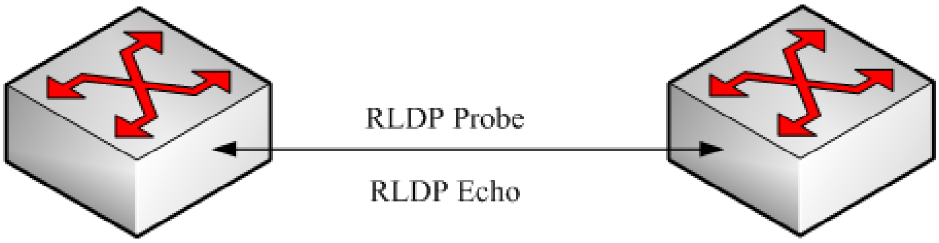
The Rapid Link Detection Protocol (RLDP) is one of Qtech's proprietary link protocol designed to detect Ethernet link fault quickly.

General Ethernet link detection mechanism only makes use of the status of the physical connections and detects the connectivity of the link via the auto-negotiation of the physical layer. This detection mechanism has restrictions and sometimes cannot provide reliable link detection information for the user. For example, if the optical fiber receiving line pair on the optical interface is misconnected, due to the existence of the optical converter, the related port of the device is "linkup" physically but actually the corresponding layer-2 link cannot work for communications. Here is another example. There is an intermediate network between two Ethernet devices. Due to the existence of the network transmission relay devices, the same problem may occur if those relay devices are faulty.

The RLDP enables easy detection of Ethernet device link fault, including the one-way link fault, two-way link fault and loop link fault.

The RLDP implements the detection by exchanging the RLDP messages at the two ends of the link, as shown in Figure-1:

Figure-1:



The RLDP defines two protocol messages: Probe message and Echo message. The RLDP sends the Probe message of this port to the port with RLDP configured and in linkup status on regular basis, and waits for the Echo message from the neighbor port and waits for the Probe message sent by the neighbor ports. If a link is correct both physically and logically, a port shall be able to receive the Echo message of the neighbor port as well as the Probe message of the neighbor port. Otherwise, the link is considered abnormal.

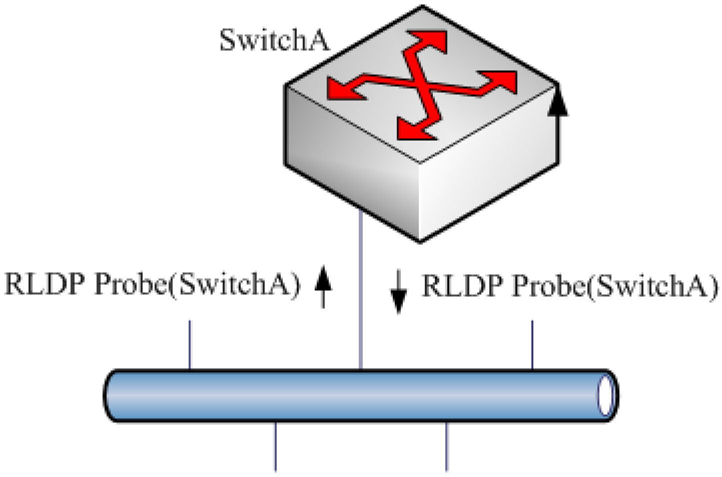
Note

To make use of the one-way detection and two-way detection functions of the RLDP, it is necessary to ensure the RLDP is enabled on the ports at both ends of the link. And, it is not allowed for a port with RLDP enabled to connect multiple neighbor ports. Otherwise, the RLDP cannot detect the health conditions of every neighbor link.

Typical Application

Loop detection:

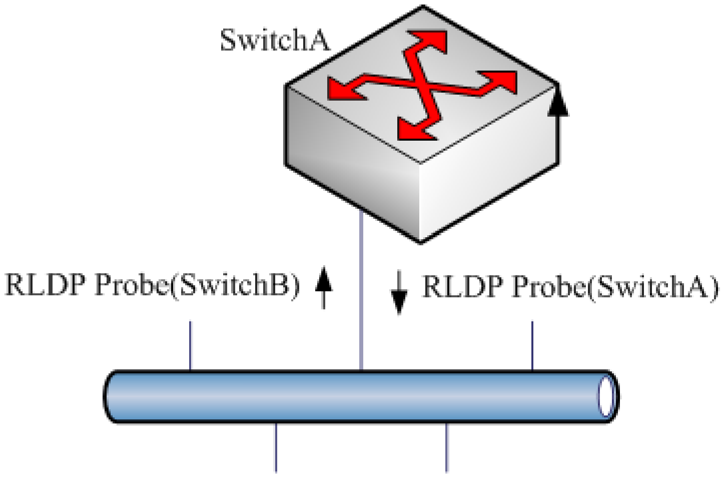
Figure-2: Loop detection



The so-called loop fault means that a loop appears on the links connected with the port. A shown above, on a port the RLDP receives the RLDP message sent from its machine, so the port is considered as loop fault. So, the RLDP deals with the fault according to the user configurations, including alarming, setting port violation, turning off the SVI with that port, turning off the port learning forwarding, and more.

One-way link detection:

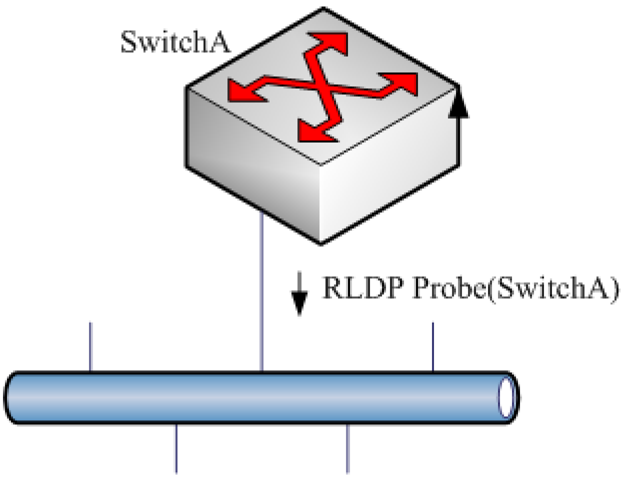
Figure-3: One-way link detection



The so-called one-way link detection means the link connected with the port can receive message only or send messages only (due to misconnection of the optical receiving line pair, for example). As shown above, the RLDP only receives the detection message from the neighbor port on a port, so it is considered one-way link fault. So, the RLDP deals with the fault accordingly according to the user configurations. In addition, if the port cannot receive any RLDP detection message, it is also considered one-way link fault.

Two-way link detection:

Figure-4:Two-way link detection



This means that fault occurs at the frame transmission/receiving at both ends of the link. As shown above, the port of the device sends the RLDP probe message but has never received the Echo message or the Probe message from the neighbors. So, it is considered two-way link fault. From the nature of the fault, the two-way fault actually includes the one-way fault.

Note

If the party at one of the two link ends has not enabled the RLDP, the diagnosis also shows two-way or one-way link fault. So, in configuring two-way link detection or one-way link detection, the administrator shall make sure that the RLDP is enabled at both ends to avoid the incorrect diagnosis information.

1.2 Default Configuration

Global RLDP status	Disabled
Port RLDP status	Disabled
Detection interval	3 seconds
Maximum detection times	2

Caution

The RLDP can be configured only on the basis of the switching interface (including AP) and the routing interface.

Caution

All RLDP frames are untagged.

Caution

In the RLDP fault processing type, the block function and the STP are mutually exclusive. In other words, if the fault processing type configured on the port is "block", it is recommended to disable STP; otherwise, since the STP cannot recognize one-way link, possibly the STP allows port forwarding but the RLDP is configured with port blocking.

1.3 Configuring RLDP

1.3.1 Enabling RLDP Globally

The RLDP works on the port only when the global RLDP is enabled.

In global configuration mode, follow these steps to enable RLDP:

Command	Function
Qtech(config)# **rldp enable**	Enables global RLDP function.
Qtech(config)# **end**	Returns to privileged EXEC mode.

Use the **no** form of this command to disable global RLDP.

1.3.2 Configuring RLDP on the Port

The RLDP operation is port-based, so the user needs to explicitly configure which ports shall run RLDP. In configuring the port RLDP, it is required to specify the diagnosis type and the troubleshooting method for the port at the same time. The diagnosis types include unidirection-detect, bidirection-detect and loop-detect. The troubleshooting methods include warning, block, shutdown-port, and shutdown-svi.

In global configuration mode, follow these steps to configure the RLDP on the port:

Command	Function
Qtech(config)# **interface** *interface-id*	Enters interface configuration mode.
Qtech(config-if)# **rldp port** {**unidirection-detect** | **bidirection-detect** | **loop-detect** } {**warning** | **shutdown-svi** | **shutdown-port** | **block**}	Enables RLDP on the port and configures the diagnosis type and troubleshooting method at the same time.
Qtech(config-if)# **end**	Returns to privileged EXEC mode.

The **no** form of this command disables RLDP on the port and the configured detection types one by one.

In the example below, the RLDP is configured on GigabitEthernet 0/5, and multiple diagnosis types and troubleshooting methods are specified:

```
Qtech# configure terminal
Qtech(config)# interface gigabitEthernet 0/5
Qtech(config-if)# rldp port unidirection-detect
shutdown-svi
Qtech(config-if)# rldp port bidirection-detect warning
Qtech(config-if)# rldp port loop-detect block
Qtech(config-if)# end
Qtech# show rldp interface gigabitEthernet 0/5
port state      : normal
local bridge    : 00d0.f822.33ac
neighbor bridge : 0000.0000.0000
neighbor port   :
unidirection detect information:
action : shutdown svi
state  :  normal
bidirection detect information :
action : warnning
state  :  normal
loop detect  information     :
action : block
state  :  normal
```

Several precautions in configuring port detection:

- The routing interface does not support the shutdown-svi error handling method, so this method is not executed in case of the occurring of detection error.

- In configuring loop detection, the neighbor devices downward connected with the port cannot enable the RLDP detection; otherwise, the port cannot have correct detection.
- If the block method is configured on the aggregated port and the link detection error happens, do not change the member port relations of the aggregate port before the port reset detection; otherwise, the forwarding status of the member interface may have unexpected effects of forwarding status.
- If the RLDP detects link error, alarm information will be given. The user can send the alarm information to the log server by configuring the log function. At least 3 levels of log shall be ensured.
- You are recommended to specify the diagnosis type of the loop detection to shutdown-port for the reason that for some devices, even if the device detects the loop and specifies the block port, a large amount of packets will be sent to the CPU for the hardware chip limitation.
- If you configure RLDP on AP port, you are recommended to specify the diagnosis type of loop detection to shutdown-port.

1.3.3 Configuring RLDP Detection Interval

The port with the RLDP function enabled will send the RLDP Probe messages on a regular basis.

In global configuration mode, follow these steps to configure the RERP detection interval:

Command	Function
Qtech(config)# **rldp detect-interval** interval	Configures the detection interval within the range 2-15s, 3s by default.
Qtech(config)# **end**	Returns to privileged EXEC mode.

The **no** form of the command restores the value to its default.

1.3.4 Configuring the Maximum RLDP Detection Times

If the port with RLDP enabled cannot receive messages from neighbors in the maximum detection period (maximum detection times X detection interval), that port will be diagnosed as faulty. See the Overview for details of the fault types.

In global configuration mode, follow these steps to configure the RERP maximum detection times:

Command	Function
Qtech(config)# **rldp detect-max** Num	Configures the maximum detection times, num range 2-10, 2 by default.
Qtech(config)# **end**	Returns to privileged EXEC mode.

The **no** form of this command restores the value to its default.

Note

The maximum detection times only take effect in the unidirectional link detection and bidirectional link detection, and will not take effect if only loop detection is enabled on a port.

1.3.5 Restoring the RLDP Status of the Port

The port with shutdown-port troubleshooting method configured cannot resume the RLDP detection actively after a fault occurs. If the user confirms the fault removed, run the recovery command to restart the RLDP on the shutdown port. This command sometimes may make the other ports with detection errors resume.

In privileged EXEC mode, follow these steps to resume the RLDP detection of the port:

Command	Function
Qtech# **rldp reset**	Makes any port with RLDP detection failure resume the detection.

Note

The **errdisable recover** command can be used in the global configuration mode to restart, instantly or at fixed time, the RLDP detection of the port that is set violation by RLDP. It is worth mentioning that when there are some relay devices between rldp ports, if you use **errdisable recover interval** to restore the fault timely, you need to set the value of rldp detection time greater than that of **errdisable recover interval**, that is, the value of detect-interval* detect-max total time is greater than that of **errdisable recover interval** to prevent error judgment.

1.3.6 Enabling RLDP Neighbor Negotiation

RLDP neighbor negotiation is disabled by default.

In global configuration mode, follow these steps to configure RLDP neighbor negotiation:

Command	Function
Qtech(config)# **rldp neighbor-negotiation**	Enables RLDP neighbor negotiation.
Qtech(config)# **no rldp neighbor-negotiation** Qtech(config)# **default rldp neighbor-negotiation**	Disables RLDP neighbor negotiation.

Note

With neighbor negotiation enabled, RLDP unidirectional-/bidirectional-link detection starts only after the neighbor negotiation is successful. (Receiving the Prob message from the neighbor indicates the neighbor negotiation is successful.)

1.4 Displaying RLDP Configuration

1.4.1 Displaying the RLDP Status of All Ports

In privileged EXEC mode, run the following commands to display the RLDP global configuration and the port detection information with RLDP detection configured:

Command	Function
Qtech# **show rldp**	Displays the RLDP global configuration and the port detection information with RLDP detection configured

In the example below, the **show rldp** command is used to display the detection information of all RLDP ports:

```
Qtech# show rldp
rldp state          : enable
rldp hello interval        : 2
rldp max hello      : 3
rldp local bridge   : 00d0.f8a6.0134
------------------------------------------------
interface GigabitEthernet 0/1
port state:normal
neighbor bridge     : 00d0.f800.41b0
neighbor port       : GigabitEthernet 0/2
unidirection detect information:
action        : shutdown svi
state         : normal

interface GigabitEthernet 0/24
port state:error
neighbor bridge     : 0000.0000.0000
neighbor port       :
bidirection detect information :
action        : warnning
state         : error
```

As shown above, port GigabitEthernet 0/1 is configured with unidirection detection. No error is detected now, and the port status is normal. Port GigabitEthernet 0/24 is configured with bidirection detection, and bidirection fault is detected.

1.4.2 Displaying the RLDP Status of the Specified Port

In privileged EXEC mode, run the following command to display the RLDP detection information of the specified port:

Command	Function
Qtech# **show rldp interface** *interface-id*	Displays the RLDP detection information of interface-id.

In the example below, the **show rldp interface GigabitEthernet** *0/1* command is used to display the RLDP detection information of port fas0/1:

```
Qtech# show rldp int GigabitEthernet 0/1
port state      :error
local bridge    : 00d0.f8a6.0134
neighbor bridge : 00d0.f822.57b0
neighbor port   : GigabitEthernet 0/1
unidirection detect information:
action: shutdown svi
state :  normal
bidirection detect information :
action : warnning
state :  normal
loop detect  information    :
action: shutdown svi
state :  error
```

As shown above, the port GigabitEthernet 0/1 is configured with three detection types: unidirection detection, bidirection detection and loop detection. The troubleshooting methods are shutdown-svi and warning. Error is found in loop detection so the current port status is error. Accordingly, the SVI of the port is shutdown.

2 CONFIGURING MSTP

2.1 STP and RSTP Overview

Qtech series supports both the STP protocol and the RSTP protocol, as well as complying with the IEEE 802.1D and IEEE 802.1w standards.

The STP protocol can prevent broadcast storm caused by link loops and provide link redundancy and backup.

For the layer 2 Ethernet, there is only one active channel between two LANs to avoid broadcast storm. However, it is necessary to set up redundant links to improve the reliability of a LAN. Furthermore, some channels should be in the backup status in order to take up its work when a link fails. It is obviously hard to control this process by manual. The STP protocol can complete this work automatically. It enables a device in a LAN to:

- Discover and activate an optimal tree-type topology of the LAN.
- Detect and fix failures and automatically update the network topology to offer the possible optimal tree-type structure at any time.

The LAN topology is automatically calculated by a set of bridge parameters set by the administrator. The proper configuration of these parameters is helpful to offer an optimal solution.

The RSTP protocol is completely compatible with the 802.1D STP protocol downward. As with traditional protocol, the RSTP protocol can prevent loop and offer link redundency. The most critical feature of the RSTP protocol is quickness. If the bridges in a LAN support the RSTP protocol and are configured appropriately by administrators, it will take no more than 1 second to re-span the topology tree once the network topology changes (it takes about 50 seconds for traditional STP protocol to re-span the topology tree).

⚠ Caution

For the switch buffer control, see the chapter *Buffer Control* in *Configuring QOS*.

2.1.1 Bridge Protocol Data Units (BPDU):

A stable tree-type topology depends on the following elements :

- The unique bridge ID of each bridge consists of the bridge priority and the MAC address.
- The root path cost refers to the cost from a bridge to the root bridge.
- Each port ID consists of the port priority and port number.

By exchanging the Bridge Protocol Data Units (BPDU) frame destined to the multicast address 01-80-C2-00-00-00 (in hex), bridges gets the information necessary for building the optimal tree-type topology.

A BPDU is comprised of the following elements:

- Root Bridge ID (root bridge ID that a bridge considers)
- Root Path cost (Root Path cost of a bridge).
- Bridge ID (ID of a bridge).
- Message age (the live time of the message)

- Port ID (port ID sending the message).
- Forward-Delay Time, Hello Time and Max-Age: time parameters.
- Other flag bits, such as network topology change and port status.

Once a port of a bridge receives a BPDU message whose priority is higher than its priority (or smaller bridge ID and smaller root path cost), the bridge will store this message on the port while updating and propagating them to all other ports. If the BPDU with lower priority is received, the bridge will discard this message.

This mechanism propagates a BPDU message of higher priority in the whole network. As a result:

- A bridge is elected to be the root bridge in the network.
- Each bridge other than the root bridge has a root port that offers a shortest path to the root bridge.
- Each bridge will calculate the shortest path to the root bridge.
- Each LAN has a designated bridge that lies in the shortest path between this LAN and the root bridge. The port for connecting the designated bridge and the LAN is referred to as the designated port.
- The root port and the designated port are in the forwarding status.
- Other ports beyond the spanning tree are in the discarding status.

2.1.2 Bridge ID

As specified in IEEE 802.1W standard, each bridge has an unique bridge ID based on which the root bridge is elected in spanning tree algorithm. The bridge ID consists of eight bytes, in which the last six bytes are the MAC address of the bridge, and the first two bytes are shown in the table below. Of which, the first four bits denote the priority, while the last twelve bits denote the system ID for extending the protocol in the future. This value is 0 in the RSTP, so the priority of the bridge should be configured as the multiple of 4096.

	Priority value				**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

2.1.3 Spanning-Tree Timers

The following describes three timers impacting the performance of spanning tree.

- Hello timer: Interval to send the BUDU message.
- Forward-Delay timer: Interval to change the port status, that is, the time interval at which the port switches from the listening status to the learning status and vice versa when the RSTP protocol runs in the compatible STP protocol mode.
- Max-Age timer: The longest time for the BPDU message. The system will discard the message when the timer times out.

2.1.4 Port Roles and Status

A port plays a role to present its function in the network topology.

- Root port: The port that provides the shortest path to the root bridge.
- Designated port: The port through which each LAN is connected to the root bridge.
- Alternate port: The alternate port of the root port that will take up its work when the root port fails.
- Backup port: The backup port of the designated port. If two ports of a bridge are connected to a LAN, the port with higher priority is the designated port and the other one is the backup port.
- Disable port: The port that is not in the active status, namely, the ports whose operation status is down.

Figure 1, Figure 2 and Figure 3 below show the roles of various ports:

R = Root port D = Designated port A = Alternate port B = Backup port

Unless otherwise stated, the priorities of these ports are in the descending order from left to right.

Figure-1

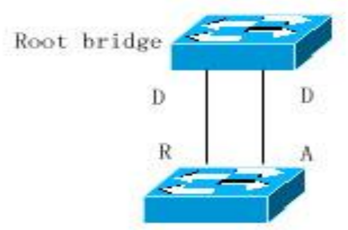


Figure-2

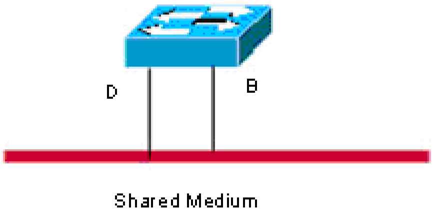
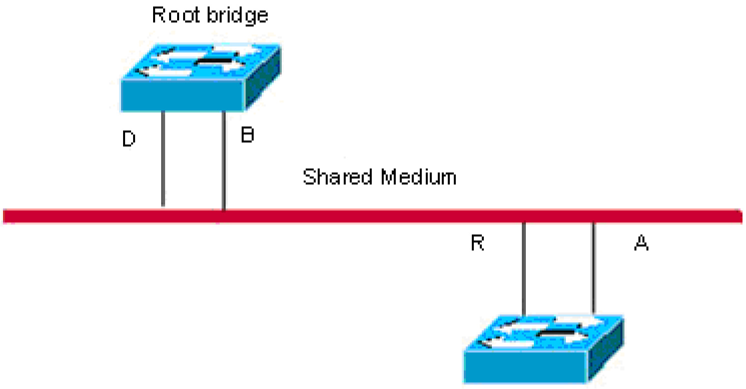


Figure-3



There are three port states for every port to indicate whether the data packet is forwarded and control the topology of the whole spanning tree.

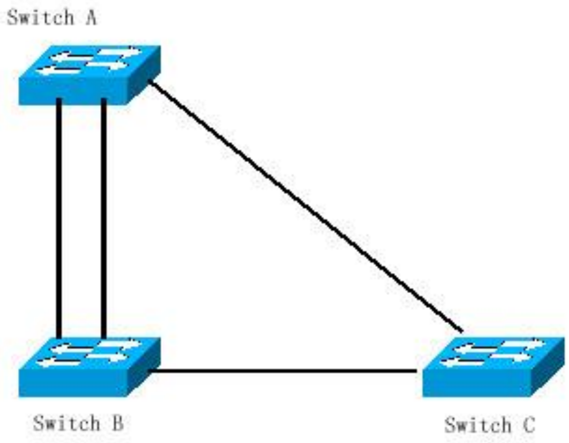
- Discarding: Neither forward the received frame nor learn about the source Mac address.
- Learning: Do not forward the received frame, but learn about the source Mac address, so it is a transitional status.
- Forwarding: Both forward the received frame and learn about the source Mac address.

For the stable network topology, only the root port and designated port can be the forwarding status, while other ports are only in the discarding status.

2.1.5 Generating a Network Topology Tree (Typical Application Solution)

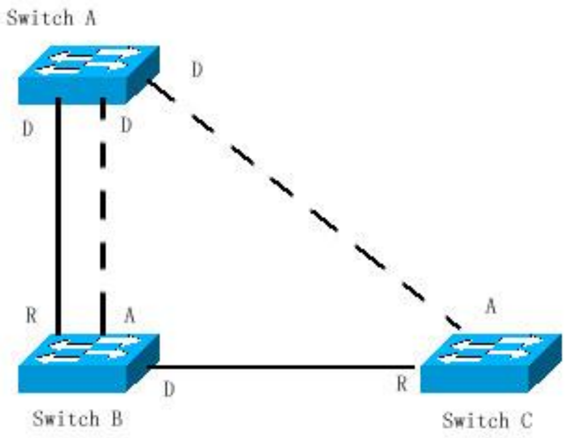
We now describe how the STP and RSTP protocols span a tree-type structure by the mixed network topology. As shown in Figure 4, the bridge IDs of Switches A, B and C are assumed in the ascending order. Namely, Switch A presents the highest priority. There is the 1000M link between switch A and switch B, and the 10M link between switch A and switch C, while it is the 100M link between switch B and switch C. Switch A acts as the backbone switch of this network and implements the link redundancy for both Switch B and Switch C. Obviously, broadcast storm would occur if all these links are active.

Figure-4



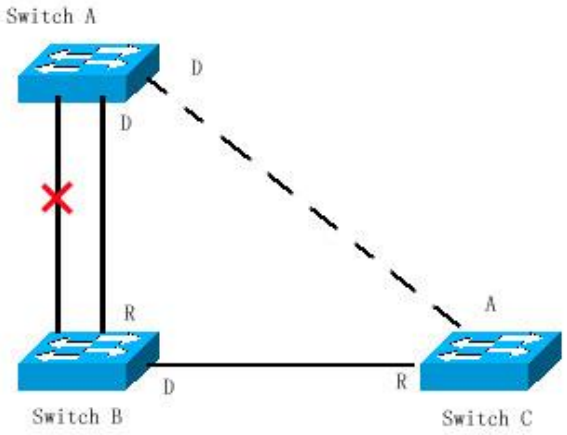
If all of these three switches enable the Spanning Tree protocol, they will select switch A as the root bridge by exchanging BPDU message. Once Switch B detects that two ports are connected to Switch A, it will select the port with the highest priority as the root port, while another one is selected as the alternate port. Meanwhile, Switch C detects that it can reach Switch A through Switch B or directly. However, Switch C discovers that the cost of the path from Switch B to Switch A is lower than that directly (For the costs corresponding to various paths, refer to table ***), so Switch C selects the port connected with Switch B as the root port, while the one that connected with Switch A as the alternate port. Various ports enter the corresponding status after their roles are determined. As a result, the network topology is generated as shown in Figure 5.

Figure-5



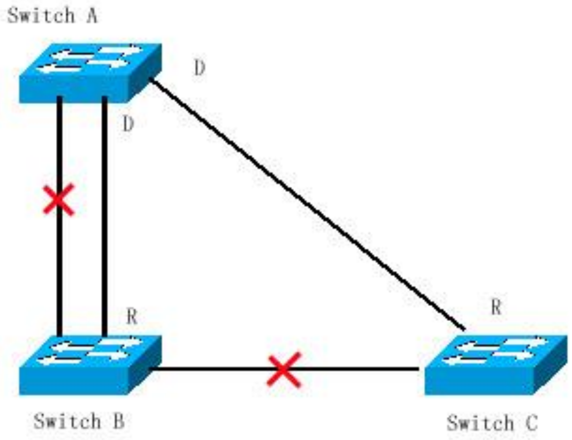
If the active path between Switch A and Switch B fails, the backup path will work. Consequently, the network topology is generated as shown in Figure 6.

Figure-6



If the path between Switch B and Switch C fails, Switch C will automatically switch the alternate port to the root port. Consequently, the network topology is generated as shown in Figure 7.

Figure-7



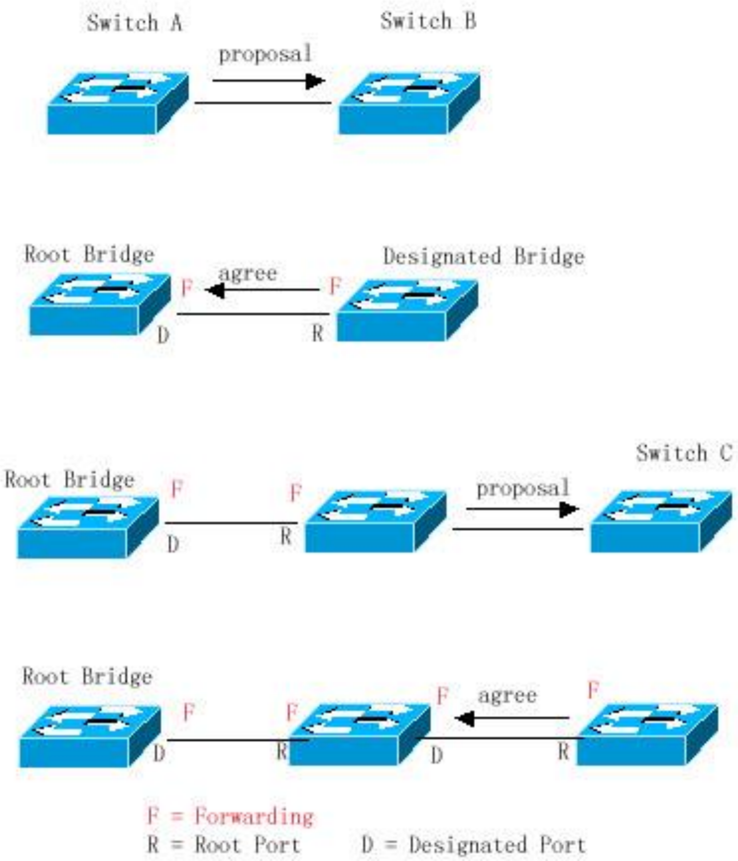
2.1.6 Rapid Convergence of RSTP

The following introduces the special function of RSTP: enabling rapid forwarding on a port.

The STP protocol will forward packets after 30s since the port roles are selected, which is twice as the Forward-Delay Time (you can set the Forward-Delay Time, which is 15s by default). Furthermore, the root port and designated port of each bridge will carry out the forwarding again after 30s, so it will take about 50s to stabilize the tree-type structure of the whole network topology.

The forwarding procedure of the RSTP protocol is different from that of the STP protocol. As shown in Figure 8, Switch A sends the specific proposal message of the RSTP protocol. Switch B detects that the priority of Switch A is higher than itself, takes the Switch A as the root bridge and the port that receives the message as the root port and forwards the proposal message. Then it sends the Agree message to Switch A through the root port. Upon the receipt of the proposal message, Switch A will forward the message through its designated port. After that, Switch sends the proposal message through the designated port to extend the spanning tree in turn. In theory, the RSTP protocol can immediately restore the tree-type network structure to implement rapid convergence when the network topology changes.

Figure-8



Caution

Point-to-point Connection" between ports is required for the above "handshaking" process. In order to make full use of you device, do not use non-point-to-point connection between devices.

Other than Figure 9, other schematics in this chapter are the point-to-point connection. The following lists the example figure of the non point-to-point connection.

Example of Non Point-to-point Connection:

Figure-9

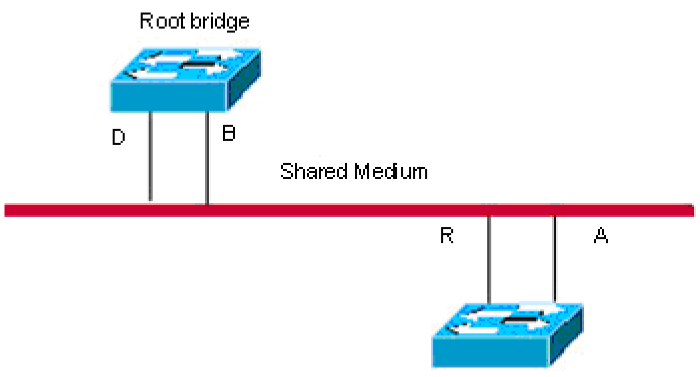
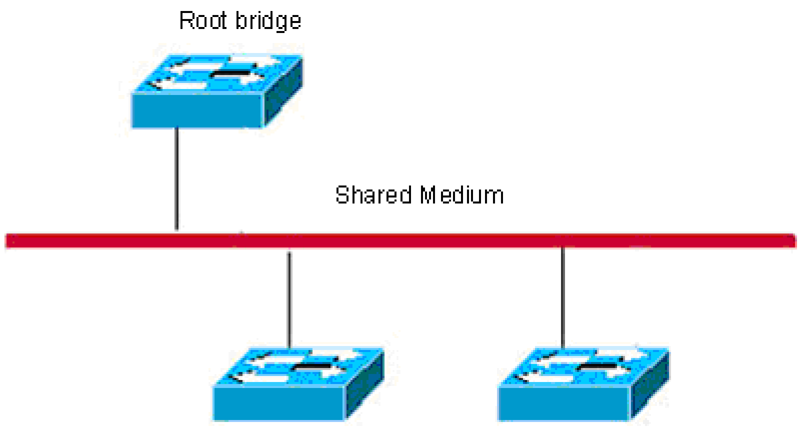
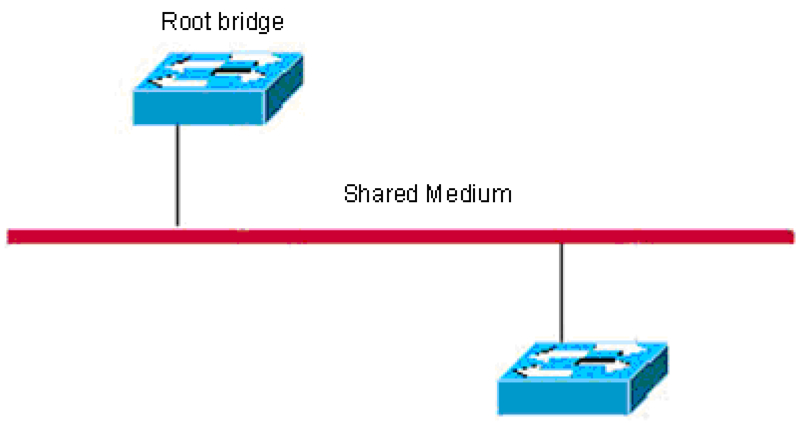


Figure-10



In addition, the following figure is a point-to-point connection and should be differentiated by users carefully.

Figure-11



2.1.7 Compatibility of RSTP and STP

The RSTP protocol is completely compatible with the STP protocol. It will judge whether the connected bridge supports the STP protocol or the RSTP protocol by the version number of the received BPDU message automatically. Only the forwarding process of the STP protocol is executed in the case of that the bridge supports the STP protocol. This cannot maximize the performance of the RSTP protocol.

Furthermore, using the RSTP protocol and the STP protocol will cause a problem. As shown in Figure 17-12, Switch A supports the RSTP protocol, while Switch B supports the STP protocol. Both switches are connected with each other. Switch A will send the STP BPDU message to Switch B for compatibility. However, if Switch A is connected with the RSTP-enabled Switch C, Switch A still sends the STP BPDU message, and thus causing that Switch C considers Switch A a STP-enabled bridge. As a result, two RSTP-supported switches run the STP protocol, reducing their efficiency greatly.

For this reason, the RSTP protocol provides the protocol-migration function to send the RSTP BPDU message forcibly in case that the peer bridge must support RSTP. In this way, Switch C will detect the bridge connected with it supports the RSTP protocol, so both two devices can run the RSTP protocol as shown in Figure 13.

Figure-12 Protocol Migration

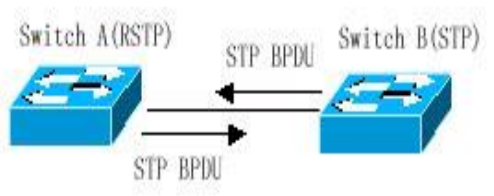
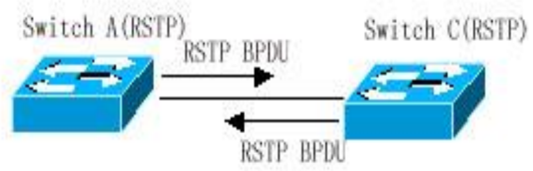


Figure-13



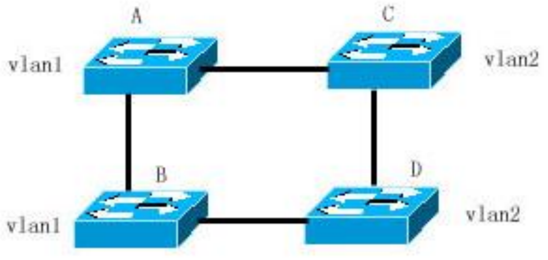
2.2 MSTP Overview

Qtech series supports the MSTP protocol, a new spanning-tree protocol derived from the traditional STP and RSTP protocols that includes the rapid forwarding mechanism of the RSTP protocol itself.

Since traditional spanning tree protocols are not related to a VLAN, the following problems may occur in a specific network topology.

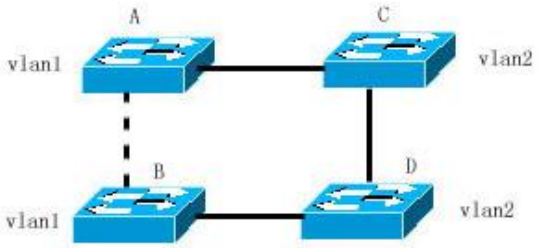
As shown in Figure 14, Switches A and B are located in Vlan1, and switches C and D in Vlan2. They form a loop.

Figure-14



If the cost of the path from Switch A through Switch C, Switch D to Switch B is smaller than that of the direct path from Switch A to Switch B, the latter path will be torn down, as shown in Figure 15. Packets in Vlan1 can not be forwarded because Switches C and D do not contain Vlan1. In this way, Vlan1 of Switch A cannot communicate with Vlan1 of Switch B.

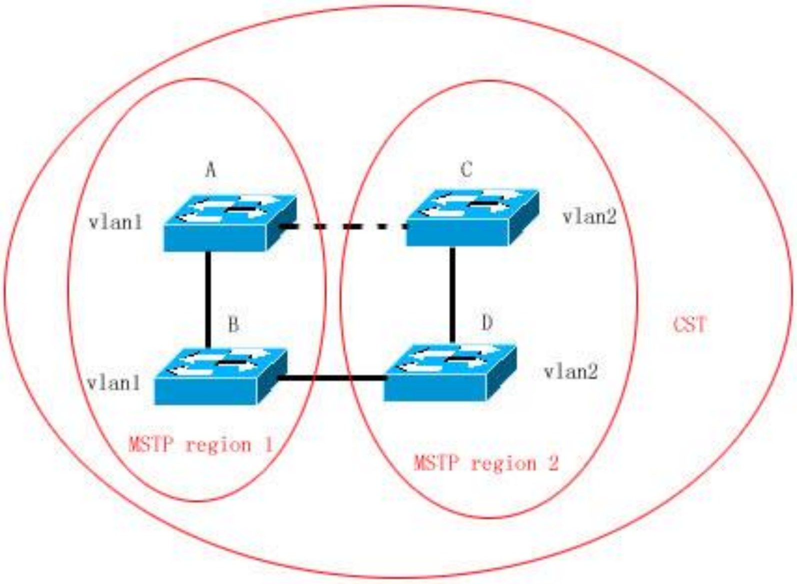
Figure-15



The MSTP protocol is developed to address this problem. It partitions one or more vlans of the switch into an instance, so the switches with the same instance configuration form a region (MST region) to run a separated spanning tree (this internal spanning-tree is referred to as the IST). The MST region is equivalent to a large device, which executes the spanning tree algorithm with other MST regions to obtain a whole spanning tree, referred to as the common spanning tree (CST).

With this algorithm, the above mentioned network can form the topology shown in Figure 16. Switches A and B are within the MSTP region 1 without a loop, so no path is discarded. This is also the case in the MSTP region 2. Region 1 and region 2 serve as two large devices respectively. There is a loop between them, so one path is discarded according to related configuration.

Figure-16



In this way, no loop occurs and the communication between the devices in a VLAN works as well.

2.2.1 How to Partition MSTP regions

According to above description, MSTP regions should be partitioned rationally and the switches in a MSTP region should be configured similarly for the MSTP protocol to work properly.

The MST configuration information contains:

- MST region name (name): A string of up to 32 bytes identifying the MSTP region.
- MST revision number: A revision number of 16 bits identifying the MSTP region.
- MST instance-vlan table: Each device can create up to 64 instances with IDs ranging from 1 to 64). Instance 0 always exists, so the system totally supports 65 instances. You can allocate 1 to 4094 VLANs for different instances (0 to 64) as needed, and the unallocated VLANs belong to instance 0 by default. In this way, each MSTI (MST instance) is a VLAN group and executes the spanning tree algorithm within the MSTI according to the MSTI information of the BPDU without the effect of the CIST and other MSTIs.

You can use the **spanning-tree mst configuration** command in the global configuration mode to enter the MST configuration mode and configure above information.

The MSTP BPDU carries above information. If a device has received the same MST configuration information of the BPDU as that of itself, it considers that the device connecting to this port belong to the same MST region as itself.

You are recommended to configure the instance-vlan table while the STP protocol is disable, and then enable the MSTP protocol to ensure the stability and convergence of the network topology.

2.2.2 Spanning Tree within a MSTP region (IST)

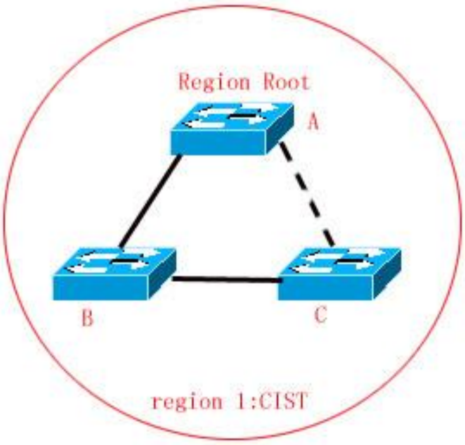
After MSTP regions are partitioned, a root bridge is elected for every instance within a region and the port role is determined for every port on a switch. A port is forwarded or discarded within an instance depneds on its role.

In this way, the IST (Internal Spanning Tree) is formed by exchanging the MSTP BPDU message, and various instances have their own spanning trees (MSTI). The spanning tree corresponding to the instance 0 is referred to as the CIST (Common Instance Spanning Tree) in conjunection with CST. That is to say, each instance provides each VLAN group with a single network topology without loop.

As shown in the following figure, Switches A, B and C form a loop within the region 1.

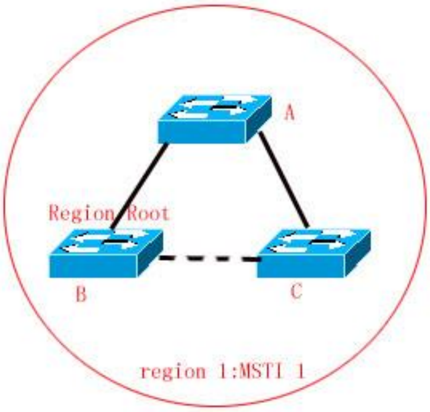
Switch A with the highest priority is selected as the region root in the CIST (instance 0). Then, the path between Switches A and C is discarded according to other parameters. Hence, for the VLAN group of instance 0, only the path from switch A to B and switch B to switch C are available, which break the loop of the VLAN group.

Figure-17



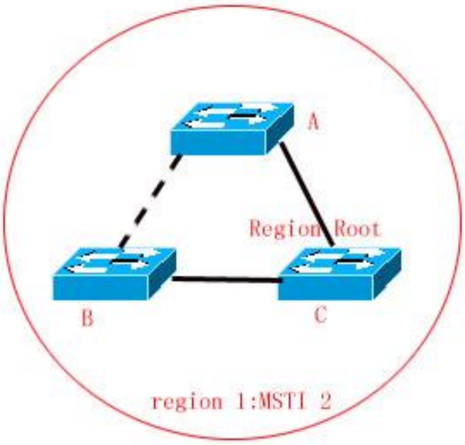
As shown in Figure 18, switch C with the highest priority is selected as the region root in the MSTI 1 (instance 1). Then, the path between switch A and B is discarded according to other parameters. Hence, for the VLAN group of instance 1, only the path from switch A to switch B and switch A to switch C are available, which break the loop of the VLAN group.

Figure-18



As shown in Figure 19, switch B with the highest priority is selected as the region root in the MSTI 2 (instance 2). Then, the path between switch B and switch C is discarded according to other parameters. Hence, for the VLAN group of instance 2, only the path from switch A to switch B and switch B to switch C are available, which break the loop of the VLAN group.

Figure-19

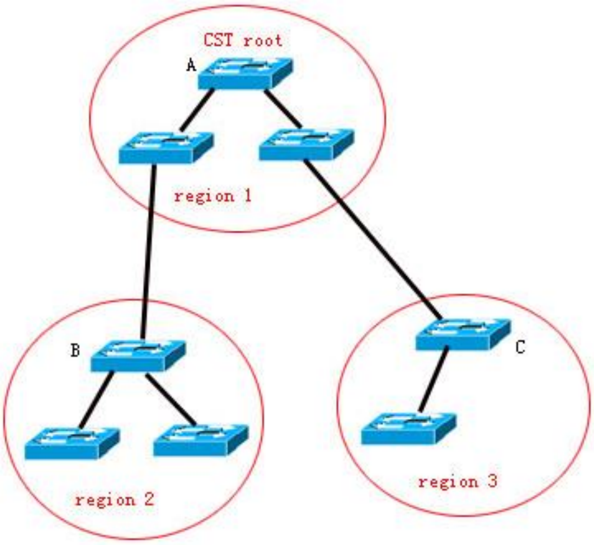


It should note that the MSTP protocol is not concerned on which VLAN a port belongs to, so users should configure corresponding path costs and priorities for ports according to actual VLAN configuration to prevent the MSTP protocol from breaking the loop unnecessarily.

2.2.3 Spanning Tree between MSTP regions (CST)

For CST, each MSTP region is equivalent to a large-sized device, and different MSTP regions also form a large-sized network topology tree, referred to as CST (common spanning tree). As shown in Figure 20, for CST, switch A with the smallest bridge ID is selected as the root of the entire CST (CST Root) and the CIST Regional Root in this region. In Region 2, since the root path cost from switch B to the CST root is the lowest one, switch B is selected as the CIST Regional Root in this region. Similarly, switch C is selected as the CIST Regional Root in Region 3.

Figure-20



The CIST Regional Root is not necessarily the device with the smallest bridge ID in that region. It is the device in the region that has the lowest root path cost to the CST root.

At the same time, the root port of the CIST regional root takes a new port role for the MSTI, namely the **Master port,** as the outlet of all instances, which is forwarded to all instances. In order to make the topology more stable, it is recommended to configure the outlet of the regions to the CST root on one device of this region as much as possible!

2.2.4 Hop Count

The IST and MSTI will not take the message age and Max age to calculate whether the BPDU message is timeout. Instead, they use the mechanism similar to the TTL of IP packets, namely hop count.

You can set it by using the **spanning-tree max-hops** command in the global configuration mode. The hop count is reduced by 1 when the BPDU message psses through a device in a region starting from the region root bridge until it is 0, which means the BPDU message is timeout. A device will discard the BPDU message whose hop count is 0.

In order to be compatible with the STP protocol and the RSTP protocol out a region, the MSTP protocol still remains the Message age and Max age mechanisms.

2.2.5 Compatibility of MSTP with RSTP and STP

For the STP protocol, the MSTP protocol will send the STP BPDU to be compatible with it like the RSTP protocol. For detailed information, refer to the Compatibility of RSTP and STP section.

For the RSTP protocol, it will process the CIST part of the MSTP BPDU, so it is not necessary for the MSTP to send the RSTP BPDU to be compatible with it.

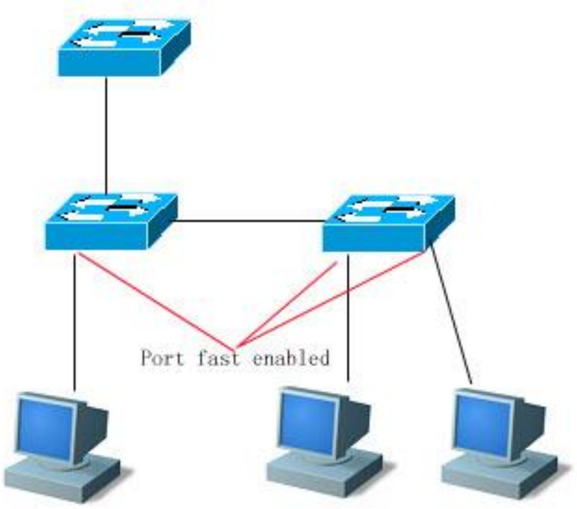
Each device that runs the STP or RSTP protocol is an independent region, and does not form the same region with any other device.

2.3 Overview of Optional Features of MSTP

2.3.1 Understanding Port Fast

If a port of a device is connected with the network terminal directly, this port can be set as the Port Fast to forward packets directly. The port does not need to wait 30 seconds before forwarding packets, which is the case when the port is not set to Port Fast. The following figure indicates which ports of a device can be set to Port Fast.

Figure-21



If the BPDU message is received from the Port Fast enabled port, its Port Fast operational state is disabled. At this time, this port will execute the forwarding by normal STP algorithm.

2.3.2 Understanding AutoEdge

If the specified port doesn’t receive the BPDU message sent by the downstream port within a certain period of time (3 seconds), the port will be considered that it connects a network device and set as an edge port to enter the Forwarding status directly. An edge port will be automatically identified as a non-edge port after receiving the BPDU message.

You can disable the automatic identification function of the edge port by the **spanning-tree autoedge disabled** command.

This function is enabled by default.

Caution

When the AutoEdge function conflicts with the manually-configured Port Fast function, the latter shall prevail.

AutoEdge function can be used for rapid negotiation forwarding between the designated port and the downstream port, so the STP protocol doesn’t support AutoEdge. If the designated port is in the forwarding status, Autoedge does not take effect on the port. It will take effect during repaid renegotiation such as pluging/unpluging network cables.

If a port enables the BPUD Filter, it forwards the BPDU message directly, but not be identified as the edge port automatically.

AutoEdge function is only applicable for the designated port.

AutoEdge complies with the standard definition of IEEE 802.1D (version 2004), in which the parameter range of Bridge Hello Time has been modified as 1.0-2.0. Therefore, you shall confirm that the Hello Time value is within the range when using AutoEdge function, or the risk of temporary loop will occur. It is recommended to disable AutoEdge function if it is neccesary to exceed the range of Hello Time.

2.3.3 Understanding BPDU Guard

The BPDU guard can be enabled globally or on individual interface. There are some slightly difference between these two ways.

You can use the **spanning-tree portfast bpduguard default** command to open the global BPDU guard enabled status in the global configuration mode. In this status, if the BPDU message is received through a Port Fast-enabled port or a AutoEdge port, this port will enter the error-disabled status, indicating the configuration error. At the same time, the port will be closed to show that some illegal users may add a network device to the network, which change the network topology.

You can also use the **spanning-tree bpduguard enable** command to enable BPDU guard on individual interface in the interface configuration mode (it is not related to whether it is AutoEdge port or not). Under this situation, it will enter the error-disabled status if this interface receives the BPDU message.

2.3.4 Understanding BPDU Filter

The BPDU filter can be enabled globally or on individual interface. There are some slightly difference between these two ways.

You can use the **spanning-tree portfast bpdufilter default** command to enable the BPDU filter globally in the global configuration mode. In this status, the BPDU messages can not be received or sent through a Port Fast-enabled port or a AutoEdge port, leading to no BPDU messages received by the host directly connecting the port. The BPDU filter will be disabled when the Port Fast is disabled for the AutoEdge port receives the BPDU message.

You can also use the **spanning-tree bpdufilter enable** command to enable the BPDU filter on individual interface in the interface configuration mode (it is not related to whether it is AutoEdge port or not). In this situation, this interface will not receive or transmit the BPDU message, but execute the forwarding directly.

2.3.5 Understanding Tc-protection

TC-BPDU messages are BPDU messages carrying with TC flag. When the L2 switch receives these messages, the network topology will change and the MAC address table will be deleted. And for L3 switch, the route table will be deleted and the port state in the ARP entry will change. To prevent the switch from processing abovementioned operations when pseudo TC-BPDU messages attack maliciously, too-heavy burden and network turbulance, the TC-protection function comes into being.

Tc-protection can only be enabled or disabled globally. It is enabled by default.

Once Tc-protection is enabled, the switch will delete the message within a certain period of time (usually 4 seconds) after receiving the TC-BPDU message while monitoring the TC-BPDU message. If it receives the TC-BPDU message during this period, it will perform the delete operation again after this period expires. This eliminates the need of frequently deleting MAC address entries and ARP entries.

2.3.6 Understanding TC Guard

The Tc-Protection function can reduce the removal of MAC address entries and ARP entries when a lot number of TC messages are generated in a network. However, you need to do more delete oeprations in case of TC message attack. Furthermore, the TC message is propagated and will have an effect on the whole network. The TC Guard function allows you to disable the propagation of the TC message globally or on ports. When TC Guard function is configured globally or on a port, the port will shield the TC messages received or produced to prevent from propagating them to other ports. In this way, this funciton can manage TC message attack in the network and maintain the network stability. Moreover, this function can prevent from interrupting core routes due to the oscillation of the devices on the access layer.

⚠ Caution

Network communication will be broken off if you use tc-guard function incorrectly.

You are recommended to enable this function when you ensure that there is illegal tc message attack in the network.

If you enable global tc-guard, then all the ports will not spread tc message. It is applicable for those devices that are accessed on the desk to enable this function.

If you enable interface tc-guard, then the topology change and tc message received on this port will not be spreaded to other ports. It is applicable for up-link ports especially aggregated ports to enable this function.

2.3.7 Understanding TC Filtering

With the TC Guard function enabled, the port will not propagate TC message to other ports participating in the spanning tree calculation on the local device. The TC message here includes the TC message received on the port, and the TC mesasge produced by the port itself. The latter one refers to the TC message generated when the forwarding state of the port changes (For example, port state change from block to forwarding), which indicates the topology may be changed.

As TC message propagation is prevented by TC Guard, the device will not clear the MAC addresses of the coppresponding ports when the topology changes, resulting in data forwarding failure.

TC filtering is introduced to solve the above problems. TC filtering will process the TC message in the condition of normal topology change instead of the TC message received on the port, so that address clearing and core route interruption caused by frequent UP/DOWN on the port without Portfast configured can be solved, and the core routing entries can be updated in time when the topology changes.

⚠ Caution

By default, the TC filtering function is disabled.

2.3.8 Understanding BPDU Source MAC Check

The gobal of the BPDU source MAC check funciton is to prevent malicious attack on the switch by sending the BPDU message manually and thus cause the MSTP protocol work abnormally. When the peer switch connected to a port in the point-to-point mode is determined, enabling the BPDU source MAC check function can receive only the BPDU message from the remote switch and discard all other BPDU messages to protect against malicious attacks. You can configure the corresponding MAC addresses for the BPDU source MAC check fucntion on a specific port in the interface mode. Only one MAC address is configured for one port. BPDU source MAC check can be disabled by using the **no bpdu src-mac-check** command. In this case, any BPDU message is received on the port.

2.3.9 Understanding Invalid Length Filtering for BPDU

When the Ethernet length field of the BPDU message exceeds 1500 bits, this BPDU message is discarded in order to avoid receiving invalid BPDU messages.

2.3.10 Understanding ROOT Guard

In network design, root bridge and backup root bridge are always divided in the same region. Due to error configuration of accendant and malicious attack in the network, it is possible that root bridge receives configuration message of higher priority and loses the current root bridge position, leading to error turbulance of network topology, which Root Guard function can prevent from occuring.

When enabling Root Guard, it enforces the port role of all the instances as specified port. Once the port receives configuration message of higher priority, Root Guard will set the interface as root-inconsistent (blocked). If there is no configuration message of higher priority during the time long enough, the port will be restored to be the original normal status.

You shall disable ROOT Guard function if this function results in the blocked status for interfaces and it needs manual configuration to restore to the normal status. You can use the command **spanning-tree guard none** in the interface configuration mode to disable Root Guard function. This function is enabled by default.

Caution

Incorrectly using ROOT Guard leads to network link breakdown.

If you enable ROOT Guard on non-designated port, the non-designated port will be enforced as designated port and show BKN status (blocking status).

If MST0 enters BKN status because it receives configuration message of higher priority on a port, ROOT Guard will enforce the port in all the other instances to enter BKN status.

ROOT Guard or LOOP Guard takes effect at the same time. That is , they cannot both take effect at the same time .

The AutoEdge function is disabled when enabling the ROOT Guard-enabled port.

2.3.11 Understanding LOOP Guard

Due to breakdown of one-way link, root port or backup port becomes designated port, being ready to forward because they can not receive BPDU, causing the loop in the network, which Loop Guard function can prevent.

For the ports configured loop guard, if they can not receive BPDU, the port roles will be migrated. However, the port state is always set as discarding till the port receive BPDU again and recalculate spanning tree.

Caution

You can enable LOOP Guard based on global or interface.

ROOT Guard or LOOP Guard takes effect at the same time. That is , they can not both take effect at the same time .

The AutoEdge function on all interfaces is ineffective when enabling LOOP Guard function globally.

The AutoEdge function on the interface is ineffective when enabling LOOP Guard function in the interface configuration mode.

2.4 Configuring MSTP

2.4.1 Default Spanning Tree Configuration

The following table lists the default configuration of the Spanning Tree protocol.

Item	Default value
Enable State	Disable
STP MODE	MSTP
STP Priority	32768
STP port Priority	128
STP port cost	Automatically determine according to port rate.
Hello Time	2 seconds
Forward-delay Time	15 seconds
Max-age Time	20 seconds
Default calculation method of the Path Cost	Long
Tx-Hold-Count	3
Link-type	Automatically determine by the duplex status of the port.
Maximum hop count	20
Corresponding relationship between vlan and instance	All VLANs belong to instance 0 Only instance 0 exists

You can restore the STP parameters to its default configuration (except for disabling STP) by using the **spanning-tree reset** command.

2.4.2 Enabling and Disabling the Spanning Tree Protocol

The spanning tree protocol is disabled on the device by default.

To enable the spanning tree protocol, execute the following command in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree**	Enable the spanning tree protocol.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show spanning-tree**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To disable the spanning tree protocol, use the **no spanning-tree** command in the global configuration mode.

2.4.3 Configuring the Spanning Tree Mode

According to the 802.1-related protocols, it is not necessary for administrators to set much for three versions of the spanning tree protocols such as the STP, RSTP and MSTP. These versions are compatible with one another naturally. However, given that some manufacturers will not develop the spanning tree protocol by standards, it may cause some compatibility problem. Hence, we provide a command to facilitate administrators to switch to the lower version of the spanning tree protocol for compatibility when they detect that this device is not compatible with that of other manufacturers.

Note: When you switch to the RSTP or STP version from the MSTP version, all information about MSTP Region will be cleared.

The default mode of the device is MSTP.

To enable the spanning tree protocol, execute the following command in the privileged EXEC mode:

Command	Function

Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree mode** [**stp** | **rstp** | **mstp**]	Set the spanning tree version.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show spanning-tree**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the spanning tree mode to the default value, use the **no spanning-tree mode** command in the global configuration mode.

2.4.4 Configuring Switch Priority

Switch priority allows you to select the root and draw the topology of a network. It is recommended that administrators set the core device with higher priority (or smaller value) to facilitate the stablization of the whole network. You can assign different switch priorities for various instances so that various instances can run separate spanning tree protocol.Only the priority of CIST (Instance 0) is related to the devices between different regions.

As mentioned in Bridge ID, there are 16 values for the priority, and all of them are multiples of 4096, which are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. The default value is 32768.

To configure switch priority, execute the following command in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree** [**mst** *instance-id*] **priority** *priority*	Configure different switch priorities for different instances. This command configures the switch priority for instance 0 without the instance-id parameter. *instance-id*: ID of the instance in the range from 0 to 64. *priority*: switch priority in the range from 0 to 61440 and is increased by the integral multiple of 4096, 32768 by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the switch priority to the default value, use the **no spanning-tree mst** *instance-id* **priority** command in the global configuration mode.

2.4.5 Configuring Port Priority

When two ports are connected to the shared media, the device will set the one of the higher priority (or smaller value) to be the forwarding status and the one of the lower priority (or larger value) to be the discarding status.If the two ports are of the same priority, the device will set the one with the smaller port number to the forwarding status. You can assign different port priorities to various instances on one port, by which various instances can run the separated spanning tree protocols.

Same as device priority, it has 16 values, all a multiple of 16. They are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240 respectively. The default value is 128.

To configure a port priority, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. A legal interface contains a physical port and an aggregate Link.

Qtech(config-if)# **spanning-tree** [**mst** *instance-id*] **port-priority** *priority*	Configure different priorities for different instances. The command without the *instance-id* parameter will configure a port priority for instance 0. *instance-id*: Interface ID in the range from 0 to 64. *priority*: Port priority of an instance in the range from 0 to 240. Furthermore, it is increased by the integral multiple of 16, 128 by default.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show spanning-tree** [**mst** *instance-id*] **interface** *interface-id*	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the port priority to the default value, execute the **no spanning-tree mst** *instance-id* **port-priority** command in the interface configuration mode.

2.4.6 Configuring Path Cost of a Port

The switch determines a root port upon the total of the path costs along the path from a port to the boot bridge. The port the total of paths costs from the port to the root brdige is the smallest is elected the root port. Its default value is calculated by the media speed of the port automatically. The higher the media speed, the smaller the cost is. It is not necessary for administrators to change it for the path cost calculated in this way is most scientific. You can assign different cost paths for various instances on one port, by which various instances can run the separated spanning tree protocols.

To configure the path cost of a port, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. A legal interface contains a physical port and an aggregate Link.
Qtech(config-if)# **spanning-tree** [**mst** *instance-id*] **cost** *cost*	Configure different priorities for different instances. The command without the *instance-id* parameter will configure a port priority for instance 0. *instance-id*: Interface ID in the range of 0 to 64. *cost*: Path cost of the port in the range of 1 to 200,000,000. The default value is calculated by the media rate of the port automatically.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show spanning-tree** [**mst** *instance-id*] **interface** *interface-id*	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the path cost of a port to the default value, execute the **no spanning-tree mst** *cost* command in the interface configuration mode.

2.4.7 Configuring the Default Calculation Method of Path Cost (path cost method)

If the path cost of a port is the default value, the device will calculate the path cost of this port by port rate. However, IEEE 802.1d-1998 and IEEE 802.1t specify different path cost values for a port rate respectively. The value range of the 802.1d-1998 is short (1 to 65535), while the value range of the 802.1t is long (1 to 200,000,000).

There are two modes for the Cost value of AP: 1) our private mode fixes it to: the Cost value of the physical port * 95%; 2) the standard value is 20,000,000,000/ the actual link bandwidth of AP (The actual link bandwidth is: the bandwidth of member port * the number of UP member ports). Administrators should unify the path cost standard of the whole network. The default mode is long (IEEE 802.1t Mode).

The following table lists the path costs set for different port rates in two standards.

Port Rate	Interface	IEEE 802.1d (short)	IEEE 802.1t (long)	IEEE 802.1t (long standard)
10M	Common Port	100	2000000	2000000
	Aggregate Link	95	1900000	2000000 ÷ linkupcnt
100M	Common Port	19	200000	200000
	Aggregate Link	18	190000	200000 ÷ linkupcnt
1000M	Common Port	4	20000	20000
	Aggregate Link	3	19000	20000 ÷ linkupcnt
10000M	Common Port	2	2000	2000
	Aggregate Link	1	1900	2000 ÷ linkupcnt

Caution

1. The default path cost mode is long. After changing the path cost to the standard mode, the cost of AP will vary with the number of UP member ports. The change of port cost value may result in network topology change.

2. For the static AP, the linkupcnt in the table is the number of UP member ports; for the LACP AP, the linkupcnt refers to the number of member ports participating in AP data forwarding. If there is no linkup on the member port, the linkupcnt is 1. For detailed configurations about AP and LACP, refer to *AP-SCG* and *LACP-SCG*.

To configure the default calculation method of path cost, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree pathcost method** { { **long** [**standard**] } | **short** }	Configure the default calculation method of the port path cost as long, standard long or short, with long by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the setting to the default value, execute the **no spanning-tree pathcost** method command in the global configuration mode.

2.4.8 Configuring Hello Time

Configure the interval of sending the BPDU message. The default value is 2s.

To configure the Hello Time, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree hello-time** *seconds*	Configure the hello time ranging from 1 to 10s, 2s by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the hello time to the default value, execute the **no spanning-tree hello-time** command in the global configuration mode.

2.4.9 Configuring Forward-Delay Time

Configure the interval for changing port status. The default value is 15s.

To configure the forward-delay time, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree forward-time** *seconds*	Configure the forward delay time ranging from 4 to 30 seconds, 15 seconds by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the forward-delay time to the default value, execute the **no spanning-tree forward-time** command in the global configuration mode.

2.4.10 Configuring Max-Age Time

Configure the maximum period of time before the BPDU message is aged out. The default value is 20s.

In the privilege mode, perform these steps to configure the Max-Age Time:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree max-age** *seconds*	Configure the max age time ranging from 6 to 40 seconds, 20 seconds by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the max age time to the default value, execute the **no spanning-tree max-age** command in the global configuration mode.

Caution

Hello Time, Forward-Delay Time and Max-Age Time have their own value ranges. Meanwhile, the following condition must be addressed: 2*(Hello Time + 1.0 seconds) <= Max-Age Time <= 2*(Forward-Delay – 1.0 second). Otherwise, it may cause the topology instability.

2.4.11 Configuring Tx-Hold-Count

Configure the maximum number of the BPDU message sent per second, 3 by default.

To configure the Tx-Hold-Count, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree tx-hold-count** *numbers*	Configure the maximum number of the BPDU message sent per second in the range of 1 to10, 3 by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the default setting, execute the **no spanning-tree tx-hold-count** command in the global configuration mode.

2.4.12 Configuring Link-type

Configure the link-type of a port. This is crucial for rapid RSTP convergence. For details, refer to Rapid RSTP Convergence. Without configuration, the device will set the link type of a port according to its duplex status automatically, with point-to-point for the full duplex port and shared for the half duplex port.

To configure the link type of a port, execute the following commands in the interface configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode.
Qtech(config-if)# **spanning-tree link-type point-to-point/shared**	Configure the link type of the interface, with point-to-point for the full duplex port and shared for the half duplex port. Point-to-point indicates the rapid forwarding is enabled on the port.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the link type of a port to the default value, execute the **no spanning-tree link-type** command in the interface configuration mode.

2.4.13 Configuring Protocol Migration Processing

This command is to check the version globally or on individual port. For related information, refer to Compatibility of RSTP and STP.

Command	Function
Qtech# **clear spanning-tree detected-protocols**	Forcibly check the version on all ports.
Qtech# **clear spanning-tree detected-protocols interface** *interface-id*	Check the version forcibly on the port.

2.4.14 Configuring an MSTP Region

To deploy several devices in the same MSTP Region, you have to configure these devices with the same name, the same revision number, and the same Instance-VLAN table.

You can assign a VLAN to instances 0 to 64 respectively as required. The remaining VLANs will be automatically assigned to instance 0. One vlan can only be of an instance.

It is recommended to configure the Instance-VLAN table when the MSTP protocol is disabled. After configuration, you should enable the MSTP protocol again to ensure the stability and convergence of the network topology.

To configure an MSTP region, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree mst configuration**	Enter the MST configuration mode.

Qtech(config-mst)# **instance** *instance-id* **vlan** *vlan-range*	Add a VLAN group to a MST instance. *instance-id*: Instance ID ranging from 0 to 64. *vlan-range:* VLAN range in the range 1 to 4094. For instance: The **instance 1 vlan 2-200** command is to add VLAN 2-200 to instance 1. The **instance 1 vlan 2,20,200** command is to add VLAN 2, VLAN 20 and VLAN 200 to instance 1. You can use the **no** option of this command to delete a VLAN from an instance, and the deleted VLAN will be added to instance 0 automatically.
Qtech(config-mst)# **name** *name*	Specify the MST configuration name, a string of up to 32 bytes.
Qtech(config-mst)# **revision** *version*	Specify the MST revision number in the range 0 to 65535. The default value is 0.
Qtech(config-mst)# **show spanning-tree mst configuration**	Verify the configuration.
Qtech(config-mst)# **end**	Return to the privileged EXEC mode.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the MST region configuration to the default value, execute the **no spanning-tree mst configuration** command in the global configuration mode. You can use the **no instance** *instance-id* command to delete an instance. Similarly, the **no name** and **no revision** commands can be used to restore the MST name and MST revision number settings to the default value, respectively.

The following is the example of configuration:

```
Qtech(config)# spanning-tree mst configuration
Qtech(config-mst)# instance 1 vlan 10-20
Qtech(config-mst)# name region1
Qtech(config-mst)# revision 1
Qtech(config-mst)# show spanning-tree mst configuration
Multi spanning tree protocol : Enable Name [region1]
Revision 1
Instance Vlans Mapped
-------- ---------------------
0 1-9,21-4094
1 10-20
-------------------------------
Qtech(config-mst)# exit
Qtech(config)#
```

Caution

Before configuring vlan and instance mapping relationship, please ensure that all configured VLANs have been created. Otherwise, the association of vlan and instance on part of the products may be failed.

2.4.15 Configuring Maximum-Hop Count

Maximum-Hop Count means how many devices the BPDU message will pass through in a MSTP region before being discarded. This parameter takes effect for all instances.

To configure the Maximum-Hop Count, execute the following commands in the global configuraiton mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree max-hops** *hop-count*	Configure the Maximum-Hop Count ranging from 1 to 40, 20 by default.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the Maxium-Hop Count to the default value, execute the **no spanning-tree max-hops** command in the global configuration mode.

2.4.16 Configuring Intereface Compatibility Mode

In interface compatibility mode, when a port sends BPDU, it will carry different MSTI information according to the current port attribute to realize interconnection with other vendors.

To configure the interface compatibility mode, execute the following commands in the interface configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)#**interface** *interface-id*	Enter the Interface configuration mode.
Qtech(config-if)# **spanning-tree compatible enable**	Enable interface compatibility mode.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Check configuration items.
Qtech# **copy running-config startup-config**	Save the configuration.

To remove the settings, you can execute command **no spanning-tree compatible enable.**

2.5 Configuring Optional MSTP Features

2.5.1 Default Setting of Optional Spanning Tree Features

All the optional features are disabled by default, except for AutoEdge function.

2.5.2 Enabling Port Fast

Enabling Port Fast lets a port directly forward the BPDU message. When Port Fast is disabled due to the receipt of the BPDU message, the port will participate in the STP algorithm and forward the BPDU message normally.

To enable Port Fast, execute the following commands in the global configuraiton mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. A legal interface contains a physical port and an Aggregate Link.
Qtech(config-if)# **spanning-tree Portfast**	Enable Port Fast on the interface.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show spanning-tree interface** *interface-id* **portfast**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To disable Port Fast, execute the **spanning-tree portfast disable** command in the interface configuration mode.

You can use the **spanning-tree portfast default** command in the global configuration mode to enable Port Fast on all ports.

2.5.3 Enabling BPDU Guard

After BPDU Guard is enabled, a port will in the error-disabled status after receiving the BPDU packet.

To configure the BPDU guard, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree portfast Bpduguard default**	Enable the BPDU Guard globally.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. A legal interface contains a physical port and an aggregate link.
Qtech(config-if)# **spanning-tree portfast**	Enable Port Fast on the interface before the bpduguard configuration takes effect globally.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To disable BPDU Guard, execute the **no spanning-tree portfast bpduguard default** command in the global configuration command.

To enable or disable BPDU Guard on an interface, execute the **spanning-tree bpduguard enable** command or the **spanning-tree bpduguard disable** command on the interface respectively.

2.5.4 Enabling BPDU Filter

A port neither transmit nor receive the BPDU message after the BPDU filter is enabled.

To configure the BPDU Filter, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree portfast bpdufilter default**	Enable BPDU filter globally.
Qtech(config)# **interface** *Interface-id*	Enter the interface configuration mode. A legal interface contains a physical port and an aggregate link.
Qtech(config-if)# **spanning-tree Portfast**	Enable portfast on this interface before the bpduguard configuration takes effect globally.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To restore the default setting, execute the **no spanning-tree portfast bpdufilter default** command in the global configuration mode.

To enable or disable BPDU Filter on an interface, execute the **spanning-tree bpdufilter enable** command or the **spanning-tree bpdufilter disable** command in the interface configuration mode.

2.5.5 Enabling Tc_Protection

To configure Tc_Protection, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree tc-protection**	Enable Tc-Protection

Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To disable Tc_Protection, execute the **no spanning-tree tc-protection** command in the global configuration mode.

2.5.6 Enabling TC Guard

To enable TC Guard globally, execute the following commands in the global configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree tc-protection tc-guard**	Enable TC Guard globally.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To configure TC Guard on an interface, execute the following commands in the interface configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *Interface-id*	Enter the interface configuration mode. A legal interface includes a physical port and an aggregate link.
Qtech(config-if)# **spanning-tree tc-guard**	Enable TC Guard on this interface.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

2.5.7 Enabling TC Filtering

To enable TC filtering, execute the following commands:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter interface configuration mode. The valid interface includes physical port and Aggregate Link.
Qtech(config)# **spanning-tree ignore tc**	Enable TC filtering for this interface.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To disable TC filtering, use the **no spanning-tree ignore tc** command in interface configuration mode.

2.5.8 Enabling BPDU Source MAC check

After the BPDU source MAC check is enabled, the switch accepts only the BPDU message from the specified MAC address.

To configure the BPDU source MAC check, execute the following commands in the interface configuration mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. A legal interface includes a physical port and an aggregate link.
Qtech(config-if)#**bpdu src-mac-check** H.H.H	Enable the BPDU source MAC address check function on the interface.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.

Qtech# **copy running-config startup-config**	Save the configuration.

To disable BPDU source MAC check, execute the **no bpdu src-mac-check** command in the interface mode.

2.5.9 Enabling Root Guard

To configure interface ROOT Guard, execute the following commands in the privileged mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. Valid interface includes physical port and Aggregate Link.
Qtech(config-if)# **spanning-tree guard root**	Enable interface ROOT Guard.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

2.5.10 Enabling Loop Guard

To configure global LOOP Guard, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **spanning-tree loopguard default**	Enable global LOOP Guard.
Qtech(config)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

To configure interface LOOP Guard, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. Valid interface includes physical port and Aggregate Link.
Qtech(config-if)# **spanning-tree guard loop**	Enable interface Loop Guard.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

2.5.11 Disabling Interface Guard

To disable interface ROOT or LOOP Guard, execute the following commands in the privileged EXEC mode:

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **interface** *interface-id*	Enter the interface configuration mode. Valid interface includes physical port and Aggregate Link.
Qtech(config-if)# **spanning-tree guard none**	Disable interface Loop Guard.
Qtech(config-if)# **end**	Return to the privileged EXEC mode.
Qtech# **show running-config**	Verify the configuration.
Qtech# **copy running-config startup-config**	Save the configuration.

2.6 Showing MSTP Configuration and Status

You can use the following show commands to view the configuration of MSTP:

Command	Meaning
Qtech# **show spanning-tree**	Display the information on the parameters and topology of MSTP.
Qtech# **show spanning-tree summary**	Display the information on various instances and port forwarding status of MSTP.
Qtech# **show spanning-tree inconsistentports**	Display the block port due to root guard or loop guard.
Qtech# **show spanning-tree mst configuration**	Display the configuration information of the MST region.
Qtech# **show spanning-tree mst** *instance-id*	Display the MSTP information of an instance.
Qtech# **show spanning-tree mst** *instance-id* **interface** *interface-id*	Display the MSTP information of the specified instance of the interface.
show spanning-tree mst *instance-id* **topochange record**	Display the STP topology change record.
Qtech# **show spanning-tree interface** *interface-id*	Display the MSTP information of all the instances of the interface.
Qtech# **show spanning-tree forward-time**	Display forward-time.
Qtech# **show spanning-tree Hello Time**	Display Hello time.
Qtech# **show spanning-tree max-hops**	Display max-hops.
Qtech# **show spanning-tree tx-hold-count**	Display tx-hold-count.
Qtech# **show spanning-tree pathcost Method**	Display pathcost method.

3 CONFIGURING LLDP

3.1 LLDP Overview

Drafted by IEEE 802.1AB, LLDP (Link Layer Discovery Protocol) can detect network topology change and identify what the change is. With LLDP, a device sends local device information as TLV (Type, Length and Value) triplets in LLDP Data Units (LLDPDUs) to the neighbor devices, and at the same time, stores the device information received in LLDPDUs sent from the LLDP neighbors in a standard management information base (MIB) to be accessed by the network management system.

Through LLDP, the network management system can learn about the state of topological connections, such as which ports of the device are connected to other devices, the rate of ports on both sides of link, and whether the duplex mode is matched. The network administrator can quickly locate and eliminate faults according to such information.

3.1.1 Basic Concepts

LLDPDU

LLDPDU refers to the data units encapsulated in LLDP packets, and comprises multiple TLV sequences, including three fixed TLVs, a number of optional TLVs and an End of TLV. The detailed format of LLDPDU is shown in Fig 1:

Fig 1-1 LLDPDU format

Chassis ID TLV	Port ID TLV	Time To Live TLV	Optional TLV	…	Optional TLV	End Of LLDPDU TLV
M	M	M				M

- * M refers to fixed TLV.
- In LLDPDU, Chassis ID TLV, Port ID TLV, Time To Live TLV and End Of LLDPDU TLV are fixed TLVs, while other TLVs are optional.

LLDPDU Encapsulation Format

LLDP packet supports two encapsulation formats: Ethernet II and SNAP (Subnetwork Access Protocols).

Ethernet II encapsulated LLDPDU format is shown in Fig 2:

Figure 1-2 Ethernet II encapsulated LLDPDU format

Destination Address	Source Address	Ethertype	LLDPDU	FCS

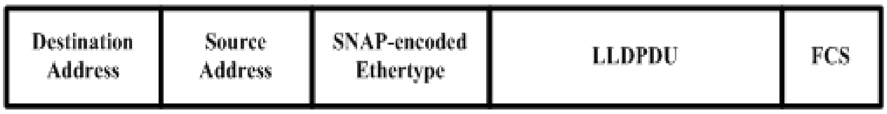
Specifically:

- Destination Address: destination MAC address. It is fixed to 01-80-C2-00-00-0E, a multicast address.
- Source Address: source MAC address, layer-2 MAC address of device.
- Ethertype: the Ethernet type, 0x88CC.
- LLDPDU: LLDP Data Unit.

- FCS: frame check sequence.

SNAP-encapsulated LLDPDU format is shown in Fig 3:

Figure 1-3 SNAP-encapsulated LLDPDU format



Specifically:

- Destination Address: destination MAC address. It is fixed to 01-80-C2-00-00-0E, a multicast address.
- Source Address: source MAC address, layer-2 MAC address of device.
- SNAP-encoded Ethertype: SNAP-encapsulated Ethernet type, AA-AA-03-00-00-00-88-CC.
- LLDPDU: LLDP Data Unit.
- FCS: frame check sequence.

TLV

TLVs encapsulated in LLDPDU can fall into two broad categories:

- Basic management TLVs
- Organizationally specific TLVs

Basic management TLVs are a group of basic TLVs for network management. The organizationally specific TLVs are TLVs defined by standards organizations and other organizations, such as IEEE 802.1, IEEE 802.3 and etc.

1) Basic management TLVs

Basic management TLVs include two types of TLVs: fixed TLVs and optional TLVs. Fixed TLVs must be included in LLDPDU, while optional TLVs can be included or excluded according to need.

Basic management TLVs are shown in Table 1:

Type	Description	Use in LLDPDU
End Of LLDPDU TLV	End mark of LLDPDU, occupying 2 bytes	Fixed
Chassis ID TLV	Used to identify the device, and is generally represented with MAC address	Fixed
Port ID TLV	ID of the LLDPDU sending port	Fixed
Time To Live TLV	Life of local information on the neighbor device. When TLV with 0 TTL is received, the corresponding neighbor information must be deleted.	Fixed
Port Description TLV	Port description of LLDPDU sending port	Optional

System Name TLV	Name of the sending device	Optional
System Description TLV	Description of the sending device, including hardware/software version, operating system and etc.	Optional
System Capabilities TLV	Identifies the primary functions of the sending device, such as bridging, routing and relaying.	Optional
Management Address TLV	Management address, including interface number and OID (object Identifier).	Optional

Table 1 Basic management TLV

☑ Basic management TLVs are supported by the LLDP protocol used by Qtech switch products.

2) Organizationally specific TLVs

Different organizations (such as IEEE 802.1, IEEE 802.3, IETF or device suppliers) may define specific TLVs to advertise specific information about the device, and OUI (Organizationally Unique Identifier) is used to identify different organizations.

Organizationally specific TLVs are optional TLVs advertised in LLDPDU according to user's actual needs. Currently, commonly found organizationally specific TLVs include:

1) IEEE 802.1 organizationally specific TLVs

IEEE 802.1 organizationally specific TLVs are shown in Table 2:

Type	Description
Port VLAN ID TLV	VLAN identifier of the sending port
Port And Protocol VLAN ID TLV	Protocol VLAN identifier of the sending port
VLAN Name TLV	Name of VLAN with which the device is configured
Protocol Identity TLV	Protocols supported by the port

Table 2 IEEE 802.1 organizationally specific TLVs

☑ LLDP protocol used by Qtech switch products doesn't support the sending of Protocol Identity TLV, but allows the reception of such TLV.

2) IEEE 802.3 organizationally specific TLVs

IEEE 802.3 organizationally specific TLVs are shown in Table 3:

Type	Description
MAC/PHY Configuration/Status TLV	The bit-rate and duplex capabilities of the sending port and support for auto negotiation.
Power Via MDI TLV	Power supply capability of the port
Link Aggregation TLV	Indicate the link aggregation capability of the port and the aggregation status.
Maximum Frame Size TLV	The maximum frame size supported by the port.

Table 3 IEEE 802.3 organizationally specific TLVs

☑ IEEE 802.3 organizationally specific TLVs are supported by the LLDP protocol used by Qtech switch products.

3) LLDP-MED TLVs

LLDP-MED is the extension of IEEE 802.1AB LLDP protocol, so that the user can conveniently deploy VoIP (Voice Over IP) network and fault detection. It provides multiple applications such as network policy configuration, device detection, PoE management and directory management, providing a cost-effective and easy-to-use solution for deploying voice devices in Ethernet.

LLDP-MED TLVs are shown in Table 4:

Type	Description
LLDP-MED Capabilities TLV	Whether the device supports LLDP-MED, the type of LLDP-MED TLV encapsulated in LLDPDU, and the type of current device (network connection device or endpoint)
Network Policy TLV	Advertise VLAN configuration of the specific port, supported applications (voice and video, for example), and the Layer 2 priorities.
Location Identification TLV	Location identifier information for an endpoint, used to accurately locate the endpoint in applications such as network topology collection.
Extended Power-via-MDI TLV	Provide more advanced power supply management.
Inventory – Hardware Revision TLV	Hardware version of MED device
Inventory – Firmware Revision TLV	Firmware version of MED device
Inventory – Software Revision TLV	Software version of MED device

Inventory – Serial Number TLV	Serial number of MED device
Inventory – Manufacturer Name TLV	Vendor name of MED device
Inventory – Model Name TLV	Model name of MED device
Inventory – Asset ID TLV	Asset ID of MED device, used for directory management and asset tracking.

Table 4 LLDP-MED TLVs

☑ LLDP-MED TLVs are supported by the LLDP protocol used by Qtech switch products.

3.1.2 Working Principles

Operating Modes of LLDP

LLDP provides three operating modes:

- TxRx: sending and receiving LLDPDUs.
- Rx Only: only sending LLDPDUs.
- Tx Only: only receiving LLDPDUs.

When the LLDP operating mode of a port changes, the port will initialize the protocol state machine. To prevent LLDP from being initialized too frequently during times of frequent operating mode change, you can configure a re-initialization delay.

Mechanism for Transmitting LLDPDUs

An LLDP-enabled port operating in TxRx mode or Tx Only mode will send LLDPDUs both periodically and when the local device information changes. To avoid frequent LLDPDU sending during times of frequent local device information change, an interval is introduced between two successive LLDPDUs. This interval can be configured manually.

LLDP provides two types of packets:

- Standard LLDPDUs: including the management and configuration information about local device.
- Shutdown LLDPDU: When LLDP sending mode is disabled or when the port is administratively shut down, shutdown LLDPDU will be sent. Shutdown LLDPDU generally comprises Chassis ID TLV, Port ID TLV, Time To Live TLV and End Of LLDP TLV, with the TTL in Time To Live TLV being 0. When the device receives shutdown LLDPDUs, it will consider the neighbor on longer available and delete neighbor information.

When LLDP operating mode changes from shutdown or Rx to TxRx or Tx, or when a new neighbor is detected (namely new LLDPDUs are received and no such neighbor information is stored locally), to allow the neighbor device to quickly study the information about this device, the fast sending mechanism will be initiated. The fast sending mechanism adjusts the LLDPDU sending interval to 1 second and continuously transmits a certain number of LLDPDUs.

Mechanism for Receiving LLDPDUs

A LLDP-enabled port operating in TxRx mode or Rx Only mode will be able to receive LLDPDUs, and will check the validity of received LLDPDUs to verify they are new neighbor information or updates of existing neighbor information.

The neighbor information will be stored on the local device. Meanwhile, an aging timer will be set according to the value in TTL TLV carried in the LLDPDU. If the TTL value is zero, the information is aged out immediately.

3.1.3 Protocol Specifications

The protocols and standards related to LLDP include:

- IEEE 802.1AB 2005: Station and Media Access Control Connectivity Discovery
- ANSI/TIA-1057: Link Layer Discovery Protocol for Media Endpoint Devices

3.2 Configuring LLDP Basic Functions

Function	Default setting
Globally enable LLDP	Enabled
Enable LLDP on the port	Enabled
Operating mode of LLDP	TxRx
Port re-initialization delay	2 seconds
LLDPDU transmit interval	30 seconds
LLDPDU transmit delay	2 seconds
Neighbor information aging timer	120 seconds
LLDPDU encapsulation format	Ethernet II
Enable LLDP Trap	Disabled
LLDP error detection	Enabled

3.2.1 Enabling LLDP

By default, LLDP is enabled globally and on each port. To make LLDP take effect on certain ports, you must enable LLDP both globally and on these ports.

Execute the following steps to disable LLDP globally and on each port.

Command	Function

Qtech(config)#**no lldp enable**	Disable LLDP globally.
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.
Qtech(config-if)#**no lldp enable**	Disable LLDP on the interface.
Qtech(config-if)#**show lldp status**	Display LLDP state.

To enable LLDP globally or on the port, execute "lldp enable" command.

Caution

Disabling the LLDP globally will disable LLDP on the device. Meanwhile, the device will send Shutdown LLDPDUs to neighbor devices in order to delete the corresponding LLDP information.

Note

The port can learn up to 5 neighbors.
If a neighbor device does not support the LLDP, but its downlink device does, the information of non-directly connected devices may be learnt on the port as the neighbor device may forward the LLDP packets.

Configuration example:

# Globally disable LLDP and display LLDP state.

```
Qtech(config)#no lldp enable

Qtech(config)#show lldp status

Global status of LLDP: Disable
```

3.2.2 Configuring LLDP Operating Mode

By default, LLDP is enabled on the interface and operates in TxRx mode. The user can change the operating mode to Tx mode or Rx mode as needed. Execute the following steps to configure LLDP operating mode.

Command	**Function**
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.
Qtech(config-if)#**lldp mode** { **tx** | **rx** | **txrx** }	Configure LLDP operating mode. The configurable operating

	modes include Tx, Rx and TxRx.
Qtech(config-if)#**show lldp status interface** *interface-name*	Display LLDP state on the interface.

Configuration example:

# Configure LLDP operating mode as Tx on the interface and display LLDP state on the interface

```
Qtech(config)#interface gigabitethernet 0/1

Qtech(config-if)#lldp mode tx

Qtech(config-if)#show lldp status interface gigabitethernet 0/1

Port [GigabitEthernet 0/1]

Port status of LLDP                : Enable

Port state                         : UP

Port encapsulation                 : Ethernet II

Operational mode                   : TxOnly

Notification enable                : NO

Error detect enable                : YES

Number of neighbors                : 0

Number of MED neighbors            : 0
```

3.2.3 Configuring the Advertisable TLVs

By default, all TLVs other than Location Identification TLV can be advertised on the interface. Execute the following steps to configure advertisable TLVs on the interface.

Command	**Function**
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.
Qtech(config-if)# **lldp tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** } | **dot1-tlv** { **all** | **port-vlan-id** | **protocol-vlan-id** [*vlan-id*] | **vlan-name** [*vlan-id*] } | **dot3-tlv** { **all** | **link-**	Configure the TLV types that the interface allows the port to advertise. By default, all TLVs other than Location Identification TLV can be advertised on the interface.

aggregation | **mac-physic** | **max-frame-size** | **power** } | **med-tlv** { **all** | **capability** | **inventory** | **location** { **civic-location** | **elin** } **identifier** *id* | **network-policy profile** [*profile-num*] | **power-over-ethernet** } }	
Qtech(config-if)# **show lldp tlv-config interface** *interface-name*	Display the attributes of advertisable TLVs.

Note

When configuring basic management TLVs, IEEE 802.1 organizationally specific TLVs and IEEE 802.3 organizationally specific TLVs, if "all" parameter is specified, all corresponding optional TLVs will be advertised.

When configuring LLDP-MED TLVs, if "all" parameter is specified, all LLDP-MED TLVs other than Location Identification TLV will be advertised.

Configure to allow the advertisement of LLDP-MED MAC/PHY TLVs before that of LLDP-MED Capability TLVs.

Configure to cancel the advertisement of LLDP-MED Capability TLVs before that of LLDP-MED MAC/PHY TLVs.

When configuring LLDP-MED TLVs, the LLDP-MED Capability TLV shall be configured as advertisable in order to further configure other LLDP-MED TLVs as advertisable.

In order not to advertise LLDP-MED Capability TLV, other LLDP-MED TLVs shall be configured as non-advertisable, so that LLDP-MED TLVs are not advertised.

For the meaning of respective key words of "lldp tlv-enable", please refer to the descriptions given in "LLDP-CREF".

When associating the device with an IP phone, you can configure the network policy TLV delivery policy to the IP phone if it supports LLDP-MED. Then, the IP phone modifies the voice flow tag and QoS. At this time, the voice VLAN function is not required, but it is required to configure the port connecting to the IP phone as the QoS trusted port. If the IP phone does not support LLDP-MED, the voice VLAN configuration is required and the phone MAC address must be manually configured to the voice VLAN OUI list.

Configuration example:

# Configure to disable the advertisement of Port And Protocol VLAN ID TLV specified by IEEE 802.1.

```
Qtech(config)#interface gigabitethernet 0/1

Qtech(config-if)#no lldp tlv-enable dot1-tlv protocol-vlan-id

Qtech(config-if)#show lldp tlv-config interface gigabitethernet 0/1

LLDP tlv-config of port [GigabitEthernet 0/1]

               NAME                    STATUS  DEFAULT

------------------------------ ------ -------

Basic optional TLV:
```

```
Port Description TLV              YES    YES
System Name TLV                    YES    YES
System Description TLV            YES    YES
System Capabilities TLV           YES    YES
Management Address TLV            YES    YES


IEEE 802.1 extend TLV:
Port VLAN ID TLV                   YES    YES
Port And Protocol VLAN ID TLV    NO     YES
VLAN Name TLV                       YES    YES


IEEE 802.3 extend TLV:
MAC-Physic TLV                      YES    YES
Power via MDI TLV                  YES    YES
Link Aggregation TLV               YES    YES
Maximum Frame Size TLV             YES    YES


LLDP-MED extend TLV:
Capabilities TLV                    YES    YES
Network Policy TLV                  YES    YES
Location Identification TLV       NO     NO
Extended Power via MDI TLV        YES    YES
Inventory TLV                        YES    YES
```

3.2.4 Configuring the Management address Advertised in LLDPDU

The management address of a device is used by the network management system to identify and manage the device.

Execute the following steps to configure the management address to be advertised in LLDPDU:

Command	Function
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.
Qtech(config-if)#**lldp management-address-tlv** [*ip-address*]	Configure the management address advertised in the LLDP packet.
Qtech(config-if)#**show lldp local-information interface** *interface-name*	Display LLDP local information about a specific interface.

Note

By default, the management address is advertised in LLDPDU, and is the IPv4 address of the lowest-ID VLAN carried on the port. If IPv4 address is not configured for this VLAN, the next lowest-ID VLAN carried on the port will be tried until the IPv4 address is obtained. If the IPv4 address is still not found, the IPv6 address of the lowest-ID VLAN carried on the port will be tried. If the IPv6 address is still not found, the MAC address of the device will be advertised as the management address.

Configuration example:

# Configure the management address advertised in LLDPDU as 192.168.1.1 and display the corresponding configuration.

```
Qtech(config)#interface gigabitethernet 0/1
Qtech(config-if)#lldp management-address-tlv 192.168.1.1
Qtech(config-if)#show  lldp local-information interface GigabitEthernet 0/1
Lldp local-information of port [GigabitEthernet 0/1]
    Port ID type                      : Interface name
  Port id                           : GigabitEthernet 0/1
  Port description                  :

  Management address subtype        : ipv4
  Management address                : 192.168.1.1
  Interface numbering subtype       : ifIndex
  Interface number                  : 0

  Object identifier                 :



  802.1 organizationally information

  Port VLAN ID                      : 1

  Port and protocol VLAN ID(PPVID)  : 1

      PPVID Supported               : YES

      PPVID Enabled                 : NO

  VLAN name of VLAN 1               : VLAN0001

  Protocol Identity                 :



  802.3 organizationally information

  Auto-negotiation supported        : YES

  Auto-negotiation enabled          : YES

  PMD auto-negotiation advertised   : 1000BASE-T full duplex mode, 100BASE-TX full
duplex mode, 100BASE-TX half duplex mode, 10BASE-T full duplex mode, 10BASE-T half
duplex mode

  Operational MAU type              : dot3MauType100BaseTXFD: 2 pair category 5 UTP,
full duplex mode

  PoE support                       : NO

 Link aggregation supported        : YES

  Link aggregation enabled          : NO

  Aggregation port ID               : 0

  Maximum frame Size                : 1500



  LLDP-MED organizationally information

  Power-via-MDI device type         : PD
  Power-via-MDI power source          : Local
  Power-via-MDI power priority        :
  Power-via-MDI power value           :
  Model name                          : Model name
```

3.2.5 Configuring the Number of Fast Sent LLDPDUs

When a new neighbor is detected or when LLDP operating mode changes from shutdown or Rx to TxRx or Tx, to allow the neighbor device to quickly study the information about this device, the fast sending mechanism will be initiated. The fast sending mechanism shortens the LLDPDU sending interval to 1 second and continuously transmits a certain number of LLDPDUs before restoring to the normal transmit interval.

Command	Function
Qtech(config)#**lldp fast-count** *count*	Configure the number of fast sent LLDPDUs in the range from 1 to 10. The default is 3.
Qtech(config-if)#**show lldp status**	Display LLDP state.

Configuration example:

# Configure the number of fast sent LLDPDUs to 5.

```
Qtech(config)#lldp fast-count 5
Qtech(config)#show lldp status
Global status of LLDP                  : Enable
Neighbor information last changed time :
Transmit interval                      : 30s
Hold multiplier                        : 4
Reinit delay                           : 2s
Transmit delay                         : 2s
Notification interval                  : 5s
Fast start counts                      : 5
```

3.2.6 Configuring TTL Multiplier and LLDPDU Transmit interval

The value of Time To Live TLV in LLDPDU = TTL multiplier × LLDPDU transmit interval + 1. Therefore, the TTL of local device information on the neighbor device can be controlled by adjusting TTL multiplier.

The LLDPDU transmit interval can be adjusted. Execute the following steps to configure TTL multiplier and LLDPDU transmit interval.

Command	Function
Qtech(config)#**lldp hold-multiplier** *value*	Configure TTL multiplier in the range from 2 to 10. The default is 4.
Qtech(config)#**lldp timer tx-interval** *seconds*	Configure LLDPDU transmit interval in the range from 5 to 32768 in the unit of seconds. The default is 30.
Qtech(config-if)#**show lldp status**	Display LLDP state.

Configuration example:

# Configure TTL multiplier to 3 and LLDPDU transmit interval to 20 seconds. By this time, the TTL of local device information on the neighbor device is 61 seconds.

```
Qtech(config)#lldp hold-multiplier 3

Qtech(config)#lldp timer tx-interval 20

Qtech(config)#show lldp status

Global status of LLDP                : Enable

Neighbor information last changed time :

Transmit interval                    : 20s

Hold multiplier                      : 3

Reinit delay                         : 2s

Transmit delay                       : 2s

Notification interval                : 5s

Fast start counts                    : 3
```

3.2.7 Configuring LLDPDU Transmit Delay

An LLDP-enabled port will send LLDPDUs when the local device information changes. To avoid frequent LLDPDU sending during times of frequent local device information change, we can configure LLDPDU transmit delay to control the frequent transmission of LLDPDUs. The default transmit delay is 2 seconds. Execute the following steps to configure the LLDPDU transmit delay.

Command	Function
Qtech(config)#**lldp timer tx-delay** *seconds*	Configure LLDPDU transmit delay

Qtech(config)#**show lldp status**	Display LLDP state.

Configuration example:

# Configure LLDPDU transmit delay to 3 seconds and display LLDP state.

```
Qtech(config)#lldp timer tx-delay 3

Qtech(config)#show lldp status

Global status of LLDP                   : Enable

Neighbor information last changed time :

Transmit interval                       : 30s

Hold multiplier                         : 4

Reinit delay                            : 2s

Transmit delay                          : 3s

Notification interval                   : 5s

Fast start counts                       : 3
```

3.2.8 Configuring Port Re-initialization Delay

When the LLDP operating mode of a port changes, the port will initialize the protocol state machine. To prevent LLDP from being initialized too frequently during times of frequent operating mode change, you can configure port re-initialization delay. Execute the following steps to configure port re-initialization delay:

Command	Function
Qtech(config)#**lldp timer reinit-delay** *seconds*	Configure port re-initialization delay.
Qtech(config)#**show lldp status**	Display LLDP state.

Configuration example:

# Configure the port re-initialization delay to 3 seconds and display LLDP state.

```
Qtech(config)#lldp timer reinit-delay 3

Qtech(config)#show lldp status

Global status of LLDP                   : Enable

Neighbor information last changed time :
```

```
Transmit interval                       : 30s
Hold multiplier                         : 4
Reinit delay                            : 3s
Transmit delay                          : 2s
Notification interval                   : 5s
Fast start counts                       : 3
```

3.2.9 Configuring LLDP Trap

By configuring LLDP Trap, the LLDP information of local device (such as information about the detection of new neighbor or the fault on the communication link) can be sent to the network management server. The administrator can monitor the network operation status according to such information.

To prevent excessive LLDP traps from being sent, you can set an interval for sending LLDP Traps. If LLDP information change is detected during this interval, traps will be sent to the network management server.

By default, LLDP Trap is disabled.

Execute the following steps to configure LLDP Trap:

Command	Function
Qtech(config)#**lldp timer notification-interval** *seconds*	Configure the interval for sending LLDP Traps in the range from 5 to 3600 in the unit of seconds. The default is 5.
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.
Qtech(config-if)#**lldp notification remote-change enable**	Enable LLDP Trap. LLDP Trap is disabled by default.
Qtech(config-if)#**show lldp status**	Display LLDP state.

Configuration example:

# Enable LLDP Trap and configure the interval for sending LLDP Traps to 10 seconds.

```
Qtech(config)#lldp timer notification-interval 10
Qtech(config)#interface gigabitethernet 0/1
Qtech(config-if)#lldp notification remote-change enable
Qtech(config-if)#show lldp status
Global status of LLDP                   : Enable
```

```
Neighbor information last changed time :

Transmit interval                       : 30s

Hold multiplier                         : 4

Reinit delay                            : 2s

Transmit delay                          : 2s

Notification interval                   : 10s

Fast start counts                       : 3

------------------------------------------------------------

Port [GigabitEthernet 0/1]

------------------------------------------------------------

Port status of LLDP              : Enable

Port state                       : UP

Port encapsulation               : Ethernet II

Operational mode                 : RxAndTx

Notification enable              : YES

Error detect enable              : YES

Number of neighbors              : 0

Number of MED neighbors          : 0
```

3.2.10 Configuring LLDP Error Detection

Configure LLDP error detection, including the detection of VLAN configurations on both sides of the link, port state detection, port aggregation configuration detection, MTU configuration detection and loop detection. If any error is detected by LLDP, LOG information will be printed to notify the administrator.

Execute the following steps to configure LLDP error detection:

Command	Function
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.

Qtech(config-if)#**lldp error-detect**	Configure LLDP error detection. LLDP error detection is enabled by default.
Qtech(config-if)#**show lldp status interface** *interface-name*	Display LLDP state on the interface.

Configuration example:

# Configure LLDP error detection.

```
Qtech(config)#interface gigabitethernet 0/1

Qtech(config-if)#lldp error-detect

Qtech(config-if)#show lldp status interface gigabitethernet 0/1

Port [GigabitEthernet 0/1]

Port status of LLDP              : Enable

Port state                       : UP

Port encapsulation               : Ethernet II

Operational mode                 : RxAndTx

Notification enable              : NO

Error detect enable              : YES

Number of neighbors              : 0

Number of MED neighbors          : 0
```

3.2.11 Configuring LLDPDU Encapsulation Format

By default, LLDPDUs are encapsulated in Ethernet II frames. The configurable encapsulation formats include Ethernet II and SNAP.

When configured to Ethernet II format, the device can only send and receive Ethernet II-encapsulated LLDP packets.

When configured to SNAP format, the device can only send and receive SNAP-encapsulated LLDP packets.

Execute the following steps to configure LLDPDU encapsulation format:

Command	Function
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode. LLDP runs on the actual physical interface (for AP port, it runs on AP's member port). LLDP is not supported on stacking port or VSL port.

Qtech(config-if)#**lldp encapsulation snap**	Configure LLDPDU encapsulation format to SNAP.
Qtech(config-if)#**show lldp status interface** *interface-name*	Display LLDP state on the interface.

Caution

To guarantee normal communication between local device and neighbor device, the same LLDPDU encapsulation format must be used.

Configuration example:

# Configure LLDPDU encapsulation format to SNAP and display the corresponding configuration.

```
Qtech(config)#interface gigabitethernet 0/1

Qtech(config-if)#lldp encapsulation snap

Qtech(config-if)#show lldp status interface gigabitethernet 0/1

Port [GigabitEthernet 0/1]

Port status of LLDP             : Enable

Port state                      : UP

Port encapsulation              : Snap

Operational mode                : RxAndTx

Notification enable             : NO

Error detect enable             : YES

Number of neighbors             : 0

Number of MED neighbors         : 0
```

3.2.12 Displaying and Clearing Configurations

Command	**Function**
Qtech(config)# **lldp network-policy profile** *profile-num*	Enter the LLDP network-policy configuration mode.
Qtech(config-lldp-network-policy)# { **voice** | **voice-signaling** } **vlan** { { *vlan-id* [**cos** ***cvalue*** | **dscp** *dvalue*] } | { **dot1p** [**cos** ***cvalue*** | **dscp** *dvalue*] } |	Configure the LLDP network-policy.

none | **untagged** } **no** { **voice** | **voice-signaling** } **vlan**	

Configuration example:

# Configure the LLDP packet advertised from interface 1 as follows:
Network Policy TLV: **1**
voice VLAN ID: **3**
cos: **4**
dscp: **6**

```
Qtech#config

Qtech(config)#lldp network-policy profile 1

Qtech(config-lldp-network-policy)# voice vlan 3 cos 4

Qtech(config-lldp-network-policy)# voice vlan 3 dscp 6

Qtech(config-lldp-network-policy)#exit

Qtech(config)# interface gigabitethernet 0/1

Qtech(config-if-GigabitEthernet 0/1)# lldp tlv-enable med-tlv network-policy profile 1
```

3.2.13 Configuring the Civic Address Information of a Device

Run the commands listed in the following table to configure the address information of a device.

Command	Function
Qtech(config)# **lldp location civic-location identifier** *id*	Enters the LLDP Civic Address configuration mode
Qtech(config-lldp-civic)**# device-type** *device-type*	Configure the device type. The default device type is a switch.
Qtech(config-lldp-civic)# **civic-location** { **country** | **state** | **county** | **city** | **division** | **neighborhood** | **street-group** | **leading-street-dir** | **trailing-street-suffix | street-suffix** | **number** | **street-number-suffix** | **landmark** | **additional-location-information** | **name** | **postal-code** | **building** | **unit** | **floor** | **room** | **type-of-place | postal-community-name** | **post-office-box** | **additional-code** } *ca-word*	Configure the LLDP civic address information.

Configuration example:

# Configure the address of device interface 1 as follows:
Device type: switch
Country: CH
City: Fuzhou
Postal-code: 350000

```
Qtech#config
```

```
Qtech(config)#lldp location civic-location identifier 1
Qtech(config-lldp-civic)# country CH
Qtech(config-lldp-civic)# city Fuzhou
Qtech(config-lldp-civic)# postal-code 350000
Qtech(config-lldp-civic)# exit
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-if-GigabitEthernet 0/1)# lldp tlv-enable location civic-location identifier 1
```

3.2.14 Configuring the Emergency Call Number

Run the commands listed in the following table to configure the emergency call number of a device.

Command	**Function**
Qtech(config)# **lldp location elin identifier** *id* **elin-location** *tel-number*	Configure the emergency call number.

Configuration example:

# Configure the emergency call number of device interface 1 as 085285555556.

```
Qtech#config
Qtech(config)#lldp location elin identifier 1 elin-location 085283671111
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-if-GigabitEthernet 0/1)# lldp tlv-enable location elin identifier 1
```

3.2.15 Viewing and Clearing Configurations

Command	**Function**
show lldp local-information [**global** | **interface** *interface-name*]	Show the device information to be sent to a neighbor.
show lldp location { **civic-location** | **elin** } { **identifier** ***id*** | **interface** *interface-name* | **static** }	Show the civic address information or emergency call number of a local device.
show lldp neighbors [**interface** *interface-name*] [**detail**]	Show the device information about an adjacent neighbor.

show lldp network-policy profile [*profile-num*]	Show the LLDP network-policy configuration.
show lldp statistics [**global** | **interface** *interface-name*]	Show the LLDP statistics.
show lldp status [**interface** *interface-name*]	Show the LLDP status.
show lldp tlv-config [**interface** *interface-name*]	Show the optional TLVs that can be advertised.
clear lldp statistics [**interface** *interface-name*]	Clear LLDP statistics.
clear lldp table [**interface** *interface-name*]	Clear the information about LLDP neighbors.

Configuration example:

# Show the device information about an adjacent neighbor connecting a specified port.

```
Qtech# show lldp neighbors detail
Lldp neighbor-information of port [GigabitEthernet 0/1]
  Neighbor index                    : 1
  Device type                       : LLDP Device
  Update time                       : 12minutes 40seconds
Aging time                          : 5seconds
  Chassis ID type                   : MAC address
  Chassis id                        : 00d0.f822.33cd
  System name                       : System name
  System description              : System description
  System capabilities supported     : Repeater, Bridge, Router
  System capabilities enabled       : Repeater, Bridge, Router
```

```
Management address subtype        : 802 mac address
Management address                : 00d0.f822.33cd
Interface numbering subtype       :
Interface number                  : 0
Object identifier                 :

LLDP-MED capabilities             :
Device class                      :
HardwareRev                       :
FirmwareRev                       :
SoftwareRev                       :
SerialNum                         :
Manufacturer name                 :
Asset tracking identifier         :

Port ID type                      : Interface name
Port id                           : GigabitEthernet 0/2
Port description                  :

802.1 organizationally information
Port VLAN ID                      : 1
Port and protocol VLAN ID(PPVID)  : 1
    PPVID Supported               : YES
    PPVID Enabled                 : NO
```

```
  VLAN name of VLAN 1                 : VLAN0001

  Protocol Identity                   :

  802.3 organizationally information

  Auto-negotiation supported          : YES

  Auto-negotiation enabled            : YES

  PMD auto-negotiation advertised     : 1000BASE-T full duplex mode, 100BASE-TX full
duplex mode, 100BASE-TX half duplex mode, 10BASE-T full duplex mode, 10BASE-T half
duplex mode

  Operational MAU type                : speed(100)/duplex(Full)

  PoE support                         : NO

  Link aggregation supported          : YES

  Link aggregation enabled            : NO

  Aggregation port ID                 : 0

  Maximum frame Size                  : 1500



  LLDP-MED organizationally information

  Power-via-MDI device type           :

  Power-via-MDI power source          :

  Power-via-MDI power priority        :

  Power-via-MDI power value           :
```

Note

For details about LLDP output information, see the description in *LLDP Command Reference*.

3.3 Typical LLDP Configuration Examples

3.3.1 Use LLDP to View Topological Connections

Networking Requirements

Devices required

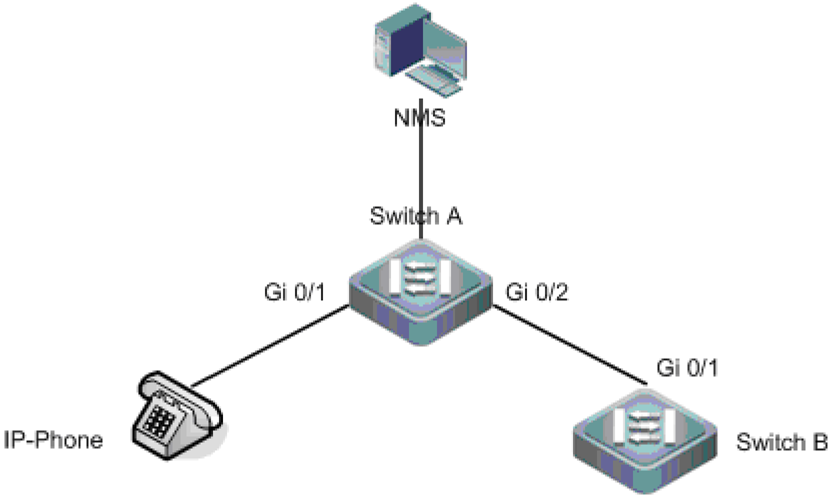
Two Ethernet switches (Switch A and Switch B), one MED device (taking IP Phone as the example) and one NMS (Network management System).

Configuration required

LLDP is enabled by default. No further configuration is needed.

Network Tpology

Fig 4 Basic topological diagram of LLDP



Configuration Tips

- LLDP operating mode on the port is TxRx.
- LLDPDU transmit times will use default values, namely LLDPDU transmit interval is 30 seconds and LLDPDU transmit delay is 2 seconds.

Configuration Steps

By default, LLDP is enabled, and no further configuration is needed.

Verification

- Display the information about the neighbor device connecting with Switch A.

# Display the information about the neighbor device on Switch A.

```
Qtech# show lldp neighbors gigabitethernet 0/2

Capability codes:

    (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device

    (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other



Local Intf  Port ID  Capability   Aging-time

Gi 0/2       Gi 0/1   B,R           120
```

```
Total entries displayed: 1
```

The above messages show that the MAC address of neighbor device connected to port 2 of switch A is 00d0-f822-33cd and the port connected is Gi 0/1. The neighbor device allows bridging and routing.

# Display the detailed information about the neighbor device connected to port Gi 0/2 of Switch A.

```
Qtech# show lldp neighbor-information interface gigabitethernet 0/2

Lldp neighbor-information of port [GigabitEthernet 0/2]

  Neighbor index                    : 1

  Device type                       : LLDP Device

  Update time                       : 5minute 39second



  Chassis ID type                   : MAC address

  Chassis id                        : 00d0.f822.33cd

  System name                       : System name

  System description                : System description

  System capabilities supported     : Repeater, Bridge, Router

  System capabilities enabled       : Repeater, Bridge, Router



  Management address subtype        : 802 mac address

  Management address                : 00d0.f822.33cd

  Interface numbering subtype       :

  Interface number                  : 0

  Object identifier                 :





  LLDP-MED capabilities             :

  Device class                      :
```

```
  HardwareRev                       :
  FirmwareRev                       :
  SoftwareRev                       :
  SerialNum                         :
  Manufacturer name                 :
  Asset tracking identifier         :

  Port ID type                      : Interface name
  Port id                           : GigabitEthernet 0/1
  Port description                  :

  802.1 organizationally information
  Port VLAN ID                      : 1
  Port and protocol VLAN ID(PPVID)  : 1
      PPVID Supported               : YES
      PPVID Enabled                 : NO
  VLAN name of VLAN 1               : VLAN0001
  Protocol Identity                 :


  802.3 organizationally information
  Auto-negotiation supported        : YES
  Auto-negotiation enabled          : YES
  PMD auto-negotiation advertised   : 1000BASE-T full duplex mode, 100BASE-TX full
duplex mode, 100BASE-TX half duplex mode, 10BASE-T full duplex mode, 10BASE-T half
duplex mode
```

```
  Operational MAU type              : dot3MauType1000BaseTFD: Four-pair Category 5
UTP, full duplex mode

  PoE support                       : NO

  Link aggregation supported        : YES

  Link aggregation enabled          : NO

  Aggregation port ID               : 0

  Maximum frame Size                : 1500



  LLDP-MED organizationally information

  Power-via-MDI device type         :

  Power-via-MDI power source        :

  Power-via-MDI power priority      :

              Power-via-MDI power value         :
```

3.3.2 Use LLDP Error Detection Feature to Perform Error Detection

Networking Requirements

- Devices required

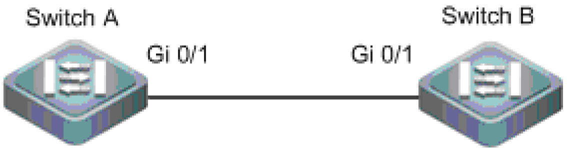
two Ethernet switches (Switch A and Switch B)

- Configuration required

LLDP is enabled by default. No further configuration is needed.

Network Topology

Fig 5 Basic topological diagram of LLDP



Configuration Tips

- LLDP operating mode on the port is TxRx.
- LLDPDU transmit times will use default values, namely LLDPDU transmit interval is 30 seconds and LLDPDU transmit delay is 2 seconds.
- LLDP error detection is enabled by default. No further configuration is needed.

Configuration Steps

1. Configure the bit-rate of port Gi 0/1 of Switch A to 100M.

```
Qtech#config

Qtech(config)#interface gigabitethernet 0/1

Qtech(config-if)#speed 100

%Warning: the speed/duplex of port GigabitEthernet 0/1 may not match with it's
neighbor.
```

The above messages show that bit-rate and duplex capabilities of port 1 may not match with that of port on neighbor device.

Verification

While the administrator is carrying out VLAN configuration, port bit-rate and duplex configuration, aggregation port configuration and port MTU configuration, if the information doesn't match with the configurations of neighbor device the corresponding error messages will be prompted.

4 CONFIGURING VRRP

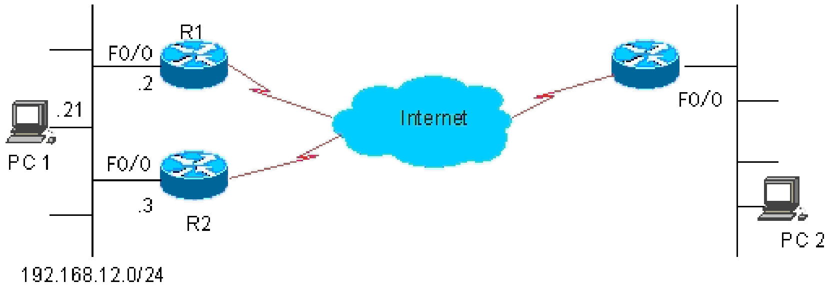
4.1 Overview

The Virtual Router Redundancy Protocol (VRRP) is designed to work in master/backup mode, so that traffic can switch over to a backup router without affecting internal or external data communication when the master router fails. In this process, parameters of the internal network do not need to be modified. Multiple routers in a VRRP group map to one virtual router. VRRP ensures that only one router transmits packets at a time, whereas hosts send data packets to the virtual router. The router that forwards data packets is elected as the master router. If the master router cannot work due to certain reasons at a time, a backup router is used to perform tasks of the original master router. Using VRRP allows all hosts in a local area network (LAN) looking like using only one router, and guarantees route connectivity even if the first-hop router fails.

RFC 2338, RFC 3768, and RFC 5798 define the format and operating mechanism of VRRP packets. A VRRP packet is a multicast packet with a specified destination address. It is sent by the master router to indicate that the master router is running properly or used to elect the master router. VRRP allows another router to automatically take over the router that fails to support the routing and forwarding function in an IP LAN, therefore implementing the hot backup and error tolerance of IP routes. VRRP also guarantees the communication continuity and reliability of hosts inside the LAN. One VRRP group consists of multiple routers in redundancy mode. At any moment, however, only one router serves as the master router to perform routing and forwarding functions. All other routers in the VRRP group are backup routers. The switchover between routers in the VRRP group is completely transparent to hosts in the LAN. RFC defines the following router switchover rules:

- VRRP elects the master router using a simple election method. First, it compares the VRRP priority of interfaces on various routers in the VRRP group and elects the router with the highest interface priority as the master router. The status of the elected router changes to Master. If the routers have the same priority, VRRP compares the primary IP address of network interfaces of the routers. The router with the biggest primary IP address becomes the master router to provide actual routing and forwarding services.
- After the master router is elected, other routers serve as backup routers whose status changes to Backup. These backup routers monitor the status of the master router by receiving VRRP packets that are periodically sent by the master router. When working normally, the master router sends a VRRP multicast packet that is known as an advertisement packet at a certain interval to inform the backup routers that the master router itself is running properly. If a backup router in the VRRP group does not receive any advertisement packet from the master router within the specified time, it sets its own status to Master. If multiple routers in the VRRP group exist in Master state, an election process as described previously is implemented again so that a router with the highest priority is selected as the new master router to implement the backup function of the VRRP.

Figure 1 Operating principles of VRRP



Once the master router is selected out of the VRRP group, the packets of all hosts in the LAN are routed and forwarded by the master router. Figure 1 shows the specific communication process. Routers R1 and R2 are connected to the LAN segment 192.168.12.0/24 through the Ethernet interface F0/0. VRRP is enabled on the Ethernet interface F0/0 of routers R1 and R2. The virtual router IP address of the VRRP group is set as the default gateway on all hosts in the LAN. Hosts in the LAN can detect only the virtual router of the VRRP group, whereas the master router that practically performs routing and forwarding functions is transparent to all of them. For example, the host PC 1 in the LAN sends a data packet to PC 2 by using the virtual router of the VRRP group as the default gateway so as to communicate with PC 2 in the same LAN. Upon receipt of the data packet, the master router of the VRRP group forwards the data packet to PC 2. In this communication process, PC 1 can detect only the virtual router but does not know whether R1 or R2 is the master router that plays the role of the virtual router. The master router of the VRRP group is selected between R1 and R2. Once the master router fails, the other router takes over traffic and becomes the new master router.

RFC 5798 redefines the format of a VRRP packet. The RGOS IPv6 VRRP complies with RFC 5798. In later descriptions, RGOS IPv6 VRRP is called VRRPv3 for short whereas the original VRRP implementation is simply called VRRPv2. In IPv4 VRRP, VRRPv2 and VRRPv3 are strictly distinguished from each other. The two VRRP standards define different fields in a VRRP packet. For this reason, RGOS IPv4 VRRP supports VRRPv3 and provide compatibility with VRRPv2. In contrast, IPv6 does not distinguish VRRPv2 from VRRPv3, because IPv6 VRRP is defined only in VRRPv3.

Currently, VRRP is defined in the following three protocols:

- RFC 2338
- RFC 3768
- RFC 5798

RFC 3768 is an update of RFC 2338 and defines mechanisms such as IPv4 VRRP. RFC 5798 is an improvement and extension of RFC 3768, and defines IPv4 VRRP and IPv6 VRRP.

⚠ Caution

To provide compatibility with widely deployed devices that do not support VRRPv3, IPv6 VRRP, however, invariably uses VRRPv3.

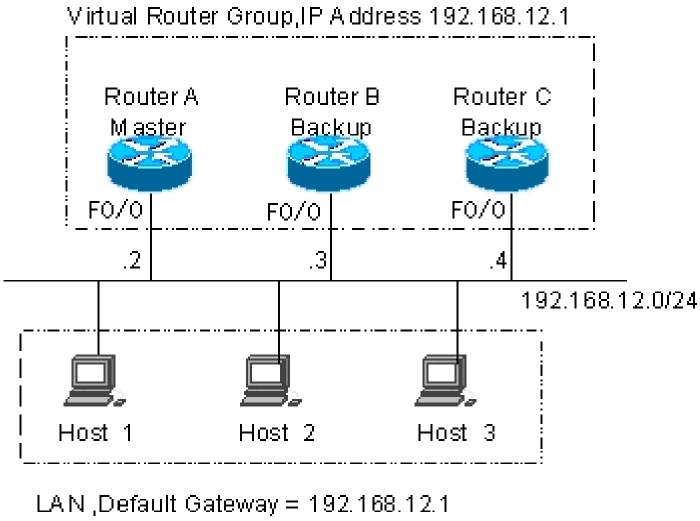
VRRP Application

VRRP supports two application modes: basic applications and advanced applications. In basic applications, only one VRRP group is used to implement simple route redundancy. In advanced applications, multiple VRRP groups are used to implement route redundancy and load balancing.

4.1.1 Route Redundancy

Figure 2 shows an example of basic VRRP applications.

Figure 2 Example of basic VRRP applications

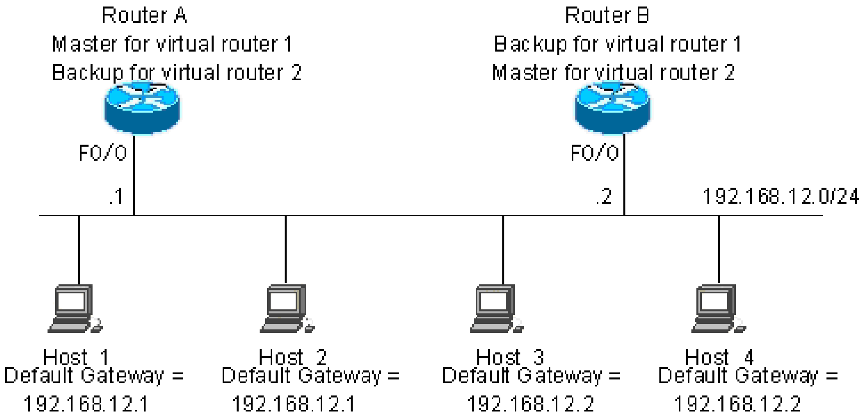


As shown in Figure 2, routers A, B, and C are connected to a LAN through Ethernet interfaces on which VRRP is enabled. Routers A, B, and C belong to the same VRRP group. The virtual IP address of the VRRP group is 192.168.12.1. Router A is elected as the master router of the VRRP group, whereas routers B and C are backup routers. The virtual router IP address 192.168.12.1 is set as the default gateway on hosts 1, 2, and 3 in the LAN. Data packets from hosts in the LAN to other networks are routed and forwarded by the master router A. If router A fails, a new master router is elected between routers B and C to route and forwards packets as a virtual router, therefore implementing simple route redundancy.

4.1.2 Load Balancing

Figure 3 shows an example of advanced VRRP applications.

Figure 3 Example of advanced VRRP applications



As shown in Figure 3, two virtual routers are set. For virtual router 1, the IP address 192.168.12.1 of the Ethernet interface F0/0 on router A is set as the IP address of the virtual router, so router A is the master router and router B is a backup router. For virtual router 2, the IP address 192.168.12.2 of the Ethernet interface F0/0 on router B is set as the IP address of the virtual router, so router B is the master router and router A is a backup router. The IP address 192.168.12.1 of virtual router 1 is set as the default gateway on hosts 1 and 2, and the IP address 192.168.12.2 of virtual router 2 is set as the default gateway on hosts 3 and 4 in the LAN. In this VRRP application example, routers A and B back up each other to implement route redundancy and share traffic from the LAN to implement load balancing.

4.2 Configuring VRRP

VRRP Configuration Task List

VRRP is applicable to multicast or broadcast LANs, such as Ethernets. VRRP configurations are mostly Ethernet interface configurations and involve the following configuration tasks:

- Enabling the VRRP function (Mandatory)
- Setting the authentication string of the VRRP group (Optional)
- Setting the advertisement interval of the VRRP group (Optional)
- Setting the preemption mode of the router in the VRRP group (Optional)
- Setting the Accept_Mode of the IPv6 VRRP virtual router
- Setting the priority of the router in the VRRP group (Optional)
- Setting the tracked interface of the VRRP group (Optional)
- Setting the tracked IP address of the VRRP group (Optional)
- Setting the periodic learning function of VRRP advertisement packets (Optional)
- Setting the description string of the VRRP group on the router (Optional)
- Setting the start delay of the VRRP group (Optional)
- Setting the IPv4 VRRP version (Optional)

You can determine which tasks to be configured based on your actual requirement.

4.2.1 Enabling the VRRP Function

You can add a VRRP group to a specific LAN segment by setting a group number and a virtual IPv4/IPv6 address for the VRRP group so as to enable the VRRP function on the corresponding Ethernet interface.

Command	**Function**
Qtech(config-if)# **vrrp** *group* **ip** *ipaddress* [**secondary**]	Enables IPv4 VRRP.
Qtech(config-if)# **no vrrp** *group* **ip** *ipaddress* [**secondary**]	Disables IPv4 VRRP.
Or:	
Qtech(config-if)# **vrrp** *group* **ipv6** *ipv6-address*	Enables IPv6 VRRP.
Qtech(config-if)# **no vrrp** *group* **ipv6** *ipv6-address*	Disables IPv6 VRRP.

The group number specified by the *group* parameter ranges from 1 to 255. If the virtual IP address is not specified, the router does not participate in the VRRP group. If the *secondary* parameter is not specified, the specified IP address becomes the primary IP address of the virtual router. The system does not identify whether an IPv6 address is a primary or secondary address. The first virtual IP address configured for an IPv6 VRRP group, however, must be a link-local address.

⚠ Caution

If the virtual IP address (primary or secondary) or virtual IPv6 address (link-local or non-link-local) of the VRRP group is consistent with the IP address (primary or secondary) or IPv6 address (link-local or non-link-local) of an Ethernet interface, the VRRP group is considered as owning the real IP address of the Ethernet interface. In this case, the priority of the VRRP group is 255. If the Ethernet interface is available, the VRRP group is automatically in Master state.

☑ The NMX-2GEH line card supports listening to a maximum of 15 MAC addresses. The number of configurable VRRP groups depends on the number of MAC addresses supported by the current line card. If the number of configured VRRP groups is larger than the maximum number allowed by the line card, an error message is displayed. Note that the line card may also need to support MAC address listening according to other protocols, such as Open Shortest Path First (OSPF) and Routing Information Protocol (RIP).

4.2.2 Setting the Authentication String of the VRRP Group

VRRP supports two authentication modes: plain text authentication and no-authentication. When setting the authentication string of a VRRP group, you can also set the authentication mode of the VRRP group to plain text authentication or no-authentication. All members of the VRRP group must be set to the same authentication mode so as to normally communicate with one another. In plain text authentication mode, all routers in the VRRP group must have the same authentication password. The plain text authentication password does not guarantee security but is used only to prevent or prompt VRRP configuration errors.

Command	**Function**
Qtech(config-if)# **vrrp group authentication** *string*	Sets the authentication string of the IPv4 VRRP group.
Qtech(config-if)# **no vrrp group authentication**	Sets the authentication mode of the IPv4 VRRP group to no-authentication.

By default, the authentication mode of a VRRP group is no-authentication. If the plain text authentication mode is specified, the plain text authentication password consists of up to eight bytes.

⚠ Caution

The authentication mode is already abandoned in RFC 5798 and no longer adopted in new specifications. Therefore, the user-specified authentication mode is applicable to VRRPv2 packets only.

4.2.3 Setting the Advertisement Interval of the VRRP Group

Command	**Function**
Qtech(config-if)# **vrrp group timers advertise** { *advertise-interval* | **csec** *centisecond-interval* }	Sets the VRRP advertisement interval of the IPv4 VRRP master router.
Qtech(config-if)# **no vrrp group timers advertise**	Restores the default VRRP advertisement interval of the IPv4 VRRP master router.
Or:	
Qtech(config-if)# **vrrp ipv6 group timers advertise**{ *advertise-interval* | **csec** *centisecond-interval* }	Sets the VRRP advertisement interval of the IPv6 VRRP master router.
Qtech(config-if)# **no vrrp ipv6 group timers advertise**	Restores the default VRRP advertisement interval of the IPv6 VRRP master router.

If the current router is the master router of the VRRP group, it sends VRRP advertisement packets at the set interval to advertise its own VRRP status, priority, and other information. By default, the master router sends VRRP advertisement packets at an interval of one second. The time for a VRRP backup router to switch over to the master router is defined in RFC 2338, RFC 3768, and RFC 5798. It is three times the advertisement interval plus a Skew_Time. The Skew_Time is calculated with the following formula: Skew_Time = (((256 – Priority of the VRRP group) x VRRP advertisement interval) / 256). VRRPv3 supports a VRRP advertisement interval of the master router ranging from 50 to 99 milliseconds to accelerate VRRP convergence time without correlation with BFD. If the network traffic is heavy, an interval in milliseconds is not recommended, as the backup router may fail to receive the VRRP advertisement packets from the master router within the interval due to heavy traffic, causing status change.

⚠ Caution

If periodic VRRP learning is not enabled on routers, the same VRRP advertisement interval must be set on all routers in a VRRP group; otherwise, backup routers discard received VRRP advertisement packets.

4.2.4 Setting the Preemption Mode of the Router in the VRRP Group

A router in a VRRP group that works in preemption mode preempts other routers in the VRRP group to become the master router once detecting that its own priority is higher than the priority of the existing master router. If the VRRP group works in non-preemption mode, the router does not preempt other routers in the VRRP group to become the master router even when detecting that its own priority is higher than the priority of the existing master router. Setting the preemption mode is insignificant for a VRRP group whose virtual router address is an Ethernet interface IP address, because the router configured with the Ethernet interface IP address has the highest priority and automatically becomes the master router of the VRRP group.

Command	Function
Qtech(config-if)# **vrrp** *group* **preempt** [**delay** *seconds]*	Sets the IPv4 VRRP group to the preemption mode.
Qtech(config-if)# **no vrrp** *group* **preempt** [**delay]**	Sets the IPv4 VRRP group to the non-preemption mode or restores the default delay.
or	
Qtech(config-if)# **vrrp ipv6** *group* **preempt** [**delay** *seconds]*	Sets the IPv6 VRRP group to the preemption mode.
Qtech(config-if)# **no vrrp ipv6** *group* **preempt** [**delay]**	Sets the IPv6 VRRP group to the non-preemption mode or restores the default delay.

The optional parameter *delay seconds* defines a delay before a backup VRRP router advertises itself as the master router of the VRRP group. It is 0 seconds by default. Once the VRRP function is enabled, the VRRP group works in preemption mode by default.

4.2.5 Setting the Accept_Mode of the IPv6 VRRP Virtual Router

The Accept_Mode can be set for the IPv6 VRRP virtual router that serves as the master router to determine whether to receive and process packets destined to the IP address of the IPv6 VRRP virtual router itself. If the Accept_Mode is enabled, the IPv6 VRRP virtual router receives and processes packets destined to the IP address of the virtual router itself. If the Accept_Mode is not enabled, the IPv6 VRRP virtual router discards packets destined to the IP address of the virtual router itself but does not discard NA and NS packets. By default, the Accept_Mode is disabled. In addition, the IPv6 VRRP virtual router in Owner state receives and processes packets destined to the IP address of the virtual router itself, no matter whether the Accept_Mode is enabled or disabled.

Command	Function
Qtech(config-if)# **vrrp ipv6** *group* **accept_mode**	Enables the Accept_Mode for an IPv6 VRRP group.
Qtech(config-if)# **no vrrp ipv6** *group* **accept_mode**	Disables the Accept_Mode for an IPv6 VRRP group.

4.2.6 Setting the Priority of the Router in the VRRP Group

According to VRRP, the role of a router in a VRRP group is determined by the priority of the router in the VRRP group. A router in a VRRP group becomes the active or master router of the VRRP group if it works in preemption mode, has the highest priority, and has obtained a virtual IP address. The other routers that have a priority lower than the priority of the master router in the VRRP group becomes backup or listening routers. Once the VRRP function is enabled on a router, the priority of the router in a VRRP group is 100 by default.

Command	Function
Qtech(config-if)# **vrrp** *group* **priority** *level*	Sets the priority of a router in an IPv4 VRRP group.
Qtech(config-if)# **no vrrp** *group* **priority**	Restores the default priority of a router in an IPv4 VRRP group.
Or:	
Qtech(config-if)# **vrrp ipv6** *group* **priority** *level*	Sets the priority of a router in an IPv6 VRRP group.

Qtech(config-if)# **no vrrp ipv6** *group* **priority**	Restores the default priority of a router in an IPv6 VRRP group.

The priority defined by the *level* parameter ranges from 1 to 254. If the virtual IP address of a VRRP group is consistent with the real IP address of an Ethernet interface on the local router, the priority of the local router in the VRRP group is 255. In this case, the VRRP group configured on the router is automatically in Master state as long as the Ethernet interface is available, no matter whether the VRRP group works in preemption mode or not.

4.2.7 Setting the Tracked Interface of the VRRP Group

After a tracked interface is set for the VRRP group, the system dynamically adjusts the priority of the local router according to the status of the tracked interface. When the tracked interface becomes unavailable, the local router decreases its VRRP group priority based on settings. At this time, another router in the VRRP group may become the active or master router of the VRRP group if its status is more stable and its priority is higher.

Command	Function
Qtech(config-if)# **vrrp** *group* **track** *interface-type number* [*interface-priority*]	Sets the tracked interface of an IPv4 VRRP group.
Qtech(config-if)# **no vrrp** *group* **track** *interface-type number*	Cancels the tracked interface set for an IPv4 VRRP group.
Or:	
Qtech(config-if)# **vrrp ipv6** *group* **track** *interface-type number* [*interface-priority*]	Sets the tracked interface of an IPv6 VRRP group.
Qtech(config-if)# **no vrrp ipv6** *group* **track** *interface-type number*	Cancels the tracked interface set for an IPv6 VRRP group.

By default, no tracked interface is set for a VRRP group. The value of the *interface-priority* parameter ranges from 1 to 255. It is 10 by default if not specified.

Note

The tracked interface can only be a reachable logical Layer 3 (L3) interface, such as a routed port, SVI, loopback interface, or tunnel interface.

4.2.8 Setting the Tracked IPv4/IPv6 Address of the VRRP Group

After a tracked IP address is set for the VRRP group, the system dynamically adjusts the priority of the local router depending on whether the tracked IP address is reachable. When the tracked IP address is unreachable and cannot be pinged, the local router decreases its VRRP group priority based on settings. At this time, another router in the VRRP group may become the active or master router of the VRRP group if its priority is higher. In the following commands, the optional parameter *interval* defines an interval at which the system detects whether the destination address is reachable, the optional parameter *timeout* defines a timeout interval that is used to determine that the destination is unreachable, and the optional parameter *retry* defines the number of retries when the destination is unreachable.

Command	Function
Qtech(config-if)# **vrrp** *group* **track** *ip-address* [**interval** *interval-value]* [**timeout** *timeout-value]* [**retry** *retry-value] [priority]*	Sets the tracked IP address of an IPv4 VRRP group.
Qtech(config-if)# **no vrrp** *group* **track** *ip-address*	Cancels the tracked IP address set for an IPv4 VRRP group.
Or:	
Qtech(config-if)# **vrrp ipv6** *group* **track** { *ipv6-global-address* | { *ipv6-linklocal-address interface-type number* } } [**interval** *interval-value]* [**timeout** *timeout-value]* [**retry** *retry-value] [priority]*	Sets the tracked IP address of an IPv6 VRRP group.
Qtech(config-if)# **no vrrp ipv6** *group* **track** { *ipv6-global-address* | { *ipv6-linklocal-address interface-type number* } }	Cancels the tracked IP address set for an IPv6 VRRP group.

By default, no tracked IP address is set for a VRRP group. The value of the *interval-value* parameter ranges from 1 to 3600 seconds. It is 3 seconds by default if not specified. The value of the *timeout-value* parameter ranges from 1 to 60 seconds. It is 1 second by default if not specified.

The value of the *timeout-value* parameter must be smaller than or equal to that of the *interval-value* parameter. The value of the *retry-value* parameter ranges from 1 to 60. It is 1 by default if not specified. The value of the p*riority* parameter ranges from 1 to 255. It is 10 seconds by default if not specified. A VRRP IPv6 link-local IP address is preferred as the tracked IP address of an IPv6 VRRP group. If you set the tracked IP address to a link-local IP address, you must also set the specified interface.

4.2.9 Setting the Periodic Learning of VRRP Advertisement Packets

If the periodic learning function is enabled on the local router that is a VRRP backup router, the local router learns a VRRP advertisement interval from VRRP advertisement packets sent by the master router and calculates a VRRP master invalidity interval using the learned VRRP advertisement interval instead of the VRRP advertisement interval set on the local router itself. This command enables a backup router to synchronize the VRRP advertisement interval locally set on itself to the VRRP advertisement interval of the master router.

Command	Function
Qtech(config-if)# **vrrp** *group* **timers learn**	Enables the periodic learning of IPv4 VRRP advertisement packets.
Qtech(config-if)# **no vrrp** *group* **timers learn**	Disables the periodic learning of IPv4 VRRP advertisement packets.
Or:	
Qtech(config-if)# **vrrp ipv6** *group* **timers learn**	Enables the periodic learning of IPv6 VRRP advertisement packets.
Qtech(config-if)# **no vrrp ipv6** *group* **timers learn**	Disables the periodic learning of IPv6 VRRP advertisement packets.

By default, the periodic learning function is disabled for a VRRP group.

Note

If the advertisement interval that a VRRP backup router learns from a received VRRP advertisement packet is inconsistent with the VRRP advertisement interval locally set on the VRRP backup router and the periodic learning function is disabled on the VRRP backup router, the VRRP backup router discards the VRRP advertisement packet. Otherwise, the VRRP backup router receives the VRRP advertisement packet and calculates a VRRP master invalidity interval using the advertisement interval carried in the VRRP advertisement packet.

4.2.10 Setting the Description String of the VRRP Group on the Router

You can set a description string for a VRRP group to distinguish it from other VRRP groups.

Command	Function
Qtech(config-if)# **vrrp** *group* **description** *text*	Sets the description string of an IPv4 VRRP group.
Qtech(config-if)# **no vrrp** *group* **description**	Cancels the description string set for an IPv4 VRRP group.
Or:	
Qtech(config-if)# **vrrp ipv6** *group* **description** *text*	Sets the description string of an IPv6 VRRP group.
Qtech(config-if)# **no vrrp ipv6** *group* **description**	Cancels the description string set for an IPv6 VRRP group.

By default, no description string is set for a VRRP group. The description string of a VRRP group consists of at most 80 bytes.

Note

If the description string of a VRRP group contains blanks, the quotation mark (") must be used to identify the description string.

4.2.11 Setting the Start Delay of the VRRP Group

You can set the start delay of a VRRP group on a certain interface. The system supports two types of delay: system start delay and interface activity delay, which can be configured separately or together.

In non-preemption mode, a router with a higher VRRP group priority does not preempt the master router in the same VRRP group when it is started. In some cases, however, a router newly started preempts other routers to become the VRRP master router, even if it is set to the non-preemption mode. This is because the VRRP group on the interface does not receive the VRRP advertisement packet from the master router in the same VRRP group in time when the router is started or the interface becomes active.

To resolve the preceding problem, you can run a command to configure a start delay for the VRRP group. Then the VRRP group on the interface waits for a certain time before being started when the router is started or the interface becomes active, so that the non-preemption configuration takes effect.

If a VRRP advertisement packet is received on the interface after the start delay is set, the start delay is canceled and VRRP is immediately started on the interface.

Command	Function
Qtech(config-if)# **vrrp delay** { **minimum** *min-seconds* | **reload** *reload-seconds* }	Sets the start delay of the VRRP group on the interface.
Qtech(config-if)# **no vrrp delay**	Cancels the start delay set for the VRRP group on the interface.

By default, no start delay is configured for the VRRP group on an interface. Both the system start delay and the interface activity delay as mentioned previously range from 0 to 60 seconds. After this command is configured on an interface, the configurations apply to both IPv4 VRRP and IPv6 VRRP groups on the interface.

4.2.12 Setting the IPv4 VRRP Version

You can set the version of IPv4 VRRP to VRRPv2 or VRRPv3. By default, VRRPv2 is applied.

Command	Function
Qtech(config-if)# **vrrp** *group* **version** { **2 | 3** }	Sets the IPv4 VRRP version.
Qtech(config-if)# **no vrrp** *group* **version**	Uses the VRRPv2 by default.

4.3 Monitoring and Maintenance of VRRP

The **show vrrp**, **show ipv6 vrrp**, and **debug vrrp** commands are available for monitoring and maintaining VRRP. You can run the **show vrrp** command to check the IPv4 VRRP status of the local router, the **show ipv6 vrrp** command to check the IPv6 VRRP status of the local router, and the **debug vrrp** command to check VRRP information, such as status changes to a VRRP group, VRRP advertisement transmitting/receiving, and VRRP events.

show vrrp

Run the following **show vrrp** commands to check the IPv4 VRRP status of the local router:

Command	Function
Qtech# **show** **[ipv6] vrrp** **[brief** | *group]*	Displays the IPv4 or IPv6 VRRP status of the local router.
Qtech# **show** **[ipv6] vrrp interface** *type number* **[brief]**	Displays the IPv4 or IPv6 VRRP status of a specific network interface.

Command examples:

3) show [ipv6] vrrp

```
Qtech# show vrrp
FastEthernet 0/0 - Group 1
State is Backup
Virtual IP address is 192.168.201.1 configured
Virtual MAC address is 0000.5e00.0101
Advertisement interval is 3 sec
Preemption is enabled
min delay is 0 sec
Priority is 100
Master Router is 192.168.201.213 , priority is 120
Master Advertisement interval is 3 sec
Master Down interval is 10.82 sec
FastEthernet 0/0 - Group 2
State is Master
Virtual IP address is 192.168.201.2 configured
Virtual MAC address is 0000.5e00.0102
Advertisement interval is 3 sec
Preemption is enabled
min delay is 0 sec
```

```
Priority is 120
Master Router is 192.168.201.217 (local), priority is 120
Master Advertisement interval is 3 sec
Master Down interval is 10.59 sec
Qtech#show ipv6 vrrp
GigabitEthernet 0/13 - Group 1
  State is Master
  Virtual IPv6 address is as follows:
    FE80::2
    1::2
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 1 sec
  Preemption is enabled
    min delay is 0 sec
  Priority is 100
  Master Router is FE80::1 (local), priority is 100
  Master Advertisement interval is 1 sec
  Master Down interval is 3.60 sec
```

The command outputs include the following information:

- Names of Ethernet interfaces where IPv4/IPv6 VRRP groups are configured
- ID, status, priority, preemption mode, VRRP advertisement interval, virtual IP address, and virtual MAC address of each VRRP group configured on the interfaces
- IP address, priority, advertisement interval, and invalidity interval of the master router in each VRRP group
- Tracked interface and priority change metric of each VRRP group

4) show [ipv6] vrrp brief

```
Qtech# show vrrp brief
Interface          Grp  Pri  Time  Own  Pre  State   Master addr      Group addr
FastEthernet 0/0   1    100  3.60  -    P    Backup  192.168.201.213  192.168.201.1
FastEthernet 0/0   2    120  3.53  -    P    Master  192.168.201.217  192.168.201.2
Qtech#show ipv6 vrrp brief
Interface              Grp  Pri  timer  Own  Pre  State   Master addr   Group addr
GigabitEthernet 0/13   1    100  3.60   -    P    Master  FE80::1       FE80::2
```

- The command outputs include the following information:
- Names of Ethernet interfaces where IPv4/IPv6 VRRP groups are configured
- ID, status, priority, preemption mode, and virtual IP address of each VRRP group configured on the interfaces
- IP address of the mater router in each VRRP group

5) show [ipv6] vrrp interface

```
Qtech# show vrrp interface FastEthernet 0/0
FastEthernet 0/0 - Group 1
State is Backup
Virtual IP address is 192.168.201.1 configured
Virtual MAC address is 0000.5e00.0101
Advertisement interval is 3 sec
Preemption is enabled
VRRP standard version is V3
min delay is 0 sec
Priority is 100
Master Router is 192.168.201.213 , priority is 120
Master Advertisement interval is 3 sec
Master Down interval is 10.82 sec
FastEthernet 0/0 - Group 2
State is Master
Virtual IP address is 192.168.201.2 configured
Virtual MAC address is 0000.5e00.0102
Advertisement interval is 3 sec
Preemption is enabled
min delay is 0 sec
Priority is 120
Master Router is 192.168.201.217 (local), priority is 120
Master Advertisement interval is 3 sec
Master Down interval is 10.59 sec
Qtech#
Qtech#show ipv6 vrrp inter gig 0/13
```

```
GigabitEthernet 0/13 - Group 1
  State is Master
  Virtual IPv6 address is as follows:
    FE80::2
    1::2
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 1 sec
  Preemption is enabled
    min delay is 0 sec
  Priority is 100
  Master Router is FE80::1 (local), priority is 100
  Master Advertisement interval is 1 sec
  Master Down interval is 3.60 sec
```

The command outputs include the following information:

- Names of Ethernet interfaces where IPv4/IPv6 VRRP groups are configured
- ID, status, priority, preemption mode, VRRP advertisement interval, virtual IP address, and virtual MAC address of each VRRP group configured on the interfaces
- IP address, priority, advertisement interval, and invalidity interval of the master router in each VRRP group
- Tracked interface and priority change metric of each VRRP group

Debug vrrp

You can run the following **debug [ipv6] vrrp** commands to enable or disable VRRP debugging on the local router:

Command	Function
Qtech# **debug** [ipv6] **vrrp errors**	Enables VRRP error debugging.
Qtech# **no debug** [ipv6] **vrrp errors**	Disables VRRP error debugging.
Qtech# **debug** [ipv6] **vrrp events**	Enables VRRP event debugging.
Qtech# **no debug** [ipv6] **vrrp events**	Disables VRRP event debugging.
Qtech# **debug** [ipv6] **vrrp packets**	Enables VRRP packet debugging.
Qtech# **no debug** [ipv6] **vrrp packets**	Disables VRRP packet debugging.
Qtech# **debug** [ipv6] **vrrp state**	Enables VRRP status debugging..
Qtech# **no debug** [ipv6] **vrrp state**	Disables VRRP status debugging.
Qtech# **debug** [ipv6] **vrrp**	Enables VRRP debugging.
Qtech# **no debug** [ipv6] **vrrp**	Disables VRRP debugging.

Command examples:

6) debug [ipv6] vrrp

```
Qtech# debug vrrp
Qtech#
%VRRP-6-STATECHANGE: FastEthernet 0/0 IPv4 VRRP Grp 1 state Master -> Backup
VRRP: IPv4 VRRP Grp 1 Advertisement from 192.168.201.213 has invalid virtual address
192.168.1.1
VRRP: IPv4 VRRP Grp 1 on interface Gi0/13 is sending IPv4 VRRP V2 advertisement
checksum a352.
Qtech# debug ipv6 vrrp
Qtech#
VRRP: IPv6 VRRP Grp 1 Event - Advert higher or equal priority
%VRRP-6-STATECHANGE: FastEthernet 0/0 IPv6 VRRP  Grp 1 state Backup -> Master
Qtech#
VRRP: IPv6 VRRP Grp 1 on interface Gi0/13 is sending IPv6 VRRP v3 advertisement
checksum 6de3.
```

The **debug [ipv6] vrrp** command is equivalent to a combination of commands **debug [ipv6] vrrp errors**, **debug [ipv6] vrrp events**, **debug [ipv6] vrrp packets**, and **debug [ipv6] vrrp state**.

7) debug [ipv6] vrrp errors

```
Qtech# debug vrrp errors
Qtech#
VRRP: IPv4 VRRP Grp 1 Advertisement from 192.168.1.1 has wrong checksum.
VRRP: IPv4 VRRP Grp 1 Advertisement from 192.168.1.1 has wrong checksum.
VRRP: IPv4 VRRP Grp 1 Advertisement from 192.168.1.1 has wrong checksum.
```

The preceding information indicates that the local router has received VRRP advertisement packets that contain checksum errors for IPv4 VRRP group 1 from 192.168.1.1.

```
Qtech# debug ipv6 vrrp errors
```

```
Qtech#
VRRP: IPv6 VRRP Grp 1 Advertisement from FE80::2D0:F8FF:FE22:DE00 has different IP
address.
VRRP: IPv6 VRRP Grp 1 Advertisement from FE80::2D0:F8FF:FE22:DE00 has different IP
address.
VRRP: IPv6 VRRP Grp 1 Advertisement from FE80::2D0:F8FF:FE22:DE00 has different IP
address.
```

The preceding information indicates that the local router has received VRRP advertisement packets that carry different IPv6 group addresses for the same IPv6 VRRP group.

8) debug [ipv6] vrrp events

```
Qtech# debug vrrp events
Qtech#
VRRP: IPv4 VRRP Grp 1 Event - Advert higher or equal priority
VRRP: IPv4 VRRP Grp 1 Event - Advert higher or equal priority
Qtech# debug ipv6 vrrp events
VRRP: IPv6 VRRP Grp 1 Event - Advert higher or equal priority
Qtech#
```

The preceding information indicates that the local router has received VRRP advertisement packets with a priority higher than or equal to the local priority for local IPv4 VRRP and IPv6 VRRP groups.

9) debug [ipv6] vrrp packets

```
Qtech# debug vrrp packets
Qtech#
VRRP: IPv4 VRRP Grp 1 on interface Gi0/13 is sending IPv4 VRRP V2 advertisement
checksum a352.
VRRP: IPv4 VRRP Grp 1 on interface Gi0/13 is sending IPv4 VRRP V2 advertisement
checksum a352.
Qtech# debug ipv6 vrrp packets
VRRP: IPv6 VRRP Grp 1 on interface Gi0/13 is sending IPv6 VRRP v3 advertisement
checksum 6de3.
VRRP: IPv6 VRRP Grp 1 on interface Gi0/13 is sending IPv6 VRRP v3 advertisement
checksum 6de3.
```

The preceding information indicates that local IPv4 VRRP group 1 and local IPv6 VRRP group 1 are sending VRRP advertisement packets.

```
Qtech# debug vrrp packets
Qtech#
VRRP: IPv4 VRRP Grp 1 on interface Gi0/13 received ipv4 v2 advertisement priority 100,
source 192.168.1.1.
Qtech# debug ipv6 vrrp packets

VRRP: IPv6 VRRP Grp 1 on interface Gi0/13 received ipv6 v3 advertisement priority 100,
source FE80::1.
```

The preceding information indicates that the local router has received a VRRP advertisement packet with the priority of 100 for IPv4 VRRP group 1 from 192.168.1.1 and also a VRRP advertisement for IPv6 VRRP group 1 from fe80::1.

10) debug [ipv6] vrrp state

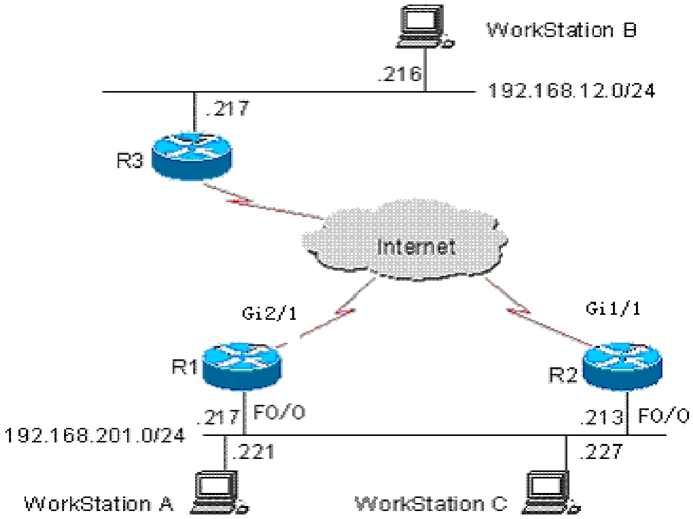
```
Qtech# debug vrrp state
Qtech#
VRRP: IPv4 VRRP Grp 1 add primary virtual IP, startup
Qtech# debug ipv6 vrrp state
VRRP: IPv6 VRRP Grp 1 add primary virtual IP, startup.
```

The preceding information indicates that both the IPv4 VRRP group and the IPv6 VRRP group on the interface FastEthernet 0/0 are configured with a primary IP address and started.

4.3.1 Example of Configuring an IPv4 VRRP Group

As shown in Figure 4, a VRRP group is configured on routers R1 and R2 to provide the VRRP service for the internal network segment 192.168.201.0/24, whereas only the common routing function instead of any VRRP group is enabled on R3. This example shows VRRP-related configurations on routers R1 and R2 only.

Figure 4 VRRP network topology



In the following example, the configuration of router R3 is invariable. The following shows configurations on R3:

```
!
!
hostname "R3"
!
!
!
interface FastEthernet 0/0
/* The no switchport command needs to be run on a switch only*/
no switchport
ip address 192.168.12.217 255.255.255.0
!
interface GigabitEthernet 1/1
/* The no switchport command needs to be run on a switch only*/
no switchport
ip address 60.154.101.5 255.255.255.0
!
interface GigabitEthernet 2/1
/* The no switchport command needs to be run on a switch only*/
no switchport
ip address 202.101.90.61 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.12.0 0.0.0.255 area 10
network 60.154.101.0 0.0.0.255 area 10
!
!
!
end
```

Example of Configuring a VRRP Group

Devices are connected, as shown in Figure 1-4. In this example, a workstation group (192.168.201.0/24) uses a VRRP group formed by routers R1 and R2. Its gateway is set to the virtual router IP address 192.168.201.1 of the VRRP group, so that the workstation group can access a remote workstation group whose network address is 192.168.12.0/24 through the virtual router 192.168.201.1. Here, R1 is set as the master router of the VRRP group. In normal cases, R1 provides the gateway (192.168.201.1) function. If R1 is unreachable because it is shut down or faulty, R2 takes the place of R1 to provide the gateway function. Below are related configurations on R1 and R2.

Configurations on R1:

```
!
!
hostname "R1"
```

```
!
!
interface FastEthernet 0/0
ip address 192.168.201.217 255.255.255.0
vrrp 1 priority 120
vrrp 1 version 3

vrrp 1 timers advertise 3
vrrp 1 ip 192.168.201.1
!
interface GigabitEthernet 2/1
ip address 202.101.90.63 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
```

Configurations on R2:

```
!
hostname "R2"
!
interface FastEthernet 0/0
ip address 192.168.201.213 255.255.255.0
vrrp 1 ip 192.168.201.1
 vrrp 1 version 3
vrrp 1 timers advertise 3
!
interface GigabitEthernet 1/1
/* The no switchport command needs to be run on a switch only*/
no switchport
ip address 60.154.101.3 255.255.255.0
!
!
router ospf
network 60.154.101.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

As can be seen, R1 and R2 belong to IPv4 VRRP group 1. Both routers use VRRPv3, point to the same virtual router IP address 192.168.201.1, and work in VRRP preemption mode. Since the priority of R1 in the IPv4 VRRP group is 120 but that of R2 is the default value 100, R1 works as the master router of the IPv4 VRRP group in normal cases.

Example of Configuring the Tracked Interface of an IPv4 VRRP Group

Devices are connected, as shown in Figure 4. In this example, a workstation group (192.168.201.0/24) uses a VRRP group formed by routers R1 and R2. Its gateway is set to the virtual router IP address 192.168.201.1 of the VRRP group, so that the workstation group can access a remote workstation group whose network address is 192.168.12.0/24 through the virtual router 192.168.201.1. Here, R1 is set as the master router of the VRRP group. Different from the example as described previously for configuring a single VRRP group, a VRRP tracked interface (GigabitEthernet 2/1) is set on R1. In normal cases, R1 provides the virtual gateway (192.168.201.1) function. If R1 is unreachable because it is shut down or faulty, R2 takes the place of R1 to provide the virtual gateway function. In particular, when the interface GigabitEthernet 2/1 on R1 to connect to a wide area network (WAN) is unavailable, R1 decreases its VRRP group priority based on settings, so that R2 has a chance to become the master router and provide the virtual gateway function. If the interface GigabitEthernet 2/1 on R1 is recovered later, R1 restores its own VRRP group priority and then become the master router to provide the virtual gateway function. Below are related configurations on R1 and R2.

Configurations on R1:

```
!
!
hostname "R1"
```

```
!
!
interface FastEthernet 0/0
ip address 192.168.201.217 255.255.255.0
vrrp 1 priority 120
vrrp 1 timers advertise 3
vrrp 1 ip 192.168.201.1
vrrp 1 track GigabitEthernet 2/1 30
!

interface GigabitEthernet 2/1
ip address 202.101.90.63 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

Configurations on R2:

```
!
!
hostname "R2"
!
interface FastEthernet 0/0
ip address 192.168.201.213 255.255.255.0
vrrp 1 ip 192.168.201.1
vrrp 1 timers advertise 3
!
interface GigabitEthernet 1/1
ip address 60.154.101.3 255.255.255.0
!
router ospf
network 60.154.101.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

As can be seen, R1 and R2 belong to VRRP group 1. Both routers use the same VRRP group authentication mode (no-authentication), point to the same virtual router IP address 192.168.201.1, and work in VRRP preemption mode. The VRRP advertisement interval is set to three seconds on both R1 and R2. Since the priority of R1 in the VRRP group is 120 but that of R2 is the default 100, R1 works as the master router of the VRRP group in normal cases. If R1 detects that its interface GigabitEthernet 2/1 to the WAN is unavailable, it decreases its VRRP group priority by 30 to 90, so that R2 becomes the master router. If R1 detects later that its interface GigabitEthernet 2/1 to the WAN is available again, it increases its own VRRP group priority by 30 to 120, so that R1 again becomes the master router.

4.3.2 Configuring Multiple IPv4 VRRP Groups

Multiple VRRP groups can be configured on one Ethernet interface to implement load balancing and backup one another to provide more reliable and stable network services.

Devices are connected, as shown in Figure 4. In this example, a workstation group (192.168.201.0/24) uses two VRRP groups formed by routers R1 and R2. The gateway of some workstations such as workstation A is set to the virtual IP address 192.168.201.1 of VRRP group 1, and that of the rest workstations such as workstation C is set to the virtual IP address 192.168.201.2 of VRRP group 2. R1 serves as the master router of VRRP group 2 and the backup router of VRRP group 1, whereas R2 serves as the master router of VRRP group 1 and the backup router of VRRP group 2. Below are related configurations on R1 and R2.

Configurations on R1:

```
!
!
hostname "R1"
```

```
!
interface FastEthernet 0/0
ip address 192.168.201.217 255.255.255.0
vrrp 1 timers advertise 3
vrrp 1 ip 192.168.201.1
vrrp 2 priority 120
vrrp 2 timers advertise 3
vrrp 2 ip 192.168.201.2
vrrp 2 track GigabitEthernet 2/1 30
!
interface GigabitEthernet 2/1
ip address 202.101.90.63 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

Configurations on R2:

```
!
!
hostname "R2"
!
interface FastEthernet 0/0
ip address 192.168.201.213 255.255.255.0
vrrp 1 ip 192.168.201.1
vrrp 1 timers advertise 3
vrrp 1 priority 120
vrrp 2 ip 192.168.201.2
vrrp 2 timers advertise 3
!
interface GigabitEthernet 1/1
ip address 60.154.101.3 255.255.255.0
!
router ospf
network 60.154.101.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
!
end
```

As can be seen, R1 and R2 back up each other. They serve as the master router in VRRP group 1 or 2 to provide different virtual gateways.

4.3.3 Example of Configuring an IPv6 VRRP Group

Configuring a VRRP Group

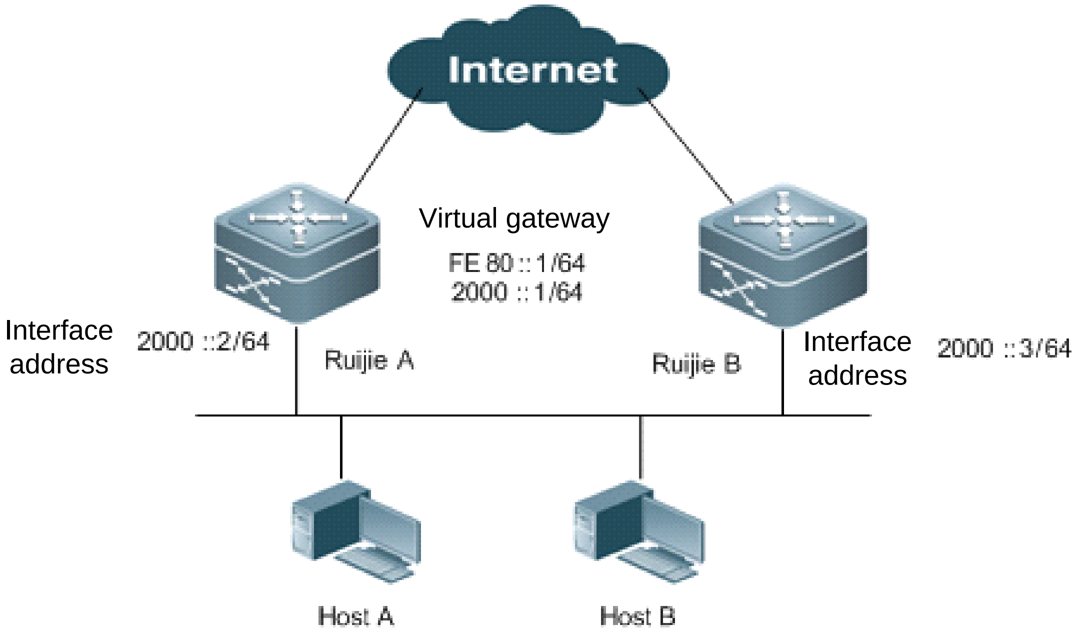
Networking Requirements

This configuration instance is applicable to both switches and routers.

- Hosts A and B access the Internet through a gateway. The default gateway is 2000::1/64 on both hosts.
- Qtech A and Qtech B are two routers that form IPv6 VRRP group 1. The virtual addresses are 2000::1/64 and FE80::1.
- When Qtech A works properly, the packets of Host A to the Internet are forwarded by Qtech A. When Qtech A fails, the packets of Host A to the Internet are forwarded by Qtech B.

Networking Topology

Figure 5 Network topology of the example for configuring an IPv6 VRRP group



Configuration Steps

Configurations on Qtech A:

# Configure an IPv6 address on an interface to enable the IPv6 service on the interface.

```
interface FastEthernet 0/1
```

/* The **no switchport** command needs to be run on a switch only*/

```
no switchport
ipv6 address 2000::2/64
!
```

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
vrrp 1 ipv6 FE80::1
vrrp 1 ipv6 2000::1
```

# Change the priority of IPv6 VRRP group 1 to 120.

```
vrrp ipv6 1 priority 120
```

# Change the advertisement interval of IPv6 VRRP group 1 to 3s.

```
vrrp ipv6 1 timers advertise 3
```

# Set the Accept_Mode of the IPv6 VRRP group.

```
vrrp ipv6 1 accept_mode
!
```

Configurations on Qtech B:

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/1
```

/* The **no switchport** command needs to be run on a switch only*/

```
no switchport
ipv6 address 2000::3/64
!
```

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
vrrp 1 ipv6 FE80::1
vrrp 1 ipv6 2000::1
```

# Change the priority of IPv6 VRRP group 1 to 120.

```
vrrp ipv6 1 priority 100
```

# Change the advertisement interval of IPv6 VRRP group 1 to 3s.

```
vrrp ipv6 1 timers advertise 3
```

# Set the Accept_Mode of the IPv6 VRRP group.

```
vrrp ipv6 1 accept_mode
```

As can be seen, Qtech A and Qtech B belong to IPv6 VRRP group 1, point to the same virtual router IPv6 address (2000::1), and work in VRRP preemption mode. Since the priority of Qtech A in the IPv6 VRRP group is 120 and that of Qtech B is the default 100, Qtech A works as the master router of the IPv6 VRRP group in normal cases.

Verification

Run the **show ipv6 vrrp 1** command to check VRRP configuration information after the configuration is complete.

# Show configurations on Qtech A

```
Qtech#show ipv6 vrrp 1
FastEthernet 0/1 - Group 1
  State is Master
  Virtual IPv6 address is as follows:
FE80::1
2000::1
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 120
  Master Router is FE80::1234 (local), priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.59 sec
```

# Show configurations on Qtech B

```
Qtech#show ipv6 vrrp 1
FastEthernet 0/1 - Group 1
  State is Backup
  Virtual IPv6 address is as follow:
FE80::1
2000::1
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 100
  Master Router is FE80::1234, priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.82 sec
```

4.3.4 Example of Configuring the Tracked Interface of an IPv6 VRRP Group

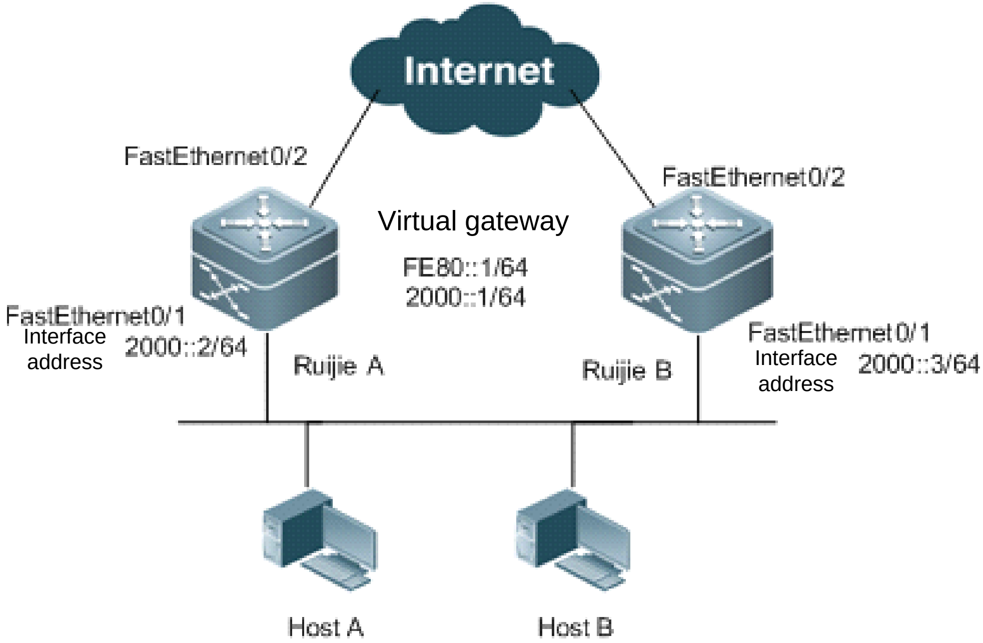
4.3.5 Networking Requirements

This configuration instance is applicable to both switches and routers.

- Host A and Host B access the Internet through a gateway. The default gateway is 2000::1/64 on both hosts.
- Qtech A and Qtech B are two routers that form IPv6 VRRP group 1. The virtual addresses are 2000::1/64 and FE80::1.
- Qtech A tracks the interface FastEthernet 0/2 to the Internet. When the interface FastEthernet 0/2 is unavailable, Qtech A decreases its VRRP group priority so that Qtech B serves as the master router to provide the gateway function.

Networking Topology

Figure 6 Network topology of the example for configuring the tracked interface of an IPv6 VRRP group



Configuration Steps

Configurations on Qtech A:

# Configure an IPv6 address on an interface to enable the IPv6 service on the interface.

```
interface FastEthernet 0/0
```

/* The **no switchport** command needs to be run on a switch only*/

```
no switchport
ipv6 address 2000::2/64
!
```

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
vrrp 1 ipv6 FE80::1
vrrp 1 ipv6 2000::1
!
```

# Change the priority of IPv6 VRRP group 1 to 120.

```
vrrp ipv6 1 priority 120
!
```

# Change the advertisement interval of IPv6 VRRP group 1 to 3s.

```
vrrp ipv6 1 timers advertise 3
!
```

# Configure the tracked interface FastEthernet 0/2 for IPv6 VRRP group 1.

```
vrrp ipv6 1 track FastEthernet 0/2 50
```

# Set the Accept_Mode of the IPv6 VRRP group.

```
vrrp ipv6 1 accept_mode
```

Configurations on Qtech B:

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
```

/* The **no switchport** command needs to be run on a switch only*/

```
no switchport
```

```
ipv6 address 2000::3/64
!
```

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
vrrp 1 ipv6 FE80::1
vrrp 1 ipv6 2000::1
```

# Change the priority of IPv6 VRRP group 1 to 100.

```
vrrp ipv6 1 priority 100
!
```

# Change the advertisement interval of IPv6 VRRP group 1 to 3s.

```
vrrp ipv6 1 timers advertise 3
```

# Set the Accept_Mode of the IPv6 VRRP group.

```
vrrp ipv6 1 accept_mode
```

As can be seen, Qtech A and Qtech B belong to IPv6 VRRP group 1, point to the same virtual router IPv6 address (2000::1), and work in IPv6 VRRP preemption mode. Since the priority of Qtech A in the IPv6 VRRP group is 120 and that of Qtech B is the default 100, Qtech A works as the master router of the IPv6 VRRP group in normal cases. If Qtech A detects that its interface FastEthernet 0/2 is unavailable, it decreases its VRRP group priority by 50 to 70, so that Qtech B becomes the master router. If Qtech A detects later that its interface FastEthernet 0/2 is available again, it increases its VRRP group priority by 50 to 120, so that Qtech A again becomes the master router.

Verification

Run the **show ipv6 vrrp 1** command to check VRRP configuration information after the configuration is complete.

# Show configurations on Qtech A

```
Qtech#show ipv6 vrrp 1
FastEthernet 0/1 - Group 1
  State is Master
  Virtual IPv6 address is as follows:
FE80::1
2000::1
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 120
  Master Router is FE80::1234 (local), priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.59 sec
Tracking state of 1 interface, 1 up:
    up FastEthernet 0/2 priority decrement=50
```

# Show configurations on Qtech B

```
Qtech#show ipv6 vrrp 1
FastEthernet 0/1 - Group 1
  State is Backup
  Virtual IPv6 address is as follow:
FE80::1
2000::1
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 100
  Master Router is FE80::1234, priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.82 sec
```

4.3.6 Example of Configuring Multiple IPv6 VRRP Groups

Multiple VRRP groups can be configured on one Ethernet interface to implement load balancing and backup one another to provide more reliable and stable network services.

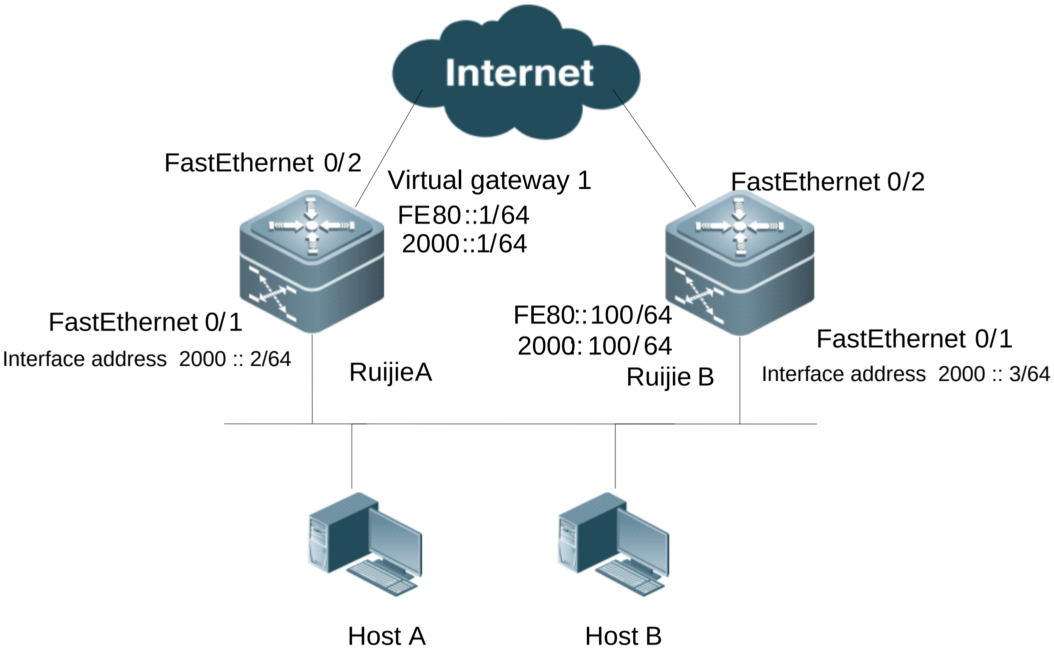
Networking Requirements

This configuration instance is applicable to both switches and routers.

- Host A and Host B access the Internet through gateways. The default gateway for Host A is 2000::1/64, and that for Host B is 2000::100/64.
- Qtech A and Qtech B are two routers that form IPv6 VRRP group 1. The virtual addresses are 2000::1/64 and FE80::1.
- Qtech A and Qtech B also form IPv6 VRRP group 2. The virtual addresses are 2000::100/64 and FE80::100.
- Qtech A and Qtech B serve as gateways to forward traffic and back up each other.

Networking Topology

Figure 7 Network topology of the example for configuring multiple IPv6 VRRP groups



Configuration Steps

Configurations on Qtech A:

# Configure an IPv6 address on an interface to enable the IPv6 service on the interface.

```
interface FastEthernet 0/0
```

/* The **no switchport** command needs to be run on a switch only*/

```
no switchport
ipv6 address 2000::2/64
!
```

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
vrrp 1 ipv6 FE80::1
vrrp 1 ipv6 2000::1
!
```

# Change the priority of IPv6 VRRP group 1 to 120.

```
vrrp ipv6 1 priority 120
!
```

# Change the advertisement interval of IPv6 VRRP group 1 to 3s.

```
vrrp ipv6 1 timers advertise 3
```

# Set the Accept_Mode of IPv6 VRRP group 1.

```
vrrp ipv6 1 accept_mode
!
```

# Create IPv6 VRRP group 2 and configure virtual IPv6 addresses FE80::100 and 2000::100.

```
vrrp 2 ipv6 FE80::100
vrrp 2 ipv6 2000::100
!
```

# Change the priority of IPv6 VRRP group 2 to 100.

```
vrrp ipv6 2 priority 100
```

# Change the advertisement interval of IPv6 VRRP group 2 to 3s.

```
vrrp ipv6 2 timers advertise 3
```

# Set the Accept_Mode of IPv6 VRRP group 2.

```
vrrp ipv6 2 accept_mode
```

Configurations on Qtech B:

```
interface FastEthernet 0/0
```

/* The **no switchport** command needs to be run on a switch only*/

```
no switchport
ipv6 address 2000::3/64
!
```

# Create IPv6 VRRP group 1 and configure virtual IPv6 addresses FE80::1 and 2000::1.

```
interface FastEthernet 0/0
vrrp 1 ipv6 FE80::1
vrrp 1 ipv6 2000::1
```

# Change the priority of IPv6 VRRP group 1 to 100.

```
vrrp ipv6 1 priority 100
!
```

# Change the advertisement interval of IPv6 VRRP group 1 to 3s.

```
vrrp ipv6 1 timers advertise 3
```

# Set the Accept_Mode of IPv6 VRRP group 1.

```
vrrp ipv6 1 accept_mode
!
!
```

# Create IPv6 VRRP group 2 and configure virtual IPv6 addresses FE80::100 and 2000::100.

```
vrrp 2 ipv6 FE80::100
vrrp 2 ipv6 2000::100
!
```

# Change the priority of IPv6 VRRP group 2 to 120.

```
vrrp ipv6 2 priority 120
```

# Change the advertisement interval of IPv6 VRRP group 2 to 3s.

```
vrrp ipv6 2 timers advertise 3
!
```

# Set the Accept_Mode of IPv6 VRRP group 2.

```
vrrp ipv6 2 accept_mode
!
```

As can be seen, Qtech A and Qtech B belong to IPv6 VRRP group 1, point to the same virtual router IPv6 address (2000::1), and work in IPv6 VRRP preemption mode. Since the priority of Qtech A in IPv6 VRRP group 1 is 120 and that of Qtech B is the default 100, Qtech A works as the master router of IPv6 VRRP group 1 in normal cases. In IPv6 VRRP group 2, however, the priority of Qtech A is 100 and that of Qtech B is 120 and IPv6 VRRP group 2 works in preemption mode. Therefore, Qtech B works as the master router of IPv6 VRRP group 2 in normal cases. For hosts in the same LAN, Host A uses IPv6 VRRP group 1 as the default gateway whereas Host B uses IPv6 VRRP group 2 as the default gateway. Route redundancy is implemented between Qtech A and Qtech B, which share LAN traffic and implement load balancing. In this example, default gateways must be manually set on IPv6 hosts to implement load balancing based on IPv6 VRRP groups.

Verification

Run the **show ipv6 vrrp** command to check VRRP configuration information after the configuration is complete.

# Show configurations on Qtech A

```
Qtech#show ipv6 vrrp
FastEthernet 0/1 - Group 1
  State is Master
  Virtual IPv6 address is as follows:
FE80::1
2000::1
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 120
  Master Router is FE80::1234 (local), priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.59 sec
FastEthernet 0/1 - Group 2
  State is Backup
  Virtual IPv6 address is as follows:
FE80::100
2000::100
  Virtual MAC address is 0000.5e00.0202
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 100
  Master Router is FE80::5678, priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.82 sec
```

# Show configurations on Qtech B

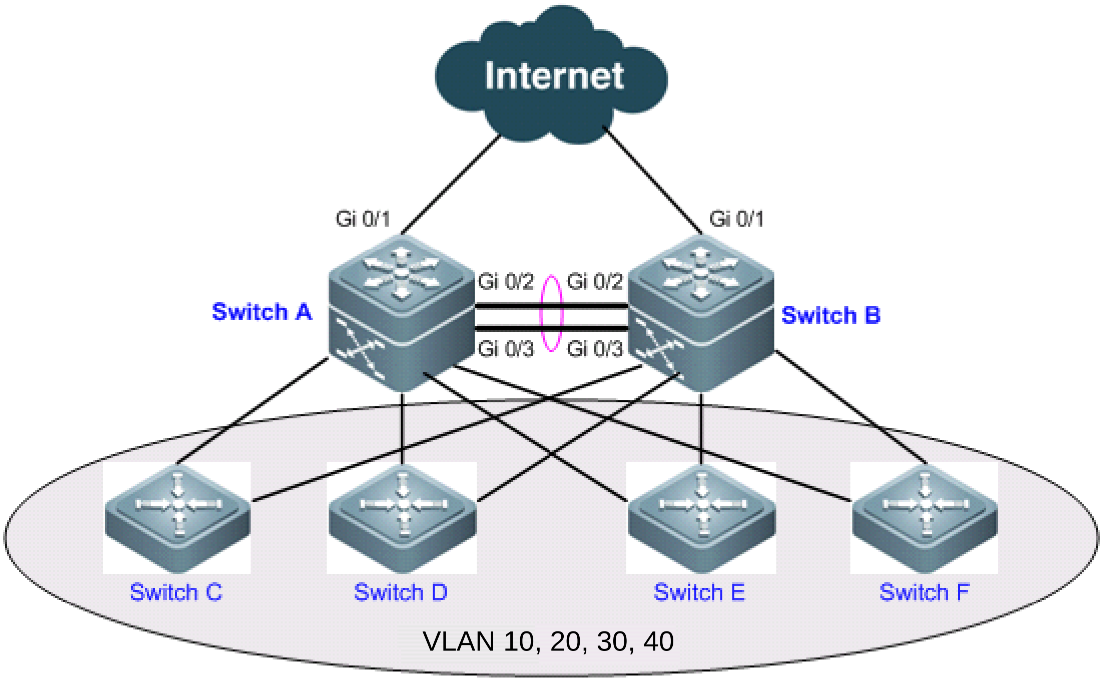
```
Qtech#show ipv6 vrrp 1
FastEthernet 0/1 - Group 1
  State is Backup
  Virtual IPv6 address is as follow:
FE80::1
2000::1
  Virtual MAC address is 0000.5e00.0201
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 100
  Master Router is FE80::1234, priority is 120
  Master Advertisement interval is 3 sec
  Master Down interval is 10.82 sec

FastEthernet 0/1 - Group 2
  State is Master
  Virtual IPv6 address is as follows:
FE80::100
2000::100
  Virtual MAC address is 0000.5e00.0202
  Advertisement interval is 3 sec
  Accept_Mode is enabled
  Preemption is enabled
    min delay is 0 sec
  Priority is 120
  Master Router is FE80::5678(local), priority is 120
```

```
 Master Advertisement interval is 3 sec
 Master Down interval is 10.59 sec
```

4.3.6.1 Configuring VRRP+MSTP

Networking Topology

Figure 8 Network topology of the VRRP dual-core solution



Networking Requirements

Figure 8 shows a typical network topology of the dual-core solution. This configuration instance is applicable to switches or switching cards of routers only. Users access Switches C, D, E, and F, which belong to VLAN 10, 20, 30, and 40 respectively. Switches A and B serve as gateways to enable users to communicate with external networks. The specific application requirements are described as follows:

- The Multiple Spanning Tree Protocol (MSTP) runs on devices to back up physical links and avoid loops. Different VLAN packets are forwarded along respective instances to implement layer 2 traffic load balancing.
- VRRP runs on devices to back up gateway routes and share LAN traffic.
- All links from access switches to the master router are monitored. When a link to the master router fails, the backup router immediately takes over the master router to forward data.

Configuration Tips

- Enable the MSTP function on devices (Switches A, B, C, D, E, and F in this example), configure mappings between VLANs and instances (VLANs 10 and 20 map to instance 1, VLANs 30 and 40 map to instance 2, and the rest VLANs map to instance 0 in this example), and set gateways (Switches A and B in this example) as the root bridges of respective instances.
- Add the switch virtual instances (SVIs) of various VLANs to respective VRRP groups, and set the master and backup routers of respective VRRP groups as gateways. The following table shows the specific configurations.

Gateway	VLAN ID	SVI	VRRP Group	Virtual IP Address	Status
Switch A	10	192.168.10.2	VRRP 10	192.168.10.1	Master
Switch B		192.168.10.3			Backup
Switch A	20	192.168.20.2	VRRP 20	192.168.20.1	Master
Switch B		192.168.20.3			Backup
Switch A	30	192.168.30.2	VRRP 30	192.168.30.1	Backup

Gateway	VLAN ID	SVI	VRRP Group	Virtual IP Address	Status
Switch B		192.168.30.3			Master
Switch A	40	192.168.40.2	VRRP 40	192.168.40.1	Backup
Switch B		192.168.40.3			Master

- Set the uplink ports on the master routers of VRRP groups as the tracked interfaces of the master routers. In this example, the tracked interfaces are two ports Gi 0/1 on Switches A and B.

Caution

When setting the tracked interface of a VRRP group, ensure that the value of the *Priority decrement* parameter is larger than the difference between the priority of the master router and the priority of the backup router. The system automatically decreases or increases the priority value of a router according to the status of the tracked interface on the router.

Configuration Steps

In this example, only VRRP+MSTP configurations on Switches A and B are listed. This example does not provide details about how to define VLANs on Switches C, D, E, and F or how to configure MSTP on devices. For details about MSTP configuration, see *MSTP Configuration*.

- Step 1: Create VLANs on devices.

! Create VLANs 10, 20, 30, and 40 on Switch A:

```
SwitchA#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SwitchA(config)#vlan range 10,20,30,40
SwitchA(config-vlan-range)#exit
```

! Configure the same as above on Switch B.

- Step 2: Configure the Multiple Spanning Tree (MST) domain.

! Configure mappings from VLANs 10 and 20 to instance 1, from VLANs 20 and 30 to instance 2, and from the rest VLANs to instance 0 on Switch A.

```
SwitchA(config)#spanning-tree mst configuration
SwitchA(config-mst)#instance 1 vlan 10,20
%Warning: you must create vlans before configuring instance-vlan relationship
SwitchA(config-mst)#instance 2 vlan 30,40
%Warning: you must create vlans before configuring instance-vlan relationship
SwitchA(config-mst)#exit
```

! Configure the same as above on Switch B.

Step 3: Set Switch A as the root bridges of MST instances 0 and 1, and set Switch B as the root bridge of MST instance 2.

! Set the priority of MST instances 0 and 1 to 4096 and that of MST instance 2 to 8192 on Switch A.

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

! Set the priority of MST instances 0 and 1 to 8192 and that of MST instance 2 to 4096 on Switch B.

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

- Step 4: Enable MSTP.

! Enable MSTP on Switch A.

```
SwitchA(config)#spanning-tree
Enable spanning-tree.
```

! Configure the same as above on Switch B.

- Step 5: Configure the SVIs of VLANs, add the SVIs to VRRP groups, and set the virtual IP addresses of VRRP groups. For details, see the preceding table.

! Configurations on Switch A:

```
SwitchA(config)#interface vlan 10
SwitchA(config-if-VLAN 10)#ip address 192.168.10.2 255.255.255.0
SwitchA(config-if-VLAN 10)#vrrp 10 ip 192.168.10.1
SwitchA(config-if-VLAN 10)#exit
SwitchA(config)#interface vlan 20
SwitchA(config-if-VLAN 20)#ip address 192.168.20.2 255.255.255.0
SwitchA(config-if-VLAN 20)#vrrp 20 ip 192.168.20.1
SwitchA(config-if-VLAN 20)#exit
SwitchA(config)#interface vlan 30
SwitchA(config-if-VLAN 30)#ip address 192.168.30.2 255.255.255.0
SwitchA(config-if-VLAN 30)#vrrp 30 ip 192.168.30.1
SwitchA(config-if-VLAN 30)#exit
SwitchA(config)#interface vlan 40
SwitchA(config-if-VLAN 40)#ip address 192.168.40.2 255.255.255.0
SwitchA(config-if-VLAN 40)#vrrp 40 ip 192.168.40.1
SwitchA(config-if-VLAN 40)#exit
```

! Configurations on Switch B:

```
SwitchB(config)#interface vlan 10
SwitchB(config-if-VLAN 10)#ip address 192.168.10.3 255.255.255.0
SwitchB(config-if-VLAN 10)#vrrp 10 ip 192.168.10.1
SwitchB(config-if-VLAN 10)#exit
SwitchB(config)#interface vlan 20
SwitchB(config-if-VLAN 20)#ip address 192.168.20.3 255.255.255.0
SwitchB(config-if-VLAN 20)#vrrp 20 ip 192.168.20.1
SwitchB(config-if-VLAN 20)#exit
SwitchB(config)#interface vlan 30
SwitchB(config-if-VLAN 30)#ip address 192.168.30.3 255.255.255.0
SwitchB(config-if-VLAN 30)#vrrp 30 ip 192.168.30.1
SwitchB(config-if-VLAN 30)#exit
SwitchB(config)#interface vlan 40
SwitchB(config-if-VLAN 40)#ip address 192.168.40.3 255.255.255.0
SwitchB(config-if-VLAN 40)#vrrp 40 ip 192.168.40.1
SwitchB(config-if-VLAN 40)#exit
```

- Step 6: Configure the master and backup routers of VRRP groups.

! Raise the priority of VRRP groups 10 and 20 on Switch A to 120, so that Switch A works as the master router of VRRP groups 10 and 20.

```
SwitchA(config)#interface vlan 10
SwitchA(config-if-VLAN 10)#vrrp 10 priority 120
SwitchA(config-if-VLAN 10)#exit
SwitchA(config)#interface vlan 20
SwitchA(config-if-VLAN 20)#vrrp 20 priority 120
SwitchA(config-if-VLAN 20)#exit
```

! Similarly, raise the priority of VRRP groups 30 and 40 on Switch B to 120.

```
SwitchB(config)#interface vlan 30
SwitchB(config-if-VLAN 30)#vrrp 30 priority 120
SwitchB(config-if-VLAN 30)#exit
SwitchB(config)#interface vlan 40
SwitchB(config-if-VLAN 40)#vrrp 40 priority 120
SwitchB(config-if-VLAN 40)#exit
```

- Step 7: Set the uplink ports on the master routers of VRRP groups as the tracked interfaces of VRRP groups. Ensure that the configured tracked interfaces are L3 interfaces.

! Set the port Gi 0/1 on Switch A as a route port and its IP address to 10.10.1.1/24.

```
SwitchA(config)#interface gigabitEthernet 0/1
SwitchA(config-if-GigabitEthernet 0/1)#no switchport
SwitchA(config-if-GigabitEthernet 0/1)#ip address 10.10.1.1 255.255.255.0
SwitchA(config-if-GigabitEthernet 0/1)#exit
```

! Set the port Gi 0/1 on Switch A as the tracked interface of VRRP groups 10 and 20, and *Priority decrement* to 30.

```
SwitchA(config)#interface vlan 10
SwitchA(config-if-VLAN 10)#vrrp 10 track gigabitEthernet 0/1 30
SwitchA(config-if-VLAN 10)#exit
SwitchA(config)#interface vlan 20
SwitchA(config-if-VLAN 20)#vrrp 20 track gigabitEthernet 0/1 30
SwitchA(config-if-VLAN 20)#exit
```

! Set the port Gi 0/1 on Switch B as a route port and its IP address to 10.10.2.1/24.

```
SwitchB(config)#interface gigabitEthernet 0/1
SwitchB(config-if-GigabitEthernet 0/1)#no switchport
SwitchB(config-if-GigabitEthernet 0/1)#ip address 10.10.2.1 255.255.255.0
SwitchB(config-if-GigabitEthernet 0/1)#exit
```

! Set the port Gi 0/1 on Switch B as the tracked interface of VRRP groups 30 and 40, and *interface-priority* to 30.

```
SwitchB(config)#interface vlan 30
SwitchB(config-if-VLAN 30)#vrrp 30 track gigabitEthernet 0/1 30
SwitchB(config-if-VLAN 30)#exit
SwitchB(config)#interface vlan 40
SwitchB(config-if-VLAN 40)#vrrp 40 track gigabitEthernet 0/1 30
SwitchB(config-if-VLAN 40)#exit
```

- Step 8: Set the interconnection ports between the two core devices (Switches A and B) as aggregation ports.

! Configurations on Switch A:

# Set ports Gi 0/2 and Gi 0/3 as aggregation ports, which serve as trunk ports.

```
Qtech#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Qtech(config)#interface range gigabitEthernet 0/2-3
Qtech(config-if-range)#port-group 1
Qtech(config)#interface aggregateport 1
Qtech(config-if-AggregatePort 1)#switchport mode trunk
```

! Configure the same as above on Switch B.

Verification

- Step 1: Check configuration information on devices.

! Check configuration information on Switch A.

```
SwitchA#show running-config
!
vlan 10
!
vlan 20
!
vlan 30
!
vlan 40
!
spanning-tree
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 20
 instance 2 vlan 30, 40
spanning-tree mst 0 priority 4096
spanning-tree mst 1 priority 4096
spanning-tree mst 2 priority 8192
interface GigabitEthernet 0/1
 no switchport
 no ip proxy-arp
 ip address 10.10.1.1 255.255.255.0
!
interface GigabitEthernet 0/2
 port-group 1
!
interface GigabitEthernet 0/3
 port-group 1
!
interface AggregatePort 1
 switchport mode trunk
!
```

```
interface VLAN 10
 no ip proxy-arp
 ip address 192.168.10.2 255.255.255.0
 vrrp 10 priority 120
 vrrp 10 ip 192.168.10.1
 vrrp 10 track GigabitEthernet 0/1 30
!
interface VLAN 20
 no ip proxy-arp
 ip address 192.168.20.2 255.255.255.0
 vrrp 20 priority 120
 vrrp 20 ip 192.168.20.1
 vrrp 20 track GigabitEthernet 0/1 30
!
interface VLAN 30
 no ip proxy-arp
 ip address 192.168.30.2 255.255.255.0
 vrrp 30 ip 192.168.30.1
!
interface VLAN 40
 no ip proxy-arp
 ip address 192.168.40.2 255.255.255.0
 vrrp 40 ip 192.168.40.1
```

! Check configuration information on Switch B.

```
SwitchB#show running-config
!
vlan 10
!
vlan 20
!
vlan 30
!
vlan 40
!
spanning-tree
spanning-tree mst configuration
 instance 0 vlan 1-9, 11-19, 21-29, 31-39, 41-4094
 instance 1 vlan 10, 20
 instance 2 vlan 30, 40
spanning-tree mst 0 priority 8192
spanning-tree mst 1 priority 8192
spanning-tree mst 2 priority 4096
interface GigabitEthernet 0/1
 no switchport
 no ip proxy-arp
 ip address 10.10.2.1 255.255.255.0
!
interface GigabitEthernet 0/2
 port-group 1!
interface GigabitEthernet 0/3
 port-group 1
!
interface AggregatePort 1
 switchport mode trunk
!
interface VLAN 10
 no ip proxy-arp
 ip address 192.168.10.3 255.255.255.0
 vrrp 10 ip 192.168.10.1
!
interface VLAN 20
 no ip proxy-arp
 ip address 192.168.20.3 255.255.255.0
 vrrp 20 ip 192.168.20.1
!
interface VLAN 30
 no ip proxy-arp
 ip address 192.168.30.3 255.255.255.0
 vrrp 30 priority 120
 vrrp 30 ip 192.168.30.1
 vrrp 30 track GigabitEthernet 0/1 30
!
interface VLAN 40
 no ip proxy-arp
 ip address 192.168.40.3 255.255.255.0
 vrrp 40 priority 120
 vrrp 40 ip 192.168.40.1
 vrrp 40 track GigabitEthernet 0/1 30
```

- Step 2: Check the VRRP status of each device.

! Check the VRRP status of Switch A.

```
SwitchA#show vrrp brief
Interface  Grp   Pri   timer  Own   Pre   State    Master addr    Group addr
VLAN 10    10    120   3      -     P     Master   192.168.10.2   192.168.10.1
VLAN 20    20    120   3      -     P     Master   192.168.20.2   192.168.20.1
VLAN 30    30    100   3      -     P     Backup   192.168.30.3   192.168.30.1
VLAN 40    40    100   3      -     P     Backup   192.168.40.3   192.168.40.1
```

! Check the VRRP status of Switch B.

```
SwitchB#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    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    120   3      -     P     Master   192.168.40.3   192.168.40.1
```

As can be seen from above, Switch A serves as the master router of VRRP groups 10 and 20 and has a priority of 120 when links are normal, whereas Switch B serves as the backup routers of VRRP groups 10 and 20.

- Step 3: Disconnect the uplink of Switch A, and then check the VRRP status of Switches A and B.

! Check the VRRP status of Switch A.

```
SwitchA#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    90    3      -     P     Backup   192.168.20.3   192.168.20.1
VLAN 30    30    100   3      -     P     Backup   192.168.30.3   192.168.30.1
VLAN 40    40    100   3      -     P     Backup   192.168.40.3   192.168.40.1
```

! Check the VRRP status of Switch B.

```
SwitchB#show vrrp brief
Interface  Grp   Pri   timer  Own   Pre   State    Master addr    Group addr
VLAN 10    10    100   3      -     P     Master   192.168.10.3   192.168.10.1
VLAN 20    20    100   3      -     P     Master   192.168.20.3   192.168.20.1
VLAN 30    30    120   3      -     P     Master   192.168.30.3   192.168.30.1
VLAN 40    40    120   3      -     P     Master   192.168.40.3   192.168.40.1
```

As can be seen from above, when the uplink of Switch A fails, the system automatically decreases the priority of VRRP groups 10 and 20 to 90 and changes the VRRP status of Switch A to Backup, so that Switch B becomes the master router of VRRP groups 10 and 20.

4.4 Fault Diagnosis and Clearance

You can analyze and clear VRRP faults by checking configuration and debugging information. Below are some common faults and analysis methods:

Symptom

The virtual IPv4/IPv6 address of a VRRP group cannot be pinged.

Analysis

- Ensure that at least one router is active in the VRRP group.
- If the virtual IPv4/IPv6 address cannot be pinged from a device on another network, possibly the fault is caused because a short time is needed for VRRP status switching. Run the **show [ipv6] vrrp** command to check VRRP information and verify the root cause.
- If the virtual IPv4/IPv6 address cannot be pinged from a local network device in the same network segment as the virtual router, check whether the ARP table or neighbor discovery (ND) table on the local network device contains an ARP entry for the virtual IPv4/IPv6 address. If there is no ARP entry for the virtual IPv4/IPv6 address, check network lines.
- If the virtual IPv4/IPv6 address cannot be pinged from a local network device in a network segment that is different from the network segment where the virtual router resides, check whether a route to the virtual IPv4/IPv6 address has been configured on the local network device.

Symptom

Multiple master routers exist in a VRRP group.

Analysis

- Ethernet interfaces on the routers of the VRRP group are set to different VRRP group authentication modes.
- For VRRPv2, Ethernet interfaces on the routers of the VRRP group are set to the same plain text authentication mode but the configured authentication strings are inconsistent.
- The cables of the Ethernet interfaces on the routers of the VRRP group are disconnected, but the routers fail to detect the disconnection.
- The VRRP advertisement intervals configured on routers of the VRRP group are inconsistent, and the periodic learning of VRRP advertisement packets is disabled on the routers.
- Different virtual IPv4/IPv6 addresses are set for routers of the VRRP group.

5 CONFIGURING HOT SWAPPING

5.1 Understanding Hot Swap

5.1.1 Overview

Hot swap refers to the ability of allowing removing a faulty line card and inserting the standby line card in a high-reliable system without restarting and shutting down the system.

Qtech designs hot-swap products by complying with the rule of separating software settings from hardware application and separating the existence of line cards from the startup of line cards.

Qtech employs the install/no install concept so that users can run the **install** command to pre-install and pre-configure a certain type of line card in a slot even if no line card is inserted in the slot. Management software can reserve all configurations of the line card. Whether to configure hardware depends on actual situations.

After users save settings, all pre-configuration information is stored. After routers are reset, this pre-configuration information is still effective.

During normal running of routers, removing and then inserting a line card will not lead to the loss of related hardware settings.

If the actual type of the newly inserted line card is different from the pre-configured type, the inserted line card is not enabled and users need to uninstall the original configuration to enable the line card.

The concept of “Two Hosts” is introduced in RSR30-X series routers. Every RSR30-X series router includes an LPU expansion box like an LPU-4HNM expansion box. One such LPU expansion box is under the management of two PSP30-X hosts with one host as a MASTER and another as SLAVE. MASTER serves as the major manager and SLAVE as the standby manager. When MASTER is faulty, SLAVE manages LPU. In this way, two hosts guarantee the reliability of LPU.

⚠ Caution

Comply with the following rules during hot swap to avoid causing abnormalities of software and hardware or even damaging line cards:

Operation Limits	Description
Remove Line Cards	1. Run the **remove** or **no install** commands, and remove line cards after the commands are executed. Print log similar to the one below to confirm that the **remove** or **no install** commands are fully executed. *May 17 10:24:03: %HOTPLUG-5-DISABLE_LINE_CARD_BEGIN: Begin to disable the line card in slot 1/0. *May 17 10:24:07: %HOTPLUG-5-DISABLE_LINE_CARD_OK: Disable line card in slot 1/0 OK. 2. Remove one line card only at a time; 3. Smoothly and securely remove line cards; 4. During system startup, the line card must not be inserted; 5. Print log similar to the one below to confirm that line cards are removed. *May 17 10:28:38: %HOTPLUG-5-PULL_LINE_CARD_BEGIN: Begin process PULL event in slot 1/0. *May 17 10:28:38: %HOTPLUG-5-PULL_LINE_CARD_OK: PULL event in slot 1/0 process OK.
Insert Line Cards	1. Line cards can only be inserted into slots after the completion of execution of the hot swap commands or previous removal of line cards. 2. Insert one line card only at a time;

	3. Smoothly and securely insert line cards; 4. During system startup, the line card must not be inserted; 5. Line cards are only allowed to be inserted again after they are completely removed (slots and line cards are fully disconnected).

5.2 Configuring Hot Swap

5.2.1 Hot Swap Configuration Task List

Hot swap can be configured by following the task list below:

5.2.2 Installing and Uninstalling a Line Card Module

The pre-configuration function of Qtech products allows users to run the **install** command to virtualize a line card module of the specified type (the line card module is not actually inserted in the slot), and then configure the line card module. After the line card module is inserted in the slot, all configuration automatically takes effect.

Command	Function
Qtech(config)# **install** *slot-num moduletype*	Installs a line card module.
Qtech(config)# **no install** *slot-num*	Uninstalls a line card module.

5.2.3 Inserting and Removing a Line Card Module

Follow the rules in the "Overview" section to insert and remove a line card.

You do not need to configure a command to insert a line card.

Before removing a line card, run the **remove** command.

Command	Function
Qtech(config)# **remove** *slot-num*	Removes a line card.
Qtech(config)# **no remove** *slot-num*	Restores line card configuration.

Note

When you run the **no remove** command, or directly insert the NMX-8E1/CE1 or NMX-4E1/CE1 line card, the communication over the interface that is not added to the fast forwarding group is interrupted for about 30 seconds. Therefore, it is not recommended that you carry out such operation during peak hours of communication.
During system startup, the line card must not be inserted or removed.
A line card can be inserted and removed only after the system is started completely (namely, the console prompts “%SYS-5-WARMSTART:System warmstart.” or “%SYS-5-WARMSTART:System coldstart.”).

5.2.4 Resetting a Line Card Module Hot Swap

Hot swap resetting includes a series of operations: run the **remove** command, remove a line card, run the **no install** command, run the **install** command, and insert the line card. After the hot swap resetting is complete, line card hardware is reset and configuration information on the software is initialized again.

Command	Function
Qtech(config)# **reset** *slot-num*	Resets a line card module hot swappable.

5.2.5 Monitoring and Maintaining Hot Swap

The **show version slots** command is used to display the information of the line card modules in each slot, including line card types configured by users, actual line card types, and hot swap status of line cards.

Command	Function
Qtech# **show version slots, or** Qtech(config)#**show version slots**	Display information about line-card modules in all slots, including the type of the line cards configured by users, the type of the inserted line cards, the hos swap status of the line cards, and information about MASTER and SLAVE. 3 status of the management board: none- No management board is inserted in the slot; slave- The management board in the slot is SLAVE. MASTER- The management board in the slot is MASTER. 8 status of line cards: ● None: Neither a line card nor a pre-configured line card is in the slot. ● Installed: No line card is in the slot, but a line card is configured. ● running-config: MASTER is downloading configuration for the line card. ● running: The line card is running properly. ● run-remove: The user has run the **remove** command but has not removed the line card. ● conflict: The inserted line card is not the type pre-configured by the user. ● unins-remove: The user has run the **no install** command but has not removed the line card. ● standby: this status only works for the LPU expansion box and the device is SLAVE.

5.2.5.1 *MASTER/SLAVE Switchover of LPU Expansion Boxes*

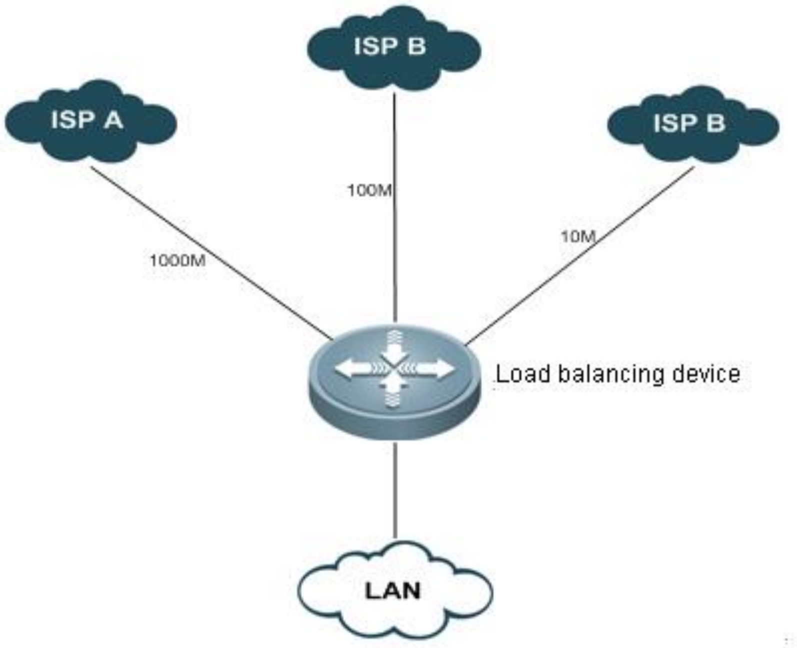
Under the mode of “Two Hosts”, when MASTER is faulty, MASTER/SLAVE switchover will happen automatically. However, manual switchover is demanded in some cases, such as demonstration of the function. So, the **switchover** command is offered to serve this aim.

Command	Function
Qtech(config)#**switchover** *slot-num*	Realizes manual switchover between MASTER and SLAVE.

6 CONFIGURING MULTILINK GATEWAY LOAD BALANCING

6.1 Introduction to Multilink Gateway Load Balancing

Fig 1 Typical application diagram of multilink gateway load balancing



6.1.1 Overview

The network gateway is generally connected with two or more ISP links. For example, the gateway of an educational institution will be connected with an education communication link and a Telecom/CNC link; the gateway of a governmental agency may be connected with a Telecom link and a CNC link. Multiple ISP links handles traffic as per certain policy or act as the backup link.

Multilink load balancing allows reasonable flow distribution among multiple links as per certain policy, well improving the utilization efficiency of link resources.

6.1.2 Basic Concept

6.1.2.1 Link Bandwidth

Link bandwidth is the indicator for measuring available resources, and is different from the transmission rate of physical interface. It is the maximum transmission rate provided by ISP, and generally refers to the inbound bandwidth.

6.1.2.2 Link Latency

When there are multiple gateways, packets can reach the same destination address through different gateways. The link latency used in load balancing refers to the different response times when packets reach the same destination address through different gateways. It is used to compare the access speed of different gateways, focusing mainly on the difference between links in terms of latency.

6.1.2.3 Link Load

Link load refers to the current resource utilization rate of the link. It can be calculated by dividing the packet reception rate of physical interface by link bandwidth.

6.1.2.4 Load Balancing Policy

The policy to share traffic between different links. You can choose load balancing according to bandwidth, access speed, link utilization rate, comprehensive link bandwidth, latency, or load.

6.1.3 Working Principle

In accordance with the load balancing policy selected, the system will calculate the weight of respective links according to link bandwidth, latency and load, and regenerate WCMP routes as per the weight of each link. The subsequent traffic will select the gateway according to the routes generated, thus controlling the traffic handled by each link.

6.1.4 Protocol Specification

NA

6.2 Default Configurations

The following table describes the default configurations of multilink gateway load balancing.

Function	Default setting
Configure multilink gateway load balancing	Disabled

6.3 Configuring Multilink Gateway Load Balancing

The following section describes how to configure multilink load balancing:

- (Required) Enable/disable multilink load balancing
- (Optional) Configure link bandwidth
- (Optional) Configure link load threshold
- (Optional) Configure load balancing policy
- (Optional) Configure weight base
- Display configurations

6.3.1 Enabling/Disabling Multilink Load Balancing

By default, multilink load balancing function is disabled on the device. Enter privilege mode and execute the following steps to enable multilink load balancing:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **mllb enable**	Enable multilink load balancing
Qtech(config)# **no mllb enable**	Disable multilink load balancing

Configuration example:

# Enable global multilink load balancing

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# mllb
```

# Disable global multilink load balancing

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# no mllb
```

6.3.2 Configuring Link Bandwidth

To configure the bandwidth of interface, execute the following commands in interface configuration mode:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **interface** *interface-name*	Enter interface configuration mode.
Qtech(config-if)# **bandwidth** *kilobits*	Configure link bandwidth.
Qtech(config-if)# **no bandwidth**	Restore link bandwidth to default value.

Configuration example:

# Configure link bandwidth to 1M:

```
Qtech# configure terminal
Enter configuation commands, one per line.  End with CNTL/Z.
Qtech(config)# interface fastEthernet 0/1
Qtech(config-if)# bandwidth 1000
```

# Remove link bandwidth configurations:

```
Qtech# configure terminal
Enter configuation commands, one per line.  End with CNTL/Z.
Qtech(config)# interface fastEthernet 0/1
Qtech(config-if)# no bandwidth
```

6.3.3 Configuring Link Load Threshold

If the link load exceeds the threshold configured, the system will then no longer use this link as the gateway link for load balancing. This threshold shall apply to all gateway links. The load threshold will be expressed in percentage (1-100).

To configure link load threshold, execute the following steps:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **mllb threshold** *percent*	Configure link load threshold to "percent" (integer between 1-100).
Qtech(config)# **no mllb threshold**	Restore the link load threshold to the default value of 100.

Configuration example:

# Configure link load threshold to 95:

```
Qtech# configure terminal
Enter configuation commands, one per line.  End with CNTL/Z.
Qtech(config)# mllb threshold 95
```

# Restore link load threshold to the default value:

```
Qtech# configure terminal
Enter configuation commands, one per line.  End with CNTL/Z.
Qtech(config)# no mllb threshold
```

6.3.4 Configuring Load Balancing Policy

Load balancing policy provides four different load balancing options for the user. In case of bandwidth based load balancing, the gateway traffic will be shared according to gateway bandwidth; in case of latency based load balancing, the link with shorter latency will handle more traffic and the link with longer latency will handle less traffic; in case of load based load balancing, the traffic will be shared according to the load situation of each gateway link, so that loads are balanced on links; in case of intelligent load balancing, the traffic will be routed by giving comprehensive consideration to bandwidth, latency and load. The user can further configure the weight base of these three factors (see "Configure weight base"), so as to adjust the specific influence of the corresponding factor on flow distribution.

To configure load balancing policy, execute the following steps:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **mllb policy** { **bandwidth** | **latency** | **load** | **intelligent** }	Specify load balancing policy as bandwidth/latency/load/intelligent policy, with bandwidth being the default policy.
Qtech(config)# **no mllb policy**	Restore load balancing policy to default setting

Configuration example:

# Configure load balancing policy to bandwidth:

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# mllb policy bandwidth
```

6.3.5 Configuring Weight Base

This refers to the corresponding weight base for calculating the link weight as per bandwidth, latency and load when intelligent is selected as the load balancing policy. By adjusting the weight base of bandwidth, latency and load, we can change the specific influence of these three factors on the link weight.

Configure the weight base of bandwidth, latency and load according to the following steps.

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **mllb policy intelligent** [**bandwidth** *base1*] [**latency** *base2*] [**load** *base3*] }	Configure the weight bases of bandwidth, latency and load to base1, base2 and base3 respectively (1-100, with 1 being the default value).
Qtech(config)# **no mllb policy intelligent**	Restore all weight bases to default value.

Configuration example:

# Configure load balancing policy to bandwidth, and configure the weight bases of bandwidth, latency and load to 20, 50 and 100 respectively:

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# mllb policy intelligent bandwidth 50 latency 20 load 100
```

Note

Weight base will only take effect when load balancing policy is configured to intelligent.

6.3.6 Displaying Configurations

In privilege mode, execute "show mllbconfig" command to display configurations related to multilink load balancing.

Command	Function
Qtech (config)# **show mllb config**	Display system configurations, including load balancing configurations.

Configuration example:

# In privilege mode, execute "**show running-config**" command to display load balancing configurations:

```
Qtech# show mllb config
muti-link load balance configure:
muti-link load balance state: enabled
muti-link load balance threshold: 95
muti-link load balance policy: intelligent
     bandwidth weight base = 100
     latency weight base = 100
     load weight base = 100
```

6.4 Typical Multilink Load Balancing Configuration Example

6.4.1.1 Networking Requirements

One 1000M telecom link is connected to interface gigabitEthernet 0/1 of the device, while two CNC links (with bandwidth being 100M and 10M respectively) are connected to interface gigabitEthernet 0/2 and interface gigabitEthernet 0/3 of the device.

6.4.1.2 Network Topology

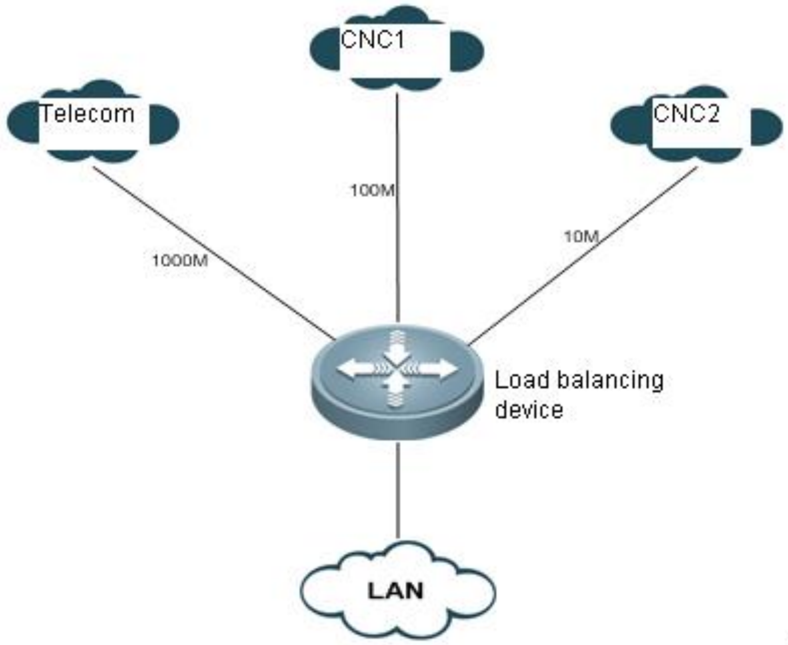


Fig 2 Networking topology of multilink load balancing

6.4.1.3 Configuration Steps

1) Enable multilink load balancing

# Enable global multilink load balancing

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# mllb enable
```

2) Configure link bandwidth

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
```

# Configure the bandwidth of telecom link

```
Qtech(config)# interface gigabitEthernet 0/1
Qtech(config-if)# bandwidth 1000000
Qtech# exit
```

# Configure the bandwidth of CNC link 1

```
Qtech(config)# interface gigabitEthernet 0/2
Qtech(config-if)# bandwidth 100000
Qtech# exit
```

# Configure the bandwidth of CNC link 2

```
Qtech(config)# interface gigabitEthernet 0/3
Qtech(config-if)# bandwidth 10000
Qtech# exit
```

3) Configure load balancing policy to bandwidth load balancing

```
# Configure load balancing policy to bandwidth
Qtech(config)#mllb policy bandwidth
```

4) Configure link load threshold

# Configure link load threshold to 95

```
Qtech(config)# mllb threshold 95
```

7 CONFIGURING RNS & TRACK

7.1 Introduction to RNS

RNS (Qtech Network Service) monitors the integrity of end-to-end connection by detecting whether the reply message is sent by the peer device. The icmp echo packets and dns request packets can be sent.

7.2 List of RNS Configuration Tasks

To configure rns, please follow these tasks:

- Configuring rns to detect icme echo
- Configuring rns to detect dns

7.2.1 Configuring RNS to Detect Icme Echo

General steps:

Configure a RNS object to send ICMP echo packets
1 **enable**
2 **configure terminal**
3 **ip rns** *operation-number*
4 **icmp-echo** destination-hostname [**source-ipaddr** *ip-address*] [**out-interface** *type number* [**next-hop** *nhop-ip*]]
5 **frequency** *milliseconds*
6 **timeout** *milliseconds*
7 **ntime** *number*
8 **exit**

Command	Function
Qtech> **enable**	Enter privilege mode
Qtech# **configure terminal**	Enter global configuration mode
Qtech(config)# **ip rns** *operation-number*	Enter IP RNS configuration mode
Qtech(config-ip-rns)# **icmp-echo destination-hostname** [*source-ipaddr* ip-address] [**out-interface** *type number* [**next-hop** *nhop-ip*]]	Configure an IP RNS object to send ICMP packets, and Enter icmp echo configuration mode If the optional parameter of “out-interface” is applied, note the following points: If out-interface is an Ethernet port, then next-hop has to be configured; if not, then next-hop is optional.
Qtech(config-ip-rns-icmp-echo)# **frequency** *milliseconds*	(Optional) Set an interval of sending packets. The value of the interval should exceed or equal the timeout.
Qtech(config-ip-rns-icmp-echo)# **timeout** *milliseconds*	(Optional) Set timeout of sending packets
Qtech(config-ip-rns-icmp-echo)# **ntime** *number*	(Optional) Set detection times of packets

Configuration example:

```
Qtech> enable
Qtech# configure terminal
```

```
Qtech(config)# ip rns 1
Qtech(config-ip-rns)# icmp-echo 10.1.1.1
Qtech(config-ip-rns-icmp-echo)# ntime 3
```

Display the configurations of RNS object:

```
Qtech# show ip rns configuration
Ip rns id:1
Type of operation to perform: icmp-echo
Target address/Source address:10.1.1.1/0.0.0.0
Out interface/next hop: /0.0.0.0
Operation timeout (milliseconds):5000
Vrf Name:
Operation frequency (milliseconds):60000
Operation ntime: 3
```

Display statistics of RNS object:

```
Qtech# show ip rns statistics
IP rns index    1
Number of successes:0
Number of failures:174
Round-trip min/avg/max = 0/0/0 ms
```

7.2.2 Configuring RNS to Detect DNS

General steps:

Configure a RNS object to send ICMP echo packets

1 **enable**
2 **configure terminal**
3 **ip rns** *operation-number*
4 **dns** *work* **name-server** *a.b.c.d*
5 **frequency** *milliseconds*
6 **timeout** *milliseconds*
7 **ntime** *number*
8 **exit**

Command	Function
Qtech> **enable**	Enter privilege mode
Qtech# **configure terminal**	Enter global configuration mode
Qtech(config)# **ip rns** *operation-number*	Enter IP RNS configuration mode
Qtech(config-ip-rns)# **dns** *work* **name-server** *a.b.c.d*	Configure an IP RNS object to send DNS packets and enter DNS configuration mode.
Qtech(config-ip-rns-dns)# **frequency** *milliseconds*	(Optional) Set an interval of sending packets. The value of the interval should exceed or equal the timeout.
Qtech(config-ip-rns-dns)# **timeout** *milliseconds*	(Optional) Set timeout of sending packets
Qtech(config-ip-rns-dns)# **ntime** *number*	(Optional) Set detection times of packets

Configuration example:

```
Qtech> enable
Qtech# configure terminal
Qtech(config)# ip rns 1
Qtech(config-ip-rns) # dns www.Qtech.com name-server 1.1.1.1
Qtech(config-ip-rns-icmp-echo) # frequency 50000
```

Display the configurations of RNS object:

```
Qtech# show ip rns configuration
Ip rns id:1
Type of operation to perform: dns
Domain name: www.Qtech.com
Name server:1.1.1.1
Operation timeout (milliseconds):9000
Operation frequency (milliseconds):50000
Operation ntime: 1
```

Display statistics of RNS object:

```
Qtech# show ip rns statistics
IP rns index    1
Number of successes:0
Number of failures:8
Round-trip min/avg/max = 0/0/0 ms
```

7.3 Introduction to Track

A track object can track whether an IP address is reachable and whether an interface is UP. The track feature separates a tracked object from the modules which are interested in this object, such as PBR and VRRP. When the states of track object changes, they can take different actions.

7.4 List of Track Configuration Tasks

To configure track, please follow these tasks:

- □Tracking the link state of an interface
- □Tracking the state of a RNS object

7.4.1 Tracking the Link State of an Interface

Execute this task to track the link state of an interface. A layer-2 interface is considered UP as long as the interface is powered up; a layer-3 interface is considered UP as long as the layer-2 interface of this interface-3 interface is UP; a logic interface like loopback interface is considered UP as long as it is not shut down.

General steps:

1. **enable**
2. **configure terminal**
3. **track** *object-number* **interface** *type number* **line-protocol**
4. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
5. **end**
6. **show track** *object-number*

Detailed steps:

Command	Function
Qtech> **enable**	Enter privilege mode
Qtech# **configure terminal**	Enter global configuration mode
Qtech(config)# **track** *object-number* **interface** *type number* **line-protocol**	Track the state of an interface and enter track mode
Qtech(config-track)# **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a delay time after which the state of track object will change when interface state changes. There is no delay by default.
Qtech(config-track)# **show track** *object-number*	(Optional) Display the information about track object. You can use this command to verify whether the configurations are correct.

Configuration example:

```
Qtech> enable
Qtech# configure terminal
Qtech(config)# track 3 interface FastEthernet 1/0 line-protocol
Qtech(config-track)# delay up 30
Qtech(config-track)# show track 3
```

Configure a track object to track the state of an interface. The following example shows relevant information:

```
Qtech# show track 3
Track 3
interface FastEthernet 1/0
The state is Up
1 change,current state last:11 secs
Delay up 10 secs,down 10 secs
```

7.4.2 Tracking the State of a RNS Object

We uses a track object to track the state of RNS object; If the RNS object receives the reply packets, then the state of track object is UP; otherwise, the state of track object is DOWN.

⚠ Caution When track object is used to track a nonexistent RNS object, the state of this track object is UP.

General steps:

First configure an IP RNS object
1. **enable**
2. **configure terminal**
3. **ip rns** *operation-number*
4. **icmp-echo destination-hostname** [*source-ipaddr ip-address*]
5. **frequency** *seconds*
6. **exit**

Then configure a track object:
1. **enable**
2. **configure terminal**
3. **track** *object-number* **rns** *entry-number*
4. **delay up** *seconds* **down** *seconds*
5. **end**
6. **show track object-number**

Configure a route-map and apply the aforementioned track object:
1. **route-map map-tag [permit | deny]** [*sequence-number*]
2. **set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*]

Apply policy-based routing to an interface:
1. **interface** *type number*
2. **ip address** *ip-address* **mask** *[secondary]*
3. **ip policy route-map map-tag**
4. **exit**

Detailed steps:

Command	**Function**
Qtech(config)# **track** *object-number* **rns** *entry-number*	Track the state of an IP RNS object and enter track mode.
Qtech(config-track)# **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a delay time after which the state of track object will change when its state changes. There is no delay by default.
Qtech(config-track)# **exit**	Return to global configuration mode.
Qtech(config)# **interface** *type number*	Enter interface configuration mode.

Qtech(config-if)# **ip address** *ip-address* **mask** [*secondary*]	Configure an IP address for the interface.
Qtech(config-if)# **ip policy route-map map-tag**	Apply policy-based routing to this interface.
Qtech(config-if)# **exit**	Return to global configuration mode.
Qtech(config)# **route-map map-tag** [**permit** | **deny**] [*sequence-number*]	Configure a route-map.
Qtech(config-route-map)# **set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*]	The route map configured is used to track the state of a track object

Configuration example:

```
Qtech(config)# track 123 rns 1
Qtech(config-track)# delay up 30
Qtech(config-track)# exit
Qtech(config)# interface ethernet 0
Qtech(config-if)# ip address 10.1.1.11 255.0.0.0
Qtech(config-if)# ip policy route-map alpha
Qtech(config-if)# exit
Qtech(config)# route-map alpha
Qtech(config-route-map)# set ip next-hop verify-availability 10.1.1.1 10 track 123
```

Display the state of a track object:

```
Track 2
Qtech Network Service 1
The state is Down
1 change,current state last:7 secs
Delay up 30 secs,down 0 secs
```

7.5 Application of Track Feature

We can achieve the following function by tracking the UP/DOWN state of track object:

- Associating the next hop of PBR with a track object

7.5.1 Associating the Next Hop of PBR with a Track Object

In policy-based routing, we can associate the next hop of PBR with a track object. When the state of track object becomes DOWN, this next hop will be disabled, namely PBR will not use this next hop as the next hop for packets.

General steps:

First configure an IP RNS object (please refer RNS configurations given above).

Then configure a track object (please refer to track configurations given above).

Configure a route-map and apply the track object configured above:

1. **route-map map-tag [permit | deny]** [*sequence-number*]
2. **set ip next-hop verify-availability track** *object*

Apply policy-based routing to an interface:

1. **interface** *type number*
2. **ip address** ip-address **mask** [secondary]
3. **ip policy route-map map-tag**
4. **exit**

Detailed steps:

Command	Function
Qtech(config)# **interface** *type number*	Enter interface configuration mode.

Qtech(config-if)# **ip address** *ip-address mask* [*secondary*]	Configure an IP address for the interface.
Qtech(config-if)# **ip policy route-map** *map-tag*	Apply policy-based routing to this interface.
Qtech(config-if)# **exit**	Return to global configuration mode.
Qtech(config)# **route-map** *map-tag* [**permit** | **deny**]	Configure a route-map.
Qtech(config-route-map)# **set ip next-hop verify-availability track** *object*	The route map configured is used to track the state of a track object.

Configuration example:

```
Qtech(config)# interface ethernet 0
Qtech(config-if)# ip address 10.1.1.11 255.0.0.0
Qtech(config-if)# ip policy route-map alpha
Qtech(config-if)# exit
Qtech(config)# route-map alpha
Qtech(config-route-map)# set ip next-hop verify-availability 10.1.1.1  track 123
```

The policy-based routing to be achieved: When packets are received by fa 0/0 and the IP address of 10.1.1.1 is reachable, the next hop of packets is configured as 10.1.1.1 (IP address of the interface on router 2). If 10.1.1.1 is unreachable, the next hop of packets is configured as 10.2.2.2 (IP address of the interface on router 3). If 10.2.2.2 is also unreachable, PBR fails. Packets will be forwarded as per ordinary route according to the query result of core routing table.

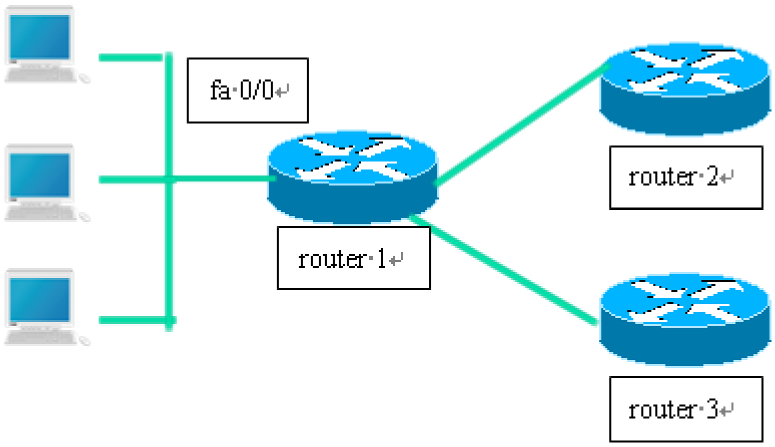


Figure 1

Configurations of Router 1:

# Define two IP RNS objects to track whether the remote IP address is reachable.

```
ip rns 1
icmp-echo 10.1.1.1
ip rns 2
icmp-echo 10.2.2.2
```

# Define track object

```
track 123 rns 1
track 124 rns 2
```

# Apply PBR to the interface

```
interface FastEthernet 0/0
ip address 10.4.4.4 255.255.255.0
ip policy route-map alpha
```

# 10.1.1.1 is the following interface

```
interface fa 0/1
ip address 10.1.1.254 255.255.255.0
```

# 10.2.2.2 is the following interface

```
interface fa 0/2
ip address 10.2.2.254 255.255.255.0
```

# Configure a route-map; the availability of next hop depends on the reachability of track object.

```
route-map alpha
set ip next-hop verify-availability 10.1.1.1  track 123
set ip next-hop verify-availability 10.2.2.2  track 124
```