Руководство пользователя

QSR-2830

Оглавление

1 CONFIGURING IP MULTICAST ROUTING

1.1 Overview

This chapter describes how to configure IPv4 multicast routing protocols. To obtain complete descriptions of multicast routing commands, see the "Multicast Routing Commands" section.

The traditional IP transmission only allows one host to transmit packets to a single host (unicast communication) or all hosts (broadcast communication). Multicast, however, allows one host to send the packets to some hosts (also known as group members).

The destination addresses of packets sent to the group member are Class-D network addresses (224.0.0.0–239.255.255.255). Multicast packets are UDP packets with best effort service. It does not provide reliable transmission and error control as TCP.

The multicast application consists of the sender and receiver. The sender can send multicast packets without needing to join a group. In contrast, the receiver can receive the multicast packets from the group only after joining the group.

Group members are dynamic. A host can join or leave a group at any time. Furthermore, there is no limit on the position or number of group members. A host can join more than one group simultaneously if necessary. Consequently, the active status and the number of members of a group vary from time to time.

Devices run a multicast routing protocol such as the Protocol-Independent Multicasting-Dense Mode (PIM-DM) and the Protocol-Independent Multicasting-Sparse Mode (PIM-SM) to maintain their routing tables to forward multicast messages, and use the Internet Group Management Protocol (IGMP) to learn the status of the members within a group on their directly attached subnets. A host can join or leave an IGMP group by sending IGMP Report messages.

IP multicast applies to one-to-many multimedia applications.

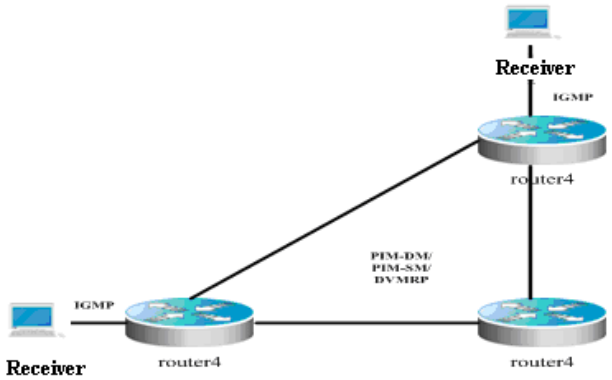
1.1.1 Implementation of IP Multicast Routing

Multicast routing protocols include:

- IGMP: Runs between the routers and the hosts to trace the relationship of group members.
- PIM-DM: A multicast routing protocol in dense mode, which runs between multicast devices to establish the multicast routing table for forwarding.
- PIM-SM: A multicast routing protocol in sparse mode, which runs between multicast devices to establish the multicast routing table for forwarding.
- DVMRP: Distance Vector Multicast Routing Protocol, which runs between multicast devices to establish the multicast routing table for forwarding.

Figure 1 shows IP multicast routing protocols used within the IP multicast environment.

Figure 1 IP Multicast Routing Protocols within the IP Multicast Environment



1.1.2 IGMP

To enable IP multicast, hosts and routers must support the IGMP protocol. This protocol is used by hosts to report their group memberships to multicast routers on the directly-connected network, allowing the multicast routers to determine how to forward multicast traffic.

By using the information obtained from IGMP, multicast routers create an interface-based multicast group member list. The list is activated only when at least one host on an interface is a member of the group. IGMPv1, IGMPv2 and IGMPv3 are currently supported.

1.1.3 IGMPv1

There are only two types of IGMP messages defined in IGMPv1:

- Membership query
- Membership report

A host sends a membership report to indicate that it is interest in joining a group, and the router sends membership queries periodically to ensure that the group has at least one host. When there is no hosts in that group, the device will delete it.

1.1.4 IGMPv2

In IGMPv2, there are only four types of IGMP messages:
- Membership query
- Version 1 membership report
- Version 2 membership report
- Leave group

IGMPv2 is basically the same as IGMPv1, except that IGMPv2 creates a Leave group message for hosts. For IGMPv2, hosts report leave messages to routers which then send queries to check whether there is a host in the multicast group. This makes joining and leaving a group more efficient.

In the multicast network running IGMP, a multicast router is dedicated for sending IGMP query messages. This router is called a querier which is selected through an election mechanism. At first, all routers are queriers. If a router receives a query message from another router with a lower IP address, it becomes a non-querier. Consequently, there is only one querier which has the lowest IP address among all multicast routers on the network.

If a querier is invalid, new querier will be elected. Non-queriers keep a timer for Other Querier Present Interval. Every time when a router receives a membership query packet, it resets the timer. If the timer expires, the router starts to send query messages and selects new querier again.

Queriers must periodically send membership queries to ensure that other routers on the network know that the querier still works. For this purpose, the querier maintains one query interval timer. When it sends membership query messages, this timer will be reset. When the interval timer times out, the querier sends another membership query.

When a new router appers, it sends a series of general query messages to solicit membership information. The number of general query packets depends on the Startup Query Count configured on the router. The initial general query interval is defined by the Startup Query Interval.

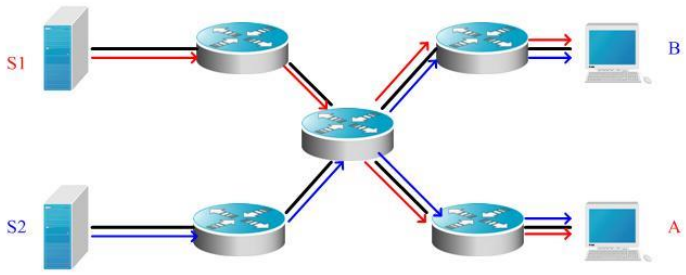
When a querier receives a leave group message from a host, it must send a group-specific membership query to see whether the host is the last one to leave the group. Before the querier stops forwarding packets to the group, it sends a series of such packets, the number of which is equal to the Last Member Query Count. The querier sends multiple group-specific membership queries to ensure that there is no member in the group. Such a query is sent every other the Last Member Query Interval seconds. When no response is received, the querier stops forwarding multicast packets to the group on the specified interface.

1.1.5 IGMPv3

Both IGMPv1 and IGMPv2 have the following defects:
- Lack of efficient measures to control multicast sources
- Difficult to establish multicast paths due to ignorance of multicast source locations.
- Difficult to find a unique multicast address. It is possible that multicast groups are using the same multicast address.

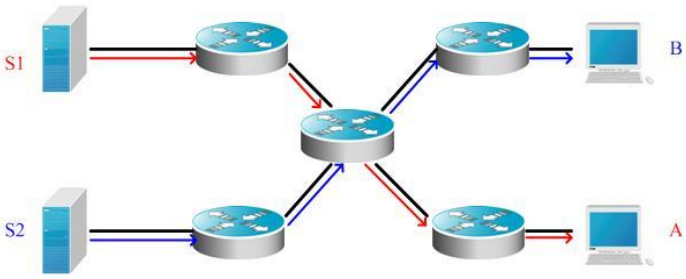
On the basis of IGMPv1 and IGMPv2, IGMPv3 provides an additional source filtering multicast function. In IGMPv1 or IGMPv2, hosts determines whether to join a group by group address and, once it joins the group, it receives multicast traffic forwarded from any source to that group address. In IGMPv3, hosts are enabled to report the multicast group they desire to join in and the multicast source from which they expect to receive traffic. A host specifies sources from which they want to receive multicast traffic through an INCLUDE list or an EXCLUDE list. Besides, IGMPv3 saves bandwidth by preventing unnecessary, illegal multicast data flows from occupying network bandwidth. It is particularly useful in the case where multiple multicast sources share one multicast address. IGMPv1 and IGMPv2 can also implement "source address filtering" in some sense, which, however, is performed on hosts receiving multicast traffic. As shown in the following diagram, two multicast sources (S1 and S2) send out traffic directed to the same multicast group address (G). This multicast traffic from S1 and S2 will be sent to all hosts receiving traffic from G. If host A only wants to receive multicast traffic from S1, it has to filter out traffic from S2 by running approriate client software.

Figure 2 Multicast traffic forwarded without source filtering



If multicast routers on the network support IGMPv3, host A wants to receive traffic from S1 only, it sends out an IGMPv3 packet in the form of “join G include S1”. host B wants to receive traffic from S2 only, it sends out an IGMPv3 packet in the form of “ join G include S2”. In this way, the traffic is forwarded as shown in Figure 3. This saves bandwidth.

Figure 3 Multicast traffic forwarded with source filtering



Based on IGMPv2, IGMPv3 adds the following two kinds of messages:

- Membership query
- Version 3 membership report

There are three types of membership query:

- General Query: used to learn information of all multicast members on an interface.
- Group-Specific Query: used to learn information of members of a specific group on an interface.
- Group-and-Source-Specific Query: a new type specified in IGMPv3 used to learn whether there is a member on an interface wants to receive group-specific multicast traffic from sources in the specified source list.

Membership report in IGMPv3 is different from that defined in IGMPv2. The IGMPv3 membership reports are always sent with an destination address of 224.0.0.22. Besides, an IGMPv3 membership report can contain one or more group records, containing a group address and an list of source addresses. Group records have the following types.

- IS_IN: indicates that the filter mode between a multicast group and the multicast source list is INCLUDE, that is, only multicast data sent from the specified multicast source list to the multicast group is received. If the specified multicast source list is empty, it indicates leaving the multicast group, which is equivalent to the leave packet in IGMPv2.

- IS_EX: indicates that the filter mode between a multicast group and the multicast source list is EXCLUDE, that is, only multicast data sent from the multicast sources not included in the specified multicast source list to the multicast group is received.
- TO_IN: indicates that the filter mode between a multicast group and the multicast source list is changed from EXCLUDE to INCLUDE.
- TO_EX: indicates that the filter mode between a multicast group and the multicast source list is changed from INCLUDE to EXCLUDE.
- ALLOW: indicates that multicast data from certain multicast sources are allowed. If the current relationship is INCLUDE, these multicast sources are added to the existing multicast source list. If the current relationship is EXCLUDE, these multicast sources are deleted from the existing multicast source list.
- BLOCK: indicates that multicast data from certain multicast sources are prohibited. If the current relationship is INCLUDE, these multicast sources are deleted from the existing multicast source list. If the current relationship is EXCLUDE, these multicast sources are added to the existing multicast source list.

For compatibility consideration, IGMPv3 can identify packets of IGMPv1 and IGMPv2.

1.1.6 PIM-DM

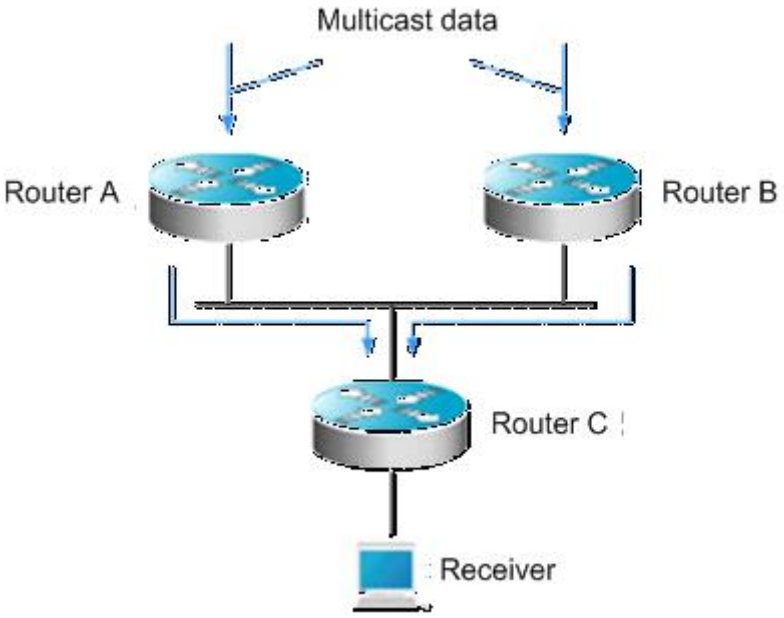
Protocol Independent Multicast-Dense Mode (PIM-DM) is a dense-mode multicast routing protocol, which is suitable for small-sized networks with densely distributed multicast members. As PIM-DM does not rely on any specific unicast routing protocol, it is called protocol independent multicast routing protocol. PIM-DM is defined in RFC 3973.

PIM-DM devices discover neighbors through Hello messages. After startup, a PIM-DM device sends a Hello message to each PIM-DM enabled interface periodically. The Hello message has a field of **Hello Hold Time**, which defines the maximum duration that a neighbor waits for the next message. If the neighbor does not receive another Hello message from the sender within this duration, this device will be removed from the adjacency list.

PIM-DM builds a shortest path tree (SPT) through flood and prune. PIM-DM assumes that when a multicast source begins to send a multicast packet, all the systems in the network need to receive this packet. As a result, this packet is forwarded to every system. The reverse path forwarding (RPF) check is performed for the packets received from the upstream interface. Those packets that fail to pass the check will be discarded. For the packets passing the check, the outgoing interface is computed based on the (S, G) pair of the packets, that is, source address and group address. If the outgoing interface is not null, an outgoing interface entry is created from the (S, G) pair and the multicast packet is forwarded through this outgoing interface. If the computed outgoing interface is null, a prune message is sent to the RPF neighbor, informing the upstream neighbor not to forward the multicast packets from the (S, G) pair to this interface. After the prune message is received on the upstream interface, the device marks the sending interface as pruned state and sets a pruned state timer. In this way, an SPT is created with the multicast source as its root.

PIM-DM uses the Assert mechanism to eliminate redundant routes.

Figure 4 Assert mechanism of PIM-DM



As shown in Figure-4, the multicast data arrives at Routers A and B at the same time, which forward the data to Router C. In this case, Router C receives duplicated data, which is not allowed. So there must be a mechanism to select Router A or B to forward the multicast data to Router C. This is the Assert mechanism of PIM-DM.

PIM-DM uses the state refresh message to update network state. The device directly connected to the multicast source sends the state refresh message to the downstream devices periodically to advertise the network topology changes. The devices receiving the message add their topology state to the state refresh message by modifying some fields, and then send it to the downstream devices. When the refresh message arrives at the leaf devices, the entire network state is updated.

PIM-DM uses the Graft mechanism to reestablish the connection with upstream devices. If the network topology of a downstream device in pruned state changes and the device needs to receive multicast data from a (S, G) pair, it sends the graft message to the upstream device. Upon receiving the graft message, the upstream device responds with a Graft-Ack message and forwards the multicast data to the downstream device again.

1.1.7 DVMRP

The Distance Vector Multicast Routing Protocol (DVMRP) is the first widely used multicast routing protocol in the Internet. It is also in dense mode. Like PIM-DM, DVMRP also uses the reverse path multicast mechanism to establish a distribution tree to forward multicast packets. The difference between the two protocols is that PIM-DM does not rely on specific unicast routing protocols and DVMRP replies on RIP.

A DVMRP device advertises itself, learns neighbor addresses and establishes adjacency through Probe packets. The DVMRP device establishes the adjacency relationship when the Probe packet received from a neighbor contains the IP address of the DVMRP device.

DVMRP neighbors exchange source route information by periodically sending Report packets. The information includes the source network mask and hop count. Such information is stored in the DVMRP routing table that is independent of the unicast routing table and used for RPF check during source tree creation.

DVMRP is also a multicast routing protocol in dense mode and creates SPTs for each multicast source. Initial multicast traffic is forwarded along the entire SPT, but DVMRP does not forward redundant paths. For the specified SPT, the device will send Prune packets to the upstream device after acknowledging that it does not need to receive multicast traffic. The device does not need to receive specified multicast traffic if no downstream neighbor exists and no multicast member information exists. As DVMRP is a multicast routing protocol in dense mode, multicast traffic is redistributed once pruning times out.

In addition, to enable the multicast receiver to join the SPT quickly, DVMRP supports the Graft and Graft-Ack mechanisms. The Graft mechanism adds the pruned paths to the SPT quickly, while the Graft-Ack mechanism avoids loss of Graft messages due to busy networks.

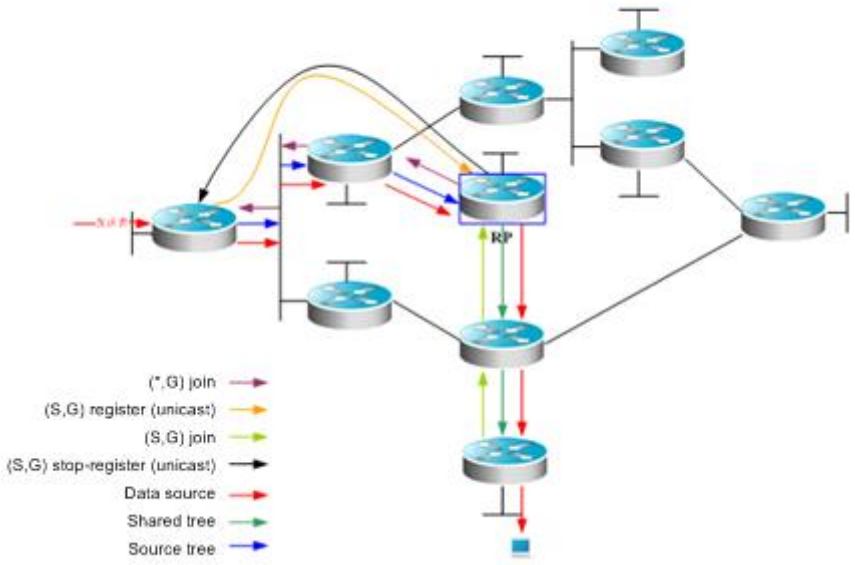
This product supports the complete DVMRP protocol.

1.1.8 PIM-SM

The Protocol Independent Multicast (PIM) is designed by the Inter-Domain Multicast Routing (idmr) working group. As its name implied, PIM does not rely on any specific unicast routing protocol. It can use a unicast routing table established by any unicast routing protocol to perform the RPF check function, instead of maintaining separate multicast routing tables to implement multicast forwarding. As PIM is not required to receive or distribute route updates, compared to other multicast routing protocols, it costs much less. PIM is designed to support shortest path trees (SPTs) and rendezvous point trees (RPTs) simultanenously and enable flexible conversion between them, so that their advantages can be used to improve multicast efficiency. There are two PIM modes: dense mode and sparse mode.

The Protocol Independent Multicast – Sparse Mode (PIM-SM) is a multicast routing protocol of sparse mode. In a PIM-SM domain, the PIM-SM-enabled device periodically sends Hello messages to discover adjacent PIM-SM devices and selects a designated router (DR) in a multi-access network. The DR is responsible for sending Join/Prune messages towards the root of themulticast distribution tree from its directly connected group member, or itsdirectly connected multicast source.

Figure 5 Explicit join mechanism of PIM-SM



PIM-SM forwards multicast data packets by establishing a multicast distribution tree. The multicast distribution tree is divided into two types: Shared Tree that takes the RP of the group G as the root and Shortest Path Tree that takes the multicast source as the root. PIM-SM establishes and maintains the multicast distribution tree by use of the explicit join/prune mechanism. As shown in Figure-5,
When the DR at the receiving end receives a report packet from the receiving end, it sends a (*.G)join packet towards the RP of group G to join the shared tree.
When the DR at the data source receives multicast data from the source host, it encapsulates the multicast data into a register message and unicasts it to the RP. Then the RP will forward the decapsulated data packets to group members along the shared tree.
The RP sends an (S, G)join packet to the first-hop device in the source direction to join the shortest path tree of this source. In this way, the source's packets are sent to the RP without encapsulation along its shortest path tree.
When the first multicast data reaches along the SPT, the RP sends the stop-register message to the DR at the source, notifying the DR of stopping register encapsulation. Afterwards, the DR at the source does not encapsulate register packets but sends them to the RP along its shortest path tree, which then forwards the packets to group members along the shared tree. When no multicast data is required, the DR at the receiving end multicasts the prune message to group G's RP hop by hop to prune the shared tree.

PIM-SM also offers a mechanism of selecting the root point (RP). One or more Candidate-BSRs are configured in a PIM-SM domain. PIM-SM selects a BSR by following a certain rule. There are also Candidate-RPs in a PIM-SM domain that unicast the packets including their IP addresses and available multicast groups to the BSR. The BSR will periodically generate a BSR message which includes a series ofcandidate RPs and corresponding multicast group addresses. The BSR messages are sent hop-by-hop within the entire domain. The device receives and saves these BSR messages. If the DR receives a report on the member relationship of a multicast group from its directly connected host but has no route entries of the multicast group, the DR will use a Hash algorithm to map the multicast group address to a candidate RP that can serve this group. Then, the DR multicasts the Join/Prune message to the RP hop-by-hop. If the DR receives multicast data packets from its directly connected host but has no route entries of the multicast group, the DR will usea Hash algorithm to map the multicast group address to a candidate RP that can serve this group. Then the DR encapsulates multicast data packets into a register message and unicasts it to the RP.

The main difference between PIM-SM and the flood/prune model-based PIM-DM is that PIM-SM is based on the explicit join model. In other words, the receiver sends the join message to the RP, while the router only forwards the packets of that multicast group on the outgoing interface that has joined a multicast group. PIM-SM uses the shared tree to forward multicast packets. Each group has a Rendezvous Point (RP). The multicast source sends data to the RP along the shortest path, and then the RP sends the data to the receivers along the shortest path. This is similar to CBT, but PIM-SM does not use the concept of core. One of the major advantages of PIM-SM is that it not only receives multicast messages through the shared tree but also provides a shared tree-to-SPT conversion mechanism. Such conversion reduces network delay and possible congestion on the RP, but it consumes enormous router resources. So it is suitable for the case where there are only a few multicast data sources and network groups.

PIM-SM uses the shared tree and SPT to distribute multicast frames. At this time, it is assumed that other devices don't want to receive these multicasts unless otherwise stated definitely. When a host joins a group, the equipment connected to the host must notify the root (or the RP) by using the PIM join message. This join message is transferred one after another through the routers to create a shared tree structure. Therefore, the RP records the transfer path and also the register message from the first hop router (DR) of the multicast source, and improves the shared tree upon these two messages. The branch/leaf messages are updated by periodically querying messages. With the shared tree, the multicast source first sends multicast packets to the RP, guaranteeing that all the receivers can receive them. The notation (*.G) represents a tree. The asterisk (*) represents all sources and G represents a specific multicast address. The prune message is also used in the shared tree. That is, the branch/leaf will send prune messages once it is not expecting to receive multicast frames.

PIMv2 BSR is a method of distributing group-to-RP messages to all devices without the need of setting an RP for them. BSR distributes mapping information by propagating BSR messages hop by hop. At first, BSR is selected among routers in the same process as selecting a root bridge based on priority level among layer 2 bridges. Each BSR checks the BSR messages and only forwards those having a priority higher than or equal to its own (higher IP address). The selected BSR sends its BSR message to the all-PIM-routers multicast group (224.0.0.13), where TTL is 1. After the adjacent PIMv2 router receives the message, it multicasts it while setting the TTL to 1. In this way, the BSR message is received by all devices hop by hop. Since the message contains the IP address of the BSR, the candidate BSR can know which router is the current BSR based on this message. The candidate RPs send candidate RP advertisements to announce in which address ranges they can become an RP. The BSR stores them in its local candidate RP cache. The BSP notifies all PIM routers of its local candidate RPs periodically. These messages reach various devices hop by hop in the same way.RPF Check

Multicast routing protocols depend on existing unicast route messages, MBGP routes or static multicast routes to create multicast routing entries. When creating multicast routing entries, multicast routing protocols run the Reverse Path Forwarding (RPF) check mechanism to ensure that multicast packets are transmitted along proper paths while avoiding loops.

RPF check is on the basis of unicast routes, MBGP routes or static multicast routes.

- The unicast routing table summarizes the shortest paths to each destination network segment.
- The MBGP routing table directly offers multicast routes.
- The multicast static routing table lists the RPF route messages that the user configures statically and manually.

1.1.9 RPF Check Process

The multicast routing protocol searches the unicast routing table, the MBGP routing table and the static multicast routing while performing an RPF check. The process is as follows:

1) First of all, select an optimal route from the unicast routing table, the MBGP routing table and the static multicast routing table, respectively.

- Select an optimal route from the unicast routing table for RPF check:

- Use the IP address of the packet source as the destination address to search the unicast routing table and select an optimal unicast route.
- If the unicast route has only one next hop, check whether multicast is enabled on the egress of the next hop.
  - If no, the unicast route is not suitable for RPF check.
  - If yes, the unicast route is suitable for RPF check and the egress serves as the RPF interface.
- If the unicast route has more than one next hop, traverse all next hops and check whether multicast is enabled on the egress of one next hop.
  - If no, traverse the next hop.
  - If yes, the unicast route is suitable for RPF check and the egress serves as the RPF interface.
  - If no multicast is enabled on the egress after all next hops are traversed, there is no unicast route suitable for RPF check.
- If no optimal route exists, there is no unicast route suitable for RPF check.

- Select an optimal route from the MBGP routing table for RPF check:
  - Use the IP address of the packet source as the destination address to search the MBGP routing table and select an optimal MBGP route.
  - The MBGP route has only one next hop. Check whether multicast is enabled on the egress of the next hop.
    - If no, the MBGP route is not suitable for RPF check.
    - If yes, the MBGP route is suitable for RPF check.
  - If no optimal route exists, there is no MBGP route suitable for RPF check.
- Select an optimal route from the static multicast routing table for RPF check:
  - Use the IP address of the packet source as the destination address to search the static multicast routing table and select an optimal static multicast route.
  - If the static multicast route has only one next hop, check whether multicast is enabled on the egress of the next hop.
    - If no, the static multicast route is not suitable for RPF check.
    - If yes, further check whether there is a unicast route available for RPF check.
      - If the next hop does not associate with the unicast protocol number, the static multicast route is suitable for RPF check.
      - If there is no unicast route available for RPF check, the static multicast route is suitable for RPF check.
      - If there is a unicast route available for RPF check, but the unicast protocol number is inconsistent with the one associated with the next hop of the static multicast route, the static multicast route is not suitable for RPF check.
      - If there is a unicast route available for RPF check, and the unicast protocol number is consistent with the one associated with the next hop of the static multicast route, the static multicast route is suitable for RPF check.
  - If the static multicast route has more than one next hop, traverse all next hops and check whether multicast is enabled on the egress of one next hop.
    - If no, traverse the next hop.
    - If yes, further check whether there is a unicast route available for RPF check.
      - If the next hop does not associate with the unicast protocol number, the static multicast route is suitable for RPF check. The outbound interface is the RPF interface.
      - If there is no unicast route available for RPF check, the static multicast route is suitable for RPF check. The outbound interface is the RPF interface.
      - If there is a unicast route available for RPF check, but the unicast protocol number is inconsistent with the one associated with the next hop of the static multicast route, traverse the next hop.
      - If there is a unicast route available for RPF check, and the unicast protocol number is consistent with the one associated with the next hop of the static multicast route, the static multicast route is suitable for RPF check. The outbound interface is the RPF interface.
      - If no multicast is enabled on the egress after all next hops are traversed, there is no static multicast route suitable for RPF check.
- If no optimal route exists, there is no static multicast route suitable for RPF check.

Select one from these three optimal routes for RPF check.

- If the longest match routing rule is configured, select the longest match route; if these three routes are of the same mask, select the one of the highest priority; if they are of the same priority, select the one in the order of static multicast route, MBGP route and unicast route.
- If the longest match routing rule is not configured, select the one of the highest priority; if they are of the same priority, select the one in the order of static multicast route, MBGP route and unicast route.


Caution

The effectiveness of MBGP routes recurs on unicast routes rather than distance. The implementation of current MBGP protocols does not support equal-cost routes.
The effectiveness of static multicast routes recurs on unicast routes rather than distance.
If the static unicast route is selected as RPF route, the route must be configured with the next hop IP address. The PIM protocol will select RPF neighbors based on this next hop IP address. If the static unicast route is not configured with the next hop IP address, the PIM protocol cannot obtain RPF neighbors.

☑ For the commands including the VRF parameters, only the RSR20, RSR30, RSR50 and RSR50E devices support the VRF parameters.

1.2 Configuring IP Multicast Routing

1.2.1 Enabling IP Multicast Forwarding

Multicast data packets and protocol packets can be received and processed by related multicast protocols only after the multicast routing forwarding function is enabled.

Command	Function
Qtech(config)# **ip multicast-routing** [**vrf** *vrf-name*]	Enables multicast routing forwarding. Enable multicast routing based on VRF if it is carried; Enable the default multicast routing globally if VRF is not carried.

1.2.2 Enabling IP Multicast Routing Protocols

Use the following commands to enable the IP multicast function on an interface.

Command	Function
Qtech(config-if)# **ip pim dense-mode**	Enters the interface on which PIM-DM is to be enabled and enables PIM-DM in interface configuration mode. This command must be configured on a layer 3 interface.
Qtech(config-if)# **ip pim sparse-mode**	Enters the interface on which PIM-SM is to be enabled and enables PIM-SM in interface configuration mode. This command must be configured on a layer 3 interface.
Qtech(config-if)# **ip dvmrp enable**	Enters the interface on which DVMRP is to be enabled and enables DVMRP in interface configuration mode. This command must be configured on a layer 3 interface.

The following example shows how to configure PIM-DM on interface GE 0/3:

```
Qtech(config)# ip multicast-routing
Qtech(config)# interface  gigabitEthernet 0/3
Qtech(config-if)# ip address 192.168.194.2 255.255.255.0
Qtech(config-if)# ip pim dense-mode
```


Note

If IPv4 multicast routing is enabled globally, enabling IP multicast routing protocols on an interface will also enable the IGMP function. Only the IP multicast routing protocols of one mode can be enabled on one interface.


Caution

After the layer 3 multicasting is enabled on the Private VLAN and Super VLAN, if the multicast source exists in the sub-VLAN, one more route entry must be duplicated and the ingress is the sub-VLAN in

which the multicast streams enter as the ingress validity check is required for multicast forwarding. As a result, one more multicast hardware entry is occupied, that is, multicast capacity is reduced by 1.

1.2.3 Enabling IGMP

The IGMP protocol is enabled with the enabling of IP multicast route forwarding and IP multicast routing protocols.

1.2.4 Configuring IP Multicast Routing

1.2.5 Configuring TTL Threshold

To limit the TTL of the data packets allowed to pass an interface, configure the TTL threshold.

Use the following command to configure the TTL threshold of multicast packets allowed to pass an interface. Use the **no** form of this command to restore the default value. The default value is **0**.

Command	Function
Qtech (config-if) # **ip multicast ttl-threshold** *ttl-value*	Configures the TTL threshold of an interface. *ttl-value* ranges from 0 to 255.

1.2.6 Limiting the Number of Entries to Be Added to the IP Multicast Routing Table

Use the following command to limit the number of entries to be added to the IP multicast routing table in global configuration mode. Use the **no** form of this command to restore the default value. The default value is **1024**.

Command	Function
Qtech (config) # **ip multicast route-limit** *limit* [*threshold*]	Limits the number of entries to be added to the IP multicast routing table. *limit* specifies the number of entries to be added to the multicast routing table. The range is from 1 to 2147483647. The default is 1024. *threshold* (optional) specifies the number of routes triggering alarm. The default is 2147483647.


Caution

As the hardware is limited for different models, the routes exceeding the hardware entry threshold need to be forwarded through software, resulting in performance degrade.

1.2.7 Configuring IP Multicast Boundary for a Specific IP Group

Use the following command to set IP multicast boundary for a specific IP group in interface configuration mode. Use the **no** form of this command to restore the default value.

Command	Purpose
Qtech (config-if) # **ip multicast boundary** *access-list* [**in** | **out**]	Sets IP multicast boundary for a specific IP group. Numerical standard ACL or name can be used to specify an IP group.

This command filters the IGMP, PIM-SM and PIM-DM packets associated with the IP group. Multicast packets will not flow in or out through the multicast boundary interface.


Note

The ACL in this command supports matching of destination IP addresses, not group IP addresses or source IP addresses.

1.2.8 Configuring IP Multicast Static Route

IP static multicast route enables multicast packet forwarding through a path different from IP unicast path. RPF check is always performed for IP multicast packet forwarding. The real interface receiving packets is the expected one, namely the next hop interface of IP unicast route used to transmit to the sender. The check is reasonable when

IP unicast topology is consistent with IP multicast topology. In some cases, however, it is better to make difference between IP unicast path and IP multicast path.

Static multicast route enables devices to execute RPF check according to configurations rather than the IP unicast routing table. Consequently, tunnel technology is used for IP multicast packet forwarding, not IP unicast packet forwarding. IP static multicast route is stored locally rather than be advertised or forwarded.

Use the following command to configure IP multicast static routes.

Command	Function
Qtech (config) # **ip mroute** [**vrf** *vrf-name*] *source-address mask* { **fallback-lookup** { **global** | **vrf** *vrf-name* } | [**bgp** | **isis** | **ospf** | **static**] {*v4rpf-address* | *interface-type interface-number* } } [*distance*]	Configures IP multicast static route. The routing protocol type can be set. *distance*: In the range of 0 to 255

Caution

To set the egress of the static multicast route not to be the next hop IP address, the egress must be a point-to-point interface.

1.2.9 Configuring Longest-match-based Routing

Use the following command to configure longest-match-based routing.

Command	Function
Qtech(config)# **ip multicast** [**vrf** *vrf-name*] **rpf longest-match**	Selects an optimal route from the static multicast, MBGP and unicast routing tables respectively in compliance with the RPF rules. Select the one with the longest mask matching from the three routes as the RPF route. If the three are of the same priority, select one in the order of multicast static route, MBGP route and unicast route.

The static multicast route, MBGP route and unicast route used for RPF check are elected from the static multicast routing table, MBGP routing table and unicast routing by RPF rules, respectively.

- By default, the one of highest priority is selected from these three routes. If they are of the same priority, select one in the order of static multicast route, MBGP route and unicast route.
- If the route selection based on the longest match has been configured, the one with the longest mask matching is selected from the three routes as the RPF route. If the masks of the three routes are the same, the one of the highest priority will be selected; if the three are of the same priority, a route is selected in the order of multicast static route, MBGP route and unicast route.

1.2.10 Configuring the Selection Method for PROXY in RPF Vector

Use the following commands to configure the selection method for proxy in RPF vector in global configuration mode.

Command	Function
ip multicast [**vrf** *vrf-name*] **rpf proxy** [**rd**] {**vector | disable**}	Configures the selection method for proxy. There are three selection methods for proxy: When the **rd** parameter is configured, the RPF Vector with RD information carried is used. This parameter takes effect only when the vrf is specified. When the **vector** parameter is configured, RFP Vector is used. When the **disable** parameter is configured, receiving RPF Vector is prohibited.
no ip multicast [**vrf** *vrf-name*] **rpf proxy** [**rd**] {**vector | disable**}	Deletes the selection method for proxy.

☑ Currently, this function is supported by RSR20. RSR30, RSR50 and RSR50E.

1.2.11 Configuring the Multicast Hardware Table Overflow Override Mechanism

This command deletes the hardware forwarding entry created earliest if the number of hardware forwarding table is full during creation of a new entry.

Command	Function
Qtech(config)# **msf ipmc-overflow override**	Deletes the hardware forwarding entry created earliest and adds the new entry if the hardware forwarding table is full during creation of a new entry.

1.2.12 Monitoring and Maintaining IP Multicast Routing

Use the following command to show the IPv4 multicast forwarding table in privileged EXEC mode.

Command	Function
show ip mroute [**vrf** *vrf-name*] [*group-or-source-address* [*group-or-source-address*]] [**dense** | **sparse**] [**summary** | **count**]	Shows the IPv4 multicast forwarding table.

Use the following command to delete the IPv4 multicast forwarding table in privileged EXEC mode.

Command	Function
clear ip mroute [vrf *vrf-name***]** {***** **|** *v4group-address* [*v4source -address*]}	Deletes the IPv4 multicast forwarding table.

Use the following command to reset the IPv4 multicast forwarding table statistics in privileged EXEC mode.

Command	Function
clear ip mroute [vrf *vrf-name***] statistics** {* | *v4group-address* [*v4source-address*]}	Resets the IPv4 multicast forwarding table statistics.

Use the following command to show the IPv4 static multicast routing information in privileged EXEC mode.

Command	Function
show ip mroute [**vrf** *vrf-name*] **static**	Shows the IPv4 static multicast routing information.

Use the following command to show the IPv4 static multicast routing information in privileged EXEC mode.

Command	Function
show ip rpf [**vrf** *vrf-name*] { *source-address* [*group-address*] [**rd** *route-distinguisher*] } [**metric**]	Shows the RPF information of specific IPv4 source address.

☑ Only RSR20, RSR30, RSR50 and RSR50E support the parameters of **group address**, **rd** and **metric**.

Use the following command to show the IPv4 static multicast interface information in privileged EXEC mode.

Command	Function
show ip mvif [**vrf** *vrf-name*] [*interface-type interface-number*]	Shows the IPv4 multicast interface information.

Use the following command to show the IPv4 layer 3 multicast forwarding table in privileged EXEC mode.

Command	Function
show ip mrf [**vrf** *vrf-name*] **mfc**	Shows the IPv4 layer 3 multicast forwarding table.

Use the following command to show the operation of multicast core in privileged mode.

Command	Function
debug nsm mcast [**vrf** *vrf-name*] **all**	Shows the operation of multicast core.

Use the following command to show the communication between the core of IPv4 multicast and multicast protocols in privileged mode.

Command	Function
debug nsm mcast [**vrf** *vrf-name*] **fib-msg**	Shows the communication between the core of IPv4 multicast and multicast protocols.

Use the following command to show the operation on the interface of the core of IPv4 multicast in privileged mode.

Command	Function
debug nsm mcast [**vrf** *vrf-name*] **vif**	Shows the operation on the interface of the core of IPv4 multicast.

Use the following command to show the operation of interface and statistics of the core of IPv4 multicast in privileged mode.

Command	Function
debug nsm mcast [**vrf** *vrf-name*] **stats**	Shows the operation of interface and statistics of the core of IPv4 multicast.

Use the following command to show the packet forwarding on Layer 3 of IPv4 multicast in privileged mode.

Command	Function
debug ip mrf [**vrf** *vrf-name*] **forwarding**	Shows the packet forwarding on Layer 3 of IPv4 multicast.

Use the following command to show the operation of forwarding entries on Layer 3 of IPv4 multicast in privileged mode.

Command	Function
debug ip mrf [**vrf** *vrf-name*] **mfc**	Shows the operation of forwarding entries on Layer 3 of IPv6 multicast.

Use the following command to show the operation of forwarding events on Layer 3 of IPv4 multicast in privileged mode.

Command	Function
debug ip mrf [**vrf** *vrf-name*] **event**	Shows the operation of forwarding events on Layer 3 of IPv4 multicast.

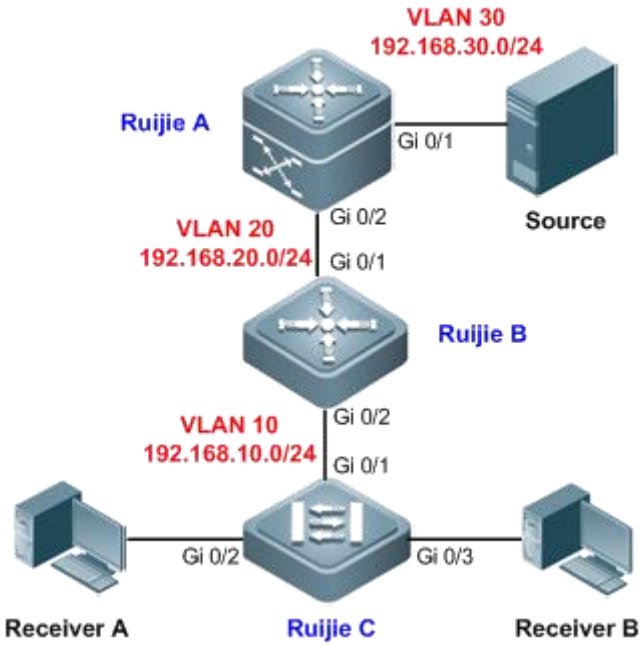
1.3 Configuration Examples

1.3.1 PIM-DM Configuration Example

1.3.1.1 Networking Topology

As shown in Figure 6, Qtech A and Qtech B are layer 3 devices; Qtech C is a layer-2 access device, with downlink users belonging to VLAN 10. The multicast source belongs to VLAN 30, and resides in a network segment different from the multicast receivers.

Figure 6 Topology for multicast routing network



1.3.1.2 Networking Requirements

- IGMP is running between multicast source and multicast receiver to establish and maintain the membership of a multicast group. For a dense-mode multicast network, PIM-DM can be applied to realize layer-3 routing of multicast data, while IGMP Snooping can be applied on the layer-2 device to realize layer-2 forwarding of multicast data.
- Only hosts belonging to VLAN 10 can join the multicast group 225.0.0.0/8, and a host can join up to 200 multicast groups.
- PIM adjacency established between the edge multicast router (Qtech B) and the downlink device by receiving PIM data packets should be avoided.

1.3.1.3 Configuration Tips

- Configure unicast routing protocol on the layer-3 devices (Qtech A and Qtech B in this example), and ensure the route connectivity between different network segments. Static route is configured in this example.
- On the layer-3 interface of multicast routing devices (SVI of VLANs 10, 20 and 30), configure PIM-DM to enable IGMP automatically (IGMPv2 is the default version).
- Configure IGMP Snooping on the layer-2 device (Qtech C in this example). Here only the IVGL mode of IGMP Snooping is enabled, and detailed configurations are not provided here. For details, see *Configuring IGMP Snooping*.
- Configure multicast group access control on the layer-3 interface (SVI for VLAN 10 of Qtech B) of multicast router to limit the range of multicast groups the downlink hosts can join. Configure the maximum number of IGMP group members on this interface (in this example, the maximum number is set to 200).
- Configure PIM neighbor filtering on the layer-3 interface of Qtech B for connecting to the layer-2 device. By setting up filtering conditions for ACL, only the PIM packets from uplink neighbors can be received.

1.3.1.4 Configuration Steps

- Step 1: Configure SVI for each VLAN.

! On Qtech A, create VLAN 20 and VLAN 30, and configure the SVI for VLAN 20 as 192.168.20.1/24 and the SVI for VLAN 30 as 192.168.30.1/24.

```
QtechA#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
QtechA(config)#vlan 20
QtechA(config-vlan)#exit
QtechA(config)#vlan 30
QtechA(config-vlan)#exit
QtechA(config)#interface vlan 20
QtechA(config-if-VLAN 20)#ip address 192.168.20.1 255.255.255.0
QtechA(config-if-VLAN 20)#exit
QtechA(config)#interface vlan 30
QtechA(config-if-VLAN 30)#ip address 192.168.30.1 255.255.255.0
QtechA(config-if-VLAN 30)#exit
```

! On Qtech B, create VLAN 10 and VLAN 20, and configure the SVI for VLAN 10 as 192.168.10.1/24 and the SVI for VLAN 20 as 192.168.20.2/24.

```
QtechB#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
QtechB(config)#vlan 10
QtechB(config-vlan)#exit
QtechB(config)#vlan 20
QtechB(config-vlan)#exit
QtechB(config)#interface vlan 10
QtechB(config-if-VLAN 10)#ip address 192.168.10.1 255.255.255.0
QtechB(config-if-VLAN 10)#exit
QtechB(config)#interface vlan 20
QtechB(config-if-VLAN 20)#ip address 192.168.20.2 255.255.255.0
QtechB(config-if-VLAN 20)#exit
```

- Step 2: Configure the attributes of each port.

! On Qtech A, configure Gi 0/1 as an access port belonging to VLAN 30 and Gi 0/2 as a trunk port.

```
QtechA(config)#interface gigabitEthernet 0/1
QtechA(config-if-GigabitEthernet 0/1)#switchport access vlan 30
```

```
QtechA(config-if-GigabitEthernet 0/1)#exit
QtechA(config)#interface gigabitEthernet 0/2
QtechA(config-if-GigabitEthernet 0/2)#switchport mode trunk
QtechA(config-if-GigabitEthernet 0/2)#exit
```

! On Qtech B, configure Gi 0/1 and Gi 0/2 as trunk ports.

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

! On Qtech C, configure Gi 0/1 as a trunk port and Gi 0/2-3 as an access port belonging to VLAN 10.

```
QtechC(config)#interface gigabitEthernet 0/1
QtechC(config-if-GigabitEthernet 0/1)#switchport mode trunk
QtechC(config-if-GigabitEthernet 0/1)#exit
QtechC(config)#interface range gigabitEthernet 0/2-3
QtechC(config-if-range)#switchport access vlan 10
QtechC(config-if-range)#exit
```

- Step 3: Configure static route on the layer-3 device.

! On Qtech B, configure the next-hop IP address for 192.168.30.0 as 192.168.20.1.

```
QtechB(config)#ip route 192.168.30.0 255.255.255.0 192.168.20.1
```

! On Qtech A, configure the next-hop IP address for 192.168.10.0 as 192.168.20.2.

```
QtechA(config)#ip route 192.168.10.0 255.255.255.0 192.168.20.2
```

- Step 4: Enable multicast routing on the layer-3 interface.

! On Qtech A, enable multicast routing globally, and enable PIM-DM on each interface.

```
QtechA(config)#ip multicast-routing
QtechA(config)#interface vlan 20
QtechA(config-if-VLAN 20)#ip pim dense-mode
QtechA(config-if-VLAN 20)#exit
QtechA(config)#interface vlan 30
QtechA(config-if-VLAN 30)#ip pim dense-mode
QtechA(config-if-VLAN 30)#exit
```

! On Qtech B, enable multicast routing globally, and enable PIM-DM on each interface.

```
QtechB(config)#ip multicast-routing
QtechB(config)#interface vlan 10
QtechB(config-if-VLAN 10)#ip pim dense-mode
QtechB(config-if-VLAN 10)#exit
QtechB(config)#interface vlan 20
QtechB(config-if-VLAN 20)#ip pim dense-mode
QtechB(config-if-VLAN 20)#exit
```

- Step 5: Enable IGMP Snooping on the layer-2 device.

! In global configuration mode, configure IGMP Snooping to operate in IVGL mode.

```
QtechC(config)#ip igmp snooping ivgl
```

- Step 6: Configure multicast group access control on the layer-3 interface and configure the maximum number of IGMP group members.

! On Qtech B, create ACL to permit the IP address 225.0.0.0/8.

```
QtechB(config)#ip access-list standard 1
QtechB(config-std-nacl)#permit 225.0.0.0 0.255.255.255
QtechB(config-std-nacl)#exit
```

! On the SVI for VLAN 10, configure multicast group access control and associate ACL.

```
QtechB(config)#interface vlan 10
QtechB(config-if-VLAN 10)#ip igmp access-group 1
```

! On the SVI for VLAN 10, configure the maximum number of allowed multicast groups as 200.

```
QtechB(config-if-VLAN 10)#ip igmp limit 200
QtechB(config-if-VLAN 10)#exit
```

- Step 7: Configure PIM neighbor filtering.

! On Qtech B, create the ACL to deny all IP addresses.

```
QtechB(config)#ip access-list standard 2
QtechB(config-std-nacl)#deny any
QtechB(config-std-nacl)#exit
```

! Configure PIM neighbor filtering on the SVI for VLAN 10 and associate ACL so that this interface does not receive PIM packets from other devices or establish adjacencies with them.

```
QtechB(config)#interface vlan 10
QtechB(config-if-VLAN 10)#ip pim neighbor-filter 2
QtechB(config-if-VLAN 10)#exit
```

1.3.1.5 Verification

- Step 1: Display device configurations.

! Configurations on Switch A

```
QtechA#show running-config
!
vlan 20
!
vlan 30
!
ip multicast-routing
!
interface GigabitEthernet 0/1
 switchport access vlan 30
!
interface GigabitEthernet 0/2
 switchport mode trunk
!
interface VLAN 20
 ip pim dense-mode
 no ip proxy-arp
 ip address 192.168.20.1 255.255.255.0
!
interface VLAN 30
 ip pim dense-mode
 no ip proxy-arp
 ip address 192.168.30.1 255.255.255.0
!
ip route 192.168.10.0 255.255.255.0 192.168.20.2
```

! Configurations on Switch B

```
SwtichB#show running-config
!
vlan 10
!
vlan 20
!
ip multicast-routing
!
ip access-list standard 1
 10 permit 225.0.0.0 0.255.255.255
!
ip access-list standard 2
 10 deny any
```

```
!
interface GigabitEthernet 0/1
 switchport mode trunk
!
interface GigabitEthernet 0/2
 switchport mode trunk
!
interface VLAN 10
 ip pim dense-mode
 ip pim neighbor-filter 2
 ip igmp access-group 1
 ip igmp limit 200
 no ip proxy-arp
 ip address 192.168.10.1 255.255.255.0
!
interface VLAN 20
 ip pim dense-mode
 no ip proxy-arp
 ip address 192.168.20.2 255.255.255.0
!
ip route 192.168.30.0 255.255.255.0 192.168.20.1
```

- Step 2: Display PIM-DM information of the interface (Qtech A is used as an example).

```
QtechA#show ip pim dense-mode interface detail
VLAN 20 (vif-id: 1):
  Address 192.168.20.1
  Hello period 30 seconds, Next Hello in 30 seconds
  Over-ride interval 2500 milli-seconds
  Propagation-delay 500 milli-seconds
  Neighbors:
    192.168.20.2
VLAN 30 (vif-id: 2):
  Address 192.168.30.1
  Hello period 30 seconds, Next Hello in 25 seconds
  Over-ride interval 2500 milli-seconds
  Propagation-delay 500 milli-seconds
  Neighbors: none
```

The preceding information shows the ID and address of the PIM-DM enabled interface and the corresponding PIM-DM neighbor.

- Step 3: Display the next hop information of PIM-DM (Qtech B is used as an example).

```
QtechB#show ip pim dense-mode nexthop
Destination    Nexthop    Nexthop          Nexthop          Metric Pref
                  Num        Addr             Interface
192.168.30.2     1        192.168.20.1     VLAN 20             0      1
```

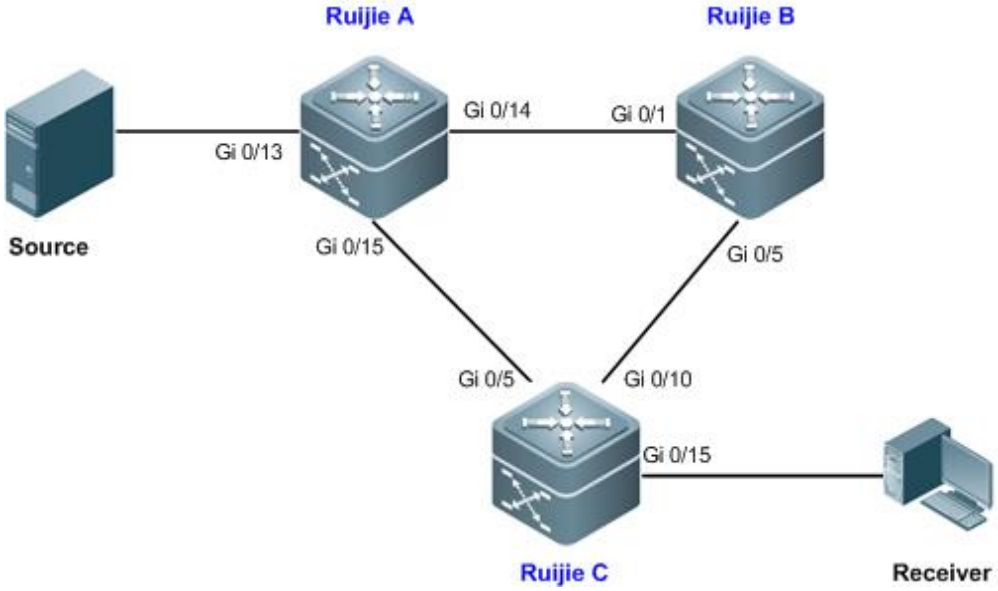
☑ This example applies to both layer-3 switches and routers. However, VLAN-related configuration commands are not supported on routers. Therefore, to establish the topology in this example on routers, directly replace the SVI interfaces with use common layer-3 interfaces.

1.3.2 PIM-SM Configuration Example (I)

1.3.2.1 Networking Topology

As shown in Figure 7, three layer-3 devices are interconnected through the routed ports. Multicast source and receiver are in different network segments.

Figure 7



The interface addresses of the devices are listed in the following table.

Device	Port Number	IP Address
Qtech A	Gi 0/13	192.168.1.1/24
	Gi 0/14	192.168.2.1/24
	Gi 0/15	192.168.3.1/24
Qtech B	Gi 0/1	192.168.2.2/24
	Gi 0/5	192.168.4.1/24
	Loopback1	10.1.1.1/24
Qtech C	Gi 0/5	192.168.3.2/24
	Gi 0/10	192.168.4.2/24
	Gi 0/15	192.168.5.1/24

1.3.2.2 Networking Requirements

- IGMP is running between multicast source and multicast receiver to establish and maintain the membership of a multicast group. For a sparse-mode multicast network, PIM-SM can be applied to realize layer-3 routing of multicast data.
- Unauthenticated multicast source should be avoided from sending multicast data in the PIM-SM domain.

1.3.2.3 Configuration Tips

- Configure unicast routing protocols on the layer-3 devices, and ensure the route connectivity between different network segments. This example configures the OSPF protocol. For details, see *Configuring OSPF*.
- After enabling PIM-SM on each interface, IGMP will be enabled automatically (IGMPv2 is the default version).
- In the entire PIM-SM domain, at least one RP must be configured (serving all multicast groups by default) to act as the root node of shared tree (in this example, one interface of Qtech B is configured as static RP).
- On the RP, configure address filtering of register messages (on Qtech B in this example).

1.3.2.4 Configuration Steps

- Step 1: Configure the IP address on the interface of each device.

! Configure the IP address of Qtech A's interface.

```
QtechA#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
QtechA(config)#interface gigabitEthernet 0/13
QtechA(config-if-GigabitEthernet 0/13)#no switchport
QtechA(config-if-GigabitEthernet 0/13)#ip address 192.168.1.1 255.255.255.0
QtechA(config-if-GigabitEthernet 0/13)#exit
QtechA(config)#interface gigabitEthernet 0/14
```

```
QtechA(config-if-GigabitEthernet 0/14)#no switchport
QtechA(config-if-GigabitEthernet 0/14)#ip address 192.168.2.1 255.255.255.0
QtechA(config-if-GigabitEthernet 0/14)#exit
QtechA(config)#interface gigabitEthernet 0/15
QtechA(config-if-GigabitEthernet 0/15)#no switchport
QtechA(config-if-GigabitEthernet 0/15)#ip address 192.168.3.1 255.255.255.0
QtechA(config-if-GigabitEthernet 0/15)#exit
```

! Configure the IP address of Qtech B's interface, and configure a Loopback interface as well.

```
QtechB(config)#interface gigabitEthernet 0/1
QtechB(config-if-GigabitEthernet 0/1)#no switchport
QtechB(config-if-GigabitEthernet 0/1)#ip address 192.168.2.2 255.255.255.0
QtechB(config-if-GigabitEthernet 0/1)#exit
QtechB(config)#interface gigabitEthernet 0/5
QtechB(config-if-GigabitEthernet 0/5)#no switchport
QtechB(config-if-GigabitEthernet 0/5)#ip address 192.168.4.1 255.255.255.0
QtechB(config-if-GigabitEthernet 0/5)#exit
QtechB(config)#interface loopback 1
QtechB(config-if-Loopback 1)#ip address 10.1.1.1 255.255.255.0
QtechB(config-if-Loopback 1)#exit
```

! Configure the IP address of Qtech C's interface.

```
QtechC(config)#interface gigabitEthernet 0/5
QtechC(config-GigabitEthernet 0/5)#no switchport
QtechC(config-GigabitEthernet 0/5)#ip address 192.168.3.2 255.255.255.0
QtechC(config-GigabitEthernet 0/5)#exit
QtechC(config)#interface gigabitEthernet 0/10
QtechC(config-GigabitEthernet 0/10)#no switchport
QtechC(config-GigabitEthernet 0/10)#ip address 192.168.4.2 255.255.255.0
QtechC(config-GigabitEthernet 0/10)#exit
QtechC(config)#interface gigabitEthernet 0/15
QtechC(config-GigabitEthernet 0/15)#no switchport
QtechC(config-GigabitEthernet 0/15)#ip address 192.168.5.1 255.255.255.0
```

- Step 2: Interconnect devices and configure the corresponding OSPF protocol on the devices.

! Configure Switch A

```
QtechA(config)#route ospf 1
QtechA(config-router)#network 192.168.1.0 0.0.0.255 area 0
QtechA(config-router)#network 192.168.2.0 0.0.0.255 area 0
QtechA(config-router)#network 192.168.3.0 0.0.0.255 area 0
QtechA(config-router)#exit
```

! Configure Qtech B

```
QtechB(config)#route ospf 1
QtechB(config-router)#network 10.1.1.0 0.0.0.255 area 0
QtechB(config-router)#network 192.168.2.0 0.0.0.255 area 0
QtechB(config-router)#network 192.168.4.0 0.0.0.255 area 0
QtechB(config-router)#exit
```

! Configure Qtech C

```
QtechC(config)#route ospf 1
QtechC(config-router)#network 192.168.3.0 0.0.0.255 area 0
QtechC(config-router)#network 192.168.4.0 0.0.0.255 area 0
QtechC(config-router)#network 192.168.5.0 0.0.0.255 area 0
QtechC(config-router)#exit
```

- Step 3: Globally enable multicast routing on the devices and enable PIM-SM on each interface.

! Configure Qtech A

```
QtechA(config)#ip multicast-routing
QtechA(config)#interface gigabitEthernet 0/13
```

```
QtechA(config-if-GigabitEthernet 0/13)#ip pim sparse-mode
QtechA(config-if-GigabitEthernet 0/13)#exit
QtechA(config)#interface gigabitEthernet 0/14
QtechA(config-if-GigabitEthernet 0/14)#ip pim sparse-mode
QtechA(config-if-GigabitEthernet 0/14)#exit
QtechA(config)#interface gigabitEthernet 0/15
QtechA(config-if-GigabitEthernet 0/15)#ip pim sparse-mode
QtechA(config-if-GigabitEthernet 0/15)#exit
```

! Configurations on Qtech B and Qtech C (including the Loopback interface on Qtech B) are the same as the preceding configurations.

- Step 4: Configure RP.

! Select the Loopback interface of Switch B as the static RP of PIM-SM domain. Note: Static RP must be configured identically on all PIM devices.

```
QtechA(config)#ip pim rp-address 10.1.1.1
```

! Configurations on Qtech B and Qtech C are the same as the preceding configuration.

- Step 5: Configure address filtering of register messages on RP.

! On Qtech B, create ACL to permit register messages with source IP address being 192.168.1.2 and group address range being 225.0.0.0/8–226.0.0.0/8.

```
QtechB(config)#ip access-list extended 100
QtechB(config-ext-nacl)#permit ip host 192.168.1.2 225.0.0.0 0.255.255.255
QtechB(config-ext-nacl)#permit ip host 192.168.1.2 226.0.0.0 0.255.255.255
QtechB(config-ext-nacl)#deny ip any any
QtechB(config-ext-nacl)#exit
```

! Associate this ACL with the register message address filtering of RP.

```
QtechB(config)#ip pim accept-register list 100
```

1.3.2.5 Verification

- Step 1: Display device configurations.

! Configurations on Qtech A.

```
QtechA#show running-config
!
ip pim rp-address 10.1.1.1
!
ip multicast-routing
!
interface GigabitEthernet 0/13
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.1.1 255.255.255.0
!
interface GigabitEthernet 0/14
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.2.1 255.255.255.0
!
interface GigabitEthernet 0/15
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.3.1 255.255.255.0
!
router ospf 1
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```

```
 network 192.168.3.0 0.0.0.255 area 0
!
```

! Configurations on Qtech B.

```
QtechB#show running-config
!
ip pim rp-address 10.1.1.1
ip pim accept-register list 100
!
ip multicast-routing
!
ip access-list extended 100
 10 permit ip host 192.168.1.2 225.0.0.0 0.255.255.255
 20 permit ip host 192.168.1.2 226.0.0.0 0.255.255.255
 30 deny ip any any
!
interface GigabitEthernet 0/1
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.2.2 255.255.255.0
!
interface GigabitEthernet 0/5
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.4.1 255.255.255.0
!
interface Loopback 1
 ip pim sparse-mode
 ip address 10.1.1.1 255.255.255.0
!
router ospf 1
 network 10.1.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.4.0 0.0.0.255 area 0
```

! Configurations on Qtech C.

```
QtechC#show running-config
!
ip pim rp-address 10.1.1.1
!
ip multicast-routing
!
interface GigabitEthernet 0/5
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.3.2 255.255.255.0
!
interface GigabitEthernet 0/10
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.4.2 255.255.255.0
!
interface GigabitEthernet 0/15
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.5.1 255.255.255.0
!
```

```
router ospf 1
 network 192.168.3.0 0.0.0.255 area 0
 network 192.168.4.0 0.0.0.255 area 0
 network 192.168.5.0 0.0.0.255 area 0
```

- Step 2: Display PIM-SM interface information (Qtech B is used as an example).

```
QtechB#show ip pim sparse-mode interface detail
GigabitEthernet 0/1 (vif 1):
  Address 192.168.2.2, DR 192.168.2.2
  Hello period 30 seconds, Next Hello in 1 seconds
  Triggered Hello period 5 seconds
  Neighbors:
   192.168.2.1
GigabitEthernet 0/5 (vif 2):
  Address 192.168.4.1, DR 192.168.4.2
  Hello period 30 seconds, Next Hello in 10 seconds
  Triggered Hello period 5 seconds
  Neighbors:
   192.168.4.2
Loopback 1 (vif 3):
  Address 10.1.1.1, DR 10.1.1.1
  Hello period 30 seconds
  Triggered Hello period 5 seconds
  Neighbors:
```

The preceding information shows the IP address of each interface and the IP addresses of the DR and PIM-SM neighbor in the corresponding network segment.

- Step 3: Display current RP information (Qtech B is used as an example).

```
QtechB#show ip pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): 224.0.0.0/4, Static
    RP: 10.1.1.1 , Static
         Uptime: 01:43:07
```

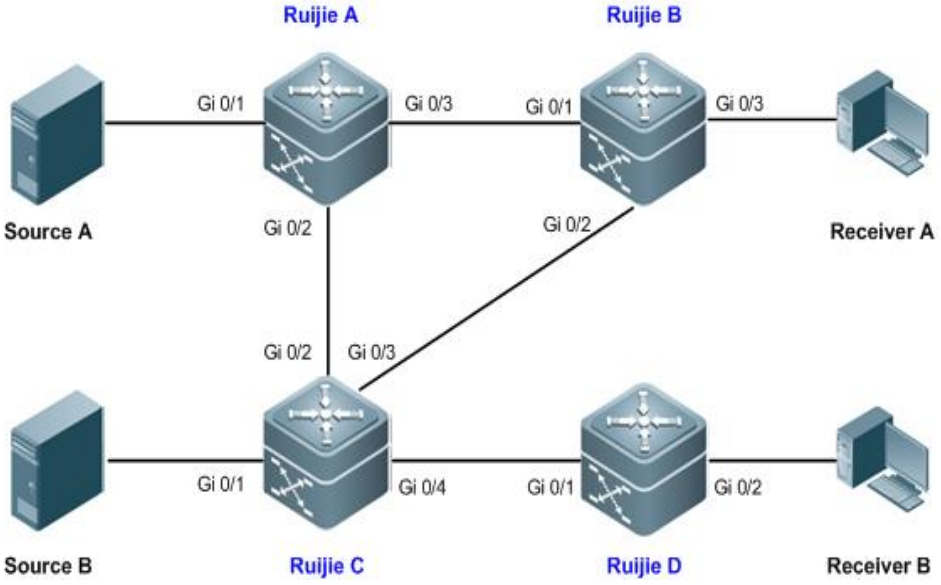
☑ This example applies to both layer-3 switches and routers. However, if you need to configure the switch port as a layer-3 port, you must run the **no switchport** command (not needed for routers).

1.3.3 PIM-SM Configuration Example (II)

1.3.3.1 Networking Topology

As shown in Figure 8, four layer-3 devices are interconnected through the routed ports. Multicast sources (Source A and Source B) and receivers (Receiver A and Receiver B) are in different network segments.

Figure 8



The interface addresses of each device are listed in the following table.

Device	Port Number	IP Address
Qtech A	Gi 0/1	192.168.1.1/24
	Gi 0/2	192.168.2.1/24
	Gi 0/3	192.168.3.1/24
Qtech B	Gi 0/1	192.168.3.2/24
	Gi 0/2	192.168.4.1/24
	Gi 0/3	192.168.5.1/24
	Loopback 1	10.1.1.1/24
	Loopback 2	10.1.2.1/24
Qtech C	Gi 0/1	192.168.6.1/24
	Gi 0/2	192.168.2.2/24
	Gi 0/3	192.168.4.2/24
	Gi 0/4	192.168.7.1/24
	Loopback 1	10.1.3.1/24
Qtech D	Gi 0/1	192.168.7.2/24
	Gi 0/2	192.168.8.1/24

1.3.3.2 Networking Requirements

- Multicast data forwarding between multicast routers is achieved through PIM-SM. One BSR in the PIM-SM domain is responsible for collecting and advertising RP information in the domain, while multiple candidate RPs serve different multicast groups to divert network traffic.
- The multicast router receives only BSM messages from a valid BSR.
- The BSR receives only advertisement packets from valid candidate RPs.

1.3.3.3 Configuration Tips

- Enable multicast routing on all multicast routers and enable PIM-SM multicast routing protocol on the interconnected interface. Note: Enabling PIM-SM will automatically enable IGMP.
- Specify one interface (Loopback 1 on Qtech B) as the candidate BSR and two other interfaces (Loopback 2 on Qtech B and Loopback 1 on Qtech C) as the candidate RPs, and configure the multicast groups served by the candidate RPs.
- On the multicast router needing to filter BSM messages, configure the range of valid BSR (in this example, enable Qtech C to permit only the BSM messages sent from Loopback 1 of Qtech B).
- On BSR (Qtech B in this example), configure the elected BSR to limit the valid C-RP address range and the multicast group range served.

1.3.3.4 Configuration Steps

- Step 1: Configure the interface IP addresses of each device.

! Configure Qtech A

```
QtechA#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
QtechA(config)#interface gigabitEthernet 0/1
QtechA(config-if-GigabitEthernet 0/1)#no switchport
QtechA(config-if-GigabitEthernet 0/1)#ip address 192.168.1.1 255.255.255.0
QtechA(config-if-GigabitEthernet 0/1)#exit
QtechA(config)#interface gigabitEthernet 0/2
QtechA(config-if-GigabitEthernet 0/2)#no switchport
QtechA(config-if-GigabitEthernet 0/2)#ip address 192.168.2.1 255.255.255.0
QtechA(config-if-GigabitEthernet 0/2)#exit
QtechA(config)#interface gigabitEthernet 0/3
QtechA(config-if-GigabitEthernet 0/3)#no switchport
QtechA(config-if-GigabitEthernet 0/3)#ip address 192.168.3.1 255.255.255.0
QtechA(config-if-GigabitEthernet 0/3)#exit
```

! Configurations on Qtech B, Qtech C and Qtech D are the same as the preceding configurations.

! On Qtech B, configure the IP address of Loopback 1 as 10.1.1.1/24 and the IP address of Loopback 2 as 10.1.2.1/24.

```
QtechB(config)#interface loopback 1
QtechB(config-if-Loopback 1)#ip address 10.1.1.1 255.255.255.0
QtechB(config-if-Loopback 1)#exit
QtechB(config)#interface loopback 2
QtechB(config-if-Loopback 2)#ip address 10.1.2.1 255.255.255.0
QtechB(config-if-Loopback 2)#exit
```

! On Qtech C, configure the IP address of Loopback 1 as 10.1.3.1/24.

```
QtechC(config)#interface loopback 1
QtechC(config-Loopback 1)#ip address 10.1.3.1 255.255.255.0
QtechC(config-Loopback 1)#exit
```

- Step 2: Interconnect devices and configure the corresponding OSPF protocol on the devices.

! Configure Qtech A

```
QtechA(config)#route ospf 1
QtechA(config-router)#network 192.168.1.0 0.0.0.255 area 0
QtechA(config-router)#network 192.168.2.0 0.0.0.255 area 0
QtechA(config-router)#network 192.168.3.0 0.0.0.255 area 0
QtechA(config-router)#exit
```

! Configurations on Qtech B, Qtech C and Qtech D are the same as the preceding configurations.

Step 2: Enable multicast routing on each device and enable PIM-SM multicast routing protocol on each interface.

! Configure Qtech A

```
QtechA(config)#ip multicast-routing
QtechA(config)#interface range gigabitEthernet 0/1-3
QtechA(config-if-range)#ip pim sparse-mode
```

! Configurations on Qtech B, Qtech C and Qtech D are the same as the preceding configurations. Note: PIM-SM must be enabled on the Loopback interface.

- Step 3: Configure the candidate BSR and candidate RP.

! On Qtech B, configure Loopback 1 as the candidate BSR.

```
QtechB(config)#ip pim bsr-candidate loopback 1 24
```

! On Qtech B, create standard ACL to permit the address range of 225.0.0.0/8 to 226.0.0.0/8.

```
QtechB(config)#ip access-list standard 1
```

```
QtechB(config-std-nacl)#permit 225.0.0.0 0.255.255.255
QtechB(config-std-nacl)#permit 226.0.0.0 0.255.255.255
QtechB(config-std-nacl)#exit
```

! Configure Loopback 2 of Qtech B as the candidate RP and associate the ACL.

```
QtechB(config)#ip pim rp-candidate loopback 2 group-list 1
```

! On Qtech C, create the standard ACL to permit the address range of 227.0.0.0/8 to 228.0.0.0/8.

```
QtechC(config)#ip access-list standard 1
QtechC(config-std-nacl)#permit 227.0.0.0 0.255.255.255
QtechC(config-std-nacl)#permit 228.0.0.0 0.255.255.255
QtechC(config-std-nacl)#exit
```

! On Qtech C, configure Loopback 1 as the candidate RP and associate the ACL.

```
QtechC(config)#ip pim rp-candidate loopback 1 group-list 1
```

- Step 4: Configure the range of valid BSR.

! On Qtech C, create the standard ACL named "bsr_acl" to permit only packets with IP address being 10.1.1.1.

```
QtechC(config)#ip access-list standard bsr_acl
QtechC(config-std-nacl)#permit host 10.1.1.1
QtechC(config-std-nacl)#exit
```

! On Qtech C, configure the range of valid BSR and associate ACL "bsr_acl".

```
QtechC(config)#ip pim accept-bsr list bsr_acl
```

- Step 5: Configure the elected BSR to limit the valid C-RP address range and the multicast group range served.

! On Qtech B, create extended ACL named "rp_acl" to permit only packets with IP address being 10.1.3.1 and multicast address range being 227.0.0.0/8–228.0.0.0/8.

```
QtechB(config)#ip access-list extended rp_acl
QtechB(config-ext-nacl)#permit ip host 10.1.3.1 227.0.0.0 0.255.255.255
QtechB(config-ext-nacl)#permit ip host 10.1.3.1 228.0.0.0 0.255.255.255
QtechB(config-ext-nacl)#exit
```

! On Qtech B, configure the elected BSR to limit the valid C_RP.

```
QtechB(config)#ip pim accept-crp list rp_acl
```

1.3.3.5 Verification

- Step 1: Display device configurations.

! Configurations on Qtech A.

```
QtechA#show running-config
!
ip multicast-routing
!
interface GigabitEthernet 0/1
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.1.1 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.2.1 255.255.255.0
!
interface GigabitEthernet 0/3
 no switchport
```

```
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.3.1 255.255.255.0
!
router ospf 1
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0

 network 192.168.3.0 0.0.0.255 area 0
```

! Configurations on Qtech B.

```
QtechB#show running-config
!
ip pim accept-crp list rp_acl
ip pim bsr-candidate Loopback 1 24
ip pim rp-candidate Loopback 2 group-list 1
!
ip multicast-routing
!
ip access-list standard 1
 10 permit 225.0.0.0 0.255.255.255
 20 permit 226.0.0.0 0.255.255.255
!
ip access-list extended rp_acl
 10 permit ip host 10.1.3.1 227.0.0.0 0.255.255.255
 20 permit ip host 10.1.3.1 228.0.0.0 0.255.255.255
!
interface GigabitEthernet 0/1
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.3.2 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.4.1 255.255.255.0
!
interface GigabitEthernet 0/3
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.5.1 255.255.255.0
!
interface Loopback 1
 ip pim sparse-mode
 ip address 10.1.1.1 255.255.255.0
!
interface Loopback 2
 ip pim sparse-mode
 ip address 10.1.2.1 255.255.255.0
!
router ospf 1
 network 10.1.1.0 0.0.0.255 area 0
 network 10.1.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
 network 192.168.4.0 0.0.0.255 area 0
 network 192.168.5.0 0.0.0.255 area 0
```

! Configurations on Qtech C.

```
QtechC#show running-config
```

```
!
ip pim accept-bsr list bsr_acl
ip pim rp-candidate Loopback 1 group-list 1
!
ip multicast-routing
!
ip access-list standard 1
 10 permit 227.0.0.0 0.255.255.255
 20 permit 228.0.0.0 0.255.255.255
!
ip access-list standard bsr_acl
 10 permit host 10.1.1.1
!
interface GigabitEthernet 0/1
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.6.1 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.2.2 255.255.255.0
!
interface GigabitEthernet 0/3
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.4.2 255.255.255.0
!
interface GigabitEthernet 0/4
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.7.1 255.255.255.0
!
interface Loopback 1
 ip pim sparse-mode
 ip address 10.1.3.1 255.255.255.0
!
router ospf 1
 network 10.1.3.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.4.0 0.0.0.255 area 0
 network 192.168.6.0 0.0.0.255 area 0
 network 192.168.7.0 0.0.0.255 area 0
```

! Configurations on Qtech D.

```
QtechD#show running-config
!
ip multicast-routing
!
interface GigabitEthernet 0/2
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
 ip address 192.168.8.1 255.255.255.0
!
interface GigabitEthernet 0/11
 no switchport
 ip pim sparse-mode
 no ip proxy-arp
```

```
 ip address 192.168.7.2 255.255.255.0
!
router ospf 1
 network 192.168.7.0 0.0.0.255 area 0
 network 192.168.8.0 0.0.0.255 area 0
```

- Step 2: Display RPs in the PIM-SM domain and the corresponding service multicast information (Qtech A is used as an example).

```
QtechA#show ip pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): 225.0.0.0/8
  RP: 10.1.2.1
    Info source: 10.1.1.1, via bootstrap, priority 192
         Uptime: 01:15:16, expires: 00:02:00
Group(s): 226.0.0.0/8
  RP: 10.1.2.1
    Info source: 10.1.1.1, via bootstrap, priority 192
         Uptime: 01:15:16, expires: 00:02:00
Group(s): 227.0.0.0/8
  RP: 10.1.3.1
    Info source: 10.1.1.1, via bootstrap, priority 192
         Uptime: 01:13:30, expires: 00:02:00
Group(s): 228.0.0.0/8
  RP: 10.1.3.1
    Info source: 10.1.1.1, via bootstrap, priority 192
         Uptime: 01:13:30, expires: 00:02:00
```

By limiting the multicast address range served by each candidate RP, the multicast sources in the PIM-SM domain can be limited. The receiver cannot receive multicast data sent from a multicast source that is not within the address range served (225.0.0.0/8–228.0.0.0/8).

In addition, invalid BSMs (the source address is not 10.1.1.1) are directly dropped on the device configured with valid BSR limitation. If invalid C_RP advertisement messages are sent to BSR, after valid C_RP limitation is configured, BSR will filter invalid C_RP advertisement messages.

☑ This example applies to both layer-3 switches and routers. However, if you need to configure the switch port as a layer-3 port, you must run the **no switchport** command (not needed for routers).

2 CONFIGURING IPV6 MULTICAST

2.1 Overview

Traditional IP transmission allows one host to transmit packets to a single host (unicast communication) or all hosts (broadcast communication). (Note that IPv6 no longer supports broadcast). Multicast, however, allows one host to send packets to some hosts (also known as group members).

The multicast application consists of the sender and the receiver. The sender can send multicast packets without needing to join a group. In contrast, the receiver can receive the multicast packets from the group only after joining the group.

Group members are dynamic. A host can join in or leave from a group at any time. Furthermore, there is no limit on the position and number of group members. A host can join in more than one group simultaneously if necessary. Consequently, the active status and the number of members of a group vary with time.

The device maintains the routing table for forwarding multicast packets by running IPv6 multicast routing protocol (for instance, PIM-SMv6) and learns the status of group members on the direct segment by running the MLDv1/v2 protocol. The device joins inan IPv6 multicast group by sending the MLD report message.

IPv6 multicast applies to one-to-many multimedia applications.

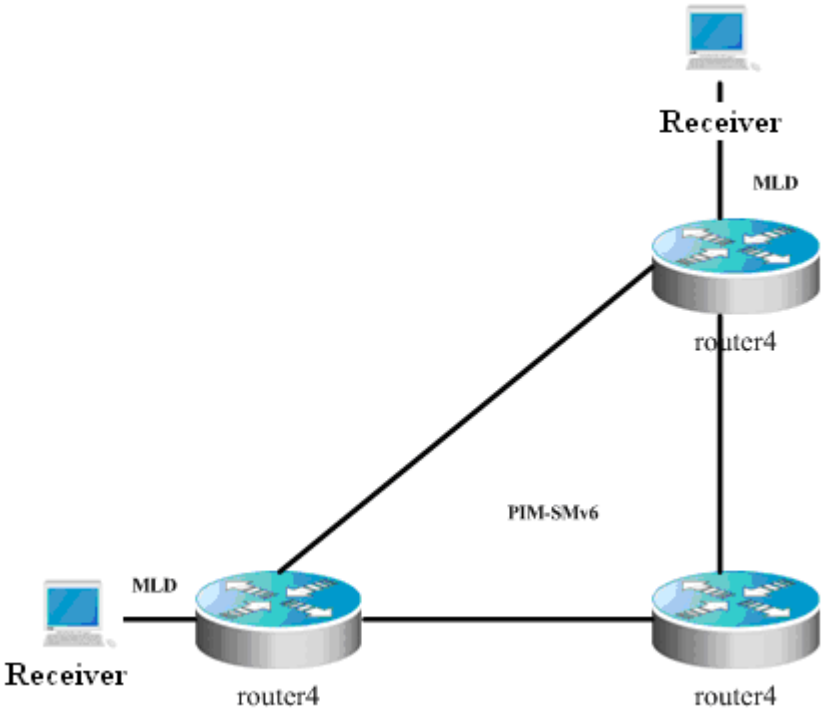
2.1.1 Implementation of IPv6 Multicast Routing

The IPv6 multicast routing protocol includes:

- MLD: Runs between the multicast deivce and the host to learn the relation of group members.
- PIM-SMv6: Runs between the multicast devices to enable multicast packet forwarding by setting up the multicast routing table.

The following figure illustrates the function of the multicast protocols used in IPv6 multicast packet forwarding:

Figure 1 Multicast protocols used in IPv6 multicast environment



2.1.2 MLD Overview

To enable IPv6 multicast, the multicast hosts and devices must support the MLD protocol, which maintains the multicast group member relation between the multicast device and the unicast device to determine the forwarding of multicast streams.

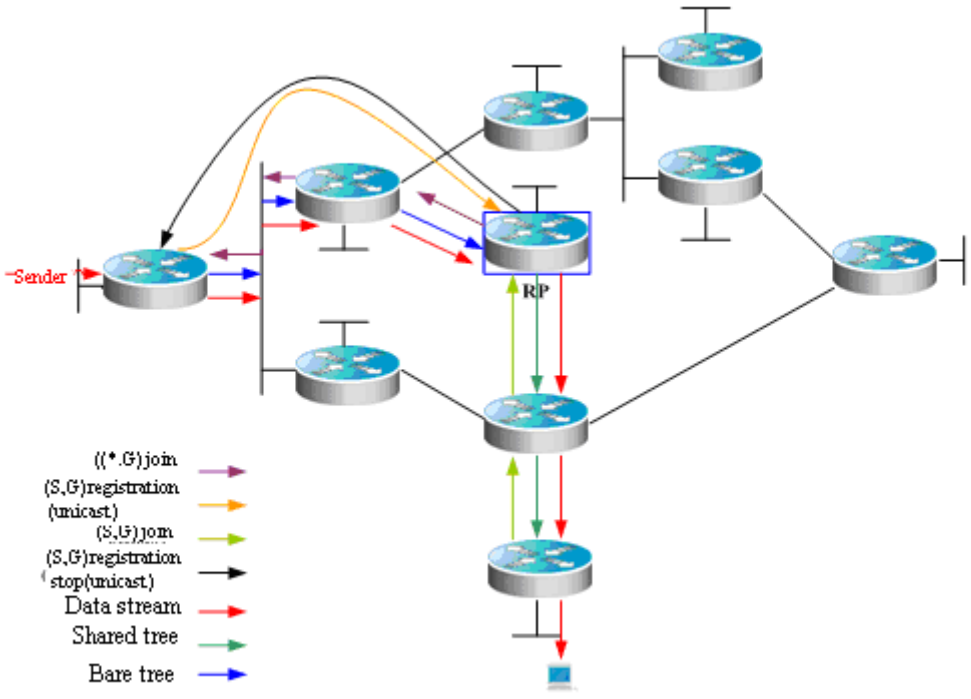
Based on the messages obtained from the MLD protocol, the device maintains a multicast group member table per interface, which is activated when at least one host on the interface is group member.

2.1.3 PIM-SMv6 Overview

PIM (Protocol Independent Multicast) is designed by IDMR (Inter-domain Multicast Routing) working group. As its name implied, PIM does not depend on a specific unicast routing protocol. It utilizes the unicast routing table established by various unicast routing protocols to enable the RPF check function instead of maintaining a separate multicast routing table for forwarding multicast packets. Compared with other multicast protocols, PIM overhead falls down at large extent for PIM does not need to receive and send multicast route update. The concept behind PIM design is that support and flexible transformation between SPT and the shared tree is enabled for higher multicast efficiency. There are two kinds of PIM modes-dense mode and sparse mode.

PIM-SMv6 (Protocol Independent Multicast Sparse Mode) is a multicast routing protocol in sparse mode. In the PIM-SMv6 domain, the device running PIM-SMv6 sends the Hello message at a specific interval to discover adjacent devices running PIM-SMv6 and be in charge of DR election. Here DR sends the “join/prune” message to its direct group members in the direction of the root node of the multicast distribution tree or sends the data from the direct multicast source to the multicast distribution tree.

Figure 2 PIM-SMv6 explicit join mechanism



PIM-SMv6 forwards multicast data packets by setting up a multicast distribution tree. There are two kinds of multicast distribution tree: the shared tree using G’s RP as root and the shortest path tree using the multicast source as root. With the explicit join/prune mechanism, PIM-SMv6 sets up and maintains the multicast distribution tree. As shown in the above figure, when the DR receives a join report message from the receiver, it multicasts a (*.G) join message in the direction of G’s RP hop by hop to join the shared tree. The source host sends multicast packets. Upon receiving the packets, its DR encapsulates them in the registration message, and then unicasts to the RP. Then the RP forwards the decapsulated packets to every group member along the shared tree. The RP sends the

(S,G) join message to the first hop device along the source direction to join the shortest path tree of the source. In this way, the source sends packets along the shortest path tree to the RP without encapsulation. Upon receiving the first multicast data, the RP sends the registration stop message to the source DR to stop encapsulation. After that, the source sends multicast packets without registration and encapsulation alogn its shortest path tree to the RP, which then forwards the packets to every group members. When there is no necessary to receive multicast packets, the DR on the receiving side sends the multicast prune message to G's RP hop by hop to prune the shared tree.

RP election is also involved in PIM-SMv6. when there is one or more candidate BSRs configured in the PIM-SMv6 domain, some rule is applied to elect BSR. Candidate RPs are also configured in the PIM-SMv6 domain, which send the packets containing their addresses and serviceable multicast groups to BSRs in unicast form. BSRs generate bootstrap messages with a series of candidate RPs and the addresses of corresponding multicast groups periodically. These bootstrap messages are transmitted in the overall domain hop by hop. Devices will receive and save these bootstrap messages. Upon receiving the member relation report of a multicast group from the directly connected device, the DR will use a hash algorithm and map the multicast group address to a candidate RP who can serve this group if it has not the routing entry of this group. Then the DR will send the join/prune message in multicast form hop by hop along the way to the RP. On the other hand, upon receiving multicast packets from the directly connected device, the DR will use a hash algorithm and map the multicast group address to a candidate RP who can serve this group, and then encapsulate these multicast packets in the registration message and send it to the PR in unicast form.

The essential difference between PIM-SMv6 and PIM-DM is that PIM-SMv6 is based on explicit join mode and PIM-DM is based on flood/prune mode. For PIM-SMv6, the receiver sends a join message to the PR, but the device forwards the packets of a multicast group only on the interface joining this multicast group. PIM-SMv6 forwards multicast packets through the shared tree. Each multicast group has a rendezvous point. The multicast source sends packets to the RP along the shortest path, and then the RP sends the packets to every receiver along the shortest path. This process is similar to CBT. However, the core concept is not used in PIM-SMv6. One of the main advantages of PIM-SMv6 is that it not only receives multicast packets through the shared tree but also offers the shared tree-to-SPT transformation mechanism. This transformation consumes a lot number of resources, even though it reduces network delay and possible block on the RP. It is suitable for the environment where there are many pairs of multicast sources yet fewer networks.

PIM-SMv6 distributes multicast packets through the shared tree and SPT. Assume that other devices do not need to receive these multicast packets, unless otherwise specified. When a host joins a multicast group, the devices connecting to the host notify the root (or RP) through the PIM join message. This message is transferred among these devices in order to set up the structure of a shared tree. So, the RP records this transmission path and the registration message from the first hop device (DR) of the sending multicast source, and perfects the shared tree based on these two messages. Update of leaf messages is enabled on periodic query message. For the shared tree, the multicast source sends multicast packets to the RP so that all receivers can receive these multicast packets.

PIMv2 BSR distributes the group-to-RP message to all devices without the necessity for configuring RP for every device. The BSR distributes the mapping message through the hop-by-hop flooding BSR message. First of all, the BSR is elected among devices. This election procedure is similar to electing the root bridge in STP by priority. Every BSR device checks the BSR message, and only forwards the BSR messages with higher or equivalent priority (or higher IP address). The elected BSR sends the BSR message to the all-PIM-routers multicast group (ff02::d) with TTL 1. Upon receiving the BSR message, the adjacent PIMv2 device sents it out in multicast form and then reset TTL to 1. In this way, the BSR message is sent to all devices hop by hop. Since the BSR message includes the IP addresses of BSR devices, the candidate BSR can determine which device is the current BSR device. The candidate RP sends the candidate RP advertisement and alleges in which address ranges it can become RP. The BSR stores the advertisement message in its local candidate RP cache, and notifies all PIM devices of local candidate RPs periodically. Also in this way, the message is sent to all devices hop by hop.

2.1.4 RPF Rules

Multicast routing protocols depend on existing unicast route messages, MBGP routes or static multicast routes to create multicast routing entries. When creating multicast routing entries, multicast routing protocols run the Reverse Path Forwarding (RPF) check mechanism to guarantee that multicast packets are transmitted along proper paths while avoiding loops.

RPF check is on the basis of unicast routes, MBGP routes or static multicast route:
- The unicast routing table summarizes the shortest paths to each destination segments.
- The MBGP routing table directly offers multicast routes.
- The static multicast routing table lists the RPF route messages that user configures statically and manually.

The multicast routing protocol searches the unicast routing table, the MBGP routing table and the static multicast routing while run RPF check, as shown below:

1. First of all, select an optimal route from the unicast routing table, the MBGP routing table and the static multicast routing table, respectively.

a. Select an optimal route from the unicast routing table for RPF check:

- Use the IP address of the packet source as the destination address to search the unicast routing table and select an optimal unicast route.
- If the unicast route has only one next hop, check whether multicast is enabled on the egress of the next hop or not.
  - If not, the unicast route is not suitable for RPF check.
  - If so, the unicast route is suitable for RPF check and the egress serves as the RPF interface.
- If the unicast route has more than one next hop, traverse all next hops and check whether multicast is enabled on the egress of one next hop.
  - If not, traverse the next hop.
  - If so, the unicast route is suitable for RPF check and the egress serves as the RPF interface.
  - If no multicast is enabled on the egress after traversing all next hops, there is no unicast route suitable for RPF check.
- If no optical route exists, there is no unicast route suitable for RPF check.

2. Select an optimal route from the MBGP routing table for RPF check:

- Use the IP address of the packet source as the destination address to search the MBGP routing table and select an optimal MBGP route.
- The MBGP route has only one next hop. Check whether multicast is enabled on the egress of the next hop or not.
  - If not, the MBGP route is not suitable for RPF check.
  - If so, the MBGP route is suitable for RPF check.
- If no optical route exists, there is no MBGP route suitable for RPF check.

3. Select an optimal route from the static multicast routing table for RPF check:

- Use the IP address of the packet source as the destination address to search the static multicast routing table and select an optimal static multicast route.
- If the static multicast route has only one next hop, check whether multicast is enabled on the egress of the next hop or not.
  - If not, the static multicast route is not suitable for RPF check.
  - If so, further check whether there is a unicast route available for RPF check.
    - If the next hop does not associate with the unicast protocol number, the static multicast route is suitable for RPF check.
    - If there is no unicast route available for RPF check, the static multicast route is suitable for RPF check.
    - If there is a unicast route available for RPF check, but the unicast protocol number is inconsistent with the one associated with the next hop of the static multicast route, the static multicast route is not suitable for RPF check.
    - If there is a unicast route available for RPF check, and the unicast protocol number is consistent with the one associated with the next hop of the static multicast route, the static multicast route is suitable for RPF check.
- If the static multicast route has more than one next hop, traverse all next hops and check whether multicast is enabled on the egress of one next hop.
  - If not, traverse the next hop.

- If so, further check whether there is a unicast route available for RPF check.
  - If the next hop does not associate with the unicast protocol number, the static multicast route is suitable for RPF check.
  - If there is no unicast route available for RPF check, the static multicast route is suitable for RPF check.
  - If there is a unicast route available for RPF check, but the unicast protocol number is inconsistent with the one associated with the next hop of the static multicast route, traverse the next hop.
  - If there is a unicast route available for RPF check, and the unicast protocol number is consistent with the one associated with the next hop of the static multicast route, the static multicast route is suitable for RPF check.
  - If no multicast is enabled on the egress after traversing all next hops, there is no static multicast route suitable for RPF check.

◆ If no optical route exists, there is no static multicast route suitable for RPF check.

4. Then, select one from these three optimal routes for RPF check.
   a. If the longest match routing rule is configured, select the longest match route; if these three routes are of the same mask, select the one of the highest priority; if they are of the same priority, select the one in the order of static multicast route, MBGP route and unicast route.
   b. If the longest match routing rule is not configured, select the one of the highest priority; if they are of the same priority, select the one in the order of static multicast route, MBGP route and unicast route.


Caution

The effectiveness of MBGP routes recurs on unicast routes rather than distance. The implementation of current MBGP protocols does not support equal-cost routes.


Caution

The effectiveness of static multicast routes recurs on unicast routes rather than distance.


Caution

If the static unicast route is selected as RPF route, the route must be configured with the next hop IP address. PIM protocol will select RPF neighbors based on this next hop IP address. If the static unicast route is not configured with the next hop IP address, PIM protocol can get RPF neighbors.

2.2 Basic IPv6 Multicast Route Configuration

Basic IPv6 multicast route configuration includes:
- Enable IPv6 multicast route forwarding (mandatory)
- Enable IPv6 multicast route protocol (mandatory)

2.2.1 Enabling IPv6 Multicast Route Forwarding

This function enables software to forward multicast packets.

To enable IPv6 multicast route forwarding, run the following command in the global configuration mode:

Command	Function
Qtech(config)# **ipv6 multicast-routing**	Enable IPv6 multicast routing.

2.2.2 Enabling IPv6 Multicast Route Protocol

To enable IPv6 PIM-SM multicast route protocol, run the following command in the interface configuration mode:

Command	Function
Qtech(config-if)# **ipv6 pim sparse-mode**	Enable PIM-SM. This command should run on Layer 3 interfaces.

The following example configures PIM-SM on GigabitEthernet 0/3.

```
Qtech(config)# ipv6 multicast-routing
Qtech(config)# interface  gigabitEthernet 0/3
Qtech(config-if)# ipv6 address 3333::3333/64
Qtech(config-if)# ipv6 pim sparse-mode
```

Note

When IPv6 multicast route is enabled globally, enabling PIM-SMv6 on an interface will enable MLD on the corresponding interface. Only one kind of multicast routing protocol is allowed to run on an interface.

Caution

After enabling the layer3 multicasting on the Private VLAN and Super VLAN, if the multicast source exists in the Sub-VLAN, one more route entry is needed to be duplicated and the ingress is the Sub-VLAN in which the multicast streams enter as the ingress validity check is required when multicast forwarding, resulting in occupying one more multicast hardware entry with 1 less multicast capacity.

Caution

S5760 supports running RPF check on SVI interface, not on routed interface. Consequently, if the routed interface becomes RPF interface, multicast streams from a non-RPF interface can also be forwarded according to route entries.

2.2.3 Enabling MLD

Enabling IPv6 multicast route forwarding and IPv6 multicast route protocol will enable MLD.

2.3 Advanced IPv6 Multicast Core Function Configuration

Advanced IPv6 multicast core function configuration includes:

- Limit the number of the routes that are allowed to join the IPv6 multicast routing table (optional)
- Set the IPv6 multicast border for the specific IPv6 group range (optional)
- Configure static IPv6 multicast route (optional)
- IPV6 multicast route monitoring and maintenance (optional)

2.3.1 Limiting the Number of the Routes That are Allowed to Join the IPv6 Multicast Routing Table

In the global configuration mode, use the **ipv6 multicast route-limit** *limit* [*threshold*] command to limit the number of the routes that are allowed to join the multicast routing table. Use the no form of this command to restore it to the default value, or 1024.

Command	Function
Qtech(config-if)# **ipv6 multicast route-limit** *limit* [*threshold*]	Limits the number of the routes that are allowed to join the multicast routing table. *limit*: Number of the routes that are allowed to join the multicast routing table in the range 1 to 2147483647, 1024 by default. Threshold: (optional) Number of multicast routes triggering alarm, 2147483647 by default. Note: Given the hardware resource for different models of devices, the routes exceeding the hardware entry threshold need to be forwarded through software, and resulting in decrease in performance.

2.3.2 Setting IPv6 Multicast Border for Specific IPv6 Group Range

In interface configuration mode, use the **ipv6 multicast boundary** *access-list-name* command to set IPv6 multicast border for specific IPv6 group range. Use no form of this command to restore it to the default value, namely no multicast border.

Command	Function
Qtech(config-if)# **ipv6 multicast boundary** *access-list-name* [**in	out**]	Set IPv6 multicast border for specific IPv6 group range. ACL can be used to specify the IPv6 group range. Note: The ACL associated with this command supports only matching destination IP address, not multicast IP address and source IP address.

This command filters MLD and PIM-SMv6 protocol packets correlating with the specific IPv6 group range. Multicast streams will income and outgoing through the multicast border interface.

2.3.3 Configuring Static IPv6 Multicast Route

IPv6 static multicast route enables multicast packet forwarding through a path different from IPv6 unicast path. RPF check is always performed while forwarding. The real interface receiving packets is the expected one, namely the next hop interface of IPv6 unicast route used to transmit to the sender. The check is reasonable when IPv6 unicast topology is in accord with IPv6 multicast topology. In some cases, however, it is better to make difference between IPv6 unicast path and IPv6 multicast path.

Static multicast route enables devices to execute RPF check according to configurations rather than the IPv6 unicast routing table. Consequently, tunnel technology is used for IPv6 multicast packet forwarding, not IPv6 unicast packet forwarding. IPv6 static multicast route is stored locally rather than be advertised or forwarded.

In the global configuration mode, use the following command to configure IPv6 static multicast route.

Command	Function
Qtech(config)# **ipv6 mroute** *ipv6-prefix/prefix-length* [**bgp**	**isis**	**ospfv3**	**ripng**	**static**]{ *ipv6-prefix*	*interface-type interface-number*} [*distance*]	Configures IPv6 static multicast route. Routing protocol can be set at the same time. Distance: <1-255>


Note

To set the egress of the static multicast route not to be the IPv6 address of next hop, the egress must be an point-to-point interface.

2.3.4 Configuring Longest-match-based Routing

The static multicast route, MBGP route and unicast route used for RPF check are elected from the static multicast routing table, MBGP routing table and unicast routing by RPF rules, respectively.

By default, the one of highest priority is selected from these three routes. If they are of the same priority, select one in the order of static multicast route, MBGP route and unicast route.

Use this command to select the route matching the longest mask from these three routes. If they are of the same priority level, the order of choice is: multicast static route, MBGP route and unicast route.

Command	Function
Qtech(config)# **ipv6 multicast rpf longest-match**	Selects the route matching the longest mask.

2.3.5 Multicast Route Monitoring and Maintenance

In the privileged EXEC configuration mode, run the following command to show the information of IPv6 multicast forwarding table.

Command	Function
show ipv6 mroute [*v6group-address*] [*v6source-address*] [**dense**]|[**sparse**] } {[**summary**]|[**count**]}	Show the information of IPv6 multicast forwarding table.

In the privileged EXEC configuration mode, run the following command to delete the IPv6 multicast forwarding table.

Command	Function
clear ipv6 mroute {* | *v6group-address* [*v6source -address*]}	Delete the IPv6 multicast forwarding table.

In the privileged EXEC configuration mode, run the following command to reset the statics of the IPv6 multicast forwarding table.

Command	Function
clear ipv6 mroute statistics {* | *v6group-address* [*v6source-address*]}	Reset the statics of the IPv6 multicast forwarding table.

In the privileged EXEC configuration mode, run the following command to show the RPF information of the specific IPv6 source address.

Command	Function
show ipv6 rpf *v6source-address*	Show the RPF information of the specific IPv6 source address.

In the privileged EXEC configuration mode, run the following command to show the information of static IPv6 multicast route.

Command	Function

show ipv6 mroute static	Show the information of static IPv6 multicast route.

In the privileged EXEC configuration mode, run the following command to show the information of IPv6 multicast interface.

Command	Function
show ipv6 mvif [*interface-type interface-number*]	Show the information of IPv6 multicast interface.

In the privileged EXEC configuration mode, run the following command to show the IPv6 Layer 3 multicast forwarding table.

Command	Function
show ipv6 mrf mfc	Show the IPv6 Layer 3 multicast forwarding table.

In the privileged EXEC configuration mode, run the following command to show the operation of the core of IPv6 multicast.

Command	Function
debug nsm mcast6 all	Show the operation of the core of IPv6 multicast.

In the privileged EXEC configuration mode, run the following command to show the communication between the core of IPv6 multicast and multicast protocols.

Command	Function
debug nsm mcast6 fib-msg	Show the communication between the core of IPv6 multicast and multicast protocols.

In the privileged EXEC configuration mode, run the following command to show the operation on the interface of the core of IPv6 multicast.

Command	Function
debug nsm mcast6 mif	Show the operation on the interface of the core of IPv6 multicast.

In the privileged EXEC configuration mode, run the following command to show the operation of interface and statistics of the core of IPv6 multicast.

Command	Function
debug nsm mcast6 stats	Show the operation of interface and statistics of the core of IPv6 multicast.

In the privileged EXEC configuration mode, run the following command to show the packet forwarding on Layer 3 of IPv6 multicast.

Command	Function
debug ipv6 mrf forwarding	Show the packet forwarding on Layer 3 of IPv6 multicast.

In the privileged EXEC configuration mode, run the following command to show the operation of forwarding entries on Layer 3 of IPv6 multicast.

Command	Function
debug ipv6 mrf mfc	Show the operation of forwarding entries on Layer 3 of IPv6 multicast.

In the privileged EXEC configuration mode, run the following command to show the operation of forwarding events on Layer 3 of IPv6 multicast.

Command	Function
debug ipv6 mrf event	Show the operation of forwarding events on Layer 3 of IPv6 multicast.

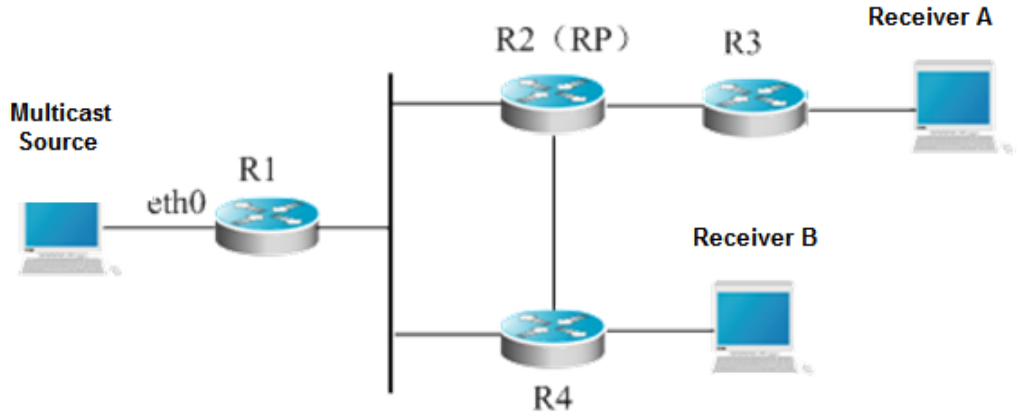
2.4 Multicast Route Configuration Example

2.5 PIM-SMv6 Configuration Example

2.5.1 Configuration Requirements

Figure 3 is network topology. R1 and the multicast source are located in one network. R2 is set to be RP. R3 and the Receiver A are located in the same network. R4 and the Receiver B are located in the same network. Assume that devices and hosts are connected properly, IPv6 is enabled on every interface and IPv6 unicast is enabled on every device.

Figure 3 Network topology of PIM-SMv6 configuration example



2.5.2 Device Configuration

Step1: Enable IPv6 multicast route.

Enable IPv6 multicast route on R1. The configurations on R2, R3 and R4 are similar.

```
Qtech# configure terminal
Qtech(config)# ipv6 multicast-routing
```

Step 2: Enable PIM-SMv6 on the interface.

Enable PIM-SMv6 on R1's Gi 0/1. The configurations on R2, R3 and R4 are similar.

```
Qtech(config)# interface GigabitEthernet 0/1
Qtech(config-if)# ipv6 pim sparse-mode
Qtech(config-if)# end
```

Step 3: Configure the candidate BSR and the candidate RP.

Set R2's loopback1 to be C-BSR and C-RP

```
Qtech(config)# interface loopback 1
Qtech(config-if)# ipv6 address 2008:1::1/64
Qtech(config-if)# ipv6 pim sparse-mode
Qtech(config-if)# exit
Qtech(config)# ipv6 pim bsr-candidate loopback 1
Qtech(config)# ipv6 pim rp-candidate loopback 1
Qtech(config-if)# end
```

Add the receiver into the multicast group. After the multicast source sends multicast streams, you can run **show** commands to monitor operation.

Note When you enable PIM-SMV6, MLD automatically runs on every interface, respectively.

3 CONFIGURING IGMP

3.1 IGMP Overview

IP multicast refers to a network technology that allows one or more sender (multicast source) to send one packet to more than one receiver simultaneously. The multicast source sends packets to a specific multicast group and only hosts joining the group can receive the packets. Multicast can save network bandwidth greatly because there is only a single packet transmitted on any link of the network, no matter how many receivers are deployed.

Multicast uses Class-D IP addresses specified by the Internet Assigned Numbers Authority (IANA). The four high-order bits of Class-D IP addresses are binary 1110. So, the range of multicast address is from 224.0.0.0 to 239.255.255.255. However, not all addresses in this range can be assigned to users. Some addresses are reserved for protocols or other use. For instance, the address 224.0.0.1 is assigned to all multicast hosts and 224.0.0.2 is assigned to all multicast routers.

Any hosts, no matter whether they are multicast group members or not, can be multicast sources. However, only multicast group members can receive multicast frames. A multicast group member is able to dynamically join or leave the group. Forwarding of multicast frames in the network is implemented by multicast routers running multicast routing protocols.

To enable IP multicast, hosts and routers must support the Internet Group Management Protocol (IGMP). This protocol is used by hosts to report their group memberships to multicast routers on the directly-connected network, allowing the multicast routers to determine how to forward multicast traffic. By using the information obtained from IGMP, multicast routers create an interface-based multicast group member list. The list is activated only when at least one host on an interface is a member of the group.

IGMPv1, IGMPv2 and IGMPv3 are currently supported. On the basis of IGMPv1, IGMPv2 adds a leave message for a host to actively request to leave a multicast group. IGMP behaviors includes behaviors of hosts and devices.

3.1.1 IGMPv1

There are only two types of IGMP messages defined in IGMPv1:
- Membership query
- Membership report

A host sends a membership report to indicate that it is interest in joining a group, and the router sends membership queries periodically to ensure that the group has at least one host. When there is no hosts in that group, the device will delete it.

3.1.2 IGMPv2

In IGMPv2, there are only four types of IGMP messages:
- Membership query
- Version 1 membership report
- Version 2 membership report
- Leave group

IGMPv2 is basically the same as IGMPv1, except that IGMPv2 creates a Leave group message for hosts. For IGMPv2, hosts report leave messages to routers which then send queries to check whether there is a host in the multicast group. This makes joining and leaving a group more efficient.

In the multicast network running IGMP, a multicast router is dedicated for sending IGMP query messages. This router is called a querier which is selected through an election mechanism. At first, all routers are queriers. If a router receives a query message from another router with a lower IP address, it becomes a non-querier. Consequently, there is only one querier which has the lowest IP address among all multicast routers on the network.

If a querier is invalid, new querier will be elected. . Non-queriers keep a timer for Other Querier Present Interval. Every time when a router receives a membership query packet, it resets the timer. If the timer expires, the router starts to send query messages and selects new querier again.

Queriers must periodically send membership queries to ensure that other routers on the network know that the querier still works. For this purpose, the querier maintains one query interval timer. When it sends membership query messages, this timer will be reset. When the interval timer times out, the querier sends another membership query.

When a new router appers, it sends a series of general query messages to solicit membership information. The number of general query packets depends on the Startup Query Count configured on the router. The initial general query interval is defined by the Startup Query Interval.

When a querier receives a leave group message from a host, it must send a group-specific membership query to see whether the host is the last one to leave the group. Before the querier stops forwarding packets to the group, it sends a series of such packets, the number of which is equal to the Last Member Query Count. The querier sends multiple group-specific membership queries to ensure that there is no member in the group. Such a query is sent every other the Last Member Query Interval seconds. When no response is received, the querier stops forwarding multicast packets to the group on the specified interface.

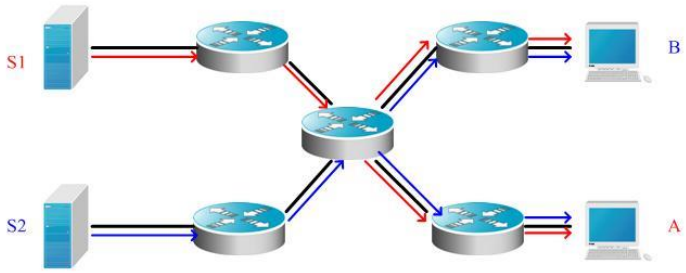
3.1.3 IGMPv3

Both IGMPv1 and IGMPv2 have the following defects:

- Lack of efficient measures to control multicast sources
- Difficult to establish multicast paths due to ignorance of multicast source locations.
- Difficult to find a unique multicast address. It is possible that multicast groups are using the same multicast address.

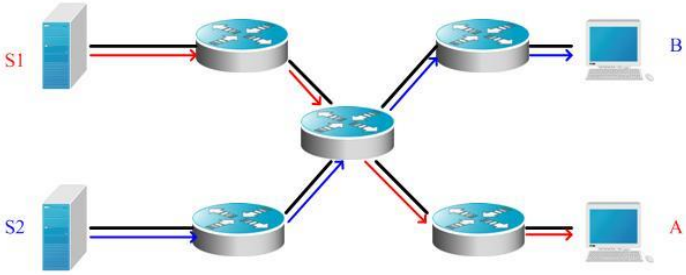
On the basis of IGMPv1 and IGMPv2, IGMPv3 provides an additional source filtering multicast function. In IGMPv1 or IGMPv2, hosts determines whether to join a group by group address and, once it joins the group, it receives multicast traffic forwarded from any source to that group address. In IGMPv3, hosts are enabled to report the multicast group they desire to join in and the multicast source from which they expect to receive traffic. A host specifies sources from which they want to receive multicast traffic through an INCLUDE list or an EXCLUDE list. Besides, IGMPv3 saves bandwidth by preventing unnecessary, illegal multicast data flows from occupying network bandwidth. It is particularly useful in the case where multiple multicast sources share one multicast address. IGMPv1 and IGMPv2 can also implement "source address filtering" in some sense, which, however, is performed on hosts receiving multicast traffic. As shown in the following diagram, two multicast sources (S1 and S2) send out traffic directed to the same multicast group address (G). This multicast traffic from S1 and S2 will be sent to all hosts receiving traffic from G. If host A only wants to receive multicast traffic from S1, it has to filter out traffic from S2 by running approriate client software.

Figure 9 Multicast traffic forwarded without source filtering



If multicast routers on the network support IGMPv3, host A wants to receive traffic from S1 only, it sends out an IGMPv3 packet in the form of “join G include S1”. host B wants to receive traffic from S2 only, it sends out an IGMPv3 packet in the form of “ join G include S2”. In this way, the traffic is forwarded as shown in Figure 2. This saves bandwidth.

Figure 10 Multicast traffic forwarded with source filtering



Based on IGMPv2, IGMPv3 adds the following two kinds of messages:

- Membership query

- Version 3 membership report

There are three types of membership query:
- General Query: used to learn information of all multicast members on an interface.
- Group-Specific Query: used to learn information of members of a specific group on an interface.
- Group-and-Source-Specific Query: a new type specified in IGMPv3 used to learn whether there is a member on an interface wants to receive group-specific multicast traffic from sources in the specified source list.

Membership Report in IGMPv3 is different from that defined in IGMPv2. The IGMPv3 membership reports are always sent with an destination address of 224.0.0.22. Besides, an IGMPv3 membership report can contain information of multiple groups.

IGMPv3 can identify membership report messages of IGMPv1 and IGMPv2 and leave group messges of IGMPv2.

IGMPv3 works the same way as IGMPv2. It is backward compatible with IGMPv1 and IGMPv2.


Caution

At most 1017 sources are allowed to forward traffic to a specific multicast group on Layer-3 interfaces of Qtech's switching routers. You can configure unicasting traffic from a specific allowed source. At most 1017 sources can be filtered from forwarding traffic to a specific multicast group on Layer-3 interfaces of Qtech's switching routers. You cannot configure unicasting traffic from the filtered sources.

3.2 IGMP Configuration Tasks

This section describes IGMP configuration tasks. Only some tasks are mandatory, and other tasks are optional depending on network requirements.


Caution

All commands described in this section must be configured on layer-3 interfaces.

3.2.1 Default Configuration

The following table describes the default configuration of IGMP.

Feature	Default Setting
IGMP version	IGMPv2 is supported on all interfaces.
Query response interval	10 seconds
Query interval	125 seconds
Access to multicast group	All multicast groups are permitted.
Other querier present interval	255 seconds
Robustness variable	2
Last member query interval	1 second
Last member query count	2
IGMP	Disabled

3.2.2 Enabling IGMP

Use the following commands to enable IGMP in interface configuration mode.

Command	Function
Qtech (config-if) # **ip pim** { **sparse-mode** | **dense-mode** }	Enables IGMP.
Qtech (config-if) # **no ip pim** { **sparse-mode** | **dense-mode** }	Disables IGMP.

3.2.3 Configuring IGMP Version

Use the following commands to configure the IGMP version in interface configuration mode.

Command	Function
Qtech (config-if) # **ip igmp version** { 1 | *2* | 3 }	Configures the IGMP version, version 2 by default.
Qtech (config-if) # **no ip igmp version**	Restores to the default value.

3.2.4 Configuring Last Member Query Interval

After receiving a leave message from a multicast group, the querier sends a group-specific membership query to verify whether there is any member in the group. If no report is received during the last member query interval, the querier will regard the host that is leaving the group is the last member of that group, and then delete the information of the group. The default value of the last member query interval is 10, in units of 1/10 second. .

Use the following commands to configure the last member query interval in interface configuration mode.

Command	Function
Qtech (config-if) # **ip igmp last-member-query-interval** *interval*	Configures the last member query interval. The *interval* specifies the range from 1 to 255. The unit is 1/10 second.
Qtech (config-if) # **no ip igmp last-member-query-interval**	Restores to the default value.

3.2.5 Configuring Last Member Query Count

To prevent loss of group-specific membership query packets, it is required to send the packets for several times to ensure reliability. Therefore, you are advised to configure the last member query count to greater than 1.

Use the following commands to configure the last member query count in interface configuration mode.

Command	Function
Qtech (config-if) # **ip igmp last-member-query-count** *count*	Configures the last member query count. The range is from 2 to 7. The default is 2.
Qtech (config-if) # **no ip igmp last-member-query-count**	Restores to the default value.

3.2.6 Configuring General Query Interval

A querier sends general query messages at intervals to all hosts to verify the current membership. The destination address of the messages is the all-hosts group address, 224.0.0.1, Time To Live (TTL) is 1 and the default value is 125 seconds.

Use the following commands to configure the general query interval in interface configuration mode.

Command	Function
Qtech (config-if) # **ip igmp query-interval** *seconds*	Configures the general query interval in seconds. The range is from 1 to 18000 seconds. The default is 125 seconds.
Qtech (config-if) # **no ip igmp query-interval**	Restores to the default value.

3.2.7 Configuring the Max Response Time

The max response time is specified in the membership query message sent by the querier. Shortening this response time can allow the querier to know change of members earlier. However, it can also result in increase of the member reports diffusing in the network. Network administrators can consider a tradeoff between the two factors and then decide a proper value for the period, 10 seconds by default. Another consideration in configuring the response time is that it must be shorter than the query interval.

Use the following commands to configure the max response time in interface configuration mode.

Command	Function
Qtech (config-if) # **ip igmp query-max-response-time** *seconds*	Configures the max response time in seconds. The range is from 1 to 25 seconds. The default value is 10 seconds.
Qtech (config-if) # **no ip igmp query-max-response-time**	Restores to the default value.

3.2.8 Configuring Other Querier Present Interval

Once the timer times out, the querier considers that there is no other queriers on the network. This is helpful for the election of querier. You can reduce the value of this timer in the circumstance where the querier changes frequently to speed up response.

Use the following commands to configure the other querier present interval in interface configuration mode.

Command	Function
Qtech (config-if) # **ip igmp query-timeout** *seconds*	Configures other querier present interval in seconds. The range is from 60 to 300 seconds. The default is 255 seconds.
Qtech (config-if) # **no ip igmp query-timeout**	Restores to the default value.

3.2.9 Configuring Access Control to Multicast Groups

By default, hosts on an interface can join any multicast group. You can limit the range of multicast groups that hosts can join by configuring a standard IP ACL and applying it to the specific interface.

Use the following commands to create a standard access control list.

Command	Function
Qtech# **config terminal**	Enters global configuration mode.
Qtech (config) # **access-list** *access-list-num* **permit** *A.B.C.D A.B.C.D*	Defines an ACL.
Qtech (config)# **interface** *interface-id*	Enters interface configuration mode.
Qtech (config-if) # **ip igmp access-group** *access-list-name*	Applies the access control list to an interface. The *access-list-name* specifies the range of group addresses that hosts on the interface can join.
Qtech (config-if) # **no ip igmp access-group**	Deletes the access control list.

Suppose this command is associated with extended ACL. When the received IGMP report message is (S1,S2,S3…Sn,G), this command has to use an ACL entry to match the (0, G) message. Therefore you must explicitly configure an extended ACL with the (0,G) entry so that the (S1,S2,S3…Sn,G) message can be filtered.

3.2.10 Configuring Immediate-leave Group

In IGMPv2, you can execute this command to r reduce the leave latency of multicast group members . After this command is enabled, a host leaves a multicast group as long as it sends a leave message, without waiting the querier to send a group-specific query message. This command is available only when there is only one receiver host on an interface.

Use the following commands to configure an immdiate-leave group list. .

Command	Function
Qtech# **config terminal**	Enters global configuration mode.
Qtech (config) # **access-list** *access-list-num* **permit** *A.B.C.D A.B.C.D*	Defines an ACL.
Qtech (config)# **interface** *interface-id*	Enters interface configuration mode.
Qtech(config-if)# **ip igmp immediate-leave group-list** *access-list-name*	Creates an immediate-leave group list.
Qtech (config-if) # **end**	Enters privileged EXEC mode.

3.2.11 Configuring Join-Group

This command configures a routeras a member of a multicast group. Use the **no** form of this command to remove the router from the multicast group.

Command	Function
Qtech# **config terminal**	Enters global configuration mode.
Qtech (config)# **interface** *interface-id*	Enters interface configuration mode.
Qtech(config-if)# **ip igmp join-group** *group-address*	Configures a router to join a multicast group.
Qtech (config-if) # **end**	Enters privileged EXEC mode.

Use the **no ip igmp join-group** *group-address* command to remove the router from the multicast group.

3.2.12 Configuring Static-Group

This command configures a statically joined group. Use the **no** form of this command to remove the statically joined multicast group.

Command	Function
Qtech# **config terminal**	Enters global configuration mode.
Qtech (config)# **interface** *interface-id*	Enters interface configuration mode.
Qtech(config-if)# **ip igmp static-group** *group-address*	Configures a statically joined group.
Qtech (config-if) # **end**	Enters privileged EXEC mode.

Use the **no ip igmp static-group** *group-address* command to remove the statically joined group. .

3.2.13 Configuring Limit on the Number of IGMP Group Members

Use this command to limit the number of IGMP group members globally. Membership messages that exceed the limit will not be cached or forwarded.

You can configure this command on each interface in interface or global configuration mode.

Command	Function
Qtech(config) # **ip igmp limit** *number*	Limits the number of IGMP members in global configuration mode. The range depends on specific products.
Qtech(config-if) # **ip igmp limit** *number*	Limits the number of IGMP members in interface configuration mode. . The range depends on specific products. By default, it is 1024.

Use the **no ip igmp limit** command to restore the default configuration.

3.2.14 Configuring IGMP PROXY-SERVICE

This command enables services on all the downlink mroute-proxy interfaces. After you configure this command on an interface, the interface becomes the uplink interface of the corresponding mroute-proxy service. Moreover, it associates all its downlink interfaces and maintains their propagated multicast group information.

Up to 32 proxy services can be configured using this command. The interface number with the IGMP Proxy enabled is limited by the multicast interface number supported by the device. Upon the receipt of query message, the proxy-service interface responds accordingly based on the member information that it maintains from the interfaces with mroute-proxy configured. Consequently, configuring proxy-service on an interface equals to performing host behaviors rather than router behaviors on the interface. Use the following command to configure IGMP proxy-service in interface configuration mode.

Command	Function

Command	Function
Qtech(config-if)# **ip igmp proxy-service**	Configures proxy-service on the interface.

3.2.15 Configuring IGMP MROUTE-PROXY

This command lets an interface to forward messages to its corresponding uplink interface. The uplink interface can forward IGMP messages received from its members only when it is set to a proxy-service interface.

Use the following command to configure IGMP mroute-proxy in interface configuration mode.

Command	Function
Qtech(config-if)# **ip igmp mroute-proxy** *interfacename*	Configures mroute-proxy on the interface. *interfacename* specifies the name of the uplink interface.

3.2.16 Enabling IGMP SSM-MAP

This command forcibly appends the relevant multicast source messages to the dynamically learned multicast group messages. It is usually used in conjunction with the **ip igmp ssm-map static** command.

Use the following command to enable IGMP SSM-MAP in global configuration mode.

Command	Function
Qtech(config)# **ip igmp** [**vrf** *vrf-name*] **ssm-map enable**	Enables the SSM-MAP function under specified VRF.

3.2.17 Configuring IGMP SSM-MAP STATIC

This command is used in conjunction with the **ip igmp ssm-map enable** command. After this command is configured, the received messages whose version is earlier than version 3 will be mapped to the corresponding multicast source record.

Use the following command to configure IGMP SSM-MAP static in global configuration mode.

Command	Function
Qtech(config)# **ip igmp** [**vrf** *vrf-name*] **ssm-map static** *access-list-num A.B.C.D*	Maps all groups matched ACL *access-list-num* under specified VRF to source address *A.B.C.D*.

3.3 Monitoring and Maintaining IGMP and Membership Information

3.3.1 Clearing the Dynamic Group Member Messages Obtained from the Response Message in the IGMP Cache

Use the following command to clear the dynamic group member messages obtained from the response message in the IGMP cache in privileged EXEC mode.

Command	Function
Qtech# **clear ip igmp** [**vrf** *vrf-name*] **group**	Clears the dynamic group member messages obtained from the response message in the IGMP cache in the specified VRF.

3.3.2 Clearing All the Information on the Interface in IGMP Cache

Use the following command to clear all the information on the interface in IGMP cache in privileged EXEC mode.

Command	Function
Qtech# **clear ip igmp** [**vrf** *vrf-name*] **interface** *interface-type interface-number*	Clears all the information on the interface in IGMP cache in the specified VRF.

3.3.3 Displaying the Status of All IGMP Group Members in the Directly-Connected Subnet

Use the following command to show the status of all IGMP group members in the directly-connected subnet in privileged EXEC mode.

Command	Function
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups**	Shows the status of all IGMP groups under the specified VRF in the directly-connected subnet.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups detail**	Shows the details of all IGMP groups under the specified VRF in the directly-connected subnet.

Command	Function
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups** *A.B.C.D*	Shows the status of the specified group under the specified VRF in the directly-connected subnet.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups** *A.B.C.D* **detail**	Shows the details of the specified group under the specified VRF in the directly-connected subnet.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **interface** *interface-type interface-number*	Shows the information of the specified interface under the specified VRF in the directly-connected subnet.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups** *interface-type interface-number* **detail**	Shows the details of all the groups of the specified interface under the specified VRF in the directly-connected subnet.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups** *interface-type interface-number A.B.C.D*	Shows the information of the specific group of the specified interface under the specified VRF in the directly-connected subnet.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **groups** *interface-type interface-number A.B.C.D* **detail**	Shows the details of the specific group of the specified interface under the specified VRF in the directly-connected subnet.

3.3.4 Displaying IGMP Interface Configuration

Use the following commands to show the IGMP interface configuration in privileged EXEC mode.

Command	Function
Qtech# **show ip igmp** [**vrf** *vrf-name*] **interface** [*interface-type interface-number*]	Shows the configuration information of the IGMP interface in the specified VRF.
Qtech# **show ip igmp** [**vrf** *vrf-name*] **interface**	Shows the configuration information of all the IGMP interfaces in the specified VRF.

3.3.5 Displaying IGMP SSM-MAP Configuration

Use the following commands to show the IGMP SSM-MAP configuration in privileged EXEC mode.

Command	Function
Qtech# **show ip igmp** [**vrf** *vrf-name*] **ssm-mapping**	Shows the configuration information of IGMP SSM-MAP under the specified VRF.
Qtech# **show ip igmp ssm-mapping** *A.B.C.D*	Shows the mapping information from IGMP SSM-MAP under the specified VRF to the multicast group *A.B.C.D*.

3.3.6 Displaying the Status of the IGMP Debugging Switch

Use the following command to show the status of the IGMP debugging switch in privileged EXEC mode.

Command	Function
Qtech# **show debugging**	Shows the status of the IGMP debugging switch.

3.3.7 Turning on IGMP Debugging Switches

Use the following commands to turn on IGMP debugging switches to observe IGMP behaviors in privileged EXEC mode.

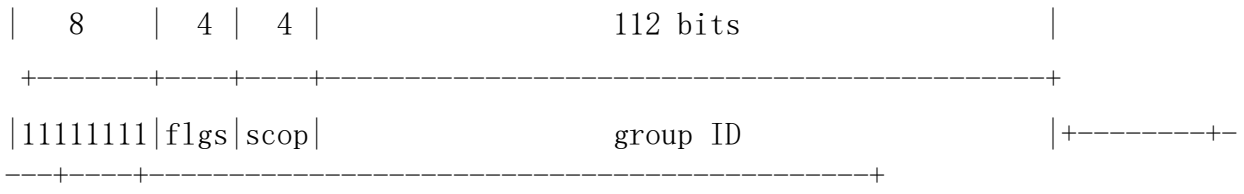
Command	Function
Qtech# **debug ip igmp** [**vrf** *vrf-name*] **all**	Turns on all IGMP debugging switches in the specified VRF.
Qtech# **debug ip igmp** [**vrf** *vrf-name*] **decode**	Turns on decode debugging switch in the specified VRF.
Qtech# **debug ip igmp** [**vrf** *vrf-name*] **encode**	Turns on encode debugging switch in the specified VRF.
Qtech# **debug ip igmp** [**vrf** *vrf-name*] **events**	Turns on event debugging switch in the specified VRF.
Qtech# **debug ip igmp** [**vrf** *vrf-name*] **fsm**	Turns on final-state-machine debugging switch in the specified VRF.
Qtech# **debug igmp** [**vrf** *vrf-name*] **tib**	Turns on tree debugging switch in the specified VRF.
Qtech# **debug ip igmp** [**vrf** *vrf-name*] **warning**	Turns on warning debugging switch in the specified VRF.

4 CONFIGURING MLD

4.1 MLD Overview

IPv6 multicast is a network technology allowing one or more senders (multicast sources) to send a single packet to multiple receivers (once and simultaneous). The multicast source sends the packet to a specific multicast group, and only hosts having addresses of this multicast group can receive the packet. Multicasting can substantially save network bandwidth, because only one packet is sent on any link of the entire network no matter how many destination addresses exist.

The format of IPv6 multicast address is different from that of IPv6 unicast address, and can only be used for destination address. The first byte of address format all consists of "1" (multicast address). The multicast address occupies 1/256 of the IPv6 address space. Except for parts other than the first byte, the multicast address format includes the following fields:



- Flag field: (starting from high-order bit) The first bit is the reserved bit set to 0; the second bit is R flag: R = 1 indicates a multicast address that embeds the address of the RP, while R = 0 indicates a multicast address that does not embed the address of the RP; the third bit is P bit: R = 1 indicates a multicast address is created on basis of unicast prefix, otherwise it is set to 0.
- The scop field has 4 bits and indicates the scope of multicasting. According to the definition given in RFC 4291: 1 (hexadecimal number system) indicates interface-local scope, 2 indicates link-local scope, 4 indicates admin-local scope, 5 indicates site-local scope, 8 indicates organization-local scope, and E indicates global scope.
- Group ID has 112 bits, and is used to identify the multicast group. According to the multicast address (transient or well-known) and the scope of address, the same multicast ID can indicate different groups. Well-known multicast address uses specific group ID with special meanings.

No matter whether a member of multicast group or not, any host can be a multicast source. However, only the member of multicast group can receive the multicast frames. Members of the multicast group are determined dynamically, and the host can dynamically join or leave the group. The multicast frames are forwarded by the multicast device, which will run the multicast routing protocol.

To participate in IPv6 multicast, the multicast host/device shall be able to support MLD operations. This protocol allows the interactive multicasting of member relationship between host and routing device, so as to decide on the forwarding of multicast steam. By utilizing the information obtained from MLD, the device will maintain a multicast listener state table, which is based on each interface. When the local link of an interface has at least one host being a member of the group, the multicast listener state table will then be activated.

Currently, MLD has to versions. MLDv2 was developed on the basis of MLDv1 by adding the source filtering mechanism. The behaviors of MLD protocol can be divided into two parts: host behavior and router behavior.

4.2 Introduction to Messages of Different MLD versions

4.2.1 MLD Version 1

In MLDv1, there are three types of messages:

- Multicast Listener Query
- Multicast Listener Report
- Multicast Listener Done

In the multicast network running MLD protocol, there will be querier responsible for sending MLD query messages. Such querier is determined through election. In the beginning, all devices are of the querier state. When the devices receives multicast listener query from a device with a lower IP address, they will change from querier state to non-

querier state. Therefore, only one device will be of the querier state, and this device has the lowest IP address among all multicast devices on the network.

MLDv1 also has the corresponding mechanism to handle the failure of querier device. Non-querier devices will maintain the current interval timer of other queriers. The device will reset this timer when every time it receives the multicast listener query message. If this timer times out, then the corresponding device will restart sending query message, and the new round of querier device election will begin again.

The querier device must periodically send out the multicast listener query to ensure that non-querier devices on the network know that the querier device is still workable. In realize this function, the querier device maintains a query interval timer, which will be reset when the multicast listener query message is sent out. When the query interval timer is set to zero or no longer useful, the querier device will send out another multicast listener query.

When MLD protocol is initiated, it will send a number of general query messages to discover which multicast groups shall be forwarded on the specific interface. The number of ordinary query messages sent by the device will be based on the Startup Query Count configured by the current device. The space between startup general queries will be determined by the value of Startup Query Interval. 。

When the querier device receives the leave message, it must send a group-specific multicast listener query message to see whether the host is the last listener to leave the group. The device will send a number of group-specific query messages before it stops forwarding data messages for the specific group, and the number of messages equals to the number of last listener queries. The device will send multiple group-specific multicast listener queries to ensure there is no more listener in that group. A group-specific query will be sent at every other last listener query interval in order to partition the queries. When no response is received, the device will delete the corresponding group record, and stop forwarding multicast data messages on that specific interface for this group address.

4.2.2 MLD Version 2

MLDV1 has the following defects in application:

- Lack of effective means to control multicast sources.
- Being unaware of the location of multicast source, it is comparatively difficult to establish the multicast route.
- It is very difficult to discover an only multicast address. Multiple multicast groups may use the same multicast address.

On the basis of MLDv1, MLDv2 provides the additional multicast source filtering mode (INCLUDE/EXCLUDE). In MLDv1, the host determines to join a certain group only according to the group address, and receives multicast streams sent from any source to this group address. In MLDv2, the host informs the host of the multicast group it desires to join in, and also notifies this host of addresses of multicast sources that it is willing/unwilling to receive. The host can indicate which sources to receive multicast streams via an INCLUDE list or an EXCLUDE list. When the host joins a multicast group:

- If the host only needs to receive the data streams sent from source { s1,s2,s3…}, then its Report message can contain the information of INCLUDE{ s1,s2,s3…}.
- If the host doesn't want to receive the data streams sent from source { s1,s2,s3…}, then its Report message can contain the information of EXCLUDE{ s1,s2,s3…}.

Through source address filtering, MLDv2 can save network bandwidth and prevent unnecessary and invalid multicast data streams from occupying the network bandwidth. This is especially useful when multiple multicast sources are sharing one multicast address. Although MLDv1 can also realize "source address filtering" in a certain sense, it is done on the receiving end of multicast streams. As shown in Fig 1, S1 and S2 are multicast sources sending data streams with the same multicast address of G. The multicast streams of S1 and S2 will be sent to all hosts receiving G. If host A only wants to receive the data streams of S1, in order to avoid the disturbance of data streams from S2, we can only use the corresponding Client software to perform filtering.

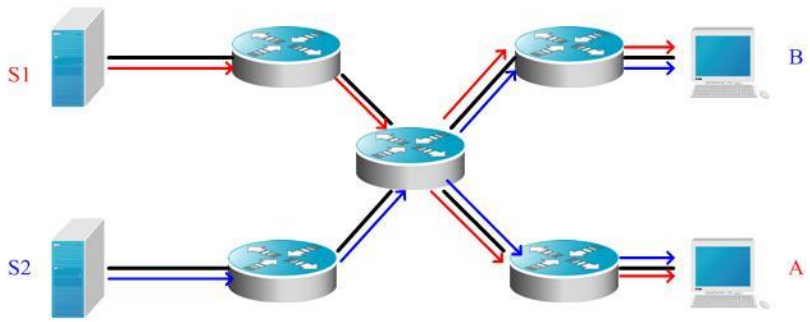


Fig 4 MLDv1 forwarding diagram

If the devices on network can support MLDv2: host A can send the MLDv2 message of join G include { S1 } if it only wants to receive data streams from S1, and host B can send the MLDv2 message of join G include { S2 } if it only wants to receive data streams from S2. The forwarding of data streams will be shown as follows, allowing the saving of partial bandwidth.

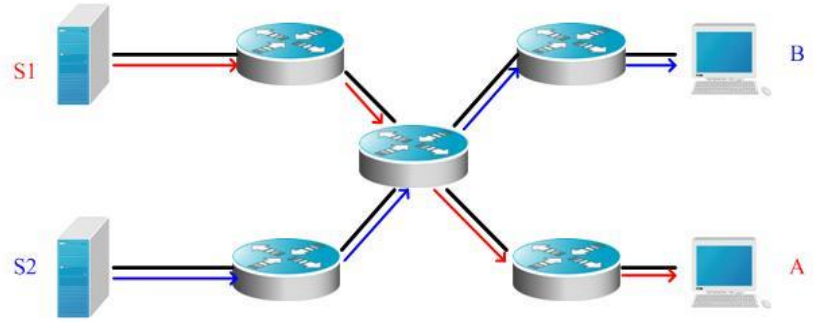


Fig 5 MLDv2 forwarding diagram

Compared with version 2, version 2 defines the following two types of messages:

- Membership Listener Query
- Membership Listener Report

Membership Query can be classified into:

- General Query: Used to learn the information of all multicast listeners on the attached link;
- Group-Specific Query: Used to learn the information of group-specific listeners on the attached link;
- Group-and-Source-Specific Query: This type is newly added by MLDv2, and is used to learn if any of listeners on the attached link needs to receive the group-specific multicast streams sent by the specified source list.

Different from the Membership Report described in MLD Version1, the Membership Report message sent by MLD Version2 may contain the information of multiple groups. The destination address of Report message is FF02::16 in MLDv2, and can carry one ore more group records, each of which contains the group address and source address. The types of group records are shown below:

- IS_IN: indicates that filtering mode between multicast group and multicast source list is INCLUDE, i.e., only the multicast data sent from the specified multicast source list to this multicast group will be received.
- IS_EX: indicates that filtering mode between multicast group and multicast source list is EXCLUDE, i.e., only the multicast data sent from multicast sources outside the specified multicast source list to this multicast group will be received.
- TO_IN: indicates that the filtering mode between multicast group and multicast source list will change from EXCLUDE mode to INCLUDE mode.
- TO_EX: indicates that the filtering mode between multicast group and multicast source list will change from INCLUDE mode to EXCLUDE mode.
- ALLOW: allows the receipt of multicast data from certain multicast sources. If the current corresponding relationship is INCLUDE, these multicast sources will be added into the existing multicast source list; if the current corresponding relationship is EXCLUDE, these multicast sources will be deleted from the existing multicast source list.
- BLOCK: no longer expects to receive multicast data from certain multicast sources. If the current corresponding relationship is INCLUDE, these multicast sources will be deleted from the existing multicast source list; if the current corresponding relationship is EXCLUDE, these multicast sources will be added into the existing multicast source list.

In consideration of compatibility, MLD Version2 can also recognize the Membership Report message and Done message of version 1.

Caution

The multicast layer-3 interface of Qtech router can only allow up to 253 specific sources for specific groups. The user can perform the unicast of allowed specific sources.


Caution

The multicast layer-3 interface of Qtech router can only filter up to 253 specific sources for specific groups. The user cannot perform the unicast of filtered specific sources.

4.2.3 MLD Protocol Specifications

The current MLD protocols include:
- RFC2710: Multicast Listener Discovery (MLD) for IPv6
- RFC3810: Multicast Listener Discovery Version 2 (MLDv2) for IPv6

4.3 MLD Configuration Task List

MLD configuration tasks include the following items, but only some configurations are compulsory, while other tasks are optional according to the specific needs of the network. Note: The following command must be configured on the layer-3 interface.
- (mandatory) Enabling MLD protocol
- (mandatory) Configuring MLD version
- (optional) Configuring Last Listener Query Interval
- (optional) Configuring Last Listener Query Count
- (optional) Configuring General Listener Query Interval
- (optional) Configuring Maximum Response Interval
- (optional) Configuring Other Querier Timer Interval
- (optional) Configuring Multicast Group Access Control
- (optional) Configuring Immediate Leave Group
- (optional) Configuring MLD Listener Number Limit
- (optional) Configure Host-Behavior Multicast Group Joining
- (optional) Configuring Static Multicast Group Joining
- (optional) Configuring MLD P
- (optional) Configuring MLD P
- (optional) Enabling MLD SSM-MAP
- (optional) Configuring MLD SSM-MAP S

4.3.1 Default MLD Configurations

Function	Default setting
MLD version	All interfaces support version 2
Query Response Interval	10 seconds
Query Interval	125 seconds
Multicast Group Access Control	All groups allowed
Other Querier Present Interval	255 seconds
Robustness Variable	2
Last Listener Query Interval	10 (0.1s)
Last Listener Query Count	2
MLD state	Disabled

4.3.2 Enabling MLD Protocol

MLD needs to be used with multicast routing protocol. If the multicast routing protocol is enabled, the MLD protocol will also be enabled. Run the following command in the interface configuration mode:

Command	Function
Qtech (config-if) # **ipv6 pim sparse-mode**	Enable the multicast routing protocol and MLD.
Qtech (config-if) # **no ipv6 pim sparse-mode**	Disable the multicast routing protocol and MLD.

4.3.3 Configuring MLD Version

To configure MLD version, run the following command in the interface configuration mode:

Command	Function
Qtech (config-if) # **ipv6 mld version** {**1**|**2**}	Configure the MLD version.

Qtech (config-if) # **no ipv6 mld version**	Recover the MLD version to the default version2.

4.3.4 Configuring Last Listener Query Interval

After receiving a group leave message, the querier device will send a group-specific listener query to identify that whether there are still listeners in the group. During the interval of sending last listener query, if no report is received, the device will assume that the leaving device is the last listener of the group, and will the information of this group. The default value is 1s. The Last Listener Query Interval determines the leaving speed of listeners.

Run the following command in the interface configuration mode:

Command	Function
Qtech (config-if) # **ipv6 mld last-member-query-interval** *interval*	Configure the last listener query interval *interval*: the valid range is 1-255, in 0.1s.
Qtech (config-if) # **no ipv6 mld last-member-query-interval**	Configure Last Listener Query Interval to default value.

4.3.5 Configuring Last Listener Query Count

To avoid the loss of group-specific listener query messages sent by the querier device, the messages need to be sent for several times to guarantee reliability. Therefore, the Last Listener Query Count shall be configured.

Run the following command in the interface configuration mode:

Command	Function
Qtech(config-if) # **ipv6 mld last-member-query-count** *count*	Configure the last listener query count. *count:* the valid range is 2-7, and the default value is 2.
Qtech(config-if) # **no ipv6 mld last-member-query-count**	Configure the last listener query count to the default value.

4.3.6 Configuring General Listener Query Interval

At every other listener query interval, the querier will periodically send out the listener query messages to verify the relationship of current listeners. The destination address of listener query message is all-hosts, the multicast address is FF02::1, and the TTL is 1. The default value is 125 seconds.

Run the following command in the interface configuration mode:

Command	Function
Qtech (config-if) # **ipv6 mld query-interval** *seconds*	Configure the general listener query interval. *seconds:* the valid range is 1-18000.
Qtech (config-if) # **no ipv6 mld query-interval**	Configure the general listener query interval to the default value of 125s.

4.3.7 Configuring Maximum Response Interval

This means the maximum response time required in the multicast listener query messages sent by the querier device. Less time can allow the device to rapidly learn the changes in listeners but will lead to corresponding increase in the number of potential listener reports. The network administrator can consider both factors to determine the most appropriate value, which is 10 seconds by default. Furthermore, this time shall be less than the Query Interval.

Run the following command in the interface configuration mode:

Command	Function
Qtech (config-if)**# ipv6 mld query-max-response-time** *seconds*	Configure the maximum response time. *seconds:* the valid range is 1-25.
Qtech (config-if)**# no ipv6 mld query-max-response-time**	Configure the maximum response time to the default value of 10s.

4.3.8 Configuring Other Querier Timer Interval

The configuration of other querier timer interval can help adjust the querier device election time. This value can be decreased to increase the response speed when the querier devices are prone to change.

Run the following command in the interface configuration mode:

Command	Function

Qtech (config-if)# **ipv6 mld query-timeout** *seconds*	Configure other querier timer interval. *seconds:* the valid range is 60-300, and the default value is 255s.
Qtech (config-if)# **no ipv6 mld query-timeout**	Configure other querier timer interval to the default value.

4.3.9 Configuring Multicast Group Access Control

By default, the host on one interface can join any multicast group. This feature can be used when the administrator expects to limit the scope of multicast groups which can be joined by the host. Configure an IP access list to allow and limit the scope of multicast group addresses, and apply the list to the specific interface.

Run the following commands:

Command	Function
Qtech# **config terminal**	Enter the global configuration mode.
Qtech (config) # **ipv6 access-list** *access-list-name* Qtech (config-std-nacl) # **permit ipv6** source_address *group_address*	Define an access control list.
Qtech (config)# **interface** *interface-type interface-id*	Enter the interface configuration mode.
Qtech (config-if) # **ipv6 mld access-group** *access-list-name*	Configure the group addresses or source addresses so that multicast groups covered by the address scope of access-list-name can access this interface.
Qtech (config-if) # **no ipv6 mld access-group**	Delete the access control list and allow the access of all groups.


Caution

The Multicast Group Access Control is associated with Extended ACL. When the MLD report message received is (S1,S2,S3…Sn,G), this command will perform the corresponding ACL based check of (0,G) information. Therefore, an explicit entry of (0,G) shall be configured for extended ACL to allow the normal filtering of (S1,S2,S3…Sn,G). This (0,G) mentioned herein refers to all sources of the specific group.

4.3.10 Configuring Immediate Leave Group

In MLD, this command can reduce the leave latency. When there is only one host on the attached link needs to receive group information, after the host sends out the leave message, it can leave immediately without the need for the querier to device to send group-specific query. This command is used when one interface has only one receiver host.

Command	Function
Qtech# **config terminal**	Enter the global configuration mode.
Qtech (config)# **interface** *interface-id*	Enter the interface configuration mode.
Qtech (config-if)# **ipv6 mld immediate-leave group-list** *access-list-name*	Configure the immediate leave group list to access-list-name.
Qtech (config-if) # **exit**	Return to the global configuration mode.
Qtech (config)# **ipv6 access-list** *access-list-name* Qtech (config-std-nacl)#**permit** *x:x:x:x::x:x/< 0-128>*	Configure the address scope of group list.

4.3.11 Configuring MLD Listener Number Limit

This command can be used to limit the number of listeners that can be learned by MLD. Listener messages will not be learned after the listener number has exceeded the limit, and no group record will be generated.

This command can be used to configure each interface, and the interface and global can be configured separately. If exceeding the number limit of interface or global configuration, the listener messages will be dropped. Run the following commands:

Command	Function

Qtech (config)# **ipv6 mld limit** *number*	Global configuration of MLD listener number limit Please note that different models provide different limit ranges. Please identify the default value according to the capacity indicator of different models.
Qtech (config-if) # **ipv6 mld limit** *number*	Interface configuration of MLD listener number limit Please note that different models provide different limit ranges. The default value is 1024.
Qtech (config-if) # **ipv6 mld limit** *number* **except** *access-list-name*	The key word of Except indicates that groups covered by ACL will be limited in number.

Use the **no ipv6 mld limit** command to restore the relevant configurations to the default values.

4.3.12 Configuring Host-Behavior Multicast Group Joining

Use this command to configure relevant switch interfaces to implement host behavior and join the corresponding multicast group. In this way, the sub-switches can initiatively learn the corresponding group information. This configuration can be used when a listener is required to be assigned to the interface. Use the **no** form of this command to cancel the joining operation.

Command	Function
Qtech# **config terminal**	Enter the global configuration mode.
Qtech (config)# **interface** *interface-id*	Enter the interface configuration mode.
Qtech (config-if)# **ipv6 mld join-group** *group-address*	Enable the interface to join host group.
Qtech (config-if) # **exit**	Return to the global configuration mode.

Use the **no ipv6 mld join-group** *group-address* command to leave the corresponding multicast group.

Caution

Some of the local link addresses have been reserved for use by the IP layer. If such addresses are used as the local link address, no group record will be generated.

4.3.13 Configuring Static Multicast Group Joining

Use this command to directly assign a listener to the relevant switch interface when a listener is required to be assigned to the interface. Use the **no** form of this command to cancel the joining operation.

Command	Function
Qtech# **config terminal**	Enter the global configuration mode.
Qtech (config)# **interface** *interface-id*	Enter the interface configuration mode.
Qtech (config-if)# **ipv6 mld static-group** *group-address*	Enable the interface to join static group.
Qtech (config-if) # **exit**	Return to the global configuration mode.

Use the **no ipv6 mld static-group** *group-address* command to delete the static group assigned to the interface.

4.3.14 Configuring MLD Proxy-service

Use this command to enable the service of all downlink mroute-proxy interfaces, enabling the interface to become the uplink interface of corresponding mroute-proxy. It will bind all attached downlink interfaces and maintain the group information advertised by the downlink interface.

The maximum configurable number of this command is limited to 32. The number of interfaces with MLD Proxy function enabled depends on the number of multicast interfaces supported by the device. When the interface receives the query message, the proxy-service interface will give the corresponding reply according to the listener information it maintains. Such information is acquired from the mroute-proxy interface. Therefore, proxy-service interface means that it will only implement host behavior instead of router behavior. Run the following command in the interface configuration mode:

Command	Function
Qtech (config-if)# **ipv6 mld proxy-service**	Configure the interface to proxy-service state.

Use the **no ipv6 mld proxy-service** command to disable the proxy function on the corresponding interface.

4.3.15 Configuring MLD Mroute-proxy

This command will allow the interface to forward messages to the corresponding uplink interface. When the corresponding uplink interface is configured to proxy-service interface, this interface can then forward various MLD protocol messages forwarded by its listeners.

Run the following command in the interface configuration mode:

Command	Function
Qtech (config-if) # **ipv6 mld mroute-proxy** *interfacename*	Enables the interface to forward messages to the correspongding uplink interface.

Use the **no ipv6 mld mroute-proxy** command to cancel the binding between downlink interface and uplink interface, and disable the proxy function of downlink interface.

4.3.16 Enabling MLD SSM-MAP

Use this command to add the dynamically learned group information to the bound source record information, and is generally co-used with the **ipv6 mld ssm-map static** command.

Run the following command in the global configuration mode:

Command	Function
Qtech (config)# **ipv6 mld ssm-map enable**	Enable the ssm-map function.

Use the **no ipv6 mld ssm-map enable** command to disable the SSM-MAP function.

4.3.17 Configuring MLD SSM-MAP Static

Use this command together with the **ipv6 mld ssm-map enable** command. After configuring this command, the MLDv1 messages received will be mapped to the corresponding source records.

Run the following command in the global configuration mode:

Command	Function
Qtech (config) # **ipv6 mld ssm-map static** *acl_name src_address*	Map all group records complying with acl_name to the source address of src_address.

Use the **no ipv6 mld ssm-map static** command to cancel the mapping relationship between related group record and source address.

4.4 Monitoring and Maintaining MLD State and Listener Information

4.4.1 Clearing the Dynamic Listener Information in the MLD Cache

Use this comamnd to clear the dynamic listener information in the MLD cache. Note that this command cannot delete the statically added listener information. Run the following command in the privilege mode:

Command	Function
Qtech# **clear ipv6 mld group** [group-address] [interface-type interface-number]	Clear all dynamic listener information in the MLD cache learned from response message. Clear all mld group information if there is no parameter.

4.4.2 Clearing All Information in MLD Cache on the Specific Interface

To clear all information in MLD cache on the specific interface, run the following command in the privilege mode:

Command	Function
Qtech# **clear ipv6 mld interface** *interface-type*	Clear all information in MLD cache on the interface.

4.4.3 Displaying the State of Listeners on the Attached Subnetwork

To display the state of all listeners on the attached subnetwork, run the following commands in the privilege mode:

Command	Function
Qtech# **show ipv6 mld groups**	Display the information of all listeners learned by MLD.

Qtech# **show ipv6 mld groups detail**	Display the detailed information of all listeners learned by MLD.
Qtech# **show ipv6 mld groups** *x:x:x:x::x*	Display the information of group-specific listeners learned by MLD.
Qtech# **show ipv6 mld groups** *x:x:x:x::x* detail	Display the detailed information of group-specific listeners learned by MLD.
Qtech# **show ipv6 mld ssm-mapping**	Display the configuration information of ssm-mapping.
Qtech# **show ipv6 mld ssm-mapping** *x:x:x:x::x*	Display the source address mapping information of specific group.
Qtech# **show ipv6 mld groups** *interface-type interface-number* **detail**	Display the detailed information of all groups on the specific interface.
Qtech# **show ipv6 mld groups** *interface-type interface-number x:x:x:x::x*	Display the information of specific groups on the specific interface.
Qtech# **show ipv6 mld groups** *interface-type interface-number x:x:x:x::x* **detail**	Display the detailed information of specific groups on the specific interface.

4.4.4 Displaying the Configuration Information of MLD Interface

To display the configuration information of MLD interface, run the following commands in the privilege mode:

Command	Function
Qtech# **show ipv6 mld interface** *interface-type interface-number*	Display the MLD interface configurations.
Qtech# **show ipv6 mld interface**	Display the configurations of all MLD interfaces.

4.4.5 Displaying the On/Off State of MLD Debug Switch

To display the on/off state of MLD debug switch, run the following command in the privilege mode:

Command	Function
Qtech# **show debugging**	Display the on/off state of MLD debug switch.

4.4.6 Turning on MLD Debug Information Switch and Observing MLD Behaviors

To turn on MLD debug information switch and observe MLD behaviors, run the following command in the privilege mode:

Command	Function
Qtech# **debug ipv6 mld all**	Turn on all MLD debug information switches
Qtech# **debug ipv6 mld decode**	Turn on MLD debug message decoding switch.
Qtech# **debug ipv6 mld encode**	Turn on MLD debug message encoding switch.
Qtech# **debug ipv6 mld events**	Turn on MLD debug event information switch.
Qtech# **debug ipv6 mld fsm**	Turn on MLD debug state switch.
Qtech# **debug ipv6 mld tib**	Turn on MLD debug member tree structure information switch.
Qtech# **debug ipv6 mld warnning**	Turn on MLD debug warning switch.

5 CONFIGURING PIM-DM

5.1 PIM-DM Overview

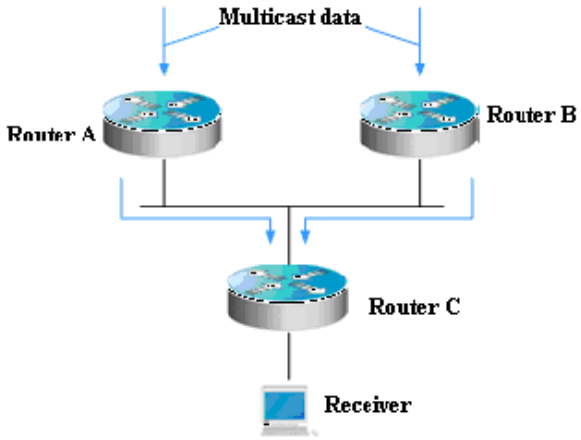
Protocol Independent Multicast-Dense Mode (PIM-DM) is a dense-mode multicast routing protocol, which is suitable for small-sized networks with densely distributed multicast members. As PIM-DM does not rely on any specific unicast routing protocol, it is called protocol independent multicast routing protocol. PIM-DM is defined in RFC 3973.

PIM-DM devices discover neighbors through Hello messages. After startup, a PIM-DM device sends a Hello message to each PIM-DM enabled interface periodically. The Hello message has a field of **Hello Hold Time**, which defines the maximum duration that a neighbor waits for the next message. If the neighbor does not receive another Hello message from the sender within this duration, this device will be removed from the adjacency list.

PIM-DM builds a shortest path tree (SPT) through flood and prune. PIM-DM assumes that when a multicast source begins to send a multicast packet, all the systems in the network need to receive this packet. As a result, this packet is forwarded to every system. The reverse path forwarding (RPF) check is performed for the packets received from the upstream interface. Those packets that fail to pass the check will be discarded. For the packets passing the check, the outgoing interface is computed based on the (S, G) pair of the packets, that is, source address and group address. If the outgoing interface is not null, an outgoing interface entry is created from the (S, G) pair and the multicast packet is forwarded through this outgoing interface. If the computed outgoing interface is null, a prune message is sent to the RPF neighbor, informing the upstream neighbor not to forward the multicast packets from the (S, G) pair to this interface. After the prune message is received on the upstream interface, the device marks the sending interface as pruned state and sets a pruned state timer. In this way, an SPT is created with the multicast source as its root.

PIM-DM uses the Assert mechanism to eliminate redundant routes.

Figure 11 Assert mechanism of PIM-DM



As shown in Figure-1, the multicast data arrives at Routers A and B at the same time, which forward the data to Router C. In this case, Router C receives duplicated data, which is not allowed. So there must be a mechanism to select Router A or B to forward the multicast data to Router C. This is the Assert mechanism of PIM-DM.

PIM-DM uses the state refresh message to update network state. The device directly connected to the multicast source sends the state refresh message to the downstream devices periodically to advertise the network topology changes. The devices receiving the message add their topology state to the state refresh message by modifying some fields, and then send it to the downstream devices. When the refresh message arrives at the leaf devices, the entire network state is updated.

PIM-DM uses the Graft mechanism to reestablish the connection with upstream devices. If the network topology of a downstream device in pruned state changes and the device needs to receive multicast data from a (S, G) pair, it sends the graft message to the upstream device. Upon receiving the graft message, the upstream device responds with a Graft-Ack message and forwards the multicast data to the downstream device again.

5.2 PIM-DM Configuration Tasks

The PIM-DM configuration tasks include the following items. However, only the first and second one are mandatory, and others are optional.

5.2.1 Enabling Multicast Routing

PIM-DM can forward multicast packets only after the multicast routing function is enabled.

Use the following command to enable or disable the multicast routing function in global configuration mode.

Command	Function
Qtech (config) # **ip multicast-routing**	Enables multicast routing globally.
Qtech (config) # **no ip multicast-routing**	Disables multicast routing globally.

5.2.2 Enabling PIM-DM

PIM-DM must be enabled on each interface. A device can exchange PIM-DM control messages with other devices, maintain and update the multicast routing table and forward multicast messages only after PIM-DM is enabled on the interface of the device.

Use the following commands to enable PIM-DM in interface configuration mode.

Command	Function
Qtech(config-if)# **ip pim dense-mode**	Enables the PIM-DM protocol on the interface.
Qtech(config-if)**# no ip pim dense-mode**	Disables the PIM-DM protocol on the interface.


Caution

Enabling PIM-DM will take effect on an interface only after the multicast routing is enabled in global configuration mode.
When this command is configured, if the system displays "Failed to enable PIM-DM on <*interface name*>, resource temporarily unavailable, please try again", configure this command again.
When this command is configured, if the system displays "PIM-DM Configure failed! VIF limit exceeded in NSM!!!", the number of configured multicast interfaces reaches the threshold. If you still need to enable PIM-DM on the interface, remove some unnecessary PIM-DM, PIM-SM or DVMRP interfaces.
It is not recommended that different IPv4 multicast routing protocols be configured on different interfaces of a switch or router.
If the interface is of tunnel-type, only 4Over4 configuration tunnel, 4Over4 GRE tunnel, 4Over6 configuration tunnel and 4Over6 GRE tunnel support the IPv4 multicasting. The multicasting function can also be enabled on other tunnel interfaces that do not support IPv4 multicasting, but no error message will be displayed and no multicast packets will be received or sent.
Multicast tunnels can be created on Ethernet interfaces only. Nested tunnel and multicast data QoS/ACL are not supported.

5.2.3 Setting the Interval of Sending the Hello Message

After PIM-DM is enabled on an interface, the interface sends the Hello message to the interfaces of adjacent devices periodically. You can modify the interval according to the network situation.

Use the following command to configure the interval of sending the Hello message in interface configuration mode.

Command	Function
Qtech(config-if)# **ip pimquery-interval** *interval-seconds*	Sets the interval of sending the Hello message on the interface. *interval-seconds*: in the range of 1 to 65535 seconds
Qtech(config-if)# **no ip pimquery-interval**	Restores the setting to the default value.

By default, the interval of sending the Hello message on the interface is 30 seconds.


Caution

When the interval of sending the Hello message is updated, Hello hold time will be updated as 3.5 times of the Hello sending interval automatically. If the interval of sending Hello message multiplying 3.5 is larger than 65535, Hello hold time is updated to 65535.

5.2.4 Configuring Propagation Delay of Hello Message

Options can be added to Hello messages. The default value of **propagation-delay** in **LAN Prune Delay Option** is 500 milliseconds.

Use the following command to configure the propagation delay of Hello messages in interface configuration mode.

Command	Function
Qtech(config-if)# **ip pim propagation-delay** *interval-milliseconds*	Sets the propagation delay in the range of 1 to 32767 milliseconds.
Qtech(config-if)# **no ip pim propagation-delay**	Restores the setting to the default value.

5.2.5 Configuring Override Interval of Hello Message

Options can be added to Hello messages. The default value of **override-interval** in **LAN Prune Delay Option** is 2500 milliseconds.

Use the following command to configure the override interval in interface configuration mode.

Command	Function
Qtech(config-if)# **ip pim override-interval** *interval-milliseconds*	Sets the override interval in the range of 1 to 65535 milliseconds.
Qtech(config-if)# **no ip pim override-interval**	Restores the setting to the default value.

5.2.6 Configuring PIM-DM Neighbor Filtering

The neighbor filtering function can be configured on the interface to enhance network security. With neighbor filtering enabled, PIM-DM does not establish adjacency with the neighbor or deletes the established adjacency with the neighbor as long as a neighbor is denied by the access list.

Use the following command to configure the PIM neighbor filtering function in interface configuration mode.

Command	Function
Qtech(config-if)#**ip pim neighbor-filter** *access-list*	Enables the PIM neighbor filtering function on the current interface.
Qtech(config-if)#**no ip pim neighbor-filter** *access-list*	Disables the PIM neighbor filtering function on the current interface.

The PIM neighbor filtering function is disabled by default on an interface.


Note

ip pim neighbor-filter command description:
Only neighbor addresses permitted by the ACL can be the PIM neighbors of the current interface.

5.2.7 Configuring PIM-DM State Refresh

After PIM-DM is enabled on a device, if the RPF interface is directly connects to the multicast source (that is, the PIM interface is in the same network segment as the multicast source), the device periodically sends state refresh messages to downstream devices to update the entire network state. You can disable processing or forwarding of PIM state refresh messages in global configuration mode.

Use the following command to configure PIM-DM state refresh in global configuration mode.

Command	Function
Qtech(config-if)#**no ip pim state-refresh disable**	Enables processing or forwarding PIM-DM state refresh messages.
Qtech(config-if)# **ip pim state-refresh disable**	Disables processing or forwarding PIM-DM state refresh message.

The PIM-DM state refresh function is enabled by default.


Caution

Disabling the state refresh messages may cause the re-convergence of the converged PIM-DM multicast forward tree, resulting in unnecessary bandwidth waste and routing table flapping. Therefore, it is better not to disable the state refresh function in normal cases.

5.2.8 Configuring the Interval of Sending PIM-DM State Refresh Message

After PIM-DM is enabled on a device, if an interface is directly connected to the multicast source, the device periodically sends state refresh messages to downstream devices to update the entire network state. You can modify the interval of sending PIM state refresh message on an interface according to the network situation.

Use the following command to configure the interval of sending PIM state messages on the interface in interface configuration mode.

Command	Function
Qtech(config-if)**#ip pim state-refresh origination-interval** *seconds*	Configures the interval of sending PIM state refresh messages on the current interface, in the range of 1 to 100 seconds.
Qtech(config-if)**#no ip pim state-refresh origination-interval**	Restores the setting to the default value.

By default, the interval of sending PIM state refresh messages on the interface is 60 seconds.


Note

Only the devices directly connected to multicast source can periodically send the PIM state refresh message to the downstream interfaces. Therefore, if the devices are not directly connected to the multicast source, the interval of sending PIM state refresh messages configured on the downstream interface is invalid.

5.3 Monitoring and Maintaining PIM-DM

PIM-DM provides the following commands to monitor and maintain PIM-DM.

5.3.1 Displaying PIM-DM State

Command	Function
show ip pim dense-mode interface [*interface-type interface-number*] [**detail**]	Shows the PIM-DM information on the interface.
show ip pim dense-mode neighbor [*interface-type interface-number*]	Shows the PIM-DM neighbor information.
show ip pim dense-mode nexthop	Shows the next hop information of PIM-DM.
show ip pim dense-mode mroute [*group-or-source-address [group-or-source-address*]] [**summary**]	Shows the PIM-DM routing table.
show ip pim dense-mode track	Shows the number of PIM packets sent and received since the statistic beginning time.

For details about the preceding commands, see *PIM-DM Command References*.

Here are some examples of the commands:
show ip pim dense-mode interface detail command:

```
Qtech# show ip pim dense-mode interface detail
FastEthernet 0/1 (vif-id: 3):
Address 10.10.10.10
Hello period 30 seconds, Next Hello in 15 seconds
Over-ride interval 2500 milli-seconds
Propagation-delay 500 milli-seconds
Neighbors:
```

```
10.10.10.1
FastEthernet 0/2 (vif-id: 2):
Address 50.50.50.50
Hello period 30 seconds, Next Hello in 2 seconds
Over-ride interval 2500 milli-seconds
Propagation-delay 500 milli-seconds
Neighbors:
50.50.50.1
```

In the preceding example, the IP address of FastEthernet 0/1 is 10.10.10.10, the Hello message sending interval is 30 seconds, the next Hello message is to be sent in 15 seconds, and the neighbor address is 10.10.10.1. The information about FastEthernet 0/2 is similar to that of FastEthernet 0/1.

show ip pim dense-mode neighbor command:

```
Qtech# show ip pim dense-mode neighbor
Neighbor-Address Interface              Uptime/Expires     Ver
10.10.10.1       FastEthernet 0/1       00:19:29/00:01:21  v2
50.50.50.1       FastEthernet 0/2       00:22:09/00:01:39  v2
```

In the preceding example, the device has two neighbors. Neighbor 10.10.10.1 connects to FastEthernet 0/1 and has been alive for 19 minutes and 29 seconds, with the TTL to expire in one minute and 21 seconds. Neighbor 50.50.50.1 is similar.

show ip pim dense-mode nexthop command:

```
Qtech# show ip pim dense-mode nexthop
Destination  Nexthop   Nexthop       Nexthop              Metric Pref
                 Num       Addr          Interface
1.1.1.111        1        50.50.50.1    FastEthernet 0/2    0      1
```

In the preceding example, the next hop neighbor address to multicast source 1.1.1.111 is 50.50.50.1 and the egress is FastEthernet 0/2.

show ip pim dense-mode mroute command:

```
Qtech# show ip pim dense-mode mroute
PIM-DM Multicast Routing Table
(1.1.1.111, 229.1.1.1)
MRT lifetime expires in 205 seconds
RPF Neighbor: 50.50.50.1, Nexthop: 50.50.50.1, FastEthernet 0/2
Upstream IF: FastEthernet 0/2
Upstream State: Pruned, PLT:200
Assert State: NoInfo
Downstream IF List:
FastEthernet 0/1:
Downstream State: NoInfo
Assert State: Loser, AT:170
```

The preceding example shows two entries: 1.1.1.111 and 229.1.1.1. The MRG aging time is 205 seconds, RPF neighbor is 50.50.50.1, the next hop is 50.50.50.1, and the egress to the next hop is FastEthernet 0/2. The upstream interface of these entries is FastEthernet 0/2 in Pruned state, indicating that there is no downstream forwarding egress. The downstream interface is FastEthernet 0/1 in NoInfo state. The Assert state of the interface is Loser. FastEthernet 0/1 is not a forwarding egress.

show ip pim dense-mode track command:

```
Qtech# show ip pim dense-mode track
PIM packet counters
Elapsed time since counters cleared: 00:04:03

                                received     sent
Valid PIMDM packets:                   1             8
Hello:                                 1             8
Join/Prune:                      0             0
Graft:                                        0                    0
Graft-Ack:                                    0                    0
Assert:                                       0                    0
State-Refresh:                         0                    0
PIM-SM-Register:                 0                    0
PIM-SM-Register-Stop:            0                    0
PIM-SM-BSM:                            0                    0
PIM-SM-C-RP-ADV:                       0                    0
```

```
Unknown Type:                          0

Errors:
Malformed packets:                 0
Bad checksums:                         0
Unknown PIM version:                   0
Send errors:                           0
```

5.3.2 Deleting PIM-DM State Information

Use the following command to delete the PIM-DM state information:

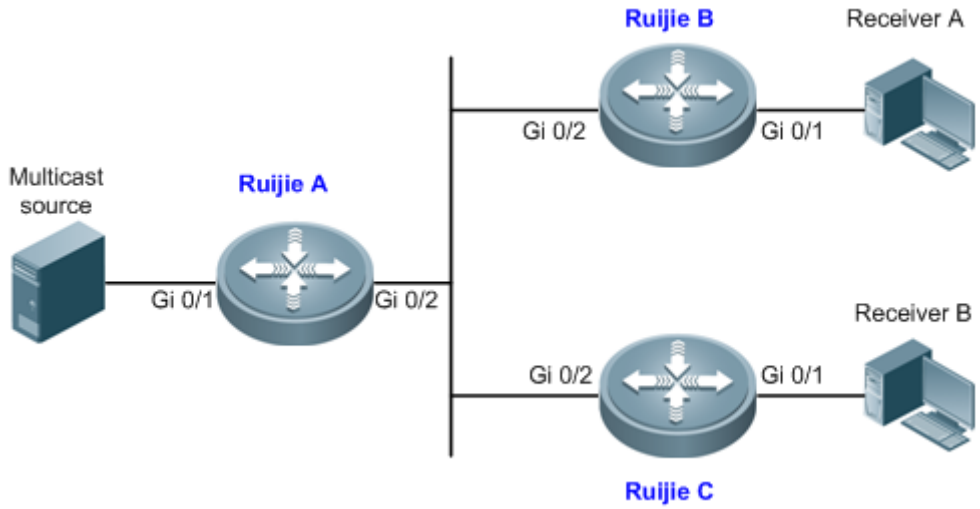
Command	Function
Qtech# **clear ip pim dense-mode track**	Resets the start time of packet statistics and clears PIM packet counter.

5.4 PIM-DM Configuration Example

5.4.1 Configuration Requirements

Figure 12 shows the network topology. Qtech A and the multicast source are in the same network, Qtech B and receiver A are in the same network, and Qtech C and receiver B are in the same network. Assume that the devices are properly connected to the hosts, and the IP addresses and unicast routes are configured.

Figure 12 Networking topology for PIM-DM configuration



5.4.2 Device Configuration

The following example shows how to configure PIM-DM on Qtech A. The configurations on Qtech B and Qtech C are similar to those on Qtech A.

- Step 1: Enable multicast routing.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
```

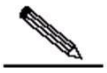
- Step 2: Enable PIM-DM on the interface Gi 0/1.

```
Qtech(config)# interface GigabitEthernet 0/1
Qtech(config-if)# ip pim dense-mode
Qtech(config-if)# exit
```

- Step 3: Enable PIM-DM on the interface Gi 0/2 and return to the privileged EXEC mode.

```
Qtech(config)# interface GigabitEthernet 0/2
Qtech(config-if)# ip pim dense-mode
Qtech(config-if)# end
```

The configuration on Qtech B and Qtech C is similar to Qtech A, that is, enable multicast routing and enable PIM-DM on each interface.


Note

Enabling PIM-DM will automatically enable IGMP on each interface.
This example applies to both layer-3 switches and routers. However, if you need to configure the switch port as a layer-3 port, you must run the **no switchport** command (not needed for routers).

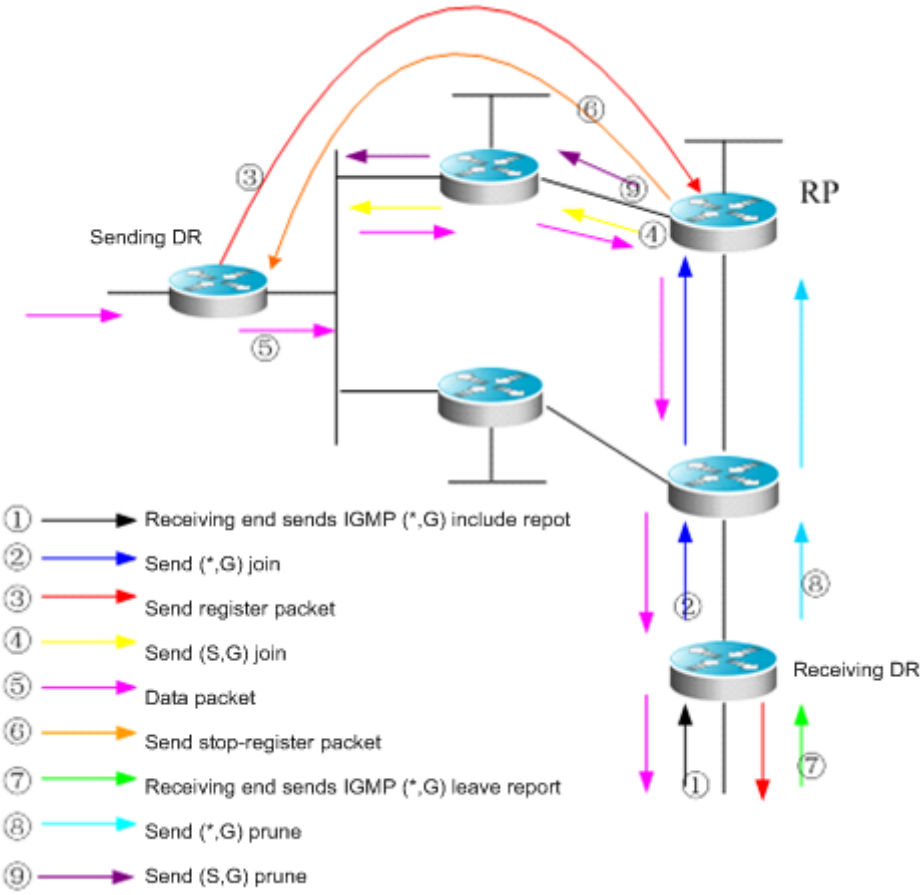
6 CONFIGURING PIM-SM

6.1 PIM-SM Overview

The Protocol Independent Multicast (PIM) is designed by the Inter-Domain Multicast Routing (idmr) working group. As its name implied, PIM does not rely on any specific unicast routing protocol. It can use a unicast routing table established by any unicast routing protocol to perform the RPF check function, instead of maintaining separate multicast routing tables to implement multicast forwarding. As PIM is not required to receive or distribute route updates, compared to other multicast routing protocols, it costs much less. PIM is designed to support shortest path trees (SPTs) and rendezvous point trees (RPTs) simultanenously and enable flexible conversion between them, so that their advantages can be used to improve multicast efficiency. There are two PIM modes: dense mode and sparse mode.

The Protocol Independent Multicast – Sparse Mode (PIM-SM) is a multicast routing protocol of sparse mode. In a PIM-SM domain, the PIM-SM-enabled device periodically sends Hello messages to discover adjacent PIM-SM devices and selects a designated router (DR) in a multi-access network. The DR is responsible for sending Join/Prune messages towards the root of themulticast distribution tree from its directly connected group member, or itsdirectly connected multicast source.

Figure 13 Explicit Join/Prune mechanism of PIM-SM



PIM-SM forwards multicast data packets by establishing a multicast distribution tree. The multicast distribution tree is divided into two types: Shared Tree that takes the RP of the group G as the root and Shortest Path Tree that takes the multicast source as the root. PIM-SM establishes and maintains the multicast distribution tree by use of the explicit join/prune mechanism. As shown in Figure-1,
The DR at the receiving end receives an IGMP (*,G)include report packet from the receiving end.

If the DR at the receiving end is not the RP of this group G, it will send a (*.G)join packet towards the RP. The upstream router receiving this (*.G)join packet will send it towards the RP. In this way, the (*.G)join packet is sent hop by hop until the RP of the group G receives the (*.G)join packet. It is indicated that the DR has joined the shared tree.
When the source host sends multicast data to the group, the source data is encapsulated into a register message and unicasted by the DR at the data source to the RP. Then the RP will forward the decapsulated data packets to group members along the shared tree.
The RP will send a (S, G)join packet to the DR in the direction of the source to join the shortest path tree of this source.
In this way, the source's packets are sent to the RP without encapsulation along its shortest path tree after the SPT from the RP to the DR at the source is established.
When the first multicast data reaches along the SPT, the RP will send the register - stop message to the DR at the source, notifying the DR of stopping register encapsulation. When the DR at the source received the register - stop message, it will not encapsulate register packets, but send them to the RP along its shortest path tree, which will forward them to group members along the shared tree.
When a receiving end needs no multicast data, it will send an IGMP leave message.
The DR at the receiving end multicasts the prune message to the group G's RP hop by hop to prune the shared tree. This prune message will finally arrive at the RP or a router with other (*G) receivers on the way to the RP. Therefore, the data packets will not be sent toward that receiving end.
If there is no downstream receiver on the RP, the RP will send the (S,G) prune packet toward the data source. As the (S, G) prune packets are sent to the DR at the source end one by one, the DR at the source end will prune the interface receiving the (S,G) prune packet. As a result, the data packets are filtered at the DR at the source end.

PIM-SM also offers a mechanism of selecting the root point (RP). One or more Candidate-BSRs are configured in a PIM-SM domain. PIM-SM selects a BSR by following a certain rule. There are also Candidate-RPs in a PIM-SM domain that unicast the packets including their IP addresses and available multicast groups to the BSR. The BSR will periodically generate a BSR message which includes a series ofcandidate RPs and corresponding multicast group addresses. The BSR messages are sent hop-by-hop within the entire domain. The device receives and saves these BSR messages. If the DR receives a report on the member relationship of a multicast group from its directly connected host but has no route entries of the multicast group, the DR will use a Hash algorithm to map the multicast group address to a candidate RP that can serve this group. Then, the DR multicasts the Join/Prune message to the RP hop-by-hop. If the DR receives multicast data packets from its directly connected host but has no route entries of the multicast group, the DR will usea Hash algorithm to map the multicast group address to a candidate RP that can serve this group. Then the DR encapsulates multicast data packets into a register message and unicasts it to the RP.

The main difference between PIM-SM and the flood/prune model-based PIM-DM is that PIM-SM is based on the explicit join model. In other words, the receiver sends the join message to the RP, while the router only forwards the packets of that multicast group on the outgoing interface that has joined a multicast group. PIM-SM uses the shared tree to forward multicast packets. Each group has a Rendezvous Point (RP). The multicast source sends data to the RP along the shortest path, and then the RP sends the data to the receivers along the shortest path. This is similar to CBT, but PIM-SM does not use the concept of core. One of the major advantages of PIM-SM is that it not only receives multicast messages through the shared tree but also provides a shared tree-to-SPT conversion mechanism. Such conversion reduces network delay and possible congestion on the RP, but it consumes enormous router resources. So it is suitable for the case where there are only a few multicast data sources and network groups.

PIM-SM uses the shared tree and SPT to distribute multicast frames. At this time, it is assumed that other devices don't want to receive these multicasts unless otherwise stated definitely. When a host joins a group, the equipment connected to the host must notify the root (or the RP) by using the PIM join message. This join message is transferred one after another through the routers to create a shared tree structure. Therefore, the RP records the transfer path and also the register message from the first hop router (DR) of the multicast source, and improves the shared tree upon these two messages. The branch/leaf messages are updated by periodically querying messages. With the shared tree, the multicast source first sends multicast packets to the RP, guaranteeing that all the receivers can receive them. The notation (*.G) represents a tree. The asterisk (*) represents all sources and G represents a specific multicast address. The prune message is also used in the shared tree. That is, the branch/leaf will send prune messages once it is not expecting to receive multicast frames.

PIMv2 BSR is a method of distributing group-to-RP messages to all devices without the need of setting an RP for them. BSR distributes mapping information by propagating BSR messages hop by hop. At first, BSR is selected among routers in the same process as selecting a root bridge based on priority level among layer 2 bridges. Each BSR checks the BSR messages and only forwards those having a priority higher than or equal to its own (higher IP address). The selected BSR sends its BSR message to the all-PIM-routers multicast group (224.0.0.13), where TTL is 1. After the adjacent PIMv2 router receives the message, it multicasts it while setting the TTL to 1. In this way, the

BSR message is received by all devices hop by hop. Since the message contains the IP address of the BSR, the candidate BSR can know which router is the current BSR based on this message. The candidate RPs send candidate RP advertisements to announce in which address ranges they can become an RP. The BSR stores them in its local candidate RP cache. The BSP notifies all PIM routers of its local candidate RPs periodically. These messages reach various devices hop by hop in the same way.

☑ The VRF parameters only apply to RSR20, RSR30, RSR50 and RSR50E

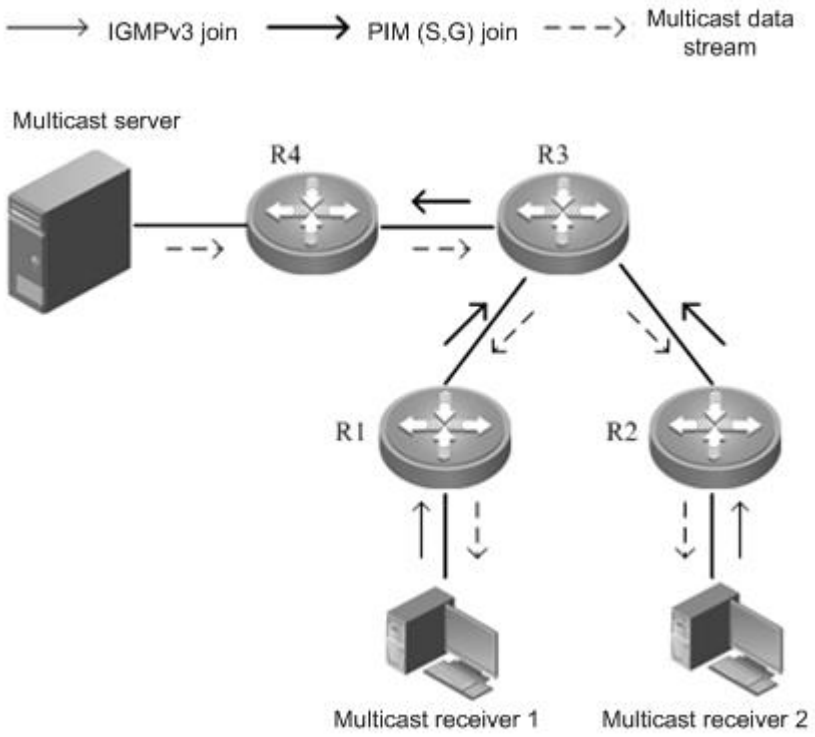
6.1.1 SSM Model

PIM-SM allows two multicast models,: Any-Source Multicast (ASM) and Source-Specific Multicast (SSM). In the ASM model, multicast receivers only specify a multicast group G to join but not a multicast source S. In the SSM model, multicast receivers can specify both a multicast source S and multicast group G.

The PIM-based SSM model provides implementation solutions for specified source multicast. It requires IGMPv3 to manage the membership between hosts and routers and PIM-SM to connect routers.

In the SSM model, multicast receivers have known the multicast source information (S, G) by some means such as accessing the server and accepting advertisements. Then, when a multicast receiver requests a multicast service, it can send IGMP(S, G) join directly to the last hop router. As shown in Figure 2, multicast receiver 1 sends the IGMP(S, G) join report to request the multicast service (S, G). The last hop router sends PIM (S, G) join to the multicast source hop by hop after receiving the IGMP (S, G) join from the multicast receiver. As shown in Figure 2, R1 sends the PIM(S, G) join to R3 after receiving the IGMP(S, G) join report from multicast receiver 1, and then R3 sends the PIM(S, G) join to R4. In this way, a shortest path tree from the multicast receiver to the multicast source is created.

Figure 14 SSM Model



Implementation of an SSM model requires that,

- The multicast receiver obtains the information about the multicast source (S, G) in advance through some channel; the multicast receiver initiates IGMP(S, G) join for the desired multicast services.
- IGMPv3 must be enabled on the interface of the last hop router connected to the multicast receiver. IGMPv1/IGMPv2 does not support the SSM.
- It is recommended that PIM-SSM be enabled on the routers on the way from the multicast receiver to the multicast source. AS PIM-SSM is compatible with PIM-SM, it is feasible to enable only PIM-SM.

Note

With SSM enabled, the default group range of SSM is 232/8. The group range of SSM can be modified through commands or SSM can be disabled. For details, see the "Configuring SSM" section.

The SSM has the following features:

- In the SSM model, multicast receivers can obtain the information about the multicast source in advance through some channels, for example, advertisements or access to the specified server.
- The SSM model is a specific subset of PIM-SM. It only processes the PIM(S,G) join and PIM(S,G) prune messages and drops the RPT-related messages within the range of SSM, e.g. PIM(*,G) join/prune messages. It will respond immediately with the register - stop packet to the register packets within the range of SSM.
- In the SSM model, no PR or the election and distribution of RP messages is required. In the SSM, all the multicast distribution trees created are the shortest path trees (SPT).

6.2 Preparation before Configuring PIM-SM

Before configuring PIM-SM, enable a routing protocol such as OSPF to automatically discover routes.

6.3 PIM-SM Configuration Tasks

The PIM-SM configuration tasks cover the following items. However, only the first and second one are mandatory, and others are optional.

6.3.1 Enabling Multicast Routing

PIM-SM can forward multicast packets only after the multicast routing function is enabled.

Use the following command to enable the multicast routing function in global configuration mode.

Command	Function
ip multicast-routing [**vrf** *vrf-name*]	Enables multicast routing. If VRF is carried, multicast routing is enabled based on the VRF; if VRF is not carried, multicast routing is enabled globally.
no ip multicast-routing [**vrf** *vrf-name*]	Disables multicast routing.

6.3.2 Enabling PIM-SM

PIM-SM must be enabled on each interface. A device can exchange PIM-SM control messages with other devices, maintain and update multicast routing table, and forward multicast packets only after PIM-SM is enabled on its interface.

Use the following command to enable PIM-SM on the interface in interface configuration mode.

Command	Function
ip pim sparse-mode	Enables PIM-SM on the interface.
no ip pim sparse-mode	Disables PIM-SM on the interface.

Note PIM-SM can be enabled on an interface only after multicast routing is enabled in global configuration mode.

Note If the system prompts "Failed to enable PIM-SM on <*interface name*>, resource temporarily unavailable, please try again", run this command again.

Note If the system prompts "PIM-SM Configure failed! VIF limit exceeded in NSM!!!", the number of configured interfaces exceeds the upper limit of the multicast interfaces. Remove some unnecessary PIM-DM, PIM-SM or DVMRP interfaces.

Note It is not recommended that different IPv4 multicast protocols be configured on different interfaces of a switch or router.

Note If the interface is of tunnel-type, only 4Over4 configuration tunnel, 4Over4 GRE tunnel, 4Over6 configuration tunnel and 4Over6 GRE tunnel support the IPv4 multicasting. Multicast can be also enabled on other tunnel interfaces that do not support the multicasting, but no error message will be displayed and no multicast packets will be received or sent.

Note The multicast tunnel can be created on the Ethernet interface only. Nested tunnel and multicast data QoS/ACL are not supported.

6.3.3 Configuring the Interval of Sending Hello Messages

After PIM-SM is enabled on the interface, the device periodically sends Hello messages to the interfaces of neighbors. You can set the interval of sending Hello messages according to the actual network situation.

Use the following command to configure the interval of sending the Hello message in interface configuration mode.

Command	Function
ip pim query-interval *interval-seconds*	Sets the interval of sending Hello messages, in seconds. *interval-seconds*: in the range of 1 to 65535 seconds
no ip pim query-interval	Restores the setting to the default value.

By default, the interval of sending Hello messages on the interface is 30 seconds.

Note

When the interval of sending Hello messages is updated, Hello hold time will automatically be updated to 3.5 times of the interval of sending Hello messages. If the result is greater than 65535, Hello hold time is updated to 65535.

6.3.4 Configuring Propagation-Delay in Hello Message Option

Options can be added to Hello messages. The default value of propagation-delay in LAN Prune Delay Option is 500 milliseconds.

Use the following command to configure the propagation delay of Hello messages in interface configuration mode.

Command	Function
ip pim propagation-delay *interval-milliseconds*	Sets the propagation delay in the range of 1 to 32767 milliseconds.
no ip pim propagation-delay	Restores the setting to the default value.

Note

Modifying propagation delay or prune deny delay will affect J/P-override-interval. As specified in the protocol, J/P-override-interval is less than the hold time of Join-Prune message; otherwise streams may be interrupted temporarily. This can be ensured by network administrators.

6.3.5 Configuring Override-Interval in Hello Message Option

Options can be added to Hello messages. The default value of override-interval in LAN Prune Delay Option is 2500 milliseconds.

Use the following command to configure the override interval in interface configuration mode.

Command	Function
ip pim override-interval *interval-milliseconds*	Sets the override interval in the range of 1 to 65535 milliseconds.
no ip pim override-interval	Restores the setting to the default value, namely, 2500 milliseconds.

Note

Modifying propagation delay or prune deny delay will affect J/P-override-interval. As specified in the protocol, J/P-override-interval must be less than the hold time of Join-Prune message; otherwise streams may be interrupted temporarily. This can be ensured by network administrators.

6.3.6 Configuring Neighbor-Tracking in Hello Message Option

The T bit of the LAN Prune Delay Option of the Hello message indicates whether to enable join restriction on the interface. When join restriction is enabled on the interface, the Join message to be sent from the interface to the upstream neighbor will be restricted, upon receipt of the Join message from its neighbor to the upstream neighbor. If this function is disabled, the Join message to be sent from the interface to the upstream neighbor will still be sent. Moreover, if join restriction is enabled on all downstream receivers, the upstream router can trace these receivers by received Join messages. By default, join restriction is enabled on the interface.

Use the following command to disable join restriction on the interface in interface mode.

Command	Function
ip pim neighbor-tracking	Disables join restriction on the interface.
no ip pim neighbor-tracking	Enables join restriction on the interface.

6.3.7 Configuring Triggered Hello Delay of Hello Messages

When a router starts or detects new neighbor, the device will send Hello messages after a random period of time to prevent congestion of Hello packets. This random interval can be calculated based on triggered hello delay, which is 5 seconds by default.

Use the following command to configure the triggered Hello delay in interface configuration mode.

Command	Function
ip pim triggered-hello-delay *interval-seconds*	Sets the triggered hello delay, in seconds. *interval-seconds*: in the range of 1 to 5 seconds
no ip pim triggered-hello-delay	Restores the setting to the default value.

6.3.8 Configuring PIM-SM Neighbor Filtering

You can filter neighbors on an interface to enhance network security. With this function enabled, when a neighbor is denied by an ACL, the PIM-SM will not establish the adjacency relationship with that neighbor or remove the currently established adjacency relationship with that neighbor.

Use the following command to configure PIM-SM neighbor filtering in interface configuration mode:

Command	Function
ip pim neighbor-filter *access-list*	Enables the PIM-SM neighbor filtering function on the interface.
no ip pim neighbor-filter *access-list*	Disables the PIM-SM neighbor filtering function on the interface.

By default, the PIM-SM neighbor filtering function is disabled on an interface.

Note

ip pim neighbor-filter command description:
When the associated ACL rule is permit, only the neighbor address in the ACL list can be used as the PIM neighbor of the current interface; when the associated ACL rule is deny, the neighbor address in the ACL list cannot be used as the PIM neighbor of the current interface.

6.3.9 Configuring the Priority of DR

This command is used to configure the priority of the designated router (DR). Higher weight means higher priority.

Use the following command to configure the DR priority in interface configuration mode.

Command	Function
ip pim dr-priority *priority-value*	Sets the DR priority. *priority-value*: in the range of 0 to 4294967294
no ip pim dr-priority	Restores the setting to the default value, namely 1.

6.3.10 Configuring Static RP

In a small network, you can configure static RP to use PIM-SM. All the devices in the PIM-SM domain have the same static RP configuration, ensuring no ambiguity of the PIM-SM multicast routes.

Use the following command to configure static RP in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **rp-address** *rp-address* [*access-list*]	Configures the static RP address. *access-list*: supports the numerical ACL in the range of 1 to 99 and 1300 to 1999; also supports the named ACL. All multicast groups are permitted by default.
no ip pim [**vrf** *vrf-name*] **rp-address** *rp-address* [*access-list*]	Removes the static RP configuration.

Note

If the static RP and the dynamic RP take effect at the same time, the dynamic RP takes precedence. .

Note

The static RP address can be configured for multiple multicast groups (by ACL) or all multicast groups (not by ACL). However, a static RP address cannot be configured for several times.

Note

If more than one static RP are configured for a multicast group, the one with the highest IP address takes effect.

Note

Only the permitted addresses defined in the ACL are invalid multicast groups. By default, 0.0.0.0/0 refers to filtering all multicast groups (224/4).

Note

After configuration, the static RP source address is inserted into the tree of group-based static RP group. Each static multicast group maintains the link table structure of a static RP group. The link tables are ordered in descending sequence by IP addresses. When a RP is selected for a group, the first element, namely, the RP with the highest IP address is firstly selected.

Note

Deleting a static RP address deletes the address from all groups that has this address, and one address is selected from the existing tree structure as the RP address.

6.3.11 Configuring Candidate BSR

A globally unique BSR is elected from candidate BSRs configured on an interface in a PIM-SM domain. This BSR will collect and distribute RPs in the domain to ensure the uniqueness of RP mapping in the domain.

Use the following command to configure the candidate BSR in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **bsr-candidate** *interface-type interface-number* [*hash-mask-length* [*priority-value*]]	Specifies an interface as a candidate BSR. The global BSR is elected through BSM message learning and competition. *hash-mask-length* ranges from 0 to 32, 10 by default. *priority-value* ranges from 0 to 255, 64 by default.
no ip pim [**vrf** *vrf-name*] **bsr-candidate** *interface-type interface-number*	Removes the configuration of a candidate BSR.

6.3.12 Configuring BSR Border

To restrict BSM flooding, you can set the BSR border on the interface so that BSM will be dropped immediately rather than being forwarded.

Use the following command to configure the BSR border in interface configuration mode.

Command	Function
ip pim bsr-border	Configures the BSR border on an interface.
no ip pim bsr-border	Removes the BSR border on an interface.

6.3.13 Ignoring RP Priorities in RP-SET

When an RP is selected for a multicast address, if several RPs can serve this multicast address, you can use this command to ignore the RP priority when comparing two RPs. If this command is not configured, the RP priority will be taken into account during comparison.

Command	Function
ip pim [**vrf** *vrf-name*] **ignore-rp-set-priority**	Ignores the RP priority in RP-Set.
no ip pim [**vrf** *vrf-name*] **ignore-rp-set-priority**	Considers the RP priority in RP-Set.

6.3.14 Configuring Candidate RP

Candidate RP advertisement is sent to the BSR at intervals and then propagated to all the PIM-SM devices in the domain, thus ensuring the uniqueness of RP mapping.

Use the following command to configure the candidate RP in global configuration mode.

Command	Function
ip pim rp-candidate *interface-type interface-number* [**priority** *priority-value*] [**interval** *interval-seconds*] [**group-list** *access-list*]	Configures the device as the candidate RP. When **priority** is default, *priority-value* ranges from 0 to 255, 192 by default. When **interval** is default, *interval-seconds* ranges from 1 to 16383, 60 seconds by default. When **group-list** is default, *access-list* permits all multicast groups by default, namely 224/4.
no ip pim rp-candidate *interface-type interface-number*	Removes the candidate RP configuration.


Note

You can use the ACL to specify an interface as the candidate RP of a specific group. It should be noted that the group calculation is based on the permit ACE, not the deny ACE. The source range of the ACE is matched as a specific group range.

6.3.15 Checking Reachability of Register Messages

You can use this command to check whether an RP is reachable. With this command configured, the DR checks whether RP is reachable before sending a register packet, that is whether there is a route to the RP by checking the unicast and static multicast routing tables. If there is no such a route, no register packet will be sent.

Use the following command to check the RP reachability in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **register-rp-reachability**	Checks whether a register packet can reach the destination device. No check is conducted by default.
no ip pim [**vrf** *vrf-name*] **register-rp-reachability**	Disables this function.


Note

If there is a static multicast route to the RP and the next hop of the route is reachable in the unicast routing table, PIM-SM considers that a route to the RP exists even if the RP is not reachable in the unicast routing table.

6.3.16 Configuring Address-based Filtering for Register Packets

You can use this command to filter the register packets that have arrived at an RP by the source address and group address contained in the packets. Otherwise, every reached register packet is permitted. With this command configured, only the register packets with the source addresses and group addresses permitted by the ACL can be processed.

Use the following command to configure address-based filtering for register packets in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **accept-register list** *access-list*	Enables filter of register packets by source addresses and group addresses. *access-list* ranges from 100 to 199 and 2000 to 2699; also supports named ACL.
no ip pim [**vrf** *vrf-name*] **accept-register**	Disables filter of register packets by source addresses and group addresses.

6.3.17 Configuring Rate Limit on Sending Register Packets

Use this command to configure the rate of sending register packets by DR. Use the **no** form of this command to cancel the rate limit. This command configures the rate of sending register packets for each (S, G) state, not the register packets in the whole system.

Use the following command to configure the rate limit on sending RPs in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **register-rate-limit** *rate*	Sets the maximum number of register packets sent per second. *rate*: in the range of 1-65535
no ip pim [**vrf** *vrf-name*] **register-rate-limit**	Removes the rate limit.

6.3.18 Configuring the Whole-Packet Method for Calculating the Register Packet Checksum

Use this command to calculate the whole PIM packet including the multicast data packet encapsulated when calculating the register packet checksum. Otherwise, the checksum of register packets is calculated using the default method specified by the protocol.

Use the following command to configure the whole-packet method for calculating the register packet checksum in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **register-checksum-wholepkt** [**group-list** *access-list*]	Configures the whole-packet method for calculating the register packet checksum. group-list *access-list*: Apply this configuration to all multicast addresses by default.
no ip pim [**vrf** *vrf-name*] **register-checksum-wholepkt** [**group-list** *access-list*]	Removes the whole-packet method for calculating the register packet checksum. group-list *access-list*: By default, this configuration is removed for all the multicast addresses.

Note

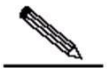
Some devices from other vendors make checksum calculation of register packets based on the overall packets. This function is introduced in Qtech's devices to be compatible with those devices.
If a device from those vendors serves as the RP and Qtech's device serves as the source DR, you can use this command on the source DR; if the device from other vendors serves as the source DR and Qtech's device serves as the RP, you can use this command on the RP.

6.3.19 Configuring the RP to Forward Multicast Packets to Downstream Interfaces after Decapsulating Register Packets

Use this command to decapsulate a register packet and forward its multicast packets. Without this command, the multicast packets in the register packet are not de-capsulated or forwarded.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **register-decapsulate-forward**	Decapsulates the register packet and forwards its multicast packets
no ip pim [**vrf** *vrf-name*] **register-decapsulate-forward**	Removes the configuration.



Note

Since the register packet is decapsulated and its multicast packets are forwarded through software, in case of decapsulation and forwarding of many register packets, this function incurs additional workload to CPU. So it is not recommended.

6.3.20 Limiting the Range of Legal BSRs

Use this command to limit the range of legal BSRs. Without this function, PIM-SM-enabled routers will receive all external BSM messages.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **accept-bsr list** *accept-bsr list*	Filters the BSM packet of BSR. The range is from 1 to 99, 1300 to 1999, or can be characters.
no ip pim [**vrf** *vrf-name*] **accept-bsr**	Removes the filtering of the BSM packet of BSR.


Note

This command filters the BSR address field of the BSM message. If this address is denied by ACL, the BSM message is filtered.

6.3.21 Configuring the Electing BSR to Limit the Legal CRP Address Range and the Multicast Group Range It Serves

Use this command to configure the electing BSR to limit the legal CRP address range and the multicast group range it serves. Without this function, the electing BSR will receive all external advertisement messages of candidate RPs.

In this command, the *source* parameter of ACL rule specifies the C-RP address and the *destination* parameter specifies the multicast group range the C-RP serves. If both addresses are denied by ACL, the group of the C-RP will be filtered.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **accept-crp list** *accpet-crp list*	Enables the electing BSR to filter the candidate RP advertisement. The range is from 100 to 199, 2000 to 2699 or can be characters.
no ip pim [**vrf** *vrf-name*] **accept-crp**	Disables the electing BSR to filter the candidate RP advertisement.

6.3.22 Configuring the Electing BSR to Receive the C-RP-ADV Message Whose Prefix-count Is 0

Use this command to configure the electing BSR to receive the C-RP-ADV message whose prefix-count is 0. Without this command, the electing BSR will not process the C-RP-ADV packet whose prefix-count is 0.

With this function, the electing BSR considers that the C-RP supports all groups after receiving the C-RP-ADV message whose prefix-count is 0.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **accept-crp-with-null-group**	Configures the electing BSR to receive the C-RP-ADV message whose prefix-count is 0.
no ip pim [**vrf** *vrf-name*] **accept-crp-with-null-group**	Removes the configuration.

6.3.23 Configuring the Source IP Address of Register Packets

This command sets the source IP address of register packets sent from DR. With this command not configured or the **no** form of this command, the DR interface address connected to the multicast source is used as the source address of the register packet. If the address parameter of this command is used, the configured address must be reachable for unicast routes. If the interface parameter of this command is used, it is generally a loopback interface, but can also be other types. This interface address must have been advertised by the unicast route.

Use the following command in global configuration mode.

Command	Function
ip pim register-source { *local_address* | *Interface-type interface-number* }	Configures the source IP address used in RPs.
no ip pim register-source	Sets the RPF interface address as the source IP address of register packets.

6.3.24 Configuring Register Suppression Time

This command configures the registersuppression time. It will modify the register suppression time defined on the DR. If the **ip pim rp-register-kat command** is not configured, defining the register suppression time in the RP will change RP keepalive period.

Use the following command to configure the register suppression time in global configuration mode.

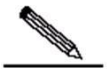
Command	Function
ip pim [**vrf** *vrf-name*] **register-suppression** *seconds*	Configures the register suppression time. *seconds* ranges from 1 to 65535 seconds.
no ip pim [**vrf** *vrf-name*] **register-suppression**	Sets the suppression time to 60 seconds.

6.3.25 Configuring the Probe Interval of Null Register Packet

The source DR can send Null-Register packet to the RP in a period of time before the registersuppression time expires. This period is called probe interval, 5 seconds by default.

Use the following command to configure the probe time of null register packets in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **probe-interval** *interval-seconds*	Configures the probe time of null register packets. *interval-seconds*: in the range of 1 to 65535 seconds
no ip pim [**vrf** *vrf-name*] **probe-interval**	Restores the probe time to 5s.



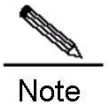
Note

The probe time should be less than half of register suppression time. Moreover, the register suppression time times three and plus the probe time should not be greater than 65535 seconds; otherwise, the system displays a warning message.

6.3.26 Configuring the RP KAT Timer

Use this command to configure the keepalive time of the (S, G) state created by register packets on the RP in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **rp-register-kat** *seconds*	Configures KAT timer. *seconds*: in the range of 1 to 65535 seconds
no ip pim [**vrf** *vrf-name*] **rp-register-kat**	Uses the default KAT value, namely, register suppression time times three and plus register probe time.


Note

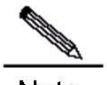
The value of the timer should be greater than register suppression time of source DR times three and plus register probe time. Otherwise, the RP may timeout the (S, G) state before the source DR sends the register packet again, causing temporary interruption of the multicast stream.

6.3.27 Configuring the Interval of Sending the Join/Prune Message

By default, the Join/Prune message is sent at the interval of 60 seconds by default. Use this command to modify this interval. With this command not configured, the default sending interval of join/prune packets is 60 seconds.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **jp-timer** *seconds*	Sets the interval of sending the Join/Prune message. *interval-seconds*: in the range of 1 to 65535 seconds.
no ip pim [**vrf** *vrf-name*] **jp-timer** [*interval-seconds*]	Restores the setting to the default value, namely 60s.


Note

When the sending interval of the join/prune packets is configured, if the interval times3.5 is more than 65535, a warning message is displayed and the interval is changed to 65535/3.5 seconds.

6.3.28 Allowing the Last Hop Device to Switch from the Shared Tree to the Shortest Path Tree

With this command configured, when the first (S, G) packet is received, a PIM join message is triggered and a source tree is created. If the keyword **group-list** is defined, all the groups specified will switch to the source tree. Use the **no** form of this command to enable the device to switch back to the shared tree and send a prune message.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **spt-threshold** [**group-list** *access-list*]	If **group-list** is defined, allows the last hop device of a specific group to switch from the shared tree to the shortest path tree. If **group-list** is not defined, allows all multicast groups.
no ip pim [**vrf** *vrf-name*] **spt-threshold** [**group-list** *access-list*]	Disables this function.

6.3.29 Configuring MIB in Dense Mode

Use this command to configure MIB in dense mode; With this command not configured, MIB is in sparse mode.

Use the following command in global configuration mode.

Command	Function
ip pim mib dense-mode	Uses MIB in dense mode.
no ip pim mib dense-mode	Uses MIB in sparse mode,

6.3.30 Enabling SSM

In the SSM mode, multicast packets can be directly received from the multicast source instead of through the RP tree.

Use the following command in global configuration mode.

Command	Function
ip pim [**vrf** *vrf-name*] **ssm** { **default** | **range** *access-list* }	Enables SSM.
no ip pim [**vrf** *vrf-name*] **ssm**	Disables SSM.

6.3.31 Monitoring and Maintaining PIM-SM

PIM-SM provides the following commands to monitor and maintain PIM-SM.

6.3.31.1 Displaying PIM-SM Information

Use the following commands to show information about PIM-SM on the local device.

Command	Function
show debugging	Shows the status of the debugging switch.
show ip pim sparse-mode [**vrf** *vrf-name*] **bsr-router**	Shows BSR details.
show ip pim sparse-mode [**vrf** *vrf-name*] **interface** [*interface-type interface-number*] [**detail**]	Shows the PIM-SM information of the interface.
show ip pim sparse-mode [**vrf** *vrf-name*] **local-members** [*interface-type interface-number*]	Shows local IGMP information about a PIM-SM interface.
show ip pim sparse-mode [**vrf** *vrf-name*] **mroute** [*group-or-source-address* [*group-or-source-address*]] [**proxy**]	Shows information about a PIM-SM multicast routing table. With the parameter **proxy**, this command shows the RPF Vector information about PIM-SM entries.
show ip pim sparse-mode [**vrf** *vrf-name*] **neighbor** [**detail**]	Shows information about the PIM-SM neighbors.
show ip pim sparse-mode [**vrf** *vrf-name*] **nexthop**	Shows the next hop of PIM-SM from NSM.
show ip pim sparse-mode [**vrf** *vrf-name*] rp-**hash** *group-address*	Shows a specified group address. *group-address* corresponds to RP information.
show ip pim sparse-mode [**vrf** *vrf-name*] **rp mapping**	Shows information about all current RPs and the groups they serve.
show ip pim sparse-mode [**vrf** *vrf-name*] **track**	Shows the number of PIM packets sent and received from the start time of statistics till now

☑ The parameter **proxy** in the command **show ip pim sparse-mode mroute** is only supported by RSR20, RSR30, RSR50 and RSR50E.

6.3.31.2 Deleting Internal Information About PIM-SM

Use the following commands to delete internal information about PIM-SM on the local device.

Command	Function
clear ip mroute [**vrf** *vrf-name*] { * | *group_address* [*source_address*] }	Deletes a multicast routing table entry.
clear ip mroute [**vrf** *vrf-name*] **statistics** { * | *group_address* [*source_address*] }	Deletes the statistics of a multicast routing table entry.
clear ip pim sparse-mode [**vrf** *vrf-name*] **bsr rp-set** *	Deletes RP-SET.
clear ip pim sparse-mode [**vrf** *vrf-name*] **track**	Resets the start time of statistics and clears the PIM packet counter

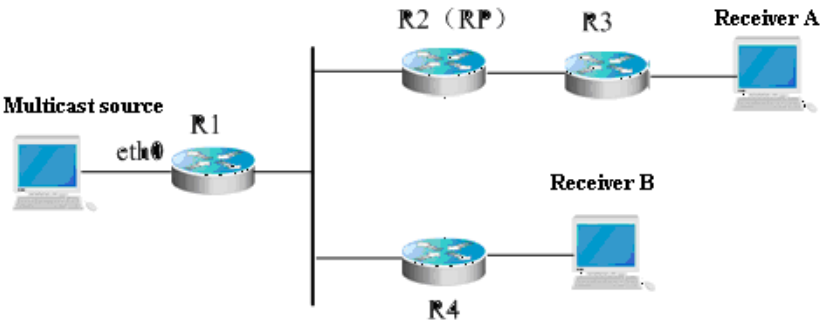
For details about the preceding commands, see *PIM-SM Commands*.

6.4 PIM-SM Configuration Example

Configuration Requirements

Figure 3 shows the network topology. R1 and the multicast source are in the same network. R2 will be set as an RP. R3 and receiver A are in the same network, and R4 and receiver B are in the same network. Assume that the devices connect to the host properly, IP addresses are configured on each interface, and IP unicast is enabled on each device.

Figure 15 Network topology for PIM-SM configuration



Device Configuration

Step 1: Enable multicast routing.

R1 is used as an example to show how to enable IP multicast routing. The configurations on R2, R3 and R4 are similar to R1.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
```

Step 2: Enable PIM-SM on the interface.

The following shows how to enable PIM-SM on Gi 0/1 of R1. The configurations on interfaces of R1, R2, R3 and R4 are similar.

```
Qtech(config)# interface GigabitEthernet 0/1
Qtech(config-if)# ip pim sparse-mode
Qtech(config-if)# end
```

Step 3: Configure the candidate BSR and the candidate RP.

Configure loopback 1 of R2 as C-BSR and C-RP.

```
Qtech(config)# interface loopback 1
Qtech(config-if)# ip address 100.1.1.1 255.255.255.0
Qtech(config-if)# ip pim sparse-mode
Qtech(config-if)# exit
Qtech(config)# ip pim bsr-candidate loopback 1
Qtech(config)# ip pim rp-candidate loopback 1
Qtech(config-if)# end
```

After the receivers join the group and the multicast source sends multicast streams, you can use the **show** command provided by PIM-SM to monitor the running status.

Note Enabling PIM-DM will automatically enable IGMP on each interface.

Note This example applies to both layer-3 switches and routers. However, if you need to configure the switch port as a layer-3 port, you must run the **no switchport** command (not needed for routers).

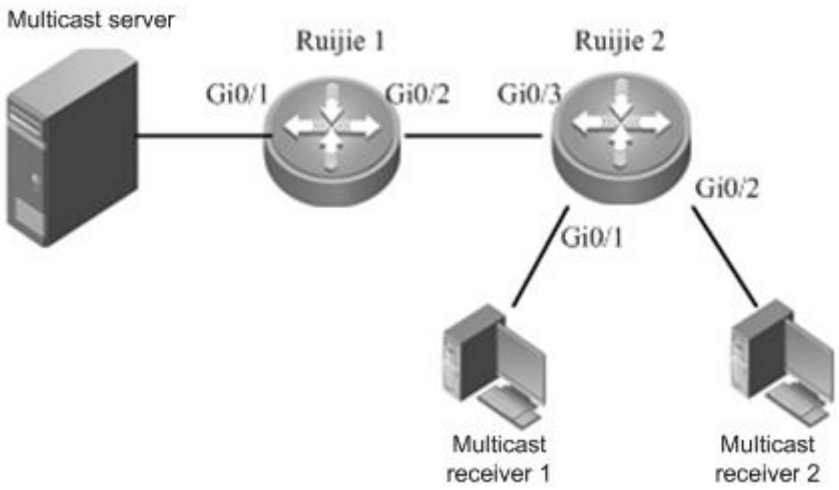
6.4.1 Enabling SSM based on PIM-SM

Networking Requirements

- The network must be interconnected on layer 3, for example, based on OSPF.
- The multicast receivers can obtain the information about the multicast source through some channels, depending on application software of hosts and deployment of network administrators.
- The PIM-SM protocol is applied within the network.

Networking Topology

Figure 16 Network topology for SSM configuration



Device name	Interface	IP address
Qtech 1	Gi 0/1	2.2.2.1/24
	Gi 0/2	3.3.3.1/24
Qtech 2	Gi 0/3	3.3.3.2/24
	Gi 0/1	4.4.4.1/24
	Gi 0/2	5.5.5.1/24

Configuration Steps

- Perform basic configuration on interfaces of Qtech 1 and Qtech 2.

Perform configurations on interfaces of Qtech 1 and Qtech 2 based on the IP addresses specified in the networking topology.

- Enable interworking on layer 3.

Enable OSPF on all interfaces of Qtech 1 and Qtech 2, implementing interworking on layer 3.

- Configure multicast on Qtech 1 and Qtech 2.

Firstly, enable the multicast routing on Qtech 1 and Qtech 2; secondly, enable PIM-SM on the interfaces of Qtech 1 and Qtech 2; at last, enable IGMPv3 on the interfaces where Qtech 2 connects to the multicast receivers. Qtech 2 is used as an example below.

Enable the multicast routing on Qtech 2.

```
Qtech(conf)#ip multicast-routing
```

Enable PIM-SM on all the interfaces of Qtech 2. Interface Gi 0/1 is used as an example below:

```
Qtech(conf-GigabitEthernet0/1)#ip pim sparse-mode
```

Enable IGMPv3 on the interfaces where Qtech 2 connects to the multicast receivers. Interface Gi 0/1 is used as an example below:

```
Qtech(conf-GigabitEthernet0/1)#ip igmp version 3
```

The configuration procedure for Qtech 1 is similar to that for Qtech 2.

- Enable the SSM function on Qtech 1 and Qtech 2.

Enable SSM on Qtech 2. The default group range of SSM is used, namely 232/8.

```
Qtech(conf)#ip pim ssm default
```

Enable SSM on Qtech 1. The default group range of SSM is used, namely 232/8.

```
Qtech(conf)#ip pim ssm default
```

Verification

- Multicast receivers 1 and 2 request the multicast service (2.2.2.2, 232.0.0.1) by sending IGMP (2.2.2.2, 232.0.0.1) join.

Protocol entry (2.2.2.2, 232.0.0.1) will be created on Qtech 2. Use the **show ip pim sparse-mode mroute** command to view it.

```
Qtech#show ip pim sparse-mode mroute
IP Multicast Routing Table
(*,*,RP) Entries: 0
(*,G) Entries: 0
(S,G) Entries: 1
(S,G,rpt) Entries: 1
FCR Entries: 0
REG Entries: 0

(2.2.2.2, 232.0.0.1)
RPF nbr: 3.3.3.1
RPF idx: GigabitEthernet0/3
SPT bit: 0
Upstream State: JOINED
kat expires in 175 seconds
  00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27
28 29 30 31
Local
0.i
i  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .
1  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .  .  .  .
Joined
0  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .  .  .  .
1  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .  .  .  .
Asserted
0  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .  .  .  .
1  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .  .  .  .
Outgoing
0.o
o  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .
1  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 .  .  .  .
```

As shown in the preceding information, the protocol entry (2.2.2.2, 232.0.0.1) has been created on Qtech 2 and there are two multicast receivers.

- The multicast server sends data stream (2.2.2.2, 232.0.0.1).

Multicast forwarding table (2.2.2.2, 232.0.0.1) will be created on Qtech 2. Use the **show ip mroute** command to view it.

```
Qtech#show ip mroute

IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed, R - RPT, S - SPT
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(2.2.2.2, 232.0.0.1), uptime 00:19:31, stat expires 00:02:53
Owner PIMSM, Flags: TFSs
  Incoming interface: GigabitEthernet 0/3
  Outgoing interface list:
      GigabitEthernet 0/1(1)
      GigabitEthernet 0/2(1)
```

As shown in the preceding information, the multicast forwarding table (2.2.2.2, 232.0.0.1) has been created on Qtech 2. Multicast egresses Gi 0/1 and Gi 0/2 are connected to multicast receivers 1 and 2 respectively. The forwarding table has been marked with the flag “s", which indicates that the table is in the SSM model.

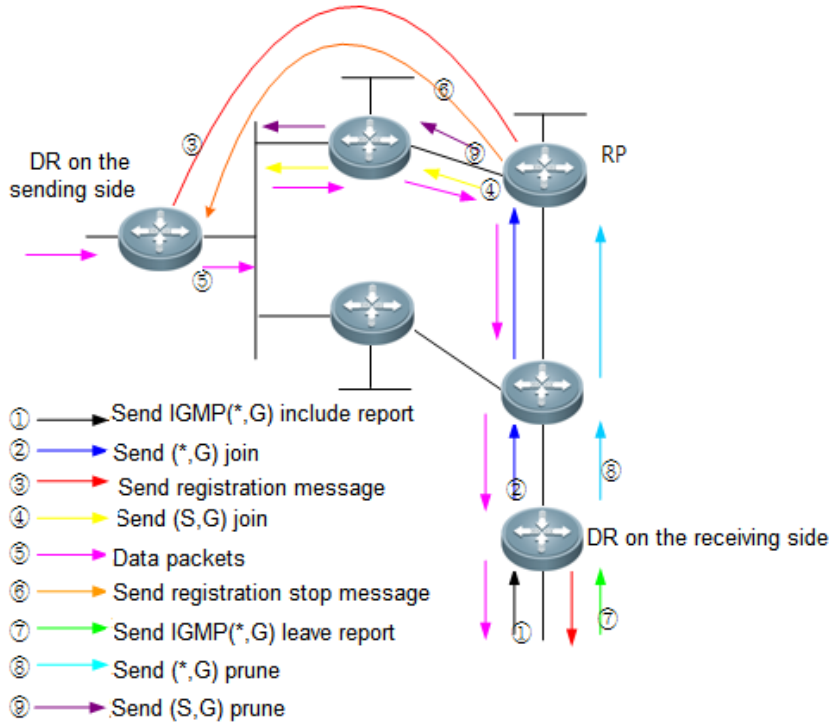
7 CONFIGURING PIM-SMV6

7.1 PIM-SM Overview

PIM (Protocol Independent Multicast) is designed by IDMR (Inter-domain Multicast Routing) working group. As its name implied, PIM does not depend on a specific unicast routing protocol. It utilizes the unicast routing table established by various unicast routing protocols to enable the RPF check function instead of maintaining a separate multicast routing table for forwarding multicast packets. Compared with other multicast protocols, PIM overhead falls down at large extent for PIM does not need to receive and send multicast route update. The concept behind PIM design is that support and flexible transformation between SPT and the shared tree is enabled for higher multicast efficiency. There are two kinds of PIM modes-dense mode and sparse mode.

PIM-SM (Protocol Independent Multicast Sparse Mode) is a multicast routing protocol in sparse mode. In the PIM-SM domain, the device running PIM-SM sends the Hello message at a specific interval to discover adjacent devices running PIM-SM and be in charge of DR election. Here DR sends the “join/prune” message to its direct group members in the direction of the root node of the multicast distribution tree or sends the data from the direct multicast source to the multicast distribution tree.

Figure 17 PIM-SM explicit join/prune mechanism



PIM-SM forwards multicast data packets by setting up a multicast distribution tree. There are two kinds of multicast distribution tree: the shared tree using G’s RP as root and the shortest path tree using the multicast source as root. With the explicit join/prune mechanism, PIM-SM sets up and maintains the multicast distribution tree.

As illustrated in the above figure:

1. The DR on the receiving side receives the MLD (*,G) report message from a host on the same side.
2. If the DR on the receiving side is not the RP of G, it will send a (*.G) Join message to the RP. Upon receiving this message, the upstream router will send the (*.G) Join message to the RP without the forwarding entry of the corresponding group. In this way, the (*,G) Join message is transmitted hop to hop until G’s RP receives the (*,G) Join message, indicating that the DR joins the shared tree.
3. When the source host sends multicast data to a group, the data is encapsulated in the registration message and sent by the DR on the source side to the RP in unicast form. Then, the RP decapsulates the registration message, extracts the data and forwards the data to every member of the group along the shared tree.
4. The RP sends the (S,G) Join message to the DR on the source side to join its shortest path tree.

5. After the shortest path tree from the RP to the DR on the source side is established, data packets are sent to the RP along this STP without encapsulation.
6. When the first multicast data arrives along the SPT, the RP sends the registration stop message to the DR on the source side, notifying the DR of stopping registration and encapsulation. Upon receiving the registration stop message, the DR no longer registers and encapsulates data packets. Instead, it sends data packets to the RP along the shortest path tree, which then forwards data packets to every group member along the shared tree.
7. When a receiving end does not need multicast packets, it sends the MLD leave message.
8. The DR on the receiving side sends the prune message in multicast form hop by hop to G's RP, the prune message arrives at the RP or the router along the way to the RP which has other (*,G) receiver so that data packets are not sent to this receiving side any more.
9. If the RP has no downstream receiver at present, it sends (S,G) prune message to the DR of the data source hop by hop. Consequently, the DR on the source side prunes the interface receiving this (S,G) prune message. In this way, data packets are filtered on the DR of the data source.

RP election is also involved in PIM-SM. when there is one or more candidate BSRs configured in the PIM-SM domain, some rule is applied to elect BSR. Candidate RPs are also configured in the PIM-SM domain, which send the packets containing their addresses and serviceable multicast groups to BSRs in unicast form. BSRs generate bootstrap messages with a series of candidate RPs and the addresses of corresponding multicast groups periodically. These bootstrap messages are transmitted in the overall domain hop by hop. Devices will receive and save these bootstrap messages. Upon receiving the member relation report of a multicast group from the directly connected device, the DR will use a hash algorithm and map the multicast group address to a candidate RP who can serve this group if it has not the routing entry of this group. Then the DR will send the join/prune message in multicast form hop by hop along the way to the RP. On the other hand, upon receiving multicast packets from the directly connected device, the DR will use a hash algorithm and map the multicast group address to a candidate RP who can serve this group, and then encapsulate these multicast packets in the registration message and send it to the PR in unicast form.

The essential difference between PIM-SM and PIM-DM is that PIM-SM is based on explicit join mode and PIM-DM is based on flood/prune mode. For PIM-SM, the receiver sends a join message to the PR, but the device forwards the packets of a multicast group only on the interface joining this multicast group. PIM-SM forwards multicast packets through the shared tree. Each multicast group has a rendezvous point. The multicast source sends packets to the RP along the shortest path, and then the RP sends the packets to every receiver along the shortest path. This process is similar to CBT. However, the core concept is not used in PIM-SMv6. One of the main advantages of PIM-SMv6 is that it not only receives multicast packets through the shared tree but also offers the shared tree-to-SPT transformation mechanism. This transformation consumes a lot number of resources, even though it reduces network delay and possible block on the RP. It is suitable for the environment where there are many pairs of multicast sources yet fewer networks.

PIM-SM distributes multicast packets through the shared tree and SPT. Assume that other devices do not need to receive these multicast packets, unless otherwise specified. When a host joins a multicast group, the devices connecting to the host notify the root (or RP) through the PIM join message. This message is transferred among these devices in order to set up the structure of a shared tree. So, the RP records this transmission path and the registration message from the first hop device (DR) of the sending multicast source, and perfects the shared tree based on these two messages. Update of leaf messages is enabled on periodic query message. For the shared tree, the multicast source sends multicast packets to the RP so that all receivers can receive these multicast packets. *.G indicates a tree, in which * indicates all sources and G indicates the specific multicast address. The prune message is also used in the shared tree when leafs do not need to receive multicast packets.

PIMv2 BSR distributes the group-to-RP message to all devices without the necessity for configuring RP for every device. The BSR distributes the mapping message through the hop-by-hop flooding BSR message. First of all, the BSR is elected among devices. This election procedure is similar to electing the root bridge in STP by priority. Every BSR device checks the BSR message, and only forwards the BSR messages with higher or equivalent priority (or higher IP address). The elected BSR sends the BSR message to the all-PIM-routers multicast group (ff02::d) with TTL 1. Upon receiving the BSR message, the adjacent PIMv2 device sends it out in multicast form and then reset TTL to 1. In this way, the BSR message is sent to all devices hop by hop. Since the BSR message includes the IP addresses of BSR devices, the candidate BSR can determine which device is the current BSR device. The candidate RP sends the candidate RP advertisement and alleges in which address ranges it can become RP. The BSR stores the advertisement message in its local candidate RP cache, and notifies all PIM devices of local candidate RPs periodically. Also in this way, the message is sent to all devices hop by hop.

Embedded RP is a new feature introduced in IPv6 multicast. It provides a new type of mapping from IPv6 multicast group to RP to replace the RP configured statically or the one calculated by BSR mechanism dynamically.

Some special IPv6 multicast addresses are called that of the embedded RP, the range of which is FF70::/12. You can obtain IPv6 unicast addresses by calculating the special IPv6 multicast group addresses, which means a RP address is embedded in each IPv6 multicast group address.

If the multicast group address extracted from MLD packets, PIM packets and multicast data packets by routers is the IPv6 multicast address of embedded RP, the router can calculate the RP address from this multicast group address directly. The embedded-RP multicast groups have the following advantages:

1) Since an IPv6 multicast group address is bound with a RP address. You can calculate the RP address directly rather than learning and updating the group RP mapping.

2) It can be deployed not only for intra-domain multicast, but also for inter-domain multicast. For IPv6 multicast group addresses of the same embedded RP, the same RP address is calculation no matter where the router is located.

7.2 SSM Model for PIM

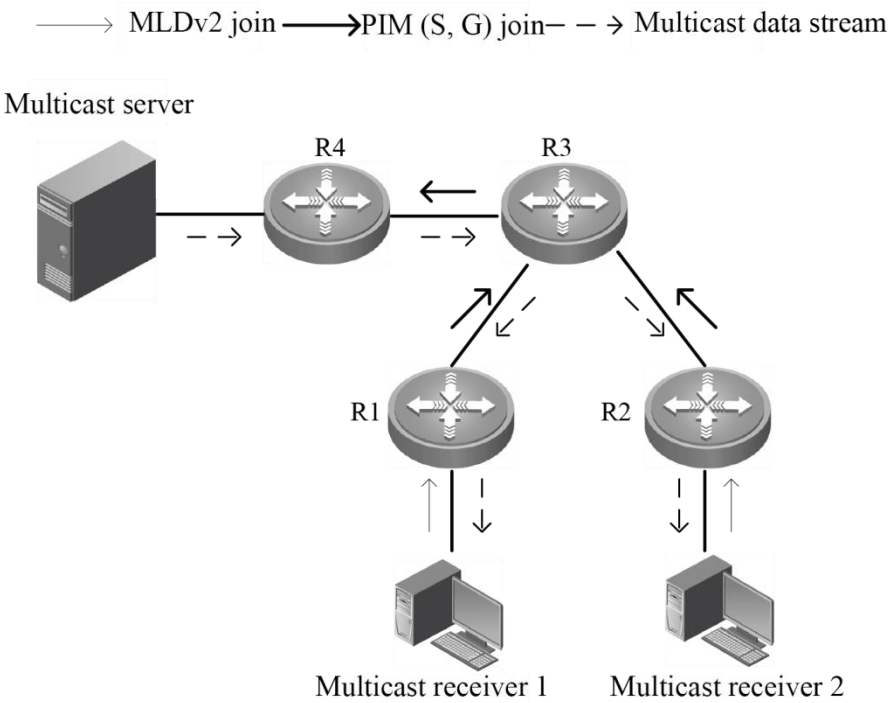
The traditional PIM-SM mode and PIM-DM mode are Any-Source Multicast (ASM) models, in which multicast data receivers specify only a specific multicast group G.

Source-Specific Multicast (SSM) and ASM are two independent models. The SSM model can be implemented by the PIM-SM technology, in which multicast data receivers specify both the multicast source S and multicast group G.

The PIM-based SSM model provides a solution for specifying the source multicast. It requires MLDv2 to manage the membership between hosts and routers and uses PIM-SM to connect hosts and routers.

In the SSM model, multicast receivers learn information about the multicast source (S,G) through a channel, such as accessing servers or accepting advertisements. Therefore, when the multicast receiver requests a multicast service, it can send the MLD (S, G) Join message directly to the last-hop router, as shown in Figure 2, and the multicast receiver 1 sends the MLD (S, G) Join message to request the multicast service (S, G). After receiving the MLD (S, G) Join message from the multicast receiver, the last-hop router sends PIM (S ,G) to the multicast source hop by hop, as shown in Figure 2. After receiving the MLD (S, G) Join message from multicast receiver 1, R1 sends PIM (S ,G) Join message to R3, and then R3 sends PIM (S ,G) Join message to R4. As a result, a shortest path tree from multicast receivers to the multicast source is established, as shown in Figure 2.

Figure 18 SSM working model



Requirements for SSM models:

1) The multicast receiver learns information about the multicast source (S, G) through a channel and initiates an MLD (S, G) Join message to the multicast service.
2) The last-hop router connects to the interface of multicast receivers and must be enabled with MLDv2 but not MLDv1. MLDv1 cannot processing the MLD Join message of the specified source.
3) It is recommended to enable PIM-SSM on path routers from multicast receivers to the multicast source. Since PIM-SSM is compatible with PIM-SM, you can enable only PIM-SM without enabling PIM-SSM.

By default, the multicast group range reserved for SSM is FF3x::/32. However, you can expand or shrink the multicast group range of SSM, or even disable the SSM function at the CLI. For more information, see the commands described in Configuring the Specific Source Multicast.

Compared to PIM-SM, PIM-SM-based SSM has the following features:

1) In the SSM model, multicast receivers can obtain information about the multicast source in advance through a channel, such as advertisements or accessing specified servers.
2) As a specific subset of PIM-SM, the SSM model processes only PIM (S, G) Join message and PIM (S, G) prune messages and discards RPT-associated messages that fall in the range of SSM, such as PIM (*, G) Join message and prune messages. For register messages fall into the range of SSM, the model will respond to the register-stop message immediately.
3) In the SSM model, RP information selection and distribution are unnecessary. The established multicast distribution tree in SSM are shortest path tree (SPT).

7.3 PIM-SMv6 Configuration Preparation

Enable a unicast routing protocol, for instance OSPFv3, to automatically discover routes.

7.4 PIM-SMv6 Configuration Task List

PIM-SMv6 configuration includes the following tasks, but only the first and the second tasks are mandatory, and you can determine whether to configure the other tasks based on actual networks.

Enabling multicase routing

Enabling PIM-SMv6

Configuring Hello message transmission interval

Configuring the propagation delay of the Hello message

Configuring the override interval of the Hello message

Configuring the neignbor tracking of the Hello message

Configuring the triggered Hello delay of the Hello message

Configuring PIM-SMv6 neighbor filtering

Configuring DR priority

Configuring static RP

Configure candidate BSR

Configure BSR border

Ignoring the RP priority of RP-SET

Configuring candidate RP

Checking the reachability of RP registration message

Filtering the addresses of registration packets on RP

Limiting the rate to send registration packets

Configuring the calculation method of checksum of registration packets

Limiting the range of legal BSRs

Configuring elected BSR to limit the address range of legal candidate RP and the multicast group range it serves

Enabling elected BSR to receive the candidate RP advertisement whose prefix-count is 0

7.4.1

Please configure this command to make elected BSR receive C-RP-ADV whose prefix-count is 0. If this command is not configured, the elected BSR cannot deal with C-RP-ADV whose prefix-count is 0.

Please run the following command in the global configuration mode.

Command	Function
ipv6 pim accept-crp-with-null-group	Elected BSR can receive C-RP-ADV whose prefix-count is 0.
no ipv6 pim accept-crp-with-null-group	Elected BSR cannot receive C-RP-ADV whose prefix-count is 0.

7.4.2 Enabling Multicast Routing

Multicast packets can be forwarded and enabling PIM-SMv6 makes sense only after multicast routing is enabled.

To enable or disable multicast routing, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 multicast-routing**	Enable multicast routing globally.
Qtech(config)# **no ipv6 multicast-routing**	Disable multicast routing globally.

7.4.3 Enabling PIM-SMv6

In order for one device to interact with other devices on PIM-SMv6 control messages, maintain and update the multicast routing table and forward multicast packets, PIM-SMv6 must be enabled on every interface.

To enable PIM-SMv6 on the interface, run the following commands in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim sparse-mode**	Enable PIM-SMv6 on the interface.
Qtech(config-if)# **no ipv6 pim sparse-mode**	Disable PIM-SMv6 on the interface.

Enabling PIM-SMv6 brings into effect only after multicast routing is enabled globally.

If an interface is the tunnel interface, only IPv6 over IPv4 tunnel, IPv6 over IPv4 GRE tunnel, IPv6 over IPv6 tunnel and IPv6 over IPv6 GRE tunnel support IPv6 multicast. You can also enable multicast on tunnel interfaces not supporting multicast. However, the system neither prompts error messages nor receives or sends multicast packets.


Caution

Multicast tunnels can be set up only on Ethernet interfaces. They do not support embedded tunnels or QoS/ACL of multicast data.

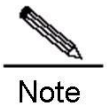
7.4.4 Configuring the Hello Message Transmission Interval

After enabling PIM-SMv6, the interface sends the Hello message to the ones of adjacent devices at an interval. This transmission interval can be set as required.

To configure the Hello message transmission interval, run the following command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim query-interval** *interval-seconds* Qtech(config-if)# **no ipv6 pim query-interval**	Set the Hello message transmission interval. *interval-seconds*: in the range 1 to 65535 seconds Restore the setting to the default value.

By default, the Hello message transmission interval is 30 seconds on the interface.


Note

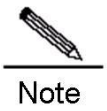
When the Hello message transmission interval is updated every time, the Hello message hold time is updated 3.5 times of the Hello message transmission interval. If the Hello message transmission interval multiplying 3.5 is larger than 65535, the Hello message hold time is updated to 65535.

7.4.5 Configuring the Propagation Delay of the Hello Message

After the interface sends the Hello message, you can set the options of the Hello message. For LAN prune delay, the Propagation_Delay field is 500 ms by default.

To configure the propagation delay, run the following command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim propagation-delay** *interval-milliseconds* Qtech(config-if)# **no ipv6 pim propagation-delay**	Set the propagation delay. *interval-milliseconds:* in the range of 1-32767 ms. Restore the propagation delay setting to the default value, namely 500 ms.


Note

Change of propagation delay or prune delay will influence the override interval of Join/prune message. As specified in the protocol, the override interval of Join/prune message must be less than its hold time or otherwise this will cause temporary interruption.

7.4.6 Configuring the Override Interval of the Hello Message

After the interface sends the Hello message, you can set the options of the Hello message. For LAN prune delay, the override-interval field is defaulted to 2500ms.

To configure the override-interval field, run the following command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim override-interval** *interval-milliseconds*	Set the override-interval. *interval-milliseconds:* In the range of 1 to 65535 seconds.

Qtech(config-if)# **no ipv6 pim override-interval**	Restore the setting to the default value.

Note

Change of propagation delay or prune delay will influence the override interval of Join/prune message. As specified in the protocol, the override interval of Join/prune message must be less than its hold time or otherwise this will cause temporary interruption.

7.4.7 Configuring the Neighbor Tracking of the Hello Message

After an interface sends the Hello message, the LAN Prune Delay option of the Hello message has a T bit, indicating whether join constraint is enabled on the interface. With join constraint enabled, the interface is constrained not to send its Join message to the upstream neighbor when it receives the Join message that its neighbor sends to the upstream neighbor. On the other hand, with join constrain disabled, the interface will send its Join message to the upstream neighbor when it receives the Join message that its neighbor sends to the upstream neighbor. This function allows upstream routers to track how many receivers in downstream in accord with all received Join messages. Join constraint is enabled on the interface by default.

To disable join constraint on the interface, run the following command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim neighbor-tracking**	Disable join constraint on the interface.
Qtech(config-if)# **no ipv6 pim neighbor-tracking**	Enable join constraint on the interface

7.4.8 Configuring the Triggered Hello Delay of the Hello Message

When the router starts or detects a new neighbor starts, it sends the Hello message. To avoid congestion, the router will send the Hello message at random. This random time is calculated by triggered hello delay, which is 5s by default.

To configure triggered hello delay, run the command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim triggered-hello-delay** *interval-seconds*	Configure triggered hello delay. *interval-seconds*: In the range of 1 to 5 seconds
Qtech(config-if)# **no ipv6 pim triggered-hello-delay**	Restore the setting to the default value.

7.4.9 Configuring PIM-SMv6 Neighbor Filtering

Neighbor filtering can be enabled on the interface for security. PIM-SMv6 will not establish adjacency relation with a neighbor device and delete the established adjacency relation as long as the neighbor device is denied by the filtering access list.

To configure neighbor filtering, run the following command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim neighbor-filter** *ipv6_access-list*	Enable neighbor filtering on the interface.
Qtech(config-if)# **no ipv6 pim neighbor-filter** *ipv6_access-list*	Disable neighbor filtering on the interface.

By default, neighbor filtering is disabled on the interface.


Note

Only the neighbors whose IP address matches ACL filtering can serve as the PIM neighbors of the interface. The neighbor addresses filtered by ACL cannot function as the PIM neighbors of the interface.

7.4.10 Configuring DR Priority

Use this command to set the priority of a device. The higher the value, the higher the priority.

Run the following commands in the interface mode:

Command	Function
Qtech(config-if)# **ipv6 pim dr-priority** *priority-value*	Configure DR priority in the range of 0 to 4294967294.
Qtech(config-if)# **no ipv6 pim dr-priority** *priority-value*	Restore DR priority to the default value, namely 1.

7.4.11 Configuring Static RP

In a smaller network, you can use PIM-SMv6 by configuring static RP. All devices in the PIM-SMv6 domain should be configured with similar static RP for consistent PIM-SMv6 multicast routes.

To configure static RP, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim rp-address** *ipv6_rp-address* [*ipv6_access-list*]	Configure static RP.
Qtech(config)# **no ipv6 pim rp-address** *ipv6_rp-address* [*ipv6_access-list*]	Remove the configuration.

Note that:

- When BSR and static RP take effect simultaneously, static RP is preferred.
- Static RP can be configured for many multicast groups (by using ACL) or all multicast groups. However, one static RP can be configured only once.
- If more than one IPv6 address is configured to be RP, the highest IPv6 address is adapted first.
- Only the IPv6 addresses defined and permitted by ACL are valid multicast groups. By default, all multicast groups are permitted.
- After configuration, the source address of static RP is inserted in the tree structure of the group range-based static RP group. Each group range-based static RP group maintains a chain structure that lists static RP groups in the descending order of IPv6 addresses. When a group range selects a RP, the first element, or the highest IPv6 address is selected.
- Deleting a static RP will delete it from all multicast groups and a new one is selected from the static RP tree structure as static RP.

7.4.12 Configuring Candidate BSR

Configuration of candidate BSR produces globally unique BSR in the PIM-SMv6 domain, which collects and distributes the RPs in the domain for the uniqueness of RP mapping.

To configure candidate BSR, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim bsr-candidate** *interface-type interface-number* [*hash-mask-length*] [*priority-value*]	Set the device as the candidate BSR. *hash-mask-length:* In the range 0 to 128, 126 by default. *priority-value:* In the range 0 to 255, 64 by default.
Qtech(config)# **no ipv6 pim bsr-candidate**	Remove the configuration.


Note

To set an interface as the candidate BSR, it must be configured with a global unicast address used for unicast routes or a local address. The first global unicast address or local address is elected as the candidate BSR.

7.4.13 Configuring BSR Border

To restrain BSM flooding, configure BSR border on the interface so that the interface drops BSM messages upon receiving them.

To configure BSR border, run the following command in the interface configuration mode.

Command	Function
Qtech(config-if)# **ipv6 pim bsr-border**	Set the interface to be BSR border.
Qtech(config-if)# **no ipv6 pim bsr-border**	Remove the configuration.

7.4.14 Ignoring the RP Priority of RP-Set

When you select a RP for a multicast address, ignore their priorities in comparison if there are more than one RP. If this command is not configured, the priority is used to compare two RPs.

To ignore RP priority, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim ignore-rp-set-priority**	Ignore RP priority of RP-SET.
Qtech(config)# **no ipv6 pim ignore-rp-set-priority**	Compare the RP priority of RP-SET.

7.4.15 Configuring Candidate RP

Configure candidate RP to periodically send candidate RP advertisement to the BSR so that the candidate RP advertisement is propagated to all PIM-SMv6 devices in the domain and guarantee the uniqueness of RP mapping.

To configure candidate RP, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim rp-candidate** *interface-type interface-number* [**priority** *priority-value*] [**interval** *interval-seconds*] [**group-list** *ipv6_access-list*]	Configure candidate RP. *priority-value*: In the range 0 to 255, 192 by default *interval-seconds*: In the range 1 to 16383s, 60s by default *ipv6_access-list*: All multicast groups are allowed by default.
Qtech(config)# **no ipv6 pim rp-candidate**	Remove the configuration.


Note

To set an interface as the candidate RP of the specific group range, use this command with ACL option. Note that the calculation of group range is only based on the ACE with permit rule, not deny rule.

7.4.16 Checking the Reachability of RP Registration Message

This command can be used to check whether the RP is reachable before the DR sends the registration message to the RP. In other words, the DR queries the unicast routing and static multicast routing tables to learn whether a route to the RP exists. If not, DR does not send the register packet.

To check the reachability of RP, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim register-rp-reachability**	Check the reachability of the RP.
Qtech(config)# **no ipv6 pim register-rp-reachability**	Remove the configuration.


Caution

If there is a static multicast route to the RP and the next hop of this static multicast route is reachable in the unicast routing table, PIM-SMv6 considers that the RP is reachable, even though the RP is not reachable in the unicast routing table.

7.4.17 Filtering the Addresses of Registration Packets on RP

This command filters the source addresses and group addresses of the registration packets arrived on RP. If this command is not configured, each register packet is allowed to reach the RP. Only the registration packets whose source addresses and group addresses are permitted by ACL or the route map are processed. Other registration packets are filtered and the Register-stop message is sent back.

Run the following command in global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim accept-register** {**list** *ipv6_access-list* | **route-map** *map-name* } Qtech(config)# **no ipv6 pim accept-register**	Filter the source addresses and group addresses of registration packets. Remove the configuration.

7.4.18 Limiting the Rate to Send Registration Packets

This command applies to the registration packets in (S, G) state, not the overall system.

To limit the rate to send registration packets, run the following commands in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim register-rate-limit** *rate* Qtech(config)# **no ipv6 pim register-rate-limit**	Set the maximum number of registration packets sent per second, in the range of 1 to 65535. Remove the configuration.

7.4.19 Configuring the Calculation Method of Checksum of Registration Packets

This command calculates the checksum of all the packets of PIM protocol, including encapsulated multicast packets.

Without this command, the checksum of registration packets is calculated by default methods of PIM protocol.

To configure the calculation method of checksum of registration packets, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim register-checksum-wholepkt** [**group-list** *ipv6_access-list*]	Calculate the checksum for registering packets based on the overall packets. When the **group-list** *ipv6 access-list* value is the default value, apply the configuration to all multicast addresses.

Qtech(config)# **no ipv6 pim register-checksum-wholepkt** [**group-list** *ipv6_access-list*]	Remove the checksum calculation for registering packets. When the **group-list** *ipv6 access-list* value is the default value, apply the configuration to all multicast addresses.

Note

Some vendors calculate checksum based on the overall registration packets. Qtech Networks introduces this function for the compatibility with these vendors. If the device of other vendors serves as the RP and the Qtech device serves as the source DR, you can use this command on the source DR. If the device of other vendors serves as the source DR and the Qtech device serves as the RP, you can use this command on the RP.

7.4.20 Limiting the Range of Legal BSRs

This command limits the range of legal BSRs. Without this command the PIM-SMv6 enabled router receives all external BSM packets.

To limit the range of legal BSRs, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim accept-bsr list** *WORD*	Filter the BSM packets of BSRs.
Qtech(config)# **no ipv6 pim accept-bsr list**	Remove the configuration.

Note

This command filters the BSR address field of BSM packets. If the address of a BSM packet is denied by ACL, the BSM packet is filtered.

7.4.21 Configuring Elected BSR to Limit the Address Range of Legal Candidate RP and the Multicast Group Range it Serves

This command configures elected BSR to limit the address range of legal candidate RP and the multicast group range it serves. Without this command, the elected BSR receives all external advertisements of candidate RPs.

For the ACL rule of this command, source specifies the address of candidate RP, and destination specifies the multicast group range that the candidate RP serves. If the ACL denies both addresses, the multicast group range of the candidate RP is filtered.

To configure elected BSR to limit the address range of legal candidate RP and the multicast group range it serves, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim accept-crp list** *WORD*	Elected BSR filters the advertisement of candidate RP.
Qtech(config)# **no ipv6 pim accept-crp**	Remove the configuration.

7.4.22 Enabling Elected BSR to Receive the Candidate RP Advertisement whose Prefix-count is 0

This command enables the elected BSR to receive the candidate RP advertisement whose prefix-count is 0. Without this command, the elected BSR will not process this kind of packets.

Once configured, the elected BSR considers that the candidate RP supports all multicast groups upon receiving the candidate RP advertisement whose prefix-count is 0.

To enable the elected BSR to receive the candidate RP advertisement whose prefix-count is 0, run this command in the global configuration mode.

Command	Function
ipv6 pim accept-crp-with-null-group	The elected BSR can receive the C-RP-ADV packet whose prefix-count is 0.
no ipv6 pim accept-crp-with-null-group	The elected BSR cannot receive the C-RP-ADV packet whose prefix-count is 0.

7.4.23 Configuring the Source Address of Registration Packets

This command configures the source address of registration packets. Without this command or with the no form of this command, the interface address of DR connecting to the multicast source is used. For address parameter of this command, the address to be set must be reachable to unicast routes. For interface parameter of this command, the interface to be set must be loopback interface or other type of interface whose address is advertised by unicast routes, in which the first non-local link address of the interface serves as the source address of registration packets.

To configure the source address of registration packets, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim register-source** {*ipv6_local_address* | *interface-type interface-number*}	Configure the source address of registration packets.
Qtech(config)# **no ipv6 pim register-source**	Use the RPF's interface address as the source address of registration packets.

7.4.24 Configuring the Suppression Time of Registration Packets

When the receiver does not receive the data packets destined to a multicast group from RP (namely RP does not serve this multicast group) or RP begins to receive multicast packets from the multicast source, RP sends the registration stop message to the DR on the multicast source side. Upon receiving this message, DR stops sending the registration packets encapsulated with multicast packets and transfers into the register suppression state.

During registration suppression, DR sends null registration packets, namely registration packets not encapsulated with multicast packets), to DR, indicating that the multicast source is still active. Probe time refers to the period that DR is allowed to send null registration packets before the registration suppression state is timed out. When registration suppression is timed out, DR starts to send registration packets. The smaller registration suppression timeout means the higher frequency that RP receives multicast packets; the larger timeout means the higher delay for a receiver to join a multicast group.

Running this command on DR will change the registration packet suppression time defined on DR.

To configure the registration packet suppression time, run this command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim register-suppression** *seconds*	Configure registration packet suppression time. *Seconds*: in the range *1 to 65535*s
Qtech(config)# **no ipv6 pim register-suppression**	Configure the suppression time to 60s.

7.4.25 Configuring the Probe Time of Null Registration Packet

The DR can send the null registration message to the RP in a period before the registration suppression time expires. This period is call probe time, 5 seconds by default

To configure the probe time of null registration packet, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim probe-interval** *interval-seconds*	Configure the probe time of null registration packet. *interval-seconds*: in the range 1 to 65535 seconds
Qtech(config)# **no ipv6 pim probe-interval**	Restore the probe time to 5s.


Note

The probe time must be less than half of registration suppression time. Furthermore, 3* registration suppression time plus registration probe time should be no more than 65535s or otherwise the system triggers an alarm.

7.4.26 Configuring RP KAT Timer

This command configures the hold time of (S, G) state that registration packets set up on RP.

To configure RP KAT timer, run this command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim rp-register-kat** *seconds*	Configure the KAT timer, which ranges from 1 to 65535 seconds.
Qtech(config)# **no ipv6 pim rp-register-kat**	Restore the setting to the default value, namely 3* registration suppression time plus registration probe time.


Caution

The timer should be larger than 3* registration suppression time plus registration probe time on the source DR, or otherwise the RP may time out the (S, G) state before the source DR sends the registration packet and thus leading to temporary interruption of multicast packets.

7.4.27 Configuring the Join/Prune Message Sending Interval

By default, the Join/Prune message is sent at the interval of 60s. You can use this command to change the sending interval.

To configure the Join/Prune message sending interval, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim jp-timer** *interval-seconds*	Configure the Join/Prune message sending interval. *interval-seconds*: in the range 1 to 65535s
Qtech(config)# **no ipv6 pim jp-timer**	Restore the setting to the default value, namely 60s.


Note

When you configure the Join/Prune message sending interval, if the sending interval * 3.5 is larger than 65535s, the system triggers an alarm and the sending interval is changed to be 65535/3.5 seconds.

7.4.28 Enabling the Last Hop Device to Transfer from the Shared Tree to the Shortest Path Tree

With this command, a PIM join message is triggered and a source tree is constructed upon the receipt of the first (S, G) message. The keyword **group-list** means all the groups in the list transfer to the source tree. After you use the **no** form of this command without the keyword **group-list**, to disable switching from the shared tree to the SPT, the device re-directs to the shared tree and sends a prune message to the source. If you use the previous command with a configured ACL, the association between the ACL and the **group-list** is removed, and all groups can switch from the shared tree to the SPT.

To enable the last hop device to transfer from the shared tree to the shortest path tee, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim spt-threshold** [**group-list** *ipv6_access-list*]	If group-list is configured, the last hop device of this multicast group is enabled to transfer from the shared tree to the shortest path tree. Without group-list, all multicast groups are permitted.
Qtech(config)# **no ipv6 pim spt-threshold** [**group-lis**t *ipv6_access-list*]	If the group-list is defined and the carried ACL is configured, the association between the ACL and the group-list is removed, and all multicast groups can switch from the shared tree to the shortest path tree. If group-list is not defined, the function of switching from the shared tree to the SPT is disabled.

7.4.29 Configuring the Specific Source Multicast

This command enables the device to directly receive multicast packets from the specific multicast source rather than the PR tree.

To configure the specific source multicast, run the following command in the global configuration mode..

Command	Function
Qtech(config)# **ipv6 pim ssm** {**default** |**range** *ipv6_access-list*]}	Configure the specific source multicast, FF3x::/32 by default. x is an any valid range. If the ACL is specified, SSM is applied only to multicast groups that fall into this ACL.
Qtech(config)# **no ipv6 pim ssm**	Remove the configuration.

Note

Interfaces on the last-hop switch or router connected to multicast receivers must enable MLDv2. Otherwise, the last-hop switch or router cannot process the MLD (S, G) Join message, and the specified source multicast will fail.

7.4.30 Configuring Static RP Preference

To configure the static RP's priority higher than the one elected through BSR mechanism, run the following command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim static-rp-preferred**	Configure the static RP's priority higher than the one elected through BSR mechanism.
Qtech(config)# **no ipv6 pim static-rp-preferred**	Configure the RP priority elected through BSR mechanism higher than the one of static RP.

7.4.31 Enabling Embedded RP

Embedded RP is the special RP discovery mechanism for IPv6 PIM that uses its IPv6 multicast address, from which the multicast router can directly resolve RP's address.

By default, embedded RP is enabled for the IPv6 multicast addresses of all embedded RP addresses. To enable embedded RP for the IPv6 multicast address of some embedded RP address, run this command in the global configuration mode.

Command	Function
Qtech(config)# **ipv6 pim rp embedded** [**group-list** *ipv6_acl_name*]	Without *ipv6_acl_name,* enable embedded RP for the IPv6 multicast addresses of all embedded RP addresses. With *ipv6_acl_name,* enable embedded RP for the IPv6 multicast address of some embedded RP address.

Qtech(config)# **no ipv6 pim rp embedded**	Disable embedded RP for the IPv6 multicast addresses of all embedded RP addresses. After embedded RP is disabled, the group address of the embedded RP is considered as an ordinary one, which means the RP mapping to the group address is no longer the embedded RP. It must calculate the embedded RP based on the learned RP mapping or the one configured statically.

Note

In addition to enabling embedded RP on the device, which is enabled by default, you also need to configure static RP on devices acting as embedded RP, or otherwise the device cannot serve as RP, even though its interface has the same address as embedded RP.

7.5 PIM-SMv6 Monitoring and Maintenance

PIM-SMv6 offers show commands to show the information on PIM-SMv6 interface, multicast group and multicast routing table.

7.5.1 Showing PIM-SMv6 Status

Use the following commands to show PIM-SMv6 status.

Command	Function
Qtech# **show debugging**	Show debugging switches.
Qtech# **show ipv6 pim sparse-mode bsr-router**	Show BSR details.
Qtech# **show ipv6 pim sparse-mode interface** [*interface-type interface-number* [**detail**]]	Show PIM-SMv6 interface information.
Qtech# **show ipv6 pim sparse-mode local-members** [*interface-type interface-number*]	Show local MLD information of PIM-SMv6 interface.
Qtech# **show ipv6 pim sparse-mode mroute** {*ipv6_group_address* | *ipv6_source_address* | }	Show PIM-SMv6 multicast routing information.
Qtech# **show ipv6 pim sparse-mode neighbor** [**detail**]	Show PIM-SMv6 neighbors.
Qtech# **show ipv6 pim sparse-mode nexthop**	Show PIM-SMv6 next hop information from NSM.
Qtech# **show ipv6 pim sparse-mode rp-hash** *ipv6_group-address*	Show the RP information of the specific multicast group address.
Qtech# **show ipv6 pim sparse-mode rp mapping**	Show all RPs and the groups they serve.
Qtech# **show ipv6 pim sparse-mode track**	Show the number of received and sent PIMv6 packets.

7.5.2 Deleting Internal PIM-SMv6 Messages

The following commands delete internal PIM-SMv6 messages.

Command	Function
Qtech# **clear ipv6 mroute** {* | *group_address* [*source_address*] }	Clear multicast routing entries.
Qtech# **clear ipv6 mroute statistics** {* | *group_address* [*source_address*] }	Clear multicast routing entry statistics.
Qtech# **clear ipv6 pim sparse-mode bsr rp-set ***	Clear RP-SET.
Qtech# **clear ipv6 pim sparse-mode track**	Reset the beginning time of statistics and reset PIMv6 packet counter.

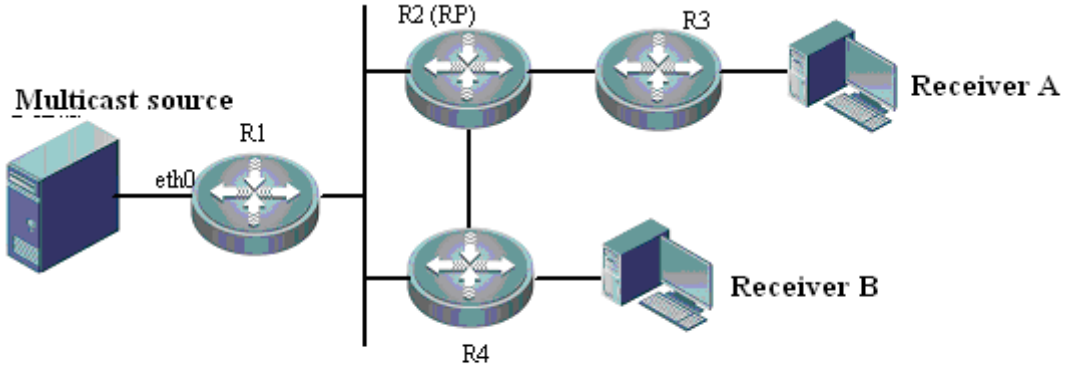
For details, refer to *PIM-SMv6 Command Reference*.

7.6 PIM-SMv6 Configuration Example

7.6.1 Configuration Requirements

Figure 3 illustrates network topology. R1 and the multicast source are located in one network. R2 is set to be RP. R3 and receiver A are located in the same network. R4 and receiver B are in the same network. Assume that devices are connected properly, IPv6 is enabled on each interface and IPv6 unicast is enabled on every device.

Figure 19 PIM-SMv6 topology



7.6.2 Configuration steps

Step1: Enable multicast routing

Enable IPv6 multicast routing on R1. The configuration is similar on R2, R3 and R4.

```
Qtech# configure terminal
Qtech(config)# ipv6 multicast-routing
```

Step 2: Enable PIM-SMv6 on the interface

Enable PIM-SMv6 on R1's eth0. This configuration is similar on the interfaces of R1, R2, R3 and R4.

```
Qtech(config)# interface eth 0
Qtech(config-if)# ipv6 pim sparse-mode
Qtech(config-if)# end
```

Step 3: Configure candidate BSR and candidate RP.

Set R2's loopback1 to be C-BSR and C-RP

```
Qtech(config)# interface loopback 1
Qtech(config-if)# ipv6 address 2008:1::1/64
Qtech(config-if)# ipv6 pim sparse-mode
Qtech(config-if)# exit
Qtech(config)# ipv6 pim bsr-candidate loopback 1
Qtech(config)# ipv6 pim rp-candidate loopback 1
```

After the receiver joins the multicast group and the multicast source sends multicast packets, you can use show commands to monitor operation status.

Note MLD is automatically enabled on each interface while PIM-SMv6 is enabled.

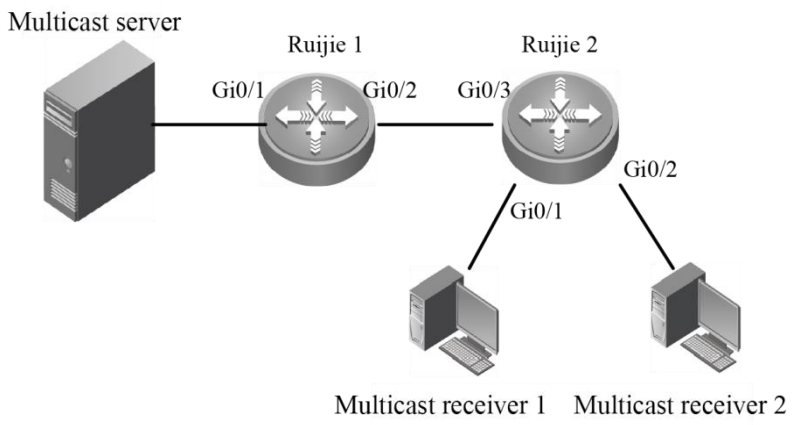
7.7 Configuration Examples of SSM Based on PIM

7.7.1 Networking Requirements

1. Network communication must be available at Layer 3. For example, run OSPFv3.
2. Multicast receivers obtain information about the multicast source through a channel depending on application software and deployment of network administrators on hosts.
3. Run PIM-SMv6 on the network.

7.7.2 Networking Topology

Figure 20 SSM configuration example



Basic configuration of devices and interfaces in Figure 4:

Device name	Interface	Basic configuration
Multicast server		Provides multicast data (2222::2, FF33::1111)
Qtech 1	Gi 0/1	IPv6 address is 2222::1/64
Qtech 1	Gi 0/2	IPv6 address is 3333::1/64
Qtech 2	Gi 0/3	IPv6 address is 3333::2/64
Qtech 2	Gi 0/1	IPv6 address is 4444::1/64
Qtech 2	Gi 0/2	IPv6 address is 5555::1/64

7.7.3 Configuration Procedure

1. Basic configuration of interfaces on Qtech 1 and Qtech 2:

Configure interfaces on Qtech 1 and Qtech 2 according to the IPv6 addresses specified in the topology. The procedure is omitted.

2. Enable network interconnection on Layer 3:

Enable OSPFv3 on all interfaces of Qtech 1 and Qtech 2 and ensure devices are interconnected at Layer 3. The procedure is omitted.

3. Configure multicast on Qtech 1 and Qtech 2:

Enable the multicast routing on Qtech 1 and Qtech 2. Then, enable PIM-SMv6 on interfaces of Qtech 1 and Qtech 2. Lastly, enable MLDv2 on the interface connecting to multicast receivers of Qtech 2. The following description uses Qtech 2 as an example.

Enable multicast routing on Qtech 2:
```
Qtech(conf)#ipv6 multicast-routing
```

Enable PIM-SMv6 on all interfaces of Qtech 2, such as interface Gi 0/1:
```
Qtech(conf-GigabitEthernet0/1)#ipv6 pim sparse-mode
```

Enable MLDv2 on the interface connected to the multicast receiver of Qtech 2, such as interface Gi 0/1:
```
Qtech(conf-GigabitEthernet0/1)#ipv6 mld version 2
```

The configuration steps of Qtech 1 are similar to that of Qtech 2. The procedure for configuring Qtech 1 is omitted.

4. Enable SSM on Qtech 1 and Qtech 2:

Enable SSM on Qtech 2 and use the default SSM group range: FF3x::/32
```
Qtech(conf)#ipv6 pim ssm default
```

Enable SSM on Qtech 1 and use the default SSM group range: FF3x::/32
```
Qtech(conf)#ipv6 pim ssm default
```

7.7.4 Verifying the Configuration

1. Multicast receivers 1 and 2 request the multicast service (2222::2, FF33::1111) by sending MLD (2222::2, FF33::1111) join.

The protocol table of (2222::2, FF33::1111) will be created on Qtech 2. You can view the protocol table by running the **show ipv6 pim sparse-mode mroute** command.

Qtech#show ipv6 pim sparse-mode mroute

IPv6 Multicast Routing Table

(*,*,RP) Entries: 0

(*,G) Entries: 0

(S,G) Entries: 1

(S,G,rpt) Entries: 1

FCR Entries: 0

REG Entries: 0

(2222::2, FF33::1111)

RPF nbr: 3333::1

RPF idx: GigabitEthernet0/3

SPT bit: 0

Upstream State: JOINED

kat expires in 175 seconds

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Local

0.i i . . . . . .

1 . . . . . .

Joined

0 . . . . . .

1 . . . . . .

Asserted

0 . . . . . .

1 . . . . . .

Outgoing

0.o o . . . . . .

1 . . . . . .

The output shows that the entry of (2222::2, FF33::1111) is created on Qtech 2 and two multicast receivers exist.

The multicast server sends the data flow (2222::2, FF33::1111):

The multicast forwarding table of (2222::2, FF33::1111) will be created on Qtech 2. You can view the multicast forwarding table with the **show ipv6 mroute** command.

Qtech#show ipv6 mroute

IPv6 Multicast Routing Table

Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed, R - RPT, S - SPT

Timers: Uptime/Stat Expiry

Interface State: Interface (TTL)

(2222::2, FF33::1111), uptime 00:19:31, stat expires 00:02:53

Owner PIMSMV6, Flags: TFSs

Incoming interface: GigabitEthernet 0/3

Outgoing interface list:

GigabitEthernet 0/1(1)

GigabitEthernet 0/2(1)

The output indicates that the multicast forwarding table of (2222::2, FF33::1111) is established on Qtech 2 and the multicast egresses are Gi 0/1 and Gi 0/2, which are connected with multicast receiver 1 and multicast receiver 2 respectively. In addition, the Flags of this forwarding table is tagged with ‘s’, indicating that this forwarding table uses the SSM model.

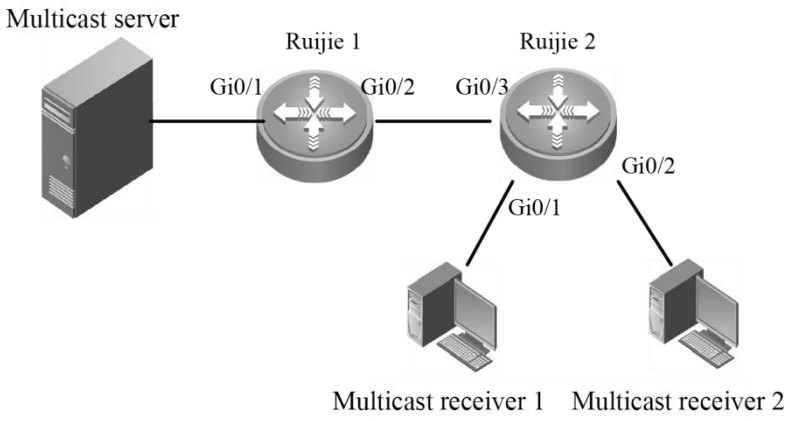
7.8 Configuration Examples of Embedded RP

7.8.1 Networking Requirements

1. Network communication must be available at Layer 3. For example, run OSPFv3.
2. Run PIM-SMv6 on the network.

7.8.2 Networking Topology

Figure 21 Configuring embedded RP



Basic configuration of devices and interfaces in Figure 5:

Device name	Interface	Basic configuration
Multicast server		Provides multicast source (2222::2, FF7E:0540:6666::1)
Qtech 1	Gi0/1	IPv6 address is 2222::1/64
Qtech 1	Gi0/2	IPv6 address is 3333::1/64
Qtech 2	Gi0/3	IPv6 address is 3333::2/64
Qtech 2	Gi0/1	IPv6 address is 4444::1/64
Qtech 2	Gi0/2	IPv6 address is 5555::1/64
Qtech 2	Loopback0	IPv6 address is 6666::5/64

7.8.3 Configuration Procedure

1. Basic configuration of interfaces on Qtech 1 and Qtech 2:

Configure interfaces on Qtech 1 and Qtech 2 according to the IPv6 addresses specified in the topology. The procedure is omitted.

2. Configure the networks to interoperate on layer 3:

Enable OSPFv3 on all interfaces of Qtech 1 and Qtech 2 and ensure that networks are interoperable on layer 3. The procedure is omitted.

3. Configure the multicast on Qtech 1 and Qtech 2:

Enable multicast routing on Qtech 1 and Qtech 2, and then enable PIM-SMv6 on all interfaces on Qtech 1 and Qtech 2, and configure the static RP on Qtech 2 to enable it acting as the embedded RP.

Enable multicast routing on Qtech 2:

```
Qtech(conf)# ipv6 multicast-routing
```

Enable PIM-SMv6 on all interfaces of Qtech 2, with Gi 0/1 as an example:

```
Qtech(conf-GigabitEthernet0/1)# ipv6 pim sparse-mode
```

Configure static RP 6666::5 on Qtech 2:

```
Qtech(conf)# ipv6 pim rp-address 6666::5
```

The configuration steps of Qtech 1 are similar with that of Qtech 2; therefore, the procedure for configuring Qtech 1 is omitted.

4. Enable embedded RP on Qtech 1 and Qtech 2:

By default, embedded RP is enabled. Enable it if it is disabled.

Enable embedded RP on Qtech 2:
```
Qtech(conf)# ipv6 pim rp embedded
```

Enable embedded RP on Qtech 1:
```
Qtech(conf)# ipv6 pim rp embedded
```

7.8.4 Verifying the Configuration

1. Multicast receivers 1 and 2 send MLD (*,FF7E:0540:6666::1) join:

The protocol table of (*,FF7E:0540:6666::1) is created on Qtech 2. You can view the protocol table by running the show ipv6 pim sparse-mode mroute command. Meanwhile, the group RP mapping corresponding to the IPv6 multicast address FF7E:0540:6666::1 of the embedded RP will be saved on Qtech 2. You can will the group RP mapping by running the show ipv6 pim sparse-mode rp mapping command.

Qtech#sh ipv6 pim sparse-mode rp mapping

PIM Group-to-RP Mappings

Group(s): ff7e:540:6666::/96

RP: 6666::5

Info source: Embedded RP

Uptime: 00:00:12, expires: 00:00:48

The preceding output indicates that on Qtech 2, the group RP mapping of the IPv6 multicast address FF7E:0540:6666::/96 of the embedded RP is created automatically and the RP address is 6666::5.

Qtech#show ipv pim sparse-mode mroute

IPv6 Multicast Routing Table

(*,*,RP) Entries: 0

(*,G) Entries: 1

(S,G) Entries: 0

(S,G,rpt) Entries: 0

FCR Entries: 0

REG Entries: 0

(*, ff7e:540:6666::1)

RP: 6666::5

RPF nbr: ::

RPF idx: None

Upstream State: JOINED

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Local

0.i i . . . . . .

1 . . . . . .

Joined

0 . . . . . .

1 . . . . . .

Asserted

0 . . . . . .

1 . . . . . .

FCR:

The preceding display indicates that the protocol table of (*,ff7e:540:6666::1) has been created and two multicast receivers exist.

2. The multicast server sends the multicast data (2222::2, FF7E:0540:6666::1):

The multicast forwarding table of (2222::2, FF7E:0540:6666::1) will be created on Qtech 2. You can view the forwarding table by running the show ipv6 mroute command.

Qtech#show ipv6 mroute

IPv6 Multicast Routing Table

Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed, R - RPT, S - SPT

Timers: Uptime/Stat Expiry

Interface State: Interface (TTL)

(2222::2, FF7E:0540:6666::1), uptime 00:19:31, stat expires 00:02:53

Owner PIMSMV6, Flags: TFS

Incoming interface: GigabitEthernet 0/3

Outgoing interface list:

GigabitEthernet 0/1(1)

GigabitEthernet 0/2(1)

8 CONFIGURING RMEF

8.1 RMEF Overview

IP multicast realizes efficient point-to-multipoint data transmission over IP networks. Since IP multicast can effectively save network bandwidth and reduce network load, it is widely applied in real-time data transmission, multimedia conferencing, data copying, gaming and simulation.

Qtech Multicast Express Forward (RMEF) maintains a mirror image of control-plane multicast routing table at the express forwarding data plane, so that forwarding of multicast packets can be done at the express forwarding data plane, thus improving the performance and efficiency of multicast forwarding.

8.2 Configuring REMF

8.2.1 Enabling/Disabling Multicast Express Forwarding On the Interface

Use the following commands to enable/disable multicast express forwarding on the interface in interface configuration mode.

Command	Function
Qtech(config-if)# **ip ref**	Enables multicast express forwarding on the interface.
Qtech(config-if)# **no ip ref**	Disables multicast express forwarding on the interface.

By default, multicast express forwarding is enabled on the interface.

8.2.2 Displaying RMEF Configuration and Status

Use the following commands to show RMEF configurations and statistics.

Command	Function
show ip ref mcast route [*ip address ip address*]	Shows the multicast express forwarding table. When no parameters are set, this command shows all table entries. When parameters are set (the first parameter is the IP address of the multicast source and the second one is the IP address of the multicast group), this command shows the table entries that match the two IP addresses.
show ip ref mcast info	Shows RMEF information, including the RMEF status, whether RMEF is enabled, number of express forwarding tables, and data packet uploading rate of RMEF in the case of no forwarding table or in data packet uploading process.
show ip ref mcast statistics interface *interface-type interface-number*	Shows statistics on the packets forwarded on the multicast express forwarding interface.
show ip ref mcast statistics mfc	Shows statistics on data packets forwarded on all multicast express forwarding entries.

8.3 Configuration Examples

The following examples show typical configurations on a router:

```
RSR20-04# config
```

! Enable multicast forwarding.

```
RSR20-04(config)# ip multicast-routing
```

! Enter the related interface and enable the multicast protocol.

```
RSR20-04(config)# interface fastEthernet 0/0
RSR20-04(config-if)# ip pim dense-mode
```

! Enable express forwarding on the interface.

```
RSR20-04(config-if)# ip ref
```

The following example shows how to debug multicast express forwarding:

! Display all multicast express forwarding tables.

```
Qtech (config-if)# show ip ref mcast route
IP Multicast EF Routing Table
Interface State: Interface (Interface Index)
(30.1.1.2, 224.1.1.2)
 In_interface: GigabitEthernet 0/1.100(8)
 Hit: Yes
 To_cpu: No
 Oif_list: GigabitEthernet 0/2.100 (12)
```

! Display multicast express forwarding information.

```
Qtech (config-if)# show ip ref mcast info
---------------------------------
IP RMEF is open
total RMEF MFC NUM = 1
to_cpu ratelimit PPS in one second = 10
no_mfc ratelimit PPS in one second = 10
---------------------------------
```

! Display multicast express forwarding statistics.

```
Qtech (config-if)# show ip ref mcast statistics mfc
(30.1.1.2, 224.1.1.2)
 In_interface: GigabitEthernet 0/1.100(8)
 Match_PKTNUM: 17058555
 Match_PKTBYTES: 1091747520
 WRONG_IN_IF_PKTNUM: 0
 TO_CPU_RESERVE_PACKET: 0
 TO_CPU_DROP_PACKET: 0
 Oif_list: GigabitEthernet 0/2.100(12)
```

9 CONFIGURING MSDP

9.1 Overview

In PIM-SM network, network domains with same group-RP mapping information form a special multicast network domain, which is called PIM-SM domain.

1) The need for cross-domain multicasting

In the basic PIM-SM mode, a multicast source registers only with the RP in the local PIM-SM domain, and the multicast source information of a PIM-SM domain is isolated from that of another PIM-SM domain. As a result, the RP is aware of the source information only within the local domain and a multicast distribution tree is built only within the local domain; since RP is not aware of the multicast source information of other PIM-SM domains, it cannot join the multicast sources in other domains and build the multicast distribution tree. Therefore, cross-domain multicast application becomes impossible.

Given the feature of PIM-SM domain and the aforementioned problem, a mechanism is needed to allow RPs of different PIM-SM domains to share their multicast source information, so that the RP can learn the multicast source information of other PIM-SM domains. In this way, the local RP will be able to join multicast sources in other domains and the cross-domain multicast distribution tree can be built to carry out cross-domain multicast applications.

2) The need for Anycast-RP deployment

In a single PIM-SM domain, RP is the root node of share tree. RP is a critical device in PIM-SM domain, and it may fail due to excessive loads. Therefore, a RP load sharing mechanism will be needed to ease the burden of a single RP, and this mechanism is called Anycast-RP. In practice, we configure the same IP address for multiple RPs in a single PIM-SM domain; the multicast receiver joins the nearest RP according to the route, while the multicast source will register the multicast source information with the nearest RP according to the route, namely each RP will share certain PIM-SM activities.

During the deployment of Anycast-RP, we can encounter one problem: since the multicast source only registers the multicast source information with the nearest RP according to the route, other RPs acting as Anycast-RP will be unaware of the multicast source information and thus will not join the multicast source, leading to the failure of Anycast-RP deployment.

Given the aforementioned problem, a mechanism is needed to allow RPs acting as Anycast-RP to share their multicast source information, so that each RP acting as Anycast-RP can learn the multicast source information of the local domain. In this way, each RP will be able to join multicast sources and build the multicast distribution tree, allowing successful Anycast-RP deployment.

3) MSDP

Given the needs for cross-domain multicasting and Anycast-RP deployment, MSDP (Multicast Source Discovery Protocol) is introduced. With MSDP peer relationships established between different RPs, the RPs are interconnected with one another. If different RPs are distributed in different PIM-SM domains, then RP can discover the multicast source information in other PIM-SM domains, allowing cross-domain multicasting; if different RPs are distributed in the same PIM-SM domain, then RPs can share the multicast source information in the domain, allowing Anycast-RP deployment.

9.1.1 Basic Concepts

9.1.1.1 MSDP Peer

By establishing a special MSDP relationship between two layer-3 devices running MSDP protocol, these two layer-3 devices can share the multicast source information with each other. Such a special MSDP relationship is called MSDP peer relationship. Two layer-3 devices with MSDP peer relationship are called a pair of MSDP peers.

MSDP peer relationship is based on TCP connection and utilizes TCP 639 port. As long as the layer-3 route can reach, the MSDP peer relationship can be established between any two layer-3 devices.

Note The layer-3 device as mentioned herein refers to the generic router or layer-3 switch.

9.1.1.2 *Peer-RPF Check of SA messages*

MSDP device acting as RP will encapsulate the multicast source information registered with RP in SA messages, and forward SA messages to MSDP peers, which will then be able to learn the multicast source information.

Since the MSDP peers of the local MSDP device may form a loop. To avoid looping, MSDP protocol has introduced Peer-RPF check of SA messages.

The basic principle of Peer-RPF check of SA messages: According to the source address of SA messages, MSDP device will select the only MSDP peer for each SA message, and this MSDP peer is called Peer-RPF neighbor. When the local MSDP device receives SA message from Peer-RPF neighbor, it will accept this SA message and forward to other MSDP peers. Otherwise, it will drop this SA message.

9.1.1.3 *Peer-RPF Forwarding of SA Messages*

Once a MSDP device receives a SA message from the Peer-RPF neighbor, it will need to forward this SA message to other MSDP peers. This kind of behavior is called: Peer-RPF forwarding.

9.1.1.4 *Mesh Group*

An MSDP mesh group refers to a group of MSDP peers that have MSDP peering relationships among one another, namely any two layer-3 devices form a pair of MSDP peers.

When a SA messages is accepted by a MSDP peers in the mesh group, this SA message will not be forwarded to other MSDP peers in the same mesh group. This has reduced the flooding of SA messages and simplified Peer-RPF forwarding of SA messages.

9.1.2 Working Principle

1. MSDP peer

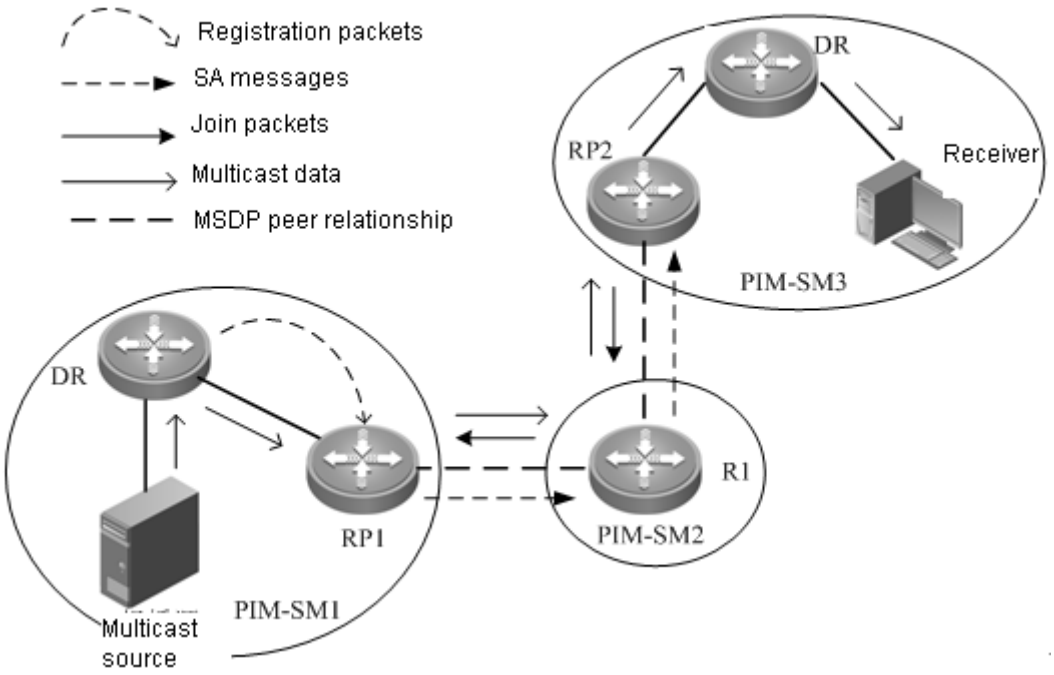
By configuring one or more pairs of MSDP peers in the multicast network, different RPs will be interconnected and send the multicast source information learned to MSDP peers, so as to advertise the multicast source information to other RPs.

If the aforementioned RPs are in different PIM-SM domains, then these PIM-SM domains can be interconnected. If the aforementioned RPs are in the same PIM-SM domain, then the multicast source information of the same PIM-SM domain will be shared among RPs.

2. Create MSDP peer

You can create MSDP peer on any layer-3 device. Devices that assume different multicast roles function differently. These peers can be classified into MSDP peers created on RP and MSDP peers created on non-RP.

Fig 1 MSDP Peers



(1) Creating MSDP peers on RPs

Source-side MSDP peer

According to PIM-SM protocol, the multicast source information is resisted with RP via registration message, indicating that the RP nearest to the multicast source (source-side RP) is aware of the multicast source information. The source-side RP creates SA messages according to the multicast source information registered and sends the messages to other RPs. Therefore, the MSDP peer must be created on the source-side RP; otherwise it will not be able to advertise the multicast source information.

As shown in Figure 1, the multicast source is in PIM-SM1, while RP1 is the RP nearest to the multicast source. The multicast source information is registered with RP1. Therefore, the peer must be created on RP1, so that RP1 can advertise the multicast source information. For example, the MSDP peer relationship is created between RP1 and R1 in PIM-SM2, so that the multicast source information can be sent to R1.

Receiver-side MSDP peer

If the receiver-side RP is unable to learn the multicast source information, it will not be able to join the multicast source and build the multicast distribution tree. Therefore, the MSDP peer must be created on the RP nearest to the receiver and being unaware of the multicast source information.

As shown in Figure 1, the receiver is in PIM-SM3, while RP2 is the nearest RP. Therefore, the peer must be created on RP2 in order to learn multicast source information from other PIM-SM domains. For example: MSDP peer relationship is established between RP2 and R1 in PIM-SM2; R1 sends the multicast source information learned from RP1 to RP2, so that RP2 can learn the same multicast source information. Then, RP2 joins the multicast source and build the multicast distribution tree from PIM-SM1 to PIM-SM3.

Intermediate MSDP Peer

The RP which is neither nearest to the multicast source nor nearest to the multicast receiver, functioning as a relay of multicast source information.

(2) MSDP created on non-RP: It will only forward the SA messages. As indicated in Figure 1, R1 in PIM-SM2 only forwards the SA messages to other MSDP peers. For example: R1 forwards SA messages to RP2.

9.1.3 Protocol Specification

Please refer to RFC3618.

9.2 Default Configurations

The following table describes the default configurations of MSDP.

Function	Default setting
Create MSDP peer	No MSDP peer is created.
Specify the default MSDP peer	No default MSDP peer is specified.
Configure MSDP peer description	No MSDP peer description is configured.
Configure SA request filtering	SA request messages are not filtered.
Configure mesh group	No MSDP peer will join the mesh group.
Change the originator address of SA messages	The originator address of SA messages is the address of RP.
Configure MD5 key for MSDP peer	MD5 encryption is not applied on the TCP connections between MSDP peers.
Configure redistribution filtering	All multicast source information registered with RP will be propagated to the MSDP peers.
Configure SA filtering	All SA messages from MSDP peers will be accepted; Each SA message received will be forwarded to the MSDP peer.
Configure the upper threshold of SA learning	No upper threshold for the SA information learned from MSDP peers.
Deactivate MSDP peer	MSDP peer is not deactivated.
Configure TTL upper threshold	TTL upper threshold is not configured. If the SA messages received carry the multicast data, then the SA messages forwarded to other MSDP peers will still carry the multicast data.

9.3 Configuring MSDP Peer

(Required) Creating MSDP peer

(Optional) Configuring MSDP peer description

(Optional) Configuring MD5 encryption for MSDP peer

9.3.1 Creating MSDP Peer

To enable MSDP, you need to configure MSDP peer. Execute the following command in the global configuration mode.

Command	Function
Qtech(config)# **ip msdp peer** *peer-address* **connect-source** *interface-type interface-number*	Create MSDP peer The main address of the interface specified by the key word of "connect-source" will be used as the source address of MSDP TCP connection.

9.3.2 Configuring MSDP Peer Description

Each MSDP peer is an IP address with poor readability. In order to better distinguish different MSDP peers, you can configure text descriptions for MSDP peers.

To configure MSDP peer description, execute the following command in the global configuration mode:

Command	Function
Qtech(config)# **ip msdp description** *peer-address text*	Configure text description for MSDP peer.

9.3.3 Configuring MD5 Encryption for MSDP Peer

The connection between MSDP peers is based on TCP connection. By default, the TCP connection will not use MD5 encryption, which is to say, TCP connection will not be subject to identity authentication. Under circumstances with high security needs, the identity authentication of TCP connections will be needed. To enable the identity authentication of TCP connections, execute the following commands in the configuration mode.

Command	Function
Qtech(config)# **ip msdp password peer** *peer-address [encryption-type] string*	Enable MD5 encryption of the TCP connections between MSDP peers.

9.4 Controlling the Propagation of Multicast Source Information

(Optional) Redistribution filtering

(Optional) Filtering SA request messages

9.4.1 Redistribution Filtering

When the multicast source registers with RP, RP will learn the multicast source information and generate SA messages. By default, all registered multicast source information will be distributed by RP.

To control which registered sources can be distributed, execute the following commands in the global configuration mode.

Command	Function
Qtech(config)# **ip msdp redistribute** [**list** *access-list-name*] [**route-map** *route-map-name*]	Only multicast source information passing the filtering rule will be propagated to the MSDP peers. 1) If the "list" or "route-map" is not specified, then no multicast source information (S, G) will be propagated. 2) If the "list" or "route-map" is specified, then only the multicast source information (S, G) meeting the access list or route map rule will be propagated to MSDP peers. 3) If the "list" and "route-map" is specified at the same time, then only the multicast source information (S, G) meeting both the access list and route map rule will be propagated to MSDP peers.

9.4.2 Filtering SA Request Messages

MSDP peer will request the multicast source information about a certain multicast group from local MSDP device by sending SA request messages. If the local device has the multicast source information about a certain multicast group, then the local MSDP device will need to send SA reply message to this MSDP peer, so as to inform the MSDP peer of the relevant multicast source information.

By default, the MSDP device will accept all SA request messages from the MSDP peer and give replies. However, you can configure to ignore all SA request messages, or only accept the SA request messages from certain multicast groups and ignore those from other multicast groups through the standard ACL.

To filter SA request messages, execute the following command in the global configuration mode:

Command	Function
Qtech(config)# **ip msdp filter-sa-request** *peer-address* [**list** *access-list-name*]	Filter SA request messages from the specified MSDP peer, and only rely to SA request messages passing the filtering rule. 1) If the "list" is not specified, then all SA request messages from this MSDP peer will be rejected. 2) If the "list" is specified, then only SA request messages meeting ACL will be accepted.

9.5 Controlling the Forwarding of Multicast Source Information

(Optional) Using MSDP filter

(Optional) Using TTL to limit the multicast data carried in SA messages

9.5.1 Using MSDP Filter

After accepting a SA message, the MSDP device will need to forward the SA message to other MSDP peers. By default, MSDP device will forward the received SA messages to other MSDP peers without any change. However, in certain deployment environment, we need to limit the multicast source information that can be learned by other MSDP peers. In such a case, we will need to control which multicast source information can be forwarded to other MSDP peers.

To control which multicast source information that can be forwarded to MSDP peers, execute the following commands in the global configuration mode:

Mode	Function
Qtech(config)# **ip msdp sa-filter out** *peer-address* [**list** *access-list-name*] [**route-map** *map-name*] [**rp-list** *rp-access-list-name*] [**rp-route-map** *rp-route-map-name*]	Configure SA message filtering rule for the specified MSDP peer. Only the multicast source information passing the filtering rule will be forwarded to MSDP peers. 1) If no key word is specified, SA messages will not be sent to the specified MSDP peer. 2) If one or more key words are specified, only multicast source information (S,G) meeting all rules corresponding to these key words will be forwarded the MSDP peer.

9.5.2 Using TTL to Limit the Multicast Data Carried in SA Messages

The SA messages can carry multicast data. In certain deployment environment, for efficiency related considerations, the TTL value of multicast data may be required to reach a certain threshold. If the TTL value of multicast data is

less than the threshold, then the multicast data will be separated from SA messages and discarded. For example: By limiting the TTL value of multicast data packets to 8, the TTL of multicast data packets sent to MSDP peers shall be greater than or equal to 8.

To configure TTL threshold, execute the following commands in the global configuration mode.

Command	Function
Qtech(config)# **ip msdp ttl-threshold** *peer-address ttl-value*	Limit the TTL value of multicast data carried in SA messages. When a SA message is received from one MSDP peer, if this SA message carries multicast data with TTL value less than the TTL threshold configured, then the multicast data will be separated from the SA message, and the SA message without multicast data will then be forwarded to other MSDP peers.

9.6 Controlling the Acceptance of Multicast Source Information

By default, MSDP device will accept all SA messages from the Peer-RPF neighbor. However, we can configure filtering rule to control which SA messages will be accepted by the local device.

There are several ways to control the multicast source information from MSDP peers:

- Filter all SA messages from a certain MSDP peer;
- Filter by using the specified access control list;
- Filter based on route map.

To control the multicast source information that can be accepted by the local device, apply the filter in the global configuration mode.

Command	Function
Qtech(config)# **ip msdp sa-filter in** *peer-address* [**list** *access-list-name*] [**route-map** *route-map-name*] [**rp-list** *rp-access-list-name*] [**rp-route-map** *rp-route-map-name*]	Configure inbound SA message filtering rule for the specified MSDP peer. Only the multicast source information passing the filtering rule will be accepted. 1) If no key word is specified, SA messages from the specified MSDP peer will not rejected; 2) If one or more key words are specified, only multicast source information (S,G) meeting all rules corresponding to these key words will be accepted.

9.7 Configuring Default Peer

When the MSDP device receives an SA message, MSDP device will need to determine whether this SA message is from Peer-RPF neighbor; if not, the MSDP device will discard this SA message. To avoid the absence of Peer-RPF neighbor, you can configure the default peer, which is also a default Peer-RPF neighbor.

In the global configuration mode, execute the following command to specify the default MSDP peer:

Command	Function
Qtech(config)# **ip msdp default-peer** *peer-address* [**prefix-list** *prefix-list-name*]	Specify the default MSDP peer. If "prefix-list" is not specified, then all SA messages from this default MSDP peer will be accepted; otherwise, only SA messages meeting prefix-list rule will be accepted.

9.8 Configuring MSDP Mesh Group

MSDP mesh group consists of a group of fully connected MSDP peers. The SA messages received by one MSDP peer in the mesh group will not be forwarded to other MSDP peers in the same mesh group. Therefore, it can reduce the flooding of SA messages and simplify Peer-RPF computation.

To create mesh group, execute the following command in the global configuration mode.

Command	Function
Qtech(config)# **ip msdp mesh-group** *mesh-name peer-address*	Configure a MSDP mesh group and specify one MSDP peer to join this mesh group.

9.9 Deactivating MSDP Peer

If you don't want to activate a MSDP peer but deleting this MSDP peer will lose the relevant configurations, you can then deactivate this MSDP peer. After the MSDP peer is deactivated, only the TCP connection is terminated, and the configurations of this MSDP peer won't be deleted.

In the global configuration mode, execute the following command to deactivate the MSDP peer:

Command	Function
Qtech(config)# **ip msdp shutdown** *peer-address*	Deactivate MSDP peer.

9.10 Configuring the Address of Originator other than RP

By default, the originator address of SA messages uses the address of RP in PIM-SM domain.

During the deployment of Anycast-RP, MSDP peer relationship will be established between RPs with the same IP address. When the multicast source registers with a RP nearest to the route, this RP will send SA messages to other RPs. By default, the originator address of such SA messages will be the address of RP. Upon receipt of such SA messages, other RPs will find out that the originator address is the local address, and the computation of Peer-RPF neighbor may fail. There are three solutions to the abovementioned problem:

- Configure mesh group
- Configure default MSDP peer
- Change the originator address of SA messages

To change the originator address of SA messages, execute the following command in the global configuration mode.

Command	Function
Qtech(config)# **ip msdp originator-id** *interface-type interface-number*	Configure the originator address of SA messages to the main IP address of the specified interface.

9.11 Monitoring and Maintaining MSDP

Use the following commands to monitor MSDP.

Command	Function
Qtech# **debug ip msdp peer** [*peer-address*]	Debug a specific peer or all peers, including: peer connection establishment/shutdown, MSDP protocol packets reception/forwarding, etc.
Qtech# **show ip msdp count** [*as-number*]	Display the number of sources and groups indicated in the SA messages generated by each autonomous system.

Qtech# **show ip msdp mesh-group**	Display the information of mesh group.
Qtech# **show ip msdp peer** [*peer-address*]	Display the detailed information of a specific or all MSDP peers.
Qtech# **show ip msdp rpf-peer** *ip-address*	Display the information of RPF-Peer with the specified address.
Qtech# **show ip msdp sa-cache** [*group-address*|*source–address*] [*group-address*|*source–address*] [*as-number*]	Display the status of multicast source (S,G) learned from MSDP peer.
Qtech**# show ip msdp sa-originated**	Display the SA originated multicast source (S, G).
Qtech# **show ip msdp summary**	Display the brief information of MSDP peers.
Qtech# **clear ip msdp peer** *peer-address*	Reset the TCP connection with a specific MSDP peer; reset all MSDP message counters.
Qtech# **clear ip msdp statistics** [*peer-address*]	Clear the statistics of a specific or all MSDP peers.
Qtech# **clear ip msdp sa-cache** [*group-address*]	Clear all SA cache entries or those of a specific group.

9.12 Typical MSDP Configuration Example

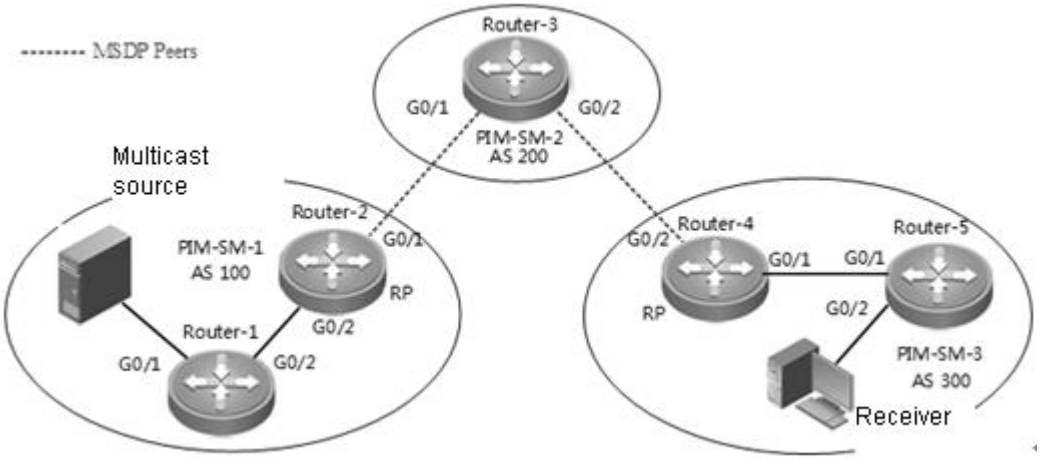
9.12.1 Cross-domain Multicasting

9.12.1.1 Networking Requirements

1) Three ISPs maintains autonomous system AS 100, AS 200 and AS 300 respectively. OSPF is adopted for the internal communication within each AS, while BGP is adopted for exchanging routing information between the autonomous systems.
2) PIM-SM-1 belongs to AS 100, PIM-SM-2 belongs to AS 200, and PIM-SM-3 belongs to AS 300. Each PIM-SM domain has 0 or 1 multicast source and 0 or 1 receiver.
3) Configure the Loopback1 interface of Router-2 to C-BSR and C-RP of PIM-SM-1, and configure the Loopback1 interface of Router-4 to C-BSR and C-RP of PIM-SM-3.
4) Establish MSDP Peer relationship between Router-2 and Router-3 via EBGP, and establish MSDP Peer relationship between Router-3 and Router-4 via EBGP.

***9.12.1.2** Network Topology*

Fig 2 Cross-domain multicasting



Basic configurations of major interfaces in Figure 2.

```
Device        Interface           IP address
Router-1      G0/1                200.200.200.1/24
G0/2                100.100.100.1/24
Router-2      G0/2                100.100.100.2/24
G0/1                1.1.1.1/24
Loopback1           111.111.111.111/32
Loopback0           10.10.10.10/32
Router-3      G0/1                1.1.1.2/24
G0/2                2.2.2.1/24
Loopback0           30.30.30.30/32
Loopback1           40.40.40.40/32
Router-4      G0/2                2.2.2.2/24
G0/1                3.3.3.1/24
Loopback1           222.222.222.222/32
Loopback0           20.20.20.20/32
Router-5      G0/1                3.3.3.2/24
G0/2                4.4.4.1/24
```

9.12.1.3 Configuration Steps

(1) Configure the IP address and unicast routing protocol for the interfaces of respective devices

Configure the IP address and subnet mask of respective interfaces according to Figure 2. Configure OSPF protocol to connect the devices in each domain, and make sure the devices are interconnected at the network layer.

(2) Enable IP multicast routing, and enable PIM-SM on each interface.

On Router-1, enable IP multicast routing, and enable PIM-SM on each interface.

```
Qtech# conf t
Qtech(config)# ip multicast-routing
Qtech(config)# int g 0/1
Qtech(config-if)# ip pim sparse-mode
```

Proceed with the similar configurations on other interfaces of the local device and on the interfaces of other devices.

(3) Configure BSR border

Configure BSR border on G0/1 of Router-2.

```
Qtech(config)# int g 0/1
Qtech(config-if)# ip pim bsr-border
```

Configure BSR border on G0/1 and G0/2 of Router-3 and on G0/2 of Router-4.

(4) Configure C-BSR and C-RP

Configure C-BSR and C-RP on Loopback1 of Router-2.

```
Qtech# conf t
Qtech(config)# int loopback 1
Qtech(config-if)# ip pim sparse-mode
Qtech(config-if)# ip address 111.111.111.111 255.255.255.255
Qtech(config-if)# exit
Qtech(config)# ip pim bsr-candidate loopback 1
Qtech(config)# ip pim rp-candidate loopback 1
```

Configure Lookback1 of Router-4 to C-BSR and C-RP as shown above.

(5) Configure BGP routing protocol between the autonomous systems, and configure mutual route redistribution between BGP and OSPF

We need to configure EBGP Peer between Router-2 and Router-3 and between Router-3 and Router-4, and configure mutual route redistribution between BGP and OSPF. Please refer to BGP configuration guide for the detailed configuration process. You can execute BGP commands to display the route learning conditions between autonomous systems.

(6) Configure MSDP Peer

Configure Router-2 and Router-3 to MSDP peers.

On Router-2, configure the main address of interface Loopback0 to the source address of TCP connection.

```
Qtech(config)# ip msdp peer 30.30.30.30 connect-source loopback 0
```

On Router-3, configure the main address of interface Loopback0 to the source address of TCP connection.

```
Qtech(config)# ip msdp peer 10.10.10.10 connect-source loopback 0
```

Configure Router-3 and Router-4 to MSDP peers.

On Router-4, configure the main address of interface Loopback0 to the source address of TCP connection.

```
Qtech(config)# ip msdp peer 40.40.40.40 connect-source loopback 0
```

On Router-3, configure the main address of interface Loopback1 to the source address of TCP connection.

```
Qtech(config)# ip msdp peer 20.20.20.20 connect-source loopback 1
```

9.12.1.4 Verification

(1) Multicast source information

Multicast source sending multicast data (200.200.200.200, 225.1.1.1).

(2) Display MSDP peer status on Router-4

```
Qtech#sh ip msdp summary
Msdp Peer Status Summary
Peer Address        As            State        Uptime/Downtime     Reset-Count
       SA-Count     Peer-Name
40.40.40.40         200           Up                 00:01:42                  0
                                  1                  ?
```

(3) Display the multicast source information learned by Router-4

```
Qtech#sh ip msdp sa-cache

MSDP Source-Active Cache: 1 entries
  (200.200.200.200,225.1.1.1),RP:111.111.111.111,MBGP/AS 100, 00:00:18/00:01:57, Peer
40.40.40.40
    Learned from peer 40.40.40.40, RPF peer 40.40.40.40,
   SAs received: 1, Encapsulated data received: 1
```

9.12.2 Deploying Anycast-RP

In Anycast-RP, two or more RPs with the same address are configured in the same PIM-SM domain, and MSDP peer relationship is established between these RPs, allowing load sharing and redundant backup between RPs in the domain.

The significance of Anycast-RP is shown below:

- RP with the shortest path: Multicast source registers with the nearest RP and build the SPT; the receiver joins the nearest RP and build the RPT.
- Load sharing between RPs: Each RP only maintains partial source/group information in the PIM-SM domain, allowing the load sharing between RPs.
- Redundant backup between RPs: When one RP fails, the multicast source registering with this RP or the receiver joining this RP will automatically select the nearest RP for registration or joining, allowing redundant backup between RPs.

9.12.2.1 Networking Requirements

In the PIM-SM domain, there are multiple multicast data senders and receivers; the OSPF protocol is running in the PIM-SM domain to allow intercommunication between devices at the network layer.

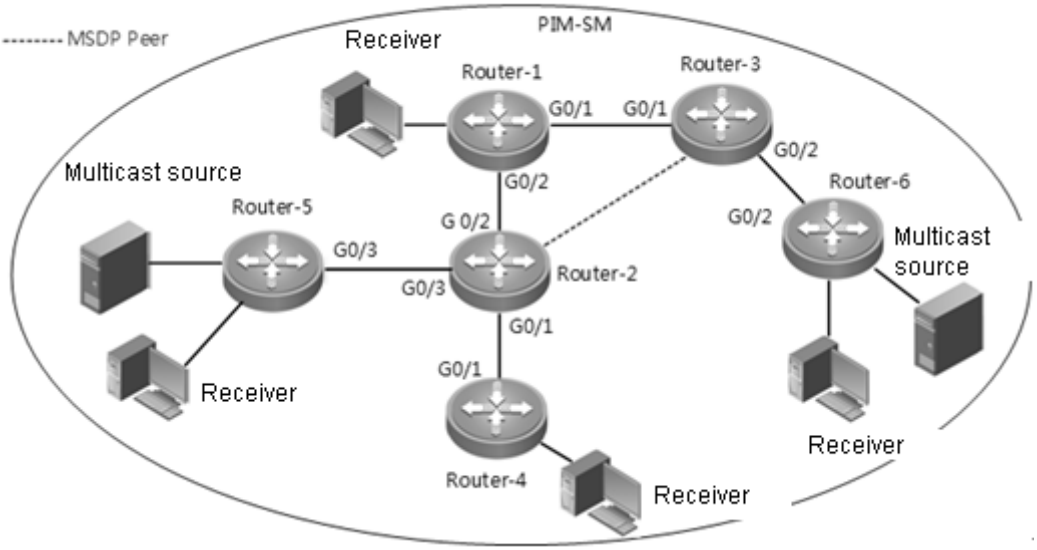
By configuring Anycast-RP in the PIM-SM domain, when a new member joins the multicast group, the receiver-side DR or router acting as DR can join the RP with the nearest unicast distance.

Establish MSDP Peer relationship between Router-2 and Router-3.

Configure C-BSR on Loopback0 of Router-1, and configure C-RP on Loopback1 of Router-2 and Router-3.

9.12.2.2 Network Topology

Fig 3 Anycast-RP deployment



Basic configurations of major interfaces in Figure 3.

```
Device        Interface           IP address
Router-1      G0/1                1.1.1.1/24
                    G0/2                2.2.2.1/24
                    Loopback0           100.100.100.100/32
Router-2      G0/1                3.3.3.1/24
                    G0/2                2.2.2.2/24
                    G0/3                4.4.4.1/24
                    Loopback0           20.20.20.20/32
                    Loopback1           10.10.10.10/32
Router-3      G0/1                1.1.1.2/24
                    G0/2                5.5.5.1/24
                    Loopback0           30.30.30.30/32
                    Loopback1           10.10.10.10/32
Router-4      G0/1                3.3.3.2/24
Router-5      G0/3                4.4.4.2/24
Router-6      G0/2                5.5.5.2/24
```

9.12.2.3 Configuration Steps

(1) Configure the IP address and unicast routing protocol for the interfaces of respective routers

Configure the IP address and subnet mask of respective interfaces according to Figure 3. Configure to interconnect the routers via OSPF protocol.

(2) Enable IP multicast routing, and enable PIM-SM on each interface.

On Router-1, enable IP multicast routing, and enable PIM-SM on each interface.

```
Qtech# conf t
Qtech(config)# ip multicast-routing
Qtech(config)# int g 0/1
Qtech(config-if)# ip pim sparse-mode
```

Proceed with the similar configurations on other interfaces of the local device and on the interfaces of other devices.

(3) Configure C-BSR and C-RP

Configure C-BSR on Router-1

```
Qtech(config)# int loopback 0
Qtech(config-if)# ip pim sparse-mode
Qtech(config-if)# ip address 100.100.100.100 255.255.255.255
Qtech(config-if)# exit
Qtech(config)# ip pim bsr-candidate loopback 0
```

Configure Loopback1 of Router-2 and Loopback1 of Router-3 to the same address, and configure both interfaces to C-RP;

```
Qtech# conf t
Qtech(config)# int loopback 1
Qtech(config-if)# ip pim sparse-mode
Qtech(config-if)# ip address 10.10.10.10 255.255.255.255
Qtech(config-if)# exit
Qtech(config)# ip pim rp-candidate loopback 1
```

(4) Configure MSDP Peer

Configure Router-2 and Router-3 to MSDP peers.

On Router-2, configure the main address of interface Loopback0 to the source address of connection.

```
Qtech(config)#ip msdp peer 30.30.30.30 connect-source loopback 0
```

On Router-3, configure the main address of interface Loopback0 to the source address of connection.

```
Qtech(config)# ip msdp peer 20.20.20.20 connect-source loopback 0
```

9.12.2.4 Verification

(1) Multicast source information

There are two multicast source in Figure 2: (6.6.6.6, 225.1.1.1) and (7.7.7.7, 225.1.1.1). (6.6.6.6, 225.1.1.1) is registered with Router-2, while (7.7.7.7, 225.1.1.1) is registered with Router-3.

(2) Display MSDP peer status on Router-2

```
Qtech#sh ip msdp summary
Msdp Peer Status Summary
Peer Address        As           State        Uptime/Downtime     Reset-Count
       SA-Count     Peer-Name
30.30.30.30         ?            Up                  00:01:42                  0
                                 1                   ?
```

(3) Display the multicast source information learned by Router-2

The following messages show that Router-2 has learned the multicast source information (7.7.7.7, 225.1.1.1) registered with Router-3.

```
Qtech#sh ip msdp sa-cache
```

```
MSDP Source-Active Cache: 1 entries
  (7.7.7.7,225.1.1.1),RP:10.10.10.10,MBGP/AS ?, 00:00:18/00:01:57, Peer 30.30.30.30
    Learned from peer 30.30.30.30, RPF peer 30.30.30.30,
SAs received: 1, Encapsulated data received: 1
```

10 CONFIGURING MULTICAST VPN

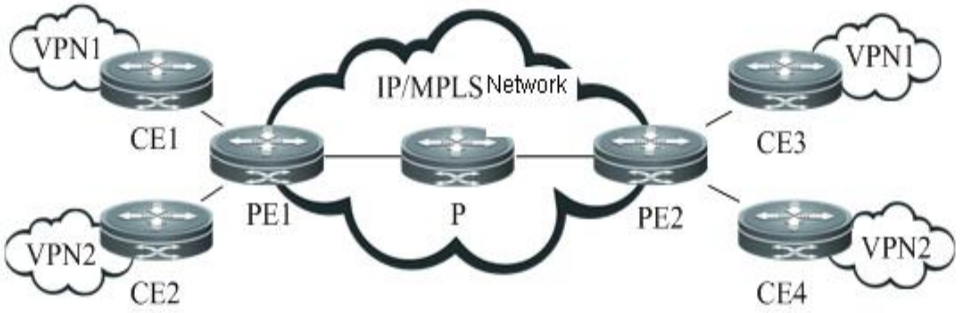
10.1 Introduction to multicast VPN

10.1.1 Overview

IP multicast is being widely applied, and it is being used as the solution to many applications in many sectors. Meanwhile, the application of VPN technology is getting more and more popular in the enterprise networks. BGP/MPLS L3VPN based architecture can be widely found in E-government network, electric power data network and other enterprise networks. By setting different departments into different VPNs, the data of different departments can be well isolated. Meanwhile, the intra-department video conferencing, data sharing and other multicast services will also involve VPN-based data isolation, leading to the increasingly pressing needs for multicast VPN.

Multicast VPN is a BGP/MPLS L3VPN network based technology to transmit multicast data within the VPN network, with network structure similar to that of BGP/MPLS L3VPN, as shown below:

Figure 22 Basic structure of multicast VPN



- CE

CE (Customer Edge Router) is an edge device of customer network, and is logically a customer VPN, with certain interface directly connected to the ISP device. CE can be a host, router or switch, such as CE1, CE2, CE3 and CE4 shown above.

- PE

PE (Provider Edge Router) is an edge device of SP backbone network (it can be a router, ATM switch, frame relay switch and etc), such as PE1 and PE2 as shown in the figure. Logically belonging to the service provider, PE is directly connected with CE, and one PE can be connected with multiple CEs.

- P

P (Provider Router) is the core device within the SP backbone network, such as P1, P2 and P3 as shown in the figure. P is not connected with CE, and is responsible for routing and fast forwarding on the public network.

As shown above, the multicast protocol is run on customer VPN to establish inter-VPN multicast routes and forward multicast traffic for the private network. When PE1 receives the multicast traffic from CE1 and CE2, it will convert the traffic into multicast data format that can be transmitted over the public network, and forward the traffic to PE2 of other VPN sites. PE2 will then de-encapsulate the traffic and forward to the corresponding VPN site.

10.1.2 Multicast Domain

MD (Multicast-Domain) is a scheme to realize multicast VPN, and is achieved by maintaining a multicast domain for each VPN in the service provider network. MD scheme will create a MDT (Multicast Distribution Tree) for reach VPN in the public network to transmit private network multicast protocol packets and data packets, which will be converted into public network multicast data packets on PE and be forwarded via MDT. There are two types of MDTs in the MD scheme:

- Default-MDT

Each multicast domain will create a Default-MDT to connect all PEs in the domain. Through this Default-MDT, the customer network can treat the multicast domain as a LAN, which is to say the CE multicast data transmitted to the ingress PE will be forwarded over the public network to all egress PEs in this multicast domain, while such multicast data are transmitted on Default-MDT in the form of multicast. The egress PE will determine whether or not to forward these multicast data by verifying whether there is multicast receiver in the VPN sites connected. If no receiver exists in the sites, the multicast data will be discarded.

- Data-MDT

Data-MDT is an optional optimized means to transmit multicast traffic. It will only forward to multicast traffic to the egress PE in need of such multicast traffic. When the ingress PE intends to create a Data-MDT, it will first send a MDT-join packet through Default-MDT, and such packets carries such information as C-S (source address of customer multicast traffic), C-G (group address of customer multicast traffic) and P-G (group address for transmission over public network), indicating that C-S and C-G identified multicast traffic will be transmitted along P-G identified Data-MDT. Upon receipt of MDT-Join packets, the egress PE will only join the Data-MDT when it is in need of such multicast traffic. Therefore, the multicast traffic will arrive at the egress PE in need of the multicast traffic along Data-MDT, thus saving the resources of irrelevant egress PEs and the bandwidth of public network. Hence, if the private network contains certain high-rate multicast traffic, we can forward such multicast traffic by configuring Data-MDT, so as to avoid bandwidth wasting caused by the use of Default-MDT.

10.1.3 Single-AS Multicast VPN

In the single-AS multicast VPN, we can use PIM-SM/SSM to create Default-MDT in order to interconnect different PEs.

- Use PIM-SM to create Default-MDT

In this scheme, by running PIM-SM on the public network, each PE uses the same group address (group address of Default-MDT) to join the RP on the public network, thus forming a shared tree (RPT) connecting RP and each PE. Then, each PE will register with RP, so that RP creates the SPT reaching each PE. In this way, the multicast packets sent by each PE can be transmitted through such SPT and RPT, and eventually reach other PEs. The Default-MDT is hence successfully created.

- Use PIM-SSM to create Default-MDT

In this scheme, by running PIM-SSM on the public network, each PE uses the MDT address-family routes of BGP to inform other PEs of its NLRI information (including PE address and Default-MDT group address). When one PE receives the MDT address-family route of BGP, it will use the PE address and Default-MDT group address contained therein to join SPT with the PE providing this route. A SPT is eventually formed with PE initiating the MDT address-family routes being the root and PEs receiving such MDT address-family routes being the leaves. Since each PE will initiate the MDT address-family route, multiple SPTs will be formed eventually. In this way, the multicast packets sent by each PE can be transmitted to other PEs via these SPTs, and Default-MDT is created accordingly.

10.1.4 Multi-AS Multicast VPN

When VPN needs to cross multiple ASes, there are three unicast solutions to address the PE communication problem between ASes. There are also different multicast VPN options based on these three unicast solutions.

- OptionA: VRF-to-VRF

VRF is configured for each VPN on the ASBR, which act as PE, and ASBRs are interconnected through their respective VPN instance and regard each other as a CE device. In this approach, ASBR acts as PE and the MDT is established within each AS. The multicast data is transmitted over the public network through MDTs of multiple ASes.

- OptionB: Single-hop EBGP

PE uses to IBGP to advertise the VPN route to the ASBR of local AS, and then ASBRs of different ASes will advertise the VPN route of other ASes to the local ASes through EBGP. In this way, ASBR no long acts as PE, and the MDT of multicast VPN must cross each AS in order to connect PEs of respective ASes.

- OptionC: Multi-hop EBGP

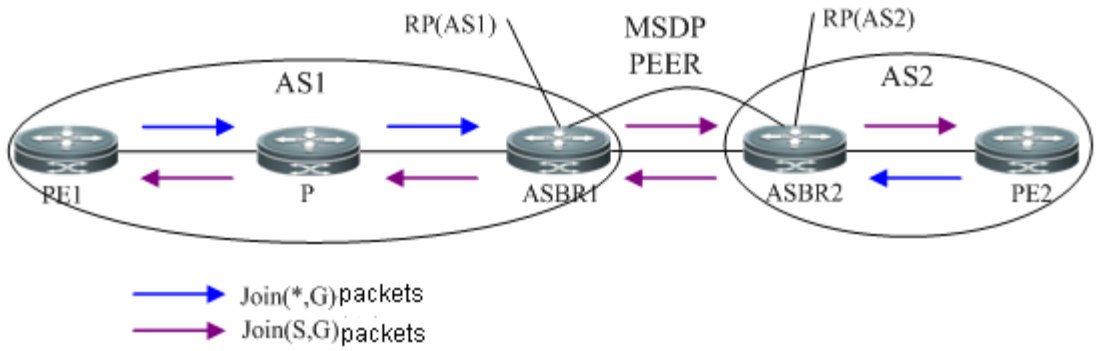
In this way, PEs of different ASes are interconnected via Multihop-EBGP, and ASBR will neither participate in the reception and advertisement of VPN route nor act as PE. In this approach, the MDT of multicast VPN must cross each AS in order to connect PEs of respective ASes.

In OptionA, since MDT is established within AS, the process is similar to that of single-AS multicast VPN. In OptionB and OptionC, when MDT is created using PIM-SM and PIM-SSM, there will be two different scenarios:

- Use PIM-SM to create multi-AS MDT

In order to interconnect different PEs in respective ASes via the same MDT, we can use MSDP to share multicast source information between RPs of two ASes. The process is shown below:

Figure 23 Use PIM-SM to create multi-AS MDT

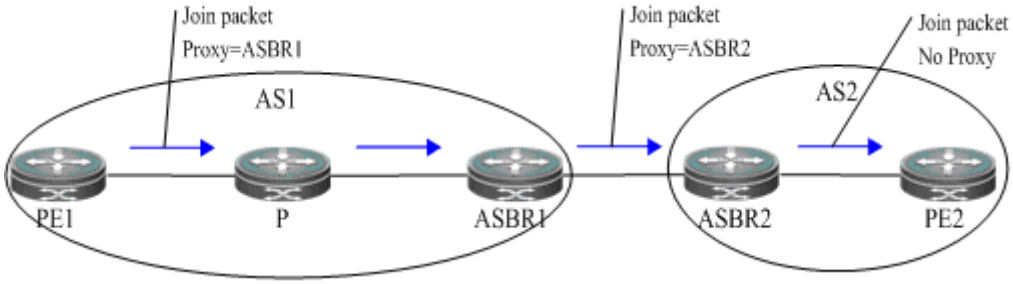


In this scenario, the ASBR of each AS is configured to MSDP peer, which is used to learn the multicast source information of other ASes, so as to initiate the grafting of SPT to other ASes. As shown above, PE1 and PE2 first initiate shared tree grafting to local RPs in order to create Default-MDT. When RP (ASBR1) in AS1 learns the multicast source information (PE2) of AS2 via MSDP, it will send Join (S,G) packets and eventually create a SPT with ASBR1 as the leave and PE2 as the root. Likewise, when ASBR2 learns the multicast source information (PE1) of AS1, a SPT with ASBR2 as the leave and PE1 as the root will also be created. In this way, the multicast packets sent by PE1 can be forwarded to ASBR2 along SPT, while ASBR2 will forward such packets to PE2 along the shared tree. Therefore, through the shared tree and SPT tree, PEs in different ASes can then be interconnected.

- Use PIM-SSM to create multi-AS MDT

In the process of using PIM-SSM to create multi-AS SPT, the P router in one AS may not have the route to the PE in another AS, and this will lead to the failure in SPT creation. To address this problem, PIM-SSM extends the attribute of Join packet so that it carries proxy information, while P router will use such proxy information to create SPT, as shown below:

Figure 24 Create multi-AS SPT



In the above figure, when PE1 initiates to create SPT, it will select ASBR1 as the proxy and include its IP address in the Join packet. When P receives this Join packet, it will use proxy as the destination address to discover RPF neighbors and initiate SPT grating to neighbor ASBR1 that leads to the proxy. After ASBR1 receives the Join packet, it will find out that the proxy address is its own IP address, and it will reselect a new proxy (ASBR2) and initiate SPT grafting. In this way, a SPT is formed through proxy grafting, so that PEs in respective ASes can be interconnected.

Note

When using PIM-SM to create multi-AS MDT, all routers along the path from RPs in respective ASes to the multicast source (namely PEs in other ASes) must have the unicast route to the multicast source. Therefore, this scenario requires advertising the route of each PE to the IGP of each AS.

Note

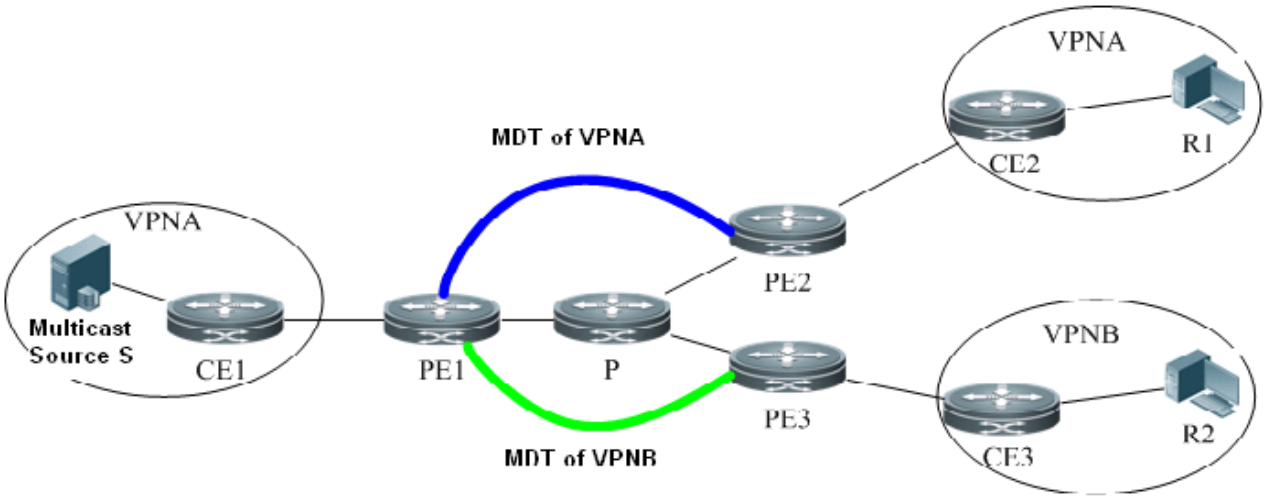
When using PIM-SSM to create multi-AS MDT, the P router is not required to have the route to PEs in other ASes. This scenario boasts higher flexibility, but will require PEs on the public network and P router to support proxy-based SPT creation.

10.1.5 Extranet Application of Multicast VPN

Extranet multicast VPN (Extranet-MVPN) enables the multicast source of a multicast VPN to send multicast streams to other multicast VPN sites. On the Extranet-MVPN, a cross-VPN sender is called an "Extranet-Source". The VRF where the sender is located is called the source MVRF. The PE connected to the source MVRF is called the ingress PE. A cross-VPN receiver is called an Extranet-Receiver. The VRF where the receiver is located is called a receiver MVRF. The PE connected to the receiver MVRF is called an egress PE. See the following figure. The multicast source on VPN A can forward the multicast stream to VPN B, so that the receiver on VPN B can receive the multicast stream. In this scenario, the VRF (on PE1) connected to the VPNA can be referred to as the source MVRF. Multicast source S can be referred to as the Extranet-Source. The VRF (on PE3) connected to VPN B can be referred to as the receiver MVRF. The receiver R2 can be referred to as the Extranet-Receiver.

On the Extranet-MVPN, a PE establishes a cross-VRF multicast distribution tree for introducing multicast streams of other VPN sites to the PE. Meanwhile, the PE uses the cross-VRF multicast stream forwarding function to forward streams from one VPN to another. The following figure shows the detailed process.

Figure 25 Work process of the Extranet-MVPN



As shown in the preceding figure, VPN A works as the sharing VPN; VPN B works as the receiver VPN; The VPNA unicast route is imported for VPN B on PE1 and PE3, so that CE3 can locate the route leading to the multicast source S. The corresponding VRF is configured for every VPN on all PEs and the MD function is enabled. They form the Extranet-MVPN for R2 between the multicast source S and VPN B.

- Establishment Process of MDTs on a Public Network

As shown in the preceding figure, the corresponding VRF is configured for every VPN on all PEs and the MD function is enabled (the VRFs corresponding to VPN A and VPN B are both configured on PE1). In this case, an MDT is created to connect PE1 VPN A to PE2 VPN A for transmitting multicast streams of VPN A on the public network; another MDT is created to connect PE1 VPN B to PE2 VPN B for transmitting multicast streams of VPN B on the public network.

➢ Process of Creating MDTs

The processes of creating MDTs are similar for PIM-SM and PIM-SSM. The only difference lies in that a multicast router initiates the graft process in the RP direction for PIM-SM, while a multicast router initiates the graft process in the multicast source direction for PIM-SSM. The process of creating an MDT for PIM-SSM is set out as follows as an example.

1) During the transmission from R1 to S, CE2 sends a Join packet to PE2 to create a MDT, and then PE2 uses the MDT of VPN A to send the Join packet to PE1. Upon receiving the Join packet, PE1 initiates the graft process to CE1, and eventually an MDT from CE1 to CE2 is created on VPN A.

2) During the transmission from R2 to S, CE3 sends a Join packet to PE3 to create an MDT, and then PE3 uses the MDT of VPN B to send the Join packet to PE1. Upon receiving the Join packet, PE1 performs the RPF check and finds that the RPF interface belongs to VRF A. PE1 notifies the PIM instance on VRF A of the receiving request. Then, the PIM instance on VRF A takes over the initiating of the MDT creation process. The PIM instance on VRF A sends the Join packet to CE1 for grafting. Finally, an MDT from CE1 (on VPN A) to CE3 (on VPN B) is created.

➢ Multicast Stream Forwarding Process

1) After a multicast stream of the multicast source S is forwarded through CE1 to PE1, PE1 forwards the multicast stream to PE2 and eventually to R1 through the MDT of VPN A.

2) After a multicast stream of the multicast source S is forwarded through CE1 to PE1, PE1 can detect VRF B's request on the multicast stream because the PIM instance of VRF B has notified the PIM instance of VRF A of the multicast-receving request when the MDT is created previously. In this case, PE 1 forwards the multicast stream to VRF B, transmits the multicast stream on the public network through the MDT on VPN B first to PE3 and eventually to R2.


Note

If a private network runs PIM-SM, the multicast source and RP must be on a same VPN. Otherwise, the source DR fails to register with RP and an MDT fails to be created.

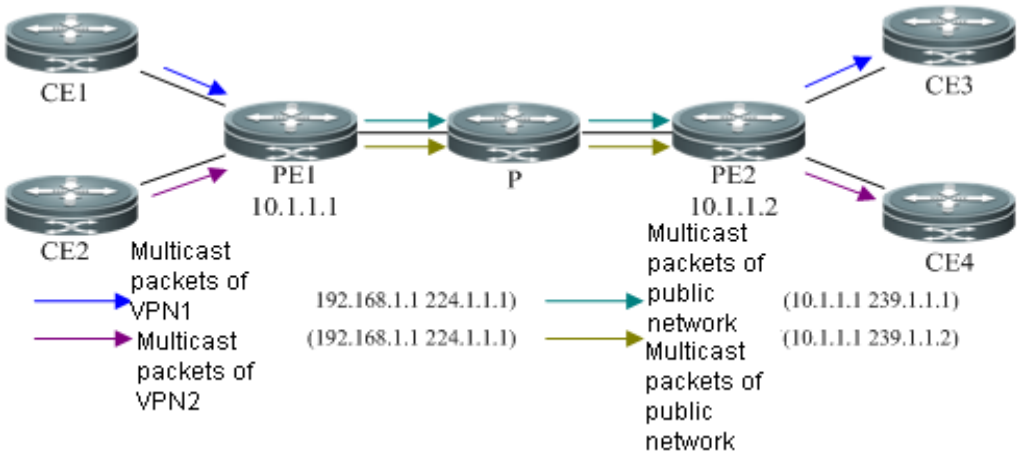
10.1.6 Multicast VRF

One PE may be connected with multiple CEs of different VPN sites. In order to differentiate the multicast traffic from different VPN sites on PE, the MD scheme also uses multiple VRFs to forward the multicast traffic from multiple VPN sites, and these VRFs are called MVRF (Multicast VPN Routing and Forwarding). The connection between PE and the CE of each VPN site needs to be associated with one VRF. When PE receives the multicast packets from CE, it will look up the multicast forwarding table under corresponding MVRF and forward packets as per such forwarding table. In this way, MVRF addresses the problem of local multicast route conflict on PE and isolate the multicast traffic in each VPN site.

The creation and maintenance process of MVRF is similar to that of VRF, yet multicast route must be initiated for each MVRF so that it can generate relevant multicast forwarding table. In the MD scheme, an MVRF is created on PE for each VPN to forward the corresponding multicast traffic. Meanwhile, each MVRF contains one Multicast Tunnel Interface (MTI), which is a GRE Tunnel interface converting private network multicast traffic into public network multicast traffic, which will then be transmitted over the public network through the public network multicast forwarding table.

Through the forwarding of private network multicast traffic by MVRF and the conversion of multicast traffic by MTI interface, the multicast data packets will eventually be forwarded from one VPN site to another VPN site over the public network. The packet forwarding process is shown below:

Figure 26 Forwarding process of multicast data packets



As shown above, the forwarding process of multicast data packets is shown below:

1) CE1 and CE2 forward the multicast data packets in respective VPNs to PE1. The source address and destination address carried by data packets of two VPNs could be identical (as shown above, they are both 192.168.1.1, 224.1.1.1).

2) After PE1 receives the data packets transmitted from CE1 and CE2, it will select different MVRFs to forward these two multicast traffic according to the VPN. When each multicast traffic is forwarded to the MTI of respective MVRF, the multicast traffic will be encapsulated into the corresponding public network multicast traffic. As shown above, the multicast traffic of VPN1 is encapsulated into (10.1.1.1, 239.1.1.1), and that of VPN2 is encapsulated into (10.1.1.1, 239.1.1.2). These two multicast traffic will be transmitted to PE2 through public network multicast routing and forwarding.

3) PE2 will de-encapsulate these two public network multicast traffic and restore to the private network multicast traffic, which will then be forwarded by the MVRFs corresponding to their own VPNs and reach CE3 and CE4 respectively. Such traffic will eventually reach the multicast receivers in respective VPNs.

Caution

To enhance the efficiency of packet processing, no GRE checksum will be calculated or configured during packet encapsulation on the MTI interface of ingress PE.

Caution

Before packet encapsulation, the MTI interface of ingress PE cannot proceed with fragmentation first according to IP MTU of MTI interface.

10.2 Configuring multicast VPN

10.2.1 Configuring Single-AS Multicast VPN

Before configuring single-AS multicast VPN, the following network configurations must be completed first:

- Configure a unicast routing protocol to enable routes in AS
- Configure BGP/MPLS L3VPN

The configuration of single-AS multicast VPN mainly involves the following:

- Configure Default-MDT (Required)
- Configure Data-MDT (Optional)
- Configure multicast routing on VRF (Required)
- Configure PIM-SSM (Optional)
- Configure BGP MDT address family (optional)

10.2.1.1 Configuring Default-MDT

Steps of Default-MDT configuration are shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **ip vrf** *vrf_name*	Create a VRF and enter VRF configuration mode.
Qtech(config-vrf)# **rd** *route_distinguisher*	Configure a RD for VRF; RD information will be need when BGP distributes VPN routes and MDT-SAFI routes.
Qtech(config-vrf)# **route-target** {**both	export	import**} *rt-value*	Configure RT value.
Qtech(config-vrf)# **mdt default** *group_address*	Configure Default-MDT: specify the group address and enable MD.


Note

After executing "**mdt default**" command, a MTI interface will be generated automatically. This MTI interface can only be dynamically generated by MD control plane instead of user configuration.


Note

Execute "**neighbor** *ip-address* **activate**" command under vpnv4 address family; after establishing the connection between device and neighbor, MTI interface will take the interface automatically acquired to establish connection with neighbor as the source interface, and take the source address to establish connection with neighbor as its IP address.


Note

After configuration, the group address of Default-MDT cannot be changed by running the command. To change the group address, you can only delete the existing one and configure a new one. If you run the **no mdt default** command to delete Default-MDT, the MTI interface and the configuration command of Data-MDT corresponding to the VRF will also be deleted.

Configure Default-MDT

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1: 100
Qtech(config-vrf)# route-target both 1: 100
Qtech(config-vrf)# mdt default 239.1.1.1
```

10.2.1.2 *Configuring Data-MDT*

To configure Data-MDT forwarding for some high-rate private network multicast traffic, you need to configure Data-MDT under VRF mode, as shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **ip vrf** *vrf_name*	Create a VRF and enter VRF configuration mode.
Qtech(config-vrf)# **mdt data** group_address wildcard_bits [**list** access-list]	Configure Data-MDT.


Note

After executing "mdt data" command, all private network multicast traffic meeting ACL filtering conditions will reach PE, which will create a Data-MDT. Each PE can only create a limited number of Data-MDT. Therefore, it is suggested that Data-MDT can only be configured for certain high-rate private network multicast traffic.


Note

Data-MDT cannot be reused, i.e., one Data-MDT can only be used to forward one private network multicast traffic.

Configure Data-MDT, so that the private network multicast traffic specified by ACL will be forwarded with Data-MDT.

```
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# mdt data 238.1.1.0 0.0.0.255 list vpn1-datamdt
Qtech(config-vrf)# exit
Qtech(config-vrf)# ip access-list standard vpn1-datamdt
Qtech(config-std-nacl)# permit 224.1.1.0 0.0.0.255
```

10.2.1.3 Configuring Multicast Routing on VRF

Configuration of multicast routing on VRF will need to enable multicast routing on VRF and configure the PIM-SM instance on this VRF. When vrf parameter is not specified, the following commands shall mean the configuration of public network multicast routing. The following table describes the procedure.

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **ip multicast-routing vrf** *vrf_name*	Enable multicast routing of certain VRF.
Qtech(config)# **ip pim** [**vrf** *vrf-name*] **rp-address** *rp-address* [*access_list*]	Configure static RP on certain VRF.
Qtech(config)**# interface** *type ID*	Configure interface.
Qtech(config-if-*type ID*)# **ip vrf forwarding** *vrf_name*	Bind the interface with certain VRF.
Qtech(config-if-*type ID*)# **ip address** *ip-address mask*	Configure IP address on interface.
Qtech(config-if-*type ID*)# **ip pim sparse-mode**	Enable PIM-SM on interface.

Configure multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# exit
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip pim vrf vpn1 rp-address 1.1.1.1
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 0/1)# ip address 1.1.1.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
```

10.2.1.4 Configuring PIM-SSM

This step intends to create Default-MDT in order to use PIM-SSM, as shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **ip pim ssm** { **default** | **range** *access_list*}	Configure the scope of PIM-SSM. After configuring PIM-SSM, the group address in "mdt default" command must fall within the scope of SSM. Only in this way we can use PIM-SSM to create Default-MDT.

Configure the scope of SSM

```
Qtech# configure terminal
Qtech(config)# ip pim ssm default
```

10.2.1.5 Configuring BGP MDT Address Family

When using PIM-SSM to create Default-MDT, you will need to configure BGP MDT address family. Through the routing of MDT address family, PE can discover other PE addresses and initiate the grating of SPT to other PEs. The configuration steps of MDT address family are shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **router bgp** *asn-num*	Create BGP and enter BGP configuration mode.
Qtech(config-router)# **neighbor** *ip-address* **remote-as** *asn-number*	Configure BGP session.
Qtech(config-router)# **neighbor** *ip-address* **update-source** *interface-name*	Configure to use interface address as the source address when MP-IBGP session is established; usually, the Loopback interface address is used as the source address.
Qtech(config-router)# **address-family ipv4 mdt**	Enter MDT address family.
Qtech(config-router-af)# **neighbor** *ip-address* **activate**	Activate the route to exchange MDT address family on BGP session.
Qtech(config-router-af)# **neighbor** *ip-address* **next-hop-self**	Change the next-hop route to self; this command can be executed on ASBR in OptionB.

Configure MDT address family

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.3 remote-as 2
Qtech(config-router)# neighbor 10.1.1.3 update-source loopback 0
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighobr 10.1.1.3 activate
```

10.2.2 Configuring Multi-AS Multicast VPN

The configuration of multi-AS multicast VPN could be different according to different unicast schemes. In OptionA, the configuration of multicast VPN is similar to that of single AS, and is hence not introduced herein. We will mainly introduce the configurations of OptionB and OptionC. Before configuring multi-AS multicast VPN, the following network configurations must have been completed:

- Configure multi-AS BGP/MPLS L3VPN
- Configure Default-MDT

The configuration of multi-AS multicast VPN mainly involves the following:

- Multicast VPN configuration when using PIM-SM
- Multicast VPN configuration in OptionB when using PIM-SSM

- Multicast VPN configuration in OptionC when using PIM-SSM

10.2.2.1 Multicast VPN Configuration when Using PIM-SM

When configuring MSDP-based multi-AS multicast VPN, the multicast configuration on PE shall be the same as the configuration to create Default-MDT with PIM-SM. MSDP must be configured on ASBR, as shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **ip msdp peer** *peer-address* **connect-source** *interface-type interface-number*	Configure MSDP peer.

Configure MSDP peer

```
Qtech# configure terminal
Qtech(config)# ip msdp peer 10.0.5.9 connect-source gi 0/1
```

10.2.2.2 Multicast VPN Configuration in OptionB when Using PIM-SSM

Configure multicast VPN in OptionB, so that it can use PIM-SSM to create Default-MDT, as shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech(config)# **router bgp** *asn-num*	Create BGP and enter BGP configuration mode.
Qtech(config-router)# **neighbor** *ip-address* **remote-as** *asn-number*	Configure BGP session.
Qtech(config-router)# **neighbor** *ip-address* **update-source** *interface-name*	Configure to use interface address as the source address when BGP session is established; usually, the Loopback interface address is used as the source address.
Qtech(config-router)# **address-family ipv4 mdt**	Enter MDT address family.
Qtech(config-router-af)# **neighbor** *ip-address* **activate**	Activate the route to exchange MDT-SAFI on BGP session.
Qtech(config-router-af)# **exit**	Exit the address family.
Qtech(config-router)# **exit**	Exit BGP mode.
Qtech(config)# **ip multicast** [**vrf** *vrf-name*] **rpf proxy rd vector**	Configure whether to enable RPF Vector on certain VRF. In OptionB, the rd parameter of this command must be configured to carry RD information in Join attribute. This is because in OptionB, the next-hop of route will change when MDT-SAFI route passes the ASBR. Therefore, the next hop may be different for route entries with same destination address but different group addresses. By this time, we will need to use RD+ destination address as the index for discovering routes.
Qtech(config)# **ip pim ssm default**	Configure PIM-SSM.

Configure multicast VPN in OptionB

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.3 remote-as 1
Qtech(config-router)# neighbor 10.1.1.3 update-source loopback 0
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 10.1.1.3 activate
Qtech(config-router-af)# neighbor 10.1.1.3 next-hop-self
Qtech(config-router-af)# exit
Qtech(config-router)# exit
```

```
Qtech(config)# ip multicast vrf vpn1 rpf proxy rd vector
Qtech(config)# ip pim ssm default
```

10.2.2.3 *Multicast VPN Configuration in OptionC when Using PIM-SSM*

Configure multicast VPN in OptionC, so that it can use PIM-SSM to create Default-MDT, as shown below:

Command	Function
Qtech# **configure terminal**	Enter global configuration mode.
Qtech (config)# **router bgp** *asn-num*	Create BGP and enter BGP configuration mode.
Qtech (config-router)# **neighbor** *ip-address* **remote-as** *asn-number*	Configure BGP session.
Qtech(config-router)# **neighbor** *ip-address* **update-source** *interface-name*	Configure to use interface address as the source address when BGP session is established; usually, the Loopback interface address is used as the source address.
Qtech(config-router)# **neighbor** *ebgp-peer-address* **ebgp-multihop**	Configure multi-hop attribute.
Qtech (config-router)# **address-family ipv4 mdt**	Enter MDT address family.
Qtech (config-router-af)# **neighbor** *ip-address* **activate**	Activate the route to exchange MDT-SAFI on BGP session.
Qtech (config-router-af)# **exit**	Exit the address family.
Qtech (config-router)# **exit**	Exit BGP mode.
Qtech (config)# **ip multicast rpf proxy vector**	Enable RPF Vector.
Qtech (config)# **ip pim ssm default**	Configure PIM-SSM.

Configure multicast VPN in OptionC.

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.5 remote-as 2
Qtech(config-router)# neighbor 10.1.1.5 update-source loopback 0
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 10.1.1.5 activate
Qtech(config-router-af)# exit
Qtech(config-router)# exit
Qtech(config)# ip multicast rpf proxy vector
Qtech(config)# ip pim ssm default
```

10.2.3 Extranet-MVPNConfiguring Extranet-MVPN

Ensure that the following network configurations are completed before configuring Extranet-MPVN:

- Configuration of single-AS or cross-AS multicast VPN

To ensure that the multicast stream can be transmitted from the ingress PE to the egress PE on the public network, the following two ways to configure Extranet-MVPN:

- Configuring the receiver MVRF on the ingress PE.
- Configuring the source MVRF on the egress PE.

10.2.3.1 Configuring the Receiver MVRF on the Ingress PE

After this method is used, the ingress PE is added to the Mobile Data Terminal (MDT) of the receiver MVRF. In this case, the ingress PE can send the multicast stream to the public network MDT of the egress PE, and eventually to the egress PE. The following table describes the procedure.

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **ip vrf** *vrf_name*	Create a VRF and enter the VRF configuration mode.
Qtech(config-vrf)# **rd** *route_distinguisher*	Configure an RD, BGP distribution VPN routing, and RD information used during MDT-SAFI routing for the VRF.
Qtech(config-vrf)# **route-target import** *rt-value*	Configure the RT value to be imported. Ensure that the receiver MVRF of the ingress PE imports the VPN route where the multicast source resides, so that the PE can create the cross-VRF multicast distribution tree.
Qtech(config-vrf)# **mdt default** *group_address*	Configure Default-MDT. The command specifies the group address and enables the MD. The group address must be the same as that of the MVRF of the egress PE.

If multiple receiver VPNs exist, you need to configure multiple receiver MVRFs on the ingress PE. When forwarding the multicast stream, the ingress PE makes a copy of the multicast steam and broadcasts it on the public network for the MDT of each receiver MVRF. If the traffic of the multicast steam is high, making copies of the multicast stream will cause heave waste of the public network's bandwidth. In this case, you can configure the source MVRF on the egress PE. The method saves the trouble of configuring the source MVRF on all egress PEs. However, it may waste bandwidth. Therefore, it should only be used when the traffic of the multicast stream is low.

Configure the receiver MVRF.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn2
Qtech(config-vrf)# rd 2: 200
Qtech(config-vrf)# route-target import 1: 100
Qtech(config-vrf)# mdt default 239.2.2.2
```

10.2.3.2 *Configuring the Source MVRF on the Egress PE*

This method is to configure the MVRF consistent with the ingress PE on the egress PE, so that the egress PE can join the public-network MDT of the source MVRF and receive the multicast stream sent by the ingress PE on the public network.

The following table describes the procedure.

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **ip vrf** *vrf_name*	Create a VRF and enter the VRF configuration mode.
Qtech(config-vrf)# **rd** *route_distinguisher*	Configure an RD, BGP distribution VPN routing, and RD information used during MDT-SAFI routing for the VRF.
Qtech(config-vrf)# **mdt default** *group_address*	Configure Default-MDT. This command specifies the group address and enables the MD. The group address must be the same as that of the MVRF of the ingress PE.
Qtech(config-vrf)# **ip mroute** [**vrf** *vrf-name*] *source-address mask* **fallback-lookup** { **global** | **vrf** *vrf-name* } [*distance*]	Configure a static multicast route and specify the VRF to be selected during the RPF check. Ensure that the VRF is consistent with the source MVRF.

This method is recommended for saving public-network bandwidth when the traffic of the multicast stream send by the multicast source is high. Pay attention to the following points when using this method:

- You must configure the source MVRF on all egress PEs. The configuration is more complicated than configuring the receiver PE on the ingress PE.

- When querying the multicast RPF on the VRF (connected to the receiver VPN) of the egress PE, you need to locate the source MVRF and use the MDT of the source MVRF to send the multicast receiving request to the ingress PE through the public network.

Configure the source MVRF.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1: 100
Qtech(config-vrf)# route-target import 1: 100
Qtech(config-vrf)# mdt default 239.1.1.1
Qtech(config-vrf)# ip mroute vrf vpn2 192.168.0.0 255.255.255.0 fallback-lookup vrf
vpn1
```

10.2.3.3 *Configuring the VRF to Be Selected by the Static Multicast Route*

The unicast route is used in the RPF check by default. Therefore, the same VRF is used in the multicast RPF check and unicast routing. You can configure the static multicast route to change default RPF configuration for distinguishing the topology of the multicast route from the topology of the unicast route and enabling the unicast and multicast routes to use different VRFs.

Specifically, you need to specify a VRF (not used by the unicast route) to be used by the static multicast route for the multicast RPF check on the network segment. The VRF must be different from that used by the unicast route.

This configuration method allows the multicast stream of a VPN site to be released on the public network. BGP cannot deliver a VPN route to the routing table on the public network. As a result, the RPF check on the public network cannot locate the next-hop leading to the multicast source of the VPN site. The configuration of the static multicast route, however, allows specifying a VRF used in the RPF check. Thereafter, the next-hop leading to the multicast source of the VPN site can be located.

The following table describes the procedure.

Command	Function
Qtech# **configure terminal**	Enter the global configuration mode.
Qtech(config)# **ip mroute** [**vrf** *vrf-name*] *source-address mask* **fallback-lookup** {**global | vrf** *source-vrf-name*} [*distance*]	Specify a VRF in the static multicast route. Set vrf *vrf-name* to the VRF where this command functions for the RPF check. Set *source-address mask* to the route prefix and mask for specifying the route range where the command functions for the RPF check. Use the keyword global to indicate that the RPF check is performed on the public network. Set vrf *source-vrf-name* to the VRF used in the RPF check. Set *distance* to the distance value of the route.
Qtech(config)# **end**	Exit the configuration mode.
Qtech# **show ip mroute** [**vrf** *vrf-name*] static	Show the configuration result of static multicast route.

Specify the VRF to be selected by the static multicast route.

```
Qtech# configure terminal
Qtech(config)# ip mroute vrf vpn1 1.1.1.0 255.255.255.0 fallback-lookup vrf vpn2
Qtech(config)# end
```

10.2.4 Verifying the Operating Status of Multicast VPN

This section will introduce how to verify the configurations of multicast VPN and routing information. Enter privilege mode and execute the following commands:

Command	Function
Qtech# **show ip pim mdt** [**bgp**]	Display MDT information.
Qtech# **show ip pim vrf** *vrf-name* **mdt send**	Display the sending status of MDT-Join message.

Qtech# **show ip pim vrf** *vrf-name* **mdt receive**	Display the reception status of MDT-Join message.
Qtech# **show ip pim sparse-mode** [**vrf** *vrf-name*] **neighbor [detail]**	Display the establishment status of PIM-SM neighbor.
Qtech# **show ip pim sparse-mode mroute proxy**	Display the RPF Vector information of PIM-SM.

Display MDT information
```
Qtech# show ip pim mdt
* implies group is the MDT default group
  MDT Group       Interface   Source                  VRF
* 239.1.1.1       Tunnel0     Loopback0               vpn1
```

Display the sending and reception information of MDT-Join packets.
```
Qtech# show ip pim vrf vpn1 mdt send
MDT-data send list for VRF: vpn1
  (source, group)                    MDT-data group
  (192.168.1.107, 224.1.1.1)         239.1.2.0

Qtech# show ip pim vrf vpn1 mdt receive
Joined MDT-data [group : source]  uptime/expires for VRF: vpn1
 [239.1.2.0 : 0.0.0.0]  00:00:28/00:02:35
```

Display PIM-SM neighbor
```
Qtech# show ip pim sparse-mode vrf vpn1 neighbor
Neighbor    Interface           Uptime/Expires      Ver   DR
Address                                                   Priority/Mode
1.1.1.20    FastEthernet 0/1   00:02:25/00:01:20   v2    1 / P
```

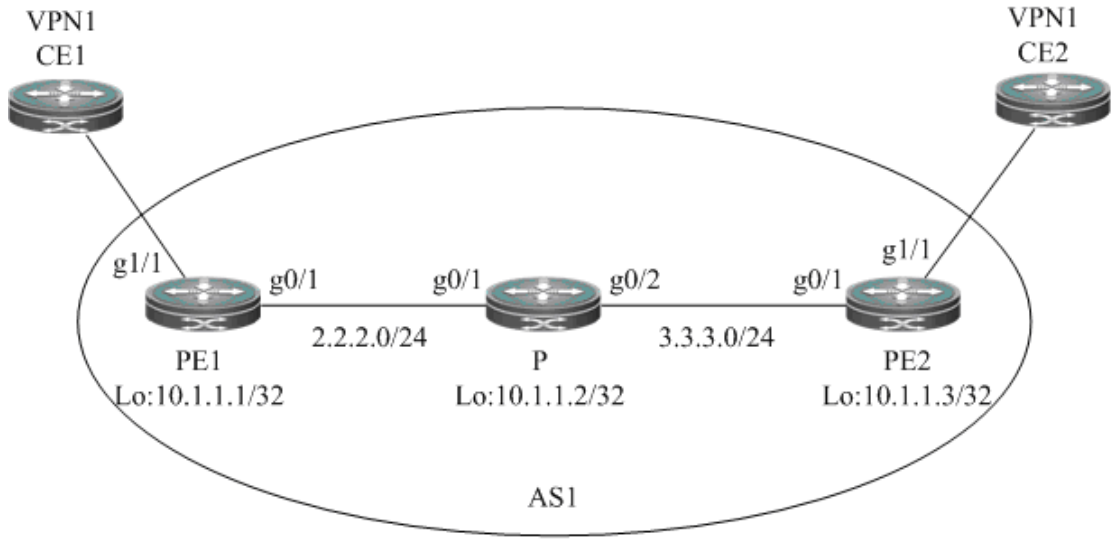
Display PIM-SM RPF Vector
```
Qtech# show ip pim sparse-mode mroute proxy
(20.1.1.1, 232.1.1.1)
  Proxy                    Assigner         Origin    Uptime/Expire
  1:100/10.1.1.3           0.0.0.0          BGP MDT   00:09:25/stopped
```

10.3 Multicast VPN Configuration Example

10.3.1 Example of Single-AS Multicast VPN Configuration

Requirement: The unicast routing has been configured for VPN and public network, and MPLS has also been configured for the public network, with topology shown below:

Figure 27 Topology of single-AS multicast VPN



Configuration steps:

- CE1:

Configure multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-GigabitEthernet 0/1)# ip address 192.168.0.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
```

- PE1:

Configure VRF

Create a VRF of "vpn1", define RD value and RT value and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 239.1.1.1
Qtech(config-vrf)# end
```

Configure private network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.0.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP protocol to establish IBGP session with PE2 and configure VPNv4 address family

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.3 remote-as 1
Qtech(config-router)# neighbor 10.1.1.3 update-source loopback 0
Qtech(config-router)# address-family vpnv4
Qtech(config-router-af)# neighbor 10.1.1.3 activate
Qtech(config-router-af)# neighbor 10.1.1.3 send-community extended
Qtech(config-router-af)# end
```

- P:

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.2 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface gigabitethernet 0/2
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/2)# no switchport
Qtech(config-GigabitEthernet 0/2)# ip address 3.3.3.1 255.255.255.0
Qtech(config-GigabitEthernet 0/2)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/2)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.2 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
```

- PE3:

The configuration steps are similar to that of PE1.

- CE2:

The configuration steps are similar to that of CE1.

- Verify configurations:

Execute "**show ip pim mdt**" command on PE1 to display MDT established.

```
Qtech# show ip pim mdt
* implies group is the MDT default group
  MDT Group       Interface   Source                  VRF
* 239.1.1.1       Tunnel0     Loopback0               vpn1
```

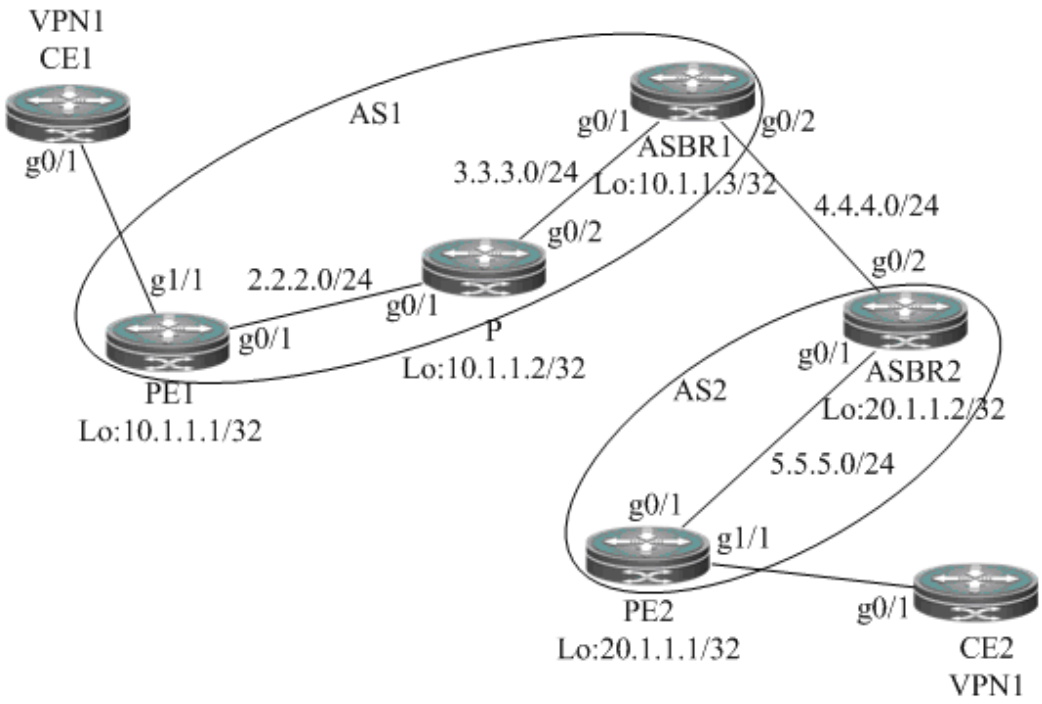
Execute "**show ip pim sparse-mode neighbor**" command on PE1 to display the neighbors established under VRF.

```
Qtech# show ip pim sparse-mode vrf vpn1 neighbor
Neighbor    Interface             Uptime/Expires      Ver   DR
Address                                                     Priority/Mode
192.168.0.1 Gigabitethernet 0/1 00:00:01/00:01:44  v2    1 / P
10.1.1.3     Tunnel0               00:02:25/00:01:20  v2    1 / P
```

10.3.2 Example of Multicast VPN Configuration in OptionC when Using PIM-SM

Requirement: BGP/MPLS L3VPN has been configured according to the scheme of OptionC, and PE route has been distributed to respective ASes through BGP.

Figure 28 Topology of multi-AS multicast VPN



Configuration steps:

- CE1:

The configuration steps are similar to that of CE1 in the scheme of single-AS multicast VPN.

- CE2:

The configuration steps are similar to that of CE2 in the scheme of single-AS multicast VPN.

- PE1:

Configure VRF

Create a VRF of "vpn1", define RD value and RT value and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 239.1.1.1
Qtech(config-vrf)# end
```

Configure private network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.0.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
```

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

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

- P:

The configuration steps are similar to that of P in the scheme of single-AS multicast VPN.

- ASBR1:

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 3.3.3.2 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface gigabitethernet 0/2
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/2)# no switchport
Qtech(config-GigabitEthernet 0/2)# ip address 4.4.4.1 255.255.255.0
Qtech(config-GigabitEthernet 0/2)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/2)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.3 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure candidate BSR, candidate RP and MSDP Peer, and configure BSR boundary on the interface connecting with ASBR2, so that two unicast domains can be isolated to different RP domains.

```
Qtech# configure terminal
Qtech(config)# ip pim bsr-candidate loopback 0
Qtech(config)# ip pim rp-candidate loopback 0
Qtech(config)# ip msdp peer 4.4.4.2 connect-source gi 0/2
Qtech(config)# interface gigabitethernet 0/2
Qtech(config-GigabitEthernet 0/2)# ip pim bsr-border
```

- ASBR2:

The configuration steps are similar to that of ASBR1.

- PE2:

The configuration steps are similar to that of PE1.

- Verify configurations:

Execute "**show ip pim mdt**" command on PE1 to display the MDT established.

```
Qtech# show ip pim mdt
* implies group is the MDT default group
  MDT Group       Interface   Source                  VRF
* 239.1.1.1       Tunnel0     Loopback0              vpn1
```

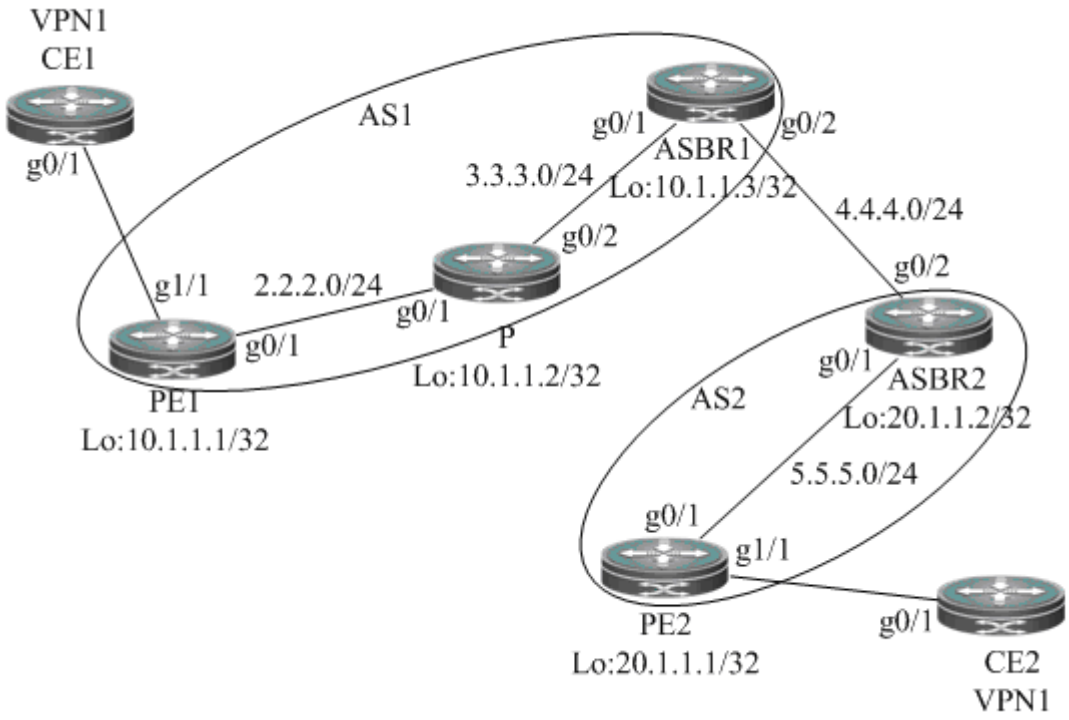
Execute "**show ip pim sparse-mode neighbor**" command on PE1 to display the neighbors established under VRF.

```
Qtech# show ip pim sparse-mode vrf vpn1 neighbor
Neighbor    Interface             Uptime/Expires      Ver    DR
Address                                                     Priority/Mode
192.168.0.1 Gigabitethernet 0/1 02:34:20/00:01:31  v2    1 / P
20.1.1.1     Tunnel0               01:55:19/00:01:37  v2    1 / P
```

10.3.3 Example of Multicast VPN Configuration in OptionB when Using PIM-SSM

Requirement: BGP/MPLS L3VPN has been configured according to the scheme of OptionB.

Figure 29 Topology of multi-AS multicast VPN (OptionB)



Configuration steps:

- CE1:

The configuration steps are similar to that of CE1 in the scheme of single-AS multicast VPN.

- CE2:

The configuration steps are similar to that of CE2 in the scheme of single-AS multicast VPN.

- PE1:

Configure VRF

Create a VRF of "vpn1", define RD value and RT value and configure Default-MDT. The group address of Default-MDT must fall within the scope of PIM-SSM.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 232.1.1.1
Qtech(config-vrf)# end
```

Configure private network multicast routing

```
Qtech# configure terminal
```

```
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.0.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure public network multicast routing, and configure to use RPF Vector that carries RD information.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip multicast vrf vpn1 rpf proxy rd vector
Qtech(config)# ip pim ssm default
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP protocol to establish IBGP session with ASBR1 and configure MDT address family

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.3 remote-as 1
Qtech(config-router)# neighbor 10.1.1.3 update-source loopback 0
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 10.1.1.3 activate
Qtech(config-router-af)# end
```

- P:

The configuration steps are similar to that of P in the scheme of single-AS multicast VPN.

- ASBR1:

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 3.3.3.2 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface gigabitethernet 0/2
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/2)# no switchport
Qtech(config-GigabitEthernet 0/2)# ip address 4.4.4.1 255.255.255.0
Qtech(config-GigabitEthernet 0/2)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/2)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.3 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
```

```
Qtech(config-Loopback 0)# end
```

Configure BGP protocol to establish IBGP session with PE1 and EBGP session with ASBR2, and configure MDT address family

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.1 remote-as 1
Qtech(config-router)# neighbor 10.1.1.1 update-source loopback 0
Qtech(config-router)# neighbor 4.4.4.2 remote-as 2
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 10.1.1.1 activate
Qtech(config-router-af)# neighbor 4.4.4.2 activate
```

- ASBR2:

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 5.5.5.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface gigabitethernet 0/2
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/2)# no switchport
Qtech(config-GigabitEthernet 0/2)# ip address 4.4.4.2 255.255.255.0
Qtech(config-GigabitEthernet 0/2)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/2)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 20.1.1.2 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP protocol to establish IBGP session with PE2 and EBGP session with ASBR1, and configure MDT address family

```
Qtech# configure terminal
Qtech(config)# router bgp 2
Qtech(config-router)# neighbor 20.1.1.1 remote-as 2
Qtech(config-router)# neighbor 20.1.1.1 update-source loopback 0
Qtech(config-router)# neighbor 4.4.4.1 remote-as 1
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 20.1.1.1 activate
Qtech(config-router-af)# neighbor 4.4.4.1 activate
```

- PE2:

The configuration steps are similar to that of PE1.

- Verify configurations:

Execute "**show ip pim mdt**" command on PE1 to display the MDT established.

```
Qtech# show ip pim mdt
* implies group is the MDT default group
  MDT Group       Interface   Source                VRF
* 232.1.1.1       Tunnel0     Loopback0             vpn1
```

Execute "**show ip pim sparse-mode neighbor**" command on PE1 to display the neighbors established under VRF.

```
Qtech# show ip pim sparse-mode vrf vpn1 neighbor
Neighbor    Interface            Uptime/Expires      Ver   DR
Address                                                    Priority/Mode
192.168.0.1 Gigabitethernet 0/1 02:34:20/00:01:31  v2    1 / P
```

```
20.1.1.1      Tunnel0                 01:55:19/00:01:37  v2     1 / P
```

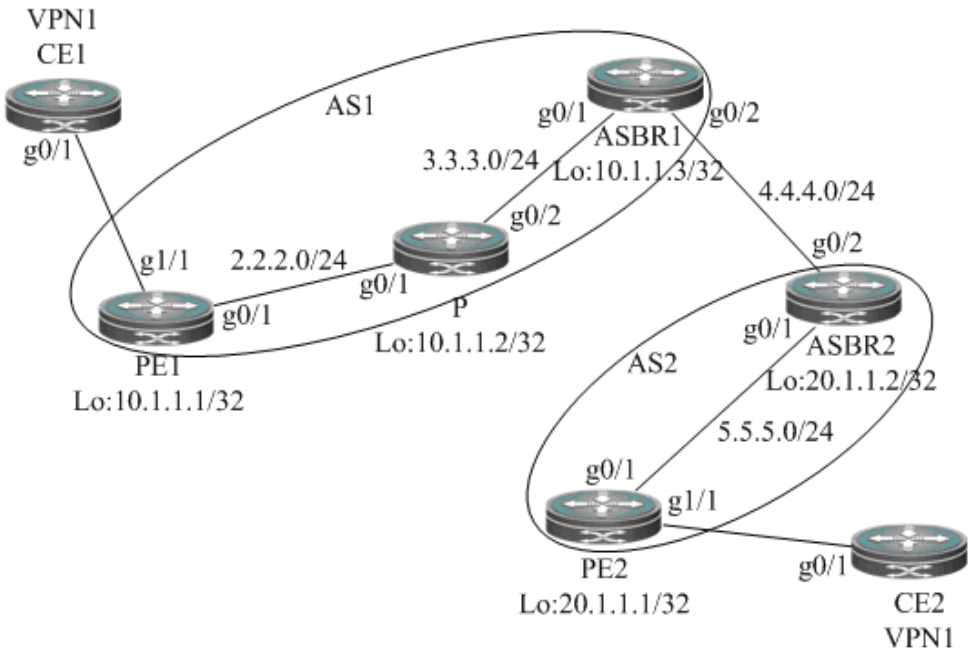
Execute "**show ip pim sparse-mode mroute proxy**" command on PE1 to display RPF Vector information.

```
Qtech# show ip pim sparse-mode mroute proxy
(20.1.1.1, 232.1.1.1)
  Proxy                       Assigner          Origin    Uptime/Expire
  1:100/10.1.1.3            0.0.0.0           BGP MDT   00:09:25/stopped
```

10.3.4 Example of Multicast VPN Configuration in OptionC when Using PIM-SSM

Requirement: BGP/MPLS L3VPN has been configured according to the scheme of OptionC.

Figure 30 Topology of multi-AS multicast VPN (OptionC)



Configuration steps:

- CE1:

The configuration steps are similar to that of CE1 in the scheme of single-AS multicast VPN.

- CE2:

The configuration steps are similar to that of CE2 in the scheme of single-AS multicast VPN.

- PE1:

Configure VRF

Create a VRF of "vpn1", define RD value and RT value and configure Default-MDT. The group address of Default-MDT must fall within the scope of PIM-SSM.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 232.1.1.1
Qtech(config-vrf)# end
```

Configure private network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
```

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

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.0.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip multicast rpf proxy vector
Qtech(config)# ip pim ssm default
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP protocol to establish mhop-EBGP session with PE2 and configure MDT address family

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 20.1.1.1 remote-as 2
Qtech(config-router)# neighbor 20.1.1.1 ebgp-multihop 255
Qtech(config-router)# neighbor 20.1.1.1 update-source loopback 0
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 20.1.1.1 activate
Qtech(config-router-af)# end
```

- P:

The configuration steps are similar to that of P in the scheme of single-AS multicast VPN.

- ASBR1:

Configure public network multicast routing

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# interface gigabitethernet 0/1
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 3.3.3.2 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface gigabitethernet 0/2
```

"no switchport" command is used to switch the port mode to "Routed Port" mode on switch products, and is not applicable to the router. Therefore, you don't need to execute this command on router products.

```
Qtech(config-GigabitEthernet 0/2)# no switchport
Qtech(config-GigabitEthernet 0/2)# ip address 4.4.4.1 255.255.255.0
Qtech(config-GigabitEthernet 0/2)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/2)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.3 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Execute BGP/MPLS L3VPN (OptionC) related configurations on ASBR; there is no need to configure MDT address family.

- ASBR2:

The configuration steps are similar to that of ASBR1.

- PE2:

The configuration steps are similar to that of PE1.

- Verify configurations:

Execute "**show ip pim mdt**" command on PE1 to display the MDT established.

```
Qtech# show ip pim mdt
* implies group is the MDT default group
  MDT Group       Interface   Source                  VRF
* 232.1.1.1       Tunnel0     Loopback0               vpn1
```

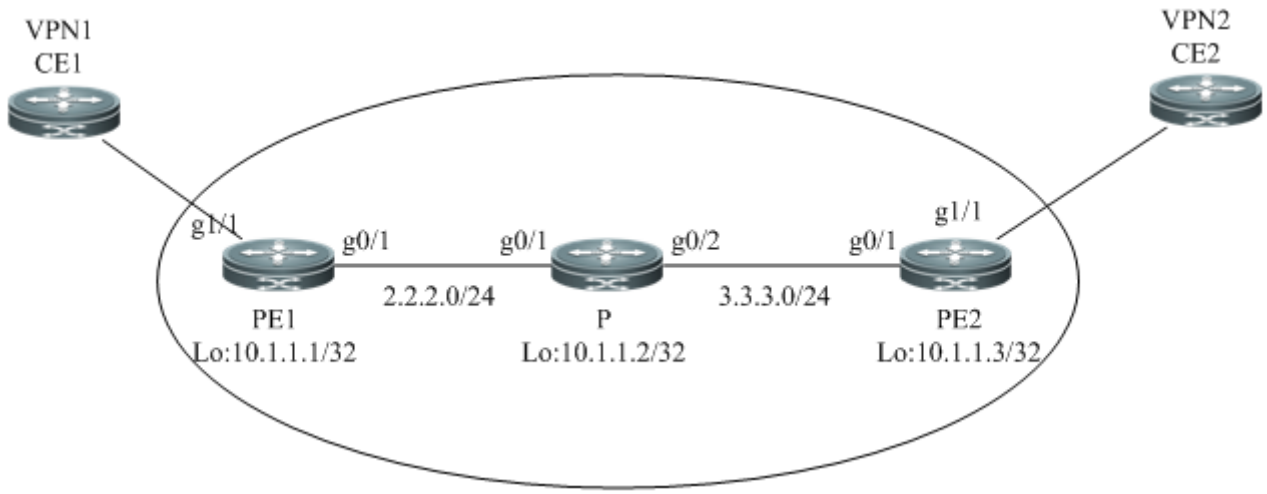
Execute "**show ip pim sparse-mode neighbor**" command on PE1 to display the neighbors established under VRF.

```
Qtech# show ip pim sparse-mode vrf vpn1 neighbor
Neighbor    Interface             Uptime/Expires      Ver    DR
Address                                                     Priority/Mode
192.168.0.1 Gigabitethernet 0/1 02:34:20/00:01:31  v2     1 / P
20.1.1.1     Tunnel0               01:55:19/00:01:37  v2     1 / P
```

10.3.5 Extranet-MVPN Configuration Instance of Configuring the Receiver MVRF on the Ingress PE

Prerequisite: Multicast VPN has been configured in single AS mode. Extranet-MVPN configuration in the cross-AS topology is the same as that in the single AS topology.

Figure 31 Topology of configuring the receiver MVRF on the ingress PE



Procedure:

- CE1:

Configure the multicast route.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-GigabitEthernet 0/1)# ip address 192.168.0.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
```

- PE1:

Configure the VRF.

Configure an MVRF (vpn1), define the RD value and the RT value, and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 239.1.1.1
Qtech(config-vrf)# end
```

Configure a receiver MVRF (vpn2), import the vpn1 route, and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn2
Qtech(config-vrf)# rd 2:200
Qtech(config-vrf)# route-target import 1:100
Qtech(config-vrf)# mdt default 239.2.2.2
Qtech(config-vrf)# end
```

Configure the multicast route on a private network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip multicast-routing vrf vpn2
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
Qtech(config)# ip pim vrf vpn2 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

The **no switchport** command is used to switch the port mode to **Routed Port** on a switch and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.0.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure the multicast route on a public network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface gigabitethernet 0/1
```

The no switchport command is used to switch the port mode to Routed Port and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP, establish the IBGP session with PE2, and configure the VPNv4 address family.

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.3 remote-as 1
Qtech(config-router)# neighbor 10.1.1.3 update-source loopback 0
Qtech(config-router)# address-family vpnv4
Qtech(config-router-af)# neighbor 10.1.1.3 activate
Qtech(config-router-af)# neighbor 10.1.1.3 send-community extended
Qtech(config-router-af)# exit
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 10.1.1.3 activate
Qtech(config-router-af)# neighbor 10.1.1.3 send-community extended
Qtech(config-router-af)# end
```

- P:

The procedure is similar to the procedure for configuring P on the single-AS multicast VPN.

- PE2:

Configure the VRF.

Configure an MVRF (vpn1), define the RD value and the RT value, import the vpn1 route, and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn2
Qtech(config-vrf)# rd 2:200
Qtech(config-vrf)# route-target both 2:200
Qtech(config-vrf)# route-target import 1:100
Qtech(config-vrf)# mdt default 239.2.2.2
Qtech(config-vrf)# end
```

Configure the multicast route on a private network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn2
Qtech(config)# ip pim vrf vpn2 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

The **no switchport** command is used to switch the port mode to **Routed Port** on a switch and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn2
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.1.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure the multicast route on a public network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface gigabitethernet 0/1
```

The **no switchport** command is used to switch the port mode to **Routed Port** and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 3.3.3.3 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.3 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP, establish the IBGP session with PE2, and configure the VPNv4 address family.

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.1 remote-as 1
Qtech(config-router)# neighbor 10.1.1.1 update-source loopback 0
Qtech(config-router)# address-family vpnv4
Qtech(config-router-af)# neighbor 10.1.1.1 activate
Qtech(config-router-af)# neighbor 10.1.1.1 send-community extended
Qtech(config-router-af)# exit
Qtech(config-router)# address-family ipv4 mdt
Qtech(config-router-af)# neighbor 10.1.1.1 activate
Qtech(config-router-af)# neighbor 10.1.1.1 send-community extended
Qtech(config-router-af)# end
```

- CE2:

Configure the multicast route.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 192.168.0.1
```

```
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-GigabitEthernet 0/1)# ip address 192.168.1.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
```

- Verify the configuration result.

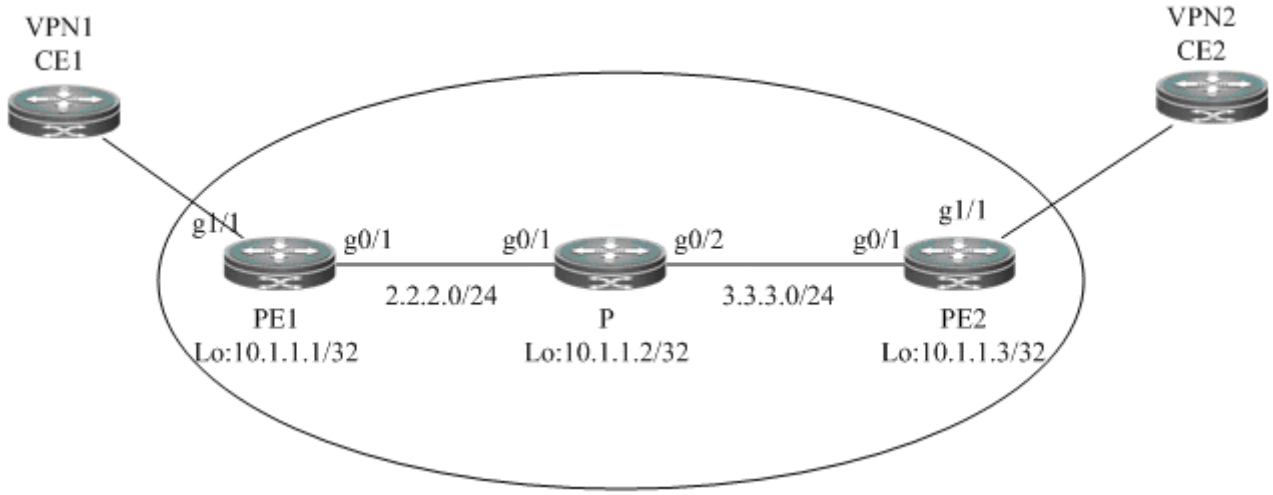
Run the **show ip mroute** command on PE1 to check the establishment of each VPN multicast forwarding table.

```
Qtech# show ip mroute vrf vpn1

IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed, R - RPT, S - SPT
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(1.1.1.1, 224.1.1.1), uptime 00:00:32, stat expires 00:02:58
Owner PIMSM, Flags: TFR
  Incoming interface: GigabitEthernet 1/1
  Outgoing interface list:

Extranet receivers in vrf vpn2:
(1.1.1.1, 224.1.1.1), 00:00:32/stopped

Qtech# show ip mroute vrf vpn2

IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed, R - RPT, S - SPT
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(1.1.1.1, 224.1.1.1), uptime 00:00:32, stat expires 00:02:58
Owner PIMSM, Flags: TFR
  Incoming interface: using vrf vpn1
  Outgoing interface list:
Tunnel 0 (1)
```

10.3.6 Extranet-MVPN Configuration Instance of Configuring the Source MVRF on the Egress PE

Prerequisite: Multicast VPN has been configured in single AS mode. Extranet-MVPN configuration in the cross-AS topology is the same as that in the single AS topology.

Figure 32 Topology of configuring the receiver MVRF on the ingress PE



Procedure:

- CE1:

Configure the multicast route.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-GigabitEthernet 0/1)# ip address 192.168.0.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
```

- PE1:

Configure the VRF.

Configure an MVRF (vpn1), define the RD value and the RT value, and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 239.1.1.1
Qtech(config-vrf)# end
```

Configure the multicast route on a private network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

The **no switchport** command is used to switch the port mode to **Routed Port** on a switch and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn1
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.0.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure the multicast route on a public network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface gigabitethernet 0/1
```

The no switchport command is used to switch the port mode to Routed Port on a switch and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 2.2.2.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP, establish the IBGP session with PE2, and configure the VPNv4 address family.

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.3 remote-as 1
Qtech(config-router)# neighbor 10.1.1.3 update-source loopback 0
Qtech(config-router)# address-family vpnv4
Qtech(config-router-af)# neighbor 10.1.1.3 activate
Qtech(config-router-af)# neighbor 10.1.1.3 send-community extended
Qtech(config-router-af)# end
```

- P:

The procedure is similar to the procedure for configuring P on the single-AS multicast VPN.

- PE2:

Configure the VRF.

Configure an MVRF (vpn2), define the RD value and the RT value, and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn2
Qtech(config-vrf)# rd 2:200
Qtech(config-vrf)# route-target both 2:200
Qtech(config-vrf)# mdt default 239.2.2.2
Qtech(config-vrf)# end
```

Configure a source MVRF (vpn1), define the RD value and the RT value, and configure Default-MDT.

```
Qtech# configure terminal
Qtech(config)# ip vrf vpn1
Qtech(config-vrf)# rd 1:100
Qtech(config-vrf)# route-target both 1:100
Qtech(config-vrf)# mdt default 239.1.1.1
Qtech(config-vrf)# end
```

Specify the VRF to be selected by the static multicast route.

```
Qtech# configure terminal
Qtech(config)# ip mroute vrf vpn2 192.168.0.0 255.255.255.0 fallback-lookup vrf vpn1
```

Configure the multicast route on a private network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing vrf vpn1
Qtech(config)# ip multicast-routing vrf vpn2
Qtech(config)# ip pim vrf vpn1 rp-address 192.168.0.1
Qtech(config)# ip pim vrf vpn2 rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 1/1
```

The **no switchport** command is used to switch the port mode to **Routed Port** on a switch and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 1/1)# no switchport
Qtech(config-GigabitEthernet 1/1)# ip vrf forwarding vpn2
Qtech(config-GigabitEthernet 1/1)# ip address 192.168.1.2 255.255.255.0
Qtech(config-GigabitEthernet 1/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 1/1)# end
```

Configure the multicast route on a public network.

```
Qtech# configure terminal
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 10.1.1.2
Qtech(config)# interface gigabitethernet 0/1
```

The no switchport command is used to switch the port mode to Routed Port on a switch and is inapplicable to a router. Therefore, it does not need to be run on a router.

```
Qtech(config-GigabitEthernet 0/1)# no switchport
Qtech(config-GigabitEthernet 0/1)# ip address 3.3.3.3 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
Qtech(config-GigabitEthernet 0/1)# exit
Qtech(config)# interface loopback 0
Qtech(config-Loopback 0)# ip address 10.1.1.3 255.255.255.255
Qtech(config-Loopback 0)# ip pim sparse-mode
Qtech(config-Loopback 0)# end
```

Configure BGP, establish the IBGP session with PE2, and configure the VPNv4 address family.

```
Qtech# configure terminal
Qtech(config)# router bgp 1
Qtech(config-router)# neighbor 10.1.1.1 remote-as 1
Qtech(config-router)# neighbor 10.1.1.1 update-source loopback 0
Qtech(config-router)# address-family vpnv4
Qtech(config-router-af)# neighbor 10.1.1.1 activate
Qtech(config-router-af)# neighbor 10.1.1.1 send-community extended
Qtech(config-router-af)# end
```

- CE2:

Configure the multicast route.

```
Qtech# configure terminal
```

```
Qtech(config)# ip multicast-routing
Qtech(config)# ip pim rp-address 192.168.0.1
Qtech(config)# interface gigabitethernet 0/1
Qtech(config-GigabitEthernet 0/1)# ip address 192.168.1.1 255.255.255.0
Qtech(config-GigabitEthernet 0/1)# ip pim sparse-mode
```