# PIM



## Feature history for PIM

This table provides release and platform support information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature name and description | Supported platform |
|---|---|---|
| **Cisco IOS XE 17.18.1** | PIM: PIM is IP routing protocol-independent and operates independently of any specific unicast routing protocol. | Cisco C9350 Series Smart Switches<br>Cisco C9610 Series Smart Switches |

## Understand PIM

The Protocol Independent Multicast (PIM) protocol maintains the current IP multicast service mode of receiver-initiated membership. PIM operates independently of any specific unicast routing protocol. It is IP routing protocol-independent. PIM uses available unicast routing protocols to populate the unicast routing table, including Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), ECMP, and static routes. PIM uses unicast routing information to perform the multicast forwarding function.

Although PIM is called a multicast routing protocol, it actually uses the unicast routing table to perform the reverse path forwarding (RPF) check function instead of building up a completely independent multicast routing table. PIM differs from other routing protocols as it does not send or receive routing updates between routers.

PIM is defined in RFC 4601, Protocol Independent Multicast - Sparse Mode (PIM-SM).

# PIM versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution function that enables routers and multilayer switches to dynamically learn the group-to-RP mappings.
- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages sent to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

# Multicast source discovery protocol

Multicast Source Discovery Protocol (MSDP) is used for inter-domain source discovery when PIM SM is used. Each PIM administrative domain has its own RP. MSDP signals new sources between RPs across different domains.

When RP in a domain receives a PIM register message for a new source, with MSDP configured it sends a new source-active (SA) message to all its MSDP peers in other domains. Each MSDP peer floods the SA message from the originating RP. The MSDP peers install this SA message in their MSDP sa-cache.

If RPs in other domains have join requests for the group's SA message (shown by a (*,G) entry with a non-empty outgoing interface list), the domain is interested in the group. The RP then triggers an (S,G) join toward the source.

# PIM sparse mode

PIM sparse mode (PIM-SM) is a multicast routing protocol designed to efficiently route IP multicast traffic in networks where receivers are sparsely distributed. Unlike dense mode protocols that flood multicast traffic to all parts of the network initially, PIM Sparse Mode uses a more controlled approach to conserve bandwidth and resources.

PIM-SM uses a pull model to deliver multicast traffic. Only network segments with active receivers that have explicitly requested the data will receive the traffic.

Sparse mode interfaces are added to the multicast routing table when Join messages arrive from downstream routers or when a connected member is on the interface. When forwarding from a LAN, sparse mode operation happens only if an RP is recognized for the group. If so, the packets are encapsulated and sent toward the RP. When there is sufficient multicast traffic from a source, the receiver's first hop router may send Join messages toward the source to create a source-based distribution tree.

PIM-SM distributes information about active sources by forwarding data packets on the shared tree. Because PIM-SM uses shared trees (at least, initially), it requires the use of a rendezvous point (RP). You must configure the RP in the network.

In sparse mode, routers do not forward multicast packets for a group unless an explicit request for traffic is received. When hosts join a multicast group, the directly connected routers send PIM Join messages toward the RP. The RP keeps track of multicast groups. Hosts that send multicast packets are registered with the RP by the first hop router of that host. The RP then sends Join messages toward the source, and at this point, packets are forwarded on a shared distribution tree. If the multicast traffic from a specific source is sufficient, the first hop router of the host may send Join messages toward the source to build a source-based distribution tree.

Sources register with the RP and then data is forwarded down the shared tree to the receivers. Edge routers learn about a particular source when they receive data packets that travel on the shared tree from that source through the RP. The edge router then sends PIM (S,G) Join messages toward that source. Each router on the reverse path compares the RP address's unicast metric to the source address's metric. If the metric for the source address is better, it will forward a PIM (S,G) Join message toward the source. If the metric for the RP is the same or better, then the PIM (S,G) Join message will be sent in the same direction as the RP. In this case, the shared tree and the source tree would be considered congruent.

If the shared tree is not an optimal path between the source and the receiver, the routers dynamically create a source tree and stop traffic from flowing down the shared tree. This behavior is the default behavior in software. Network administrators can force traffic to stay on the shared tree by using the **ip pim spt-threshold infinity** command.

PIM-SM scales well to a network of any size, including those with WAN links. The explicit join mechanism prevents unwanted traffic from flooding the WAN links.

# PIM stub routing

The PIM stub routing feature moves routed traffic closer to the end user to reduce resource usage. This feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces: uplink PIM interfaces and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic; it only passes and forwards IGMP traffic.

In a network using PIM stub routing, IP traffic to the user must pass through a device configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast (IGMP) receivers and sources are permitted in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

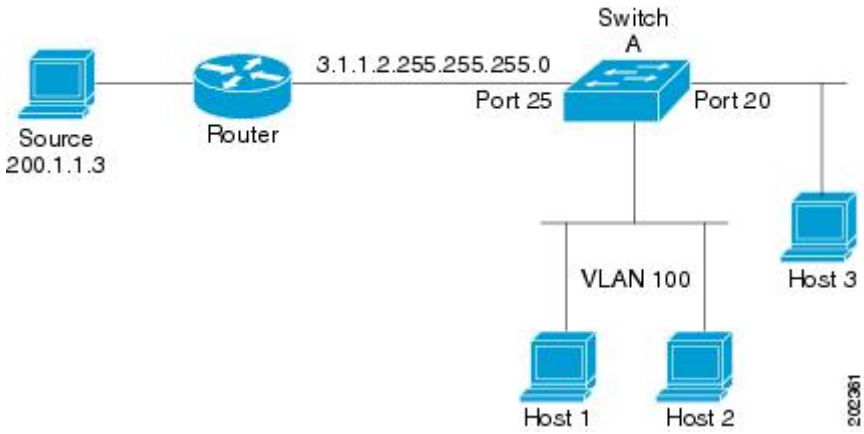
When using PIM stub routing, you should configure the distribution and remote routers to use IP multicast routing. Configure only the device as a PIM stub router. The device does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the device. The device uplink port cannot be used with SVIs. If you need PIM for an SVI uplink port, you should upgrade to the Network Advantage license.

**Note** You must also configure EIGRP stub routing when configuring PIM stub routing on the device.

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM asset and designated router election mechanisms are not supported on the PIM passive interfaces. Only the nonredundant access router topology is supported by the PIM stub feature. By using a nonredundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

*Figure 1: PIM stub router configuration*

In this figure, the Device A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source 200.1.1.3.




# Rendezvous points

A rendezvous point (RP) is a role that a device performs when operating in PIM-SM. An RP is required only in networks running PIM SM. In the PIM-SM model, traffic is forwarded only to network segments with active receivers that have explicitly requested multicast data.

An RP acts as the meeting place for sources and receivers of multicast data. In a PIM-SM network, sources must send their traffic to the RP. This traffic is then forwarded to receivers down a shared distribution tree. By default, when the first hop device of the receiver learns about the source, it will send a Join message directly to the source, creating a source-based distribution tree from the source to the receiver. This source tree does not include the RP unless the RP is located within the shortest path between the source and receiver.

In most cases, the placement of the RP in the network is not a complex decision. By default, the RP is needed only to start new sessions with sources and receivers. Consequently, the RP experiences little overhead from traffic flow or processing. In PIM version 2, the RP performs less processing than in PIM version 1. This occurs because sources need only periodically register with the RP to create state.

## Auto-RP

Auto-RP is a PIM-SM feature that:

- automates the distribution of group-to-RP mappings in a PIM network,
- allows easy configuration of multiple RPs to serve different groups and enables load splitting, and
- prevents inconsistent, manual RP configurations that may cause connectivity issues.

In the initial version of PIM-SM, static RP configuration required manual RP address entry on leaf routers. Auto-RP simplifies this by allowing designated RP-mapping agents to manage announcements and resolve conflicts, enabling automatic group-to-RP discovery across the network.

The RP-mapping agent receives RP-announcements, adjudicates any discrepancies, and sends consistent mappings to other routers. This automated process is crucial for large, complex networks where manual configuration is tedious.

**Note** If router interfaces are configured in sparse mode, Auto-RP can still be used if all routers are configured with a static RP address for the Auto-RP groups.

To use Auto-RP, designate a router as an RP mapping agent to receive RP announcements and arbitrate conflicts. Thus, all routers automatically discover which RP to use for the groups. The Internet Assigned Numbers Authority (IANA) has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP. One advantage of Auto-RP is that any change to the RP designation must be configured only on the routers that are RPs and not on the leaf routers. Another advantage of Auto-RP is that it offers the ability to scope the RP address within a domain. Scoping can be achieved by defining the time-to-live (TTL) value allowed for the Auto-RP advertisements.

Each method for configuring an RP has strengths, weaknesses, and complexity. In conventional IP multicast network scenarios, use Auto-RP to configure RPs because it is easy to set up, well-tested, and stable. The alternative ways to configure an RP are static RP, Auto-RP, and bootstrap router.

## Auto-RP in a PIM network

Auto-RP automates the distribution of group-to-rendezvous point mappings in a PIM network. To make Auto-RP work, a device must be designated as an RP mapping agent. The agent receives RP announcement messages from the RPs and arbitrates conflicts.

You can automatically discover the RP to use for the groups you support. The IANA has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP.

The mapping agent receives announcements of intention to become the RP from Candidate-RPs. The mapping agent then announces the winner of the RP election. This announcement is made independently of the decisions by the other mapping agents.

## Benefits of Auto-RP in a PIM network

- Auto-RP enables changes to the RP designation to be configured exclusively on RP devices and not on leaf routers.
- Auto-RP allows the scoping of the RP address within a domain.

## Auto-RP sparse-dense mode

An interface configured in sparse-dense mode operates in sparse or dense mode, based on the multicast group's mode. If a multicast group has a known RP, the interface is treated in sparse mode. If a group has no known RP, by default the interface is treated in dense mode and data will be flooded over this interface. Configure all interfaces in sparse-dense mode using the **ip pim sparse-dense-mode** interface configuration command for Auto-RP.

We recommend configuring a sink RP (also known as RP of last resort) to successfully implement Auto-RP and prevent any groups other than 224.0.1.39 and 224.0.1.40 from operating in dense mode. A sink RP is a statically configured RP that may or may not actually exist in the network. Configuring a sink RP does not interfere with Auto-RP operation because, by default, Auto-RP messages supersede static RP configurations. We recommend configuring a sink RP for all possible multicast groups in your network because an unknown or unexpected source can become active. If no RP is configured to limit source registration, the group may revert to dense mode operation and be flooded with data.

# Multicast boundaries

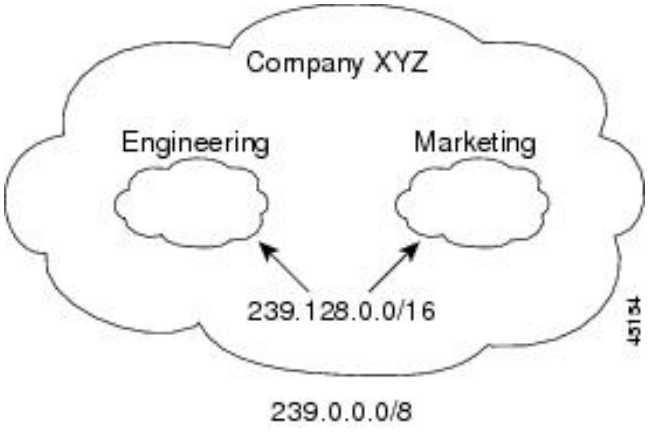
Multicast boundaries are used in multicast routing to control and limit the scope of multicast traffic within a network. They act as filters or boundaries that prevent multicast traffic from crossing certain points in the network, thereby containing multicast traffic to specific areas and reducing unnecessary load on routers and links outside those areas.

Use administratively-scoped boundaries to limit multicast traffic forwarding outside a domain or subdomain. This method uses a specific range of multicast addresses, named administratively-scoped addresses, to create the boundary mechanism. If you configure an administratively-scoped boundary on a routed interface, multicast traffic whose multicast group addresses fall in this range cannot enter or exit this interface, which provides a firewall for multicast traffic in this address range.

**Note** Multicast boundaries and TTL thresholds control the scoping of multicast domains; however, TTL thresholds are not supported by the device. Use multicast boundaries instead of TTL thresholds to limit the forwarding of multicast traffic outside a domain or subdomain.

***Figure 2: Administratively-scoped boundaries***

This figure shows that Company XYZ has an administratively-scoped boundary set for the multicast address range 239.0.0.0/8 on all routed interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have an administratively-scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0 through 239.128.255.255 from entering or leaving their respective networks.




You can define an administratively-scoped boundary on a routed interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is defined, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The IANA has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively-scoped addresses. The addresses can be reused in domains managed by different organizations. The addresses would be considered local, not globally unique.

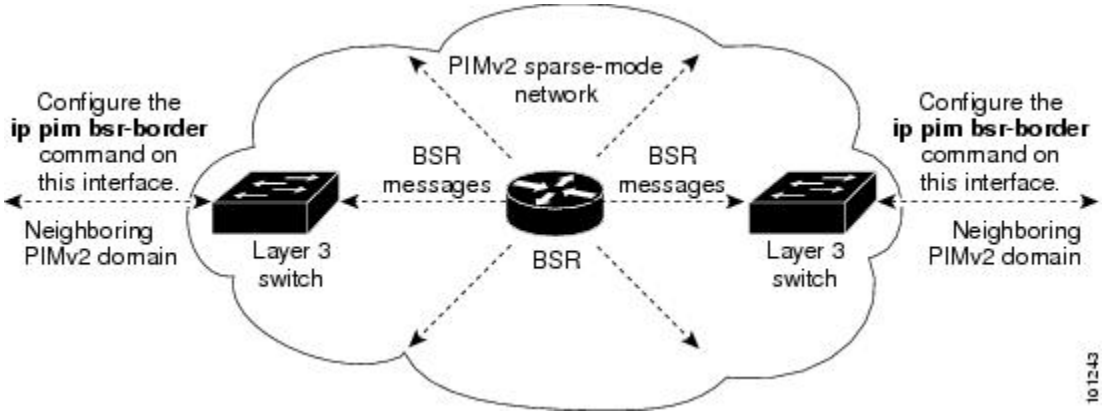
You can configure the **filter-autorp** keyword to examine and filter Auto-RP discovery and announcement messages at the administratively scoped boundary. Any Auto-RP group range announcements from the Auto-RP packets that are denied by the boundary access control list (ACL) are removed. The boundary permits

and passes an Auto-RP group range announcement only if all addresses in the Auto-RP group range are allowed by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

## PIM domain border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain increases. Because two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. If messages leak across the domain borders, it could negatively impact the normal BSR election process by electing a single BSR for all bordering domains and mixing candidate RP advertisements, which may lead to the election of RPs in the incorrect domain.

This figure displays how you can configure the PIM domain border by using the **ip pim bsr-border** command.




## PIMv2 bootstrap router

PIMv2 Bootstrap Router (BSR) is another method to distribute group-to-RP mapping information to all PIM routers and multilayer devices in the network. It eliminates the need to manually configure RP information in every router and switch in the network. Instead of using IP multicast for distributing group-to-RP mapping information, BSR uses hop-by-hop flooding of special BSR messages.

The BSR is elected from a set of candidate routers and switches in the domain that have been configured to function as BSRs. The election mechanism is similar to the root-bridge election mechanism used in bridged LANs. The BSR election is based on the BSR priority of the device contained in the BSR messages that are sent hop-by-hop through the network. Each BSR device examines the message and forwards out all interfaces only the message that has either a higher BSR priority than its BSR priority or the same BSR priority, but with a higher BSR IP address. Using this method, the BSR is elected.

The elected BSR sends BSR messages with a TTL of 1. Neighboring PIMv2 routers or multilayer devices receive the BSR message and multicast it out all other interfaces (except the one on which it was received) with a TTL of 1. In this way, BSR messages travel hop-by-hop throughout the PIM domain. Because BSR messages contain the IP address of the current BSR, the flooding mechanism enables candidate RPs to automatically learn which device is the elected BSR.

Candidate RPs send advertisements to the BSR showing the group range for which they are responsible. The BSR then stores this information in its local candidate-RP cache. The BSR periodically advertises the contents of this cache in BSR messages to all other PIM devices in the domain. These messages travel hop-by-hop through the network to all routers and switches, which store the RP information in the BSR message in their local RP cache. The routers and switches select the same RP for a given group because a common RP hashing algorithm is used by all of them.

# Multicast forwarding

Forwarding of multicast traffic is accomplished by multicast-capable routers. These routers create distribution trees that control the path that IP multicast traffic takes through the network in order to deliver traffic to all receivers.

Multicast traffic flows from the source to the multicast group over a distribution tree. This tree connects all sources to all receivers in the group and may either be shared by all sources (a shared tree) or be built separately for each source (a source tree).

Before describing the structure of source and shared trees, it is helpful to explain the notations that are used in multicast routing tables. These notations include these:

- (S,G) = (unicast source for the multicast group G, multicast group G)
- (*,G) = (any source for the multicast group G, multicast group G)

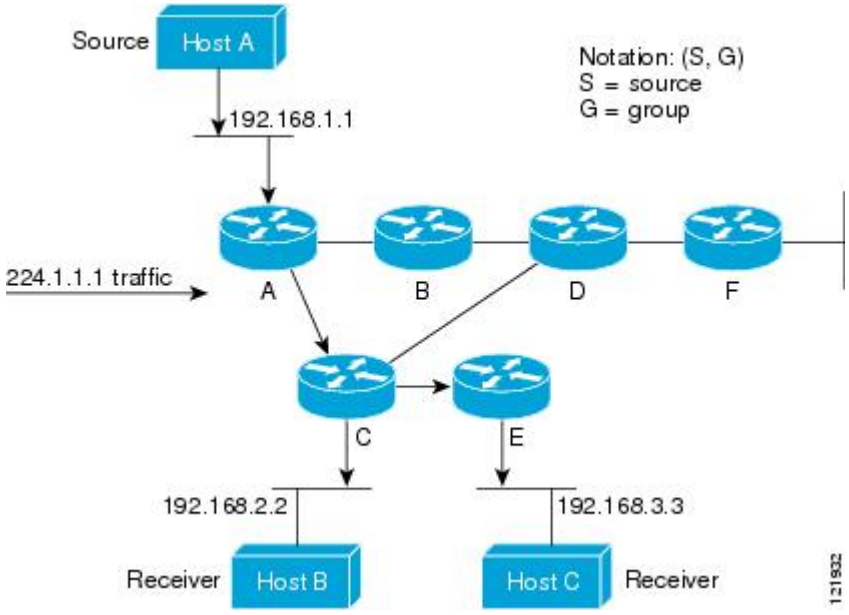
The notation of (S,G), pronounced “S comma G,” enumerates a shortest path tree where S is the IP address of the source and G is the multicast group address.

Shared trees are (*,G) and the source trees are (S,G) and always routed at the sources.

## Multicast distribution source tree

The simplest form of a multicast distribution tree is a source tree. A source tree has its root at the source host and has branches forming a spanning tree through the network to the receivers. This tree is referred to as a shortest path tree (SPT) because it uses the shortest path through the network.

The figure shows an example of an SPT for group 224.1.1.1. It is rooted at the source, Host A, and connects two receivers, Hosts B and C.




Using standard notation, the SPT for the example would be (192.168.1.1, 224.1.1.1).

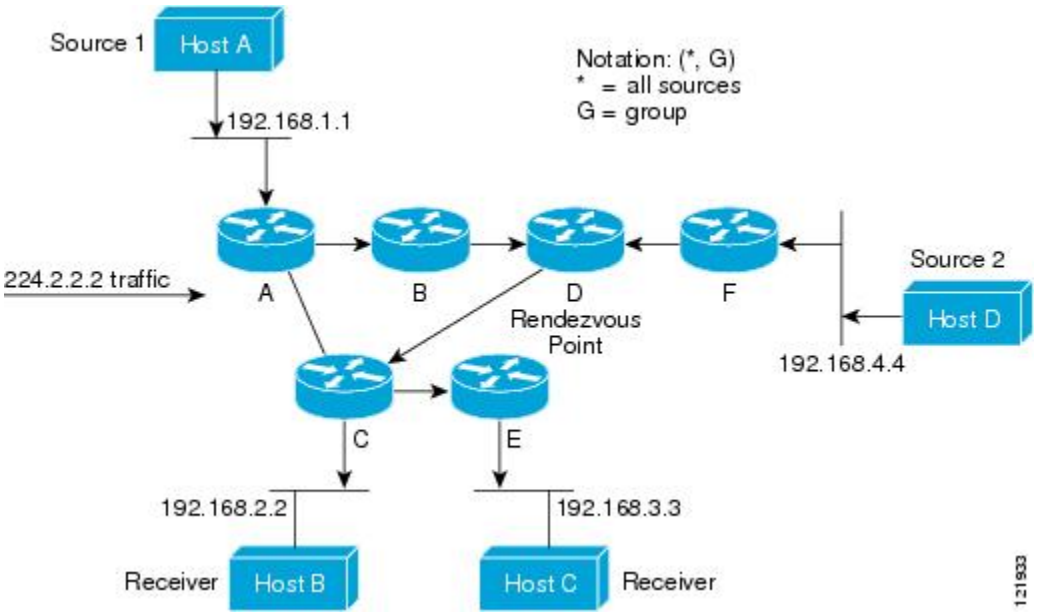
The (S,G) notation implies that a separate SPT exists for each individual source sending to each group--which is correct.

## Multicast distribution shared tree

Source trees have their root at the source, while shared trees use a single common root placed at a chosen point in the network. This shared root is called a rendezvous point (RP).

This figure shows a shared tree for the group 224.2.2.2 with the root located at Router D. This shared tree is unidirectional. Source traffic is sent towards the RP on a source tree. The traffic is then forwarded down the shared tree from the RP to reach all the receivers, except when the receiver is located between the source and the RP in which case it will be serviced directly.

***Figure 3: Shared tree***




In this example, multicast traffic from the sources, Hosts A and D, travels to the root (Router D) and then down the shared tree to the two receivers, Hosts B and C. Because all sources in the multicast group use a common shared tree, a wildcard notation written as (*, G), pronounced "star comma G", represents the tree. In this case, * means all sources, and G represents the multicast group. Therefore, the shared tree shown in the figure would be written as (*, 224.2.2.2).

Both source trees and shared trees are loop-free. Messages are replicated only where the tree branches. Members of multicast groups can join or leave at any time; therefore the distribution trees must be dynamically updated. When all the active receivers on a particular branch stop requesting the traffic for a particular multicast group, the routers prune that branch from the distribution tree and stop forwarding traffic down that branch. If one receiver on that branch becomes active and requests the multicast traffic, the router will dynamically modify the distribution tree and start forwarding traffic again.

## Source tree advantage

Source trees have the advantage of creating the optimal path between the source and the receivers. This advantage reduces network latency for multicast traffic. However, this optimization comes at a cost. The routers must maintain path information for each source. In large networks consisting of many sources and groups, this overhead can quickly become a resource issue for routers. Network designers must consider how the size of the multicast routing table affects memory consumption.

## Shared tree advantage

Shared trees have the advantage of requiring the minimum amount of state in each router. This advantage lowers the overall memory requirements for a network that only allows shared trees. Shared trees can have non-optimal paths between the source and receivers, which may introduce latency in packet delivery. For example, in the figure above the shortest path between Host A (source 1) and Host B (a receiver) would be Router A and Router C. Because we are using Router D as the root for a shared tree, the traffic must traverse Routers A, B, D and then C. Carefully consider where to place the rendezvous point (RP) when implementing a shared tree-only environment.

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans its routing table for the destination address and forwards a single unicast packet toward the destination.

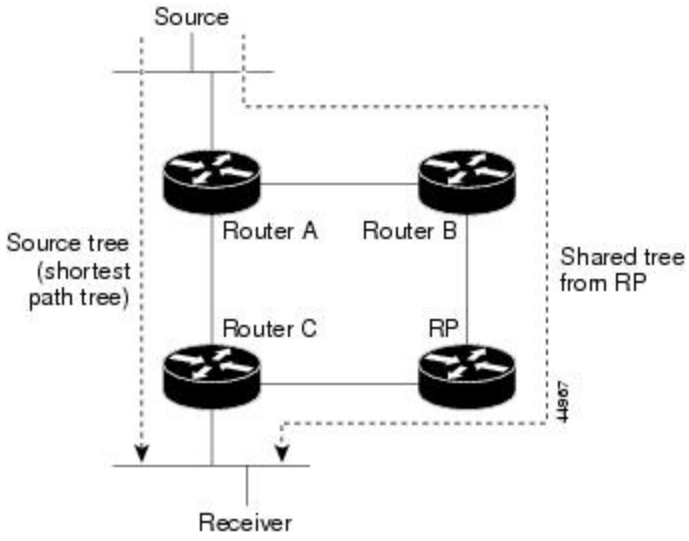
Multicast forwarding involves routing traffic to a group of hosts identified by a multicast group address. The multicast router must determine the upstream direction (toward the source) and the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric), which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF).

## PIM shared tree and source tree

By default, you receive data from senders routed through a single data-distribution tree rooted at the RP.

***Figure 4: Shared tree and source tree (shortest-path tree)***

This figure shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.




If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software transitions to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP adds a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, the first data packet prompts Router C to send a join message to the source.
7. When Router C receives data on (S, G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S, G). The RP triggers a prune message toward the source.

Join and prune messages are sent for both sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

The shared tree is used by multiple sources sending to groups. You can configure the PIM device to stay on the shared tree.

The change from shared to source tree happens when the first data packet arrives at the last-hop router. This change depends upon the threshold that is configured by using the **ip pim spt-threshold** global configuration command.

The shortest-path tree requires more memory than the shared tree but reduces delay. You may want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

## Reverse path forwarding

Reverse Path Forwarding (RPF) forwards multicast traffic away from the source instead of to the receiver. RPF is an algorithm used for forwarding multicast datagrams.

PIM creates a distribution tree using unicast routing information along the reverse path from receivers to the source. The multicast routers then forward packets along the distribution tree from the source to the receivers. RPF is a key concept in multicast forwarding. It enables routers to correctly forward multicast traffic down the distribution tree. RPF makes use of the existing unicast routing table to determine the upstream and downstream neighbors. Routers forward multicast packets only if they are received on the upstream interface. This RPF check helps to guarantee that the distribution tree will be loop-free.

# RPF check

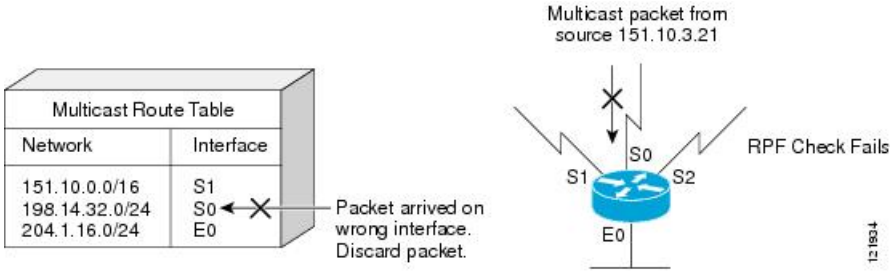
When a multicast packet arrives at a router, the router performs an RPF check on the packet. If the RPF check succeeds, the packet is forwarded. Otherwise, it is dropped.

For traffic flowing down a source tree, the RPF check procedure works as follows:

1. The router looks up the source address in the unicast routing table to determine if the packet has arrived on the interface that is on the reverse path back to the source.
2. If the packet has arrived on the interface leading back to the source, the RPF check succeeds and the packet is forwarded out the interfaces present in the outgoing interface list of a multicast routing table entry.
3. If the RPF check in Step 2 fails, the packet is dropped.

An example of an unsuccessful RPF check is shown in the figure.

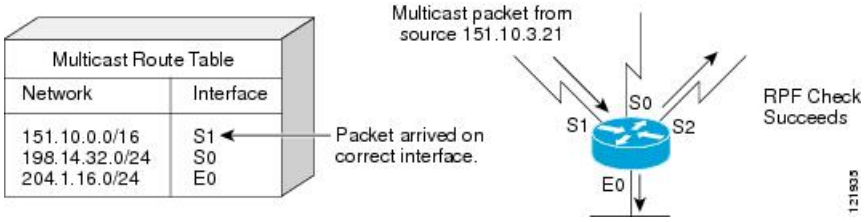
***Figure 5: RPF check fails***




As the figure illustrates, a multicast packet from source 151.10.3.21 is received on serial interface 0 (S0). A check of the unicast route table shows that S1 is the interface this router would use to forward unicast data to 151.10.3.21. Because the packet has arrived on interface S0, the packet is discarded.

The figure shows an example of a successful RPF check.

***Figure 6: RPF check succeeds***




In this example, the multicast packet has arrived on interface S1. The router refers to the unicast routing table and finds that S1 is the correct interface. The RPF check passes, and the packet is forwarded.

PIM uses both source trees and RP-rooted shared trees to forward datagrams. The RPF check is performed differently for each:

- If a PIM router or multilayer switch has a source-tree state (that is, an (S, G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer switch has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP address (which is known when members join the group).

**Note** DVMRP is not supported on the switch.

Sparse-mode PIM uses the RPF lookup function to decide where it needs to send joins and prunes:

- (S, G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

# High availability on PIM

**Note** This feature is applicable to Tenant Routed Multicast (TRM) deployments only.

High availability on PIM feature improves the multicast convergence time after Stateful Switchover (SSO) in a chassis with dual supervisors, StackWise Virtual Link (SVL) and Stacking devices.

After SSO, the PIM protocol generates a new ID and populates multicast route states using the PIM join/prune or IGMP report messages from downstream routers. The Join message populates the Source IP, Multicast Group address, and Outgoing Interface list. For incoming interface, PIM places a route watch request with Rendezvous Point (RP) address as the prefix for (*, G) entries, and source address as the prefix for (S,G) entries.

Once the unicast Routing Information Base (RIB) converges, the route watch update provides the Reverse Path Forward (RPF) interface and RPF neighbor address details. Reverse Path Forward (RPF) checking ensures that multicast traffic arrives on the expected router interface before further processing. If multicast packets fail the RPF check, they are discarded.

Now the PIM has complete multicast route information to program the forwarding table and to send a PIM join/prune message towards upstream PIM neighbor. In multicast deployments where unicast RIB convergence takes more than 3 minutes after SSO, PIM running in the new active device does not have incoming interface to program the forwarding table and RPF neighbor address to send join/prune messages towards upstream PIM neighbor. The time interval of 3 minutes is determined by the default value of the PIM join/prune interval.

If PIM join/prune message is not received, the upstream PIM neighbor removes outgoing interface from the multicast route entry. This affects multicast traffic depending on unicast protocol convergence time. The standby device stores the RPF interface and RPF neighbor address details to improve multicast traffic convergence. After SSO, the new active device uses this stored RPF information to program the forwarding table until the unicast RIB converges. It also sends a join/prune message to the upstream PIM neighbor.

# Default PIM configuration

This table displays the default PIM routing configuration for the device.

***Table 1: Default PIM routing configuration***

| Feature | Default setting |
|---|---|
| Multicast routing | Disabled on all interfaces. |
| PIM version | Version 2. |
| PIM mode | No mode is defined. |
| PIM stub routing | None configured. |
| PIM RP address | None configured. |
| PIM domain border | Disabled. |
| PIM multicast boundary | None. |
| Candidate BSRs | Disabled. |
| Candidate RPs | Disabled. |
| Shortest-path tree threshold rate | 0 kb/s. |
| PIM router query message interval | 30 seconds. |

# Prerequisites for PIM

Decide which PIM mode you will use before starting the PIM configuration process. This is based on the applications you intend to support on your network. Use these guidelines:

- In general, if the application is one-to-many or many-to-many in nature, then PIM-SM can be used successfully.
- Use SSM for optimal one-to-many application performance if IGMP version 3 is supported.

Ensure you meet these conditions before configuring PIM stub routing:

- You must have IP multicast routing configured on both the stub router and the central router. You must also have PIM mode configured on the uplink interface of the stub router.
- You must also configure either Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing or Open Shortest Path First (OSPF) stub routing on the device.
- The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior.

# Restrictions for PIM

These are the PIM configuration restrictions:

- Use ACLs to designate a port as a multicast host port, not a multicast router port. Multicast router control packets received on this port are dropped.
- PIM nonbroadcast multiaccess (NBMA) mode is not supported on an ethernet interface.

## PIMv1 and PIMv2 interoperability

To avoid misconfiguring multicast routing on your device, review the information in this section.

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. You can configure PIM Versions 1 and 2 on different routers and multilayer switches within one network. Internally, all routers and multilayer switches on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer switches in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, is separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF.

**Note** We recommend that you use PIMv2. The BSR function interoperates with Auto-RP on Cisco routers and multilayer switches.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer switch in the group. Not all routers and switches in the domain use the PIMv2 hash function to select multiple RPs.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we recommend using Auto-RP throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP.

## Restrictions for PIM stub routing

- Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. Access domains do not support the PIM protocol.
- In a network using PIM stub routing, IP traffic to the user must pass through a device configured with PIM stub routing.

- The PIM stub feature supports only nonredundant access router topology; redundant PIM stub router topology is unsupported.

# Restrictions for auto-RP and BSR

Consider your network configuration and these restrictions when configuring Auto-RP and BSR:

### Restrictions for auto-RP

These are restrictions for configuring Auto-RP (if used in your network configuration):

- If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.
- If routed interfaces are configured in sparse mode and you enter the **ip pim autorp listener** global configuration command, Auto-RP can still be used even if all devices are not configured with a manual RP address for the Auto-RP groups.

### Restrictions for BSR

These are the restrictions for configuring BSR (if used in your network configuration):

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

### Restrictions and guidelines auto-RP and BSR

These are restrictions for configuring Auto-RP and BSR (if used in your network configuration):

- If your network is all Cisco routers and multilayer switches, you can use either Auto-RP or BSR. The simultaneous deployment of Auto-RP and BSR is not supported.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers, multilayer switches, and non-Cisco routers, both Auto-RP and BSR are required. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure no PIMv1 device is on the path between the BSR and any non-Cisco PIMv2 device.

**Note** There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents messages from reaching routers and multilayer switches across your network. If your network includes a PIMv1 device and Cisco routers and multilayer switches, use Auto-RP.

- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer switch. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.
- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR.

# Configure PIM

This section provides information about the various tasks to configure PIM.

## Enable PIM stub routing

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **interface** *interface-id*<br>**Example:**<br>`Device(config)# interface gigabitethernet 1/0/1` | Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode.<br>• A routed port: A physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command.<br>• An SVI: A VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. |
| **Step 4** | **ip pim passive**<br>**Example:** | Configures the PIM stub feature on the interface. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device(config-if)# ip pim passive` | |
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 6** | **show ip pim interface**<br>**Example:**<br>`Device# show ip pim interface` | (Optional) Displays the PIM stub that is enabled on each interface. |
| **Step 7** | **show ip igmp groups detail**<br>**Example:**<br>`Device# show ip igmp groups detail` | (Optional) Displays the interested clients that have joined the specific multicast source group. |
| **Step 8** | **show ip mroute**<br>**Example:**<br>`Device# show ip mroute` | (Optional) Displays the IP multicast routing table. |
| **Step 9** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 10** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Configure a rendezvous point

A rendezvous point (RP) is required if the interface is in sparse-dense mode and if handling the group as sparse is desired. You can use these methods:

- Manually assign an RP to multicast groups.
- Use a standalone, Cisco-proprietary protocol separate from PIMv1.

- Utilize a standards track protocol in the Internet Engineering Task Force (IETF) by configuring PIMv2 BSR.

**Note** You can use Auto-RP, BSR, or a combination of both, depending on the PIM version that you are running and the types of routers in your network.

## Manually assign an RP to multicast groups

If the rendezvous point (RP) for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you do not need to perform this task for that RP.

Senders of multicast traffic announce their existence through register messages, which are received from the source first-hop router (designated router) and then forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages.

**Note** RPs are not members of the multicast group; they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer switch responds to the group as dense and uses the dense-mode PIM techniques.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **ip pim rp-address** *ip-address* [*access-list-number*] [**override**]<br>**Example:**<br>`Device(config)# ip pim rp-address 10.1.1.1 20 override` | Configures the address of a PIM RP.<br>By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer switches (including the RP).<br>**Note** |

| | Command or Action | Purpose |
|---|---|---|
| | | If there is no RP configured for a group, the device treats the group as dense, using the dense-mode PIM techniques.<br>A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access list conditions specify the groups for which the device is an RP.<br>• For *ip-address*, enter the unicast address of the RP in dotted-decimal notation.<br>• (Optional) For *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups.<br>• (Optional) The **override** keyword indicates that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails. |
| **Step 4** | **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]<br>**Example:**<br>`Device(config)# access-list 25 permit 10.5.0.1 255.224.0.0` | Creates a standard access list, repeating the command as many times as necessary.<br>• For *access-list-number*, enter the access list number specified in Step 2.<br>• The **deny** keyword denies access if the conditions are matched.<br>• The **permit** keyword permits access if the conditions are matched.<br>• For *source*, enter the multicast group address for which the RP should be used.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.<br>The access list is always terminated by an implicit deny statement for everything. |
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 7** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

## Set up Auto-RP in a new internetwork

**Note** If you want to configure a PIM router as the RP for the local group, omit step 3.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command.<br>**Note**<br>This step is not required for spare-dense-mode environments.<br><br>The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is |

| | Command or Action | Purpose |
|---|---|---|
| | | desirable to use a second RP for the local groups. |
| **Step 3** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 4** | **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*<br>**Example:**<br>`Device(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120` | Configures another PIM device to be the candidate RP for local groups.<br>• For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs.<br>• For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255.<br>• For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups.<br>• For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383. |
| **Step 5** | **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]<br>**Example:**<br>`Device(config)# access-list 10 permit 10.10.0.0` | Creates a standard access list, repeating the command as many times as necessary.<br>• For *access-list-number*, enter the access list number specified in Step 3.<br>• The **deny** keyword denies access if the conditions are matched.<br>• The **permit** keyword permits access if the conditions are matched.<br>• For *source*, enter the multicast group address range for which the RP should be used.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation |

| | Command or Action | Purpose |
|---|---|---|
| | | to be applied to the source. Place ones in the bit positions that you want to ignore.<br>**Note** Recall that the access list is always terminated by an implicit deny statement for everything. |
| **Step 6** | **ip pim send-rp-discovery scope** *ttl*<br>**Example:**<br>`Device(config)# ip pim send-rp-discovery scope 50` | Finds a device with stable connectivity and assigns it the role of RP-mapping agent.<br>For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255. |
| **Step 7** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 8** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 9** | **show ip pim rp mapping**<br>**Example:**<br>`Device# show ip pim rp mapping` | Displays active RPs that are cached with associated multicast routing entries. |
| **Step 10** | **show ip pim rp**<br>**Example:**<br>`Device# show ip pim rp` | Displays the information cached in the routing table. |
| **Step 11** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Add Auto-RP to an existing sparse-mode cloud

This section contains suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command.<br>**Note** This step is not required for spare-dense-mode environments.<br>The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups. |
| **Step 3** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 4** | **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*<br>**Example:**<br>`Device(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval` | Configures another PIM device to be the candidate RP for local groups.<br>• For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs.<br>• For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is |

| | Command or Action | Purpose |
|---|---|---|
| | `120` | high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255.<br>• For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups.<br>• For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383. |
| **Step 5** | **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]<br>**Example:**<br>`Device(config)# access-list 10 permit 224.0.0.0 15.255.255.255` | Creates a standard access list, repeating the command as many times as necessary.<br>• For *access-list-number*, enter the access list number specified in Step 3.<br>• The **deny** keyword denies access if the conditions are matched.<br>• The **permit** keyword permits access if the conditions are matched.<br>• For *source*, enter the multicast group address range for which the RP should be used.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.<br>Recall that the access list is always terminated by an implicit deny statement for everything. |
| **Step 6** | **ip pim send-rp-discovery scope** *ttl*<br>**Example:**<br>`Device(config)# ip pim send-rp-discovery scope 50` | Finds a device whose connectivity is not likely to be interrupted, and assigns it the role of RP-mapping agent.<br>For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255. |

| | Command or Action | Purpose |
|---|---|---|
| | | **Note** To remove the device as the RP-mapping agent, use the **no ip pim send-rp-discovery** global configuration command. |
| **Step 7** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 8** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 9** | **show ip pim rp mapping**<br>**Example:**<br>`Device# show ip pim rp mapping` | Displays active RPs that are cached with associated multicast routing entries. |
| **Step 10** | **show ip pim rp**<br>**Example:**<br>`Device# show ip pim rp` | Displays the information cached in the routing table. |
| **Step 11** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

## Prevent join messages to false RPs

Use the **show running-config** privileged EXEC command to determine whether the **ip pim accept-rp** command was configured across the network previously. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In routers or multilayer switches where the **ip pim accept-rp** command is already configured, enter the command again to accept the newly advertised RP.

## Filter incoming RP announcement messages

Add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*<br>**Example:**<br>`Device(config)# ip pim rp-announce-filter rp-list 10 group-list 14` | Filters incoming RP announcement messages.<br>Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default.<br>For **rp-list** *access-list-number*, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the **group-list** *access-list-number* variable. If this variable is omitted, the filter applies to all multicast groups.<br>If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the group-to-RP mapping information. |
| **Step 4** | **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]<br>**Example:**<br>`Device(config)# access-list 10 permit 10.8.1.0 255.255.224.0` | Creates a standard access list, repeating the command as many times as necessary.<br>• For *access-list-number*, enter the access list number specified in Step 2.<br>• The **deny** keyword denies access if the conditions are matched.<br>• The **permit** keyword permits access if the conditions are matched.<br>• Create an access list that specifies from which routers and multilayer switches the mapping agent accepts candidate RP announcements (rp-list ACL). |

| | Command or Action | Purpose |
|---|---|---|
| | | • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL).<br>• For *source*, enter the multicast group address range for which the RP should be used.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.<br>The access list is always terminated by an implicit deny statement for everything. |
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 6** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 7** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Configure PIMv2 BSR

The process for configuring PIMv2 BSR may involve these optional tasks:

- Defining the PIM domain border
- Defining the IP multicast boundary
- Configuring candidate BSRs
- Configuring candidate RPs

## Define the PIM domain border

Perform these steps to configure the PIM domain border. This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **interface** *interface-id*<br>**Example:**<br>`Device(config)# interface gigabitethernet 1/0/1` | Specifies the interface to be configured, and enters interface configuration mode. |
| **Step 4** | **ip pim bsr-border**<br>**Example:**<br>`Device(config-if)# ip pim bsr-border` | Defines a PIM bootstrap message boundary for the PIM domain.<br>Enter this command on each interface that connects to other bordering PIM domains. This command instructs the device to neither send nor receive PIMv2 BSR messages on this interface.<br>**Note**<br>To remove the PIM border, use the **no ip pim bsr-border** interface configuration command. |
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 6** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 7** | **copy running-config startup-config**<br>**Example:** | (Optional) Saves your entries in the configuration file. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device# copy running-config startup-config` | |

## Define the IP multicast boundary

Define a multicast boundary to prevent Auto-RP messages from entering the PIM domain by creating an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **access-list** *access-list-number* **deny** *source* [*source-wildcard*]<br>**Example:**<br>`Device(config)#`<br>`access-list 12 deny 224.0.1.39`<br>`access-list 12 deny 224.0.1.40` | Creates a standard access list, repeating the command as many times as necessary.<br>• For *access-list-number*, the range is 1 to 99.<br>• The **deny** keyword denies access if the conditions are matched.<br>• For *source*, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.<br>The access list is always terminated by an implicit deny statement for everything. |
| **Step 4** | **interface** *interface-id*<br>**Example:** | Specifies the interface to be configured, and enters interface configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device(config)# interface gigabitethernet 1/0/1` | |
| **Step 5** | **ip multicast boundary** *access-list-number*<br>**Example:**<br>`Device(config-if)# ip multicast boundary 12` | Configures the boundary, specifying the access list you created in Step 2. |
| **Step 6** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 7** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 8** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

## Configure candidate BSRs

Configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device# configure terminal` | |
| **Step 3** | **ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]<br>**Example:**<br>`Device(config)# ip pim bsr-candidate gigabitethernet 1/0/3 28 100` | Configures your device to be a candidate BSR.<br>• For *interface-id*, enter the interface on this device from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs.<br>• For *hash-mask-length*, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter.<br>• (Optional) For *priority*, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0. |
| **Step 4** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 5** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 6** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Configure candidate RPs

Configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP is capable of serving the complete IP multicast address space or just a segment of it. Candidate RPs send candidate RP advertisements to the BSR.

This procedure is optional.

### Before you begin

When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer switches where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer switches and with routers from other vendors, any device can be used as an RP.
- In a network including Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure RPs using only Cisco PIMv2 routers and multilayer switches.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]<br>**Example:**<br>`Device(config)# ip pim rp-candidate gigabitethernet 1/0/5 group-list 10` | Configures your device to be a candidate RP.<br>• For *interface-id*, specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs.<br>• (Optional) For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no group-list is specified, the device is a candidate RP for all groups. |
| **Step 4** | **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*] | Creates a standard access list, repeating the command as many times as necessary. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Device(config)# access-list 10 permit 239.0.0.0 0.255.255.255` | • For *access-list-number*, enter the access list number specified in Step 2.<br>• The **deny** keyword denies access under matched conditions. The **permit** keyword grants access under matched conditions.<br>• For *source*, enter the number of the network or host from which the packet is being sent.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.<br>The access list is always terminated by an implicit deny statement for everything. |
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 6** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 7** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Configure sparse mode with Auto-RP

### Before you begin

All access lists that are needed when Auto-RP is configured should be configured prior to beginning the configuration task.

**Note**

- If a group has no known RP when the interface is configured to sparse-dense mode, it is treated as dense mode, causing data to flood the interface. To avoid this data flooding, configure the Auto-RP listener and then configure the interface as sparse mode.
- When configuring Auto-RP, you must either configure the Auto-RP listener feature (Step 5) or specify sparse mode (Step 7).
- When you configure sparse-dense mode, dense mode failover may result in a network dense-mode flood. To avoid this condition, use PIM sparse mode with the Auto-RP listener feature.

Follow this procedure to configure auto-rendezvous point (Auto-RP). Auto-RP can also be optionally used with anycast RP.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **ip multicast-routing**<br>**Example:**<br>`Device(config)# ip multicast-routing` | Enables IP multicast routing. |
| **Step 4** | Either perform Steps 5 through 7 or perform Steps 6 and 8. | -- |
| **Step 5** | **interface** *type number*<br>**Example:**<br>`Device(config)# interface Gigabitethernet 1/0/0` | Selects an interface that is connected to hosts on which PIM can be enabled. |
| **Step 6** | **ip pim sparse-mode**<br>**Example:**<br>`Device(config-if)# ip pim sparse-mode` | Enables PIM sparse mode on an interface. When configuring Auto-RP in sparse mode, you must also configure the Auto-RP listener in the next step.<br>• Skip this step if you are configuring sparse-dense mode in Step 8. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 7** | **exit**<br>**Example:**<br>`Device(config-if)# exit` | Exits interface configuration mode and returns to global configuration mode. |
| **Step 8** | Repeat Steps 1 through 9 on all PIM interfaces. | -- |
| **Step 9** | **ip pim send-rp-announce** {*interface-type interface-number* \| *ip-address*} **scope** *ttl-value* [**group-list** *access-list*] [**interval** *seconds*]<br>**Example:**<br>`Device(config)# ip pim send-rp-announce loopback0 scope 31 group-list 5` | Sends RP announcements out all PIM-enabled interfaces.<br>• Perform this step on the RP device only.<br>• Use the*interface-type* and *interface-number* arguments to define which IP address is to be used as the RP address.<br>• Use the*ip-address* argument to specify a directly connected IP address as the RP address.<br>**Note** If the*ip-address* argument is configured for this command, the RP-announce message will be sourced by the interface to which this IP address is connected (that is, the source address in the IP header of the RP-announce message is the IP address of that interface).<br>• This example shows that the interface is enabled with a maximum of 31 hops. The IP address by which the device wants to be identified as RP is the IP address associated with loopback interface 0. Access list 5 describes the groups for which this device serves as RP. |
| **Step 10** | **ip pim send-rp-discovery** [*interface-type interface-number*] **scope** *ttl-value* [**interval** *seconds*]<br>**Example:**<br>`Device(config)# ip pim send-rp-discovery loopback 1 scope 31` | Configures the device to be an RP mapping agent.<br>• Perform this step on RP mapping agent devices or on combined RP/RP mapping agent devices.<br>**Note** Auto-RP allows the RP function to run separately on one device.<br>Alternatively, it can deploy both the RP and RP mapping agent on a combined RP/RP mapping agent device. |

| | Command or Action | Purpose |
|---|---|---|
| | | • Use the optional *interface-type* and *interface-number* arguments to define which IP address is to be used as the source address of the RP mapping agent.<br>• Use the **scope** keyword and *ttl-value* argument to specify the Time-to-Live (TTL) value in the IP header of Auto-RP discovery messages.<br>• Use the optional **interval** keyword and *seconds* argument to specify the interval at which Auto-RP discovery messages are sent.<br>**Note** Lowering the interval at which Auto-RP discovery messages are sent from the default value of 60 seconds results in more frequent floodings of the group-to-RP mappings. In some network environments, the disadvantages of lowering the interval (more control packet overhead) may outweigh the advantages (more frequent group-to-RP mapping updates).<br>• The example shows limiting the Auto-RP discovery messages to 31 hops on loopback interface 1. |
| **Step 11** | **ip pim rp-announce-filter rp-list** *access-list* **group-list** *access-list*<br>**Example:**<br>`Device(config)# ip pim rp-announce-filter rp-list 1 group-list 2` | Filters incoming RP announcement messages sent from candidate RPs (C-RPs) to the RP mapping agent.<br>• Perform this step on the RP mapping agent only. |
| **Step 12** | **interface** *type number*<br>**Example:**<br>`Device(config)# interface gigabitethernet 1/0/0` | Selects an interface that is connected to hosts on which PIM can be enabled. |
| **Step 13** | **ip multicast boundary** *access-list* [**filter-autorp**]<br>**Example:**<br>`Device(config-if)# ip multicast boundary 10 filter-autorp` | Configures an administratively scoped boundary.<br>• Perform this step on the interfaces that are boundaries to other devices.<br>• The access list is not shown in this task. |

| | Command or Action | Purpose |
|---|---|---|
| | | • An access list entry that uses the **deny** keyword creates a multicast boundary for packets that match that entry. |
| **Step 14** | **end**<br>**Example:**<br>`Device(config-if)# end` | Returns to global configuration mode. |
| **Step 15** | **show ip pim autorp**<br>**Example:**<br>`Device# show ip pim autorp` | (Optional) Displays the Auto-RP information. |
| **Step 16** | **show ip pim rp** [**mapping**] [*rp-address*]<br>**Example:**<br>`Device# show ip pim rp mapping` | (Optional) Displays RPs known in the network and shows how the device learned about each RP. |
| **Step 17** | **show ip igmp groups** [*group-name* \| *group-address*\| *interface-type interface-number*] [**detail**]<br>**Example:**<br>`Device# show ip igmp groups` | (Optional) Displays the multicast groups having receivers that are directly connected to the device and that were learned through Internet Group Management Protocol (IGMP).<br>• A receiver must be active on the network at the time that this command is issued in order for receiver information to be present on the resulting display. |
| **Step 18** | **show ip mroute** [*group-address* \| *group-name*] [*source-address* \| *source-name*] [*interface-type interface-number*] [**summary**] [**count**] [**active** *kbps*]<br>**Example:**<br>`Device# show ip mroute cbone-audio` | (Optional) Displays the contents of the IP multicast routing (mroute) table. |

# Delay PIM shortest-path tree

Configure a traffic rate threshold for switching multicast routing from the source tree to the shortest-path tree.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable** | Enables privileged EXEC mode. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Device> enable` | • Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]<br>**Example:**<br>`Device(config)# access-list 16 permit 225.0.0.0 0.255.255.255` | Creates a standard access list.<br>• For *access-list-number*, the range is 1 to 99.<br>• The **deny** keyword denies access if the conditions are matched.<br>• The **permit** keyword permits access if the conditions are matched.<br>• For *source*, specify the multicast group to which the threshold will apply.<br>• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.<br>The access list is always terminated by an implicit deny statement for everything. |
| **Step 4** | **ip pim spt-threshold** {*kbps* \| **infinity**} [**group-list** *access-list-number*]<br>**Example:**<br>`Device(config)# ip pim spt-threshold infinity group-list 16` | Specifies the threshold that must be reached before moving to shortest-path tree (spt).<br>• For *kbps*, specify the traffic rate in kilobits per second. The default is 0 kbps.<br>**Note** Because of device hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967.<br>• Specify **infinity** if you want all sources for the specified group to use the shared tree, never switching to the source tree.<br>• (Optional) For **group-list** *access-list-number*, specify the access list created in Step 2. When the value is 0 or the group list is unused, the threshold applies to all groups. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 6** | **show running-config**<br>**Example:**<br>`Device# show running-config` | Verifies your entries. |
| **Step 7** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Modify the PIM router-query message interval

PIM routers and multilayer switches send PIM router-query messages to determine the designated router (DR) for each LAN segment (subnet). The DR sends IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM-SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

This procedure is optional.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **interface** *interface-id*<br>**Example:**<br>`Device(config)# interface gigabitethernet 1/0/1` | Specifies the interface to be configured, and enters interface configuration mode. |
| **Step 4** | **ip pim query-interval** *seconds*<br>**Example:**<br>`Device(config-if)# ip pim query-interval 45` | Configures the frequency at which the device sends PIM router-query messages.<br>The default is 30 seconds. The range is 1 to 65535. |
| **Step 5** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |
| **Step 6** | **show ip igmp interface** [*interface-id*]<br>**Example:**<br>`Device# show ip igmp interface` | Verifies your entries. |
| **Step 7** | **copy running-config startup-config**<br>**Example:**<br>`Device# copy running-config startup-config` | (Optional) Saves your entries in the configuration file. |

# Enable high availability on PIM using RPF

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device# configure terminal` | |
| **Step 3** | **ip multicast redundancy rpf-sync**<br>**Example:**<br>`Device# ip multicast redundancy rpf-sync` | Synchronizes the RPF information into PIM. RPF sync can also be enabled by enabling the **evpn-mcast** command. |
| **Step 4** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |

# Monitor and troubleshoot PIM

This section provides command information to monitor and troublehsoot PIM configuration.

## Monitor PIM information

Use the privileged EXEC commands in this table to monitor your PIM configurations.

***Table 2: PIM monitoring commands***

| Command | Purpose |
|---|---|
| **show ip pim all-vrfs tunnel** [**tunnel** *tunnel_number* \| **verbose**] | Displays all VRFs. |
| **show ip pim autorp** | Displays global auto-RP information. |
| **show ip pim boundary** | Displays information about mroutes filtered by administratively scoped IPv4 multicast boundaries configured on an interface. |
| **show ip pim interface** | Displays information about interfaces configured for PIM. |
| **show ip pim neighbor** | Displays the PIM neighbor information. |
| **show ip pim rp**[*group-name* \| *group-address*] | Displays RP routers associated with a sparse-mode multicast group. This command is available in all software images. |
| **show ip pim tunnel** [**tunnel** \| **verbose**] | Displays information about PIM tunnel interfaces |

| Command | Purpose |
|---|---|
| **show ip pim vrf** { **word** { **all-vrfs** \| **autorp** \| **boundary** \| **bsr-router** \| **interface** \| **mdt** \| **neighbor** \| **rp** \| **rp-hash** \| **tunnel** } } | Displays the VPN routing/forwarding instance. |
| **show ip igmp groups detail** | Displays the interested clients that have joined the specific multicast source group. |

# Monitor the RP mapping and BSR information

Use the privileged EXEC mode in this table to verify the consistency of group-to-RP mappings:

***Table 3: RP mapping monitoring commands***

| Command | Purpose |
|---|---|
| **show ip pim rp** [ *hostname* or *IP address* \| **mapping** [ *hostname* or *IP address* \| **elected** \| **in-use** ] \| **metric** [ *hostname* or *IP address* ] ] | Displays all available RP mappings and metrics. This tells you how the device learns of the RP (through the BSR or the Auto-RP mechanism).<br>• (Optional) For the *hostname*, specify the IP name of the group about which to display RPs.<br>• (Optional) For the *IP address*, specify the IP address of the group about which to display RPs.<br>• (Optional) Use the **mapping** keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP).<br>• (Optional) Use the **metric** keyword to display the RP RPF metric. |
| **show ip pim rp-hash** *group* | Displays the RP that was selected for the specified group. That is, on a PIMv2 router or multilayer switch, confirms that the same RP is the one that a PIMv1 system chooses. For *group*, enter the group address for which to display RP information. |

Use the privileged EXEC commands in this table to monitor BSR information:

***Table 4: BSR monitoring commands***

| Command | Purpose |
|---|---|
| **show ip pim bsr** | Displays information about the elected BSR. |
| **show ip pim bsr-router** | Displays information about the BSRv2. |

# Troubleshoot PIMv1 and PIMv2 interoperability problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure that the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

# Configuration examples for PIM

This section provides configuration examples for PIM.

## Example: Enable PIM stub routing

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode** enabled. PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20.

```
Device(config)# ip multicast-routing
Device(config)# interface GigabitEthernet3/0/25
Device(config-if)# no switchport
Device(config-if)# ip address 3.1.1.2 255.255.255.0
Device(config-if)# ip pim sparse-dense-mode
Device(config-if)# exit
Device(config)# interface vlan100
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface GigabitEthernet3/0/20
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface vlan100
Device(config-if)# ip address 100.1.1.1 255.255.255.0
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface GigabitEthernet3/0/20
Device(config-if)# no switchport
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# ip pim passive
Device(config-if)# end
```

## Example: Verify PIM stub routing

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** command in privileged EXEC mode:

```
Device# show ip pim interface

Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet3/0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet3/0/20 v2/P 0 30 1 10.1.1.1
```

# Example: Manually assign an RP to multicast groups

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Device(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Device(config)# ip pim rp-address 147.106.6.22 1
```

# Example: Configure auto-RP

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this device serves as RP:

```
Device(config)# ip pim send-rp-announce gigabitethernet1/0/1 scope 31 group-list 5
Device(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

# Example: Sparse mode with auto-RP

This example configures sparse mode with Auto-RP:

```
Device(config)# ip multicast-routing
Device(config)# ip pim autorp listener
Device(config)# ip pim send-rp-announce Loopback0 scope 16 group-list 1
Device(config)# ip pim send-rp-discovery Loopback1 scope 16
Device(config)# no ip pim dm-fallback
Device(config)# access-list 1 permit 239.254.2.0 0.0.0.255
Device(config)# access-list 1 permit 239.254.3.0 0.0.0.255
.
.
.
Device(config)# access-list 10 permit 224.0.1.39
Device(config)# access-list 10 permit 224.0.1.40
Device(config)# access-list 10 permit 239.254.2.0 0.0.0.255
Device(config)# access-list 10 permit 239.254.3.0 0.0.0.255
```

# Example: Define IP multicast boundary to deny auto-RP information

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Device(config)# access-list 1 deny 224.0.1.39
Device(config)# access-list 1 deny 224.0.1.40
Device(config)# access-list 1 permit all
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip multicast boundary 1
```

# Example: Filter incoming RP announcement messages

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Device(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Device(config)# access-list 10 permit host 172.16.5.1
Device(config)# access-list 10 permit host 172.16.2.1
Device(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Device(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

The mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1 for specific multicast groups. These groups must fall in the range of 224.0.0.0 to 239.255.255.255. Otherwise, the mapping agent does not accept candidate RP announcements from any other devices. Furthermore, the mapping agent does not accept announcements from 172.16.5.1 and 172.16.2.1 for groups in the 239.0.0.0 to 239.255.255.255 range. This range is the administratively scoped address range.

# Example: Prevent join messages to false RPs

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Device(config)# ip pim accept-rp 172.10.20.1 1
Device(config)# access-list 1 permit 224.0.1.39
Device(config)# access-list 1 permit 224.0.1.40
```

# Example: Configure candidate BSRs

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip address 172.21.24.18 255.255.255.0
Device(config-if)# ip pim sparse-mode
Device(config-if)# ip pim bsr-candidate gigabitethernet1/0/2 30 10
```

# Example: Configure candidate RPs

This example shows how to configure the device to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Device(config)# ip pim rp-candidate gigabitethernet1/0/2 group-list 4
Device(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```