

Configuring IP Multicast Routing

This chapter describes how to configure IP multicast routing. For a complete description of the IP multicast routing commands in this chapter, refer to the “IP Multicast Routing Commands” chapter of the *Network Protocols Command Reference, Part 1*. To locate documentation of other commands in this chapter, use the command reference master index or search online.

Traditional IP communication allows a host to send packets to a single host (*unicast transmission*) or to all hosts (*broadcast transmission*). IP multicast provides a third scheme, allowing a host to send packets to a subset of all hosts (*group transmission*). These hosts are known as group members.

Packets delivered to group members are identified by a single multicast group address. Multicast packets are delivered to a group using best-effort reliability, just like IP unicast packets.

The multicast environment consists of senders and receivers. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

A multicast address is chosen for the receivers in a multicast group. Senders use that address as the destination address of a datagram to reach all members of the group.

Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time.

How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it may be very short-lived. Membership in a group can change constantly. A group that has members may have no activity.

Routers executing a multicast routing protocol, such as Protocol-Independent Multicast (PIM), maintain forwarding tables to forward multicast datagrams. Routers use the Internet Group Management Protocol (IGMP) to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending IGMP report messages.

Many multimedia applications involve multiple participants. IP multicast is naturally suitable for this communication paradigm.

Cisco's Implementation of IP Multicast Routing

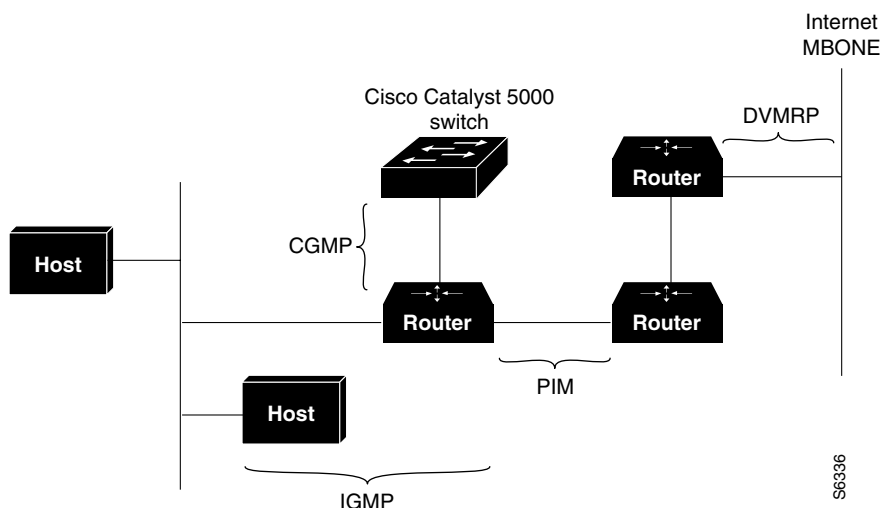
The Cisco IOS software supports the following protocols to implement IP multicast routing:

- Internet Group Management Protocol (IGMP) is used between hosts on a LAN and the router(s) on that LAN to track of which multicast groups the hosts are members.
- Protocol-Independent Multicast (PIM) is used between routers so that they can track which multicast packets to forward to each other and to their directly connected LANs.

- Distance Vector Multicast Routing Protocol (DVMRP) is the protocol used on the MBONE (the multicast backbone of the Internet). The Cisco IOS software supports PIM-to-DVMRP interaction.
- Cisco Group Management Protocol (CGMP) is a protocol used on routers connected to Cisco Catalyst switches to perform tasks similar to those performed by IGMP.

Figure 34 shows where these protocols operate within the IP multicast environment. The protocols are further described after the figure.

Figure 34 IP Multicast Routing Protocols



Internet Group Management Protocol

IP hosts use Internet Group Management Protocol (IGMP) to report their group membership to directly connected multicast routers. IGMP is an integral part of IP. IGMP is defined in RFC 1112, *Host Extensions for IP Multicasting*.

IGMP uses group addresses, which are Class D IP addresses. The high-order four bits of a Class D address are 1110. This means that host group addresses can be in the range 224.0.0.0 to 239.255.255.255. The address 224.0.0.0 is guaranteed not to be assigned to any group. The address 224.0.0.1 is assigned to all systems on a subnet. The address 224.0.0.2 is assigned to all routers on a subnet.

Protocol-Independent Multicast Protocol

The Protocol-Independent Multicast (PIM) protocol maintains the current IP multicast service mode of receiver-initiated membership. It is not dependent on a specific unicast routing protocol.

PIM is defined in the following IETF Internet drafts:

- *Protocol Independent Multicast (PIM): Motivation and Architecture*
- *Protocol Independent Multicast (PIM), Dense Mode Protocol Specification*
- *Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification*

- *IGMP Router Extensions for Routing to Dense Multicast Groups*
- *IGMP Router Extensions for Routing to Sparse Multicast Groups*

PIM can operate in dense mode, sparse mode, or sparse-dense mode.

In dense mode, a router assumes that all other routers want to forward multicast packets for a group. If a router receives a multicast packet and has no directly connected members or PIM neighbors present, a Prune message is sent back to the source. Subsequent multicast packets are not flooded to this router on this pruned branch. PIM builds source-based multicast distribution trees.

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

Distance Vector Multicast Routing Protocol

Cisco routers run PIM, and know enough about Distance Vector Multicast Routing Protocol (DVMRP) to successfully forward multicast packets to and receive packets from a DVMRP neighbor. It is also possible to propagate DVMRP routes into and through a PIM cloud. However, PIM only uses this information. Cisco routers do not implement DVMRP to forward multicast packets.

DVMRP builds a parent-child database using a constrained multicast model to build a forwarding tree rooted at the source of the multicast packets. Multicast packets are initially flooded down this source tree. If redundant paths are on the source-tree, packets are not forwarded along those paths. Forwarding occurs until Prune messages are received on those parent-child links, which further constrain the broadcast of multicast packets.

DVMRP is implemented in the equipment of many vendors and is based on the public-domain mrouterd program.

The Cisco IOS software supports dynamic discovery of DVMRP routers and can interoperate with them over traditional media (such as Ethernet and FDDI), or over DVMRP-specific tunnels.

Cisco Group Management Protocol (CGMP)

Cisco Group Management Protocol (CGMP) is a protocol used on routers connected to Cisco Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary because the Catalyst switch cannot tell the difference between IP multicast data packets and IGMP Report messages, which are both MAC-level addressed to the same group address.

Basic IP Multicast Routing Tasks

The IP multicast routing tasks are divided into basic and advanced tasks, which are discussed in the following sections. The first two basic tasks are required to configure IP multicast routing; the remaining basic and advanced tasks are optional.

- Enable IP Multicast Routing
- Enable PIM on an Interface

- Configure Auto-RP
- Configure IGMP Features
- Configure the TTL Threshold
- Disable Fast Switching of IP Multicast
- Configure sdr Listener Support
- Configure Basic DVMRP Interoperability Features
- Enable the Functional Address for IP Multicast over Token Ring LANs

Advanced IP Multicast Routing Tasks

Advanced, optional IP multicast routing tasks are described in the following sections:

- Configure Advanced PIM Features
- Configure Advanced DVMRP Interoperability Features
- Configure an IP Multicast Static Route
- Control the Transmission Rate to a Multicast Group
- Configure RTP Header Compression
- Configure IP Multicast over ATM Point-to-Multipoint Virtual Circuits
- Configure an IP Multicast Boundary
- Configure an Intermediate IP Multicast Helper
- Store IP Multicast Headers
- Enable CGMP
- Configure Stub IP Multicast Routing
- Load Split IP Multicast Traffic across Equal-Cost Paths
- Monitor and Maintain IP Multicast Routing

See the “IP Multicast Configuration Examples” at the end of this chapter for examples of multicast routing configurations.

Enable IP Multicast Routing

Enabling IP multicast routing allows the Cisco IOS software to forward multicast packets. To enable IP multicast routing on the router, perform the following task in global configuration mode:

Task	Command
Enable IP multicast routing.	ip multicast-routing

Enable PIM on an Interface

Enabling PIM on an interface also enables IGMP operation on that interface. An interface can be configured to be in dense mode, sparse mode, or sparse-dense mode. The mode determines how the router populates its multicast routing table and how the router forwards multicast packets it receives from its directly connected LANs. You must enable PIM in one of these modes for an interface to perform IP multicast routing.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic Join messages are received from downstream routers, or when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router may send joins toward the source to build a source-based distribution tree.

There is no default mode setting. By default, multicast routing is disabled on an interface.

Enable Dense Mode

To configure PIM on an interface to be in dense mode, perform the following task in interface configuration mode:

Task	Command
Enable dense-mode PIM on the interface.	ip pim dense-mode

See the “PIM Dense Mode Example” section at the end of this chapter for an example of how to configure a PIM interface in dense mode.

Enable Sparse Mode

To configure PIM on an interface to be in sparse mode, perform the following task in interface configuration mode:

Task	Command
Enable sparse-mode PIM on the interface.	ip pim sparse-mode

See the “PIM Sparse Mode Example” section at the end of this chapter for an example of how to configure a PIM interface in sparse mode.

Enable Sparse-Dense Mode

If you configure either **ip pim sparse-mode** or **ip pim dense-mode**, then sparseness or denseness is applied to the interface as a whole. However, some environments might require PIM to run in a single region in sparse mode for some groups and in dense mode for other groups.

An alternative to enabling only dense mode or only sparse mode is to enable sparse-dense mode. In this case, the interface is treated as dense mode if the group is in dense mode; the interface is treated in sparse mode if the group is in sparse mode. You must have an RP if the interface is in sparse-dense mode, and you want to treat the group as a sparse group.

If you configure sparse-dense mode, the idea of sparseness or denseness is applied to the group on the router, and the network manager should apply the same concept throughout the network.

Another benefit of sparse-dense mode is that Auto-RP information can be distributed in a dense-mode manner; yet, multicast groups for user groups can be used in a sparse-mode manner. Thus, there is no need to configure a default RP at the leaf routers.

When an interface is treated in dense mode, it is populated in a multicast routing table's outgoing interface list when either of the following is true:

- There are members or DVMRP neighbors on the interface.
- There are PIM neighbors and the group hasn't been pruned.

When an interface is treated in sparse mode, it is populated in a multicast routing table's outgoing interface list when either of the following is true:

- There are members or DVMRP neighbors on the interface.
- An explicit Join has been received by a PIM neighbor on the interface.

To enable PIM to operate in the same mode as the group, perform the following task in interface configuration mode:

Task	Command
Enable PIM to operate in sparse or dense mode, depending on the group.	ip pim sparse-dense-mode

Configure a Rendezvous Point (RP)

If you configure PIM to operate in sparse mode, you must also choose one or more routers to be RPs. You do not have to configure the routers to be RPs; they learn this themselves. RPs are used by senders to a multicast group to announce their existence and by receivers of multicast packets to learn about new senders. The Cisco IOS software can be configured so that packets for a single multicast group can use one or more RPs.

You must configure the IP address of RPs in leaf routers only. *Leaf routers* are those routers that are directly connected either to a multicast group member or to a sender of multicast messages.

The RP address is used by first-hop routers to send PIM register messages on behalf of a host sending a packet to the group. The RP address is also used by last-hop routers to send PIM join/prune messages to the RP to inform it about group membership. The RP does not need to know it is an RP. You must configure the RP address only on first-hop and last-hop routers (leaf routers).

A PIM router can be an RP for more than one group. A group can have more than one RP. The conditions specified by the access list determine for which groups the router is an RP.

To configure the address of the RP, perform the following task on a leaf router in global configuration mode:

Task	Command
Configure the address of a PIM rendezvous point (RP).	ip pim rp-address <i>ip-address</i> [<i>access-list-number</i>] [<i>override</i>]

Configure Auto-RP

Auto-RP is a feature that automates the distribution of group-to-RP mappings in a PIM network. This feature has the following benefits:

- It is easy to use multiple RPs within a network to serve different group ranges.
- It allows load splitting among different RPs and arrangement of RPs according to the location of group participants.
- It avoids inconsistent, manual RP configurations that can cause connectivity problems.

Multiple RPs can be used to serve different group ranges or serve as hot backups of each other. To make Auto RP work, a router must be designated as an *RP-mapping agent*, which receives the RP-announcement messages from the RPs and arbitrates conflicts. The RP-mapping agent then sends the consistent group-to-RP mappings to all other routers. Thus, all routers automatically discover which RP to use for the groups they support.

One way to start is to place (preserve) the default RP for all global groups at or near the border router of your routing domain, while placing another RP in a more centrally located router for all local groups using the administratively scoped addresses (239.x.x.x).

Note If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must statically configure an RP as described in the section “Assign an RP to Multicast Groups” later in this chapter.

Set Up Auto-RP in a New Internetwork

You do not need a default RP in this case. Follow the process described in the section “Add Auto-RP to an Existing Sparse-Mode Cloud,” except that you should skip the first step of choosing a default RP.

Add Auto-RP to an Existing Sparse-Mode Cloud

The following sections contain some suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to provide experience and allow minimal disruption of the existing multicast infrastructure.

Choose a Default RP

Sparse-mode environments need a default RP; sparse-dense-mode environments do not. If you have sparse-dense mode configured everywhere, you do not need to choose a default RP.

Adding Auto-RP to a sparse-mode cloud requires a default RP. In an existing PIM sparse mode region, at least one RP is defined across the network that has good connectivity and availability. That is, the **ip pim rp-address** command is already configured on all routers in this network.

Use that RP for the global groups (for example, 224.x.x.x and other global groups). There is no need to reconfigure the group address range that RP serves. RPs discovered dynamically through Auto-RP take precedence over statically configured RPs. Assume it is desirable to use a second RP for the local groups.

Announce the RP and the Group Range it Serves

Find another router to serve as the RP for the local groups. The RP-mapping agent can double as an RP itself. Assign the whole range of 239.x.x.x to that RP, or assign a subrange of that (for example, 239.2.x.x).

To designate that a router is the RP, perform the following task in global configuration mode:

Task	Command
Configure a router to be the RP.	ip pim send-rp-announce <i>type number scope ttl</i> group-list <i>access-list-number</i>

To change the group ranges this RP optimally serves in the future, change the announcement setting on the RP. If the change is valid, all other routers automatically adopt the new group-to-RP mapping.

The following example advertises the IP address of Ethernet 0 as the RP for the administratively scoped groups:

```
ip pim send-rp-announce ethernet0 scope 16 group-list 1
access-list 1 permit 239.0.0.0 0.255.255.255
```

Assign the RP Mapping Agent

The RP mapping agent is the router that sends the authoritative Discovery packets telling other routers which group-to-RP mapping to use. Such a role is necessary in the event of conflicts (such as overlapping group-to-RP ranges).

Find a router whose connectivity is not likely to be interrupted and assign it the role of RP-mapping agent. All routers within *t* hops from the source router receive the Auto-RP Discovery messages. To assign the role of RP mapping agent, in that router perform the following task in global configuration mode:

Task	Command
Assign the RP mapping agent.	ip pim send-rp-discovery scope <i>t</i>

Verify the Group-to-RP Mapping

To see if the group-to-RP mapping has arrived, perform one of the following tasks in EXEC mode on the designated routers:

Task	Command
Display active RPs that are cached with associated multicast routing entries. Information learned by configuration or Auto-RP.	show ip pim rp mapping
Display information actually cached in the routing table.	show ip pim rp [<i>group-name</i> <i>group-address</i>] [<i>mapping</i>]

Start Using IP Multicast

Use your IP multicast application software to start joining and sending to a group.

Prevent Join Messages to False RPs

Note the **ip pim accept-rp** commands previously configured throughout the network. If that command is not configured on any router, this problem can be addressed later. In those routers already configured with **ip pim accept-rp** command, you must specify the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** command.

If all interfaces are in sparse mode, a default configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto RP relies on 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 default RP must be configured, as follows:

```
ip pim accept-rp default RP address 1
access-list 1 permit 224.0.1.39
access-list 1 permit 224.0.1.40
```

Filter Incoming RP Announcement Messages

To filter incoming RP announcement messages, perform the following task in global configuration mode:

Task	Command
Filter incoming RP announcement messages.	ip pim rp-announce-filter rp-list <i>access-list-number</i> group-list <i>access-list-number</i>

Configure IGMP Features

To configure IGMP features, perform the tasks in the following sections:

- Configure a Router to Be a Member of a Group
- Control Access to IP Multicast Groups
- Modify the IGMP Host-Query Message Interval
- Change the IGMP Version
- Change the IGMP Query Timeout
- Change the Maximum Query Response Time
- Configure the Router as a Statically Connected Member

Configure a Router to Be a Member of a Group

Cisco routers can be configured to be members of a multicast group. This is useful for determining multicast reachability in a network. If a device is configured to be a group member and supports the protocol that is being transmitted to the group, it can respond (for example, the **ping** command). The device responds to ICMP echo request packets addressed to a group of which it is a member. Another example is the multicast traceroute tools provided in the Cisco IOS software.

To have the router join a multicast group and enable IGMP, perform the following task in interface configuration mode:

Task	Command
Join a multicast group.	ip igmp join-group <i>group-address</i>

Control Access to IP Multicast Groups

Multicast routers send IGMP host-query messages to determine which multicast groups have members of the router's attached local networks. The routers then forward to these group members all packets addressed to the multicast group. You can place a filter on each interface that restricts the multicast groups that hosts on the subnet serviced by the interface can join.

To filter multicast groups allowed on an interface, perform the following task in interface configuration mode:

Task	Command
Control the multicast groups that hosts on the subnet serviced by an interface can join.	ip igmp access-group <i>access-list-number</i>

Modify the IGMP Host-Query Message Interval

Multicast routers send IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-systems group address of 224.0.0.1 with a TTL of 1.

Multicast routers send host-query messages periodically to refresh their knowledge of memberships present on their networks. If, after some number of queries, the Cisco IOS software discovers that no local hosts are members of a multicast group, the software stops forwarding onto the local network multicast packets from remote origins for that group and sends a prune message upstream toward the source.

Multicast routers elect a PIM designated router for the LAN (subnet). This is the router with the highest IP address. The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router.

By default, the designated router sends IGMP host-query messages once a minute in order to keep the IGMP overhead on hosts and networks very low. To modify this interval, perform the following task in interface configuration mode:

Task	Command
Configure the frequency at which the designated router sends IGMP host-query messages.	ip igmp query-interval <i>seconds</i>

Change the IGMP Version

By default, the router uses IGMP Version 2, which allows such features as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The router does not automatically detect Version 1 systems and switch to Version 1, as did earlier releases of the Cisco IOS software.

Configure the router for Version 1 if your hosts do not support Version 2.

To control which version of IGMP the router uses, perform the following task in interface configuration mode:

Task	Command
Select the IGMP version that the router uses.	ip igmp version {2 1}

Change the IGMP Query Timeout

You can specify the period of time before the router takes over as the querier for the interface, after the previous querier has stopped doing so. By default, the router waits 2 times the query interval controlled by the **ip igmp query-interval** command. After that time, if the router has received no queries, it becomes the querier. This feature requires IGMP Version 2.

To change the query timeout, perform the following task in interface configuration mode:

Task	Command
Set the IGMP query timeout.	ip igmp query-timeout <i>seconds</i>

Change the Maximum Query Response Time

By default, the maximum query response time advertised in IGMP queries is 10 seconds. If the router is using IGMP Version 2, you can change this value. The maximum query response time allows a router to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value allows the router to *prune* groups faster.

To change the maximum query response time, perform the following task in interface configuration mode:

Task	Command
Set the maximum query response time advertised in IGMP queries.	ip igmp query-max-response-time <i>seconds</i>

Configure the Router as a Statically Connected Member

Sometimes either there is no group member on a network segment or a host cannot report its group membership using IGMP. However, you may want multicast traffic to go to that network segment. The following are two ways to pull multicast traffic down to a network segment:

- Use the **ip igmp join-group** command. With this method, the router accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the router from fast switching.
- Use the **ip igmp static-group** command. With this method, the router does not accept the packets itself, but only forwards them. Hence, this method allows fast switching. The outgoing interface appears in the IGMP cache, but the router itself is not a member, as evidenced by lack of an “L” (local) flag in the multicast route entry.

To configure the router itself to be a statically connected member of a group (and allow fast switching), perform the following task in interface configuration mode:

Task	Command
Configure the router as a statically connected member of a group.	ip igmp static-group <i>group-address</i>

Configure the TTL Threshold

The time-to-live (TTL) value controls whether packets are forwarded out of an interface. You specify the TTL value in hops. Only multicast packets with a TTL greater than the interface TTL threshold are forwarded on the interface. The default value is 0, which means that all multicast packets are forwarded on the interface. To change the default TTL threshold value, perform the following task in interface configuration mode:

Task	Command
Configure the TTL threshold of packets being forwarded out an interface.	ip multicast ttl-threshold <i>ttl</i>

Disable Fast Switching of IP Multicast

Fast switching of IP multicast packets is enabled by default on all interfaces (including GRE and DVMRP tunnels), with one exception: It is disabled and not supported over X.25 encapsulated interfaces. Keep the following in mind:

- If fast switching is disabled on an *incoming* interface for a multicast routing table entry, the packet is sent at process level for all interfaces in the outgoing interface list.
- If fast switching is disabled on an *outgoing* interface for a multicast routing table entry, the packet is process-level switched for that interface, but may be fast switched for other interfaces in the outgoing interface list.

Disable fast switching if you want to log debug messages, because when fast switching is enabled, debug messages are not logged.

To disable fast switching of IP multicast, perform the following task in interface configuration mode:

Task	Command
Disable fast switching of IP multicast.	no ip mroute-cache

Configure sdr Listener Support

The tasks in the following sections configure Session Directory Protocol (sdr) listener support:

- Enable sdr Listener Support
- Limit How Long an sdr Cache Entry Exists

Enable sdr Listener Support

The multicast backbone (MBONE) allows efficient, many-to-many communication and is widely used for multimedia conferencing. To help announce multimedia conference sessions and provide the necessary conference setup information to potential participants, the Session Directory Protocol Version 2 (sdr) tool is available. A session directory client announcing a conference session periodically multicasts an announcement packet on a well-known multicast address and port.

To enable session directory listener support, perform the following task in interface configuration mode:

Task	Command
Enable sdr listener support.	ip sdr listen

Limit How Long an sdr Cache Entry Exists

By default, entries are never deleted from the sdr cache. You can limit how long an sdr cache entry stays active in the cache. To do so, perform the following task in global configuration mode:

Task	Command
Limit how long an sdr cache entry stays active in the cache.	ip sdr cache-timeout <i>minutes</i>

Configure Basic DVMRP Interoperability Features

The following sections describe some basic tasks that allow interoperability with DVMRP machines:

- Configure DVMRP Interoperability
- Configure a DVMRP Tunnel
- Advertise Network 0.0.0.0 to DVMRP Neighbors

For more advanced DVMRP features, see the section “Configure Advanced DVMRP Interoperability Features” later in this chapter.

Configure DVMRP Interoperability

Cisco multicast routers using PIM can interoperate with non-Cisco multicast routers that use the Distance Vector Multicast Routing Protocol (DVMRP).

PIM routers dynamically discover DVMRP multicast routers on attached networks. Once a DVMRP neighbor has been discovered, the router periodically transmits DVMRP Report messages advertising the unicast sources reachable in the PIM domain. By default, directly connected subnets and networks are advertised. The router forwards multicast packets that have been forwarded by DVMRP routers and, in turn, forwards multicast packets to DVMRP routers.

You can configure what sources are advertised and what metrics are used by configuring the **ip dvmrp metric** command. You can also direct all sources learned via a particular unicast routing process to be advertised into DVMRP.

The mroute protocol is a public-domain implementation of DVMRP. It is necessary to use mroute Version 3.8 (which implements a nonpruning version of DVMRP). When Cisco routers are directly connected to DVMRP routers or interoperate with DVMRP routers over an MBONE tunnel, DVMRP advertisements produced by the Cisco IOS software can cause older versions of mroute to corrupt their routing tables and those of their neighbors. Any router connected to the MBONE should have an access-list to limit the number of unicast routes that are advertised via DVMRP.

To configure the sources that are advertised and the metrics that are used when transmitting DVMRP Report messages, perform the following task in interface configuration mode:

Task	Command
Configure the metric associated with a set of destinations for DVMRP reports.	ip dvmrp metric <i>metric</i> [list <i>access-list-number</i>] [[<i>protocol process-id</i>] [dvmrp]]

A more sophisticated way to achieve the same results as the preceding command is to use a route map instead of an access list. Thus, you have a finer granularity of control. To subject unicast routes to route-map conditions before being injected into DVMRP, perform the following task in interface configuration mode:

Task	Command
Subject unicast routes to route-map conditions before being injected into DVMRP	ip dvmrp metric <i>metric</i> route-map <i>map-name</i>

Responding to MRINFO Requests

The Cisco IOS software answers mrinfo requests sent by mrouted systems and Cisco routers. The software returns information about neighbors on DVMRP tunnels and all of the router’s interfaces. This information includes the metric (which is always set to 1), the configured TTL threshold, the status of the interface, and various flags. The **mrinfo** command can also be used to query the router itself, as in the following example:

```
mm1-7kd# mrinfo
171.69.214.27 (mm1-7kd.cisco.com) [version cisco 11.1] [flags: PMS]:
171.69.214.27 -> 171.69.214.26 (mm1-r7kb.cisco.com) [1/0/pim/querier]
171.69.214.27 -> 171.69.214.25 (mm1-45a.cisco.com) [1/0/pim/querier]
171.69.214.33 -> 171.69.214.34 (mm1-45c.cisco.com) [1/0/pim]
171.69.214.137 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
171.69.214.203 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
171.69.214.18 -> 171.69.214.20 (mm1-45e.cisco.com) [1/0/pim]
171.69.214.18 -> 171.69.214.19 (mm1-45c.cisco.com) [1/0/pim]
171.69.214.18 -> 171.69.214.17 (mm1-45a.cisco.com) [1/0/pim]
```

See the “DVMRP Interoperability Example” section at the end of this chapter for an example of how to configure a PIM router to interoperate with a DVMRP router.

Configure a DVMRP Tunnel

The Cisco IOS software supports DVMRP tunnels to the MBONE (the multicast backbone of the Internet). You can configure a DVMRP tunnel on a router if the other end is running DVMRP. The software then sends and receives multicast packets over the tunnel. This allows a PIM domain to connect to the DVMRP router in the case where all routers on the path do not support multicast routing. You cannot configure a DVMRP tunnel between two routers.

When a Cisco router runs DVMRP over a tunnel, it advertises sources in DVMRP Report messages much as it does on real networks. In addition, the software caches DVMRP Report messages it receives and uses them in its Reverse Path Forwarding (RPF) calculation. This allows the software to forward multicast packets received over the tunnel.

When you configure a DVMRP tunnel, you should assign a tunnel an address in the following two cases:

- To enable the sending of IP packets over the tunnel
- To indicate whether the Cisco IOS software should perform DVMRP summarization

You can assign an IP address either by using the **ip address** interface configuration command, or by using the **ip unnumbered** interface configuration command to configure the tunnel to be unnumbered. Either of these two methods allows IP multicast packets to flow over the tunnel. The software will not advertise subnets over the tunnel if the tunnel has a different network number from the subnet. In this case, the software advertises only the network number over the tunnel.

To configure a DVMRP tunnel, perform the following tasks in interface configuration mode:

Task	Command
Step 1 Specify a tunnel interface in global configuration mode. This puts the router into interface configuration mode.	interface tunnel <i>number</i>
Step 2 Set the tunnel interface’s source address. This is the IP address of the interface on the router.	tunnel source <i>ip-address</i>

Task	Command
Step 3 Set the tunnel interface's destination address. This is the IP address of the mrouterd multitask router.	tunnel destination <i>ip-address</i>
Step 4 Configure a DVMRP tunnel.	tunnel mode dvmrp
Step 5 Assign an IP address to the interface. or Configure the interface as unnumbered.	ip address <i>address mask</i> ip unnumbered <i>type number</i>
Step 6 Configure PIM on the interface.	ip pim [dense-mode sparse-mode]
Step 7 Configure an acceptance filter for incoming DVMRP reports.	ip dvmrp accept-filter <i>access-list-number</i> [<i>distance</i> neighbor-list <i>access-list-number</i>]

See the "DVMRP Tunnel Example" section at the end of this chapter for an example of how to configure a DVMRP tunnel.

Advertise Network 0.0.0.0 to DVMRP Neighbors

The mrouterd protocol is a public-domain implementation of DVMRP. If your router is a neighbor to an mrouterd Version 3.6 machine, you can configure the Cisco IOS software to advertise network 0.0.0.0 to the DVMRP neighbor. Do not advertise the DVMRP default into the MBONE. You must specify whether only route 0.0.0.0 is advertised or if other routes can also be specified.

To advertise network 0.0.0.0 to DVMRP neighbors on an interface, perform the following task in interface configuration mode:

Task	Command
Advertise network 0.0.0.0 to DVMRP neighbors.	ip dvmrp default-information { originate only }

Enable the Functional Address for IP Multicast over Token Ring LANs

By default, IP multicast datagrams on Token Ring LAN segments used the MAC-level broadcast address 0xFFFF.FFFF.FFFF. That places an unnecessary burden on all devices that do not participate in IP multicast. The IP multicast over Token Ring LANs feature defines a way to map IP multicast addresses to a single Token Ring MAC address.

This feature defines the Token Ring functional address (0xc000.0004.0000) that should be used over Token Ring. A functional address is a severely restricted form of multicast addressing implemented on Token Ring interfaces. Only 31 functional addresses are available. A bit in the destination MAC address designates it as a functional address.

The implementation used by Cisco Systems complies with RFC 1469, IP Multicast over Token-Ring Local Area Networks.

If you configure this feature, IP multicast transmissions over Token Ring interfaces are more efficient than they formerly were. This feature reduces the load on other machines that do not participate in IP multicast because they do not process these packets.

The following restrictions apply to the Token Ring functional address:

- This feature can be configured only on a Token Ring interface.
- Neighboring devices on the Token Ring on which this feature is used should also use the same functional address for IP multicast traffic.

- Because there are a limited number of Token Ring functional addresses, it is possible there are other protocols assigned to the Token Ring functional address 0xc000.0004.0000. Therefore, not every frame sent to the functional address is necessarily an IP multicast frame.

To enable the mapping of IP multicast addresses to the Token Ring functional address 0xc000.0004.0000, perform the following task in interface configuration mode:

Task	Command
Enable the mapping of IP multicast addresses to the Token Ring functional address.	ip multicast use-functional

For an example of configuring the functional address, see the section “Functional Address for IP Multicast over Token Ring LAN Example” at the end of this chapter.

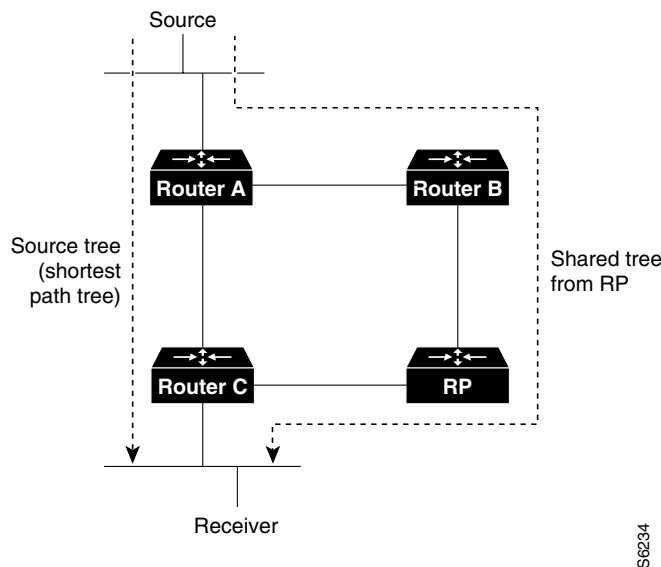
Configure Advanced PIM Features

Perform the optional tasks in the following sections to configure PIM features:

- Understand PIM Shared Tree and Source Tree (Shortest Path Tree)
- Delay the Use of PIM Shortest Path Tree
- Understand Reverse-Path Forwarding (RPF)
- Assign an RP to Multicast Groups
- Increase Control over RPs
- Modify the PIM Router-Query Message Interval
- Enable PIM Nonbroadcast, Multiaccess (NBMA) Mode

Understand PIM Shared Tree and Source Tree (Shortest Path Tree)

By default, members of a group receive data from senders to the group across a single data distribution tree rooted at the rendezvous point (RP). This type of distribution tree is called *shared tree*, as shown in Figure 35. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Figure 35 Shared Tree and Source Tree (Shortest Path Tree)

If the data rate warrants, leaf routers on the shared tree may initiate a switch to 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 Cisco IOS software switches to a source tree upon receiving the first data packet from a source.

The following steps describe the move from shared tree to source tree in more detail:

- Step 1** Receiver joins a group; leaf Router C sends a Join message toward RP.
- Step 2** RP puts link to Router C in its outgoing interface list.
- Step 3** Source sends data; Router A encapsulates data in Register and sends it to RP.
- Step 4** RP forwards data down the shared tree to Router C and sends a Join message toward Source. At this point, data may arrive twice at Router C, once encapsulated and once natively.
- Step 5** When data arrives natively (unencapsulated) at RP, RP sends a Register-Stop message to Router A.
- Step 6** By default, reception of the first data packet prompts Router C to send a Join message toward Source.
- Step 7** When Router C receives data on (S,G), it sends a Prune message for Source up the shared tree.
- Step 8** RP deletes the link to Router C from outgoing interface of (S,G). RP triggers a Prune message toward Source.

Join and Prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM router 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.

Multiple sources sending to groups used the shared tree.

The network manager can configure the router to stay on the shared tree, as described in the section “Delay the Use of PIM Shortest Path Tree.”

Delay the Use of PIM Shortest Path Tree

The switch from shared to source tree happens upon the arrival of the first data packet at the last hop router (Router C in Figure 35). This occurs because the **ip pim spt-threshold** command controls that timing, and its default setting is 0 kbps.

The shortest path tree requires more memory than the shared tree, but reduces delay. You might want to postpone its use. Instead of allowing the leaf router to move to the shortest path tree immediately, 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 router triggers a PIM Join message toward the source to construct a source tree (shortest path tree). If **infinity** is specified, all sources for the specified group use the shared tree, never switching to the source tree.

The group list is a standard access list that controls what groups the shortest path tree threshold applies to. If a value of 0 is specified or the group list is not used, the threshold applies to all groups.

To configure a traffic rate threshold that must be reached before multicast routing is switched from the source tree to the shortest path tree, perform the following task in interface configuration mode:

Task	Command
Specify the threshold that must be reached before moving to shortest path tree (spt).	ip pim spt-threshold { <i>kbps</i> infinity } [group-list <i>access-list-number</i>]

Understand Reverse-Path Forwarding (RPF)

Reverse-Path Forwarding (RPF) is an algorithm used for forwarding multicast datagrams. It functions as follows:

- If a router receives a datagram on an interface it uses to send unicast packets to the source, the packet has arrived on the RPF interface.
- If the packet arrives on the RPF interface, a router forwards the packet out the interfaces present in the outgoing interface list of a multicast routing table entry.
- If the packet does not arrive on the RPF interface, the packet is silently discarded to prevent loops.

PIM uses both source trees and RP-rooted shared trees to forward datagrams; the RPF check is performed differently for each, as follows:

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

Sparse-mode PIM uses the RPF lookup function to determine 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.

DVMRP and dense-mode PIM use only source trees and use RPF as described previously.

Assign an RP to Multicast Groups

If you have configured PIM sparse mode, you must configure a PIM rendezvous point (RP) for a multicast group. An RP can either be configured statically in each box, or learned through a dynamic mechanism. This section explains how to statically configure an RP. If the RP for a group is learned through a dynamic mechanism (such as Auto-RP), you need not perform this task for that RP. You should use Auto-RP, which is described in the section “Configure Auto-RP” earlier in this chapter.

PIM Designated Routers forward data from directly connected multicast sources to the RP for distribution down the shared tree.

Data is forwarded to the RP in one of two ways. It is encapsulated in Register packets and unicast directly to the RP, or, if the RP has itself joined the source tree, it is multicast forwarded per the RPF forwarding algorithm described in the preceding section, “Understand Reverse-Path Forwarding (RPF).” Last-hop routers directly connected to receivers may, at their discretion, join themselves to the source tree and prune themselves from the shared tree.

A single RP can be configured for multiple groups defined by an access list. If there is no RP configured for a group, the router treats the group as dense using the dense-mode PIM techniques.

If a conflict exists between the RP configured with this command and one learned by Auto-RP, the Auto-RP information is used, unless the **override** keyword is configured.

To assign an RP to one or more multicast groups, perform the following task in global configuration mode:

Task	Command
Assign an RP to multicast groups.	ip pim rp-address <i>ip-address</i> [<i>group-access-list-number</i>] [override]

Increase Control over RPs

You can take a defensive measure to prevent a misconfigured leaf router from interrupting PIM service to the remainder of a network. To do so, configure the local router to accept Join messages only if they contain the RP address specified, when the group is in the group range specified by the access list. To configure this feature, perform the following task in global configuration mode:

Task	Command
Control which RPs the local router will accept Join messages to.	ip pim accept-rp { <i>address</i> auto-rp } [<i>access-list-number</i>]

Modify the PIM Router-Query Message Interval

Route-query messages are used to elect a PIM designated router. The designated router is responsible for sending IGMP host-query messages. By default, multicast routers send PIM router-query messages every 30 seconds. To modify this interval, perform the following task in interface configuration mode:

Task	Command
Configure the frequency at which multicast routers send PIM router-query messages.	ip pim query-interval <i>seconds</i>

Enable PIM Nonbroadcast, Multiaccess (NBMA) Mode

PIM nonbroadcast, multiaccess (NBMA) mode allows the Cisco IOS software to replicate packets for each neighbor on the NBMA network. Traditionally, the software replicates multicast and broadcast packets to all “broadcast” configured neighbors. This might be inefficient when not all neighbors want packets for certain multicast groups. NBMA mode enables you to reduce bandwidth on links leading into the NBMA network, as well as CPU cycles in switches and attached neighbors.

Configure this feature on ATM, Frame Relay, SMDS, PRI ISDN, or X.25 networks only, especially when these media do not have native multicast available. Do not use this feature on multicast-capable LANs (such as Ethernet or FDDI).

You should use sparse-mode PIM with this feature. Therefore, when each join is received from NBMA neighbors, PIM stores each neighbor IP address/interface in the outgoing interface list for the group. When a packet is destined for the group, the software replicates the packet and unicasts (data-link unicasts) it to each neighbor that has joined the group.

To enable PIM nonbroadcast, multicaccess mode on your serial link, perform the following task in interface configuration mode:

Task	Command
Enable PIM nonbroadcast, multiaccess mode.	ip pim nbma-mode

Consider the following two factors before enabling PIM NBMA mode:

- If the number of neighbors grows, the outgoing interface list gets large. This costs memory and replication time.
- If the network (Frame Relay, SMDS, or ATM) supports multicast natively, you should use it so that replication is performed at optimal points in the network.

Configure Advanced DVMRP Interoperability Features

Cisco routers run PIM and know enough about DVMRP to successfully forward multicast packets to receivers and receive multicast packets from senders. It is also possible to propagate DVMRP routes into and through a PIM cloud. PIM uses this information; however, Cisco routers do not implement DVMRP to forward multicast packets.

The basic DVMRP features are described in the section “Configure Basic DVMRP Interoperability Features” earlier in this chapter. To configure more advanced DVMRP interoperability features on a Cisco router, perform the optional tasks in the following sections:

- Enable DVMRP Unicast Routing
- Limit the Number of DVMRP Routes Advertised
- Change the DVMRP Route Threshold
- Configure a DVMRP Summary Address
- Disable DVMRP Auto-Summarization
- Add a Metric Offset to the DVMRP Route
- Reject a DVMRP Nonpruning Neighbor
- Configure a Delay between DVMRP Reports

Enable DVMRP Unicast Routing

Since policy for multicast routing and unicast routing require separate topologies, PIM must follow the multicast topology to build loopless distribution trees. Using DVMRP unicast routing, Cisco routers and mroute-based machines exchange DVMRP unicast routes, to which PIM can then Reverse Path Forward.

Cisco routers do not perform DVMRP multicast routing among each other, but they can exchange DVMRP routes. The DVMRP routes provide a multicast topology that may differ from the unicast topology. This allows PIM to run over the multicast topology, thereby allowing sparse-mode PIM over the MBONE topology.

When DVMRP unicast routing is enabled, the router caches routes learned in DVMRP Report messages in a DVMRP routing table. PIM prefers DVMRP routes to unicast routes by default, but that preference can be configured.

DVMRP unicast routing can run on all interfaces, including GRE tunnels. On DVMRP tunnels, it runs by virtue of DVMRP multicast routing. This feature does not enable DVMRP multicast routing among Cisco routers. However, if there is a DVMRP-capable multicast router, the Cisco router will do PIM/DVMRP multicast routing interaction.

To enable DVMRP unicast routing, perform the following task in interface configuration mode:

Task	Command
Enable DVMRP unicast routing.	ip dvmrp unicast-routing

Limit the Number of DVMRP Routes Advertised

By default, only 7000 DVMRP routes will be advertised over an interface enabled to run DVMRP (that is, a DVMRP tunnel, an interface where a DVMRP neighbor has been discovered, or an interface configured to run **ip dvmrp unicast-routing**).

To change this limit, perform the following task in global configuration mode:

Task	Command
Change the number of DVMRP routes advertised over an interface enabled to run DVMRP.	ip dvmrp route-limit <i>count</i>

Change the DVMRP Route Threshold

By default, 10,000 DVMRP routes may be received per interface within a 1-minute interval. When that rate is exceeded, a syslog message is issued, warning that there might be a route surge occurring. The warning is typically used to quickly detect when people have misconfigured their routers to inject a large number of routes into the MBONE.

To change the threshold number of routes that trigger the warning, perform the following task in global configuration mode:

Task	Command
Configure the number of routes that trigger a syslog message.	ip dvmrp routehog-notification <i>route-count</i>

Use the **show ip igmp interface** command to display a running count of routes. When the count is exceeded, “*** ALERT ***” is appended to the line.

Configure a DVMRP Summary Address

You can customize the summarization of DVMRP routes if the default classful auto-summarization does not suit your needs. To summarize such routes, specify a summary address by performing the following task in interface configuration mode:

Task	Command
Specify a DVMRP summary address.	ip dvmrp summary-address <i>address mask</i> [metric <i>value</i>]

Note At least one, more specific route must be present in the unicast routing table before a configured summary address will be advertised.

Disable DVMRP Auto-Summarization

By default, the Cisco IOS software performs some level of DVMRP summarization automatically. Disable this function if you want to advertise all routes, not just a summary. If you configure the **ip dvmrp summary-address** command and did not configure **no ip dvmrp auto-summary**, you get both custom and auto-summaries.

To disable DVMRP auto-summarization, perform the following task in interface configuration mode:

Task	Command
Disable DVMRP auto-summarization.	no ip dvmrp auto-summary

Add a Metric Offset to the DVMRP Route

By default, the router increments by 1 the metric of a DVMRP route advertised in incoming DVMRP reports. You can change the metric if you want to favor or not favor a certain route. The DVMRP metric is a hop-count. Therefore, a very slow serial line of one hop is preferred over a route that is two hops over FDDI or another fast medium.

For example, perhaps a route is learned by Router A and the the same route is learned by Router B with a higher metric. If you want to use the path through Router B because it is a faster path, you can apply a metric offset to the route learned by Router A to make it larger than the metric learned by Router B, allowing you to choose the path through Router B.

To change the default metric, perform the following task in interface configuration mode:

Task	Command
Change the metric added to DVMRP routes advertised in incoming reports.	ip dvmrp metric-offset [in out] <i>increment</i>

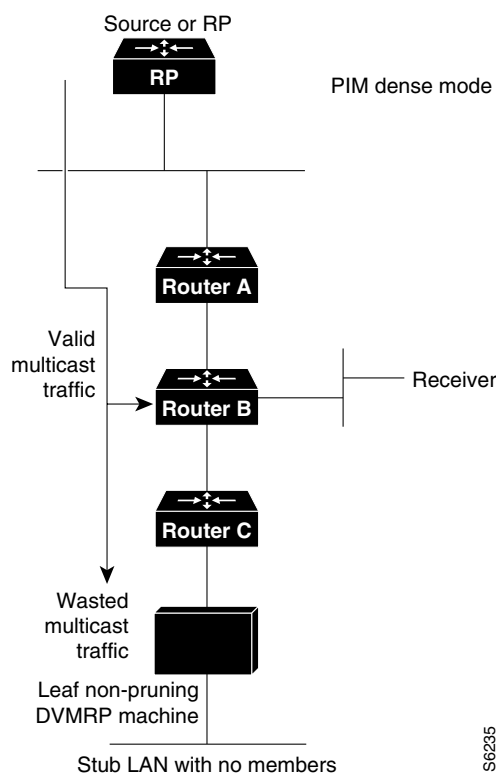
Similar to the **metric** keyword in mroute configuration files, the following is true.

- When you specify **in** or no keyword, the *increment* is added to incoming DVMRP reports and is reported in mroute replies. The default value for **in** is 1.
- When you specify **out**, the *increment* is added to outgoing DVMRP reports for routes from the DVMRP routing table. The default value for **out** is 0.

Reject a DVMRP Nonpruning Neighbor

By default, Cisco routers accept all DVMRP neighbors as peers, regardless of their DVMRP capability or lack thereof. However, some non-Cisco machines run old versions of DVMRP that cannot prune, so they will continuously receive forwarded packets unnecessarily, wasting bandwidth. Figure 36 shows this scenario.

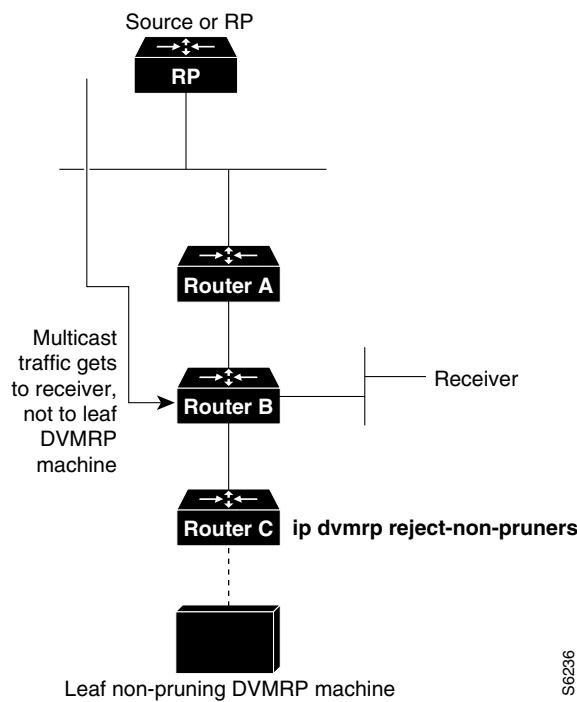
Figure 36 Leaf Nonpruning DVMRP Neighbor



56235

You can prevent a router from peering (communicating) with a DVMRP neighbor if that neighbor does not support DVMRP pruning or grafting. To do so, configure Router C (which is a neighbor to the leaf, nonpruning DVMRP machine) with the **ip dvmrp reject-non-pruners** command on the interface to the nonpruning machine. Figure 37 illustrates this scenario. In this case, when the router receives a DVMRP Probe or Report message without the Prune-Capable flag set, the router logs a syslog message and discards the message.

Figure 37 Router Rejects Nonpruning DVMRP Neighbor



Note that the **ip dvmrp reject-non-pruners** command prevents peering with neighbors only. If there are any nonpruning routers multiple hops away (downstream toward potential receivers) that are not rejected, then a nonpruning DVMRP network might still exist.

To prevent peering with nonpruning DVMRP neighbors, perform the following task in interface configuration mode:

Task	Command
Prevent peering with non-pruning DVMRP neighbors.	ip dvmrp reject-non-pruners

Configure a Delay between DVRMP Reports

You can configure an interpacket delay of a DVMRP report. The delay is the number of milliseconds that elapse between transmissions of sets of packets that constitute a report. The number of packets in the set is determined by the *burst* value, which defaults to 2 packets. The *milliseconds* value defaults to 100 milliseconds.

To change the default values of the delay, perform the following task in interface configuration mode:

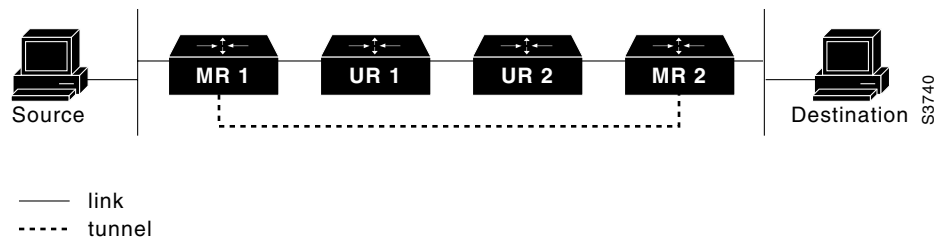
Task	Command
Configure an inter-packet delay between DVMRP reports.	ip dvmrp output-report-delay <i>milliseconds</i> [<i>burst</i>]

Configure an IP Multicast Static Route

IP multicast static routes (mroutes) allow you to have multicast paths diverge from the unicast paths. When using PIM, the router expects to receive packets on the same interface where it sends unicast packets back to the source. This is beneficial if your multicast and unicast topologies are congruent. However, you might want unicast packets to take one path and multicast packets to take another.

The most common reason for using separate unicast and multicast paths is tunneling. When a path between a source and a destination does not support multicast routing, a solution is to configure two routers with a GRE tunnel between them. In Figure 38, the UR routers support unicast packets only; the MR routers support multicast packets.

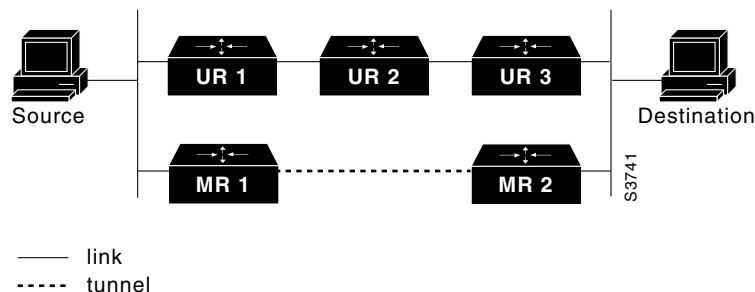
Figure 38 Tunnel for Multicast Packets



In Figure 38, Source delivers multicast packets to Destination by using MR1 and MR2. MR2 accepts the multicast packet only if it thinks it can reach Source over the tunnel. If this is true, when Destination sends unicast packets to Source, MR2 sends them over the tunnel. This could be slower than natively sending the unicast packet through UR2, UR1, and MR1.

Prior to multicast static routes, the configuration in Figure 39 was used to overcome the problem of both unicasts and multicasts using the tunnel. In this figure, MR1 and MR2 are used as multicast routers only. When Destination sends unicast packets to Source, it uses the (UR3,UR2,UR1) path. When Destination sends multicast packets, the UR routers do not understand or forward them. However, the MR routers forward the packets.

Figure 39 Separate Paths for Unicast and Multicast Packets



To make the configuration in Figure 39 work, MR1 and MR2 must run another routing protocol (typically a different instantiation of the same protocol running in the UR routers), so that paths from sources are learned dynamically.

A multicast static route allows you to use the configuration in Figure 38 by configuring a static multicast source. The Cisco IOS software uses the configuration information instead of the unicast routing table. This allows multicast packets to use the tunnel without having unicast packets use the tunnel. Static mroutes are local to the router they are configured on and not advertised or redistributed in any way to any other router.

To configure a multicast static route, perform the following task in global configuration mode:

Task	Command
Configure an IP multicast static route.	ip mroute <i>source mask [protocol as-number]</i> <i>{rpf-address type number} [distance]</i>

Control the Transmission Rate to a Multicast Group

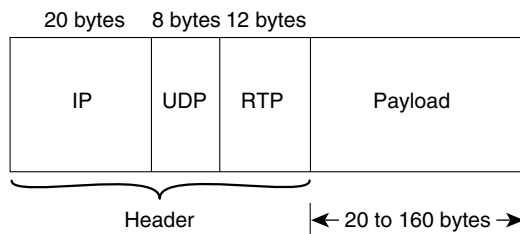
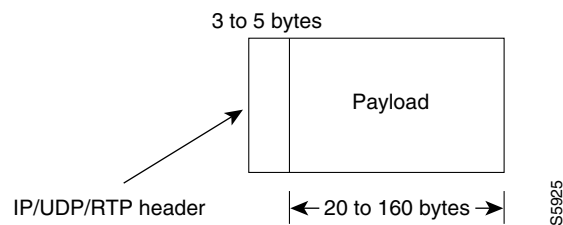
By default, there is no limit as to how fast a sender can transmit packets to a multicast group. To control the rate that the sender from the source list can send to a multicast group in the group list, perform the following task in interface configuration mode:

Task	Command
Control transmission rate to a multicast group.	ip multicast rate-limit { in out } [video whiteboard] [group-list <i>access-list</i>] [source-list <i>access-list</i>] <i>kbps</i>

Configure RTP Header Compression

Real-time Transport Protocol (RTP) is a protocol used for carrying packetized audio and video traffic over an IP network. RTP, described in RFC 1889, is not intended for data traffic, which uses TCP or UDP. RTP provides end-to-end network transport functions intended for applications with real-time requirements (such as audio, video, or simulation data over multicast or unicast network services).

The minimal 12 bytes of the RTP header, combined with 20 bytes of IP header and 8 bytes of UDP header, create a 40-byte IP/UDP/RTP header, as shown in Figure 40. The RTP packet has a payload of approximately 20 to 150 bytes for audio applications that use compressed payloads. It is very inefficient to transmit the IP/UDP/RTP header without compressing it.

Figure 40 RTP Header Compression**Before RTP header compression:****After RTP header compression:**

The RTP header compression feature compresses the IP/UDP/RTP header in an RTP data packet from 40 bytes to approximately 2 to 5 bytes, as shown in Figure 40. It is a hop-by-hop compression scheme similar to RFC 1144 for TCP header compression. Using RTP header compression can benefit both telephony voice and multicast backbone (MBONE) applications running over slow links.

RTP header compression is supported on serial lines using Frame Relay, HDLC, or PPP encapsulation. It is also supported over ISDN interfaces.

Enabling compression on both ends of a low-bandwidth serial link can greatly reduce the network overhead if there is a lot of RTP traffic on that slow link. This compression is beneficial especially when the RTP payload size is small (for example, compressed audio payloads of 20-50 bytes). Although the MBONE-style RTP traffic has higher payload sizes, compact encodings such as Compressed Encoding for Linear Prediction (CELP) can also help considerably.

Before you can enable RTP header compression, you must have configured a serial line that uses either Frame Relay, HDLC, or PPP encapsulation, or an ISDN interface. To configure RTP header compression, perform the tasks in the following sections. Either one of the first two tasks is required.

- Enable RTP Header Compression on a Serial Interface
- Enable RTP Header Compression with Frame Relay Encapsulation
- Change the Number of Header Compression Connections

You can compress the IP/UDP/RTP headers of RTP traffic to reduce the size of your packets, making audio or video communication more efficient. You must enable compression on both ends of a serial connection.

Enable RTP Header Compression on a Serial Interface

To enable RTP header compression for serial encapsulations HDLC or PPP, perform the following task in interface configuration mode:

Task	Command
Enable RTP header compression.	ip rtp header-compression [passive]

If you include the **passive** keyword, the software compresses outgoing RTP packets only if incoming RTP packets on the same interface are compressed. If you use the command without the **passive** keyword, the software compresses all RTP traffic.

Enable RTP Header Compression with Frame Relay Encapsulation

To enable RTP header compression with Frame Relay encapsulation, perform one of the following tasks in interface configuration mode:

Task	Command
Enable RTP header compression on the physical interface and all the interface maps will inherit it. Subsequently, all maps will perform RTP/IP header compression.	frame-relay ip rtp header-compression [passive]
Enable RTP header compression only on the particular map specified.	frame-relay map ip <i>ip-address dlci</i> [broadcast] rtp header-compression [active passive]
Enable both RTP and TCP header compression on this link.	frame-relay map ip <i>ip-address dlci</i> [broadcast] compress

Change the Number of Header Compression Connections

By default, the software supports a total of 16 RTP header compression connections on an interface. To change that number, perform the following task in interface configuration mode:

Task	Command
Specify the total number of RTP header compression connections supported on an interface.	ip rtp compression connections <i>number</i>

Configure IP Multicast over ATM Point-to-Multipoint Virtual Circuits

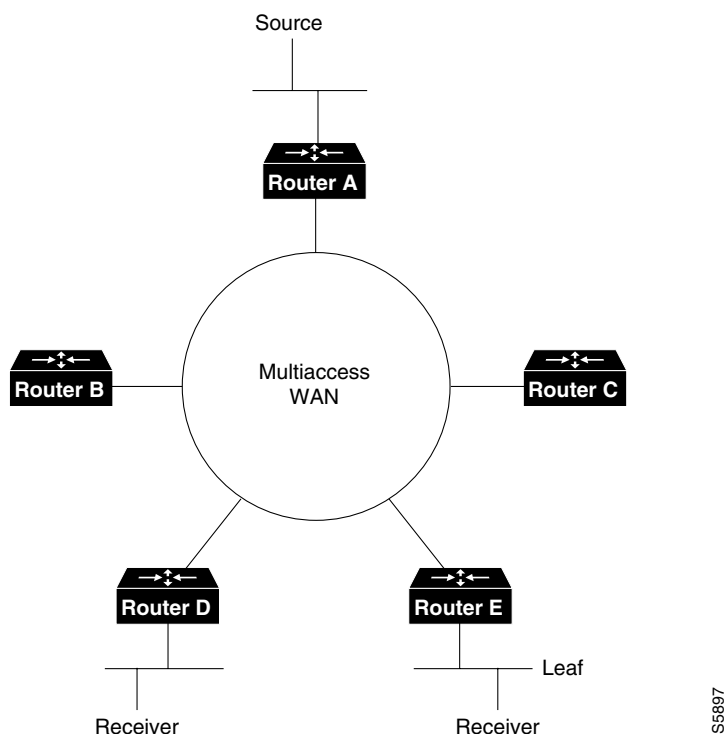
IP multicast over ATM point-to-multipoint virtual circuits is a feature that dynamically creates ATM point-to-multipoint SVCs to handle IP multicast traffic more efficiently.

The feature can enhance router performance and link utilization because packets are not replicated and sent multiple times over the ATM interface.

Traditionally, over nonbroadcast, multiaccess (NBMA) networks, Cisco routers would perform a pseudobroadcast to get broadcast or multicast packets to all neighbors on a multiaccess network. For example, assume in Figure 41 that Routers A, B, C, D, and E were running Open Shortest Path First

(OSPF) protocol. Router A must deliver to Routers D and E. When A sends an OSPF Hello, the data-link layer replicates the Hello and sends one to each neighbor, known as *pseudobroadcast*, which results in four copies being sent over the link from Router A to the multi-access WAN.

Figure 41 Environment for IP Multicast over ATM Point-to-Multipoint Virtual Circuits



With the advent of IP multicast, where high-rate multicast traffic can occur, that approach does not scale. Furthermore, in the preceding example, Routers B and C would get data traffic they do not need. To handle this problem, PIM can be configured in NBMA mode using the **ip pim nbma-mode** command. PIM in NBMA mode works only for sparse-mode groups. This would allow only routers D and E to get the traffic without distributing to B and C. However, two copies are still delivered over the link from A to the multiaccess WAN.

If the underlying network supported multicast capability, the routers could handle this situation more efficiently. If the multiaccess WAN were an ATM network, IP multicast could use multipoint virtual circuits.

This works by having routers A, B, C, D, and E run sparse-mode PIM. Suppose the Receiver directly connected to D joins a group and A is the PIM Rendezvous Point (RP). The following sequence occurs:

- 1 Router D will send a PIM Join message to A.
- 2 When A receives the PIM Join, it sets up a multipoint virtual circuit (VC) for the multicast group.
- 3 Later, when the Receiver directly connected to E joins the same group, E will send a PIM Join to A.
- 4 Router A will see there is a multipoint VC already associated with the group, and will add E to the existing multipoint VC.

- 5 When the Source sends a data packet, A can send a single packet over its link that gets to both D and E. The replication occurs in the ATM switches at the topological diverging point from A to D and E.

If a host sends an IGMP report over an ATM interface to a router, the router adds the host to the multipoint VC for the group.

This feature can be used over ATM subinterfaces also.

You must have ATM configured for multipoint signaling. Depending on which router platform you have, refer to the section called “Configure Point-to-Multipoint Signaling” in one of the following ATM chapters in the *Wide-Area Networking Configuration Guide*:

- “Configuring ATM on the AIP for Cisco 7000 and 7500 Series Routers”
- “Configure ATM on the ATM Port Adapter for Cisco 7000, 7200, and 7500 Series Routers”
- “Configure ATM on the NPM for Cisco 4500 and 4700 Routers”

You also must have IP multicast routing and PIM sparse mode configured. This feature does not work with dense-mode PIM.

Perform the tasks in the following sections to configure IP multicast over ATM point-to-multipoint virtual circuits. The first task is required; the remaining tasks are optional.

- Enable IP Multicast over ATM Point-to-Multipoint VCs
- Limit the Number of Virtual Circuits

Enable IP Multicast over ATM Point-to-Multipoint VCs

To enable PIM to open ATM point-to-multipoint virtual circuits for each multicast group that a receiver joins, perform the following tasks in interface configuration mode on the ATM interface:

Task	Command
Enable IP multicast over ATM point-to-multipoint virtual circuits.	ip pim multipoint-signalling
Enable point-to-multipoint signaling to the the ATM switch.	atm multipoint-signaling

The **atm multipoint-signaling** command is required so that static-map multipoint VCs can be opened. The router uses existing static map entries that include the **broadcast** keyword to establish multipoint calls. You must have the map list to act like a static ARP table.

Use the **show ip pim vc** command to display ATM VC status information for multipoint VCs opened by PIM.

Limit the Number of Virtual Circuits

By default, PIM can open a maximum of 200 virtual circuits. When the router reaches this number, it deletes inactive virtual circuits so it can open VCs for new groups that might have activity. To change the maximum number of VCs that PIM can open, perform the following task in interface configuration mode:

Task	Command
Change the maximum number of VCs that PIM can open.	ip pim vc-count <i>number</i>

Idling Policy

An idling policy uses the **ip pim vc-count** *number* to limit the number of VCs created by PIM. When the router stays at or below this *number* value, no idling policy is in effect. When the next VC to be opened will exceed the *number* value, an idling policy is exercised. An idled virtual circuit does not mean that the multicast traffic is not forwarded; the traffic is switched to vc 0. The vc 0 is the broadcast virtual circuit that is open to all neighbors listed in the map list. The name “vc0” is unique to PIM and the mrouting table.

How the Idling Policy Works

The idling policy works as follows:

- The only VCs eligible for idling are those with a current 1-second activity rate less than or equal to the value configured by the **ip pim minimum-vc-rate** command on the ATM interface. Activity level is measured in packets per second (pps).
- The VC with the least amount of activity below the configured **ip pim minimum-vc-rate** pps rate is idled.
- If the **ip pim minimum-vc-rate** command is not configured, all VCs are eligible for idling.
- If there are other VCs at the same activity level, the VC with the highest fanout (number of leaf routers on the multipoint VC) is idled.
- The activity level is rounded to three orders of magnitude (less than 10 pps, 10 to 100 pps, and 100 to 1000 pps). Therefore, a VC that has 40 pps activity and another that has 60 pps activity are considered to have the same rate, and the fanout count determines which one is idled. If the first VC has a fanout of 5 and the second has a fanout of 3, the first one is idled.
- Idling a VC means releasing the multipoint VC that is dedicated for the multicast group. The group's traffic continues to be sent; it is moved to the static-map VC. Packets will flow over a shared multipoint VC that delivers packets to all PIM neighbors.
- If all VCs have a 1-minute rate greater than pps, the new group (that exceeded the **ip pim vc-count** *number*) will use the shared multipoint VC.

Keep VCs from Idling

You can configure the minimum rate required to keep VCs from being idled. By default, all VCs are eligible for idling. To configure a minimum rate, perform the following task in interface configuration mode:

Task	Command
Set the minimum activity rate required to keep VCs from being idled.	ip pim minimum-vc-rate <i>pps</i>

Configure an IP Multicast Boundary

You can set up an administratively scoped boundary on an interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is set up, 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. Then this range of addresses can be reused in domains administered by different organizations. They would be considered local, not globally unique.

To set up an administratively scoped boundary, perform the following tasks beginning in global configuration mode:

Task	Command
Step 1 Create a standard access list, repeating the command as many times as necessary.	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>]
Step 2 Configure an interface.	interface <i>type number</i>
Step 3 Configure the boundary, specifying the access list you created in Step 1.	ip multicast boundary <i>access-list-number</i>

See the section “Administratively Scoped Boundary Example” at the end of this chapter for an example of configuring a boundary.

Configure an Intermediate IP Multicast Helper

When a multicast-capable internetwork is between two subnets with broadcast-only capable hosts, you can convert broadcast traffic to multicast at the first hop router, and convert it back to broadcast at the last hop router to deliver the packets to the broadcast clients. Thus, you can take advantage of the multicast capability of the intermediate multicast internetwork. This feature prevents unnecessary replication at the intermediate routers and can take advantage of multicast fast switching in the multicast internetwork.

See Figure 43 and the example of this feature in the section “IP Multicast Helper Example” at the end of this chapter.

An extended IP access list controls which broadcast packets are translated, based on the UDP port number.

To configure an intermediate IP multicast helper, perform the following tasks on the first hop router beginning in global configuration mode:

Task	Command
Specify an interface.	interface <i>type number</i>
Configure a first hop router to convert broadcast traffic to multicast traffic.	ip multicast helper-map broadcast <i>multicast-address</i> <i>extended-access-list-number</i>
Configure an access list.	access-list <i>access-list-number</i> { deny permit } udp <i>source</i> <i>source-wildcard</i> <i>destination</i> <i>destination-wildcard</i> <i>port</i>
Configure IP to forward the protocol you are using.	ip forward-protocol udp [<i>port</i>]

Then perform the following tasks on the last hop router beginning in global configuration mode:

Task	Command
Specify an interface.	interface <i>type number</i>
Configure a last hop router to convert multicast traffic to broadcast traffic.	ip multicast helper-map <i>group-address</i> <i>broadcast-address</i> <i>extended-access-list-number</i>
Configure an access list.	access-list <i>access-list-number</i> { deny permit } udp <i>source</i> <i>source-wildcard</i> <i>destination</i> <i>destination-wildcard</i> <i>port</i>

Task	Command
Configure IP to forward the protocol you are using.	ip forward-protocol udp [<i>port</i>]

Note On the last hop router, the **ip multicast helper-map** command automatically introduces **ip igmp join-group** *group-address* on that interface. This command must stay there for this feature to work. If you remove the **ip igmp join-group** command, the feature will fail.

Store IP Multicast Headers

You can store IP multicast packet headers in a cache and then display them to determine any of the following information:

- Who is sending IP multicast packets to what groups
- Inter-packet delay
- Duplicate IP multicast packets (if any)
- Multicast forwarding loops in your network (if any)
- Scope of the group
- UDP port numbers
- Packet length

Note This feature allocates a circular buffer of approximately 32 kilobytes.

To allocate a circular buffer to store IP multicast packet headers that the router receives, perform the following task in global configuration mode:

Task	Command
Allocate a buffer to store IP multicast packet headers.	ip multicast cache-headers

Use the **show ip mpacket** command to display the buffer.

Enable CGMP

Cisco Group Management Protocol (CGMP) is a protocol used on routers connected to Cisco Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary because the Catalyst switch cannot tell the difference between IP multicast data packets and IGMP Report messages, which are both MAC-level addressed to the same group address.

Enabling CGMP triggers a CGMP Join message. CGMP should be enabled only on 802 or ATM media, or LANE over ATM. CGMP should be enabled only on routers connected to Catalyst switches.

To enable CGMP for IP multicast on a LAN, perform the following task in interface configuration mode:

Task	Command
Enable CGMP.	ip cgmp [proxy]

When the **proxy** keyword is specified, the CGMP proxy function is enabled. That is, any router that is not CGMP-capable will be advertised by the proxy router. The proxy router advertises the existence of other non CGMP-capable routers by sending a CGMP Join message with the non CGMP-capable router's MAC address and a group address of 0000.0000.0000.

Configure Stub IP Multicast Routing

When using PIM in a large network, there are often stub regions over which the administrator has limited control. To reduce the configuration and administration burden, you can configure a subset of PIM functionality that provides the stub region with connectivity, but does not allow it to participate in or potentially complicate any routing decisions.

Stub IP multicast routing allows simple multicast connectivity and configuration at stub networks. It eliminates periodic flood-and-prune behavior across slow-speed links (ISDN and below) using dense mode. It does this by using forwarded IGMP reports as a type of Join message and using selective PIM message filtering.

Stub IP multicast routing allows stub sites to be configured quickly and easily for basic multicast connectivity, without the flooding of multicast packets and subsequent group pruning that occurs in dense-mode, and without excessive administrative burden at the central site.

Before configuring stub IP multicast routing, you must have IP multicast routing configured on both the stub router and the central router. You must also have PIM dense mode configured on both the incoming and outgoing interfaces of the stub router.

Two steps are required to enable stub IP multicast routing. One task is performed on the stub router, and the other is performed on a central router one hop away from the stub router. By definition, a stub region is marked by a leaf router. That is, the stub router (leaf router) is the last stop before any hosts receiving multicast packets or the first stop for anyone sending multicast packets.

The first step is to configure the stub router to forward all IGMP Host Reports and Leave messages received on the interface to an IP address. The reports are resent out the next-hop interface toward the IP address, with that interface's source address. This action enables a sort of "dense-mode" Join, allowing stub sites not participating in PIM to indicate membership in multicast groups.

To configure the stub router to forward IGMP Host Reports and Leave messages, perform the following task in interface configuration mode. Specify the IP address of an interface on the central router. When the central router receives IGMP Host Report and Leave messages, it appropriately adds or removes the interface from its outgoing list for that group.

Task	Command
Step 1 On the stub router, forward all IGMP Host Reports and Leave messages to the specified IP address on a central router.	ip igmp helper-address <i>ip-address</i>

The second step is to configure an access list on the central router to filter all PIM control messages from the stub router. Thus, the central router does not by default add the stub router to its outgoing interface list for any multicast groups. This task has the side benefit of preventing a misconfigured PIM neighbor from participating in PIM.

To filter PIM control messages, perform the following task in interface configuration mode:

Task	Command
Step 2 On the central router, filter all PIM control messages based on the specified access list.	ip pim neighbor-filter <i>access-list-number</i>

For an example of stub IP multicast routing, see the section “Stub IP Multicast Example” at the end of this chapter.

Load Split IP Multicast Traffic across Equal-Cost Paths

You can now configure load splitting of IP multicast traffic across equal-cost paths. Prior to this feature, when there were equal-cost paths between routers, IP multicast packets traversed only one path. If a tunnel was configured, the same next hop was always used, and no load splitting occurred.

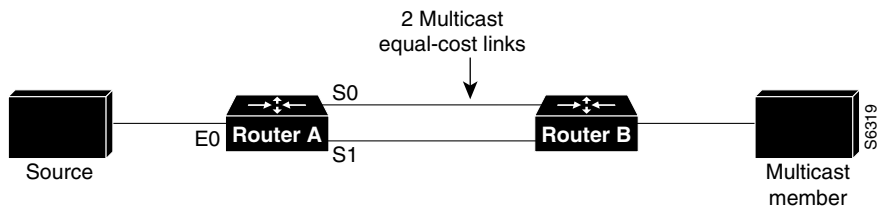
IP multicast load splitting is accomplished indirectly by consolidating the available bandwidth of all the physical links into a single tunnel interface. The underlying physical connections then use existing unicast load-splitting mechanisms for the tunnel (multicast) traffic.

Note This feature is load splitting the traffic, not load balancing the traffic.

By configuring load splitting among equal-cost paths, you can use your links between routers more efficiently when sending IP multicast traffic.

Due to reverse-path forwarding (RPF) issues, splitting IP multicast traffic across physical interfaces is nearly impossible. Consider the sample topology in Figure 42, where Router A and Router B are connected with two equal-cost multicast links. Once a router chooses its RPF interface (Serial 0 or Serial 1), all subsequent multicast traffic is accepted only from that interface (assuming there are no routing changes). Hence, all multicast traffic uses only one link.

Figure 42 Two Multicast Links without Load Splitting



The solution is to consolidate all the bandwidth from the equal-cost links for multicast traffic by configuring a multicast (GRE) tunnel between Router A and Router B. The routers should be made to RPF to the tunnel interface and not to any of the physical equal-cost interfaces between them. The multicast packets are then unicast across the tunnel and the underlying unicast mechanisms perform load splitting of these now unicast packets across the equal cost links.

Use the configuration tasks in the following sections to achieve this solution. The first three tasks are required.

- Configure the Access Router
- Configure the Router at the Opposite End of the Tunnel
- Configure Both Routers to RPF
- Verify the Load Splitting

Configure the Access Router

To configure the access router end of the tunnel (the end of the tunnel near the source), perform the following tasks, beginning in global configuration mode. The tunnel mode is GRE IP by default.

Task	Command
Step 1 Configure a tunnel interface.	interface tunnel <i>number</i>
Step 2 Enable IP processing without assigning an IP address to the interface.	ip unnumbered <i>type number</i>
Step 3 Enable PIM on the tunnel interface.	ip pim { dense-mode sparse-mode sparse-dense-mode }
Step 4 Configure the tunnel source.	tunnel source { <i>ip-address</i> <i>type number</i> }
Step 5 Configure the tunnel destination.	tunnel destination { <i>hostname</i> <i>ip-address</i> }

Configure the Router at the Opposite End of the Tunnel

Next, perform the following tasks on the router at the opposite end of the tunnel, beginning in global configuration mode:

Task	Command
Step 1 Configure a tunnel interface.	interface tunnel <i>number</i>
Step 2 Enable IP processing without assigning an IP address to the interface.	ip unnumbered <i>type number</i>
Step 3 Enable PIM on the tunnel interface.	ip pim { dense-mode sparse-mode sparse-dense-mode }
Step 4 Configure the tunnel source. This matches the tunnel destination at the opposite end of the tunnel.	tunnel source { <i>ip-address</i> <i>type number</i> }
Step 5 Configure the tunnel destination. This matches the tunnel source at the opposite end of the tunnel.	tunnel destination { <i>hostname</i> <i>ip-address</i> }

Configure Both Routers to RPF

Since the use of the tunnel makes the multicast topology incongruent with the unicast topology, and only multicast traffic traverses the tunnel, you must configure the routers to RPF correctly over the tunnel. The following sections describe the two ways to do this, depending on your topology:

- Load Splitting to a Stub Network
- Load Splitting to the Middle of a Network

Load Splitting to a Stub Network

If you are load splitting to a stub network, you can use a static multicast route. First perform the following task on the stub router in global configuration mode:

Task	Command
Configure a static multicast route over which to RPF from the stub router to the other end of the tunnel.	ip mroute 0.0.0.0 0.0.0.0 tunnel number

Then perform the following tasks on the router at the opposite end of the tunnel from the stub router, in global configuration mode:

Task	Command
Step 1 Configure a static route over which to RPF from the access router to the other end of the tunnel. Configure the <i>source</i> to be the network address of the network connected to the stub router.	ip mroute source mask tunnel number
Step 2 Repeat Step 1 for each network connected to the stub router.	ip mroute source mask tunnel number

Load Splitting to the Middle of a Network

You can use static mroutes in this case also, but you must make sure that Router A would RPF to the tunnel for source networks behind Router B, and Router B would RPF to the tunnel for source networks behind Router A.

Another option is to run a separate unicast routing protocol with a better administrative distance to provide the RPF. You must make sure that your multicast routers do not advertise the tunnel to your real network. For details, refer to the “Configure an IP Multicast Static Route” section in this chapter.

If you are using a DVMRP routing table for RPF information within your network, you could configure the **ip dvmrp unicast-routing** command on your tunnel interfaces to make the routers RPF correctly over the tunnel.

Verify the Load Splitting

Load splitting works for both fast switching and process switching, but splitting the traffic among the physical interfaces is performed differently for each case. Fast switching occurs if both the incoming and outgoing interfaces are configured with the **ip mroute-cache** command. IP multicast fast switching is enabled by default. Keep the following in mind:

- With process switching, load splitting occurs on a per-packet basis by round-robin on the equal-cost links. To verify that load splitting is working, look at the interface statistics using the **show interfaces accounting** command, and verify that the packet count is about equal for the underlying interfaces that provide the equal-cost paths.
- With fast switching, load splitting occurs on a per-flow basis. A flow is a set of traffic with the same source and destination. Once the cache is populated for the (S,G) pair, that flow is pinned to the physical interface assigned on the cache (the outgoing interface used by the first packet of the flow). If the cached interface goes down, the cache entry for the (S,G) pair is torn down and the flow is automatically switched to a different physical interface.

In the case of fast switching, you can verify that load splitting is occurring by viewing the multicast fast-switched cache with the **show ip mcache** command. The flows should be split among the underlying interfaces, as shown in the example that follows:

```
Router# show ip mcache

IP Multicast Fast-Switching Cache
(100.1.1.6/32, 224.1.1.1), Ethernet0, Last used: 00:00:00
Tunnel0      MAC Header: 0F000800 (Serial1)
(100.1.1.6/32, 224.1.1.2), Ethernet0, Last used: 00:00:00
Tunnel0      MAC Header: 0F000800 (Serial1)
(100.1.1.5/32, 224.1.1.3), Ethernet0, Last used: 00:00:00
Tunnel0      MAC Header: 0F000800 (Serial0)
(100.1.1.5/32, 224.1.1.4), Ethernet0, Last used: 00:00:00
Tunnel0      MAC Header: 0F000800 (Serial0)
```

For an example of load splitting IP multicast traffic across equal-cost paths, see the section “Load Splitting IP Multicast Traffic across Equal-Cost Paths Example” at the end of this chapter.

Monitor and Maintain IP Multicast Routing

You can remove all contents of a particular cache, table, or database. You also can display specific statistics. The following sections describe each of these tasks.

Clear Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database can become necessary when the contents of the particular structure have become, or are suspected to be, invalid.

The following table lists the tasks associated with clearing IP multicast caches, tables, and databases. Perform these tasks in EXEC mode:

Task	Command
Clear all group entries the Catalyst switches have cached.	clear ip cgmp
Delete routes from the DVMRP routing table.	clear ip dvmrp route { * route }
Delete entries from the IGMP cache.	clear ip igmp group [group-name group-address interface]
Delete entries from the IP multicast routing table.	clear ip mroute { * group [source] }
Clear the Auto-RP cache.	clear ip pim auto-rp rp-address
Clear RTP header compression structures and statistics.	clear ip rtp header-compression [type number]
Delete the Session Directory Protocol Version 2 cache or an sdr cache entry.	clear ip sdr [group-address “session-name”]

Display System and Network Statistics

You can display specific statistics such as the contents of IP routing tables, caches, and databases. Information provided can be used to determine resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device’s packets are taking through the network.

To display various routing statistics, perform the following tasks in EXEC mode:

Task	Command
Query a multicast router about which neighboring multicast routers are peering with it.	mrinfo <i>[hostname-or-address]</i> <i>[source-address-or-interface]</i>
Display IP multicast packet rate and loss information.	mstat <i>source</i> <i>[destination]</i> <i>[group]</i>
Trace the path from a source to a destination branch for a multicast distribution tree for a given group.	mtrace <i>source</i> <i>[destination]</i> <i>[group]</i>
Send an ICMP Echo Request to a multicast group address.	ping <i>[group-address-or-name]</i>
Display Frame Relay RTP header compression statistics.	show frame-relay ip rtp header-compression <i>[interface type number]</i>
Display the entries in the DVMRP routing table.	show ip dvmrp route <i>[ip-address]</i>
Display the multicast groups that are directly connected to the router and that were learned via IGMP.	show ip igmp groups <i>[group-name group-address type number]</i>
Display multicast-related information about an interface.	show ip igmp interface <i>[type number]</i>
Display the contents of the IP fast-switching cache.	show ip mcache <i>[group [source]]</i>
Display the contents of the circular cache-header buffer.	show ip mpacket <i>[source-address-or-name]</i> <i>[group-address-or-name]</i> [detail]
Display the contents of the IP multicast routing table.	show ip mroute <i>[group-name group-address]</i> <i>[source]</i> [summary] [count] [active kbps]
Display information about interfaces configured for PIM.	show ip pim interface <i>[type number]</i> [count]
List the PIM neighbors discovered by the router.	show ip pim neighbor <i>[type number]</i>
Display the RP routers associated with a sparse-mode multicast group.	show ip pim rp <i>[group-name group-address]</i>
Display ATM VC status information for multipoint VCs opened by PIM.	show ip pim vc <i>[group-or-name]</i> <i>[type number]</i>
Display how the router is doing Reverse-Path Forwarding (that is, from the unicast routing table, DVMRP routing table, or static mroutes).	show ip rpf <i>source-address-or-name</i>
Display RTP header compression statistics.	show ip rtp header-compression <i>[type number]</i> [detail]
Display the Session Directory Protocol Version 2 cache.	show ip sdr <i>[group "session-name"]</i> [detail]

IP Multicast Configuration Examples

This section provides the following IP multicast routing configuration examples:

- PIM Dense Mode Example
- PIM Sparse Mode Example
- DVMRP Interoperability Example
- DVMRP Tunnel Example

- RTP Header Compression Examples
- IP Multicast over ATM Point-to-Multipoint VC Example
- Functional Address for IP Multicast over Token Ring LAN Example
- Administratively Scoped Boundary Example
- IP Multicast Helper Example
- Stub IP Multicast Example
- Load Splitting IP Multicast Traffic across Equal-Cost Paths Example

PIM Dense Mode Example

The following example configures dense-mode PIM on an Ethernet interface of the router:

```
ip multicast-routing
interface ethernet 0
 ip pim dense-mode
```

PIM Sparse Mode Example

The following example configures the Cisco IOS software to operate in sparse-mode PIM. The RP router is the router whose address is 10.8.0.20.

```
ip multicast-routing
 ip pim rp-address 10.8.0.20 1
interface ethernet 1
 ip pim sparse-mode
```

DVMRP Interoperability Example

The following example configures DVMRP interoperability for configurations when the PIM router and the DVMRP router are on the same network segment. In this example, access list 1 advertises the networks (98.92.35.0, 198.92.36.0, 198.92.37.0, 131.108.0.0, and 150.136.0.0) to the DVMRP router, and access list 2 is used to prevent all other networks from being advertised (**ip dvmrp metric 0**).

```
interface ethernet 0
 ip address 131.119.244.244 255.255.255.0
 ip pim dense-mode
 ip dvmrp metric 1 list 1
 ip dvmrp metric 0 list 2

access-list 1 permit 198.92.35.0 0.0.0.255
access-list 1 permit 198.92.36.0 0.0.0.255
access-list 1 permit 198.92.37.0 0.0.0.255
access-list 1 permit 131.108.0.0 0.0.255.255
access-list 1 permit 150.136.0.0 0.0.255.255
access-list 1 deny 0.0.0.0 255.255.255.255
access-list 2 permit 0.0.0.0 255.255.255.255
```

DVMRP Tunnel Example

The following example configures a DVMRP tunnel:

```
!
ip multicast-routing
!
```



```
interface tunnel 0
 ip unnumbered ethernet 0
 ip pim dense-mode
 tunnel source ethernet 0
 tunnel destination 192.70.92.133
 tunnel mode dvmrp
!
interface ethernet 0
 description Universitat DMZ-ethernet
 ip address 192.76.243.2 255.255.255.0
 ip pim dense-mode
```

RTP Header Compression Examples

The following example enables RTP header compression for a serial, ISDN, or asynchronous interface. For ISDN, you also need a broadcast dialer map.

```
interface serial 0 :or interface bri 0
 ip rtp header-compression
 encapsulation ppp
 ip rtp compression-connections 25
```

The following example is for Frame Relay encapsulation. It enables RTP header compression on the specified map.

```
interface serial 0
 ip address 1.0.0.2 255.0.0.0
 encapsulation frame-relay
 no keepalive
 clockrate 64000
 frame-relay map ip 1.0.0.1 17 broadcast rtp header-compression
```

IP Multicast over ATM Point-to-Multipoint VC Example

The following example enables IP multicast over ATM point-to-multipoint virtual circuits:

```
interface ATM2/0
 ip address 171.69.214.43 255.255.255.248
 ip pim sparse-mode
 ip pim multipoint-signalling
 ip ospf network broadcast
 atm nsap-address 47.00918100000000410B0A1981.333333333333.00
 atm pvc 1 0 5 qsaal
 atm pvc 2 0 16 ilmi
 atm multipoint-signalling
 map-group mpvc
 router ospf 9
 network 171.69.214.0 0.0.0.255 area 0
!
 ip classless
 ip pim rp-address 171.69.10.13 98
!
 map-list mpvc
 ip 171.69.214.41 atm-nsap 47.00918100000000410B0A1981.111111111111.00 broadcast
 ip 171.69.214.42 atm-nsap 47.00918100000000410B0A1981.222222222222.00 broadcast
 ip 171.69.214.43 atm-nsap 47.00918100000000410B0A1981.333333333333.00 broadcast
```

Functional Address for IP Multicast over Token Ring LAN Example

In the following example, any IP multicast packets going out Token Ring interface 0 are mapped to MAC address 0xc000.0004.0000:

```
interface token 0
ip address 1.1.1.1 255.255.255.0
ip pim dense-mode
ip multicast use-functional
```

Administratively Scoped Boundary Example

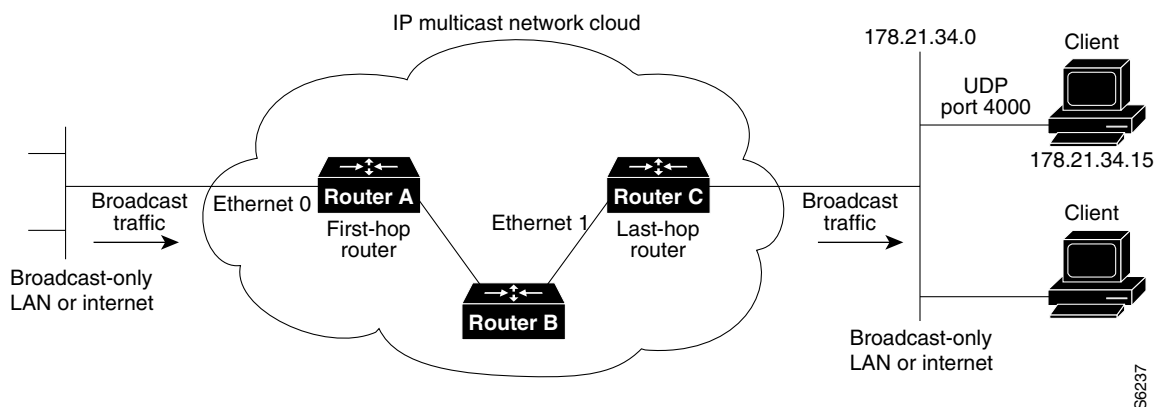
The following example sets up a boundary for all administratively scoped addresses:

```
access-list 1 deny 239.0.0.0 0.255.255.255
access-list 1 permit 224.0.0.0 15.255.255.255
interface ethernet 0
ip multicast boundary 1
```

IP Multicast Helper Example

Figure 43 illustrates how a helper address on two routers converts from broadcast to multicast and back to broadcast.

Figure 43 IP Multicast Helper Scenario



The configuration on the first hop router converts a broadcast stream arriving at incoming interface Ethernet 0 destined for UDP port 4000 to a multicast stream. The access list denies other traffic from being forwarded into the multicast cloud. The traffic is sent to group address 224.5.5.5. Because fast switching does not perform such a conversion, the **ip forward-protocol** command causes the proper process level to perform the conversion.

The second configuration on the last hop router converts the multicast stream at incoming interface Ethernet 1 back to broadcast. Again, all multicast traffic emerging from the multicast cloud should not be converted to broadcast, only the traffic destined for UDP port 4000.

First Hop Router

```
interface ethernet 0
 ip multicast helper-map broadcast 224.5.5.5 120
 ip pim dense-mode
!
access-list 120 permit any any udp 4000
access-list 120 deny any any udp
 ip forward-protocol udp 4000
```

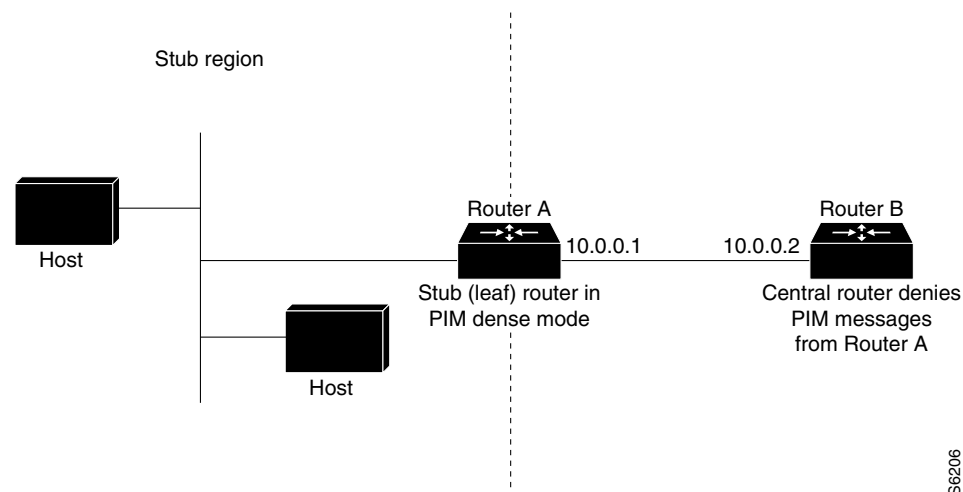
Last Hop Router

```
interface ethernet 1
 ip multicast helper-map 224.5.5.5 178.21.34.255 135
 ip pim dense-mode
!
access-list 135 permit any any udp 4000
access-list 135 deny any any udp
 ip forward-protocol udp 4000
```

Stub IP Multicast Example

The following example configures stub IP multicast routing for Router A. Figure 44 illustrates the example. On stub Router A, the interfaces must be configured for PIM dense mode. The helper address is configured on the host interfaces. Central site Router B can be configured for either sparse-mode or dense-mode PIM. The access list on Router B denies any PIM messages from Router A.

Figure 44 Stub IP Multicast Routing Example



Router A

```
ip multicast-routing
 ip pim dense-mode
 ip igmp helper-address 10.0.0.2
```

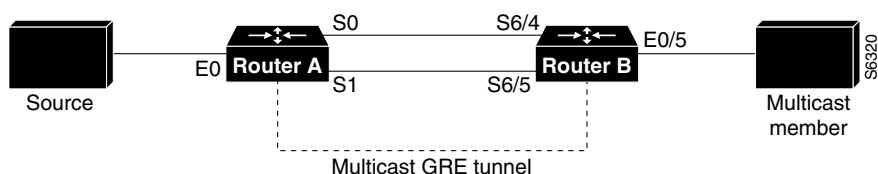
Router B

```
ip multicast-routing
ip pim dense-mode : or ip pim sparse-mode
ip pim neighbor-filter 1
access-list 1 deny 10.0.0.1
```

Load Splitting IP Multicast Traffic across Equal-Cost Paths Example

This example configures a GRE tunnel between Router A and Router B. Figure 45 illustrates the tunneled topology. The configurations follow the figure.

Figure 45 IP Multicast Load Splitting across Equal-Cost Paths



Router A

```
interface tunnel 0
ip unnumbered Ethernet0
ip pim dense-mode : or sparse-mode or sparse-dense-mode
tunnel source 100.1.1.1
tunnel destination 100.1.5.3
!
interface ethernet 0
ip address 100.1.1.1 255.255.255.0
ip pim dense-mode : or sparse-mode or sparse-dense-mode
!
interface Serial0
ip address 100.1.2.1 255.255.255.0
bandwidth 125
clock rate 125000
!
interface Serial1
ip address 100.1.3.1 255.255.255.0
bandwidth 125
```

Router B

```
interface tunnel 0
ip unnumbered ethernet 0/5
ip pim dense-mode : or sparse-mode or sparse-dense-mode
tunnel source 100.1.5.3
tunnel destination 100.1.1.1
!
interface ethernet 0/5
ip address 100.1.5.3 255.255.255.0
ip pim dense-mode : or sparse-mode or sparse-dense-mode
!
```

```
interface serial 6/4
 ip address 100.1.2.3 255.255.255.0
 bandwidth 125
!
interface Serial6/5
 ip address 100.1.3.3 255.255.255.0
 bandwidth 125
 clock rate 125000
```

