C H A P T E R **32**

Configuring IP Multicast

This chapter describes IP multicast routing on the Catalyst 4500 series switch. It also provides procedures and examples to configure IP multicast routing.

Note For more detailed information on IP Multicast, refer to this URL:

http://www.cisco.com/en/US/products/ps6552/products_ios_technology_home.html

Note For complete syntax and usage information for the switch commands used in this chapter, first look at the *Cisco Catalyst 4500 Series Switch Command Reference* and related publications at this location:

http://www.cisco.com/en/US/products//hw/switches/ps4324/index.html

If the command is not found in the *Catalyst 4500 Command Reference*, it will be found in the larger Cisco IOS library. Refer to the *Cisco IOS Command Reference* and related publications at this location:

http://www.cisco.com/en/US/products/ps6350/index.html

This chapter includes the following major sections:

Overview of IP Multicast

This section includes these subsections:

At one end of the IP communication spectrum is IP unicast, where a source IP host sends packets to a specific destination IP host. In IP unicast, the destination address in the IP packet is the address of a single, unique host in the IP network. These IP packets are forwarded across the network from the source

to the destination host by routers. At each point on the path between source and destination, a router uses a unicast routing table to make unicast forwarding decisions, based on the IP destination address in the packet.

At the other end of the IP communication spectrum is an IP broadcast, where a source host sends packets to all hosts on a network segment. The destination address of an IP broadcast packet has the host portion of the destination IP address set to all ones and the network portion set to the address of the subnet. IP hosts, including routers, understand that packets, which contain an IP broadcast address as the destination address, are addressed to all IP hosts on the subnet. Unless specifically configured otherwise, routers do not forward IP broadcast packets, so IP broadcast communication is normally limited to a local subnet.

IP multicasting falls between IP unicast and IP broadcast communication. IP multicast communication enables a host to send IP packets to a *group* of hosts anywhere within the IP network. To send information to a specific group, IP multicast communication uses a special form of IP destination address called an IP *multicast group address*. The IP multicast group address is specified in the IP destination address field of the packet.

To multicast IP information, Layer 3 switches and routers must forward an incoming IP packet to all output interfaces that lead to *members* of the IP multicast group. In the multicasting process on the Catalyst 4500 Metro series switch, a packet is replicated in the Integrated Switching Engine, forwarded to the appropriate output interfaces, and sent to each member of the multicast group.

It is not uncommon for people to think of IP multicasting and video conferencing as almost the same thing. Although the first application in a network to use IP multicast is often video conferencing, video is only one of many IP multicast applications that can add value to a company's business model. Other IP multicast applications that have potential for improving productivity include multimedia conferencing, data replication, real-time data multicasts, and simulation applications.

This section contains the following subsections:

- IP Multicast Protocols, page 32-2
- IP Multicast on the Catalyst 4500 Series Switch, page 32-4
- Unsupported Feature, page 32-12

IP Multicast Protocols

The Catalyst 4500 Metro series switch primarily uses these protocols to implement IP multicast routing:

- Internet Group Management Protocol (IGMP)
- Protocol Independent Multicast (PIM)
- IGMP snooping and Cisco Group Management Protocol

Figure 32-1 shows where these protocols operate within the IP multicast environment.

Figure 32-1   IP Multicast Routing Protocols
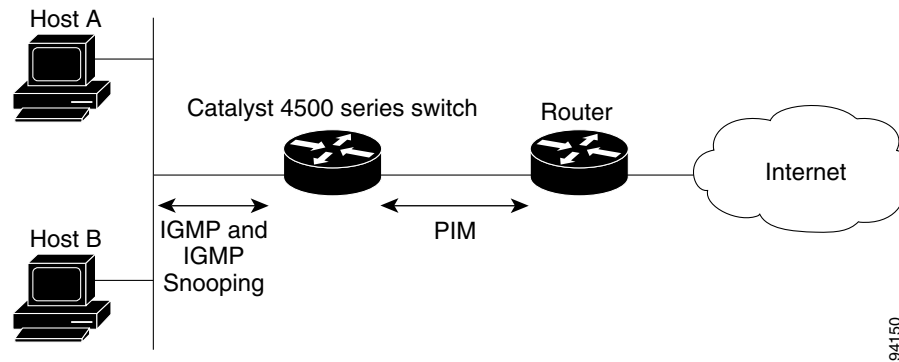
Internet Group Management Protocol

IGMP messages are used by IP multicast hosts to send their local Layer 3 switch or router a request to join a specific multicast group and begin receiving multicast traffic. With some extensions in IGMPv2, IP hosts can also send a request to a Layer 3 switch or router to leave an IP multicast group and not receive the multicast group traffic.

Using the information obtained via IGMP, a Layer 3 switch or router maintains a list of multicast group memberships on a per-interface basis. A multicast group membership is active on an interface if at least one host on the interface sends an IGMP request to receive multicast group traffic.

Protocol-Independent Multicast

PIM is *protocol independent* because it can leverage whichever unicast routing protocol is used to populate the unicast routing table, including EIGRP, OSPF, BGP, or static route, to support IP multicast. PIM also uses a unicast routing table to perform the reverse path forwarding (RPF) check function instead of building a completely independent multicast routing table. PIM does not send and receive multicast routing updates between routers like other routing protocols do.

PIM Dense Mode

PIM Dense Mode (PIM-DM) uses a *push* model to flood multicast traffic to every corner of the network. PIM-DM is intended for networks in which most LANs need to receive the multicast, such as LAN TV and corporate or financial information broadcasts. It can be an efficient delivery mechanism if there are active receivers on every subnet in the network.

For more detailed information on PIM Dense Mode, refer to this URL:

http://www.cisco.com/en/US/docs/ios-xml/ios/ipmulti_optim/configuration/12-2sx/imc_pim_dense_rfrsh.html

PIM Sparse Mode

PIM Sparse Mode (PIM-SM) uses a *pull* model to deliver multicast traffic. Only networks with active receivers that have explicitly requested the data will be forwarded the traffic. PIM-SM is intended for networks with several different multicasts, such as desktop video conferencing and collaborative computing, that go to a small number of receivers and are typically in progress simultaneously.

Rendezvous Point (RP)

If you configure PIM to operate in sparse mode, you must also choose one or more routers to be rendezvous points (RPs). Senders to a multicast group use RPs to announce their presence. Receivers of multicast packets use RPs to learn about new senders. You can configure Cisco IOS software so that packets for a single multicast group can use one or more RPs.

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 and prune messages to the RP to inform it about group membership. You must configure the RP address on all routers (including the RP router).

A PIM router can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain for the same group. The conditions specified by the access list determine for which groups the router is an RP (as different groups can have different RPs).

IGMP Snooping

IGMP snooping is used for multicasting in a Layer 2 switching environment. With IGMP snooping, a Layer 3 switch or router examines Layer 3 information in the IGMP packets in transit between hosts and a router. When the switch receives the IGMP Host Report from a host for a particular multicast group, the switch adds the host's port number to the associated multicast table entry. When the switch receives the IGMP Leave Group message from a host, it removes the host's port from the table entry.

Because IGMP control messages are transmitted as multicast packets, they are indistinguishable from multicast data if only the Layer 2 header is examined. A switch running IGMP snooping examines every multicast data packet to determine whether it contains any pertinent IGMP control information. If IGMP snooping is implemented on a low end switch with a slow CPU, performance could be severely impacted when data is transmitted at high rates. On the Catalyst 4500 series switches, IGMP snooping is implemented in the forwarding ASIC, so it does not impact the forwarding rate.

IP Multicast on the Catalyst 4500 Series Switch

The Catalyst 4500 Metro series switch supports an ASIC-based Integrated Switching Engine that provides Ethernet bridging at Layer 2 and IP routing at Layer 3. Because the ASIC is specifically designed to forward packets, the Integrated Switching Engine hardware provides very high performance with ACLs and QoS enabled. At wire-speed, forwarding in hardware is significantly faster than the CPU subsystem software, which is designed to handle exception packets.

The Integrated Switching Engine hardware supports interfaces for inter-VLAN routing and switchports for Layer 2 bridging. It also provides a physical Layer 3 interface that can be configured to connect with a host, a switch, or a router.

Figure 32-2 shows a logical view of Layer 2 and Layer 3 forwarding in the Integrated Switching Engine hardware.

Figure 32-2   Logical View of Layer 2 and Layer 3 Forwarding in Hardware
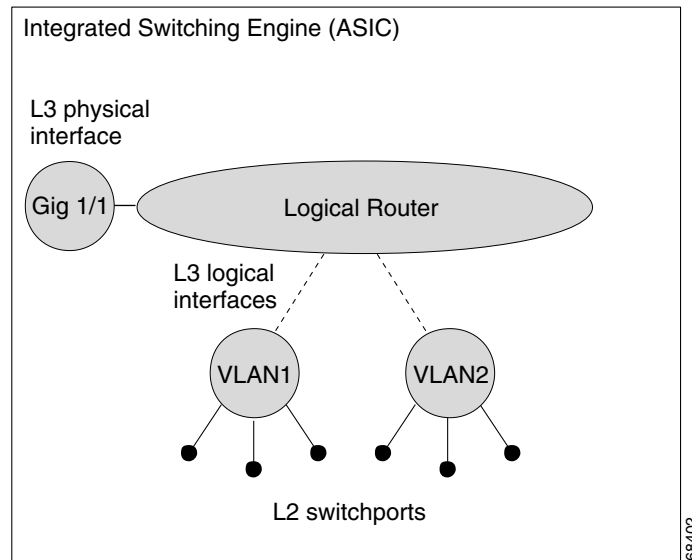
This section contains the following subsections:

CEF, MFIB, and Layer 2 Forwarding

The implementation of IP multicast on the Catalyst 4500 Metro series switch is an extension of centralized Cisco Express Forwarding (CEF). CEF extracts information from the unicast routing table, which is created by unicast routing protocols, such as BGP, OSPF, and EIGR and loads it into the hardware Forwarding Information Base (FIB). With the unicast routes in the FIB, when a route is changed in the upper-layer routing table, only one route needs to be changed in the hardware routing state. To forward unicast packets in hardware, the Integrated Switching Engine looks up source and destination routes in ternary content addressable memory (TCAM), takes the adjacency index from the hardware FIB, and gets the Layer 2 rewrite information and next-hop address from the hardware adjacency table.

The new Multicast Forwarding Information Base (MFIB) subsystem is the multicast analog of the unicast CEF. The MFIB subsystem extracts the multicast routes that PIM and IGMP create and refines them into a protocol-independent format for forwarding in hardware. The MFIB subsystem removes the protocol-specific information and leaves only the essential forwarding information. Each entry in the MFIB table consists of an (S,G) or (*,G) route, an input RPF VLAN, and a list of Layer 3 output interfaces. The MFIB subsystem, together with platform-dependent management software, loads this multicast routing information into the hardware FIB and hardware multicast expansion table (MET).

Note   On the Supervisor Engine 6-E, MET has been replaced by the RET (Replica Expansion Table).

The Catalyst 4500 Metro series switch performs Layer 3 routing and Layer 2 bridging at the same time. There can be multiple Layer 2 switchports on any VLAN interface. To determine the set of output switchports on which to forward a multicast packet, the Supervisor Engine III combines Layer 3 MFIB information with Layer 2 forwarding information and stores it in the hardware MET for packet replication.

Figure 32-3 shows a functional overview of how the Catalyst 4500 Metro series switch combines unicast routing, multicast routing, and Layer 2 bridging information to forward in hardware.

Figure 32-3   Combining CEF, MFIB, and Layer 2 Forwarding Information in Hardware
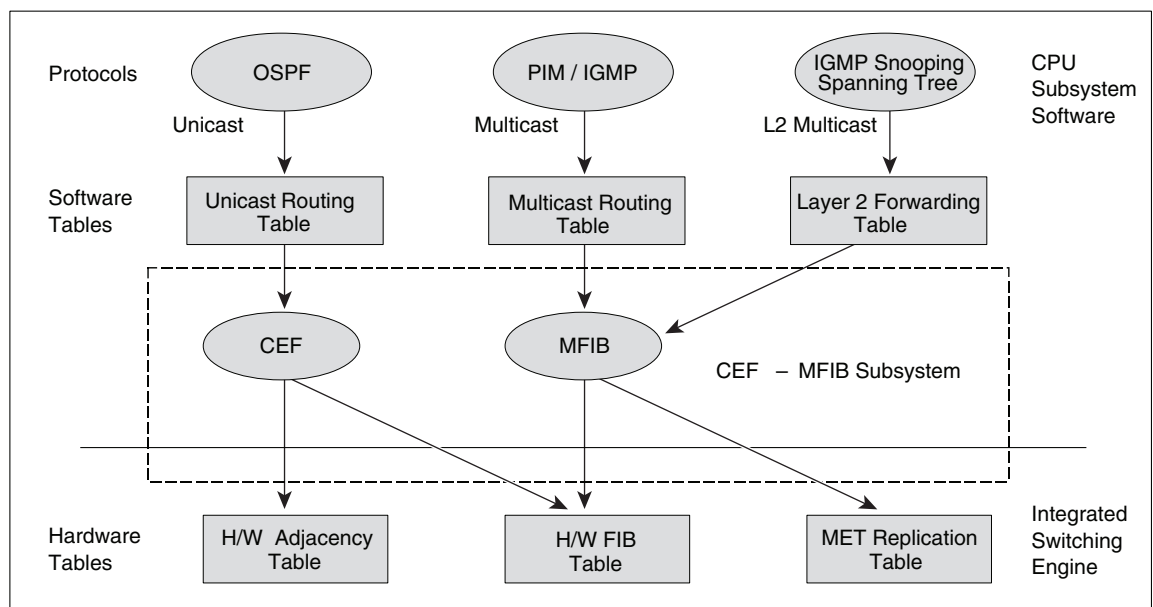
Like the CEF unicast routes, the MFIB routes are Layer 3 and must be merged with the appropriate Layer 2 information. The following example shows an MFIB route:

```
(*,224.1.2.3)
RPF interface is Vlan3
Output Interfaces are:
Vlan 1
Vlan 2
```

The route (*,224.1.2.3) is loaded in the hardware FIB table and the list of output interfaces is loaded into the MET. A pointer to the list of output interfaces, the MET index, and the RPF interface are also loaded in the hardware FIB with the (*,224.1.2.3) route. With this information loaded in hardware, merging of the Layer 2 information can begin. For the output interfaces on VLAN1, the Integrated Switching Engine must send the packet to all switchports in VLAN1 that are in the spanning tree forwarding state. The same process applies to VLAN 2. To determine the set of switchports in VLAN 2, the Layer 2 Forwarding Table is used.

When the hardware routes a packet, in addition to sending it to all of the switchports on all output interfaces, the hardware also sends the packet to all switchports (other than the one it arrived on) in the input VLAN. For example, assume that VLAN 3 has two switchports in it, Gig 3/1 and Gig 3/2. If a host

on Gig 3/1 sends a multicast packet, the host on Gig 3/2 might also need to receive the packet. To send a multicast packet to the host on Gig 3/2, all of the switchports in the ingress VLAN must be added to the portset that is loaded in the MET.
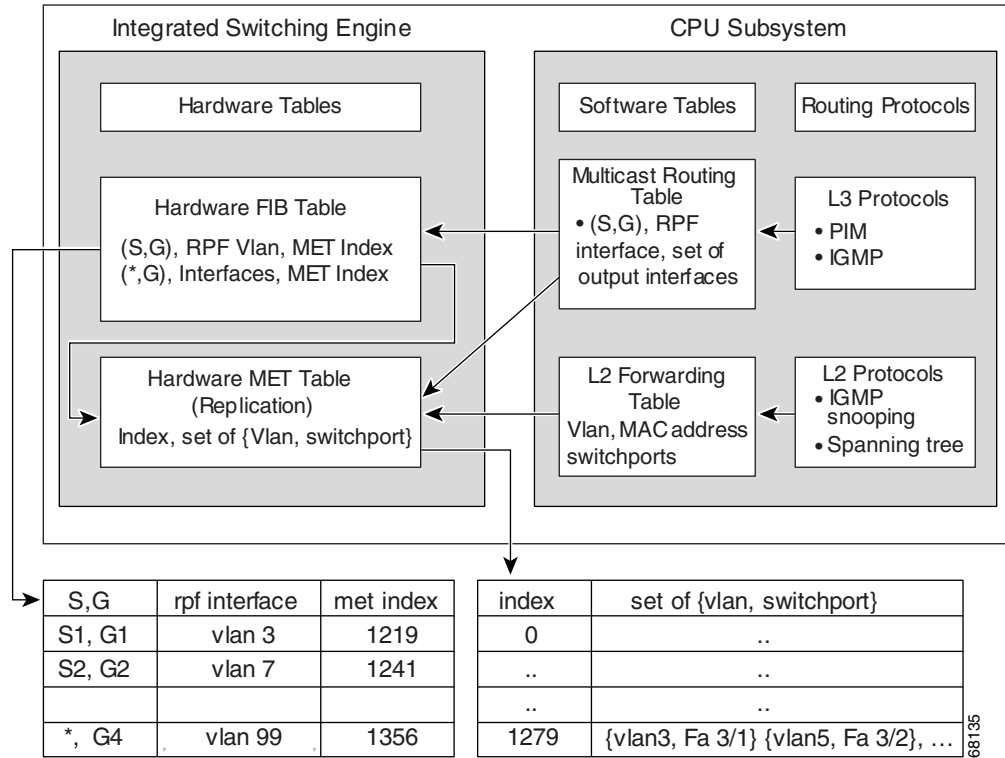
If VLAN 1 contains 1/1 and 1/2, VLAN 2 contains 2/1 and 2/2, and VLAN 3 contains 3/1 and 3/2, the MET chain for this route would contain these switchports: (1/1,1/2,2/1,2/2,3/1, and 3/2).

If IGMP snooping is on, the packet should not be forwarded to all output switchports on VLAN 2. The packet should be forwarded only to switchports where IGMP snooping has determined that there is either a group member or router. For example, if VLAN 1 had IGMP snooping enabled, and IGMP snooping determined that only port 1/2 had a group member on it, then the MET chain would contain these switchports: (1/1,1/2, 2/1, 2/2, 3/1, and 3/2).

IP Multicast Tables

Figure 32-4 shows some key data structures that the Catalyst 4500 Metro series switch uses to forward IP multicast packets in hardware.

Figure 32-4   IP Multicast Tables and Protocols

S,G	rpf interface	met index	index	set of {vlan, switchport}
S1, G1	vlan 3	1219	0	..
S2, G2	vlan 7	1241	..	..
			..	..
*, G4	vlan 99	1356	1279	{vlan3, Fa 3/1} {vlan5, Fa 3/2}, …

The Integrated Switching Engine maintains the hardware FIB table to identify individual IP multicast routes. Each entry consists of a destination group IP address and an optional source IP address. Multicast traffic flows on primarily two types of routes: (S,G) and (*,G). The (S,G) routes flow from a source to a group based on the IP address of the multicast source and the IP address of the multicast group destination. Traffic on a (*,G) route flows from the PIM RP to all receivers of group G. Only sparse-mode groups use (*,G) routes. The Integrated Switching Engine hardware contains space for a total of 128,000 routes, which are shared by unicast routes, multicast routes, and multicast fast-drop entries.

Output interface lists are stored in the multicast expansion table (MET). The MET has room for up to 32,000 output interface lists. (For RET, we can have up to 102K entries (32K used for floodsets, 70K used for mcast entries)). The MET resources are shared by both Layer 3 multicast routes and by Layer 2 multicast entries. The actual number of output interface lists available in hardware depends on the specific configuration. If the total number of multicast routes exceed 32,000, multicast packets might not be switched by the Integrated Switching Engine. They would be forwarded by the CPU subsystem at much slower speeds.

Note   For RET, a maximum of 102K entries is supported (32K used for floodsets, 70K used for mcast entries).

Note   Partial routing is not supported on Supervisor Engine 6-E; only hardware and software routing are supported.

Hardware and Software Forwarding

The Integrated Switching Engine forwards the majority of packets in hardware at very high rates of speed. The CPU subsystem forwards exception packets in software. Statistical reports should show that the Integrated Switching Engine is forwarding the vast majority of packets in hardware.

Figure 32-5 shows a logical view of the hardware and software forwarding components.

Figure 32-5   Hardware and Software Forwarding Components
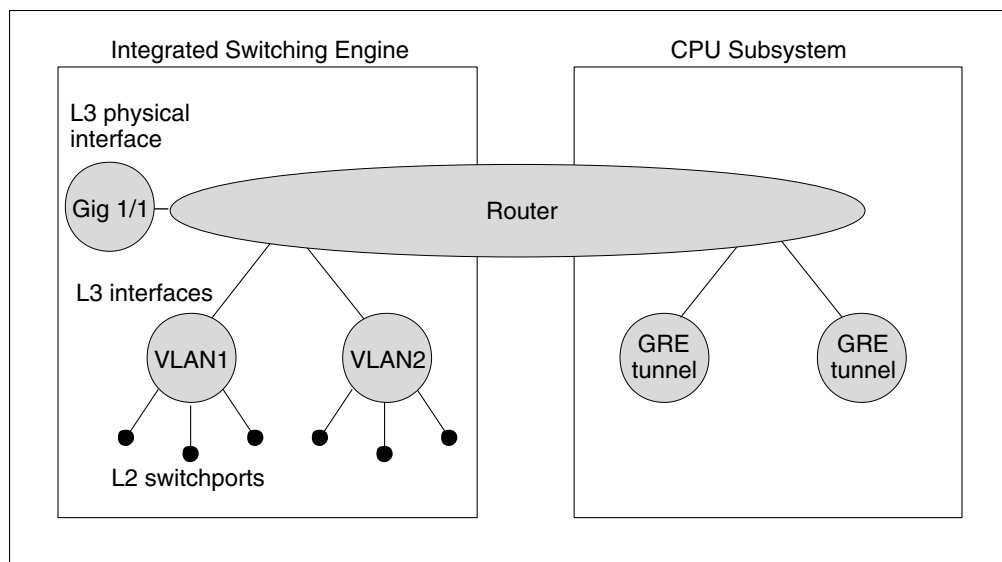
In the normal mode of operation, the Integrated Switching Engine performs inter-VLAN routing in hardware. The CPU subsystem supports generic routing encapsulation (GRE) tunnels for forwarding in software.

Replication is a particular type of forwarding where, instead of sending out one copy of the packet, the packet is replicated and multiple copies of the packet are sent out. At Layer 3, replication occurs only for multicast packets; unicast packets are never replicated to multiple Layer 3 interfaces. In IP multicasting, for each incoming IP multicast packet that is received, many replicas of the packet are sent out.

IP multicast packets can be transmitted on the following types of routes:

- Hardware routes
- Software routes
- Partial routes

Hardware routes occur when the Integrated Switching Engine hardware forwards all replicas of a packet. Software routes occur when the CPU subsystem software forwards all replicas of a packet. Partial routes occur when the Integrated Switching Engine forwards some of the replicas in hardware and the CPU subsystem forwards some of the replicas in software.

Partial Routes

> **Note**   The conditions listed below cause the replicas to be forwarded by the CPU subsystem software, but the performance of the replicas that are forwarded in hardware is not affected.

The following conditions cause some replicas of a packet for a route to be forwarded by the CPU subsystem:

- The switch is configured with the **ip igmp join-group** command as a member of the IP multicast group on the RPF interface of the multicast source.
- The switch is the first-hop to the source in PIM sparse mode. In this case, the switch must send PIM-register messages to the RP.

Software Routes

> **Note**   If any one of the following conditions is configured on the RPF interface or the output interface, all replication of the output is performed in software.

The following conditions cause all replicas of a packet for a route to be forwarded by the CPU subsystem software:

- The interface is configured with multicast helper.
- The interface is a generic routing encapsulation (GRE) or Distance Vector Multicast Routing Protocol (DVMRP) tunnel.
- The interface uses non-Advanced Research Products Agency (ARPA) encapsulation.

The following packets are always forwarded in software:

- Packets sent to multicast groups that fall into the range 224.0.0.* (where * is in the range from 0 to 255). This range is used by routing protocols. Layer 3 switching supports all other multicast group addresses.
- Packets with IP options.

Non-Reverse Path Forwarding Traffic

Traffic that fails an Reverse Path Forwarding (RPF) check is called non-RPF traffic. Non-RPF traffic is forwarded by the Integrated Switching Engine by filtering (persistently dropping) or rate limiting the non-RPF traffic.

In a redundant configuration where multiple Layer 3 switches or routers connect to the same LAN segment, only one device forwards the multicast traffic from the source to the receivers on the outgoing interfaces. Figure 32-6 shows how Non-RPF traffic can occur in a common network configuration.

Figure 32-6   Redundant Multicast Router Configuration in a Stub Network
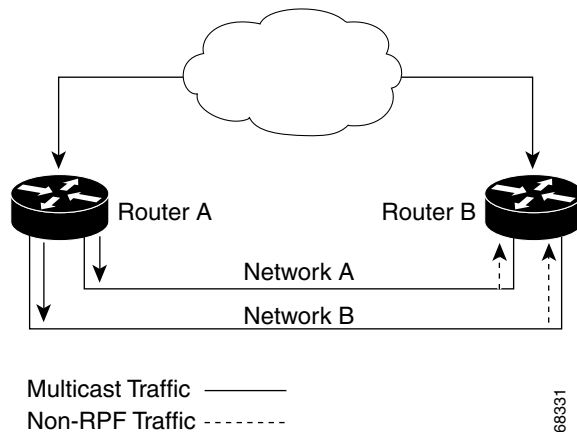
In this kind of topology, only Router A, the PIM designated router (PIM DR), forwards data to the common VLAN. Router B receives the forwarded multicast traffic, but must drop this traffic because it has arrived on the wrong interface and fails the RPF check. Traffic that fails the RPF check is called non-RPF traffic.

Multicast Fast Drop

In IP multicast protocols, such as PIM-SM and PIM-DM, every (S,G) or (*,G) route has an incoming interface associated with it. This interface is referred to as the reverse path forwarding interface. In some cases, when a packet arrives on an interface other than the expected RPF interface, the packet must be forwarded to the CPU subsystem software to allow PIM to perform special protocol processing on the packet. One example of this special protocol processing that PIM performs is the PIM Assert protocol.

By default, the Integrated Switching Engine hardware sends all packets that arrive on a non-RPF interface to the CPU subsystem software. However, processing in software is not necessary in many cases, because these non-RPF packets are often not needed by the multicast routing protocols. The problem is that if no action is taken, the non-RPF packets that are sent to the software can overwhelm the CPU.

To prevent this from happening, the CPU subsystem software loads fast-drop entries in the hardware when it receives an RPF failed packet that is not needed by the PIM protocols running on the switch. A fast-drop entry is keyed by (S,G, incoming interface). Any packet matching a fast-drop entry is bridged in the ingress VLAN, but is not sent to the software, so the CPU subsystem software is not overloaded by processing these RPF failures unnecessarily.

Protocol events, such as a link going down or a change in the unicast routing table, can impact the set of packets that can safely be fast dropped. A packet that was correctly fast dropped before might, after a topology change, need to be forwarded to the CPU subsystem software so that PIM can process it. The CPU subsystem software handles flushing fast-drop entries in response to protocol events so that the PIM code in IOS can process all the necessary RPF failures.

The use of fast-drop entries in the hardware is critical in some common topologies because it is possible to have persistent RPF failures. Without the fast-drop entries, the CPU would be exhausted by RPF failed packets that it did not need to process.

Multicast Forwarding Information Base

The Multicast Forwarding Information Base (MFIB) subsystem supports IP multicast routing in the Integrated Switching Engine hardware on the Catalyst 4500 Metro series switch. The MFIB logically resides between the IP multicast routing protocols in the CPU subsystem software (PIM, IGMP, MSDP, MBGP, and DVMRP) and the platform-specific code that manages IP multicast routing in hardware. The MFIB translates the routing table information created by the multicast routing protocols into a simplified format that can be efficiently processed and used for forwarding by the Integrated Switching Engine hardware.

To display the information in the multicast routing table, use the **show ip mroute** command. To display the MFIB table information, use the **show ip mfib** command.

The MFIB table contains a set of IP multicast routes. There are several types of IP multicast routes, including (S,G) and (*,G) routes. Each route in the MFIB table can have one or more optional flags associated with it. The route flags indicate how a packet that matches a route should be forwarded. For example, the Internal Copy (IC) flag on an MFIB route indicates that a process on the switch needs to receive a copy of the packet. The following flags can be associated with MFIB routes:

- Internal Copy (IC) flag—set on a route when a process on the router needs to receive a copy of all packets matching the specified route
- Signalling (S) flag—set on a route when a process needs to be notified when a packet matching the route is received; the expected behavior is that the protocol code updates the MFIB state in response to receiving a packet on a signalling interface
- Connected (C) flag-—when set on an MFIB route, has the same meaning as the Signalling (S) flag, except that the C flag indicates that only packets sent by directly connected hosts to the route should be signalled to a protocol process

A route can also have a set of optional flags associated with one or more interfaces. For example, an (S,G) route with the flags on VLAN 1 indicates how packets arriving on VLAN 1 should be treated, and they also indicate whether packets matching the route should be forwarded onto VLAN 1. The per-interface flags supported in the MFIB include the following:

- Accepting (A)—set on the interface that is known in multicast routing as the RPF interface. A packet that arrives on an interface that is marked as Accepting (A) is forwarded to all Forwarding (F) interfaces.
- Forwarding (F)—used in conjunction with the Accepting (A) flag as described above. The set of Forwarding interfaces that form what is often referred to as the multicast "olist" or output interface list.
- Signalling (S)—set on an interface when some multicast routing protocol process in IOS needs to be notified of packets arriving on that interface.

> **Note**   When PIM-SM routing is in use, the MFIB route might include an interface like in this example: PimTunnel [1.2.3.4]. This is a virtual interface that the MFIB subsystem creates to indicate that packets are being tunnelled to the specified destination address. A PimTunnel interface cannot be displayed with the normal **show interface** command.

S/M, 224/4

An (S/M, 224/4) entry is created in the MFIB for every multicast-enabled interface. This entry ensures that all packets sent by directly connected neighbors can be Register-encapsulated to the PIM-SM RP. Typically, only a small number of packets would be forwarded using the (S/M,224/4) route, until the (S,G) route is established by PIM-SM.

For example, on an interface with IP address 10.0.0.1 and netmask 255.0.0.0, a route would be created matching all IP multicast packets in which the source address is anything in the class A network 10. This route can be written in conventional subnet/masklength notation as (10/8,224/4). If an interface has multiple assigned IP addresses, then one route is created for each such IP address.

Unsupported Feature

The following IP multicast feature is not supported in this release:

- Controlling the transmission rate to a multicast group

Configuring IP Multicast Routing

The following sections describe IP multicast routing configuration tasks:

- Default Configuration in IP Multicast Routing, page 32-12
- Enabling IP Multicast Routing, page 32-13
- Enabling PIM on an Interface, page 32-13
- Enabling PIM-SSM Mapping, page 32-15
- Configuring a Rendezvous Point, page 32-15
- Configuring a Single Static RP, page 32-19

For more detailed information on IP multicast routing, such as Auto-RP, PIM Version 2, and IP multicast static routes, refer to the *Cisco IOS IP and IP Routing Configuration Guide, Cisco IOS Release 12.3.*

Default Configuration in IP Multicast Routing

Table 32-1 shows the IP multicast default configuration.

Table 32-1   Default IP Multicast Configuration

Feature	Default Value
Rate limiting of RPF	Enabled globally
IP multicast routing	Disabled globally **Note** When IP multicast routing is disabled, IP multicast traffic data packets are not forwarded by the Catalyst 4500 Metro series switch. However, IP multicast control traffic will continue to be processed and forwarded. Therefore, IP multicast routes can remain in the routing table even if IP multicast routing is disabled.

Table 32-1   Default IP Multicast Configuration

Feature	Default Value
PIM	Disabled on all interfaces
IGMP snooping	Enabled on all VLAN interfaces **Note** If you disable IGMP snooping on an interface, all output ports are forwarded by the Integrated Switching Engine. When IGMP snooping is disabled on an input VLAN interface, multicast packets related to that interface are sent to all forwarding switchports in the VLAN.

Note   Source-specific multicast and IGMP v3 are supported.

For more information about source-specific multicast with IGMPv3 and IGMP, see the following URL:

http://www.cisco.com/en/US/docs/ios/ipmulti/configuration/guide/imc_cfg_ssm.html

Enabling IP Multicast Routing

Enabling IP multicast routing allows the Catalyst 4500 Metro series switch to forward multicast packets. To enable IP multicast routing on the router, perform this task in global configuration mode:

Command	Purpose
Switch(config)# **ip multicast-routing**	Enables IP multicast routing.

Enabling 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 Layer 3 switch or router populates its multicast routing table and how the Layer 3 switch or 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.

When the switch populates 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 can send join messages 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.

Enabling Dense Mode

To configure PIM on an interface to be in dense mode, perform this task:

Command	Purpose
Switch(config-if)# **ip pim dense-mode**	Enables dense-mode PIM on the interface.

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.

Enabling Sparse Mode

To configure PIM on an interface to be in sparse mode, perform this task:

Command	Purpose
Switch(config-if)# **ip pim sparse-mode**	Enables sparse-mode PIM on the interface.

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.

Enabling Sparse-Dense Mode

When you enter either the **ip pim sparse-mode** or **ip pim dense-mode** command, 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. If you want to treat the group as a sparse group, and the interface is in sparse-dense mode, you must have an RP.

If you configure sparse-dense mode, the idea of sparseness or denseness is applied to the group on the switch, 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:

- When there are members or DVMRP neighbors on the interface
- When there are PIM neighbors and the group has not 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:

- When there are members or DVMRP neighbors on the interface
- When 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 this task:

Command	Purpose
`Switch(config-if)# ip pim sparse-dense-mode`	Enables PIM to operate in sparse or dense mode, depending on the group.

Enabling PIM-SSM Mapping

Catalyst 4500 series switch supports SSM mapping. This enables an SSM transition in cases where neither URD nor IGMP v3lite is available, or when supporting SSM on the end system is impossible or unwanted due to administrative or technical reasons. With SSM mapping, you can leverage SSM for video delivery to legacy set-top boxes (STBs) that do not support IGMPv3 or for applications that do not take advantage of the IGMPv3 host stack.

For more details, refer to this URL:

http://www.cisco.com/en/US/docs/ios/12_3t/12_3t2/feature/guide/gtssmma.html

Configuring a Rendezvous Point

A rendezvous point (RP) is required in networks running Protocol Independent Multicast sparse mode (PIM-SM). In PIM-SM, traffic will be forwarded only to network segments with active receivers that have explicitly requested multicast data.

There are different ways to configure a Rendezvous Point. The most commonly used, described here, are the usage of Static RP and the usage of the Auto-RP protocol. Another way, not described here, are the use of the Bootstrap Router (BSR) protocol. For more information see http://www.cisco.com/en/US/docs/ios/12_2/ip/configuration/guide/1cfmulti.html.

Configuring Auto-RP

Auto-rendezvous point (Auto-RP) automates the distribution of group-to-rendezvous point (RP) mappings in a PIM network. 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 by way of dense mode flooding.

Thus, all routers automatically discover which RP to use for the groups they support. The Internet Assigned Numbers Authority (IANA) has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP.

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

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing**

4. **interface** *type number*

5. **ip pim [sparse-mode | sparse-dense-mode]**

6. **exit**

7. Repeat Steps 4 and 5 on all PIM interfaces.

8. **ip pim autorp listener**

9. **ip pim send-rp-announce** {**interface-type interface-number** | **ip-address**} **scope ttl-value** [*group-list access-list*] [*interval seconds*] [*bidir*]

10. **ip pim send-rp-discovery** [*interface-type interface-number*] **scope ttl-value** [*interval seconds*]

11. **ip pim rp-announce-filter** *rp-list access-list group-list access-list*

12. **interface** *type number*

13. **ip multicast boundary access-list** [*filter-autorp*]

14. **end**

15. **show ip pim autorp**

16. **show ip pim rp** [*mapping*] [*rp-address*]

17. **show ip igmp groups** [*group-name | group-address | interface-type interface-number*] [**detail**]

18. **show ip mroute** [*group-address | group-name*] [*source-address | source-name*] [*interface-type interface-number*] [**summary**] [**count**] [**active kbps**]

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip multicast-routing** **Example:** `Router(config)# ip multicast-routing`	Enables IP multicast routing.
Step 4	**interface** *type number* **Example:** `Router(config)# interface ethernet 1`	Selects an interface that is connected to hosts on which PIM can be enabled.
Step 5	**ip pim** [**sparse-mode**	**sparse-dense-mode**] **Example:** `Router(config-if)# ip pim sparse-mode`	Enables PIM sparse or sparse-dense mode on an interface. When configuring Auto-RP in sparse mode, you must also configure the Auto-RP listener in the next step.

	Command or Action	Purpose

Step 6	`exit`
	
	Example:
	`Router(config-if)# end`
		Returns to global configuration mode.

Step 7	Repeat Steps 4 and 5 on all PIM interfaces.	—

Step 8	`ip pim autorp listener`
	
	Example:
	`Router(config)# ip pim autorp listener`
		Causes IP multicast traffic for the two Auto-RP groups 224.0.1.39 and 224.0.1.40 to be PIM dense mode flooded across interfaces operating in PIM sparse mode.
		
		• Skip this step if you are configuring sparse-dense mode in Step 8.

Step 9	`ip pim send-rp-announce {`*interface-type interface-number* `|` *ip-address*`}` **scope** *ttl-value* [**group-list** *access-list*] [**interval** *seconds*] [**bidir**] **Example:** `Router(config)# ip pim send-rp-announce loopback0 scope 31 group-list 5`	Sends RP announcements out all PIM-enabled interfaces. • Perform this step on the RP router only. • Use the *interface-type* and *interface-number* arguments to define which IP address is to be used as the RP address. • Use the *ip-address* argument to specify a directly connected IP address as the RP address. **Note** If the *ip-address* argument is configured for this command, the RP-announce message will be sourced by the interface to which this IP address is connected (that is, the source address in the IP header of the RP-announce message is the IP address of that interface). • This example shows that the interface is enabled with a maximum of 31 hops. The IP address by which the router wants to be identified as RP is the IP address associated with loopback interface 0. Access list 5 describes the groups for which this router serves as RP.

Command or Action	Purpose
Step 10 `ip pim send-rp-discovery` [*interface-type interface-number*] **scope** *ttl-value* [**interval** *seconds*] **Example:** `Router(config)# ip pim send-rp-discovery loopback 1 scope 31`	Configures the router to be an RP mapping agent. • Perform this step on the RP router only. • Use the optional *interface-type* and *interface-number* arguments to define which IP address is to be used as the source address of the RP mapping agent. • Use the **scope** keyword and *ttl-value* argument to specify the Time-to-Live (TTL) value in the IP header of Auto-RP discovery messages. • Use the optional **interval** keyword and *seconds* argument to specify the interval at which Auto-RP discovery messages are sent. **Note** Lowering the interval at which Auto-RP discovery messages are sent from the default value of 60 seconds results in more frequent floodings of the group-to-RP mappings. In some network environments, the disadvantages of lowering the interval (more control packet overhead) may outweigh the advantages (more frequent group-to-RP mapping updates). • The example shows limiting the Auto-RP discovery messages to 31 hops on loopback interface 1.

Step 11 `ip pim rp-announce-filter rp-list` *access-list* **group-list** *access-list*
 	
Example:
`Router(config)# ip pim rp-announce-filter rp-list 1 group-list 2`
	Filters incoming Auto-RP announcement messages coming from the RP.
	
	• Perform this step on the RP router only.
	
	• Two example access lists that apply to this step could be:
	
	`access-list 1 permit 10.0.0.1`
	`access-list 1 permit 10.0.0.2`
	`access-list 2 permit 224.0.0.0 15.255.255.255`

Step 12 `interface` *type number* **Example:** `Router(config)# interface ethernet 1`	Selects an interface that is connected to hosts on which PIM can be enabled.
Step 13 `ip multicast boundary` *access-list* [**filter-autorp**] **Example:** `Router(config-if)# ip multicast boundary 10 filter-autorp`	Configures an administratively scoped boundary. • Perform this step on the interfaces that are boundaries to other routers. • The access list is not shown in this task. • An access list entry that uses the **deny** keyword creates a multicast boundary for packets that match that entry.
Step 14 `end` **Example:** `Router(config-if)# end`	Returns to EXEC mode.

Command or Action	Purpose
Step 15 `show ip pim autorp` **Example:** `Router# show ip pim autorp`	(Optional) Displays the Auto-RP information.
Step 16 `show ip pim rp` [`mapping`] [*rp-address*] **Example:** `Router# show ip pim rp mapping`	(Optional) Displays RPs known in the network and shows how the router learned about each RP.
Step 17 `show ip igmp groups` [*group-name* | *group-address* | *interface-type interface-number*] [`detail`] **Example:** `Router# show ip igmp groups`	(Optional) Displays the multicast groups having receivers that are directly connected to the router and that were learned through Internet Group Management Protocol (IGMP). • A receiver must be active on the network at the time that this command is issued in order for receiver information to be present on the resulting display.
Step 18 `show ip mroute` [*group-address* | *group-name*] [*source-address* | *source-name*] [*interface-type interface-number*] [`summary`] [`count`] [`active kbps`] **Example:** `Router# show ip mroute cbone-audio`	(Optional) Displays the contents of the IP multicast routing (mroute) table.

Configuring a Single Static RP

If you are configuring PIM sparse mode, you must configure a PIM RP for a multicast group. An RP can either be configured statically in each device, or learned through a dynamic mechanism. This task explains how to statically configure an RP, as opposed to the router learning the RP through a dynamic mechanism such as Auto-RP.

PIM designated routers (DRs) 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. 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 that are defined by an access list. If no RP is configured for a group, the router treats the group as dense using the PIM dense mode techniques. (You can prevent this occurrence by configuring the no ip pim dm-fallback command.)

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

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing**

4. **interface** *type number*

5. **ip pim [sparse-mode | sparse-dense-mode]**

6. Repeat Steps 4 and 5 on every instance that uses IP multicast.

7. **exit**

8. **ip pim rp-address rp-address [***access-list***] [***override***]**

9. **end**

10. **show ip pim rp** [*mapping*] [*rp-address*]

11. **show ip igmp groups** [*group-name | group-address | interface-type interface-number*] [**detail**]

12. **show ip mroute** [*group-address | group-name*] [*source-address | source-name*] [*interface-type interface-number*] [**summary**] [**count**] [**active kbps**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		
		• Enter your password if prompted.

Step 2	`configure terminal`
	
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`ip multicast-routing`
	
	Example:
	`Router(config)# ip multicast-routing`
		Enables IP multicast routing.

Step 4	`interface` *type number* **Example:** `Router(config)# interface ethernet 1`	Selects an interface that is connected to hosts on which PIM can be enabled.
Step 5	`ip pim [sparse-mode	sparse-dense-mode]`  **Example:** `Router(config-if)# ip pim sparse-mode`	Enables PIM on an interface. You must use sparse mode.
Step 6	Repeat Steps 4 and 5 on every interface that uses IP multicast.	—
Step 7	`exit` **Example:** `Router(config-if)# `**`exit`**	Returns to global configuration mode.

	Command or Action	Purpose
Step 8	**ip pim rp-address** *rp-address* [*access-list*] [**override**] **Example:** Router(config)# ip pim rp-address 192.168.0.0	Configures the address of a PIM RP for a particular group. • Perform this step on any router. • The *access-list* argument specifies the number or name of an access list that defines for which multicast groups the RP should be used. • The **override** keyword specifies that if there is a conflict between the RP configured with this command and one learned by Auto-RP, the RP configured with this command prevails.
Step 9	**end** **Example:** Router(config)# end	Ends the current configuration session and returns to EXEC mode.
Step 10	**show ip pim rp** [**mapping**] [*rp-address*] **Example:** Router# show ip pim rp mapping	(Optional) Displays RPs known in the network and shows how the router learned about each RP.
Step 11	**show ip igmp groups** [*group-name* | *group-address* | *interface-type interface-number*] [**detail**] **Example:** Router# show ip igmp groups	(Optional) Displays the multicast groups having receivers that are directly connected to the router and that were learned through IGMP. • A receiver must be active on the network at the time that this command is issued in order for receiver information to be present on the resulting display.
Step 12	**show ip mroute** [*group-address* | *group-name*] [*source-address* | *source-name*] [*interface-type interface-number*] [**summary**] [**count**] [**active** *kbps*] **Example:** Router# show ip mroute cbone-audio	(Optional) Displays the contents of the IP multicast routing (mroute) table.

Monitoring and Maintaining 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 how to monitor and maintain IP multicast:

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables 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, you can perform any of these tasks:

Command	Purpose
Switch# **ping** [*group-name* | *group-address*]	Sends an ICMP Echo Request to a multicast group address.
Switch# **show ip mroute** [*hostname* | *group_number*]	Displays the contents of the IP multicast routing table.
Switch# **show ip pim interface** [*type number*] [**count**]	Displays information about interfaces configured for PIM.
Switch# **show ip interface**	Displays PIM information for all interfaces.

Displaying the Multicast Routing Table

The following is sample output from the **show ip mroute** command for a router operating in dense mode. This command displays the contents of the IP multicast FIB table for the multicast group named cbone-audio.

```
Switch# show ip mroute cbone-audio

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
R - RP-bit set, F - Register flag, T - SPT-bit set
Timers: Uptime/Expires
Interface state: Interface, Next-Hop, State/Mode

(*, 224.0.255.1), uptime 0:57:31, expires 0:02:59, RP is 0.0.0.0, flags: DC
    Incoming interface: Null, RPF neighbor 0.0.0.0, Dvmrp
    Outgoing interface list:
    Ethernet0, Forward/Dense, 0:57:31/0:02:52
    Tunnel0, Forward/Dense, 0:56:55/0:01:28

(198.92.37.100/32, 224.0.255.1), uptime 20:20:00, expires 0:02:55, flags: C
    Incoming interface: Tunnel0, RPF neighbor 10.20.37.33, Dvmrp
    Outgoing interface list:
    Ethernet0, Forward/Dense, 20:20:00/0:02:52
```

The following is sample output from the **show ip mroute** command for a router operating in sparse mode:

```
Switch# show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
R - RP-bit set, F - Register flag, T - SPT-bit set
Timers: Uptime/Expires
Interface state: Interface, Next-Hop, State/Mode

(*, 224.0.255.3), uptime 5:29:15, RP is 198.92.37.2, flags: SC
```

```
    Incoming interface: Tunnel0, RPF neighbor 10.3.35.1, Dvmrp
    Outgoing interface list:
       Ethernet0, Forward/Sparse, 5:29:15/0:02:57

(198.92.46.0/24, 224.0.255.3), uptime 5:29:15, expires 0:02:59, flags: C
  Incoming interface: Tunnel0, RPF neighbor 10.3.35.1
  Outgoing interface list:
     Ethernet0, Forward/Sparse, 5:29:15/0:02:57
```

Note Interface timers are not updated for hardware-forwarded packets. Entry timers are updated approximately every five seconds.

The following is sample output from the **show ip mroute** command with the **summary** keyword:

```
Switch# show ip mroute summary

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
Timers: Uptime/Expires
Interface state: Interface, Next-Hop, State/Mode

(*, 224.255.255.255), 2d16h/00:02:30, RP 171.69.10.13, flags: SJPC

(*, 224.2.127.253), 00:58:18/00:02:00, RP 171.69.10.13, flags: SJC

(*, 224.1.127.255), 00:58:21/00:02:03, RP 171.69.10.13, flags: SJC

(*, 224.2.127.254), 2d16h/00:00:00, RP 171.69.10.13, flags: SJCL
  (128.9.160.67/32, 224.2.127.254), 00:02:46/00:00:12, flags: CLJT
  (129.48.244.217/32, 224.2.127.254), 00:02:15/00:00:40, flags: CLJT
  (130.207.8.33/32, 224.2.127.254), 00:00:25/00:02:32, flags: CLJT
  (131.243.2.62/32, 224.2.127.254), 00:00:51/00:02:03, flags: CLJT
  (140.173.8.3/32, 224.2.127.254), 00:00:26/00:02:33, flags: CLJT
  (171.69.60.189/32, 224.2.127.254), 00:03:47/00:00:46, flags: CLJT
```

The following is sample output from the **show ip mroute** command with the **active** keyword:

```
Switch# show ip mroute active

Active IP Multicast Sources - sending >= 4 kbps

Group: 224.2.127.254, (sdr.cisco.com)
   Source: 146.137.28.69 (mbone.ipd.anl.gov)
     Rate: 1 pps/4 kbps(1sec), 4 kbps(last 1 secs), 4 kbps(life avg)

Group: 224.2.201.241, ACM 97
   Source: 130.129.52.160 (webcast3-e1.acm97.interop.net)
     Rate: 9 pps/93 kbps(1sec), 145 kbps(last 20 secs), 85 kbps(life avg)

Group: 224.2.207.215, ACM 97
   Source: 130.129.52.160 (webcast3-e1.acm97.interop.net)
     Rate: 3 pps/31 kbps(1sec), 63 kbps(last 19 secs), 65 kbps(life avg)
```

The following is sample output from the **show ip mroute** command with the **count** keyword:

```
Switch# show ip mroute count

IP Multicast Statistics - Group count: 8, Average sources per group: 9.87
Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second

Group: 224.255.255.255, Source count: 0, Group pkt count: 0
  RP-tree: 0/0/0/0
```

```
            Group: 224.2.127.253, Source count: 0, Group pkt count: 0
              RP-tree: 0/0/0/0

            Group: 224.1.127.255, Source count: 0, Group pkt count: 0
              RP-tree: 0/0/0/0

            Group: 224.2.127.254, Source count: 9, Group pkt count: 14
              RP-tree: 0/0/0/0
              Source: 128.2.6.9/32, 2/0/796/0
              Source: 128.32.131.87/32, 1/0/616/0
              Source: 128.125.51.58/32, 1/0/412/0
              Source: 130.207.8.33/32, 1/0/936/0
              Source: 131.243.2.62/32, 1/0/750/0
              Source: 140.173.8.3/32, 1/0/660/0
              Source: 146.137.28.69/32, 1/0/584/0
              Source: 171.69.60.189/32, 4/0/447/0
              Source: 204.162.119.8/32, 2/0/834/0

            Group: 224.0.1.40, Source count: 1, Group pkt count: 3606
              RP-tree: 0/0/0/0
              Source: 171.69.214.50/32, 3606/0/48/0, RPF Failed: 1203

            Group: 224.2.201.241, Source count: 36, Group pkt count: 54152
              RP-tree: 7/0/108/0
              Source: 13.242.36.83/32, 99/0/123/0
              Source: 36.29.1.3/32, 71/0/110/0
              Source: 128.9.160.96/32, 505/1/106/0
              Source: 128.32.163.170/32, 661/1/88/0
              Source: 128.115.31.26/32, 192/0/118/0
              Source: 128.146.111.45/32, 500/0/87/0
              Source: 128.183.33.134/32, 248/0/119/0
              Source: 128.195.7.62/32, 527/0/118/0
              Source: 128.223.32.25/32, 554/0/105/0
              Source: 128.223.32.151/32, 551/1/125/0
              Source: 128.223.156.117/32, 535/1/114/0
              Source: 128.223.225.21/32, 582/0/114/0
              Source: 129.89.142.50/32, 78/0/127/0
              Source: 129.99.50.14/32, 526/0/118/0
              Source: 130.129.0.13/32, 522/0/95/0
              Source: 130.129.52.160/32, 40839/16/920/161
              Source: 130.129.52.161/32, 476/0/97/0
              Source: 130.221.224.10/32, 456/0/113/0
              Source: 132.146.32.108/32, 9/1/112/0
```

> **Note** Multicast route byte and packet statistics are supported only for the first 1024 multicast routes. Output interface statistics are not maintained.

Displaying IP MFIB

You can display all routes in the MFIB, including routes that might not exist directly in the upper-layer routing protocol database but that are used to accelerate fast switching. These routes appear in the MFIB, even if dense-mode forwarding is in use.

To display various MFIB routing routes, perform one of these tasks:

Command	Purpose
Switch# **show ip mfib**	Displays the (S,G) and (*,G) routes that are used for packet forwarding. Displays counts for fast, slow, and partially-switched packets for every multicast route.
Switch# **show ip mfib all**	Displays all routes in the MFIB, including routes that may not exist directly in the upper-layer routing protocol database, but that are used to accelerate fast switching.These routes include the (S/M,224/4) routes.
Switch# **show ip mfib log** [n]	Displays a log of the most recent n MFIB related events, most recent first.
Switch# **show ip mfib counters**	Displays counts of MFIB related events. Only non-zero counters are shown.

The following is sample output from the **show ip mfib** command.

```
IP Multicast Forwarding Information Base
Entry Flags: C - Directly Connected, S - Signal,
             IC - Internal Copy
Interface Flags: A - Accept, F - Forward, S - Signal,
             NP - Not platform switched
Packets: Fast/Partial/Slow Bytes: Fast/Partial/Slow:
(171.69.10.13, 224.0.1.40), flags (IC)
   Packets: 2292/2292/0, Bytes: 518803/0/518803
   Vlan7 (A)
   Vlan100 (F NS)
   Vlan105 (F NS)
(*, 224.0.1.60), flags ()
   Packets: 2292/0/0, Bytes: 518803/0/0
   Vlan7 (A NS)
(*, 224.0.1.75), flags ()
   Vlan7 (A NS)
(10.34.2.92, 239.192.128.80), flags ()
   Packets: 24579/100/0, 2113788/15000/0 bytes
   Vlan7 (F NS)
   Vlan100 (A)
(*, 239.193.100.70), flags ()
   Packets: 1/0/0, 1500/0/0 bytes
   Vlan7 (A)
..
```

The fast-switched packet count represents the number of packets that were switched in hardware on the corresponding route.

The partially switched packet counter represents the number of times that a fast-switched packet was also copied to the CPU for software processing or for forwarding to one or more non-platform switched interfaces (such as a PimTunnel interface).

The slow-switched packet count represents the number of packets that were switched completely in software on the corresponding route.

Displaying PIM Statistics

The following is sample output from the **show ip pim interface** command:

```
Switch# show ip pim interface

Address          Interface         Mode     Neighbor   Query       DR
                                            Count      Interval
198.92.37.6      Ethernet0         Dense    2          30          198.92.37.33
198.92.36.129    Ethernet1         Dense    2          30          198.92.36.131
10.1.37.2        Tunnel0           Dense    1          30          0.0.0.0
```

The following is sample output from the **show ip pim interface** command with a **count**:

```
Switch# show ip pim interface count

Address          Interface         FS   Mpackets In/Out
171.69.121.35    Ethernet0         *    548305239/13744856
171.69.121.35    Serial0.33        *    8256/67052912
198.92.12.73     Serial0.1719      *    219444/862191
```

The following is sample output from the **show ip pim interface** command with a **count** when IP multicast is enabled. The example lists the PIM interfaces that are fast-switched and process-switched, and the packet counts for these. The H is added to interfaces where IP multicast is enabled.

```
Switch# show ip pim interface count

States: FS - Fast Switched, H - Hardware Switched
Address         Interface        FS   Mpackets In/Out
192.1.10.2      Vlan10           * H 40886/0
192.1.11.2      Vlan11           * H 0/40554
192.1.12.2      Vlan12           * H 0/40554
192.1.23.2      Vlan23           *   0/0
192.1.24.2      Vlan24           *   0/0
```

Clearing Tables and Databases

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

To clear IP multicast caches, tables, and databases, perform one of these tasks:

Command	Purpose
Switch# **clear ip mroute**	Deletes entries from the IP routing table.
Switch# **clear ip mfib counters**	Deletes all per-route and global MFIB counters.

Note   IP multicast routes can be regenerated in response to protocol events and as data packets arrive.

Configuration Examples

The following sections provide IP multicast routing configuration examples:

PIM Dense Mode: Example

This example is a configuration of dense-mode PIM on an Ethernet interface:

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

PIM Sparse Mode: Example

This example is a configuration of sparse-mode PIM. The RP router is the router with the address 10.8.0.20.

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

Sparse Mode with a Single Static RP: Example

The following example sets the PIM RP address to 192.168.1.1 for all multicast groups and defines all groups to operate in sparse mode:

```
ip multicast-routing
interface ethernet 1
 ip pim sparse-mode
ip pim rp-address 192.168.1.1
no ip pim dm-fallback
```

Note   The same RP cannot be used for both bidirectional and sparse mode groups.

The following example sets the PIM RP address to 172.16.1.1 for the multicast group 225.2.2.2 only:

```
access list 1 225.2.2.2 0.0.0.0
 ip pim rp-address 172.17.1.1
```

Sparse Mode with Auto-RP: Example

The following example configures sparse mode with Auto-RP:

```
ip multicast-routing
ip pim autorp listener
ip pim send-rp-announce Loopback0 scope 16 group-list 1
```

```
ip pim send-rp-discovery Loopback1 scope 16
no ip pim dm-fallback
access-list 1 permit 239.254.2.0 0.0.0.255
access-list 1 permit 239.254.3.0 0.0.0.255
.
.
.
access-list 10 permit 224.0.1.39
access-list 10 permit 224.0.1.40
access-list 10 permit 239.254.2.0 0.0.0.255
access-list 10 permit 239.254.3.0 0.0.0.255
```