

## Chapter 1

# Multicast Overview

The JUNOS software routing protocol process supports a wide variety of routing protocols. These routing protocols carry network information among routers not only for *unicast* traffic streams sent between one pair of clients and servers, but also for *multicast* traffic streams containing video, audio, or both, between a single server source and many client receivers. The routing protocols used for multicast differ in many key ways from unicast routing protocols.

This chapter discusses the following topics:

What Is Multicast? on page 4

IP Multicast Uses on page 5

IP Multicast Terminology on page 6

IP Multicast Building Blocks on page 9

The RPF Table on page 14

Protocols for Multicast on page 15

## What Is Multicast?

---

Information is delivered over a network by three basic methods: unicast, broadcast, and multicast.

The differences among unicast, broadcast, and multicast can be summarized as follows:

Unicast: One-to-one, from one source to one destination.

Broadcast: One-to-all, from one source to all possible destinations.

Multicast: One-to-many, from one source to multiple destinations expressing an interest in receiving the traffic.



**NOTE:** This list does not include a special category for many-to-many applications such as online gaming or videoconferencing, where there are many sources for the same receiver and receivers often double as sources. Many-to-many is a service model that repeatedly employs one-to-many multicast and therefore requires no unique protocol. The original multicast specification, RFC 1112, supports both the any-source multicast (ASM) many-to-many model and the source-specific multicast (SSM) one-to-many model.

---

With unicast traffic, many streams of IP packets that travel across networks flow from a single source, such as a Web site server, to a single destination such as a client PC. This is still the most common form of information transfer on networks.

Broadcast traffic flows from a single source to all possible destinations reachable on the network, which is usually a LAN. Broadcasting is the easiest way to make sure traffic reaches its destinations.

Television networks use broadcasting to distribute video and audio. Even if the television network is a cable television (CATV) system, the source signal reaches all possible destinations, which is the main reason that some channels' content is scrambled. Broadcasting is not feasible on the public Internet because of the enormous amount of unnecessary information that would constantly arrive at each end user's device, the complexities and impact of scrambling, and related privacy issues.

Multicast traffic lies between the extremes of unicast (one source, one destination) and broadcast (one source, all destinations). Multicast is a "one source, many destinations" method of traffic distribution, meaning only the destinations that explicitly indicate their need to receive the information from a particular source will receive the traffic stream.

On an IP network, because destinations (clients) do not often communicate directly with sources (servers), the routers between source and destination must be able to determine the topology of the network from the unicast or multicast perspective to avoid routing traffic haphazardly. Multicast routers replicate packets received on one input interface and send the copies out on multiple output interfaces.

In IP multicast, the source and destination are almost always hosts and not routers. Multicast routers distribute the multicast traffic across the network from source to destinations. The multicast router must find multicast sources on the network, send out copies of packets on several interfaces, prevent routing loops, connect interested destinations with the proper source, and keep the flow of unwanted packets to a minimum. Standard multicast routing protocols provide most of these capabilities, but some router architectures cannot send multiple copies of packets and so do not support multicasting directly.

## IP Multicast Uses

---

Multicast allows an IP network to support more than just the unicast model of data delivery that prevailed in the early stages of the Internet. Multicast, originally defined as a host extension in RFC 1112 in 1989, provides an efficient method for delivering traffic flows that can be characterized as one-to-many or many-to-many.

Unicast traffic is not strictly limited to data applications. Telephone conversations, wireless or not, contain digital audio samples and might contain digital photographs or even video and still flow from a single source to a single destination. In the same way, multicast traffic is not strictly limited to multimedia applications. In some data applications, the flow of traffic is from a single source to many destinations that require the packets, as in a news or stock ticker service delivered to many PCs. For this reason, the term *receiver* is preferred to *listener* for multicast destinations, although both terms are common.

Network applications that can function with unicast but are better suited for multicast include collaborative groupware, teleconferencing, periodic or “push” data delivery (stock quotes, sports scores, magazines, newspapers, and advertisements), server or Web site replication, and distributed interactive simulation (DIS) such as wargames or virtual reality. Any IP network concerned with reducing network resource overhead for one-to-many or many-to-many data or multimedia applications with multiple receivers benefits from multicast.

If unicast were employed by radio or news ticker services, each radio or PC would have to have a separate traffic session for each listener or viewer at a PC (this is actually the method for some Web-based services). The processing load and bandwidth consumed by the server would increase linearly as more people “tune in” to the server. This is extremely inefficient when dealing with the global scale of the Internet. Unicast places the burden of packet duplication on the server and consumes a lot of backbone bandwidth as the number of users grows.

If broadcast were employed instead, the source could generate a single IP packet stream using a broadcast destination address. Although this eliminates the server packet duplication issue, this is not a good solution for IP because IP broadcasts can only be sent to a single subnetwork, and IP routers normally isolate IP subnetworks on separate interfaces. Even if an IP packet stream could be addressed to literally go everywhere, and there were no need to “tune” to any source at all, broadcast would be extremely inefficient due to the bandwidth strain and need for uninterested hosts to discard large numbers of packets. Broadcast places the burden of packet rejection on each host and consumes the maximum amount of backbone bandwidth.

For radio station or news ticker traffic, multicast provides the most efficient and effective outcome, with none of the drawbacks and all of the advantages of the other methods. A single source of multicast packets finds its way to every *interested* receiver. As with broadcast, the transmitting host only generates a single stream of IP packets, so the load remains constant whether there is one receiver or one million. The network routers replicate the packets and deliver the packets to the proper receivers, but only the replication role is a new one for routers. The links leading to subnets consisting of entirely uninterested receivers carry no multicast traffic. Multicast minimizes the burden placed on sender, network, and receiver.

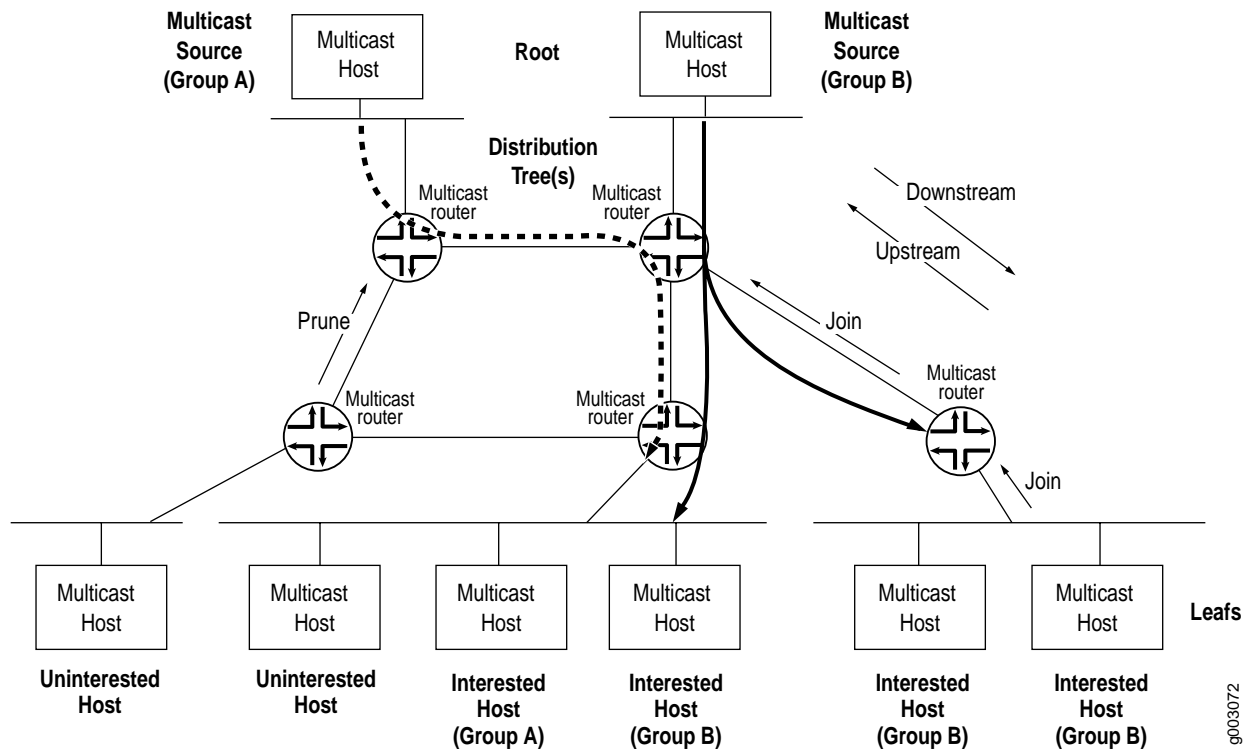
## IP Multicast Terminology

---

Multicast has its own particular set of terms and acronyms that apply to IP multicast routers and networks. Figure 1 on page 7 shows a general view of some of the terms commonly used in an IP multicast network.

In a multicast network, the key component is the *router*, able to replicate packets and therefore multicast-capable. The routers in the IP multicast network, which has exactly the same topology as the unicast network it is based on, use a *multicast routing protocol* to build a *distribution tree* to connect receivers (preferred to the multimedia implications of listeners, but listeners is also used) to *sources*. The distribution tree is rooted at the source. The interface on the router leading toward the source is the *upstream* interface, although the less precise terms *incoming* or *inbound* interface are used as well. To keep bandwidth use to a minimum, there should only be one upstream interface on the router receiving multicast packets. The interface on the router leading toward the receivers is the *downstream* interface, although the less precise terms *outgoing* or *outbound* interface are used as well. There can be 0 to  $N - 1$  downstream interfaces on a router, where  $N$  is the number logical interfaces on the router. To prevent looping, the upstream interface should never receive copies of downstream multicast packets.

Figure 1: Multicast Terminology in an IP Network



Routing loops are disastrous in multicast networks because of the risk of repeatedly replicated packets. One of the complexities of modern multicast routing protocols is the need to avoid routing loops, packet by packet, much more rigorously than in unicast routing protocols.

Multicast terminology includes two more complex concepts:

Leaf and Branch on page 7

Protocols for Multicast Networks on page 8

## Leaf and Branch

Each subnetwork with hosts on the router that has at least one interested receiver is a *leaf* on the distribution tree. Routers can have multiple *leaves* on different interfaces and must send a copy of the IP multicast packet out on each interface with a leaf. When a new leaf subnetwork is added to the tree (that is, the interface to the host subnetwork previously received no copies of the multicast packets), a new *branch* is built and the leaf is joined to the tree and replicated packets are now sent out on the interface. The number of leaves on a particular interface does not affect the router. The action is the same for one leaf or a hundred.

When a branch contains no leaves because there are no interested hosts on the router interface leading to that IP subnetwork, the branch is *pruned* from the distribution tree and no multicast packets are sent out that interface. Packets are replicated and sent out multiple interfaces only where the distribution tree branches at a router, and no link ever carries a duplicate flow of packets.

Collections of hosts all receiving the same stream of IP packets, usually from the same multicast source, are called *groups*. In IP multicast networks, traffic is delivered to multicast groups based on an IP multicast address or *group address*. The groups determine the location of the leafs and the leafs determine the branches on the multicast network.

## Protocols for Multicast Networks

The actions of receivers suggests two basic strategies for protocols to handle joining and pruning branches among a collection of multicast routers:

Dense-mode multicast—The assumption could be made that almost all possible subnets have at least one receiver wanting to receive the multicast traffic from a source, so the network is *flooded* with traffic on all possible branches, then pruned back as branches do not express an interest in receiving the packets, explicitly (by message) or implicitly (time-out silence). This is the *dense mode* of multicast operation. LANs are appropriate networks for dense mode operation.

Sparse-mode multicast—Alternately, the assumption could be made that very few of the possible receivers want packets from this source, so the network only establishes and sends packets on branches that have at least one leaf indicating (by message) a desire for the traffic. This is the *sparse mode* of multicast operation. WANs are appropriate networks for sparse mode operation, and indeed a common multicast guideline is not to run dense mode on a WAN under any circumstances.

Some multicast routing protocols, especially older ones, only support dense mode operation, which makes them inappropriate for use on the public Internet. Others allow sparse mode as well. If *sparse-dense mode* is supported, the multicast routing protocol allows some multicast groups to be sparse and other groups to be dense.

There is also a difference between the multicast protocols used between host and router and between the multicast routers themselves. Hosts on a given subnetwork only need to inform their router whether or not they are interested in receiving packets from a certain multicast group. The source host needs to inform only their routers that it is the source of traffic for a particular multicast group. In other words, no detailed knowledge of the distribution tree is needed by any hosts, only a *group membership protocol to inform routers of their participation in a multicast group*. Between adjacent routers, on the other hand, the multicast routing protocols must avoid loops as they build a detailed sense of the network topology and distribution tree from source to leaf. So, different multicast protocols are used for the host-router portion and the router-router portion of the multicast network.

## IP Multicast Building Blocks

---

Implementing an IP multicast network is made simpler by using a number of standardized building blocks. These IP multicast building blocks include a special IP multicast address space, administrative scoping to prevent large-scale routing loops, upstream and downstream interface lists, reverse path forwarding (RPF) to prevent small-scale routing loops, a shortest-path tree (SPT) algorithm to build a minimal distribution tree, and a rendezvous point (RP) and associated rendezvous-point tree (RPT) to allow sparse mode receivers to find sources.

There are six major IP multicast building blocks:

IP Multicast Addressing on page 9

Administrative Scoping on page 10

Interface Lists on page 10

Reverse Path Forwarding (RPF) on page 11

Shortest-Path Tree (SPT) on page 12

Rendezvous Point (RP), Shared Trees, and the Rendezvous-Point Tree (RPT) on page 13

### **IP Multicast Addressing**

Multicast uses the Class D IP address range (224.0.0.0 through 239.255.255.255). Class D addresses are commonly referred to as *multicast addresses* because the entire classful address concept is obsolete. Multicast addresses can never appear as the source address in an IP packet and can only be the destination of a packet.

Multicast addresses usually have a prefix length of /32, although other prefix lengths are allowed. Multicast addresses represent logical groupings of receivers and not physical collections of devices. Blocks of multicast addresses can still be described in terms of prefix length in traditional notation, but only for convenience. For example, the multicast address range from 232.0.0.0 through 232.255.255.255 can be written as 232.0.0.0/8 or 232/8.

ISPs do not typically allocate multicast addresses to their customers because multicast addresses are concerned more with content than physical devices. Receivers are not assigned their own multicast addresses, but need to know only the multicast address of the content. Sources need to be assigned multicast addresses only in order to produce the content, not to identify their place in the network. Every source and receiver still needs an ordinary, unicast IP address.

Multicast addressing most often references the receivers, and the source of multicast content is usually not even a member of the multicast group for which it produces content. If the source needs to monitor the packets it produces, this can be done locally, and there is no need to make the packets traverse the network.

Many applications have been assigned a range of multicast addresses for their own use. These applications assign multicast addresses to sessions created by that application. You do not usually need to statically assign a multicast address, but you can do so.

## Administrative Scoping

Routing loops must be avoided in IP multicast networks. Because multicast routers must replicate packets for each downstream branch, not only will looping packets not arrive at a destination, but each pass around the loop will multiply the number of looping packets, eventually overwhelming the network.

*Scoping* limits the routers and interfaces that can be used to forward a multicast packet. Scoping can use the time-to-live (TTL) field in the IP packet header, but TTL scoping depends on intimate knowledge of the network topology by the network administrator. This topology can change as links fail and are restored, making TTL scoping a poor solution for multicast.

Multicast scoping is *administrative* in the sense that a range of multicast addresses is reserved for scoping purposes, as described in RFC 2365. Routers at the boundary must be able to filter multicast packets and make sure the packets do not stray beyond the established limit.

Administrative scoping is much better than TTL scoping, but in many cases the dropping of administratively scoped packets is still determined by the network administrator. For example, the multicast address range 239/8 is defined in RFC 2356 as administratively scoped and packets using this range should not be forwarded beyond a network “boundary,” usually a routing domain. But only the network administrator knows where the border routers are and can implement the scoping correctly.

Multicast groups used by unicast routing protocols, such as 224.0.0.5 for all OSPF routers, are administratively scoped for that LAN only. This allows the same multicast address to be used without conflict on every LAN running OSPF.

## Interface Lists

To avoid multicast routing loops, every multicast router must always be aware of the interface that leads to source of that multicast group content by the shortest path. This is the upstream (incoming) interface and packets should never be forwarded back towards a multicast source. All other interfaces are potential downstream (outgoing) interfaces, depending on the number of branches on the distribution tree.

Routers closely monitor the status of the incoming and outgoing interfaces, a process that determines the *multicast forwarding state*. A router with multicast forwarding state for a particular multicast group is essentially “turned on” for that group content. The incoming interface for a group is sometimes called the *IIF*. The outgoing interface list for a group is sometimes called the *OIL*. Interfaces on the router’s OIL send copies of the group’s packets received on the IIF for that group. The IIF and OIL might be different for different multicast groups, of course.

The multicast forwarding state in a router is usually written in either (S,G) or (\*,G) notation. These are pronounced “ess comma gee” and “star comma gee” respectively. In (S,G), the S refers to the unicast IP address of the source for the multicast traffic and the G refers to the particular multicast group IP address for which S is the source. All multicast packets sent from this source have S as the source address and G as the destination address.

The asterisk (\*) in the (\*,G) notation is a wild card that indicates the state applies to any multicast application source sending to group G. So, if two sources are originating exactly the same content for multicast group 224.1.1.2, a router could use (\*, 224.1.1.2) to represent the state of a router forwarding traffic from both sources to the group.

For more information on the use of multicast forwarding state notations in different types of distribution trees, see “Rendezvous Point (RP), Shared Trees, and the Rendezvous-Point Tree (RPT)” on page 13. For more information on the use of multicast notations in different multicast routing protocols, see “Protocols for Multicast” on page 15.

### **Reverse Path Forwarding (RPF)**

Unicast forwarding decisions are typically based on the destination address of the packet arriving at a router. The unicast routing table is organized by destination subnet and mainly set up to forward the packet toward the destination.

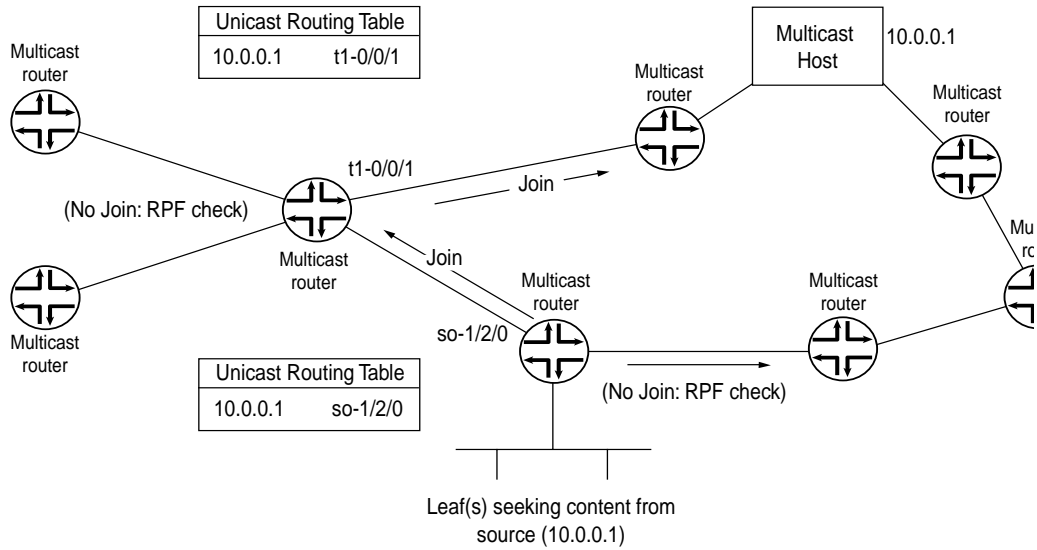
In multicast, the router forwards the packet *away* from the source to make progress along the distribution tree and prevent routing loops. The router’s multicast forwarding state runs more logically by organizing tables based on the reverse path, from the receiver back to the root of the distribution tree. This process is known as *reverse path forwarding (RPF)*.

The router adds a branch to a distribution tree depending on whether the request for traffic from a multicast group passes the *reverse-path-forwarding check (RPF check)*. Every multicast packet received must pass an RPF check before it is eligible to be replicated or forwarded on any interface.

The RPF check is essential for every router’s multicast implementation. When a multicast packet is received on an interface, the router interprets the *source* address in the multicast IP packet as the *destination* address for a unicast IP packet. The source multicast address is found in the unicast routing table and the outgoing interface determined. If the outgoing interface found in the unicast routing table is the same as the interface that the multicast packet was received on, the packet passes the RPF check. Multicast packets that fail the RPF check are dropped because the incoming interface is not on the *shortest path* back to the source.

Figure 2 shows how multicast routers can use the unicast routing table to perform an RPF check, and how the results obtained at each router determine where join messages are sent.

Figure 2: Multicast Routers and the RPF Check



Routers can build and maintain separate tables for RPF purposes. The router must have some way to determine its *RPF interface* for the group, which is the interface topologically closest to the root. The distribution tree should follow the *shortest-path tree* (SPT) topology for efficiency. The RPF check helps to construct this tree.

### Shortest-Path Tree (SPT)

The distribution tree used for multicast is rooted at the source and is the shortest-path tree (SPT) as well. Consider a set of multicast routers without any active multicast traffic for a certain group (that is, they have no multicast forwarding state for that group). When a router learns that an interested receiver for that group is on one of its directly connected subnets, the router will attempt to join the tree for that group.

To join the distribution tree, the router determines the unicast IP address of the source for that group. This address can be a simple static configuration in the router, or as complex as a set of protocols.

To build the SPT for that group, the router executes an RPF check on the source address in its routing table. The RPF check produces the interface closest to the source, which is where multicast packets from this source for this group should flow into the router.

The router next sends a *join message* out on this interface using the proper multicast protocol to inform the upstream router that it wishes to join the distribution tree for that group. This message is an (S,G) join message because both S and G are known. The router receiving the (S,G) join message adds the interface on which the message was received to its OIL for the group and also performs an RPF check on the source address. The upstream router then sends an (S,G) join message out the RPF interface toward the source informing the upstream router that it also wants to join the group.

Each upstream router repeats this process, propagating joins out the RPF interface, building the SPT as it goes. The process stops when the join message:

- Reaches the router directly connected to the host that is the source, or
- Reaches a router that already has multicast forwarding state for this source-group pair.

In either case, the branch is created and each of the routers has multicast forwarding state for the source-group pair, and packets can flow down the distribution tree from source to receiver. The RPF check at each router makes sure that the tree is an SPT.

SPTs are always the shortest path, but they are not necessarily short. That is, sources and receivers tend to be on the periphery of a router network, not on the backbone, and multicast distribution trees have a tendency to sprawl across almost every router in the network. Because multicast traffic can overwhelm a slow interface, and one packet can easily become a hundred or a thousand on the opposite side of the backbone, it makes sense to provide a *shared tree* as a distribution tree so that the multicast source could be located more centrally in the network, on the backbone. This sharing of distribution trees with roots in the core network is accomplished by a multicast *rendezvous point (RP)*. For more information about RPs, see “Rendezvous Point (RP), Shared Trees, and the Rendezvous-Point Tree (RPT)” on page 13.

### ***Rendezvous Point (RP), Shared Trees, and the Rendezvous-Point Tree (RPT)***

In a shared tree, the root of the distribution tree is a router, not a host, and located somewhere in the core of the network. In the primary sparse mode multicast routing protocol, *Protocol Independent Multicast sparse mode (PIM SM)*, the core router at the root of the shared tree is the *rendezvous point (RP)*. Packets from the upstream source and join messages from the downstream routers “rendezvous” at this core router.

In the RP model, other routers do not need to know the addresses of the sources for every multicast group. All they need to know is the IP address of the RP router. The RP router knows the sources for all multicast groups.

The RP model shifts the burden of finding sources of multicast content from each router (the (S,G) notation) to the network (the (\*,G) notation knows only the RP). Exactly how the RP finds the unicast IP address of the source varies, but there must be some method to determine the proper source for multicast content for a particular group.

Consider a set of multicast routers without any active multicast traffic for a certain group. When a router learns that an interested receiver for that group is on one of its directly connected subnets, the router will attempt to join the distribution tree for that group back to RP, not to the actual source of the content.

To join the shared tree, or *rendezvous-point tree (RPT)* as it called in PIM sparse mode, the router must do the following:

Determine the IP address of the RP for that group. This can be as simple as static configuration in the router, or as complex as a set of nested protocols.

Build the shared tree for that group. The router executes an RPF check on the RP address in its routing table, which produces the interface closest to the RP. The router now knows that multicast packets from this RP for this group should flow into the router on this RPF interface.

Send a join message out on this interface using the proper multicast protocol (probably PIM sparse mode) to inform the upstream router that it wishes to join the shared tree for that group. This message is an (\*,G) join message because S is not known, only the RP, and the RP is not actually the source of the multicast packets. The router receiving the (\*,G) join message adds the interface on which the message was received to its OIL for the group and also performs an RPF check on the RP address. The upstream router then sends an (\*,G) join message out the RPF interface toward the source informing the upstream router that it also wants to join the group.

Each upstream router repeats this process, propagating joins out the RPF interface, building the shared tree as it goes. The process stops when the join message:

Reaches the RP for that group, or

Reaches a router that already has multicast forwarding state for this group along the RPT.

In either case, the branch is created and packets can flow from the source to the RP, and from the RP to the receiver. Note that there is no guarantee that the shared tree (RPT) is the shortest path tree to the source, and most likely it is not. However, there are ways to “migrate” a shared tree to an SPT once the flow of packets begins. In other words, the forwarding state can transition from (\*,G) to (S,G). The formation of both types of tree depends heavily on the operation of the RPF check and the *RPF table*. For more information on the RPF table, see “The RPF Table” on page 14.

## The RPF Table

---

The RPF table plays the key role in the multicast router. The RPF table is consulted for every RPF check, which is performed at intervals on multicast packets entering the multicast router. Distribution trees of all types rely on the RPF table to form properly, and multicast forwarding state also depends on the RPF table.

RPF checks are performed only on unicast addresses to find the upstream interface for the multicast source or RP.

## RPF Checks

The routing table used for RPF checks can be the same routing table used to forward unicast IP packets, or it can be a separate routing table used only for multicast RPF checks. In either case, the RPF table contains only unicast routes, because the RPF check is performed on the source address of the multicast packet, not the multicast group destination address, and a multicast address is forbidden from appearing in the source address field of an IP packet header. The unicast address can be used for RPF checks because there is only one source host for a particular stream of IP multicast content for a multicast group address, although the same content could be available from multiple sources.

## Populating the RPF Table

If the same routing table used to forward unicast packets is also used for the RPF checks, the routing table is populated and maintained by the traditional unicast routing protocols such as RIP, OSPF, IS-IS, and BGP. If a dedicated multicast RPF table is used, this table must be populated by some other method. Some multicast routing protocols (such as the Distance Vector Multicast Routing Protocol [DVMRP]) essentially duplicate the operation of a unicast routing protocol and populate a dedicated RPF table. Others, such as PIM, do not duplicate routing protocol functions and must rely on some other routing protocol to set up this table, which is why PIM is *protocol independent*.

Some traditional routing protocols such as BGP and IS-IS now have extensions to differentiate between different sets of routing information sent between routers for unicast and multicast. For example, there is multiprotocol Border Gateway Protocol (MBGP) and multitopology routing in IS-IS (M-ISIS). Multicast Open Shortest Path First (MOSPF) also extends OSPF for multicast use, but goes further than MBGP or M-ISIS and makes MOSPF into a complete multicast routing protocol on its own. When these routing protocols are used, routes can be tagged as multicast RPF routers and used by the receiving router differently than the unicast routing information.

Using the main unicast routing table for RPF checks provides simplicity. A dedicated routing table for RPF checks allows a network administrator to set up separate paths and routing policies for unicast and multicast traffic, allowing the multicast network to function more independently of the unicast network.

## Protocols for Multicast

---

The protocols used among a collection of multicast-capable IP routers fall into two major categories:

- Multicast group membership protocols that are used between host and router (and host to host).

- Multicast routing protocols that are used between routers.

The following sections describe:

- Multicast Group Membership Protocols on page 16

- Multicast Routing Protocols on page 17

## Multicast Group Membership Protocols

Multicast group membership protocols allow a router to know when a host on a directly attached subnet, typically a LAN, wants to receive traffic from a certain multicast group. Even if more than one host on the LAN wants to receive traffic for that multicast group, the router has to send only one copy of each packet for that multicast group out on that interface because of the inherent broadcast nature of LANs. Only when the router is informed by the multicast group membership protocol that there are no interested hosts on the subnet can the packets be withheld and that leaf pruned from the distribution tree.

There is one standard IP multicast group membership protocol: the Internet Group Management Protocol (IGMP). However, IGMP has several versions that are supported by hosts and routers. There are currently three versions of IGMP:

IGMPv1—The original protocol defined in RFC 1112. An explicit join message is sent to the router, but a time-out is used to determine when hosts leave a group. This wastes processing cycles on the router, especially on older or smaller routers.

IGMPv2—Defined in RFC 2236. Among other features, IGMPv2 adds an explicit leave message to the join message so that routers can more easily determine when a group has no interested listeners on a LAN.

IGMPv3—Defined in RFC 3376. Among other features, IGMPv3 optimizes support for a single source of content for a multicast group, or *source-specific multicast (SSM)*. (RFC 1112 supported both many-to-many and one-to-many multicast, but one-to-many is considered the more viable model for the Internet at large.)

Although the various versions of IGMP are backward compatible, it is common for a router to run multiple versions of IGMP on LAN interfaces because backward compatibility is achieved by dropping back to the most basic of all versions run on a LAN. For example, if one host is running IGMPv1, any router attached to the LAN running IGMPv2 will drop back to IGMPv1 operation, effectively eliminating the IGMPv2 advantages. Running multiple IGMP versions ensures both IGMPv1 and IGMPv2 hosts find peers for their versions on the router.

## Multicast Routing Protocols

Multicast routing protocols enable a collection of multicast routers to build (join) distribution trees when a host on a directly attached subnet, typically a LAN, wants to receive traffic from a certain multicast group.

There are five multicast routing protocols:

**DVMRP**—The first of the multicast routing protocols and hampered by a number of limitations that make this method unattractive for large-scale Internet use. DVMRP is a dense-mode-only protocol, and uses the flood-and-prune or implicit join method to deliver traffic everywhere and then determine where the uninterested receivers are. DVMRP uses source-based distribution trees in the form (S,G).

**MOSPF**—Extends OSPF for multicast use, but only for dense mode. However, MOSPF has an explicit join message, so routers do not have to flood their entire domain with multicast traffic from every source. MOSPF uses source-based distribution trees in the form (S,G).

**PIM dense mode**—This is PIM operating in dense mode (PIM DM), but the differences from PIM sparse mode are profound enough to consider the two modes separately. PIM also supports sparse-dense mode, with mixed sparse and dense groups, but there is no special notation for that operational mode. In contrast to DVMRP and MOSPF, PIM dense mode allows a router to use any unicast routing protocol and performs RPF checks using the unicast routing table. PIM dense mode has an implicit join message, so routers use the flood-and-prune method to deliver traffic everywhere and then determine where the uninterested receivers are. PIM dense mode uses source-based distribution trees in the form (S,G), as do all dense-mode protocols.

**PIM sparse mode**—Allows a router to use any unicast routing protocol and performs RPF checks using the unicast routing table. However, PIM sparse mode has an *explicit* join message, so routers determine where the interested receivers are and send join messages upstream to their neighbors, building trees from receivers to RP. PIM sparse mode uses an RP router as the initial source of multicast group traffic and therefore builds distribution trees in the form (\*,G), as do all sparse-mode protocols. However, PIM sparse mode migrates to an (S,G) source-based tree if that path is shorter than through the RP for a particular multicast group's traffic.

**CBT**—Shares all of the characteristics of PIM sparse mode (sparse mode, explicit join, and shared (\*,G) trees), but is said to be more efficient at finding sources than PIM sparse mode. CBT is rarely encountered outside of academic discussions. There are no large-scale deployments of CBT, commercial or otherwise.

The differences among the five multicast routing protocols are summarized in Table 2.

**Table 2: Multicast Routing Protocols Compared**

Multicast Routing Protocol	Dense Mode?	Sparse Mode?	Implicit Join?	Explicit Join?	(S,G) SBT?	(* ,G) Shared Tree?
DVMRP	Yes	No	Yes	No	Yes	No
MOSPF	Yes	No	No	Yes	Yes	No
PIM dense mode	Yes	No	Yes	No	Yes	No
PIM sparse mode	No	Yes	No	Yes	Yes, maybe	Yes, initially
CBT	No	Yes	No	Yes	No	Yes

It is important to realize that retransmissions due to a high bit-error-rate on a link or overloaded router can make multicast as inefficient as repeated unicast. Therefore, there is a trade-off in many multicast applications regarding the session support when using TCP (but TCP always resends missing segments) or the simple drop-and-continue strategy of the UDP datagram service (but reordering can become an issue). Modern multicast uses UDP almost exclusively.