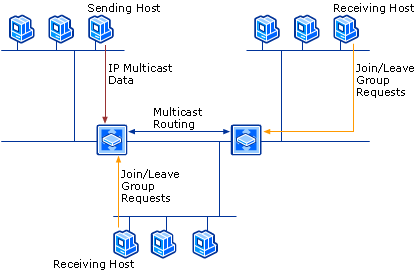
IP multicasting overview

Unicasting is the sending of network traffic to an endpoint. Multicasting is the sending of network traffic to a group of endpoints. Only those members in the group of endpoints that are listening for the multicast traffic (the multicast group) process the multicast traffic. All other nodes ignore the multicast traffic.

The concept of group membership is central to IP multicasting. IP multicast datagrams are sent to a group, and only members of the group receive the datagrams. A group is identified by a single IP multicast address, which is an IP address in the Class D range of 224.0.0.0 to 239.255.255.255 (designated as 224.0.0.0/4 in classless interdomain routing (CIDR) notation). These Class D addresses are known as group addresses. A source host sends multicast datagrams to a group address. Destination hosts inform a local router that they need to join the group.

In an IP multicast-enabled intranet, any host can send IP multicast datagrams to any group address, and any host can receive IP multicast datagrams from any group address regardless of its location. To facilitate this capability, the hosts and routers on the intranet must support IP multicasting. Hosts use the Internet Group Management Protocol (IGMP) for establishing group membership. Routers use multicast routing protocols for forwarding multicast data.

The following figure illustrates a multicast-enabled intranet.



In this illustration, the hosts and routers are multicast-enabled so that the following can occur:

* The sending host sends multicast datagrams to a designated group address.
* The routers forward the multicast datagrams to any network segments that include group members. Routers can forward multicast traffic across a network, between networks, and across the Internet.
* The receiving hosts inform a local router to join the group, and then they receive all subsequent datagrams sent to the group address.
* If a receiving host leaves the group and detects that it might be the last group member on the subnet, it can contact the local router to leave the group, informing the router to stop forwarding the multicast datagrams to that subnet.

Benefits of IP multicasting

Multicasting provides an efficient way to support high-bandwidth, one-to-many applications on a network:

* Multicasting can dramatically reduce network traffic by sending a single copy of the data.
* Hosts can be configured for multicasting without hardware upgrades.
* Because newer routers already support multicast forwarding and multicast routing protocols, enabling multicasting on a network is practical and cost-effective.

Multicasting is useful for many types of one-to-many applications, such as the following:

* Multimedia, such as video conferencing and collaborative computing.
* Automatic discovery of resources in a network (in Windows Server® 2008 for example, TCP/IP router discovery uses multicasting by default, and WINS uses multicasting during automatic discovery of replication partners).
* Datacasting, such as file distribution or database synchronization.
* Mobile computer support, such as remote address book updating.
* Distribution of organizational publications.

IP multicasting with Routing and Remote Access

Windows Server 2008 does not provide multicast routing protocols, such as Distance Vector Multicast Routing Protocol (DVMRP), Multicast Extensions to Open Shortest Path First (MOSPF), and Protocol Independent Multicast (PIM), although Routing and Remote Access does support multicast routing protocols developed by independent software vendors (ISVs).

As an alternative, you can use the Routing and Remote Access service to forward multicast traffic. In this case, the Routing and Remote Access service uses IGMP as an IP routing protocol component. Router interfaces are configured in one of two operating modes: IGMP router mode or IGMP proxy mode. The purpose of IGMP router mode is to forward multicast traffic in a single-router intranet. The purpose of IGMP proxy mode is to connect a single-router intranet to a multicast-capable intranet or the Internet.

Although Routing and Remote Access uses IGMP in a limited way to enable multicast forwarding on an intranet, it is not the equivalent of a true multicast routing protocol. The Routing and Remote Access IGMP routing protocol component supports multicast forwarding for several network topologies.

Multicast Listener Discovery (MLD)

Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) are the Multicast Group Membership Discovery (MGMD) protocols. They are essentially the same protocol, with IGMP used for IPv4 multicast groups and MLD used for IPv6 multicast groups. These protocols are used between end systems (often desktops) and the multicast router to request data for a given multicast group.There have been three versions of IGMP, and two versions of MLD. IGMPv2 is equivalent in function to MLDv1 and IGMPv3 is equivalent to MLDv2. All versions of IGMP/MLD are widely deployed.

This protocol is used by an IPv6 router to discover the presence of multicast listeners (that is, nodes wishing to receive multicast packets) on its directly attached links, and to discover specifically which multicast addresses are of interest to those neighboring nodes. This protocol is referred to as Multicast Listener Discovery or MLD. MLD is derived from version 2 of IPv4's

Internet Group Management Protocol, IGMPv2. One important difference to note is that MLD uses ICMPv6 (IP Protocol 58) message types, rather than IGMP (IP Protocol 2) message types.

Introduction

The purpose of Multicast Listener Discovery (MLD) is to enable each

IPv6 router to discover the presence of multicast listeners (that is,

nodes wishing to receive multicast packets) on its directly attached

links, and to discover specifically which multicast addresses are of

interest to those neighboring nodes. This information is then

provided to whichever multicast routing protocol is being used by the

router, in order to ensure that multicast packets are delivered to

all links where there are interested receivers.

MLD is an asymmetric protocol, specifying different behaviors for

multicast listeners and for routers. For those multicast addresses

to which a router itself is listening, the router performs both parts

of the protocol, including responding to its own messages.

If a router has more than one interface to the same link, it need

perform the router part of MLD over only one of those interfaces.

Listeners, on the other hand, must perform the listener part of MLD

on all interfaces from which an application or upper-layer protocol

has requested reception of multicast packets.

**Message Format**

MLD is a sub-protocol of ICMPv6, that is, MLD message types are a

subset of the set of ICMPv6 messages, and MLD messages are identified

in IPv6 packets by a preceding Next Header value of 58. All MLD

messages described in this document are sent with a link-local IPv6

Source Address, an IPv6 Hop Limit of 1, and an IPv6 Router Alert

option [[RTR-ALERT](https://tools.ietf.org/html/rfc2710#ref-RTR-ALERT)] in a Hop-by-Hop Options header. (The Router Alert

option is necessary to cause routers to examine MLD messages sent to

multicast addresses in which the routers themselves have no

interest.)

MLD messages have the following format:

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

| Type | Code | Checksum |

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

| Maximum Response Delay | Reserved |

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

| |

+ +

| |

+ Multicast Address +

| |

+ +

| |

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

### Type

There are three types of MLD messages:

Multicast Listener Query (Type = decimal 130)

There are two subtypes of Multicast Listener Query messages:

- General Query, used to learn which multicast addresses have

listeners on an attached link.

- Multicast-Address-Specific Query, used to learn if a

particular multicast address has any listeners on an attached

link.

These two subtypes are differentiated by the contents of the

Multicast Address field, as described in [section 3.6](https://tools.ietf.org/html/rfc2710#section-3.6).

Multicast Listener Report (Type = decimal 131)

Multicast Listener Done (Type = decimal 132)

### Checksum

The standard ICMPv6 checksum, covering the entire MLD message plus a

"pseudo-header" of IPv6 header fields [[ICMPv6](https://tools.ietf.org/html/rfc2710#ref-ICMPv6),[IPv6](https://tools.ietf.org/html/rfc2710#ref-IPv6)].

### [3.4](https://tools.ietf.org/html/rfc2710" \l "section-3.4). Maximum Response Delay

The Maximum Response Delay field is meaningful only in Query

messages, and specifies the maximum allowed delay before sending a

responding Report, in units of milliseconds. In all other messages,

it is set to zero by the sender and ignored by receivers.

Varying this value allows the routers to tune the "leave latency"

(the time between the moment the last node on a link ceases listening

to a particular multicast address and moment the routing protocol is

notified that there are no longer any listeners for that address), as

discussed in [section 7.8](https://tools.ietf.org/html/rfc2710#section-7.8). It also allows tuning of the burstiness of

MLD traffic on a link, as discussed in [section 7.3](https://tools.ietf.org/html/rfc2710#section-7.3).

### Reserved

Initialized to zero by the sender; ignored by receivers.

### [3.6](https://tools.ietf.org/html/rfc2710" \l "section-3.6). Multicast Address

In a Query message, the Multicast Address field is set to zero when

sending a General Query, and set to a specific IPv6 multicast address

when sending a Multicast-Address-Specific Query.

In a Report or Done message, the Multicast Address field holds a

specific IPv6 multicast address to which the message sender is

listening or is ceasing to listen, respectively.

### [3.7](https://tools.ietf.org/html/rfc2710" \l "section-3.7). Other fields

The length of a received MLD message is computed by taking the IPv6

Payload Length value and subtracting the length of any IPv6 extension

headers present between the IPv6 header and the MLD message. If that

length is greater than 24 octets, that indicates that there are other

fields present beyond the fields described above, perhaps belonging

to a future backwards-compatible version of MLD. An implementation

of the version of MLD specified in this document MUST NOT send an MLD

message longer than 24 octets and MUST ignore anything past the first

24 octets of a received MLD message. In all cases, the MLD checksum

MUST be computed over the entire MLD message, not just the first 24

octets.

## [4](https://tools.ietf.org/html/rfc2710" \l "section-4). Protocol Description

Note that defaults for timer values are described later in this

document. Timer and counter names appear in square brackets.

Routers use MLD to learn which multicast addresses have listeners on

each of their attached links. Each router keeps a list, for each

attached link, of which multicast addresses have listeners on that

link, and a timer associated with each of those addresses. Note that

the router needs to learn only that listeners for a given multicast

address are present on a link; it does NOT need to learn the identity

(e.g., unicast address) of those listeners or even how many listeners

are present.

For each attached link, a router selects one of its link-local

unicast addresses on that link to be used as the IPv6 Source Address

in all MLD packets it transmits on that link.

For each interface over which the router is operating the MLD

protocol, the router must configure that interface to listen to all

link-layer multicast address that can be generated by IPv6

multicasts. For example, an Ethernet-attached router must set its

Ethernet address reception filter to accept all Ethernet multicast

addresses that start with the hexadecimal value 3333 [[IPv6-ETHER](https://tools.ietf.org/html/rfc2710#ref-IPv6-ETHER)]; in

the case of an Ethernet interface that does not support the filtering

of such a range of multicast address, it must be configured to accept

ALL Ethernet multicast addresses, in order to meet the requirements

of MLD.

With respect to each of its attached links, a router may assume one

of two roles: Querier or Non-Querier. There is normally only one

Querier per link. All routers start up as a Querier on each of their

attached links. If a router hears a Query message whose IPv6 Source

Address is numerically less than its own selected address for that

link, it MUST become a Non-Querier on that link. If [Other Querier

Present Interval] passes without receiving, from a particular

attached link, any Queries from a router with an address less than

its own, a router resumes the role of Querier on that link.

A Querier for a link periodically [Query Interval] sends a General

Query on that link, to solicit reports of all multicast addresses of

interest on that link. On startup, a router SHOULD send [Startup

Query Count] General Queries spaced closely together [Startup Query

Interval] on all attached links in order to quickly and reliably

discover the presence of multicast listeners on those links.

General Queries are sent to the link-scope all-nodes multicast

address (FF02::1), with a Multicast Address field of 0, and a Maximum

Response Delay of [Query Response Interval].

When a node receives a General Query, it sets a delay timer for each

multicast address to which it is listening on the interface from

which it received the Query, EXCLUDING the link-scope all-nodes

address and any multicast addresses of scope 0 (reserved) or 1

(node-local). Each timer is set to a different random value, using

the highest clock granularity available on the node, selected from

the range [0, Maximum Response Delay] with Maximum Response Delay as

specified in the Query packet. If a timer for any address is already

running, it is reset to the new random value only if the requested

Maximum Response Delay is less than the remaining value of the

running timer. If the Query packet specifies a Maximum Response

Delay of zero, each timer is effectively set to zero, and the action

specified below for timer expiration is performed immediately.

When a node receives a Multicast-Address-Specific Query, if it is

listening to the queried Multicast Address on the interface from

which the Query was received, it sets a delay timer for that address

to a random value selected from the range [0, Maximum Response

Delay], as above. If a timer for the address is already running, it

is reset to the new random value only if the requested Maximum

Response Delay is less than the remaining value of the running timer.

If the Query packet specifies a Maximum Response Delay of zero, the

timer is effectively set to zero, and the action specified below for

timer expiration is performed immediately.

If a node's timer for a particular multicast address on a particular

interface expires, the node transmits a Report to that address via

that interface; the address being reported is carried in both the

IPv6 Destination Address field and the MLD Multicast Address field of

the Report packet. The IPv6 Hop Limit of 1 (as well as the presence

of a link-local IPv6 Source Address) prevent the packet from

traveling beyond the link to which the reporting interface is

attached.

If a node receives another node's Report from an interface for a

multicast address while it has a timer running for that same address

on that interface, it stops its timer and does not send a Report for

that address, thus suppressing duplicate reports on the link.

When a router receives a Report from a link, if the reported address

is not already present in the router's list of multicast address

having listeners on that link, the reported address is added to the

list, its timer is set to [Multicast Listener Interval], and its

appearance is made known to the router's multicast routing component.

If a Report is received for a multicast address that is already

present in the router's list, the timer for that address is reset to

[Multicast Listener Interval]. If an address's timer expires, it is

assumed that there are no longer any listeners for that address

present on the link, so it is deleted from the list and its

disappearance is made known to the multicast routing component.

When a node starts listening to a multicast address on an interface,

it should immediately transmit an unsolicited Report for that address

on that interface, in case it is the first listener on the link. To

cover the possibility of the initial Report being lost or damaged, it

is recommended that it be repeated once or twice after short delays

[Unsolicited Report Interval]. (A simple way to accomplish this is

to send the initial Report and then act as if a Multicast-Address-

Specific Query was received for that address, and set a timer

appropriately).

When a node ceases to listen to a multicast address on an interface,

it SHOULD send a single Done message to the link-scope all-routers

multicast address (FF02::2), carrying in its Multicast Address field

the address to which it is ceasing to listen. If the node's most

recent Report message was suppressed by hearing another Report

message, it MAY send nothing, as it is highly likely that there is

another listener for that address still present on the same link. If

this optimization is implemented, it MUST be able to be turned off

but SHOULD default to on.

When a router in Querier state receives a Done message from a link,

if the Multicast Address identified in the message is present in the

Querier's list of addresses having listeners on that link, the

Querier sends [Last Listener Query Count] Multicast-Address-Specific

Queries, one every [Last Listener Query Interval] to that multicast

address. These Multicast-Address-Specific Queries have their Maximum

Response Delay set to [Last Listener Query Interval]. If no Reports

for the address are received from the link after the response delay

of the last query has passed, the routers on the link assume that the

address no longer has any listeners there; the address is therefore

deleted from the list and its disappearance is made known to the

multicast routing component. This process is continued to its

resolution (i.e. until a Report is received or the last Multicast-

Address-Specific Query is sent with no response) despite any

transition from Querier to Non-Querier on this link.

Routers in Non-Querier state MUST ignore Done messages.

When a router in Non-Querier state receives a Multicast-Address-

Specific Query, if its timer value for the identified multicast

address is greater than [Last Listener Query Count] times the Maximum

Response Delay specified in the message, it sets the address's timer

to that latter value.

[https://tools.ietf.org/html/rfc2710]

### Distance Vector Multicast Routing (DVMRP)

Currently used across the Internet Multicast Backbone or MBONE, DVMRP use reverse path flooding i.e. when it receives a packet it floods the packet out of all interfaces except the one that leads back to the source. Prune messages are sent up the distribution tree to prevent subsequent packets travelling to where no members exist. Periodic flooding occurs as DVMRP tries to establish if there are any other potential new group members.

To determine which interfaces lead back to the source DVMRP uses a unicast hop-based routing protocol which is similar to RIP.

[**RFC 1075**](http://www.ietf.org/rfc/rfc1075.txt) describes DVMRP.

### Multicast Open Shortest Path First (MOSPF)

MOSPF depends on OSPF running and includes multicast information in OSPF Link State Advertisements. MOSPF builds a distribution tree for each Source/Group pair and a tree for the active sources. These trees are held in a cache and are updated everytime that there is a link change. An unstable environment with many source/group pairs would be most unsuitable for MOSPF so it does not scale well.

[**RFC 1584**](http://www.ietf.org/rfc/rfc1584.txt) describes MOSPF.

### Protocol Independent Multicast Dense Mode (PIM DM)

This protocol operates in the same way that DVMRP does except that it is not dependent on a unicast routing protocol.

PIMv1 uses IP protocol **2** and queries are multicast to all routers using **224.0.0.2** every 30 seconds by default (PIMv2 sends queries to all PIM routers on**224.0.0.13** and uses IP protocol **103**). On a multiaccess network like Ethernet a Designated Router is elected (the highest IP address wins) and this router is the one that sends IGMP host query messages on a LAN.

PIM first floods the multicast packets to all routers and then prunes those that do not have members of particular groups or if the path to members is redundant and not the shortest route. Ideally senders and receivers should be close to one another and the ratio of receivers to senders should be large. Dense mode is fine if the volume of multicast traffic is large and constant.

### Sparse Mode Routing Protocols

If bandwidth is low and the group members are sparsely distributed then a Sparse Mode protocol is more appropriate.

Flooding will cause problems so more selective techniques are used to create the trees. The trees start off empty of any branches until there are explicit requests to join the distribution tree i.e. no packets are sent unless specifically asked for them.

Two sparse mode routing protocols are **Core-Based Trees (CBT)** and **Protocol Independent Multicast Sparse Mode (PIM SM)**.

### Core-Based Tree (CBT)

A single tree is constructed that is shared by all members of a group and all multicast traffic is sent over this tree regardless of the source i.e. the sources share the tree. This means that individual routers do not have to maintain such a high level of multicast information.

A Core router takes responsibility for constructing the tree and the other routers join the tree by sending join messages to this core router. The core sends a join acknowledgement over the reverse path thus forming a branch. If a router already on a branch receives a join message then it acknowledges the message instead of the core router, thereby minimising traffic.

[**RFC 2201**](http://www.ietf.org/rfc/rfc2201.txt) describes CBT.

### Protocol Independent Multicast Sparse Mode (PIM SM)

If multicast traffic is intermittent and there are few receivers then the idea of **Rendezvous Points (RP)** is introduced. If a source wants to send data it first sends it to the Rendezvous Point (another router designated as such). When a receiver wants to receive this data it registers with the Rendezvous Point, this is the **Explicit Join Model**. At this point data flows and the sender and receiver work out the most direct route between themselves automatically. The first-hop router receives PIM register messages by hosts wanting to send data to a group. These register messages are sent to the RP. The last-hop routers send PIM join or prune messages to the RP.

You can think of the Rendezvous Point as the root of a shared distribution tree. In fact, multiple shared trees are built around the Rendezvous Point.

With Source trees, Reverse Path Forwarding (RPF) only accepts multicast traffic if it is received on an interface that has the route to the source out of that interface. With Shared trees, multicast traffic is only accepted when it is received on an interface that has the route to the RP.

PIM can support Dense Mode for some multicast groups and Sparse Mode for others. If no Rendezvous Point is found then Dense Mode could be used instead. An RP can act for many groups and one group can have multiple RPs.