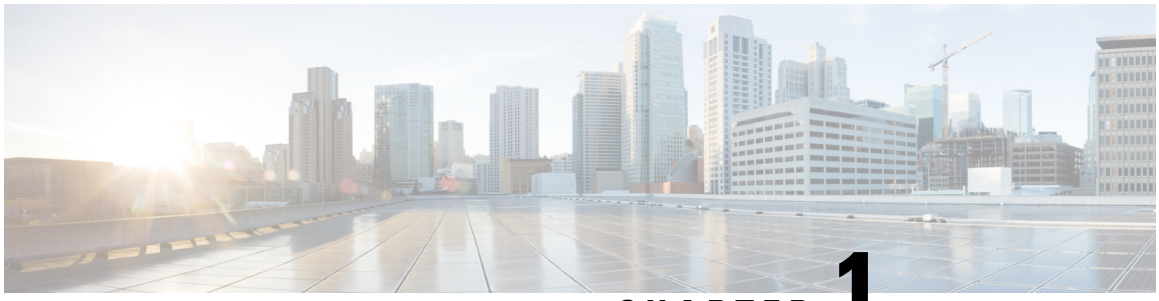
Multicast Configuration Guide for Cisco NCS 560 Series Routers, Cisco IOS XR Release 7.1.x

First Published: 2020-01-29

Last Modified: 2020-05-05

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Implementing Layer-3 Multicast Routing

Multicast routing allows a host to send packets to a subset of all hosts as a group transmission rather than to a single host, as in unicast transmission, or to all hosts, as in broadcast transmission. The subset of hosts is known as group members and are identified by a single multicast group address that falls under the IP Class D address range from 224.0.0.0 through 239.255.255.255.

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

The following protocols are supported to implement multicast routing:

- IGMP—IGMP is used between hosts on a network (for example, LAN) and the routers on that network to track the multicast groups of which hosts are members.
- PIM SSM— Protocol Independent Multicast in Source-Specific Multicast (PIM-SSM) has the ability to report interest in receiving packets from specific source addresses (or from all but the specific source addresses), to an IP multicast address.

Prerequisites for Implementing Multicast Routing

- You must install and activate the multicast RPM package.
- You must be familiar with IPv4 multicast routing configuration tasks and concepts.
- Unicast routing must be operational.

Enabling Multicast

Configuration Example

Enables multicast routing and forwarding on all new and existing interfaces.

```
Router#config
Router(config)#multicast-routing
Router(config-mcast)#address-family ipv4
Router(config-mcast-default-ipv4)#interface all enable
*/In the above command, you can also indicate a specific interface (For example, interface
 TenGigE0/11/0/0)
for enabling multicast only on that interface/*
Router(config-mcast-default-ipv4)#commit
```

Running Configuration

```
Router#show running multicast routing
multicast-routing
 address-family ipv4
  interface all enable
  !
```

Verification

Verify that the Interfaces are enabled for multicast.

```
Router#show mfib interface location 0/RP0/cpu0
Interface : FINT0/RP0/cpu0 (Enabled)
SW Mcast pkts in : 0, SW Mcast pkts out : 0
TTL Threshold : 0
Ref Count : 2
Interface : TenGigE0/11/0/0 (Enabled)
SW Mcast pkts in : 0, SW Mcast pkts out : 0
TTL Threshold : 0
Ref Count : 3
Interface : TenGigE0/11/0/1 (Enabled)
SW Mcast pkts in : 0, SW Mcast pkts out : 0
TTL Threshold : 0
Ref Count : 13
Interface : Bundle-Ether1 (Enabled)
SW Mcast pkts in : 0, SW Mcast pkts out : 0
TTL Threshold : 0
Ref Count : 4
```

```
Interface : Bundle-Ether1.1 (Enabled)
SW Mcast pkts in : 0, SW Mcast pkts out : 0
TTL Threshold : 0
```

Supported Mulitcast Features

- Hardware Offloaded BFD for PIMv4 is supported.
- IPv4 and IPV6 static groups for both IGMPv2/v3 and MLDv1/v2 are supported.
- Protocol Independent Multicast in Source-Specific Multicast (PIM-SSM) mapping is supported.
- PIMv4 SSM over Bundle sub-interface is supported with the exception of PIMv6 SSM over Bundle sub-interface.
- Loadbalancing for multicast traffic for ECMP links and bundles is supported.
- Router needs to be reloaded to recover, if TCAM space is exceeded.
- Multicast MAC and multicast IP address should be matched for both Layer 2 and Layer 3 traffic, else traffic may be dropped by ASIC. L2 flooding is not supported.
- Multicast traffic fragmentation in hardware is not supported.
- Multicast traffic without Spanning-Tree protocol is supported at Layer 2 for multicast traffic without snooping enabled.
- IPv6 multicast MLD joins are subjected to hop by hop LPTS punt policer. Tweaking this policer to a higher value achieves convergence at higher scale.

  Also, adjust the ICMP control traffic LPTS hardware policer to a higher value for optimal convergence at higher scale.

IGMP Snooping Features

Supported Features

- IGMP Snooping on bridge domain is supported
- Multicast on BVI is supported.
- EVPN IGMP State Sync using IGMP snooping profile is supported.

Restrictions for Implementing Multicast Routing

- PIM SM is not supported for any of the supported MLDP profiles.
- DATA MDT with rate based or policy based switchover is not supported.
- Auto RP is not supported.

Protocol Independent Multicast

Protocol Independent Multicast (PIM) is a multicast routing protocol used to create multicast distribution trees, which are used to forward multicast data packets.

Proper operation of multicast depends on knowing the unicast paths towards a source or an RP. PIM relies on unicast routing protocols to derive this reverse-path forwarding (RPF) information. As the name PIM implies, it functions independently of the unicast protocols being used. PIM relies on the Routing Information Base (RIB) for RPF information. Protocol Independent Multicast (PIM) is designed to send and receive multicast routing updates.

PIM on Bundle-Ethernet subinterface is supported.

PIM BFD Overview

The BFD Support for Multicast (PIM) feature, also known as PIM BFD, registers PIM as a client of BFD. PIM can then utilize BFD's fast adjacency failure detection. When PIM BFD is enabled, BFD enables faster failure detection without waiting for hello messages from PIM.

At PIMs request, as a BFD client, BFD establishes and maintains a session with an adjacent node for maintaining liveness and detecting forwarding path failure to the adjacent node. PIM hellos will continue to be exchanged between the neighbors even after BFD establishes and maintains a BFD session with the neighbor. The behavior of the PIM hello mechanism is not altered due to the introduction of this feature. Although PIM depends on the Interior Gateway Protocol (IGP) and BFD is supported in IGP, PIM BFD is independent of IGP's BFD.

Protocol Independent Multicast (PIM) uses a hello mechanism for discovering new PIM neighbors between adjacent nodes. The minimum failure detection time in PIM is 3 times the PIM Query-Interval. To enable faster failure detection, the rate at which a PIM hello message is transmitted on an interface is configurable. However, lower intervals increase the load on the protocol and can increase CPU and memory utilization and cause a system-wide negative impact on performance. Lower intervals can also cause PIM neighbors to expire frequently as the neighbor expiry can occur before the hello messages received from those neighbors are processed. When PIM BFD is enabled, BFD enables faster failure detection without waiting for hello messages from PIM.

Configure PIM BFD

Note PIM BFD for IPv6 is not supported.

This section describes how you can configure PIM BFD

```
Router# configure
Router(config)# router pim address-family ipv4
Router(config-pim-default-ipv4)# interface HundredGigE0/9/0/0
Router(config-pim-ipv4-if)#  bfd minimum-interval 10
Router(config-pim-ipv4-if)#  bfd fast-detect
Router(config-pim-ipv4-if)#  bfd multiplier 3
Router(config-pim-ipv4)#  exit
Router(config-pim-default-ipv4)# interface TenGigE0/11/0/0
```

```
Router(config-pim-ipv4-if)#  bfd minimum-interval 50
Router(config-pim-ipv4-if)#  bfd fast-detect
Router(config-pim-ipv4-if)#  bfd multiplier 3
Router(config-pim-ipv4-if)#  exit
```

Running Configuration

```
router pim
 address-family ipv4
  interface HundredGigE 0/9/0/0
   bfd minimum-interval 10
   bfd fast-detect
   bfd multiplier 3
  !
  interface TenGigE 0/11/0/0
   bfd minimum-interval 50
   bfd fast-detect
   bfd multiplier 3
  !

  !
    !

  !
 !
!
```

Verification

The show outputs given in the following section display the details of the configuration of the PIM BFD, and the status of their configuration.

```
Router# show bfd session
Wed Nov 22 08:27:35.952 PST
Interface           Dest Addr    Local det time(int*mult)  State      Echo       Async
H/W     NPU
------------------- --------------- ----------------- ----------------- -----      -----
-----    -----
Hu0/9/0/0           10.12.12.2      0s(0s*0) 90ms(30ms*3)     UP                  Yes
0/RP0/CPU0


Te0/11/0/0      10.112.112.2   0s(0s*0) 90ms(30ms*3)     UP                Yes   0/RP0/CPU0




Router# show bfd client

Name             Node       Num sessions
---------------- ---------- --------------
L2VPN_ATOM       0/RP0/CPU0 0
MPLS-TR          0/RP0/CPU0 0
bgp-default      0/RP0/CPU0 0
bundlemgr_distrib 0/RP0/CPU0 14
isis-1           0/RP0/CPU0 0
object_tracking 0/RP0/CPU0 0
pim6             0/RP0/CPU0 0
```

```
pim              0/RP0/CPU0 0
service-layer    0/RP0/CPU0 0
```

Reverse Path Forwarding

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

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

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

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

Sparse-mode PIM uses the RPF lookup function to determine where it needs to send joins and prunes. (S,G) joins (which are source-tree states) are sent toward the source. (*,G) joins (which are shared-tree states) are sent toward the RP.

Setting the Reverse Path Forwarding Statically

Configuration Example

The following example configures the static RPF rule for IP address 10.0.0.1:

```
Router#configure
Router(config)#multicast-routing
Router(config-mcast)#address-family ipv4
Router(config-mcast)#static-rpf 10.0.0.1 32 TenGigE 0/0/0/1 192.168.0.2
Router(config-mcast)#commit
```

Running Configuration

```
multicast-routing
   address-family ipv4
      static-rpf 10.10.10.2 32 TenGigE0/0/0/1 192.168.0.2
```

Verification

Verify that RPF is chosen according to the static RPF configuration for 10.10.10.2

```
Router#show pim rpf
Table: IPv4-Unicast-default
* 10.10.10.2/32 [0/0]
    via GigabitEthernet0/0/0/1 with rpf neighbor 192.168.0.2
```

RPF Vector Encoding Using IETF Standard

RPF vector is a PIM proxy that lets core routers without RPF information forward join and prune messages for external sources (for example, a MPLS-based BGP-free core, where the MPLS core router is without external routes learned from BGP). The RPF vector encoding is now compatible with the new IETF encoding. The new IETF standard encodes PIM messages using PIM Hello option 26.

Configuring RPF Vector (IETF Standard Encoding)

This example shows how to enable RPF encoding using IETF standard:

```
(config)# router pim
(config-pim-default-ipv4)# address-family ipv4
(config-pim-default-ipv4)# rpf-vector use-standard-encoding
!
(config)# multicast-routing
(config-mcast)# interface TenGigE 0/11/0/0
(config-mcast)# interface TenGigE 0/11/0/1
```

Verification

```
Router#show pim neighbor
Tue Apr 17 10:15:40.961 PDT

PIM neighbors in VRF default
Flag: B - Bidir capable, P - Proxy capable, DR - Designated Router,
      E - ECMP Redirect capable
      * indicates the neighbor created for this router

Neighbor Address             Interface              Uptime    Expires  DR pri   Flags

25.25.25.1                   TenGigE 0/11/0/0      1w3d      00:01:36 1      B P
25.25.25.2*                TenGigE 0/11/0/0       1w3d      00:01:41 1 (DR) B P E
32.32.32.2*                  TenGigE 0/11/0/1
       1w4d      00:01:40 1      B P E
32.32.32.3                   TenGigE 0/11/0/1
       1w4d      00:01:42 1 (DR) B P
```

In the above output, you can see "P" tag on the multicast enabled interfaces.

PIM Bootstrap Router

The PIM bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution mechanism that simplifies the Auto-RP process. This feature is enabled by default allowing routers to dynamically learn the group-to-RP mappings.

PIM uses the BSR to discover and announce RP-set information for each group prefix to all the routers in a PIM domain. This is the same function accomplished by Auto-RP, but the BSR is part of the PIM specification. The BSR mechanism interoperates with Auto-RP on Cisco routers.

To avoid a single point of failure, you can configure several candidate BSRs in a PIM domain. A BSR is elected among the candidate BSRs automatically.

Candidates use bootstrap messages to discover which BSR has the highest priority. The candidate with the highest priority sends an announcement to all PIM routers in the PIM domain that it is the BSR.

Routers that are configured as candidate RPs unicast to the BSR the group range for which they are responsible. The BSR includes this information in its bootstrap messages and disseminates it to all PIM routers in the domain. Based on this information, all routers are able to map multicast groups to specific RPs. As long as a router is receiving the bootstrap message, it has a current RP map.

Configuring PIM Bootstrap Router

Configuration Example

Configures the router as a candidate BSR with a hash mask length of 30:

```
Router#config
Router(config)#router pim
Router(config-pim-default-ipv4)#bsr candidate-bsr 1.1.1.1 hash-mask-len 30 priority 1
Router(config-pim-default-ipv4-if)#commit
```

Configures the router to advertise itself as a candidate rendezvous point to the BSR in its PIM domain. Access list number 4 specifies the prefix associated with the candidate rendezvous point address 1.1.1.1 . This rendezvous point is responsible for the groups with the prefix 239.

```
Router#config
Router(config)#router pim
Router(config-pim-default-ipv4)#bsr candidate-rp 1.1.1.1 group-list 4 priority 192 interval
 60

Router(config-pim-default-ipv4)#exit
Router(config)#ipv4 access-list 4
Router(config-ipv4-acl)#permit ipv4 any 239.0.0.0 0.255.255.255
Router(config-ipv4-acl)#commit
```

Running Configuration

```
Router#show run router pim
router pim
 address-family ipv4
  bsr candidate-bsr 1.1.1.1 hash-mask-len 30 priority 1
  bsr candidate-rp 1.1.1.1 group-list 4 priority 192 interval 60
```

Verification

```
Router#show pim rp mapping
PIM Group-to-RP Mappings
Group(s) 239.0.0.0/8
  RP 1.1.1.1 (?), v2
    Info source: 1.1.1.1 (?), elected via bsr, priority 192, holdtime 150
      Uptime: 00:02:50, expires: 00:01:54

Router#show pim bsr candidate-rp
PIM BSR Candidate RP Info
Cand-RP          mode  scope priority uptime    group-list
1.1.1.1           BD    16      192       00:04:06    4

Router#show pim bsr election
PIM BSR Election State
Cand/Elect-State           Uptime   BS-Timer     BSR                                  C-BSR
Elected/Accept-Pref     00:03:49 00:00:25 1.1.1.1 [1, 30]          1.1.1.1 [1, 30]
```

PIM-Source Specific Multicast

When PIM is used in SSM mode, multicast routing is easier to manage. This is because RPs (rendezvous points) are not required and therefore, no shared trees (*,G) are built.

There is no specific IETF document defining PIM-SSM. However, RFC4607 defines the overall SSM behavior.

In the rest of this document, we use the term PIM-SSM to describe PIM behavior and configuration when SSM is used.

PIM in Source-Specific Multicast operation uses information found on source addresses for a multicast group provided by receivers and performs source filtering on traffic.

- By default, PIM-SSM operates in the 232.0.0.0/8 multicast group range for IPv4 and FF3x::/32 for IPv6. To configure these values, use the **ssm range** command.
- If SSM is deployed in a network already configured for PIM-SM, only the last-hop routers must be upgraded with Cisco IOS XR Software that supports the SSM feature.
- No MSDP SA messages within the SSM range are accepted, generated, or forwarded.
- SSM can be disabled using the **ssm disable** command.
- The **ssm allow-override** command allows SSM ranges to be overridden by more specific ranges.

In many multicast deployments where the source is known, protocol-independent multicast-source-specific multicast (PIM-SSM) mapping is the obvious multicast routing protocol choice to use because of its simplicity. Typical multicast deployments that benefit from PIM-SSM consist of entertainment-type solutions like the ETTH space, or financial deployments that completely rely on static forwarding.

In SSM, delivery of data grams is based on (S,G) channels. Traffic for one (S,G) channel consists of datagrams with an IP unicast source address S and the multicast group address G as the IP destination address. Systems receive traffic by becoming members of the (S,G) channel. Signaling is not required, but receivers must subscribe or unsubscribe to (S,G) channels to receive or not receive traffic from specific sources. Channel subscription signaling uses IGMP to include mode membership reports, which are supported only in Version 3 of IGMP (IGMPv3).

To run SSM with IGMPv3, SSM must be supported on the multicast router, the host where the application is running, and the application itself. Cisco IOS XR Software allows SSM configuration for an arbitrary subset of the IP multicast address range 224.0.0.0 through 239.255.255.255.

When an SSM range is defined, existing IP multicast receiver applications do not receive any traffic when they try to use addresses in the SSM range, unless the application is modified to use explicit (S,G) channel subscription.

Benefits of PIM-SSM over PIM-SM

PIM-SSM is derived from PIM-SM. However, whereas PIM-SM allows for the data transmission of all sources sending to a particular group in response to PIM join messages, the SSM feature forwards traffic to receivers only from those sources that the receivers have explicitly joined. Because PIM joins and prunes are sent directly towards the source sending traffic, an RP and shared trees are unnecessary and are disallowed. SSM is used to optimize bandwidth utilization and deny unwanted Internet broad cast traffic. The source is provided by interested receivers through IGMPv3 membership reports.

IGMPv2

To support IGMPv2, SSM mapping configuration must be added while configuring IGMP to match certain sources to group range.

Configuring Example

Configures the access-list (mc1):

```
Router#configure
Router(config)#ipv4 access-list mc1
Router(config-ipv4-acl)#permit ipv4 any 232.1.1.0 0.0.0.255
Router(config-ipv4-acl)#commit
```

Configures the multicast source (1.1.1.1) as part of a set of sources that map SSM groups described by the specified access-list (mc1):

```
Router#configure
Router(config)#router igmp
Router(config-igmp)#ssm map static 1.1.1.1 mc1
Router(config-igmp)#commit
```

Running Configuration

```
Router#show run router igmp
router igmp
ssm map static 1.1.1.1 mc1
```

Multipath Option

The multipath option is available under `router pim` configuration mode. After multipath option is enabled, SSM selects different path to reach same destination instead of choosing common path. The multipath option helps load balance the SSM traffic.

Configuring Multipath Option

```
Router#configure
Router(config)#router pim address-family ipv4
Router(config-pim-default-ipv4)#multipath hash source
Router(config-pim-default-ipv4)#commit
```

Running Configuration

```
Router#show running router pim
router pim
 address-family ipv4
  dr-priority 100
  multipath hash source    /*SSM traffic takes different path to reach same destination
based on source hash value.*/
```

Verification

The Bundle-Ether132 and TenGigE0/11/0/1.132 are two paths to reach the destination router Turnin-56. Since we have enabled multipath option, the source has two IP addresses 50.11.30.12 and 50.11.30.11. The Multicast traffic from two sources take two different paths Bundle-Ether132 and TenGigE0/11/0/1.132 to reach same destination.

This show run output shows that Bundle-Ether132 and TenGigE0/11/0/1.132 are connected to same destination router Turnin-56:

```
Router#show run int TenGigE0/11/0/2.132
interface TenGigE0/1/11/2/3.132
 description Connected to Turin-56 ten0/11/0/1.132
 ipv4 address 13.0.2.1 255.255.255.240
 ipv6 address 2606::13:0:2:1/120
 encapsulation dot1q 132
!

Router#show run int be132
interface Bundle-Ether132
 description Bundle between Fretta-56 and Turin-56
 ipv4 address 28.0.0.1 255.255.255.240
 ipv6 address 2606::28:0:0:1/120
 load-interval 30

Router#show mrib route 50.11.30.11 detail

IP Multicast Routing Information Base
Entry flags: L - Domain-Local Source, E - External Source to the Domain,
    C - Directly-Connected Check, S - Signal, IA - Inherit Accept,
    IF - Inherit From, D - Drop, ME - MDT Encap, EID - Encap ID,
    MD - MDT Decap, MT - MDT Threshold Crossed, MH - MDT interface handle
    CD - Conditional Decap, MPLS - MPLS Decap, EX - Extranet
    MoFE - MoFRR Enabled, MoFS - MoFRR State, MoFP - MoFRR Primary
    MoFB - MoFRR Backup, RPFID - RPF ID Set, X - VXLAN
Interface flags: F - Forward, A - Accept, IC - Internal Copy,
    NS - Negate Signal, DP - Don't Preserve, SP - Signal Present,
    II - Internal Interest, ID - Internal Disinterest, LI - Local Interest,
    LD - Local Disinterest, DI - Decapsulation Interface
    EI - Encapsulation Interface, MI - MDT Interface, LVIF - MPLS Encap,
    EX - Extranet, A2 - Secondary Accept, MT - MDT Threshold Crossed,
    MA - Data MDT Assigned, LMI - mLDP MDT Interface, TMI - P2MP-TE MDT Interface
    IRMI - IR MDT Interface

(50.11.30.11,225.255.11.1) Ver: 0x523cc294 RPF nbr: 50.11.30.11 Flags: L RPF, FGID: 11453,
 -1, -1
  Up: 4d15h
  Incoming Interface List
    HundredGigE0/9/0/3.1130 Flags: A, Up: 4d15h
  Outgoing Interface List
    TenGigE0/11/0/6 Flags: F NS, Up: 4d15h
    TenGigE0/1/0/6/3.132 Flags: F NS, Up: 4d15h
    TenGigE0/11/0/1.122 Flags: F NS, Up: 4d15h

Router#show mrib route 50.11.30.12 detail

IP Multicast Routing Information Base
Entry flags: L - Domain-Local Source, E - External Source to the Domain,
    C - Directly-Connected Check, S - Signal, IA - Inherit Accept,
    IF - Inherit From, D - Drop, ME - MDT Encap, EID - Encap ID,
    MD - MDT Decap, MT - MDT Threshold Crossed, MH - MDT interface handle
    CD - Conditional Decap, MPLS - MPLS Decap, EX - Extranet
    MoFE - MoFRR Enabled, MoFS - MoFRR State, MoFP - MoFRR Primary
    MoFB - MoFRR Backup, RPFID - RPF ID Set, X - VXLAN
Interface flags: F - Forward, A - Accept, IC - Internal Copy,
    NS - Negate Signal, DP - Don't Preserve, SP - Signal Present,
    II - Internal Interest, ID - Internal Disinterest, LI - Local Interest,
    LD - Local Disinterest, DI - Decapsulation Interface
    EI - Encapsulation Interface, MI - MDT Interface, LVIF - MPLS Encap,
    EX - Extranet, A2 - Secondary Accept, MT - MDT Threshold Crossed,
    MA - Data MDT Assigned, LMI - mLDP MDT Interface, TMI - P2MP-TE MDT Interface
    IRMI - IR MDT Interface
```

```
(50.11.30.12,226.255.12.1) Ver: 0x5fe02e5b RPF nbr: 50.11.30.12 Flags: L RPF, FGID: 12686,
 -1, -1
  Up: 4d15h
  Incoming Interface List
    HundredGigE0/9/0/1.1130 Flags: A, Up: 4d15h
  Outgoing Interface List
    Bundle-Ether121 Flags: F NS, Up: 4d15h
    Bundle-Ether132 Flags: F NS, Up: 4d15h
    TenGigE0/11/0/6.117 Flags: F NS, Up: 4d15h
```

Configuring PIM-SSM

Configuration Example

Configures SSM service for the IPv4 address range defined by access list 4.

```
Router#config
Router(config)#ipv4 access-list 4
Router(config-ipv4-acl)#permit ipv4 any 224.2.151.0 0.0.0.255
Router(config-ipv4-acl)#exit
Router(config)#multicast-routing
Router(config-mcast)#address-family ipv4
Router(config-mcast-default-ipv4)#ssm range 4
Router(config-mcast-default-ipv4)#commit
Router(config-mcast-default-ipv4)#end
```

Running Configuration

```
Router#show running multicast-routing
multicast-routing
 address-family ipv4
  ssm range 4
  interface all enable
 !
```

Verification

Verify if the SSM range is configured according to the set parameters:

```
Router#show access-lists 4
ipv4 access-list 4
 10 permit ipv4 any 224.2.151.0 0.0.0.255

*/Verify if the SSM is configured for 224.2.151.0/24/*:

Router#show pim group-map
IP PIM Group Mapping Table
(* indicates group mappings being used)
Group Range         Proto Client   Groups RP address      Info
224.0.1.39/32*      DM    perm     1      0.0.0.0
224.0.1.40/32*      DM    perm     1      0.0.0.0
224.0.0.0/24*       NO    perm     0      0.0.0.0
224.2.151.0/24*     SSM   config   0      0.0.0.0
```

Configuring PIM Parameters

To configure PIM-specific parameters, the router pim configuration mode is used. The default configuration prompt is for IPv4 and will be seen as config-pim-default-ipv4. To ensure the election of a router as PIM DR

on a LAN segment, use the **dr-priority** command. The router with the highest DR priority will win the election. By default, at a preconfigured threshold, the last hop router can join the shortest path tree to receive multicast traffic. To change this behavior, use the command **spt-threshold infinity** under the router pim configuration mode. This will result in the last hop router permanently joining the shared tree. The frequency at which a router sends PIM hello messages to its neighbors can be configured by the hello-interval command. By default, PIM hello messages are sent once every 30 seconds. If the hello-interval is configured under router pim configuration mode, all the interfaces with PIM enabled will inherit this value. To change the hello interval on the interface, use the **hello-interval** command under interface configuration mode, as follows:

Configuration Example

```
Router#configure
Router(config)#router pim
Router(config-pim-default)#address-family ipv4
Router(config-pim-default-ipv4)#dr-priority 2
Router(config-pim-default-ipv4)#spt-threshold infinity
Router(config-pim-default-ipv4)#interface TenGigE0/11/0/1
Router(config-pim-ipv4-if)#dr-priority 4
Router(config-pim-ipv4-if)#hello-interval 45
Router(config-pim-ipv4-if)#commit
```

Running Configuration

```
Router#show run router pim
router pim
 address-family ipv4
  dr-priority 2
  spt-threshold infinity
  interface TenGigE0/11/0/1
   dr-priority 4
   hello-interval 45
```

Verification

Verify if the parameters are set according to the configured values:

```
Router#show pim interface te0/11/0/1
PIM interfaces in VRF default
Address               Interface                PIM  Nbr   Hello  DR    DR Count Intvl
  Prior
100.1.1.1             TenGigE0/11/0/1           on   1     45     4     this system
```

Multicast Source Discovery Protocol

Multicast Source Discovery Protocol (MSDP) is a mechanism to connect multiple PIM sparse-mode domains. MSDP allows multicast sources for a group to be known to all rendezvous points (RPs) in different domains. Each PIM-SM domain uses its own RPs and need not depend on RPs in other domains.

An RP in a PIM-SM domain has MSDP peering relationships with MSDP-enabled routers in other domains. Each peering relationship occurs over a TCP connection, which is maintained by the underlying routing system.

MSDP speakers exchange messages called Source Active (SA) messages. When an RP learns about a local active source, typically through a PIM register message, the MSDP process encapsulates the register in an SA message and forwards the information to its peers. The message contains the source and group information for the multicast flow, as well as any encapsulated data. If a neighboring RP has local joiners for the multicast

group, the RP installs the S, G route, forwards the encapsulated data contained in the SA message, and sends PIM joins back towards the source. This process describes how a multicast path can be built between domains.

Note Although you should configure BGP or Multiprotocol BGP for optimal MSDP interdomain operation, this is not considered necessary in the Cisco IOS XR Software implementation. For information about how BGP or Multiprotocol BGP may be used with MSDP, see the MSDP RPF rules listed in the Multicast Source Discovery Protocol (MSDP), Internet Engineering Task Force (IETF) Internet draft.

Interconnecting PIM-SM Domains with MSDP

To set up an MSDP peering relationship with MSDP-enabled routers in another domain, you configure an MSDP peer to the local router.

If you do not want to have or cannot have a BGP peer in your domain, you could define a default MSDP peer from which to accept all Source-Active (SA) messages.

Finally, you can change the Originator ID when you configure a logical RP on multiple routers in an MSDP mesh group.

Before you begin

You must configure MSDP default peering, if the addresses of all MSDP peers are not known in BGP or multiprotocol BGP.

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config)# interface loopback 0`	(Optional) Enters interface configuration mode to define the IPv4 address for the interface. **Note** This step is required if you specify an interface type and number whose primary address becomes the source IP address for the TCP connection.
Step 3	**ipv4 address** *address mask* **Example:** `RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.0.1.3 255.255.255.0`	(Optional) Defines the IPv4 address for the interface. **Note** This step is required only if you specify an interface type and number whose primary address becomes the source IP address for the TCP connection. See optional for information about configuring the **connect-source** command.

	Command or Action	Purpose
Step 4	**exit** **Example:** `RP/0/RP0/CPU0:router(config-if)# end`	Exits interface configuration mode.
Step 5	**router msdp** **Example:** `RP/0/RP0/CPU0:router(config)# router msdp`	Enters MSDP protocol configuration mode.
Step 6	**default-peer** *ip-address* [**prefix-list** *list*] **Example:** `RP/0/RP0/CPU0:router(config-msdp)# default-peer 172.23.16.0`	(Optional) Defines a default peer from which to accept all MSDP SA messages.
Step 7	**originator-id** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config-msdp)# originator-id /1/1/0`	(Optional) Allows an MSDP speaker that originates a (Source-Active) SA message to use the IP address of the interface as the RP address in the SA message.
Step 8	**peer** *peer-address* **Example:** `RP/0/RP0/CPU0:router(config-msdp)# peer 172.31.1.2`	Enters MSDP peer configuration mode and configures an MSDP peer. • Configure the router as a BGP neighbor. • If you are also BGP peering with this MSDP peer, use the same IP address for MSDP and BGP. You are not required to run BGP or multiprotocol BGP with the MSDP peer, as long as there is a BGP or multiprotocol BGP path between the MSDP peers.
Step 9	**connect-source** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config-msdp-peer)# connect-source loopback 0`	(Optional) Configures a source address used for an MSDP connection.
Step 10	**mesh-group** *name* **Example:**	(Optional) Configures an MSDP peer to be a member of a mesh group.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-msdp-peer)# mesh-group internal`	
Step 11	**remote-as** *as-number* **Example:** `RP/0/RP0/CPU0:router(config-msdp-peer)# remote-as 250`	(Optional) Configures the remote autonomous system number of this peer.
Step 12	**commit**	
Step 13	**show msdp** [**ipv4**] **globals** **Example:** `RP/0/RP0/CPU0:router# show msdp globals`	Displays the MSDP global variables.
Step 14	**show msdp** [**ipv4**] **peer** [*peer-address*] **Example:** `RP/0/RP0/CPU0:router# show msdp peer 172.31.1.2`	Displays information about the MSDP peer.
Step 15	**show msdp** [**ipv4**] **rpf** *rpf-address* **Example:** `RP/0/RP0/CPU0:router# show msdp rpf 172.16.10.13`	Displays the RPF lookup.

Controlling Source Information on MSDP Peer Routers

Your MSDP peer router can be customized to control source information that is originated, forwarded, received, cached, and encapsulated.

When originating Source-Active (SA) messages, you can control to whom you will originate source information, based on the source that is requesting information.

When forwarding SA messages you can do the following:

- Filter all source/group pairs
- Specify an extended access list to pass only certain source/group pairs
- Filter based on match criteria in a route map

When receiving SA messages you can do the following:

- Filter all incoming SA messages from an MSDP peer
- Specify an extended access list to pass certain source/group pairs
- Filter based on match criteria in a route map

In addition, you can use time to live (TTL) to control what data is encapsulated in the first SA message for every source. For example, you could limit internal traffic to a TTL of eight hops. If you want other groups to go to external locations, you send those packets with a TTL greater than eight hops.

By default, MSDP automatically sends SA messages to peers when a new member joins a group and wants to receive multicast traffic. You are no longer required to configure an SA request to a specified MSDP peer.

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**router msdp** **Example:** `RP/0/RP0/CPU0:router(config)# router msdp`	Enters MSDP protocol configuration mode.
Step 3	**sa-filter** {**in** | **out**} [**list** *access-list-name*] [**rp-list** *access-list-name*] **Example:** `RP/0/RP0/CPU0:router(config-msdp)# sa-filter out list 100`	Configures an incoming or outgoing filter list for messages received from the specified MSDP peer. • If you specify both the **list** and **rp-list** keywords, all conditions must be true to pass any source, group (S, G) pairs in outgoing Source-Active (SA) messages. • You must configure the **ipv4 access-list** command in Step 7, on page 18. • If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** filters routes.
Step 4	**cache-sa-state** [**list** *access-list-name*] [**rp-list** *access-list-name*] **Example:** `RP/0/RP0/CPU0:router(config-msdp)# cache-sa-state list 100`	Creates and caches source/group pairs from received Source-Active (SA) messages and controls pairs through access lists.
Step 5	**ttl-threshold** *ttl-value* **Example:**	(Optional) Limits which multicast data is sent in SA messages to an MSDP peer.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-msdp)# ttl-threshold 8`	• Only multicast packets with an IP header TTL greater than or equal to the *ttl-value* argument are sent to the MSDP peer specified by the IP address or name. • Use this command if you want to use TTL to examine your multicast data traffic. For example, you could limit internal traffic to a TTL of 8. If you want other groups to go to external locations, send those packets with a TTL greater than 8. • This example configures a TTL threshold of eight hops.
Step 6	**exit** **Example:** `RP/0/RP0/CPU0:router(config-msdp)# exit`	Exits the current configuration mode.
Step 7	**ipv4 access-list** *name* [*sequence-number*] **permit** *source* [*source-wildcard*] **Example:** `RP/0/RP0/CPU0:router(config)# ipv4 access-list 100 20 permit 239.1.1.1 0.0.0.0`	Defines an IPv4 access list to be used by SA filtering. • In this example, the access list 100 permits multicast group 239.1.1.1. • The **ipv4 access-list** command is required if the keyword **list** is configured for SA filtering in Step 3, on page 17.
Step 8	**commit**	

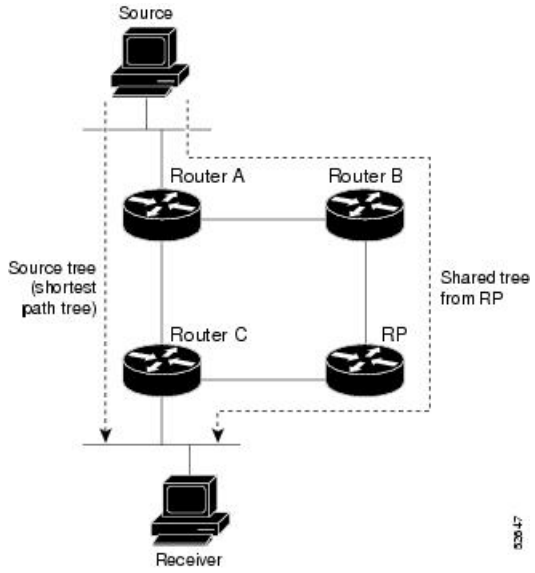
PIM-Sparse Mode

Typically, PIM in sparse mode (PIM-SM) operation is used in a multicast network when relatively few routers are involved in each multicast. Routers do not forward multicast packets for a group, unless there is an explicit request for traffic. Requests are accomplished using PIM join messages, which are sent hop by hop toward the root node of the tree. The root node of a tree in PIM-SM is the rendezvous point (RP) in the case of a shared tree or the first-hop router that is directly connected to the multicast source in the case of a shortest path tree (SPT). The RP keeps track of multicast groups, and the sources that send multicast packets are registered with the RP by the first-hop router of the source.

As a PIM join travels up the tree, routers along the path set up the multicast forwarding state so that the requested multicast traffic is forwarded back down the tree. When multicast traffic is no longer needed, a router sends a PIM prune message up the tree toward the root node to prune (or remove) the unnecessary traffic. As this PIM prune travels hop by hop up the tree, each router updates its forwarding state appropriately. Ultimately, the forwarding state associated with a multicast group or source is removed. Additionally, if prunes are not explicitly sent, the PIM state will timeout and be removed in the absence of any further join messages.

This image shows IGMP and PIM-SM operating in a multicast environment.

Figure 1: Shared Tree and Source Tree (Shortest Path Tree)



In PIM-SM, the rendezvous point (RP) is used to bridge sources sending data to a particular group with receivers sending joins for that group. In the initial set up of state, interested receivers receive data from senders to the group across a single data distribution tree rooted at the RP. This type of distribution tree is called a shared tree or rendezvous point tree (RPT) as illustrated in Figure 4: Shared Tree and Source Tree (Shortest Path Tree), above. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Unless the command is configured, this initial state gives way as soon as traffic is received on the leaf routers (designated router closest to the host receivers). When the leaf router receives traffic from the RP on the RPT, the router initiates a switch to a data distribution tree rooted at the source sending traffic. This type of distribution tree is called a shortest path tree or source tree. By default, the Cisco IOS XR Software switches to a source tree when it receives the first data packet from a source.

The following process describes the move from shared tree to source tree in more detail:

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

9. Join and prune messages are sent for sources and RPs. They are sent hop by hop and are processed by each PIM router along the path to the source or RP. Register and register-stop messages are not sent hop by hop. They are exchanged using direct unicast communication between the designated router that is directly connected to a source and the RP for the group.

Note The **spt-threshold infinity** command lets you configure the router so that it never switches to the shortest path tree (SPT).

Designated Routers

Cisco routers use PIM to forward multicast traffic and follow an election process to select a designated router (DR) when there is more than one router on a LAN segment.

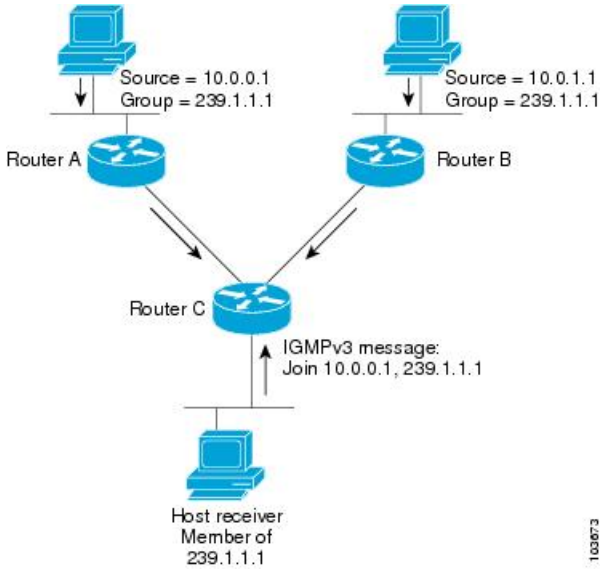
The designated router is responsible for sending PIM register and PIM join and prune messages toward the RP to inform it about host group membership.

If there are multiple PIM routers on a LAN, a designated router must be elected to avoid duplicating multicast traffic for connected hosts. The PIM router with the highest IP address becomes the DR for the LAN unless you choose to force the DR election by use of the **dr-priority** command. The DR priority option allows you to specify the DR priority of each router on the LAN segment (default priority = 1) so that the router with the highest priority is elected as the DR. If all routers on the LAN segment have the same priority, the highest IP address is again used as the tiebreaker.

Note DR election process is required only on multi access LANs. The last-hop router directly connected to the host is the DR.

The figure "Designated Router Election on a Multiaccess Segment", below illustrates what happens on a multi access segment. Router A (10.0.0.253) and Router B (10.0.0.251) are connected to a common multi access Ethernet segment with Host A (10.0.0.1) as an active receiver for Group A. As the Explicit Join model is used, only Router A, operating as the DR, sends joins to the RP to construct the shared tree for Group A. If Router B were also permitted to send (*,G) joins to the RP, parallel paths would be created and Host A would receive duplicate multicast traffic. When Host A begins to source multicast traffic to the group, the DR's responsibility is to send register messages to the RP. Again, if both routers were assigned the responsibility, the RP would receive duplicate multicast packets.

Figure 2: Designated Router Election on a Multiaccess Segment



If the DR fails, the PIM provides a way to detect the failure of Router A and to elect a failover DR. If the DR (Router A) were to become inoperable, Router B would detect this situation when its neighbor adjacency with Router A timed out. Because Router B has been hearing IGMP membership reports from Host A, it already has IGMP state for Group A on this interface and immediately sends a join to the RP when it becomes the new DR. This step reestablishes traffic flow down a new branch of the shared tree using Router B. Additionally, if Host A were sourcing traffic, Router B would initiate a new register process immediately after receiving the next multicast packet from Host A. This action would trigger the RP to join the SPT to Host A, using a new branch through Router B.

Note Two PIM routers are neighbors if there is a direct connection between them. To display your PIM neighbors, use the show pim neighbor command in EXEC mode.

- They are not used for unicast routing but are used only by PIM to look up an IPv4 next hop to a PIM source.
- They are not published to the Forwarding Information Base (FIB).
- When multicast-intact is enabled on an IGP, all IPv4 destinations that were learned through link-state advertisements are published with a set equal-cost mcast-intact next-hops to the RIB. This attribute applies even when the native next-hops have no IGP shortcuts.
- In IS-IS, the max-paths limit is applied by counting both the native and mcast-intact next-hops together. (In OSPFv2, the behavior is slightly different.)

Configuration Example

Configures the router to use DR priority 4 for TenGigE interface 0/11/0/1, but other interfaces will inherit DR priority 2:

```
Router#configure
Router(config)#router pim
```

```
Router(config-pim-default)#address-family ipv4
Router(config-pim-default-ipv4)#dr-priority 2
Router(config-pim-default-ipv4)#interface TenGigE0/11/0/1
Router(config-pim-ipv4-if)#dr-priority 4
Router(config-ipv4-acl)#commit
```

Running Configuration

```
Router#show run router pim
router pim
 address-family ipv4
  dr-priority 2
  spt-threshold infinity
  interface TenGigE0/11/0/1
   dr-priority 4
   hello-interval 45
```

Verification

Verify if the parameters are set according to the configured values:

```
Router#show pim interface
PIM interfaces in VRF default
Address           Interface           PIM  Nbr   Hello  DR    DR Count Intvl  Prior
100.1.1.1        TenGigE0/11/0/1        on   1     45     4     this system
```

Internet Group Management Protocol

Cisco IOS XR Software provides support for Internet Group Management Protocol (IGMP) over IPv4.

IGMP provides a means for hosts to indicate which multicast traffic they are interested in and for routers to control and limit the flow of multicast traffic throughout the network. Routers build state by means of IGMP messages; that is, router queries and host reports.

A set of routers and hosts that receive multicast data streams from the same source is called a multicast group. Hosts use IGMP messages to join and leave multicast groups.

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

Restrictions

IGMP snooping under VPLS bridge domain is not supported.

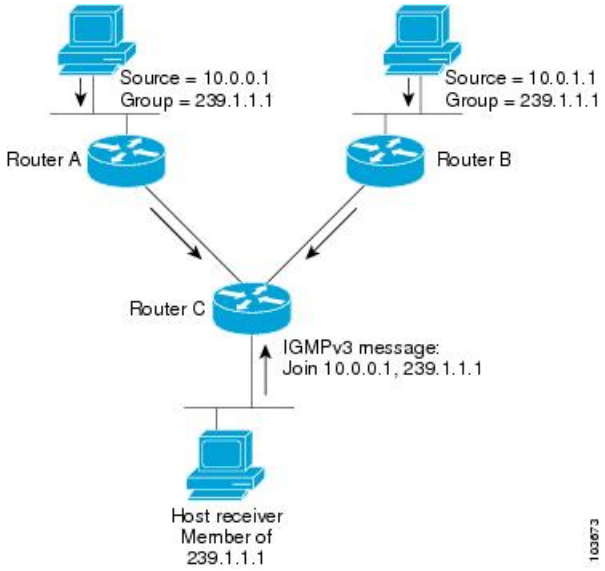
Functioning of IGMP Routing

The following image "IGMP Singaling" , illustrates two sources, 10.0.0.1 and 10.0.1.1, that are multicasting to group 239.1.1.1.

The receiver wants to receive traffic addressed to group 239.1.1.1 from source 10.0.0.1 but not from source 10.0.1.1.

The host must send an IGMPv3 message containing a list of sources and groups (S, G) that it wants to join and a list of sources and groups (S, G) that it wants to leave. Router C can now use this information to prune traffic from Source 10.0.1.1 so that only Source 10.0.0.1 traffic is being delivered to Router C.

Figure 3: IGMP Signaling



Configuring Maximum IGMP Per Interface Group Limit

The IGMP Per Interface States Limit sets a limit on creating OIF for the IGMP interface. When the set limit is reached, the group is not accounted against this interface but the group can exist in IGMP context for some other interface.

- If a user has configured a maximum of 20 groups and has reached the maximum number of groups, then no more groups can be created. If the user reduces the maximum number of groups to 10, the 20 joins will remain and a message of reaching the maximum is displayed. No more joins can be added until the number of groups has reached less than 10.
- If a user already has configured a maximum of 30 joins and add a max of 20, the configuration occurs displaying a message that the maximum has been reached. No state change occurs and also no more joins can occur until the threshold number of groups is brought down below the maximum number of groups.

Configuration Example

Configures all interfaces with 4000 maximum groups per interface except TenGigE interface 0/11/0/0, which is set to 3000:

```
Router#config
Router(config)#router igmp
Router(config-igmp)#maximum groups-per-interface 4000
Router(config-igmp)#interface TenGigE0/11/0/0
Router(config-igmp-default-if)#maximum groups-per-interface 3000
Router(config-igmp-default-if)#commit
```

Running Configuration

```
router igmp
 interface TenGigE0/11/0/0
  maximum groups-per-interface 3000
 !
 maximum groups-per-interface 4000
!
```

Verification

```
Router#show igmp summary
Robustness Value 2
No. of Group x Interfaces 37
Maximum number of Group x Interfaces 50000
Supported Interfaces   : 9
Unsupported Interfaces: 0
Enabled Interfaces     : 8
Disabled Interfaces    : 1
MTE tuple count        : 0
Interface                      Number  Max #
                               Groups  Groups
Loopback0                       4       4000
TenGigE0/11/0/0                  5       4000
TenGigE0/11/0/1                  5       4000
TenGigE0/11/0/2                  0       4000
TenGigE0/11/0/3                  5       4000
TenGigE0/11/0/4                  5       3000
TenGigE0/11/0/5                 5       4000
TenGigE0/11/0/6                 5       4000
TenGigE0/11/0/6.1                3       4000
```

SSM Static Source Mapping

Configure a source (1.1.1.1) as part of a set of sources that map SSM groups described by the specified access-list (4).

Configuration Example

```
Router#configure
Router(config)#ipv4 access-list 4
Router(config-ipv4-acl)#permit ipv4 any 229.1.1.0 0.0.0.255
Router(config-ipv4-acl)#exit
Router(config)# multicast-routing
Router(config-mcast)#address-family ipv4
Router(config-mcast-default-ipv4)#ssm range 4
Router(config-mcast-default-ipv4)#exit
Router(config-mcast)#exit
Router(config)#router igmp
Router(config-igmp)#ssm map static 1.1.1.1 4
*/Repeat the above step as many times as you have source addresses to include in the set
for SSM mapping/*
Router(config-igmp)#interface TenGigE0/11/0/3
Router(config-igmp-default-if)#static-group 229.1.1.1
Router(config-igmp-default-if)#commit
```

Running Configuration

```
Router#show run multicast-routing
multicast-routing
 address-family ipv4
```

```
  ssm range 4
  interface all enable
 !
!
Router#show access-lists 4
ipv4 access-list 4
 10 permit ipv4 any 229.1.1.0 0.0.0.255

Router#show run router igmp
router igmp
 interface TenGigE0/11/0/3
 static-group 229.1.1.1
 !
 ssm map static 1.1.1.1 4
```

Verification

Verify if the parameters are set according to the configured values:

```
Router#show mrib route 229.1.1.1 detail
IP Multicast Routing Information Base
Entry flags: L - Domain-Local Source, E - External Source to the Domain,
    C - Directly-Connected Check, S - Signal, IA - Inherit Accept,
    IF - Inherit From, D - Drop, ME - MDT Encap, EID - Encap ID,
    MD - MDT Decap, MT - MDT Threshold Crossed, MH - MDT interface handle
    CD - Conditional Decap, MPLS - MPLS Decap, EX - Extranet
    MoFE - MoFRR Enabled, MoFS - MoFRR State, MoFP - MoFRR Primary
    MoFB - MoFRR Backup, RPFID - RPF ID Set, X - VXLAN
Interface flags: F - Forward, A - Accept, IC - Internal Copy,
    NS - Negate Signal, DP - Don't Preserve, SP - Signal Present,
    II - Internal Interest, ID - Internal Disinterest, LI - Local Interest,
    LD - Local Disinterest, DI - Decapsulation Interface
    EI - Encapsulation Interface, MI - MDT Interface, LVIF - MPLS Encap,
    EX - Extranet, A2 - Secondary Accept, MT - MDT Threshold Crossed,
    MA - Data MDT Assigned, LMI - mLDP MDT Interface, TMI - P2MP-TE MDT Interface
    IRMI - IR MDT Interface
(1.1.1.1,229.1.1.1) RPF nbr: 1.1.1.1 Flags: RPF
  Up: 00:01:11
  Incoming Interface List
    Loopback0 Flags: A, Up: 00:01:11
  Outgoing Interface List
    TenGigE0/11/0/3 Flags: F NS LI, Up: 00:01:11
```

IPv6 Multicast for Multiple Sources

Multicast Route Statistics

Multicast route statistic feature provides information about the multicast routes. The multicast statistics information includes the rate at which packets are received.

Before enabling multicast route statistics, you must configure an ACL to specify which of the IP route statistics to be captured.

Restrictions for Implementing Multicast Route Statistics Feature

These are the points that you should consider before implementing multicast route statistics feature:

- Multicast route statistics are available for <S,G> routes only. The statistics for <*,G> routes are not available.
- IPv6 multicast route statistics are not supported.
- Multicast route statistics for egress direction is not supported.
- When ACL is mapped with **hw-module router-stats** configuration, you can't modify the ACL. To modify ACLs that are mapped with router-stats, remove the existing **hw-module router-stats** configuration and update the ACL entries. Then, configure the **hw-module router-stats** again.

This feature supports only:

- L3 Multicast traffic.
- Default VRFs.

Configure Multicast Route Statistics

Configuring multicast route statistics includes these main tasks:

- Configuring an ACL
- Enabling multicast route statistics for the configured ACLs

```
RP0/0/RP0/CPU0:router# configure

/* Configure an ACL matching the (S,G) routes for which statistics have to be captured:*/
RP0/0/RP0/CPU0:router(config)# ipv4 access-list mcast-counter
RP0/0/RP0/CPU0:router(config-acl)# 10 permit ipv4 host 10.1.1.2 host 224.2.151.1
RP0/0/RP0/CPU0:router(config-acl)# 30 permit ipv4 10.1.1.0/24 232.0.4.0/22
RP0/0/RP0/CPU0:router(config-acl)# 50 permit ipv4 192.168.0.0/24 232.0.4.0/22
RP0/0/RP0/CPU0:router(config-acl)#commit
RP0/0/RP0/CPU0:router(config-acl)#exit

/* Enable multicast route statistics for the configured ACL  on the default VRF. */
RP0/0/RP0/CPU0:router(config)# hw-module route-stats l3mcast vrf default ipv4 egress
mcast-router
RP0/0/RP0/CPU0:router(config)# hw-module route-stats l3mcast vrf default ipv4 ingress
mcast-router
```

Note

- If you are enabling the route stats for a router on the global table, use **vrf default**. If you are enabling the route stats for specific vrf, use the **vrf** *vrfname* option.
- In case, you want to enable route stats for all tables, do not use the **vrf**.

  For example:

  ```
  RP0/0/RP0/CPU0:router(config)#hw-module route-stats l3mcast ipv4  mcast-counter
  ```

Verification

Use the **show mfib route rate** command to verify if the multicast route information is captured for the traffic that matches the ACL:

Note The ingress stats are always per S, G.

```
RP0/0/RP0/CPU0:router# show mfib route rate
Thu Aug 16 18:04:47.312 PDT

IP Multicast Forwarding Rates
(Source Address, Group Address)
     Incoming rate:
           Node: (Incoming node) : pps/bps
     Outgoing rate:
           Node: (Outgoing node) : pps/bps

(10.1.1.2,232.0.0.1)
     Incoming rate :
           Node : 0/0/CPU0 : 4593 / 18153671
     Outgoing rate :
           Node : 0/0/CPU0 : 0 / 0
```

The above output shows that the multicast source **10.1.1.2** is sending packets to multicast group **232.0.0.1** and is received at **4593** pps.

IPv6 Egress Multicast Route Statistics Example

```
RP0/0/RP0/CPU0:router# configure

/* Configure an ACL matching the (S,G) routes for which statistics have to be captured:*/
RP0/0/RP0/CPU0:router(config)# ipv6 access-list 12
RP0/0/RP0/CPU0:router(config-acl)# 10 permit ipv6 any ff33:1:3::1/48
RP0/0/RP0/CPU0:router(config-acl)#commit
RP0/0/RP0/CPU0:router(config-acl)#exit

/* Enable multicast route statistics for the configured ACL in the egress direction on the
 named VRF. */
RP0/0/RP0/CPU0:router(config)# hw-module route-stats l3mcast vrf vrf1 ipv6 egress 12
```

For the information on the interface accounting stats, use the show interface accounting command. The following show command displays interface accounting stats for ingress:

```
Router# show int tenGigE 0/0/0/15 accounting
Mon Nov 12 10:26:20.592 UTC
TenGigE0/0/0/15
  Protocol              Pkts In         Chars In     Pkts Out        Chars Out
  IPV6_MULTICAST    22125711958    1814308380556            0                0
  IPV6_ND                     0                0         1243           128960
```

Cisco IOS XR Release 7.4.1 and later support YANG data model for multicast interface counters.

- Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/protocols/protocol
- Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface[interface-name=TenGigE0/0/0/18]/protocols/protocol

Note The YANG model does not support ingress and egress multicast route stats.

The following show command displays interface accounting stats for egress:

```
Router# show interfaces bundle-ether 100.1001 accounting rates
Mon Aug 26 15:56:41.738 IST
Bundle-Ether100.1001
                                Ingress                        Egress
  Protocol             Bits/sec         Pkts/sec     Bits/sec         Pkts/sec
  IPV4_MULTICAST              0                0     11455000              990
  IPV6_MULTICAST              0                0     11455000              990
  ARP                         0                0            0                0
  IPV6_ND                     0                0            0                0
```

Use Case: Video Streaming

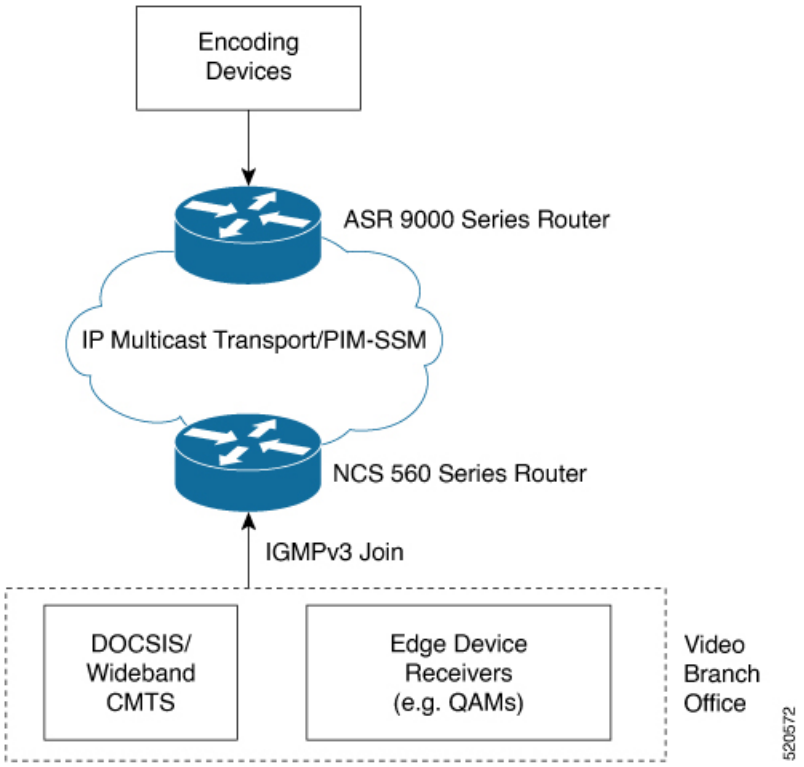
In today's broadcast video networks, proprietary transport systems are used to deliver entire channel line-ups to each video branch office. IP based transport network would be a cost efficient/convenient alternative to deliver video services combined with the delivery of other IP based services. (Internet delivery or business services)

By its very nature, broadcast video is a service well-suited to using IP multicast as a more efficient delivery mechanism to reach end customers.

The IP multicast delivery of broadcast video is explained as follows:

1. Encoding devices in digital primary headends, encode one or more video channels into a Moving Pictures Expert Group (MPEG) stream which is carried in the network via IP multicast.
2. Devices at video branch office are configured by the operator to request the desired multicast content via IGMP joins.
3. The network, using PIM-SSM as its multicast routing protocol, routes the multicast stream from the digital primary headend to edge device receivers located in the video branch office. These edge devices could be edge QAM devices which modulate the MPEG stream for an RF frequency, or CMTS for DOCSIS.

Figure 4: Video Streaming



Multicast Label Distribution Protocol (MLDP) for Core

Multicast Label Distribution Protocol (MLDP) provides extensions to the Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) Label Switched Paths (LSPs) in Multiprotocol Label Switching (MPLS) networks.

MLDP eleminates the use of native multicast PIM to transport multicast packets across the core. In MLDP multicast traffic is label switched across the core. This saves a lot of control plane processing effort.

Characteristics of MLDP Profiles on Core

The following MLDP profiles are supported when the router is configured as a core router:

- Profile 5—Partitioned MDT - MLDP P2MP - BGP-AD - PIM C-mcast Signaling
- Profile 6—VRF MLDP - In-band Signaling
- Profile 7—Global MLDP In-band Signaling
- Profile 12—Default MDT - MLDP - P2MP - BGP-AD - BGP C-mcast Signaling
- Profile 17—Default MDT - MLDP - P2MP - BGP-AD - PIM C-mcast Signaling

Point-to-Multipoint Profiles on Core and Edge Routers

The following profiles are supported when the router is configured as a core router and edge router for p2mp:

- Profile 8—Global P2MP-TE
- Profile 10—VRF Static-P2MP-TE with BGP AD

Label Switched Multicast (LSM) Multicast Label Distribution Protocol (mLDP) based Multicast VPN (mVPN) Support

Label Switch Multicast (LSM) is MPLS technology extensions to support multicast using label encapsulation. Next-generation MVPN is based on Multicast Label Distribution Protocol (mLDP), which can be used to build P2MP and MP2MP LSPs through a MPLS network. These LSPs can be used for transporting both IPv4 and IPv6 multicast packets, either in the global table or VPN context. mLDP is supported on both core and edge routers.

When router is positioned as the core router running mLDP, it only supports the Profiles 5, 6, 7, 12, 14, and 17 irrespective of the profiles supported on the edge router.

When router is positioned as the edge router running mLDP, it only supports the Profiles 6 and 7.

Note IPv6 is not supported for profile 10. Also IPv4 SM is not supported for mLDP profiles on an edge router.

For more information about the characteristics of each of the mLDP Profiles,

Benefits of LSM MLDP based MVPN

LSM provides these benefits when compared to GRE core tunnels that are currently used to transport customer traffic in the core:

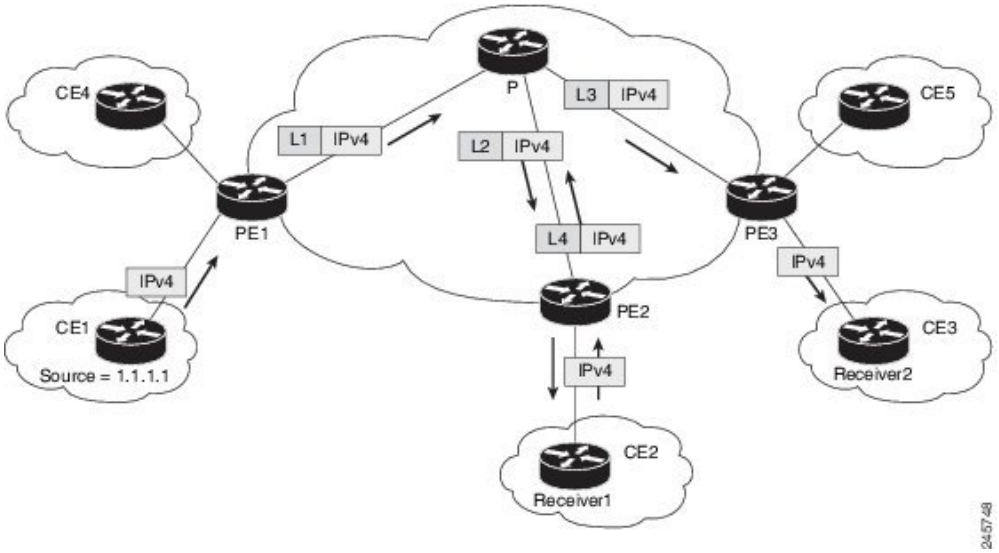
- It leverages the MPLS infrastructure for transporting IP multicast packets, providing a common data plane for unicast and multicast.
- It applies the benefits of MPLS to IP multicast such as Fast ReRoute (FRR) and
- It eliminates the complexity associated PIM.

Configuring MLDP MVPN

The MLDP MVPN configuration enables IPv4 multicast packet delivery using MPLS. This configuration uses MPLS labels to construct default and data Multicast Distribution Trees (MDTs). The MPLS replication is used as a forwarding mechanism in the core and edge network. For MLDP MVPN configuration to work, ensure that the global MPLS MLDP configuration is enabled. To configure MVPN extranet support, configure the source multicast VPN Routing and Forwarding (mVRF) on the receiver Provider Edge (PE) router or configure the receiver mVRF on the source PE. MLDP MVPN is supported for both intranet and extranet.

Note When a Cisco NCS560 Series Routers is positioned as terminal node, it drops the IPv6 traffic that it receives from Cisco ASR 9000 Series Routers which is acting as a head node since EXP NULL label is sent at the Bottom of stack (BOS) over MLDP tunnel for IPV6 traffic.

Figure 5: MLDP based MPLS Network for Core and Edge Routers



Packet Flow in mLDP-based Multicast VPN

For each packet coming in, MPLS creates multiple out-labels. Packets from the source network are replicated along the path to the receiver network. The CE1 router sends out the native IP multicast traffic. The Provider Edge1 (PE1) router imposes a label on the incoming multicast packet and replicates the labeled packet towards the MPLS core network. When the packet reaches the core router (P), the packet is replicated with the appropriate labels for the MP2MP default MDT or the P2MP data MDT and transported to all the egress PEs. Once the packet reaches the egress PE (edge routers), the label is removed and the IP multicast packet is replicated onto the VRF interface. Basically, the packets are encapsulated at headend and decapsulated at tailend on the PE routers.

Realizing a mLDP-based Multicast VPN

There are different ways a Label Switched Path (LSP) built by mLDP can be used depending on the requirement and nature of application such as:

- P2MP LSPs for global table transit Multicast using in-band signaling.
- P2MP/MP2MP LSPs for MVPN based on MI-PMSI or Multidirectional Inclusive Provider Multicast Service Instance (Rosen Draft).
- P2MP/MP2MP LSPs for MVPN based on MS-PMSI or Multidirectional Selective Provider Multicast Service Instance (Partitioned E-LAN).

The router performs the following important functions for the implementation of MLDP:

1. Encapsulating VRF multicast IP packet with Label and replicating to core interfaces (imposition node).
2. Replicating multicast label packets to different interfaces with different labels (Mid node).
3. Decapsulate and replicate label packets into VRF interfaces (Disposition node).

Restrictions for mLDP on Edge Routers

The restrictions applicable for mLDP on edge routers are as follows:

- NETCONF/YANG on MVPN for Profile 6 and Profile 7 is not supported.
- MLDP ping traceroute is not supported.
- IPv6 BVI is not supported.
- Netflow for MPLS-encapsulated multicast packets is not supported.
- MLDP Fast-Reroute is supported for Profile 14.

Multicast MLDP for Edge Router

The following MLDP and P2MP-TE profiles are supported when the router is configured as an edge router:

- Profile 6—VRF MLDP - In-Band Signaling
- Profile 7—Global MLDP In-band Signaling
- Profile 8—Global Static - P2MP-TE
- Profile 10—VRF Static - P2MP TE - BGP-AD
- Profile 14—MLDP Partitioned MDT P2MP with BGP AD and BGP-C Multicast Signaling

Multicast MLDP Profile 14 support on an Edge Router

Table 1: Feature History Table

The MLDP Profile 14 is supported when the router is configured as an edge router.

IP based transport network is a cost efficient and convenient alternative to deliver video services combined with the delivery of other IP based services. To deliver IPTV content MLDP Profile 14 also called as the partitioned MDT, is supported when a router is configured as an edge router.

These are the characteristics of the profile 14:

- Customer traffic is SSM.
- Inter-AS Option A, B and C is supported.
- All PEs must have a unique BGP Route Distinguisher (RD) value.

Configuration Example for mLDP Profile 14 on Edge Routers

```
vrf one
 address-family ipv4 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !

router pim
 vrf one
  address-family ipv4
   rpf topology route-policy rpf-for-one
   mdt c-multicast-routing bgp
   !
   interface GigabitEthernet0/1/0/0
    enable
   !
  !
 !
!

route-policy rpf-for-one
  set core-tree mldp-partitioned-p2mp
end-policy
!

multicast-routing
 vrf one
  address-family ipv4
   mdt source Loopback0
   mdt partitioned mldp ipv4 p2mp
   rate-per-route
   interface all enable
   bgp auto-discovery mldp
   !
   accounting per-prefix
  !
 !
!

mpls ldp
 mldp
  logging notifications
  address-family ipv4
  !
 !
!
```

P2MP-TE and MLDP Scale Number

The following table lists the scaling numbers for MLDP and P2MP-TE:

Feature	Scale
OLE Scale (Egress replication) per P2MP tree	250 (with packet size of 128)
Number of P2MP Trees	100

Feature	Scale
Maximum (S,G) flows	00 (S,G) flows distributed among max 100 VRFs
Maximum OLE scale per system	64 * 250 = 16000
VLAN interfaces in Egress	400
Number of Active ISIS interfaces	400
Number of ISIS Peers	400
Number of iBGP v4 Peers	400
Number of iBGP v6 Peers	400

Configure VRF MLDP In-Band Signaling on Edge Routers

To configure VRF MLDP in-band signaling (Profile 6) on edge routers, you must complete the following tasks:

1. Assign a route policy in PIM to select a reverse-path forwarding (RPF) topology.
2. Configure route policy to set the Multicast Distribution Tree (MDT) type to MLDP inband.
3. Enable MLDP-inband signaling in multicast routing.
4. Enable MPLS for MLDP.

Configuration

/* Assign a route policy in PIM to select a reverse-path forwarding (RPF) topology */

```
RP/0/RP0/CPU0:router(config)#router pim
RP/0/RP0/CPU0:router(config-pim)#vrf one
RP/0/RP0/CPU0:router(config-pim-one)#address-family ipv4
RP/0/RP0/CPU0:router(config-pim-one-ipv4)#rpf topology route-policy rpf-vrf-one
```

/* Configure route policy to set the MDT type to MLDP inband */

```
RP/0/RP0/CPU0:router(config)#route-policy rpf-vrf-one
RP/0/RP0/CPU0:router(config-rpl)#set core-tree mldp-inband
RP/0/RP0/CPU0:router(config-rpl)#end-policy
```

/* Enable MLDP-inband signaling in multicast routing */

```
RP/0/RP0/CPU0:router(config)#multicast-routing
RP/0/RP0/CPU0:router(config-mcast)#vrf one
RP/0/RP0/CPU0:router(config-mcast-one)#address-family ipv4
RP/0/RP0/CPU0:router(config-mcast-one-ipv4)#mdt source loopback 0
RP/0/RP0/CPU0:router(config-mcast-one-ipv4)#mdt mldp in-band-signaling ipv4
RP/0/RP0/CPU0:router(config-mcast-one-ipv4)#interface all enable
```

/* Enable MPLS MLDP */

```
RP/0/RP0/CPU0:router(config)#mpls ldp
RP/0/RP0/CPU0:router(config-ldp)#mldp
```

Configure Global MLDP In-band Signaling on Edge Routers

To configure global MLDP in-band signaling (Profile 7) on edge routers, you must complete the following tasks:

1. Assign a route policy in PIM to select a reverse-path forwarding (RPF) topology.
2. Configure route policy to set the MDT type to MLDP Inband.
3. Enable MLDP inband signaling in multicast routing.
4. Enable MPLS MLDP.

Configuration

/* Assign a route policy in PIM to select a reverse-path forwarding (RPF) topology */

```
RP/0/RP0/CPU0:router(config)#router pim
RP/0/RP0/CPU0:router(config-pim)#address-family ipv4
RP/0/RP0/CPU0:router(config-pim-default-ipv4)#rpf topology route-policy rpf-global
RP/0/RP0/CPU0:router(config-pim-default-ipv4)#interface TenGigE 0/11/0/1
RP/0/RP0/CPU0:router(config-pim-ipv4-if)#enable
```

/* Configure route policy to set the MDT type to MLDP inband */

```
RP/0/RP0/CPU0:router(config)#route-policy rpf-global
RP/0/RP0/CPU0:router(config-rpl)#set core-tree mldp-inband
RP/0/RP0/CPU0:router(config-rpl)#end-policy
```

/* Enable MLDP-inband signaling in multicast routing */

```
RP/0/RP0/CPU0:router(config)#multicast-routing
RP/0/RP0/CPU0:router(config-mcast)#address-family ipv4
RP/0/RP0/CPU0:router(config-mcast-default-ipv4)#interface loopback 0
RP/0/RP0/CPU0:router(config-mcast-default-ipv4-if)#enable
RP/0/RP0/CPU0:router(config-mcast-default-ipv4-if)#exit
RP/0/RP0/CPU0:router(config-mcast-default-ipv4)#mdt source loopback 0
RP/0/RP0/CPU0:router(config-mcast-default-ipv4)#mdt mldp in-band-signaling ipv4
RP/0/RP0/CPU0:router(config-mcast-default-ipv4)#interface all enable
```

/* Enable MPLS MLDP */

```
RP/0/RP0/CPU0:router(config)#mpls ldp
RP/0/RP0/CPU0:router(config-ldp)#mldp
```

Configuration Examples for Inband mLDP Profiles on Edge Routers

Running Configuration for VRF MLDP In-Band Signaling (Profile 6)

```
router pim
 vrf one
  address-family ipv4
   rpf topology route-policy rpf-vrf-one

   route-policy rpf-vrf-one
```

```
     set core-tree mldp-inband
   end-policy

multicast-routing
 vrf one
  address-family ipv4
   mdt source Loopback0
   mdt mldp in-band-signaling ipv4
   interface all enable

mpls ldp
 mldp
```

Running Configuration for Global MLDP In-band Signaling (Profile 7)

```
router pim
  address-family ipv4
   rpf topology route-policy rpf-global
   interface TenGigE0/11/0/1
    enable

route-policy rpf-global
  set core-tree mldp-inband
end-policy

multicast-routing
 address-family ipv4
  interface Loopback0
   enable
  !
  mdt source Loopback0
  mdt mldp in-band-signaling ipv4
  interface all enable
 !
mpls ldp
 mldp
```

Verification of MLDP Configuration on Edge Routers

Use the following commands to verify the MLDP configuration on edge routers.

To check the MLDP neighbors, use the **show mpls mldp neighbor** command.

```
RP/0/RP0/CPU0:Head# show mpls mldp neighbors
mLDP neighbor database
MLDP peer ID      : 2.2.2.2:0, uptime 07:47:59 Up,
  Capabilities     : GR, Typed Wildcard FEC, P2MP, MP2MP
  Target Adj       : No
  Upstream count   : 1
  Branch count     : 1
  LDP GR           : Enabled
                   : Instance: 1
  Label map timer  : never
  Policy filter in :
  Path count       : 1
  Path(s)          : 12.1.1.2          TenGigE0/11/0/1 LDP
  Adj list         : 12.1.1.2          TenGigE0/11/0/1
  Peer addr list   : 2.25.32.2
                   : 2.2.2.2
                   : 11.1.1.1
```

```
                    : 12.1.1.2
                    : 13.10.1.1
```

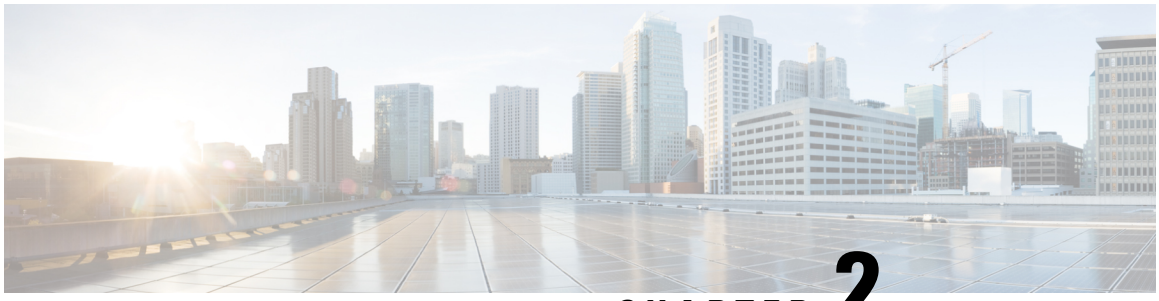
To display the contents of the Label Information Base (LIB), use the **show mpls mldp bindings** command.

```
RP/0/RP0/CPU0:Head#show mpls mldp bindings
mLDP MPLS Bindings database

LSP-ID: 0x00001 Paths: 7 Flags:
0x00001 P2MP   5.5.5.5 [vpnv6 1:1 2015:1:1::3 ff3e::1]
   Local Label: 70009
   Remote Label: 64018 NH: 12.1.1.2 Inft: TenGigE0/11/0/1
   Remote Label: 64022 NH: 50.1.1.1 Inft: TenGigE0/11/0/1
   Remote Label: 30002 NH: 30.10.1.2 Inft: Bundle-Ether56
   Remote Label: 64023 NH: 60.1.1.2 Inft: HundredGigE0/0/1/1
   Remote Label: 64024 NH: 70.1.1.1 Inft: TenGigE0/11/0/2
   Remote Label: 64022 NH: 40.1.1.1 Inft: TenGigE0/11/0/3
```

To display the MLDP event traces, use the **show mpls mldp trace** command.

```
RP/0/RP0/CPU0:Head#show mpls mldp trace
3535 wrapping entries (631040 possible, 35584 allocated, 0 filtered, 3535 total)
May 30 23:30:21.121 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Trace pre-init iox success
May 30 23:30:21.121 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Debug pre-init iox success
May 30 23:30:21.121 MLDP GLO 0/RP0/CPU0 t6746 GEN  : API pre-init iox success
May 30 23:30:21.121 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Bitfield pre-init iox success
May 31 12:08:39.465 MLDP GLO 0/RP0/CPU0 t6746 GEN  : mldp_evm 0x563de8f01698 allocated
May 31 12:08:39.465 MLDP GLO 0/RP0/CPU0 t6746 GEN  : EVM init iox success
May 31 12:08:39.472 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Registered EDM on active success
May 31 12:08:39.472 MLDP GLO 0/RP0/CPU0 t6746 GEN  : EDM Ac/St init iox again
May 31 12:08:39.472 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Registered EDM Location on active
success
May 31 12:08:39.472 MLDP GLO 0/RP0/CPU0 t6746 GEN  : EDM Loc init iox success
May 31 12:08:39.475 MLDP GLO 0/RP0/CPU0 t6746 GEN  : LMRIB init iox success
May 31 12:08:39.475 MLDP GLO 0/RP0/CPU0 t18944 MRIB : MRIB connection established
May 31 12:08:39.475 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Interface manager init iox success
May 31 12:08:39.475 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Async init iox success
May 31 12:08:39.475 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Boolean init iox success
May 31 12:08:39.475 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Timers init iox success
May 31 12:08:39.479 MLDP GLO 0/RP0/CPU0 t6746 GEN  : RUMP init iox success
May 31 12:08:39.479 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Chunks init iox success
May 31 12:08:39.509 MLDP ERR 0/RP0/CPU0 t6746 RIB  : RIB not ready
May 31 12:08:39.509 MLDP ERR 0/RP0/CPU0 t6746 RIB  : RIB not ready
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 GEN  : mldp_ens_event_ctx_chunk is NULL
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 GEN  : Context Table init iox success
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 GEN  : mldp_rib_main_evm 0x563de8fd23e8
allocated
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 GEN  : RIB Thread EVM init rib success
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 GEN  : RIB Thread Chunk init rib success
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 GEN  : RIB Thread queue init rib success
May 31 12:08:39.512 MLDP GLO 0/RP0/CPU0 t6746 RIB  : Bound to RIB, fd: 354
```



CHAPTER 2

Implementing Layer-2 Multicast with IGMP Snooping

Internet Group Management Protocol (IGMP) snooping restricts multicast flows at Layer 2 to only those segments with at least one interested receiver. This module describes how to implement IGMP snooping.

Prerequisites for IGMP Snooping

Before implementing IGMP snooping, make sure that the network is configured with a Layer 2 VPN (L2VPN).

Restrictions for IGMP Snooping

- IGMP snooping is supported only under L2VPN bridge domains.
- Explicit host tracking (an IGMPv3 snooping feature) is not supported.
- IPv6 Multicast Listener Discovery (MLD) snooping is not supported.
- IGMPv1 is not supported.
- IGMP snooping with VPLS on bridge domain is not supported.
- IGMP snooping over Access Pseudo-wire is not supported.

Note IPv4 multicast is supported for a multicast source that is behind the BVI interface. For example, the below configuration shows how to configure source behind BVI for IPv4 multicast:

```
l2vpn
bridge group 1
  bridge-domain 1
   multicast-source ipv4
   igmp snooping profile grp1
   !
   interface TenGigE0/0/0/3.32
   !
   routed interface BVI1
```

IGMP snooping for bridge domains without Bridged Virtual Interface (BVI) is supported with the following design consideration: You must configure the multicast-source ipv4 command in the source switch where bridge domain and IGMP snooping are enabled.

IGMP Snooping Overview

Description of Basic Functions

IGMP snooping provides a way to constrain multicast traffic at Layer 2. By snooping the IGMP membership reports sent by hosts in the bridge domain, the IGMP snooping application can set up Layer 2 multicast forwarding tables to deliver traffic only to ports with at least one interested member, significantly reducing the volume of multicast traffic.

Configured at Layer 3, IGMP provides a means for hosts in an IPv4 multicast network to indicate which multicast traffic they are interested in and for routers to control and limit the flow of multicast traffic in the network at Layer 3.

IGMP snooping uses the information in IGMP membership report messages to build corresponding information in the forwarding tables to restrict IP multicast traffic at Layer 2. The forwarding table entries are in the form <Route, OIF List>, where:

- Route is a <*, G> route or <S, G> route, where * is any source, G is group and S is the source.
- OIF List comprises all bridge ports that have sent IGMP membership reports for the specified route.

Implemented in a multicast network, IGMP snooping has the following attributes:

- In its basic form, it reduces bandwidth consumption by reducing multicast traffic that would otherwise flood an entire bridge domain.
- With the use of some optional configurations, it provides security between bridge domains by filtering the IGMP reports received from hosts on one bridge port and preventing leakage towards the hosts on other bridge ports.

High Availability Features

All high availability features apply to the IGMP snooping processes with no additional configuration beyond enabling IGMP snooping. The following high availability features are supported:

- Process restarts
- RP Failover
- Stateful Switch-Over (SSO)
- Non-Stop Forwarding (NSF)—Forwarding continues unaffected while the control plane is restored following a process restart or route processor (RP) failover.
- Line card online insertion and removal (OIR)

Bridge Domain Support

IGMP snooping operates at the bridge domain level. When IGMP snooping is enabled on a bridge domain, the snooping functionality applies to all ports under the bridge domain, including:

- Physical ports under the bridge domain.
- Ethernet flow points (EFPs)—An EFP can be a VLAN.
- Ethernet bundles—Ethernet bundles include IEEE 802.3ad link bundles and Cisco EtherChannel bundles. From the perspective of the IGMP snooping application, an Ethernet bundle is just another EFP. The forwarding application in the randomly nominates a single port from the bundle to carry the multicast traffic.

IGMP snooping for bridge domains without Bridged Virtual Interface (BVI) is supported with the following design consideration:

You must configure the **multicast-source ipv4** command in the source switch where IGMP snooping is enabled as seen in the following example:

```
l2vpn
bridge group 1
bridge-domain 1
multicast-source ipv4
igmp snooping profile grp1
!
interface TenGigE0/0/0/3.31 //Source
!
interface TenGigE0/0/0/3.32
!
routed interface BVI1
```

Multicast Host Ports

IGMP snooping classifies each port (for example, EFPs, physical ports, or EFP bundles) as a host ports, that is, any port that is not an mrouter port is a host port.

Multicast Traffic Handling within a Bridge Domain with IGMP Snooping Enabled

The following tables describe traffic handling behaviors by IGMP snooping and host ports.

By default, IGMP snooping supports IGMPv2 and IGMPv3. The version of the IGMP querier discovered in the bridge domain determines the operational version of the snooping processes. If you change the default, configuring IGMP snooping to support a minimum version of IGMPv3, IGMP snooping ignores any IGMPv2 queriers.

Table 2: Multicast Traffic Handling for an IGMPv2 Querier

Traffic Type	Received on Host Ports
IP multicast source traffic	Forwards to all host ports that indicate interest.
IGMP general queries	Forwarded to all the ports that are part of the bridge domain
IGMP group-specific queries	Dropped
IGMPv2 joins	Examines (snoops) the reports. • If report suppression is enabled, forwards first join for a new group or first join following a general query for an existing group. • If report suppression is disabled, forwards on all mrouter ports.
IGMPv3 reports	Ignores
IGMPv2 leaves	Invokes last member query processing.

Table 3: Multicast Traffic Handling for an IGMPv3 Querier

Traffic Type	Received on Host Ports
IP multicast source traffic	Forwards to all mrouter ports and to host ports that indicate interest.
IGMP general queries	Forwarded to all the ports that are part of the bridge domain
IGMP group-specific queries	Forwarded to all the ports that are part of the bridge domain
IGMPv2 joins	Handles as IGMPv3 IS_EX{} reports.
IGMPv3 reports	• If proxy reporting is enabled—For state changes or source-list changes, generates a state change report on all mrouter ports. • If proxy reporting is disabled—Forwards on all mrouter ports.
IGMPv2 leaves	Handles as IGMPv3 IS_IN{} reports.

Information About IGMP Snooping Configuration Profiles

To enable IGMP snooping on a bridge domain, you must attach a profile to the bridge domain. The minimum configuration is an empty profile. An empty profile enables the default configuration options and settings for IGMP snooping, as listed in the Default IGMP Snooping Configuration Settings, on page 44.

You can attach IGMP snooping profiles to bridge domains or to ports under a bridge domain. The following guidelines explain the relationships between profiles attached to ports and bridge domains:

- Any IGMP Snooping profile attached to a bridge domain, even an empty profile, enables IGMP snooping. To disable IGMP snooping, detach the profile from the bridge domain.
- An empty profile configures IGMP snooping on the bridge domain and all ports under the bridge using default configuration settings.
- A bridge domain can have only one IGMP snooping profile attached to it (at the bridge domain level) at any time. Profiles can be attached to ports under the bridge, one profile per port.
- Port profiles are not in effect if the bridge domain does not have a profile attached to it.
- IGMP snooping must be enabled on the bridge domain for any port-specific configurations to be in effect.
- If a profile attached to a bridge domain contains port-specific configuration options, the values apply to all of the ports under the bridge, including all mrouter and host ports, unless another port-specific profile is attached to a port.
- When a profile is attached to a port, IGMP snooping reconfigures that port, disregarding any port configurations that may exist in the bridge-level profile.

Creating Profiles

To create a profile, use the **igmp snooping profile** command in global configuration mode.

Attaching and Detaching Profiles

To attach a profile to a bridge domain, use the **igmp snooping profile** command in l2vpn bridge group bridge domain configuration mode. To attach a profile to a port, use the **igmp snooping profile** command in the interface configuration mode under the bridge domain. To detach a profile, use the **no** form of the command in the appropriate configuration mode.

When you detach a profile from a bridge domain or a port, the profile still exists and is available for use at a later time. Detaching a profile has the following results:

- If you detach a profile from a bridge domain, IGMP snooping is deactivated in the bridge domain.
- If you detach a profile from a port, IGMP snooping configuration values for the port are instantiated from the bridge domain profile.

Changing Profiles

You cannot make changes to an active profile. An active profile is one that is currently attached.

- If the active profile is configured under the bridge, you must detach it from the bridge, and reattach it.
- If the active profile is configured under a specific bridge port, you must detach it from the bridge port, and reattach it.

Another way to do this is to create a new profile incorporating the desired changes and attach it to the bridges or ports, replacing the existing profile. This deactivates IGMP snooping and then reactivates it with parameters from the new profile.

Default IGMP Snooping Configuration Settings

Table 4: IGMP Snooping Default Configuration Values

Scope	Feature	Default Value
Bridge Domain	IGMP snooping	Disabled on a bridge domain until an enabling IGMP snooping profile is attached to the bridge domain.
	internal querier	By default Internal Querier is disabled. To enable Internal Querier, add it to the IGMP snooping profile. Internal Querier is not recommended, when BVI and IGMP snooping is configured under a bridge.
	last-member-query-count	2
	last-member-query-interval	1000 (milliseconds)
	minimum-version	2 (supporting IGMPv2 and IGMPv3)
	querier query-interval	60 (seconds) **Note** This is a nonstandard default value.
	report-suppression	Enabled (enables report suppression for IGMPv2 and proxy-reporting for IGMPv3)
	querier robustness-variable	2
	router alert check	Enabled
	tcn query solicit	Disabled
	tcn flood	Enabled
	ttl-check	Enabled
	unsolicited-report-timer	1000 (milliseconds)

Scope	Feature	Default Value
Port	immediate-leave	Disabled
	mrouter	No static mrouters configured; dynamic discovery occurs by default.
	router guard	Disabled
	static group	None configured

IGMP Snooping Configuration at the Bridge Domain Level

IGMP Minimum Version

The **minimum-version** command determines which IGMP versions are supported by IGMP snooping in the bridge domain:

- When minimum-version is 2, IGMP snooping intercepts IGMPv2 and IGMPv3 messages. This is the default value.
- When minimum-version is 3, IGMP snooping intercepts only IGMPv3 messages and drops all IGMPv2 messages.

IGMPv1 is not supported. The scope for this command is the bridge domain. The command is ignored in a profile attached to a port.

Group Membership Interval, Robustness Variable, and Query Interval

The group membership interval (GMI) controls when IGMP snooping expires stale group membership states. The **show igmp snooping group** command shows groups with an expiry time of 0 until that stale state is cleaned up following the next query interval.

The GMI is calculated as:

GMI = (robustness-variable * query-interval) + maximum-response-time

where:

- maximum-response-time (MRT) is the amount of time during which receivers are required to report their membership state.
- robustness-variable is an integer used to influence the calculated GMI.
- query-interval is the amount of time between general queries.

Values for the components in the GMI are obtained as follows:

- MRT is advertised in the general query, for both IGMPv2 and IGMPv3.
- If the querier is running IGMPv2, IGMP snooping uses the IGMP-snooping-configured values for the robustness-variable and query-interval. These parameter values must match the configured values for the querier. In most cases, if you are interacting with other Cisco routers, you should not need to explicitly configure these values—the default values for IGMP snooping should match the default values of the

querier. If they do not, use the **querier robustness-variable** and **querier query-interval** commands to configure matching values.

- IGMPv3 general queries convey values for robustness-variable and query-interval (QRV and QQI, respectively). IGMP snooping uses the values from the query, making the IGMP snooping GMI exactly match that of the querier.

Multicast over Integrated Routing Bridging Active/Active Multihome

Multicast over integrated routing bridging active/active multihome feature enables the routers to quickly and safely switch traffic between routers, during failure, without any traffic loss. This feature comprises of the following four sub features that work together as a solution:

- First, IGMPv2 snooping is enabled for the peer routers to know which Layer 2 interface has receiver interested in a particular group.
- After snooping, this information is synced to the peer routers with the Layer 2 EVPN sync feature.
- After both peer routers are synced, they act like a last hop router and send PIM join upstream.
- Once the traffic arrives on both the peer routers, only one peer router forwards the traffic to the receiver with the designated forwarder election feature.

How to Configure IGMP Snooping

The first two tasks are required to configure basic IGMP snooping configuration.

Creating an IGMP Snooping Profile

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**igmp snooping profile** *profile-name* **Example:** `RP/0/RP0/CPU0:router(config)# igmp snooping profile default-bd-profile`	Enters IGMP snooping profile configuration mode and creates a named profile. The default profile enables IGMP snooping. You can commit the new profile without any additional configurations, or you can include additional configuration options to the profile. You can also return to the profile later to add configurations, as described in other tasks in this module.
Step 3	Optionally, add commands to override default configuration values.	If you are creating a bridge domain profile, consider the following:

	Command or Action	Purpose
		• An empty profile is appropriate for attaching to a bridge domain. An empty profile enables IGMP snooping with default configuration values. • You can optionally add more commands to the profile to override default configuration values. • If you include port-specific configurations in a bridge domain profile, the configurations apply to all ports under the bridge, unless another profile is attached to a port. If you are creating a port-specific profile, consider the following: • While an empty profile could be attached to a port, it would have no effect on the port configuration. • When you attach a profile to a port, IGMP snooping reconfigures that port, overriding any inheritance of configuration values from the bridge-domain profile. You must repeat the commands in the port profile if you want to retain those configurations. You can detach a profile, change it, and reattach it to add commands to a profile at a later time.
Step 4	**commit**	

Where to Go Next

You must attach a profile to a bridge domain or to a port to have it take effect. See one of the following tasks:

Attaching a Profile and Activating IGMP Snooping on a Bridge Domain

To activate IGMP snooping on a bridge domain, attach an IGMP snooping profile to the bridge domain, as described in the following steps.

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**l2vpn** **Example:**	Enters Layer 2 VPN configuration mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config)# l2vpn`	
Step 3	**bridge group** *bridge-group-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn)# bridge group GRP1`	Enters Layer 2 VPN bridge group configuration mode for the named bridge group.
Step 4	**bridge-domain** *bridge-domain-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain ISP1`	Enters Layer 2 VPN bridge group bridge domain configuration mode for the named bridge domain.
Step 5	**multicast-source ipv4** **Example:** `RP/0/RP0/CPU0:router(config)# multicast-source ipv4`	Configures Layer 2 multicast routes with IGMP snooping.
Step 6	**igmp snooping profile** *profile-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# igmp snooping profile default-bd-profile`	Attaches the named IGMP snooping profile to the bridge domain, enabling IGMP snooping on the bridge domain.
Step 7	**commit**	
Step 8	**show igmp snooping bridge-domain detail** **Example:** `RP/0/RP0/CPU0:router# show igmp snooping bridge-domain detail`	(Optional) Verifies that IGMP snooping is enabled on a bridge domain and shows the IGMP snooping profile names attached to bridge domains and ports.
Step 9	**show l2vpn bridge-domain detail** **Example:** `RP/0/RP0/CPU0:router# show l2vpn bridge-domain`	(Optional) Verifies that IGMP snooping is implemented in the forwarding plane (Layer 2) on a bridge domain.

Detaching a Profile and Deactivating IGMP Snooping on a Bridge Domain

To deactivate IGMP snooping on a bridge domain, remove the profile from the bridge domain using the following steps.

Note A bridge domain can have only one profile attached to it at a time.

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**l2vpn** **Example:** `RP/0/RP0/CPU0:router(config)# l2vpn`	Enters Layer 2 VPN configuration mode.
Step 3	**bridge group** *bridge-group-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn)# bridge group GRP1`	Enters Layer 2 VPN bridge group configuration mode for the named bridge group.
Step 4	**bridge-domain** *bridge-domain-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain ISP1`	Enters Layer 2 VPN bridge group bridge domain configuration mode for the named bridge domain.
Step 5	**no igmp snooping disable** **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# no igmp snooping disable`	Detaches the IGMP snooping profile from the bridge domain, disabling IGMP snooping on that bridge domain. **Note** Only one profile can be attached to a bridge domain at a time. If a profile is attached, IGMP snooping is enabled. If a profile is not attached, IGMP snooping is disabled.
Step 6	**commit**	
Step 7	**show igmp snooping bridge-domain detail** **Example:** `RP/0/RP0/CPU0:router# show igmp snooping`	(Optional) Verifies that IGMP snooping is disabled on a bridge domain.

	Command or Action	Purpose
	`bridge-domain detail`	
Step 8	**show l2vpn bridge-domain detail** **Example:** `RP/0/RP0/CPU0:router# show l2vpn bridge-domain`	(Optional) Verifies that IGMP snooping is disabled in the forwarding plane (Layer 2) on a bridge domain.

Attaching and Detaching Profiles to Ports Under a Bridge

Before you begin

IGMP snooping must be enabled on the bridge domain for port-specific profiles to affect IGMP snooping behavior.

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**l2vpn** **Example:** `RP/0/RP0/CPU0:router(config)# l2vpn`	Enters Layer 2 VPN configuration mode.
Step 3	**bridge group** *bridge-group-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn)# bridge group GRP1`	Enters Layer 2 VPN bridge group configuration mode for the named bridge group.
Step 4	**bridge-domain** *bridge-domain-name* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain ISP1`	Enters Layer 2 VPN bridge group bridge domain configuration mode for the named bridge domain.
Step 5	**interface** *interface-type interface-number* **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# interface gig 1/1/1/1`	Enters Layer 2 VPN bridge group bridge domain interface configuration mode for the named interface or PW.

	Command or Action	Purpose
Step 6	**multicast-source ipv4** **Example:** `RP/0/RP0/CPU0:router(config)# multicast-source ipv4`	Configures L2 multicast routes in L2 multicast with IGMP Snooping.
Step 7	Do one of the following: • **igmp snooping profile** *profile-name* • **no igmp snooping** **Example:** `RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-if)# igmp snooping profile mrouter-port-profile`	Attaches the named IGMP snooping profile to the port. **Note** A profile on a port has no effect unless there is also a profile attached to the bridge. The **no** form of the command detaches a profile from the port. Only one profile can be attached to a port.
Step 8	**commit**	
Step 9	**show igmp snooping bridge-domain detail** **Example:** `RP/0/RP0/CPU0:router# show igmp snooping bridge-domain detail`	(Optional) Verifies that IGMP snooping is enabled on a bridge domain and shows the IGMP snooping profile names attached to bridge domains and ports.
Step 10	**show l2vpn bridge-domain detail** **Example:** `RP/0/RP0/CPU0:router# show l2vpn bridge-domain`	(Optional) Verifies that IGMP snooping is implemented in the forwarding plane (Layer 2) on a bridge domain.

Verifying Multicast Forwarding

Procedure

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**show l2vpn forwarding bridge-domain** [*bridge-group-name*:*bridge-domain-name*] **mroute ipv4** [**group** *group_IPaddress*] [**hardware** {**ingress** | **egress**}] [**detail**]**location** *node-id* **Example:** `RP/0/RP0/CPU0:routershow l2vpn`	Displays multicast routes as they are converted into the forwarding plane forwarding tables. Use optional arguments to limit the display to specific bridge groups or bridge domains. If these routes are not as expected, check the control plane configuration and correct the corresponding IGMP snooping profiles.

	Command or Action	Purpose
	`forwarding bridge-domain bg1:bd1 mroute ipv4 group 234.192.4.1 hardware ingress detail location 0/1/cPU0`	
Step 3	**show l2vpn forwarding bridge-domain** [*bridge-group-name*:*bridge-domain-name*] **mroute ipv4 summary location** *node-id* **Example:** `RP/0/RP0/CPU0:router# show l2vpn forwarding bridge-domain bg1:bd1 mroute ipv4 summary location 0/3/CPU0`	Displays summary-level information about multicast routes as stored in the forwarding plane forwarding tables. Use optional arguments to limit the display to specific bridge domains.

Configuration Examples for IGMP Snooping

The following examples show how to enable IGMP snooping on Layer 2 bridge domains on :

Configuring IGMP Snooping on Physical Interfaces Under a Bridge: Example

1. Create two profiles.

```
igmp snooping profile profile1
!
igmp snooping profile profile2

!
```

2. Configure two physical interfaces for L2 transport.

```
interface GigabitEthernet0/8/0/38
   negotiation auto
   l2transport
   no shut
   !
!
interface GigabitEthernet0/8/0/39
   negotiation auto
   l2transport
   no shut
   !
!
```

3. Add interfaces to the bridge domain. Attach bridge_profile to the bridge domain and port_profile to one of the Ethernet interfaces. The second Ethernet interface inherits IGMP snooping configuration attributes from the bridge domain profile.

```
l2vpn
  bridge group bg1
     bridge-domain bd1
     igmp snooping profile profile1
     interface GigabitEthernet0/8/0/38
```

```
      igmp snooping profile profile2
     interface GigabitEthernet0/8/0/39

!
  !
!
```

4. Verify the configured bridge ports.

```
show igmp snooping port
```

Configuring IGMP Snooping on VLAN Interfaces Under a Bridge: Example

1. Configure two profiles.

```
multicast-source ipv4
igmp snooping profile profile1
multicast-source ipv4
igmp snooping profile profile2
!
```

2. Configure VLAN interfaces for L2 transport.

```
interface GigabitEthernet0/8/0/8
   negotiation auto
   no shut
   !
!
interface GigabitEthernet0/8/0/8.1 l2transport
   encapsulation dot1q 1001
   mtu 1514
   !
!
interface GigabitEthernet0/8/0/8.2 l2transport
   encapsulation dot1q 1002
   mtu 1514
   !
!
```

3. Attach a profile and add interfaces to the bridge domain. Attach a profile to one of the interfaces. The other interface inherits IGMP snooping configuration attributes from the bridge domain profile.

```
l2vpn
  bridge group bg1
     bridge-domain bd1
     multicast-source ipv4
     igmp snooping profile profile1
     interface GigabitEthernet0/8/0/8.1
       multicast-source ipv4
       igmp snooping profile profile2
     interface GigabitEthernet0/8/0/8.2

     !
  !
```

```
!
```

4. Verify the configured bridge ports.

```
show igmp snooping port
```

Configuring IGMP Snooping on Ethernet Bundles Under a Bridge: Example

1. Configure two IGMP snooping profiles.

```
multicast-source ipv4
       igmp snooping profile profile1
       !
       multicast-source ipv4
       igmp snooping profile profile2
```

2. Configure interfaces as bundle member links.

```
       interface GigabitEthernet0/0/0/0
         bundle id 1 mode on
         negotiation auto
       !
       interface GigabitEthernet0/0/0/1
         bundle id 1 mode on
         negotiation auto
       !
       interface GigabitEthernet0/0/0/2
         bundle id 2 mode on
         negotiation auto
       !
       interface GigabitEthernet0/0/0/3
         bundle id 2 mode on
         negotiation auto
       !
```

3. Configure the bundle interfaces for L2 transport.

```
interface Bundle-Ether 1
          l2transport
          !
       !
       interface Bundle-Ether 2
          l2transport
          !
       !
```

4. Add the interfaces to the bridge domain and attach IGMP snooping profiles.

```
       l2vpn
         bridge group bg1
            bridge-domain bd1
            multicast-source ipv4
            igmp snooping profile profile1
            interface bundle-Ether 1
              multicast-source ipv4
```

```
    igmp snooping profile profile2
   interface bundle-Ether 2

   !
  !
 !
```

5. Verify the configured bridge ports.

```
show igmp snooping port
```

Configuring Multicast over Integrated Routing Bridging Active/Active Multihome

Configurations performed on peer 1:

1. Layer 2 Base Configuration

```
hostname peer1
!
interface Bundle-Ether2
!
interface Bundle-Ether2.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface TenGigE0/0/0/0
 bundle id 2 mode on
 no shut
!
```

2. EVPN Configuration

```
hostname peer1
!
router bgp 100
 bgp router-id 1.1.1.1
 bgp graceful-restart
 address-family l2vpn evpn
 !
 neighbor 3.3.3.3
  remote-as 100
  update-source Loopback0
  address-family l2vpn evpn
  !
 !
!
evpn
 evi 2
  advertise-mac
  !
 !
 interface Bundle-Ether2
  ethernet-segment
   identifier type 0 02.02.02.02.02.02.02.02.02
   bgp route-target 0002.0002.0002
  !
 !
!
```

3. IGMPv2 Snoop Configurations

```
hostname peer1
!
router igmp

  version 2
 !
!
l2vpn
 bridge group VLAN2
  bridge-domain VLAN2
   multicast-source ipv4
   igmp snooping profile 1
   interface Bundle-Ether2.2
   !

   evi 2
   !
  !
 !
multicast-source ipv4
igmp snooping profile 1
!
```

Configurations Performed on Peer 2:

1. Layer 2 Base Configuration

```
hostname peer2
!
interface Bundle-Ether2
!
interface Bundle-Ether2.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface TenGigE0/0/0/0
 bundle id 2 mode on
 no shut
!
```

2. EVPN Configuration

```
hostname peer2
!
router bgp 100
 bgp router-id 2.2.2.2
 bgp graceful-restart
 address-family l2vpn evpn
 !
 neighbor 3.3.3.3
  remote-as 100
  update-source Loopback0
  address-family l2vpn evpn
  !
 !
!
evpn
 evi 2
  advertise-mac
  !
 !
 interface Bundle-Ether2
  ethernet-segment
   identifier type 0 02.02.02.02.02.02.02.02.02
```

```
   bgp route-target 0002.0002.0002
  !
 !
!
```

3. IGMPv2 Snoop Configurations

```
hostname peer2
!
router igmp

  version 2
 !
!
l2vpn
 bridge group VLAN2
  bridge-domain VLAN2
   multicast-source ipv4
   igmp snooping profile 1
   interface Bundle-Ether2.2
   !

   evi 2
   !
  !
 !
multicast-source ipv4
igmp snooping profile 1
!
```

Verifying IGMP Snooping and EVPN Sync

In this example, the receiver sends an IGMPv2 join for the group 239.0.0.2. On Peer2, this group has a D Flag, that means the actual IGMP joined peer2, but not peer1. On Peer1, this group has a B flag, that means this group is learnt from BGP with the EVPN sync feature.

```
RP/0/RP0/CPU0:peer1#show igmp snooping group
Fri Aug 31 22:27:46.363 UTC

Key: GM=Group Filter Mode, PM=Port Filter Mode
Flags Key: S=Static, D=Dynamic, B=BGP Learnt, E=Explicit Tracking, R=Replicated

                           Bridge Domain VLAN10:VLAN10

Group           Ver GM Source          PM Port                       Exp   Flgs
-----           --- -- ------          -- ----                       ---   ----
239.0.0.2       V2  -  *               -  BE2.2                      never B


RP/0/RP0/CPU0:peer2#show igmp snooping group
Fri Aug 31 22:27:49.686 UTC

Key: GM=Group Filter Mode, PM=Port Filter Mode
Flags Key: S=Static, D=Dynamic, B=BGP Learnt, E=Explicit Tracking, R=Replicated

                           Bridge Domain VLAN10:VLAN10

Group           Ver GM Source          PM Port                       Exp   Flgs
-----           --- -- ------          -- ----                       ---   ----
239.0.0.2       V2  -  *               -  BE2.2                      74    D
```

Verifying Dual DR PIM Uplink

In this example, when the source 126.0.0.100 sends traffic to group 239.0.0.2, you see both Peer1 and Peer2 are sending PIM join upstream. The incoming interface for (*,G) and (S,G) should be the interface toward the RP and source respectively. For both Peer1 and Peer2, the outgoing interface should be the BVI interface facing the receiver.

```
RP/0/RP0/CPU0:peer1#show mrib route
:
:

(*,239.0.0.2) RPF nbr: 30.0.0.4 Flags: C RPF
  Up: 00:13:41
  Incoming Interface List
    HundredGigE0/0/0/1 Flags: A NS, Up: 00:13:41
  Outgoing Interface List
    BVI2 Flags: F NS LI, Up: 00:13:41

(126.0.0.100,239.0.0.2) RPF nbr: 30.0.0.4 Flags: RPF
  Up: 00:03:34
  Incoming Interface List
    HundredGigE0/0/0/1 Flags: A, Up: 00:03:34
  Outgoing Interface List
    BVI2 Flags: F NS, Up: 00:03:34
:
:

RP/0/RP0/CPU0:peer2#show mrib route
:
:

(*,239.0.0.2) RPF nbr: 50.0.0.4 Flags: C RPF
  Up: 00:13:33
  Incoming Interface List
    HundredGigE0/0/0/2 Flags: A NS, Up: 00:13:33
  Outgoing Interface List
    BVI2 Flags: F NS LI, Up: 00:13:33

(126.0.0.100,239.0.0.2) RPF nbr: 50.0.0.4 Flags: RPF
  Up: 00:03:24
  Incoming Interface List
    HundredGigE0/0/0/2 Flags: A, Up: 00:03:24
  Outgoing Interface List
    BVI2 Flags: F NS, Up: 00:03:24
:
:
```

Verifying Designated Forwarder Election

As described in the previous example, both peer1 and peer2 have BVI2 as outgoing interface. However, only one of the peer should forward the traffic. Designated forwarder election elects one of them to do the forwarding. In this example, peer2 is selected as the forwarder. Peer1 has Bundle-Ether2.2 marked as NDF.

```
RP/0/RP0/CPU0:peer1#show l2vpn forwarding bridge-domain VLAN2:VLAN2 mroute ipv4 hardware
ingress detail location 0/0/cPU0
Bridge-Domain: VLAN2:VLAN2, ID: 0
:
:

Bridge-Domain: VLAN2:VLAN2, ID: 0
```

```
  Prefix: (0.0.0.0,239.0.0.2/32)
  P2MP enabled: N
  IRB platform data: {0x0, 0x2d, 0x0, 0x0}, len: 32
  Bridge Port:
  EVPN, Xconnect id: 0x80000001 NH:2.2.2.2
  Bundle-Ether2.2, Xconnect id: 0xa0000015 (NDF)

RP/0/RP0/CPU0:peer2#show l2vpn forwarding bridge-domain VLAN2:VLAN2 mroute ipv4 hardware
ingress detail location 0/0/cPU0
:
:

Bridge-Domain: VLAN2:VLAN2, ID: 0
  Prefix: (0.0.0.0,239.0.0.2/32)
  P2MP enabled: N
  IRB platform data: {0x0, 0x30, 0x0, 0x0}, len: 32
  Bridge Port:
  EVPN, Xconnect id: 0x80000001 NH:1.1.1.1
  Bundle-Ether2.2, Xconnect id: 0xa0000029
```

Additional References

Related Documents

Related Topic	Document Title
Configuring MPLS VPLS bridges	Implementing Virtual Private LAN Services on Cisco IOS XR Software module in the *MPLS Configuration Guide*
Getting started information	
Configuring EFPs and EFP bundles	*Interface and Hardware Component Configuration Guide for Cisco NCS 560 Series Routers*

Standards

Standards[1]	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

[1] Not all supported standards are listed.

MIBs

MIBs	MIBs Link
No MIBs support IGMP snooping.	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC-4541	Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport