

Implementing Layer 2 Multicast

- Implementing IGMP Snooping, on page 1
- Prerequisites for IGMP Snooping, on page 2
- Supported Features and Restrictions for IGMP Snooping, on page 2
- Information About IGMP Snooping, on page 3
- EVPN All-Active Multi-homed Multicast Source Behind a BVI, on page 10
- How to Configure IGMP Snooping, on page 15
- Configuration Examples for IGMP Snooping, on page 21
- Additional References, on page 28
- MLD Snooping , on page 29
- Creating a MLD Snooping Profile, on page 37
- Deactivating MLD Snooping on a Bridge Domain, on page 38
- Configuring Static Mrouter Ports (MLD), on page 38
- Configuring Router Guard (MLD), on page 39
- Configuring Immediate-leave for MLD, on page 40
- Configuring Internal Querier for MLD, on page 41
- Configuring Static Groups for MLD, on page 42
- Configuring MLD Snooping, on page 43
- Configuring MLD Snooping on Ethernet Bundles, on page 45
- MLD Snooping Synchronization for EVPN Multi-Homing, on page 47
- Configure MLD Snooping Synchronization for EVPN Multi-Homing, on page 50
- Verify MLD Snooping Synchronization for EVPN Multi-Homing, on page 51
- Multicast IRB, on page 56

Implementing 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.

Note Multicast traffic without Spanning-Tree protocol is supported at Layer 2 for multicast traffic without snooping enabled.

Prerequisites for IGMP Snooping

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

Supported Features and Restrictions for IGMP Snooping

- EVPN dual-homed Active Active (AA) IGMP State Sync using IGMP snooping profile is supported.
- BVI under bridge domain is supported.
- IGMP snooping is supported only under L2VPN bridge domains.
- Explicit host tracking (an IGMPv3 snooping feature) is not supported.
- IGMPv1 is not supported.
- IGMP snooping with VPLS on bridge domain is not supported.
- IGMP snooping over access and core Pseudo-wire is not supported.
- ISSU is not supported on Layer 2 Multicast.
- IGMPv3-exclude is not supported in EVPN multi-homing or proxy scenarios.
- For EVPN AA, IGMPv2 and IGMPv3 joins for same groups are not supported.
- **router-alert-check disable** configuration command is not supported.
- EVPN dual-home source AA is supported only on NCS 540 series router.
- EVPN configuration must have the **control-word-disable** configuration.
- PIM control packets (join and hello) processing is not supported when snooping is enabled, so a multicast router selection based on PIM packets won't occur.
- In an EVPN dual-home AA scenario:
    - If the multicast source and receiver are in the same bridge domain (BD), the receiver might receive permanent traffic duplication.
    - In an EVPN dual-home receiver AA scenario, transient traffic duplication is expected when the DH node role changes from DF to nDF and vice versa.
    - Source=ESI1=BE-X.A, Receiver=ESI1=BE-X.B under the same BD is not supported (where X.A and X.B represent two AC ports for the bundle interface BE).
    - Source=ESI1=BE-X.A (for NCS 5700 line cards), Receiver=ESI2=BE-Y.A (for NCS 5500 line cards) under the same BD is not supported (where X.A and Y.A represent two AC ports for the bundle interface BE).

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.

Information About IGMP Snooping

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 Cisco NCS 540 Series Routers randomly nominates a single port from the bundle to carry the multicast traffic.



Note The **efp-visibility** configuration is required when a bridge has attachment circuits as VLAN sub-interfaces from the same bundle-ether or physical interface.

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

You must configure **multicast-source ipv4** under L2VPN if snooping is enabled and mutlicast traffic source is located behind the AC port.

Configuration Example:

```
Router(config)# l2vpn
Router(config-l2vpn)# bridge group 1
Router(config-l2vpn-bg)#bridge-domain 1
Router(config-l2vpn-bg-bd)#multicast-source ipv4
Router(config-l2vpn-bg-bd)#efp-visibility
Router(config-l2vpn-bg-bd)#igmp snooping profile igmpsn
Router(config-l2vpn-bg-bd)#exit
Router(config-l2vpn-bg-bd)#interface TenGigE0/0/0/3.31
Router(config-l2vpn-bg-bd-ac)#exit
Router(config-l2vpn-bg-bd)#interface TenGigE0/0/0/3.32
Router(config-l2vpn-bg-bd-ac)#exit
Router(config-l2vpn-bg-bd)#routed interface BVI1
Router(config-l2vpn-bg-bd-bvi)#exit
```

Multicast Router Port

A Multicast router (Mrouter) port is a port that connects to a Multicast router. The device includes the Multicast router port(s) numbers when it forwards the Multicast streams and IGMP registration messages. This is required so that the Multicast routers can, in turn, forward the Multicast streams and propagate the registration messages to other subnets. The reports would be re-injected over mrouter ports.

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. Table 1: Multicast Traffic Handling for an IGMPv2 Querier, on page 5 describes traffic handling for an IGMPv2 querier. Table 2: Multicast Traffic Handling for an IGMPv3 Querier, on page 5 applies to an IGMPv3 querier.

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 1: Multicast Traffic Handling for an IGMPv2 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	—
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 2: 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	—
IGMP group-specific queries	—
IGMPv2 joins	Handles as IGMPv3 IS_EX{} reports.

Traffic Type	Received on Host Ports
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.

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 if BVI is configured. An empty profile enables the default configuration options and settings for IGMP snooping, as listed in the Default IGMP Snooping Configuration Settings, on page 8.

Note You must configure the **system-ip-address** and **internal-querier** when the BVI is not configured, and no other queriers are present in the same domain.

Configuration Example:

```
Router(config)#igmp snooping profile igmpsn
Router(config-igmp-snooping-profile)#system-ip-address 192.0.2.1
Router(config-igmp-snooping-profile)#internal-querier
```

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.
- 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 3: 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)
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.

EVPN All-Active Multi-homed Multicast Source Behind a BVI

Table 4: Feature History Table

Feature Name	Release Information	Feature Description
EVPN All-Active Multi-homed Multicast Source Behind a BVI	Release 7.11.1	We have enhanced multicast routing efficiency, load balancing, and latency in EVPN topology by optimizing redundancy and enabling support for All-Active (AA) multicast multi-homed sources. The multi-homed multicast data sources are located behind a Bridge-Group Virtual Interface (BVI), while multicast receivers can be in either the core or a bridge domain. This feature introduces the following changes: • **CLI** • The **ole-collapse-disable** keyword is introduced in the **hw-module multicast evpn** command. • **YANG Data Model** • New leaf **evpn-ole-collapse-disable** added in Cisco-IOS-XR-fia-hw-profile-cfg.yang (see GitHub, YANG Data Models Navigator).

EVPN AA multi-homed refers to a specific deployment model within the EVPN technology. In the multi-homed setup, a customer site or device (CE) is connected to multiple provider edge (PE) routers or attachment circuits (ACs). Multi-homing provides redundancy and load balancing by allowing a CE to connect to multiple PE routers, enabling traffic to be distributed across different paths. In case of a link (CE to PE and local PE to remote PE) or router failure, traffic can be quickly redirected to an alternate path.

In multi-homing, an AA mode means that all the links or paths between the EVPN sites are active and forwarding traffic simultaneously. This is in contrast to other deployment models, such as Single-Active or Port-Active Load-balancing mode, where only a subset of the links is active at any given time.

Placing the CE device behind the BVI interface has the following advantages:

- It allows for a simplified configuration on the CE side. The CE only needs to be configured with a single default gateway, which is the BVI interface. The CE doesn't have to manage multiple interfaces or deal with complex routing protocols.
- The BVI interface also enables efficient replication and forwarding of multicast traffic to the appropriate multicast distribution trees within the service provider network. This eliminates the need for the CE to handle multicast replication, reducing its processing load and potentially improving overall multicast performance.
- Placing the CE behind the BVI accept interface allows for greater flexibility in multi-homing scenarios. The CE can connect to multiple provider edge (PE) routers through the BVI accept interface, enabling seamless failover and load balancing between the PE routers during link or router failures.

Prerequisites

The network must support the following topology, protocols, and features to use the EVPN AA multi-homed multicast source feature:

- EVPN Control Plane with BGP
- BVI
- IGMP Snooping and MLD Snooping
- MLDP, MPLS, and OSPF (for L3 multicast receivers at core)
- Native multicast, MVPN GRE, or mVPN Profile 14 (core)

For more information related to EVPN technology and supported protocols, refer *EVPN Features* chapter in *L2VPN and Ethernet Services Configuration Guide for Cisco NCS 5500 Series Routers*.

For more information related to IGMP Snooping and MLD Snooping features, refer *Implementing Layer 2 Multicast* chapter in *Multicast Configuration Guide for Cisco NCS 5500 Series Routers*.

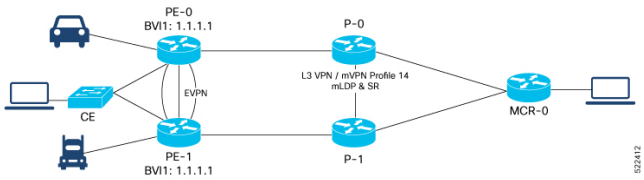
The EVPN AA multi-homed multicast source feature enables multicast data packet support for multi-homed sources in an EVPN AA (All-Active) topology.

In this setup, the multicast traffic is forwarded to the core by EVPN with BVI as the accept interface.

This deployment model combines the benefits of AA forwarding and multi-homing. It’s particularly useful in scenarios where high availability, fault tolerance, and optimized bandwidth utilization are essential requirements.

The following illustration shows the multicast data traffic route between a multi-homed source and the multi-homed receivers.

Figure 1: EVPN All-Active Multi-homed Multicast Source behind a BVI Topology



In this illustration, the multicast data sources are connected behind a CE, which is multi-homed to PE-0 and PE-1. PE-1 is configured with a BVI that has an anycast IP address. The image displays an example where

the BVI has the IP address 1.1.1.1. The receiver that is behind the MCR-0 has a PIM connection toward the multicast data source.

The data packet flow between the multicast data source and receiver occurs in the following manner:

1. The receiver, located behind Multicast Receiver (MCR-0), initiates an Internet Group Management Protocol (IGMP) join, which triggers a Protocol Independent Multicast (PIM) join towards the source.
2. The PIM join message reaches one of the PE routers (either PE-0 or PE-1) with the incoming or accept interface being the BVI and the outgoing interface leading towards the core network.
3. When the source sends traffic, it reaches one of the PE routers (PE-0 or PE-1). The next path for the traffic depends on the following IGMP snooping configurations:
   - If IGMP snooping is enabled and the multicast source is configured for both IPv4 and IPv6 traffic, the traffic is forwarded to either a route with a BVI interface or the default IGMP snoopig route.
   - If IGMP snooping is disabled, the traffic floods the multicast ID (MCID) on the bridge. As part of the flood MCID logic, the packet is recycled for the BVI and flooded to all the ACs, including the EVPN Optimized Local Egress (OLE). The recycled packets for the BVI undergo Layer 3 lookup. If there is a route with the BVI as an accepted interface, the packet is forwarded to the Olist for Layer 3 forwarding.



Note The same packet is not sent back to the CE device due to SHL (Split Horizon Label) filtering for EVPN traffic.

Usage Guidelines and Limitations

The supported scenarios for AA MH multicast are as follows:

- IPv4 SSM with BVI as accept interface is supported.
- IPv4 SM with BVI as accept interface is supported.
- IPv6 SSM with BVI as accept interface is supported.
- IPv4 SSM without BVI (only layer 2 multicast) and multicast source behind L2 is supported.
- IPv4 SM without BVI as accept interface (only layer 2 multicast) is supported.
- IPv6 SSM without BVI as accept interface (only layer 2 multicast) is supported.
- IPv6 SM without BVI as accept interface (only layer 2 multicast) is supported.

This feature has the following limitations:

- IPv6 SM with BVI as accept interface is not supported.
- Dual-homed source and Dual-homed receiver over MLDP profile on the same BD is not supported. It is recommended to disable MVPN peering between the MH nodes to prevent redundant traffic path formation in the core.
- Layer 2 IPv6 traffic is only supported on NCS 5700 fixed port routers and NCS 5500 modular routers (NCS 5700 line cards [Mode: Native]).
- In an EVPN dual-home AA scenario:

- If the multicast source and receiver are in the same BD, the receiver might receive permanent traffic duplication.
- Transient traffic duplication might occur when the DH node role changes between DF and nDF.
- In a BD, the following EVPN configuration is not supported:
  - Multicast source—ESI1=BE-X.A
  - Multicast receiver—ESI1=BE-X.B



Note ESI is the Ethernet Segment identifier, whereas X.Aand X.B represents two AC ports for the bundle interface BE.

- In a BD, the following EVPN configuration is not supported:
  - Multicast source—ESI1=BE-X.A (NCS 5700 line cards)
  - Multicast receiver—ESI1=BE-Y.A (NCS 5500 line cards)



Note ESI is the Ethernet Segment identifier, whereas X.Aand Y.A represents two AC ports for the bundle interface BE.

Configure EVPN All-Active Multi-homed Multicast Source with a BVI Interface

To configure an EVPN All-Active Multi-homed multicast source with a BVI interface, use the following example configuration:

```
Router#configure
Router(config)#l2vpn
Router(config-l2vpn)#bridge group BG1
Router(config-l2vpn-bg)#bridge-domain BD1
Router(config-bg-bd)#multicast-source ipv4-ipv6
Router(config-bg-bd)#mld snooping profile mldsn
Router(config-bg-bd)#igmp snooping profile igmpsn
Router(config-bg-bd)#interface Bundle-Ether1
Router(config-bg-bd-ac)#exit
Router(config-bg-bd)#interface TenGigE0/0/0/23.1
Router(config-bg-bd-ac)#exit
Router(config-bg-bd)#routed interface BVI1
Router(config-bg-bd)#evi 3000
Router(config-bg-bd-bvi)#commit
```

Running Configuration

This section shows the EVPN All-Active Multi-homed multicast source with BVI as accept interface running configuration.

```
l2vpn
bridge group bg1
```

```
bridge-domain bd1
multicast-source ipv4-ipv6
mld snooping profile mldsn
igmp snooping profile igmpsn
interface Bundle-Ether1
!
interface TenGigE0/0/0/23.1
!
routed interface BVI1
!
evi 3000
!
!
!
!
```

Disable EVPN Core Replications

Default behavior in EVPN involves collapsing core replications into L2 multicast routes (BD, S, G). To modify this behaviour and collapse EVPN Core to Bridge ingress multicast ID (MCID) and Snooping default routes instead of L2 multicast routes, use the following command:

```
Router(config)# hw-module multicast evpn ole-collapse-disable
```

Sample Configuration

```
Router(config)# hw-module multicast evpn ole-collapse-disable
Mon Apr 3 20:37:39.218 UTC

/*To apply the disable or re-enable EVPN OLE collapse settings, you must reload the chassis
 and all the installed line cards*/
Router# commit
Mon Apr 3 20:37:46.886 UTC
Router# end

Router# admin
Mon Apr 3 20:37:52.234 UTC
lab connected from 1.1.1.1 using ssh on sysadmin-vm:0_RP0
Reloading the RP in Order to apply the HW-cli Evpn ole collapse disable command to set

sysadmin-vm:0_RP0# hw-module location 0/RP0 reload
Mon Apr 3 20:38:15.290 UTC+00:00
Reload hardware module ? [no,yes]

/*Verification After Reload*/
Router# sh dpa objects global location 0/0/cPU0 | i evpn
Mon Apr 3 20:48:38.939 UTC
ofa_bool_t mcast_evpn_ole_collapse_disable => TRUE.

Router# sh running-config | i hw-
Mon Apr 3 20:48:43.575 UTC
hw-module multicast evpn ole-collapse-disable
```

Verification

Verify that you have configured multicast over BVI. The BVI acts as a forwarding interface for the L3 multicast packets.

```
/*PE-0*/
Router# show mrib vrf green ipv4 route 40.0.0.5
Mon May 8 12:15:44.924 UTC
(40.0.0.5,232.0.0.1) RPF nbr: 40.0.0.5 Flags: RPF
```

```
Up: 00:04:03
Incoming Interface List
BVI1 Flags: F A LI, Up: 00:04:03
Outgoing Interface List
BVI1 Flags: F A LI, Up: 00:04:03

/*Local L3 multicast join*/
TenGigE0/0/0/0.2 Flags: F NS LI, Up: 00:04:03

/*PE-1*/
Router# show mrib vrf green ipv4 route 40.0.0.5 detail
Thu May 11 09:19:07.958 UTC
(40.0.0.5,232.0.0.1) Ver: 0x1008 RPF nbr: 40.0.0.5 Flags: RPF EID, FGID: 15481, Statistics
 enabled: 0x0, Tunnel RIF: 0xffffffff, Tunnel LIF: 0xffffffff
  Up: 05:29:49
  RPF-ID: 0, Encap-ID: 262146
  Incoming Interface List
    BVI1 Flags: F A LI, Up: 05:29:49
  Outgoing Interface List
    BVI1 Flags: F A LI, Up: 05:29:49
/*Remote L3 join from multicast receiver learnt on PE-1. Multicast traffic to remote L3
multicast receiver is forwarded from PE-1*/
    Lmdtgreen Flags: F LMI TR, Up: 05:27:02, Head LSM-ID: 0x00001

/*Local L3 multicast join*/
    TenGigE0/0/0/23.2 Flags: F NS LI, Up: 05:29:48
```

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: • An empty profile is appropriate for attaching to a bridge domain. An empty

	Command or Action	Purpose
		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:**	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
	RP/0/RP0/CPU0:routershow l2vpn 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 Cisco NCS 540 Series Routers:

Configuring IGMP Snooping on Physical Interfaces Under a Bridge: Example

1. Create two profiles.

```
igmp snooping profile bridge_profile
!
igmp snooping profile port_profile

!
```

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 bridge_profile
     interface GigabitEthernet0/8/0/38
       igmp snooping profile port_profile
     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 bridge_profile

igmp snooping profile port_profile

!
```

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
   rewrite ingress tag pop 1 symmetric
   !
!
interface GigabitEthernet0/8/0/8.2 l2transport
   encapsulation dot1q 1002
   rewrite ingress tag pop 1 symmetric
   !
!
```

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 bridge_profile
     interface GigabitEthernet0/8/0/8.1
       igmp snooping profile port_profile
```

```
    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 bridge_profile
       !
       multicast-source ipv4
       igmp snooping profile port_profile

       !
```

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 bridge_profile
     interface bundle-Ether 1
       multicast-source ipv4
       igmp snooping profile port_profile
     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 for Cisco NCS 540 Series Routers*
Getting started information	
Configuring EFPs and EFP bundles	*Interface and Hardware Component Configuration Guide for Cisco NCS 540 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

MLD Snooping

Multicast Listener Discovery (MLD) snooping provides a way to constrain multicast traffic at Layer 2. By snooping the MLD membership reports sent by hosts in the bridge domain, the MLD 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.

MLD snooping uses the information in MLD membership report messages to build corresponding information in the forwarding tables to restrict IPv6 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.
- OIF List comprises all bridge ports that have sent MLD membership reports for the specified route plus all multicast router (mrouter) ports in the bridge domain.

For more information regarding MLD snooping, refer the *Multicast Configuration Guide for Cisco NCS 540 Series Routers*.

Prerequisites for MLD Snooping

- The network must be configured with a layer2 VPN.

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Supported Features and Restrictions for MLD Snooping

- BVI under bridge domain is supported.
- Receiver behind L2 ACs in the same L2 bridge domain is supported.
- Source behind L2 ACs in the same L2 bridge domain is only supported on NCS 5700 fixed port routers and NCS 5700 line cards [Mode: Compatibility; Native].
- MLDv1 not supported over BVI.
- EVPN MLD sync is not supported.
- VPLS is not supported.
- On the NCS 5700 line cards, MLD snooping can be enabled alongside IGMP snooping only.
- The **router-alert-check disable** configuration command is not supported.
- EVPN dual-home source AA is not supported on the NCS 5500 line cards line cards.
- Both IGMP and MLD snooping configurations are necessary to enable MLD snooping on the NCS 5700 line cards.
- EVPN configuration must have the **control-word-disable** configuration.
- PIM control packets (join and hello) processing is not supported when snooping is enabled, so a multicast router selection based on PIM packets won't occur.
- Explicit host tracking.
- Multicast Admission Control.
- Security filtering.
- Report rate limiting.
- Multicast router discovery.
- IPv6 multicast is not supported for a multicast source that is behind the BVI interface. For example, the below configuration is not supported:

```
l2vpn
bridge group 1
  bridge-domain 1
   multicast-source ipv6
   mld snooping profile grp1
```

- In an EVPN dual-home AA scenario:
  - If the multicast source and receiver are in the same bridge domain (BD), the receiver might receive permanent traffic duplication.
  - In an EVPN dual-home receiver AA scenario, transient traffic duplication is expected when the DH node role changes from DF to nDF and vice versa.

- Source=ESI1=BE-X.A, Receiver=ESI1=BE-X.B under the same BD is not supported (where X.A and X.B represent two AC ports for the bundle interface BE).
- Source=ESI1=BE-X.A (for NCS 5700 line cards), Receiver=ESI2=BE-Y.A (for NCS 5500 line cards) under the same BD is not supported (where X.A and Y.A represent two AC ports for the bundle interface BE).

Advantages of MLD Snooping

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

High Availability (HA) features for MLD

MLD supports the following HA features:

- 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 for MLD

MLD snooping operates at the bridge domain level. When MLD 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, VLAN range, list of VLANs, or an entire interface port.
- Ethernet bundles—Ethernet bundles include IEEE 802.3ad link bundles and Cisco EtherChannel bundles. From the perspective of the MLD snooping application, an Ethernet bundle is just another EFP. The forwarding application in the Cisco NCS 540 Series Routers randomly nominates a single port from the bundle to carry the multicast traffic.



Note The **efp-visibility** configuration is required when a bridge has attachment circuits as VLAN sub-interfaces from the same bundle-ether or physical interface.

Multicast Router and Host Ports

MLD snooping classifies each port as one of the following:

- Multicast router ports (mrouter ports)—These are ports to which a multicast-enabled router is connected. Mrouter ports are usually dynamically discovered, but may also be statically configured. Multicast traffic is always forwarded to all mrouter ports, except when an mrouter port is the ingress port.
- Host ports—Any port that is not an mrouter port is a host port.

Multicast Router Discovery for MLD

MLD snooping discovers mrouter ports dynamically. You can also explicitly configure a port as an emrouter port.

- Discovery- MLD snooping identifies upstream mrouter ports in the bridge domain by snooping mld query messages and Protocol Independent Multicast Version 2 (PIMv2) hello messages. Snooping PIMv2 hello messages identifies mld nonqueriers in the bridge domain.
- Static configuration—You can statically configure a port as an mrouter port with the **mrouter** command in a profile attached to the port. Static configuration can help in situations when incompatibilities with non-Cisco equipment prevent dynamic discovery.

Multicast Traffic Handling for MLD

The following tables describe the traffic handling behavior by MLD mrouters and host ports.

Table 5: Multicast Traffic Handling for a MLDv1 Querier

Traffic Type	Received on MRouter Ports	Received on Host Ports
IP multicast source traffic	Forwards to all mrouter ports and to host ports that indicate interest.	Forwards to all mrouter ports and to host ports that indicate interest.
MLD general queries	Forwards to all ports.	—
MLD group-specific queries	Forwards to all other mrouter ports.	Dropped
MLDv1 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.	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.
MLDv2 reports	Ignores	Ignores
MLDv1 leaves	Invokes last member query processing.	Invokes last member query processing.

Table 6: Multicast Traffic Handling for a MLDv2 Querier

Traffic Type	Received on MRouter Ports	Received on Host Ports
IP multicast source traffic	Forwards to all mrouter ports and to host ports that indicate interest.	Forwards to all mrouter ports and to host ports that indicate interest.
MLD general queries	Forwards to all ports.	—
MLD group-specific queries	If received on the querier port floods on all ports.	—
MLDv1 joins	Handles as MLDv2 IS_EX{} reports.	Handles as MLDv2 IS_EX{} reports.
MLDv2 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.	• 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.
MLDv1 leaves	Handles as MLDv2 IS_IN{} reports.	Handles as MLDv2 IS_IN{} reports.

Multicast Listener Discovery over BVI

Multicast IPv6 packets received from core, which has BVI as forwarding interface, is forwarded to access over snooped L2 AC or interface.

Note

- As per MLDv2 RFC recommendation the MLDv2 reports should carry the Hop-by-Hop options header for the reports to get punted up.
- MLDv2 is supported over BVI only when BVI is configured as a forwarding interface.

MLD and BVI Overview

Routers use the Internet Group Management Protocol (IGMP) (IPv4) and Multicast Listener Discovery (MLD) (IPv6) to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending IGMP or MLD report messages.

MLDv1 and MLDv2 are supported on NCS 540. However, MLDv2 is enabled when you configure MLD by default.

MLDv2 shares feature parity with IGMPv3 with respect to all supported interface types with the exception of PPoE and subinterfaces. MLDv2 enables a node to report interest in listening to packets only from specific multicast source addresses.

A BVI interface is a routed interface representing a set of interfaces (bridged) in the same L2 broadcast domain. MLD join messages coming in or out of this broadcast domain passes through the BVI interface.

Multicast Traffic Over Layer 2 IPv6 Network

Table 7: Feature History Table

Feature Name	Release Information	Feature Description
Multicast Traffic over Layer 2 IPv6 Network	Release 7.9.1	This feature allows you to forward the IPv6 multicast packets only to the interested MLD-snooped Access Controllers (AC), whereas in the default case, the bridge floods the IPv6 multicast packets to all AC. Routers use Multicast Listener Discovery (MLD) protocol to discover the devices in a network and create route entries in an IPv6 multicast network. This feature introduces following CLI: • **multicast-source ipv6**

The Multicast Traffic over Layer 2 IPv6 Network (L2MC IPv6) is an optimized forwarding technique, and it helps in saving the bandwidth. By default, the bridge floods IPv6 multicast packets to all AC, whereas the L2MC IPv6 feature allows you to forward the IPv6 multicast packets only to the interested MLD-snooped AC.

When IPv6 multicast packets are received over Layer 2 AC and interfaces, the lookup gets done for Virtual Switch Interfaces (VSI), Groups (G), and Services (S) or for VSI and G. The VSI details show the VLAN or VXLAN segment to which the packet belongs, while the G and S identify the multicast groups and services to which the packet should be forwarded. Based on this lookup, the traffic is forwarded to the interested receivers connected to the Layer 2 AC.

The MLD control packets received over Layer 2 AC are snooped and punted to create the route entries. This route entries are needed to avail the following supports:

- Layer 2 Multicast IPv6 support.
- EVPN sync support for IPv4 routes.

Hardware Supported

This feature is supported on routers that have the Cisco NC57 line cards installed and operate in native and compatible modes.

Limitations and Restrictions

- This feature doesn’t support MLD sync.
- With L2MC IPv6 support, the existing L2MC IPv4 scale reduces proportionally.

Configuration Example

The L2MC IPv6 feature is not enabled by default. Following is a configuration example that shows how to enable the feature.

```
router(config)# l2vpn
 router(config-l2vpn)# bridge group 1
 router(config-l2vpn-bg)#bridge-domain 1
 router(config-l2vpn-bg-bd)#multicast-source ipv6
 router(config-l2vpn-bg-bd)#efp-visibility
 router(config-l2vpn-bg-bd)#mld snooping profile prof1
 router(config-l2vpn-bg-bd)#igmp snooping profile prof1
 router(config-l2vpn-bg-bd)#interface TenGigE0/0/0/0
 router(config-l2vpn-bg-bd-ac)#exit
 router(config-l2vpn-bg-bd)#interface TenGigE0/0/0/4.1
 router(config-l2vpn-bg-bd-ac)#exit
 router(config-l2vpn-bg-bd)#interface TenGigE0/0/0/4.2
 router(config-l2vpn-bg-bd-ac)#exit
 router(config-l2vpn-bg-bd)#routed interface BVI1
 router(config-l2vpn-bg-bd-bvi)#exit
 !
 !


 router(config-l2vpn-bg-bd)#mld snooping profile prof1
 router(config-l2vpn-bg-bd)#internal-querier
!
 router(config-l2vpn-bg-bd)#igmp snooping profile prof1
 router(config-l2vpn-bg-bd)#system-ip-address 1.2.3.4
 router(config-l2vpn-bg-bd)#internal-querier
```

Note With BVI configurations, there is no need to have internal queries address configured MLD snooping profile. It implies that you can make BVI as querier under BVI configuration.

Verification

The following command shows the information about group membership in the Layer 2 Forwarding tables.

```
router# show mld snooping group

Flags Key: S=Static, D=Dynamic, E=Explicit Tracking

                 Bridge Domain bg1:bd1

Group           Ver GM  Source           PM  Port                     Exp Flg

Ff12:1:1::1     V2  Exc  -               -   GigabitEthernet0/1/1/0   122  DE
Ff12:1:1::1     V2  Exc  2002:1::1       Inc GigabitEthernet0/1/1/1     5  DE
Ff12:1:1::1     V2  Exc  2002:1::1       Inc GigabitEthernet0/1/1/2 never   S
Ff12:1:1::1     V2  Exc  2002:1::1       Exc GigabitEthernet0/1/1/3     -  DE
Ff12:1:1::1     V2  Exc  2002:1::2       Inc GigabitEthernet0/1/1/0   202  DE
Ff12:1:1::1     V2  Exc  2002:1::2       Exc GigabitEthernet0/1/1/1     -  DE
Ff12:1:1::2     V2  Exc  2002:1::1       Inc GigabitEthernet0/1/1/0   145  DE
Ff12:1:1::2     V2  Exc  2002:1::1       Inc GigabitEthernet0/1/1/1     0  DE
Ff12:1:1::2     V2  Exc  2002:1::1       Exc GigabitEthernet0/1/1/2    11  DE


                 Bridge Domain bg1:bd4

Group           Ver GM  Source           PM  Port                     Exp Flg
```

```
Ff24:1:1::2     V1  Exc  -               -   GigabitEthernet0/1/1/0   122  DE
Ff28:1:1::1     V1  -    -               -   GigabitEthernet0/1/1/1    33  DE
Ff29:1:2::3     V1  Exc  -               -   GigabitEthernet0/1/2/0   122  DE
Ff22:1:2::3     V2  Exc  2000:1:1::2     Exc GigabitEthernet0/1/2/1     5  DE
```

The following command summarizes the number of bridge domains, mrouter ports, host ports, groups, and sources configured on the router.

```
router#show mld snooping summary
  Bridge Domains:                                1
  MLD  Snooping Bridge Domains:                  1
  Ports:                                         3
  MLD  Snooping Ports:                           3
  Mrouters:                                      0
  STP Forwarding Ports:                          0
  ICCP Group Ports:                              0
  MLD  Groups:                                   0
    Member Ports:                                0
  MLD  Source Groups:                            0
    Static/Include/Exclude:                  0/0/0
    Member Ports (Include/Exclude):            0/0
```

IPv6 Multicast Listener Discovery Snooping over BVI

Multicast Listener Discovery (MLD) snooping provides a way to constrain multicast traffic at L2. By snooping the MLD membership reports sent by hosts in the bridge domain, the MLD snooping application can set up L2 multicast forwarding tables. This table is later used to deliver traffic only to ports with at least one interested member, significantly reducing the volume of multicast traffic.

MLDv2 support over BVI enables implementing IPv6 multicast routing over a L2 segment of the network that is using an IPv6 VLAN. The multicast routes are bridged via BVI interface from L3 segment to L2 segment of the network.

MLDv2 snooping over BVI enables forwarding MLDv2 membership reports received over the L2 domain to MLD snooping instead of MLD.

Restrictions

- You cannot configure `ttl-check` and disable `router-alert-check` on the router for mld messages.
- Static mrouters are not supported for MLD snooping.
- Querier is supported for MLDV2, but it is not supported on MLDV1.

Configuring Internal Querier for MLD Snooping

This configuration enables a multicast router acting as a MLD querier to send out group-and-source-specific query:

```
router# config
RP0/0/RP0/CPU0:router(config)# mld snooping profile grp1
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# system-ip-address fe80::1 link-local
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# internal-querier
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# commit
```

Verification

Use the **show mld snooping profile detail** command to verify the MLD snooping configuration:

```
router# show mld snooping profile detail
Thu Nov 22 13:58:18.844 UTC
MLD  Snoop Profile grp1:
  System IP Address:                    fe80::1
  Bridge Domain References:             2
  Port References:                      12

MLD  Snoop Profile grp10:
  System IP Address:                    fe80::5610
  Bridge Domain References:             0
  Port References:                      0
```

Creating a MLD Snooping Profile

Configuration

```
/* Enter the global configuration mode */
RP/0/RP0/CPU0:router # configure
/* Enters MLD snooping profile configuration mode and creates a named profile. */
RP/0/RP0/CPU0:router(config)# mld snooping profile default-bd-profile
RP/0/RP0/CPU0:router # commit
```

The default profile enables MLD 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.

If you are creating a bridge domain profile, consider the following:

- An empty profile is appropriate for attaching to a bridge domain. An empty profile enables MLD 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, MLD 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.

Running Configuration

```
RP/0/RP0/CPU0:router(config)# show running-config
configure
   mld snooping profile default-bd-profile
!
```

Verification

Verify that the MLD snooping profile is created:

```
RP/0/RP0/CPU0:router#show mld snooping profile

Profile                                  Bridge Domain       Port
-------                                  -------------       ----
default-bd-profile                                  0          0
grp1                                                1          2
grp10                                               1          2
```

Deactivating MLD Snooping on a Bridge Domain

To deactivate MLD snooping from a bridge domain, remove the profile from the bridge domain:



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

Configuration

```
/* Enter the global configuration mode followed by the bridge group and the bridge domain
mode */
RP0/0/RP0/CPU0:router# configuration
RP0/0/RP0/CPU0:router(config)# l2vpn
RP0/0/RP0/CPU0:router(config-l2vpn)# bridge group GRP1
RP0/0/RP0/CPU0:router(config-l2vpn-bg)# bridge domain ISP1

/* Detache the MLD snooping profile from the bridge domain. This disables MLD snooping on
that bridge domain */
/* Note: Only one profile can be attached to a bridge domain at a time. If a profile is
attached, MLD snooping is enabled.
If a profile is not attached, MLD snooping is disabled. */
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd)# no mld snooping profile
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
l2vpn
 bridge-group GRP1
  bridge-domain ISP1
   no mld snooping profile
!
```

Configuring Static Mrouter Ports (MLD)

Prerequisite

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

Note Static mrouter port configuration is a port-level option and should be added to profiles intended for ports. It is not recommended to add mrouter port configuration to a profile intended for bridge domains.

Configuration

```
/* Enter the global configuration mode */
RP0/0/RP0/CPU0:router# configuration

/* Enter the MLD snooping profile configuration mode and create a new profile or accesses
an existing profile.*/
RP0/0/RP0/CPU0:router(config)# mld snooping profile mrouter-port-profile
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# mrouter
/* Configures a static mrouter on a port. */

RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# commit
```

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
  mld snooping profile mrouter-port-profile
    mrouter
!
```

Verification

The below show command output confirms that the mrouter configuration is enabled:

```
RP0/0/RP0/CPU0:router# show mld snooping profile mrouter-port-profile

MLD  Snoop Profile mrouter-port-profile:

  Static Mrouter:                       Enabled

  Bridge Domain References:             0
  Port References:                      0
```

Configuring Router Guard (MLD)

To prevent multicast routing protocol messages from being received on a port and, therefore, prevent a port from being a dynamic mrouter port, follow these steps. Note that both router guard and static mrouter commands may be configured on the same port.

Prerequisite

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

Note Router guard configuration is a port-level option and should be added to profiles intended for ports. It is not recommended to add router guard configuration to a profile intended for bridge domains. To do so would prevent all mrouters, including MLD queriers, from being discovered in the bridge domain.

Configuration

```
/* Enter the global configuration mode and create the Bridge Group GRP1 and the Bridge
Domain ISP1*/
RP0/0/RP0/CPU0:router# configuration

/* Enter the MLD snooping profile configuration mode and create a new profile or accesses
an existing profile. */
RP0/0/RP0/CPU0:router(config)# mld snooping profile host-port-profile

/* Configure router guard. This protects the port from dynamic discovery.*/
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# router-guard
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
 mld snooping profile host-port-profile
  router-guard
 !
```

Verification

Verify that the router guard config in the named profile is enabled:

```
RP0/0/RP0/CPU0:router# show mld snooping profile host-port-profile detail
MLD  Snoop Profile host-port-profile:

  Router Guard:                          Enabled

  Bridge Domain References:              0
  Port References:                       0
```

Configuring Immediate-leave for MLD

To add the MLD snooping immediate-leave option to an MLD snooping profile:

Configuration

```
/* Enter the global configuration mode. */
RP0/0/RP0/CPU0:router# configuration

/* Enter MLD snooping profile configuration mode and create a new profile or accesses an
existing profile. */
RP0/0/RP0/CPU0:router(config)# mld snooping profile host-port-profile
/* Enable the immediate-leave option */
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# immediate-leave
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

If you add the **immediate-leave** option:

- to a profile attached to a bridge domain, it applies to all ports under the bridge.
- to a profile attached to a port, it applies to the port.

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
 mld snooping profile host-port-profile
 immediate-leave
!
```

Verification

Verify that the immediate leave config in the named profile is enabled:

```
RP0/0/RP0/CPU0:router# show mld snooping profile host-port-profile detail

MLD  Snoop Profile host-port-profile:

  Immediate Leave:                    Enabled
  Router Guard:                       Enabled

  Bridge Domain References:           0
  Port References:                    0
```

Configuring Internal Querier for MLD

Prerequisite

MLD snooping must be enabled on the bridge domain for this procedure to take effect.

Configuration

```
/* Enter the global configuration mode. */
RP0/0/RP0/CPU0:router# configuration

/* Enter MLD snooping profile configuration mode and create a new profile or accesses an
existing profile. */
RP0/0/RP0/CPU0:router(config)# mld snooping profile internal-querier-profile

/* Configure an IP address for internal querier use. The default system-ip-address value
(0.0.0.0) is not valid for the internal querier.
You must explicitly configure an IP address. Enter a valid link-local IPv6 address. */
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# system-ip-address fe80::98 link-local

/* Enable an internal querier with default values for all options.*/
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# internal-querier
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# commit
```

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
 mld snooping profile internal-querier-profile
 system-ip-address fe80::98 link-local
 internal-querier
!
```

Note Internal Querier is not recommended, when BVI and MLD snooping is configured under a bridge.

Verification

Verify that the internal querier config is enabled:

```
RP0/0/RP0/CPU0:router# show mld snooping profile internal-querier-profile detail

MLD  Snoop Profile internal-querier-profile:

  System IP Address:                    fe80::98

  Internal Querier Support:             Enabled

  Bridge Domain References:             0
  Port References:                      0
```

Configuring Static Groups for MLD

To add one or more static groups or MLDv2 source groups to an MLD snooping profile, follow these steps:

Prerequisite

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

Configuration

```
/* Enter the global configuration mode. */
RP0/0/RP0/CPU0:router# configuration

/* Enter MLD snooping profile configuration mode and create a new profile or accesses an
existing profile. */
RP0/0/RP0/CPU0:router(config)# mld snooping profile host-port-profile

/* Configure a static group. */
/* Note: Repeat this step to add additional static groups. */
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# static group fe80::99 source fe80::99
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# commit
```

If you add the **static group** option:

- to a profile attached to a bridge domain, it applies to all ports under the bridge.
- to a profile attached to a port, it applies to the port.

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
 mld snooping profile host-port-profile
 static group fe80::99 source fe80::99
!
```

Verification

```
RP0/0/RP0/CPU0:router# show mld snooping bridge-domain f1:100 detail

Bridge Domain        Profile                         Act  Ver  #Ports  #Mrtrs  #Grps
#SGs
-------------        -------                         ---  ---  ------  ------  -----
----
f1:100               grp1                             Y  v2       3       1   1000   1002


  Profile Configured Attributes:
    System IP Address:                  fe80::99
    Minimum Version:                    1
    Report Suppression:                 Enabled
    Unsolicited Report Interval:        1000 (milliseconds)
    TCN Query Solicit:                  Disabled
    TCN Membership Sync:                Disabled
    TCN Flood:                          Enabled
    TCN Flood Query Count:              2
    Router Alert Check:                 Disabled
    TTL Check:                          Enabled
    nV Mcast Offload:                   Disabled
    Internal Querier Support:           Disabled
    Querier Query Interval:             125 (seconds)
    Querier LMQ Interval:               1000 (milliseconds)
    Querier LMQ Count:                  2
    Querier Robustness:                 2
    Startup Query Interval:             31 seconds
    Startup Query Count:                2
    Startup Query Max Response Time:    10.0 seconds
    Mrouter Forwarding:                 Enabled
    P2MP Capability:                    Disabled
    Default IGMP Snooping profile:      Disabled
    IP Address:                         fe80::f278:16ff:fe63:4d81
    Port:                               BVI1000
    Version:                            v2
    Query Interval:                     125 seconds
    Robustness:                         2
    Max Resp Time:                      10.0 seconds
    Time since last G-Query:            97 seconds
  Mrouter Ports:                        1
    Dynamic:                            BVI1000
  STP Forwarding Ports:                 0
  ICCP Group Ports:                     0
  Groups:                               1000
    Member Ports:                       0
  V2 Source Groups:                     1002
    Static/Include/Exclude:             0/1002/0
    Member Ports (Include/Exclude):     1002/0
```

Configuring MLD Snooping

Configure

```
RP0/0/RP0/CPU0:router# configure
/* Create two profiles. */
RP0/0/RP0/CPU0:router(config)# mld snooping profile bridge_profile
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# mld snooping profile port_profile
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# mrouter
```

```
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# exit
RP0/0/RP0/CPU0:router(config)#

/* Configure two physical interfaces for L2 support.*/
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/8/0/38
RP0/0/RP0/CPU0:router(config-if)# negotiation auto
RP0/0/RP0/CPU0:router(config-if)# l2transport
RP0/0/RP0/CPU0:router(config-if)# no shut
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# interface GigabitEthernet0/8/0/39
RP0/0/RP0/CPU0:router(config-if)# negotiation auto
RP0/0/RP0/CPU0:router(config-if)# l2transport
RP0/0/RP0/CPU0:router(config-if)# no shut
RP0/0/RP0/CPU0:router(config-if)# exit

/* 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 MLD snooping configuration attributes from the bridge
 domain profile.*/
RP0/0/RP0/CPU0:router(config)# l2vpn
RP0/0/RP0/CPU0:router(config-l2vpn)# bridge group bg1
RP0/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain bd1
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd)# mld snooping profile bridge_profile
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd-mld-snooping)# interface GigabitEthernet0/8/0/38
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd-mld-snooping-if)# mld snooping profile port_profile
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd-mld-snooping-if)# interface GigabitEthernet0/8/0/39
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd-mld-snooping-if)# exit
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd-mld-snooping)# exit
RP0/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
 mld snooping profile bridge_profile
!
mld snooping profile port_profile
   mrouter
!

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

l2vpn
  bridge group bg1
     bridge-domain bd1
     mld snooping profile bridge_profile
     interface GigabitEthernet0/8/0/38
       mld snooping profile port_profile
     interface GigabitEthernet0/8/0/39
     !
  !
!
```

Verification

Verify the configured bridge ports.

```
RP0/0/RP0/CPU0:router# show mld snooping port


                        Bridge Domain f10:109

                                                        State
Port                                                  Oper  STP  Red   #Grps   #SGs
----                                                  ----  ---  ---   -----   ----
BVI1009                                                 Up    -    -       0      0
GigabitEthernet0/8/0/38                                 Up    -    -    1000   1000
GigabitEthernet0/8/0/39                                 Up    -    -    1000   1000
```

Configuring MLD Snooping on Ethernet Bundles

This example assumes that the front-ends of the bundles are preconfigured. For example, a bundle configuration might consist of three switch interfaces, as follows:

Configure

```
/*  Configure the front-ends of the bundles consisting of three switch interfaces.*/
RP0/0/RP0/CPU0:router# configure
RP0/0/RP0/CPU0:router(config)# interface bundle-ether 1
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/1
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/2
RP0/0/RP0/CPU0:router(config-if)# channel-group 1 mode on
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/3
RP0/0/RP0/CPU0:router(config-if)# channel-group 1 mode on
RP0/0/RP0/CPU0:router(config-if)# exit

/* Configure two MLD snooping profiles. */
RP0/0/RP0/CPU0:router(config)# mld snooping profile bridge_profile
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# exit     !
RP0/0/RP0/CPU0:router(config)# mld snooping profile port_profile
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# mrouter
RP0/0/RP0/CPU0:router(config-mld-snooping-profile)# exit

/* Configure interfaces as bundle member links. */

RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0
RP0/0/RP0/CPU0:router(config-if)# bundle id 1 mode on
RP0/0/RP0/CPU0:router(config-if)# negotiation auto
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/1
RP0/0/RP0/CPU0:router(config-if)# bundle id 1 mode on
RP0/0/RP0/CPU0:router(config-if)# negotiation auto
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/2
RP0/0/RP0/CPU0:router(config-if)# bundle id 2 mode on
RP0/0/RP0/CPU0:router(config-if)# negotiation auto
RP0/0/RP0/CPU0:router(config-if)# exit
```

```
RP0/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/3
RP0/0/RP0/CPU0:router(config-if)# bundle id 2 mode on
RP0/0/RP0/CPU0:router(config-if)# negotiation auto
RP0/0/RP0/CPU0:router(config-if)# exit

/* Configure the bundle interfaces for L2 transport. */
RP0/0/RP0/CPU0:router(config)# interface Bundle-Ether 1
RP0/0/RP0/CPU0:router(config-if)# l2transpor
RP0/0/RP0/CPU0:router(config-if)# exit
RP0/0/RP0/CPU0:router(config)# interface Bundle-Ether 2
RP0/0/RP0/CPU0:router(config-if)# l2transpor
RP0/0/RP0/CPU0:router(config-if)# exit

/* Add the interfaces to the bridge domain and attach MLD snooping profiles. */
RP0/0/RP0/CPU0:router(config)# l2vpn
RP0/0/RP0/CPU0:router(config-l2vpn)# bridge group bg1
RP0/0/RP0/CPU0:router(config-l2vpn-bg)# mld snooping profile bridge_profile
RP0/0/RP0/CPU0:router(config-l2vpn-bg-mld-snooping-profile)# interface bundle-Ether 1
RP0/0/RP0/CPU0:router(config-l2vpn-bg-mld-snooping-profile-if)# mld snooping profile
port_profile
RP0/0/RP0/CPU0:router(config-l2vpn-bg-mld-snooping-profile-if)# interface bundle-Ether 2
RP0/0/RP0/CPU0:router(config-l2vpn-bg-mld-snooping-profile-if)# commit
```

Running Configuration

```
RP0/0/RP0/CPU0:router# show running-config
configuration
 interface Port-channel1
    !
interface GigabitEthernet0/0/0/0
    !
interface GigabitEthernet0/0/0/1
!
    interface GigabitEthernet0/0/0/2
       channel-group 1 mode on
    !
    interface GigabitEthernet0/0/0/3
       channel-group 1 mode on
    !
mld snooping profile bridge_profile
       !
       mld snooping profile port_profile
          mrouter
       !
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
       !
interface Bundle-Ether 1
          l2transport
```

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

l2vpn
         bridge group bg1
            bridge-domain bd1
            mld snooping profile bridge_profile
            interface bundle-Ether 1
              mld snooping profile port_profile
            interface bundle-Ether 2
            !
          !
      !
```

Verification

```
RP0/0/RP0/CPU0:router# show mld snooping port
Bridge Domain BG1:BD1
State
Port Oper STP Red #Grps #SGs
---- ---- --- --- ----- ----
HundredGigE0/0/0/3 Up - - 1 1
HundredGigE0/0/0/7 Up - - 1 1
HundredGigE0/19/0/11 Up - - 1 1
HundredGigE0/19/0/5 Up - - 1 1
RP/0/RP1/CPU0:Router#
```

MLD Snooping Synchronization for EVPN Multi-Homing

Table 8: Feature History Table

Feature Name	Release Information	Feature Description
MLD Snooping Synchronization for EVPN Multi-Homing	Release 7.11.1	The Designated Forwarder (DF) PE router in an EVPN multi-homed network can now efficiently forward multicast traffic from the source to the interested receivers, avoiding unnecessary replication and reducing network bandwidth consumption. This is made possible by introducing support for Multicast Listener Discovery, MLDv1, and MLDv2 (IPv6) snooping state synchronization for EVPN multi-homing peers or provider edge (PE) devices, expanding the scope of the previous support for IGMP (IPv4) snooping state synchronization.

In an EVPN multi-homing network, where customer edge devices (CEs) are multi-homed to more than one PE device, the MLD snooping synchronization feature enables routers to accurately track multicast group membership information and forward multicast traffic only to the interested receivers.

In an All-Active redundancy mode, the CEs can send an MLD message to any one of the multi-homed PEs, either DF or non-DF. Only the EVPN DF forwards traffic for the bridge domain (BD) for any group. Therefore, all PEs attached to a given EVPN Segment (ES) must coordinate MLD Join and Leave Group (x, G) state, where x may be either '*' (unspecified multicast source address) or a particular source address 'S', and G represents the multicast group address, for each [EVI, broadcast domain (BD)] on that ES. This allows the DF for that ES, EVI, or BD to correctly advertise or withdraw a Selective Multicast Ethernet Tag route for that (x, G) group in that EVI or BD when needed.

In Single-Active redundancy mode, the PEs attached to a multi-homed ES coordinate the MLD Join (x, G) state. MLD join messages (MLD host membership reports) are received by the DF PE and distributed to the non-DF PEs for faster convergence. The non-DF PE also receives traffic by building the distribution tree toward the Rendezvous Point (RP) or multicast source, but doesn't forward it to the receivers in a multicast group. When a non-DF PE becomes the DF PE, it starts forwarding traffic to the CE.

Some benefits of the MLD state synchronization feature are as follows:

- **Seamless Mobility Support**—It ensures smooth mobility support for multicast listeners. When listeners move between different network devices or ports, the synchronized MLD snooping state helps maintain consistent multicast group membership information. The DF intelligently updates the forwarding information, ensuring uninterrupted multicast service delivery to mobile listeners.
- **Reduced Control Plane Overhead**—By synchronizing the MLD snooping state, we have reduced signaling messages overhead in the control plane for routing. The DF processes and propagates multicast control messages, such as MLD join and leave messages, only to the relevant ports based on the synchronized group membership information. This minimizes unnecessary control plane processing and improves network scalability.
- **Enhanced Network Stability**:—It contributes to network stability by maintaining consistent multicast group membership information across PE devices. This ensures reliable multicast service delivery and prevents disruptions or inconsistencies that could impact the network's overall performance.
- **Efficient Resource Utilization**—It uses a DF to optimize resource utilization by forwarding multicast traffic only to the ports where receivers are present. This prevents unnecessary multicast data replication and conserves network bandwidth, improving overall network efficiency.

MLD Snooping Synchronization with Proxy Querier

Each subnet has one of the two roles:

- Querier—the router with the lowest IP address in a subnet. Querier is responsible for sending the MLD or IGMP queries to know which multicast groups are active on the subnet.
- Non-Querier—the router that listens for MLD or IGMP queries and forwards them to the entire VLAN.

Initially, all multicast routers start up as a Querier on each attached network. If a router hears a Query message from a lower IP address, it becomes a Non-Querier. If a router doesn't hear a Query message for a certain period, it becomes the Querier again. The Querier router regularly sends a General Query on each attached network to gather multicast group membership information.

In this feature, two peer PEs in EVPN can both act as Queriers for the same BD. The first PE receiving the MLD join from CE sends an EVPN Join sync message to the second peer PE, which, upon receipt, sets the

"learnt via EVPN " flag on the group. The group is not expired for lack of a direct MLD Join response to the initiated query (by second PE) as long as the 'learnt via EVPN' flag is set.

Note The MLD queries are not sent over the MPLS core. ACL filter is applied on the core facing interface to drop all the MLD queries.

Usage Guidelines and Limitations

A BD can have a combination of MLDv1 and MLDv2 receivers, sending corresponding Join messages to the PE router. Additionally, MLDv2 Join messages could either be in the include or exclude mode, where a multicast receiver can specify to either listen only for packets from some list of source addresses (include) or only for packets that don't come from some list of source addresses (exclude). Handling such mixed MLD Joins involves ensuring that the PE routers can properly interpret and forward multicast traffic to hosts accordingly.

In the following table, use the supported and unsupported scenarios for the MLDv1 and MLDv2 Joins at PE as guidelines for using the MLD snooping synchronization feature:

Table 9: MLDv1 and MLDv2 Combination Joins at PE—Supported and Unsupported Scenarios

Current MLD Join State	**MLD Join Update Received**			
	v1 (*, G)	**v2 Include (S, G)**	**v2 Exclude (*, G)**	**v2 Exclude (S, G)**
No state	Accepted	Accepted	Accepted	Drop
v1 (*, G)	Accepted	Drop	Accepted	Drop
v2 Include (S, G)	Drop	Accepted	Drop	Drop
v2 Exclude (*, G)	Accepted	Drop	Accepted	Drop

This feature has the following limitations:

- If the source is directly connected to the PE where the MLD Join is received, no MLD sync route is generated.
- Any router behind an All-Active multi-homed network is not supported.
- Configuring different MLD snooping profiles on peer PEs in an All-Active multi-homed network is not supported.
- An mrouter port behind CE is not supported.
- To prevent convergence issues, per multicast route DF election is not supported.
- The IGMP and MLD snooping profiles must be enabled together.

Configure MLD Snooping Synchronization for EVPN Multi-Homing

To configure MLD Snooping Synchronization for EVPN Multi-Homing, use the following example configuration:

/* Configure the EVPN EVI */

```
Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether34
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 23.23.23.11.FF.11.11.11.11
Router(config-evpn-ac-es)# exit
Router(config-evpn-ac)# exit
Router(config-evpn)# evi 5
Router(config-evpn-instance)# advertise-mac
Router(config-evpn-instance)# exit
Router(config-evpn)# exit
```

/* Configure the L2VPN BD with MLD snooping profile and EVI */

```
Router(config)# l2vpn
Router(config-l2vpn)# bridge group bg1
Router(config-l2vpn-bg)# bridge-domain bd5
Router(config-l2vpn-bg-bd)# mld snooping profile prof1
Router(config-l2vpn-bg-bd-mld-snooping-profile)# exit
Router(config-l2vpn-bg-bd)# igmp snooping profile prof2
Router(config-l2vpn-bg-bd-igmp-snooping-profile)# exit
Router(config-l2vpn-bg-bd)# interface Bundle-Ether34.5
Router(config-l2vpn-bg-bd-ac)# exit
Router(config-l2vpn-bg-bd)# routed interface BVI5
Router(config-l2vpn-bg-bd-bvi)# exit
Router(config-l2vpn-bg-bd)# evi 5
Router(config-l2vpn-bg-bd-evpn-instance)# exit
```

/* Configure the MLD snooping profile

```
Router(config)# mld snooping profile prof1
Router(config-mld-snooping-profile)# internal-querier
Router(config-mld-snooping-profile)# internal-querier query-interval 5
Router(config-mld-snooping-profile)# commit
```

Running Configuration

```
/*EVPN EVI*/
evpn
    interface Bundle-Ether34
      ethernet-segment
        identifier type 0 23.23.23.11.FF.11.11.11.11
    !
    evi 5
      advertise-mac
      !
    !
```

```
/* Configure the L2VPN BD with MLD snooping profile and EVI */
l2vpn
 bridge group bg1
   bridge-domain bd5
     mld snooping profile prof1
     igmp snooping profile prof2
     interface Bundle-Ether34.5
     !
     routed interface BVI5
     !
     evi 5
     !

/*MLD Snooping Profile*/
mld snooping profile prof1
     internal-querier
     internal-querier query-interval 5
```

Verify MLD Snooping Synchronization for EVPN Multi-Homing

To verify the configuration for this feature, use the following example commands.

/*Verify MLD Snooping Synchronization*/

```
RP/0/RP1/CPU0:tb11-r8#show mld  snooping  group
Fri Oct  6 17:53:42.640 UTC

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

                            Bridge Domain bg-1:bd-1001

Ver GM PM Port                          Exp   Flgs Group,Source
--- -- -- ----                          ---   ---- ------------
*/B indicates MLD snooping sync through BGP*/
V2  IN IN BE1.1001                      never B    ff03::1,1108:101::100

*/D indicates MLD snooping state is locally learned through EVPN*/
V2  IN IN BE2.1001                      223   D    ff03::1,1108:101::100
```

/*Verify DF Election*/

```
RP/0/RP1/CPU0:tb11-r8#show evpn ethernet-segment carving detail

Tue Oct 17 18:14:56.607 UTC
Legend:
  B   - No Forwarders EVPN-enabled,
  C   - MAC missing (Backbone S-MAC PBB-EVPN / Grouping ES-MAC vES),
  RT  - ES-Import Route Target missing,
  E   - ESI missing,
  H   - Interface handle missing,
  I   - Name (Interface or Virtual Access) missing,
  M   - Interface in Down state,
  O   - BGP End of Download missing,
  P   - Interface already Access Protected,
  Pf  - Interface forced single-homed,
  R   - BGP RID not received,
  S   - Interface in redundancy standby state,
  X   - ESI-extracted MAC Conflict
  SHG - No local split-horizon-group label allocated
  Hp  - Interface blocked on peering complete during HA event
```

```
  Rc  - Recovery timer running during peering sequence

Ethernet Segment Id      Interface                         Nexthops
------------------------ --------------------------------- --------------------
0000.0100.ac00.0001.0a00 BE1                               7.7.7.7
                                                           8.8.8.8
  ES to BGP Gates   : Ready
  ES to L2FIB Gates : Ready
  Main port         :
     Interface name : Bundle-Ether1
     Interface MAC  : b402.1657.e485
     IfHandle       : 0x2000a164
     State          : Up
     Redundancy     : Not Defined
  ESI ID            : 1
  ESI type          : 0
     Value          : 0000.0100.ac00.0001.0a00
  ES Import RT      : 0001.00ac.0000 (from ESI)
  Source MAC        : 0000.0000.0000 (N/A)
  Topology          :
     Operational    : MH, All-active
     Configured     : All-active (AApF) (default)
  Service Carving   : Auto-selection
     Multicast      : Disabled
  Convergence       :
  Peering Details   : 2 Nexthops
     7.7.7.7 [MOD:P:00:T]
     8.8.8.8 [MOD:P:00:T]
  Service Carving Synchronization:
     Mode           : NONE
     Peer Updates   :
                 7.7.7.7 [SCT: N/A]
                 8.8.8.8 [SCT: N/A]
  Service Carving Results:
     Forwarders     : 999
     Elected        : 500
            EVI E   :     1001,    1003,    1005,    1007,    1009,    1011
            EVI E   :     1013,    1015,    1017,    1019,    1021,    1023,
            …..
            EVI E   :     1999,    2001
     Not Elected    : 499
            EVI NE  :     1002,    1004,    1006,    1008,    1010,    1012
           …
            EVI NE  :     1990,    1992,    1994,    1996,    1998,    2000,
            EVI NE  :     2002
  …..
     Main port         :
     Interface name : Bundle-Ether2
     Interface MAC  : b402.1657.e484
     IfHandle       : 0x2000a16c
     State          : Up
     Redundancy     : Not Defined
  ESI ID            : 1
  ESI type          : 0
     Value          : 0011.0200.ac00.0001.0a00
  ES Import RT      : 1102.00ac.0000 (from ESI)
  Source MAC        : 0000.0000.0000 (N/A)
  Topology          :
     Operational    : MH, All-active
     Configured     : All-active (AApF) (default)
  Service Carving   : Auto-selection
     Multicast      : Disabled
  Convergence       :
  Peering Details   : 2 Nexthops
```

```
      7.7.7.7 [MOD:P:00:T]
      8.8.8.8 [MOD:P:00:T]
   Service Carving Synchronization:
      Mode           : NONE
      Peer Updates   :
                  7.7.7.7 [SCT: N/A]
                  8.8.8.8 [SCT: N/A]
   Service Carving Results:
      Forwarders     : 998
      Elected        : 500
             EVI E   :     1001,    1003,    1005,    1007,    1009,    1011
            …
             EVI E   :     1987,    1989,    1991,    1993,    1995,    1997,
             EVI E   :     1999,    2001
      Not Elected    : 498
             EVI NE  :     1002,    1004,    1006,    1008,    1010,    1012
             EVI NE  :     1980,    1982,    1984,    1986,    1988,    1990,
             EVI NE  :     1992,    1994,    1996,    1998,    2000,    2002
   EVPN-VPWS Service Carving Results:
      Primary        : 0
      Backup         : 0
      Non-DF         : 0
   MAC Flushing mode : STP-TCN
   Peering timer     : 3 sec [not running]
   Recovery timer    : 30 sec [not running]
   Carving timer     : 0 sec [not running]
   Revert timer      : 0 sec [not running]
   HRW Reset timer   : 5 sec [not running]
   Local SHG label   : 27051
   Remote SHG labels : 1
               27051 : nexthop 7.7.7.7
   Access signal mode: Bundle OOS

N/A                       Te0/1/0/4/0                          8.8.8.8
   ES to BGP Gates   : Ready
   ES to L2FIB Gates : Ready
   Main port         :
      Interface name : TenGigE0/1/0/4/0
      Interface MAC  : b402.1657.e0a0
      IfHandle       : 0x020040c8
      State          : Up
      Redundancy     : Not Defined
   ESI ID            : 0
   ESI type          : Invalid
   ES Import RT      : 0000.0000.0000 (Incomplete Configuration)
   Source MAC        : b402.1657.e480 (PBB BSA, no ESI)
   Topology          :
      Operational    : SH
      Configured     : Single-active (AApS) (default)
   Service Carving   : Auto-selection
      Multicast      : Disabled
   Convergence       :
   Peering Details   : 1 Nexthops
      8.8.8.8 [MOD:P:00]
   Service Carving Synchronization:
      Mode           : NONE
      Peer Updates   :
                  8.8.8.8 [SCT: N/A]
   Service Carving Results:
      Forwarders     : 10
      Elected        : 10
             EVI E   :     1001,    1002,    1003,    1004,    1005,    1006
             EVI E   :     1007,    1008,    1009,    1010
      Not Elected    : 0
```

```
  EVPN-VPWS Service Carving Results:
     Primary        : 0
     Backup         : 0
     Non-DF         : 0
  MAC Flushing mode : STP-TCN
  Peering timer     : 0 sec [not running]
  Recovery timer    : 0 sec [not running]
  Carving timer     : 0 sec [not running]
  Revert timer      : 0 sec [not running]
  HRW Reset timer   : 5 sec [not running]
  Local SHG label   : None
  Remote SHG labels : 0
  Access signal mode: Unsupported
```

/*Verify EVPN IGMP Snooping*/

```
RP/0/RSP0/CPU0:tb8-r3-AVA2#show evpn igmp
Mon Nov  6 11:18:19.497 UTC

EVI   Ethernet Segment         (S,G)                                      Source
                Type
----- ---------------------- ------------------------------------------
------------------------------ ------
1001  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::1)                 Bundle-Ether1.1001
            JOIN
1001  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::1)                 Bundle-Ether2.1001
            JOIN
1001  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::1:2)               Bundle-Ether1.1001
            JOIN
1001  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::1:2)               Bundle-Ether2.1001
            JOIN
1001  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:1::1)            Bundle-Ether2.1001
            JOIN
1001  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:1::1)           Bundle-Ether1.1001
            JOIN
1001  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:1::1)           Bundle-Ether2.1001
            JOIN
1001  0000.0100.ac00.0001.0a00 (2205:101::23,ff03:13:1::1)             Bundle-Ether1.1001
            JOIN
1001  0011.0200.ac00.0001.0a00 (2205:101::23,ff03:13:1::1)             Bundle-Ether2.1001
            JOIN
1002  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::2)                 Bundle-Ether1.1002
            JOIN
1002  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::2)                 Bundle-Ether2.1002
            JOIN
1002  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:2::1)            Bundle-Ether2.1002
            JOIN
1002  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:2::1)           Bundle-Ether1.1002
            JOIN
1002  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:2::1)           Bundle-Ether2.1002
            JOIN
1002  0000.0100.ac00.0001.0a00 (2205:101::23,ff03:14:1::1)             Bundle-Ether1.1002
            JOIN
1002  0011.0200.ac00.0001.0a00 (2205:102::441,ff03:14:1::1)            Bundle-Ether2.1002
            JOIN
1003  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::3)                 Bundle-Ether1.1003
            JOIN
1003  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::3)                 Bundle-Ether2.1003
            JOIN
1003  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:3::1)            Bundle-Ether2.1003
            JOIN
1003  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:3::1)           Bundle-Ether1.1003
            JOIN
```

```
1003  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:3::1)             Bundle-Ether2.1003
          JOIN
1004  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::4)                   Bundle-Ether1.1004
          JOIN
1004  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::4)                   Bundle-Ether2.1004
          JOIN
1004  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:4::1)             Bundle-Ether2.1004
          JOIN
1004  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:4::1)            Bundle-Ether1.1004
          JOIN
1004  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:4::1)            Bundle-Ether2.1004
          JOIN
1005  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::5)                   Bundle-Ether1.1005
          JOIN
1005   0011.0200.ac00.0001.0a00 (1108:101::100,ff03::5)                     7.7.7.7
             JOIN
1005  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:5::1)             Bundle-Ether2.1005
          JOIN
1005  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:5::1)            Bundle-Ether1.1005
          JOIN
1005  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:5::1)            Bundle-Ether2.1005
          JOIN
1006  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::6)                   Bundle-Ether1.1006
          JOIN
1006  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::6)                   Bundle-Ether2.1006
          JOIN
1006  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:6::1)             Bundle-Ether2.1006
          JOIN
1006  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:6::1)            Bundle-Ether1.1006
          JOIN
1006   0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:6::1)               7.7.7.7
             JOIN
1007  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::7)                   Bundle-Ether1.1007
          JOIN
1007  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::7)                   Bundle-Ether2.1007
          JOIN
1007   0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:7::1)               7.7.7.7
             JOIN
1007  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:7::1)            Bundle-Ether1.1007
          JOIN
1007  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:7::1)            Bundle-Ether2.1007
          JOIN
1008  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::8)                   Bundle-Ether1.1008
          JOIN
1008   0011.0200.ac00.0001.0a00 (1108:101::100,ff03::8)                     7.7.7.7
             JOIN
1008  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:8::1)             Bundle-Ether2.1008
          JOIN
1008  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:8::1)            Bundle-Ether1.1008
          JOIN
1008  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:8::1)            Bundle-Ether2.1008
          JOIN
1009  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::9)                   Bundle-Ether1.1009
          JOIN
1009  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::9)                   Bundle-Ether2.1009
          JOIN
1009  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:9::1)             Bundle-Ether2.1009
          JOIN
1009  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:9::1)            Bundle-Ether1.1009
          JOIN
1009  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:9::1)            Bundle-Ether2.1009
          JOIN
```

```
1010  0000.0100.ac00.0001.0a00 (1108:101::100,ff03::a)                  Bundle-Ether1.1010
            JOIN
1010  0011.0200.ac00.0001.0a00 (1108:101::100,ff03::a)                  Bundle-Ether2.1010
            JOIN
1010  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:16:a::1)               7.7.7.7
              JOIN
1010  0000.0100.ac00.0001.0a00 (1108:101::100,ff03:123:a::1)            Bundle-Ether1.1010
            JOIN
1010  0011.0200.ac00.0001.0a00 (1108:101::100,ff03:123:a::1)            Bundle-Ether2.1010
            JOIN
```

Multicast IRB

Multicast IRB provides the ability to route multicast packets between a bridge group and a routed interface using a bridge-group virtual interface (BVI). It can be enabled with multicast-routing. THE BVI is a virtual interface within the router that acts like a normal routed interface. For details about BVI, refer *Interface and Hardware Component Configuration Guide for Cisco NCS 540 Series Routers*

BV interfaces are added to the existing VRF routes and integrated with the replication slot mask. After this integration, the traffic coming from a VRF BVI is forwarded to the VPN.

Supported Bridge Port Types

- Bundles
- Satellites
- EFPs (physical, vlans, etc)

Restrictions

- Supported only on Ethernet line cards and enhanced ethernet line cards.
- Support only for IPv4

Example

The CE-PE is collapsed into 1 router (IRB) and IGMP snooping is enabled on the BVIs.

BVI type is included in a multicast VRF. After the BVI slot mask is included in the VRF route slot mask, the traffic from the VRF BVI is forwarded to the VPN/ core.

Access Pseudowire in VPLS Bridge Domains

Table 10: Feature History Table

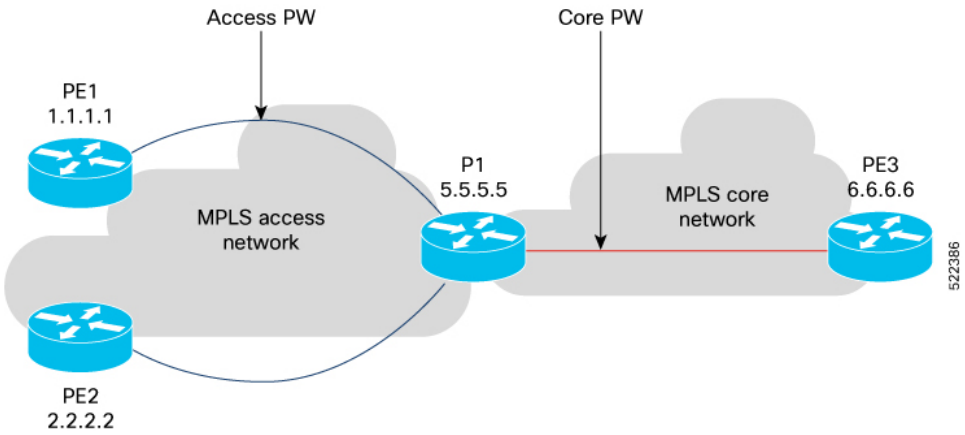
Feature Name	Release Information	Description
Access Pseudowire in VPLS Bridge Domains	Release 7.6.1	You can configure EVPN in the access node under the same bridge domain as EVPN in the core and create a pseudowire (PW) to the nearest PE that binds the access circuits using EVPN. This PW between the access PE and the single-homed PE ensures that the access nodes can leverage the benefits of EVPN.

You can enable VPLS Access Pseudowire in a Bridge Domain (BD) where flooding is enabled.

VPLS is a multipoint Layer 2 VPN technology that connects two or more customer devices using bridging techniques. In scenarios where an L3 multicast route has invalid or incorrect OLEs (Output List Element: a hardware instance of a multicast outgoing interface in a multicast route), instead of dropping the packets, they are sent again to the receiver. If the L3 multicast route already has valid OLE entries apart from the invalid ones, at the receiver end, you can see duplicate packets.

To ensure an uninterrupted flow of packets, the egress traffic management model employs a two-pass model. When you enable access pseudowire, in the two-pass model, at egress, the duplicate IP packet is recycled and gets embedded and egresses from the bundle-ether as OLE.

Following figure shows the interconnection between the provider edge (PE) routers over IP/MPLS networks. The VPLS network requires a bridge domain (Layer 2 broadcast domain) on each PE router. It is responsible for all flooding broadcast frames and multicast replications. The PEs are connected with Pseudowires (PWs).



Limitations

This feature is not supported when IGMP snooping is enabled.

The multicast L3 to L2 traffic is supported only in flood BD configuration.

Configure Access Pseudowire

To enable Access Pseudowire in a VPLS BD, use the following command:

```
Router#configure terminal
Router(config)#hw-module multicast access-pw-enable
```