Configuring Tenant Routed Multicast (TRM)

This chapter contains these sections:

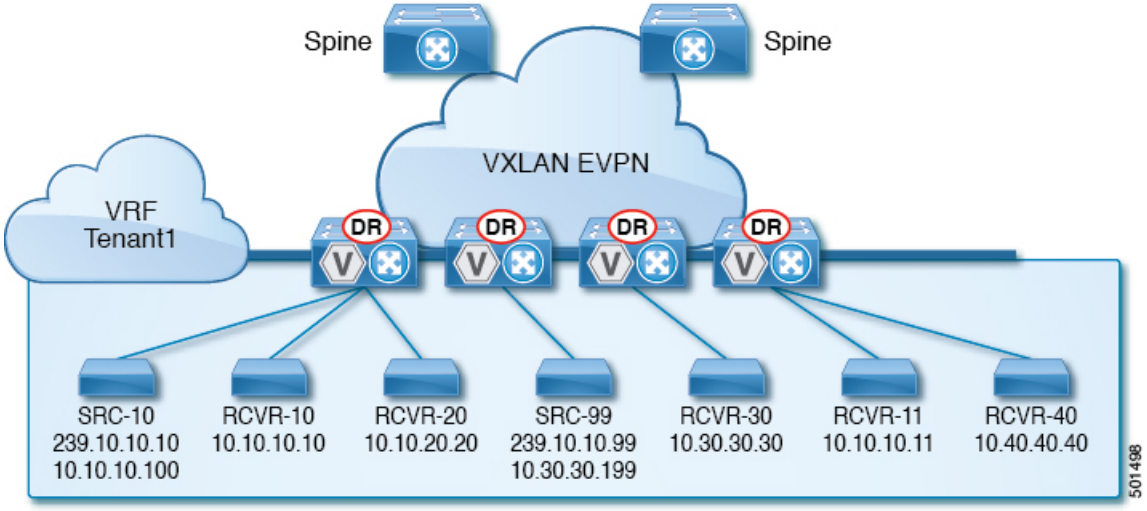
About Tenant Routed Multicast

Tenant Routed Multicast (TRM) enables multicast forwarding on the VXLAN fabric that uses a BGP-based EVPN control plane. TRM provides multi-tenancy aware multicast forwarding between senders and receivers within the same or different subnet local or across VTEPs.

This feature brings the efficiency of multicast delivery to VXLAN overlays. It is based on the standards-based next generation control plane (ngMVPN) described in IETF RFC 6513, 6514. TRM enables the delivery of customer IP multicast traffic in a multitenant fabric, and thus in an efficient and resilient manner. The delivery of TRM improves Layer-3 overlay multicast functionality in our networks.

While BGP EVPN provides the control plane for unicast routing, ngMVPN provides scalable multicast routing functionality. It follows an “always route” approach where every edge device (VTEP) with distributed IP Anycast Gateway for unicast becomes a Designated Router (DR) for Multicast. Bridged multicast forwarding is only present on the edge-devices (VTEP) where IGMP snooping optimizes the multicast forwarding to interested receivers. Every other multicast traffic beyond local delivery is efficiently routed.

Figure 1: VXLAN EVPN TRM



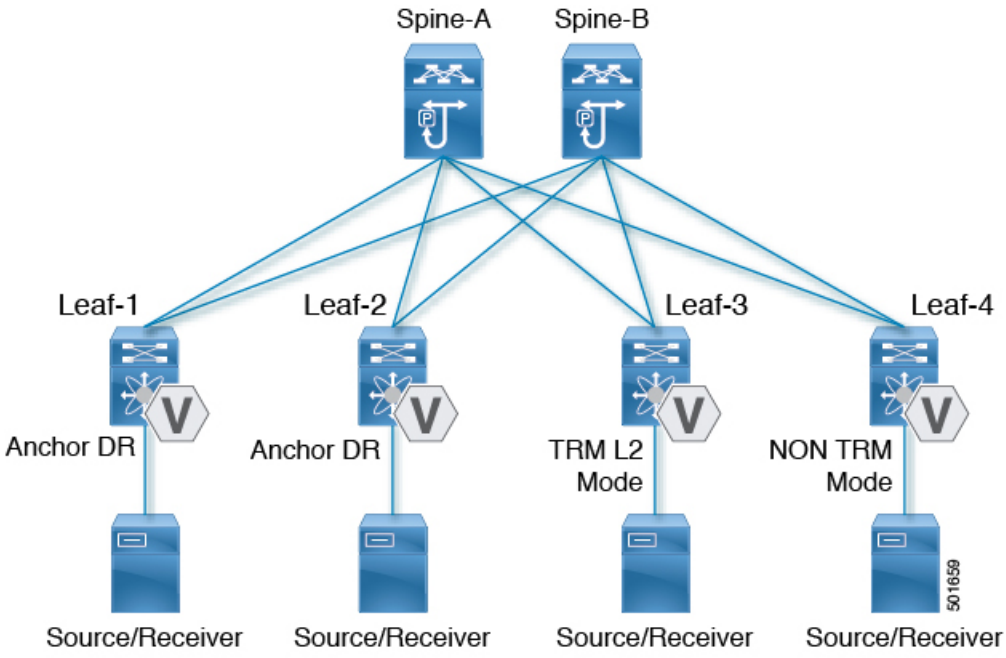
With TRM enabled, multicast forwarding in the underlay is leveraged to replicate VXLAN encapsulated routed multicast traffic. A Default Multicast Distribution Tree (Default-MDT) is built per-VRF. This is an addition to the existing multicast groups for Layer-2 VNI Broadcast, Unknown Unicast, and Layer-2 multicast replication group. The individual multicast group addresses in the overlay are mapped to the respective underlay multicast address for replication and transport. The advantage of using a BGP-based approach allows the VXLAN BGP EVPN fabric with TRM to operate as fully distributed Overlay Rendezvous-Point (RP), with the RP presence on every edge-device (VTEP).

A multicast-enabled data center fabric is typically part of an overall multicast network. Multicast sources, receivers, and multicast rendezvous points, might reside inside the data center but might also be inside the campus or externally reachable via the WAN. TRM allows a seamless integration with existing multicast networks. It can leverage multicast rendezvous points external to the fabric. Furthermore, TRM allows for tenant-aware external connectivity using Layer-3 physical interfaces or subinterfaces.

About Tenant Routed Multicast Mixed Mode

Tenant Routed Multicast (TRM) builds on the Cisco Cloud Scale ASIC enabled Cisco Nexus 9000-EX/-FX Series switches which are capable of VXLAN encapsulated multicast routing. Nevertheless, the solution is backward compatible with earlier generations of Cisco Nexus 9000 Series switches. It provides Distributed Anchor Designated Router (Anchor-DR) functionality to translate between TRM capable and non-TRM capable edge-devices (VTEPs). In this co-existence mode, multicast traffic is partially routed (on the TRM capable devices), but primarily bridged. One or more of these TRM capable edge-devices performs the necessary gateway function between the “two worlds."

Figure 2: TRM Layer 2/Layer 3 Mixed Mode



About Tenant Routed Multicast with IPv6 Overlay

Beginning with Cisco NX-OS Release 10.2(1), Tenant Routed Multicast (TRM) supports IPv6 in the overlay.

Guidelines and Limitations for TRM with IPv6 Overlay

The following are supported by TRM with IPv6 Overlay:

- Multicast IPv4 underlay within fabric. Bidir and SSM are not supported.
- IPv4 Underlay in the data center core for multisite.
- IPv4 overlay only, IPv6 overlay Only, combination of IPv4 and IPv6 overlays
- IPv6 in the underlay
- Anycast Border Gateway with Border Leaf Role

- vPC support on Border Gateway and Leaf
- Virtual MCT on Leaf
- Anycast RP (internal, external, and RP-everywhere)
- Multisite Border Gateway is supported on Cisco Nexus 9300 -FX3, -GX, GX2, -H2R, and -H1 TORs.
- RP-everywhere with Anycast RP is supported.
- TRMv6 is supported only on default system routing mode.
- MLD snooping with VxLAN VLANs with TRM
- PIM6 SVI and MLD snooping configuration on the VLAN are not supported.
- TRM with IPv6 Overlay is supported on Cisco Nexus 9300 -EX, -FX, - FX2, -FX3, -GX, - GX2, -H2R, -H1 TORs.

The following are not supported by TRM with IPv6 Overlay:

- L2 TRM
- VXLAN flood mode on L2 VLANs with L3TRM is not supported
- L2-L3 TRM Mixed Mode
- VXLAN Ingress Replication within a single site
- MLD snooping with VXLAN VLANs without TRM
- PIM6 SVI configuration without MLD snooping
- MSDP

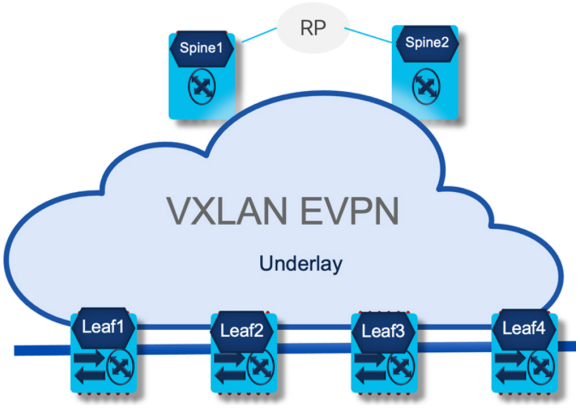
About Multicast Flow Path Visibility for TRM Flows

Beginning with Cisco NX-OS Release 10.3(2)F, the Multicast Flow Path Visualization (FPV) for TRM Flows feature is supported for TRM L3 mode and underlay multicast along with the already supported multicast flows. This feature enables you to export all multicast states in a Cisco Nexus 9000 Series switch. This helps to have a complete and reliable traceability of the flow path from the source to a receiver. To enable Multicast Flow Path Data Export on Cisco Nexus 9000 Series switches, use the **multicast flow-path export** command.

About Configuring VXLAN EVPN and TRM with IPv6 Underlay

Beginning with Cisco NX-OS Release 10.4(2)F, the support is provided for VXLAN with IPv6 Multicast in the Underlay. Hosts in the overlay can be IPv4 or IPv6. This requires IPv6 versions of the unicast routing protocols and using IPv6 multicast in the underlay (PIMv6). Any multi-destination overlay traffic (such as TRM, BUM) can use the IPv6 multicast underlay.

Figure 3: Topology - VXLAN EVPN with IPv6 Multicast Underlay



The above topology shows four leafs and two spines in a VXLAN EVPN fabric. The underlay is an IPv6 Multicast running PIMv6. RP is positioned in the spine with anycast RP.

Beginning with Cisco NX-OS Release 10.4(3)F, the combination of PIMv6 underlay on the fabric side and Ingress Replication (IPv6) on Data Center Interconnect (DCI) side is supported on Cisco Nexus 9300-FX/FX2/FX3/GX/GX2/H2R/H1 ToR switches and 9500 switches with X9716D-GX and X9736C-FX line cards.

Beginning with Cisco NX-OS Release 10.5(1)F, the underlay network supports the following combinations for VXLAN EVPN:

- In the data center fabric, both Multicast Underlay (PIMv6) and Ingress Replication (IPv6) are supported.
- In the Data Center Interconnect (DCI), only Ingress Replication (IPv6) is supported.

Guidelines and Limitations for Tenant Routed Multicast

Tenant Routed Multicast (TRM) has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.1(2), TRM Multisite with vPC BGW is supported.
- Beginning with Cisco NX-OS Release 10.2(1q)F, VXLAN TRM is supported on Cisco Nexus N9K-C9332D-GX2B platform switches.
- Beginning with Cisco NX-OS Release 10.2(3)F, VXLAN TRM is supported on Cisco Nexus 9364D-GX2A, and 9348D-GX2A platform switches.
- Beginning with Cisco NX-OS Release 10.4(1)F, VXLAN TRM is supported on Cisco Nexus 9332D-H2R switches.
- Beginning with Cisco NX-OS Release 10.4(2)F, VXLAN TRM is supported on Cisco Nexus 93400LD-H1 switches.
- Beginning with Cisco NX-OS Release 10.4(3)F, VXLAN TRM is supported on Cisco Nexus 9364C-H1 switches.
- Support straight-through FEX connected to a standalone VXLAN VTEP and multicast source/receiver behind the FEX port.

- With TRM enabled, we do not support a multicast receiver behind active-active FEX and vPC behind straight-through FEX.
- Cisco Nexus 9300-EX, 9300-FX, and 9300-FX2 switches support FEX.
- FEX is not supported on Cisco Nexus 9500 platform switches.
- Support added for Cisco Nexus 2248, 2232, and 2348 Fabric Extenders.
- If VXLAN TRM feature is enabled on a VTEP, it would stop to send IGMP messages to the VXLAN fabric.
- The Guidelines and Limitations for VXLAN also apply to TRM.
- With TRM enabled, SVI as a core link is not supported.
- If TRM is configured, ISSU is disruptive.
- TRM supports IPv4 and IPv6 multicast underlay.
- TRM supports overlay PIM ASM and PIM SSM only. PIM BiDir is not supported in the overlay.
- RP has to be configured either internal or external to the fabric.
- The internal RP must be configured on all TRM-enabled VTEPs including the border nodes.
- The external RP must be external to the border nodes.
- The RP must be configured within the VRF pointing to the external RP IP address (static RP). This ensures that unicast and multicast routing is enabled to reach the external RP in the given VRF.
- In a Transit Routing Multicast (TRM) deployment, the RP-on-stick model can sometimes lead to traffic drops if there is flapping on the Protocol Independent Multicast (PIM) enabled interface. Use the **ip pim spt-switch-graceful** command on the turnaround router that leads to the RP. This command allows for a graceful switch to the Shortest Path Tree (SPT) during flapping, which can minimize traffic drops.
- Replication of first packet is supported only on Cisco Nexus 9300 – EX, FX, FX2 family switches.
- Beginning with Cisco NX-OS Release 10.2(3)F, Replication of first packet is supported on the Cisco Nexus 9300-FX3 platform switches.
- TRM with Multi-Site is not supported on Cisco Nexus 9504-R platforms.
- TRM supports multiple border nodes. Reachability to an external RP/source via multiple border leaf switches is supported with ECMP and requires symmetric unicast routing.
- Both PIM and **ip igmp snooping vxlan** must be enabled on the L3 VNI's VLAN in a VXLAN vPC setup.
- For traffic streams with an internal source and external L3 receiver using an external RP, the external L3 receiver might send PIM S,G join requests to the internal source. Doing so triggers the recreation of S,G on the fabric FHR, and it can take up to 10 minutes for this S,G to be cleared.
- Beginning with Cisco NX-OS Release 10.3(1)F, the Real-time/flex statistics for TRM is supported on Cisco Nexus 9300-X Cloud Scale Switches.
- TRM supports vPC fabric peering leaf’s as well as vPC/Anycast BGW.

VXLAN with TRM Upgrade limitations

Caution Following changes must be done during a Maintenance window.

After upgrading a Cisco NX-OS 9000 Series switches configured with VXLAN (specifically VRF-related configurations) from Cisco NX-OS Release 7.x through 9.3 to 10.3(6) or earlier, two issues arise:

- The startup-config displays both legacy and new Layer 3 VNID configuration modes
- TRM traffic's RPF changes to the new mode for S,Gs, causing multicast traffic forwarding problems.

To avoid these issues, follow these steps:

- Enable the REST configuration input using the following commands:

```
feature nxapi
            nxapi http port 80
```

- Open a browser and enter the management IP address of the switch. This will open the Sandbox page. Use the same credentials as the switch admin login to sign in.
- In the top input textbox, enter the following command for each VRF that has an issue with the VNI ID:

```
vrf context tenant-1
        no vni 50000 l3
```

- On the right side of the page, set the Method to **NXAPI-REST(DME)** and keep the Input Type as **cli**.
- Click the **Convert (with DN)** button in the middle of the page. This will generate the XML equivalent of the configuration change.
- When the XML appears in the second textbox, click **Send** to apply the changes and remove the VNI ID configuration from the switch.
- To ensure the changes are applied, run the command:

```
copy running-config startup-config
```

Guidelines and Limitations for Layer 3 Tenant Routed Multicast

Layer 3 Tenant Routed Multicast (TRM) has the following configuration guidelines and limitations:

- When upgrading from Cisco NX-OS Release 9.3(3) to Cisco NX-OS Release 9.3(6), if you do not retain configurations of the TRM enabled VRFs from Cisco NX-OS Release 9.3(3), or if you create new VRFs after the upgrade, the auto-generation of **ip multicast multipath s-g-hash next-hop-based** CLI, when **feature ngmvpn** is enabled, will not happen. You must enable the CLI manually for each TRM enabled VRF.
- Layer 3 TRM is supported for Cisco Nexus 9200, 9300-EX, and 9300-FX/FX2/FX3/FXP and 9300-GX platform switches.
- Beginning with Cisco NX-OS Release 10.2(3)F, Layer 3 TRM is supported on the Cisco Nexus 9300-GX2 platform switches.
- Beginning with Cisco NX-OS Release 10.4(1)F, Layer 3 TRM is supported on the Cisco Nexus 9332D-H2R switches.

- Beginning with Cisco NX-OS Release 10.4(2)F, Layer 3 TRM is supported on the Cisco Nexus 93400LD-H1 switches.
- Beginning with Cisco NX-OS Release 10.4(3)F, Layer 3 TRM is supported on the Cisco Nexus 9364C-H1 switches.
- Beginning with Cisco NX-OS Release 9.3(7), Cisco Nexus N9K-C9316D-GX, N9K-C9364C-GX, and N9K-X9716D-GX platform switches support the combination of Layer 3 TRM and EVPN Multi-Site.
- Cisco Nexus 9300-GX platform switches do not support the combination of Layer 3 TRM and EVPN Multi-Site in Cisco NX-OS Release 9.3(5).
- Beginning with Cisco NX-OS Release 10.2(3)F, the combination of Layer 3 TRM and EVPN Multi-Site is supported on the Cisco Nexus 9300-GX2 platform switches.
- Beginning with Cisco NX-OS Release 10.4(1)F, the combination of Layer 3 TRM and EVPN Multi-Site is supported on the Cisco Nexus 9332D-H2R switches.
- Beginning with Cisco NX-OS Release 10.4(2)F, the combination of Layer 3 TRM and EVPN Multi-Site is supported on the Cisco Nexus 93400LD-H1 switches.
- Beginning with Cisco NX-OS Release 10.4(3)F, the combination of Layer 3 TRM and EVPN Multi-Site is supported on the Cisco Nexus 9364C-H1 switches.
- Beginning with Cisco NX-OS Release 9.3(3), the Cisco Nexus 9504 and 9508 platform switches with -R/RX line cards support TRM in Layer 3 mode. This feature is supported on IPv4 overlays only. Layer 2 mode and L2/L3 mixed mode are not supported.

  The Cisco Nexus 9504 and 9508 platform switches with -R/RX line cards can function as a border leaf for Layer 3 unicast traffic. For Anycast functionality, the RP can be internal, external, or RP everywhere.
- When configuring TRM VXLAN BGP EVPN, the following platforms are supported:
  - Cisco Nexus 9200, 9332C, 9364C, 9300-EX, and 9300-FX/FX2/FX3/FXP platform switches.
  - Cisco Nexus 9300-GX/GX2 platform switches.
  - Cisco Nexus 9300-H2R/H1 platform switches.
  - Cisco Nexus 9500 platform switches with 9700-EX line cards, 9700-FX line cards, 9700-FX3 line cards.
- Layer 3 TRM and VXLAN EVPN Multi-Site are supported on the same physical switch. For more information, see Configuring VXLAN EVPN Multi-Site.
- TRM Multi-Site functionality is not supported on Cisco Nexus 9504 platform switches with -R/RX line cards.
- If one or both VTEPs is a Cisco Nexus 9504 or 9508 platform switch with -R/RX line cards, the packet TTL is decremented twice, once for routing to the L3 VNI on the source leaf and once for forwarding from the destination L3 VNI to the destination VLAN on the destination leaf.
- TRM with vPC border leafs is supported only for Cisco Nexus 9200, 9300-EX, and 9300-FX/FX2/FX3/GX/GX2/H2R/H1 platform switches and Cisco Nexus 9500 platform switches with -EX/FX/FX3 or -R/RX line cards. The **advertise-pip** and **advertise virtual-rmac** commands must be enabled on the border leafs to support this functionality. For configuration information, see the "Configuring VIP/PIP" section.

- To support any Layer 3 source behind one of the vPC peers, whether physical or virtual MCT, a physical link configured as VRF-lite is required between the vPC peers. This setup is necessary to accommodate a receiver located behind the vPC peer, especially if it is the sole receiver in the fabric. This requirement applies to all scenarios where the vPC functions as a BGW, border Leaf, or an internal Leaf.

  On the receiving vPC peer, the VRF-lite link must have a superior reachability metric to the L3 source compared to any other paths (iBGP or eBGP) to be selected as the RPF towards the L3 source. In this configuration, traffic will flow directly to the receiver without traversing the EVPN fabric.

- Well-known local scope multicast (224.0.0.0/24) is excluded from TRM and is bridged.
- When an interface NVE is brought down on the border leaf, the internal overlay RP per VRF must be brought down.
- Beginning with Cisco NX-OS Release 10.3(1)F, TRM support for the new L3VNI mode CLIs are provided on Cisco Nexus 9300-X Cloud Scale switches.
- Beginning Cisco NXOS release 10.2(1)F, TRM Flow Path Visualization is supported for flows within a single VXLAN EVPN site.
- Beginning Cisco NXOS Release 10.3(2)F, TRM Flow Path Visualization support has been extended to below traffic patterns on Cisco Nexus 9000 Series platform switches:
  - TRM Multisite DCI Multicast
  - TRM Multisite DCI IR
  - TRM Data MDT
  - TRM on Virtual MCT vPC
  - TRM using new L3VNI
  - BUM Traffic visibility is not supported.
- Beginning with Cisco NX-OS Release 10.4(3)F, the TRM Multi-Site Anycast BGW on Cisco Nexus 9808/9804 switches with Cisco Nexus X9836DM-A and X98900CD-A line cards support the following features:
  - TRMv4
  - Ingress Replication between DCI peers across the core
  - Multicast underlay for fabric peers.
  - Only new L3VNI mode is supported. However, the traditional L3VNI mode is not supported

  TRM Multi-Site Anycast BGW on Cisco Nexus 9808/9804 switches with Cisco Nexus X9836DM-A and X98900CD-A line cards do not support the following features:
  - TRMv6
  - Data MDT
  - Multicast underlay between DCI peers across the core is not supported.

Guidelines and Limitations for Layer 2/Layer 3 Tenant Routed Multicast (Mixed Mode)

Layer 2/Layer 3 Tenant Routed Multicast (TRM) has the following configuration guidelines and limitations:

- All TRM Layer 2/Layer 3 configured switches must be Anchor DR. This is because in TRM Layer 2/Layer 3, you can have switches configured with TRM Layer 2 mode that co-exist in the same topology. This mode is necessary if non-TRM and Layer 2 TRM mode edge devices (VTEPs) are present in the same topology.
- Anchor DR is required to be an RP in the overlay.
- An extra loopback is required for anchor DRs.
- Non-TRM and Layer 2 TRM mode edge devices (VTEPs) require an IGMP snooping querier configured per multicast-enabled VLAN. Every non-TRM and Layer 2 TRM mode edge device (VTEP) requires this IGMP snooping querier configuration because in TRM multicast control-packets are not forwarded over VXLAN.
- The IP address for the IGMP snooping querier can be re-used on non-TRM and Layer 2 TRM mode edge devices (VTEPs).
- The IP address of the IGMP snooping querier in a VPC domain must be different on each VPC member device.
- When interface NVE is brought down on the border leaf, the internal overlay RP per VRF should be brought down.
- The NVE interface must be shut and unshut while configuring the **ip multicast overlay-distributed-dr** command.
- Beginning with Cisco NX-OS Release 9.2(1), TRM with vPC border leafs is supported. Advertise-PIP and Advertise Virtual-Rmac need to be enabled on border leafs to support with functionality. For configuring advertise-pip and advertise virtual-rmac, see the "Configuring VIP/PIP" section.
- Anchor DR is supported only on the following hardware platforms:
  - Cisco Nexus 9200, 9300-EX, and 9300-FX/FX2 platform switches
  - Cisco Nexus 9500 platform switches with 9700-EX/FX/FX3 line cards.
- Beginning with Cisco NX-OS Release 10.2(3)F, Anchor DR is supported on the Cisco Nexus 9300-FX3 platform switches.
- Layer 2/Layer 3 Tenant Routed Multicast (TRM) is not supported on Cisco Nexus 9300-FX3/GX/GX2/H2R/H1 platform switches.

Guidelines and Limitations for VXLAN EVPN and TRM with IPv6 in the Multicast Underlay

VXLAN EVPN and TRM with IPv6 Multicast Underlay has the following guidelines and limitations:

- Spine-based static RP is supported in underlay.
- Cisco Nexus 9300-FX, FX2, FX3, GX, GX2, H2R, and H1 ToR switches are supported as the leaf VTEP.
- Cisco Nexus X9716D-GX and X9736C-FX line cards are supported only on the spine (EoR).
- When an EoR is deployed as a spine node with Multicast Underlay (PIMv6) Any-Source Multicast (ASM), it in mandatory to configure non-default template using one of the following commands in global configuration mode:
  - **system routing template-multicast-heavy**
  - **system routing template-multicast-ext-heavy**
- OSPFv3, ISIS, eBGP underlay is supported.
- PIMv6 ASM (sparse mode) is supported in underlay.
- PIMv6 Anycast RP is supported in underlay as RP redundancy.
- Underlay IPv6 Multicast is supported.
- Underlay IPv6 Multicast is not supported on EOR platforms as a leaf.
- For overlay traffic, each Cisco Nexus 9000 leaf switch is an RP. External RP is also supported.
- EVPN TRMv4 and TRMv6 with IPv6 Multicast Underlay are supported on the Fabric.
- Fabric Peering and Multisite are not supported with IPv6 multicast underlay.
- The global mcast-group under NVE should not be configured as SSM range, and vice versa. If there is no explicit SSM configuration, then 232/8 is the default in data plane. hence 232.0.0.0/8 should not be configured as SSM and vice versea.
- GPO is not supported with IPv6 multicast underlay.
- For EVPN TRMv4 and TRMv6 with IPv6 Multicast Underlay, the TCAM region for ingress sup region must be carved to 768.
  - Check the ingress sup region using **show hardware access-list tcam region** command.
  - If the ingress sup region is not 768 or above, you must configure using the **hardware access-list tcam region ing-sup 768** command.

Note If you get an error, “Aggregate ingress TCAM allocation failure” while configuring ing-sup as 768, you must borrow the amount from other TCAM regions.

  - Reload the device after this configuration.
- Beginning with Cisco NX-OS Release 10.5(1)F, VXLAN EVPN in the data center fabric supports both Multicast Underlay (PIMv6) Any-Source Multicast (ASM) and Ingress Replication (IPv6) in the underlay. This support is available on the following switches and line cards:
  - Cisco Nexus 9300-FX, FX2, FX3, GX, GX2, H2R, and H1 ToR switches as the leaf VTEPs.

- Cisco Nexus N9K-X9716D-GX and N9K-X9736C-FX line cards as spines if the underlay is configured for Multicast Underlay (PIMv6) Any-Source Multicast (ASM).
- Cisco Nexus N9K-X9716D-GX and N9K-X9736C-FX line cards as VTEPs if the underlay uses Ingress Replication (IPv6).

Rendezvous Point for Tenant Routed Multicast

With TRM enabled Internal and External RP is supported. The following table displays the first release in which RP positioning is or is not supported.

	RP Internal	RP External	PIM-Based RP Everywhere
TRM L2 Mode	N/A	N/A	N/A
TRM L3 Mode	7.0(3)I7(1), 9.2(x)	7.0(3)I7(4), 9.2(3)	Supported in 7.0(3)I7(x) releases starting from 7.0(3)I7(5) Not supported in 9.2(x) Supported in NX-OS releases beginning with 9.3(1) for the following Nexus 9000 switches: • Cisco Nexus 9200 Series switches • Cisco Nexus 9364C platform switches • Cisco Nexus 9300-EX/FX/FX2 platform switches (excluding the Cisco Nexus 9300-FXP platform switch) Supported for Cisco Nexus 9300-FX3 platform switches beginning with Cisco NX-OS Release 9.3(5)
TRM L2L3 Mode	7.0(3)I7(1), 9.2(x)	N/A	N/A

	RP Internal	RP External
TRM L2 Mode	N/A	N/A

	RP Internal	RP External
TRM L3 Mode	7.0(3)I7(1)	7.0(3)I7(4)
TRM L2L3 Mode	7.0(3)I7(1)	N/A

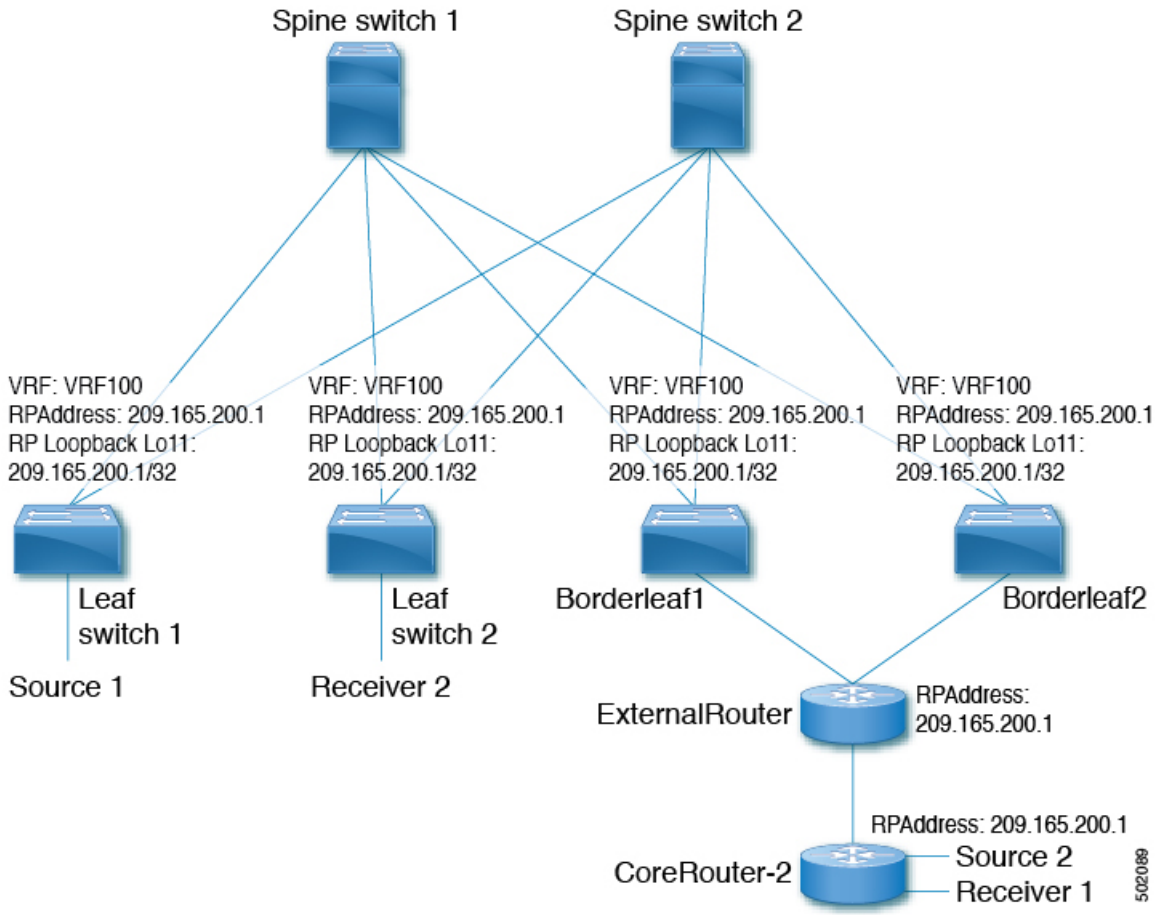
Configuring a Rendezvous Point for Tenant Routed Multicast

For Tenant Routed Multicast, the following rendezvous point options are supported:

- Configuring a Rendezvous Point Inside the VXLAN Fabric, on page 13
- Configuring an External Rendezvous Point, on page 15
- Configuring RP Everywhere with PIM Anycast, on page 17
- Configuring RP Everywhere with MSDP Peering, on page 23

Configuring a Rendezvous Point Inside the VXLAN Fabric

Configure the loopback for the TRM VRFs with the following commands on all devices (VTEP). Ensure it is reachable within EVPN (advertise/redistribute).



SUMMARY STEPS

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vxlan-number*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **vrf context** *vrf-name*
7. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

DETAILED STEPS

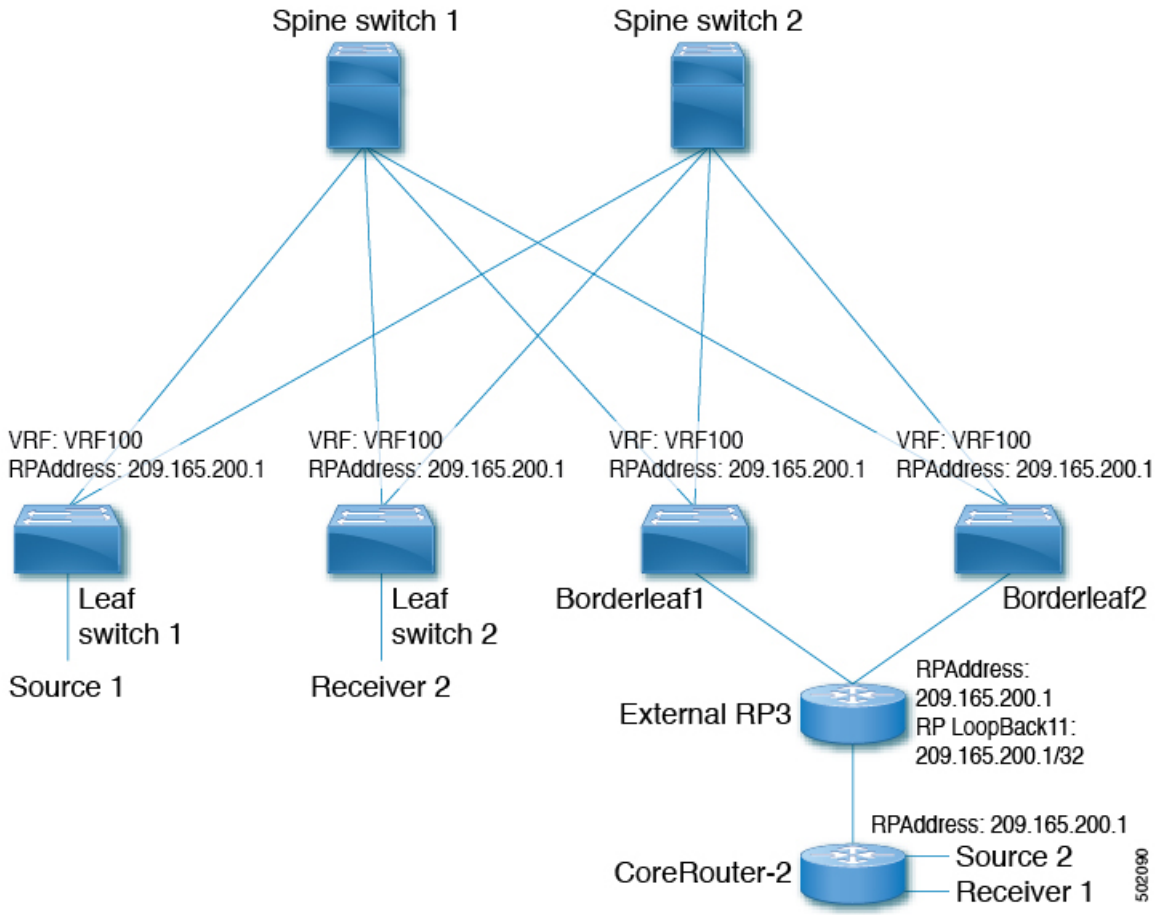
Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all TRM-enabled nodes. This enables the rendezvous point inside the fabric.
Step 3	**vrf member** *vxlan-number* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 4	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.1/32`	Specify IP address.
Step 5	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 6	**vrf context** *vrf-name* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.
Step 7	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameter is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.

Configuring an External Rendezvous Point

Configure the external rendezvous point (RP) IP address within the TRM VRFs on all devices (VTEP). In addition, ensure reachability of the external RP within the VRF via the border node. With TRM enabled and an external RP in use, ensure that only one routing path is active. Routing between the TRM fabric and the external RP must be via a single border leaf (non ECMP).



SUMMARY STEPS

1. **configure terminal**
2. **vrf context vrf100**
3. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

DETAILED STEPS

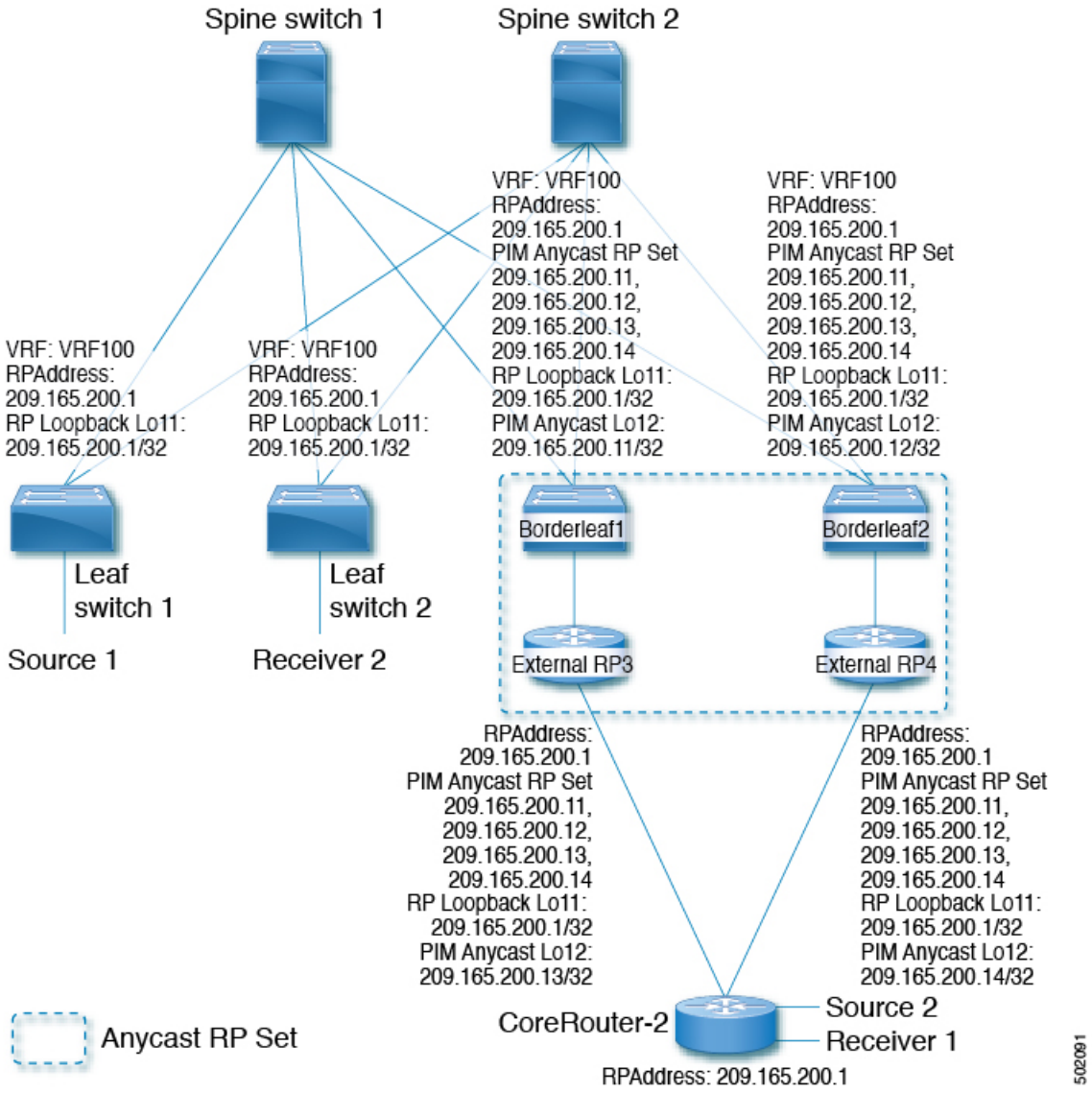
Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**vrf context vrf100** **Example:** `switch(config)# vrf context vrf100`	Enter configuration mode.

	Command or Action	Purpose
Step 3	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameter is that of the RP. The same IP address must be on all of the edge devices (VTEPs) for a fully distributed RP.

Configuring RP Everywhere with PIM Anycast

RP Everywhere configuration with PIM Anycast solution.



For information about configuring RP Everywhere with PIM Anycast, see:

Configuring a TRM Leaf Node for RP Everywhere with PIM Anycast

Configuration of Tenant Routed Multicast (TRM) leaf node for RP Everywhere.

SUMMARY STEPS

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vrf-name*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **vrf context** *vxlan*
7. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all VXLAN VTEP devices.
Step 3	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 4	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.1/32`	Specify IP address.
Step 5	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.

	Command or Action	Purpose
Step 6	**vrf context** *vxlan* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.
Step 7	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameters is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.

Configuring a TRM Border Leaf Node for RP Everywhere with PIM Anycast

Configuring the TRM Border Leaf Node for RP Anywhere with PIM Anycast.

SUMMARY STEPS

1. **configure terminal**
2. **ip pim evpn-border-leaf**
3. **interface loopback** *loopback_number*
4. **vrf member** *vrf-name*
5. **ip address** *ip-address*
6. **ip pim sparse-mode**
7. **interface loopback** *loopback_number*
8. **vrf member** *vxlan-number*
9. **ip address** *ip-address*
10. **ip pim sparse-mode**
11. **vrf context** *vrf-name*
12. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
13. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
14. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
15. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
16. **ip pim anycast-rp** *anycast-rp-address address-of-rp*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.

	Command or Action	Purpose
Step 2	**ip pim evpn-border-leaf** **Example:** `switch(config)# ip pim evpn-border-leaf`	Configure VXLAN VTEP as TRM border leaf node,
Step 3	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all VXLAN VTEP devices.
Step 4	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 5	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.1/32`	Specify IP address.
Step 6	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 7	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 12`	Configure the PIM Anycast set RP loopback interface.
Step 8	**vrf member** *vxlan-number* **Example:** `switch(config-if)# vrf member vxlan-number`	Configure VRF name.
Step 9	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.11/32`	Specify IP address.
Step 10	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 11	**vrf context** *vrf-name* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.
Step 12	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:**	The value of the *ip-address-of-router* parameters is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.

	Command or Action	Purpose
	`switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	
Step 13	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.11`	Configure PIM Anycast RP set.
Step 14	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.12`	Configure PIM Anycast RP set.
Step 15	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.13`	Configure PIM Anycast RP set.
Step 16	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.14`	Configure PIM Anycast RP set.

Configuring an External Router for RP Everywhere with PIM Anycast

Use this procedure to configure an external router for RP Everywhere.

SUMMARY STEPS

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vrf-name*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **interface loopback** *loopback_number*
7. **vrf member** *vxlan-number*
8. **ip address** *ip-address*
9. **ip pim sparse-mode**
10. **vrf context** *vxlan*
11. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
12. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
13. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
14. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
15. **ip pim anycast-rp** *anycast-rp-address address-of-rp*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all VXLAN VTEP devices.
Step 3	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vfr100`	Configure VRF name.
Step 4	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.1/32`	Specify IP address.
Step 5	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 6	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 12`	Configure the PIM Anycast set RP loopback interface.
Step 7	**vrf member** *vxlan-number* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 8	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.13/32`	Specify IP address.
Step 9	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 10	**vrf context** *vxlan* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.

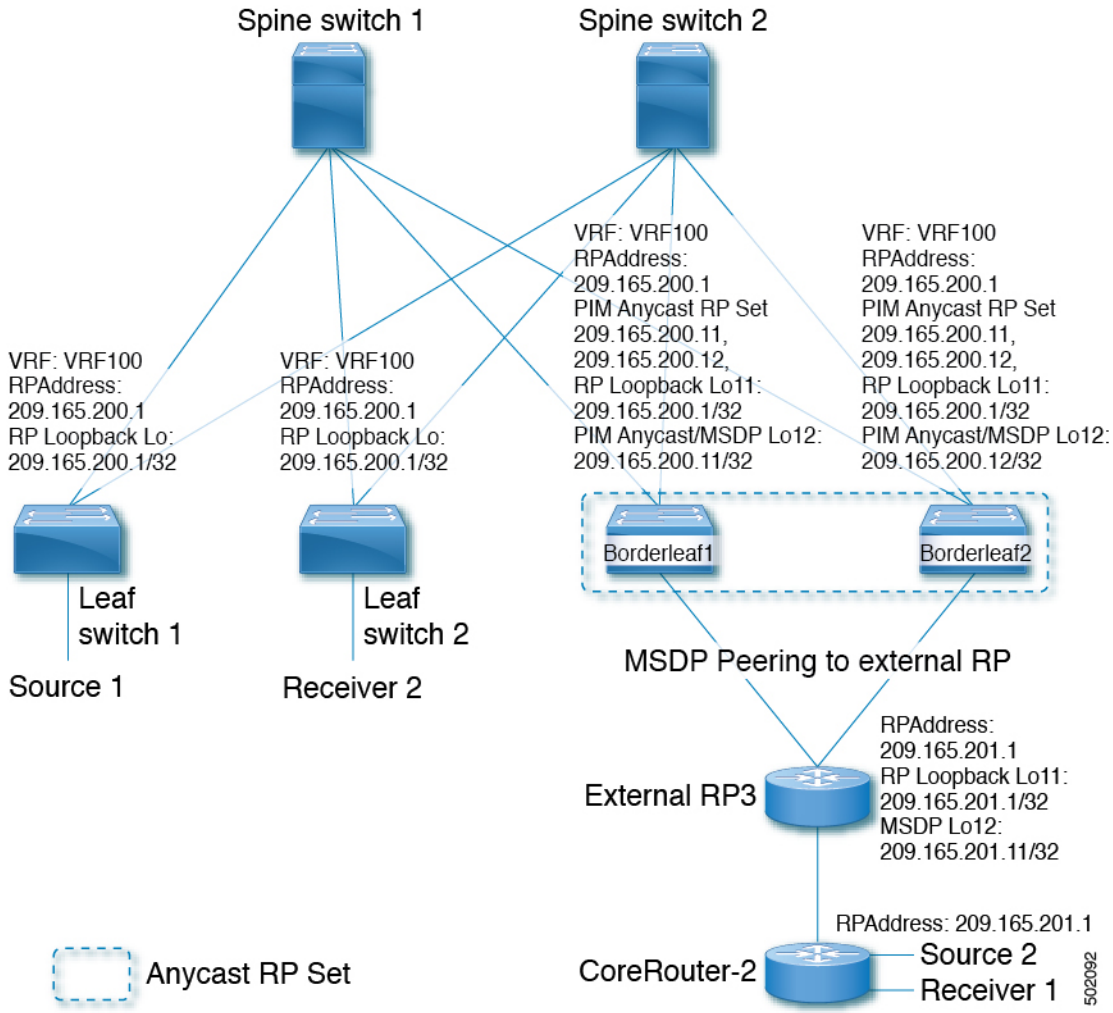
	Command or Action	Purpose
Step 11	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameters is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.
Step 12	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.11`	Configure PIM Anycast RP set.
Step 13	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.12`	Configure PIM Anycast RP set.
Step 14	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.13`	Configure PIM Anycast RP set.
Step 15	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.14`	Configure PIM Anycast RP set.

Configuring RP Everywhere with MSDP Peering

The following figure represents the RP Everywhere configuration with MSDP RP solution.

For information about configuring RP Everywhere with MSDP Peering, see:

- Configuring a TRM Leaf Node for RP Everywhere with MSDP Peering, on page 24
- Configuring a TRM Border Leaf Node for RP Everywhere with MSDP Peering, on page 25
- Configuring an External Router for RP Everywhere with MSDP Peering, on page 28



Configuring a TRM Leaf Node for RP Everywhere with MSDP Peering

Configuring a TRM leaf node for RP Everywhere with MSDP peering.

SUMMARY STEPS

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vrf-name*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **vrf context** *vrf-name*
7. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all VXLAN VTEP devices.
Step 3	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 4	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.1/32`	Specify IP address.
Step 5	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 6	**vrf context** *vrf-name* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.
Step 7	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameters is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.

Configuring a TRM Border Leaf Node for RP Everywhere with MSDP Peering

Use this procedure to configure a TRM border leaf for RP Everywhere with PIM Anycast.

SUMMARY STEPS

1. **configure terminal**
2. **feature msdp**
3. **ip pim evpn-border-leaf**
4. **interface loopback** *loopback_number*

5. **vrf member** *vrf-name*
6. **ip address** *ip-address*
7. **ip pim sparse-mode**
8. **interface loopback** *loopback_number*
9. **vrf member** *vrf-name*
10. **ip address** *ip-address*
11. **ip pim sparse-mode**
12. **vrf context** *vrf-name*
13. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
14. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
15. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
16. **ip msdp originator-id** *loopback*
17. **ip msdp peer** *ip-address* **connect-source** *loopback*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**feature msdp** **Example:** `switch(config)# feature msdp`	Enable feature MSDP.
Step 3	**ip pim evpn-border-leaf** **Example:** `switch(config)# ip pim evpn-border-leaf`	Configure VXLAN VTEP as TRM border leaf node,
Step 4	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all VXLAN VTEP devices.
Step 5	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 6	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.1/32`	Specify IP address.
Step 7	**ip pim sparse-mode** **Example:**	Configure sparse-mode PIM on an interface.

	Command or Action	Purpose
	`switch(config-if)# ip pim sparse-mode`	
Step 8	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 12`	Configure the PIM Anycast set RP loopback interface.
Step 9	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 10	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.200.11/32`	Specify IP address.
Step 11	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 12	**vrf context** *vrf-name* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.
Step 13	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameter is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.
Step 14	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.11`	Configure PIM Anycast RP set.
Step 15	**ip pim anycast-rp** *anycast-rp-address address-of-rp* **Example:** `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.12`	Configure PIM Anycast RP set.
Step 16	**ip msdp originator-id** *loopback* **Example:** `switch(config-vrf)# ip msdp originator-id loopback12`	Configure MSDP originator ID.
Step 17	**ip msdp peer** *ip-address* **connect-source** *loopback* **Example:**	Configure MSDP peering between border node and external RP router.

	Command or Action	Purpose
	`switch(config-vrf)# ip msdp peer 209.165.201.11 connect-source loopback12`	

Configuring an External Router for RP Everywhere with MSDP Peering

SUMMARY STEPS

1. **configure terminal**
2. **feature msdp**
3. **interface loopback** *loopback_number*
4. **vrf member** *vrf-name*
5. **ip address** *ip-address*
6. **ip pim sparse-mode**
7. **interface loopback** *loopback_number*
8. **vrf member** *vrf-name*
9. **ip address** *ip-address*
10. **ip pim sparse-mode**
11. **vrf context** *vrf-name*
12. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
13. **ip msdp originator-id loopback12**
14. **ip msdp peer** *ip-address* **connect-source loopback12**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**feature msdp** **Example:** `switch(config)# feature msdp`	Enable feature MSDP.
Step 3	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 11`	Configure the loopback interface on all VXLAN VTEP devices.
Step 4	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.

	Command or Action	Purpose
Step 5	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.201.1/32`	Specify IP address.
Step 6	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 7	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 12`	Configure the PIM Anycast set RP loopback interface.
Step 8	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 9	**ip address** *ip-address* **Example:** `switch(config-if)# ip address 209.165.201.11/32`	Specify IP address.
Step 10	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on an interface.
Step 11	**vrf context** *vrf-name* **Example:** `switch(config-if)# vrf context vrf100`	Create a VXLAN tenant VRF.
Step 12	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf)# ip pim rp-address 209.165.201.1 group-list 224.0.0.0/4`	The value of the *ip-address-of-router* parameters is that of the RP. The same IP address must be on all the edge devices (VTEPs) for a fully distributed RP.
Step 13	**ip msdp originator-id loopback12** **Example:** `switch(config-vrf)# ip msdp originator-id loopback12`	Configure MSDP originator ID.
Step 14	**ip msdp peer** *ip-address* **connect-source loopback12** **Example:** `switch(config-vrf)# ip msdp peer 209.165.200.11 connect-source loopback12`	Configure MSDP peering between external RP router and all TRM border nodes.

Configuring Layer 3 Tenant Routed Multicast

This procedure enables the Tenant Routed Multicast (TRM) feature. TRM operates primarily in the Layer 3 forwarding mode for IP multicast by using BGP MVPN signaling. TRM in Layer 3 mode is the main feature and the only requirement for TRM enabled VXLAN BGP EVPN fabrics. If non-TRM capable edge devices (VTEPs) are present, the Layer 2/Layer 3 mode and Layer 2 mode have to be considered for interop.

To forward multicast between senders and receivers on the Layer 3 cloud and the VXLAN fabric on TRM vPC border leafs, the VIP/PIP configuration must be enabled. For more information, see Configuring VIP/PIP.

Note TRM follows an always-route approach and hence decrements the Time to Live (TTL) of the transported IP multicast traffic.

Before you begin

VXLAN EVPN **feature nv overlay** and **nv overlay evpn** must be configured.

The rendezvous point (RP) must be configured.

To enable/disable TRM v4/v6, PIM v4/v6 must be enabled.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**feature ngmvpn** **Example:** `switch(config)# feature ngmvpn`	Enables the Next-Generation Multicast VPN (ngMVPN) control plane. New address family commands become available in BGP. **Note** The **no feature ngmvpn** command will not remove MVPN configuration under BGP. You will get a syslog message when you enable this command. The message informs you that **ip multicast multipath s-g-hash next-hop-based** is the recommended multipath hashing algorithm and you need enable it for the TRM enabled VRFs. The auto-generation of **ip multicast multipath s-g-hash next-hop-based** command does not happen after you enable the **feature ngmvpn** command. You need to configure **ip multicast multipath s-g-hash next-hop-based** as part of the VRF configuration.

	Command or Action	Purpose
Step 3	**ip igmp snooping vxlan** **Example:** `switch(config)# ip igmp snooping vxlan`	Configure IGMP snooping for VXLAN VLANs.
Step 4	**interface nve1** **Example:** `switch(config)# interface nve 1`	Configure the NVE interface.
Step 5	**member vni** *vni-range* **associate-vrf** **Example:** `switch(config-if-nve)# member vni 200100 associate-vrf`	Configure the Layer 3 virtual network identifier. The range of *vni-range* is from 1 to 16,777,214.
Step 6	**mcast-group** *ip-prefix* **Example:** `switch(config-if-nve-vni)# mcast-group 225.3.3.3`	Builds the default multicast distribution tree for the VRF VNI (Layer 3 VNI). The multicast group is used in the underlay (core) for all multicast routing within the associated Layer 3 VNI (VRF). **Note** We recommend that underlay multicast groups for Layer 2 VNI, default MDT, and data MDT not be shared. Use separate, non-overlapping groups.
Step 7	**exit** **Example:** `switch(config-if-nve-vni)# exit`	Exits command mode.
Step 8	**exit** **Example:** `switch(config-if)# exit`	Exits command mode.
Step 9	**router bgp <as-number>** **Example:** `switch(config)# router bgp 100`	Set autonomous system number.
Step 10	**neighbor** *ip-addr* **Example:** `switch(config-router)# neighbor 1.1.1.1`	Configure IP address of the neighbor.
Step 11	**address-family ipv4 mvpn** **Example:** `switch(config-router-neighbor)# address-family ipv4 mvpn`	Configure multicast VPN.

	Command or Action	Purpose
Step 12	**send-community extended** **Example:** `switch(config-router-neighbor-af)# send-community extended`	Enables ngMVPN for address family signalization. The **send community extended** command ensures that extended communities are exchanged for this address family.
Step 13	**exit** **Example:** `switch(config-router-neighbor-af)# exit`	Exits command mode.
Step 14	**exit** **Example:** `switch(config-router-neighbor)# exit`	Exits command mode.
Step 15	**vrf context** *vrf_name* **Example:** `switch(config-router)# vrf context vrf100`	Configures VRF name.
Step 16	**vni** *number* **Example:** `switch(config-router)# vni 500001 l3`	Specifies the VNI for the tenant VRF. Beginning with Cisco NX-OS Release 10.3(1)F, the **L3** keyword is provided to indicate that the new L3VNI configuration is enabled. Beginning with Cisco NX-OS Release 10.4(3)F, this command with **L3** option is supported on Cisco Nexus 9808/9804 switches with Cisco Nexus X9836DMA and X98900CD-A line cards.
Step 17	**mvpn vri id** <*id*> **Example:** `switch(config-router)#mvpn vri 100`	Generates the VRI for TRM. Run this command under router bgp <as-number> submode. The vri id range is from 1 to 65535. **Note** This command is mandatory on vPC leaf nodes, and value has to be same across vPC pair and unique in TRM domain. Also the value must not collide with any site-id value. **Note** This command is required on BGWs if site-id value is greater than 2 bytes, and value has to be same across all same site BGWs and unique in TRM domain. Also the value must not collide with any site-id value.
Step 18	**[no] mdt** [*v4*|*v6*] vxlan **Example:**	Enables TRM v4/v6 on the specified VRF. The TRM v4/v6 is enabled by default.

	Command or Action	Purpose
	`switch(config-router)#mdt v4 vxlan`	The **no** option disables the TRM v4/v6 on the specified VRF. Run this command under the sub-mode of new L3VNI config. **Note** This command is applicable only to VRFs configured with new-L3VNI.
Step 19	**ip multicast multipath s-g-hash next-hop-based** **Example:** `switch(config-vrf)# ip multicast multipath s-g-hash next-hop-based`	Configures multicast multipath and initiates S, G, nexthop hashing (rather than the default of S/RP, G-based hashing) to select the RPF interface.
Step 20	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf)# ip pim rp-address 209.165.201.1 group-list 226.0.0.0/8`	The value of the *ip-address-of-router* parameter is that of the RP. The same IP address must be on all of the edge devices (VTEPs) for a fully distributed RP. For overlay RP placement options, see the Configuring a Rendezvous Point for Tenant Routed Multicast, on page 13 section.
Step 21	**address-family ipv4 unicast** **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Configures unicast address family.
Step 22	**route-target both auto mvpn** **Example:** `switch(config-vrf-af-ipv4)# route-target both auto mvpn`	Defines the BGP route target that is added as an extended community attribute to the customer multicast (C_Multicast) routes (ngMVPN route type 6 and 7). Auto route targets are constructed by the 2-byte Autonomous System Number (ASN) and Layer 3 VNI.
Step 23	**ip multicast overlay-spt-only** **Example:** `switch(config)# ip multicast overlay-spt-only`	Gratuitously originate (S,A) route when the source is locally connected. The **ip multicast overlay-spt-only** command is enabled by default on all MVPN-enabled Cisco Nexus 9000 Series switches (typically leaf node).
Step 24	**interface***vlan_id* **Example:** `switch(config)# interface vlan11`	Configures the first-hop gateway (distributed anycast gateway for the Layer 2 VNI. No router PIM peering must ever happen with this interface.
Step 25	**no shutdown** **Example:** `switch(config-if)# no shutdown`	Disables an interface.
Step 26	**vrf member** *vrf-num* **Example:** `switch(config-if)# vrf member vrf100`	Configures VRF name.

	Command or Action	Purpose
Step 27	**ipv6 address** *ipv6_address* **Example:** `switch(config-if)# ip address 11.1.1.1/24`	Configures IP address.
Step 28	**ipv6 pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Enables IGMP and PIM on the SVI. This is required is multicast sources and/or receivers exist in this VLAN.
Step 29	**fabric forwarding mode anycast-gateway** **Example:** `switch(config-if)# fabric forwarding mode anycast-gateway`	Configures Anycast Gateway Forwarding Mode.
Step 30	**ip pim neighbor-policy** *route-map-name* **Example:** `switch(config-if)# ip pim neighbor-policy route-map1`	Creates an IP PIM neighbor policy with a suitable route-map to deny any IPv4 addresses, preventing PIM from establishing PIM neighborship on the L2VNI SVI. **Note** Do not use Distributed Anycast Gateway for PIM Peerings.
Step 31	**exit** **Example:** `switch(config-if)# exit`	Exits command mode.
Step 32	**interface** *vlan_id* **Example:** `switch(config)# interface vlan100`	Configures Layer 3 VNI.
Step 33	**no shutdown** **Example:** `switch(config-if)# no shutdown`	Disable an interface.
Step 34	**vrf member vrf100** **Example:** `switch(config-if)# vrf member vrf100`	Configures VRF name.
Step 35	**ip forward** **Example:** `switch(config-if)# ip forward`	Enable IP forwarding on interface.
Step 36	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configures sparse-mode PIM on interface. There is no PIM peering happening in the Layer-3 VNI, but this command must be present for forwarding.

Configuring TRM on the VXLAN EVPN Spine

This procedure enables Tenant Routed Multicast (TRM) on a VXLAN EVPN spine switch.

Before you begin

The VXLAN BGP EVPN spine must be configured. See Configuring iBGP for EVPN on the Spine.

SUMMARY STEPS

1. **configure terminal**
2. **route-map permitall permit 10**
3. **set ip next-hop unchanged**
4. **exit**
5. **router bgp [autonomous system]** *number*
6. **address-family ipv4 mvpn**
7. **retain route-target all**
8. **neighbor** *ip-address* [**remote-as** *number*]
9. **address-family ipv4 mvpn**
10. **disable-peer-as-check**
11. **rewrite-rt-asn**
12. **send-community extended**
13. **route-reflector-client**
14. **route-map permitall out**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**route-map permitall permit 10** **Example:** `switch(config)# route-map permitall permit 10`	Configure the route-map. **Note** The route-map keeps the next-hop unchanged for EVPN routes • Required for eBGP • Options for iBGP
Step 3	**set ip next-hop unchanged** **Example:**	Set next hop address. **Note**

	Command or Action	Purpose
	`switch(config-route-map)# set ip next-hop unchanged`	The route-map keeps the next-hop unchanged for EVPN routes • Required for eBGP • Options for iBGP
Step 4	**exit** **Example:** `switch(config-route-map)# exit`	Return to exec mode.
Step 5	**router bgp [autonomous system]** *number* **Example:** `switch(config)# router bgp 65002`	Specify BGP.
Step 6	**address-family ipv4 mvpn** **Example:** `switch(config-router)# address-family ipv4 mvpn`	Configure the address family IPv4 MVPN under the BGP.
Step 7	**retain route-target all** **Example:** `switch(config-router-af)# retain route-target all`	Configure retain route-target all under address-family IPv4 MVPN [global]. **Note** Required for eBGP. Allows the spine to retain and advertise all MVPN routes when there are no local VNIs configured with matching import route targets.
Step 8	**neighbor** *ip-address* [**remote-as** *number*] **Example:** `switch(config-router-af)# neighbor 100.100.100.1`	Define neighbor.
Step 9	**address-family ipv4 mvpn** **Example:** `switch(config-router-neighbor)# address-family ipv4 mvpn`	Configure address family IPv4 MVPN under the BGP neighbor.
Step 10	**disable-peer-as-check** **Example:** `switch(config-router-neighbor-af)# disable-peer-as-check`	Disables checking the peer AS number during route advertisement. Configure this parameter on the spine for eBGP when all leafs are using the same AS but the spines have a different AS than leafs. **Note** Required for eBGP.
Step 11	**rewrite-rt-asn** **Example:** `switch(config-router-neighbor-af)# rewrite-rt-asn`	Normalizes the outgoing route target's AS number to match the remote AS number. Uses the BGP configured neighbors remote AS. The **rewrite-rt-asn** command is required if

	Command or Action	Purpose
		the route target auto feature is being used to configure EVPN route targets.
Step 12	**send-community extended** **Example:** `switch(config-router-neighbor-af)# send-community extended`	Configures community for BGP neighbors.
Step 13	**route-reflector-client** **Example:** `switch(config-router-neighbor-af)# route-reflector-client`	Configure route reflector. **Note** Required for iBGP with route-reflector.
Step 14	**route-map permitall out** **Example:** `switch(config-router-neighbor-af)# route-map permitall out`	Applies route-map to keep the next-hop unchanged. **Note** Required for eBGP.

Configuring Tenant Routed Multicast in Layer 2/Layer 3 Mixed Mode

This procedure enables the Tenant Routed Multicast (TRM) feature. This enables both Layer 2 and Layer 3 multicast BGP signaling. This mode is only necessary if non-TRM edge devices (VTEPs) are present in the Cisco Nexus 9000 Series switches (1st generation). Only the Cisco Nexus 9000-EX and 9000-FX switches can do Layer 2/Layer 3 mode (Anchor-DR).

To forward multicast between senders and receivers on the Layer 3 cloud and the VXLAN fabric on TRM vPC border leafs, the VIP/PIP configuration must be enabled. For more information, see Configuring VIP/PIP.

All Cisco Nexus 9300-EX and 9300-FX platform switches must be in Layer 2/Layer 3 mode.

Before you begin

VXLAN EVPN must be configured.

The rendezvous point (RP) must be configured.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.

	Command or Action	Purpose
Step 2	**feature ngmvpn** **Example:** `switch(config)# feature ngmvpn`	Enables the Next-Generation Multicast VPN (ngMVPN) control plane. New address family commands become available in BGP. **Note** The **no feature ngmvpn** command will not remove MVPN configuration under BGP.
Step 3	**advertise evpn multicast** **Example:** `switch(config)# advertise evpn multicast`	Advertises IMET and SMET routes into BGP EVPN towards non-TRM capable switches.
Step 4	**ip igmp snooping vxlan** **Example:** `switch(config)# ip igmp snooping vxlan`	Configure IGMP snooping for VXLAN VLANs.
Step 5	**ip multicast overlay-spt-only** **Example:** `switch(config)# ip multicast overlay-spt-only`	Gratuitously originate (S,A) route when source is locally connected. The **ip multicast overlay-spt-only** command is enabled by default on all MVPN-enabled Cisco Nexus 9000 Series switches (typically leaf nodes).
Step 6	**ip multicast overlay-distributed-dr** **Example:** `switch(config)# ip multicast overlay-distributed-dr`	Enables distributed anchor DR function on this VTEP. **Note** The NVE interface must be shut and unshut while configuring this command.
Step 7	**interface nve1** **Example:** `switch(config)# interface nve 1`	Configure the NVE interface.
Step 8	**[no] shutdown** **Example:** `switch(config-if-nve)# shutdown`	Shuts down the NVE interface. The **no shutdown** command brings up the interface.
Step 9	**member vni** *vni-range* **associate-vrf** **Example:** `switch(config-if-nve)# member vni 200100 associate-vrf`	Configure the Layer 3 virtual network identifier. The range of *vni-range* is from 1 to 16,777,214.
Step 10	**mcast-group** *ip-prefix* **Example:** `switch(config-if-nve-vni)# mcast-group 225.3.3.3`	Configures the multicast group on distributed anchor DR.
Step 11	**exit** **Example:** `switch(config-if-nve-vni)# exit`	Exits command mode.

	Command or Action	Purpose
Step 12	**interface loopback** *loopback_number* **Example:** `switch(config-if-nve)# interface loopback 10`	Configure the loopback interface on all distributed anchor DR devices.
Step 13	**ip address** *ip_address* **Example:** `switch(config-if)# ip address 100.100.1.1/32`	Configure IP address. This IP address is the same on all distributed anchor DR.
Step 14	**ip router ospf** *process-tag* **area** *ospf-id* **Example:** `switch(config-if)# ip router ospf 100 area 0.0.0.0`	OSPF area ID in IP address format.
Step 15	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on interface.
Step 16	**interface nve1** **Example:** `switch(config-if)# interface nve1`	Configure NVE interface.
Step 17	**shutdown** **Example:** `switch(config-if-nve)# shutdown`	Disable the interface.
Step 18	**mcast-routing override source-interface loopback** *int-num* **Example:** `switch(config-if-nve)# mcast-routing override source-interface loopback 10`	Enables that TRM is using a different loopback interface than the VTEPs default source-interface. The *loopback10* variable must be configured on every TRM-enabled VTEP (Anchor DR) in the underlay with the same IP address. This loopback and the respective **override** command are needed to serve TRM VTEPs in co-existence with non-TRM VTEPs.
Step 19	**exit** **Example:** `switch(config-if-nve)# exit`	Exits command mode.
Step 20	**router bgp 100** **Example:** `switch(config)# router bgp 100`	Set autonomous system number.
Step 21	**neighbor** *ip-addr* **Example:** `switch(config-router)# neighbor 1.1.1.1`	Configure IP address of the neighbor.

	Command or Action	Purpose
Step 22	**address-family ipv4 mvpn** **Example:** `switch(config-router-neighbor)# address-family ipv4 mvpn`	Configure multicast VPN.
Step 23	**send-community extended** **Example:** `switch(config-router-neighbor-af)# send-community extended`	Send community attribute.
Step 24	**exit** **Example:** `switch(config-router-neighbor-af)# exit`	Exits command mode.
Step 25	**exit** **Example:** `switch(config-router)# exit`	Exits command mode.
Step 26	**vrf** *vrf_name* **vrf100** **Example:** `switch(config)# vrf context vrf100`	Configure VRF name.
Step 27	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* **Example:** `switch(config-vrf)# ip pim rp-address 209.165.201.1 group-list 226.0.0.0/8`	The value of the *ip-address-of-router* parameter is that of the RP. The same IP address must be on all of the edge devices (VTEPs) for a fully distributed RP. For overlay RP placement options, see the Configuring a Rendezvous Point for Tenant Routed Multicast, on page 13 - Internal RP section.
Step 28	**address-family ipv4 unicast** **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Configure unicast address family.
Step 29	**route-target both auto mvpn** **Example:** `switch(config-vrf-af-ipv4)# route-target both auto mvpn`	Specify target for mvpn routes.
Step 30	**exit** **Example:** `switch(config-vrf-af-ipv4)# exit`	Exits command mode.
Step 31	**exit** **Example:** `switch(config-vrf)# exit`	Exits command mode.

	Command or Action	Purpose
Step 32	**interface** *vlan_id* **Example:** `switch(config)# interface vlan11`	Configure Layer 2 VNI.
Step 33	**no shutdown** **Example:** `switch(config-if)# no shutdown`	Disable an interface.
Step 34	**vrf member vrf100** **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 35	**ip address** *ip_address* **Example:** `switch(config-if)# ip address 11.1.1.1/24`	Configure IP address.
Step 36	**ip pim sparse-mode** **Example:** e `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on the interface.
Step 37	**fabric forwarding mode anycast-gateway** **Example:** `switch(config-if)# fabric forwarding mode anycast-gateway`	Configure Anycast Gateway Forwarding Mode.
Step 38	**ip pim neighbor-policy** *route-map-name* **Example:** `switch(config-if)# ip pim neighbor-policy route-map1`	Creates an IP PIM neighbor policy with a suitable route-map to deny any IPv4 addresses, preventing PIM from establishing PIM neighborship on the L2VNI SVI.
Step 39	**exit** **Example:** `switch(config-if)# exit`	Exits command mode.
Step 40	**interface** *vlan_id* **Example:** `switch(config)# interface vlan100`	Configure Layer 3 VNI.
Step 41	**no shutdown** **Example:** `switch(config-if)# no shutdown`	Disable an interface.

	Command or Action	Purpose
Step 42	**vrf member vrf100** **Example:** `switch(config-if)# vrf member vrf100`	Configure VRF name.
Step 43	**ip forward** **Example:** `switch(config-if)# ip forward`	Enable IP forwarding on interface.
Step 44	**ip pim sparse-mode** **Example:** `switch(config-if)# ip pim sparse-mode`	Configure sparse-mode PIM on the interface.

Configuring VXLAN EVPN and TRM with IPv6 Multicast Underlay

Configuring IPv6 multicast underlay in the VXLAN fabric involves the following configurations:

Configuring L2-VNI Based Multicast Group in Underlay

Under NVE configuration on a leaf, IPv6 multicast group (IPv6) is configured for each L2-VNI (VLAN).

SUMMARY STEPS

1. **configure terminal**
2. **interface nve1**
3. **member vni** *vni*
4. **mcast-group** *ipv6-prefix*
5. **global mcast-group** *ipv6-multicast-group* **l2**
6. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface nve1** **Example:** `switch(config)# interface nve1`	Configures the NVE interface.

	Command or Action	Purpose
Step 3	**member vni** *vni* **Example:** `switch(config-if-nve)# member vni 10501`	Configures the Layer 2 virtual network identifier.
Step 4	**mcast-group** *ipv6-prefix* **Example:** `switch(config-if-nve-vni)# mcast-group ff04::40`	Builds the default multicast distribution tree for the Layer 2 VNI.
Step 5	**global mcast-group** *ipv6-multicast-group* **l2** **Example:** `switch(config-if-nve)# global mcast-group ff04::40 l2`	Configures the global multicast group for the Layer 2 VNI.
Step 6	**exit** **Example:** `switch(config-if-nve)# exit`	Exits configuration mode.

Configuring L3-VNI Based Multicast Group in Underlay

IPv6 multicast group (IPv6) is configured for each L3-VNI (VRF).

SUMMARY STEPS

1. **configure terminal**
2. **interface nve1**
3. **member vni** *vni* **associate-vrf**
4. **mcast-group** *ipv6-prefix*
5. **global mcast-group** *ipv6-multicast-group* **l3**
6. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface nve1** **Example:** `switch(config)# interface nve1`	Configures the NVE interface.

	Command or Action	Purpose
Step 3	**member vni** *vni* **associate-vrf** **Example:** `switch(config-if-nve)# member vni 50001 associate-vrf`	Associates L3VNI to VRF.
Step 4	**mcast-group** *ipv6-prefix* **Example:** `switch(config-if-nve-vni)# mcast-group ff10:0:0:1::1`	Builds the default multicast distribution tree for the Layer 3 VNI.
Step 5	**global mcast-group** *ipv6-multicast-group* **l3** **Example:** `switch(config-if-nve)# global mcast-group ff04::40 l3`	Configures the global multicast group for the Layer 3 VNI.
Step 6	**exit** **Example:** `switch(config-if-nve)# exit`	Exits configuration mode.

Enabling PIMv6 for Underlay

PIMv6 in and underlay is configured as follows:

SUMMARY STEPS

1. **configure terminal**
2. **interface loopback** *number*
3. **ipv6 address** *ipv6-prefix*
4. **ipv6 pim sparse-mode**
5. **interface nve1**
6. **source-interface loopback** *number*
7. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface loopback** *number* **Example:**	Configures an interface loopback. This example configures interface loopback 1.

	Command or Action	Purpose
	`switch(config)# interface loopback 1`	
Step 3	**ipv6 address** *ipv6-prefix* **Example:** `switch(config-if)# ipv6 address 11:0:0:1::1/128`	Configures an IP address for this interface. It should be a unique IP address that helps to identify this router.
Step 4	**ipv6 pim sparse-mode** **Example:** `switch(config-if)# ipv6 pim sparse-mode`	Enables PIM6 sparse mode.
Step 5	**interface nve1** **Example:** `switch(config-if)# interface nve1`	Configures the NVE interface.
Step 6	**source-interface loopback** *number* **Example:** `switch(config-if-nve)# source-interface loopback 1`	Configures an source interface loopback.
Step 7	**exit** **Example:** `switch(config-if-nve)# exit`	Exits configuration mode. **Note** For the PIMv6 configuration see the *Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide*. For the TRM configuration see the *Cisco Nexus 9000 Series NX-OS VXLAN Configuration Guide*.

Configuring Layer 2 Tenant Routed Multicast

This procedure enables the Tenant Routed Multicast (TRM) feature. This enables Layer 2 multicast BGP signaling.

IGMP Snooping Querier must be configured per multicast-enabled VXLAN VLAN on all Layer-2 TRM leaf switches.

Before you begin

VXLAN EVPN must be configured.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.

	Command or Action	Purpose
Step 2	**feature ngmvpn** **Example:** `switch(config)# feature ngmvpn`	Enables EVPN/MVPN feature. **Note** The **no feature ngmvpn** command will not remove MVPN configuration under BGP.
Step 3	**advertise evpn multicast** **Example:** `switch(config)# advertise evpn multicast`	Advertise L2 multicast capability.
Step 4	**ip igmp snooping vxlan** **Example:** `switch(config)# ip igmp snooping vxlan`	Configure IGMP snooping for VXLANs.
Step 5	**vlan configuration** *vlan-id* **Example:** `switch(config)# vlan configuration 101`	Enter configuration mode for VLAN 101.
Step 6	**ip igmp snooping querier** *querier-ip-address* **Example:** `switch(config-vlan-config)# ip igmp snooping querier 2.2.2.2`	Configure IGMP snooping querier for each multicast-enabled VXLAN VLAN.

Configuring TRM with vPC Support

This section provides steps to configure TRM with vPC support. Beginning with Cisco NX-OS Release 10.1(2), TRM Multisite with vPC BGW is supported.

SUMMARY STEPS

1. **configure terminal**
2. **feature vpc**
3. **feature interface-vlan**
4. **feature lacp**
5. **feature pim**
6. **feature ospf**
7. **ip pim rp-address** *address* **group-list** *range*
8. **vpc domain** *domain-id*
9. **peer switch**
10. **peer gateway**
11. **peer-keepalive destination** *ipaddress*
12. **ip arp synchronize**
13. **ipv6 nd synchronize**
14. Create vPC peer-link.

15. **system nve infra-vlans** *range*
16. **vlan** *number*
17. Create the SVI.
18. (Optional) **delay restore interface-vlan** *seconds*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter global configuration mode.
Step 2	**feature vpc** **Example:** `switch(config)# feature vpc`	Enables vPCs on the device.
Step 3	**feature interface-vlan** **Example:** `switch(config)# feature interface-vlan`	Enables the interface VLAN feature on the device.
Step 4	**feature lacp** **Example:** `switch(config)# feature lacp`	Enables the LACP feature on the device.
Step 5	**feature pim** **Example:** `switch(config)# feature pim`	Enables the PIM feature on the device.
Step 6	**feature ospf** **Example:** `switch(config)# feature ospf`	Enables the OSPF feature on the device.
Step 7	**ip pim rp-address** *address* **group-list** *range* **Example:** `switch(config)# ip pim rp-address 100.100.100.1 group-list 224.0.0/4`	Defines a PIM RP address for the underlay multicast group range.
Step 8	**vpc domain** *domain-id* **Example:** `switch(config)# vpc domain 1`	Creates a vPC domain on the device and enters vpn-domain configuration mode for configuration purposes. There is no default. The range is from 1 to 1000.
Step 9	**peer switch** **Example:**	Defines the peer switch.

	Command or Action	Purpose
	`switch(config-vpc-domain)# peer switch`	
Step 10	**peer gateway** **Example:** `switch(config-vpc-domain)# peer gateway`	To enable Layer 3 forwarding for packets destined to the gateway MAC address of the virtual port channel (vPC), use the **peer-gateway** command.
Step 11	**peer-keepalive destination** *ipaddress* **Example:** `switch(config-vpc-domain)# peer-keepalive destination 172.28.230.85`	Configures the IPv4 address for the remote end of the vPC peer-keepalive link. **Note** The system does not form the vPC peer link until you configure a vPC peer-keepalive link. The management ports and VRF are the defaults. **Note** We recommend that you configure a separate VRF and use a Layer 3 port from each vPC peer device in that VRF for the vPC peer-keepalive link. For more information about creating and configuring VRFs, see the Cisco Nexus 9000 NX-OS Series Unicast Routing Config Guide, 9.3(x).
Step 12	**ip arp synchronize** **Example:** `switch(config-vpc-domain)# ip arp synchronize`	Enables IP ARP synchronize under the vPC Domain to facilitate faster ARP table population following device reload.
Step 13	**ipv6 nd synchronize** **Example:** `switch(config-vpc-domain)# ipv6 nd synchronize`	Enables IPv6 nd synchronization under the vPC domain to facilitate faster nd table population following device reload.

Step 14	Create vPC peer-link.
	Example:
	`switch(config)# interface port-channel 1`
	`switch(config)# switchport`
	`switch(config)# switchport mode trunk`
	`switch(config)# switchport trunk allowed vlan 1,10,100-200`
	`switch(config)# mtu 9216`
	`switch(config)# vpc peer-link`
	`switch(config)# no shut`
	
	`switch(config)# interface Ethernet 1/1, 1/21`
	`switch(config)# switchport`
	`switch(config)# mtu 9216`
	`switch(config)# channel-group 1 mode active`
	`switch(config)# no shutdown`
		Creates the vPC peer-link port-channel interface and adds two member interfaces to it.

Step 15	**system nve infra-vlans** *range* **Example:**	Defines a non-VXLAN enabled VLAN as a backup routed path.

	Command or Action	Purpose
	`switch(config)# system nve infra-vlans 10`	
Step 16	**vlan** *number* **Example:** `switch(config)# vlan 10`	Creates the VLAN to be used as an infra-VLAN.

Step 17	Create the SVI.
	Example:
	`switch(config)# interface vlan 10`
	`switch(config)# ip address 10.10.10.1/30`
	`switch(config)# ip router ospf process UNDERLAY area 0`
	`switch(config)# ip pim sparse-mode`
	`switch(config)# no ip redirects`
	`switch(config)# mtu 9216`
	`switch(config)# no shutdown`
		Creates the SVI used for the backup routed path over the vPC peer-link.

Step 18	(Optional) **delay restore interface-vlan** *seconds* **Example:** `switch(config-vpc-domain)# delay restore interface-vlan 45`	Enables the delay restore timer for SVIs. We recommend tuning this value when the SVI/VNI scale is high. For example, when the SCI count is 1000, we recommend that you set the delay restore for **interface-vlan** to 45 seconds.

Configuring TRM with vPC Support (Cisco Nexus 9504-R and 9508-R)

SUMMARY STEPS

1. **configure terminal**
2. **feature vpc**
3. **feature interface-vlan**
4. **feature lacp**
5. **feature pim**
6. **feature ospf**
7. **ip pim rp-address** *address* **group-list** *range*
8. **vpc domain** *domain-id*
9. **hardware access-list tcam region mac-ifacl**
10. **hardware access-list tcam region vxlan 10**
11. **reload**
12. **peer switch**
13. **peer gateway**
14. **peer-keepalive destination** *ipaddress*
15. **ip arp synchronize**
16. **ipv6 nd synchronize**
17. Create vPC peer-link.

18. **system nve infra-vlans** *range*
19. **vlan** *number*
20. Create the SVI.
21. (Optional) **delay restore interface-vlan** *seconds*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter global configuration mode.
Step 2	**feature vpc** **Example:** `switch(config)# feature vpc`	Enables vPCs on the device.
Step 3	**feature interface-vlan** **Example:** `switch(config)# feature interface-vlan`	Enables the interface VLAN feature on the device.
Step 4	**feature lacp** **Example:** `switch(config)# feature lacp`	Enables the LACP feature on the device.
Step 5	**feature pim** **Example:** `switch(config)# feature pim`	Enables the PIM feature on the device.
Step 6	**feature ospf** **Example:** `switch(config)# feature ospf`	Enables the OSPF feature on the device.
Step 7	**ip pim rp-address** *address* **group-list** *range* **Example:** `switch(config)# ip pim rp-address 100.100.100.1 group-list 224.0.0/4`	Defines a PIM RP address for the underlay multicast group range.
Step 8	**vpc domain** *domain-id* **Example:** `switch(config)# vpc domain 1`	Creates a vPC domain on the device and enters vpn-domain configuration mode for configuration purposes. There is no default. The range is 1–1000.
Step 9	**hardware access-list tcam region mac-ifacl** **Example:**	Carves the TCAM region for the ACL database. **Note**

	Command or Action	Purpose
	`switch(config)# hardware access-list tcam region mac-ifacl 0`	This TCAM carving command is required to enable TRM forwarding for N9K-X9636C-RX line cards only. With no TCAM region carved for **mac-ifacl**, the TCAM resources are used for TRM instead.
Step 10	**hardware access-list tcam region vxlan 10** **Example:** `switch(config)# hardware access-list tcam region vxlan 10`	Assigns the the TCAM region for use by a VXLAN. **Note** This TCAM carving command is required to enable TRM forwarding for N9K-X9636C-RX line cards only.
Step 11	**reload** **Example:** `switch(config)# reload`	Reloads the switch config for the TCAM assignments to become active.
Step 12	**peer switch** **Example:** `switch(config-vpc-domain)# peer switch`	Defines the peer switch.
Step 13	**peer gateway** **Example:** `switch(config-vpc-domain)# peer gateway`	To enable Layer 3 forwarding for packets that are destined to the gateway MAC address of the virtual port channel (vPC), use the **peer-gateway** command.
Step 14	**peer-keepalive destination** *ipaddress* **Example:** `switch(config-vpc-domain)# peer-keepalive destination 172.28.230.85`	Configures the IPv4 address for the remote end of the vPC peer-keepalive link. **Note** The system does not form the vPC peer link until you configure a vPC peer-keepalive link. The management ports and VRF are the defaults. **Note** We recommend that you configure a separate VRF and use a Layer 3 port from each vPC peer device in that VRF for the vPC peer-keepalive link. For more information about creating and configuring VRFs, see the Cisco Nexus 9000 NX-OS Series Unicast Routing Config Guide, 9.3(x).
Step 15	**ip arp synchronize** **Example:** `switch(config-vpc-domain)# ip arp synchronize`	Enables IP ARP synchronize under the vPC Domain to facilitate faster ARP table population following device reload.
Step 16	**ipv6 nd synchronize** **Example:** `switch(config-vpc-domain)# ipv6 nd synchronize`	Enables IPv6 and synchronization under the vPC domain to facilitate faster and table population following device reload.

	Command or Action	Purpose

Step 17	Create vPC peer-link.
	Example:
	`switch(config)# interface port-channel 1`
	`switch(config)# switchport`
	`switch(config)# switchport mode trunk`
	`switch(config)# switchport trunk allowed vlan 1,10,100-200`
	`switch(config)# mtu 9216`
	`switch(config)# vpc peer-link`
	`switch(config)# no shut`
	
	`switch(config)# interface Ethernet 1/1, 1/21`
	`switch(config)# switchport`
	`switch(config)# mtu 9216`
	`switch(config)# channel-group 1 mode active`
	`switch(config)# no shutdown`
		Creates the vPC peer-link port-channel interface and adds two member interfaces to it.

Step 18	**system nve infra-vlans** *range* **Example:** `switch(config)# system nve infra-vlans 10`	Defines a non-VXLAN enabled VLAN as a backup routed path.
Step 19	**vlan** *number* **Example:** `switch(config)# vlan 10`	Creates the VLAN to be used as an infra-VLAN.

Step 20	Create the SVI.
	Example:
	`switch(config)# interface vlan 10`
	`switch(config)# ip address 10.10.10.1/30`
	`switch(config)# ip router ospf process UNDERLAY area 0`
	`switch(config)# ip pim sparse-mode`
	`switch(config)# no ip redirects`
	`switch(config)# mtu 9216`
	`switch(config)# no shutdown`
		Creates the SVI used for the backup routed path over the vPC peer-link.

Step 21	(Optional) **delay restore interface-vlan** *seconds* **Example:** `switch(config-vpc-domain)# delay restore interface-vlan 45`	Enables the delay restore timer for SVIs. We recommend tuning this value when the SVI/VNI scale is high. For example, when the SCI count is 1000, we recommend that you set the delay restore for **interface-vlan** to 45 seconds.

Flex Stats for TRM

Beginning with Cisco NX-OS Release 10.3(1)F, the Real-time/flex statistics for TRM is supported for Overlay routes on Cisco Nexus 9300-X Cloud Scale Switches. Flex Stats is not supported for Underlay Routes

Note VXLAN NVE VNI ingress and egress, NVE per-peer ingress and tunnel tx stats won't be supported.

In a VXLAN TRM setup, if you want mroute statistics for overlay mroutes you must configure the **hardware profile multicast flex-stats-enable** command in the default template. For more information on configuration, see Configuring Flex Stats for TRM, on page 53.

The following CLIs will not be supported after the flex stats CLI is enabled:

- `sh nve vni <vni_id>/<all> counters`
- `sh nve peers <peer-ip> interface nve 1 counters`
- `sh int tunnel <Tunnel interface number> counters`

Configuring Flex Stats for TRM

This procedure enables/disables the flex stats counters in a VXLAN TRM setup.

SUMMARY STEPS

1. **configure terminal**
2. **[no] hardware profile multicast flex-stats-enable**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter configuration mode.
Step 2	**[no] hardware profile multicast flex-stats-enable** **Example:** `switch(config)# hardware profile multicast flex-stats-enable`	Enables the flex stats on TRM. The **no** option disables the flex stats on TRM. **Note** To reflect the changes done during configuration, ensure that the switch is reloaded.

Configuring TRM Data MDT

About TRM Data MDT

Tenant Routed Multicast (TRM) enables multicast forwarding on the VXLAN fabric that uses a BGP-based EVPN control plane. TRM provides multi-tenancy aware multicast forwarding between senders and receivers within the same or different subnet local to the VTEP or across VTEPs.

Existing TRM solution enables multicast forwarding using default Multicast Distribution Tree (default MDT). With default MDT, nodes (PEs) will always receive traffic in the underlay irrespective of whether they have interested receiver on the overlay.

The solution described in this document enables optimized multicast forwarding using S-PMSI (data MDT). With S-PMSI, source traffic will be encapsulated in a selective multicast tunnel. Only the leafs that have interested receivers will join the selective multicast distribution tree.

Switchover to Data MDT can be immediate or based on the traffic bandwidth (threshold based configuration).

Guidelines and Limitations for TRM Data MDT

TRM Data MDT has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.3(2)F, TRM Data MDT is supported on Cisco Nexus 9300 EX/FX/FX2/FX3/GX/GX2 switches, and 9500 switches with 9700-EX/FX/GX line cards.
- Beginning with Cisco NX-OS Release 10.5(2)F, TRM Data MDT is supported on Cisco Nexus 9500 Series switches with N9K-X9736C-FX3 line card.
- Beginning with Cisco NX-OS Release 10.4(1)F, TRM Data MDT is supported on Cisco Nexus 9332D-H2R switches.
- Beginning with Cisco NX-OS Release 10.4(2)F, TRM Data MDT is supported on Cisco Nexus 93400LD-H1 switches.
- Beginning with Cisco NX-OS Release 10.4(3)F, TRM Data MDT is supported on Cisco Nexus 9364C-H1 switches.
- Data MDT in fabric is supported only with DCI IR for a given VRF. Data MDT in fabric is not supported with DCI Multicast for a given VRF on the site BGW.
- Data MDT configuration is VRF specific and configured under L3 VRF.
- The following TRM Data MDT features are supported:
    - ASM and SSM group ranges are supported for Data MDT. PIM-Bider Underlay is not supported for Data MDT.
    - Data MDT supports IPv4 and IPv6 overlay multicast traffic.
    - Data MDT will be supported by vPC, VMCT leaf's as well as vPC/Anycast BGW. Also, L2, L3 orphan/external network can be connected to vPC nodes.
    - Data MDT config per L3 VRF.
    - Data MDT origination (immediate and threshold based).

- Data MDT encap route programming delay of 3 seconds. User-defined delays are currently not supported.

- L2, L2 -L3 mixed mode will not be supported.
- New L3VNI mode is supported.
- Ensure that the total number of underlay groups (L2 BUM, default MDT, and data MDT groups) is 512.

Configuring TRM Data MDT

Follow this procedure to configure TRM Data MDT:

Before you begin

To enable switching to data MDT group based on real-time flow rate, the following command is needed:

hardware profile multicast flex-stats-enable

Note This command requires switch reloading.

SUMMARY STEPS

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **address-family {ipv4 | ipv6} unicast**
4. **[no] mdt data vxlan** <*group-range-1*> **[threshold]** [**route-map** <*value*> <*policy-name_1*>] [**seq** <*sequence-number*>]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**vrf context** *vrf-name* **Example:** `switch(config)# vrf context vrf1`	Configures the VRF.
Step 3	**address-family {ipv4 | ipv6} unicast** **Example:** For IPv4	Configures the IPv4 or IPv6 unicast address family.

	Command or Action	Purpose

	`switch(config-vrf)# address-family ipv4 unicast`
	For IPv6
	`switch(config-vrf)# address-family ipv6 unicast`	

Step 4	**[no] mdt data vxlan** <*group-range-1*> **[threshold]** [**route-map** <*value*> <*policy-name_1*>] [**seq** <*sequence-number*>] **Example:** `switch(config-vrf-af)# mdt data vxlan 224.7.8.0/24 route-map map1 10`	Data MDT can be enabled/disabled per address family. Cisco Nexus supports overlapping group ranges between VRF as well as within the VRF between the address families. • Threshold & route-maps are optional. The traffic threshold is the traffic of the source and is measured in kbps. When the threshold is exceeded, the traffic takes 3 seconds to switch over to data MDT. • Group-range is part of the command key. More than one group range can be configured per address family. • BUM & default MDT group should not overlap with data MDT group. • Data MDT can have overlapping config range.

Verifying TRM Data MDT Configuration

To display the TRM Data MDT configuration information, enter one of the following commands:

Command	Purpose
show nve vni { <*vni-id*> **| all } mdt [{ local | remote | peer-sync }] [{** <*cs*> <*cg*>**} | {** <*cs6*> <*cg6*>**}]**	Displays customer source (CS), customer group (DS), data group (DG) mapping information.
show nve vrf [x] mdt [local | remote | peer-sync] [y] [z]	Displays CS, CG allocations under VRF.
show bgp ipv4 mvpn route-type 3 detail	Displays BGP S-PMSI route information for IP
show bgp ipv6 mvpn route-type 3 detail	Displays BGP S-PMSI route information for IP
show fabric multicast [ipv4 | ipv6] spmsi-ad-route [*Source Address*] [*Group address*] vrf <*vrf_name*>	Displays fabric multicast SPMSI-AD IPV4/IPv6 tenant VRF.
show ip mroute detail vrf <*vrf_name*>	Displays IP multicast route information for defa
show l2route spmsi {all | topology <*vlan*>}	Displays CS-CG to DS-DG mapping information programming).
show forwarding distribution multicast vxlan mdt-db	Displays MFDM/MFIB data MDT db.
show nve resource multicast	Displays the resource usage of data MDT and a

Configuring IGMP Snooping

Overview of IGMP Snooping Over VXLAN

By default, multicast traffic over VXLAN is flooded in the VNI/VLAN like any broadcast and unknown unicast traffic. With IGMP snooping enabled, each VTEP can snoop IGMP reports and only forward multicast traffic towards interested receivers.

The configuration of IGMP snooping is the same in VXLAN as in the configuration of IGMP snooping in a regular VLAN domain. For more information on IGMP snooping, see the *Configuring IGMP Snooping* section in the Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide, Release 7.x.

Guidelines and Limitations for IGMP Snooping Over VXLAN

See the following guidelines and limitations for IGMP snooping over VXLAN:

- IGMP snooping over VXLAN is not supported on VLANs with FEX member ports.
- IGMP snooping over VXLAN is supported with both IR and multicast underlay.
- IGMP snooping over VXLAN is supported in BGP EVPN topologies, not flood and learn topologies.
- Beginning with NX-OS release 9.3(3), IGMP Snooping over VXLAN is supported on the Cisco Nexus C93600CD-GX.
- Beginning with NX-OS release 9.3(3), IGMP Snooping over VXLAN is supported on the Cisco Nexus C9364C-GX.
- Beginning with NX-OS release 9.3(3), IGMP Snooping over VXLAN is supported on the Cisco Nexus C9316D-GX.

Configuring IGMP Snooping Over VXLAN

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)#**ip igmp snooping vxlan**
3. switch(config)#**ip igmp snooping disable-nve-static-router-port**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)#**ip igmp snooping vxlan**	Enables IGMP snooping for VXLAN VLANs. You have to explicitly configure this command to enable snooping for VXLAN VLANs.
Step 3	switch(config)#**ip igmp snooping disable-nve-static-router-port**	Configures IGMP snooping over VXLAN to not include NVE as static mrouter port using this global CLI command. IGMP snooping over VXLAN has the NVE interface as mrouter port by default.

Verifying VXLAN EVPN and TRM with IPv6 Multicast Underlay

Use the following show command to verify the status of the IPv6 Multicast Underlay configuration:

```
switch(config)# show run interface nve 1

!Command: show running-config interface nve1
!Running configuration last done at: Wed Jul  5 10:03:58 2023
!Time: Wed Jul  5 10:04:01 2023
version 10.3(99x) Bios:version 01.08

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 10501
    mcast-group ff04::40
  member vni 50001 associate-vrf
    mcast-group ff10:0:0:1::1
```

Use the following show commands to verify the PIMv6 ASM configuration:

```
switch(config)# show ipv6 mroute
IPv6 Multicast Routing Table for VRF "default"

(*, ff04::40/128), uptime: 05:20:19, nve pim6 ipv6
  Incoming interface: Ethernet1/36, RPF nbr: fe80::23a:9cff:fe23:8367
  Outgoing interface list: (count: 1)
    nve1, uptime: 05:20:19, nve


(172:172:16:1::1/128, ff04::40/128), uptime: 05:20:19, nve m6rib pim6 ipv6
  Incoming interface: loopback1, RPF nbr: 172:172:16:1::1
  Outgoing interface list: (count: 2)
    Ethernet1/36, uptime: 01:47:03, pim6
    Ethernet1/27, uptime: 04:14:20, pim6


(*, ff10:0:0:1::10/128), uptime: 05:20:18, nve ipv6 pim6
  Incoming interface: Ethernet1/36, RPF nbr: fe80::23a:9cff:fe23:8367
  Outgoing interface list: (count: 1)
    nve1, uptime: 05:20:18, nve


(172:172:16:1::1/128, ff10:0:0:1::10/128), uptime: 05:20:18, nve m6rib ipv6 pim6
  Incoming interface: loopback1, RPF nbr: 172:172:16:1::1
  Outgoing interface list: (count: 2)
    Ethernet1/36, uptime: 04:04:35, pim6
    Ethernet1/27, uptime: 04:13:35, pim6
```

```
switch(config)# show ipv6 pim neighbor
PIM Neighbor Status for VRF "default"
Neighbor                  Interface            Uptime    Expires   DR       Bidir-  BFD
     ECMP Redirect
                                                                    Priority Capable State
   Capable
fe80::23a:9cff:fe28:5e07     Ethernet1/27         20:23:38  00:01:44  1        yes     n/a
     no
   Secondary addresses:
    27:50:1:1::2
```

```
switch(config)# show ipv6 pim rp
PIM RP Status Information for VRF "default"
BSR disabled
BSR RP Candidate policy: route-map1
BSR RP policy: route-map1

RP: 101:101:101:101::101, (0),
 uptime: 21:30:43   priority: 255,
 RP-source: (local),
 group ranges:
 ff00::/8
```

The following example provides the output for leaf switch BGP neighbor-1:

```
switch(config-if)# show ipv6 bgp neighbors

BGP neighbor is 33:52:1:1::2, remote AS 200, ebgp link, Peer index 3
  BGP version 4, remote router ID 172.17.1.1
  Neighbor previous state = OpenConfirm
  BGP state = Established, up for 00:00:16
  Neighbor vrf: default
  Peer is directly attached, interface Ethernet1/33
  Enable logging neighbor events
  Last read 0.926823, hold time = 3, keepalive interval is 1 seconds
  Last written 0.926319, keepalive timer expiry due 0.073338
  Received 23 messages, 0 notifications, 0 bytes in queue
  Sent 67 messages, 0 notifications, 0(0) bytes in queue
  Enhanced error processing: On
    0 discarded attributes
  Connections established 1, dropped 0
  Last update recd 00:00:15, Last update sent  = 00:00:15
   Last reset by us 00:08:45, due to session closed
  Last error length sent: 0
  Reset error value sent: 0
  Reset error sent major: 104 minor: 0
  Notification data sent:
  Last reset by peer never, due to No error
  Last error length received: 0
  Reset error value received 0
  Reset error received major: 0 minor: 0
  Notification data received:

  Neighbor capabilities:
  Dynamic capability: advertised (mp, refresh, gr) received (mp, refresh, gr)
  Dynamic capability (old): advertised received
  Route refresh capability (new): advertised received
  Route refresh capability (old): advertised received
  4-Byte AS capability: advertised received
  Address family IPv6 Unicast: advertised received
  Graceful Restart capability: advertised received

  Graceful Restart Parameters:
  Address families advertised to peer:
    IPv6 Unicast
```

```
  Address families received from peer:
    IPv6 Unicast
  Forwarding state preserved by peer for:
  Restart time advertised to peer: 400 seconds
  Stale time for routes advertised by peer: 300 seconds
  Restart time advertised by peer: 120 seconds
  Extended Next Hop Encoding Capability: advertised received
  Receive IPv6 next hop encoding Capability for AF:
    IPv4 Unicast  VPNv4 Unicast

  Message statistics:
                              Sent               Rcvd
  Opens:                        46                  1
  Notifications:                 0                  0
  Updates:                       2                  2
  Keepalives:                   18                 18
  Route Refresh:                 0                  0
  Capability:                    2                  2
  Total:                        67                 23
  Total bytes:                 521                538
  Bytes in queue:                0                  0

  For address family: IPv6 Unicast
  BGP table version 10, neighbor version 10
  3 accepted prefixes (3 paths), consuming 864 bytes of memory
  0 received prefixes treated as withdrawn
  2 sent prefixes (2 paths)
  Inbound soft reconfiguration allowed(always)
  Allow my ASN 3 times
  Last End-of-RIB received 00:00:01 after session start
  Last End-of-RIB sent 00:00:01 after session start
  First convergence 00:00:01 after session start with 2 routes sent

  Local host: 33:52:1:1::1, Local port: 179
  Foreign host: 33:52:1:1::2, Foreign port: 17226
  fd = 112
```

The following example provides the output for leaf switch BGP neighbor-2:

```
switch(config-if)# show bgp l2vpn evpn neighbors 172:17:1:1::1

BGP neighbor is 172:17:1:1::1, remote AS 200, ebgp link, Peer index 5
  BGP version 4, remote router ID 172.17.1.1
  Neighbor previous state = OpenConfirm
  BGP state = Established, up for 00:01:33
  Neighbor vrf: default
  Using loopback0 as update source for this peer
  Using iod 65 (loopback0) as update source
  Enable logging neighbor events
  External BGP peer might be up to 5 hops away
  Last read 0.933565, hold time = 3, keepalive interval is 1 seconds
  Last written 0.915927, keepalive timer expiry due 0.083742
  Received 105 messages, 0 notifications, 0 bytes in queue
  Sent 105 messages, 0 notifications, 0(0) bytes in queue
  Enhanced error processing: On
    0 discarded attributes
  Connections established 1, dropped 0
  Last update recd 00:01:32, Last update sent  = 00:01:32
   Last reset by us never, due to No error
  Last error length sent: 0
  Reset error value sent: 0
  Reset error sent major: 0 minor: 0
  Notification data sent:
  Last reset by peer never, due to No error
  Last error length received: 0
```

```
Reset error value received 0
Reset error received major: 0 minor: 0
Notification data received:

Neighbor capabilities:
Dynamic capability: advertised (mp, refresh, gr) received (mp, refresh, gr)
Dynamic capability (old): advertised received
Route refresh capability (new): advertised received
Route refresh capability (old): advertised received
4-Byte AS capability: advertised received
Address family IPv4 MVPN: advertised received
Address family IPv6 MVPN: advertised received
Address family L2VPN EVPN: advertised received
Graceful Restart capability: advertised received

Graceful Restart Parameters:
Address families advertised to peer:
  IPv4 MVPN  IPv6 MVPN  L2VPN EVPN
Address families received from peer:
  IPv4 MVPN  IPv6 MVPN  L2VPN EVPN
Forwarding state preserved by peer for:
Restart time advertised to peer: 400 seconds
Stale time for routes advertised by peer: 300 seconds
Restart time advertised by peer: 120 seconds
Extended Next Hop Encoding Capability: advertised received
Receive IPv6 next hop encoding Capability for AF:
  IPv4 Unicast  VPNv4 Unicast

Message statistics:
                            Sent               Rcvd
Opens:                         1                  1
Notifications:                 0                  0
Updates:                       6                  3
Keepalives:                   95                 95
Route Refresh:                 0                  0
Capability:                    6                  6
Total:                       105                105
Total bytes:                2551               2047
Bytes in queue:                0                  0

For address family: IPv4 MVPN
BGP table version 3, neighbor version 3
0 accepted prefixes (0 paths), consuming 0 bytes of memory
0 received prefixes treated as withdrawn
0 sent prefixes (0 paths)
Community attribute sent to this neighbor
Extended community attribute sent to this neighbor
Allow my ASN 3 times
Outbound route-map configured is RN_NextHop_Unchanged, handle obtained
Last End-of-RIB received 00:00:01 after session start
Last End-of-RIB sent 00:00:01 after session start
First convergence 00:00:01 after session start with 0 routes sent

For address family: IPv6 MVPN
BGP table version 3, neighbor version 3
0 accepted prefixes (0 paths), consuming 0 bytes of memory
0 received prefixes treated as withdrawn
0 sent prefixes (0 paths)
Community attribute sent to this neighbor
Extended community attribute sent to this neighbor
Allow my ASN 3 times
Outbound route-map configured is RN_NextHop_Unchanged, handle obtained
Last End-of-RIB received 00:00:01 after session start
Last End-of-RIB sent 00:00:01 after session start
```

```
First convergence 00:00:01 after session start with 0 routes sent

For address family: L2VPN EVPN
BGP table version 7, neighbor version 7
0 accepted prefixes (0 paths), consuming 0 bytes of memory
0 received prefixes treated as withdrawn
4 sent prefixes (4 paths)
Community attribute sent to this neighbor
Extended community attribute sent to this neighbor
Allow my ASN 3 times
Advertise GW IP is enabled
Outbound route-map configured is RN_NextHop_Unchanged, handle obtained
Last End-of-RIB received 00:00:01 after session start
Last End-of-RIB sent 00:00:01 after session start
First convergence 00:00:01 after session start with 4 routes sent

Local host: 172:16:1:2::1, Local port: 21132
Foreign host: 172:17:1:1::1, Foreign port: 179
fd = 113
```

Example Configuration for VXLAN EVPN and TRM with IPv6 Multicast Underlay

In the following examples, the sample configuration for the leaf, spine, and RP are shown:

- **Leaf - Sample configuration of IPv6 multicast underlay:**
  - NVE Configuration

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 10501
    mcast-group ff04::40
  member vni 50001 associate-vrf
    mcast-group ff10:0:0:1::1
```

  - PIMv6 Configuration

```
feature pim6

ipv6 pim rp-address 101:101:101:101::101 group-list ff00::/8

interface loopback1
  ipv6 address 172:172:16:1::1/128
  ipv6 pim sparse-mode

interface Ethernet1/27
  ipv6 address 27:50:1:1::1/64
  ospfv3 hello-interval 1
  ipv6 router ospfv3 v6u area 0.0.0.0
  ipv6 pim sparse-mode
  no shutdown
```

  - BGP Configuration

```
router bgp 100
    router-id 172.16.1.1
    address-family ipv4 unicast
      maximum-paths 64
```

```
      maximum-paths ibgp 64
    address-family ipv6 unicast
      maximum-paths 64
      maximum-paths ibgp 64
    address-family ipv4 mvpn
    address-family l2vpn evpn
    neighbor 172:17:1:1::1
      remote-as 100
      update-source loopback0
      address-family ipv4 mvpn
        send-community
        send-community extended
      address-family ipv6 mvpn
        send-community
        send-community extended
      address-family l2vpn evpn
        send-community
     neighbor 172:17:2:2::1
       remote-as 100
       update-source loopback0
       address-family ipv4 mvpn
         send-community
         send-community extended
       address-family ipv6 mvpn
         send-community
         send-community extended
       address-family l2vpn evpn
         send-community
         send-community extended
     vrf VRF1
       reconnect-interval 1
       address-family ipv4 unicast
         network 150.1.1.1/32
         advertise l2vpn evpn
         redistribute hmm route-map hmmAdv

evpn
  vni 10501 l2
    rd auto
    route-target import auto
    route-target export auto
vrf context VRF1
  vni 50001
rd auto
address-family ipv4 unicast
   route-target both auto
   route-target both auto mvpn
   route-target both auto evpn
address-family ipv6 unicast
   route-target both auto
   route-target both auto mvpn
   route-target both auto evpn

Note: Incase of vPC leafs, you need to configure identical "mvpn vri id" on both
the vPC nodes. For example:

router bgp 100
  mvpn vri id 2001
```

Note MVPN VRI ID must be unique within the network or setup. That is, if the network has three different sets of vPC pairs, each pair must have a different VRI ID.

- **Spine - sample configuration of IPv6 multicast underlay:**
  - NVE Configuration

```
nv overlay evpn
```

  - PIMv6 Configuration

```
feature pim6

ipv6 pim rp-address 101:101:101:101::101 group-list ff00::/8
ipv6 pim anycast-rp 101:101:101:101::101 102:102:102:102::102
ipv6 pim anycast-rp 101:101:101:101::101 103:103:103:103::103

interface loopback101
  ipv6 address 101:101:101:101::101/128
  ipv6 router ospfv3 v6u area 0.0.0.0
  ipv6 pim sparse-mode

interface loopback102
  ipv6 address 102:102:102:102::102/128
  ipv6 router ospfv3 v6u area 0.0.0.0
  ipv6 pim sparse-mode

interface Ethernet1/50/1
  ipv6 address 27:50:1:1::2/64
  ipv6 pim sparse-mode
  no shutdown
```

  - BGP Configuration

```
feature bgp

router bgp 100
        router-id 172.16.40.1
         address-family ipv4 mvpn
        address-family ipv6 mvpn
        address-family l2vpn evpn
        neighbor 172:16:1:1::1
          remote-as 100
          update-source loopback0
          address-family ipv4 mvpn
            send-community
            send-community extended
             route-reflector-client
          address-family ipv6 mvpn
           send-community
            send-community extended
             route-reflector-client
          address-family l2vpn evpn
          send-community
            send-community extended
             route-reflector-client
```