Multicast Forwarding

Feature Information for Multicast Forwarding

Table 1: Feature Information for Multicast Forwarding

Feature	Releases	Feature Information
Border leaf conversation Learning	7.2(0)N1(1)	Included a new section on *Conversational learning on Border Leaf*. Enable conversational learning on all leaf nodes.

Multicast Forwarding

Multicast forwarding allows the hardware engine to forward IP packets, instead of having multicast traffic sent to the line card CPU for slower path processing.

- PIM SM, SSM and Bidir are supported in DFA.

- Any source connected to a leaf will route traffic to local receivers and also route to fabric SVI if there are any interested receivers only.

- All receivers expecting traffic from remote source will receive routed traffic through fabric SVI.

• RP's can be configured on border leafs or any router outside of the fabric but should be reachable via border leafs only.

Enabling Fabric Multicast

Before you begin

• Feature Fabric Forwarding should be enabled

• Ensure fabric unicast is configured and unicast reachability and BGP peering is successfully established

• For multi-tenancy lite version system fabric dynamic VLANs must be configured. For multi-tenancy full version system fabric bridge-domain must be configured.

• Ensure RP is configured on border leafs or any external router connected or reachable via border leaf only

• Ensure RP reachability is from all leaf nodes and border leafs

SUMMARY STEPS

1. **configure terminal**
2. **install feature-set fabric**
3. **feature-set fabric**
4. **feature fabric forwarding**
5. **[no] feature pim**
6. **[no] feature fabric multicast**
7. **[no] ip multicast fabric-forwarding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**install feature-set fabric** **Example:** `switch(config)# install feature-set fabric`	Install feature fabric.
Step 3	**feature-set fabric** **Example:** `switch(config)# feature-set fabric`	Install feature fabric.
Step 4	**feature fabric forwarding** **Example:** `switch(config)# feature fabric forwarding`	Enable/Disable Fabric Forwarding Protocol: Host Mobility Manager (HMM).

	Command or Action	Purpose
Step 5	**[no] feature pim** **Example:** `switch(config)# feature pim`	Enables PIM. **Note** No need to enable ip pim sparse-mode under SVI/BDI configuration.
Step 6	**[no] feature fabric multicast** **Example:** `switch(config)# feature fabric multicast`	Enable/Disable NGMVPN features.
Step 7	**[no] ip multicast fabric-forwarding** **Example:** `switch(config)# ip multicast fabric-forwarding`	Enables multicast enhanced fabric forwarding.

Configuring Host VLANs

PIM is enabled automatically and operates in passive mode on a server facing SVI/BDI.

In order to enable IGMPv3, the following configuration must be done on the server facing SVI/BDI.

SUMMARY STEPS

1. **ip igmp version 3**
2. **show ip pim**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**ip igmp version 3**
	
	Example:
	Multi-tenancy lite version
	
	`Vlan 100`
	`  Mode fabricpath`
	
	`Int vlan 100`
	`   Ip address 100.1.1.1/24`
	`   Ip igmp version 3   // use this config for PIM`
	` SSM`
	
	Example:
	Multi-tenancy full version
	
	`Bridge-domain 100`
	
	`Int bdi 100`
	`   Ip address 100.1.1.1/24`
	`   Ip igmp version 3   // use this config for PIM`
		Configures ip igmp for SSM.

	Command or Action	**Purpose**
	``` SSM ```	

Step 2	**show ip pim**
	
	Example:
	
	Multi-tenancy lite version
	
	```Device# show ip pim interface vlan 77``` ``` PIM Interface Status for VRF "ORG:RED" Vlan77, Interface status: protocol-up/link-up/admin-up IP address: 10.1.1.2 IP subnet: 10.1.1.0/24 PIM DR: 10.1.1.2, DR's priority: 1 PIM neighbor count: 0 PIM hello interval: 30 secs, next hello sent in: 0.000000 PIM neighbor holdtime: 105 secs PIM configured DR priority: 1 PIM configured DR delay: 3 secs PIM border interface: no PIM GenID sent in Hellos: 0x147bad9d PIM Hello MD5-AH Authentication: disabled PIM Neighbor policy: none configured PIM Join-Prune inbound policy: none configured PIM Join-Prune outbound policy: none configured PIM Join-Prune interval: 1 minutes PIM Join-Prune next sending: 1 minutes PIM BFD enabled: no PIM passive interface: yes PIM VPC SVI: no PIM Auto Enabled: yes PIM Interface Statistics, last reset: never General (sent/received): Hellos: 0/0 (early: 0), JPs: 0/0, Asserts: 0/0 ```
	
	Example:
	
	Multi-tenancy full version
	
	```Device# show ip pim interface bdi 77``` ``` PIM Interface Status for VRF "ORG:RED" bd77, Interface status: protocol-up/link-up/admin-up IP address: 10.1.1.2 IP subnet: 10.1.1.0/24 PIM DR: 10.1.1.2, DR's priority: 1 PIM neighbor count: 0 PIM hello interval: 30 secs, next hello sent in: 0.000000 PIM neighbor holdtime: 105 secs PIM configured DR priority: 1 PIM configured DR delay: 3 secs PIM border interface: no PIM GenID sent in Hellos: 0x147bad9d PIM Hello MD5-AH Authentication: disabled PIM Neighbor policy: none configured PIM Join-Prune inbound policy: none configured PIM Join-Prune outbound policy: none configured PIM Join-Prune interval: 1 minutes PIM Join-Prune next sending: 1 minutes PIM BFD enabled: no ```
		Verifies PIM in passive state.

Command or Action	Purpose
| ```
PIM passive interface: yes
PIM VPC BDI: no
PIM Auto Enabled: yes
PIM Interface Statistics, last reset: never
General (sent/received):
Hellos: 0/0 (early: 0), JPs: 0/0, Asserts: 0/0
``` | |

Configuring BGP MVPN

For Spine configured as route reflector ensure the following configurations have been added under MVPN address family.

SUMMARY STEPS

1. **configure terminal**
2. **router bgp asn**
3. **[no] address-family**
4. **[no] neighbor** *<ip address>***remote-as***<AS number>*
5. **[no] address-family**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp asn** **Example:** `switch(config)# router bgp asn`	Creates BGP instance.
| **Step 3** | **[no] address-family**<br><br>**Example:**<br><br>```
switch(config-router)# address-family ipv4 mvpn
switch(config-router)# additional-paths send
switch(config-router)# additional-paths selection
 route-map RMAP
switch(config-router)#
switch(config-router)# address-family ipv6 mvpn
switch(config-router)# additional-paths send
switch(config-router)# additional-paths selection
 route-map RMAP

//Note: RMAP is a route map that can be configured
 as below:
switch(config-router)# route-map RMAP permit 10
switch(config-router)# set path-selection all
advertise
``` | Configures multicast VPN. |

	Command or Action	Purpose
Step 4	**[no] neighbor** *<ip address>***remote-as***<AS number>* **Example:** `switch(config)# neighbor 10.1.1.2 remote-as 65000`	Configures a BGP peer.

Step 5	**[no] address-family**
	
	Example:
	`switch(config-router)# address-family ipv4 mvpn`
	`switch(config-router)# send-community extended`
	`switch(config-router)# route-reflector-client`
	`switch(config-router)# address-family ipv6 mvpn`
	`switch(config-router)# send-community extended`
	`switch(config-router)# route-reflector-client`
		Send extended community attribute to BGP peer. Enable route reflector on spine.

Verifying BGP MVPN configuration

This example shows the BGP neighbors currently peered. Verify the BGP configs if you do not get an output:

```
switch(config)# show bgp ipv4 mvpn summary

BGP summary information for VRF default, address family IPv4 MVPN
BGP router identifier 10.1.1.1, local AS number 10
BGP table version is 62, IPv4 MVPN config peers 1, capable peers 1
6 network entries and 10 paths using 936 bytes of memory
BGP attribute entries [8/1120], BGP AS path entries [0/0]
BGP community entries [0/0], BGP clusterlist entries [2/8]

Neighbor        V    AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.1.2        4    10   32823   32787        62    0    0     3w1d 4
```

Ensure additional-paths capability is exchanged and active on the Spine-Leaf or Spine-Border Leaf peering

```
switch(config)# show bgp ipv4 mvpn neighbors | grep -A 4 Additional

Additional Paths capability: advertised received
  Additional Paths Capability Parameters:
  Receive capability advertised to Peer for AF:
    IPv4 Unicast  IPv6 Unicast  IPv4 MVPN  IPv6 MVPN
  Send capability received from Peer for AF:
    IPv4 Unicast  IPv6 Unicast  IPv4 MVPN  IPv6 MVPN
```

Border Leaf Specific Configurations

PIM RP on Border leaf

In DFA, it is required that an RP is configured on the border leaf for redundancy and load balancing. An RP on a border leaf can be configured for the desired mode (bidir / ASM) as described in the Cisco Nexus 5000 Series Switches, see Multicast Routing Configuration Guide.

The following modes of RP configuration are supported:

- Static RP

- Anycast RP

- BSR mechanism

- Auto RP

SUMMARY STEPS

1. **Show fabric multicast ipv4 ssm-range vrf all**
2. **Show fabric multicast ipv4 mroute vrf all**
3. **show ip pim group-range vrf all**
4. **show fabric multicast { ipv4 | ipv6 } rp-grange vrf all**
5. **show bgp ipv4 mvpn**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Show fabric multicast ipv4 ssm-range vrf all** **Example:** `switch(config)# Show fabric multicast ipv4 ssm-range vrf all`	Verify that ssm range is configured correctly in fabric multicast.
Step 2	**Show fabric multicast ipv4 mroute vrf all** **Example:** `switch(config)# Show fabric multicast ipv4 mroute vrf all`	Verify multicast routes are communicated correctly to fabric_mcast.
Step 3	**show ip pim group-range vrf all** **Example:** `switch(config)# Show ip pim group-range vrf all`	Verify RP is configured correctly on the border leaf. Ensure RP is configured correctly in PIM.

Step 4	**show fabric multicast { ipv4	ipv6 } rp-grange vrf all**
		Example:
		`switch(config)# show fabric multicast ipv4 rp-grange vrf all`
		`switch(config)# show fabric multicast ipv6 rp-grange vrf all`
			Verify RP is configured correctly on the border leaf. Ensure RP information is communicated correctly to fabric_mcast.

Step 5	**show bgp ipv4 mvpn** **Example:** `switch(config)# Show bgp ipv4 mvpn`	Ensure sure RP information is communicated correctly to BGP. Check for Group-to-RP mapping.

Various PIM RP Placements

RP Configured on the Border Leaf

- RP can be configured on the border leaf.

- There can be multiple border leafs for redundancy. Anycast RP can be configured between multiple RP across multiple border leafs.
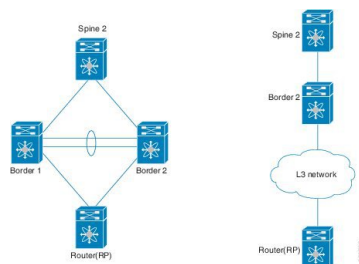
RP Configured Outside of Border Leaf

- RP can be configured outside the fabric on a router connected with Layer-3 links (physical port, subinterfaces on Layer-3 port channel) to the border leaf or border leafs in a vPC+.

> **Note** Use of legacy SVI on border leaf nodes in a vPC+ setup is not supported, as this will result in multicast traffic duplication within the DFA fabric.

- Any IGP like OSPF can be run to ensure reachability to the external RP.

Figure 1: RP configured outside of Border Leaf

Various PIM configurations supported

- Static, BSR, Anycast RP and Auto RP can be configured on the router.

- Ensure unicast reachability to RP from the border leafs with any routing protocols like OSPF.

Sample Border Leaf

Sample border leaf BGP configuration.

```
Device# show running-config bgp

!Time: Wed Feb 28 20:45:54 2001

version 7.1(0)N1(1)
feature bgp

router bgp 65101
  router-id 10.1.1.2
  fabric-soo 65101:1
```

```
address-family ipv4 unicast
  redistribute hmm route-map FABRIC-RMAP-REDIST-HOST
  maximum-paths ibgp 2
  nexthop trigger-delay critical 250 non-critical 10000
  nexthop route-map bgp_next_hop_filter
  additional-paths receive
  additional-paths selection route-map ALL-PATHS
address-family ipv6 unicast
  redistribute hmm route-map FABRIC-RMAP-REDIST-V6HOST
  maximum-paths ibgp 2
  nexthop trigger-delay critical 250 non-critical 10000
  additional-paths receive
  additional-paths selection route-map ALL-PATHS
  address-family vpnv4 unicast
  nexthop trigger-delay critical 250 non-critical 10000
  additional-paths receive
address-family vpnv6 unicast
  nexthop trigger-delay critical 250 non-critical 10000
  additional-paths receive
address-family ipv4 mvpn
  nexthop trigger-delay critical 250 non-critical 10000
  additional-paths receive
  additional-paths selection route-map ALL-PATHS
address-family ipv6 mvpn
  nexthop trigger-delay critical 250 non-critical 10000
  additional-paths receive
  additional-paths selection route-map ALL-PATHS
  neighbor 10.1.1.2 remote-as 65101
address-family ipv4 unicast
  send-community both
  route-map deny-default-route in
  next-hop-self
address-family ipv6 unicast
  send-community both
address-family vpnv4 unicast
  send-community extended
  route-map deny-default-route in
address-family vpnv6 unicast
  send-community extended
address-family ipv4 mvpn
  send-community both
address-family ipv6 mvpn
  send-community both
```

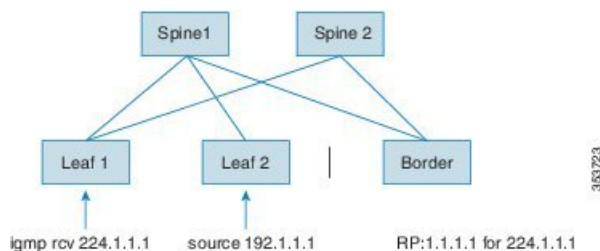# Configuring Legacy SVI

On Border Leafs:

- If an SVI over FabricPath VLAN is explicitly configured with PIM, that is configured with "ip pim sparse-mode" then such SVIs become legacy SVIs.

- Multicast forwarding using legacy SVI is same as in non-DFA environment with PIM enabled. For example, PIM in active mode, PIM hellos exchanged between multicast routers.

- Legacy SVI should be used to communicate with services outside fabric or with non-DFA routers like Cisco Nexus 5000 Series Switches or Services (for example, Firewall and Loadbalancer) .

- Use of legacy SVI on border leaf nodes in a vPC+ setup is not supported, as this will result in multicast traffic duplication within the DFA fabric.

On Leaf Nodes:

- Any VLAN without SVI is considered as a legacy VLAN.

Traffic Flow in a Spine Leaf Topology

Figure 2: Traffic flow in a simple spine leaf topology

Note The default drop route value for Layer-3 unregistered packets is 224/4. Ensure not to configure bidir with prefix of 224/4 as this may result in an unexpected behavior.

Multi-tenancy lite version

Consider the receiver VLAN 20 is on Leaf 1, the source VLAN 30 on Leaf 2 and the RP 1.1.1.1 for group 224.1.1.1 on the border leaf:

1. Consider Leaf 1 receives joins on VLAN 20, this triggers a join through BGP/NGMVPN (fabric_mcast process) towards the fabric.

   The mroute state on Leaf1 is shown in the following example:

   ```
   (*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
     Incoming interface: Vlan3001, RPF nbr: 10.1.1.4
     Outgoing interface list: (count: 1)
      Vlan20, uptime: 00:35:19, igmp
   ```

2. Border leaf listens to this join from NGMVPN imported through BGP and simulate a PIM registration behavior. Note that joins are exported through BGP/NGMVPN and not through PIM.

   The mroute state on border leaf is shown in the following example:

   ```
   (*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
     Incoming interface: Loopback0, RPF nbr: 10.1.1.4
     Outgoing interface list: (count: 1)
     Vlan3001, uptime: 00:35:19, igmp
   ```

3. In Leaf 2, when source traffic is received for a particular group on VLAN 30, NGMPVN would identify that there is an interested receiver in the fabric and adds fabric SVI as the OIF.

   ```
   (192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, fabric_mcast mrib pim ip
     Incoming interface: Vlan30, RPF nbr: 10.1.1.4
     Outgoing interface list: (count: 1) (Fabric OIF)
       Vlan3001, uptime: 00:50:19, mrib
   ```

   Once the traffic is sent to the fabric, interested receivers in remote leafs would accept it and route to all the local receivers. In this example Leaf 1 receives the traffic from Fabric SVI and route it to the local receiver VLAN 20.

The states of Leaf1 is shown in the following example:

```
(*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
  Incoming interface: Vlan3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
    Vlan20, uptime: 00:35:19, igmp

(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, igmp ip pim
  Incoming interface: Vlan3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
    Vlan20, uptime: 00:50:19, mrib
```

Traffic coming from fabric is tagged with the VNI and hence picks the right VRF and route is based on the VRF to VNI and fabric VLAN mapping.

The state of Leaf2 is shown in the following example:

```
(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, fabric_mcast mrib pim ip
  Incoming interface: Vlan30, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1) (Fabric OIF)
    Vlan3001, uptime: 00:50:19, mrib
```

The states of Border Leaf is shown in the following example:

```
(*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
  Incoming interface: Loopback0, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
    Vlan20, uptime: 00:35:19, igmp

(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, igmp ip pim
  Incoming interface: Vlan3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1) (Fabric OIF)
    Vlan3001, uptime: 00:50:19, mrib
```

Multi-tenancy full version

Consider the receiver bridge-domain 20 is on Leaf 1, the source bridge-domain 30 on Leaf 2 and the RP 1.1.1.1 for group 224.1.1.1 on the border leaf:

1. Consider Leaf 1 receives joins on bridge-domain 20, this triggers a join through BGP/NGMVPN (fabric_mcast process) towards the fabric.

   The mroute state on Leaf1 is shown in the following example:

   ```
   (*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
     Incoming interface: bd3001, RPF nbr: 10.1.1.4
     Outgoing interface list: (count: 1)
      bridge-domain20, uptime: 00:35:19, igmp
   ```

2. Border leaf listens to this join from NGMVPN imported through BGP and simulate a PIM registration behavior. Note that joins are exported through BGP/NGMVPN and not through PIM.

   The mroute state on border leaf is shown in the following example:

   ```
   (*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
     Incoming interface: Loopback0, RPF nbr: 10.1.1.4
     Outgoing interface list: (count: 1)
     bd3001, uptime: 00:35:19, igmp
   ```

3. In Leaf 2, when source traffic is received for a particular group on bridge-domain 30, NGMPVN would identify that there is an interested receiver in the fabric and adds fabric SVI as the OIF.

   ```
   (192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, fabric_mcast mrib pim ip
     Incoming interface: bridge-domain30, RPF nbr: 10.1.1.4
   ```

```
        Outgoing interface list: (count: 1) (Fabric OIF)
          bd3001, uptime: 00:50:19, mrib
```

Once the traffic is sent to the fabric, interested receivers in remote leafs would accept it and route to all the local receivers. In this example Leaf 1 receives the traffic from Fabric SVI and route it to the local receiver VLAN 20.

The states of Leaf1 is shown in the following example:

```
(*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
  Incoming interface: bd3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
    bridge-domain20, uptime: 00:35:19, igmp

(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, igmp ip pim
  Incoming interface: bd3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
    bridge-domain20, uptime: 00:50:19, mrib
```

Traffic coming from fabric is tagged with the VNI and hence picks the right VRF and route is based on the VRF to VNI and fabric VLAN mapping.

The state of Leaf2 is shown in the following example:

```
(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, fabric_mcast mrib pim ip
  Incoming interface: bridge-domain30, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1) (Fabric OIF)
    bd3001, uptime: 00:50:19, mrib
```

The states of Border Leaf is shown in the following example:

```
(*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
  Incoming interface: Loopback0, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
    bridge-domain20, uptime: 00:35:19, igmp

(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, igmp ip pim
  Incoming interface: bd3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1) (Fabric OIF)
    bd3001, uptime: 00:50:19, mrib
```

Configuring VPC+ Border Leaf

Limitation

- Multiple FabricPath MDTs are currently not supported on VPC+ border leafs. It will be supported only in the future releases.

- Use of legacy SVI on border leaf nodes in a vPC+ setup is not supported, as this will result in multicast traffic duplication within the DFA fabric.

Configure VPC Border Leaf

1. Configure VPC+ between Border 1 and Border 2. Configure similar VPC+ configuration on Border 2 and verify if VPC+ is up and running.

2. Enable enhanced multicast.

3. Ensure BGP peering with the fabric spine from both the border leafs.

4. Configure border leafs.

5. Configure RP in the border Leafs (PIM RP on Border leaf , on page 6).

6. One of the border leaf will be considered if same RP address is configured on both the border leafs in VPC+ based on unicast reachability.
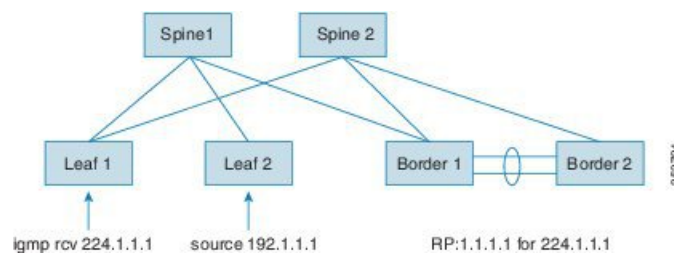
Before you begin

- VPC+ is configured between border routers

- Feature Fabric Forwarding must be enabled

- Ensure fabric unicast is configured and unicast reachability and BGP peering is successfully established

- System fabric dynamic VLANs should be configured

- Enhanced fabric multicast must be enabled

Traffic Flow with Border Leafs in VPC

Overview

Border leafs can be configured in VPC+. VPC+ configuration remains same as configuring routers in VPC+ FabricPath topologies.

Figure 3: Traffic flow with Border Leafs in VPC

Note   The default drop route value for Layer-3 unregistered packets is 224/4. Ensure not to configure bidir with prefix of 224/4 as this may result in an unexpected behavior.

Multi-tenancy lite version

1. Consider Leaf 1 receives joins on VLAN 20, this triggers a join through BGP/NGMVPN (fabric_mcast process) towards the fabric.

   The mroute state on Leaf1 is shown in the following example:

```
(*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
  Incoming interface: Vlan3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
   Vlan20, uptime: 00:35:19, igmp
```

2. One of the border leaf listens to this join from NGMVPN imported through BGP and simulate a PIM registration behavior. Note that joins are exported through BGP/NGMVPN and not through PIM.

3. In Leaf 2, when source traffic is received for a particular group on VLAN 30, NGMPVN would identify that there is an interested receiver in the fabric and adds fabric SVI as the OIF.

```
(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, fabric_mcast mrib pim ip
  Incoming interface: Vlan30, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1) (Fabric OIF)
    Vlan3001, uptime: 00:50:19, mrib
```

Once the traffic is sent to the fabric, interested receivers in remote leaf nodes would accept it and route to all the local receivers. In this example Leaf 1 receives the traffic from fabric SVI and route it to the local receiver VLAN 20.

Multi-tenancy full version

1. Consider Leaf 1 receives joins on bridge-domain 20, this triggers a join through BGP/NGMVPN (fabric_mcast process) towards the fabric.

   The mroute state on Leaf1 is shown in the following example:

```
(*, 224.1.1.1/32), uptime: 00:35:19, igmp ip pim
  Incoming interface: bd3001, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1)
   bridge-domain20, uptime: 00:35:19, igmp
```

2. One of the border leaf listens to this join from NGMVPN imported through BGP and simulate a PIM registration behavior. Note that joins are exported through BGP/NGMVPN and not through PIM.

3. In Leaf 2, when source traffic is received for a particular group on VLAN 30, NGMPVN would identify that there is an interested receiver in the fabric and adds BDI as the OIF.

```
(192.1.1.1, 224.1.1.1/32), uptime: 00:50:19, fabric_mcast mrib pim ip
  Incoming interface: bridge-domain30, RPF nbr: 10.1.1.4
  Outgoing interface list: (count: 1) (Fabric OIF)
    bd3001, uptime: 00:50:19, mrib
```

Once the traffic is sent to the fabric, interested receivers in remote leaf nodes would accept it and route to all the local receivers. In this example Leaf 1 receives the traffic from BDI and route it to the local receiver bridge-domain 20.

Conversational learning on Border Leaf

You can enable conversational learning on all leaf nodes by using the **fabric forwarding conversational-learning all** command. For this to work, the subnet needs to be instantiated on the leaf. But in case of a border leaf, this is not true as the border leaf might not have any hosts connected to it. So, the routes will always get installed in forwarding information base (FIB). But border leaf is the point of heavy load in the network and needs to conserve precious forwarding space. In this regard, we can add configuration at the border leaf for each subnet using the **fabric forwarding aggregate-subnet-prefix <> ** command.

Another way to prevent having to add this configuration for each subnet is to have border leaf learn about aggregate subnet prefixes from the leaf nodes with special cost community value. The CC value tells the software to not install an ECMP route for this prefix, but to install a glean entry in the FIB, to achieve this, the leaf nodes will need to export the anycast gateway IP into iBGP with a special cost community value as shown below.

To enable Layer-3 conversational learning-based route download into the forwarding information base (FIB), use the **fabric forwarding conversational-learning all** command. And to configure the conversational aging

timeout value, use the **fabric forwarding conversational-aging** *timeout* command. For more information on these conversational commands, see Cisco Dynamic Fabric Automation Command Reference.

Note

To enable this feature on border leaf, all the internal leaf nodes in the fabric must have a route-map. And ensure that conversational learning is enabled on border leaf.

The following example shows the leaf configuration:

Multi-tenancy lite version

```
route-map direct permit 10
  match tag 12345
  set extcommunity cost pre-bestpath 112 4294967295 transitive

router bgp 100
  router-id 100.1.1.5
  neighbor 100.1.1.1 remote-as 100
    update-source Vlan2
    address-family vpnv4 unicast
      send-community extended
  vrf vpn4
    address-family ipv4 unicast
      redistribute direct route-map direct
```

Multi-tenancy full version

```
route-map direct permit 10
  match tag 12345
  set extcommunity cost pre-bestpath 112 4294967295 transitive

router bgp 100
  router-id 100.1.1.5
  neighbor 100.1.1.1 remote-as 100
    update-source bridge-domain2
    address-family vpnv4 unicast
      send-community extended
  vrf vpn4
    address-family ipv4 unicast
      redistribute direct route-map direct
```

To prevent all direct routes from being distributed, you must tag the routes and export only the ones with the tag:

```
interface Vlan40
  no shutdown
  vrf member vpn4
  ip address 40.1.1.1/8 tag 12345
  fabric forwarding mode proxy-gateway
```

The following example shows how to configure per-vrf label allocation on border leaf:

```
router bgp 100
  vrf cisco:vrf-11
    address-family ipv4 unicast
      maximum-paths 32
```

```
maximum-paths ibgp 32
  label-allocation-mode per-vrf
address-family ipv6 unicast
  maximum-paths 32
  maximum-paths ibgp 32
  label-allocation-mode per-vrf
```

The following show command example displays the type:

```
Switch# show fabric forwarding ip aggregate-subnet-prefix vrf vpn3

HMM aggregate-subnet-prefix IPv4 routing table information for VRF vpn3

Status: *-valid, x-deleted, c-cleaned in 00:03:36

      Host                  Type              Flags
*   40.1.1.1/8          static | dynamic        0x40204
```