VyOS
The universal networking platform

Deployment Guide | Technical Doc

L2VPN IN EVPN-VXLAN DATA CENTER

April 2025

Introduction

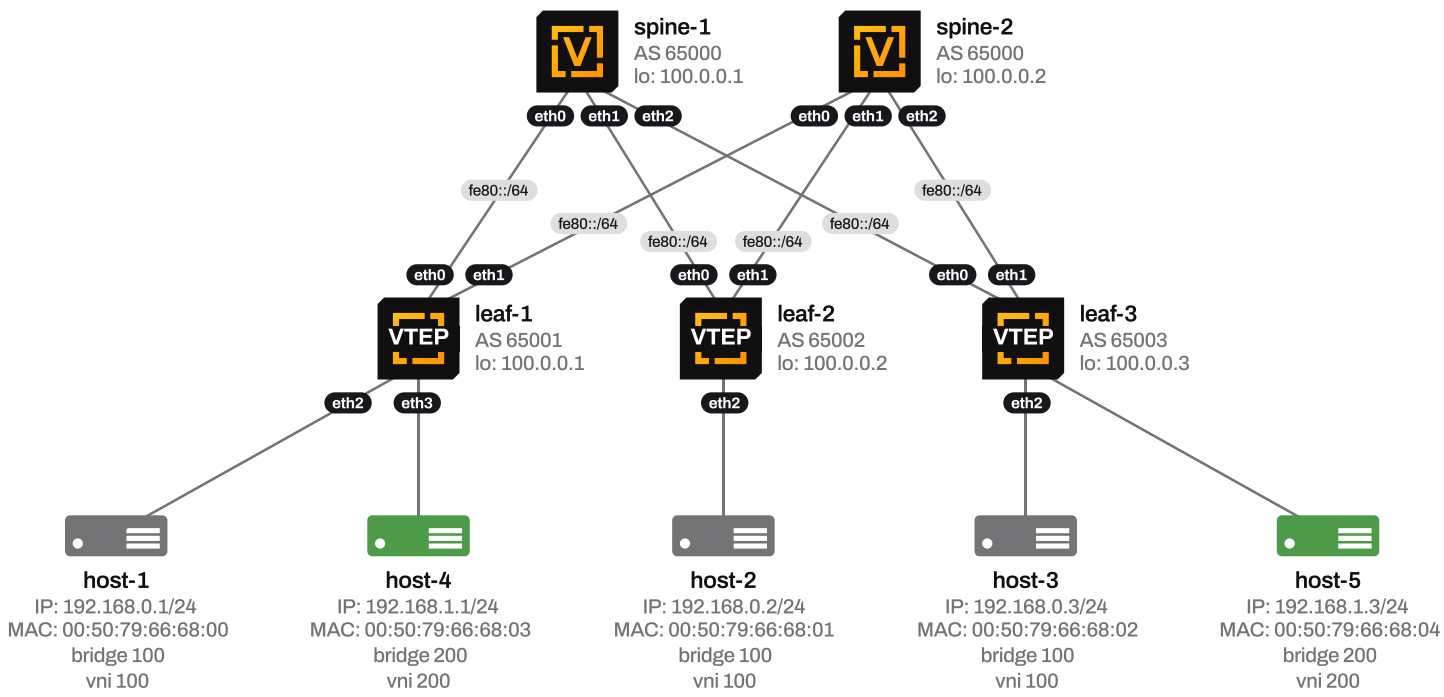
This document is intended to be a contribution for VyOS Community, and a guide to show you how to configure on VyOS a L2VPN service in a EVPN-VXLAN data-center fabric. We are also using the BGP unnumbered feature, which dynamically discovers IPv6 neighbors, reducing the burden of manually configuring an IPv6 underlay, setting up each individual BGP session. This feature is especially useful for large data centers with hundreds of leafs, saving IP addressing, reducing the number of configuration statements, and avoiding configuration errors. Although the underlay uses native IPv6, it also supports IPv4 routes with IPv6 next-hops as defined in RFC5549.

Scenario | Topology

The lab consists of a simple 2-tier Clos topology with two spine and 3 leafs routers.

As illustrated in the figure, there are 2 customers, customer “blue” and customer “green” connected to 3 different sites or leafs that needs to share a common L2 domain. A bridged overlay enables ethernet bridging between leaf devices in an EVPN network. This type of overlay extends L2 domains between leaf devices over VXLAN tunnels. It serves as an overlay solution for data center networks that require ethernet connectivity without the need for inter-VLAN routing.

In this scenario, the loopback interfaces on leaf devices function as VXLAN tunnel endpoints (VTEPs). These tunnels allow leaf devices to forward VLAN traffic to other leaf devices.



Configurations and Deployment

Underlay configuration

1. Fabric interfaces configuration and loopbacks. We just need to configure some parameters like description, MAC address if required and enable IPv6. No IPv6 address configuration is required, IPv6 link-local addresses will be used later for BGP sessions establishment. Loopback interfaces will be used as VTEP for each leaf.

```
# fabric interfaces config
set interfaces ethernet eth0 description 'to_spine-1'
set interfaces ethernet eth0 hw-id '0c:82:1e:a4:00:00'
set interfaces ethernet eth0 ipv6
set interfaces ethernet eth1 description 'to_spine-2'
set interfaces ethernet eth1 hw-id '0c:82:1e:a4:00:01'
set interfaces ethernet eth1 ipv6
set interfaces ethernet eth2 description 'to_host-1_customer-blue'
set interfaces ethernet eth2 hw-id '0c:82:1e:a4:00:02'
set interfaces ethernet eth3 description 'to_host-4_customer-green'
set interfaces ethernet eth3 hw-id '0c:82:1e:a4:00:03'
set interfaces ethernet eth4 hw-id '0c:82:1e:a4:00:04'

# loopback interface address for each leaf an spine as shown in diagram above

set interfaces loopback lo address <loopback>
```

2. IPv6 router-advertisement (RA). When the RA function is enabled, the interface periodically transmits RA messages. These messages help identify the remote neighbor's link-local IP address, which is essential for neighbor discovery and the dynamic peering process.

```
set service router-advert interface eth0
set service router-advert interface eth1
```

3. BGP. IPv4 unicast address family will be used for the advertisement of leaf's loopbacks, wich will be later used as VTEPs for VXLAN encapsulation.

   BGP sessions will be established over IPv6 link-local addresses and neighbor auto discovering using BGP unnumbered feature. That feature is enabled with the following command under bgp context: **"neighbor <interface-name> interface v6only peer-group <peer-gorup>"**.

   IPv4 routes with IPv6 next-hops (RFC5549) is enabled with the following command under bgp context: **"bgp peer-group spine capability extended-nexthop"**

Leafs

```
set protocols bgp address-family ipv4-unicast redistribute connected
set protocols bgp neighbor eth0 interface v6only peer-group 'spine'
set protocols bgp neighbor eth1 interface v6only peer-group 'spine'
set protocols bgp parameters router-id '10.0.0.[x]'
set protocols bgp peer-group spine address-family ipv4-unicast
set protocols bgp peer-group spine capability dynamic
set protocols bgp peer-group spine capability extended-nexthop
set protocols bgp peer-group spine remote-as 'external'
set protocols bgp system-as '6500[x]'
```

Spines

```
set protocols bgp address-family ipv4-unicast redistribute connected
set protocols bgp neighbor eth0 interface v6only peer-group 'leaves'
set protocols bgp neighbor eth1 interface v6only peer-group 'leaves'
set protocols bgp neighbor eth2 interface v6only peer-group 'leaves'
set protocols bgp parameters bestpath as-path multipath-relax
set protocols bgp parameters router-id '100.0.0.[x]'
set protocols bgp peer-group leaves address-family ipv4-unicast
set protocols bgp peer-group leaves capability extended-nexthop
set protocols bgp peer-group leaves remote-as 'external'
set protocols bgp system-as '65000'
```

Overlay configuration

1. BGP. L2vpn-evpn willl be used for the overlay host's MAC addresses advertisement.

```
set protocols bgp address-family l2vpn-evpn advertise-all-vni
set protocols bgp peer-group spine address-family l2vpn-evpn
```

2. VXLAN interface. Create VXLAN interfaces, set MTU, UDP port number for VXLAN encapsulation, source address and VNI. Customer bridge domain will be later associated with each VNI for VXLAN encapsulation. The same VNI value is configured on all leafs to identify the L2VPN service.

```
set interfaces vxlan vxlan100 mtu '1500'
set interfaces vxlan vxlan100 parameters nolearning
set interfaces vxlan vxlan100 port '4789'
set interfaces vxlan vxlan100 source-address '10.0.0.[x]'     # leaf's loopback as VTEP
set interfaces vxlan vxlan100 vni '100'
```

```
set interfaces vxlan vxlan200 mtu '1500'
set interfaces vxlan vxlan200 parameters nolearning
set interfaces vxlan vxlan200 port '4789'
set interfaces vxlan vxlan200 source-address '10.0.0.[x]'      # leaf's loopback as VTEP
set interfaces vxlan vxlan200 vni '200'
```

3. Bridge domains. Create bridge domains and add interfaces connected to hosts and vxlan interfaces. We create two bridge domains, one for each customer.

```
set interfaces bridge br100 description 'customer blue'
set interfaces bridge br100 member interface eth2
set interfaces bridge br100 member interface vxlan100
```

```
set interfaces bridge br200 description 'customer green'
set interfaces bridge br200 member interface eth3
set interfaces bridge br200 member interface vxlan200
```

Validation and Troubleshooting

Underlay

1. Verify fabric interfaces are up

```
vyos@spine-1# run show interfaces
 Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
 Interface    IP Address    MAC               VRF        MTU  S/L    Description
 -----------  ------------  ----------------  -------  -----  -----  -------------
 eth0         -             0c:39:65:cc:00:00  default   1500  u/u    to_leaf-1
 eth1         -             0c:39:65:cc:00:01  default   1500  u/u    to_leaf-2
 eth2         -             0c:39:65:cc:00:02  default   1500  u/u    to_leaf-3
 eth3         -             0c:39:65:cc:00:03  default   1500  u/D
 eth4         -             0c:39:65:cc:00:04  default   1500  u/u
 lo           127.0.0.1/8   00:00:00:00:00:00  default  65536  u/u
              100.0.0.1/32
              ::1/128
 [edit]
 vyos@spine-1#
```

```
vyos@leaf-1# run show interfaces
 Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
 Interface    IP Address        MAC               VRF        MTU  S/L    Description
 -----------  ----------------  ----------------  -------  -----  -----  -----------------------
 br100        192.168.0.100/24  b2:de:98:da:3b:ba  default   1500  u/u    customer blue
 br200        192.168.1.254/24  1a:f0:6f:0d:a4:68  default   1500  u/u    customer green
 eth0         -                 0c:82:1e:a4:00:00  default   1500  u/u    to_spine-1
 eth1         -                 0c:82:1e:a4:00:01  default   1500  u/u    to_spine-2
 eth2         -                 0c:82:1e:a4:00:02  default   1500  u/u    to_host-1_customer-blue
 eth3         -                 0c:82:1e:a4:00:03  default   1500  u/u    to_host-4_customer-green
 eth4         -                 0c:82:1e:a4:00:04  default   1500  u/u
 lo           127.0.0.1/8       00:00:00:00:00:00  default  65536  u/u
              10.0.0.1/32
              ::1/128
 vxlan100     -                 86:eb:61:9c:9e:67  default   1500  u/u
 vxlan200     -                 26:22:08:1c:4f:0c  default   1500  u/u
 [edit]
 vyos@leaf-1#
```

2. Verify that all devices have learned the MAC and link-local address of all directly attached IPv6 neighbors.

```
vyos@spine-1# run show ipv6 neighbors
 Address                  Interface    Link layer address    State
 -----------------------  -----------  --------------------  ---------
 fe80::e39:65ff:fecc:2    eth2         0c:39:65:cc:00:02     STALE
 fe80::e37:beff:fef2:0    eth2         0c:37:be:f2:00:00     REACHABLE
 fe80::e82:1eff:fea4:0    eth0         0c:82:1e:a4:00:00     REACHABLE
 fe80::e39:65ff:fecc:0    eth0         0c:39:65:cc:00:00     STALE
 fe80::e84:a0ff:feee:0    eth1         0c:84:a0:ee:00:00     REACHABLE
 fe80::e39:65ff:fecc:1    eth1         0c:39:65:cc:00:01     STALE
 [edit]
 vyos@spine-1#
```

3. Verify that you can ping a neighbor using its link-local address.

```
vyos@leaf-1# run ping fe80::e39:65ff:fecc:0 interface eth0
 /bin/ping6: Warning: source address might be selected on device other than: eth0
 PING fe80::e39:65ff:fecc:0(fe80::e39:65ff:fecc:0) from :: eth0: 56 data bytes
 64 bytes from fe80::e39:65ff:fecc:0%eth0: icmp_seq=1 ttl=64 time=1.05 ms
 64 bytes from fe80::e39:65ff:fecc:0%eth0: icmp_seq=2 ttl=64 time=0.900 ms
 64 bytes from fe80::e39:65ff:fecc:0%eth0: icmp_seq=3 ttl=64 time=0.936 ms
 64 bytes from fe80::e39:65ff:fecc:0%eth0: icmp_seq=4 ttl=64 time=1.12 ms
 ^C
 --- fe80::e39:65ff:fecc:0 ping statistics ---
 4 packets transmitted, 4 received, 0% packet loss, time 3004ms
 rtt min/avg/max/mdev = 0.900/1.001/1.121/0.087 ms
 [edit]
 vyos@leaf-1#
```

4. Verify all devices have established BGP peering sessions to the directly connected neighbors.

```
vyos@leaf-1# run show bgp summary

 IPv4 Unicast Summary (VRF default):
 BGP router identifier 10.0.0.1, local AS number 65001 vrf-id 0
 BGP table version 9
 RIB entries 13, using 1248 bytes of memory
 Peers 2, using 40 KiB of memory
 Peer groups 1, using 64 bytes of memory

 Neighbor        V         AS   MsgRcvd   MsgSent   TblVer  InQ OutQ  Up/Down State/PfxRcd   PfxSnt
Desc
 eth0            4      65000       107       111        9    0    0 01:21:04            3        7
N/A
 eth1            4      65000       107       115        9    0    0 01:19:36            4        7
N/A

 Total number of neighbors 2
```

5. Verify that all nodes are advertising their loopback address and learning the loopback address from other nodes. Note that IPv4 leaves loopbacks are know via two equal cost paths (one path through spine 1, the other through spine 2), due to the implementation of RFC 5549 we have two IPv6 next-hops.

```
vyos@leaf-1#   run show ip route bgp
 Codes: K - kernel route, C - connected, S - static, R - RIP,
        O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
        T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
        f - OpenFabric,
        > - selected route, * - FIB route, q - queued, r - rejected, b - backup
        t - trapped, o - offload failure

 B>* 10.0.0.2/32 [20/0] via fe80::e39:65ff:fecc:0, eth0, weight 1, 01:26:02
   *                    via fe80::eb6:51ff:fecd:0, eth1, weight 1, 01:26:02
 B>* 10.0.0.3/32 [20/0] via fe80::e39:65ff:fecc:0, eth0, weight 1, 01:26:02
   *                    via fe80::eb6:51ff:fecd:0, eth1, weight 1, 01:26:02
 B>* 100.0.0.1/32 [20/0] via fe80::e39:65ff:fecc:0, eth0, weight 1, 01:27:29
 B>* 100.0.0.2/32 [20/0] via fe80::eb6:51ff:fecd:0, eth1, weight 1, 01:26:02
 [edit]
 vyos@leaf-1#
```

6. Verify ECMP Load Balancing. We should have two equal costs routes, one through each spine device. Next, verify ECMP in both the routing and forwarding tables (RIB and FIB).

```
vyos@leaf-1# ru show ip route 10.0.0.2
 Routing entry for 10.0.0.2/32
   Known via "bgp", distance 20, metric 0, best
   Last update 01:37:16 ago
   * fe80::e39:65ff:fecc:0, via eth0, weight 1
   * fe80::eb6:51ff:fecd:0, via eth1, weight 1

 [edit]
 vyos@leaf-1#
 [edit]
 vyos@leaf-1# ru show ip route 10.0.0.3
 Routing entry for 10.0.0.3/32
   Known via "bgp", distance 20, metric 0, best
   Last update 01:37:23 ago
   * fe80::e39:65ff:fecc:0, via eth0, weight 1
   * fe80::eb6:51ff:fecd:0, via eth1, weight 1

 [edit]
 vyos@leaf-1#
```

```
vyos@leaf-1# ru show ip route 10.0.0.2
vyos@leaf-1# ru show ip route forward 10.0.0.2
 10.0.0.2 nhid 22 proto bgp metric 20
        nexthop via inet6 fe80::e39:65ff:fecc:0 dev eth0 weight 1
        nexthop via inet6 fe80::eb6:51ff:fecd:0 dev eth1 weight 1
 [edit]
 vyos@leaf-1#
 [edit]
 vyos@leaf-1# ru show ip route forward 10.0.0.3
 10.0.0.3 nhid 22 proto bgp metric 20
        nexthop via inet6 fe80::e39:65ff:fecc:0 dev eth0 weight 1
        nexthop via inet6 fe80::eb6:51ff:fecd:0 dev eth1 weight 1
 [edit]
 vyos@leaf-1#
```

7. Verify reachability. Verify that leaves have connectivity over the underlay.

```
vyos@leaf-1# ru show ip route 10.0.0.2
vyos@leaf-1# run ping 10.0.0.2
 PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
 64 bytes from 10.0.0.2: icmp_seq=1 ttl=63 time=2.04 ms
 64 bytes from 10.0.0.2: icmp_seq=2 ttl=63 time=1.79 ms
 64 bytes from 10.0.0.2: icmp_seq=3 ttl=63 time=2.02 ms
 ^C
 --- 10.0.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
 rtt min/avg/max/mdev = 1.786/1.946/2.035/0.113 ms
 [edit]
 vyos@leaf-1#
 [edit]
 vyos@leaf-1# run ping 10.0.0.3
 PING 10.0.0.3 (10.0.0.3) 56(84) bytes of data.
 64 bytes from 10.0.0.3: icmp_seq=1 ttl=63 time=2.45 ms
 64 bytes from 10.0.0.3: icmp_seq=2 ttl=63 time=2.07 ms
 64 bytes from 10.0.0.3: icmp_seq=3 ttl=63 time=1.98 ms
 ^C
 --- 10.0.0.3 ping statistics ---
 3 packets transmitted, 3 received, 0% packet loss, time 2003ms
 rtt min/avg/max/mdev = 1.975/2.165/2.448/0.203 ms
 [edit]
 vyos@leaf-1#
```

```
vyos@leaf-1# run traceroute 10.0.0.2
 traceroute to 10.0.0.2 (10.0.0.2), 30 hops max, 60 byte packets
  1  100.0.0.2 (100.0.0.2)  2.812 ms  2.746 ms  2.731 ms
  2  10.0.0.2 (10.0.0.2)  2.717 ms  2.703 ms  2.684 ms
 [edit]
 vyos@leaf-1#
 [edit]
 vyos@leaf-1# run traceroute 10.0.0.3
 traceroute to 10.0.0.3 (10.0.0.3), 30 hops max, 60 byte packets
  1  100.0.0.2 (100.0.0.2)  2.528 ms  2.473 ms  2.459 ms
  2  10.0.0.3 (10.0.0.3)  3.466 ms  3.354 ms  3.339 ms
 [edit]
 vyos@leaf-1#
```

Overlay

1. Verification of EVPN VNIs, Rds and RTs

   The BGP route-distinguisher (RD) and route-target (RT) values are auto-derived. The same VNI value is configured on all leafs to identify the L2VPN service.

```
vyos@leaf-1# run show bgp l2vpn evpn vni
Advertise Gateway Macip: Disabled
Advertise SVI Macip: Enabled
Advertise All VNI flag: Enabled
BUM flooding: Head-end replication
VXLAN flooding: Enabled
Number of L2 VNIs: 2
Number of L3 VNIs: 0
Flags: * - Kernel
  VNI        Type RD             Import RT    Export RT    MAC-VRF  Site-of-Origin  Tenant  VRF
* 100        L2   10.0.0.1:3     65001:100    65001:100                             default
* 200        L2   10.0.0.1:2     65001:200    65001:200                             default
[edit]
```

2. Verification of EVPN IMET (type-3) route distribution. Once the service is enabled, each leaf advertises an IMET route for the L2VPN service. The output displays the IMET routes received by leaf-1. The list includes an IMET route advertised by each remote leaf.

```
vyos@leaf-1# run show bgp l2vpn evpn route type multicast
BGP table version is 3, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
EVPN type-1 prefix: [1]:[EthTag]:[ESI]:[IPlen]:[VTEP-IP]:[Frag-id]
EVPN type-2 prefix: [2]:[EthTag]:[MAClen]:[MAC]:[IPlen]:[IP]
EVPN type-3 prefix: [3]:[EthTag]:[IPlen]:[OrigIP]
EVPN type-4 prefix: [4]:[ESI]:[IPlen]:[OrigIP]
EVPN type-5 prefix: [5]:[EthTag]:[IPlen]:[IP]

   Network          Next Hop            Metric LocPrf Weight Path
                    Extended Community
Route Distinguisher: 10.0.0.1:2
 *> [3]:[0]:[32]:[10.0.0.1]
                    10.0.0.1                               32768 i
                    ET:8 RT:65001:200
Route Distinguisher: 10.0.0.1:3
 *> [3]:[0]:[32]:[10.0.0.1]
                    10.0.0.1                               32768 i
                    ET:8 RT:65001:100
Route Distinguisher: 10.0.0.2:2
 *  [3]:[0]:[32]:[10.0.0.2]
                    10.0.0.2                                   0 65000 65002 i
                    RT:65002:100 ET:8
 *> [3]:[0]:[32]:[10.0.0.2]
                    10.0.0.2                                   0 65000 65002 i
                    RT:65002:100 ET:8
 *  [3]:[0]:[32]:[10.0.0.3]
```

Encapsulation Tunnel: 8-VxLAN

```
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:200 ET:8
 *> [3]:[0]:[32]:[10.0.0.3]
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:200 ET:8
Route Distinguisher: 10.0.0.3:3
 *  [3]:[0]:[32]:[10.0.0.3]
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:100 ET:8
 *> [3]:[0]:[32]:[10.0.0.3]
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:100 ET:8

Displayed 5 prefixes (8 paths) (of requested type)
[edit]
vyos@leaf-1#
```

3. Verification of EVPN IMET (type-3) route attributes.

```
vyos@leaf-1# run show bgp l2vpn evpn route detail

Route Distinguisher: 10.0.0.2:2
BGP routing table entry for 10.0.0.2:2:[3]:[0]:[32]:[10.0.0.2]
Paths: (2 available, best #2)
  Advertised to non peer-group peers:
  eth0 eth1
  Route [3]:[0]:[32]:[10.0.0.2]
  65000 65002
    10.0.0.2 from eth0 (100.0.0.1)
      Origin IGP, valid, external
      Extended Community: RT:65002:100 ET:8
      Last update: Mon Apr  7 10:50:50 2025
      PMSI Tunnel Type: Ingress Replication, label: 100
  Route [3]:[0]:[32]:[10.0.0.2]
  65000 65002
    10.0.0.2 from eth1 (100.0.0.2)
      Origin IGP, valid, external, best (Older Path)
      Extended Community: RT:65002:100 ET:8
      Last update: Mon Apr  7 10:50:49 2025
      PMSI Tunnel Type: Ingress Replication, label: 100
```

The output displays the IMET route advertised by leaf-2 in detail:

- The next-hop is set to 10.0.0.2, the loopback interface of leaf-2.
- The route target auto-derived is set to target:65100:100.
- The BGP tunnel encapsulation type is set to VxLAN (ET:8).
- The route distinguisher auto-derived is set to 10.0.0.2:2.
- A PMSI tunnel attribute is included to specify how to send BUM traffic to leaf-2: The tunnel type indicates ingress replication (IR), the VNI label to use is 100 , and the destination IP address is leaf-2's loopback interface.

4. Verification of general information about VNIs, including flooding lists (remote VTEPs).

```
vyos@leaf-1# run show evpn vni 100
VNI: 100
 Type: L2
 Vlan: 1
 Bridge: br100
 Tenant VRF: default
 VxLAN interface: vxlan100
 VxLAN ifIndex: 11
 SVI interface: br100
 SVI ifIndex: 9
 Local VTEP IP: 10.0.0.1
 Mcast group: 0.0.0.0
 Remote VTEPs for this VNI:
  10.0.0.3 flood: HER
  10.0.0.2 flood: HER
 Number of MACs (local and remote) known for this VNI: 3
 Number of ARPs (IPv4 and IPv6, local and remote) known for this VNI: 0
 Advertise-gw-macip: No
 Advertise-svi-macip: No
[edit]
vyos@leaf-1#
```

5. Verification of MAC/IP route distribution

```
vyos@leaf-1# run show bgp l2vpn evpn route type macip
BGP table version is 8, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
EVPN type-1 prefix: [1]:[EthTag]:[ESI]:[IPlen]:[VTEP-IP]:[Frag-id]
EVPN type-2 prefix: [2]:[EthTag]:[MAClen]:[MAC]:[IPlen]:[IP]
EVPN type-3 prefix: [3]:[EthTag]:[IPlen]:[OrigIP]
EVPN type-4 prefix: [4]:[ESI]:[IPlen]:[OrigIP]
EVPN type-5 prefix: [5]:[EthTag]:[IPlen]:[IP]
   Network          Next Hop            Metric LocPrf Weight Path
                    Extended Community
Route Distinguisher: 10.0.0.1:2
 *> [2]:[0]:[48]:[00:50:79:66:68:03]
                    10.0.0.1                           32768 i
                    ET:8 RT:65001:200
Route Distinguisher: 10.0.0.1:3
 *> [2]:[0]:[48]:[00:50:79:66:68:00]
                    10.0.0.1                           32768 i
                    ET:8 RT:65001:100
Route Distinguisher: 10.0.0.2:2
 *> [2]:[0]:[48]:[00:50:79:66:68:01]
                    10.0.0.2                               0 65000 65002 i
                    RT:65002:100 ET:8
 *  [2]:[0]:[48]:[00:50:79:66:68:01]
                    10.0.0.2                               0 65000 65002 i
                    RT:65002:100 ET:8
Route Distinguisher: 10.0.0.3:2
 *> [2]:[0]:[48]:[00:50:79:66:68:04]
```

```
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:200 ET:8
 *  [2]:[0]:[48]:[00:50:79:66:68:04]
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:200 ET:8
Route Distinguisher: 10.0.0.3:3
 *> [2]:[0]:[48]:[00:50:79:66:68:02]
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:100 ET:8
 *  [2]:[0]:[48]:[00:50:79:66:68:02]
                    10.0.0.3                                0 65000 65003 i
                    RT:65003:100 ET:8

Displayed 5 prefixes (8 paths) (of requested type)
[edit]
vyos@leaf-1#
```

leaf-2 and leaf-3 advertise EVPN MAC/IP routes as soon as they learn their local host MAC addresses. The output indicates that leaf-1 is receiving two EPVN MAC/IP routes for the same MAC from each leaf that corresponds to both spines neighbors.

6. Verification of MAC/IP route attributes

```
vyos@leaf-1# run show bgp l2vpn evpn route detail
Route Distinguisher: 10.0.0.2:2
BGP routing table entry for 10.0.0.2:2:[2]:[0]:[48]:[00:50:79:66:68:01]
Paths: (2 available, best #1)
  Advertised to non peer-group peers:
  eth0 eth1
  Route [2]:[0]:[48]:[00:50:79:66:68:01] VNI 100
  65000 65002
    10.0.0.2 from eth0 (100.0.0.1)
      Origin IGP, valid, external, best (Router ID)
      Extended Community: RT:65002:100 ET:8
      Last update: Mon Apr  7 13:38:40 2025
  Route [2]:[0]:[48]:[00:50:79:66:68:01] VNI 100
  65000 65002
    10.0.0.2 from eth1 (100.0.0.2)
      Origin IGP, valid, external
      Extended Community: RT:65002:100 ET:8
      Last update: Mon Apr  7 13:38:40 2025
```

7. Verify MACs learned by each VNI

```
vyos@leaf-1# run show evpn mac vni 100
Number of MACs (local and remote) known for this VNI: 3
Flags: N=sync-neighs, I=local-inactive, P=peer-active, X=peer-proxy
MAC               Type   Flags Intf/Remote ES/VTEP           VLAN  Seq #'s
00:50:79:66:68:02 remote       10.0.0.3                             0/0
00:50:79:66:68:00 local        eth2                                 0/0
00:50:79:66:68:01 remote       10.0.0.2                             0/0
[edit]
vyos@leaf-1#
```

```
vyos@leaf-1# run show evpn mac vni 200
Number of MACs (local and remote) known for this VNI: 2
Flags: N=sync-neighs, I=local-inactive, P=peer-active, X=peer-proxy
MAC               Type   Flags Intf/Remote ES/VTEP           VLAN  Seq #'s
00:50:79:66:68:03 local        eth3                                0/0
00:50:79:66:68:04 remote       10.0.0.3                            0/0
[edit]
vyos@leaf-1#
```

8. Finally, verify hosts reachability

```
host-1> ping 192.168.0.2

84 bytes from 192.168.0.2 icmp_seq=1 ttl=64 time=2.783 ms
84 bytes from 192.168.0.2 icmp_seq=2 ttl=64 time=3.260 ms
84 bytes from 192.168.0.2 icmp_seq=3 ttl=64 time=3.194 ms
84 bytes from 192.168.0.2 icmp_seq=4 ttl=64 time=3.372 ms
84 bytes from 192.168.0.2 icmp_seq=5 ttl=64 time=3.126 ms

host-1>
host-1> ping 192.168.0.3

84 bytes from 192.168.0.3 icmp_seq=1 ttl=64 time=2.496 ms
84 bytes from 192.168.0.3 icmp_seq=2 ttl=64 time=3.473 ms
84 bytes from 192.168.0.3 icmp_seq=3 ttl=64 time=3.275 ms
84 bytes from 192.168.0.3 icmp_seq=4 ttl=64 time=3.479 ms
84 bytes from 192.168.0.3 icmp_seq=5 ttl=64 time=3.244 ms

host-1>
```

```
host-4> ping 192.168.1.3

84 bytes from 192.168.1.3 icmp_seq=1 ttl=64 time=2.911 ms
84 bytes from 192.168.1.3 icmp_seq=2 ttl=64 time=3.537 ms
84 bytes from 192.168.1.3 icmp_seq=3 ttl=64 time=3.525 ms
84 bytes from 192.168.1.3 icmp_seq=4 ttl=64 time=3.375 ms
84 bytes from 192.168.1.3 icmp_seq=5 ttl=64 time=3.121 ms

host-4>
```