JUNIPER NETWORKS | Engineering Simplicity

Network Configuration Example

Over-the-Top Data Center Interconnect in an EVPN Network

Published
2023-10-31

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

About This Guide | iv

1 **Example: How to configure the OTT DCI interconnect in an EVPN Network**

Overview of OTT DCI in an EVPN Network | 2

- About This Network Configuration Example | **2**
- Use Case Overview | **2**

How to Configure OTT DCI in an EVPN Network | 4

- Requirements | **4**
- Overview | **5**
- Configure the Border Spine Devices for WAN Connectivity | **7**
  - Requirements | **7**
  - Border Spine WAN Underlay configuration | **7**
  - Border Spine WAN Underlay | **8**
- Configure the Leaf Devices to support Layer 2 DCI | **10**
  - Requirements | **13**
  - | **13**
- Configure the Leaf Devices to support Layer 3 DCI | **13**
- Alternate Policy Configuration for Importing Route Targets | **16**
  - Requirements | **16**
  - | **16**
- Verification | **17**
  - Requirements | **18**
  - Overview | **18**
  - Verify BGP Peering | **18**

Detailed Configurations for the EVPN-VXLAN Network for the Data Centers | 29

About This Guide

CHAPTER 1

Example: How to configure the OTT DCI interconnect in an EVPN Network

Overview of OTT DCI in an EVPN Network | 2

How to Configure OTT DCI in an EVPN Network | 4

Detailed Configurations for the EVPN-VXLAN Network for the Data Centers | 29

Overview of OTT DCI in an EVPN Network

IN THIS SECTION

- About This Network Configuration Example | 2
- Use Case Overview | 2

About This Network Configuration Example

Use this Network Configuration Example (NCE) to deploy Layer 2 and or Layer 3 services between two or more data center fabric networks. We focus on the over-the-top (OTT) Data Center Interconnect (DCI) model. This NCE describes how to configure the interconnection. It does not provide the step-by-step procedure for configuring and deploying the EVPN VXLAN fabric in the data center. For a detailed configuration and more information on deploying an EVPN-VXLAN in a data center, see "Detailed Configurations for the EVPN-VXLAN Network for the Data Centers" on page 29 and Data Center EVPN-VXLAN Fabric Architecture Guide.

Use Case Overview

IN THIS SECTION

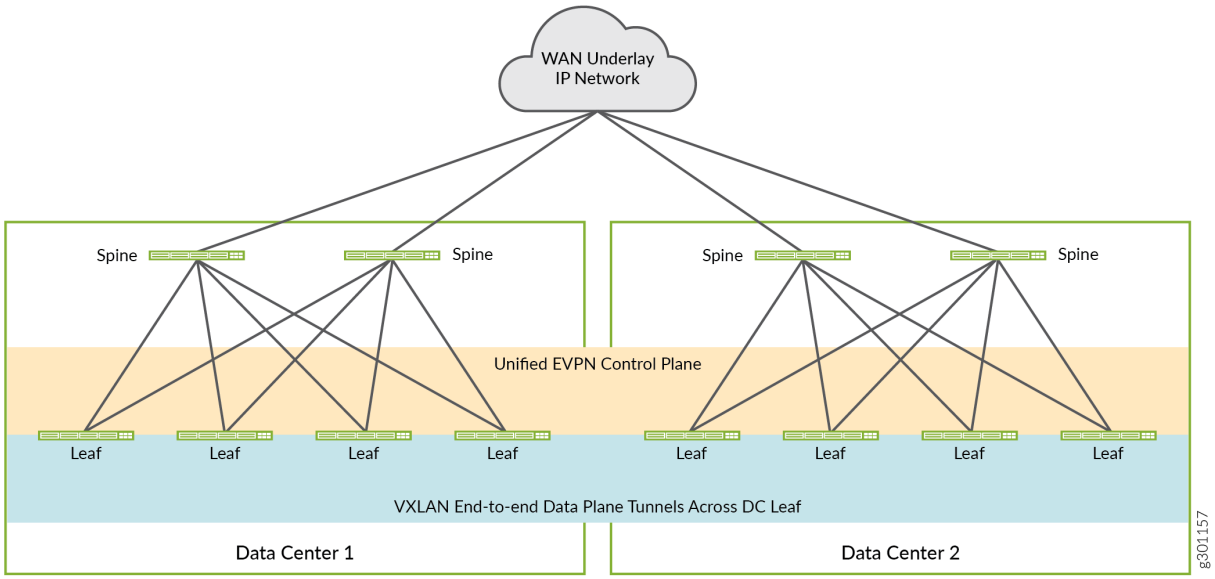
- Benefits of OTT DCI | 3
- Deployment Considerations | 3

DCI connects multiple data center fabrics. In most cases, data centers are geographically separated. In some cases, there are multiple fabrics within a data center. DCI can be used to provide overlay Layer 2 and Layer 3 services between these fabrics.

With OTT DCI, the control plane is extended across data center fabrics. The interconnected data center fabrics act as a single logical fabric. The unified control plane facilitates both Layer 2 and Layer 3 services. It is expected that IP connectivity has been established between the data center fabrics in the

underlay. The WAN underlay can be a Layer 3 VPN from a third-party, dark fiber, IPSec over Internet, etc. With OTT DCI, the leaf devices require a full mesh of VXLAN tunnels with all other leaf devices in all data centers wherever the given VLAN is configured. Figure 1 on page 3 illustrates a typical OTT DCI architecture.

Figure 1: OTT-DCI Architecture



Benefits of OTT DCI

NOTE: Our content testing team has validated and updated this example.

OTT DCI makes interconnecting data center simpler and can be deployed easily with QFX switches when you are expanding your existing data center infrastructure. Juniper recommends the following QFX series switches and roles for the deployment of OTT DCI:

- QFX 10000 series and QFX 5120-32C switches in the spine Role
- QFX5120-32C, QFX5120-48Y, and QFX5110 switches in the leaf role

Deployment Considerations

Consider the following in your DCI deployment:

- When Layer 2 services for VLANs are stretched across multiple data centers, you must have a full mesh of VXLAN tunnels among all the leaf devices in all the data centers where the VLAN is configured. As the number of leaf devices increases and the number of VLANs increases, you will start to see issues related to scaling.
- You should limit OTT DCI deployment to small and medium-sized data centers. For large data centers, we recommend using VXLAN-to-VXLAN stitching.

How to Configure OTT DCI in an EVPN Network

IN THIS SECTION

- Requirements | 4
- Overview | 5
- Configure the Border Spine Devices for WAN Connectivity | 7
- Configure the Leaf Devices to support Layer 2 DCI | 10
- Configure the Leaf Devices to support Layer 3 DCI | 13
- Alternate Policy Configuration for Importing Route Targets | 16
- Verification | 17

Requirements

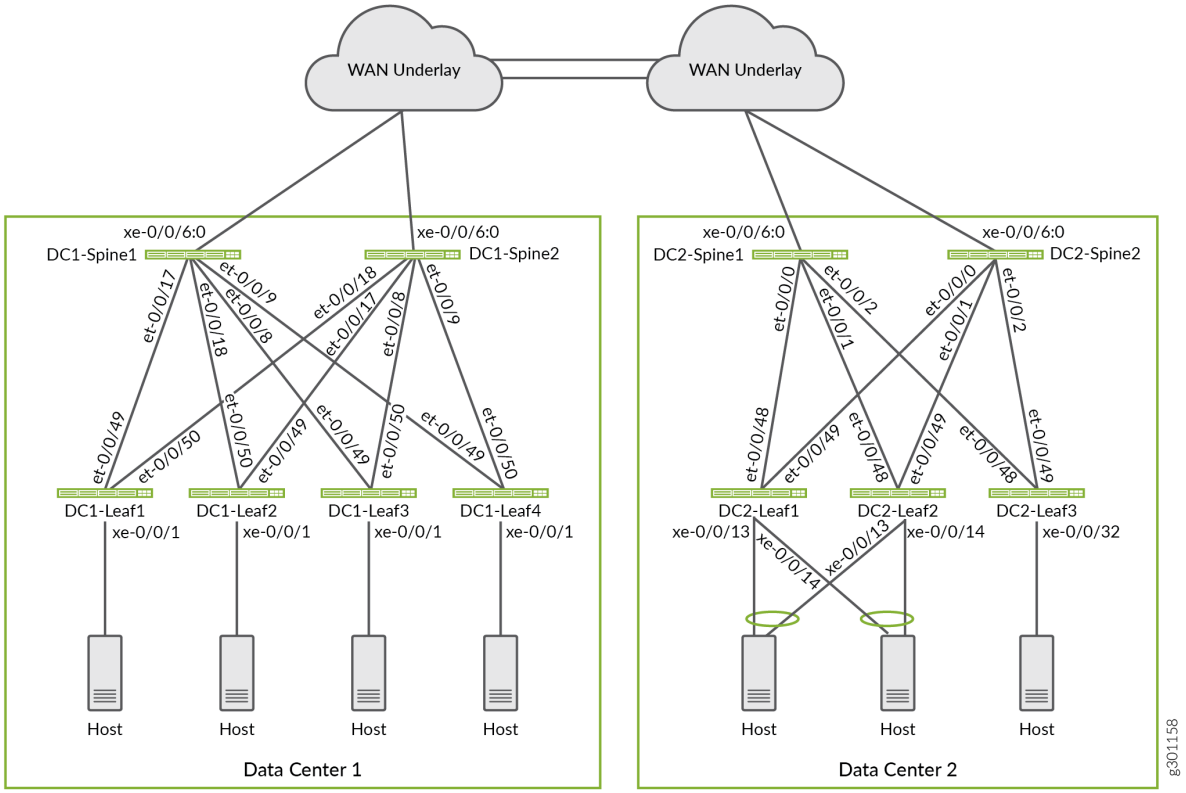
This configuration example uses the following devices and software versions:

- Spine devices consisting of QFX5120-32C switches running Junos OS Release 19.1R3-S2 in both DCs.
- Leaf devices consisting of QFX5110-48S switches running Junos OS Release 18.4R2-S5 in DC1.
- Leaf devices consisting of QFX5120-48Y switches running Junos OS Release 18.4R2-S5 in DC2.

Overview

In this deployment, we use an architecture with two data center fabric networks that have been configured in an edge-routed bridging (ERB) overlay model. The data centers are interconnected with a Layer 3 IP underlay network. Figure 2 on page 5 shows the WAN connectivity that provides the underlay transport between the two data centers. We use an over-the-top EVPN-VXLAN DCI architecture to provide both Layer 2 and Layer 3 connectivity for endpoints and virtual machines in the two data centers.

Figure 2: OTT DCI Architecture

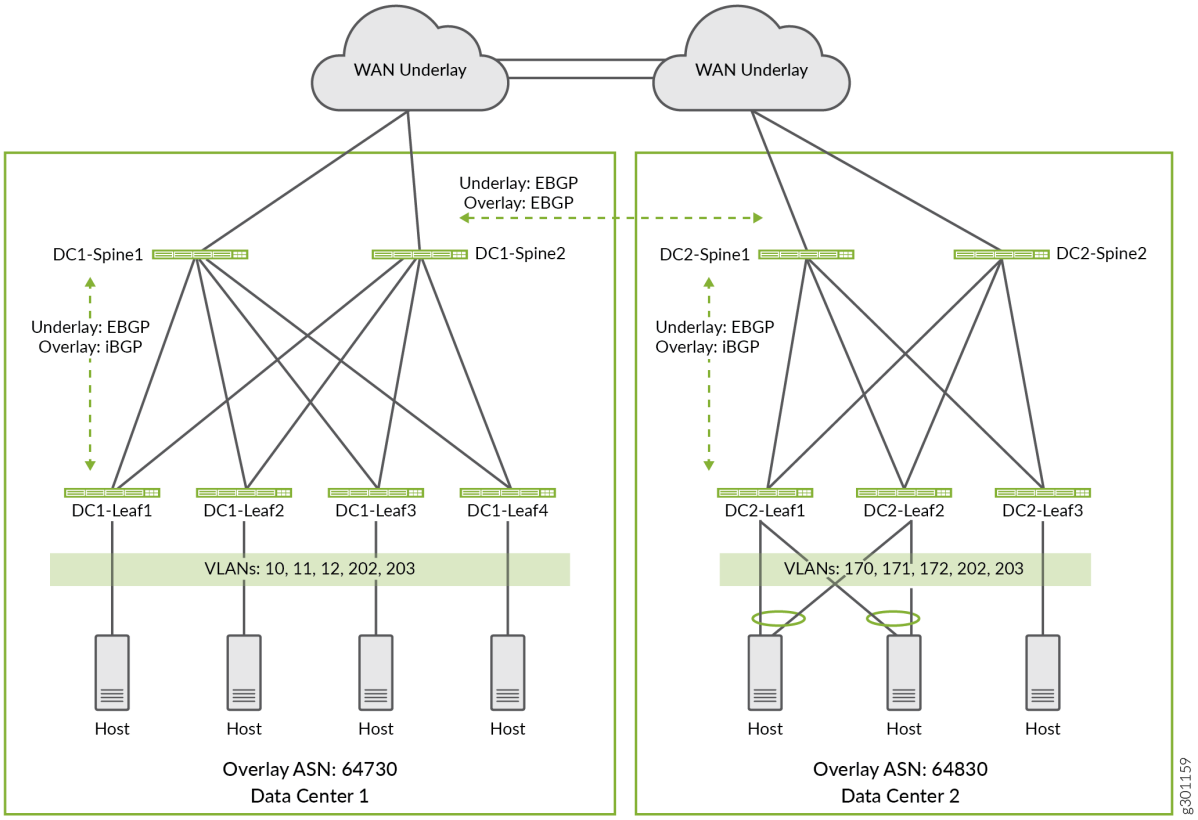


For this example, some VLANs are configured to exist only within a specific data center while other VLANs span across the two data centers. Figure 3 on page 6 shows the VLANs that are configured in the two data centers. We configure the VLANs as follows:

- VLANs 10, 11, and 12 are local to Data Center 1. The endpoints on these VLANs use Layer 3 routing to reach other VLANs in Data Center 2.
- VLANs 170, 171, and 172 are local to Data Center 2. The endpoints on these VLANs use Layer 3 routing to reach other VLANs in Data Center 1.

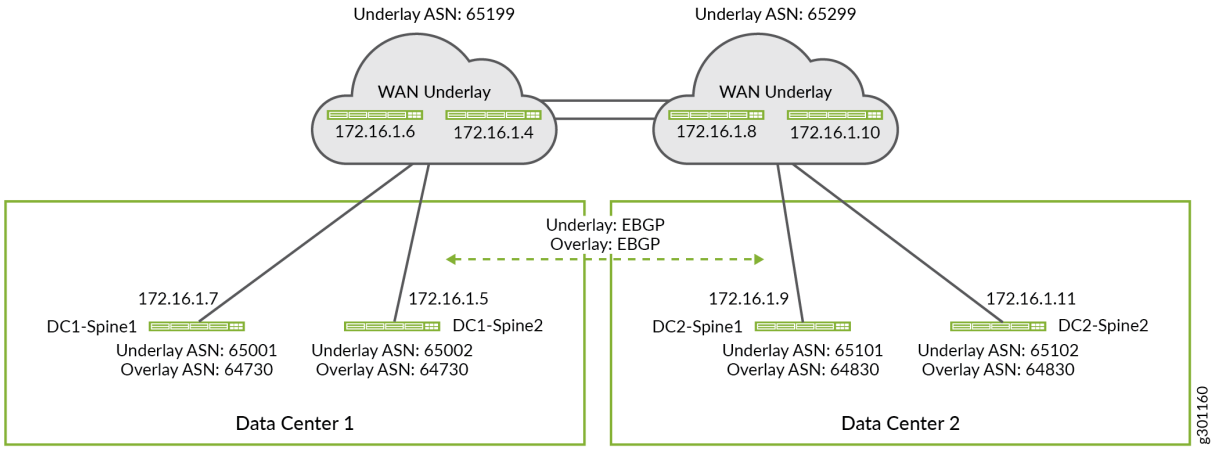
- VLANs 202 and 203 are common to Data Center 1 and Data Center 2. These VLANs stretch across all leaf devices in both data centers. The endpoints on these VLANs use Layer 2 bridging to reach the endpoints in the other data center.

Figure 3: VLANs in Data Centers



You can use any routing protocol in the underlay network. The underlay routing protocol enables devices to exchange loopback IP addresses with each other. In this example eBGP is used as the underlay routing protocol within each data center fabric. The eBGP protocol is also used to extend the underlay network across the WAN connection between the two data centers. You can use either iBGP or eBGP as the overlay routing protocol between the DCs. The choice depends on whether you use the same or different autonomous system numbers for your data centers. In this example, we will use iBGP for the overlay protocol within a fabric and eBGP between the two data centers. Figure 4 on page 7 shows the autonomous system numbers and IP addresses used in this example.

Figure 4: Details for WAN Router Peering



Configure the Border Spine Devices for WAN Connectivity

IN THIS SECTION

- Requirements | 7
- Border Spine WAN Underlay configuration | 7
- Border Spine WAN Underlay | 8

Requirements

Border Spine WAN Underlay configuration

NOTE: The configurations shown on this page are established on a pre-existing EVPN fabric with an operational underlay and overlay network. In other words, this section shows only the additional configurations that are needed to extend the DC underlay to a WAN cloud. In contrast, the detailed configurations show the full configuration of each DC device.

This section shows how to configure the WAN underlay configuration for border spine devices. For brevity, we will only show the steps for configuring one spine device in each data center. You can configure the other border spine devices by applying similar configuration steps.

NOTE: Repeat these procedures for all border spine devices in each data center.

Refer to the "Detailed Configurations for the EVPN-VXLAN Network for the Data Centers" on page 29 for the complete configurations used in this example.

Border Spine WAN Underlay

IN THIS SECTION

- Procedure | 8
- Border Spine WAN Overlay | 9

Procedure

Step-by-Step Procedure

1. Add the WAN peering session to the existing underlay group on Data Center 1 Spine 1 (DC1-Spine1). The border spine devices use the WAN underlay network to exchange loopback address information between the data centers. This in turn allows the spine devices to form an eBGP overlay connection across the wan cloud.

   ```
   set protocols bgp group UNDERLAY neighbor 172.16.1.6 peer-as 65199
   ```

   In this example the WAN router peering is added to the pre-existing `UNDERLAY` BGP peer group. Because this group is already configured as part of the DC's underlay, you only need to add the WAN router as a neighbor to the existing `UNDERLAY` group.

2. Add the WAN peering to the existing `UNDERLAY` group on Data Center 2 Spine 1 (DC2-Spine1).

   ```
   set protocols bgp group UNDERLAY neighbor 172.16.1.8 peer-as 65299
   ```

3. Configure the common WAN underlay policy used on the spine and leaf devices in both data centers. We reserve subnet 10.80.224.128/25 for loopback address in Data center 1, and subnet 10.0.0.0/24 for loopback address in Data Center 2. By specifying a reserved range of loopback addresses in our

policy, we do not have to update the policy when a device is added as long as the device is configured with a reserved loopback address.

```
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter
10.0.0.0/24 orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-EXPORT term DEFAULT then reject
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter
10.0.0.0/24 orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-IMPORT term DEFAULT then reject
```

In this example the underlay exchanges only loopback routes both within, and between, the DCs. The import and export policies for the underlay are written in such a way that they can be used by all devices in both DCs. For example, the `UNDERLAY-EXPORT` policy is configured to match both the local and remote DC loopback addresses. When applied to a device in DC1, only the `10.80.224.128/25 orlonger` term is matched. Having the extra `route-filter` term for the loopback addresses assigned to DC2 causes no harm, and it allows the same policy to be applied to the underlay in both DCs.

Border Spine WAN Overlay

Step-by-Step Procedure

For the fabric overlay, you configure a full mesh of eBGP peering among all border spine devices in both data centers using eBGP. The spine devices function as overlay route reflectors within their respective DC. The overlay peering session extends the EVPN overlay between the DCs.

Because the inter-DC overlay session runs over a WAN cloud bidirectional failure detection (BFD) is not enabled for this peer group. BFD is enabled to run within each DC, both in the underlay and overlay.

NOTE: By default eBGP changes the IP address in the Protocol next-hop field to use its own IP address when re-advertising routes. In this case, we want the protocol next-hop field to use the VTEP IP address of the leaf device that originated the route. To prevent BGP from changing the next hops of overlay routes we enable the `no-nexthop-change` option in the `OVERLAY_INTERDC` peer group.

1. Configure the WAN overlay network on DC1-Spine1.

```
set protocols bgp group OVERLAY_INTERDC type external
set protocols bgp group OVERLAY_INTERDC description "Group for overlay EBGP peering to remote
DC"
set protocols bgp group OVERLAY_INTERDC multihop no-nexthop-change
set protocols bgp group OVERLAY_INTERDC local-address 10.80.224.149
set protocols bgp group OVERLAY_INTERDC family evpn signaling delay-route-advertisements
minimum-delay routing-uptime 480
set protocols bgp group OVERLAY_INTERDC local-as 64730
set protocols bgp group OVERLAY_INTERDC multipath multiple-as
set protocols bgp group OVERLAY_INTERDC neighbor 10.0.0.2 peer-as 64830
set protocols bgp group OVERLAY_INTERDC neighbor 10.0.0.3 peer-as 64830
```

2. Configure the WAN overlay network on DC2-Spine1.

```
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group OVERLAY_INTERDC type external
set protocols bgp group OVERLAY_INTERDC description "Group for overlay EBGP peering to remote
DC"
set protocols bgp group OVERLAY_INTERDC multihop no-nexthop-change
set protocols bgp group OVERLAY_INTERDC local-address 10.0.0.2
set protocols bgp group OVERLAY_INTERDC family evpn signaling delay-route-advertisements
minimum-delay routing-uptime 480
set protocols bgp group OVERLAY_INTERDC local-as 64830
set protocols bgp group OVERLAY_INTERDC multipath multiple-as
set protocols bgp group OVERLAY_INTERDC neighbor 10.80.224.149 peer-as 64730
set protocols bgp group OVERLAY_INTERDC neighbor 10.80.224.150 peer-as 64730
```

Configure the Leaf Devices to support Layer 2 DCI

IN THIS SECTION

- Requirements | 13
- | 13

To extend Layer 2 (VLAN) connectivity across the two DCs we must configure the network to extend the Layer 2 broadcast domain for all endpoints in the shared VLAN. In this example, the endpoints on VLANs 202 and 203 belong to the same Layer 2 domain regardless of whether the endpoints are in the local or remote data centers. The goal is to extend, or stretch, these VLANs between the two DCs.

To extend Layer 2 connectivity for these shared VLANs between the DCs, you must configure the following:

- Configure unique VXLAN network identifier (VNI) based route targets for EVPN type 2 and EVPN type 3 routes that are associated with the stretched VLANs under the `protocols evpn` hierarchy.
- Apply an import policy to accept the unique route targets that are bound to the stretched VLANS using the `vrf-import` command at the `switch-options` hierarchy. The stretched VLANS are EVPN type 2 and type 3 routes that are advertised by the remote data center.

NOTE: Repeat these procedures for all leaf devices in the relevant data center. The commands shown below are added to the existing `switch-options` and `protocols evpn` hierarchies to stretch the shared VLANs between the two DCs.

Procedure

Step-by-Step Procedure

1. Configure VNI specific targets for the stretched VLANs at Data Center 1 Leaf 1 (DC1-Leaf1).

```
set protocols evpn vni-options vni 1202 vrf-target target:64730:202
set protocols evpn vni-options vni 1203 vrf-target target:64730:203
```

2. Configure a VRF import policy to match the VNI targets for the stretched VLANs at Data Center 1 Leaf 1 (DC1-Leaf1). The policy also matches the default EVPN type 1 routes used for Auto-Discovery (AD). The policy itself is defined in a later step.

```
set switch-options vrf-import OVERLAY_IMPORT
```

3. Configure VNI specific targets for the stretched VLANs at Data Center 2 Leaf 1 (DC2-Leaf1).

```
set protocols evpn vni-options vni 1202 vrf-target target:64830:202
set protocols evpn vni-options vni 1203 vrf-target target:64830:203
03
```

4. Specify a VRF import policy to match on the VNI targets for the stretched VLANs at Data Center 2 Leaf 1 (DC2-Leaf1).

```
set switch-options vrf-import OVERLAY_IMPORT
```

5. Define the VRF import policy that you configured at the `switch-options vrf-import` hierarchy in the previous step. In this example, we use unique route targets for each data center, and for each Layer 2 domain that is extended between the DCs. The policy is set to import the EVPN type 1 AD route that is associated with each DC/POD, as well as the EVPN type 2 and EVPN type 3 routes that are used by the stretched VLANs in both data centers.

   This policy should be configured on all leaf devices in both DCs.

   NOTE: In your deployment you can use a common route target for the leaf devices in your data centers. When you use a common route target, the routes from the data centers will be automatically imported as part of the implicit import policy.

```
set policy-options policy-statement OVERLAY_IMPORT term 5 from community comm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 5 then accept
set policy-options policy-statement OVERLAY_IMPORT term 10 from community comm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 10 then accept
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 then accept
set policy-options community comm_pod1 members target:64730:999
set policy-options community comm_pod2 members target:64830:999
set policy-options community shared_202_fm_pod1 members target:64730:202
set policy-options community shared_202_fm_pod2 members target:64830:202
set policy-options community shared_203_fm_pod1 members target:64730:203
set policy-options community shared_203_fm_pod2 members target:64830:203
```

The values used for the `comm_pod1` and `comm_po2` communities match the values specified with the `vrf-target statement` under the `switch-options` hierarchy of the leaf devices in DC1 and DC2, respectively. This is the route target that is added to all EVPN type 1 routes, which are used for auto discovery. This target is also attached to all other EVPN routes that do not have a VNI specific target specified at the `protocols evpn` hierarchy.

In this example the stretched VLANs are configured to use VNI specific targets. Therefore the import policy is written to match on all three of the targets used by each DC. Once again, this approach allows a common policy to be used across all leaf devices in both DCs.

Requirements

Configure the Leaf Devices to support Layer 3 DCI

IN THIS SECTION

- Procedure | 13

EVPN type 5 routes are used for Layer 3 connectivity between DCs for VLANs that are not stretched. An EVPN type 5 route is also called an IP prefix route. In other words, a Type 5 route is used when the Layer 2 broadcast domain is confined within the data center. EVPN type 5 routes enable Layer 3 connectivity by advertising the IP prefix of the IRB interface associated with non-extended VLANs. In this example, VLANs (10, 11, 12) are confined to Data Center 1, while VLANs (170, 171, 172) are confined within Data Center 2. We establish Layer 3 inter-data center connectivity between the members of these VLANs by using IP prefix routes.

Procedure

Step-by-Step Procedure

1. Configure Layer 3 DCI on DC1-Leaf1 by defining a Layer 3 VRF with support for EVPN type 5 routes.

```
set routing-instances TENANT_1_VRF description "VRF for Tenant_1"
set routing-instances TENANT_1_VRF instance-type vrf
```

```
set routing-instances TENANT_1_VRF interface irb.10
set routing-instances TENANT_1_VRF interface irb.11
set routing-instances TENANT_1_VRF interface irb.12
set routing-instances TENANT_1_VRF interface irb.202
set routing-instances TENANT_1_VRF interface irb.203
set routing-instances TENANT_1_VRF interface lo0.1
set routing-instances TENANT_1_VRF route-distinguisher 10.80.225.140:9999
set routing-instances TENANT_1_VRF vrf-import VRF1_VRF1_T5_RT_IMPORT
set routing-instances TENANT_1_VRF vrf-export VRF1_VRF1_T5_RT_EXPORT
set routing-instances TENANT_1_VRF vrf-target target:1:65001
set routing-instances TENANT_1_VRF vrf-table-label
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes vni 9999
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes export T5_EXPORT
```

2. Configure Layer 3 DCI on DC2-Leaf1 by defining a Layer 3 VRF with support for EVPN type 5 routes.

```
set routing-instances TENANT_1_VRF description "VRF for Tenant_1"
set routing-instances TENANT_1_VRF instance-type vrf
set routing-instances TENANT_1_VRF interface irb.170
set routing-instances TENANT_1_VRF interface irb.171
set routing-instances TENANT_1_VRF interface irb.172
set routing-instances TENANT_1_VRF interface irb.202
set routing-instances TENANT_1_VRF interface irb.203
set routing-instances TENANT_1_VRF interface lo0.1
set routing-instances TENANT_1_VRF route-distinguisher 10.0.1.19:9999
set routing-instances TENANT_1_VRF vrf-import VRF1_T5_RT_IMPORT
set routing-instances TENANT_1_VRF vrf-export VRF1_T5_RT_EXPORT
set routing-instances TENANT_1_VRF vrf-target target:1:65001
set routing-instances TENANT_1_VRF vrf-table-label
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes vni 9999
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes export T5_EXPORT
```

3. Configure the Layer 3 DCI policy for all leaf devices in DC1

```
set policy-options policy-statement T5_EXPORT term fm_direct from protocol direct
set policy-options policy-statement T5_EXPORT term fm_direct then accept
set policy-options policy-statement T5_EXPORT term fm_static from protocol static
set policy-options policy-statement T5_EXPORT term fm_static then accept
set policy-options policy-statement T5_EXPORT term fm_v4_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v4_host from route-filter 0.0.0.0/0
prefix-length-range /32-/32
set policy-options policy-statement T5_EXPORT term fm_v4_host then accept
set policy-options policy-statement T5_EXPORT term fm_v6_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v6_host from route-filter 0::0/0 prefix-
length-range /128-/128
set policy-options policy-statement T5_EXPORT term fm_v6_host then accept

set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then community add
target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then accept

set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 from community target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 from community target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 then accept

set policy-options community target_t5_pod1 members target:64730:9999
set policy-options community target_t5_pod2 members target:64830:9999
```

4. Configure the Layer 3 DCI policy for all leaf devices in DC2

```
set policy-options policy-statement T5_EXPORT term fm_direct from protocol direct
set policy-options policy-statement T5_EXPORT term fm_direct then accept
set policy-options policy-statement T5_EXPORT term fm_static from protocol static
set policy-options policy-statement T5_EXPORT term fm_static then accept
set policy-options policy-statement T5_EXPORT term fm_v4_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v4_host from route-filter 0.0.0.0/0
prefix-length-range /32-/32
set policy-options policy-statement T5_EXPORT term fm_v4_host then accept
set policy-options policy-statement T5_EXPORT term fm_v6_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v6_host from route-filter 0::0/0 prefix-
length-range /128-/128
set policy-options policy-statement T5_EXPORT term fm_v6_host then accept
```

```
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then community add
target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then accept

set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 from community target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 from community target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 then accept

set policy-options community target_t5_pod1 members target:64730:9999
set policy-options community target_t5_pod2 members target:64830:9999
```

Alternate Policy Configuration for Importing Route Targets

IN THIS SECTION

Requirements | 16

| 16

Requirements

In this example, the auto-derived route targets include the overlay AS number as part of its identifier. As a result you end up with different route targets for the same VNI in Data Center 1 and Data Center 2.

For a simpler policy configuration, you can automatically derive route targets by including the `vrf-target` statement with the `auto` option at the `switch-options` hierarchy. When you enable `vrf-target auto`, Junos automatically creates the route targets used for EVPN types 2 and type 3 routes. For EVPN-VXLAN, the route target is automatically derived from the VXLAN network identifier (VNI). In this example, the auto-derived EVPN type 2 and type 3 route targets for VNI 1202 are target:64730:268436658 in Data Center 1 and target:64830:268436658 in Data Center 2. These route targets are then attached to associated EVPN type 2 and type 3 routes. The `vrf-target auto` statement creates an implicit import policy to match on and import the same route target values for these VNIs.

With the simpler policy the EVPN type 2 and 3 routes for VLAN 202 and 203 are not automatically imported. To import EVPN type 2 and 3 routes from a data center with a different autonomous system number, include the `vrf-target auto import-as` statement.

Here is the configuration example for Leaf 1 in Data Center 1.

```
set switch-options vrf-import OVERLAY_IMPORT
set switch-options vrf-target target:64730:999
set switch-options vrf-target auto import-as 64830 vni-list 1202
set switch-options vrf-target auto import-as 64830 vni-list 1203
set protocols evpn extended-vni-list 110
set protocols evpn extended-vni-list 111
set protocols evpn extended-vni-list 112
set protocols evpn extended-vni-list 1202
set protocols evpn extended-vni-list 1203
set policy-options policy-statement OVERLAY_IMPORT term 5 from community comm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 5 then accept
set policy-options policy-statement OVERLAY_IMPORT term 10 from community comm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 10 then accept
set policy-options community comm_pod1 members target:64730:999
set policy-options community comm_pod2 members target:64830:999
```

Verification

IN THIS SECTION

- Requirements | **18**
- Overview | **18**
- Verify BGP Peering | **18**

Procedure

Requirements

Overview

Verify BGP Peering

IN THIS SECTION

- | 18
- Verify VTEPs in a Leaf Device | 19
- Verify EVPN type 1 Routes | 21
- Verify EVPN Type 2 Routes | 23
- Verify the Layer 3 DCI | 25

Purpose

Verify the underlay, overlay, and inter-DC BGP Peering sessions.

Action

Verify that all underlay and overlay BGP peering sessions are established. This includes the eBGP overlay peering between DCs formed across the WAN cloud.

The below is taken from the DC2-Spine1 device in DC2. All BGP peering sessions should be established on all leaf and spines devices in both DCs.

```
user@dc2spine1> show bgp summary

Threading mode: BGP I/O
Groups: 3 Peers: 9 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0
                      10          5          0          0          0          0
bgp.evpn.0
                      99         99          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State|#Active/
Received/Accepted/Damped...
```

```
10.0.0.14              64830      11634      11582       0       1     8:49:40 Establ
  bgp.evpn.0: 36/36/36/0
10.0.0.18              64830      11646      11604       0       2     8:49:40 Establ
  bgp.evpn.0: 36/36/36/0
10.0.0.19              64830      11652      11603       0       3     8:49:38 Establ
  bgp.evpn.0: 36/36/36/0
10.80.224.149          64730      11621      11622       0       6     8:49:31 Establ
  bgp.evpn.0: 27/27/27/0
10.80.224.150          64730      11630      11600       0       6     8:49:31 Establ
  bgp.evpn.0: 27/27/27/0
172.16.0.1             65019      11640      11582       0       2     8:49:39 Establ
  inet.0: 1/1/1/0
172.16.0.3             65018      11634      11583       0       1     8:49:42 Establ
  inet.0: 1/1/1/0
172.16.1.5             65229      11678      11500       0       1     8:49:43 Establ
  inet.0: 3/8/3/0
172.16.1.8             65229      11688      11581       0       1     8:49:43 Establ
  inet.0: 3/8/3/0
```

Meaning

The output on DC2-Spine1 confirms all of it's BGP sessions are established. There is an underlay and overlay session per local leaf, plus the WAN peering session, and the 2 overlay peering sessions to the spine devices in the remote DC. This brings the total number of sessions to 9. The ability to establish the overlay peering to the remote DC confirms proper underlay route exchange over the WAN cloud.

Verify VTEPs in a Leaf Device

Purpose

Verify the Layer 2 connection between two leaf devices in different data centers.

Action

Verify that VTEP interfaces are up on the leaf devices in both the local and remote data centers.

The following is a snippet of the output of VTEP status from a leaf in data center 1.

```
user@dc1leaf1> show interfaces vtep

Physical interface: vtep, Enabled, Physical link is Up
```

```
  Interface index: 641, SNMP ifIndex: 508
  Type: Software-Pseudo, Link-level type: VxLAN-Tunnel-Endpoint, MTU: Unlimited, Speed: Unlimited
  Device flags   : Present Running
  Link type      : Full-Duplex
  Link flags     : None
  Last flapped   : Never
    Input packets : 0
    Output packets: 0
.
.
.
  Logical interface vtep.32769 (Index 555) (SNMP ifIndex 539)
    Flags: Up SNMP-Traps Encapsulation: ENET2
    VXLAN Endpoint Type: Remote, VXLAN Endpoint Address: 10.80.224.141, L2 Routing Instance:
default-switch, L3 Routing Instance: default
    Input packets : 731317
    Output packets: 469531709
    Protocol eth-switch, MTU: Unlimited
      Flags: Trunk-Mode
.
.
.
  Logical interface vtep.32770 (Index 572) (SNMP ifIndex 541)
    Flags: Up SNMP-Traps Encapsulation: ENET2
    VXLAN Endpoint Type: Remote, VXLAN Endpoint Address: 10.0.0.18, L2 Routing Instance: default-
switch, L3 Routing Instance: default
    Input packets : 136973831
    Output packets: 141901343
    Protocol eth-switch, MTU: Unlimited
      Flags: Trunk-Mode
.
.
.
```

Meaning

The output on DC1-Leaf1 shows that Layer 2 VXLAN tunnels are created between this leaf device and the other leaf devices in the local data center fabric (the VTEP address of leaf 2 in DC1 is shown in the snippet). The output also shows VTEPs are learned between this leaf device and the other leaf devices in the remote data center fabric (the VTEP for leaf 2 in DC2 is shown in the snippet). The input and output packet counters verify successful data transmission between endpoints on these VTEPs.

Verify EVPN type 1 Routes

Purpose

Verify EVPN type 1 routes are being added to the routing table.

Action

Verify that the EVPN type1 (AD/ESI) route has been received and installed..

```
user@dc1leaf1> show  route | match 1:10.0.0.18

1:10.0.0.18:0::0202020201::FFFF:FFFF/192 AD/ESI
1:10.0.0.18:0::0202020202::FFFF:FFFF/192 AD/ESI

user@dc1leaf1> show route extensive | find 1:10.0.0.18:0::0202020201:

1:10.0.0.18:0::0202020201::FFFF:FFFF/192 AD/ESI (1 entry, 0 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 10.0.0.18:0
                Next hop type: Indirect, Next hop index: 0
                Address: 0xbd78c94
                Next-hop reference count: 50
                Source: 10.80.224.149
                Protocol next hop: 10.0.0.18
                Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                State: <Active Int Ext>
                Local AS: 64730 Peer AS: 64730
                Age: 1:22       Metric2: 0
                Validation State: unverified
                Task: BGP_64730.10.80.224.149
                AS path: 64830 I
                Communities: target:64830:999 encapsulation:vxlan(0x8) esi-label:0x0:all-active
(label 0)
                Import Accepted
                Route Label: 1
                Localpref: 100
                Router ID: 10.80.224.149
                Secondary Tables: default-switch.evpn.0
                Indirect next hops: 1
                        Protocol next hop: 10.0.0.18
                        Indirect next hop: 0x2 no-forward INH Session ID: 0x0
```

```
                        Indirect path forwarding next hops: 1
                                Next hop type: Router
                                Next hop: 10.80.224.2 via et-0/0/49.0
                                Session Id: 0x0
                                10.0.0.18/32 Originating RIB: inet.0
                                  Node path count: 1
                                  Forwarding nexthops: 1
                                        Nexthop: 10.80.224.2 via et-0/0/49.0
                                        Session Id: 0
```

Display EVPN routing instance information. The snippet below is taken at leaf 1 in DC1.

```
user@dc1leaf1> show evpn instance extensive

Instance: __default_evpn__
  Route Distinguisher: 10.80.224.140:0
  Number of bridge domains: 0
  Number of neighbors: 0

. . .
    ESI: 00:00:00:00:00:02:02:02:02:01
      Status: Resolved
      Number of remote PEs connected: 2
        Remote PE        MAC label  Aliasing label  Mode
        10.0.0.19        1203       0               all-active
        10.0.0.18        0          0               all-active
    ESI: 00:00:00:00:00:02:02:02:02:02
      Status: Resolved
      Number of remote PEs connected: 2
        Remote PE        MAC label  Aliasing label  Mode
        10.0.0.19        0          0               all-active
        10.0.0.18        0          0               all-active

. . .
```

Meaning

These outputs show that DC1-Leaf1 has received EVPN type1 (AD/ESI) routes from the remote data center. You can use the output of the `show route extensive` command to show that these routes have a route target of “target:64830:999" and that these routes were successfully installed. The ESI

02:02:02:02:01 and 02:02:02:02:02 are Ethernet segments in the remote data center. This output also shows the remote PEs with IP addresses of 10.0.0.18 and 10.0.0.19 are part of these Ethernet segments.

Verify EVPN Type 2 Routes

Purpose

Verify EVPN type 2 routes are being added to the routing table.

Action

Verify that an EVPN type 2 route has been received and installed in the route table on a leaf device in data center 1.

Use show route to find an endpoint in VLAN 203 that is in the remote data center.

```
user@dc1leaf1>show route | match 00:10:94:00:00:06

2:10.0.0.18:1::1203::00:10:94:00:00:06/304 MAC/IP
2:10.0.0.18:1::1203::00:10:94:00:00:06::10.1.203.151/304 MAC/IP
```

Use the detailed view for show route to see more details.

```
user@dc1leaf1> show route extensive | find 2:10.0.0.18:1::1203::00:10:94:00:00:06::10.1.203.151/304

2:10.0.0.18:1::1203::00:10:94:00:00:06::10.1.203.151/304 MAC/IP (1 entry, 0 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 10.0.0.18:1
                Next hop type: Indirect, Next hop index: 0
                Address: 0xbd78c94
                Next-hop reference count: 58
                Source: 10.80.224.149
                Protocol next hop: 10.0.0.18
                Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                State: <Active Int Ext>
                Local AS: 64730 Peer AS: 64730
                Age: 1:31       Metric2: 0
                Validation State: unverified
                Task: BGP_64730.10.80.224.149
                AS path: 64830 I
```

```
                Communities: target:64830:203 encapsulation:vxlan(0x8)
                Import Accepted
                Route Label: 1203
                ESI: 00:00:00:00:00:00:00:00:01:01
                Localpref: 100
                Router ID: 10.80.224.149
                Secondary Tables: default-switch.evpn.0
                Indirect next hops: 1
                        Protocol next hop: 10.0.0.18
                        Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                        Indirect path forwarding next hops: 1
                                Next hop type: Router
                                Next hop: 10.80.224.2 via et-0/0/49.0
                                Session Id: 0x0
                                10.0.0.18/32 Originating RIB: inet.0
                                  Node path count: 1
                                  Forwarding nexthops: 1
                                        Nexthop: 10.80.224.2 via et-0/0/49.0
                                        Session Id: 0
```

Use the `show vlans` command to find the information on VLAN 203.

```
user@dc1leaf1> show vlans v203

Routing instance        VLAN name             Tag          Interfaces
default-switch          v203                  203
                                                           esi.1860*
                                                           vtep.32769*
                                                           vtep.32770*
                                                           vtep.32772*
                                                                                   xe-0/0/1.0*
```

The above output shows that VLAN 203 is configured on VTEPs “vtep.32769 - 10.80.224.141 – DC1-Leaf2”, “vtep.32772 – 10.0.0.18 – DC2-Leaf2”, “vtep.32770 - 10.0.0.19 – DC2-Leaf1”. The output also indicates that there are endpoints connected to VLAN v203 on these VTEPs. You can use `show interfaces VTEP` to get the VTEP IP address for these VTEPs. Use the `show evpn database` and the `show ethernet-switching-table` commands to find the MAC address and IP address for these endpoints.

Use the `show evpn database` to find entries in the EVPN database.

```
user@dc1leaf1> show evpn database neighbor 10.0.0.18

Instance: default-switch
VLAN  DomainId  MAC address        Active source                  Timestamp        IP address
     1202       3c:8c:93:2e:a8:c0  10.0.0.18                      Jul 18 23:41:54  10.1.202.18
     1203       00:10:94:00:00:06  00:00:00:00:00:02:02:02:02:01  Jul 19 03:16:42  10.1.203.151
     1203       3c:8c:93:2e:a8:c0  10.0.0.18                      Jul 18 23:41:54  10.1.203.18
```

Use the `show ethernet-switching table` command to display information for a particular MAC address.

```
user@dc1leaf1> show ethernet-switching table 00:10:94:00:00:06


MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static
           SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)


Ethernet switching table : 29 entries, 29 learned
Routing instance : default-switch
   Vlan                MAC                 MAC      Logical                Active
   name                address             flags    interface              source
   v203                00:10:94:00:00:06   DR       esi.1874
00:00:00:00:00:02:02:02:02:01
```

Meaning

The output shows that the MAC address from a device on VLAN 203 in the remote PE has been installed in the EVPN database and is in Ethernet Switching table.

Verify the Layer 3 DCI

Purpose

Verify Layer 3 routes are installed in the routing table.

Action

Verify that an EVPN type 5 route has been received and installed in the route table on a leaf device in data center 1.

Verify that you are receiving EVPN type 5 routes from the 10.1.170.0/24, which is the subnet for VLAN 170. This VLAN is local to data center 2. To reach this subnet from Data Center 1 requires Layer 3 routing.

```
user@dc1leaf1> show route | match 5: | match 10.1.170.0

5:10.0.1.18:9999::0::10.1.170.0::24/248
5:10.0.1.19:9999::0::10.1.170.0::24/248

. . .
```

You can view more details about the EVPN type 5 route for the 10.1.170.0/24 network. Traffic sent from this leaf device for this subnet is encapsulated in VXLAN tunnel with VNI 9999 and sent to remote leaf 10.0.0.18 in data center 2.

```
user@dc1leaf1> show route 10.1.170.0/24 extensive

TENANT_1_VRF.inet.0: 28 destinations, 57 routes (28 active, 0 holddown, 0 hidden)
10.1.170.0/24 (3 entries, 1 announced)
TSI:
KRT in-kernel 10.1.170.0/24 -> {composite(1742)}
        *EVPN   Preference: 170/-101
                Next hop type: Indirect, Next hop index: 0
                Address: 0xdfacd74
                Next-hop reference count: 9
                Next hop type: Router, Next hop index: 1738
                Next hop: 10.80.224.2 via et-0/0/49.0, selected
                Session Id: 0x0
                Protocol next hop: 10.0.0.18
                Composite next hops: 1
                        Protocol next hop: 10.0.0.18
                        Composite next hop: 0xb959d60 1740 INH Session ID: 0x0
                          VXLAN tunnel rewrite:
                            MTU: 0, Flags: 0x0
                            Encap table ID: 0, Decap table ID: 11
                            Encap VNI: 9999, Decap VNI: 9999
                            Source VTEP: 10.80.224.140, Destination VTEP: 10.0.0.18
```

```
                        SMAC: e4:5d:37:ea:98:00, DMAC: 3c:8c:93:2e:a8:c0
                    Indirect next hop: 0xc710304 131071 INH Session ID: 0x0
                    Indirect path forwarding next hops: 1
                            Next hop type: Router
                            Next hop: 10.80.224.2 via et-0/0/49.0
                            Session Id: 0x0
                            10.0.0.18/32 Originating RIB: inet.0
                              Node path count: 1
                              Forwarding nexthops: 1
                                    Nexthop: 10.80.224.2 via et-0/0/49.0
                                    Session Id: 0
```

The output confirms that routes for the 10.1.170.0/24 subnet are present in this leaf device. Since we are exporting and importing host routes, you see the specific EVPN type 5 host routes.

```
user@dc1leaf1> show route table TENANT_1_VRF.inet.0 10.1.170.0/24

TENANT_1_VRF.inet.0: 21 destinations, 35 routes (21 active, 0 holddown, 0 hidden)
@ = Routing Use Only, # = Forwarding Use Only
+ = Active Route, - = Last Active, * = Both

10.1.170.0/24      @[EVPN/170] 1d 06:38:11
                    >  to 10.80.224.2 via et-0/0/49.0
                    >  to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 1d 06:38:09
                    >  to 10.80.224.2 via et-0/0/49.0
                    >  to 10.80.224.12 via et-0/0/50.0
                   #[Multipath/255] 1d 06:38:09, metric2 0
                    >  to 10.80.224.2 via et-0/0/49.0
                    >  to 10.80.224.12 via et-0/0/50.0
10.1.170.100/32    @[EVPN/170] 00:00:23
                    >  to 10.80.224.2 via et-0/0/49.0
                    >  to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 00:00:22
                    >  to 10.80.224.2 via et-0/0/49.0
                    >  to 10.80.224.12 via et-0/0/50.0
                   #[Multipath/255] 00:00:22, metric2 0
                    >  to 10.80.224.2 via et-0/0/49.0
                    >  to 10.80.224.12 via et-0/0/50.0
```

The output shows routes for 10.1.203.0/24 subnet in this leaf device. In this case, VLAN 203 stretches across both data centers. When you only limit the EVPN type 5 routes to subnet routes, you will have

asymmetric the inter-vni routing for L2 stretched VLANs. If you prefer to deploy a symmetric model for inter-vni routing for L2 stretched VLANs, you must export and import EVPN type 5 host routes.

This example uses the `T5_EXPORT` policy applied to the `TENANT_1_VRF` as an export policy for the EVPN protocol to effect advertisement of /32 host routes. As a result this example demonstrates symmetric routing for inter-VLAN routing when those VLANs are stretched between DCs.

```
user@dc1leaf1>show route table TENANT_1_VRF.inet.0 10.1.203.0/24

TENANT_1_VRF.inet.0: 28 destinations, 57 routes (28 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.203.0/24      *[Direct/0] 1d 05:42:50
                    >  via irb.203
                    [EVPN/170] 1d 05:42:50
                    >  to 10.80.224.2 via et-0/0/49.0
                       to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 1d 05:42:50
                    >  to 10.80.224.2 via et-0/0/49.0
                       to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 1d 05:42:50
                    >  to 10.80.224.2 via et-0/0/49.0
                       to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 1d 05:42:50
                    >  to 10.80.224.2 via et-0/0/49.0
                       to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 01:06:19
                    >  to 10.80.224.2 via et-0/0/49.0
                       to 10.80.224.12 via et-0/0/50.0
                    [EVPN/170] 01:17:41
                    >  to 10.80.224.2 via et-0/0/49.0
                       to 10.80.224.12 via et-0/0/50.0
10.1.203.1/32      *[Local/0] 1d 05:42:50
                       Local via irb.203
10.1.203.11/32     *[Local/0] 1d 05:42:50
                       Local via irb.203
10.1.203.52/32     *[EVPN/7] 02:52:51
                    >  via irb.203
```

Meaning

This output shows that the both the 10.1.203.0/24 subnet and the specific host route for 10.1.203.52/32 (the IP address of an endpoint in Data Center 2) have been installed. The more EVPN type 5 route is preferred over the EVPN type 2 route for 10.1.203.52/32 route. This results in symmetric Inter-VNI routing over Layer 2 stretched VLANs.

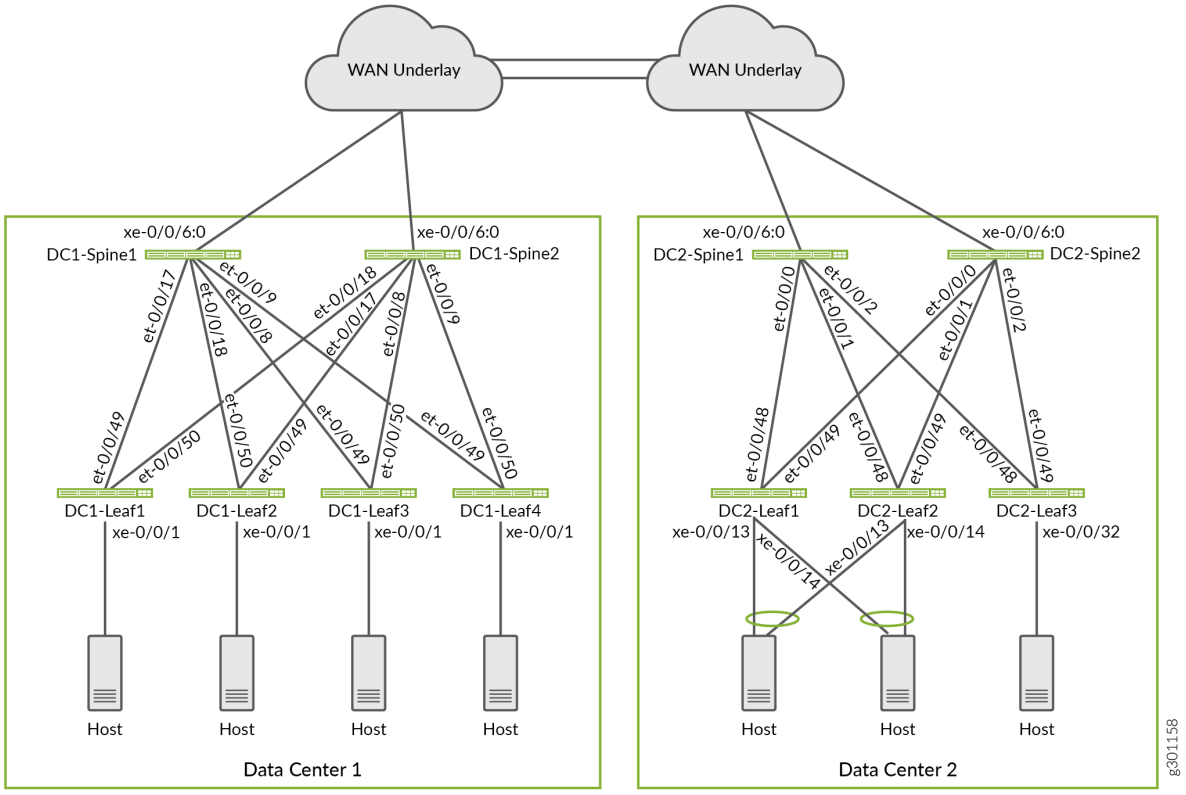
Detailed Configurations for the EVPN-VXLAN Network for the Data Centers

This section includes the complete configurations for the following devices:

- Data Center 1 Border Spine 1 and Border Spine 2 (DC1-Spine1 and DC1-Spine2)
- Data Center 1 Leaf 1 and Leaf 2 (DC1-Leaf1 and DC1-Leaf2)
- Data Center 2 Border Spine 1 and Border Spine 2 (DC2-Spine1 and DC1-Spine2)
- Data Center 2 Leaf 1 and Leaf 2 (DC2-Leaf1 and DC2-Leaf2)
- A simplified WAN router configuration that can be used for testing

Figure 5 on page 30 shows the detailed topology used in this example.

Figure 5: OTT DCI Topology



NOTE: The configurations are shown as typed, rather than as displayed with a `show configuration | display set` command. As a result the BGP authentication key appears in plain text instead of the obfuscated form. The authentication key used is for testing purposes and should be changed to suit your environment.

The need to channelize for 10 GE interfaces varies by QFX switch model. Based on the equipment used in this example channelization is not needed at the leaf switches, but is used on the spines for the 10 GE links to the WAN cloud.

Border Spine Device 1 in Data Center 1

```
set chassis fpc 0 pic 0 port 6 channel-speed 10
set interfaces xe-0/0/6:0 mtu 9200
set interfaces xe-0/0/6:0 unit 0 family inet address 172.16.1.7/31
set interfaces et-0/0/8 description "Fabric link"
set interfaces et-0/0/8 mtu 9200
set interfaces et-0/0/8 unit 0 description "** to DC1-Leaf3"
```

```
set interfaces et-0/0/8 unit 0 family inet address 10.80.224.6/31
set interfaces et-0/0/9 description "Fabric link"
set interfaces et-0/0/9 mtu 9200
set interfaces et-0/0/9 unit 0 description "** to DC1-Leaf4"
set interfaces et-0/0/9 unit 0 family inet address 10.80.224.8/31
set interfaces et-0/0/17 description "Fabric link"
set interfaces et-0/0/17 mtu 9200
set interfaces et-0/0/17 unit 0 description "** to DC1-Leaf1"
set interfaces et-0/0/17 unit 0 family inet address 10.80.224.2/31
set interfaces et-0/0/18 description "Fabric link"
set interfaces et-0/0/18 traps
set interfaces et-0/0/18 mtu 9200
set interfaces et-0/0/18 unit 0 description "** to DC1-Leaf2"
set interfaces et-0/0/18 unit 0 family inet address 10.80.224.4/31
set interfaces lo0 unit 0 description "** DC1-Spine1"
set interfaces lo0 unit 0 family inet address 10.80.224.149/32
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-EXPORT term DEFAULT then reject
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-IMPORT term DEFAULT then reject
set routing-options forwarding-table export ECMP-POLICY
set routing-options forwarding-table ecmp-fast-reroute
set routing-options autonomous-system 64730
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC description "Reflector group for overlay IBGP peering with
leaves"
set protocols bgp group EVPN_FABRIC local-address 10.80.224.149
```

```
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC vpn-apply-export
set protocols bgp group EVPN_FABRIC cluster 10.80.224.149
set protocols bgp group EVPN_FABRIC local-as 64730
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.140
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.141
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.142
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.143
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for underlay EBGP peering"
set protocols bgp group UNDERLAY import UNDERLAY-IMPORT
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export UNDERLAY-EXPORT
set protocols bgp group UNDERLAY local-as 65001
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 10.80.224.3 peer-as 65012
set protocols bgp group UNDERLAY neighbor 10.80.224.5 peer-as 65013
set protocols bgp group UNDERLAY neighbor 10.80.224.7 peer-as 65014
set protocols bgp group UNDERLAY neighbor 10.80.224.9 peer-as 65015
set protocols bgp group UNDERLAY neighbor 172.16.1.6 peer-as 65199
set protocols bgp group OVERLAY_INTERDC type external
set protocols bgp group OVERLAY_INTERDC description "Group for overlay EBGP peering to remote DC"
set protocols bgp group OVERLAY_INTERDC multihop no-nexthop-change
set protocols bgp group OVERLAY_INTERDC local-address 10.80.224.149
set protocols bgp group OVERLAY_INTERDC family evpn signaling delay-route-advertisements minimum-
delay routing-uptime 480
set protocols bgp group OVERLAY_INTERDC local-as 64730
set protocols bgp group OVERLAY_INTERDC multipath multiple-as
set protocols bgp group OVERLAY_INTERDC neighbor 10.0.0.2 peer-as 64830
set protocols bgp group OVERLAY_INTERDC neighbor 10.0.0.3 peer-as 64830
set protocols lldp interface all
```

Border Spine Device 2 in Data Center 1

```
set chassis fpc 0 pic 0 port 33 channel-speed 10g
set interfaces xe-0/0/6:0 mtu 9200
set interfaces xe-0/0/6:0 unit 0 family inet address 172.16.1.5/31
```

```
set interfaces et-0/0/8 description "Fabric link"
set interfaces et-0/0/8 mtu 9200
set interfaces et-0/0/8 unit 0 description "** to DC1-Leaf3"
set interfaces et-0/0/8 unit 0 family inet address 10.80.224.16/31
set interfaces et-0/0/9 description "Fabric link"
set interfaces et-0/0/9 mtu 9200
set interfaces et-0/0/9 unit 0 description "** to DC1-Leaf4"
set interfaces et-0/0/9 unit 0 family inet address 10.80.224.18/31
set interfaces et-0/0/17 description "Fabric link"
set interfaces et-0/0/17 mtu 9200
set interfaces et-0/0/17 unit 0 description "** to DC1-Leaf2"
set interfaces et-0/0/17 unit 0 family inet address 10.80.224.14/31
set interfaces et-0/0/18 description "Fabric link"
set interfaces et-0/0/18 mtu 9200
set interfaces et-0/0/18 unit 0 description "** to DC1-Leaf1"
set interfaces et-0/0/18 unit 0 family inet address 10.80.224.12/31
set interfaces lo0 unit 0 description "** DC1-Spine2"
set interfaces lo0 unit 0 family inet address 10.80.224.150/32
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-EXPORT term DEFAULT then reject
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-IMPORT term DEFAULT then reject
set routing-options forwarding-table export ECMP-POLICY
set routing-options forwarding-table ecmp-fast-reroute
set routing-options autonomous-system 64730
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC description "Reflector group for overlay IBGP peering with
leaves"
set protocols bgp group EVPN_FABRIC local-address 10.80.224.150
```

```
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC vpn-apply-export
set protocols bgp group EVPN_FABRIC cluster 10.80.224.150
set protocols bgp group EVPN_FABRIC local-as 64730
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.140
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.141
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.142
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.143
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for underlay EBGP peering"
set protocols bgp group UNDERLAY import UNDERLAY-IMPORT
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export UNDERLAY-EXPORT
set protocols bgp group UNDERLAY local-as 65002
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 10.80.224.13 peer-as 65012
set protocols bgp group UNDERLAY neighbor 10.80.224.19 peer-as 65015
set protocols bgp group UNDERLAY neighbor 10.80.224.17 peer-as 65014
set protocols bgp group UNDERLAY neighbor 10.80.224.15 peer-as 65013
set protocols bgp group UNDERLAY neighbor 172.16.1.4 peer-as 65199
set protocols bgp group OVERLAY_INTERDC type external
set protocols bgp group OVERLAY_INTERDC description "Group for overlay EBGP peering to remote DC"
set protocols bgp group OVERLAY_INTERDC multihop no-nexthop-change
set protocols bgp group OVERLAY_INTERDC local-address 10.80.224.150
set protocols bgp group OVERLAY_INTERDC family evpn signaling delay-route-advertisements minimum-
delay routing-uptime 480
set protocols bgp group OVERLAY_INTERDC local-as 64730
set protocols bgp group OVERLAY_INTERDC multipath multiple-as
set protocols bgp group OVERLAY_INTERDC neighbor 10.0.0.2 peer-as 64830
set protocols bgp group OVERLAY_INTERDC neighbor 10.0.0.3 peer-as 64830
set protocols lldp interface all
```

Leaf Device 1 in Data Center 1

```
set system arp aging-timer 5
set interfaces xe-0/0/1 description "DC1 Server1"
set interfaces xe-0/0/1 unit 0 family ethernet-switching vlan members v203
```

```
set interfaces et-0/0/49 description "Fabric interface"
set interfaces et-0/0/49 mtu 9200
set interfaces et-0/0/49 unit 0 description "** to DC1-Spine1"
set interfaces et-0/0/49 unit 0 family inet address 10.80.224.3/31
set interfaces et-0/0/50 description "Fabric interface"
set interfaces et-0/0/50 mtu 9200
set interfaces et-0/0/50 unit 0 description "** to DC1-Spine2"
set interfaces et-0/0/50 unit 0 family inet address 10.80.224.13/31
set interfaces irb unit 10 virtual-gateway-accept-data
set interfaces irb unit 10 description "** L3 interface for v10 in Tenant_1"
set interfaces irb unit 10 family inet address 10.1.10.11/24 preferred
set interfaces irb unit 10 family inet address 10.1.10.11/24 virtual-gateway-address 10.1.10.1
set interfaces irb unit 11 virtual-gateway-accept-data
set interfaces irb unit 11 description "** L3 interface for v11 in Tenant_1"
set interfaces irb unit 11 family inet address 10.1.11.11/24 preferred
set interfaces irb unit 11 family inet address 10.1.11.11/24 virtual-gateway-address 10.1.11.1
set interfaces irb unit 12 virtual-gateway-accept-data
set interfaces irb unit 12 description "** L3 interface for v12 in Tenant_1"
set interfaces irb unit 12 family inet address 10.1.12.11/24 preferred
set interfaces irb unit 12 family inet address 10.1.12.11/24 virtual-gateway-address 10.1.12.1
set interfaces irb unit 202 virtual-gateway-accept-data
set interfaces irb unit 202 description "** L3 interface for v202 in Tenant_1"
set interfaces irb unit 202 family inet address 10.1.202.11/24 preferred
set interfaces irb unit 202 family inet address 10.1.202.11/24 virtual-gateway-address 10.1.202.1
set interfaces irb unit 203 virtual-gateway-accept-data
set interfaces irb unit 203 description "** L3 interface for v203 in Tenant_1"
set interfaces irb unit 203 family inet address 10.1.203.11/24 preferred
set interfaces irb unit 203 family inet address 10.1.203.11/24 virtual-gateway-address 10.1.203.1
set interfaces lo0 unit 0 description "** DC1-Leaf1"
set interfaces lo0 unit 0 family inet address 10.80.224.140/32
set interfaces lo0 unit 1 family inet address 10.80.225.140/32
set forwarding-options vxlan-routing overlay-ecmp
set forwarding-options vxlan-routing next-hop 32768
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement OVERLAY_IMPORT term 5 from community comm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 5 then accept
set policy-options policy-statement OVERLAY_IMPORT term 10 from community comm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 10 then accept
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod1
```

```
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 then accept
set policy-options policy-statement T5_EXPORT term fm_direct from protocol direct
set policy-options policy-statement T5_EXPORT term fm_direct then accept
set policy-options policy-statement T5_EXPORT term fm_static from protocol static
set policy-options policy-statement T5_EXPORT term fm_static then accept
set policy-options policy-statement T5_EXPORT term fm_v4_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v4_host from route-filter 0.0.0.0/0 prefix-
length-range /32-/32
set policy-options policy-statement T5_EXPORT term fm_v4_host then accept
set policy-options policy-statement T5_EXPORT term fm_v6_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v6_host from route-filter 0::0/0 prefix-
length-range /128-/128
set policy-options policy-statement T5_EXPORT term fm_v6_host then accept
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then community add target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 from community target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 from community target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 then accept
set policy-options community comm_pod1 members target:64730:999
set policy-options community comm_pod2 members target:64830:999
set policy-options community shared_202_fm_pod1 members target:64730:202
set policy-options community shared_202_fm_pod2 members target:64830:202
set policy-options community shared_203_fm_pod1 members target:64730:203
set policy-options community shared_203_fm_pod2 members target:64830:203
set policy-options community target_t5_pod1 members target:64730:9999
set policy-options community target_t5_pod2 members target:64830:9999
set routing-instances TENANT_1_VRF instance-type vrf
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes vni 9999
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes export T5_EXPORT
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF description "VRF for Tenant_1"
set routing-instances TENANT_1_VRF interface irb.10
set routing-instances TENANT_1_VRF interface irb.11
set routing-instances TENANT_1_VRF interface irb.12
set routing-instances TENANT_1_VRF interface irb.202
set routing-instances TENANT_1_VRF interface irb.203
set routing-instances TENANT_1_VRF interface lo0.1
```

```
set routing-instances TENANT_1_VRF route-distinguisher 10.80.225.140:9999
set routing-instances TENANT_1_VRF vrf-import VRF1_T5_RT_IMPORT
set routing-instances TENANT_1_VRF vrf-export VRF1_T5_RT_EXPORT
set routing-instances TENANT_1_VRF vrf-target target:1:65001
set routing-instances TENANT_1_VRF vrf-table-label
set routing-options forwarding-table export ECMP-POLICY
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress evpn
set routing-options router-id 10.80.224.140
set routing-options autonomous-system 64730
set protocols evpn vni-options vni 110 vrf-target target:64730:110
set protocols evpn vni-options vni 111 vrf-target target:64730:111
set protocols evpn vni-options vni 112 vrf-target target:64730:112
set protocols evpn vni-options vni 1202 vrf-target target:64730:202
set protocols evpn vni-options vni 1203 vrf-target target:64730:203
set protocols evpn encapsulation vxlan
set protocols evpn default-gateway no-gateway-community
set protocols evpn extended-vni-list 110
set protocols evpn extended-vni-list 111
set protocols evpn extended-vni-list 112
set protocols evpn extended-vni-list 1202
set protocols evpn extended-vni-list 1203
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC description "Group for IBGP peering to reflectors"
set protocols bgp group EVPN_FABRIC local-address 10.80.224.140
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC local-as 64730
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.149
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.150
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for EBGP peering in underlay"
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export FROM_Lo0
set protocols bgp group UNDERLAY local-as 65012
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 10.80.224.12 peer-as 65002
```

```
set protocols bgp group UNDERLAY neighbor 10.80.224.2 peer-as 65001
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols l2-learning global-mac-table-aging-time 600
set protocols l2-learning global-mac-ip-table-aging-time 300
set protocols lldp interface all
set switch-options vtep-source-interface lo0.0
set switch-options route-distinguisher 10.80.224.140:1
set switch-options vrf-import OVERLAY_IMPORT
set switch-options vrf-target target:64730:999
set vlans v10 description "Tenant_1 - VLAN id 10"
set vlans v10 vlan-id 10
set vlans v10 l3-interface irb.10
set vlans v10 vxlan vni 110
set vlans v11 description "Tenant_1 - VLAN id 11"
set vlans v11 vlan-id 11
set vlans v11 l3-interface irb.11
set vlans v11 vxlan vni 111
set vlans v12 description "Tenant_1 - VLAN id 12"
set vlans v12 vlan-id 12
set vlans v12 l3-interface irb.12
set vlans v12 vxlan vni 112
set vlans v202 description "Tenant_1 - VLAN id 202"
set vlans v202 vlan-id 202
set vlans v202 l3-interface irb.202
set vlans v202 vxlan vni 1202
set vlans v203 description "Tenant_1 - VLAN id 203"
set vlans v203 vlan-id 203
set vlans v203 l3-interface irb.203
set vlans v203 vxlan vni 1203
```

Leaf Device 2 in Data Center 1

```
set system arp aging-timer 5
set interfaces xe-0/0/1:0 description "DC1 Server2"
set interfaces xe-0/0/1:0 unit 0 family ethernet-switching vlan members v203
set interfaces et-0/0/49 description "Fabric interface"
set interfaces et-0/0/49 mtu 9200
set interfaces et-0/0/49 unit 0 description "** to DC1-Spine2"
set interfaces et-0/0/49 unit 0 family inet address 10.80.224.15/31
set interfaces et-0/0/50 description "Fabric interface"
set interfaces et-0/0/50 traps
```

```
set interfaces et-0/0/50 mtu 9200
set interfaces et-0/0/50 unit 0 description "** to DC1-Spine1"
set interfaces et-0/0/50 unit 0 family inet address 10.80.224.5/31
set interfaces irb unit 10 virtual-gateway-accept-data
set interfaces irb unit 10 description "** L3 interface for v10 in Tenant_1"
set interfaces irb unit 10 family inet address 10.1.10.12/24 preferred
set interfaces irb unit 10 family inet address 10.1.10.12/24 virtual-gateway-address 10.1.10.1
set interfaces irb unit 11 virtual-gateway-accept-data
set interfaces irb unit 11 description "** L3 interface for v11 in Tenant_1"
set interfaces irb unit 11 family inet address 10.1.11.12/24 preferred
set interfaces irb unit 11 family inet address 10.1.11.12/24 virtual-gateway-address 10.1.11.1
set interfaces irb unit 12 virtual-gateway-accept-data
set interfaces irb unit 12 description "** L3 interface for v12 in Tenant_1"
set interfaces irb unit 12 family inet address 10.1.12.12/24 preferred
set interfaces irb unit 12 family inet address 10.1.12.12/24 virtual-gateway-address 10.1.12.1
set interfaces irb unit 202 virtual-gateway-accept-data
set interfaces irb unit 202 description "** L3 interface for v202 in Tenant_1"
set interfaces irb unit 202 family inet address 10.1.202.12/24 preferred
set interfaces irb unit 202 family inet address 10.1.202.12/24 virtual-gateway-address 10.1.202.1
set interfaces irb unit 203 virtual-gateway-accept-data
set interfaces irb unit 203 description "** L3 interface for v203 in Tenant_1"
set interfaces irb unit 203 family inet address 10.1.203.12/24 preferred
set interfaces irb unit 203 family inet address 10.1.203.12/24 virtual-gateway-address 10.1.203.1
set interfaces lo0 unit 0 description "** DC1-Leaf2"
set interfaces lo0 unit 0 family inet address 10.80.224.141/32
set interfaces lo0 unit 1 family inet address 10.80.225.141/32
set forwarding-options vxlan-routing overlay-ecmp
set forwarding-options vxlan-routing next-hop 32768
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement OVERLAY_IMPORT term 5 from community comm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 5 then accept
set policy-options policy-statement OVERLAY_IMPORT term 10 from community comm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 10 then accept
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 then accept
set policy-options policy-statement T5_EXPORT term fm_direct from protocol direct
set policy-options policy-statement T5_EXPORT term fm_direct then accept
```

```
set policy-options policy-statement T5_EXPORT term fm_static from protocol static
set policy-options policy-statement T5_EXPORT term fm_static then accept
set policy-options policy-statement T5_EXPORT term fm_v4_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v4_host from route-filter 0.0.0.0/0 prefix-
length-range /32-/32
set policy-options policy-statement T5_EXPORT term fm_v4_host then accept
set policy-options policy-statement T5_EXPORT term fm_v6_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v6_host from route-filter 0::0/0 prefix-
length-range /128-/128
set policy-options policy-statement T5_EXPORT term fm_v6_host then accept
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then community add target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 from community target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 from community target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 then accept
set policy-options community comm_pod1 members target:64730:999
set policy-options community comm_pod2 members target:64830:999
set policy-options community shared_202_fm_pod1 members target:64730:202
set policy-options community shared_202_fm_pod2 members target:64830:202
set policy-options community shared_203_fm_pod1 members target:64730:203
set policy-options community shared_203_fm_pod2 members target:64830:203
set policy-options community target_t5_pod1 members target:64730:9999
set policy-options community target_t5_pod2 members target:64830:9999
set routing-instances TENANT_1_VRF instance-type vrf
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes vni 9999
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes export T5_EXPORT
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF description "VRF for Tenant_1"
set routing-instances TENANT_1_VRF interface irb.10
set routing-instances TENANT_1_VRF interface irb.11
set routing-instances TENANT_1_VRF interface irb.12
set routing-instances TENANT_1_VRF interface irb.202
set routing-instances TENANT_1_VRF interface irb.203
set routing-instances TENANT_1_VRF interface lo0.1
set routing-instances TENANT_1_VRF route-distinguisher 10.80.225.141:9999
set routing-instances TENANT_1_VRF vrf-import VRF1_T5_RT_IMPORT
set routing-instances TENANT_1_VRF vrf-export VRF1_T5_RT_EXPORT
set routing-instances TENANT_1_VRF vrf-target target:1:65001
set routing-instances TENANT_1_VRF vrf-table-label
set routing-options forwarding-table export ECMP-POLICY
```

```
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress evpn
set routing-options router-id 10.80.224.141
set routing-options autonomous-system 64730
set protocols evpn vni-options vni 110 vrf-target target:64730:110
set protocols evpn vni-options vni 111 vrf-target target:64730:111
set protocols evpn vni-options vni 112 vrf-target target:64730:112
set protocols evpn vni-options vni 1202 vrf-target target:64730:202
set protocols evpn vni-options vni 1203 vrf-target target:64730:203
set protocols evpn encapsulation vxlan
set protocols evpn default-gateway no-gateway-community
set protocols evpn extended-vni-list 110
set protocols evpn extended-vni-list 111
set protocols evpn extended-vni-list 112
set protocols evpn extended-vni-list 1202
set protocols evpn extended-vni-list 1203
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC description "Group for IBGP peering to reflectors"
set protocols bgp group EVPN_FABRIC local-address 10.80.224.141
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC local-as 64730
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.149
set protocols bgp group EVPN_FABRIC neighbor 10.80.224.150
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for EBGP peering in underlay"
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export FROM_Lo0
set protocols bgp group UNDERLAY local-as 65013
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 10.80.224.4 peer-as 65001
set protocols bgp group UNDERLAY neighbor 10.80.224.14 peer-as 65002
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols l2-learning global-mac-table-aging-time 600
set protocols l2-learning global-mac-ip-table-aging-time 300
set protocols lldp interface all
```

```
set switch-options vtep-source-interface lo0.0
set switch-options route-distinguisher 10.80.224.141:1
set switch-options vrf-import OVERLAY_IMPORT
set switch-options vrf-target target:64730:999
set vlans v10 description "Tenant_1 - VLAN id 10"
set vlans v10 vlan-id 10
set vlans v10 l3-interface irb.10
set vlans v10 vxlan vni 110
set vlans v11 description "Tenant_1 - VLAN id 11"
set vlans v11 vlan-id 11
set vlans v11 l3-interface irb.11
set vlans v11 vxlan vni 111
set vlans v12 description "Tenant_1 - VLAN id 12"
set vlans v12 vlan-id 12
set vlans v12 l3-interface irb.12
set vlans v12 vxlan vni 112
set vlans v202 description "Tenant_1 - VLAN id 202"
set vlans v202 vlan-id 202
set vlans v202 l3-interface irb.202
set vlans v202 vxlan vni 1202
set vlans v203 description "Tenant_1 - VLAN id 203"
set vlans v203 vlan-id 203
set vlans v203 l3-interface irb.203
set vlans v203 vxlan vni 1203
```

Border Spine 1 in Data Center 2

```
set chassis fpc 0 pic 0 port 6 channel-speed 10g
set interfaces et-0/0/0 description "Fabric interface"
set interfaces et-0/0/0 mtu 9200
set interfaces et-0/0/0 unit 0 description "** to DC2-Leaf1"
set interfaces et-0/0/0 unit 0 family inet address 172.16.0.0/31
set interfaces et-0/0/1 description "Fabric interface"
set interfaces et-0/0/1 mtu 9200
set interfaces et-0/0/1 unit 0 description "** to DC2-Leaf2"
set interfaces et-0/0/1 unit 0 family inet address 172.16.0.2/31
set interfaces et-0/0/2 description "Fabric interface"
set interfaces et-0/0/2 mtu 9200
set interfaces et-0/0/2 unit 0 description "** to DC2-Leaf3"
set interfaces et-0/0/2 unit 0 family inet address 172.16.0.4/31
set interfaces xe-0/0/6:1 description "To WAN Router"
set interfaces xe-0/0/6:1 mtu 9200
```

```
set interfaces xe-0/0/6:1 unit 0 family inet address 172.16.1.9/31
set interfaces lo0 unit 0 description "** DC2-Spine1"
set interfaces lo0 unit 0 family inet address 10.0.0.2/32
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-EXPORT term DEFAULT then reject
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-IMPORT term DEFAULT then reject
set routing-options forwarding-table export ECMP-POLICY
set routing-options forwarding-table ecmp-fast-reroute
set routing-options autonomous-system 64830
set routing-options forwarding-table chained-composite-next-hop ingress evpn
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for EBGP peering in underlay"
set protocols bgp group UNDERLAY import UNDERLAY-IMPORT
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export UNDERLAY-EXPORT
set protocols bgp group UNDERLAY local-as 65101
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 172.16.0.3 peer-as 65018
set protocols bgp group UNDERLAY neighbor 172.16.0.1 peer-as 65019
set protocols bgp group UNDERLAY neighbor 172.16.0.5 peer-as 65020
set protocols bgp group UNDERLAY neighbor 172.16.1.8 peer-as 65229
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC description "Group for overlay IBGP peering to reflectors"
set protocols bgp group EVPN_FABRIC local-address 10.0.0.2
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC cluster 10.0.0.2
```

```
set protocols bgp group EVPN_FABRIC local-as 64830
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.19
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.18
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.14
set protocols bgp group EVPN_FABRIC vpn-apply-export
set protocols bgp group OVERLAY_INTERDC type external
set protocols bgp group OVERLAY_INTERDC description "Group for overlay EBGP peering to remote DC"
set protocols bgp group OVERLAY_INTERDC multihop no-nexthop-change
set protocols bgp group OVERLAY_INTERDC local-address 10.0.0.2
set protocols bgp group OVERLAY_INTERDC family evpn signaling delay-route-advertisements minimum-
delay routing-uptime 480
set protocols bgp group OVERLAY_INTERDC local-as 64830
set protocols bgp group OVERLAY_INTERDC multipath multiple-as
set protocols bgp group OVERLAY_INTERDC neighbor 10.80.224.149 peer-as 64730
set protocols bgp group OVERLAY_INTERDC neighbor 10.80.224.150 peer-as 64730
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols lldp interface all
```

Border Spine Device 2 in Data Center 2

```
set chassis fpc 0 pic 0 port 6 channel-speed 10g
set interfaces et-0/0/0 description "Fabric interface"
set interfaces et-0/0/0 mtu 9200
set interfaces et-0/0/0 unit 0 description "** to DC2-Leaf1"
set interfaces et-0/0/0 unit 0 family inet address 172.16.0.6/31
set interfaces et-0/0/1 description "Fabric interface"
set interfaces et-0/0/1 mtu 9200
set interfaces et-0/0/1 unit 0 description "** to DC2-Leaf2"
set interfaces et-0/0/1 unit 0 family inet address 172.16.0.8/31
set interfaces et-0/0/2 description "Fabric interface"
set interfaces et-0/0/2 mtu 9200
set interfaces et-0/0/2 unit 0 description "** to DC2-Leaf3"
set interfaces et-0/0/2 unit 0 family inet address 172.16.0.10/31
set interfaces xe-0/0/6:1 description "To WAN Router"
set interfaces xe-0/0/6:1 mtu 9200
set interfaces xe-0/0/6:1 unit 0 family inet address 172.16.1.11/31
set interfaces lo0 unit 0 description "** DC2-Spine2"
set interfaces lo0 unit 0 family inet address 10.0.0.3/32
```

```
set forwarding-options vxlan-routing overlay-ecmp
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-EXPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-EXPORT term DEFAULT then reject
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter
10.80.224.128/25 orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK from route-filter 10.0.0.0/24
orlonger
set policy-options policy-statement UNDERLAY-IMPORT term LOOPBACK then accept
set policy-options policy-statement UNDERLAY-IMPORT term DEFAULT then reject
set routing-options forwarding-table export ECMP-POLICY
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress evpn
set routing-options autonomous-system 64830
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for EBGP peering in underlay"
set protocols bgp group UNDERLAY import UNDERLAY-IMPORT
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export UNDERLAY-EXPORT
set protocols bgp group UNDERLAY local-as 65102
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 172.16.0.9 peer-as 65018
set protocols bgp group UNDERLAY neighbor 172.16.0.7 peer-as 65019
set protocols bgp group UNDERLAY neighbor 172.16.0.11 peer-as 65020
set protocols bgp group UNDERLAY neighbor 172.16.1.10 peer-as 65229
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC description "Group for overlay IBGP peering to reflectors"
set protocols bgp group EVPN_FABRIC local-address 10.0.0.3
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC cluster 10.0.0.3
set protocols bgp group EVPN_FABRIC local-as 64830
set protocols bgp group EVPN_FABRIC multipath
```

```
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.19
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.18
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.14
set protocols bgp group EVPN_FABRIC vpn-apply-export
set protocols bgp group OVERLAY_INTERDC type external
set protocols bgp group OVERLAY_INTERDC description "Group for overlay EBGP peering to remote DC"
set protocols bgp group OVERLAY_INTERDC multihop no-nexthop-change
set protocols bgp group OVERLAY_INTERDC local-address 10.0.0.3
set protocols bgp group OVERLAY_INTERDC family evpn signaling delay-route-advertisements minimum-
delay routing-uptime 480
set protocols bgp group OVERLAY_INTERDC local-as 64830
set protocols bgp group OVERLAY_INTERDC multipath multiple-as
set protocols bgp group OVERLAY_INTERDC neighbor 10.80.224.149 peer-as 64730
set protocols bgp group OVERLAY_INTERDC neighbor 10.80.224.150 peer-as 64730
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols lldp interface all
```

Leaf Device 1 in Data Center 2

```
set system arp aging-timer 5
set chassis aggregated-devices ethernet device-count 20
set interfaces xe-0/0/13 ether-options 802.3ad ae1
set interfaces xe-0/0/14 ether-options 802.3ad ae2
set interfaces et-0/0/48 description "Fabric interface"
set interfaces et-0/0/48 mtu 9200
set interfaces et-0/0/48 unit 0 description "** to DC2-Spine1"
set interfaces et-0/0/48 unit 0 family inet address 172.16.0.1/31
set interfaces et-0/0/49 description "Fabric interface"
set interfaces et-0/0/49 mtu 9200
set interfaces et-0/0/49 unit 0 description "** to DC2-Spine2"
set interfaces et-0/0/49 unit 0 family inet address 172.16.0.7/31
set interfaces ae1 description "DC2 Server1"
set interfaces ae1 esi 00:00:00:00:00:02:02:02:02:01
set interfaces ae1 esi all-active
set interfaces ae1 aggregated-ether-options link-speed 10g
set interfaces ae1 aggregated-ether-options lacp active
set interfaces ae1 aggregated-ether-options lacp periodic fast
set interfaces ae1 aggregated-ether-options lacp system-id 00:02:02:02:02:01
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk
```

```
set interfaces ae1 unit 0 family ethernet-switching vlan members v170
set interfaces ae1 unit 0 family ethernet-switching vlan members v171
set interfaces ae1 unit 0 family ethernet-switching vlan members v172
set interfaces ae1 unit 0 family ethernet-switching vlan members v202
set interfaces ae1 unit 0 family ethernet-switching vlan members v203
set interfaces ae2 description "DC2 Server2"
set interfaces ae2 esi 00:00:00:00:00:02:02:02:02:02
set interfaces ae2 esi all-active
set interfaces ae2 aggregated-ether-options link-speed 10g
set interfaces ae2 aggregated-ether-options lacp active
set interfaces ae2 aggregated-ether-options lacp periodic fast
set interfaces ae2 aggregated-ether-options lacp system-id 00:02:02:02:02:02
set interfaces ae2 unit 0 family ethernet-switching interface-mode trunk
set interfaces ae2 unit 0 family ethernet-switching vlan members v170
set interfaces ae2 unit 0 family ethernet-switching vlan members v171
set interfaces ae2 unit 0 family ethernet-switching vlan members v172
set interfaces ae2 unit 0 family ethernet-switching vlan members v202
set interfaces ae2 unit 0 family ethernet-switching vlan members v203
set interfaces irb unit 170 virtual-gateway-accept-data
set interfaces irb unit 170 description "** L3 interface for v170 in Tenant_1"
set interfaces irb unit 170 family inet address 10.1.170.19/24 preferred
set interfaces irb unit 170 family inet address 10.1.170.19/24 virtual-gateway-address 10.1.170.1
set interfaces irb unit 171 virtual-gateway-accept-data
set interfaces irb unit 171 description "** L3 interface for v171 in Tenant_1"
set interfaces irb unit 171 family inet address 10.1.171.19/24 preferred
set interfaces irb unit 171 family inet address 10.1.171.19/24 virtual-gateway-address 10.1.171.1
set interfaces irb unit 172 virtual-gateway-accept-data
set interfaces irb unit 172 description "** L3 interface for v172 in Tenant_1"
set interfaces irb unit 172 family inet address 10.1.172.19/24 preferred
set interfaces irb unit 172 family inet address 10.1.172.19/24 virtual-gateway-address 10.1.172.1
set interfaces irb unit 202 virtual-gateway-accept-data
set interfaces irb unit 202 description "** L3 interface for v202 in Tenant_1"
set interfaces irb unit 202 family inet address 10.1.202.19/24 preferred
set interfaces irb unit 202 family inet address 10.1.202.19/24 virtual-gateway-address 10.1.202.1
set interfaces irb unit 203 virtual-gateway-accept-data
set interfaces irb unit 203 description "** L3 interface for v203 in Tenant_1"
set interfaces irb unit 203 family inet address 10.1.203.19/24 preferred
set interfaces irb unit 203 family inet address 10.1.203.19/24 virtual-gateway-address 10.1.203.1
set interfaces lo0 traps
set interfaces lo0 unit 0 description "** DC2-Leaf1"
set interfaces lo0 unit 0 family inet address 10.0.0.19/32
set interfaces lo0 unit 1 family inet address 10.0.1.19/32
set forwarding-options vxlan-routing overlay-ecmp
```

```
set forwarding-options vxlan-routing next-hop 32768
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement OVERLAY_IMPORT term 5 from community comm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 5 then accept
set policy-options policy-statement OVERLAY_IMPORT term 10 from community comm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 10 then accept
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 then accept
set policy-options policy-statement T5_EXPORT term fm_direct from protocol direct
set policy-options policy-statement T5_EXPORT term fm_direct then accept
set policy-options policy-statement T5_EXPORT term fm_static from protocol static
set policy-options policy-statement T5_EXPORT term fm_static then accept
set policy-options policy-statement T5_EXPORT term fm_v4_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v4_host from route-filter 0.0.0.0/0 prefix-
length-range /32-/32
set policy-options policy-statement T5_EXPORT term fm_v4_host then accept
set policy-options policy-statement T5_EXPORT term fm_v6_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v6_host from route-filter 0::0/0 prefix-
length-range /128-/128
set policy-options policy-statement T5_EXPORT term fm_v6_host then accept
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then community add target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 from community target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 from community target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 then accept
set policy-options community comm_pod1 members target:64730:999
set policy-options community comm_pod2 members target:64830:999
set policy-options community shared_202_fm_pod1 members target:64730:202
set policy-options community shared_202_fm_pod2 members target:64830:202
set policy-options community shared_203_fm_pod1 members target:64730:203
set policy-options community shared_203_fm_pod2 members target:64830:203
set policy-options community target_t5_pod1 members target:64730:9999
set policy-options community target_t5_pod2 members target:64830:9999
set routing-instances TENANT_1_VRF description "VRF for Tenant_1"
set routing-instances TENANT_1_VRF instance-type vrf
set routing-instances TENANT_1_VRF interface irb.170
```

```
set routing-instances TENANT_1_VRF interface irb.171
set routing-instances TENANT_1_VRF interface irb.172
set routing-instances TENANT_1_VRF interface irb.202
set routing-instances TENANT_1_VRF interface irb.203
set routing-instances TENANT_1_VRF interface lo0.1
set routing-instances TENANT_1_VRF route-distinguisher 10.0.1.19:9999
set routing-instances TENANT_1_VRF vrf-import VRF1_T5_RT_IMPORT
set routing-instances TENANT_1_VRF vrf-export VRF1_T5_RT_EXPORT
set routing-instances TENANT_1_VRF vrf-target target:1:65001
set routing-instances TENANT_1_VRF vrf-table-label
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes vni 9999
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes export T5_EXPORT
set routing-options router-id 10.0.0.19
set routing-options autonomous-system 64830
set routing-options forwarding-table export ECMP-POLICY
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress evpn
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC description "Group for overlay IBGP peering to reflectors"
set protocols bgp group EVPN_FABRIC local-address 10.0.0.19
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC local-as 64830
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.2
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.3
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for EBGP peering in underlay"
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export FROM_Lo0
set protocols bgp group UNDERLAY local-as 65019
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 172.16.0.0 peer-as 65101
```

```
set protocols bgp group UNDERLAY neighbor 172.16.0.6 peer-as 65102
set protocols evpn vni-options vni 1170 vrf-target target:64830:1170
set protocols evpn vni-options vni 1171 vrf-target target:64830:1171
set protocols evpn vni-options vni 1172 vrf-target target:64830:1172
set protocols evpn vni-options vni 1202 vrf-target target:64830:202
set protocols evpn vni-options vni 1203 vrf-target target:64830:203
set protocols evpn encapsulation vxlan
set protocols evpn default-gateway no-gateway-community
set protocols evpn extended-vni-list 1170
set protocols evpn extended-vni-list 1171
set protocols evpn extended-vni-list 1172
set protocols evpn extended-vni-list 1202
set protocols evpn extended-vni-list 1203
set protocols l2-learning global-mac-table-aging-time 600
set protocols l2-learning global-mac-ip-table-aging-time 300
set protocols lldp interface all
set switch-options vtep-source-interface lo0.0
set switch-options route-distinguisher 10.0.0.19:1
set switch-options vrf-import OVERLAY_IMPORT
set switch-options vrf-target target:64830:999
set vlans v170 description "Tenant_1 - VLAN id 170"
set vlans v170 vlan-id 170
set vlans v170 l3-interface irb.170
set vlans v170 vxlan vni 1170
set vlans v171 description "Tenant_1 - VLAN id 171"
set vlans v171 vlan-id 171
set vlans v171 l3-interface irb.171
set vlans v171 vxlan vni 1171
set vlans v172 description "Tenant_1 - VLAN id 172"
set vlans v172 vlan-id 172
set vlans v172 l3-interface irb.172
set vlans v172 vxlan vni 1172
set vlans v202 description "Tenant_1 - VLAN id 202"
set vlans v202 vlan-id 202
set vlans v202 l3-interface irb.202
set vlans v202 vxlan vni 1202
set vlans v203 description "Tenant_1 - VLAN id 203"
set vlans v203 vlan-id 203
set vlans v203 l3-interface irb.203
set vlans v203 vxlan vni 1203
```

Leaf Device 2 in Data Center 2

```
set system arp aging-timer 5
set chassis aggregated-devices ethernet device-count 20
set interfaces xe-0/0/13 ether-options 802.3ad ae1
set interfaces xe-0/0/14 ether-options 802.3ad ae2
set interfaces et-0/0/48 description "Fabric interface"
set interfaces et-0/0/48 mtu 9200
set interfaces et-0/0/48 unit 0 description "** to DC2-Spine1"
set interfaces et-0/0/48 unit 0 family inet address 172.16.0.3/31
set interfaces et-0/0/49 description "Fabric interface"
set interfaces et-0/0/49 mtu 9200
set interfaces et-0/0/49 unit 0 description "** to DC2-Spine2"
set interfaces et-0/0/49 unit 0 family inet address 172.16.0.9/31
set interfaces ae1 description "DC2 Server1"
set interfaces ae1 mtu 9200
set interfaces ae1 esi 00:00:00:00:00:02:02:02:02:01
set interfaces ae1 esi all-active
set interfaces ae1 aggregated-ether-options link-speed 10g
set interfaces ae1 aggregated-ether-options lacp active
set interfaces ae1 aggregated-ether-options lacp periodic fast
set interfaces ae1 aggregated-ether-options lacp system-id 00:02:02:02:02:01
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk
set interfaces ae1 unit 0 family ethernet-switching vlan members v170
set interfaces ae1 unit 0 family ethernet-switching vlan members v171
set interfaces ae1 unit 0 family ethernet-switching vlan members v172
set interfaces ae1 unit 0 family ethernet-switching vlan members v202
set interfaces ae1 unit 0 family ethernet-switching vlan members v203
set interfaces ae2 description "DC2 Server2"
set interfaces ae2 mtu 9200
set interfaces ae2 esi 00:00:00:00:00:02:02:02:02:02
set interfaces ae2 esi all-active
set interfaces ae2 aggregated-ether-options link-speed 10g
set interfaces ae2 aggregated-ether-options lacp active
set interfaces ae2 aggregated-ether-options lacp periodic fast
set interfaces ae2 aggregated-ether-options lacp system-id 00:02:02:02:02:02
set interfaces ae2 unit 0 family ethernet-switching interface-mode trunk
set interfaces ae2 unit 0 family ethernet-switching vlan members v170
set interfaces ae2 unit 0 family ethernet-switching vlan members v171
set interfaces ae2 unit 0 family ethernet-switching vlan members v172
set interfaces ae2 unit 0 family ethernet-switching vlan members v202
set interfaces ae2 unit 0 family ethernet-switching vlan members v203
```

```
set interfaces irb unit 170 virtual-gateway-accept-data
set interfaces irb unit 170 description "** L3 interface for v170 in Tenant_1"
set interfaces irb unit 170 family inet address 10.1.170.18/24 preferred
set interfaces irb unit 170 family inet address 10.1.170.18/24 virtual-gateway-address 10.1.170.1
set interfaces irb unit 171 virtual-gateway-accept-data
set interfaces irb unit 171 description "** L3 interface for v171 in Tenant_1"
set interfaces irb unit 171 family inet address 10.1.171.18/24 preferred
set interfaces irb unit 171 family inet address 10.1.171.18/24 virtual-gateway-address 10.1.171.1
set interfaces irb unit 172 virtual-gateway-accept-data
set interfaces irb unit 172 description "** L3 interface for v172 in Tenant_1"
set interfaces irb unit 172 family inet address 10.1.172.18/24 preferred
set interfaces irb unit 172 family inet address 10.1.172.18/24 virtual-gateway-address 10.1.172.1
set interfaces irb unit 202 virtual-gateway-accept-data
set interfaces irb unit 202 description "** L3 interface for v202 in Tenant_1"
set interfaces irb unit 202 family inet address 10.1.202.18/24 preferred
set interfaces irb unit 202 family inet address 10.1.202.18/24 virtual-gateway-address 10.1.202.1
set interfaces irb unit 203 virtual-gateway-accept-data
set interfaces irb unit 203 description "** L3 interface for v203 in Tenant_1"
set interfaces irb unit 203 family inet address 10.1.203.18/24 preferred
set interfaces irb unit 203 family inet address 10.1.203.18/24 virtual-gateway-address 10.1.203.1
set interfaces lo0 unit 0 description "** DC2-Leaf2"
set interfaces lo0 unit 0 family inet address 10.0.0.18/32
set interfaces lo0 unit 1 family inet address 10.0.1.18/32
set forwarding-options vxlan-routing next-hop 32768
set forwarding-options vxlan-routing overlay-ecmp
set forwarding-options vxlan-routing next-hop 32768
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
set policy-options policy-statement FROM_Lo0 term 10 from interface lo0.0
set policy-options policy-statement FROM_Lo0 term 10 then accept
set policy-options policy-statement FROM_Lo0 term 20 then reject
set policy-options policy-statement OVERLAY_IMPORT term 5 from community comm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 5 then accept
set policy-options policy-statement OVERLAY_IMPORT term 10 from community comm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 10 then accept
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_202_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod2
set policy-options policy-statement OVERLAY_IMPORT term 20 from community shared_203_fm_pod1
set policy-options policy-statement OVERLAY_IMPORT term 20 then accept
set policy-options policy-statement T5_EXPORT term fm_direct from protocol direct
set policy-options policy-statement T5_EXPORT term fm_direct then accept
set policy-options policy-statement T5_EXPORT term fm_static from protocol static
set policy-options policy-statement T5_EXPORT term fm_static then accept
```

```
set policy-options policy-statement T5_EXPORT term fm_v4_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v4_host from route-filter 0.0.0.0/0 prefix-
length-range /32-/32
set policy-options policy-statement T5_EXPORT term fm_v4_host then accept
set policy-options policy-statement T5_EXPORT term fm_v6_host from protocol evpn
set policy-options policy-statement T5_EXPORT term fm_v6_host from route-filter 0::0/0 prefix-
length-range /128-/128
set policy-options policy-statement T5_EXPORT term fm_v6_host then accept
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then community add target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_EXPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 from community target_t5_pod1
set policy-options policy-statement VRF1_T5_RT_IMPORT term t1 then accept
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 from community target_t5_pod2
set policy-options policy-statement VRF1_T5_RT_IMPORT term t2 then accept
set policy-options community comm_pod1 members target:64730:999
set policy-options community comm_pod2 members target:64830:999
set policy-options community shared_202_fm_pod1 members target:64730:202
set policy-options community shared_202_fm_pod2 members target:64830:202
set policy-options community shared_203_fm_pod1 members target:64730:203
set policy-options community shared_203_fm_pod2 members target:64830:203
set policy-options community target_t5_pod1 members target:64730:9999
set policy-options community target_t5_pod2 members target:64830:9999
set routing-instances TENANT_1_VRF description "VRF for Tenant_1"
set routing-instances TENANT_1_VRF instance-type vrf
set routing-instances TENANT_1_VRF interface irb.170
set routing-instances TENANT_1_VRF interface irb.171
set routing-instances TENANT_1_VRF interface irb.172
set routing-instances TENANT_1_VRF interface irb.202
set routing-instances TENANT_1_VRF interface irb.203
set routing-instances TENANT_1_VRF interface lo0.1
set routing-instances TENANT_1_VRF route-distinguisher 10.0.1.18:9999
set routing-instances TENANT_1_VRF vrf-import VRF1_T5_RT_IMPORT
set routing-instances TENANT_1_VRF vrf-export VRF1_T5_RT_EXPORT
set routing-instances TENANT_1_VRF vrf-target target:1:65001
set routing-instances TENANT_1_VRF vrf-table-label
set routing-instances TENANT_1_VRF routing-options multipath
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes vni 9999
set routing-instances TENANT_1_VRF protocols evpn ip-prefix-routes export T5_EXPORT
set routing-options router-id 10.0.0.18
set routing-options autonomous-system 64830
set routing-options forwarding-table export ECMP-POLICY
```

```
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress evpn
set protocols bgp hold-time 10
set protocols bgp log-updown
set protocols bgp group EVPN_FABRIC type internal
set protocols bgp group EVPN_FABRIC bfd-liveness-detection minimum-interval 1000
set protocols bgp group EVPN_FABRIC bfd-liveness-detection multiplier 3
set protocols bgp group EVPN_FABRIC description "Group for overlay IBGP peering to reflectors"
set protocols bgp group EVPN_FABRIC local-address 10.0.0.18
set protocols bgp group EVPN_FABRIC family evpn signaling
set protocols bgp group EVPN_FABRIC authentication-key "samplepassword-fortesting"
set protocols bgp group EVPN_FABRIC local-as 64830
set protocols bgp group EVPN_FABRIC multipath
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.2
set protocols bgp group EVPN_FABRIC neighbor 10.0.0.3
set protocols bgp group UNDERLAY type external
set protocols bgp group UNDERLAY description "Group for EBGP peering in underlay"
set protocols bgp group UNDERLAY family inet unicast
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export FROM_Lo0
set protocols bgp group UNDERLAY local-as 65018
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY neighbor 172.16.0.2 peer-as 65101
set protocols bgp group UNDERLAY neighbor 172.16.0.8 peer-as 65102
set protocols evpn vni-options vni 1170 vrf-target target:64830:1170
set protocols evpn vni-options vni 1171 vrf-target target:64830:1171
set protocols evpn vni-options vni 1172 vrf-target target:64830:1172
set protocols evpn vni-options vni 1202 vrf-target target:64830:202
set protocols evpn vni-options vni 1203 vrf-target target:64830:203
set protocols evpn encapsulation vxlan
set protocols evpn default-gateway no-gateway-community
set protocols evpn extended-vni-list 1170
set protocols evpn extended-vni-list 1171
set protocols evpn extended-vni-list 1172
set protocols evpn extended-vni-list 1202
set protocols evpn extended-vni-list 1203
set protocols l2-learning global-mac-table-aging-time 600
set protocols l2-learning global-mac-ip-table-aging-time 300
set protocols lldp interface all
set switch-options vtep-source-interface lo0.0
set switch-options route-distinguisher 10.0.0.18:1
```

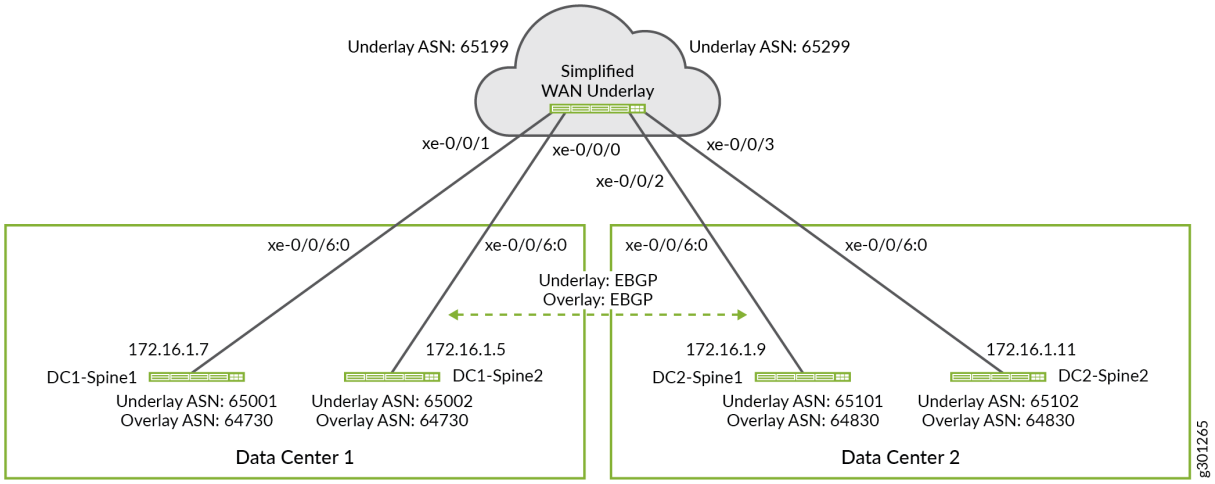
```
set switch-options vrf-import OVERLAY_IMPORT
set switch-options vrf-target target:64830:999
set vlans v170 description "Tenant_1 - VLAN id 170"
set vlans v170 vlan-id 170
set vlans v170 l3-interface irb.170
set vlans v170 vxlan vni 1170
set vlans v171 description "Tenant_1 - VLAN id 171"
set vlans v171 vlan-id 171
set vlans v171 l3-interface irb.171
set vlans v171 vxlan vni 1171
set vlans v172 description "Tenant_1 - VLAN id 172"
set vlans v172 vlan-id 172
set vlans v172 l3-interface irb.172
set vlans v172 vxlan vni 1172
set vlans v202 description "Tenant_1 - VLAN id 202"
set vlans v202 vlan-id 202
set vlans v202 l3-interface irb.202
set vlans v202 vxlan vni 1202
set vlans v203 description "Tenant_1 - VLAN id 203"
set vlans v203 vlan-id 203
set vlans v203 l3-interface irb.203
set vlans v203 vxlan vni 1203
```

Simplified WAN router configuration for testing.

The focus of this example is on the configuration and operation of the spine devices for OTT DCI. As a result the WAN underlay is treated opaquely. From the perspective of the spine devices the WAN routers, and whatever complexity is used in the WAN cloud, for example an MPLS-based Layer 3 VPN, is of no concern. The spines simply use their local BGP peering to exchange underlay routes for the loopback addresses used in each DC.

A simplified WAN router configuration is provided to help in testing and to better illustrate the functionality provided by the WAN cloud, from the perspective of the spine devices. Figure 6 on page 56 provides a topology that can be used for testing with a single routing device acting as a WAN cloud.

Figure 6: A Simplified WAN Underlay




```
set interfaces xe-0/0/0 unit 0 family inet address 172.16.1.4/31
set interfaces xe-0/0/1 unit 0 family inet address 172.16.1.6/31
set interfaces xe-0/0/2 unit 0 family inet address 172.16.1.8/31
set interfaces xe-0/0/3 unit 0 family inet address 172.16.1.10/31
set routing-options autonomous-system 65299
set routing-options forwarding-table export ECMP-POLICY
set protocols bgp group UNDERLAY authentication-key "samplepassword-fortesting"
set protocols bgp group UNDERLAY export dci
set protocols bgp group UNDERLAY bfd-liveness-detection minimum-interval 350
set protocols bgp group UNDERLAY bfd-liveness-detection multiplier 3
set protocols bgp group UNDERLAY multipath multiple-as
set protocols bgp group UNDERLAY neighbor 172.16.1.9 peer-as 65101
set protocols bgp group UNDERLAY neighbor 172.16.1.11 peer-as 65102
set protocols bgp group UNDERLAY neighbor 172.16.1.7 peer-as 65001
set protocols bgp group UNDERLAY neighbor 172.16.1.7 local-as 65199
set protocols bgp group UNDERLAY neighbor 172.16.1.5 peer-as 65002
set protocols bgp group UNDERLAY neighbor 172.16.1.5 local-as 65199
set policy-options policy-statement dci term 1 from protocol direct
set policy-options policy-statement dci term 1 then accept
set policy-options policy-statement ECMP-POLICY then load-balance per-packet
```

RELATED DOCUMENTATION

- Data Center EVPN-VXLAN Fabric Architecture Guide