Troubleshoot ACI External Forwarding

Contents

Introduction

This document describes steps to understand and troubleshoot an L3out in ACI

Background Information

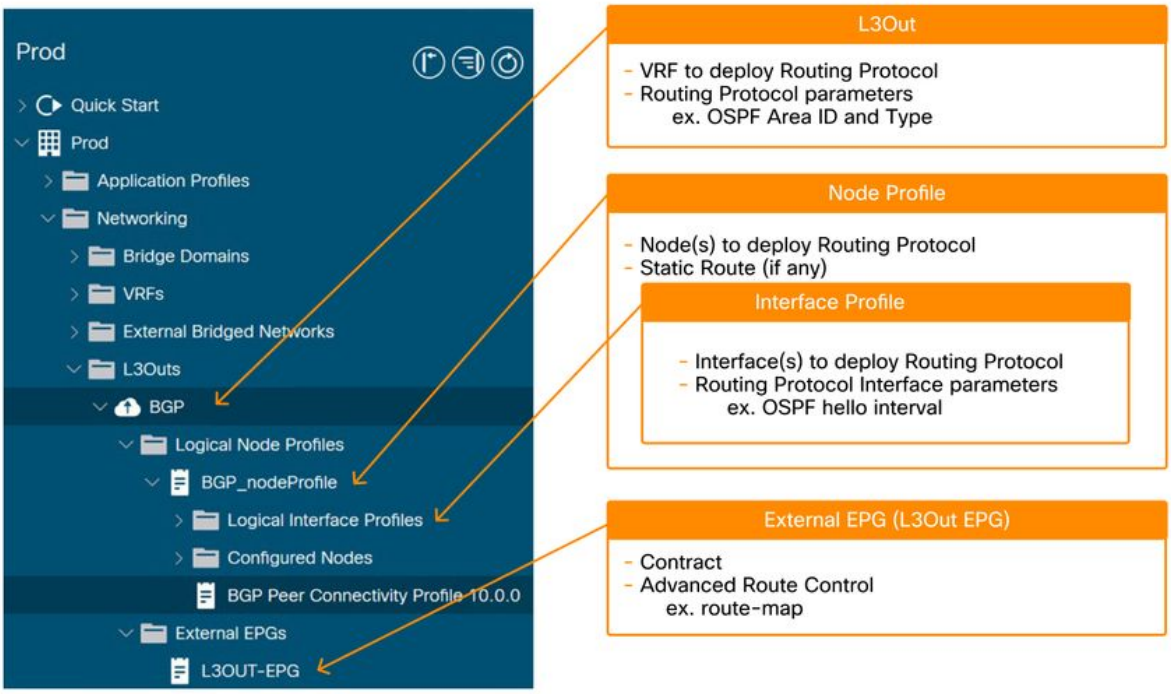
The material from this document was extracted from the Troubleshooting Cisco Application Centric Infrastructure, Second Edition book specifically the **External Forwarding - Overview, External Forwarding - Adjacencies, External Forwarding - Route advertisement, External Forwarding - Contract and L3out** and **External Forwarding - Share L3out** chapters.

Overview

L3Out components

The following picture illustrates the major building blocks required to configure an L3 Outside (L3Out).

Major components of a L3Out



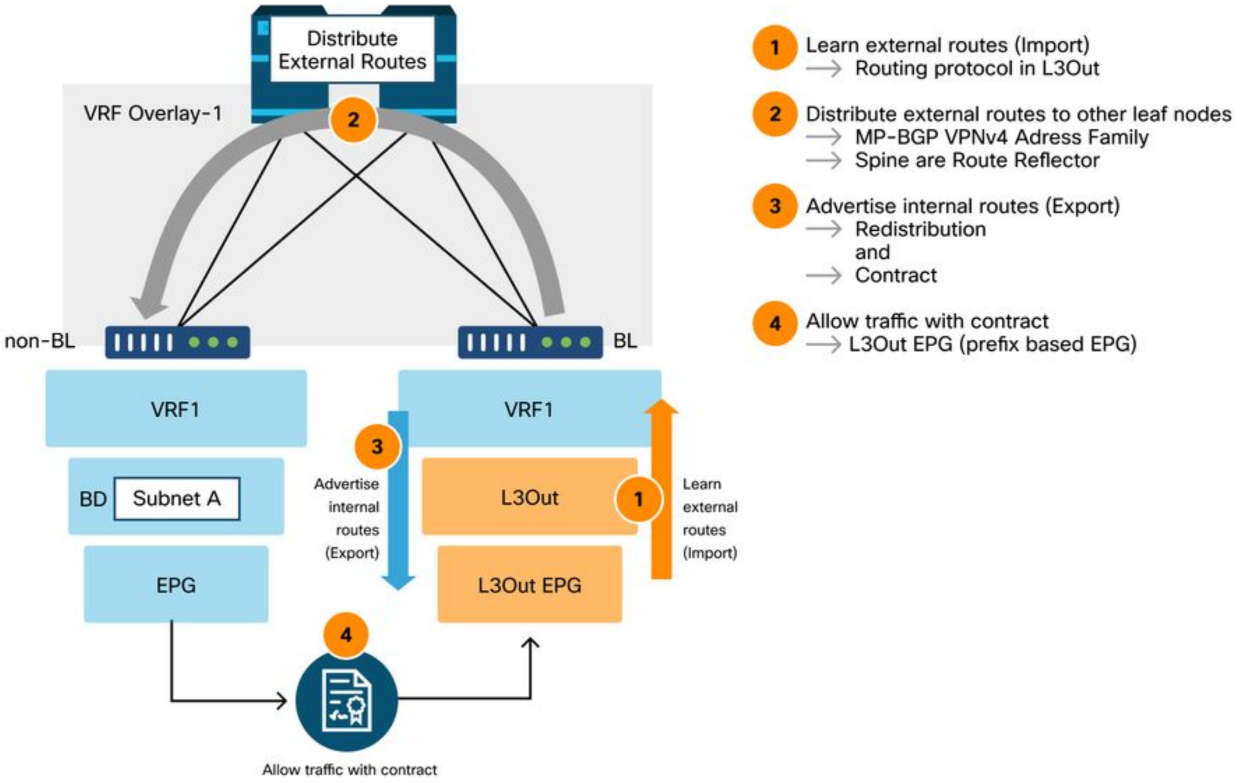
1. Root of L3Out: Select a routing protocol to deploy (such as OSPF, BGP).Select a VRF to deploy the routing protocol.Select an L3Out Domain to define available leaf interfaces and VLAN for the L3Out.
2. Node Profile: Select leaf switches to deploy the routing protocol. These are typically known as 'Border Leaf Switches' (BL).Configure Router-ID (RID) for the routing protocol on each border leaf. Unlike a normal router, ACI does not automatically assign Router-ID based on an IP address on the switch.Configure a static route.
3. Interface profile: Configure leaf interfaces to run the routing protocol.
i.e. Interface type (SVI, routed-port, sub-interface), interface ID and IP addresses etc.Select a policy for interface level routing protocol parameters (such as hello interval).
4. External EPG (L3Out EPG): An 'External EPG' is a hard requirement to deploy all policy tied

to the L3Out, such as IP addresses or SVIs to establish neighbors. Details on how to use external EPGs will be covered later.

External routing

The following diagram shows the high-level operation involved for external routing.

High-level external routing flow

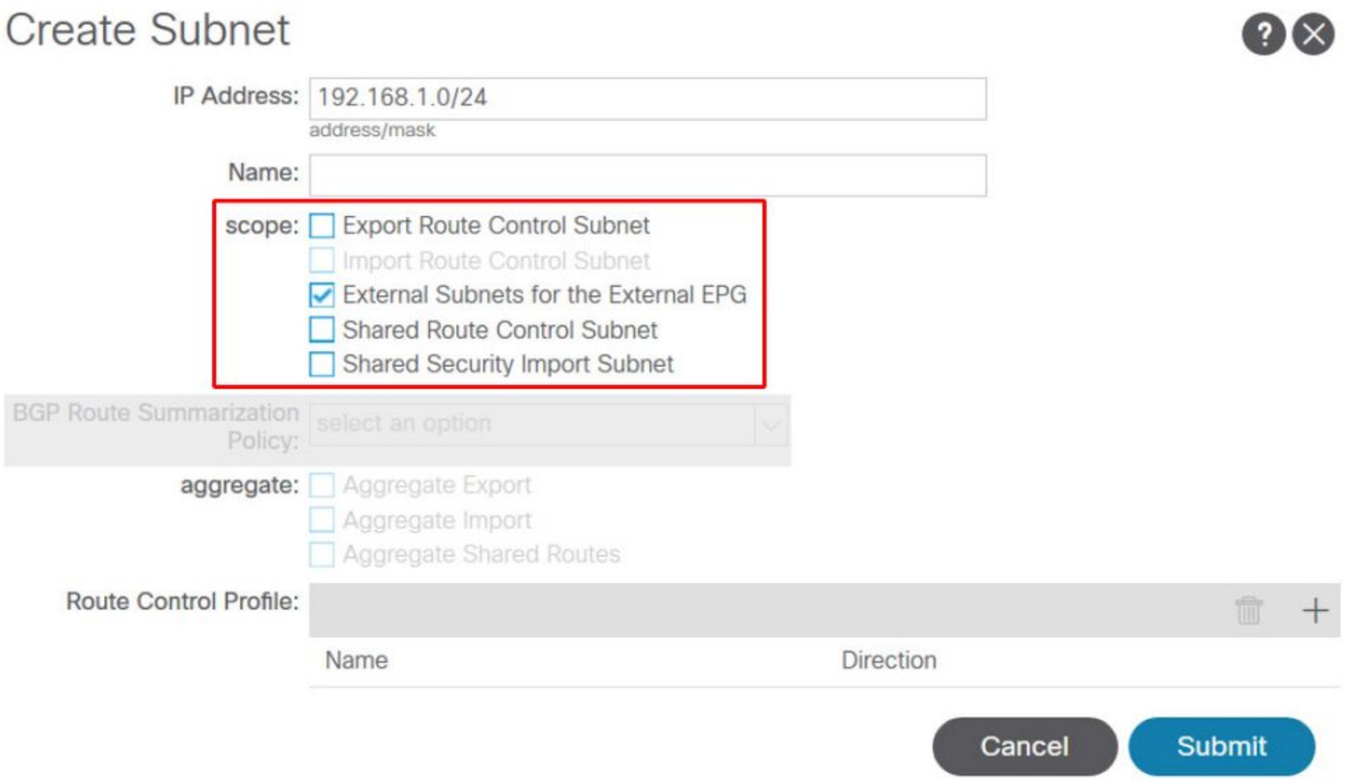

Distribute External Routes
VRF Overlay-1
2
non-BL
BL
VRF1
VRF1
BD
Subnet A
EPG
3
Advertise internal routes (Export)
L3Out
1
Learn external routes (Import)
L3Out EPG
4
Allow traffic with contract
1 Learn external routes (Import)
→ Routing protocol in L3Out
2 Distribute external routes to other leaf nodes
→ MP-BGP VPNv4 Adress Family
→ Spine are Route Reflector
3 Advertise internal routes (Export)
→ Redistribution
and
→ Contract
4 Allow traffic with contract
→ L3Out EPG (prefix based EPG)

1. The BL(s) will establish routing protocol adjacencies with external routers.
2. Route prefixes are received from external routers and are injected in MP-BGP as the VPNv4 address-family path. At a minimum, two spine nodes must be configured as BGP route reflectors to reflect external routes to all leaf nodes.
3. Internal prefixes (BD subnets) and/or prefixes received from other L3Out must be explicitly redistributed into the routing protocol to be advertised to the external router.
4. Security enforcement: an L3Out contains at least one L3Out EPG. A contract must be consumed or provided on the L3Out EPG (also called l3extInstP from the class name) to allow traffic in/out of the L3Out.

L3Out EPG configuration options

In the L3Out EPG section, subnets can be defined with a series of 'Scope' and 'Aggregate' options as illustrated below:

An L3Out subnet being defined including 'scope' definition



Create Subnet

IP Address: 192.168.1.0/24
address/mask

Name:

scope: □ Export Route Control Subnet
□ Import Route Control Subnet
☑ External Subnets for the External EPG
□ Shared Route Control Subnet
□ Shared Security Import Subnet

BGP Route Summarization Policy: select an option

aggregate: □ Aggregate Export
□ Aggregate Import
□ Aggregate Shared Routes

Route Control Profile:

'Scope' options:

- **Export Route Control Subnet:** This scope is to advertise (export) a subnet from ACI to outside via the L3Out. Although this is mainly for Transit Routing, this could also be used to advertise a BD subnet as described in the "ACI BD subnet advertisement" section.
- **Import Route Control Subnet:** This scope is about learning (importing) an external subnet from the L3Out. By default, this scope is disabled, hence it’s greyed out, and a border leaf (BL) learns any routes from a routing protocol. This scope can be enabled when it needs to limit external routes learned via OSPF and BGP. This is not supported for EIGRP. To use this scope, 'Import Route Control Enforcement' needs to be enabled first on a given L3Out.
- **External Subnets for the External EPG (import-security):** This scope is used to allow packets with the configured subnet from or to the L3Out with a contract. It classifies a packet into the configured L3Out EPG based on the subnet so that a contract on the L3Out EPG can be applied to the packet. This scope is a Longest Prefix Match instead of an exact match like other scopes for routing table. If 10.0.0.0/16 is configured with 'External Subnets for the External EPG' in L3Out EPG A, any packets with IP in that subnet, such as 10.0.1.1, will be classified into the L3Out EPG A to use a contract on it. This does not mean 'External Subnets for the External EPG' scope installs a route 10.0.0.0/16 in a routing table. It will create a different internal table to map a subnet to an EPG (pcTag) purely for a contract. It does not have any effects on routing protocol behaviors. This scope is to be configured on a L3Out that is learning the subnet.
- **Shared Route Control Subnet:** This scope is to leak an external subnet to another VRF. ACI uses MP-BGP and Route Target to leak an external route from one VRF to another. This scope creates a prefix-list with the subnet, which is used as a filter to export/import routes with route target in MP-BGP. This scope is to be configured on a L3Out that is learning the subnet in the original VRF.
- **Shared Security Import Subnet:** This scope is used to allow packets with the configured subnet when the packets are moving across VRFs with a L3Out. A route in a routing table is

leaked to another VRF with 'Shared Route Control Subnet' as mentioned above. However, another VRF has yet to know which EPG the leaked route should belong to. The 'Shared Security Import Subnet' informs another VRF of the L3Out EPG which the leaked route belongs to. Hence, this scope can be used only when 'External Subnets for the External EPG' is also used, otherwise the original VRF doesn’t know which L3Out EPG the subnet belongs to either. This scope is also the Longest Prefix Match.

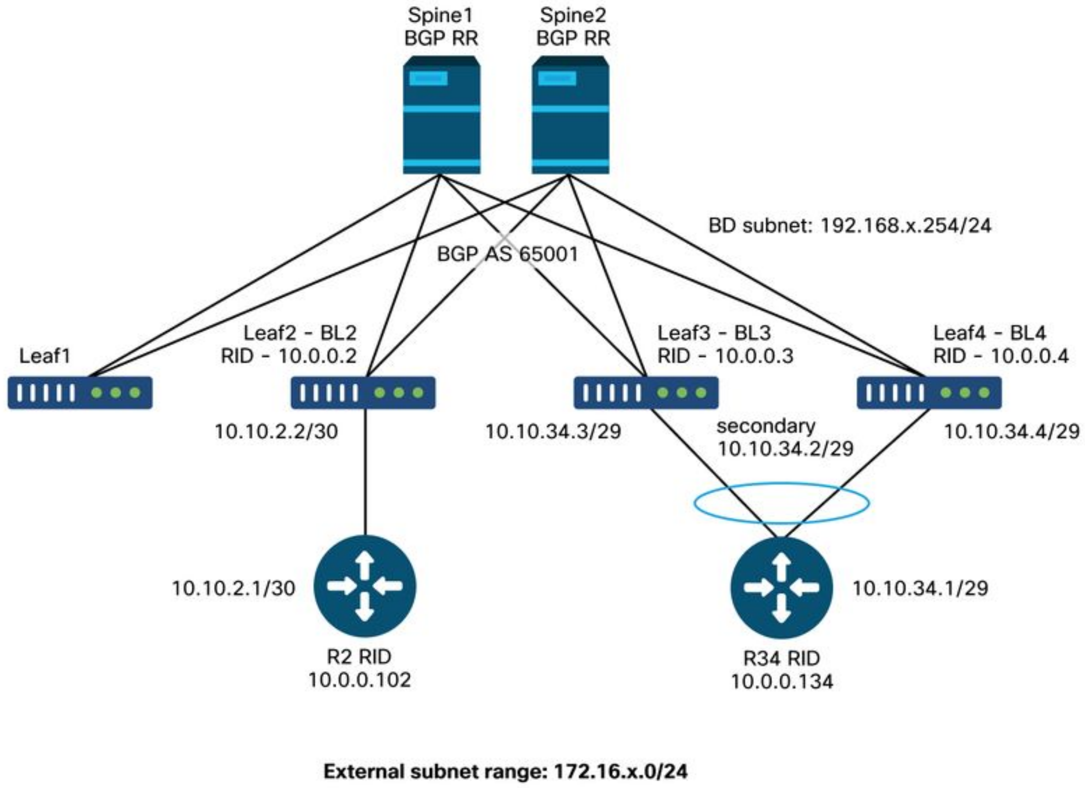
'Aggregate' options:

- **Aggregate Export:** This option can be used only for 0.0.0.0/0 with 'Export Route Control Subnet'. When both 'Export Route Control Subnet' and 'Aggregate Export' are enabled for 0.0.0.0/0; it creates a prefix-list with '0.0.0.0/0 le 32' which matches any subnets. Hence, this option can be used when a L3Out needs to advertise (export) any routes to the outside. When more granular aggregation is required, Route Map/Profile with an explicit prefix-list can be used.
- **Aggregate Import:** This option can be used only for 0.0.0.0/0 with 'Import Route Control Subnet'. When both 'Import Route Control Subnet' and 'Aggregate Import' are enabled for 0.0.0.0/0, it creates a prefix-list with '0.0.0.0/0 le 32' which matches any subnets. Hence, this option can be used when a L3Out needs to learn (import) any routes from outside. However, the same thing can be accomplished by disabling 'Import Route Control Enforcement' which is the default. When more granular aggregation is required, Route Map/Profile with an explicit prefix-list can be used.
- **Aggregate Shared Routes:** This option can be used for any subnets with 'Shared Route Control Subnet'. When both 'Shared Route Control Subnet' and 'Aggregate Shared Routes' are enabled for 10.0.0.0/8 as an example, it creates a prefix-list with '10.0.0.0/8 le 32' which matches 10.0.0.0/8, 10.1.0.0/16 and so on.

L3Out topology used in this section

The following topology will be used in this section:

L3Out topology



Adjacencies

This section explains how to troubleshoot and verify routing protocol adjacencies on L3Out interfaces.

Below are a few parameters to check that will be applicable for all ACI external routing protocols:

- **Router ID:** In ACI, each L3Out needs to use the same Router ID in the same VRF on the same leaf even if routing protocols are different. Also, only one of those L3Outs on the same leaf can create a loopback with the Router ID, which is typically BGP.
- **MTU:** Although the hard requirement of MTU is only for OSPF adjacency, it is recommended to match MTU for any routing protocols to ensure any jumbo packets used for route exchange/updates can be transmitted without fragmentation, as most of control plane frames will be sent with the DF (don't fragment) bit set, which will drop the frame if its size exceeds the maximum MTU of the interface.
- **MP-BGP Router Reflector:** Without this, the BGP process will not start on leaf nodes. Although, this is not required for OSPF or EIGRP just to establish a neighbor, it is still required for BLs to distribute external routes to other leaf nodes.
- **Faults:** Always be sure to check faults during and after configuration is complete.

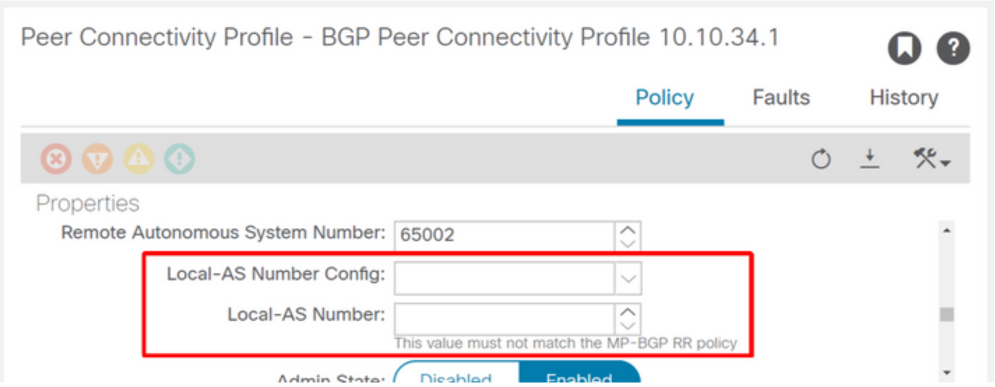
BGP

This section uses an example of an eBGP peering between the loopback on BL3, BL4, and R34 from the topology in the Overview section. The BGP AS on R34 is 65002.

Verify the following criteria when establishing a BGP adjacency.

- Local BGP AS number (ACI BL side).

Peer Connectivity Profile — Local-AS



Peer Connectivity Profile - BGP Peer Connectivity Profile 10.10.34.1

Properties

Remote Autonomous System Number: 65002

Local-AS Number Config:

Local-AS Number:

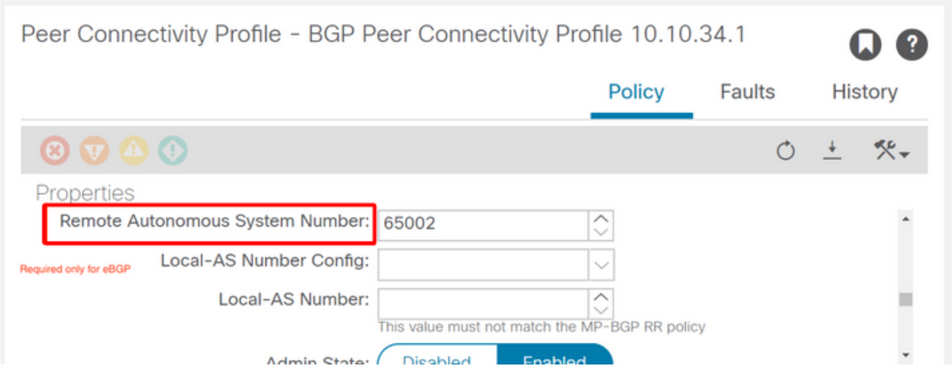
This value must not match the MP-BGP RR policy

Admin State: Disabled Enabled

The BGP AS number of a user L3Out will automatically be the same as the BGP AS for the infra-MP-BGP that is configured in the BGP Route Reflector policy. The 'Local AS' configuration in the BGP Peer Connectivity Profile is not required unless one needs to disguise the ACI BGP AS to the outside world. This means external routers should point to the BGP AS configured in the BGP Route Reflector.

NOTE — The scenario where Local AS configuration is required is the same as the standalone NX-OS 'local-as' command.

- Remote BGP AS number (external side) **Peer Connectivity Profile — Remote AS**



Peer Connectivity Profile - BGP Peer Connectivity Profile 10.10.34.1

Properties

Remote Autonomous System Number: 65002

Required only for eBGP

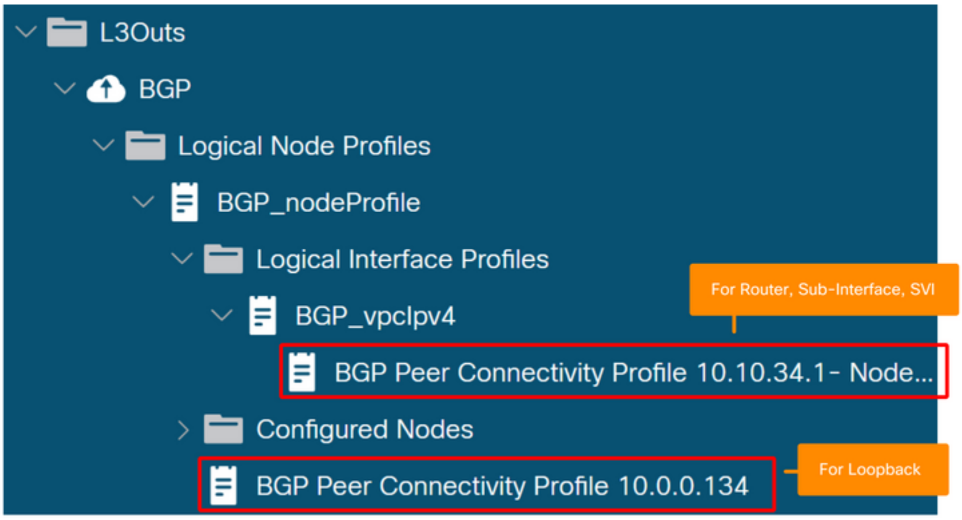
Local-AS Number Config:

Local-AS Number:

This value must not match the MP-BGP RR policy

Admin State: Disabled Enabled

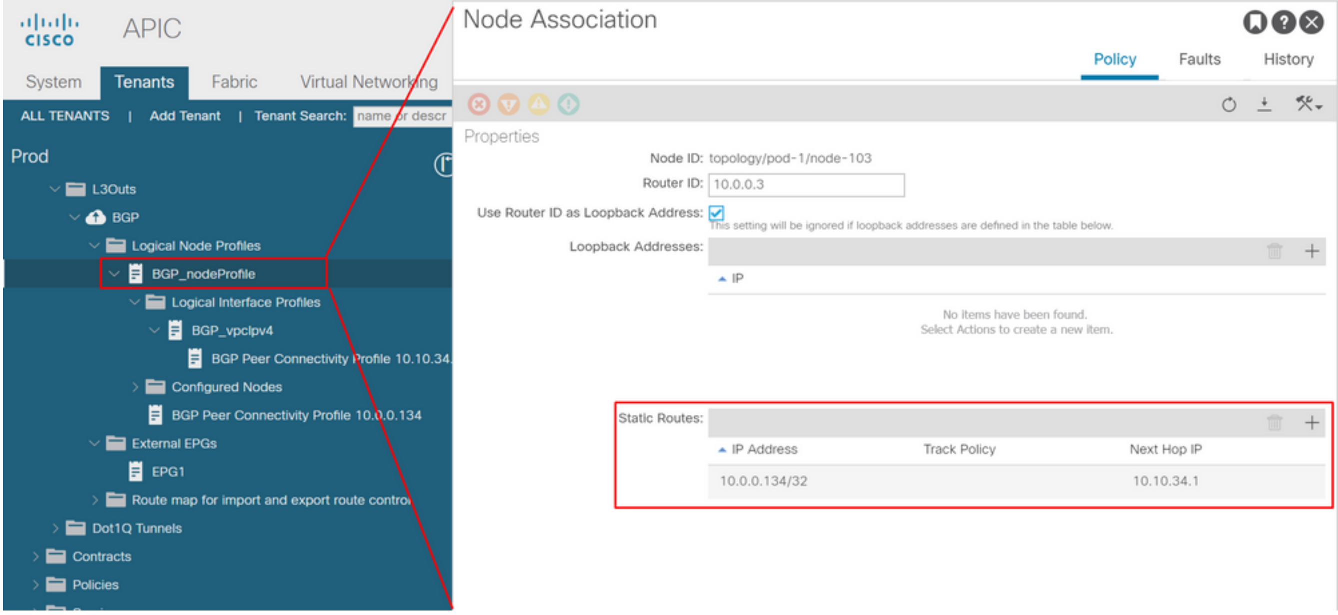
The Remote BGP AS number is required only for eBGP where the neighbor's BGP AS is different from ACI BGP AS.Source IP for BGP peer session.**L3Out — BGP Peer Connectivity Profile**


L3Outs
BGP
Logical Node Profiles
BGP_nodeProfile
Logical Interface Profiles
BGP_vpcIpv4
BGP Peer Connectivity Profile 10.10.34.1- Node...
Configured Nodes
BGP Peer Connectivity Profile 10.0.0.134

For Router, Sub-Interface, SVI

For Loopback

ACI supports sourcing a BGP session from the loopback interface on top of a typical ACI L3Out interface type (routed, sub-interface, SVI).When a BGP session needs to be sourced from a loopback, configure the BGP Peer Connectivity Profile under the Logical **Node** Profile.When the BGP session needs to be sourced from a routed/sub-interface/SVI, configure the BGP Peer Connectivity Profile under the Logical **Interface** Profile.BGP peer IP reachability.**Logical Node Profile — Node Association**


APIC
System
Tenants
Fabric
Virtual Networking
ALL TENANTS | Add Tenant | Tenant Search: name or descr
Prod
L3Outs
BGP
Logical Node Profiles
BGP_nodeProfile
Logical Interface Profiles
BGP_vpcIpv4
BGP Peer Connectivity Profile 10.10.34.
Configured Nodes
BGP Peer Connectivity Profile 10.0.0.134
External EPGs
EPG1
Route map for import and export route control
Dot1Q Tunnels
Contracts
Policies
Policy
Faults
History

Node Association

Properties

Node ID: topology/pod-1/node-103

Router ID: 10.0.0.3

Use Router ID as Loopback Address: ☑

This setting will be ignored if loopback addresses are defined in the table below.

Loopback Addresses:

IP

No items have been found.
Select Actions to create a new item.

Static Routes:

IP Address	Track Policy	Next Hop IP
10.0.0.134/32		10.10.34.1

When the BGP peer IPs are loopbacks, make sure the BL and the external router have reachability to the peer's IP address. Static routes or OSPF can be used to gain reachability to the peer IPs.**BGP CLI Verification (eBGP with loopback example)**The CLI outputs for the following steps are collected from BL3 in the topology from the Overview section.**1. Check if the BGP session is established**'State/PfxRcd' in the following CLI output means the BGP session is established.

```
f2-leaf3# show bgp ipv4 unicast summary vrf Prod:VRF1
BGP summary information for VRF Prod:VRF1, address family IPv4 Unicast
BGP router identifier 10.0.0.3, local AS number 65001

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
```

```
10.0.0.134      4 65002      10      10       10    0    0 00:06:39 0
```

If the 'State/PfxRcd' shows Idle or Active, BGP packets are not being exchanged with the peer yet. In such a scenario, check the following and move on to the next step.

- Ensure the external router is pointing to the ACI BGP AS correctly (local AS number 65001).
- Ensure the ACI BGP Peer Connectivity Profile is specifying the correct neighbor IP from which the external router is sourcing the BGP session (10.0.0.134).
- Ensure the ACI BGP Peer Connectivity Profile is specifying the correct neighbor AS of the external router (Remote Autonomous System Number in GUI which shows up in CLI as AS 65002).

2. Check BGP Neighbor details (BGP Peer Connectivity Profile)

The following command shows the parameters that are key for BGP neighbor establishment.

- Neighbor IP: 10.0.0.134.
- Neighbor BGP AS: remote AS 65002.
- Source IP: Using loopback3 as update source.
- eBGP multi-hop: External BGP peer might be upto 2 hops away.

```
f2-leaf3# show bgp ipv4 unicast neighbors vrf Prod:VRF1
BGP neighbor is 10.0.0.134,  remote AS 65002, ebgp link,  Peer index 1
 BGP version 4, remote router ID 10.0.0.134
 BGP state = Established, up for 00:11:18
 Using loopback3 as update source for this peer
 External BGP peer might be upto 2 hops away

...

  For address family: IPv4 Unicast
...
Inbound route-map configured is permit-all, handle obtained
Outbound route-map configured is exp-l3out-BGP-peer-3047424, handle obtained
Last End-of-RIB received 00:00:01 after session start
Local host: 10.0.0.3, Local port: 34873
 Foreign host: 10.0.0.134, Foreign port: 179
 fd = 64
```

Once the BGP peer is established correctly, the 'Local host' and 'Foreign host' appear at the bottom of the output.

3. Check IP reachability for the BGP peer

```
f2-leaf3# show ip route vrf Prod:VRF1
10.0.0.3/32, ubest/mbest: 2/0, attached, direct
   *via 10.0.0.3, lo3, [0/0], 02:41:46, local, local
   *via 10.0.0.3, lo3, [0/0], 02:41:46, direct
10.0.0.134/32, ubest/mbest: 1/0
   *via 10.10.34.1, vlan27, [1/0], 02:41:46, static    <--- neighbor IP reachability via static
route
10.10.34.0/29, ubest/mbest: 2/0, attached, direct
   *via 10.10.34.3, vlan27, [0/0], 02:41:46, direct
   *via 10.10.34.2, vlan27, [0/0], 02:41:46, direct
10.10.34.2/32, ubest/mbest: 1/0, attached
```

```
   *via 10.10.34.2, vlan27, [0/0], 02:41:46, local, local
10.10.34.3/32, ubest/mbest: 1/0, attached
   *via 10.10.34.3, vlan27, [0/0], 02:41:46, local, local
```

Ensure ping to the neighbor IP works from ACI BGP's source IP.

```
f2-leaf3# iping 10.0.0.134 -V Prod:VRF1 -S 10.0.0.3
PING 10.0.0.134 (10.0.0.134) from 10.0.0.3: 56 data bytes
64 bytes from 10.0.0.134: icmp_seq=0 ttl=255 time=0.571 ms
64 bytes from 10.0.0.134: icmp_seq=1 ttl=255 time=0.662 ms
```

4. Check the same thing on the external router

The following is an example of configuration on the external router (standalone NX-OS).

```
router bgp 65002
 vrf f2-bgp
   router-id 10.0.0.134
   neighbor 10.0.0.3
     remote-as 65001
     update-source loopback134
     ebgp-multihop 2
     address-family ipv4 unicast
   neighbor 10.0.0.4
     remote-as 65001
     update-source loopback134
     ebgp-multihop 2
     address-family ipv4 unicast

interface loopback134
 vrf member f2-bgp
 ip address 10.0.0.134/32

interface Vlan2501
 no shutdown
 vrf member f2-bgp
 ip address 10.10.34.1/29

vrf context f2-bgp
 ip route 10.0.0.0/29 10.10.34.2
```

5. Additional Step — tcpdump

On ACI leaf nodes, the tcpdump tool can sniff the 'kpm_inb' CPU interface to confirm if the protocol packets reached the leaf CPU. Use L4 port 179 (BGP) as a filter.

```
f2-leaf3# tcpdump -ni kpm_inb port 179
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on kpm_inb, link-type EN10MB (Ethernet), capture size 65535 bytes
20:36:58.292903 IP 10.0.0.134.179 > 10.0.0.3.34873: Flags [P.], seq 3775831990:3775832009, ack
807595300, win 3650, length 19: BGP, length: 19
20:36:58.292962 IP 10.0.0.3.34873 > 10.0.0.134.179: Flags [.], ack 19, win 6945, length 0
20:36:58.430418 IP 10.0.0.3.34873 > 10.0.0.134.179: Flags [P.], seq 1:20, ack 19, win 6945,
length 19: BGP, length: 19
20:36:58.430534 IP 10.0.0.134.179 > 10.0.0.3.34873: Flags [.], ack 20, win 3650, length 0
```

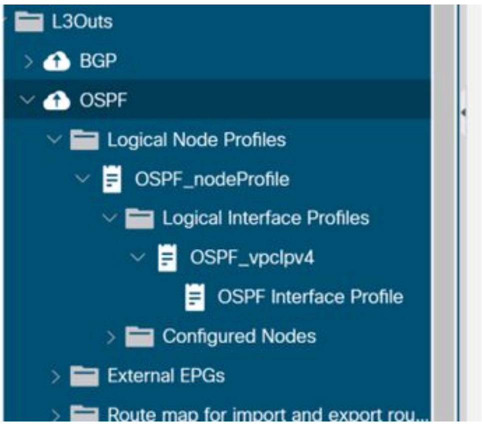
OSPF

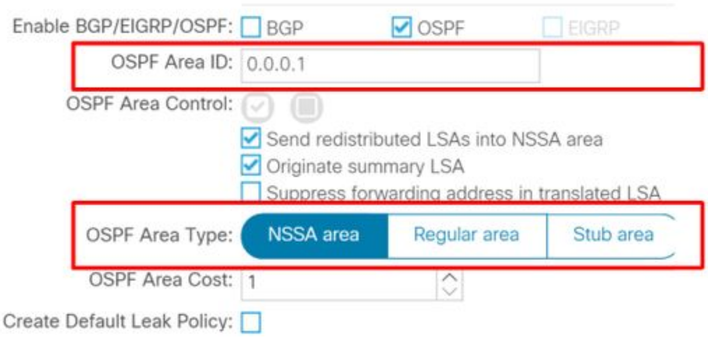
This section uses an example of OSPF neighborships between BL3, BL4, and R34 from the topology in Overview section with OSPF AreaID 1 (NSSA).

The following are the common criteria to check for OSPF adjacency establishment.

- OSPF Area ID and Type

L3Out — OSPF Interface Profile — Area ID and Type





Enable BGP/EIGRP/OSPF: □ BGP ☑ OSPF □ EIGRP

OSPF Area ID: 0.0.0.1

OSPF Area Control:

☑ Send redistributed LSAs into NSSA area

☑ Originate summary LSA

□ Suppress forwarding address in translated LSA

OSPF Area Type: [✓ NSSA area] Regular area Stub area

OSPF Area Cost: 1

Create Default Leak Policy: □

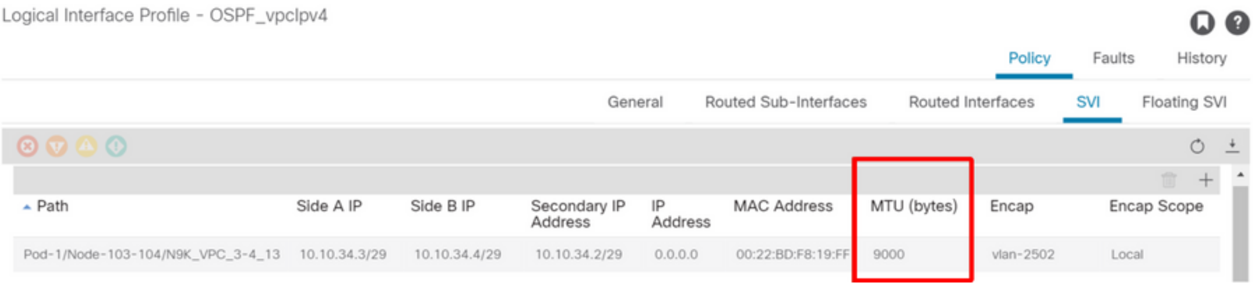
Just like any routing device, OSPF Area ID and Type need to match on both neighbors. Some ACI specific limitations on OSPF Area ID configurations include:

- One L3Out can have only one OSPF Area ID.
- Two L3Outs can use the same OSPF Area ID in the same VRF only when they are on two different leaf nodes.

Although the OSPF ID does not need to be backbone 0, in the case of Transit Routing it is required between two OSPF L3Outs on the same leaf; one of them must use OSPF Area 0 because any route exchange between OSPF areas must be through OSPF Area 0.

- MTU

Logical Interface Profile — SVI

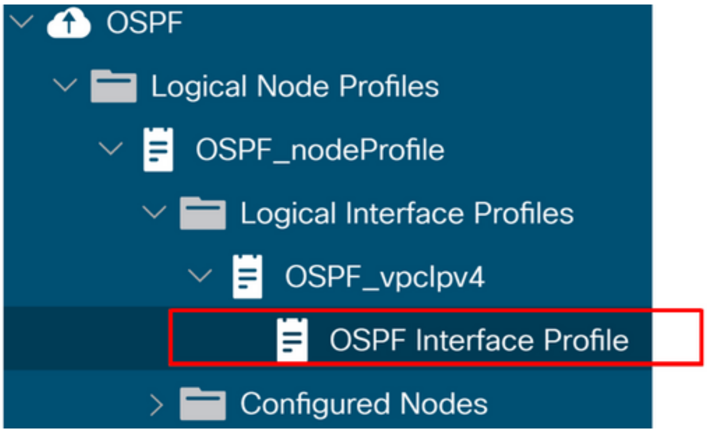


Path	Side A IP	Side B IP	Secondary IP Address	IP Address	MAC Address	MTU (bytes)	Encap	Encap Scope
Pod-1/Node-103-104/N9K_VPC_3-4_13	10.10.34.3/29	10.10.34.4/29	10.10.34.2/29	0.0.0.0	00:22:BD:F8:19:FF	9000	vlan-2502	Local

The default MTU on ACI is 9000 bytes, instead of 1500 bytes, which is typically the default used on traditional routing devices. Ensure the MTU matches with the external device. When OSPF neighbor establishment fails due to MTU, it gets stuck at EXCHANGE/DROTHER.

- IP Subnet mask. OSPF requires the neighbor IP to use the same subnet mask.
- OSPF Interface Profile.

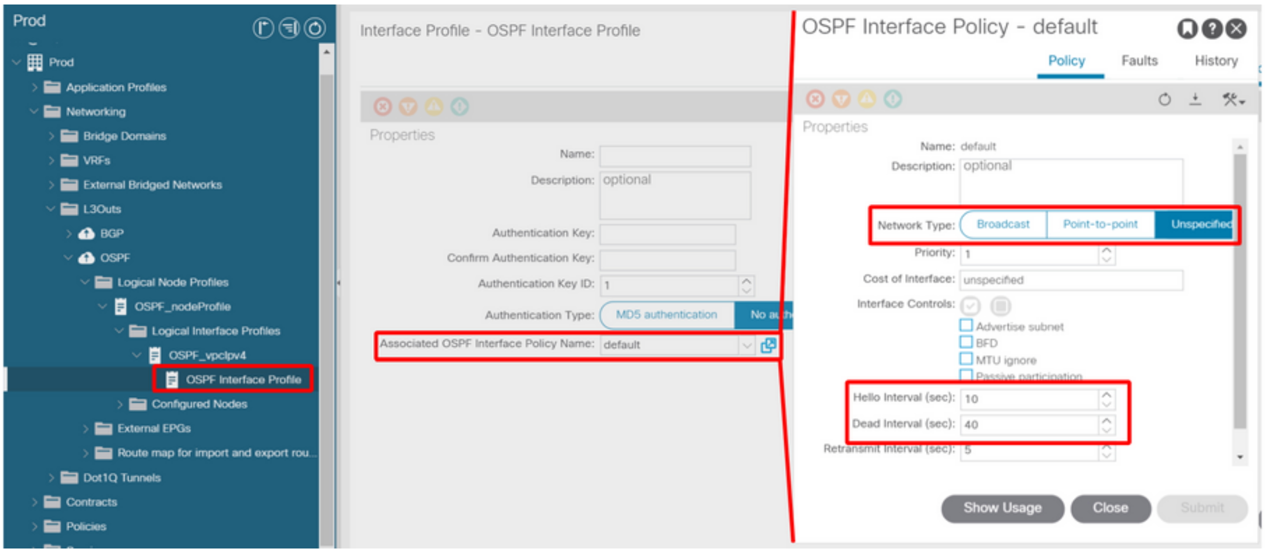
OSPF Interface Profile



This is equivalent to 'ip router ospf <tag> area <area id>' on a standalone NX-OS config to enable OSPF on the interface. Without this, the leaf interfaces will not join OSPF.

- OSPF Hello / Dead Timer, Network Type

OSPF Interface Profile — Hello / Dead timer and Network Type



Interface Profile - OSPF Interface Profile

Properties

Name:

Description: optional

Authentication Key:

Confirm Authentication Key:

Authentication Key ID: 1

Authentication Type: MD5 authentication | No au...

Associated OSPF Interface Policy Name: default

OSPF Interface Policy - default

Policy | Faults | History

Properties

Name: default

Description: optional

Network Type: Broadcast | Point-to-point | [✓ Unspecified]

Priority: 1

Cost of Interface: unspecified

Interface Controls:

□ Advertise subnet

□ BFD

□ MTU ignore

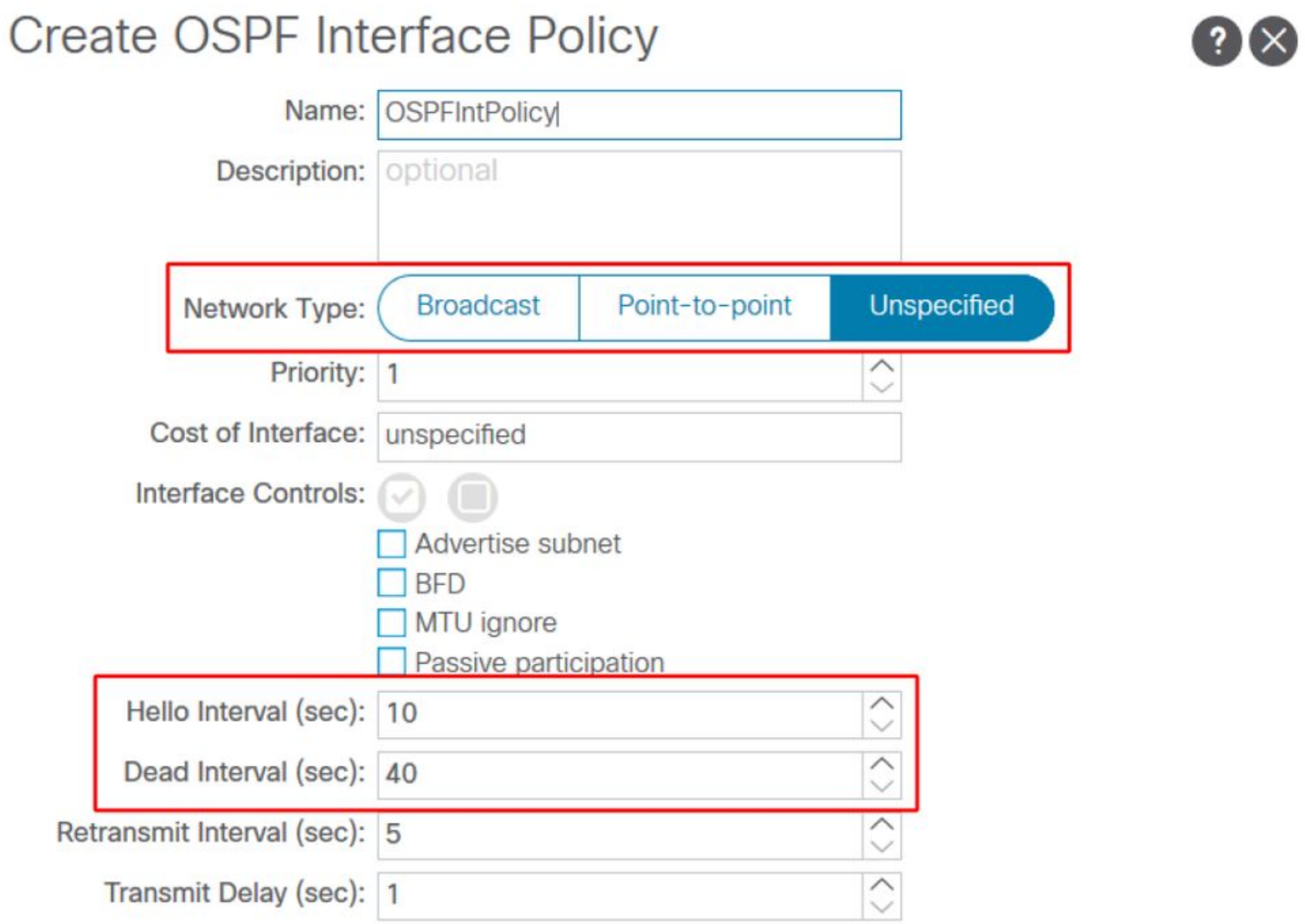
□ Passive participation

Hello Interval (sec): 10

Dead Interval (sec): 40

Retransmit Interval (sec): 5

OSPF Interface Policy details



Create OSPF Interface Policy

Name: OSPFIntPolicy

Description: optional

Network Type: Broadcast | Point-to-point | [✓ Unspecified]

Priority: 1

Cost of Interface: unspecified

Interface Controls:

☐ Advertise subnet
☐ BFD
☐ MTU ignore
☐ Passive participation

Hello Interval (sec): 10

Dead Interval (sec): 40

Retransmit Interval (sec): 5

Transmit Delay (sec): 1

OSPF requires the Hello and Dead Timers to match on each neighbor device. These are configured in the OSPF Interface Profile.

Ensure the OSPF Interface Network Type matches with the external device. When the external device is using type Point-to-Point, the ACI side needs to explicitly configure Point-to-Point as well. These are also configured in the OSPF Interface Profile.

OSPF CLI verification

The CLI outputs in the following steps are collected from BL3 in the topology from "Overview" section.

1. Check OSPF neighbor status

If the 'State' is 'FULL' in the following CLI, the OSPF neighbor is established correctly. Otherwise, move on to the next step to check parameters.

```
f2-leaf3# show ip ospf neighbors vrf Prod:VRF2
OSPF Process ID default VRF Prod:VRF2
Total number of neighbors: 2
Neighbor ID     Pri State            Up Time  Address         Interface
10.0.0.4          1 FULL/DR          00:47:30 10.10.34.4      Vlan28      <--- neighbor with BL4
10.0.0.134        1 FULL/DROTHER     00:00:21 10.10.34.1      Vlan28      <--- neighbor with R34
```

In ACI, BLs will form OSPF neighborships with each other on top of external routers when using the same VLAN ID with an SVI. This is because ACI has an internal flooding domain called L3Out

BD (or External BD) for each VLAN ID in the L3Out SVIs. Note that the VLAN ID 28 is an internal VLAN called PI-VLAN (Platform-Independent VLAN) instead of the actual VLAN (Access Encap VLAN) used on wire. Use the following command to verify the access encap VLAN ('vlan-2502').

```
f2-leaf3# show vlan id 28 extended
 VLAN Name                             Encap            Ports
 ---- -------------------------------- ---------------- ------------------------
 28   Prod:VRF2:l3out-OSPF:vlan-2502   vxlan-14942176,  Eth1/13, Po1
                                       vlan-2502
```

One could get the same output via access encap VLAN ID as well.

```
f2-leaf3# show vlan encap-id 2502 extended
 VLAN Name                             Encap            Ports
 ---- -------------------------------- ---------------- ------------------------
 28   Prod:VRF2:l3out-OSPF:vlan-2502   vxlan-14942176,  Eth1/13, Po1
                                       vlan-2502
```

2. Check OSPF area

Ensure the OSPF area ID and Type is identical to the neighbors. If the OSPF interface profile is missing, the interface will not join OSPF and it will not show up in the OSPF CLI output.

```
f2-leaf3# show ip ospf interface brief vrf Prod:VRF2
OSPF Process ID default VRF Prod:VRF2
Total number of interface: 1
Interface               ID     Area            Cost   State    Neighbors Status
Vlan28                  94     0.0.0.1         4      BDR      2         up
f2-leaf3# show ip ospf vrf Prod:VRF2
Routing Process default with ID 10.0.0.3 VRF Prod:VRF2
...
  Area (0.0.0.1)
       Area has existed for 00:59:14
       Interfaces in this area: 1 Active interfaces: 1
       Passive interfaces: 0  Loopback interfaces: 0
       This area is a NSSA area
       Perform type-7/type-5 LSA translation
       SPF calculation has run 10 times
        Last SPF ran for 0.001175s
       Area ranges are
       Area-filter in 'exp-ctx-proto-3112960'
       Area-filter out 'permit-all'
       Number of LSAs: 4, checksum sum 0x0
```

3. Check OSPF interface details

Ensure interface level parameters meet the requirements for OSPF neighbor establishment such as IP subnet, Network Type, Hello/Dead Timer. Please note the VLAN ID to specify the SVI is PI-VLAN (vlan28)

```
f2-leaf3# show ip ospf interface vrf Prod:VRF2
Vlan28 is up, line protocol is up
   IP address 10.10.34.3/29, Process ID default VRF Prod:VRF2, area 0.0.0.1
   Enabled by interface configuration
   State BDR, Network type BROADCAST, cost 4
   Index 94, Transmit delay 1 sec, Router Priority 1
   Designated Router ID: 10.0.0.4, address: 10.10.34.4
   Backup Designated Router ID: 10.0.0.3, address: 10.10.34.3
   2 Neighbors, flooding to 2, adjacent with 2
   Timer intervals: Hello 10, Dead 40, Wait 40, Retransmit 5
     Hello timer due in 0.000000
   No authentication
   Number of opaque link LSAs: 0, checksum sum 0

f2-leaf3# show interface vlan28
Vlan28 is up, line protocol is up, autostate disabled
  Hardware EtherSVI, address is  0022.bdf8.19ff
  Internet Address is 10.10.34.3/29
  MTU 9000 bytes, BW 10000000 Kbit, DLY 1 usec
```

4. Check IP reachability to the neighbor

Although OSPF Hello packets are Link Local Multicast packets, OSPF DBD packets required for the first OSPF LSDB exchange are unicast. Therefore, unicast reachability also needs to be verified for the OSPF neighborship establishment.

```
f2-leaf3# iping 10.10.34.1 -V Prod:VRF2
PING 10.10.34.1 (10.10.34.1) from 10.10.34.3: 56 data bytes
64 bytes from 10.10.34.1: icmp_seq=0 ttl=255 time=0.66 ms
64 bytes from 10.10.34.1: icmp_seq=1 ttl=255 time=0.653 ms
```

5. Check the same on the external router

The following are examples of configurations on the external router (standalone NX-OS)

```
router ospf 1
    vrf f2-ospf
    router-id 10.0.0.134
    area 0.0.0.1 nssa

interface Vlan2502
    no shutdown
    mtu 9000
    vrf member f2-ospf
    ip address 10.10.34.1/29
    ip router ospf 1 area 0.0.0.1
```

Make sure to verify the MTU as well on the physical interface.

6. Additional step — tcpdump

On ACI leaf nodes, the user can perform tcpdump on the 'kpm_inb' CPU interface to verify if the protocol packets have reached the leaf CPU. Although there are multiple filters for OSPF, the IP Protocol Number is the most comprehensive filter.

- IP Protocol Number: proto 89 (IPv4) or ip6 proto 0x59 (IPv6)
- IP address of the neighbor: host <ip>
- OSPF Link Local Mcast IP: host 224.0.0.5 or host 224.0.0.6

```
f2-leaf3# tcpdump -ni kpm_inb proto 89
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on kpm_inb, link-type EN10MB (Ethernet), capture size 65535 bytes
22:28:38.231356 IP 10.10.34.4 > 224.0.0.5: OSPFv2, Hello, length 52
22:28:42.673810 IP 10.10.34.3 > 224.0.0.5: OSPFv2, Hello, length 52
22:28:44.767616 IP 10.10.34.1 > 224.0.0.5: OSPFv2, Hello, length 52
22:28:44.769092 IP 10.10.34.3 > 10.10.34.1: OSPFv2, Database Description, length 32
22:28:44.769803 IP 10.10.34.1 > 10.10.34.3: OSPFv2, Database Description, length 32
22:28:44.775376 IP 10.10.34.3 > 10.10.34.1: OSPFv2, Database Description, length 112
22:28:44.780959 IP 10.10.34.1 > 10.10.34.3: OSPFv2, LS-Request, length 36
22:28:44.781376 IP 10.10.34.3 > 10.10.34.1: OSPFv2, LS-Update, length 64
22:28:44.790931 IP 10.10.34.1 > 224.0.0.6: OSPFv2, LS-Update, length 64
```

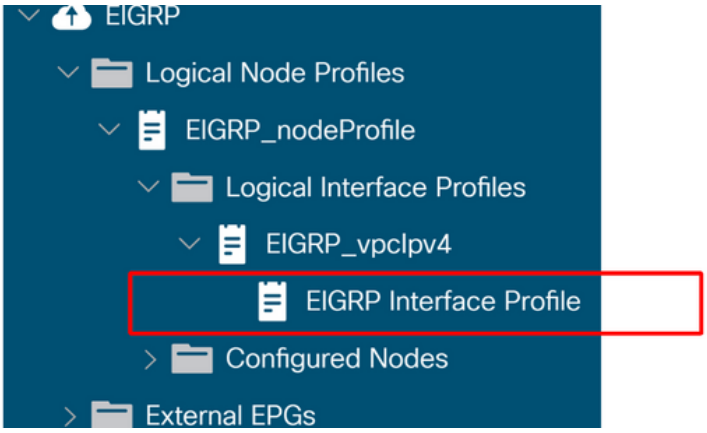
EIGRP

This section uses an example of EIGRP neighborship between BL3, BL4 and R34 from the topology in "Overview" section with EIGRP AS 10.

The following are the common criteria for EIGRP adjacency establishment.

- EIGRP AS: a L3Out is assigned one EIGRP AS. This needs to match with the external device.
- EIGRP Interface Profile.

EIGRP Interface Profile



This is equivalent to the 'ip router eigrp <as>' configuration on a standalone NX-OS device. Without this, the leaf interfaces will not join EIGRP.

- MTU

Although this does not have to match to simply establish the EIGRP neighborship, the EIGRP topology exchange packets may become larger than the maximum MTU allowed on the interfaces between the peers, and since these packets are not allowed to be fragmented, they are dropped and as a result the EIGRP neighborship will flap.

EIGRP CLI Verification

The CLI outputs in the following steps are collected from BL3 in the topology from the "Overview" section.

1. Check EIGRP neighbor status

```
f2-leaf3# show ip eigrp neighbors vrf Prod:VRF3
EIGRP neighbors for process 10 VRF Prod:VRF3
H   Address                 Interface       Hold  Uptime  SRTT   RTO  Q  Seq
                                           (sec)         (ms)       Cnt Num
0   10.10.34.4              vlan29          14   00:12:58  1    50    0   6    <--- neighbor
with BL4
1   10.10.34.1              vlan29          13   00:08:44  2    50    0   4    <--- neighbor
with R34
```

In ACI, BLs will form an EIGRP neighborship with each other on top of external routers when they use the same VLAN ID with SVI. This is because an ACI has an internal flooding domain called L3Out BD (or External BD) for each VLAN ID in L3Out SVIs.

Please note that the VLAN ID 29 is an internal VLAN called PI-VLAN (Platform-Independent VLAN) instead of the actual VLAN (Access Encap VLAN) used on wire. Use the following command to verify the access encap VLAN (vlan-2503).

```
f2-leaf3# show vlan id 29 extended
 VLAN Name                             Encap            Ports
 ---- -------------------------------- ---------------- ------------------------
 29   Prod:VRF3:l3out-EIGRP:vlan-2503  vxlan-15237052,  Eth1/13, Po1
                                       vlan-2503
```

One could get the same output via access encap VLAN ID as well.

```
f2-leaf3# show vlan encap-id 2503 extended
 VLAN Name                             Encap            Ports
 ---- -------------------------------- ---------------- ------------------------
 29   Prod:VRF3:l3out-EIGRP:vlan-2503  vxlan-15237052,  Eth1/13, Po1
                                       vlan-2503
```

2. Check EIGRP interface details

Ensure EIGRP is running on the expected interface. If not, check Logical Interface Profile and EIGRP Interface Profile.

```
f2-leaf3# show ip eigrp interfaces vrf Prod:VRF3
EIGRP interfaces for process 10 VRF Prod:VRF3
                        Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface        Peers  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
vlan29             2        0/0         1       0/0          50             0
  Hello interval is 5 sec
  Holdtime interval is 15 sec
  Next xmit serial: 0
  Un/reliable mcasts: 0/2     Un/reliable ucasts: 5/10
  Mcast exceptions: 0    CR packets: 0    ACKs suppressed: 2
  Retransmissions sent: 2    Out-of-sequence rcvd: 0
  Classic/wide metric peers: 2/0
```

```
f2-leaf3# show int vlan 29
Vlan29 is up, line protocol is up, autostate disabled
  Hardware EtherSVI, address is  0022.bdf8.19ff
  Internet Address is 10.10.34.3/29
  MTU 9000 bytes, BW 10000000 Kbit, DLY 1 usec
```

3. Check the same on the external router

The following the example config on the external router (standalone NX-OS).

```
router eigrp 10
 vrf f2-eigrp

interface Vlan2503
 no shutdown
 vrf member f2-eigrp
 ip address 10.10.34.1/29
 ip router eigrp 10
```

4. Additional step — tcpdump

On ACI leaf nodes, the user can perform tcpdump on the 'kpm_inb' CPU interface to confirm if the protocol packets reached the leaf's CPU. Use IP protocol number 88 (EIGRP) as a filter.

```
f2-leaf3# tcpdump -ni kpm_inb proto 88
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on kpm_inb, link-type EN10MB (Ethernet), capture size 65535 bytes
23:29:43.725676 IP 10.10.34.3 > 224.0.0.10: EIGRP Hello, length: 40
23:29:43.726271 IP 10.10.34.4 > 224.0.0.10: EIGRP Hello, length: 40
23:29:43.728178 IP 10.10.34.1 > 224.0.0.10: EIGRP Hello, length: 40
23:29:45.729114 IP 10.10.34.1 > 10.10.34.3: EIGRP Update, length: 20
23:29:48.316895 IP 10.10.34.3 > 224.0.0.10: EIGRP Hello, length: 40
```

Route advertisement

This section focusses on the verification and troubleshooting of route advertisement in ACI. Specifically, it looks at examples involving:

- Bridge Domains Subnet Advertisement.
- Transit Route Advertisement.
- Import and Export Route Control.

This section discusses route-leaking as it pertains to shared L3Outs in later sections.

Bridge domain route advertisement workflow

Before looking at common troubleshooting the user should familiarize themselves with how Bridge Domain advertisement is supposed to work.

BD advertisement, when the BD and L3Out are in the same VRF, involves:

- Having a contract relationship between the L3Out and the internal EPG.
- Associating the L3Out to the Bridge Domain.
- Selecting 'Advertise Externally' on the BD subnet.

In addition, it is also possible to control Bridge Domain advertisement using export route-profiles which prevent the need to associate the L3Out. However, 'Advertise Externally' should still be selected. This is a less common use-case so it won't be discussed here.

The contract relationship between the L3Out and the EPG is required in order to cause the BD pervasive static route to get pushed to the BL. The actual route-advertisement is handled via redistribution of the static route into the external protocol. Lastly, the redistribution route-maps will only be installed within the L3Outs that are associated to the BD. In this way the route isn't advertised out all L3Outs.

In this case, the BD subnet is 192.168.1.0/24 and it should be advertised via OSPF L3Out.

Before applying the contract between the L3Out and internal EPG

```
leaf103# show ip route 192.168.1.0/24 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF
Route not found
```

Notice that the BD route isn't present yet on the BL.

After applying the contract between the L3Out and internal EPG

At this point no other configuration has been done. The L3Out isn't yet associated to the BD and the 'Advertise Externally' flag isn't set.

```
leaf103# show ip route 10.0.1.0/24 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF
192.168.1.0/24, ubest/mbest: 1/0, attached, direct, pervasive
   *via 10.0.120.34%overlay-1, [1/0], 00:00:08, static, tag 4294967294
        recursive next hop: 10.0.120.34/32%overlay-1
```

Notice that the BD subnet route (indicated by the pervasive flag) is now deployed on the BL. Notice, however, that the route is tagged. This tag value is an implicit value assigned to BD routes before being configured with 'Advertise Externally'. All external protocols deny this tag from being redistributed.

After selecting 'Advertise Externally' on the BD Subnet

The L3Out still hasn't been associated to the BD. However, notice that the tag has cleared.

```
leaf103# show ip route 192.168.1.0/24 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF
```

```
192.168.1.0/24, ubest/mbest: 1/0, attached, direct, pervasive *via 10.0.120.34%overlay-1, [1/0],
00:00:06, static recursive next hop: 10.0.120.34/32%overlay-1
```

At this point the route still isn't being advertised externally because there is no route-map and prefix-list that matches this prefix for redistribution into the external protocol. This can be verified with the following commands:

```
leaf103# show ip ospf vrf Prod:Vrf1
Routing Process default with ID 10.0.0.3 VRF Prod:Vrf1
Stateful High Availability enabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
Table-map using route-map exp-ctx-2392068-deny-external-tag
Redistributing External Routes from
  static route-map exp-ctx-st-2392068
  direct route-map exp-ctx-st-2392068
  bgp route-map exp-ctx-proto-2392068
  eigrp route-map exp-ctx-proto-2392068
  coop route-map exp-ctx-st-2392068
```

The BD route is programmed as a static route, so check the static redistribution route-map by running 'show route-map <route-map name>' and then 'show ip prefix-list <name>' on any prefix-lists that are present in the route-map. Do this in the next step.

After associating the L3Out to the BD

As mentioned earlier, this step results in the prefix-list that matches the BD subnet being installed in the static to external protocol redistribution route-map.

```
leaf103# show route-map exp-ctx-st-2392068
route-map exp-ctx-st-2392068, deny, sequence 1
 Match clauses:
   tag: 4294967294
 Set clauses:

...
route-map exp-ctx-st-2392068, permit, sequence 15803
 Match clauses:
   ip address prefix-lists: IPv4-st16390-2392068-exc-int-inferred-export-dst
   ipv6 address prefix-lists: IPv6-deny-all
 Set clauses:
   tag 0
```

Verify the prefix-list:

```
leaf103# show ip prefix-list IPv4-st16390-2392068-exc-int-inferred-export-dst
ip prefix-list IPv4-st16390-2392068-exc-int-inferred-export-dst: 1 entries
  seq 1 permit 192.168.1.1/24
```

The BD subnet is being matched to redistribute into OSPF.

At this point the configuration and verification workflow is complete for the advertisement of the BD subnet out of the L3Out. Past this point, verification would be protocol specific. For instance:

- For EIGRP, verify that the route is being installed in the topology table with 'show ip eigrp topology vrf <name>'
- For OSPF, verify that the route is being installed in the database table as an External LSA

with 'show ip ospf database vrf <name>'

- For BGP, verify that the route is in the BGP RIB with 'show bgp ipv4 unicast vrf <name>'

BGP route advertisement

For BGP, all static routes are implicitly permitted for redistribution. The route-map which matches the BD subnet is applied at the BGP neighbor level.

```
leaf103# show bgp ipv4 unicast neighbor 10.0.0.134 vrf Prod:Vrf1 | grep Outbound
 Outbound route-map configured is exp-l3out-BGP-peer-2392068, handle obtained
```

In the above example, 10.0.0.134 is the BGP neighbor configured within the L3Out.

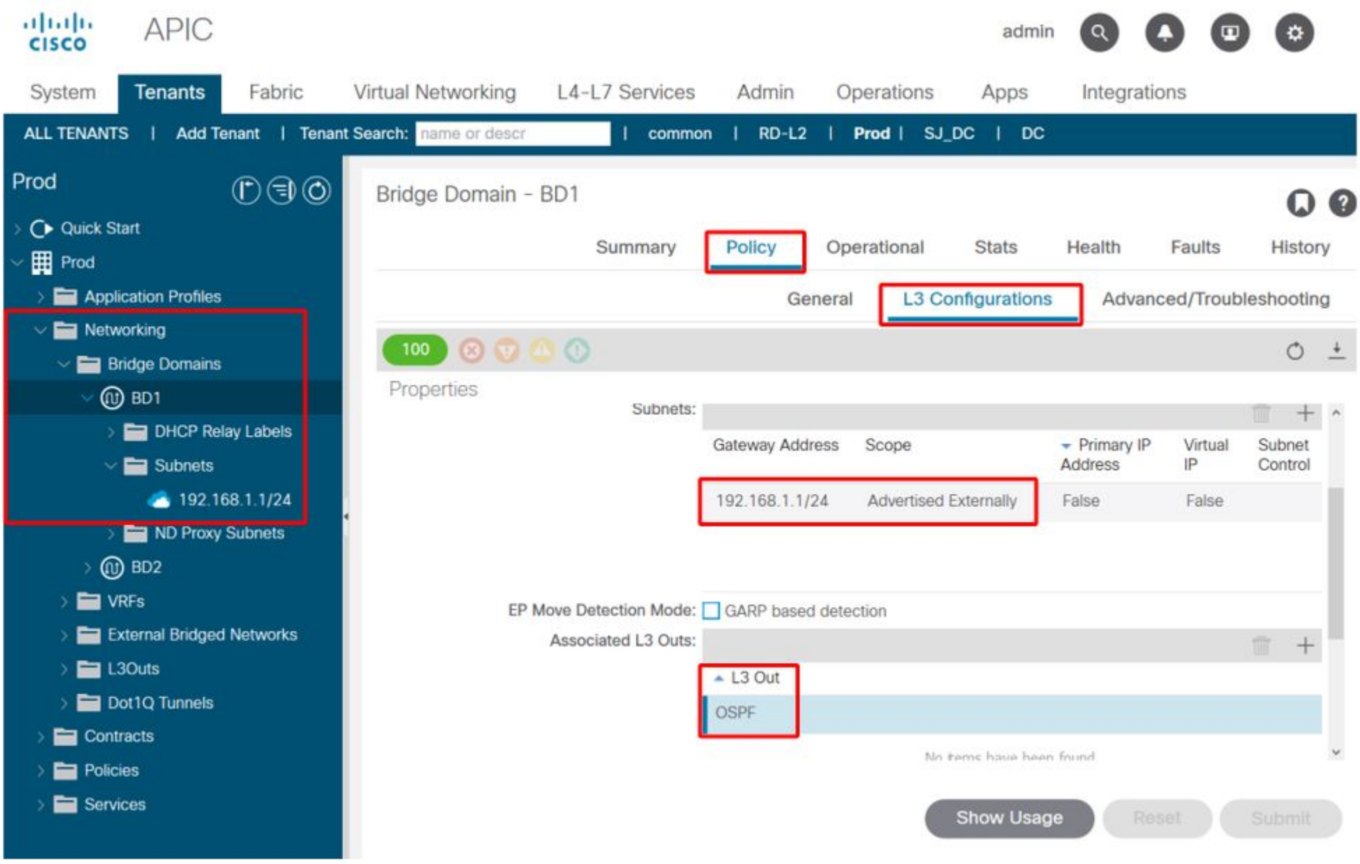
EIGRP route advertisement

Like OSPF, a route-map is used to control Static to EIGRP redistribution. In this way only subnets associated to the L3Out and set to 'Advertise Externally' should be redistributed. This can be verified with this command:

```
leaf103# show ip eigrp vrf Prod:Vrf1
IP-EIGRP AS 100 ID 10.0.0.3 VRF Prod:Vrf1
 Process-tag: default
 Instance Number: 1
 Status: running
 Authentication mode: none
 Authentication key-chain: none
 Metric weights: K1=1 K2=0 K3=1 K4=0 K5=0
 metric version: 32bit
 IP proto: 88 Multicast group: 224.0.0.10
 Int distance: 90 Ext distance: 170
 Max paths: 8
 Active Interval: 3 minute(s)
 Number of EIGRP interfaces: 1 (0 loopbacks)
 Number of EIGRP passive interfaces: 0
 Number of EIGRP peers: 2
 Redistributing:
   static route-map exp-ctx-st-2392068
   ospf-default route-map exp-ctx-proto-2392068
   direct route-map exp-ctx-st-2392068
   coop route-map exp-ctx-st-2392068
   bgp-65001 route-map exp-ctx-proto-2392068
```

The final working BD configuration is shown below.

Bridge Domain L3 Configuration



Subnets:

Gateway Address	Scope	Primary IP Address	Virtual IP	Subnet Control
192.168.1.1/24	Advertised Externally	False	False	

EP Move Detection Mode: □ GARP based detection

Associated L3 Outs:

L3 Out
OSPF

No items have been found

Bridge domain route advertisement troubleshooting scenario

In this case, the typical symptom would normally be that a configured BD subnet is not being advertised out of an L3Out. Follow the previous workflow to understand which component is broken.

Start with the configuration before getting too low-level by verifying the following:

- Is there a contract between the EPG and L3Out?
- Is the L3Out associated to the BD?
- Is the BD subnet set to advertise externally?
- Is the external protocol adjacency up?

Possible Cause: BD Not Deployed

This case would be applicable in a couple of different scenarios, such as:

- The internal EPG is using VMM integration with On Demand option and no VM endpoints have been attached to the port-group for the EPG.
- The internal EPG has been created but no static path bindings have been configured or the interface on which the static path is configured are down.

In both cases, the BD would not be deployed and, as a result, the BD static route would not get pushed to the BL. The solution here is to deploy some active resources within an EPG which is linked to this BD so that the subnet gets deployed.

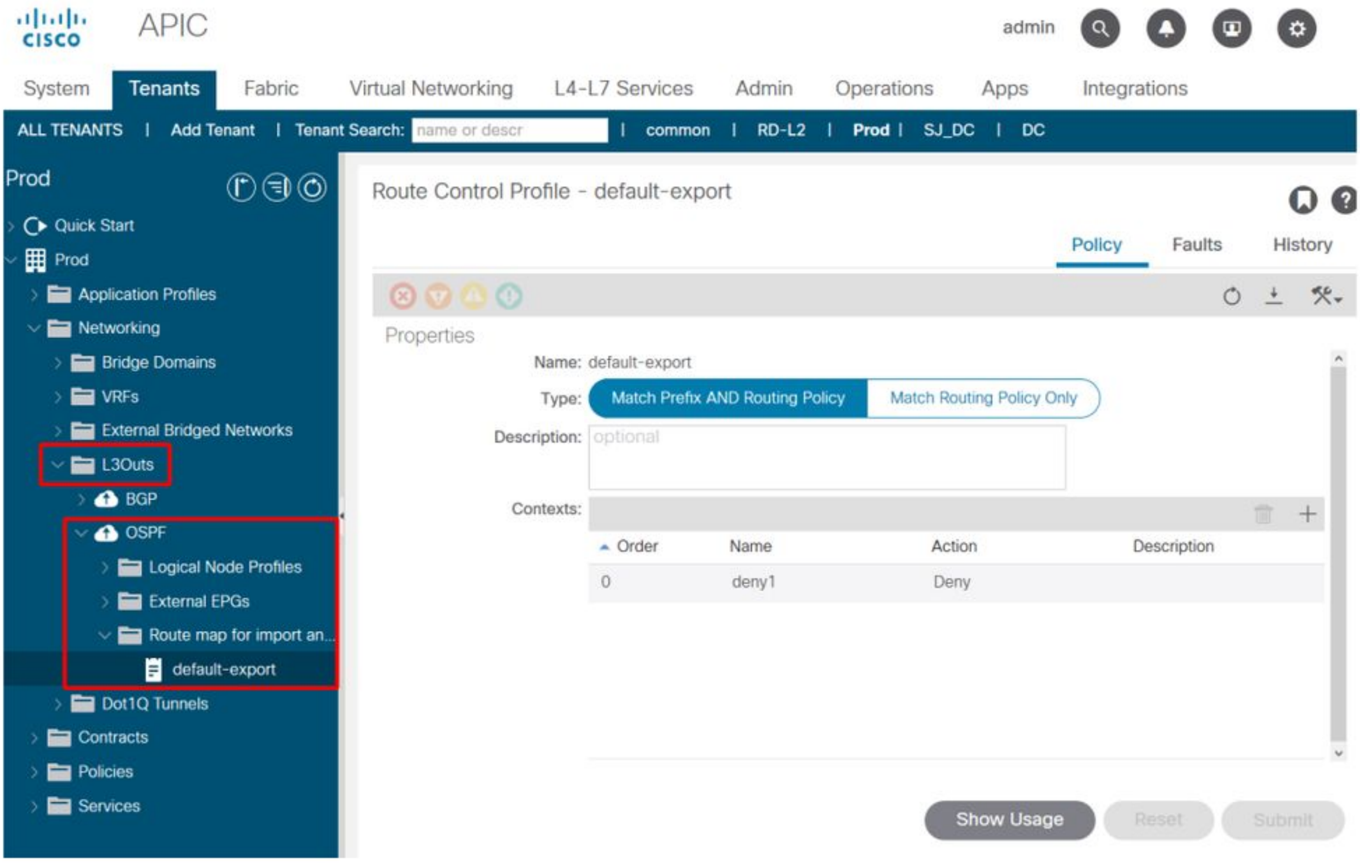
Possible Cause: OSPF L3Out is configured as 'Stub' or 'NSSA' with No Redistribution

When OSPF is used as the L3Out protocol, basic OSPF rules must still be followed. Stub areas do not allow redistributed LSAs but can advertise a default route instead. NSSA areas do allow redistributed paths but 'Send Redistributed LSAs into NSSA Area' must be selected on the L3Out. Or NSSA can also advertise a default route instead by disabling 'Originate Summary LSA' as well which is a typical scenario where 'Send Redistributed LSA's into NSSA Area' would be disabled.

Possible Cause: 'Default-Export' Route-Profile with a 'Deny' Action configured under the L3Out

When route-profiles are configured under an L3Out with the names of 'default-export' or 'default-import' they are implicitly applied to the L3Out. In addition, if the default-export route-profile is set to a deny action and configured as 'Match Prefix and Routing Policy' then BD subnets should be advertised out of this L3Out and would be implicitly denied:

Default-export Deny Route Profile



Route Control Profile - default-export

Name: default-export

Type: Match Prefix AND Routing Policy | Match Routing Policy Only

Description: optional

Contexts:

Order	Name	Action	Description
0	deny1	Deny	

Prefix-matches within the default-export route-profile will not implicitly include BD Subnets if the 'Match Routing Policy Only' option is selected.

External route import workflow

This section discusses how ACI learns external routes through an L3Out and distributes this to internal leaf nodes. It also covers transit and route-leaking use-cases in later sections

As with the previous section, the user should be aware of what happens at a higher-level.

By default, all routes learned through the external protocol are redistributed into the internal fabric BGP process. This is true regardless of what subnets are configured under the external EPG and what flags are selected. There are two examples where this is not true.

- If the 'Route Control Enforcement' option at the top level L3Out policy is set to 'Import'. In this case the route import model would go from a blocklist model (only specify what shouldn't be allowed) to a permitlist model (everything is implicitly denied unless configured otherwise).
- If the external Protocol is EIGRP or OSPF and an Interleak Route-Profile used does not match the external routes.

For an external route to be distributed to an internal leaf the following must happen:

- The route must be learned on the BL from the external router. To be a candidate to redistribute into the fabric MP-BGP process the route must be installed in the routing table rather than just in the protocol RIB.
- The route must be permitted to be redistributed or advertised into the internal BGP process. This should always happen unless import route-control enforcement or an Interleak Route-Profile is used.
- A BGP Route-Reflector Policy must be configured and applied to a Pod Policy Group which is applied to the Pod Profile. If this isn't applied, then the BGP Process will not initialize on the switches.

If the internal EPG/BD is in the same VRF as the L3Out then the above three steps are all that is required for the internal EPG/BD to use external routes.

Route is installed in BL routing table

In this case the external route that should be learned on BLs 103 and 104 is 172.16.20.1/32.

```
leaf103# show ip route 172.16.20.1 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF

172.16.20.1/32, ubest/mbest: 1/0
    *via 10.10.34.3, vlan347, [110/20], 00:06:29, ospf-default, type-2
```

It is evident that it is installed in the routing table as being learned through OSPF. If it wasn't seen here, check the individual protocol and ensure adjacencies are up. Route is redistributed into BGP The redistribution route-map can be verified, after checking that neither 'Import' enforcement or Interleak Route-Profiles are used, by looking at the route-map used for external protocol to BGP redistribution. See the following command:

```
leaf103# show bgp process vrf Prod:Vrf1

Information regarding configured VRFs:

BGP Information for VRF Prod:Vrf1
VRF Type                        : System
VRF Id                          : 85
VRF state                       : UP
VRF configured                  : yes
VRF refcount                    : 1
VRF VNID                        : 2392068
Router-ID                       : 10.0.0.3
Configured Router-ID            : 10.0.0.3
```

```
Confed-ID                      : 0
Cluster-ID                     : 0.0.0.0
MSITE Cluster-ID               : 0.0.0.0
No. of configured peers        : 1
No. of pending config peers    : 0
No. of established peers       : 1
VRF RD                         : 101:2392068
VRF EVPN RD                    : 101:2392068
...
    Redistribution
        direct, route-map permit-all
        static, route-map imp-ctx-bgp-st-interleak-2392068
        ospf, route-map permit-all
        coop, route-map exp-ctx-st-2392068
        eigrp, route-map permit-all
```

Here it is evident that the 'permit-all' route-map is used for OSPF to BGP redistribution. This is the default. From here, BL can be verified and the local route originating from BGP checked:

```
a-leaf101# show bgp ipv4 unicast 172.16.20.1/32 vrf Prod:Vrf1
BGP routing table information for VRF Prod:Vrf1, address family IPv4 Unicast
BGP routing table entry for 172.16.20.1/32, version 25 dest ptr 0xa6f25ad0
Paths: (2 available, best #2)
Flags: (0x80c0002 00000000) on xmit-list, is not in urib, exported
  vpn: version 16316, (0x100002) on xmit-list
Multipath: eBGP iBGP


  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: redist 0x408 0x1 ref 0 adv path ref 2, path is valid, is best path
  AS-Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (10.0.0.3)
      Origin incomplete, MED 20, localpref 100, weight 32768
      Extcommunity:
          RT:65001:2392068
          VNID:2392068
          COST:pre-bestpath:162:110

  VRF advertise information:
  Path-id 1 not advertised to any peer

  VPN AF advertise information:
  Path-id 1 advertised to peers:
    10.0.64.64         10.0.72.66
  Path-id 2 not advertised to any peer
```

In the above output, the 0.0.0.0/0 indicates it is originated locally. The list of peers advertised to are the spine nodes in the fabric which act as Route-Reflectors.

Verify route on internal leaf

The BL should advertise it to the spine nodes through the VPNv4 BGP Address-Family. The spine nodes should advertise it to any leaf nodes with the VRF deployed (true of non-route-leaking example). On any of these leaf nodes run 'show bgp vpnv4 unicast <route> vrf overlay-1' to verify it is in VPNv4

Use the command below to verify the route on the internal leaf.

```
leaf101# show ip route 172.16.20.1 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF

172.16.20.1/32, ubest/mbest: 2/0
    *via 10.0.72.64%overlay-1, [200/20], 00:21:24, bgp-65001, internal, tag 65001
         recursive next hop: 10.0.72.64/32%overlay-1
    *via 10.0.72.67%overlay-1, [200/20], 00:21:24, bgp-65001, internal, tag 65001
         recursive next hop: 10.0.72.67/32%overlay-1
```

In the above output the route is being learned through BGP and the next-hops should be the Physical TEPs (PTEP) of the BLs.

```
leaf101# acidiag fnvread
      ID   Pod ID                 Name    Serial Number         IP Address    Role        State
LastUpdMsgId
------------------------------------------------------------------------------------------------
--------------
     103        1            a-leaf101      FDO20160TPS      10.0.72.67/32    leaf
active   0
     104        1            a-leaf103      FDO20160TQ0      10.0.72.64/32    leaf
active   0
```

External route troubleshooting scenario

In this scenario the internal leaf (101) is not receiving an external route(s).

As always, first check the basics. Make sure that:

- Routing protocol adjacencies are up on the BLs.
- A BGP Route-Reflector Policy is applied to the Pod Policy-Group and the Pod Profile.

If the above criteria are correct, below are some more advanced examples of what could be causing the issue.

Possible Cause: VRF not deployed on the internal leaf

In this case, the issue would be that there are no EPGs with resources deployed on the internal leaf where the external route is expected. This could be caused by static path bindings only configured on down interfaces or only have On Demand mode VMM integrated EPGs present with no dynamic attachments detected.

Because the L3Out VRF is not deployed on the internal leaf (verify with 'show vrf' on internal leaf) the internal leaf will not import the BGP route from VPNv4.

To resolve this issue, the user should deploy resources within the L3Out VRF on the internal leaf.

Possible Cause: Import Route Enforcement is being used

As mentioned earlier, when import route-control enforcement is enabled the L3Out only accepts external routes that are explicitly permitted. Typically, the feature is implemented as a table-map.

A table-map sits in between the protocol RIB and the actual routing table so that it only affects what is in the routing table.

In the output below the Import Route-Control is enabled, but there aren't any explicitly permitted routes. Notice that the LSA is in the OSPF database but not in the routing table on the BL:

```
leaf103# vsh -c "show ip ospf database external 172.16.20.1 vrf Prod:Vrf1"
        OSPF Router with ID (10.0.0.3) (Process ID default VRF Prod:Vrf1)

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.20.1     10.0.0.134      455         0x80000003 0xb9a0    0
```

```
leaf103# show ip route 172.16.20.1 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF

Route not found
```

Here is the table-map that is now installed causing this behavior:

```
leaf103# show ip ospf vrf Prod:Vrf1

 Routing Process default with ID 10.0.0.3 VRF Prod:Vrf1
 Stateful High Availability enabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Table-map using route-map exp-ctx-2392068-deny-external-tag
 Redistributing External Routes from..

leaf103# show route-map exp-ctx-2392068-deny-external-tag
route-map exp-ctx-2392068-deny-external-tag, deny, sequence 1
  Match clauses:
    tag: 4294967295
  Set clauses:
route-map exp-ctx-2392068-deny-external-tag, deny, sequence 19999
  Match clauses:
    ospf-area: 0.0.0.100
  Set clauses:
```

Anything learning in area 100, which is the area configured on this L3Out, is implicitly denied by this table-map so that it is not installed in the routing table.

To resolve this issue, the user should define the subnet on the external EPG with the 'Import Route Control Subnet' flag or create an Import Route-Profile that matches the prefixes to be installed.

- Note that import enforcement is not supported for EIGRP.
- Also note that for BGP, import enforcement is implemented as an inbound route-map applied to the BGP neighbor. Check the "BGP Route Advertisement" sub-section for details on how to check this.

Possible Cause: an Interleak Profile is being used

Interleak Route-Profiles are used for EIGRP and OSPF L3Outs and intended to allow for control over what is redistributed from the IGP into BGP as well as allows the application of policy such as setting BGP attributes.

Without an interleak Route-Profile, all routes are implicitly imported to BGP.

Without an interleak Route-Profile:

```
leaf103# show bgp process vrf Prod:Vrf1

Information regarding configured VRFs:

BGP Information for VRF Prod:Vrf1
VRF Type                       : System
VRF Id                         : 85
VRF state                      : UP
VRF configured                 : yes
VRF refcount                   : 1
VRF VNID                       : 2392068
Router-ID                      : 10.0.0.3
Configured Router-ID           : 10.0.0.3
Confed-ID                      : 0
Cluster-ID                     : 0.0.0.0
MSITE Cluster-ID               : 0.0.0.0
No. of configured peers        : 1
No. of pending config peers    : 0
No. of established peers       : 1
VRF RD                         : 101:2392068
VRF EVPN RD                    : 101:2392068

...
    Peers      Active-peers    Routes     Paths      Networks   Aggregates
    1          1               7          11         0          0

    Redistribution
        direct, route-map permit-all
        static, route-map imp-ctx-bgp-st-interleak-2392068
        ospf, route-map permit-all
        coop, route-map exp-ctx-st-2392068
        eigrp, route-map permit-all
```

With an interleak route-profile:

```
a-leaf103# show bgp process vrf Prod:Vrf1

Information regarding configured VRFs:

BGP Information for VRF Prod:Vrf1
VRF Type                       : System
VRF Id                         : 85
VRF state                      : UP
VRF configured                 : yes
VRF refcount                   : 1
VRF VNID                       : 2392068
Router-ID                      : 10.0.0.3
Configured Router-ID           : 10.0.0.3
```

```
Confed-ID                       : 0
Cluster-ID                      : 0.0.0.0
MSITE Cluster-ID                : 0.0.0.0
No. of configured peers         : 1
No. of pending config peers     : 0
No. of established peers        : 1
VRF RD                          : 101:2392068
VRF EVPN RD                     : 101:2392068

...

    Redistribution
        direct, route-map permit-all
        static, route-map imp-ctx-bgp-st-interleak-2392068
        ospf, route-map imp-ctx-proto-interleak-2392068
        coop, route-map exp-ctx-st-2392068
        eigrp, route-map permit-all
```

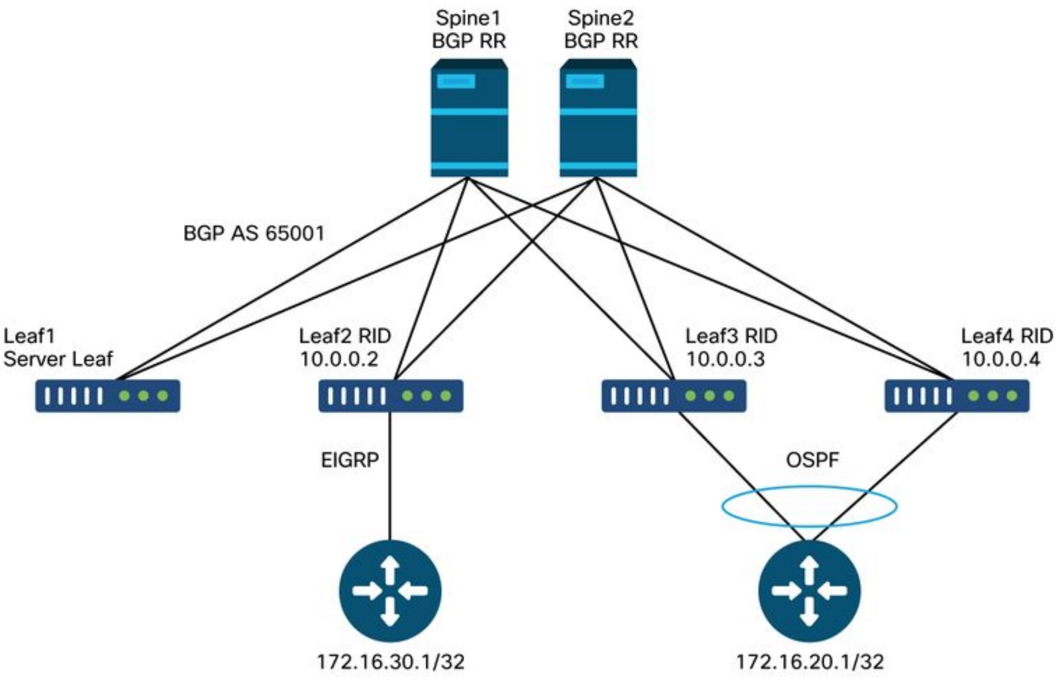
The above highlighted route-map would only permit what is explicitly matched in the configured Interleak Profile. If the external route isn't matched it will not be redistributed into BGP.

Transit route advertisement workflow

This section discusses how routes from one L3Out are advertised out another L3Out. This would also cover the scenario where static routes that are configured directly on an L3Out need to be advertised. It will not go into every specific protocol consideration, but rather through how this is implemented in ACI. It will not go into inter-VRF transit routing at this time.

This scenario will use the following topology:

Transit routing topology



The high-level flow of how 172.16.20.1 would be learned from OSPF and then advertised into

EIGRP, and verifications of the whole process and troubleshooting scenarios, are discussed below.

For the 172.16.20.1 route to get advertised into EIGRP, one of the following must be configured:

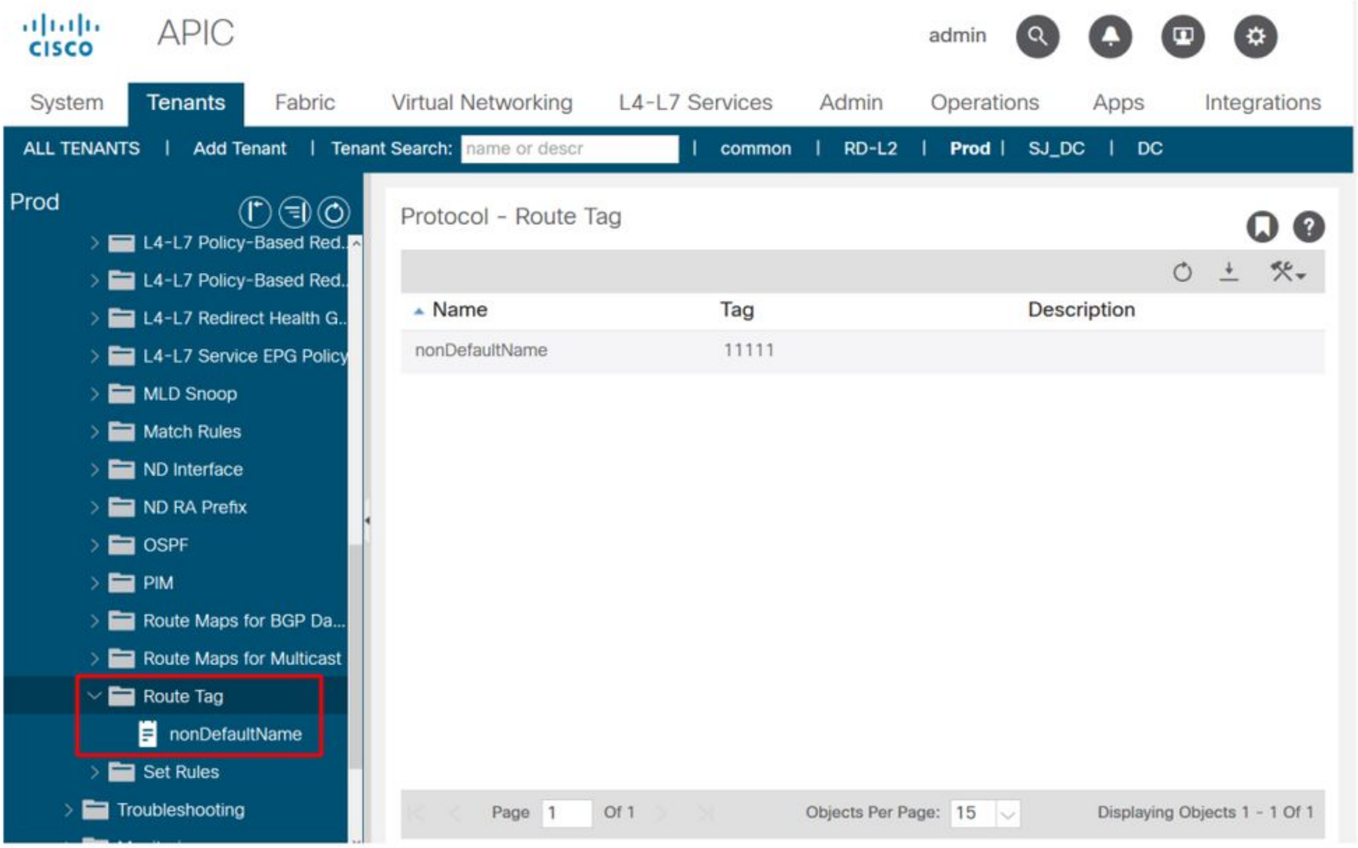
- The subnet to be advertised could be defined on the EIGRP L3Out with the 'Export Route-Control Subnet' flag. As mentioned in the overview section, this flag is used mainly for transit routing and defines the subnets that should be advertised out of that L3Out.
- Configure 0.0.0.0/0 and select both 'Aggregate Export' and 'Export Route Control Subnet'. This creates a route-map for redistribution into the external protocol that matches 0.0.0.0/0 and all prefixes that are more specific (which is an effective match any). Note that when 0.0.0.0/0 is used with 'Aggregate Export', static routes will not be matched for redistribution. This is to prevent inadvertently advertising BD routes that shouldn't be advertised.
- Lastly, it is possible to create an export route-profile that matches the prefixes to be advertised. Using this method could configure the 'Aggregate' option with prefixes besides 0.0.0.0/0.

The above configurations would result in the transit route being advertised but it still needs to have a security policy in place to allow dataplane traffic to flow. As with any EPG to EPG communication, a contract must be in place before traffic is permitted.

> Note that duplicate external subnets with the 'External Subnet for External EPG' cannot be configured in the same VRF. When configured, subnets need to be more specific than 0.0.0.0. It is important to configure 'External Subnet for External EPG' only for the L3Out where the route is being received. Don't configure this on the L3Out that should be advertising this route.

It's also important to understand that all transit routes are tagged with a specific VRF Tag. By default, this tag is 4294967295. The Route-Tag policy is configured under 'Tenant > Networking > Protocols > Route-Tag:

Route-Tag Policy



Name	Tag	Description
nonDefaultName	11111	

This Route Tag policy is then applied to the VRF. The purpose of this tag is essentially to prevent loops. This route tag is applied when the transit route is advertised back out of an L3Out. If these routes are then received back with the same route tag then the route is discarded.

Verify that the route is present on the receiving BL via OSPF

Like the last section, first verify that the BL that should initially receive the correct route.

```
leaf103# show ip route 172.16.20.1 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF

172.16.20.1/32, ubest/mbest: 1/0
    *via 10.10.34.3, vlan347, [110/20], 01:25:30, ospf-default, type-2
```

For now, assume that the advertising L3Out is on a different BL (as in the topology) (later scenarios will discuss where it is on the same BL).

Verify that the route is present in BGP on the receiving OSPF BL

For the OSPF route to be advertised to the external EIGRP router, the route needs to be advertised into BGP on the receiving OSPF BL.

```
leaf103# show bgp ipv4 unicast 172.16.20.1/32 vrf Prod:Vrf1
BGP routing table information for VRF Prod:Vrf1, address family IPv4 Unicast
BGP routing table entry for 172.16.20.1/32, version 30 dest ptr 0xa6f25ad0
Paths: (2 available, best #1)
Flags: (0x80c0002 00000000) on xmit-list, is not in urib, exported
```

```
  vpn: version 17206, (0x100002) on xmit-list
Multipath: eBGP iBGP

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: redist 0x408 0x1 ref 0 adv path ref 2, path is valid, is best path
  AS-Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (10.0.0.3)
      Origin incomplete, MED 20, localpref 100, weight 32768
      Extcommunity:
          RT:65001:2392068
          VNID:2392068
          COST:pre-bestpath:162:110

  VRF advertise information:

  Path-id 1 not advertised to any peer

  VPN AF advertise information:
  Path-id 1 advertised to peers:
    10.0.64.64         10.0.72.66
  Path-id 2 not advertised to any peer
```

The route is in BGP.

Verify on the EIGRP BL that should advertise the route that it is installed

```
leaf102# show ip route 172.16.20.1 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF

172.16.20.1/32, ubest/mbest: 2/0
    *via 10.0.72.67%overlay-1, [200/20], 00:56:46, bgp-65001, internal, tag 65001
         recursive next hop: 10.0.72.67/32%overlay-1
    *via 10.0.72.64%overlay-1, [200/20], 00:56:46, bgp-65001, internal, tag 65001
         recursive next hop: 10.0.72.64/32%overlay-1
```

It is installed in the routing table with overlay next-hops pointing to the originating border leaf nodes.

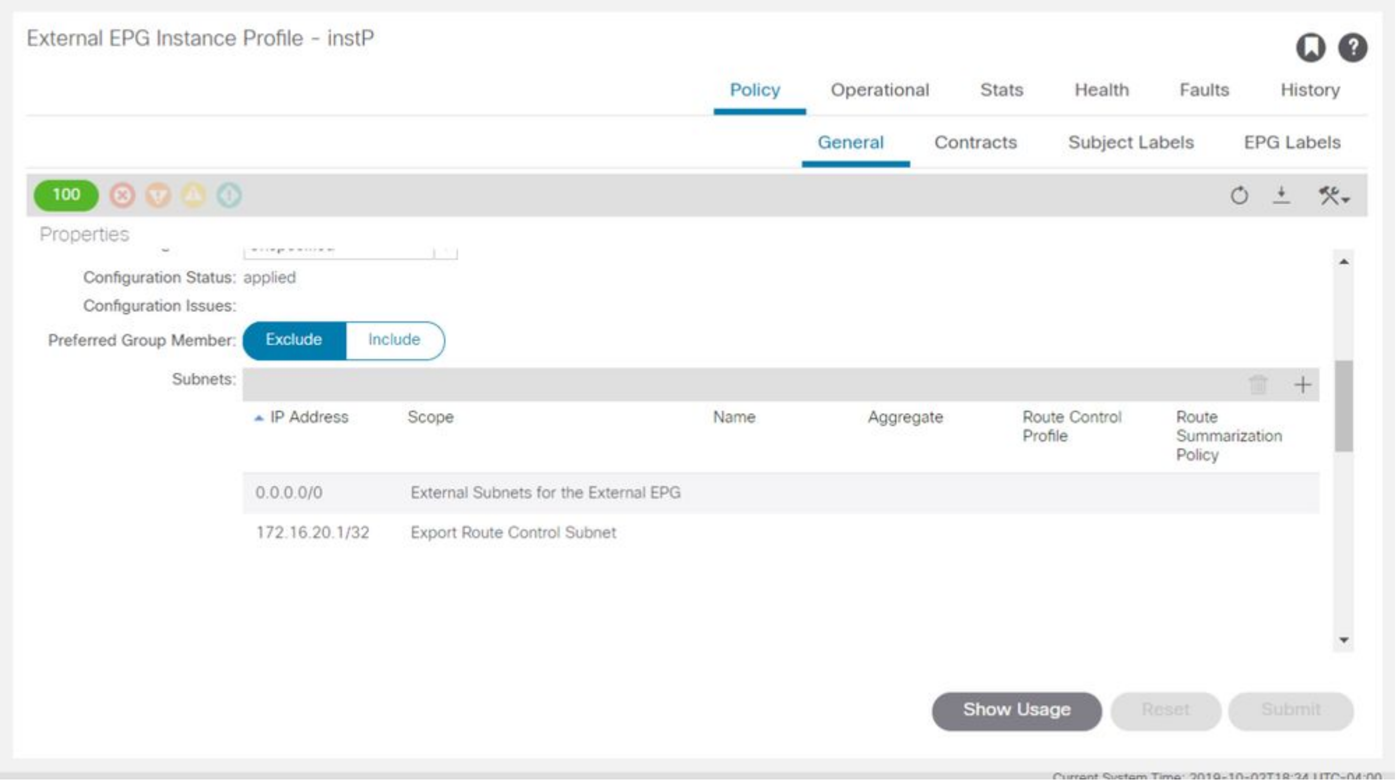
```
leaf102# acidiag fnvread

      ID   Pod ID                 Name    Serial Number         IP Address    Role        State
LastUpdMsgId
------------------------------------------------------------------------------------------------
--------------
     103        1            a-leaf101      FDO20160TPS      10.0.72.67/32    leaf
active   0
     104        1            a-leaf103      FDO20160TQ0      10.0.72.64/32    leaf
active   0
```

Verify that the route is advertised on the BL

The route will be advertised by BL 102 as a result of the 'Export Route Control Subnet' flag being set on the configured subnet:

Export Route Control



External EPG Instance Profile - instP

Configuration Status: applied

Configuration Issues:

Preferred Group Member: [✓ Exclude] Include

Subnets:

IP Address	Scope	Name	Aggregate	Route Control Profile	Route Summarization Policy
0.0.0.0/0	External Subnets for the External EPG				
172.16.20.1/32	Export Route Control Subnet				

Use the following command to view the route-map that is created as a result of this 'Export Route Control' flag:

```
leaf102# show ip eigrp vrf Prod:Vrf1
IP-EIGRP AS 101 ID 10.0.0.2 VRF Prod:Vrf1
  Process-tag: default
  Instance Number: 1
  Status: running
  Authentication mode: none
  Authentication key-chain: none
  Metric weights: K1=1 K2=0 K3=1 K4=0 K5=0
  metric version: 32bit
  IP proto: 88 Multicast group: 224.0.0.10
  Int distance: 90 Ext distance: 170
  Max paths: 8
  Active Interval: 3 minute(s)
  Number of EIGRP interfaces: 1 (0 loopbacks)
  Number of EIGRP passive interfaces: 0
  Number of EIGRP peers: 1
  Redistributing:
    static route-map exp-ctx-st-2392068
    ospf-default route-map exp-ctx-proto-2392068
    direct route-map exp-ctx-st-2392068
    coop route-map exp-ctx-st-2392068
    bgp-65001 route-map exp-ctx-proto-2392068
```

To look for the 'BGP > EIGRP redistribution', look at the route-map. But, the route-map itself should be the same regardless of whether the source protocol is OSPF, EIGRP, or BGP. Static routes will be controlled with a different route-map.

```
leaf102# show route-map exp-ctx-proto-2392068
```

```
route-map exp-ctx-proto-2392068, permit, sequence 15801
  Match clauses:
    ip address prefix-lists: IPv4-proto32771-2392068-exc-ext-inferred-export-dst
    ipv6 address prefix-lists: IPv6-deny-all
  Set clauses:
    tag 4294967295

a-leaf102# show ip prefix-list IPv4-proto32771-2392068-exc-ext-inferred-export-dst
ip prefix-list IPv4-proto32771-2392068-exc-ext-inferred-export-dst: 1 entries
   seq 1 permit 172.16.20.1/32
```

In the above output, the VRF tag is set on this prefix for loop prevention and the subnet configured with 'Export Route Control' is explicitly matched.

Transit Routing when receiving and advertising BL are the same

As discussed earlier, when the receiving and advertising BLs are different, the route must be advertised through the fabric using BGP. When the BLs are the same, the redistribution or advertisement can be done directly between the protocols on the leaf.

Below are brief descriptions of how this is implemented:

- **Transit routing between two OSPF L3Outs on the same leaf:** Route advertisement is controlled via an 'area-filter' applied to the OSPF process level. An L3Out in Area 0 must be deployed on the leaf since the routes are advertised between areas as opposed to through redistribution. Use 'show ip ospf vrf <name>' to view the filter-list. Display the contents of the filter using 'show route-map <filter name>'.
- **Transit routing between OSPF and EIGRP L3Outs on the same leaf:** Route advertisement is controlled via redistribution route-maps that can be seen with 'show ip ospf' and 'show ip eigrp'. Note that if multiple OSPF L3Outs exist on the same BL the only way to redistribute into only one of those OSPF L3Outs is if the other is a Stub or NSSA with '**Send redistributed LSAs into NSSA area' disabled so** that it doesn't allow any external LSA's.
- **Transit routing between OSPF or EIGRP and BGP on the same leaf:** Route advertisement into the IGP is controlled via redistribution route-maps. Route-advertisement into BGP is controlled via an outbound route-map applied directly to the bgp neighbor that the route should be sent do. This can be verified with 'show bgp ipv4 unicast neighbor <neighbor address> vrf <name> | grep Outbound'.
- **Transit routing between two BGP l3Outs on the same leaf:** All advertisement is controlled via route-maps applied directly to the bgp neighbor that the route should be sent to. This can be verified with 'show bgp ipv4 unicast neighbor <neighbor address> vrf <name> | grep Outbound'.

Transit routing troubleshooting scenarios #1: Transit Route not advertised

This troubleshooting scenario involves routes that should be learned through one L3Out not being sent out the other L3Out.

As always, check the basics before looking at anything ACI specific.

- Are protocol adjacencies up?
- Is the route, that ACI should be advertising, learned from an external protocol in the first

place?
- For BGP, is the path being dropped due to some BGP attribute? (as-path, etc.).
- Does the receiving L3Out have it in the OSPF database, EIGRP topology table, or BGP table?
- Is a BGP Route Reflector Policy applied to the Pod Policy Group that is applied to the Pod Profile?

If all the basic protocol verifications are configured correctly, below are some other common causes for a transit route that is not being advertised.

Possible Cause: No OSPF Area 0

If the affected topology involves two OSP L3Outs on the same border leaf, then there must be an Area 0 for routes to be advertised from one area to another. Look at the "Transit routing between two OSPF L3Outs on the same leaf" bullet above for more details.

Possible Cause: OSPF area is stub or NSSA

This would be seen if the OSPF L3Out is configured with a Stub or NSSA area that is not configured to advertise external LSAs. With OSPF, external LSAs are never advertised into Stub areas. They are advertised into NSSA areas if 'Send Redistributed LSAs into NSSA Area' is selected.

Transit routing troubleshooting scenarios #2: Transit Route not received

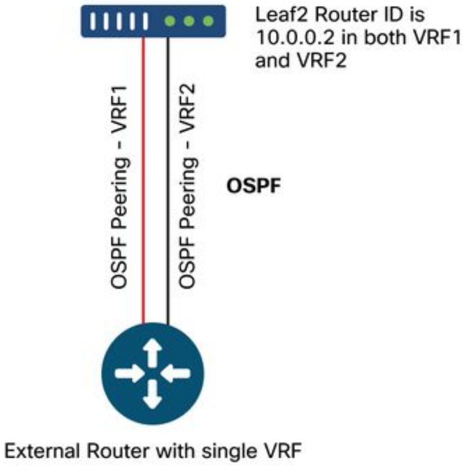
In this scenario the problem is that some routes advertised by an ACI L3Out are not being received back in another L3Out. This scenario could be applicable if the L3Outs are in two separate fabrics and are connected by external routers or if the L3Outs are in different VRFs and the routes are being passed between the VRFs by an external router.

Possible Cause: BL is Configured with the same Router ID in multiple VRFs

From a configuration perspective, a router-id cannot be duplicated within the same VRF. However, it is typically fine to use the same router-id in different VRFs as long as the two VRFs aren't attached to the same routing protocol domains.

Consider the following topology:

External router with single VRF — Transit Route not received


Leaf2 Router ID is 10.0.0.2 in both VRF1 and VRF2
OSPF Peering - VRF1
OSPF Peering - VRF2
OSPF
External Router with single VRF

The problem here would be that the ACI leaf sees LSAs with its own Router-ID being received, resulting in these not being installed in the OSPF database.

In addition, if the same setup was seen with VPC pairs, LSAs would continuously be added and deleted on some routers. For example, the router would see LSAs coming from its VPC peer with VRF and LSAs coming from the same node (with same Router-ID) that were originated in the other VRF.

To resolve this issue, the user should make sure that a node will have a different, unique router-id within each VRF that it has an L3Out in.

Possible cause: routes from one L3Out in one ACI fabric received on another fabric with same VRF tag

The default route-tag in ACI is always the same unless it is changed. If routes are advertised from one L3Out in one VRF or ACI fabric to another L3Out in another VRF or ACI fabric without changing the default VRF tags, the routes will be dropped by the receiving BLs.

The solution to this scenario is simply to use a unique Route-Tag policy for each VRF in ACI.

Transit routing troubleshooting scenarios #3 — Transit Routes unexpectantly advertised

This scenario would be seen when transit routes are advertised out an L3Out where they are not intended to be advertised.

Possible cause: usage of 0.0.0.0/0 with 'Aggregate Export'

When an external subnet is configured as 0.0.0.0/0 with 'Export Route Control Subnet' and 'Aggregate Export' the result is that a match all redistribution route-map is installed. In this case all routes on the BL that were learned through OSPF, EIGRP, or BGP are advertised out the L3Out where this is configured.

Below is the route-map that is deployed to the leaf as a result of the Aggregate Export:

```
leaf102# show ip eigrp vrf Prod:Vrf1
IP-EIGRP AS 101 ID 10.0.0.2 VRF Prod:Vrf1
```

```
 Process-tag: default
 Instance Number: 1
 Status: running
 Authentication mode: none
 Authentication key-chain: none
 Metric weights: K1=1 K2=0 K3=1 K4=0 K5=0
 metric version: 32bit
 IP proto: 88 Multicast group: 224.0.0.10
 Int distance: 90 Ext distance: 170
 Max paths: 8
 Active Interval: 3 minute(s)
 Number of EIGRP interfaces: 1 (0 loopbacks)
 Number of EIGRP passive interfaces: 0
 Number of EIGRP peers: 1
 Redistributing:
   static route-map exp-ctx-st-2392068
   ospf-default route-map exp-ctx-proto-2392068
   direct route-map exp-ctx-st-2392068
   coop route-map exp-ctx-st-2392068
   bgp-65001 route-map exp-ctx-proto-2392068
 Tablemap: route-map exp-ctx-2392068-deny-external-tag , filter-configured
 Graceful-Restart: Enabled
 Stub-Routing: Disabled
 NSF converge time limit/expiries: 120/0
 NSF route-hold time limit/expiries: 240/0
 NSF signal time limit/expiries: 20/0
 Redistributed max-prefix: Disabled
 selfAdvRtTag: 4294967295
leaf102# show route-map exp-ctx-proto-2392068
route-map exp-ctx-proto-2392068, permit, sequence 19801
 Match clauses:
   ip address prefix-lists: IPv4-proto32771-2392068-agg-ext-inferred-export-dst
   ipv6 address prefix-lists: IPv6-deny-all
 Set clauses:
   tag 4294967295
```

```
leaf102# show ip prefix-list IPv4-proto32771-2392068-agg-ext-inferred-export-dst
    ip prefix-list IPv4-proto32771-2392068-agg-ext-inferred-export-dst: 1 entries
seq 1 permit 0.0.0.0/0 le 32
```

This is the number one cause of routing loops that involve an ACI environment.

Contract and L3Out

Prefix-based EPG on L3Out

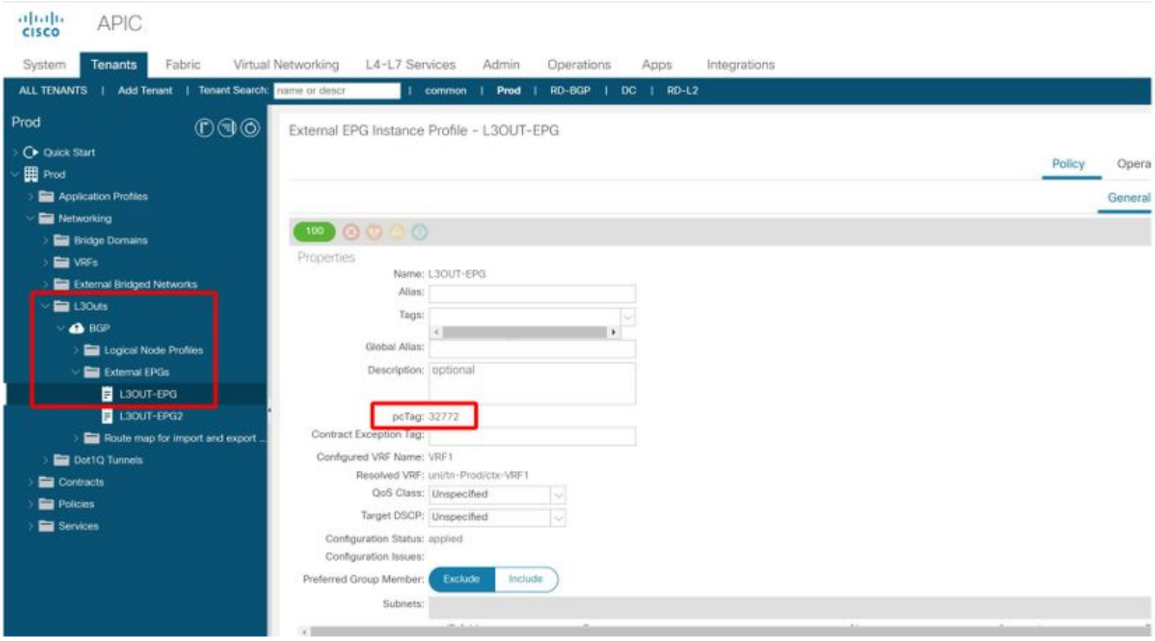
In an internal EPG (non-L3Out), contracts are enforced after deriving the pcTag of the source and the pcTag of the destination EPG. The encapsulation VLAN/VXLAN of the packet received on the downlink port is used to drive this pcTag by classing the packet into the EPG. Whenever learning a MAC address or an IP address, it is learnt along with its access encapsulation and the associated EPG pcTag. For more details on pcTag and contract enforcement, please refer to the "Security policies" chapter.

L3Outs also drive a pcTag using its L3Out EPG (External EPG) located under 'Tenant > Networking > L3OUT > Networks > L3OUT-EPG'. However, L3Outs do not rely on VLANs and interfaces to classify packets as such. Classification is instead based on source prefix/subnet in a

'Longest Prefix Match' fashion. Hence, an L3Out EPG can be referred to as a **prefix-based EPG**. After a packet is classified into an L3Out based on a subnet, it follows a similar policy enforcement pattern as a regular EPG.

The following diagram outlines where the pcTag of a given L3Out EPG can be found within the GUI.

Location of the pcTag for an L3Out



External EPG Instance Profile - L3OUT-EPG

Name: L3OUT-EPG
Alias:
Tags:
Global Alias:
Description: optional
pcTag: 32772
Contract Exception Tag:
Configured VRF Name: VRF1
Resolved VRF: uni/tn-Prod/ctx-VRF1
QoS Class: Unspecified
Target DSCP: Unspecified
Configuration Status: applied
Configuration Issues:
Preferred Group Member: Exclude
Subnets:

The user is responsible for defining the prefix-based EPG table. This is done using the 'External Subnet for External EPG' subnet scope. Each subnet set with that scope will add an entry in a static Longest Prefix Match (LPM) table. This subnet will point to the pcTag value that will get used for any IP address falling within that prefix.

Th LPM table of prefix-based EPG subnets can be verified on leaf switches using the following command:

```
vsh -c 'show system internal policy-mgr prefix'
```

Remarks:

- LPM table entries are scoped to VRF VNID. The lookup is done per vrf_vnid/src pcTag/dst pcTag.
- Each entry points to a single pcTag. As a consequence, two L3Out EPGs cannot use the same subnet with the same mask length within the same VRF.
- Subnet 0.0.0.0/0 always uses special pcTag 15. As such, it can be duplicated but should only be done so with a full understanding of the policy enforcement implications.
- This table is used in both directions. From L3Out to Leaf Local Endpoint, the source pcTag is derived using this table.From Leaf Local Endpoint to L3Out, the destination pcTag is derived using this table.
- If the VRF has the 'Ingress' enforcement setting for 'Policy Control Enforcement Direction', then the LPM prefix table will be present on the L3Out BLs as well as any leaf switches in the VRF that have a contract with the L3Out.

Example 1: Single L3Out with specific prefix

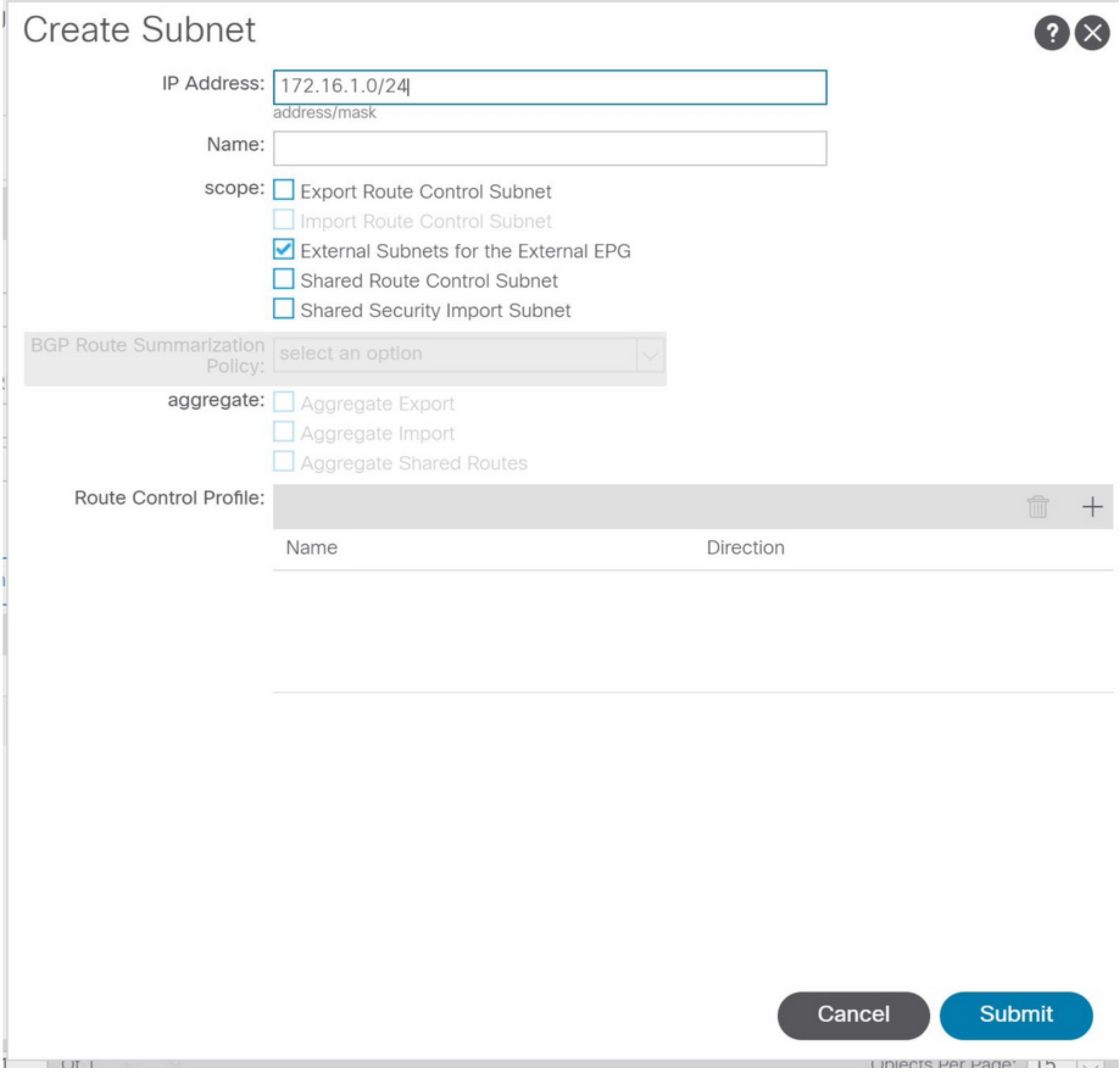
Scenario: A single BGP L3Out in vrf Prod:VRF1 with one L3Out EPG. Prefix 172.16.1.0/24 is being received from an external source so it must be classified into the L3Out EPG.

```
bdsol-aci32-leaf3# show ip route 172.16.1.0  vrf Prod:VRF1
IP Route Table for VRF "Prod:VRF1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF

172.16.1.0/24, ubest/mbest: 1/0
    *via 10.0.0.134%Prod:VRF1, [20/0], 00:56:14, bgp-132, external, tag 65002
         recursive next hop: 10.0.0.134/32%Prod:VRF1
```

First, add the subnet to the prefix table.

Subnet with 'External Subnets for the External EPG' scope



Create Subnet

IP Address: 172.16.1.0/24
address/mask

Name:

scope:
□ Export Route Control Subnet
□ Import Route Control Subnet
☑ External Subnets for the External EPG
□ Shared Route Control Subnet
□ Shared Security Import Subnet

BGP Route Summarization Policy: select an option

aggregate:
□ Aggregate Export
□ Aggregate Import
□ Aggregate Shared Routes

Route Control Profile:

Name	Direction

Verify the programming of the prefix list on the leaf switches that have the VRF of the L3Out:

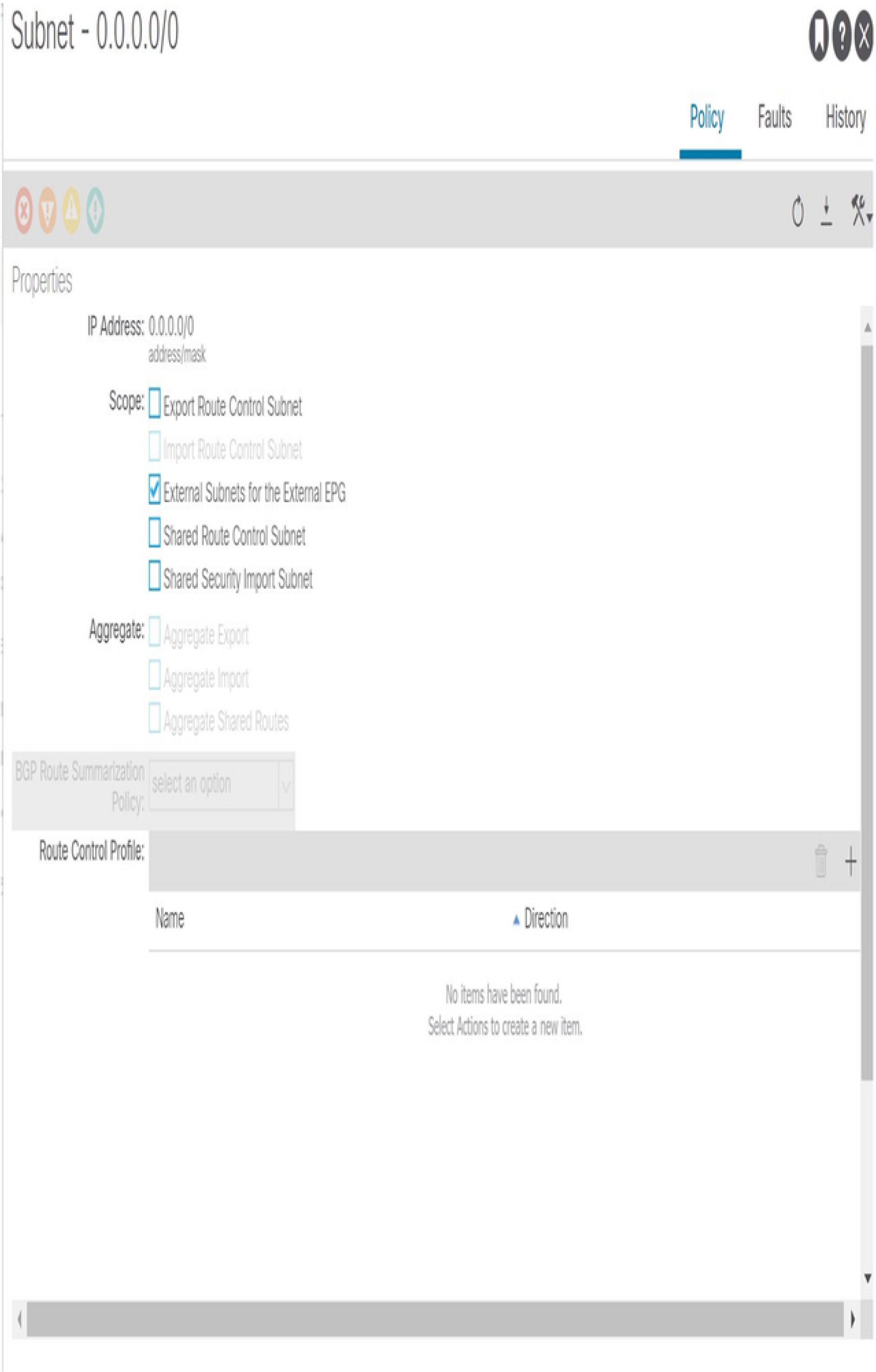
```
bdsol-aci32-leaf3# vsh -c ' show system internal  policy-mgr prefix ' |  egrep "Prod|==|Addr"
Vrf-Vni VRF-Id Table-Id Table-State  VRF-Name                    Addr
Class Shared Remote Complete
======= ======  =========== =======  ============================
================================= ====== ====== ====== ========
2097154 35     0x23          Up     Prod:VRF1
0.0.0.0/0   15      True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.1.0/24  32772   True   True   False
```

The pcTag of the L3Out EPG is 32772 in vrf scope 2097154.

Example 2: Single L3Out with multiple prefixes

Expanding on the previous example, in this scenario the L3Out is receiving multiple prefixes. While entering each prefix is functionally sound, an alternative option (depending on the intended design) is to accept all prefixes received on the L3Out.

This can be accomplished with the '0.0.0.0/0' prefix.



Subnet - 0.0.0.0/0

Properties

IP Address: 0.0.0.0/0
address/mask

Scope:
- □ Export Route Control Subnet
- □ Import Route Control Subnet
- ☑ External Subnets for the External EPG
- □ Shared Route Control Subnet
- □ Shared Security Import Subnet

Aggregate:
- □ Aggregate Export
- □ Aggregate Import
- □ Aggregate Shared Routes

BGP Route Summarization Policy: select an option

Route Control Profile:

Name	▲ Direction

No items have been found.
Select Actions to create a new item.

This results in the following policy-mgr prefix table entry:

```
bdsol-aci32-leaf3# vsh -c ' show system internal  policy-mgr prefix ' |  egrep "Prod|==|Addr"
Vrf-Vni VRF-Id Table-Id Table-State  VRF-Name                    Addr
Class Shared Remote Complete
======= ======  =========== =======  ============================
================================ ====== ====== ====== ========
2097154 35     0x23          Up     Prod:VRF1
0.0.0.0/0   15      True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.1.0/24  32772   True   True   False
```

> Note that the pcTag assigned to 0.0.0.0/0 uses value 15, not 32772. pcTag 15 is a reserved system pcTag which is only used with 0.0.0.0/0 which acts as wildcard to match all prefixes on an L3Out.

If the VRF has a single L3Out with a single L3Out EPG using the 0.0.0.0/0, then the policy-prefix remains unique and is the easiest approach to catch all.

Example 3a: Multiple L3Out EPGs in a VRF

In this scenario there are multiple L3Out EPGs in the same VRF.

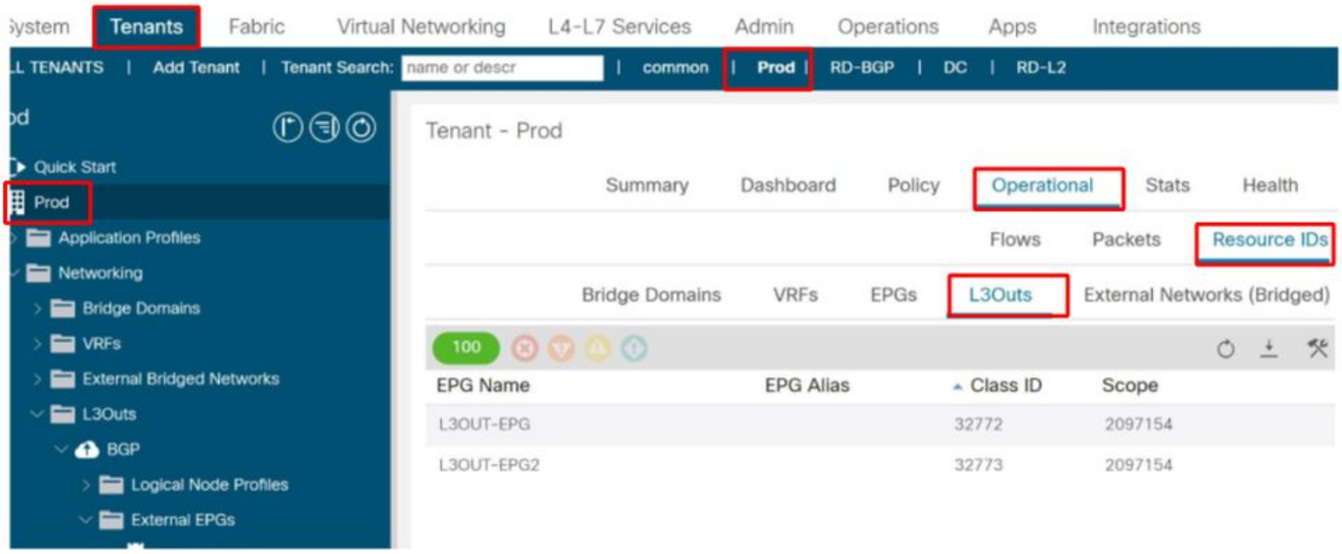
Note: From a prefix-based EPG perspective, the following two configurations will result in equivalent LPM policy-mgr prefix table entries:

1. Two L3Outs with one L3Out EPG each.
2. One L3Out with two L3Out EPGs

In both scenarios, the total number of L3Out EPGs is 2. This means that each one will have its own pcTag and associated subnets.

All pcTags of a given L3Out EPG can be viewed in the GUI at 'Tenant > Operational > Resource id > L3Outs'

Verification of the L3Out pcTag



EPG Name	EPG Alias	Class ID	Scope
L3OUT-EPG		32772	2097154
L3OUT-EPG2		32773	2097154

In this scenario, the ACI fabric is receiving multiple prefixes from the external routers and the L3Out EPG definition is as follows:

- 172.16.1.0/24 assigned to L3OUT-EPG.
- 172.16.2.0/24 assigned to L3OUT-EPG2.
- 172.16.0.0/16 assigned to L3OUT-EPG (to catch the 172.16.3.0/24 prefix).

To match this, the config will be defined as follows:

- L3OUT-EPG has subnet 172.16.1.0/24 and 172.16.0.0/16 both with scope 'External Subnet for the External EPG'.
- L3OUT-EPG2 has subnet 172.16.2.0/24 with scope 'External Subnet for the External EPG'.

The resulting prefix table entries will be:

```
bdsol-aci32-leaf3# vsh -c 'show system internal  policy-mgr prefix' |  egrep "Prod|==|Addr"
Vrf-Vni VRF-Id Table-Id Table-State  VRF-Name                    Addr
Class Shared Remote Complete
======= ======  =========== =======  ============================
================================ ====== ====== ====== ========
2097154 35     0x23          Up     Prod:VRF1
0.0.0.0/0   15      True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.1.0/24  32772   True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.0.0/16  32772   True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.2.0/24  32773   True   True   False
```

172.16.2.0/24 is assigned to pcTag 32773 (L3OUT-EPG2) and 172.16.0.0/16 is assigned to 32772 (L3OUT-EPG).

In this scenario, the entry for 172.16.1.0/24 is redundant as the /16 supernet is assigned to the same EPG.

Multiple L3Out EPGs is useful when the goal is to apply different contracts to groups of prefixes within a single L3Out. The next example will illustrate how contracts come into play with multiple L3Out EPGs.

Example 3b: multiple L3Out EPGs with different contracts

This scenario contains the following setup:

- ICMP contract allowing only ICMP.
- HTTP contract allowing only tcp destination port 80.
- EPG1 (pcTag 32770) provides the HTTP contract consumed by L3OUT-EPG (pcTag 32772).
- EPG2 (pcTag 32771) provides the ICMP contract consumed by L3OUT-EPG2 (pcTag 32773).

The same policymgr prefixes from the previous example will be used:

- 172.16.1.0/24 in L3OUT-EPG should permit HTTP to EPG1
- 172.16.2.0/24 in L3OUT-EPG2 should permit ICMP to EPG2

policy-mgr prefix and zoning-rules:

```
bdsol-aci32-leaf3# vsh -c ' show system internal  policy-mgr prefix ' |  egrep "Prod|==|Addr"
Vrf-Vni VRF-Id Table-Id Table-State  VRF-Name                    Addr
Class Shared Remote Complete
======= ======  =========== =======  ============================
================================= ====== ====== ====== ========
2097154 35     0x23          Up     Prod:VRF1
0.0.0.0/0   15      True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.1.0/24  32772   True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.0.0/16  32772   True   True   False
2097154 35     0x23          Up     Prod:VRF1
172.16.2.0/24  32773   True   True   False
```

```
bdsol-aci32-leaf3# show zoning-rule scope 2097154
+---------+--------+--------+----------+----------------+---------+---------+------+----------+-
---------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |      Dir       |  operSt |  Scope  | Name |  Action  |
Priority       |
+---------+--------+--------+----------+----------------+---------+---------+------+----------+-
---------------------+
|   4326  |   0    |   0    | implicit |    uni-dir     | enabled | 2097154 |      | deny,log |
any_any_any(21)     |
|   4335  |   0    | 16387  | implicit |    uni-dir     | enabled | 2097154 |      |  permit  |
any_dest_any(16)    |
|   4334  |   0    |   0    | implarp  |    uni-dir     | enabled | 2097154 |      |  permit  |
any_any_filter(17)  |
|   4333  |   0    |   15   | implicit |    uni-dir     | enabled | 2097154 |      | deny,log |
any_vrf_any_deny(22) |
|   4332  |   0    | 16386  | implicit |    uni-dir     | enabled | 2097154 |      |  permit  |
any_dest_any(16)    |
|   4342  | 32771  | 32773  |    5     | uni-dir-ignore | enabled | 2097154 | ICMP |  permit  |
fully_qual(7)      |
|   4343  | 32773  | 32771  |    5     |     bi-dir     | enabled | 2097154 | ICMP |  permit  |
fully_qual(7)      |
|   4340  | 32770  | 32772  |    38    |    uni-dir     | enabled | 2097154 | HTTP |  permit  |
fully_qual(7)      |
|   4338  | 32772  | 32770  |    37    |    uni-dir     | enabled | 2097154 | HTTP |  permit  |
fully_qual(7)      |
+---------+--------+--------+----------+----------------+---------+---------+------+----------+-
---------------------+
```

Datapath validation using fTriage — flow allowed by policy

With an ICMP flow between 172.16.2.1 on the external network and 192.168.3.1 in EPG2, fTriage can be used to catch and analyze the flow. In this case, start fTriage on both leaf switch 103 and 104 as traffic may enter either of them:

```
admin@apic1:~> ftriage route -ii LEAF:103,104 -sip 172.16.2.1 -dip 192.168.3.1
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "InProgress", "pid": "14454",
"apicId": "1", "id": "0"}}}
Starting ftriage
Log file name for the current run is: ftlog_2019-10-02-22-30-41-871.txt
2019-10-02 22:30:41,874 INFO     /controller/bin/ftriage route -ii LEAF:103,104 -sip 172.16.2.1
-dip 192.168.3.1
2019-10-02 22:31:28,868 INFO     ftriage:     main:1165 Invoking ftriage with default password
and default username: apic#fallback\\admin
2019-10-02 22:32:15,076 INFO     ftriage:     main:839  L3 packet Seen on bdsol-aci32-leaf3
```

```
Ingress: Eth1/12 (Po1) Egress: Eth1/12 (Po1) Vnid: 11365
2019-10-02 22:32:15,295 INFO     ftriage:     main:242  ingress encap string vlan-2551
2019-10-02 22:32:17,839 INFO     ftriage:     main:271  Building ingress BD(s), Ctx
2019-10-02 22:32:20,583 INFO     ftriage:     main:294  Ingress BD(s) Prod:VRF1:l3out-BGP:vlan-
2551
2019-10-02 22:32:20,584 INFO     ftriage:     main:301  Ingress Ctx: Prod:VRF1
2019-10-02 22:32:20,693 INFO     ftriage:   pktrec:490  bdsol-aci32-leaf3: Collecting transient
losses snapshot for LC module: 1
2019-10-02 22:32:38,933 INFO     ftriage:     nxos:1404 bdsol-aci32-leaf3: nxos matching rule
id:4343 scope:34 filter:5
2019-10-02 22:32:39,931 INFO     ftriage:     main:522  Computed egress encap string vlan-2502
2019-10-02 22:32:39,933 INFO     ftriage:     main:313  Building egress BD(s), Ctx
2019-10-02 22:32:41,796 INFO     ftriage:     main:331  Egress Ctx Prod:VRF1
2019-10-02 22:32:41,796 INFO     ftriage:     main:332  Egress BD(s): Prod:BD2
2019-10-02 22:32:48,636 INFO     ftriage:     main:933  SIP 172.16.2.1 DIP 192.168.3.1
2019-10-02 22:32:48,637 INFO     ftriage:  unicast:973  bdsol-aci32-leaf3: <- is ingress node
2019-10-02 22:32:51,257 INFO     ftriage:  unicast:1202 bdsol-aci32-leaf3: Dst EP is local
2019-10-02 22:32:54,129 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: EP if(Po1) same as
egr if(Po1)
2019-10-02 22:32:55,348 INFO     ftriage:     misc:657  bdsol-aci32-leaf3:
DMAC(00:22:BD:F8:19:FF) same as RMAC(00:22:BD:F8:19:FF)
2019-10-02 22:32:55,349 INFO     ftriage:     misc:659  bdsol-aci32-leaf3: L3 packet getting
routed/bounced in SUG
2019-10-02 22:32:55,596 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: Dst IP is present in
SUG L3 tbl
2019-10-02 22:32:55,896 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: RW seg_id:11365 in
SUG same as EP segid:11365
2019-10-02 22:33:02,150 INFO     ftriage:     main:961  Packet is Exiting fabric with peer-
device: bdsol-aci32-n3k-3 and peer-port: Ethernet1/16
```

fTriage confirms the zoning-rule hit against the ICMP rule from L3OUT_EPG2 to EPG:

```
2019-10-02 22:32:38,933 INFO     ftriage:     nxos:1404 bdsol-aci32-leaf3: nxos matching rule
id:4343 scope:34 filter:5
```

Datapath validation using fTriage — flow that is not allowed by policy

With ICMP traffic sourced from 172.16.1.1 (L3OUT-EPG) towards 192.168.3.1 (EPG2), expect a policy drop.

```
admin@apic1:~> ftriage route -ii LEAF:103,104 -sip 172.16.1.1 -dip 192.168.3.1
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "InProgress", "pid": "15139",
"apicId": "1", "id": "0"}}}
Starting ftriage
Log file name for the current run is: ftlog_2019-10-02-22-39-15-050.txt
2019-10-02 22:39:15,056 INFO     /controller/bin/ftriage route -ii LEAF:103,104 -sip 172.16.1.1
-dip 192.168.3.1
2019-10-02 22:40:03,523 INFO     ftriage:     main:1165 Invoking ftriage with default password
and default username: apic#fallback\\admin
2019-10-02 22:40:43,338 ERROR    ftriage:  unicast:234  bdsol-aci32-leaf3: L3 packet getting fwd
dropped, checking drop reason
2019-10-02 22:40:43,339 ERROR    ftriage:  unicast:234  bdsol-aci32-leaf3: L3 packet getting fwd
dropped, checking drop reason
SECURITY_GROUP_DENY              condition setcast:236  bdsol-aci32-leaf3: Drop reason -
SECURITY_GROUP_DENY              condition set
2019-10-02 22:40:43,340 INFO     ftriage:  unicast:252  bdsol-aci32-leaf3: policy drop flow
sclass:32772 dclass:32771 sg_label:34 proto:1
2019-10-02 22:40:43,340 INFO     ftriage:     main:681  : Ftriage Completed with hunch: None
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "Idle", "pid": "0", "apicId": "0",
"id": "0"}}}
```

fTriage confirms that the packet is dropped with the SECURITY_GROUP_DENY (policy drop) reason and that the derived source pcTag is 32772 and destination pcTag is 32771. Checking this against zoning-rules, there are clearly no entries between those EPG.

```
bdsol-aci32-leaf3# show zoning-rule  scope 2097154 src-epg 32772 dst-epg 32771
+---------+--------+--------+----------+-----+--------+-------+------+--------+----------+
| Rule ID | SrcEPG | DstEPG | FilterID | Dir | operSt | Scope | Name | Action | Priority |
+---------+--------+--------+----------+-----+--------+-------+------+--------+----------+
+---------+--------+--------+----------+-----+--------+-------+------+--------+----------+
```

Example 4: multiple L3Outs with multiple prefixes

The scenario is setup similarly to example 3 (L3Out and L3Out EPG definitions), but the network defined on both L3Out EPGs is 0.0.0.0/0.

Contract configuration is the following:

- ICMP1 contract allowing ICMP.
- ICMP2 contract allowing ICMP.
- EPG1 (pcTag 32770) provides ICMP1 contract which is consumed by L3OUT-EPG (pcTag 32772).
- EPG2 (pcTag 32771) provides ICMP2 contract which is consumed by L3OUT-EPG2 (pcTag 32773).

This configuration may look ideal in the case where the external network is advertising many prefixes, but there are at least two chunks of prefixes that follow different allowed flow patterns. In this example, one prefix should only allow ICMP1 and the other should only allow ICMP2.

Despite to using '0.0.0.0/0' twice in the same VRF, only one prefix gets programmed in the policy-mgr prefix table:

```
bdsol-aci32-leaf3# vsh -c ' show system internal  policy-mgr prefix ' |  egrep "Prod|==|Addr"
Vrf-Vni VRF-Id Table-Id Table-State  VRF-Name                   Addr
Class Shared Remote Complete
======= ======  =========== =======  ============================
================================ ====== ====== ====== ========
2097154 35     0x23          Up     Prod:VRF1
```

Two flows reexamined below. Based on the contract configuration above, the following is expected:

1. 172.16.2.1 (L3OUT-EPG2) to 192.168.3.1 (EPG2) **should** be allowed by ICMP2
2. 172.16.2.1 (L3OUT-EPG2) to 192.168.1.1 (EPG1) **should not** be allowed as there is no contract between EPG1 and L3OUT-EPG2

Datapath validation using fTriage — flow that is allowed by policy

Run fTriage with an ICMP flow from 172.16.2.1 (L3OUT-EPG2) to 192.168.3.1 (EPG2 — pcTag 32771).

```
Starting ftriage
Log file name for the current run is: ftlog_2019-10-02-23-11-14-298.txt
2019-10-02 23:11:14,302 INFO     /controller/bin/ftriage route -ii LEAF:103,104 -sip 172.16.2.1
-dip 192.168.3.1
2019-10-02 23:12:00,887 INFO     ftriage:     main:1165 Invoking ftriage with default password
and default username: apic#fallback\\admin
2019-10-02 23:12:44,565 INFO     ftriage:     main:839  L3 packet Seen on bdsol-aci32-leaf3
Ingress: Eth1/12 (Po1) Egress: Eth1/12 (Po1) Vnid: 11365
2019-10-02 23:12:44,782 INFO     ftriage:     main:242  ingress encap string vlan-2551
2019-10-02 23:12:47,260 INFO     ftriage:     main:271  Building ingress BD(s), Ctx
2019-10-02 23:12:50,041 INFO     ftriage:     main:294  Ingress BD(s) Prod:VRF1:l3out-BGP:vlan-
2551
2019-10-02 23:12:50,042 INFO     ftriage:     main:301  Ingress Ctx: Prod:VRF1
2019-10-02 23:12:50,151 INFO     ftriage:   pktrec:490  bdsol-aci32-leaf3: Collecting transient
losses snapshot for LC module: 1
2019-10-02 23:13:08,595 INFO     ftriage:     nxos:1404 bdsol-aci32-leaf3: nxos matching rule
id:4336 scope:34 filter:5
2019-10-02 23:13:09,608 INFO     ftriage:     main:522  Computed egress encap string vlan-2502
2019-10-02 23:13:09,609 INFO     ftriage:     main:313  Building egress BD(s), Ctx
2019-10-02 23:13:11,449 INFO     ftriage:     main:331  Egress Ctx Prod:VRF1
2019-10-02 23:13:11,449 INFO     ftriage:     main:332  Egress BD(s): Prod:BD2
2019-10-02 23:13:18,383 INFO     ftriage:     main:933  SIP 172.16.2.1 DIP 192.168.3.1
2019-10-02 23:13:18,384 INFO     ftriage:  unicast:973  bdsol-aci32-leaf3: <- is ingress node
2019-10-02 23:13:21,078 INFO     ftriage:  unicast:1202 bdsol-aci32-leaf3: Dst EP is local
2019-10-02 23:13:23,926 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: EP if(Po1) same as
egr if(Po1)
2019-10-02 23:13:25,216 INFO     ftriage:     misc:657  bdsol-aci32-leaf3:
DMAC(00:22:BD:F8:19:FF) same as RMAC(00:22:BD:F8:19:FF)
2019-10-02 23:13:25,217 INFO     ftriage:     misc:659  bdsol-aci32-leaf3: L3 packet getting
routed/bounced in SUG
2019-10-02 23:13:25,465 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: Dst IP is present in
SUG L3 tbl
2019-10-02 23:13:25,757 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: RW seg_id:11365 in
SUG same as EP segid:11365
2019-10-02 23:13:32,235 INFO     ftriage:     main:961  Packet is Exiting fabric with peer-
device: bdsol-aci32-n3k-3 and peer-port: Ethernet1/16
```

This flow is allowed (as expected) by zoning-rule 4336.

Datapath validation using fTriage — flow that is not allowed by policy

Run fTriage with an ICMP flow from 172.16.2.1 (L3OUT-EPG2) to 192.168.1.1 (EPG1 — pcTag 32770):

```
admin@apic1:~> ftriage route -ii LEAF:103,104 -sip 172.16.2.1 -dip 192.168.1.1
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "InProgress", "pid": "31500",
"apicId": "1", "id": "0"}}}
Starting ftriage
Log file name for the current run is: ftlog_2019-10-02-23-53-03-478.txt
2019-10-02 23:53:03,482 INFO     /controller/bin/ftriage route -ii LEAF:103,104 -sip 172.16.2.1
-dip 192.168.1.1
2019-10-02 23:53:50,014 INFO     ftriage:     main:1165 Invoking ftriage with default password
and default username: apic#fallback\\admin
2019-10-02 23:54:39,199 INFO     ftriage:     main:839  L3 packet Seen on bdsol-aci32-leaf3
Ingress: Eth1/12 (Po1) Egress: Eth1/12 (Po1) Vnid: 11364
2019-10-02 23:54:39,417 INFO     ftriage:     main:242  ingress encap string vlan-2551
2019-10-02 23:54:41,962 INFO     ftriage:     main:271  Building ingress BD(s), Ctx
2019-10-02 23:54:44,765 INFO     ftriage:     main:294  Ingress BD(s) Prod:VRF1:l3out-BGP:vlan-
2551
```

```
2019-10-02 23:54:44,766 INFO     ftriage:     main:301  Ingress Ctx: Prod:VRF1
2019-10-02 23:54:44,875 INFO     ftriage:   pktrec:490  bdsol-aci32-leaf3: Collecting transient
losses snapshot for LC module: 1
2019-10-02 23:55:02,905 INFO     ftriage:     nxos:1404 bdsol-aci32-leaf3: nxos matching rule
id:4341 scope:34 filter:5
2019-10-02 23:55:04,525 INFO     ftriage:     main:522  Computed egress encap string vlan-2501
2019-10-02 23:55:04,526 INFO     ftriage:     main:313  Building egress BD(s), Ctx
2019-10-02 23:55:06,390 INFO     ftriage:     main:331  Egress Ctx Prod:VRF1
2019-10-02 23:55:06,390 INFO     ftriage:     main:332  Egress BD(s): Prod:BD1
2019-10-02 23:55:13,571 INFO     ftriage:     main:933  SIP 172.16.2.1 DIP 192.168.1.1
2019-10-02 23:55:13,572 INFO     ftriage:  unicast:973  bdsol-aci32-leaf3: <- is ingress node
2019-10-02 23:55:16,159 INFO     ftriage:  unicast:1202 bdsol-aci32-leaf3: Dst EP is local
2019-10-02 23:55:18,949 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: EP if(Po1) same as
egr if(Po1)
2019-10-02 23:55:20,126 INFO     ftriage:     misc:657  bdsol-aci32-leaf3:
DMAC(00:22:BD:F8:19:FF) same as RMAC(00:22:BD:F8:19:FF)
2019-10-02 23:55:20,126 INFO     ftriage:     misc:659  bdsol-aci32-leaf3: L3 packet getting
routed/bounced in SUG
2019-10-02 23:55:20,395 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: Dst IP is present in
SUG L3 tbl
2019-10-02 23:55:20,687 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: RW seg_id:11364 in
SUG same as EP segid:11364
2019-10-02 23:55:26,982 INFO     ftriage:     main:961  Packet is Exiting fabric with peer-
device: bdsol-aci32-n3k-3 and peer-port: Ethernet1/16
```

This flow is allowed (unexpected) by zoning-rule 4341. The zoning-rules must now be analyzed to understand why.

Datapath validation — zoning-rules

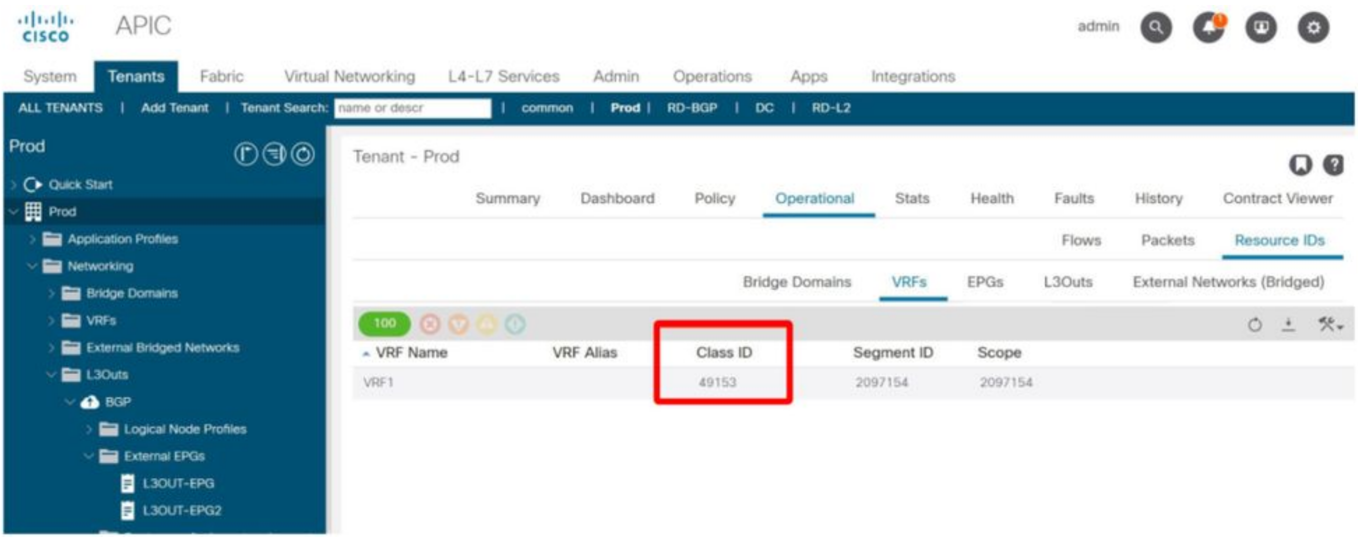
The zoning-rules corresponding to the last 2 tests are below:

- Expected — flow hits zoning-rule line 4336 (ICMP2 contract).
- Unexpected — flow hits zoning-rule line 4341 (ICMP1 contract).

```
+---------+--------+--------+----------+---------+---------+---------+-------+----------+-------
---------------+
| Rule ID | SrcEPG | DstEPG | FilterID |   Dir   |  operSt |  Scope  |  Name |  Action  |
Priority       |
+---------+--------+--------+----------+---------+---------+---------+-------+----------+-------
---------------+
|   4326  |   0    |   0    | implicit | uni-dir | enabled | 2097154 |       | deny,log |
any_any_any(21)    |
|   4335  |   0    | 16387  | implicit | uni-dir | enabled | 2097154 |       |  permit  |
any_dest_any(16)   |
|   4334  |   0    |   0    | implarp  | uni-dir | enabled | 2097154 |       |  permit  |
any_any_filter(17)  |
|   4333  |   0    |   15   | implicit | uni-dir | enabled | 2097154 |       | deny,log |
any_vrf_any_deny(22) |
|   4332  |   0    | 16386  | implicit | uni-dir | enabled | 2097154 |       |  permit  |
any_dest_any(16)   |
|   4339  | 32770  |   15   |    5     | uni-dir | enabled | 2097154 | ICMP2 |  permit  |
fully_qual(7)     |
|   4341  | 49153  | 32770  |    5     | uni-dir | enabled | 2097154 | ICMP2 |  permit  |
fully_qual(7)     |
|   4337  | 32771  |   15   |    5     | uni-dir | enabled | 2097154 | ICMP1 |  permit  |
fully_qual(7)     |
|   4336  | 49153  | 32771  |    5     | uni-dir | enabled | 2097154 | ICMP1 |  permit  |
fully_qual(7)     |
+---------+--------+--------+---------+---------+---------+---------+-------+---------+-------
---------------+
```

Both flows derive the src pcTag of 49153. This is the pcTag of the VRF. This can be verified in the UI:

Verification the pcTag of the VRF



VRF Name	VRF Alias	Class ID	Segment ID	Scope
VRF1		49153	2097154	2097154

The following happens when the 0.0.0.0/0 prefix is in use with an L3Out:

- Traffic from an internal EPG to an L3Out EPG with 0.0.0.0/0 will derive a destination pcTag of 15.
- Traffic from an L3Out EPG with 0.0.0.0/0 to an ACI internal EPG will derive a source pcTag of the VRF (49153).

The contract_parser script gives a holistic view of the zoning-rules:

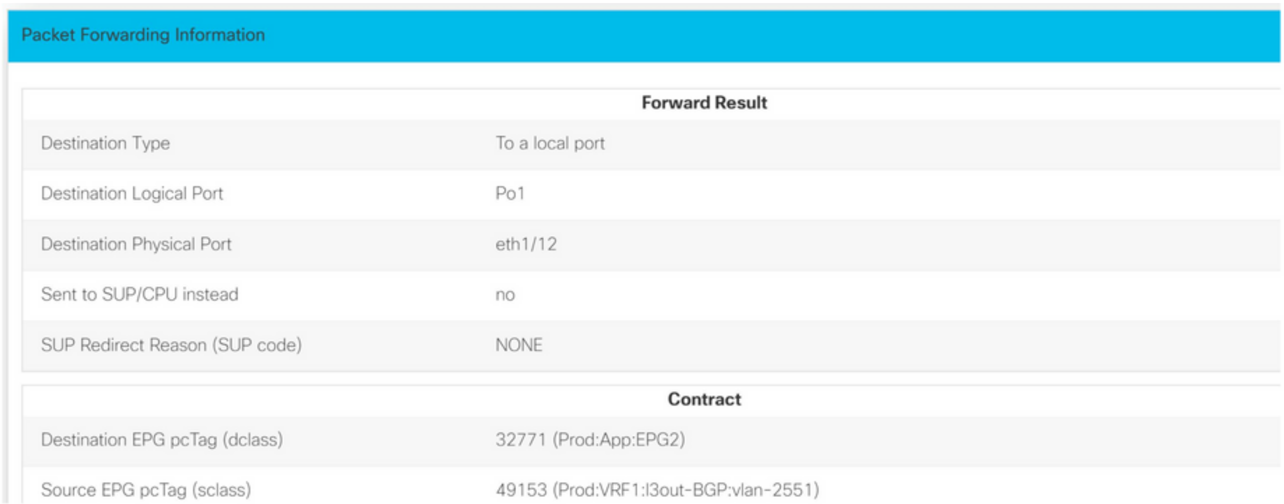
```
bdsol-aci32-leaf3# contract_parser.py --vrf Prod:VRF1
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]
[7:4339] [vrf:Prod:VRF1] permit ip icmp tn-Prod/ap-App/epg-EPG1(32770) pfx-0.0.0.0/0(15)
[contract:uni/tn-Prod/brc-ICMP2] [hit=0]
[7:4337] [vrf:Prod:VRF1] permit ip icmp tn-Prod/ap-App/epg-EPG2(32771) pfx-0.0.0.0/0(15)
[contract:uni/tn-Prod/brc-ICMP] [hit=0]
[7:4341] [vrf:Prod:VRF1] permit ip icmp tn-Prod/vrf-VRF1(49153) tn-Prod/ap-App/epg-EPG1(32770)
[contract:uni/tn-Prod/brc-ICMP2] [hit=270]
[7:4336] [vrf:Prod:VRF1] permit ip icmp tn-Prod/vrf-VRF1(49153) tn-Prod/ap-App/epg-EPG2(32771)
[contract:uni/tn-Prod/brc-ICMP] [hit=0]
```

Confirming pcTag used by the packet using ELAM Assistant app

The ELAM Assistant App gives another method to confirm the source and destination pcTag of live traffic flows.

The screen shot below shows the ELAM result for traffic from pcTag 32771 to pcTag 49153.

ELAM Assistant app output for src 32771 to dst 49153



Packet Forwarding Information

Forward Result	
Destination Type	To a local port
Destination Logical Port	Po1
Destination Physical Port	eth1/12
Sent to SUP/CPU instead	no
SUP Redirect Reason (SUP code)	NONE

Contract	
Destination EPG pcTag (dclass)	32771 (Prod:App:EPG2)
Source EPG pcTag (sclass)	49153 (Prod:VRF1:l3out-BGP:vlan-2551)

Conclusion

The usage of 0.0.0.0/0 must be carefully tracked within a VRF as every L3Out using that subnet will inherit the contracts applied to every other L3Out using it. This will likely lead to unplanned permit flows.

Shared L3Out

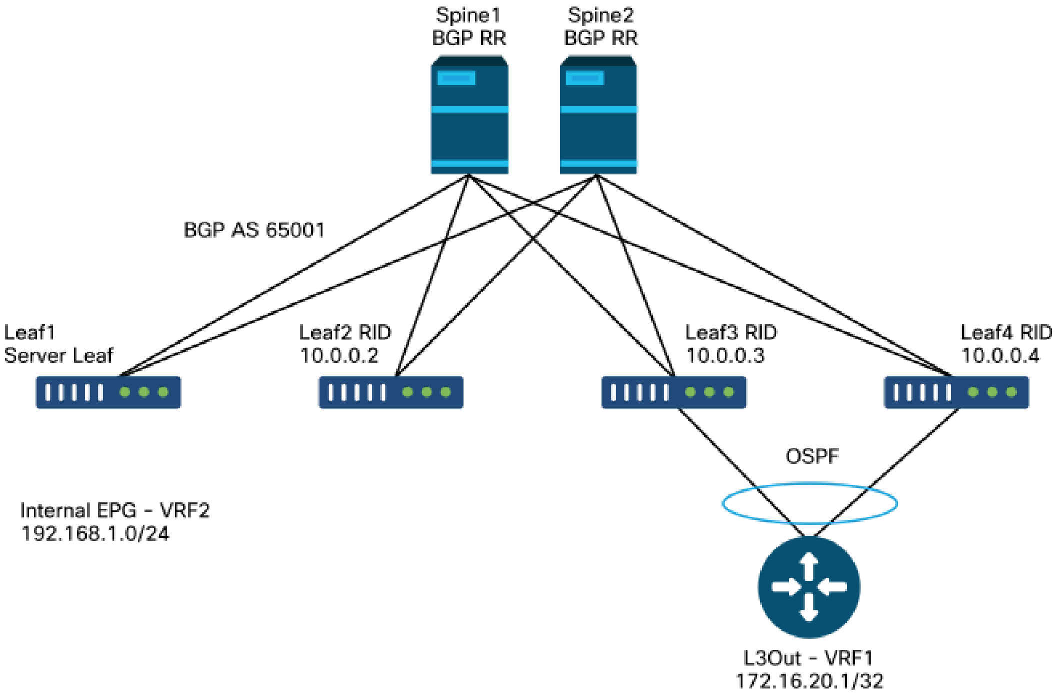
Overview

This section will discuss how to troubleshoot route-advertisement in Shared L3Out configurations. The term 'Shared L3Out' refers to the scenario where an L3Out is in one VRF but an internal EPG having a contract with the L3Out is in a another VRF. With Shared L3Outs, the route-leaking is being done internally to the ACI fabric.

This section will not go into deep detail about security policy troubleshooting. For that refer to the "Security Policies" chapter of this book. This section will also not talk in detail about External Policy Prefix classification for security purposes. Refer to the "Contract and L3Out" section in the "external forwarding" chapter.

This section uses the following topology for our examples.

Shared L3Out topology



At a high level, the following configurations must be in place for a Shared L3Out to function:

- An L3Out subnet must be configured with the 'Shared Route Control Subnet' scope to leak external routes into internal VRFs. 'Aggregate Shared' option can also be selected to leak all routes that are more specific than the configured subnet.
- An L3Out subnet must be configured with the 'Shared Security Import Subnet' scope to program the security policies necessary to allow communication through this L3Out.
- The internal BD subnet must be set to 'Shared between VRFs' and 'Advertise Externally' to program the BD subnet in the external VRF and advertise it.
- A 'tenant' or 'global' scope contract must be configured between the internal EPG and the external EPG of the shared L3Out.

The next section will go into detail about how leaked routes are advertised and learned in ACI.

Shared L3Out workflow — learning external routes

This section will outline the path of a learned external route as it is advertised into the fabric.

External route as seen on the border leaf

This command will show the external route learned from OSPF:

```
leaf103# show ip route 172.16.20.1/32 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
```

```
'%' in via output denotes VRF

172.16.20.1/32, ubest/mbest: 1/0
    *via 10.10.34.3, vlan347, [110/20], 03:59:59, ospf-default, type-2
```

Next, the route must be imported into BGP. By default, all external routes should be imported into BGP.

BGP verifications on the border leaf

The route must be in the BGP VPNv4 Address-family with a route-target to be distributed throughout the fabric. The route-target is a BGP extended community exported by the external VRF and imported by any internal VRFs that needs to receive the path.

Next, verify the route-target that is being exported by the external VRF on the BL.

```
leaf103# show bgp process vrf Prod:Vrf1

Information regarding configured VRFs:

BGP Information for VRF Prod:Vrf1
VRF Type                       : System
VRF Id                         : 85
VRF state                      : UP
VRF configured                 : yes
VRF refcount                   : 1
VRF VNID                       : 2392068
Router-ID                      : 10.0.0.3
Configured Router-ID           : 10.0.0.3
Confed-ID                      : 0
Cluster-ID                     : 0.0.0.0
MSITE Cluster-ID               : 0.0.0.0
No. of configured peers        : 1
No. of pending config peers    : 0
No. of established peers       : 0
VRF RD                         : 101:2392068
VRF EVPN RD                    : 101:2392068

...

    Wait for IGP convergence is not configured
    Export RT list:
        65001:2392068
    Import RT list:
        65001:2392068
    Label mode: per-prefix
```

The above output shows that any paths advertised from the external VRF into VPNv4 should receive a route-target of 65001:2392068.

Next, verify the bgp path:

```
leaf103# show bgp ipv4 unicast 172.16.20.1/32 vrf Prod:Vrf1
BGP routing table information for VRF Prod:Vrf1, address family IPv4 Unicast
BGP routing table entry for 172.16.20.1/32, version 30 dest ptr 0xa6f25ad0
Paths: (2 available, best #1)
Flags: (0x80c0002 00000000) on xmit-list, is not in urib, exported
```

```
  vpn: version 17206, (0x100002) on xmit-list
Multipath: eBGP iBGP

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: redist 0x408 0x1 ref 0 adv path ref 2, path is valid, is best path
  AS-Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (10.0.0.3)
      Origin incomplete, MED 20, localpref 100, weight 32768
      Extcommunity:
          RT:65001:2392068
          VNID:2392068
          COST:pre-bestpath:162:110

  VRF advertise information:
  Path-id 1 not advertised to any peer

  VPN AF advertise information:
  Path-id 1 advertised to peers:
    10.0.64.64         10.0.72.66
  Path-id 2 not advertised to any peer
```

The above output shows that the path has the correct route-target. The VPNv4 path can also be verified by using 'show bgp vpnv4 unicast 172.16.20.1 vrf overlay-1' command.

Verifications on the server leaf

For the internal EPG leaf to install the BL-advertised route, it must import the route-target (mentioned above) into the internal VRF. The internal VRF's BGP process can be checked to validate this:

```
leaf101# show bgp process vrf Prod:Vrf2

Information regarding configured VRFs:

BGP Information for VRF Prod:Vrf2
VRF Type                       : System
VRF Id                         : 54
VRF state                      : UP
VRF configured                 : yes
VRF refcount                   : 0
VRF VNID                       : 2916352
Router-ID                      : 192.168.1.1
Configured Router-ID           : 0.0.0.0
Confed-ID                      : 0
Cluster-ID                     : 0.0.0.0
MSITE Cluster-ID               : 0.0.0.0
No. of configured peers        : 0
No. of pending config peers    : 0
No. of established peers       : 0
VRF RD                         : 102:2916352
VRF EVPN RD                    : 102:2916352
...
    Wait for IGP convergence is not configured
    Import route-map 2916352-shared-svc-leak
    Export RT list:
        65001:2916352
    Import RT list:
        65001:2392068
        65001:2916352
```

The above output shows the internal VRF importing the route-target that is exported by the external VRF. Additionally, there is an 'Import Route-Map' that is referenced. The import route-map includes the specific prefixes that are defined in the shared L3Out with the 'Shared Route Control Subnet' flag.

The route-map contents can be checked to ensure it includes the external prefix:

```
leaf101# show route-map 2916352-shared-svc-leak
route-map 2916352-shared-svc-leak, deny, sequence 1
 Match clauses:
   pervasive: 2
 Set clauses:
route-map 2916352-shared-svc-leak, permit, sequence 2
 Match clauses:
   extcommunity  (extcommunity-list filter): 2916352-shared-svc-leak
 Set clauses:
route-map 2916352-shared-svc-leak, permit, sequence 1000
 Match clauses:
   ip address prefix-lists: IPv4-2392068-16387-5511-2916352-shared-svc-leak
   ipv6 address prefix-lists: IPv6-deny-all
 Set clauses:
a-leaf101# show ip prefix-list IPv4-2392068-16387-5511-2916352-shared-svc-leak
ip prefix-list IPv4-2392068-16387-5511-2916352-shared-svc-leak: 1 entries
  seq 1 permit 172.16.20.1/32
```

The above output shows the import route-map which includes the subnet to be imported.

The final verifications include checking that the route is in the BGP table and that it is installed in the routing table.

BGP table on server leaf:

```
leaf101# show bgp ipv4 unicast 172.16.20.1/32 vrf Prod:Vrf2
BGP routing table information for VRF Prod:Vrf2, address family IPv4 Unicast
BGP routing table entry for 172.16.20.1/32, version 3 dest ptr 0xa763add0
Paths: (2 available, best #1)
Flags: (0x08001a 00000000) on xmit-list, is in urib, is best urib route, is in HW
  vpn: version 10987, (0x100002) on xmit-list
Multipath: eBGP iBGP

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: internal 0xc0000018 0x40 ref 56506 adv path ref 2, path is valid, is best path
             Imported from 10.0.72.64:5:172.16.20.1/32
  AS-Path: NONE, path sourced internal to AS
    10.0.72.64 (metric 3) from 10.0.64.64 (192.168.1.102)
      Origin incomplete, MED 20, localpref 100, weight 0
      Received label 0
      Received path-id 1
      Extcommunity:
          RT:65001:2392068
          VNID:2392068
          COST:pre-bestpath:162:110
      Originator: 10.0.72.64 Cluster list: 192.168.1.102
```

The route is imported into the internal VRF BGP table and has the expected route-target.

The installed routes can be verified:

```
leaf101# vsh -c "show ip route 172.16.20.1/32 detail vrf Prod:Vrf2"
IP Route Table for VRF "Prod:Vrf2"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%' in via output denotes VRF
172.16.20.1/32, ubest/mbest: 2/0
   *via 10.0.72.64%overlay-1, [200/20], 01:00:51, bgp-65001, internal, tag 65001 (mpls-vpn)
        MPLS[0]: Label=0 E=0 TTL=0 S=0 (VPN)
        client-specific data: 548
        recursive next hop: 10.0.72.64/32%overlay-1
        extended route information: BGP origin AS 65001 BGP peer AS 65001 rw-vnid: 0x248004
table-id: 0x36 rw-mac: 0
   *via 10.0.72.67%overlay-1, [200/20], 01:00:51, bgp-65001, internal, tag 65001 (mpls-vpn)
        MPLS[0]: Label=0 E=0 TTL=0 S=0 (VPN)
        client-specific data: 54a
        recursive next hop: 10.0.72.67/32%overlay-1
        extended route information: BGP origin AS 65001 BGP peer AS 65001 rw-vnid: 0x248004
table-id: 0x36 rw-mac: 0
```

The above output uses a specific 'vsh -c' command to get the 'detail' output. The 'detail' flag includes the rewrite VXLAN VNID. This is the VXLAN VNID of the external VRF. When the BL receives dataplane traffic with this VNID, it knows to make the forwarding decision in the external VRF.

The rw-vnid value is in hex, so converting to decimal will get the VRF VNID of 2392068. Search for the corresponding VRF using 'show system internal epm vrf all | grep 2392068' on the leaf. A global search can be performed on an APIC using the 'moquery -c fvCtx -f 'fv.Ctx.seg=="2392068"' command.

The next-hop's IP should also point to the BL PTEPs and the '%overlay-1' indicates that the route lookup for the next-hop is in the overlay VRF.

Shared L3Out workflow — advertising internal routes

As in previous sections, advertising internal BD subnets out a shared L3Out is handled by the following:

- The BD subnet (internal VRF) is installed on the BL (external VRF) as a static route. This static route deployment is a result of the contract relationship between the internal EPG and the L3Out.
- The static route is redistributed into the external protocol when the 'Advertised Externally' scope is set on the BD subnet.

Verify BD static route on the BL

```
leaf103# vsh -c "show ip route 192.168.1.0 detail vrf Prod:Vrf1"
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
```

```
'%' in via output denotes VRF

192.168.1.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.120.34%overlay-1, [1/0], 00:55:27, static, tag 4294967292
         recursive next hop: 10.0.120.34/32%overlay-1
         vrf crossing information:  VNID:0x2c8000 ClassId:0 Flush#:0
```

Notice that in the above output the VNID of the internal VRF is set for the rewrite. The next-hop is also set to the proxy-v4-anycast address.

The above route is advertised externally through the same route-maps that are demonstrated in the "Route Advertisement" section.

If a BD subnet is set to 'Advertise Externally', it is redistributed into **every L3Out's external protocol** that the internal EPG has a contract relationship with.

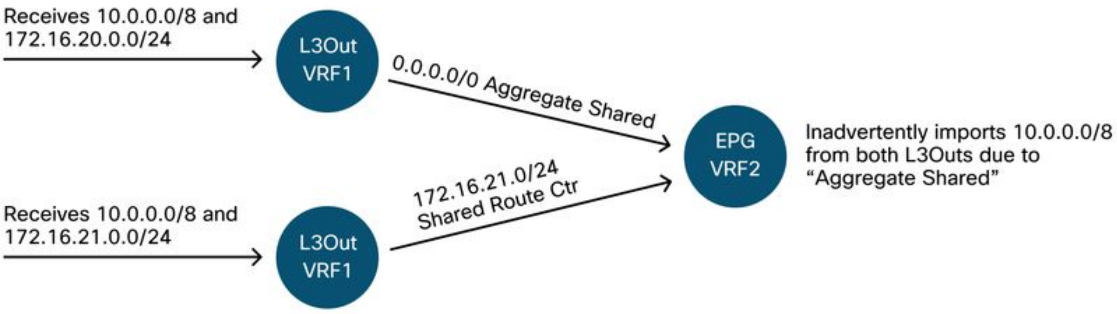
Shared L3Out troubleshooting scenario — unexpected route leaking

This scenario has multiple L3Outs in the external VRF and an internal EPG is receiving a route from an L3Out where the network **is not** defined with the 'shared' scope options.

Usage of 'Aggregate Shared'

Consider the following figure:

Unexpected route leak



The BGP import-map with the prefix-list programmed from the **'Shared Route Control Subnet'** flags is applied at the VRF level. If one L3Out in VRF1 has a subnet with 'Shared Route Control Subnet', then all routes received on L3Outs within VRF1 that match this Shared Route Control Subnet will get imported into VRF2.

The above design can result in unexpected traffic flows. If there are no contracts between the internal EPG and the unexpected advertising L3Out EPG, then there will be traffic drops.