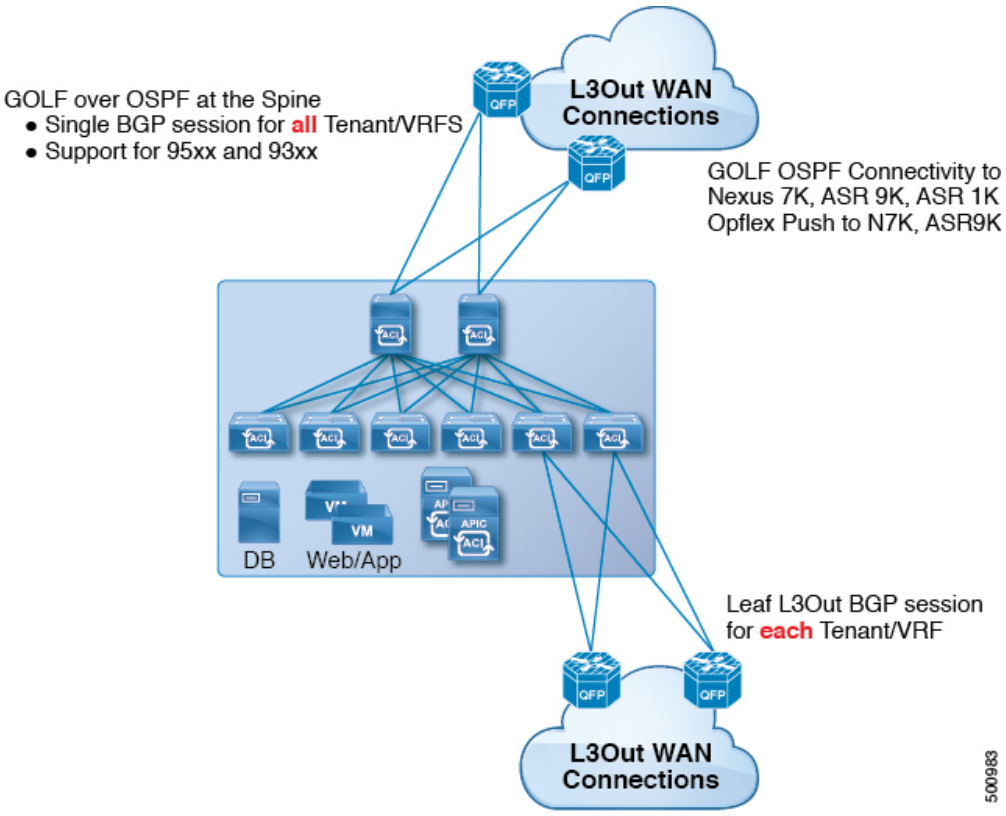
# Cisco ACI GOLF


- Cisco ACI GOLF , on page 1
- Distributing BGP EVPN Type-2 Host Routes to a DCIG, on page 15


# Cisco ACI GOLF

The Cisco ACI GOLF feature (also known as Layer 3 EVPN Services for Fabric WAN) enables much more efficient and scalable ACI fabric WAN connectivity. It uses the BGP EVPN protocol over OSPF for WAN routers that are connected to spine switches.

***Figure 1: Cisco ACI GOLF Topology***



All tenant WAN connections use a single session on the spine switches where the WAN routers are connected. This aggregation of tenant BGP sessions towards the Data Center Interconnect Gateway (DCIG) improves control plane scale by reducing the number of tenant BGP sessions and the amount of configuration required for all of them. The network is extended out using Layer 3 subinterfaces configured on spine fabric ports. Transit routing with shared services using GOLF is not supported.

A Layer 3 external outside network (`L3extOut`) for GOLF physical connectivity for a spine switch is specified under the `infra` tenant, and includes the following:

- `LNodeP` (`l3extInstP` is not required within the L3Out in the infra tenant. )
- A provider label for the `L3extOut` for GOLF in the infra tenant.
- OSPF protocol policies
- BGP protocol policies

All regular tenants use the above-defined physical connectivity. The `L3extOut` defined in regular tenants requires the following:

- An `l3extInstP` (EPG) with subnets and contracts. The scope of the subnet is used to control import/export route control and security policies. The bridge domain subnet must be set to advertise externally and it must be in the same VRF as the application EPG and the GOLF L3Out EPG.
- Communication between the application EPG and the GOLF L3Out EPG is governed by explicit contracts (not Contract Preferred Groups).
- An `l3extConsLbl` consumer label that must be matched with the same provider label of an `L3Out` for GOLF in the `infra` tenant. Label matching enables application EPGs in other tenants to consume the `LNodeP` external `L3Out` EPG.
- The BGP EVPN session in the matching provider `L3extOut` in the `infra` tenant advertises the tenant routes defined in this `L3Out`.

# Guidelines and Limitations for Cisco ACI GOLF

Observe the following Cisco ACI GOLF guidelines and limitations:

- GOLF does not support shared services.
- GOLF does not support transit routing.
- GOLF routers must advertise at least one route to Cisco Application Centric Infrastructure (ACI) to accept traffic. No tunnel is created between leaf switches and the external routers until Cisco ACI receives a route from the external routers.
- All Cisco Nexus 9000 Series Cisco ACI-mode switches and all of the Cisco Nexus 9500 platform Cisco ACI-mode switch line cards and fabric modules support GOLF. With Cisco APIC, release 3.1(x) and higher, this includes the N9K-C9364C switch.
- At this time, only a single GOLF provider policy can be deployed on spine switch interfaces for the whole fabric.
- Up to Cisco APIC release 2.0(2), GOLF is not supported with Cisco ACI Multi-Pod. In release 2.0 (2), the two features are supported in the same fabric only over Cisco Nexus 9000 switches without "EX" on

the end of the switch name; for example, N9K-9312TX. Since the 2.1(1) release, the two features can be deployed together over all the switches used in the Cisco ACI Multi-Pod and EVPN topologies.

- When configuring GOLF on a spine switch, wait for the control plane to converge before configuring GOLF on another spine switch.
- A spine switch can be added to multiple provider GOLF outside networks (GOLF L3Outs), but the provider labels have to be different for each GOLF L3Out. Also, in this case, the OSPF Area has to be different on each of the `L3extOut`s and use different loopback addresses.
- The BGP EVPN session in the matching provider `L3Out` in the `infra` tenant advertises the tenant routes defined in this `L3extOut`.
- When deploying three GOLF Outs, if only 1 has a provider/consumer label for GOLF, and 0/0 export aggregation, Cisco APIC will export all routes. This is the same as existing `L3extOut` on leaf switches for tenants.
- If you have an ERSPAN session that has a SPAN destination in a VRF instance, the VRF instance has GOLF enabled, and the ERSPAN source has interfaces on a spine switch, the transit prefix gets sent from a non-GOLF L3Out to the GOLF router with the wrong BGP next-hop.
- If there is direct peering between a spine switch and a data center interconnect (DCI) router, the transit routes from leaf switches to the ASR have the next hop as the PTEP of the leaf switch. In this case, define a static route on the ASR for the TEP range of that Cisco ACI pod. Also, if the DCI is dual-homed to the same pod, then the precedence (administrative distance) of the static route should be the same as the route received through the other link.
- The default `bgpPeerPfxPol` policy restricts routes to 20,000. For Cisco ACI WAN Interconnect peers, increase this as needed.
- In a deployment scenario where there are two `L3extOut`s on one spine switch, and one of them has the provider label `prov1` and peers with the DCI 1, the second `L3extOut` peers with DCI 2 with provider label `prov2`. If the tenant VRF instance has a consumer label pointing to any 1 of the provider labels (either prov1 or prov2), the tenant route will be sent out both DCI 1 and DCI 2.
- When aggregating GOLF OpFlex VRF instances, the leaking of routes cannot occur in the Cisco ACI fabric or on the GOLF device between the GOLF OpFlex VRF instance and any other VRF instance in the system. An external device (not the GOLF router) must be used for the VRF leaking.

**Note** Cisco ACI does not support IP fragmentation. Therefore, when you configure Layer 3 Outside (L3Out) connections to external routers, or Multi-Pod connections through an Inter-Pod Network (IPN), it is recommended that the interface MTU is set appropriately on both ends of a link. On some platforms, such as Cisco ACI, Cisco NX-OS, and Cisco IOS, the configurable MTU value does not take into account the Ethernet headers (matching IP MTU, and excluding the 14-18 Ethernet header size), while other platforms, such as IOS-XR, include the Ethernet header in the configured MTU value. A configured value of 9000 results in a max IP packet size of 9000 bytes in Cisco ACI, Cisco NX-OS, and Cisco IOS, but results in a max IP packet size of 8986 bytes for an IOS-XR untagged interface.

For the appropriate MTU values for each platform, see the relevant configuration guides.

We highly recommend that you test the MTU using CLI-based commands. For example, on the Cisco NX-OS CLI, use a command such as `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1`.

# Using Shared GOLF Connections Between Multi-Site Sites

## APIC GOLF Connections Shared by Multi-Site Sites

For APIC Sites in a Multi-Site topology, if stretched VRFs share GOLF connections, follow these guidelines to avoid the risk of cross-VRF traffic issues.

### Route Target Configuration between the Spine Switches and the DCI

There are two ways to configure EVPN route targets (RTs) for the GOLF VRFs: Manual RT and Auto RT. The route target is synchronized between ACI spines and DCIs through OpFlex. Auto RT for GOLF VRFs has the Fabric ID embedded in the format: – *ASN*: [*FabricID*] *VNID*

If two sites have VRFs deployed as in the following diagram, traffic between the VRFs can be mixed.

| Site 1 | Site 2 |
|---|---|
| ASN: 100, Fabric ID: 1 | ASN: 100, Fabric ID: 1 |
| VRF A: VNID 1000<br>Import/Export Route Target: 100: [1] 1000 | VRF A: VNID 2000<br>Import/Export Route Target: 100: [1] 2000 |
| VRF B: VNID 2000<br>Import/Export Route Target: 100: [1] 2000 | VRF B: VNID 1000<br>Import/Export Route Target: 100: [1] 1000 |

### Route Maps Required on the DCI

Since tunnels are not created across sites when transit routes are leaked through the DCI, the churn in the control plane must be reduced as well. EVPN type-5 and type-2 routes sent from GOLF spine in one site towards the DCI should not be sent to GOLF spine in another site. This can happen when the DCI to spine switches have the following types of BGP sessions:

Site1 — IBGP ---- DCI ---- EBGP ---- Site2

Site1 — EBGP ---- DCI ---- IBGP ---- Site2

Site1 — EBGP ---- DCI ---- EBGP ---- Site2

Site1 — IBGP RR client ---- DCI (RR)---- IBGP ---- Site2

To avoid this happening on the DCI, route maps are used with different BGP communities on the inbound and outbound peer policies.

When routes are received from the GOLF spine at one site, the outbound peer policy towards the GOLF spine at another site filters the routes based on the community in the inbound peer policy. A different outbound peer policy strips off the community towards the WAN. All the route-maps are at peer level.

## Recommended Shared GOLF Configuration Using the NX-OS Style CLI

Use the following steps to configure route maps and BGP to avoid cross-VRF traffic issues when sharing GOLF connections with a DCI between multiple APIC sites that are managed by Multi-Site.

## Procedure

**Step 1** Configure the inbound route map

**Example:**

```
Inbound peer policy to attach community:
 route-map multi-site-in permit 10
  set community 1:1 additive
```

**Step 2** Configure the outbound peer policy to filter routes based on the community in the inbound peer policy.

**Example:**

```
ip community-list standard test-com permit 1:1
route-map multi-site-out deny 10
  match community test-com exact-match
route-map multi-site-out permit 11
```

**Step 3** Configure the outbound peer policy to filter the community towards the WAN.

**Example:**

```
ip community-list standard test-com permit 1:1
route-map multi-site-wan-out permit 11
  set comm-list test-com  delete
```

**Step 4** Configure BGP.

**Example:**

```
router bgp 1
  address-family l2vpn evpn
  neighbor 11.11.11.11 remote-as 1
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-map multi-site-in in
  neighbor 13.0.0.2 remote-as 2
    address-family l2vpn evpn
      send-community both
      route-map multi-site-out out
```

# Configuring ACI GOLF Using the GUI

The following steps describe how to configure infra GOLF services that any tenant network can consume.

### Procedure

**Step 1** On the menu bar, click **Tenants**, then click **infra** to select the infra tenant.

**Step 2** In the **Navigation** pane, expand the **Networking** option and perform the following actions:

a) Right-click **External Routed Networks** and click **Create Routed Outside for EVPN** to open the wizard.

b) In the **Name** field, enter a name for the policy.

c) In the **Route Target** field, choose whether to use automatic or explicit policy-governed BGP route target filtering policy:

- **Automatic** - Implements automatic BGP route-target filtering on VRFs associated with this routed outside configuration.
- **Explicit** - Implements route-target filtering through use of explicitly configured BGP route-target policies on VRFs associated with this routed outside configuration.

  **Note**
  Explicit route target policies are configured in the **BGP Route Target Profiles** table on the **BGP Page** of the **Create VRF Wizard**. If you select the **Automatic** option the in **Route Target** field, configuring explicit route target policies in the **Create VRF Wizard** might cause BGP routing disruptions.

**Note**
Explicit route target policies are configured in the **BGP Route Target Profiles** table on the **BGP Page** of the **Create VRF Wizard**. If you select the **Automatic** option the in **Route Target** field, configuring explicit route target policies in the **Create VRF Wizard** might cause BGP routing disruptions.

d) Complete the configuration options according to the requirements of your Layer 3 connection.

**Note**
In the protocol check boxes area, assure that both **BGP** and **OSPF** are checked. GOLF requires both BGP and OSPF.

e) Click **Next** to display the **Nodes and Interfaces Protocol Profile** tab.

f) In the **Define Routed Outside** section, in the **Name** field, enter a name.

g) In the **Spines** table, click **+** to add a node entry.

h) In the **Node ID** drop-down list, choose a spine switch node ID.

i) In the **Router ID** field, enter the router ID.

j) In the **Loopback Addresses**, in the IP field, enter the IP address. Click **Update**.

k) In the OSPF Profile for Sub-interfaces/Routed Sub-Interfaces section in the Name field enter the name of the OSPF Profile for Sun-Interfaces.

l) Click **OK**.

**Note**
The wizard creates a **Logical Node Profile> Configured Nodes> Node Association** profile, that set the **Extend Control Peering** field to enabled.

**Step 3** In the ***infra*** > **Networking** > **External Routed Networks** section of the **Navigation** pane, click to select the Golf policy just created. Enter a **Provider Label**, (for example, *golf*) and click **Submit**.

**Step 4** In the **Navigation** pane for any tenant, expand the ***tenant_name*** > **Networking** and perform the following actions:

a) Right-click **External Routed Networks** and click **Create Routed Outside** to open the wizard.
b) In the **Identity** dialog box, in the **Name** field, enter a name for the policy.
c) Complete the configuration options according to the requirements of your Layer 3 connection.

**Note**
In the protocol check boxes area, assure that both **BGP** and **OSPF** are checked. GOLF requires both BGP and OSPF.

d) Assign a **Consumer Label**. In this example, use *golf* (that was just created above.
e) Click **Next**.
f) Configure the External EPG Networks dialog box, and click **Finish** to deploy the policy.

# Cisco ACI GOLF Configuration Example, Using the NX-OS Style CLI

These examples show the CLI commands to configure GOLF Services, which uses the BGP EVPN protocol over OSPF for WAN routers that are connected to spine switches.

### Configuring the infra Tenant for BGP EVPN

The following example shows how to configure the infra tenant for BGP EVPN, including the VLAN domain, VRF, Interface IP addressing, and OSPF:

```
configure
  vlan-domain evpn-dom dynamic
  exit
  spine 111
       # Configure  Tenant Infra VRF overlay-1 on the spine.
    vrf context tenant infra vrf overlay-1
        router-id 10.10.3.3
        exit

    interface ethernet 1/33
         vlan-domain member golf_dom
         exit
    interface ethernet 1/33.4
         vrf member tenant infra vrf overlay-1
         mtu 1500
         ip address 5.0.0.1/24
         ip router ospf default area 0.0.0.150
         exit
    interface ethernet 1/34
         vlan-domain member golf_dom
        exit
    interface ethernet 1/34.4
        vrf member tenant infra vrf overlay-1
        mtu 1500
        ip address 2.0.0.1/24
        ip router ospf default area 0.0.0.200
       exit
```

```
router ospf default
   vrf member tenant infra vrf overlay-1
       area 0.0.0.150 loopback 10.10.5.3
       area 0.0.0.200 loopback 10.10.4.3
       exit
   exit
```

### Configuring BGP on the Spine Node

The following example shows how to configure BGP to support BGP EVPN:

```
Configure
spine 111
router bgp 100
    vrf member tenant infra vrf overlay- 1
         neighbor 10.10.4.1 evpn
             label golf_aci
             update-source loopback 10.10.4.3
             remote-as 100
             exit
         neighbor 10.10.5.1 evpn
             label golf_aci2
             update-source loopback 10.10.5.3
             remote-as 100
             exit
    exit
exit
```

### Configuring a Tenant for BGP EVPN

The following example shows how to configure a tenant for BGP EVPN, including a gateway subnet which will be advertised through a BGP EVPN session:

```
configure
  tenant sky
    vrf context vrf_sky
      exit
    bridge-domain bd_sky
      vrf member vrf_sky
      exit
    interface bridge-domain bd_sky
      ip address 59.10.1.1/24
      exit
    bridge-domain bd_sky2
      vrf member vrf_sky
      exit
    interface bridge-domain bd_sky2
      ip address 59.11.1.1/24
      exit
    exit
```

### Configuring the BGP EVPN Route Target, Route Map, and Prefix EPG for the Tenant

The following example shows how to configure a route map to advertise bridge-domain subnets through BGP EVPN.

```
configure
spine 111
    vrf context tenant sky vrf vrf_sky
        address-family ipv4 unicast
```

```
            route-target export 100:1
            route-target import 100:1
             exit

        route-map rmap
            ip prefix-list p1 permit 11.10.10.0/24
            match bridge-domain bd_sky
                exit
            match prefix-list p1
                exit

         evpn export map rmap label golf_aci

          route-map rmap2
           match bridge-domain bd_sky
               exit
           match prefix-list p1
              exit
          exit

         evpn export map rmap label golf_aci2

    external-l3 epg l3_sky
      vrf member vrf_sky
      match ip 80.10.1.0/24
      exit
```

# Configuring GOLF Using the REST API

### Procedure

**Step 1** The following example shows how to deploy nodes and spine switch interfaces for GOLF, using the REST API:

**Example:**

```
POST
https://192.0.20.123/api/mo/uni/golf.xml
```

**Step 2** The XML below configures the spine switch interfaces and infra tenant provider of the GOLF service. Include this XML structure in the body of the POST message.

**Example:**

```
<l3extOut descr="" dn="uni/tn-infra/out-golf" enforceRtctrl="export,import"
    name="golf"
    ownerKey="" ownerTag="" targetDscp="unspecified">
    <l3extRsEctx tnFvCtxName="overlay-1"/>
    <l3extProvLbl descr="" name="golf"
         ownerKey="" ownerTag="" tag="yellow-green"/>
    <l3extLNodeP configIssues="" descr=""
         name="bLeaf" ownerKey="" ownerTag=""
         tag="yellow-green" targetDscp="unspecified">
         <l3extRsNodeL3OutAtt rtrId="10.10.3.3" rtrIdLoopBack="no"
             tDn="topology/pod-1/node-111">
             <l3extInfraNodeP descr="" fabricExtCtrlPeering="yes" name=""/>
             <l3extLoopBackIfP addr="10.10.3.3" descr="" name=""/>
         </l3extRsNodeL3OutAtt>
         <l3extRsNodeL3OutAtt rtrId="10.10.3.4" rtrIdLoopBack="no"
             tDn="topology/pod-1/node-112">
```

```
<l3extInfraNodeP descr="" fabricExtCtrlPeering="yes" name=""/>
<l3extLoopBackIfP addr="10.10.3.4" descr="" name=""/>
</l3extRsNodeL3OutAtt>
<l3extLIfP descr="" name="portIf-spine1-3"
    ownerKey="" ownerTag="" tag="yellow-green">
    <ospfIfP authKeyId="1" authType="none" descr="" name="">
      <ospfRsIfPol tnOspfIfPolName="ospfIfPol"/>
    </ospfIfP>
    <l3extRsNdIfPol tnNdIfPolName=""/>
    <l3extRsIngressQosDppPol tnQosDppPolName=""/>
    <l3extRsEgressQosDppPol tnQosDppPolName=""/>
    <l3extRsPathL3OutAtt addr="7.2.1.1/24" descr=""
       encap="vlan-4"
       encapScope="local"
       ifInstT="sub-interface"
       llAddr="::" mac="00:22:BD:F8:19:FF"
       mode="regular"
       mtu="1500"
       tDn="topology/pod-1/paths-111/pathep-[eth1/12]"
       targetDscp="unspecified"/>
 </l3extLIfP>
 <l3extLIfP descr="" name="portIf-spine2-1"
     ownerKey=""
     ownerTag=""
     tag="yellow-green">
     <ospfIfP authKeyId="1"
          authType="none"
          descr=""
          name="">
          <ospfRsIfPol tnOspfIfPolName="ospfIfPol"/>
     </ospfIfP>
     <l3extRsNdIfPol tnNdIfPolName=""/>
     <l3extRsIngressQosDppPol tnQosDppPolName=""/>
     <l3extRsEgressQosDppPol tnQosDppPolName=""/>
     <l3extRsPathL3OutAtt addr="7.1.0.1/24" descr=""
          encap="vlan-4"
          encapScope="local"
          ifInstT="sub-interface"
          llAddr="::" mac="00:22:BD:F8:19:FF"
          mode="regular"
          mtu="9000"
          tDn="topology/pod-1/paths-112/pathep-[eth1/11]"
          targetDscp="unspecified"/>
  </l3extLIfP>
  <l3extLIfP descr="" name="portif-spine2-2"
     ownerKey=""
     ownerTag=""
     tag="yellow-green">
     <ospfIfP authKeyId="1"
          authType="none" descr=""
          name="">
          <ospfRsIfPol tnOspfIfPolName="ospfIfPol"/>
    </ospfIfP>
    <l3extRsNdIfPol tnNdIfPolName=""/>
    <l3extRsIngressQosDppPol tnQosDppPolName=""/>
    <l3extRsEgressQosDppPol tnQosDppPolName=""/>
    <l3extRsPathL3OutAtt addr="7.2.2.1/24" descr=""
          encap="vlan-4"
          encapScope="local"
          ifInstT="sub-interface"
                 llAddr="::" mac="00:22:BD:F8:19:FF"
                 mode="regular"
                 mtu="1500"
                 tDn="topology/pod-1/paths-112/pathep-[eth1/12]"
```

```
                targetDscp="unspecified"/>
        </l3extLIfP>
        <l3extLIfP descr="" name="portIf-spine1-2"
             ownerKey="" ownerTag="" tag="yellow-green">
             <ospfIfP authKeyId="1" authType="none" descr="" name="">
                  <ospfRsIfPol tnOspfIfPolName="ospfIfPol"/>
             </ospfIfP>
             <l3extRsNdIfPol tnNdIfPolName=""/>
             <l3extRsIngressQosDppPol tnQosDppPolName=""/>
             <l3extRsEgressQosDppPol tnQosDppPolName=""/>
             <l3extRsPathL3OutAtt addr="9.0.0.1/24" descr=""
              encap="vlan-4"
              encapScope="local"
              ifInstT="sub-interface"
                   llAddr="::" mac="00:22:BD:F8:19:FF"
                   mode="regular"
                   mtu="9000"
                   tDn="topology/pod-1/paths-111/pathep-[eth1/11]"
                   targetDscp="unspecified"/>
        </l3extLIfP>
        <l3extLIfP descr="" name="portIf-spine1-1"
              ownerKey="" ownerTag="" tag="yellow-green">
              <ospfIfP authKeyId="1" authType="none" descr="" name="">
                   <ospfRsIfPol tnOspfIfPolName="ospfIfPol"/>
              </ospfIfP>
              <l3extRsNdIfPol tnNdIfPolName=""/>
              <l3extRsIngressQosDppPol tnQosDppPolName=""/>
              <l3extRsEgressQosDppPol tnQosDppPolName=""/>
              <l3extRsPathL3OutAtt addr="7.0.0.1/24" descr=""
                encap="vlan-4"
                encapScope="local"
                ifInstT="sub-interface"
                llAddr="::" mac="00:22:BD:F8:19:FF"
                mode="regular"
                mtu="1500"
                tDn="topology/pod-1/paths-111/pathep-[eth1/10]"
                     targetDscp="unspecified"/>
        </l3extLIfP>
        <bgpInfraPeerP addr="10.10.3.2"
           allowedSelfAsCnt="3"
           ctrl="send-com,send-ext-com"
           descr="" name="" peerCtrl=""
           peerT="wan"
           privateASctrl="" ttl="2" weight="0">
           <bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
           <bgpAsP asn="150" descr="" name="aspn"/>
        </bgpInfraPeerP>
        <bgpInfraPeerP addr="10.10.4.1"
           allowedSelfAsCnt="3"
           ctrl="send-com,send-ext-com" descr="" name="" peerCtrl=""
           peerT="wan"
           privateASctrl="" ttl="1" weight="0">
           <bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
           <bgpAsP asn="100" descr="" name=""/>
         </bgpInfraPeerP>
         <bgpInfraPeerP addr="10.10.3.1"
           allowedSelfAsCnt="3"
           ctrl="send-com,send-ext-com" descr="" name="" peerCtrl=""
           peerT="wan"
           privateASctrl="" ttl="1" weight="0">
           <bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
           <bgpAsP asn="100" descr="" name=""/>
        </bgpInfraPeerP>
</l3extLNodeP>
```

```
        <bgpRtTargetInstrP descr="" name="" ownerKey="" ownerTag="" rtTargetT="explicit"/>
        <l3extRsL3DomAtt tDn="uni/l3dom-l3dom"/>
        <l3extInstP descr="" matchT="AtleastOne" name="golfInstP"
                  prio="unspecified"
                  targetDscp="unspecified">
                  <fvRsCustQosPol tnQosCustomPolName=""/>
         </l3extInstP>
         <bgpExtP descr=""/>
         <ospfExtP areaCost="1"
                areaCtrl="redistribute,summary"
                areaId="0.0.0.1"
                areaType="regular" descr=""/>
</l3extOut>
```

**Step 3** The XML below configures the tenant consumer of the infra part of the GOLF service. Include this XML structure in the body of the POST message.

**Example:**

```
<fvTenant descr="" dn="uni/tn-pep6" name="pep6" ownerKey="" ownerTag="">
     <vzBrCP descr="" name="webCtrct"
          ownerKey="" ownerTag="" prio="unspecified"
          scope="global" targetDscp="unspecified">
          <vzSubj consMatchT="AtleastOne" descr=""
               name="http" prio="unspecified" provMatchT="AtleastOne"
               revFltPorts="yes" targetDscp="unspecified">
               <vzRsSubjFiltAtt directives="" tnVzFilterName="default"/>
          </vzSubj>
      </vzBrCP>
      <vzBrCP descr="" name="webCtrct-pod2"
           ownerKey="" ownerTag="" prio="unspecified"
           scope="global" targetDscp="unspecified">
           <vzSubj consMatchT="AtleastOne" descr=""
                name="http" prio="unspecified"
                provMatchT="AtleastOne" revFltPorts="yes"
                targetDscp="unspecified">
                <vzRsSubjFiltAtt directives=""
                      tnVzFilterName="default"/>
           </vzSubj>
       </vzBrCP>
       <fvCtx descr="" knwMcastAct="permit"
            name="ctx6" ownerKey="" ownerTag=""
            pcEnfDir="ingress" pcEnfPref="enforced">
            <bgpRtTargetP af="ipv6-ucast"
                 descr="" name="" ownerKey="" ownerTag="">
                 <bgpRtTarget descr="" name="" ownerKey="" ownerTag=""
                 rt="route-target:as4-nn2:100:1256"
                 type="export"/>
                 <bgpRtTarget descr="" name="" ownerKey="" ownerTag=""
                      rt="route-target:as4-nn2:100:1256"
                      type="import"/>
            </bgpRtTargetP>
            <bgpRtTargetP af="ipv4-ucast"
                 descr="" name="" ownerKey="" ownerTag="">
                 <bgpRtTarget descr="" name="" ownerKey="" ownerTag=""
                      rt="route-target:as4-nn2:100:1256"
                      type="export"/>
                 <bgpRtTarget descr="" name="" ownerKey="" ownerTag=""
                      rt="route-target:as4-nn2:100:1256"
                      type="import"/>
            </bgpRtTargetP>
            <fvRsCtxToExtRouteTagPol tnL3extRouteTagPolName=""/>
            <fvRsBgpCtxPol tnBgpCtxPolName=""/>
            <vzAny descr="" matchT="AtleastOne" name=""/>
```

```
     <fvRsOspfCtxPol tnOspfCtxPolName=""/>
     <fvRsCtxToEpRet tnFvEpRetPolName=""/>
     <l3extGlobalCtxName descr="" name="dci-pep6"/>
</fvCtx>
<fvBD arpFlood="no" descr="" epMoveDetectMode=""
     ipLearning="yes"
     limitIpLearnToSubnets="no"
     llAddr="::" mac="00:22:BD:F8:19:FF"
     mcastAllow="no"
     multiDstPktAct="bd-flood"
     name="bd107" ownerKey="" ownerTag="" type="regular"
     unicastRoute="yes"
     unkMacUcastAct="proxy"
     unkMcastAct="flood"
     vmac="not-applicable">
     <fvRsBDToNdP tnNdIfPolName=""/>
     <fvRsBDToOut tnL3extOutName="routAccounting-pod2"/>
     <fvRsCtx tnFvCtxName="ctx6"/>
     <fvRsIgmpsn tnIgmpSnoopPolName=""/>
     <fvSubnet ctrl="" descr="" ip="27.6.1.1/24"
          name="" preferred="no"
          scope="public"
          virtual="no"/>
          <fvSubnet ctrl="nd" descr="" ip="2001:27:6:1::1/64"
               name="" preferred="no"
               scope="public"
               virtual="no">
               <fvRsNdPfxPol tnNdPfxPolName=""/>
          </fvSubnet>
          <fvRsBdToEpRet resolveAct="resolve" tnFvEpRetPolName=""/>
</fvBD>
<fvBD arpFlood="no" descr="" epMoveDetectMode=""
     ipLearning="yes"
     limitIpLearnToSubnets="no"
     llAddr="::" mac="00:22:BD:F8:19:FF"
     mcastAllow="no"
     multiDstPktAct="bd-flood"
     name="bd103" ownerKey="" ownerTag="" type="regular"
     unicastRoute="yes"
     unkMacUcastAct="proxy"
     unkMcastAct="flood"
     vmac="not-applicable">
     <fvRsBDToNdP tnNdIfPolName=""/>
     <fvRsBDToOut tnL3extOutName="routAccounting"/>
     <fvRsCtx tnFvCtxName="ctx6"/>
     <fvRsIgmpsn tnIgmpSnoopPolName=""/>
     <fvSubnet ctrl="" descr="" ip="23.6.1.1/24"
          name="" preferred="no"
          scope="public"
          virtual="no"/>
     <fvSubnet ctrl="nd" descr="" ip="2001:23:6:1::1/64"
          name="" preferred="no"
          scope="public" virtual="no">
          <fvRsNdPfxPol tnNdPfxPolName=""/>
     </fvSubnet>
     <fvRsBdToEpRet resolveAct="resolve" tnFvEpRetPolName=""/>
</fvBD>
<vnsSvcCont/>
<fvRsTenantMonPol tnMonEPGPolName=""/>
<fvAp descr="" name="AP1"
     ownerKey="" ownerTag="" prio="unspecified">
     <fvAEPg descr=""
          isAttrBasedEPg="no"
          matchT="AtleastOne"
```

```
                name="epg107"
                pcEnfPref="unenforced" prio="unspecified">
                <fvRsCons prio="unspecified"
                     tnVzBrCPName="webCtrct-pod2"/>
                <fvRsPathAtt descr=""
                     encap="vlan-1256"
                     instrImedcy="immediate"
                     mode="regular" primaryEncap="unknown"
                     tDn="topology/pod-2/paths-107/pathep-[eth1/48]"/>
                <fvRsDomAtt classPref="encap" delimiter=""
                     encap="unknown"
                     instrImedcy="immediate"
                     primaryEncap="unknown"
                     resImedcy="lazy" tDn="uni/phys-phys"/>
                <fvRsCustQosPol tnQosCustomPolName=""/>
                <fvRsBd tnFvBDName="bd107"/>
                <fvRsProv matchT="AtleastOne"
                     prio="unspecified"
                     tnVzBrCPName="default"/>
           </fvAEPg>
           <fvAEPg descr=""
                isAttrBasedEPg="no"
                matchT="AtleastOne"
                name="epg103"
                pcEnfPref="unenforced" prio="unspecified">
                <fvRsCons prio="unspecified" tnVzBrCPName="default"/>
                <fvRsCons prio="unspecified" tnVzBrCPName="webCtrct"/>
                <fvRsPathAtt descr="" encap="vlan-1256"
                     instrImedcy="immediate"
                     mode="regular" primaryEncap="unknown"
                     tDn="topology/pod-1/paths-103/pathep-[eth1/48]"/>
                     <fvRsDomAtt classPref="encap" delimiter=""
                          encap="unknown"
                          instrImedcy="immediate"
                          primaryEncap="unknown"
                          resImedcy="lazy" tDn="uni/phys-phys"/>
                     <fvRsCustQosPol tnQosCustomPolName=""/>
                     <fvRsBd tnFvBDName="bd103"/>
           </fvAEPg>
      </fvAp>
      <l3extOut descr=""
           enforceRtctrl="export"
           name="routAccounting-pod2"
           ownerKey="" ownerTag="" targetDscp="unspecified">
           <l3extRsEctx tnFvCtxName="ctx6"/>
           <l3extInstP descr=""
                matchT="AtleastOne"
                name="accountingInst-pod2"
                prio="unspecified" targetDscp="unspecified">
           <l3extSubnet aggregate="export-rtctrl,import-rtctrl"
                descr="" ip="::/0" name=""
                scope="export-rtctrl,import-rtctrl,import-security"/>
           <l3extSubnet aggregate="export-rtctrl,import-rtctrl"
                descr=""
                ip="0.0.0.0/0" name=""
                scope="export-rtctrl,import-rtctrl,import-security"/>
           <fvRsCustQosPol tnQosCustomPolName=""/>
           <fvRsProv matchT="AtleastOne"
                prio="unspecified" tnVzBrCPName="webCtrct-pod2"/>
           </l3extInstP>
           <l3extConsLbl descr=""
                name="golf2"
                owner="infra"
                ownerKey="" ownerTag="" tag="yellow-green"/>
```

```
        </l3extOut>
        <l3extOut descr=""
             enforceRtctrl="export"
             name="routAccounting"
             ownerKey="" ownerTag="" targetDscp="unspecified">
             <l3extRsEctx tnFvCtxName="ctx6"/>
             <l3extInstP descr=""
                  matchT="AtleastOne"
                  name="accountingInst"
                  prio="unspecified" targetDscp="unspecified">
             <l3extSubnet aggregate="export-rtctrl,import-rtctrl" descr=""
                  ip="0.0.0.0/0" name=""
                  scope="export-rtctrl,import-rtctrl,import-security"/>
             <fvRsCustQosPol tnQosCustomPolName=""/>
             <fvRsProv matchT="AtleastOne" prio="unspecified" tnVzBrCPName="webCtrct"/>
             </l3extInstP>
             <l3extConsLbl descr=""
                  name="golf"
                  owner="infra"
                  ownerKey="" ownerTag="" tag="yellow-green"/>
        </l3extOut>
</fvTenant>
```

# Distributing BGP EVPN Type-2 Host Routes to a DCIG

## Distributing BGP EVPN Type-2 Host Routes to a DCIG

In APIC up to release 2.0(1f), the fabric control plane did not send EVPN host routes directly, but advertised public bridge domain (BD) subnets in the form of BGP EVPN type-5 (IP Prefix) routes to a Data Center Interconnect Gateway (DCIG). This could result in suboptimal traffic forwarding. To improve forwarding, in APIC release 2.1x, you can enable fabric spines to also advertise host routes using EVPN type-2 (MAC-IP) host routes to the DCIG along with the public BD subnets.

To do so, you must perform the following steps:

1. When you configure the BGP Address Family Context Policy, enable Host Route Leak.
2. When you leak the host route to BGP EVPN in a GOLF setup:
   a. To enable host routes when GOLF is enabled, the BGP Address Family Context Policy must be configured under the application tenant (the application tenant is the consumer tenant that leaks the endpoint to BGP EVPN) rather than under the infrastructure tenant.
   b. For a single-pod fabric, the host route feature is not required. The host route feature is required to avoid sub-optimal forwarding in a multi-pod fabric setup. However, if a single-pod fabric is setup, then in order to leak the endpoint to BGP EVPN, a Fabric External Connection Policy must be configured to provide the ETEP IP address. Otherwise, the host route will not leak to BGP EVPN.
3. When you configure VRF properties:
   a. Add the BGP Address Family Context Policy to the BGP Context Per Address Families for IPv4 and IPv6.

**b.** Configure BGP Route Target Profiles that identify routes that can be imported or exported from the VRF.

# Distributing BGP EVPN Type-2 Host Routes to a DCIG Using the GUI

Enable distributing BGP EVPN type-2 host routes with the following steps:

### Before you begin

You must have already configured ACI WAN Interconnect services in the infra tenant, and configured the tenant that will consume the services.

### Procedure

**Step 1** On the menu bar, click **Tenants** > **infra**,

**Step 2** In the Navigation pane, expand the **External Routed Networks**, then expand **Protocol Policies** and **BGP**.

**Step 3** Right-click **BGP Address Family Context**, select **Create BGP Address Family Context Policy** and perform the following steps:

a) Type a name for the policy and optionally add a description.
b) Click the **Enable Host Route Leak** check box.
c) Click **Submit**.

**Step 4** Click **Tenants** > **tenant-name** (for a tenant that will consume the BGP Address Family Context Policy) and expand **Networking**.

**Step 5** Expand **VRFs** and click the VRF that will include the host routes you want to distribute.

**Step 6** When you configure the VRF properties, add the **BGP Address Family Context Policy** to the **BGP Context Per Address Families** for IPv4 and IPv6.

**Step 7** Click **Submit**.

# Enabling Distributing BGP EVPN Type-2 Host Routes to a DCIG Using the NX-OS Style CLI

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Configure distributing EVPN type-2 host routes to a DCIG with the following commands in the BGP address family configuration mode.<br>**Example:**<br>`apic1(config)# leaf 101`<br>`apic1(config-leaf)# template bgp address-family bgpAf1 tenant` | This template will be available on all nodes where tenant bgp_t1 has a VRF deployment. To disable distributing EVPN type-2 host routes, enter the **no host-rt-enable** command. |

| | Command or Action | Purpose |
|---|---|---|
| | **bgp_t1**<br>apic1(config-bgp-af)# **distance 250 240 230**<br>apic1(config-bgp-af)# **host-rt-enable**<br><br>apic1(config-bgp-af)# **exit** | |

# Enabling Distributing BGP EVPN Type-2 Host Routes to a DCIG Using the REST API

Enable distributing BGP EVPN type-2 host routes using the REST API, as follows:

### Before you begin

EVPN services must be configured.

### Procedure

**Step 1** Configure the Host Route Leak policy, with a POST containing XML such as in the following example:

**Example:**

```
<bgpCtxAfPol descr="" ctrl="host-rt-leak" name="bgpCtxPol_0 status=""/>
```

**Step 2** Apply the policy to the VRF BGP Address Family Context Policy for one or both of the address families using a POST containing XML such as in the following example:

**Example:**

```
<fvCtx name="vni-10001">
<fvRsCtxToBgpCtxAfPol af="ipv4-ucast" tnBgpCtxAfPolName="bgpCtxPol_0"/>
<fvRsCtxToBgpCtxAfPol af="ipv6-ucast" tnBgpCtxAfPolName="bgpCtxPol_0"/>
</fvCtx>
```