



REST API を使用してタスクを実行する

- [Part I : レイヤ 3 の設定 \(1 ページ\)](#)
- [パートII : 外部ルーティング \(L3Out\) の設定 \(27 ページ\)](#)

Part I : レイヤ 3 の設定

REST API を使用した共通パーベシブ ゲートウェイの設定

REST API を使用した共通パーベシブ ゲートウェイの設定

始める前に

- テナント、VRF、およびブリッジ ドメインが作成されていること。

手順

共通パーベシブ ゲートウェイを設定します。

次の REST API XML の例では、太字のテキストは一般的なパーベシブ ゲートウェイの設定に関連しています。

例 :

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="test">
    <fvCtx name="test"/>

    <fvBD name="test" >vmac="12:34:56:78:9a:bc">
      <fvRsCtx tnFvCtxName="test"/>
      <!-- Primary address -->
      <fvSubnet ip="192.168.15.254/24" preferred="yes"/>
      <!-- Virtual address -->
    </fvBD>
  </fvTenant>
</polUni>
```

```

    <fvSubnet ip="192.168.15.1/24" virtual="yes"/>
  </fvBD>

  <fvAp name="test">
    <fvAEPg name="web">
      <fvRsBd tnFvBDName="test"/>
      <fvRsPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/3]" encap="vlan-1002"/>
    </fvAEPg>
  </fvAp>
</fvTenant>
</polUni>

```

REST API を使用した IP エージングの設定

REST API を使用した IP エージングの設定

このセクションでは、REST API を使用した IP エージング ポリシーを有効および無効にする方法を説明します。

手順

ステップ 1 IP エージング ポリシーを有効にするには：

例：

```

<epIpAgingP adminSt="enabled" descr="" dn="uni/infra/ipAgingP-default" name="default" ownerKey=""
ownerTag=""/>

```

ステップ 2 IP エージング ポリシーを無効にするには：

例：

```

<epIpAgingP adminSt="disabled" descr="" dn="uni/infra/ipAgingP-default" name="default" ownerKey=""
ownerTag=""/>

```

次のタスク

エンドポイントの IP アドレスをトラッキングするために使用される間隔を指定するには、次の例のように XML で post を送信することによって、エンドポイント保持ポリシーを作成します。

```

<fvEpRetPol bounceAgeIntvl="630" bounceTrig="protocol" holdIntvl="350" lcOwn="local"
localEpAgeIntvl="900" moveFreq="256" name="EndpointPol1" remoteEpAgeIntvl="350"/>

```

REST API を使用したブリッジドメインのスタティック ルートの設定

REST API を使用してブリッジドメインでのスタティック ルートの設定

- スタティック ルートのサブネットを作成するには、epg (fvAEPg で fvSubnet オブジェクト)、普及 BD (fvBD) 自体 BD しないに関連付けられているように構成されます。
- サブネットマスクが/32 にする必要があります (128/for IPv6) 1 つの IP アドレスまたは 1 つのエンドポイントをポイントします。これは、パーベイスブ BD に関連付けられている EPG に含まれます。

始める前に

テナント、VRF、BD、および EPG が作成されています。

手順

普及ゲートウェイで使用される BD のスタティック ルートを設定するには、次の例など post を入力します。

例 :

```
<fvAEPg name="ep1">
  <fvRsBd tnFvBDName="bd1"/>
  <fvSubnet ip="2002:0db8:85a3:0000:0000:8a2e:0370:7344/128" ctrl="no-default-gateway"
>
  <fvEpReachability>
    <ipNextHopEpP nhAddr="2001:0db8:85a3:0000:0000:8a2e:0370:7343/128" />
  </fvEpReachability>
</fvSubnet>
</fvAEPg>
```

REST API を使用した IPv6 ネイバー 探索の設定

REST API を使用したブリッジドメインの IPv6 ネイバー探索対応のテナント、VRF、およびブリッジドメインの作成

手順

ネイバー探索インターフェイス ポリシーとネイバー探索プレフィックス ポリシーが適用された、テナント、VRF、ブリッジドメインを作成します。

例 :

```
<fvTenant descr="" dn="uni/tn-ExampleCorp" name="ExampleCorp" ownerKey="" ownerTag="">
  <ndIfPol name="NDPol001" ctrl="managed-cfg" descr="" hopLimit="64" mtu="1500" nsIntvl="1000"
nsRetries="3" ownerKey="" ownerTag="" raIntvl="600" raLifetime="1800" reachableTime="0"
retransTimer="0"/>
  <fvCtx descr="" knwMcastAct="permit" name="pvn1" ownerKey="" ownerTag="" pcEnfPref="enforced">
  </fvCtx>
  <fvBD arpFlood="no" descr="" mac="00:22:BD:F8:19:FF" multiDstPktAct="bd-flood" name="bd1"
ownerKey="" ownerTag="" unicastRoute="yes" unkMacUcastAct="proxy" unkMcastAct="flood">
    <fvRsBDToNDP tnNdIfPolName="NDPol001"/>
    <fvRsCtx tnFvCtxName="pvn1"/>
    <fvSubnet ctrl="nd" descr="" ip="34::1/64" name="" preferred="no" scope="private">
      <fvRsNDPfxPol tnNDPfxPolName="NDPfxPol001"/>
    </fvSubnet>
    <fvSubnet ctrl="nd" descr="" ip="33::1/64" name="" preferred="no" scope="private">
      <fvRsNDPfxPol tnNDPfxPolName="NDPfxPol002"/>
    </fvSubnet>
  </fvBD>
  <ndPfxPol ctrl="auto-cfg,on-link" descr="" lifetime="1000" name="NDPfxPol001" ownerKey=""
ownerTag="" prefLifetime="1000"/>
  <ndPfxPol ctrl="auto-cfg,on-link" descr="" lifetime="4294967295" name="NDPfxPol002" ownerKey=""
ownerTag="" prefLifetime="4294967295"/>
</fvTenant>
```

(注)

外部ルーテッドを設定するときにパブリックサブネットがある場合は、ブリッジドメインを外部設定と関連付ける必要があります。

REST API を使用したレイヤ3インターフェイス上のRAによるIPv6ネイバー探索インターフェイスポリシーの設定

手順

IPv6 ネイバー検索インターフェイス ポリシーを設定し、レイヤ3 インターフェイスに関連付けます。

次の例では、非 VPC セットアップの設定が表示されます。

例：

```
<fvTenant dn="uni/tn-ExampleCorp" name="ExampleCorp">
  <ndIfPol name="NDPol001" ctrl="managed-cfg" hopLimit="64" mtu="1500" nsIntvl="1000" nsRetries="3"
raIntvl="600" raLifetime="1800" reachableTime="0" retransTimer="0"/>
  <fvCtx name="pvn1" pcEnfPref="enforced">
  </fvCtx>
  <l3extOut enforceRtctrl="export" name="l3extOut001">
    <l3extRsEctx tnFvCtxName="pvn1"/>
    <l3extLNodeP name="lnodeP001">
      <l3extRsNodeL3OutAtt rtrId="11.11.205.1" rtrIdLoopBack="yes" tDn="topology/pod-2/node-2011"/>

      <l3extLIIfP name="lifP001">
        <l3extRsPathL3OutAtt addr="2001:20:21:22::2/64" ifInstT="l3-port" llAddr="::"
mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit" tDn="topology/pod-2/paths-2011/pathep-[eth1/1]">

      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>
```

```

        <ndRsPfxPToNdPfxPol tnNdPfxPolName="NDPfxPol001"/>
    </ndPfxP>
</l3extRsPathL3OutAtt>
<l3extRsNdIfPol tnNdIfPolName="NDPol001"/>
</l3extLIfP>
</l3extLNodeP>
<l3extInstP name="instp"/>
</l3extOut>
<ndPfxPol ctrl="auto-cfg,on-link" descr="" lifetime="1000" name="NDPfxPol001" ownerKey="" ownerTag=""
prefLifetime="1000"/>
</fvTenant>

```

(注)

VPC ポートについては、ndPfxP が l3extRsNodeL3OutAtt ではなく l3extMember の子である必要があります。次のコード スニペットは、VPC のセットアップでの設定を示します。

```

<l3extLNodeP name="lnodeP001">
<l3extRsNodeL3OutAtt rtrId="11.11.205.1" rtrIdLoopBack="yes" tDn="topology/pod-2/node-2011"/>
<l3extRsNodeL3OutAtt rtrId="12.12.205.1" rtrIdLoopBack="yes" tDn="topology/pod-2/node-2012"/>
    <l3extLIfP name="lifP002">
        <l3extRsPathL3OutAtt addr="0.0.0.0" encap="vlan-205" ifInstT="ext-svi" llAddr="::"
mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-2/protpaths-2011-2012/pathep-[vpc7]" >
            <l3extMember addr="2001:20:25:1::1/64" descr="" llAddr="::" name="" nameAlias="" side="A">
                <ndPfxP >
                    <ndRsPfxPToNdPfxPol tnNdPfxPolName="NDPfxPol001"/>
                </ndPfxP>
            </l3extMember>
            <l3extMember addr="2001:20:25:1::2/64" descr="" llAddr="::" name="" nameAlias="" side="B">
                <ndPfxP >
                    <ndRsPfxPToNdPfxPol tnNdPfxPolName="NDPfxPol001"/>
                </ndPfxP>
            </l3extMember>
        </l3extRsPathL3OutAtt>
    </l3extLIfP>
</l3extLNodeP>

```

REST API を使用したネイバー探索重複アドレス検出の構成

手順

ステップ 1 サブネットの ipv6Dad エントリの値を **無効**に変更することによって、サブネットのネイバー探索重複アドレス検出プロセスを無効にします。

次の例は、2001:DB8:A::11/64 サブネットのネイバー探索重複アドレス検出エントリを **無効 disabled** に設定する方法を示しています：

(注)

次の REST API 例では、長い 1 行のテキストは \ で分けて読みやすくします。

例 :

```
<l3extRsPathL3OutAtt addr="2001:DB8:A::2/64" autostate="enabled" \
  childAction="" descr="" encap="vlan-1035" encapScope="local" \
ifInstT="ext-svi" ipv6Dad="enabled" llAddr=": : " \
  mtu="inherit" \
  rn="rspathL3OutAtt-[topology/pod-1/paths-105/pathep-[eth1/1]]"
  \
  status="" tDn="topology/pod-1/paths-105/pathep-[eth1/1]" >
<l3extIp >addr="2001:DB8:A::11/64" childAction="" descr="" \
  ipv6Dad="disabled" name="" nameAlias="" \
  rn="addr-[2001:DB8:A::11/64]" status=""/>

</l3extRsPathL3OutAtt>
</l3extLIIfP>
</l3extLNodeP>
```

ステップ2 リーフ スイッチで **show ipv6 int** コマンドを入力して、構成がリーフ スイッチに正しくプッシュされたか確認してください。例 :

```
swtb23-leaf5# show ipv6 int vrf icmpv6:v1
IPv6 Interface Status for VRF "icmpv6:v1"(9)

vlan2, Interface status: protocol-up/link-up/admin-up, iod: 73
if_mode: ext
IPv6 address:
  2001:DB8:A::2/64 [VALID] [PREFERRED]
  2001:DB8:A::11/64 [VALID] [dad-disabled]
IPv6 subnet: 2001:DB8:A::/64
IPv6 link-local address: fe80::863d:c6ff:fe9f:eb8b/10 (Default) [VALID]
```

REST API を使用した Microsoft NLB の設定

REST API を使用したユニキャスト モードでの Microsoft NLB の設定

手順

Microsoft NLB をユニキャスト モードで設定するには、次の例のように XML で POST を送信します。

例 :

```
https://apic-ip-address/api/node/mo/uni/.xml
<polUni>
  <fvTenant name="tn2" >
    <fvCtx name="ctx1"/>
    <fvBD name="bd2">
      <fvRsCtx tnFvCtxName="ctx1" />
    </fvBD>
    <fvAp name = "ap1">
      <fvAEPg name = "ep1">
        <fvRsBd tnFvBDName = "bd2"/>
        <fvSubnet ip="10.0.1.1/32" scope="public" ctrl="no-default-gateway">
```

```

        <fvEpNlb mac="12:21:21:35" mode="mode-uc"/>
      </fvSubnet>
    </fvAEPg>
  </fvAp>
</fvTenant>
</polUni>

```

REST API を使用したマルチキャスト モードでの Microsoft NLB の設定

手順

Microsoft NLB をマルチキャスト モードで設定するには、次の例のように XML で POST を送信します。

例：

```

https://apic-ip-address/api/node/mo/uni/.xml
<polUni>
  <fvTenant name="tn2" >
    <fvCtx name="ctx1"/>
    <fvBD name="bd2">
      <fvRsCtx tnFvCtxName="ctx1" />
    </fvBD>
    <fvAp name = "ap1">
      <fvAEPg name = "ep1">
        <fvRsBd tnFvBDName = "bd2"/>
        <fvSubnet ip="2001:0db8:85a3:0000:0000:8a2e:0370:7344/128" scope="public"
ctrl="no-default-gateway">
          <fvEpNlb mac="03:21:21:35" mode="mode-mcast--static"/>
        </fvSubnet>
        <fvRsPathAtt tDn="topology/pod-1/paths-101/patchep-[eth1/6]" encap="vlan-911" >
          <fvNlbStaticGroup mac = "03:21:21:35" />
        </fvRsPathAtt>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>

```

REST API を使用した IGMP モードでの Microsoft NLB の設定

手順

Microsoft NLB を IGMP モードで設定するには、次の例のように XML で POST を送信します。

例：

```

https://apic-ip-address/api/node/mo/uni/.xml
<polUni>
  <fvTenant name="tn2" >
    <fvCtx name="ctx1"/>

```

```

<fvBD name="bd2">
  <fvRsCtx tnFvCtxName="ctx1" />
</fvBD>
<fvAp name = "ap1">
  <fvAEPg name = "ep1">
    <fvRsBd tnFvBDName = "bd2"/>
    <fvSubnet ip="10.0.1.3/32" scope="public" ctrl="no-default-gateway">
      <fvEpNlb group = "224.132.18.17" mode="mode-mcast-igmp" />
    </fvSubnet>
  </fvAEPg>
</fvAp>
</fvTenant>
</polUni>

```

REST API を使用した IGMP スヌーピングの設定

REST API を使用したブリッジ ドメインへの IGMP スヌーピング ポリシーの設定と割り当て

手順の概要

1. IGMP スヌーピング ポリシーを設定してブリッジ ドメインに割り当てるには、次の例のように XML で POST を送信します。

手順の詳細

手順

IGMP スヌーピング ポリシーを設定してブリッジ ドメインに割り当てるには、次の例のように XML で POST を送信します。

例 :

```

https://apic-ip-address/api/node/mo/uni/.xml
<fvTenant name="mcast_tenant1">
  <!-- Create an IGMP snooping template, and provide the options -->
  <igmpSnoopPol name="igmp_snp_bd_21" ver="v2" adminSt="enabled" lastMbrIntvl="1" queryIntvl="125"
    rspIntvl="10" startQueryCnt="2" startQueryIntvl="31" />
  <fvCtx name="ip_video"/>
  <fvBD name="bd_21">
    <fvRsCtx tnFvCtxName="ip_video"/>
    <!-- Bind IGMP snooping to a BD -->
    <fvRsIgmpsn tnIgmpSnoopPolName="igmp_snp_bd_21"/>
  </fvBD></fvTenant>

```

この例では、次のプロパティで IGMP スヌーピング ポリシー、igmp_snp_bd_12 を作成および設定し、IGMP ポリシー、igmp_snp_bd_12 をブリッジ ドメイン bd_21 にバインドします。

- 管理状態が有効です。

- 最後のメンバクエリ間隔は、デフォルトでは、1 秒です。
- クエリ間隔は、デフォルトでは 125 です。
- クエリの応答間隔はデフォルトでは 10 秒です。
- クエリの開始カウントは、デフォルトでは 2 メッセージです。
- クエリの開始間隔は 31 秒です。
- クエリア バージョンを v2 に設定する。

REST API を使用した静的ポートでの IGMP スヌーピングとマルチキャストの有効化

EPG に静的に割り当てられているポートで、IGMP スヌーピングおよびマルチキャスト処理を有効にできます。それらのポートで有効な IGMP スヌープおよびマルチキャストトラフィックへのアクセスを許可または拒否するアクセスユーザーのグループを作成および割り当てるができます。

手順の概要

1. スタティック ポートでアプリケーション EPG を構成するには、それらのポートを IGMP スヌーピングおよびマルチキャストトラフィックを受信し処理するように有効にして、グループをアクセスに割り当てるかトラフィックへのアクセスを拒否するように割り当て、次の例のように XML で POST を送信します。

手順の詳細

手順

スタティック ポートでアプリケーション EPG を構成するには、それらのポートを IGMP スヌーピングおよびマルチキャストトラフィックを受信し処理するように有効にして、グループをアクセスに割り当てるかトラフィックへのアクセスを拒否するように割り当て、次の例のように XML で POST を送信します。

次の例では、IGMP スヌーピングが有効になっています。leaf_102 インターフェイス 1/10。これは、VLAN 202にあります。マルチキャスト IP アドレス 224.1.1.1 および 225.1.1.1 がこのポートに関連付けられています。

例：

```
https://apic-ip-address/api/node/mo/uni/.xml
<fvTenant name="tenant_A">
  <fvAp name="application">
    <fvAEPg name="epg_A">
      <fvRsPathAtt encap="vlan-202" instrImedcy="immediate" mode="regular"
tDn="topology/pod-1/paths-102/pathep-[eth1/10]">
        <!-- IGMP snooping static group case -->
        <igmpSnoopStaticGroup group="224.1.1.1" source="0.0.0.0"/>
        <igmpSnoopStaticGroup group="225.1.1.1" source="2.2.2.2"/>
      </fvRsPathAtt>
    </fvAEPg>
  </fvAp>
</fvTenant>
```

```

        </fvRsPathAtt>
    </fvAEPg>
</fvAp>
</fvTenant>

```

IGMP スヌーピングを REST API を使用するマルチキャストグループのアクセスを有効化

IGMP を有効にした後にスヌーピングおよび、EPG に静的に割り当てられているポートでマルチキャストすることができますし、作成を許可または IGMP スヌーピングへのアクセスを拒否するユーザのアクセスのグループを割り当てるおよびマルチキャストトラフィックは、これらのポートで有効になっています。

手順

アクセスグループを定義するには、F23ブローカー、次の例のような XML を POST 送信します。

例は、構成アクセスグループ F23ブローカー-tenant_A、Rmap_A、application_A、リーフ 102、1/10、インターフェイス VLAN 202 で、epg_A に関連付けられている。Rmap_A、アクセスグループとの関連付けによって F23broker マルチキャストアドレス 226.1.1.1/24 で受信したマルチキャストトラフィックへのアクセスがあり、マルチキャストアドレス 227.1.1.1/24 で受信したトラフィックへのアクセスは拒否されます。

例：

```

<!-- api/node/mo/uni/.xml --> <fvTenant name="tenant_A"> <pimRouteMapPol name="Rmap_A"> <pimRouteMapEntry
action="permit" grp="226.1.1.1/24" order="10"/> <pimRouteMapEntry action="deny" grp="227.1.1.1/24" order="20"/>
</pimRouteMapPol> <fvAp name="application_A"> <fvAEPg name="epg_A"> <fvRsPathAtt encap="vlan-202"
instrImedcy="immediate" mode="regular" tDn="topology/pod-1/paths-102/pathep-[eth1/10]"> <!-- IGMP snooping
access group case --> <igmpSnoopAccessGroup name="F23broker"> <igmpRsSnoopAccessGroupFilterRMap
tnPimRouteMapPolName="Rmap_A"/> </igmpSnoopAccessGroup> </fvRsPathAtt> </fvAEPg> </fvAp> </fvTenant>

```

REST API を使用した MLD スヌーピングの設定

REST API を使用した MLD スヌーピング ポリシーの構成とブリッジ ドメインへの割り当て

手順

MLD スヌーピング ポリシーを構成してブリッジ ドメインに割り当てるには、次の例のように XML で POST を送信します：

例：

```
https://apic-ip-address/api/node/mo/uni/.xml
<fvTenant name="mldsn">
  <mldSnoopPol adminSt="enabled" ctrl="fast-leave,querier" name="mldsn-it-fabric-querier-policy"
    queryIntvl="125"      rspIntvl="10" startQueryCnt="2" startQueryIntvl="31" status=""/>
  <fvBD name="mldsn-bd3">
    <fvRsMldsn status="" tnMldSnoopPolName="mldsn-it-policy"/>
  </fvBD>
</fvTenant>
```

この例では、MLD スヌーピング ポリシーを作成して構成します。mldsn を次のプロパティと共に構成し、MLD ポリシー mldsn-it-fabric-querier-policy をブリッジドメインへをバインドします。mldsn-bd3 :

- 高速脱退処理が有効になっています
- クエリア処理が有効になっています
- クエリ間隔は 125 に設定されています
- 最大クエリ レスポンス タイムは 10 に設定されています
- 送信する初期クエリの数は 2 に設定されます
- 初期クエリの送信時間は 31 に設定されます

REST API を使用した IP マルチキャストの設定

REST API を使用したレイヤ 3 マルチキャストの設定

手順

ステップ 1 テナントと VRF を設定し、VRF のマルチキャストを有効にします。

例 :

```
<fvTenant dn="uni/tn-PIM_Tenant" name="PIM_Tenant">
  <fvCtx knwMcastAct="permit" name="ctx1">
    <pimCtxP mtu="1500">
    </pimCtxP>
  </fvCtx>
</fvTenant>
```

ステップ 2 L3 アウトを設定し、L3 アウト上のマルチキャスト (PIM、IGMP) を有効にします。

例 :

```
<l3extOut enforceRtctrl="export" name="l3out-pim_l3out1">
  <l3extRsEctx tnFvCtxName="ctx1"/>
  <l3extLNodeP configIssues="" name="bLeaf-CTX1-101">
    <l3extRsNodeL3OutAtt rtrId="200.0.0.1" rtrIdLoopBack="yes" tDn="topology/pod-1/node-101"/>

    <l3extLIfP name="if-PIM_Tenant-CTX1" tag="yellow-green">
      <igmpIfP/>
```

```

    <pimIfP>
      <pimRsIfPol tDn="uni/tn-PIM_Tenant/pimifpol-pim_pol1"/>
    </pimIfP>
    <l3extRsPathL3OutAtt addr="131.1.1.1/24" ifInstT="l3-port" mode="regular" mtu="1500"
tDn="topology/pod-1/paths-101/pathep-[eth1/46]"/>
  </l3extLIfP>
  </l3extLNodeP>
  <l3extRsL3DomAtt tDn="uni/l3dom-l3outDom"/>
  <l3extInstP name="l3out-PIM_Tenant-CTX1-ltopo" >
  </l3extInstP>
  <pimExtP enabledAf="ipv4-mcast" name="pim"/>
</l3extOut>

```

ステップ 3 テナントで BD を設定して、BD のマルチキャストおよび IGMP を有効にします。

例 :

```

<fvTenant dn="uni/tn-PIM_Tenant" name="PIM_Tenant">
  <fvBD arpFlood="yes" mcastAllow="yes" multiDstPktAct="bd-flood" name="bd2" type="regular"
unicastRoute="yes" unkMacUcastAct="flood" unkMcastAct="flood">
    <igmpIfP>
      <fvRsBDToOut tnL3extOutName="l3out-pim_l3out1"/>
      <fvRsCtx tnFvCtxName="ctx1"/>
      <fvRsIgmpsn/>
      <fvSubnet ctrl="" ip="41.1.1.254/24" preferred="no" scope="private" virtual="no"/>
    </fvBD>
  </fvTenant>

```

ステップ 4 IGMP ポリシーを設定し、それを BD に割り当てます。

例 :

```

<fvTenant dn="uni/tn-PIM_Tenant" name="PIM_Tenant">
  <igmpIfPol grpTimeout="260" lastMbrCnt="2" lastMbrRespTime="1" name="igmp_pol"
querierTimeout="255" queryIntvl="125" robustFac="2" rspIntvl="10" startQueryCnt="2"
startQueryIntvl="125" ver="v2">
  </igmpIfPol>
  <fvBD arpFlood="yes" mcastAllow="yes" name="bd2">
    <igmpIfP>
      <igmpRsIfPol tDn="uni/tn-PIM_Tenant/igmpIfPol-igmp_pol"/>
    </igmpIfP>
  </fvBD>
</fvTenant>

```

ステップ 5 VRF のルート マップ、PIM、および RP ポリシーを設定します。

(注)

REST API を使用してファブリック RP を設定する場合、最初にスタティック RP を設定します。

例 :

スタティック RP を設定しています :

```

<fvTenant dn="uni/tn-PIM_Tenant" name="PIM_Tenant">
  <pimRouteMapPol name="rootMap">
    <pimRouteMapEntry action="permit" grp="224.0.0.0/4" order="10" rp="0.0.0.0" src="0.0.0.0/0"/>
  </pimRouteMapPol>
  <fvCtx knwMcastAct="permit" name="ctx1">
    <pimCtxP ctrl="" mtu="1500">
      <pimStaticRPPol>
        <pimStaticRPEntryPol rpIp="131.1.1.2">
          <pimRPGrpRangePol>
            <rtmcRsFilterToRtMapPol tDn="uni/tn-PIM_Tenant/rtmap-rootMap"/>
          </pimRPGrpRangePol>
        </pimStaticRPEntryPol>
      </pimStaticRPPol>
    </pimCtxP>
  </fvCtx>

```

```

        </pimRPGrpRangePol>
      </pimStaticRPEntryPol>
    </pimStaticRPPol>
  </pimCtxP>
</fvCtx>
</fvTenant>

```

ファブリック RP を設定しています：

```

<fvTenant name="t0">
  <pimRouteMapPol name="fabricrp-rtmap">
    <pimRouteMapEntry grp="226.20.0.0/24" order="1" />
  </pimRouteMapPol>
  <fvCtx name="ctx1">
    <pimCtxP ctrl="">
      <pimFabricRPPol status="">
        <pimStaticRPEntryPol rpIp="6.6.6.6">
          <pimRPGrpRangePol>
            <rtmcRsFilterToRtMapPol tDn="uni/tn-t0/rtmap-fabricrp-rtmap" />
          </pimRPGrpRangePol>
        </pimStaticRPEntryPol>
      </pimFabricRPPol>
    </pimCtxP>
  </fvCtx>
</fvTenant>

```

ステップ 6 PIM インターフェイス ポリシーを設定し、それを L3 アウトに適用します。

例：

```

<fvTenant dn="uni/tn-PIM_Tenant" name="PIM_Tenant">
  <pimIfPol authKey="" authT="none" ctrl="" drDelay="60" drPrio="1" helloItvl="30000" itvl="60"
name="pim_pol1"/>
  <l3extOut enforceRtctrl="export" name="l3out-pim_l3out1" targetDscp="unspecified">
    <l3extRsEctx tnFvCtxName="ctx1"/>
    <l3extLNodeP name="bLeaf-CTX1-101">
      <l3extRsNodeL3OutAtt rtrId="200.0.0.1" rtrIdLoopBack="yes" tDn="topology/pod-1/node-101"/>

      <l3extLIIfP name="if-SIRI_VPC_src_rcv-CTX1" tag="yellow-green">
        <pimIfP>
          <pimRsIfPol tDn="uni/tn-tn-PIM_Tenant/pimifpol-pim_pol1"/>
        </pimIfP>
      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>

```

ステップ 7 Inter-VRF マルチキャストを設定します。

例：

```

<fvTenant name="t0">
  <pimRouteMapPol name="intervrf" status="">
    <pimRouteMapEntry grp="225.0.0.0/24" order="1" status="" />
    <pimRouteMapEntry grp="226.0.0.0/24" order="2" status="" />
    <pimRouteMapEntry grp="228.0.0.0/24" order="3" status="deleted" />
  </pimRouteMapPol>
  <fvCtx name="ctx1">
    <pimCtxP ctrl="">
      <pimInterVRFPol status="">
        <pimInterVRFEntryPol srcVrfDn="uni/tn-t0/ctx-stig_r_ctx" >
          <rtmcRsFilterToRtMapPol tDn="uni/tn-t0/rtmap-intervrf" />
        </pimInterVRFEntryPol>
      </pimInterVRFPol>
    </pimCtxP>
  </fvCtx>
</fvTenant>

```

```

        </pimCtxP>
    </fvCtx>
</fvTenant>

```

REST API を使用したレイヤ 3 IPv6 マルチキャストの設定

始める前に

- 目的の VRF、ブリッジドメイン、IPv6 アドレスを持つレイヤ 3 Out インターフェイスは、PIM6 が有効になるように設定する必要があります。レイヤ 3 Out の場合、IPv6 マルチキャストが機能するために、論理ノードプロファイルのノードに IPv6 ループバック アドレスが設定されます。
- 基本的なユニキャスト ネットワークを設定する必要があります。

手順

ステップ 1 VRF で PIM6 を有効にします。

例：

```

<fvTenant name="t0">
  <fvCtx name="ctx1" pcEnfPref="unenforced" >
    <pimIPv6CtxP ctrl="" mtu="1500" />
  </fvCtx>
</fvTenant>

```

ステップ 2 レイヤ 3 Out で PIM6 を有効にします。

例：

```

<fvTenant dn="uni/tn-t0" name="t0">
  <l3extOut enforceRtctrl="export" name="bl_l3out_1">
    <pimExtP enabledAf="ipv6-mcast" name="pim"/>
  </l3extOut>
</fvTenant>

```

ステップ 3 BD で PIM6 を有効にします。

例：

```

<fvTenant name="t0" >
  <fvBD name="BD_VPC5" ipv6McastAllow="yes" >
    <fvRsCtx tnFvCtxName="ctx1" />
    <fvSubnet ip="124:1::ffff:ffff:ffff:0/64" scope="public"/>
  </fvBD>
</fvTenant>

```

ステップ 4 スタティック ランデブー ポイントの設定

例：

```

<fvTenant name="t0">
  <pimRouteMapPol dn="uni/tn-t0/rtmap-static_101_ipv6" name="static_101_ipv6">
    <pimRouteMapEntry action="permit" grp="ff00::/8" order="1" rp="2001:0:2001:2001:1:1:1:1/128"
src="::"/>
  </pimRouteMapPol>
  <fvCtx name="ctx1" pcEnfPref="unenforced">
    <pimIPv6CtxP ctrl=" " mtu="1500">
      <pimStaticRPPol>
        <pimStaticRPEntPol rpIp="2001:0:2001:2001:1:1:1:1">
          <pimRPGrpRangePol>
            <rtDmcsFilterToRtMapPol tDn="uni/tn-t0/rtmap-static_101_ipv6"/>
          </pimRPGrpRangePol>
        </pimStaticRPEntPol>
      </pimStaticRPPol>
    </pimIPv6CtxP>
  </fvCtx>
</fvTenant>

```

ステップ 5 PIM6 インターフェイス ポリシーを設定し、レイヤ 3 Out に適用します。

例 :

```

<fvTenant dn="uni/tn-t0" name="t0">
  <l3extOut enforceRtctrl="export" name="bl_l3out_1">
    <l3extLNodeP annotation=" " configIssues=" " descr=" " name="common_npl" nameAlias=" " ownerKey=" "
ownerTag=" " tag="yellow-green" targetDscp="unspecified">
      <l3extLIIfP annotation=" " descr=" " name="common_intpl_v6" nameAlias=" " ownerKey=" " ownerTag=" "
prio="unspecified" tag="yellow-green">
        <pimIPv6IfP annotation=" " descr=" " name=" " nameAlias=" ">
          <pimRsV6IfPol annotation=" " tDn="uni/tn-common/pimifpol-pimv6_policy"/>
        </pimIPv6IfP>
      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>

```

PIM6 を使用したレイヤ 3 IPv6 マルチキャストが有効になります。

REST API を使用したマルチキャスト フィルタリングの設定

ブリッジ ドメイン レベルでマルチキャスト フィルタリングを設定します。このトピックの手順を使用して、ブリッジ ドメイン レベルで送信元フィルタリングまたは受信者フィルタリング、あるいはその両方を設定します。

始める前に

- マルチキャストフィルタリングを設定するブリッジドメインはすでに作成されています。
- ブリッジドメインは PIM 対応ブリッジドメインです。
- レイヤ 3 マルチキャストは VRF レベルで有効になります。

手順

ステップ 1 ブリッジ ドメインでマルチキャスト *source* フィルタリングを有効にする場合は、次の例のように XML で POST を送信します：

例：

```
<fvBD dn="uni/tn-filter/BD-BD1520" ipv6McastAllow="no" mcastAllow="yes">
  <pimBDP annotation="" descr="" name="" nameAlias="" ownerKey="" ownerTag="">
    <pimBDFilterPol annotation="" descr="" name="" nameAlias="">
      <pimBDSrcFilterPol annotation="" descr="" name="" nameAlias="">
        <rtdmcRsFilterToRtMapPol tDn="uni/tn-filter/rtmap-test_src_filter"/>
      </pimBDSrcFilterPol>
    </pimBDFilterPol>
  </pimBDP>
</fvBD>
```

ステップ 2 ブリッジ ドメインでマルチキャスト *receiver* フィルタリングを有効にする場合は、次の例のように XML で POST を送信します：

例：

```
<fvBD dn="uni/tn-filter/BD-BD1520" ipv6McastAllow="no" mcastAllow="yes">
  <pimBDP annotation="" descr="" name="" nameAlias="" ownerKey="" ownerTag="">
    <pimBDFilterPol annotation="" descr="" name="" nameAlias="">
      <pimBDDestFilterPol annotation="" descr="" name="" nameAlias="">
        <rtdmcRsFilterToRtMapPol tDn="uni/tn-filter/rtmap-Recv_filter"/>
      </pimBDDestFilterPol>
    </pimBDFilterPol>
  </pimBDP>
</fvBD>
```

(注)

また、次の例のように XML で POST を送信することで、同じブリッジ ドメインで送信元と受信者の両方のフィルタリングを有効にすることもできます。

```
<fvBD dn="uni/tn-filter/BD-BD1520" ipv6McastAllow="no" mcastAllow="yes">
  <pimBDP annotation="" descr="" name="" nameAlias="" ownerKey="" ownerTag="">
    <pimBDFilterPol annotation="" descr="" name="" nameAlias="">
      <pimBDSrcFilterPol annotation="" descr="" name="" nameAlias="">
        <rtdmcRsFilterToRtMapPol tDn="uni/tn-filter/rtmap-test_src_filter"/>
      </pimBDSrcFilterPol>
      <pimBDDestFilterPol annotation="" descr="" name="" nameAlias="">
        <rtdmcRsFilterToRtMapPol tDn="uni/tn-filter/rtmap-Recv_filter"/>
      </pimBDDestFilterPol>
    </pimBDFilterPol>
  </pimBDP>
</fvBD>
```


REST API を使用したマルチポッドの設定

REST API を使用したマルチポッド ファブリックのセットアップ

手順

ステップ1 Cisco APIC へのログイン :

例 :

```
http://<apic-name/ip>:80/api/aaaLogin.xml

data: <aaaUser name="admin" pwd="ins3965!"/></>
```

ステップ2 TEP プールの設定 :

例 :

```
http://<apic-name/ip>:80/api/policymgr/mo/uni/controller.xml

<fabricSetupPol status=''>
  <fabricSetupP podId="1" tepPool="10.0.0.0/16" />
  <fabricSetupP podId="2" tepPool="10.1.0.0/16" status='' />
</fabricSetupPol>
```

ステップ3 ノード ID ポリシーの設定 :

例 :

```
http://<apic-name/ip>:80/api/node/mo/uni/controller.xml

<fabricNodeIdentPol>
<fabricNodeIdentP serial="SAL1819RXP4" name="ifav4-leaf1" nodeId="101" podId="1"/>
<fabricNodeIdentP serial="SAL1803L25H" name="ifav4-leaf2" nodeId="102" podId="1"/>
<fabricNodeIdentP serial="SAL1934MNY0" name="ifav4-leaf3" nodeId="103" podId="1"/>
<fabricNodeIdentP serial="SAL1934MNY3" name="ifav4-leaf4" nodeId="104" podId="1"/>
<fabricNodeIdentP serial="SAL1748H56D" name="ifav4-spine1" nodeId="201" podId="1"/>
<fabricNodeIdentP serial="SAL1938P7A6" name="ifav4-spine3" nodeId="202" podId="1"/>
<fabricNodeIdentP serial="SAL1938PHBB" name="ifav4-leaf5" nodeId="105" podId="2"/>
<fabricNodeIdentP serial="SAL1942R857" name="ifav4-leaf6" nodeId="106" podId="2"/>
<fabricNodeIdentP serial="SAL1931LA3B" name="ifav4-spine2" nodeId="203" podId="2"/>
<fabricNodeIdentP serial="FGE173400A9" name="ifav4-spine4" nodeId="204" podId="2"/>
</fabricNodeIdentPol>
```

ステップ4 インフラ L3Out および外部接続プロファイルの設定 :

例 :

```
http://<apic-name/ip>:80/api/node/mo/uni.xml

<polUni>

<fvTenant descr="" dn="uni/tn-infra" name="infra" ownerKey="" ownerTag="">

  <l3extOut descr="" enforceRtctrl="export" name="multipod" ownerKey="" ownerTag=""
targetDscp="unspecified" status=''>
    <ospfExtP areaId='0' areaType='regular' status='' />
    <l3extRsEctx tnFvCtxName="overlay-1"/>
    <l3extProvLbl descr="" name="prov_mpl" ownerKey="" ownerTag="" tag="yellow-green"/>
  </l3extOut>
</fvTenant>
</polUni>
```

```

<l3extLNodeP name="bSpine">
  <l3extRsNodeL3OutAtt rtrId="201.201.201.201" rtrIdLoopBack="no" tDn="topology/pod-1/node-201">

    <l3extInfraNodeP descr="" fabricExtCtrlPeering="yes" name="" />
    <l3extLoopBackIfP addr="201::201/128" descr="" name="" />
    <l3extLoopBackIfP addr="201.201.201.201/32" descr="" name="" />
  </l3extRsNodeL3OutAtt>

  <l3extRsNodeL3OutAtt rtrId="202.202.202.202" rtrIdLoopBack="no" tDn="topology/pod-1/node-202">

    <l3extInfraNodeP descr="" fabricExtCtrlPeering="yes" name="" />
    <l3extLoopBackIfP addr="202::202/128" descr="" name="" />
    <l3extLoopBackIfP addr="202.202.202.202/32" descr="" name="" />
  </l3extRsNodeL3OutAtt>

  <l3extRsNodeL3OutAtt rtrId="203.203.203.203" rtrIdLoopBack="no" tDn="topology/pod-2/node-203">

    <l3extInfraNodeP descr="" fabricExtCtrlPeering="yes" name="" />
    <l3extLoopBackIfP addr="203::203/128" descr="" name="" />
    <l3extLoopBackIfP addr="203.203.203.203/32" descr="" name="" />
  </l3extRsNodeL3OutAtt>

  <l3extRsNodeL3OutAtt rtrId="204.204.204.204" rtrIdLoopBack="no" tDn="topology/pod-2/node-204">

    <l3extInfraNodeP descr="" fabricExtCtrlPeering="yes" name="" />
    <l3extLoopBackIfP addr="204::204/128" descr="" name="" />
    <l3extLoopBackIfP addr="204.204.204.204/32" descr="" name="" />
  </l3extRsNodeL3OutAtt>

  <l3extLIfP name='portIf'>
    <l3extRsPathL3OutAtt descr='asr' tDn="topology/pod-1/paths-201/pathep-[eth1/1]"
    encap='vlan-4' ifInstT='sub-interface' addr="201.1.1.1/30" />
    <l3extRsPathL3OutAtt descr='asr' tDn="topology/pod-1/paths-201/pathep-[eth1/2]"
    encap='vlan-4' ifInstT='sub-interface' addr="201.2.1.1/30" />
    <l3extRsPathL3OutAtt descr='asr' tDn="topology/pod-1/paths-202/pathep-[eth1/2]"
    encap='vlan-4' ifInstT='sub-interface' addr="202.1.1.1/30" />
    <l3extRsPathL3OutAtt descr='asr' tDn="topology/pod-2/paths-203/pathep-[eth1/1]"
    encap='vlan-4' ifInstT='sub-interface' addr="203.1.1.1/30" />
    <l3extRsPathL3OutAtt descr='asr' tDn="topology/pod-2/paths-203/pathep-[eth1/2]"
    encap='vlan-4' ifInstT='sub-interface' addr="203.2.1.1/30" />
    <l3extRsPathL3OutAtt descr='asr' tDn="topology/pod-2/paths-204/pathep-[eth4/31]"
    encap='vlan-4' ifInstT='sub-interface' addr="204.1.1.1/30" />

    <ospfIfP>
      <ospfRsIfPol tnOspfIfPolName='ospfIfPol' />
    </ospfIfP>

  </l3extLIfP>
</l3extLNodeP>

  <l3extInstP descr="" matchT="AtleastOne" name="instp1" prio="unspecified"
  targetDscp="unspecified">
    <fvRsCustQosPol tnQosCustomPolName="" />
  </l3extInstP>
</l3extOut>

<fvFabricExtConnP descr="" id="1" name="Fabric_Ext_Conn_Pol1" rt="extended:as2-nn4:5:16" status=''>

  <fvPodConnP descr="" id="1" name="">
    <fvIp addr="100.11.1.1/32" />
  </fvPodConnP>
  <fvPodConnP descr="" id="2" name="">
    <fvIp addr="200.11.1.1/32" />
  </fvPodConnP>

```

```

</fvPodConnP>
<fvPeeringP descr="" name="" ownerKey="" ownerTag="" type="automatic_with_full_mesh"/>
<l3extFabricExtRoutingP descr="" name="ext_routing_prof_1" ownerKey="" ownerTag="">
  <l3extSubnet aggregate="" descr="" ip="100.0.0.0/8" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="200.0.0.0/8" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="201.1.0.0/16" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="201.2.0.0/16" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="202.1.0.0/16" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="203.1.0.0/16" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="203.2.0.0/16" name="" scope="import-security"/>
  <l3extSubnet aggregate="" descr="" ip="204.1.0.0/16" name="" scope="import-security"/>
</l3extFabricExtRoutingP>
</fvFabricExtConnP>
</fvTenant>
</polUni>

```

REST API を使用したリモート リーフ スイッチの設定

REST API を使用したリモート リーフ スイッチの構成

[Cisco APIC] を有効にして IPN ルータとリモート リーフ スイッチを検出し接続するには、このトピックの手順を実行します。

この例では、マルチポッド トポロジで、ポッドにリモート リーフ スイッチが接続されていることを前提としています。VRF オーバーレイ 1 とともに、インフラ テナントに構成されている 2 個の L3Outs が含まれます。

- 1 個は VLAN 4 に構成され、リモート リーフ スイッチとスパイン スイッチ両方が WAN ルータに接続されている必要があります。
- 1 個はマルチポッド内部 L3Out が VLAN5 で構成されており、一緒に展開する場合はマルチポッドとリモート リーフ機能に必要です。

手順

ステップ 1 ポッドに接続されるように 2 個のリモート リーフ スイッチに TEP プールを定義するには、次の例のように XML で POST を送信します。

例：

```

<fabricSetupPol>
  <fabricSetupP tepPool="10.0.0.0/16" podId="1" >
    <fabricExtSetupP tepPool="30.0.128.0/20" extPoolId="1"/>
  </fabricSetupP>
  <fabricSetupP tepPool="10.1.0.0/16" podId="2" >
    <fabricExtSetupP tepPool="30.1.128.0/20" extPoolId="1"/>
  </fabricSetupP>
</fabricSetupPol>

```

ステップ 2 ノードのアイデンティティ ポリシーを定義するには、次の例のように XML で POST を送信します。

例：

```

<fabricNodeIdentPol>
  <fabricNodeIdentP serial="SAL17267Z7W" name="leaf1" nodeId="101" podId="1" extPoolId="1"
nodeType="remote-leaf-wan"/>
  <fabricNodeIdentP serial="SAL17267Z7X" name="leaf2" nodeId="102" podId="1" extPoolId="1"
nodeType="remote-leaf-wan"/>
  <fabricNodeIdentP serial="SAL17267Z7Y" name="leaf3" nodeId="201" podId="1" extPoolId="1"
nodeType="remote-leaf-wan"/>
  <fabricNodeIdentP serial="SAL17267Z7Z" name="leaf4" nodeId="201" podId="1" extPoolId="1"
nodeType="remote-leaf-wan"/>
</fabricNodeIdentPol>

```

ステップ3 ファブリック外部接続プロファイルを構成するには、次の例のように XML で POST を送信します。

例：

```

<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="1">
  <fvFabricExtConnP dn="uni/tn-infra/fabricExtConnP-1" id="1" name="Fabric_Ext_Conn_Pol1"
rt="extended:as2-nn4:5:16" siteId="0">
    <l3extFabricExtRoutingP name="test">
      <l3extSubnet ip="150.1.0.0/16" scope="import-security"/>
    </l3extFabricExtRoutingP>
    <l3extFabricExtRoutingP name="ext_routing_prof_1">
      <l3extSubnet ip="204.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="209.2.0.0/16" scope="import-security"/>
      <l3extSubnet ip="202.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="207.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="200.0.0.0/8" scope="import-security"/>
      <l3extSubnet ip="201.2.0.0/16" scope="import-security"/>
      <l3extSubnet ip="210.2.0.0/16" scope="import-security"/>
      <l3extSubnet ip="209.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="203.2.0.0/16" scope="import-security"/>
      <l3extSubnet ip="208.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="207.2.0.0/16" scope="import-security"/>
      <l3extSubnet ip="100.0.0.0/8" scope="import-security"/>
      <l3extSubnet ip="201.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="210.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="203.1.0.0/16" scope="import-security"/>
      <l3extSubnet ip="208.2.0.0/16" scope="import-security"/>
    </l3extFabricExtRoutingP>
    <fvPodConnP id="1">
      <fvIp addr="100.11.1.1/32"/>
    </fvPodConnP>
    <fvPodConnP id="2">
      <fvIp addr="200.11.1.1/32"/>
    </fvPodConnP>
    <fvPeeringP type="automatic_with_full_mesh"/>
  </fvFabricExtConnP>
</imdata>

```

ステップ4 VLAN 4 で L3Out を構成するには、リモートリーフスイッチとスパインスイッチ両方が WAN ルータに接続され、次の例のように XML を入力する必要があります。

例：

```

<?xml version="1.0" encoding="UTF-8"?>
<polUni>
  <fvTenant name="infra">
    <l3extOut name="rleaf-wan-test">
      <ospfExtP areaId="0.0.0.5"/>
      <bgpExtP/>
      <l3extRsEctx tnFvCtxName="overlay-1"/>
      <l3extRsL3DomAtt tDn="uni/l3dom-l3extDom1"/>
      <l3extProvLbl descr="" name="prov_mp1" ownerKey="" ownerTag="" tag="yellow-green"/>
    </l3extOut>
  </fvTenant>
</polUni>

```

```

<l3extLNodeP name="rleaf-101">
  <l3extRsNodeL3OutAtt rtrId="202.202.202.202" tDn="topology/pod-1/node-101">
  </l3extRsNodeL3OutAtt>
  <l3extLIIfP name="portIf">
    <l3extRsPathL3OutAtt ifInstT="sub-interface" tDn="topology/pod-1/paths-101/pathep-[eth1/49]"
    addr="202.1.1.2/30" mac="AA:11:22:33:44:66" encap='vlan-4' />
    <ospfIfP>
      <ospfRsIfPol tnOspfIfPolName='ospfIfPol' />
    </ospfIfP>
  </l3extLIIfP>
</l3extLNodeP>
<l3extLNodeP name="rlSpine-201">
  <l3extRsNodeL3OutAtt rtrId="201.201.201.201" rtrIdLoopBack="no" tDn="topology/pod-1/node-201">

  <!--
  <l3extLoopBackIfP addr="201::201/128" descr="" name="" />
  <l3extLoopBackIfP addr="201.201.201.201/32" descr="" name="" />
  -->
  <l3extLoopBackIfP addr="::" />
</l3extRsNodeL3OutAtt>
  <l3extLIIfP name="portIf">
    <l3extRsPathL3OutAtt ifInstT="sub-interface" tDn="topology/pod-1/paths-201/pathep-[eth8/36]"
    addr="201.1.1.1/30" mac="00:11:22:33:77:55" encap='vlan-4' />
    <ospfIfP>
      <ospfRsIfPol tnOspfIfPolName='ospfIfPol' />
    </ospfIfP>
  </l3extLIIfP>
</l3extLNodeP>
<l3extInstP descr="" matchT="AtleastOne" name="instpl" prio="unspecified" targetDscp="unspecified">

  <fvRsCustQosPol tnQosCustomPolName="" />
</l3extInstP>
</l3extOut>
  <ospfIfPol name="ospfIfPol" nwT="bcast" />
</fvTenant>
</polUni>

```

ステップ 5 リリース 4.1 (2) 以前で、VLAN-5 でマルチポッド L3Out を構成するには、マルチポッドとリモート リーフ トポロジの両方と、次の例のように XML を送信する必要があります。

(注)

リリース 4.1 (2) 以降を実行している新しいリモート リーフ スイッチを導入し、それらのリモート リーフ スイッチで直接トラフィック転送をイネーブルにする場合は、この情報を入力しないでください。この場合、マルチポッドに VLAN-5 を使用して OSPF インスタンスを構成する必要はありません。

詳細については、[ダイレクトトラフィックフォワーディングについて](#) を参照してください。

例：

```

<?xml version="1.0" encoding="UTF-8"?>
<polUni>

  <fvTenant name="infra" >
    <l3extOut name="ipn-multipodInternal">
      <ospfExtP areaCtrl="inherit-ipsec,redistribute,summary" areaId="0.0.0.5" multipodInternal="yes"
      />
      <l3extRsEctx tnFvCtxName="overlay-1" />
      <l3extLNodeP name="bLeaf">
        <l3extRsNodeL3OutAtt rtrId="202.202.202.202" rtrIdLoopBack="no" tDn="topology/pod-2/node-202">

          <l3extLoopBackIfP addr="202.202.202.212" />
        </l3extRsNodeL3OutAtt>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>

```

```

<l3extRsNodeL3OutAtt rtrId="102.102.102.102" rtrIdLoopBack="no" tDn="topology/pod-1/node-102">
  <l3extLoopBackIfP addr="102.102.102.112"/>
</l3extRsNodeL3OutAtt>
<l3extLIIfP name="portIf">
  <ospfIfP authKeyId="1" authType="none">
    <ospfRsIfPol tnOspfIfPolName="ospfIfPol" />
  </ospfIfP>
  <l3extRsPathL3OutAtt addr="10.0.254.233/30" encap="vlan-5" ifInstT="sub-interface"
tDn="topology/pod-2/paths-202/pathep-[eth5/2]"/>
  <l3extRsPathL3OutAtt addr="10.0.255.229/30" encap="vlan-5" ifInstT="sub-interface"
tDn="topology/pod-1/paths-102/pathep-[eth5/2]"/>
  </l3extLIIfP>
</l3extLNodeP>
<l3extInstP matchT="AtleastOne" name="ipnInstP" />
</l3extOut>
</fvTenant>
</polUni>

```

REST API を使用した SR-MPLS ハンドオフの設定

REST API を使用した SR-MPLS インフラ L3Out の設定

- SR-MPLS インフラ L3Out は、境界リーフスイッチで設定され、SR-MPLS ハンドオフに必要なアンダーレイ BGP-LU およびオーバーレイ MP-BGP EVPN セッションを設定するために使用されます。
- SR-MPLS インフラ L3Out は、ポッドまたはリモートリーフスイッチサイトにスコープされます。
- 1 つの SR-MPLS インフラ L3Out 内の境界リーフスイッチまたはリモートリーフスイッチは、1 つ以上のルーティングドメイン内の 1 つ以上のプロバイダーエッジ (PE) ルータに接続できます。
- ポッドまたはリモートリーフスイッチサイトには、1 つ以上の SR-MPLS インフラ L3Out を設定できます。
- 各 SR-MPLS インフラ L3Out には、一意のプロバイダーラベルと 1 つのプロバイダーラベルのみが必要です。各 SR-MPLS インフラ L3Out は、プロバイダーラベルによって識別されます。

SR-MPLS インフラ L3Out を設定する場合は、次の項目を設定します。

- ノード
 - リーフスイッチのみが SR-MPLS インフラ L3Out のノードとして設定できます (境界リーフスイッチおよびリモートリーフスイッチ)。
 - 各 SR-MPLS インフラ L3Out は、1 つのポッドからの境界リーフスイッチまたは同じサイトからのリモートリーフスイッチを持つことができます。

- 各境界リーフ スイッチまたはリモートリーフ スイッチは、複数の SR-MPLS ドメインに接続する場合、複数の SR-MPLS インフラ L3Out で設定できます。
- また、ノードの下にループバックインターフェイスを設定し、ループバックインターフェイスの下にノード SID ポリシーを設定します。

• インターフェイス

- サポートされるインターフェイスのタイプは次のとおりです。
 - ルーテッド インターフェイスまたはサブインターフェイス
 - ルーテッド ポートチャネルまたはポートチャネル サブインターフェイス

サブインターフェイスでは、任意の VLAN タグがサポートされます。

- また、SR-MPLS infra L3Out のインターフェイスエリアの下にアンダーレイ BGP ピアポリシーを設定します。

• QoS ルール

- MPLS 入力ルールと MPLS 出力ルールは、SR-MPLS インフラ L3Out の MPLS QoS ポリシーを使用して設定できます。
- MPLS QoS ポリシーを作成しない場合、入力 MPLS トラフィックにはデフォルトの QoS レベルが割り当てられます。

また、SR-MPLS インフラ L3Out を使用してアンダーレイとオーバーレイを設定します。

- **アンダーレイ**：インターフェイス構成の一部としての BGP ピア IP (BGPLU ピア) 構成。
- **オーバーレイ**：論理ノードプロファイル構成の一部としての MP-BGP EVPN リモート IPv4 アドレス (MP-BGP EVPN ピア) 構成。

始める前に

- 次に提供されている SR-MPLS ガイドラインと制限事項を確認します。 [SR-MPLS のガイドラインおよび制限事項](#)特に、[SR-MPLS インフラ L3Out のガイドラインと制約事項](#)に提供されているガイドラインと制限を確認します。
- (オプション) 必要に応じて、[REST API を使用した SR-MPLS カスタム QoS ポリシー \(26 ページ\)](#) の手順を使用して MPLS カスタム QoS ポリシーを構成します。

手順

次のような情報が表示されます。

<polUni>

REST API を使用した SR-MPLS VRF L3Out の設定

```

<fvTenant name="infra">
  <mplsIfPol name="default"/>
  <mplsLabelPol name="default" >
    <mplsSrgbLabelPol minSrgbLabel="16000" maxSrgbLabel="17000" localId="1" status=""/>
  </mplsLabelPol>

  <l3extOut name="mplsOut" status="" descr="b1" mplsEnabled="yes">
    <l3extRsEctx tnFvCtxName="overlay-1"/>
    <l3extProvLbl name="mpls" />
    <mplsExtP status="" >
      <mplsRsLabelPol tDn="uni/tn-infra/mplslabelpol-default"/>
    </mplsExtP>
    <l3extLNodeP name="mplsLNP" status="">
      <l3extRsNodeL3OutAtt rtrId="100.1.1.1" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"
status="">
        <l3extLoopBackIfP addr="10.10.10.11" status="">
          <mplsNodeSidP sidoffset="2" loopbackAddr="10.1.3.11" status=""/>
        </l3extLoopBackIfP>
      </l3extRsNodeL3OutAtt>

      <l3extLIIfP name="mplsLIIfP1" status="">
        <mplsIfP status="">
          <mplsRsIfPol tnMplsIfPolName="default" />
        </mplsIfP>
        <l3extRsPathL3OutAtt addr="34.1.2.3/30" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/8]">
          <bgpPeerP addr="9.9.9.7" addrTCtrl="af-ucast,af-label-ucast" ctrl="send-ext-com"
ttl="1" status="">
            <bgpAsP asn="100"/>
          </bgpPeerP>
        </l3extRsPathL3OutAtt>
      </l3extLIIfP>
      <bgpInfraPeerP addr="20.1.1.1" ctrl="send-com,send-ext-com" peerT="sr-mpls" ttl="3" status=""
>
        <bgpAsP asn="100"/>
      </bgpInfraPeerP>
    </l3extLNodeP>

    <l3extInstP name="mplsInstP">
      <l3extSubnet aggregate="" descr="" ip="11.11.11.0/24" name="" scope="import-security"/>
    </l3extInstP>
    <bgpExtP/>
    <l3extRsL3DomAtt tDn="uni/l3dom-l3extDom1" />
  </l3extOut>

</fvTenant>
</polUni>

```

REST API を使用した SR-MPLS VRF L3Out の設定

この項の手順を使用して、SR-MPLS VRF L3Out を設定します。これは、前の手順で設定した SR-MPLS インフラ L3Out からのトラフィックの転送に使用されます。

- ユーザ テナント VRF は SR-MPLS インフラ L3Out にマッピングされ、テナントブリッジドメイン サブネットを DC-PE ルータにアドバタイズし、DC-PE から受信した MPLS VPN ルートをインポートします。

- 各 VRF の SR-MPLS VRF L3Out でルーティング ポリシーとセキュリティ ポリシーを指定する必要があります。これらのポリシーは、1 つ以上の SR-MPLS インフラ L3Out をポイントします。
- VRF ごとに 1 つの SR-MPLS VRF L3Out がサポートされます。
- 1 つの SR-MPLS VRF L3Out で複数のコンシューマ ラベルを設定でき、各コンシューマ ラベルで 1 つの SR-MPLS インフラ L3Out を識別できます。コンシューマ ラベルは、特定のポッドまたはリモート リーフ スイッチの特定の MPLS ドメインである 1 つの SR-MPLS VRF L3Out との間のトラフィックのエントリ ポイントと出口ポイントを識別します。

始める前に

- 次に提供されている SR-MPLS ガイドラインと制限事項を確認します。 [SR-MPLS のガイドラインおよび制限事項](#)特に、 [SR-MPLS VRF L3Out のガイドラインと制約事項](#)で提供されているガイドラインと制限事項を確認します。
- 次に提供されている手順に従って、SR-MPLS インフラ L3Out を構成します。 [REST API を使用した SR-MPLS インフラ L3Out の設定 \(22 ページ\)](#)。

手順

次のような情報が表示されます。

```
<polUni>
<fvTenant name="t1">
  <fvCtx name="v1">
    <!-- specify bgp evpn route-target -->
    <bgpRtTargetP af="ipv4-ucast">
      <bgpRtTarget rt="route-target:as4-nn2:100:1259" type="import"/>
      <bgpRtTarget rt="route-target:as4-nn2:100:1259" type="export"/>
    </bgpRtTargetP>
  </fvCtx>

  <!-- MPLS L3out -->
  <l3extOut name="out1" mplsEnabled="yes">
    <l3extRsEctx tnFvCtxName="v1" />

    <!-- MPLS consumer label -->
    <l3extConsLbl name="mpls1">
      <!-- route profile association -->
      <l3extRsLblToProfile tDn="uni/tn-t1/prof-rpl" direction="export" />
      <!-- InstP association -->
      <l3extRsLblToInstP tDn="uni/tn-t1/out-out1/instP-epgMpls1" />
    </l3extConsLbl>

    <!-- External-EPG -->
    <l3extInstP name="epgMpls1">
      <fvRsProv tnVzBrCPName="cp1"/>
      <l3extSubnet ip="55.1.1.1/28"/>
    </l3extInstP>
    <bgpExtP/>
  </l3extOut>
```

```

<!-- route control profile -->
<rtctrlProfile descr="" name="rp1" type="global" status="">
  <rtctrlCtxP action="permit" descr="" name="ctx1" order="0">
    <rtctrlRsCtxPToSubjP status="" tnRtctrlSubjPName="subj1"/>
  </rtctrlCtxP>
</rtctrlProfile>
<rtctrlSubjP descr="" name="subj1" status="" >
  <rtctrlMatchRtDest ip="101.1.1.1/32"/>
  <rtctrlMatchRtDest ip="102.1.1.0/24" aggregate="yes"/>
</rtctrlSubjP>

<!-- Filter and Contract (global) -->
<vzBrCP name="cp1" scope="global">
  <vzSubj name="allow-all">
    <vzRsSubjFiltAtt action="permit" tnVzFilterName="default" />
  </vzSubj>
</vzBrCP>
</fvTenant>
</polUni>

```

REST API を使用した SR-MPLS カスタム QoS ポリシー

SR MPLS カスタム QoS ポリシーは、MPLS QoS 出力 ポリシーで定義された着信 MPLS EXP 値に基づいて、SR-MPLS ネットワークから送信されるパケットのプライオリティを定義します。これらのパケットは、ACI ファブリック内にあります。また、MPLS QoS 出力ポリシーで定義された IPv4 DSCP 値に基づく MPLS インターフェイスを介して ACI ファブリックから離れるパケットの CoS 値および MPLS EXP 値をマーキングします。

カスタム出力ポリシーが定義されていない場合、デフォルトの QoS レベル（レベル3）がファブリック内のパケットに割り当てられます。カスタム イングレス ポリシーが定義されていない場合、0 のデフォルト EXP 値は、ファブリックから離れるパケットにマークされます。

手順

ステップ1 SR-MPLS QoS ポリシーの作成します。

次の POST で：

- *customqos1* を、作成する SR-MPLS QoS ポリシーの名前に置き換えます。
- qosMplsIngressRule の場合：
 - from="2" to="3" を、ポリシーに一致させる EXP 範囲に置き換えます。
 - prio =level5" を ACI ファブリック内にあるパケットの ACI QoS レベルに置き換えます。
 - target="CS5" は、パケットが一致したときに設定する DSCP 値に置き換えます。
 - targetCos =4" を、パケットが一致したときにパケットに設定する CoS 値に置き換えます。

- qosMplsEgressRule の場合：
 - from="CS2" to="CS4" を、ポリシーを照合する DSCP 範囲に置き換えます。
 - targetExp =5" を、パケットがファブリックを離れるときに設定する EXP 値に置き換えます。
 - targetCos =3" を、パケットがファブリックを離れるときに設定する CoS 値に置き換えます。

```
<polUni>
  <fvTenant name="infra">
    <qosMplsCustomPol descr="" dn="uni/tn-infra/qosmplscustom-">customqos1" name="customqos1"
status="" >
      <qosMplsIngressRule from="">2" to="3" prio="level5" target="CS5" targetCos="4" status="" />

      <qosMplsEgressRule from="">CS2" to="CS4" targetExp="5" targetCos="3" status="" />
    </qosMplsCustomPol>
  </fvTenant>
</polUni>
```

ステップ2 SR-MPLS QoS ポリシーの適用します。

次の POST で、次を置換します。 *customqos1* を前の手順で作成した SR-MPLS QoS ポリシーの名前に置き換えます。

```
<polUni>
  <fvTenant name="infra">
    <l3extOut name="mplsOut" status="" descr="bl">
      <l3extLNodeP name="mplsLNP" status="">
        <l3extRsLNodePMplsCustQosPol tDn="uni/tn-infra/qosmplscustom-">customqos1"/>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>
```

パートII：外部ルーティング（L30ut）の設定

外部ネットワークへのルーテッド接続

REST API を使用した MP-BGP ルート リフレクタの設定

REST API を使用した MP-BGP ルート リフレクタの設定

手順

ステップ1 スパイン スイッチをルート リフレクタとしてマークします。

例：

POST <https://apic-ip-address/api/policymgr/mo/uni/fabric.xml>

REST API を使用したループ防止のための BGP ドメインパス機能の設定

```
<bgpInstPol name="default">
  <bgpAsP asn="1" />
  <bgpRRP>
    <bgpRRNodePEp id="<spine_id1>" />
    <bgpRRNodePEp id="<spine_id2>" />
  </bgpRRP>
</bgpInstPol>
```

ステップ 2 次のポストを使用してポッドセクタをセットアップします。

例：

FuncP セットアップの場合：

POST <https://apic-ip-address/api/policymgr/mo/uni.xml>

```
<fabricFuncP>
  <fabricPodPGrp name="bgpRRPodGrp"      <fabricRsPodPGrpBGPGRP tnBgpInstPolName="default" />
</fabricPodPGrp>
</fabricFuncP>
```

例：

PodP セットアップの場合：

POST <https://apic-ip-address/api/policymgr/mo/uni.xml>

```
<fabricPodP name="default">
  <fabricPodS name="default" type="ALL">
    <fabricRsPodPGrp tDn="uni/fabric/funcprof/podpgrp-bgpRRPodGrp" />
  </fabricPodS>
</fabricPodP>
```

REST API を使用したループ防止のための BGP ドメインパス機能の設定

始める前に

[ループ防止のための BGP ドメインパス機能について](#)に記載されている情報を使用して、BGP ドメインパス機能に精通します。

手順

ステップ 1 ループ防止に BGP ドメインパス機能を使用する場合は、グローバル DomainIdBaseを設定します。

```
<polUni>
  <fabricInst>
    <bgpInstPol name="default">
      <bgpDomainIdBase domainIdBase="12346" />
    </bgpInstPol>
  </fabricInst>
</polUni>
```

ステップ 2 適切な L3Out で send-domain-path を有効にします。

```
<bgpPeerP addr="22.22.3.5" addrTctrl="af-ucast" allowedSelfAsCnt="3" ttl="2"
>ctrlExt="send-domain-path" ctrl="send-ext-com">
</bgpPeerP>
```

L3Out のノードとインターフェイス

REST API を使用したレイヤ 3 ルーテッド ポート チャンネルとサブインターフェイス ポート チャンネルの設定

REST API を使用したレイヤ 3 ルーテッド ポート チャンネルの構成

始める前に

- ACI ファブリックがインストールされています。[APIC] コントローラがオンラインで、[APIC] クラスターが形成され、正常に動作していることを示します。
- 必要なファブリック インフラストラクチャ設定を作成できる [APIC] ファブリック管理者 アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。
- ポートチャンネルは、L3Out インターフェイスにポートチャンネルが使用される場合に設定されます。



(注) 次の REST API 例では、長い 1 行のテキストは \ で分けて読みやすくします。

手順

REST API を使用して以前作成したポートチャンネルにレイヤ 3 ルートを構成するには、次のように XML で post を送信します：

例：

```
<polUni>
<fvTenant name=pep9>
  <l3extOut descr="" dn="uni/tn-pep9/out-routAccounting" enforceRtctrl="export" \
name="routAccounting" nameAlias="" ownerKey="" ownerTag="" \      targetDscp="unspecified">
    <l3extRsL3DomAtt tDn="uni/l3dom-Dom1"/>
    <l3extRsEctx tnFvCtxName="ctx9"/>
    <l3extLNodeP configIssues="" descr="" name="node101" nameAlias="" ownerKey="" \
ownerTag="" tag="yellow-green" targetDscp="unspecified">
      <l3extRsNodeL3OutAtt rtrId="10.1.0.101" rtrIdLoopBack="yes" \
tDn="topology/pod-1/node-101">
        <l3extInfraNodeP descr="" fabricExtCtrlPeering="no" \
fabricExtIntersiteCtrlPeering="no" name="" nameAlias="" spineRole="">
```

REST API を使用して、レイヤ3 サブインターフェイス ポート チャンネルの構成

```

</l3extRsNodeL3OutAtt>
<l3extLIfP descr="" name="lifp17" nameAlias="" ownerKey="" ownerTag="" \
tag="yellow-green">
  <ospfIfP authKeyId="1" authType="none" descr="" name="" nameAlias="">
    <ospfRsIfPol tnOspfIfPolName=""/>
  </ospfIfP>
  <l3extRsPathL3OutAtt addr="10.1.5.3/24" autostate="disabled" descr="" \
encap="unknown" encapScope="local" ifInstT="l3-port" llAddr="::" \
mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit" \
tDn="topology/pod-1/paths-101/pathep-[pol17_PolGrp]" \
targetDscp="unspecified"/>
  <l3extRsNdIfPol tnNdIfPolName=""/>
  <l3extRsIngressQosDppPol tnQosDppPolName=""/>
  <l3extRsEgressQosDppPol tnQosDppPolName=""/>
</l3extLIfP>
</l3extLNodeP>
<l3extInstP descr="" floodOnEncap="disabled" matchT="AtleastOne" \
name="accountingInst" nameAlias="" prefGrMemb="exclude" prio="unspecified" \
targetDscp="unspecified">
  <fvRsProv matchT="AtleastOne" prio="unspecified" tnVzBrCPName="webCtrct"/>
  <l3extSubnet aggregate="export-rtctrl,import-rtctrl" descr="" ip="0.0.0.0/0" \
name="" nameAlias="" scope="export-rtctrl,import-rtctrl,import-security"/>
  <l3extSubnet aggregate="export-rtctrl,import-rtctrl" descr="" ip="::/0" \
name="" nameAlias="" scope="export-rtctrl,import-rtctrl,import-security"/>
  <fvRsCustQosPol tnQosCustomPolName=""/>
</l3extInstP>
<l3extConsLbl descr="" name="golf" nameAlias="" owner="infra" ownerKey="" \
ownerTag="" tag="yellow-green"/>
</l3extOut>
</fvTenant>
</polUni>

```

REST API を使用して、レイヤ3 サブインターフェイス ポート チャンネルの構成

始める前に

- ACI ファブリックがインストールされています。[APIC] コントローラがオンラインで、[APIC] クラスターが形成され、正常に動作していることを示します。
- 必要なファブリック インフラストラクチャ構成を作成できる [APIC] ファブリック管理者 アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。
- ポート チャンネルは、「REST API を使用したポート チャンネルの構成」の手順を使用して構成されます。



(注) 次の REST API 例では、1 つ以上の行のテキストはで区分するが、\ 読みやすさを改善する文字。

手順

REST API を使用して、以前に作成したポート チャネルをレイヤ 3 サブインターフェイス ルートを構成するには、次のようには、XML で post を送信します：

例：

```
<polUni>
<fvTenant name=pep9>
  <l3extOut descr="" dn="uni/tn-pep9/out-routAccounting" enforceRtctrl="export" \
name="routAccounting" nameAlias="" ownerKey="" ownerTag="" targetDscp="unspecified">
    <l3extRsL3DomAtt tDn="uni/l3dom-Dom1"/>
    <l3extRsEctx tnFvCtxName="ctx9"/>
    <l3extLNodeP configIssues="" descr="" name="node101" nameAlias="" ownerKey="" \
ownerTag="" tag="yellow-green" targetDscp="unspecified">
      <l3extRsNodeL3OutAtt rtrId="10.1.0.101" rtrIdLoopBack="yes" \
tDn="topology/pod-1/node-101">
        <l3extInfraNodeP descr="" fabricExtCtrlPeering="no" \
fabricExtIntersiteCtrlPeering="no" name="" nameAlias="" spineRole="">
          <l3extRsNodeL3OutAtt>
            <l3extLIfP descr="" name="lifp27" nameAlias="" ownerKey="" ownerTag="" \
tag="yellow-green">
              <ospfIfP authKeyId="1" authType="none" descr="" name="" nameAlias="">
                <ospfRsIfPol tnOspfIfPolName=""/>
              </ospfIfP>
              <l3extRsPathL3OutAtt addr="11.1.5.3/24" autostate="disabled" descr="" \
encap="vlan-2001" encapScope="local" ifInstT="sub-interface" \ llAddr="::"
mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit" \
tDn="topology/pod-1/paths-101/pathep-[po27_PolGrp]" \ targetDscp="unspecified"/>
                <l3extRsNdIfPol tnNdIfPolName=""/>
                <l3extRsIngressQosDppPol tnQosDppPolName=""/>
                <l3extRsEgressQosDppPol tnQosDppPolName=""/>
              </l3extLIfP>
            </l3extLNodeP>
            <l3extInstP descr="" floodOnEncap="disabled" matchT="AtleastOne" \
name="accountingInst" nameAlias="" prefGrMemb="exclude" prio="unspecified" \
targetDscp="unspecified">
              <fvRsProv matchT="AtleastOne" prio="unspecified" tnVzBrCPName="webCtrct"/>
              <l3extSubnet aggregate="export-rtctrl,import-rtctrl" descr="" ip="0.0.0.0/0" \
name="" nameAlias="" scope="export-rtctrl,import-rtctrl,import-security"/>
              <l3extSubnet aggregate="export-rtctrl,import-rtctrl" descr="" ip="::/0" \
name="" nameAlias="" scope="export-rtctrl,import-rtctrl,import-security"/>
              <fvRsCustQosPol tnQosCustomPolName=""/>
            </l3extInstP>
            <l3extConsLbl descr="" name="golf" nameAlias="" owner="infra" ownerKey="" \
ownerTag="" tag="yellow-green"/>
          </l3extOut>
        </fvTenant>
      </polUni>
```

REST API を使用したスイッチ仮想インターフェイスの設定

REST API を使用して、SVI インターフェイスのカプセル化スコープの設定

始める前に

インターフェイス セレクタが設定されます。

手順

SVI インターフェイスのカプセル化の範囲を設定します。

例：

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- /api/node/mo/.xml -->
<polUni>
  <fvTenant name="coke">
    <l3extOut descr="" dn="uni/tn-coke/out-l3out1" enforceRtctrl="export" name="l3out1" nameAlias=""
ownerKey="" ownerTag="" targetDscp="unspecified">
      <l3extRsL3DomAtt tDn="uni/l3dom-Dom1"/>
      <l3extRsEctx tnFvCtxName="vrf0"/>
      <l3extLNodeP configIssues="" descr="" name="__ui_node_101" nameAlias="" ownerKey="" ownerTag=""
tag="yellow-green" targetDscp="unspecified">
        <l3extRsNodeL3OutAtt rtrId="1.1.1.1" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"/>
        <l3extLIIfP descr="" name="intl_11" nameAlias="" ownerKey="" ownerTag="" tag="yellow-green">
          <l3extRsPathL3OutAtt addr="1.2.3.4/24" descr="" encap="vlan-2001" encapScope="ctx"
ifInstT="ext-svi" llAddr="0.0.0.0" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/5]" targetDscp="unspecified"/>
          <l3extRsNdIfPol tnNdIfPolName=""/>
          <l3extRsIngressQosDppPol tnQosDppPolName=""/>
          <l3extRsEgressQosDppPol tnQosDppPolName=""/>
        </l3extLIIfP>
      </l3extLNodeP>
      <l3extInstP descr="" matchT="AtleastOne" name="epg1" nameAlias="" prefGrMemb="exclude"
prio="unspecified" targetDscp="unspecified">
        <l3extSubnet aggregate="" descr="" ip="101.10.10.1/24" name="" nameAlias=""
scope="import-security"/>
        <fvRsCustQosPol tnQosCustomPolName=""/>
      </l3extInstP>
    </l3extOut>
  </fvTenant>
</polUni>
```

REST API を使用した SVI 自動状態の設定

始める前に

- テナントと VRF が設定されています。
- レイヤ3アウトが設定されており、レイヤ3アウトの論理ノードプロファイルと論理インターフェイス プロファイルが設定されています。

手順

SVI の自動状態の値を有効にします。

例：

```
<fvTenant name="t1" >
  <l3extOut name="out1">
    <l3extLNodeP name="__ui_node_101" >
      <l3extLIfP descr="" name="__ui_eth1_10_vlan_99_af_ipv4" >
        <l3extRsPathL3OutAtt addr="19.1.1.1/24" autostate="enabled" descr="" encap="vlan-100"
encapScope="local" ifInstT="ext-svi" l1Addr=":" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/10]" targetDscp="unspecified" />
      </l3extLIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>
```

自動状態を無効にするには、上記の例では無効に値を変更する必要があります。次に例を示します。
autostate □ 「disabled」。

REST API を使用したルーティング プロトコルの設定

REST API を使用した BFD サポート付き BGP 外部ルーテッド ネットワークの設定

REST API を使用した BGP 外部ルーテッド ネットワークの設定

始める前に

外部ルーテッド ネットワークを設定するテナントがすでに作成されていること。

ここでは、REST API を使用して BGP 外部ルーテッド ネットワークを設定する方法を示します。

例：

手順

例：

```
<l3extOut descr="" dn="uni/tn-t1/out-l3out-bgp" enforceRtctrl="export" name="l3out-bgp" ownerKey=""
ownerTag="" targetDscp="unspecified">
  <l3extRsEctx tnFvCtxName="ctx3"/>
  <l3extLNodeP configIssues="" descr="" name="l3extLNodeP_1" ownerKey="" ownerTag="" tag="yellow-green"
targetDscp="unspecified">
    <l3extRsNodeL3OutAtt rtrId="1.1.1.1" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"/>
  <l3extLIfP descr="" name="l3extLIfP_2" ownerKey="" ownerTag="" tag="yellow-green">
```

```

<l3extRsNdIfPol tnNdIfPolName=""/>
<l3extRsIngressQosDppPol tnQosDppPolName=""/>
<l3extRsEgressQosDppPol tnQosDppPolName=""/>
<l3extRsPathL3OutAtt addr="3001::31:0:1:2/120" descr="" encap="vlan-3001" encapScope="local"
ifInstT="sub-interface" llAddr=":" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/8]" targetDscp="unspecified">
  <bgpPeerP addr="3001::31:0:1:0/120" allowedSelfAsCnt="3" ctrl="send-com,send-ext-com" descr=""
name="" peerCtrl="bfd" privateASctrl="remove-all,remove-exclusive,replace-as" ttl="1" weight="1000">

    <bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
    <bgpAsP asn="3001" descr="" name=""/>
  </bgpPeerP>
</l3extRsPathL3OutAtt>
</l3extLIfP>
<l3extLIfP descr="" name="l3extLIfP_1" ownerKey="" ownerTag="" tag="yellow-green">
  <l3extRsNdIfPol tnNdIfPolName=""/>
  <l3extRsIngressQosDppPol tnQosDppPolName=""/>
  <l3extRsEgressQosDppPol tnQosDppPolName=""/>
  <l3extRsPathL3OutAtt addr="31.0.1.2/24" descr="" encap="vlan-3001" encapScope="local"
ifInstT="sub-interface" llAddr=":" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/8]" targetDscp="unspecified">
    <bgpPeerP addr="31.0.1.0/24" allowedSelfAsCnt="3" ctrl="send-com,send-ext-com" descr="" name=""
peerCtrl="" privateASctrl="remove-all,remove-exclusive,replace-as" ttl="1" weight="100">
      <bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
      <bgpLocalAsnP asnPropagate="none" descr="" localAsn="200" name=""/>
      <bgpAsP asn="3001" descr="" name=""/>
    </bgpPeerP>
  </l3extRsPathL3OutAtt>
</l3extLIfP>
</l3extLNodeP>
<l3extRsL3DomAtt tDn="uni/l3dom-l3-dom"/>
<l3extRsDampeningPol af="ipv6-ucast" tnRtctrlProfileName="damp_rp"/>
<l3extRsDampeningPol af="ipv4-ucast" tnRtctrlProfileName="damp_rp"/>
<l3extInstP descr="" matchT="AtleastOne" name="l3extInstP_1" prio="unspecified"
targetDscp="unspecified">
  <l3extSubnet aggregate="" descr="" ip="130.130.130.0/24" name="" scope="import-rtctrl"/></l3extSubnet>

  <l3extSubnet aggregate="" descr="" ip="130.130.131.0/24" name="" scope="import-rtctrl"/>
  <l3extSubnet aggregate="" descr="" ip="120.120.120.120/32" name=""
scope="export-rtctrl,import-security"/>
  <l3extSubnet aggregate="" descr="" ip="3001::130:130:130:100/120" name="" scope="import-rtctrl"/>

</l3extInstP>
<bgpExtP descr=""/>
</l3extOut>
<rtctrlProfile descr="" dn="uni/tn-t1/prof-damp_rp" name="damp_rp" ownerKey="" ownerTag=""
type="combinable">
  <rtctrlCtxP descr="" name="ipv4_rpc" order="0">
    <rtctrlScope descr="" name="">
      <rtctrlRsScopeToAttrP tnRtctrlAttrPName="act_rule"/>
    </rtctrlScope>
  </rtctrlCtxP>
</rtctrlProfile>
<rtctrlAttrP descr="" dn="uni/tn-t1/attr-act_rule" name="act_rule">
  <rtctrlSetDamp descr="" halfLife="15" maxSuppressTime="60" name="" reuse="750" suppress="2000"
type="dampening-pol"/>
</rtctrlAttrP>

```

REST API を使用した BGP パスの設定

次のフィールドの許容値の『Cisco ACI の検証済み拡張性ガイド (Verified Scalability Guide for Cisco ACI)』を参照します。これは、[\[Cisco APIC ドキュメンテーション ページ \(Cisco APIC documentation page\)\]](#) にあります。

さらに多くのパスを構成できるようにする 2 つのプロパティは次のとおりです。maxEcmp および [maxEcmpIbgp] です。これは、bgpCtxAfPol オブジェクトにあります。これら 2 つのプロパティを構成した後、実装の残り部分に反映されます。ECMP ポリシーは VRF レベルで適用されます。

次の例では、REST API を使用して BGP 最長パス機能を構成する方法の情報を提供します：

```
<fvTenant descr="" dn="uni/tn-t1" name="t1">
  <fvCtx name="v1">
    <fvRsCtxToBgpCtxAfPol af="ipv4-ucast" tnBgpCtxAfPolName="bgpCtxPol1"/>
  </fvCtx>
  <bgpCtxAfPol name="bgpCtxPol1" maxEcmp="64" maxEcmpIbgp="64"/>
</fvTenant>
```

REST API を使用した AS パス プリペンドの設定

次の例では、REST API を使用した AS パス プリペンド機能を設定する方法の情報を提供します。

```
<?xml version="1.0" encoding="UTF-8"?>
<fvTenant name="coke">
  <rtctrlAttrP name="attrp1">
    <rtctrlSetASPath criteria="prepend">
      <rtctrlSetASPathASN asn="100" order="1"/>
      <rtctrlSetASPathASN asn="200" order="10"/>
      <rtctrlSetASPathASN asn="300" order="5"/>
    </rtctrlSetASPath>
    <rtctrlSetASPath criteria="prepend-last-as" lastnum="9" />
  </rtctrlAttrP>

  <l3extOut name="out1">
    <rtctrlProfile name="rp1">
      <rtctrlCtxP name="ctxp1" order="1">
        <rtctrlScope>
          <rtctrlRsScopeToAttrP tnRtctrlAttrPName="attrp1"/>
        </rtctrlScope>
      </rtctrlCtxP>
    </rtctrlProfile>
  </l3extOut>
</fvTenant>
```

REST API を使用した自律システム オーバーライド対応のネットワークのルーティング BGP 外部の設定

手順の概要

1. 自律型オーバーライドを有効にして、BGP 外部ルーテッド ネットワークを設定します。

手順の詳細

手順

自律型オーバーライドを有効にして、BGP 外部ルーテッド ネットワークを設定します。

(注)

太字で表示されているコードの行に設定の BGP AS オーバーライド部分が表示されます。この機能は Cisco APIC リリース 3.1(2m) で導入されました。

例 :

```
<fvTenant name="coke">

  <fvCtx name="coke" status="">
    <bgpRtTargetP af="ipv4-ucast">
      <bgpRtTarget type="import" rt="route-target:as4-nn2:1234:1300" />
      <bgpRtTarget type="export" rt="route-target:as4-nn2:1234:1300" />
    </bgpRtTargetP>
    <bgpRtTargetP af="ipv6-ucast">
      <bgpRtTarget type="import" rt="route-target:as4-nn2:1234:1300" />
      <bgpRtTarget type="export" rt="route-target:as4-nn2:1234:1300" />
    </bgpRtTargetP>
  </fvCtx>

  <fvBD name="cokeBD">
    <!-- Association from Bridge Doamin to Private Network -->
    <fvRsCtx tnFvCtxName="coke" />
    <fvRsBDToOut tnL3extOutName="routAccounting" />
    <!-- Subnet behind the bridge domain-->
    <fvSubnet ip="20.1.1.1/16" scope="public"/>
    <fvSubnet ip="2000:1::1/64" scope="public"/>
  </fvBD>

  <fvBD name="cokeBD2">
    <!-- Association from Bridge Doamin to Private Network -->
    <fvRsCtx tnFvCtxName="coke" />
    <fvRsBDToOut tnL3extOutName="routAccounting" />
    <!-- Subnet behind the bridge domain-->
    <fvSubnet ip="30.1.1.1/16" scope="public"/>
  </fvBD>

  <vzBrCP name="webCtrct" scope="global">
    <vzSubj name="http">
      <vzRsSubjFiltAtt tnVzFilterName="default"/>
    </vzSubj>
  </vzBrCP>

  <!-- GOLF L3Out -->
  <l3extOut name="routAccounting">
    <l3extConsLbl name="golf_transit" owner="infra" status="" />
    <bgpExtP/>
    <l3extInstP name="accountingInst">
      <!--
      <l3extSubnet ip="192.2.2.0/24" scope="import-security,import-rtctrl" />
      <l3extSubnet ip="192.3.2.0/24" scope="export-rtctrl"/>
      <l3extSubnet ip="192.5.2.0/24" scope="export-rtctrl"/>
      <l3extSubnet ip="64:ff9b::c007:200/120" scope="export-rtctrl" />
    </l3extInstP>
  </l3extOut>
</fvTenant>
```

```

-->
<l3extSubnet ip="0.0.0.0/0"
scope="export-rtctrl,import-security"                                aggregate="export-rtctrl"
/>
<fvRsProv tnVzBrCPName="webCtrct"/>
</l3extInstP>

<l3extRsEctx tnFvCtxName="coke"/>
</l3extOut>

<fvAp name="cokeAp">
  <fvAEPg name="cokeEPg" >
    <fvRsBd tnFvBDName="cokeBD" />
    <fvRsPathAtt tDn="topology/pod-1/paths-103/pathep-[eth1/20]" encap="vlan-100"
instrImedcy="immediate" mode="regular"/>
    <fvRsCons tnVzBrCPName="webCtrct"/>
  </fvAEPg>
  <fvAEPg name="cokeEPg2" >
    <fvRsBd tnFvBDName="cokeBD2" />
    <fvRsPathAtt tDn="topology/pod-1/paths-103/pathep-[eth1/20]" encap="vlan-110"
instrImedcy="immediate" mode="regular"/>
    <fvRsCons tnVzBrCPName="webCtrct"/>
  </fvAEPg>
</fvAp>

<!-- Non GOLF L3Out-->
<l3extOut name="NonGolfOut">
  <bgpExtP/>
  <l3extLNodeP name="bLeaf">
    <!--
    <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="20.1.13.1"/>
    -->
    <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="20.1.13.1">
    <l3extLoopBackIfP addr="1.1.1.1"/>

    <ipRouteP ip="2.2.2.2/32" >
      <ipNextHopP nhAddr="20.1.12.3"/>
    </ipRouteP>

  </l3extRsNodeL3OutAtt>
  <l3extLIfP name='portIfV4'>
    <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/17]" encap='vlan-1010'
ifInstT='sub-interface' addr="20.1.12.2/24">

    </l3extRsPathL3OutAtt>
  </l3extLIfP>
  <l3extLIfP name='portIfV6'>
    <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/17]" encap='vlan-1010'
ifInstT='sub-interface' addr="64:ff9b::1401:302/120">
      <bgpPeerP addr="64:ff9b::1401:d03" ctrl="send-com,send-ext-com" />
    </l3extRsPathL3OutAtt>
  </l3extLIfP>
  <bgpPeerP addr="2.2.2.2" ctrl="as-override,disable-peer-as-check, send-com,send-ext-com"
status=""/>
</l3extLNodeP>
<!--
  <bgpPeerP addr="2.2.2.2" ctrl="send-com,send-ext-com" status=""/>
-->
<l3extInstP name="accountingInst">
  <l3extSubnet ip="192.10.0.0/16" scope="import-security,import-rtctrl" />
  <l3extSubnet ip="192.3.3.0/24" scope="import-security,import-rtctrl" />
  <l3extSubnet ip="192.4.2.0/24" scope="import-security,import-rtctrl" />
  <l3extSubnet ip="64:ff9b::c007:200/120" scope="import-security,import-rtctrl" />

```

REST API を使用した BGP ネイバー シャットダウンおよびソフト リセットの設定

```

        <l3extSubnet ip="192.2.2.0/24" scope="export-rtctrl" />
        <l3extSubnet ip="0.0.0.0/0"
scope="export-rtctrl,import-rtctrl,import-security"
aggregate="export-rtctrl,import-rtctrl" />
        </l3extInstP>
        <l3extRsEctx tnFvCtxName="coke"/>
    </l3extOut>

</fvTenant>

```

REST API を使用した BGP ネイバー シャットダウンおよびソフト リセットの設定

REST API を使用した BGP ネイバー シャットダウンの構成

次の手順では、REST API を使用して BGP ネイバー シャットダウン機能を使用する方法について説明します。

手順

ステップ 1 ノードおよびインターフェイスを構成します。

この例は、次の VRF `v1` をノード 103（ボーダー リーフ スイッチ）に、ノード プロファイル、`nodep1`、およびルータ ID `11.11.11.103` 付きで構成します。また、インターフェイス `eth1/3` をルーテッドインターフェイス（レイヤ 3 ポート）として IP アドレス `12.12.12.1/24` およびレイヤ 3 ドメイン `dom1` 付きで構成します。

例：

```

<l3extOut name="l3out1">
  <l3extRsEctx tnFvCtxName="v1"/>
  <l3extLNodeP name="nodep1">
    <l3extRsNodeL3OutAtt rtrId="11.11.11.103" tDn="topology/pod-1/node-103"/>
    <l3extLIIfP name="ifp1"/>
    <l3extRsPathL3OutAtt addr="12.12.12.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-103/patchep-[eth1/3]"/>
  </l3extLIIfP>
</l3extLNodeP>
  <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
</l3extOut>

```

ステップ 2 BGP ルーティング プロトコルを構成し、BGP ネイバー シャットダウン機能を構成します。

この例では、IP アドレス、`15.15.15.2`、ASN 100 を持つ BGP ピアで、プライマリ ルーティング プロトコルとして BGP を構成します。

adminSt 変数は次のいずれかに設定できます：

- **[有効 (enabled)]**：BGP ネイバー シャットダウン機能を有効にします。
- **[無効 (disabled)]**：BGP ネイバー シャットダウン機能を無効にします。

次の例では、BGP ネイバー シャットダウン機能が有効になっています。

例 :

```
<l3extOut name="l3out1">
  <l3extLNodeP name="nodep1">
    <bgpPeerP addr="15.15.15.2"> adminSt="enabled"
      <bgpAsP asn="100"/>
    </bgpPeerP>
  </l3extLNodeP>
</l3extOut>
```

REST API を使用した BGP ネイバー ソフト リセットの設定

次の手順では、REST API を使用して BGP ネイバー ソフト リセット機能を使用する方法について説明します。

手順

ステップ1 ノードおよびインターフェイスを設定します。

この例は、ノード 103（ボーダー リーフ スイッチ）上で次の VRF v1 を構成します。これは、ノードプロファイルの nodep1 および ルータ ID 11.11.11.103 付きです。ルーテッドインターフェイス（レイヤー 3 ポート）としてインターフェイス Eth1/3 を IP アドレス 12.12.12.1/24 および レイヤ 3 ドメイン dom1 を構成します。

例 :

```
<l3extOut name="l3out1">
  <l3extRsEctx tnFvCtxName="v1"/>
  <l3extLNodeP name="nodep1">
    <l3extRsNodeL3OutAtt rtrId="11.11.11.103" tDn="topology/pod-1/node-103"/>
    <l3extLIfP name="ifp1"/>
    <l3extRsPathL3OutAtt addr="12.12.12.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-103/pathep-[eth1/3]"/>
  </l3extLIfP>
</l3extLNodeP>
  <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
</l3extOut>
```

ステップ2 BGP ルーティング プロトコルを構成し、BGP ネイバー ソフト リセット機能を構成します。

この例では、IP アドレス、15.15.15.2、および ASN 100 を持つ BGP ピアで、プライマリ ルーティング プロトコルとして BGP を構成します。

dir 変数は、次のいずれかに設定できます :

- **in** : ソフト ダイナミック インバウンド リセットを有効にします。
- **out** : ソフト アウトバウンド リセットを有効にします。

次の例では、ソフト ダイナミック インバウンド リセットが有効になっています。

例 :

REST API を使用した VRF ごと、ノード BGP ごとのタイマーの設定

```

<l3extOut name="l3out1">
  <l3extLNodeP name="nodep1">
    <bgpPeerP addr="15.15.15.2">
      <bgpAsP asn="100"/>
      <bgpPeerEntryClearPeerLTask>
        <attributes>
          <mode>soft</mode>
          <dir>in</dir>
          <adminSt>start</adminSt>
        </attributes>
      </children/>
    </bgpPeerEntryClearPeerLTask>
  </bgpPeerP>
</l3extLNodeP>
<bgpExtP/>
</l3extOut>

```

REST API を使用した VRF ごと、ノード BGP ごとのタイマーの設定

次の例では、ノード内の VRF ごと、ノード BGP ごとのタイマーの設定方法を示します。bgpProtP を構成します。これは、l3extLNodeP 構成の下にあります。bgpProtP の下には、関係 (bgpRsBgpNodeCtxPol) を目的の BGP コンテキスト ポリシー (bgpCtxPol) に構成します。

手順

ノード固有の BGP タイマー ポリシーを node1 で構成します。そして node2 は、ノード固有ではない BGP タイマー ポリシーと構成します。

例：

POST https://apic-ip-address/mo.xml

```

<fvTenant name="tn1" >
  <bgpCtxPol name="pol1" staleIntvl="25" />
  <bgpCtxPol name="pol2" staleIntvl="35" />
  <fvCtx name="ctx1" >
    <fvRsBgpCtxPol tnBgpCtxPolName="pol1"/>
  </fvCtx>
  <l3extout name="out1" >
    <l3extRsEctx toFvCtxName="ctx1" />
    <l3extLNodeP name="node1" >
      <bgpProtP name="protpl" >
        <bgpRsBgpNodeCtxPol tnBgpCtxPolName="pol2" />
      </bgpProtP>
    </l3extLNodeP>
    <l3extLNodeP name="node2" >
    </l3extLNodeP>

```

この例では、node1 は、pol2 ポリシーから BGP タイマー値を取得します。そして、node2 は、BGP タイマー値を pol1 から取得します。タイマー値は、bgpDom に適用されます。この bgpDom は、VRF tn1 : ctx1 に対応するものです。これは、「VRF ごと、ノード BGP ごとのタイマーの値」のセクションで説明したアルゴリズムに従って選択された、BGP タイマー ポリシーに基づきます。

削除するノード BGP タイマーが REST API を使用してごとの VRF あたり

次の例では、ノード内で既存の VRF ごとの各ノード BGP タイマーを削除する方法を示します。

手順

node1 で特定の BGP タイマー ポリシーのノードを削除します。

例：

POST https://apic-ip-address/mo.xml

```
<fvTenant name="tn1" >
  <bgpCtxPol name="pol1" staleIntvl="25" />
  <bgpCtxPol name="pol2" staleIntvl="35" />
  <fvCtx name="ctx1" >
    <fvRsBgpCtxPol tnBgpCtxPolName="pol1"/>
  </fvCtx>
  <l3extout name="out1" >
    <l3extRsEctx toFvCtxName="ctx1" />
    <l3extLNodeP name="node1" >
      <bgpProtP name="protpl" status="deleted" >
        <bgpRsBgpNodeCtxPol tnBgpCtxPolName="pol2" />
      </bgpProtP>
    </l3extLNodeP>
    <l3extLNodeP name="node2" >
    </l3extLNodeP>
```

上記の例のコードフレーズ `<bgpProtP name="protpl" status="deleted" >` では、BGP タイマー ポリシーを削除します。削除後、node1 は、node1 が関連付けられている VRF の BGP タイマー ポリシーにデフォルトします。これは、上記の例では、pol1 です。

REST API を使用したセカンダリ IP アドレスでの双方向フォワーディング検出の構成

次の例では、REST API を使用して、セカンダリ IP アドレスに Bidirectional Forwarding Detection (BFD) を構成します。

```
<l3extLIIfP dn="uni/tn-sec-ip-bfd/out-secip-bfd-l3out/lnodep-secip-bfd-l3out_nodeProfile/
  lifp-secip-bfd-l3out_interfaceProfile" name="secip-bfd-l3out_interfaceProfile"
prio="unspecified" tag="yellow-green" userdom=":all:">
  <l3extRsPathL3OutAtt addr="50.50.50.200/24" autostate="disabled" encap="vlan-2"
  encapScope="local" ifInstT="ext-svi" ipv6Dad="enabled" isMultiPodDirect="no"
  llAddr="::" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
  tDn="topology/pod-1/paths-101/pathep-[eth1/3]" targetDscp="unspecified"
  userdom=":all:">
    <l3extIp addr="9.9.9.1/24" ipv6Dad="enabled" userdom=":all:"/>
    <l3extIp addr="6.6.6.1/24" ipv6Dad="enabled" userdom=":all:"/>
  </l3extRsPathL3OutAtt>
  <l3extRsNdIfPol userdom="all"/>
  <l3extRsLIIfPCustQosPol userdom="all"/>
  <l3extRsIngressQosDppPol userdom="all"/>
  <l3extRsEgressQosDppPol userdom="all"/>
  <l3extRsArpIfPol userdom="all"/>
</l3extLIIfP>
<ipRouteP aggregate="no"
```

```

dn="uni/tn-sec-ip-bfd/out-secip-bfd-l3out/lnodep-secip-bfd-l3out_nodeProfile/
rsnodeL3OutAtt-[topology/pod-1/node-101]/rt-[6.0.0.1/24]" fromPfxLen="0" ip="6.0.0.1/24"
pref="1" rtCtrl="bfd" toPfxLen="0" userdom=":all:">
  <ipNextHopP nhAddr="6.6.6.2" pref="unspecified" type="prefix" userdom=":all:"/>
</ipRouteP>

```

グローバル REST API を使用して BFD の設定

手順

次の REST API は、(BFD) を双方向フォワーディング検出のグローバル コンフィギュレーションを示します。

例：

```

<polUni>
  <infraInfra>
    <bfdIpv4InstPol name="default" echoSrcAddr="1.2.3.4" slowIntvl="1000" minTxIntvl="150"
minRxIntvl="250" detectMult="5" echoRxIntvl="200"/>
    <bfdIpv6InstPol name="default" echoSrcAddr="34::1/64" slowIntvl="1000" minTxIntvl="150"
minRxIntvl="250" detectMult="5" echoRxIntvl="200"/>
  </infraInfra>
</polUni>

```

REST API を使用した BFD インターフェイスのオーバーライドの設定

手順

次の REST API は、(BFD) を双方向フォワーディング検出のインターフェイスのオーバーライド コンフィギュレーションを示します。

例：

```

<fvTenant name="ExampleCorp">
  <bfdIfPol name="bfdIfPol" minTxIntvl="400" minRxIntvl="400" detectMult="5" echoRxIntvl="400"
echoAdminSt="disabled"/>
  <l3extOut name="l3-out">
    <l3extLNodeP name="leaf1">
      <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="2.2.2.2"/>
      <l3extLIIfP name='portIpv4'>
        <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/11]" ifInstT='l3-port'
addr="10.0.0.1/24" mtu="1500"/>
        <bfdIfP type="sha1" key="password">
          <bfdRsIfPol tnBfdIfPolName='bfdIfPol' />
        </bfdIfP>
      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>

```

REST API を使用した BFD コンシューマ プロトコルの設定

手順

ステップ 1 次の例では、双方向の転送検出（BFD）のインターフェイス設定を示します。

例：

```
<fvTenant name="ExampleCorp">
  <bfdIfPol name="bfdIfPol" minTxIntvl="400" minRxIntvl="400" detectMult="5" echoRxIntvl="400"
  echoAdminSt="disabled"/>
  <l3extOut name="l3-out">
    <l3extLNodeP name="leaf1">
      <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="2.2.2.2"/>
      <l3extLIIfP name='portIpv4'>
        <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/11]" ifInstT='l3-port'
        addr="10.0.0.1/24" mtu="1500"/>
        <bfdIfP type="sha1" key="password">
          <bfdRsIfPol tnBfdIfPolName='bfdIfPol'/>
        </bfdIfP>
      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>
```

ステップ 2 次の例では、OSPF および EIGRP で BFD を有効にするためのインターフェイス設定を示します。

例：

リーフ スイッチ上の BFD

```
<fvTenant name="ExampleCorp">
  <ospfIfPol name="ospf_intf_pol" cost="10" ctrl="bfd"/>
  <eigrpIfPol ctrl="nh-self,split-horizon,bfd" dn="uni/tn-Coke/eigrpIfPol-eigrp_if_default" ></fvTenant>
```

例：

スパイン スイッチ上の BFD

```
<l3extLNodeP name="bSpine">

  <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-103" rtrId="192.3.1.8">
    <l3extLoopBackIfP addr="10.10.3.1" />
    <l3extInfraNodeP fabricExtCtrlPeering="false" />
  </l3extRsNodeL3OutAtt>

  <l3extLIIfP name='portIf'>
    <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-103/pathep-[eth5/10]" encap='vlan-4'
    ifInstT='sub-interface' addr="20.3.10.1/24"/>
    <ospfIfP>
      <ospfRsIfPol tnOspfIfPolName='ospf_intf_pol'/>
    </ospfIfP>
    <bfdIfP name="test" type="sha1" key="hello" status="created,modified">
      <bfdRsIfPol tnBfdIfPolName='default' status="created,modified"/>
    </bfdIfP>
  </l3extLIIfP>

</l3extLNodeP>
```

ステップ 3 次の例では、BGP 上の BFD を有効にするためのインターフェイス設定を示します。

例：

```
<fvTenant name="ExampleCorp">
  <l3extOut name="l3-out">
    <l3extLNodeP name="leaf1">
      <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="2.2.2.2"/>
      <l3extLIIfP name='portIpv4'>
        <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/11]" ifInstT='l3-port'
        addr="10.0.0.1/24" mtu="1500">
          <bgpPeerP addr="4.4.4.4/24" allowedSelfAsCnt="3" ctrl="bfd" descr="" name=""
          peerCtrl="" ttl="1">
            <bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
            <bgpAsP asn="3" descr="" name=""/>
          </bgpPeerP>
        </l3extRsPathL3OutAtt>
      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>
```

ステップ 4 次の例では、スタティック ルートで BFD を有効にするためのインターフェイス設定を示します。

例：

リーフ スイッチ上の BFD

```
<fvTenant name="ExampleCorp">
  <l3extOut name="l3-out">
    <l3extLNodeP name="leaf1">
      <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="2.2.2.2">
        <ipRouteP ip="192.168.3.4" rtCtrl="bfd">
          <ipNexthopP nhAddr="192.168.62.2"/>
        </ipRouteP>
      </l3extRsNodeL3OutAtt>
      <l3extLIIfP name='portIpv4'>
        <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/3]" ifInstT='l3-port'
        addr="10.10.10.2/24" mtu="1500" status="created,modified" />
      </l3extLIIfP>
    </l3extLNodeP>
  </l3extOut>
</fvTenant>
```

例：

スパイン スイッチ上の BFD

```
<l3extLNodeP name="bSpine">
  <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-103" rtrId="192.3.1.8">
    <ipRouteP ip="0.0.0.0" rtCtrl="bfd">
      <ipNexthopP nhAddr="192.168.62.2"/>
    </ipRouteP>
  </l3extRsNodeL3OutAtt>
  <l3extLIIfP name='portIf'>
    <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-103/pathep-[eth5/10]" encap='vlan-4'
    ifInstT='sub-interface' addr="20.3.10.1/24"/>
  </l3extLIIfP>
</l3extLNodeP>
```

```

        <bfdIfP name="test" type="sha1" key="hello" status="created,modified">
          <bfdRsIfPol tnBfdIfPolName='default' status="created,modified"/>
        </bfdIfP>
      </l3extLIfP>

    </l3extLNodeP>

```

ステップ 5 次の例では、IS-IS で BFD を有効にするためのインターフェイス設定を示します。

例：

```

<fabricInst>
  <l3IfPol name="testL3IfPol" bfdIsis="enabled"/>
  <fabricLeafP name="LeNode" >
    <fabricRsLePortP tDn="uni/fabric/leportp-leaf_profile" />
    <fabricLeafS name="spsw" type="range">
    <fabricNodeBlk name="node101" to_="102" from_="101" />
    </fabricLeafS>
  </fabricLeafP>

  <fabricSpineP name="SpNode" >
    <fabricRsSpPortP tDn="uni/fabric/spportp-spine_profile" />
    <fabricSpineS name="spsw" type="range">
      <fabricNodeBlk name="node103" to_="103" from_="103" />
    </fabricSpineS>
  </fabricSpineP>

  <fabricLePortP name="leaf_profile">
    <fabricLFPortS name="leafIf" type="range">
    <fabricPortBlk name="spBlk" fromCard="1" fromPort="49" toCard="1" toPort="49" />
      <fabricRsLePortPGrp tDn="uni/fabric/funcprof/leportgrp-LeTestPGrp" />
    </fabricLFPortS>
  </fabricLePortP>

  <fabricSpPortP name="spine_profile">
    <fabricSFPortS name="spineIf" type="range">
      <fabricPortBlk name="spBlk" fromCard="5" fromPort="1" toCard="5" toPort="2" />
      <fabricRsSpPortPGrp tDn="uni/fabric/funcprof/spportgrp-SpTestPGrp" />
    </fabricSFPortS>
  </fabricSpPortP>

  <fabricFuncP>
    <fabricLePortPGrp name = "LeTestPGrp">
    <fabricRsL3IfPol tnL3IfPolName="testL3IfPol"/>
    </fabricLePortPGrp>

    <fabricSpPortPGrp name = "SpTestPGrp">
    <fabricRsL3IfPol tnL3IfPolName="testL3IfPol"/>
    </fabricSpPortPGrp>

  </fabricFuncP>
</fabricInst>

```

REST API を使用した OSPF 外部ルーテッド ネットワークの設定

REST API を使用した管理テナントの OSPF 外部ルーテッド ネットワークの作成

- ルータ ID と論理インターフェイス プロファイルの IP アドレスが異なっていて重複していないことを確認します。
- 次の手順は、管理テナントの OSPF 外部ルーテッド ネットワークを作成するためのものです。テナントの OSPF 外部ルーテッド ネットワークを作成するには、テナントを選択し、テナント用の VRF を作成する必要があります。
- 詳細を参照してください。『Cisco APIC and Transit Routing』を参照してください。

手順

管理テナントの OSPF 外部ルーテッド ネットワークを作成します。

例：

POST: <https://apic-ip-address/api/mo/uni/tn-mgmt.xml>

```
<fvTenant name="mgmt">
  <fvBD name="bd1">
    <fvRsBDToOut tnL3extOutName="RtdOut" />
    <fvSubnet ip="1.1.1.1/16" />
    <fvSubnet ip="1.2.1.1/16" />
    <fvSubnet ip="40.1.1.1/24" scope="public" />
    <fvRsCtx tnFvCtxName="inb" />
  </fvBD>
  <fvCtx name="inb" />

  <l3extOut name="RtdOut">
    <l3extRsL3DomAtt tDn="uni/l3dom-extdom"/>
    <l3extInstP name="extMgmt">
      </l3extInstP>
      <l3extLNodeP name="borderLeaf">
        <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="10.10.10.10"/>
        <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-102" rtrId="10.10.10.11"/>
        <l3extLIIfP name='portProfile'>
          <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]" ifInstT='l3-port'
            addr="192.168.62.1/24"/>
          <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-102/pathep-[eth1/40]" ifInstT='l3-port'
            addr="192.168.62.5/24"/>
          <ospfIfP/>
        </l3extLIIfP>
      </l3extLNodeP>
      <l3extRsEctx tnFvCtxName="inb"/>
      <ospfExtP areaId="57" />
    </l3extOut>
  </fvTenant>
```

REST API を使用した EIGRP 外部ルーテッドネットワークの設定

REST API を使用した EIGRP の設定

手順

ステップ 1 EIGRP コンテキスト ポリシーを設定します。

例 :

```
<polUni>
  <fvTenant name="cisco_6">
    <eigrpCtxAfPol actIntvl="3" descr="" dn="uni/tn-cisco_6/eigrpCtxAfPol-eigrp_default_pol"
    extDist="170" intDist="90" maxPaths="8" metricStyle="narrow" name="eigrp_default_pol"
    ownerKey="" ownerTag="" />
  </fvTenant>
</polUni>
```

ステップ 2 EIGRP インターフェイス ポリシーを設定します。

例 :

```
<polUni>
  <fvTenant name="cisco_6">
    <eigrpIfPol bw="10" ctrl="nh-self,split-horizon" delay="10" delayUnit="tens-of-micro" descr=""
    dn="uni/tn-cisco_6/eigrpIfPol-eigrp_if_default" helloIntvl="5" holdIntvl="15"
    name="eigrp_if_default" ownerKey="" ownerTag="" />
  </fvTenant>
</polUni>
```

ステップ 3 EIGRP VRF を設定します。

例 :

IPv4 :

```
<polUni>
  <fvTenant name="cisco_6">
    <fvCtx name="dev">
      <fvRsCtxToEigrpCtxAfPol tnEigrpCtxAfPolName="eigrp_ctx_pol_v4" af="1" />
    </fvCtx>
  </fvTenant>
</polUni>
```

IPv6

```
<polUni>
  <fvTenant name="cisco_6">
    <fvCtx name="dev">
      <fvRsCtxToEigrpCtxAfPol tnEigrpCtxAfPolName="eigrp_ctx_pol_v6" af="ipv6-ucast" />
    </fvCtx>
  </fvTenant>
</polUni>
```

ステップ 4 外部の EIGRP Layer3 を設定します。

例 :

IPv4

```
<polUni>
  <fvTenant name="cisco_6">
```

```

    <l3extOut name="ext">
      <eigrpExtP asn="4001"/>
      <l3extLNodeP name="node1">
        <l3extLIIfP name="intf_v4">
          <l3extRsPathL3OutAtt addr="201.1.1.1/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/4]"/>
          <eigrpIfP name="eigrp_ifp_v4">
            <eigrpRsIfPol tnEigrpIfPolName="eigrp_if_pol_v4"/>
          </eigrpIfP>
        </l3extLIIfP>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>

```

IPv6

```

<polUni>
  <fvTenant name="cisco_6">
    <l3extOut name="ext">
      <eigrpExtP asn="4001"/>
      <l3extLNodeP name="node1">
        <l3extLIIfP name="intf_v6">
          <l3extRsPathL3OutAtt addr="2001::1/64" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/4]"/>
          <eigrpIfP name="eigrp_ifp_v6">
            <eigrpRsIfPol tnEigrpIfPolName="eigrp_if_pol_v6"/>
          </eigrpIfP>
        </l3extLIIfP>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>

```

IPv4 および IPv6

```

<polUni>
  <fvTenant name="cisco_6">
    <l3extOut name="ext">
      <eigrpExtP asn="4001"/>
      <l3extLNodeP name="node1">
        <l3extLIIfP name="intf_v4">
          <l3extRsPathL3OutAtt addr="201.1.1.1/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/4]"/>
          <eigrpIfP name="eigrp_ifp_v4">
            <eigrpRsIfPol tnEigrpIfPolName="eigrp_if_pol_v4"/>
          </eigrpIfP>
        </l3extLIIfP>

        <l3extLIIfP name="intf_v6">
          <l3extRsPathL3OutAtt addr="2001::1/64" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/4]"/>
          <eigrpIfP name="eigrp_ifp_v6">
            <eigrpRsIfPol tnEigrpIfPolName="eigrp_if_pol_v6"/>
          </eigrpIfP>
        </l3extLIIfP>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>

```

ステップ 5 (任意) インターフェイス ポリシー ノブを設定します。

例：


```
<polUni>
  <fvTenant name="cisco_6">
    <eigrpIfPol bw="1000000" ctrl="nh-self,split-horizon" delay="10"
delayUnit="tens-of-micro" helloIntvl="5" holdIntvl="15" name="default"/>
  </fvTenant>
</polUni>
```

[帯域幅 (delayUnit)] (bw) 属性は kbps で定義されています。 [delayUnit] 属性は、「1 万マイクロ」または「ピコ」です。

REST API を使用したルート集約の設定

BGP、OSPF、および REST API を使用して EIGRP のルート集約の設定

手順

ステップ 1 次のように、REST API を使用して BGP ルート集約を設定します。

例：

```
<fvTenant name="common">
  <fvCtx name="vrf1"/>
  <bgpRtSummPol name="bgp_rt_summ" ctrl='as-set' />
  <l3extOut name="l3_ext_pol" >
    <l3extLNodeP name="bLeaf">
      <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="20.10.1.1"/>
      <l3extLIfP name='portIf'>
        <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/31]" ifInstT='l3-port'
addr="10.20.1.3/24"/>
      </l3extLIfP>
    </l3extLNodeP>
  <bgpExtP />
  <l3extInstP name="InstP" >
    <l3extSubnet ip="10.0.0.0/8" scope="export-rtctrl">
      <l3extRsSubnetToRtSumm tDn="uni/tn-common/bgpsum-bgp_rt_summ"/>
      <l3extRsSubnetToProfile tnRtctrlProfileName="rtprof"/>
    </l3extSubnet>
  </l3extInstP>
  <l3extRsEctx tnFvCtxName="vrf1"/>
</l3extOut>
</fvTenant>
```

ステップ 2 次の REST API を使用して、OSPF のエリア間および外部の集約を設定します。

例：

```
<?xml version="1.0" encoding="utf-8"?>
<fvTenant name="t20">
  <!--Ospf Inter External route summarization Policy-->
  <ospfRtSummPol cost="unspecified" interAreaEnabled="no" name="ospfext"/>
  <!--Ospf Inter Area route summarization Policy-->
  <ospfRtSummPol cost="16777215" interAreaEnabled="yes" name="interArea"/>
</fvTenant>
```

```

<fvCtx name="ctx0" pcEnfDir="ingress" pcEnfPref="enforced"/>
<!-- L3OUT backbone Area-->
<l3extOut enforceRtctrl="export" name="l3_1" ownerKey="" ownerTag="" targetDscp="unspecified">
  <l3extRsEctx tnFvCtxName="ctx0"/>
  <l3extLNodeP name="node-101">
    <l3extRsNodeL3OutAtt rtrId="20.1.3.2" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"/>
    <l3extLIIfP name="intf-1">
      <l3extRsPathL3OutAtt addr="20.1.5.2/24" encap="vlan-1001" ifInstT="sub-interface"
tDn="topology/pod-1/paths-101/pathep-[eth1/33]"/>
    </l3extLIIfP>
  </l3extLNodeP>
  <l3extInstP name="l3InstP1">
    <fvRsProv tnVzBrCPName="default"/>
    <!--Ospf External Area route summarization-->
    <l3extSubnet aggregate="" ip="193.0.0.0/8" name="" scope="export-rtctrl">
      <l3extRsSubnetToRtSumm tDn="uni/tn-t20/ospfrtsumm-ospfext"/>
    </l3extSubnet>
  </l3extInstP>
  <ospfExtP areaCost="1" areaCtrl="redistribute,summary" areaId="backbone" areaType="regular"/>
</l3extOut>
<!-- L3OUT Regular Area-->
<l3extOut enforceRtctrl="export" name="l3_2">
  <l3extRsEctx tnFvCtxName="ctx0"/>
  <l3extLNodeP name="node-101">
    <l3extRsNodeL3OutAtt rtrId="20.1.3.2" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"/>
    <l3extLIIfP name="intf-2">
      <l3extRsPathL3OutAtt addr="20.1.2.2/24" encap="vlan-1014" ifInstT="sub-interface"
tDn="topology/pod-1/paths-101/pathep-[eth1/11]"/>
    </l3extLIIfP>
  </l3extLNodeP>
  <l3extInstP matchT="AtleastOne" name="l3InstP2">
    <fvRsCons tnVzBrCPName="default"/>
    <!--Ospf Inter Area route summarization-->
    <l3extSubnet aggregate="" ip="197.0.0.0/8" name="" scope="export-rtctrl">
      <l3extRsSubnetToRtSumm tDn="uni/tn-t20/ospfrtsumm-interArea"/>
    </l3extSubnet>
  </l3extInstP>
  <ospfExtP areaCost="1" areaCtrl="redistribute,summary" areaId="0.0.0.57" areaType="regular"/>
</l3extOut>
</fvTenant>

```

ステップ3 次の REST API を使用して EIGRP の集約を設定します。

例：

```

<fvTenant name="exampleCorp">
  <l3extOut name="out1">
    <l3extInstP name="eigrpSummInstp" >
      <l3extSubnet aggregate="" descr="" ip="197.0.0.0/8" name="" scope="export-rtctrl">
        <l3extRsSubnetToRtSumm/>
      </l3extSubnet>
    </l3extInstP>
  </l3extOut>
  <eigrpRtSummPol name="pol1" />

```

(注)

EIGRP を設定するルート集約ポリシーはありません。EIGRP の集約を有効にするために必要なだけの設定では、サマリー サブネット、InstP です。

REST API を使用したルート マップおよびルート プロファイルによるルート制御の設定

REST API を使用した BGP ピアごとのルート制御の構成

次の手順では、REST API を使用して BGP ピア単位のルート制御を構成する方法について説明します。

手順

BGP ピアごとのルート制御機能を構成します。

場所：

- **direction = 「import」** は、ルートインポートポリシー（ファブリックに許可されるルート）です。
- **direction = 「export」** は、ルートエクスポートポリシー（外部ネットワークからアドバタイズされるルート）です。

例：

```
<polUni>
  <fvTenant name="t1">
    <fvCtx name="v1"/>
    <l3extOut name="l3out1">
      <l3extRsEctx tnFvCtxName="v1"/>
      <l3extLNodeP name="nodep1">
        <l3extRsNodeL3OutAtt rtrId="11.11.11.103" tDn="topology/pod-1/node-103"/>
        <l3extLIIfP name="ifp1">
          <l3extRsPathL3OutAtt addr="12.12.12.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-103/pathep-[eth1/3]"/>
        </l3extLIIfP>
        <bgpPeerP addr="15.15.15.2">
          <bgpAsP asn="100"/>
          <bgpRsPeerToProfile direction="export" tnRtctrlProfileName="rp1"/>
        </bgpPeerP>
      </l3extLNodeP>
      <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
      <bgpExtP/>
      <ospfExtP areaId="0.0.0.0" areaType="regular"/>
      <l3extInstP name="extnw1" >
        <l3extSubnet ip="20.20.20.0/24" scope="import-security"/>
      </l3extInstP>
    </l3extOut>
    <rtctrlProfile name="rp1">
      <rtctrlCtxP name="ctxp1" action="permit" order="0">
        <rtctrlScope>
          <rtctrlRsScopeToAttrP tnRtctrlAttrPName="attrp1"/>
        </rtctrlScope>
        <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule1"/>
      </rtctrlCtxP>
    </rtctrlProfile>
    <rtctrlSubjP name="match-rule1">
      <rtctrlMatchRtDest ip="200.3.2.0/24"/>
    </rtctrlSubjP>
  </fvTenant>
</polUni>
```

```

    <rtctrlAttrP name="attrp1">
      <rtctrlSetASPath criteria="prepend">
        <rtctrlSetASPathASN asn="100" order="2"/>
        <rtctrlSetASPathASN asn="200" order="1"/>
      </rtctrlSetASPath>
    </rtctrlAttrP>
  </fvTenant>
</polUni>

```

REST API を使用して、明示的なプレフィックス リストでルート マップ/プロファイルの構成

始める前に

- テナントと VRF を構成する必要があります。

手順

明示的なプレフィックス リストを使用してルート マップ/プロファイルを構成します。

(注)

以下の **太字** のエントリは、APIC リリース 4.2 (3) 以降で使用可能な一致プレフィックスの拡張機能です。これらのフィールドの詳細については、[一致プレフィックスの機能拡張](#)を参照してください。

例 :

```

<?xml version="1.0" encoding="UTF-8"?>
<fvTenant name="PM" status="">
  <rtctrlAttrP name="set_dest">
    <rtctrlSetComm community="regular:as2-nn2:5:24" />
  </rtctrlAttrP>
  <rtctrlSubjP name="allow_dest">
    <rtctrlMatchRtDest ip="192.169.0.0/24" aggregate="yes" >fromPfxLen="26" toPfxLen="30" />
    <rtctrlMatchCommTerm name="term1">
      <rtctrlMatchCommFactor community="regular:as2-nn2:5:24" status="" />
      <rtctrlMatchCommFactor community="regular:as2-nn2:5:25" status="" />
    </rtctrlMatchCommTerm>
    <rtctrlMatchCommRegexTerm commType="regular" regex="200:*" status="" />
  </rtctrlSubjP>
  <rtctrlSubjP name="deny_dest">
    <rtctrlMatchRtDest ip="192.168.0.0/24" />
  </rtctrlSubjP>
</fvCtxt name="ctx" />
<l3extOut name="L3Out_1" enforceRtctrl="import,export" status="">
  <l3extRsEctx tnFvCtxtName="ctx" />
  <l3extLNodeP name="bLeaf">
    <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="1.2.3.4" />
    <l3extLIIfP name="portIf">
      <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/25]"
ifInstT="sub-interface" encap="vlan-1503" addr="10.11.12.11/24" />
      <ospfIfP />
    </l3extLIIfP>
    <bgpPeerP addr="5.16.57.18/32" ctrl="send-com" />
    <bgpPeerP addr="6.16.57.18/32" ctrl="send-com" />
  </l3extLNodeP>
</l3extOut>
</fvTenant>

```

```

</l3extLNodeP>
<bgpExtP />
<ospfExtP areaId="0.0.0.59" areaType="nssa" status="" />
<l3extInstP name="l3extInstP_1" status="">
  <l3extSubnet ip="17.11.1.11/24" scope="import-security" />
</l3extInstP>
<rtctrlProfile name="default-export" type="global" status="">
  <rtctrlCtxP name="ctx_deny" action="deny" order="1">
    <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="deny_dest" status="" />
  </rtctrlCtxP>
  <rtctrlCtxP name="ctx_allow" order="2">
    <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="allow_dest" status="" />
  </rtctrlCtxP>
  <rtctrlScope name="scope" status="">
    <rtctrlRsScopeToAttrP tnRtctrlAttrPName="set_dest" status="" />
  </rtctrlScope>
</rtctrlProfile>
</l3extOut>
<fvBD name="testBD">
  <fvRsBDToOut tnL3extOutName="L3Out_1" />
  <fvRsCtx tnFvCtxName="ctx" />
  <fvSubnet ip="40.1.1.12/24" scope="public" />
  <fvSubnet ip="40.1.1.2/24" scope="private" />
  <fvSubnet ip="2003::4/64" scope="public" />
</fvBD>
</fvTenant>

```

REST API を使用した、インポート制御とエクスポート制御によるルーティング制御プロトコルの設定

この例では、ネットワーク接続 BGP を使用して外部レイヤ 3 が設定されていることを前提としています。OSPF を使用してネットワークを次のタスクを実行することもできます。

始める前に

- テナント、プライベートネットワーク、およびブリッジドメインが作成されていること。
- レイヤ 3 Outside テナント ネットワークが設定されていること。

手順

インポート制御とエクスポート制御を使用するルート制御プロトコルを設定します。

例：

```

<l3extOut descr="" dn="uni/tn-Ten_ND/out-L3Out1" enforceRtctrl="export" name="L3Out1" ownerKey=""
ownerTag="" targetDscp="unspecified">
  <l3extLNodeP descr="" name="LNodeP1" ownerKey="" ownerTag="" tag="yellow-green"
targetDscp="unspecified">
    <l3extRsNodeL3OutAtt rtrId="1.2.3.4" rtrIdLoopBack="yes" tDn="topology/pod-1/node-101">

      <l3extLoopBackIfP addr="2000::3" descr="" name="" />
    </l3extRsNodeL3OutAtt>
  </l3extLNodeP>
</l3extOut>

```

```

<l3extLIfP descr="" name="IFP1" ownerKey="" ownerTag="" tag="yellow-green">
  <ospfIfP authKeyId="1" authType="none" descr="" name="">
    <ospfRsIfPol tnOspfIfPolName=""/>
  </ospfIfP>
  <l3extRsNdIfPol tnNdIfPolName=""/>
  <l3extRsPathL3OutAtt addr="10.11.12.10/24" descr="" encap="unknown" ifInstT="l3-port"
    llAddr="::" mac="00:22:BD:F8:19:FF" mtu="1500" tDn="topology/pod-1/paths-101/pathep-[eth1/17]"
    targetDscp="unspecified"/>
</l3extLIfP>
</l3extLNodeP>
<l3extRsEctx tnFvCtxName="PVN1"/>
<l3extInstP descr="" matchT="AtleastOne" name="InstP1" prio="unspecified"
targetDscp="unspecified">
  <fvRsCustQosPol tnQosCustomPolName=""/>
  <l3extSubnet aggregate="" descr="" ip="192.168.1.0/24" name="" scope=""/>
</l3extInstP>
<ospfExtP areaCost="1" areaCtrl="redistribute,summary" areaId="0.0.0.1" areaType="nssa"
descr=""/>
<rtctrlProfile descr="" name="default-export" ownerKey="" ownerTag="">
  <rtctrlCtxP descr="" name="routecontrolpvtnw" order="3">
    <rtctrlScope descr="" name="">
      <rtctrlRsScopeToAttrP tnRtctrlAttrPName="actionruleprofile2"/>
    </rtctrlScope>
  </rtctrlCtxP>
</rtctrlProfile>
</l3extOut>

```

REST API を使用したインターリーク再配布の構成

次の手順では、REST API を使用してインターリーク再配布を構成する方法について説明します。

始める前に

テナント、VRF、および L3Out を作成します。

手順

ステップ1 インターリーク再配布のルートマップを構成します。

例：

次の例では、次の二つのコンテキストを持つルートマップ [INTERLEAK_RP] を構成します。2つのコンテキスト ([ROUTES_A] および [ROUTES_ALL])。最初のコンテキスト [ROUTES_A] は、IP プレフィックスリスト 10.0.0.0/24 le 32 と一致し、set rule [COM_A] を介してコミュニティ属性を設定します。2番目のコンテキストは、すべてのルートと一致します。

```

POST: https://<APIC IP>/api/mo/uni.xml
BODY:
<fvTenant dn="uni/tn-SAMPLE">
  <!-- route map with two contexts (ROUTES_A and ROUTES_ALL)-->
  <rtctrlProfile type="global" name="INTERLEAK_RP">
    <rtctrlCtxP name="ROUTES_A" order="0" action="permit">
      <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="PFX_10-0-0-0_24"/>
    <rtctrlScope>

```

```

        <rtctrlRsScopeToAttrP tnRtctrlAttrPName="COM_A"/>
    </rtctrlScope>
</rtctrlCtxP>
<rtctrlCtxP name="ROUTES_ALL" order="9" action="permit">
    <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="ALL_PREFIX"/>
</rtctrlCtxP>
</rtctrlProfile>

<!-- match rule with an IP prefix-list -->
<rtctrlSubjP name="ALL_PREFIX">
    <rtctrlMatchRtDest ip="0.0.0.0/0" aggregate="yes"/>
</rtctrlSubjP>

<!-- match rule with an IP prefix-list -->
<rtctrlSubjP name="PFX_10-0-0-0_24">
    <rtctrlMatchRtDest ip="10.0.0.0/24" aggregate="yes"/>
</rtctrlSubjP>

<!-- setu rule for community attribute -->
<rtctrlAttrP name="COM_A">
    <rtctrlSetComm type="community" setCriteria="append" community="regular:as2-nn2:100:200"/>
</rtctrlAttrP>
</fvTenant>

```

ステップ2 構成されたルートマップを L3Out に適用します。

次の例では、ステップ 1 から L3Out l3out1 からルート マップへ特定の L3Out からのルートのインター再配布をカスタマイズするために適用します。

L3extRsInterleakPol は、特定の L3Out によって使用されるダイナミック ルーティング プロトコル (OSPF/EIGRP) ルートに適用されます。L3extRsRedistributePol は、静的ルートに適用されます。src 属性 ([静的 (static)]) で指定されているように適用されます。

例：

```

POST: https://<APIC IP>/api/mo/uni.xml
BODY:
<fvTenant dn="uni/tn-SAMPLE">
    <l3extOut name="l3out1">
        <!-- interleak redistribution for OSPF/EIGRP routes -->
        <l3extRsInterleakPol tnRtctrlProfileName="INTERLEAK_RP"/>
        <!-- interleak redistribution for static routes -->
        <l3extRsRedistributePol tnRtctrlProfileName="INTERLEAK_RP" src="static"/>
    </l3extOut>
</fvTenant>

```

REST API を使用したトランジット ルーティングの設定

REST API を使用したトランジット ルーティングの設定

次の手順では、テナントのトランジットルーティングを設定する方法を説明します。この例では、別のルータにそれぞれ接続された 2 つの境界リーフ スイッチで、1 つの VRF 内に 2 つの L3Out を展開します。

始める前に

- ノード、ポート、AEP、機能プロファイル、レイヤ 3 ドメインを構成します。
- 外部ルーテッド ドメインを作成し、L3Out のインターフェイスに関連付けます。
- ファブリック内でルートを伝播させるために、BGP ルート リフレクタ ポリシーを設定します。

手順

ステップ 1 テナントおよび VRF を構成します。

この例では、テナント t1 および VRF v1 を構成します。VRF はまだ展開されていません。

例：

```
<fvTenant name="t1">
  <fvCtx name="v1"/>
</fvTenant>
```

ステップ 2 ノードおよびインターフェイスを構成します。

この例では、2 つの境界リーフ スイッチで、テナント t1 と VRF v1 に 2 つの L3Outs を構成します。VRF は、レイヤ 3 ドメイン dom1 です。

- 最初の L3Out はノード 101 上にあり、nodep1 という名前です。ノード 101 はルータ ID 11.11.11.103 で構成されます。ルーテッド インターフェイス ifp1 は、Eth1/3 にあり、12.12.12.3/24 の IP アドレスを使用します。
- 2 番目の L3Out が ノード 102 上にあり、nodep2 という名前です。ノード 102 はルータ ID 22.22.22.203 に構成されています。これは、ルーテッド インターフェイス ifp2 は、Eth1/3 にあります。これは、23.23.23.1/24 の IP アドレスを使用します。

例：

```
<l3extOut name="l3out1">
  <l3extRsEctx tnFvCtxName="v1"/>
  <l3extLNodeP name="nodep1">
    <l3extRsNodeL3OutAtt rtrId="11.11.11.103" tDn="topology/pod-1/node-101"/>
    <l3extLIIfP name="ifp1"/>
    <l3extRsPathL3OutAtt addr="12.12.12.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/3]"/>
    </l3extLIIfP>
  </l3extLNodeP>
  <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
</l3extOut>

<l3extOut name="l3out2">
  <l3extRsEctx tnFvCtxName="v1"/>
  <l3extLNodeP name="nodep2">
    <l3extRsNodeL3OutAtt rtrId="22.22.22.203" tDn="topology/pod-1/node-102"/>
    <l3extLIIfP name="ifp2"/>
    <l3extRsPathL3OutAtt addr="23.23.23.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-102/pathep-[eth1/3]"/>
    </l3extLIIfP>
  </l3extLNodeP>
```



```
<l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
</l3extOut>
```

ステップ3 両方の境界リーフ スイッチのルーティング プロトコルを設定します。

この例では、両方の境界リーフ スイッチに対して、ASN 100 でプライマリ ルーティング プロトコルとして BGP を構成します。ノード 101 を BGP ピア 15.15.15.2 およびノード 102 を BGP ピア 25.25.25.2 と共に構成します。

例：

```
<l3extOut name="l3out1">
  <l3extLNodeP name="nodep1">
    <bgpPeerP addr="15.15.15.2/24"                ><bgpAsP asn="100"/>
    </bgpPeerP>
  </l3extLNodeP>
</l3extOut>

<l3extOut name="l3out2">
  <l3extLNodeP name="nodep2">
    <bgpPeerP addr="25.25.25.2/24"                ><bgpAsP asn="100"/>
    </bgpPeerP>
  </l3extLNodeP>
</l3extOut>
```

ステップ4 接続ルーティング プロトコルを構成します。

この例では、定期的なエリア ID 0.0.0.0 で両方の L3Outs に対して通信プロトコルとして OSPF を構成します。

例：

```
<l3extOut name="l3out1">
  <ospfExtP areaId="0.0.0.0" areaType="regular"/>
  <l3extLNodeP name="nodep1">
    <l3extLIIfP name="ifp1">
      <ospfIfP/>
    </l3extLIIfP>
  </l3extLNodeP>
</l3extOut>

<l3extOut name="l3out2">
  <ospfExtP areaId="0.0.0.0" areaType="regular"/>
  <l3extLNodeP name="nodep2">
    <l3extLIIfP name="ifp2">
      <ospfIfP/>
    </l3extLIIfP>
  </l3extLNodeP>
</l3extOut>
```

ステップ5 外部 EPG を構成します。

この例は、ノード101 上でネットワーク 192.168.1.0/24 を外部ネットワーク extnw1 として構成します。そして、ノード 102 上で 192.168.2.0/24 を外部ネットワーク extnw2 として構成します。また、ルート制御プロファイル rp1 および rp2 と外部 EPG を関連付けます。

例：

```
<l3extOut name="l3out1">
  <l3extInstP name="extnw1">
    <l3extSubnet ip="192.168.1.0/24" scope="import-security"/>
    <l3extRsInstPToProfile direction="export" tnRtctrlProfileName="rp1"/>
  </l3extInstP>
</l3extOut>
```

```

<l3extOut name="l3out2">
  <l3extInstP name="extnw2">
    <l3extSubnet ip="192.168.2.0/24" scope="import-security"/>
    <l3extRsInstPToProfile direction="export" tnRtctrlProfileName="rp2"/>
  </l3extInstP>
</l3extOut>

```

ステップ6 オプション。ルート マップを構成します。

この例では、インバウンドおよびアウトバウンド方向で各 BGP ピアのルートマップを構成します。l3out1 では、ルートマップ rp1 は、次のインポート接続先に一致するルートに適用されます。192.168.1.0/24 およびルートマップ rp2 は、192.168.2.0/24 のエクスポート接続先に一致するルートに適用されます。l3out2 では、ルートマップの方向を反転します。

例：

```

<fvTenant name="t1">
  <rtctrlSubjP name="match-rule1">
    <rtctrlMatchRtDest ip="192.168.1.0/24" />
  </rtctrlSubjP>
  <rtctrlSubjP name="match-rule2">
    <rtctrlMatchRtDest ip="192.168.2.0/24" />
  </rtctrlSubjP>
  <l3extOut name="l3out1">
    <rtctrlProfile name="rp1">
      <rtctrlCtxP name="ctxp1" action="permit" order="0">
        <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule1" />
      </rtctrlCtxP>
    </rtctrlProfile>
    <rtctrlProfile name="rp2">
      <rtctrlCtxP name="ctxp1" action="permit" order="0">
        <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule2" />
      </rtctrlCtxP>
    </rtctrlProfile>
    <l3extInstP name="extnw1">
      <l3extRsInstPToProfile direction="import" tnRtctrlProfileName="rp1" />
      <l3extRsInstPToProfile direction="export" tnRtctrlProfileName="rp2" />
    </l3extInstP>
  </l3extOut>
  <l3extOut name="l3out2">
    <rtctrlProfile name="rp1">
      <rtctrlCtxP name="ctxp1" action="permit" order="0">
        <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule1" />
      </rtctrlCtxP>
    </rtctrlProfile>
    <rtctrlProfile name="rp2">
      <rtctrlCtxP name="ctxp1" action="permit" order="0">
        <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule2" />
      </rtctrlCtxP>
    </rtctrlProfile>
    <l3extInstP name="extnw2">
      <l3extRsInstPToProfile direction="import" tnRtctrlProfileName="rp2" />
      <l3extRsInstPToProfile direction="export" tnRtctrlProfileName="rp1" />
    </l3extInstP>
  </l3extOut>
</fvTenant>

```

ステップ7 フィルタおよびコントラクトを作成し、EPG が通信できるようにします。

この例では、フィルタ `http-filter` およびコントラクト `httpCtrct` を構成します。外部 EPG およびアプリケーション EPG は、それぞれプロバイダおよびコンシューマとして、すでにコントラクト `httpCtrct` と関連付けられています。

例：

```
<vzFilter name="http-filter">
  <vzEntry name="http-e" etherT="ip" prot="tcp"/>
</vzFilter>
<vzBrCP name="httpCtrct" scope="context">
  <vzSubj name="subj1">
    <vzRsSubjFiltAtt tnVzFilterName="http-filter"/>
  </vzSubj>
</vzBrCP>
```

ステップ 8 コントラクトと外部 EPG を関連付けます。

この例では、外部 EPG `extnw1` は、プロバイダーとしておよび外部 EPG `extnw2` は、コントラクト `httpCtrct` のコンシューマとして関連付けします。

```
<l3extOut name="l3out1">
  <l3extInstP name="extnw1">
    <fvRsProv tnVzBrCPName="httpCtrct"/>
  </l3extInstP>
</l3extOut>
<l3extOut name="l3out2">
  <l3extInstP name="extnw2">
    <fvRsCons tnVzBrCPName="httpCtrct"/>
  </l3extInstP>
</l3extOut>
```

REST API の例: 中継ルーティング

次の例では、REST API を使用して、2 つの境界リーフ スイッチで 2 つの L3Outs を設定します。

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="t1">
    <fvCtx name="v1"/>
    <l3extOut name="l3out1">
      <l3extRsEctx tnFvCtxName="v1"/>
      <l3extLNodeP name="nodep1">
        <bgpPeerP addr="15.15.15.2/24">
          <bgpAsP asn="100"/>
        </bgpPeerP>
        <l3extRsNodeL3OutAtt rtrId="11.11.11.103" tDn="topology/pod-1/node-101"/>
      </l3extLNodeP>
      <l3extLIfP name="ifp1">
        <l3extRsPathL3OutAtt addr="12.12.12.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-101/pathep-[eth1/3]" />
        <ospfIfP/>
      </l3extLIfP>
    </l3extLNodeP>
    <l3extInstP name="extnw1">
      <l3extSubnet ip="192.168.1.0/24" scope="import-security"/>
      <l3extRsInstPToProfile direction="import" tnRtctrlProfileName="rp1"/>
      <l3extRsInstPToProfile direction="export" tnRtctrlProfileName="rp2"/>
    </l3extInstP>
  </l3extOut>
</fvTenant>
</polUni>
```

```

        <fvRsProv tnVzBrCPName="httpCttrct"/>
    </l3extInstP>
    <bgpExtP/>
    <ospfExtP areaId="0.0.0.0" areaType="regular"/>
    <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
    <rtctrlProfile name="rp1">
        <rtctrlCtxP name="ctxp1" action="permit" order="0">
            <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule1"/>
        </rtctrlCtxP>
    </rtctrlProfile>
    <rtctrlProfile name="rp2">
        <rtctrlCtxP name="ctxp1" action="permit" order="0">
            <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule2"/>
        </rtctrlCtxP>
    </rtctrlProfile>
</l3extOut>
<l3extOut name="l3out2">
    <l3extRsEctx tnFvCtxName="v1"/>
    <l3extLNodeP name="nodep2">
        <bgpPeerP addr="25.25.25.2/24">
            <bgpAsP asn="100"/>
        </bgpPeerP>
        <l3extRsNodeL3OutAtt rtrId="22.22.22.203" tDn="topology/pod-1/node-102"
/>

        <l3extLIfP name="ifp2">
            <l3extRsPathL3OutAtt addr="23.23.23.3/24" ifInstT="l3-port"
tDn="topology/pod-1/paths-102/pathep-[eth1/3]" />
            <ospfIfP/>
        </l3extLIfP>
    </l3extLNodeP>
    <l3extInstP name="extnw2">
        <l3extSubnet ip="192.168.2.0/24" scope="import-security"/>
        <l3extRsInstPToProfile direction="import" tnRtctrlProfileName="rp2"/>
        <l3extRsInstPToProfile direction="export" tnRtctrlProfileName="rp1"/>
        <fvRsCons tnVzBrCPName="httpCttrct"/>
    </l3extInstP>
    <bgpExtP/>
    <ospfExtP areaId="0.0.0.0" areaType="regular"/>
    <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
    <rtctrlProfile name="rp1">
        <rtctrlCtxP name="ctxp1" action="permit" order="0">
            <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule1"/>
        </rtctrlCtxP>
    </rtctrlProfile>
    <rtctrlProfile name="rp2">
        <rtctrlCtxP name="ctxp1" action="permit" order="0">
            <rtctrlRsCtxPToSubjP tnRtctrlSubjPName="match-rule2"/>
        </rtctrlCtxP>
    </rtctrlProfile>
</l3extOut>
<rtctrlSubjP name="match-rule1">
    <rtctrlMatchRtDest ip="192.168.1.0/24"/>
</rtctrlSubjP>
<rtctrlSubjP name="match-rule2">
    <rtctrlMatchRtDest ip="192.168.2.0/24"/>
</rtctrlSubjP>
<vzFilter name="http-filter">
    <vzEntry name="http-e" etherT="ip" prot="tcp"/>
</vzFilter>
<vzBrCP name="httpCttrct" scope="context">
    <vzSubj name="subj1">
        <vzRsSubjFiltAtt tnVzFilterName="http-filter"/>
    </vzSubj>
</vzBrCP>

```

```

    </fvTenant>
  </polUni>

```

共有 L3Out

REST API を使用した共有サービスの設定

共有設定の 2 つのレイヤ REST API を使用して 2 つの Vrf に 3 が記録されます。

2 つの方法が表示されますが、2 つの Vrf にレイヤ 3 が記録されるを共有する次の REST API の設定例は次の通信します。

手順

ステップ 1 プロバイダー レイヤ 3 を設定します。

例：

```

<tenant name="t1_provider">
  <fvCtx name="VRF1">
    <l3extOut name="T0-o1-L3OUT-1">
      <l3extRsEctx tnFvCtxName="o1"/>
      <ospfExtP areaId='60'/>
      <l3extInstP name="l3extInstP-1">
        <fvRsProv tnVzBrCPName="vzBrCP-1">
          </fvRsProv>
        <l3extSubnet ip="192.168.2.0/24" scope="shared-rtctrl, shared-security" aggregate=""/>
      </l3extInstP>
    </l3extOut>
  </fvCtx>
</tenant>

```

ステップ 2 レイヤ 3 Out コンシューマを設定します。

例：

```

<tenant name="t1_consumer">
  <fvCtx name="VRF2">
    <l3extOut name="T0-o1-L3OUT-1">
      <l3extRsEctx tnFvCtxName="o1"/>
      <ospfExtP areaId='70'/>
      <l3extInstP name="l3extInstP-2">
        <fvRsCons tnVzBrCPName="vzBrCP-1">
          </fvRsCons>
        <l3extSubnet ip="199.16.2.0/24" scope="shared-rtctrl, shared-security"
aggregate=""/>
      </l3extInstP>
    </l3extOut>
  </fvCtx>
</tenant>

```

REST API を使用した L3Out の QoS の設定

REST API を使用した L3Out での QoS ディレクトリの構成

この章では L3Out で QoS ディレクトリを構成する方法について説明します。これは、[Cisco APIC] リリース 4.0 (1) 以降の L3Out QoS の推奨構成方法です。

次のオブジェクトの内の 1 つで L3Out の QoS を構成できます：

- スイッチ仮想インターフェイス (SVI)
- サブ インターフェイス
- 外部ルーテッド

手順

ステップ 1 L3Out SVI に QoS プライオリティを構成します。

例：

```
<l3extLifP descr="" dn="uni/tn-DT/out-L3_4_2_24_SVI17/lnodep-L3_4_E2_24/lifp-L3_4_E2_24_SVI_19"
  name="L3_4_E2_24_SVI_19" prio="level6" tag="yellow-green">
  <l3extRsPathL3OutAtt addr="0.0.0.0" autostate="disabled" descr="SVI19" encap="vlan-19"
    encapScope="local" ifInstT="ext-svi" ipv6Dad="enabled" llAddr="::"
    mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
  tDn="topology/pod-1/protopaths-103-104/pathep-[V_L3_l4_2-24]"
  targetDscp="unspecified">
    <l3extMember addr="107.2.1.253/24" ipv6Dad="enabled" llAddr="::" side="B"/>
    <l3extMember addr="107.2.1.252/24" ipv6Dad="enabled" llAddr="::" side="A"/>
  </l3extRsPathL3OutAtt>
  <l3extRsLifPCustQosPol tnQosCustomPolName="VrfQos006"/>
</l3extLifP>
```

ステップ 2 サブインターフェイスに QoS プライオリティを構成します。

例：

```
<l3extLifP dn="uni/tn-DT/out-L4E48_inter_tenant/lnodep-L4E48_inter_tenant/lifp-L4E48"
  name="L4E48" prio="level4" tag="yellow-green">
  <l3extRsPathL3OutAtt addr="210.1.0.254/16" autostate="disabled" encap="vlan-20"
    encapScope="local" ifInstT="sub-interface" ipv6Dad="enabled" llAddr="::"
    mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
  tDn="topology/pod-1/paths-104/pathep-[eth1/48]" targetDscp="unspecified"/>
  <l3extRsNdIfPol annotation="" tnNdIfPolName=""/>
  <l3extRsLifPCustQosPol annotation="" tnQosCustomPolName=" vrfQos002"/>
</l3extLifP>
```

ステップ 3 外部ルーテッドに QoS プライオリティを構成します。

例：

```
<l3extLifP dn="uni/tn-DT/out-L2E37/lnodep-L2E37/lifp-L2E37OUT"
  name="L2E37OUT" prio="level5"
  tag="yellow-green">
  <l3extRsPathL3OutAtt addr="30.1.1.1/24" autostate="disabled" encap="unknown"
    encapScope="local" ifInstT="l3-port" ipv6Dad="enabled"
    llAddr="::"
    mac="00:22:BD:F8:19:FF" mode="regular"
    mtu="inherit" targetDscp="unspecified"
  tDn="topology/pod-1/paths-102/pathep-[eth1/37]"/>
```

```
<l3extRsNdIfPol annotation="" tnNdIfPolName=""/>
<l3extRsLIIfPCustQosPol tnQosCustomPolName="vrfQos002"/>
</l3extLIIfP>
```

REST API を使用した L3Out の QoS コントラクトの設定

この項では、コントラクトを使用して L3Out の QoS を設定する方法について説明します。



- (注) リリース 4.0 (1) 以降では、L3Out QoS 用にカスタム QoS ポリシーを使用することを推奨しています。[REST API を使用した L3Out での QoS ディレクトリの構成 \(62 ページ\)](#) で説明しています。

手順

ステップ 1 テナント、VRF、およびブリッジドメインを構成する場合は、VRF を出力モードに構成します (pcEnfDir = "egress")、ポリシーの適用が有効になっている (pcEnfPref="enforced")。次の例のように XML で post を送信します。

例：

```
<fvTenant name="t1">
  <fvCtx name="v1" pcEnfPref="enforced" pcEnfDir="egress"/>
  <fvBD name="bd1">
    <fvRsCtx tnFvCtxName="v1"/>
    <fvSubnet ip="44.44.44.1/24" scope="public"/>
    <fvRsBDToOut tnL3extOutName="l3out1"/>
  </fvBD>
</fvTenant>
```

ステップ 2 通信のため L3Out に参加して EPG を有効にする契約を作成するときは、優先順位の QoS を構成します。

この例のコントラクトには、L3Out でのトラフィック入力のための QoS 優先順位の level-1 が含まれています。または、ターゲットの DSCP 値を定義する可能性があります。QoS ポリシーは、契約またはサブジェクトのいずれかでサポートされます。

フィルタに matchDscp = 「Ef」 条件があるため、このタグを持つトラフィックがコントラクト件名で指定されたキューを通して L3out プロセスにより受信できます。

(注)

L3out インターフェイスでの QOS またはカスタム QOS では VRF の適用は入力とします。VRF の適用を出力にする必要があるのは、QOS 分類が EPG と L3out の間、または L3out から L3out へのトラフィックの契約で実行される場合に限ります。

(注)

QOS 分類が契約で設定され、VRF の適用が出力である場合、契約 QOS 分類は L3out インターフェイス QOS またはカスタム QOS 分類をオーバーライドするため、これか新しいもののいずれかを設定する必要があります。

例：

```
<vzFilter name="http-filter">
  <vzEntry name="http-e" etherT="ip" prot="tcp" matchDscp="EF"/>
</vzFilter>
<vzBrCP name="httpCtrct" prio="level1" scope="context">
  <vzSubj name="subj1">
    <vzRsSubjFiltAtt tnVzFilterName="http-filter"/>
  </vzSubj>
</vzBrCP>
```

REST API を使用した SR-MPLS カスタム QoS ポリシー

SR MPLS カスタム QoS ポリシーは、MPLS QoS 出力 ポリシーで定義された着信 MPLS EXP 値に基づいて、SR-MPLS ネットワークから送信されるパケットのプライオリティを定義します。これらのパケットは、ACI ファブリック内にあります。また、MPLS QoS 出力ポリシーで定義された IPv4 DSCP 値に基づく MPLS インターフェイスを介して ACI ファブリックから離れるパケットの CoS 値および MPLS EXP 値をマーキングします。

カスタム出力ポリシーが定義されていない場合、デフォルトの QoS レベル（レベル3）がファブリック内のパケットに割り当てられます。カスタム イングレス ポリシーが定義されていない場合、0 のデフォルト EXP 値は、ファブリックから離れるパケットにマークされます。

手順

ステップ 1 SR-MPLS QoS ポリシーの作成します。

次の POST で：

- *customqos1* を、作成する SR-MPLS QoS ポリシーの名前に置き換えます。
- *qosMplsIngressRule* の場合：
 - *from="2" to="3"* を、ポリシーに一致させる EXP 範囲に置き換えます。
 - *prio =level15* を ACI ファブリック内にあるパケットの ACI QoS レベルに置き換えます。
 - *target="CS5"* は、パケットが一致したときに設定する DSCP 値に置き換えます。
 - *targetCos =4* を、パケットが一致したときにパケットに設定する CoS 値に置き換えます。
- *qosMplsEgressRule* の場合：
 - *from="CS2" to="CS4"* を、ポリシーを照合する DSCP 範囲に置き換えます。
 - *targetExp =5* を、パケットがファブリックを離れるときに設定する EXP 値に置き換えます。
 - *targetCos =3* を、パケットがファブリックを離れるときに設定する CoS 値に置き換えます。


```
<polUni>
  <fvTenant name="infra">
    <qosMplsCustomPol descr="" dn="uni/tn-infra/qosmplscustom-">customqos1" name="customqos1"
    status="" >
      <qosMplsIngressRule from="">2" to="3" prio="level5" target="CS5" targetCos="4" status="" />

      <qosMplsEgressRule from="">CS2" to="CS4" targetExp="5" targetCos="3" status="" />
    </qosMplsCustomPol>
  </fvTenant>
</polUni>
```

ステップ2 SR-MPLS QoS ポリシーの適用します。

次の POST で、次を置換します。 *customqos1* を前の手順で作成した SR-MPLS QoS ポリシーの名前に置き換えます。

```
<polUni>
  <fvTenant name="infra">
    <l3extOut name="mplsOut" status="" descr="bl">
      <l3extLNodeP name="mplsLNP" status="">
        <l3extRsLNodePMplsCustQosPol tDn="uni/tn-infra/qosmplscustom-">customqos1"/>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>
```

REST API を使用した ACI IP SLA の設定

REST API を使用した IP SLA モニタリング ポリシーの構成

[Cisco APIC] を有効にして REST API を使用して特定の SLA タイプのモニタリング プロブを送信するには、次の手順を実行します：

手順

IP SLA モニタリング ポリシーの構成

例：

```
<?xml version="1.0" encoding="utf-8"?>
<imdata totalCount="1">
  <fvIPSLAMonitoringPol annotation="" descr=""
  dn="uni/tn-t8/ipslaMonitoringPol-ICMP-Probe" name="ICMP-Probe" nameAlias=""
  ownerKey="" ownerTag="" slaDetectMultiplier="3" slaFrequency="5" slaPort="0"
  slaType="icmp"/>
</imdata>
```

REST API を使用した IP-SLA トラック メンバーの設定

REST API を使用して IP SLA トラック メンバーを設定するには、次の手順を実行します。

手順

IP SLA トラック メンバーを設定します。

例：

```
<?xml version="1.0" encoding="utf-8"?>
<imdata totalCount="1">
  <fvTrackMember annotation="" descr="" dn="uni/tn-t8/trackmember-TM_pc_sub"
    dstIpAddr="52.52.52.1" name="TM_pc_sub" nameAlias="" ownerKey="" ownerTag=""
    scopeDn="uni/tn-t8/out-t8_l3">
    <fvRsIpslaMonPol annotation="" tDn="uni/tn-t8/ipslaMonitoringPol-TCP-Telnet"/>
  </fvTrackMember>
</imdata>
```

REST API を使用した IP-SLA トラック リストの設定

REST API を使用して IP SLA トラック リストを設定するには、次の手順を実行します。

手順

IP SLA トラック リストを設定します。

例：

```
<?xml version="1.0" encoding="utf-8"?>
<imdata totalCount="1">
  <fvTrackList annotation="" descr="" dn="uni/tn-t8/tracklist-T8_pc_sub1"
    name="T8_pc_sub1" nameAlias="" ownerKey="" ownerTag="" percentageDown="0"
    percentageUp="1" type="weight" weightDown="5" weightUp="10">
    <fvRsOtmListMember annotation="" tDn="uni/tn-t8/trackmember-TM_pc_sub"
      weight="10"/>
  </fvTrackList>
</imdata>
```

REST API を使用したスタティック ルートとトラック リストの関連付け

REST API を使用して IP SLA トラック リストをスタティック ルートに関連付けるには、次の手順を実行します。

手順

IP SLA トラック リストをスタティック ルートに関連付けます。

例：

```
<?xml version="1.0" encoding="utf-8"?>
<imdata totalCount="1">
  <ipRouteP aggregate="no" annotation="" descr=""
dn="uni/tn-t8/out-t8_l3/lnodep-t8_l3_vpc1/rsnodeL3OutAtt-[topology/pod-2/node-108]/rt-[88.88.88.2/24]"
  ip="88.88.88.2/24" name="" nameAlias="" pref="1" rtCtrl="">
    <ipRsRouteTrack annotation="" tDn="uni/tn-t8/tracklist-T8_TL1_Static"/>
    <ipNexthopP annotation="" descr="" name="" nameAlias="" nhAddr="23.23.2.3"
      pref="1" type="prefix"/>
  </ipRouteP>
</imdata>
```

REST API を使用してネクストホッププロファイルのトラックリストに関連付けをする

REST API を使用して IP SLA トラックリストをネクストホッププロファイルに関連付けるには、次の手順を実行します。

手順

IP SLA トラックリストをネクストホッププロファイルに関連付けます。

例：

```
<?xml version="1.0" encoding="utf-8"?>
<imdata totalCount="1">
  <ipRouteP aggregate="no" annotation="" descr=""
dn="uni/tn-t8/out-t8_l3/lnodep-t8_l3_vpc1/rsnodeL3OutAtt-[topology/pod-2/node-109]/rt-[86.86.86.2/24]"
  ip="86.86.86.2/24" name="" nameAlias="" pref="1" rtCtrl="">
    <ipNexthopP annotation="" descr="" name="" nameAlias="" nhAddr="25.25.25.3"
      pref="1" type="prefix">
      <ipRsNexthopRouteTrack annotation=""
tDn="uni/tn-t8/tracklist-ctx0_25.25.25.3"/>
      <ipRsNHTrackMember annotation=""
tDn="uni/tn-t8/trackmember-ctx0_25.25.25.3"/>
    </ipNexthopP>
  </ipRouteP>
</imdata>
```

REST API を使用した HSRP の設定

REST API を使用した APIC 内の HSRP の設定

リーフスイッチが設定されている場合、HSRP が有効になっています。

始める前に

- テナントおよび VRF を設定する必要があります。
- VLAN プールは、適切な VLAN 範囲が定義され、レイヤ 3 ドメインが作成されて VLAN プールに接続されている状態で設定される必要があります。

- エンティティプロファイルの接続も、レイヤ3ドメインに関連付けられている必要があります。
- リーフ スイッチのインターフェイス プロファイルは必要に応じて設定する必要があります。

手順

ステップ1 ポート セレクタを作成します。

例：

```
<polUni>
  <infraInfra dn="uni/infra">
    <infraNodeP name="TenantNode_101">
      <infraLeafS name="leafselector" type="range">
        <infraNodeBlk name="nodeblk" from_"101" to_"101">
          </infraNodeBlk>
        </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-TenantPorts_101"/>
    </infraNodeP>
    <infraAccPortP name="TenantPorts_101">
      <infraHPortS name="portselector" type="range">
        <infraPortBlk name="portblk" fromCard="1" toCard="1" fromPort="41" toPort="41">
          </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-TenantPortGrp_101"/>
      </infraHPortS>
    </infraAccPortP>
    <infraFuncP>
      <infraAccPortGrp name="TenantPortGrp_101">
        <infraRsAttEntP tDn="uni/infra/attentp-AttEntityProfTenant"/>
        <infraRsHifPol tnFabricHifPolName="default"/>
      </infraAccPortGrp>
    </infraFuncP>
  </infraInfra>
</polUni>
```

ステップ2 テナント ポリシーを作成します。

例：

```
<polUni>
  <fvTenant name="t9" dn="uni/tn-t9" descr="">
    <fvCtx name="t9_ctx1" pcEnfPref="unenforced">
      </fvCtx>
    <fvBD name="t9_bd1" unkMacUcastAct="flood" arpFlood="yes">
      <fvRsCtx tnFvCtxName="t9_ctx1"/>
      <fvSubnet ip="101.9.1.1/24" scope="shared"/>
    </fvBD>
    <l3extOut dn="uni/tn-t9/out-l3extOut1" enforceRtctrl="export" name="l3extOut1">
      <l3extLNodeP name="Node101">
        <l3extRsNodeL3OutAtt rtrId="210.210.121.121" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"/>
      </l3extLNodeP>
      <l3extRsEctx tnFvCtxName="t9_ctx1"/>
      <l3extRsL3DomAtt tDn="uni/l3dom-dom1"/>
      <l3extInstP matchT="AtleastOne" name="extEpg" prio="unspecified" targetDscp="unspecified">
        <l3extSubnet aggregate="" descr="" ip="176.21.21.21/21" name="" scope="import-security"/>
      </l3extInstP>
    </l3extOut>
  </fvTenant>
</polUni>
```

```

    </l3extInstP>
  </l3extOut>
</fvTenant>
</polUni>

```

ステップ3 LLDP インターフェイス ポリシーを作成します。

例：

```

<polUni>
  <fvTenant name="t9" dn="uni/tn-t9" descr="">
    <hsrpIfPol name="hsrpIfPol" ctrl="bfd" delay="4" reloadDelay="11"/>
  </fvTenant>
</polUni>

```

ステップ4 HSRP グループ ポリシーを作成します。

例：

```

<polUni>
  <fvTenant name="t9" dn="uni/tn-t9" descr="">
    <hsrpIfPol name="hsrpIfPol" ctrl="bfd" delay="4" reloadDelay="11"/>
  </fvTenant>
</polUni>

```

ステップ5 HSRP インターフェイス プロファイルおよび HSRP グループ プロファイルを作成します。

例：

```

<polUni>
  <fvTenant name="t9" dn="uni/tn-t9" descr="">
    <l3extOut dn="uni/tn-t9/out-l3extOut1" enforceRtctrl="export" name="l3extOut1">
      <l3extLNodeP name="Node101">
        <l3extLIIfP name="eth1-41-v6" ownerKey="" ownerTag="" tag="yellow-green">
          <hsrpIfP name="eth1-41-v6" version="v2">
            <hsrpRsIfPol tnHsrpIfPolName="hsrpIfPol"/>
            <hsrpGroupP descr="" name="HSRPV6-2" groupId="330" groupAf="ipv6" ip="fe80::3"
mac="00:00:0C:18:AC:01" ipObtainMode="admin">
              <hsrpRsGroupPol tnHsrpGroupPolName="G1"/>
            </hsrpGroupP>
          </hsrpIfP>
          <l3extRsPathL3OutAtt addr="2002::100/64" descr="" encap="unknown" encapScope="local"
ifInstT="l3-port" llAddr="::" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/41]" targetDscp="unspecified">
            <l3extIp addr="2004::100/64"/>
          </l3extRsPathL3OutAtt>
        </l3extLIIfP>
        <l3extLIIfP name="eth1-41-v4" ownerKey="" ownerTag="" tag="yellow-green">
          <hsrpIfP name="eth1-41-v4" version="v1">
            <hsrpRsIfPol tnHsrpIfPolName="hsrpIfPol"/>
            <hsrpGroupP descr="" name="HSRPV4-2" groupId="51" groupAf="ipv4" ip="177.21.21.21"
mac="00:00:0C:18:AC:01" ipObtainMode="admin">
              <hsrpRsGroupPol tnHsrpGroupPolName="G1"/>
            </hsrpGroupP>
          </hsrpIfP>
          <l3extRsPathL3OutAtt addr="177.21.21.11/24" descr="" encap="unknown" encapScope="local"
ifInstT="l3-port" llAddr="::" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/41]" targetDscp="unspecified">
            <l3extIp addr="177.21.23.11/24"/>
          </l3extRsPathL3OutAtt>
        </l3extLIIfP>
      </l3extLNodeP>
    </l3extOut>
  </fvTenant>
</polUni>

```

```
</fvTenant>
</polUni>
```

REST API を使用した Cisco ACI GOLF の設定

構成 [GOLF] REST API の使用

手順の概要

1. 次の例では、REST API を使用して [GOLF] のノードおよびスパインスイッチインターフェイスを展開する方法を示しています、REST API を使用：
2. 次の XML で、スパインスイッチインターフェイスと [GOLF] サービスのインフラテナントプロバイダを構成します。次の XML 構造を POST メッセージの本文に含めます。
3. 次の XML で、[GOLF] サービスのインフラ部分のテナント コンシューマを構成します。次の XML 構造を POST メッセージの本文に含めます。

手順の詳細

手順

ステップ 1 次の例では、REST API を使用して [GOLF] のノードおよびスパインスイッチインターフェイスを展開する方法を示しています、REST API を使用：

例：

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

ステップ 2 次の XML で、スパインスイッチインターフェイスと [GOLF] サービスのインフラテナントプロバイダを構成します。次の XML 構造を POST メッセージの本文に含めます。

例：

```
<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-l11/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-l11/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>

```

ステップ3 次の XML で、[GOLF] サービスのインフラ部分のテナント コンシューマを構成します。次の XML 構造を POST メッセージの本文に含めます。

例：

```

<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="webCtct-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="dc1-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 tnMonEPGPName=""/>
    <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=""                                instrImedcy="immediate"
encap="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>

```

REST API を使用した DCIG への BGP EVPN タイプ 2 ホスト ルート配信の有効化

次のように REST API を使用して、BGP EVPN タイプ 2 ホスト ルートの配信を有効にします。

始める前に

EVPN サービスを設定する必要があります。

手順

ステップ 1 次の例のように、XML が含まれている POST で、ホスト ルート リーク ポリシーを設定します。

例：

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

ステップ 2 次の例のように、XML が含まれている POST を使用してアドレス ファミリの一方または両方の VRF BGP アドレス ファミリ コンテキスト ポリシーに、ポリシーを適用します。

例：

```

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

```


翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。