ルーティング プロトコル サポート

この章の内容は、次のとおりです。

概要

ルーティング プロトコル サポート

Cisco ACI ファブリック内のルーティングは、BGP（BFD サポート）および OSPF または EIGRP ルーティング プロトコルを使用して実装されます。

IP 送信元ルーティングは ACI ファブリックではサポートされません。

BGP 外部ルーテッド ネットワークと BFD のサポート

BGP レイヤ 3 外部ネットワーク接続設定のガイドライン

BGP 外部ルーテッドネットワークを設定するときは、以下のガイドラインに従ってください。

- リーフ スイッチにルータ ID を作成すると、必ず内部ループバック アドレスが作成されます。リーフ スイッチに BGP 接続をセットアップする場合、ルート ID をインターフェイスの IP アドレスと同じにすることはできません。これは、その設定が ACI リーフ スイッチではサポートされていないためです。ルータ ID は、別のサブネット内の別のアドレスである必要があります。外部レイヤ 3 デバイスでは、ルータ ID はループバック アドレスまたはインターフェイス アドレスです。スタティック ルートまたは OSPF 設定のいずれかを使用して、レイヤ 3 デバイスのルーティング テーブルにリーフ ルータ ID へのルートが存在することを確認してください。また、レイヤ 3 デバイスに BGP ネイバーをセットアッ

プする場合、使用するピア IP アドレスはリーフ スイッチのルータ ID である必要があります。

- BGP を使用する 2 つの外部レイヤ 3 ネットワークを同じノードに設定する際、ループバック アドレスを明示的に定義する必要があります。このガイドラインに従わないと、BGP を確立できない可能性があります。
- 定義上、ルータ ID はループバック インターフェイスです。ルータ ID を変更してループバックに別のアドレスを割り当てるには、ループバック インターフェイス ポリシーを作成する必要があります（ループバック ポリシーは、アドレス ファミリ、IPv4、および IPv6 ごとに 1 つずつ設定できます）。ループバック ポリシーを作成しない場合は、ルータ ID ループバック（デフォルトで有効）を有効にすることができます。ルータ ID ループバックが無効である場合、導入先の特定のレイヤ 3 Outside に対するループバックは作成されません。
- この設定作業は iBGP および eBGP に適用されます。BGP 設定がループバック アドレスに対するものである場合、iBGP セッションまたはマルチホップ eBGP セッションです。ピア IP アドレスが BGP ピアが定義されている物理インターフェイスに対するものである場合、物理インターフェイスが使用されます。
- IPv6 を使用したループバックを介したピアリングを有効にするには、ユーザが IPv6 アドレスを設定する必要があります。
- 自律システム機能は eBGP ピアでしか使用できません。この機能では、ルータが実際の AS に加えて、2 番めの自律システム（AS）のメンバであるように見せることができます。ローカル AS を使用すると、ピアリングの調整を変更せずに 2 つの ISP をマージできます。マージされた ISP 内のルータは、新しい自律システムのメンバになりますが、使用者に対しては古い自律システム番号を使用し続けます。
- リリース 1.2（1x）以降、BGP `l3extOut` 接続のテナント ネットワーキング プロトコル ポリシーは、最大プレフィックス制限を使用して設定できます。これにより、ピアから受信されるルート プレフィックスの数をモニタし、制限することができます。最大プレフィックス制限を超えると、ログ エントリの記録、それ以降のプレフィックスの拒否、固定期間中にカウントがしきい値未満になった場合の接続の再起動、または接続のシャットダウンを行うことができます。一度に 1 つのオプションだけを使用できます。デフォルト設定では 20,000 プレフィックスに制限され、その後は新しいプレフィックスは拒否されます。拒否オプションが導入されると、BGP は設定されている制限よりも 1 つ多くプレフィックスを受け入れ、APIC でエラーが発生します。

（注） Cisco ACI は、IP フラグメンテーションをサポートしていません。したがって、外部ルータへのレイヤ 3 Outside（L3Out）接続、または Inter-Pod Network（IPN）を介した multipod 接続を設定する場合は、MTU が両側で適切に設定されていることが重要です。ACI、Cisco NX-OS、Cisco IOS などの一部のプラットフォームでは、設定された MTU 値は IP ヘッダーを考慮に入れています（結果として、最大パケット サイズは、ACI で 9216 バイト、NX-OS および IOS で 9000 バイトに設定されます）。ただし、IOS XR などの他のプラットフォームは、パケット ヘッダーのを除く MTU 値を設定します（結果として最大パケット サイズは 8986 バイトになります）。

各プラットフォームの適切な MTU 値については、それぞれの設定ガイドを参照してください。

CLI ベースのコマンドを使用して MTU をテストすることを強く推奨します。たとえば、Cisco NX-OS CLI で `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1` などのコマンドを使用します。

BGP の接続タイプとループバックのガイドライン

ACI では次の BGP 接続の種類をサポートし、それらのループバックのガイドラインをまとめています。

BGP 接続タイプ	ループバックが必要	ルータ ID と同じループバック	スタティック ルートまたは OSPF ルートが必要
直接 iBGP	いいえ（No）	該当なし	いいえ
iBGP ループバック ピアリング	はい（BGP ピアごとに個別のループバック）	いいえ（同じノードに複数のレイヤ 3 Out がある場合）	はい
直接 eBGP	いいえ（No）	該当なし	いいえ
eBGP ループバック ピアリング（マルチホップ）	はい（BGP ピアごとに個別のループバック）	いいえ（同じノードに複数のレイヤ 3 Out がある場合）	○

BGP 外部ルーテッド ネットワークの設定

GUI 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、VRF、およびブリッジ ドメインがすでに作成されていること。

手順

ステップ 1 [Navigation] ペインで、**[Tenant_name]** > **[Networking]** > **[External Routed Networks]** を展開します。

ステップ 2 右クリックし、[Create Routed Outside] をクリックします。

ステップ 3 [Create Routed Outside] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、外部ルーテッド ネットワーク ポリシーの名前を入力します。

b) [BGP] チェックボックスをクリックします。

（注） 次の 2 つの方法のいずれかで、BGP ピアの到達可能性を使用できるようになっている必要があります。スタティック ルートを設定するか、または OSPF を有効にする必要があります。

c) （任意） [Route Control Enforcement] フィールドで、[mport] チェックボックスをオンにします。

（注） BGP でインポート制御を適用する場合は、このチェックボックスをオンにします。

d) [VRF] フィールドのドロップダウン リストから、目的の VRF を選択します。

e) [Route Control for Dampening] フィールドを展開し、目的のアドレス ファミリ タイプとルート ダンプニング ポリシーを選択します。[Update] をクリックします。

このステップでは、ポリシーはステップ 4 で作成することができます。または、ポリシー名が選択されているドロップダウン リストでルート プロファイルを作成するオプションがあります。

f) [Nodes and Interfaces Protocol Policies] を展開します。

g) [Create Node Profile] ダイアログボックスに、ノード プロファイルの名前を入力します。

h) [Nodes] を展開します。

i) [Select Node] ダイアログボックスの [Node ID] フィールドのドロップダウン リストから、ノードを選択します。

j) [Router ID] フィールドに、ルータ ID を入力します。

k) [Loopback Address] を展開し、[IP] フィールドに IP アドレスを入力します。[Update] をクリックします。

（注） IPv6 アドレスを入力します。前のステップでルータ ID を追加しなかった場合は、[IP] フィールドに IPv4 アドレスを追加できます。

l) [OK] をクリックします。

ステップ 4 [Navigation] ペインで、**[Tenant_name]** > **[Networking]** > **[Route Profiles]** の順に展開します。[Route Profiles] を右クリックし、[Create Route Profile] をクリックします。[Create Route Profile] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、ルート制御 VRF の名前を入力します。
b) [Create Route Control Context] ダイアログボックスを展開します。
c) [Name] フィールドに、ルート制御 VRF の名前を入力します。
d) [Set Attribute] ドロップダウン リストから、[Create Action Rule Profile] を選択します。

アクション ルールを作成するときに、必要に応じてルート ダンプニング属性を設定します。

ステップ 5 [Create Interface Profiles] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、インターフェイス プロファイル名を入力します。
b) [Interfaces] 領域で、目的のインターフェイス タブを選択し、インターフェイスを展開します。

ステップ 6 [Select Routed Interface] ダイアログボックスで、次の操作を実行します。

a) [Path] ドロップダウン リストから、ノードおよびインターフェイスを選択します。
b) [IP Address] フィールドに、IP アドレスを入力します。

（注） 必要に応じて、IPv6 アドレスまたは IPv4 アドレスを追加できます。

c) （任意） 前のステップで IPv6 アドレスを入力した場合は、[Link-local Address] フィールドに IPv6 アドレスを入力します。
d) [BGP Peer Connectivity Profile] フィールドを展開します。

ステップ 7 [Create Peer Connectivity Profile] ダイアログボックスで、次の操作を実行します。

a) [Peer Address] フィールドでは、ダイナミック ネイバー機能を使用できます。必要に応じて、指定されたサブネット内のすべてのピアが BGP と通信またはルートを交換できます。

手順内の前のステップで入力した IPv4 または IPv6 のアドレスに対応する IPv4 または IPv6 のアドレスを入力します。

b) [BGP Controls] フィールドで、目的の制御をオンにします。
c) [Autonomous System Number] フィールドで、目的の値を選択します。
d) （任意） [Weight for routes from this neighbor] フィールドで、目的の値を選択します。
e) （任意） [Private AS Control] フィールドで、[Remove AS] のチェックボックスをオンにします。
f) （任意） [Local Autonomous System Number Config] フィールドで、目的の値を選択します。

eBGP ピアのローカル自律システム機能の場合にオプションで必要です。

g) （任意） [Local Autonomous System Number] フィールドで、目的の値を選択します。

eBGP ピアのローカル自律システム機能の場合にオプションで必要です。

（注） このフィールドの値は、[Autonomous System Number] フィールドの値と同じであってはなりません。

h) [OK] をクリックします。

ステップ 8 次のアクションを実行します。

a) [Select Routed Interface] ダイアログボックスで、[OK] をクリックします。
b) [Create Interface Profile] ダイアログボックスで、[OK] をクリックします。
c) [Create Node Profile] ダイアログボックスで、[OK] をクリックします。
[External EPG Networks] 領域が表示されます。
d) [Create Routed Outside] ダイアログボックスで、前に作成したノード プロファイルを選択し、[Next] をクリックします。

ステップ 9 [External EPG Networks] を展開し、[Create External Network] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、外部ネットワークの名前を入力します。
b) [Subnet] を展開します。
c) [Create Subnet] ダイアログボックスの [IP address] フィールドに、必要に応じてサブネットアドレスを入力します。

（注） 前のステップで入力した内容に応じて、IPv4 または IPv6 のアドレスを入力します。

外部サブネットを作成するときに、プレフィックス EPG の BGP ループバックの両方を設定するか、またはどちらも設定しない必要があります。BGP ループバックを 1 つのみ設定すると、BGP ネイバーシップは確立されません。

d) [Scope] フィールドで、[Export Route Control Subnet] 、[Import Route Control Subnet] 、および [Security Import Subnet] のチェックボックスをオンにします。[OK] をクリックします。

（注） BGP でインポート制御を適用する場合は、[Import Route Control Subnet] チェックボックスをオンにします。

ステップ 10 **[Create External Network]** ダイアログボックスで、**[OK]** をクリックします。

ステップ 11 **[Create Routed Outside]** ダイアログボックスで、**[Finish]** をクリックします。
eBGP は外部接続用に設定されています。

NX-OS スタイルの CLI を使用した BGP 外部ルーテッド ネットワークの設定

手順

ここでは、NX-OS CLI 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：

```
apic1(config-leaf)#template route-profile damp_rp tenant t1
This template will be available on all leaves where tenant t1 has a VRF deployment
apic1(config-leaf-template-route-profile)#set dampening 15 750 2000 60
apic1(config-leaf-template-route-profile)#exit
apic1(config-leaf)#
apic1(config-leaf)#router bgp 100
apic1(config-bgp)#vrf member tenant t1 vrf ctx3
apic1(config-leaf-bgp-vrf)# neighbor 32.0.1.0/24 l3out l3out-bgp
apic1(config-leaf-bgp-vrf-neighbor)#update-source ethernet 1/16.401
apic1(config-leaf-bgp-vrf-neighbor)#address-family ipv4 unicast
apic1(config-leaf-bgp-vrf-neighbor-af)#weight 400
apic1(config-leaf-bgp-vrf-neighbor-af)#exit
apic1(config-leaf-bgp-vrf-neighbor)#remote-as 65001
apic1(config-leaf-bgp-vrf-neighbor)#private-as-control remove-exclusive
apic1(config-leaf-bgp-vrf-neighbor)#private-as-control remove-exclusive-all
apic1(config-leaf-bgp-vrf-neighbor)#private-as-control remove-exclusive-all-replace-as
apic1(config-leaf-bgp-vrf-neighbor)#exit
apic1(config-leaf-bgp-vrf)# address-family ipv4 unicast
apic1(config-leaf-bgp-vrf-af)#inherit bgp dampening damp_rp
This template will be inherited on all leaves where VRF ctx3 has been deployed
apic1(config-leaf-bgp-vrf-af)#exit
apic1(config-leaf-bgp-vrf)# address-family ipv6 unicast
apic1(config-leaf-bgp-vrf-af)#inherit bgp dampening damp_rp
This template will be inherited on all leaves where VRF ctx3 has been deployed
apic1(config-leaf-bgp-vrf-af)#exit
```

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>
```

BGP 最大パスの設定

BGP Max Path の設定

次の機能を使用すると、等コスト マルチパスのロード バランシングを有効にするルート テーブルへのパスの最大数を追加できます。

GUI を使用した BGP Max Path の設定

始める前に

適切なテナントと BGP 外部ルーティング ネットワークが作成され、使用可能になります。

手順

ステップ 1 APIC GUI にログインし、**[Tenants]** > **<Your_Tenant>** > **[Networking]** > **[Protocol Policies]** > **[BGP]** > **[BGP Address Family Context]** をクリックし、**[Create BGP Address Family Context Policy]** を右クリックします。

ステップ 2 **[Create BGP Address Family Context Policy]** ダイアログ ボックスで、次のタスクを実行します。

a) **[Name]** フィールドにポリシーの名前を入力します。
b) **[eBGP Distance]** フィールドをクリックして、実装の値を確認します。
c) **[iBGP Distance]** フィールドをクリックして、実装の値を確認します。
d) **[Local Distance]** フィールドをクリックして、実装の値を確認します。
e) **[eBGP Max ECMP]** フィールドをクリックして、実装の値を確認します。
f) **[iBGP Max ECMP]** フィールドをクリックして、実装の値を確認します。
g) エントリを更新した後、**[Submit]** をクリックします。

ステップ 3 **[Tenants]** > **<Your_Tenant>** > **[Networking]** > **[VRFs]** > **<your_VRF>** をクリックします。

ステップ 4 対象の VRF の設定の詳細を確認します。

ステップ 5 **[BGP Context Per Address Family]** フィールドにアクセスし、[Address Family] ドロップダウン リストで [Ipv6] を選択します。

ステップ 6 **[BGP Address Family Context]** ドロップダウン リストで作成した [BGP Address Family Context] にアクセスし、それをサブジェクト VRF に関連付けます。

ステップ 7 [送信（Submit）] をクリックします。

NX-OS スタイルの CLI を使用した BGP 最大パスの設定

始める前に

適切なテナントと BGP 外部ルーテッド ネットワークが作成され、使用可能になっています。

さらに多くのパスを設定できるようにする 2 つのプロパティは、bgpCtxAfPol オブジェクトの maxEcmp と maxEcmpIbgp です。これら 2 つのプロパティを設定した後、実装の残り部分に反映されます。

BGP にログインして、次のコマンドを使用します:

```
maximum-paths [ibgp]
```

```
no maximum-paths [ibgp]
```

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# template bgp address-family newAf tenant t1
This template will be available on all nodes where tenant t1 has a VRF deployment
apic1(config-bgp-af)# maximum-paths ?
<1-16>  Maximum number of equal-cost paths for load sharing. The default is 16.
ibgp  Configure multipath for IBGP paths
apic1(config-bgp-af)# maximum-paths 10
apic1(config-bgp-af)# maximum-paths ibpg 8
apic1(config-bgp-af)# end
apic1#
no maximum-paths [ibgp]
```

REST API を使用した BGP パスの設定

次の例では、REST API を使用して BGP 最長パス機能を設定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="8" maxEcmpIbgp="4"/>
</fvTenant>
```

AS パスのプリペンドの設定

AS パス プリペンドの設定

BGP ピアは、AS パス アトリビュートの長さを増やすことで、リモート ピアでベスト パス選択の影響を与えることができます。番号として指定桁の前に付加して AS パス アトリビュートの長さを向上するために使用するメカニズムを提供する AS パス Prepend。

AS パス前に付加は、ルート マップを使用してアウト バウンド方向にのみ適用できます。パスとして前に付加が機能しない iBGP セッションで。

AS パス Prepend 機能は、次のように変更を有効に。

プリペンド	ルート マップと一致するルートの AS パスに、指定した AS 番号を付加します。 （注） • 1 個以上の AS 番号を設定できます。 • 4 バイト番号がサポートされています。 • 合計を prepend は 32 の AS 番号。AS 番号は、AS パス アトリビュートに挿入されます順序を指定する必要があります。
Prepend-最後-として	最後の前に付加 AS パス 1 から 10 までの範囲に番号として。

次の表では、AS パス Prepend の実装の選択基準について説明します。

プリペンド	1	指定された AS 番号を追加します。
Prepend-最後-として	2	最後の AS 番号を AS パスに付加します。
デフォルト	Prepend(1)	指定された AS 番号を追加します。

設定の AS パス Prepend GUI を使用して

始める前に

構成済みのテナント

手順

ステップ 1 ログインし、APIC GUI に、メニュー バーで、をクリックして **テナント** > **<Your_Tenant>** > **ネットワー キング** > **外部ルーテッド ネットワーク** > **ルート マップの設定のルール** を右クリックし、 **設定ルールの A ルート マップの作成** .</Your_Tenant>

ステップ 2 **設定ルールの A ルート マップの作成** ダイアログボックス、次のタスクを実行します。

a) [Name] フィールドに、名前を入力します
b) をクリックして、 **AS パスの設定** を開くにアイコン、 **設定 AS パスの作成** ダイアログボックス。

ステップ 3 条件を選択 **Prepend AS** に番号に付加します。

ステップ 4 AS 番号とその順序を入力し、クリックして **更新** 。複数の AS 番号の先頭を追加する必要があるかどうかを繰り返します。

ステップ 5 条件を選択 **Prepend 最後 AS** に指定された回数を番号と最後を付加します。

ステップ 6 **[カウント]**(1-10) を入力します。

ステップ 7 **設定ルールの A ルート マップの作成** 表示、AS パスに基づいて、ルールにリストされている条件を確認し、をクリックして **終了します。**

ステップ 8 APIC GUI メニュー バーで、をクリックして **テナント** > **<Your_Tenant>** > **ネットワー キング** > **外部ルーテッド ネットワーク** > **ルート マップのルールの設定** し、お客様のプロファイルを右クリックします</Your_Tenant>。

ステップ 9 確認、 **AS パスの設定** 画面の下部の値します。

NX-OS スタイルの CLI を使用した AS パスのプリペンド

このセクションでは、NX-OS スタイル コマンドライン インターフェイス (CLI) を使用して、AS パスのプリペンド機能を実現する方法について説明します。

始める前に

構成済みのテナント

手順

境界ゲートウェイ プロトコル (BGP) ルートの自動システムパス (AS パス) を変更するには、`set as-path` コマンドを使用します。set as-path コマンドは、
`apic1(config-leaf-vrf-template-route-profile)# set as-path {'prepend as-num [ ,... as-num ] | prepend-last-as num}` の形式で実行します。

例：

```
apic1(config)# leaf 103
apic1(config-leaf)# vrf context tenant t1 vrf v1
apic1(config-leaf-vrf)# template route-profile rp1
apic1(config-leaf-vrf-template-route-profile)# set as-path ?
prepend Prepend to the AS-Path
prepend-last-as Prepend last AS to the as-path
apic1(config-leaf-vrf-template-route-profile)# set as-path prepend 100, 101, 102, 103
apic1(config-leaf-vrf-template-route-profile)# set as-path prepend-last-as 8
apic1(config-leaf-vrf-template-route-profile)# exit
apic1(config-leaf-vrf)# exit
apic1(config-leaf)# exit
```

次のタスク

AS パスのプリペンドを無効にするには、示されているコマンドの no 形式を使用します:

```
apic1(config-leaf-vrf-template-route-profile)# [no] set
as-path { prepend as-num [ ,... as-num ] | prepend-last-as num}
```

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>
```

BGP 外部ルーテッド ネットワークと AS オーバーライド

BGP 自律システムのオーバーライドについて

BGP のループ防止は、自律システム パスの自律システム番号を確認することで行われます。受信側のルータが受信した BGP パケットの自律システム パスで独自の自律システム番号が表示される場合、パケットは廃棄されます。受信側のルータでは、パケットが独自の自律システムから発信され、最初に発信元から同じ場所に達したことが想定されます。この設定では、ルーティング ループが発生しないようにするためのデフォルトです。

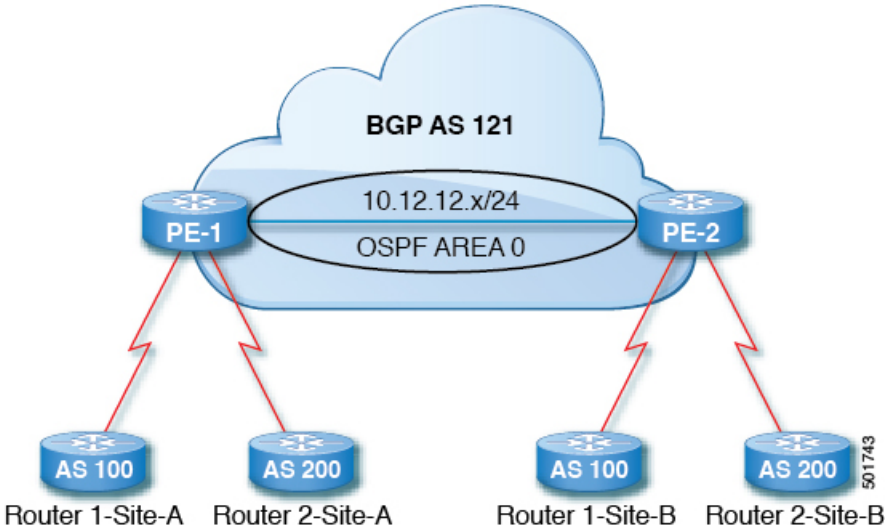
別の自立システム番号によりリンクする同一の自律システム番号を持つさまざまなサイトや禁止ユーザーのサイトを使用する場合、デフォルトルートのループが発生しないようにする設定によって問題が発生する可能性があります。このようなシナリオでは、その他のサイトが受信した場合 1 つのサイトからのルーティング更新は廃棄されます。

そのような状況が発生しないようにするには、BGP 自律システム オーバーライド機能を有効にしてデフォルト設定を上書きします。同時に、ピア AS チェックの無効化も有効にする必要があります。

自律システム オーバライド機能では、発信元のルータからの自律システム番号を、アウトバウンド ルートの AS パスの BGP ルータ送信の自律システム番号に置き換えます。アドレス ファミリごとにこの機能を有効にできます（IPv4 または IPv6）。

自律システム オーバライド機能は、GOLF レイヤ 3 設定および非 GOLF レイヤ 3 の設定でサポートされています。

図 *1:* 自律システム オーバライド機能を説明するトポロジ例



ルータ 1 およびルータ 2 は、複数のサイトを持つ 2 つの顧客です（サイト A とサイト B）。顧客ルータ 1 は AS 100 で動作し、顧客ルータ 2 は AS 200 で動作します。

上の図は、次のような自律システム（AS）オーバーライド プロセスを示しています。

1. ルータ A サイト 1 では、AS100 でルート 10.3.3.3 をアドバタイズします。
2. ルータ PE-1 は、AS100 として PE2 へ内部ルートとして反映します。
3. ルータ PE-2 は AS121 で 10.3.3.3 をプリペンドし（AS パスの 100 を 121 に置き換えます）、プレフィックスをプロパゲートします。
4. ルータ 2 サイト B は 10.3.3.3 更新プログラムを承認します。

GUI を使用して、BGP 外部ルーテッド ネットワークと有効になっている自律システム オーバーライドを設定する

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ます。
- 非 GOLF 設定の外部ルーテッド ネットワーク、論理ノード プロファイル、および BGP ピア接続プロファイルが作成されています。

手順

ステップ 1 メニュー バーで、 **Tenants** > ***Tenant_name*** > **Networking** > **External Routed Network** > ***Non-GOLF Layer 3 Out_name*** > **Logical Node Profiles** を選択します。

ステップ 2 **Navigation** ウィンドウで、適切な **BGP Peer Connectivity Profile** を選択します。

ステップ 3 **Work** ウィンドウの **Properties (BGP Peer Connectivity Profile** のもの) の下、**BGP Controls** フィールドで、次の手順を実行します:

a) **AS override** フィールドのチェック ボックスをオンにして、**Autonomous System override** 機能を有効にします。

b) **Disable Peer AS Check** フィールドのチェック ボックスをオンにします。

（注） AS オーバーライド機能を有効にするには、 **AS override** および **Disable Peer AS Check** チェック ボックスをオンにする必要があります。

c) 必要に応じてその他のフィールドを選択します。

ステップ 4 [送信（Submit）] をクリックします。

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

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。

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" />
           -->
           <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"
 />
           <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>
```

（注） 太字の例では、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。

VRF ごと、ノード BGP ごとのタイマーの値の設定

ノード BGP タイマー値ごとの各 VRF

この機能を紹介する前に、特定の VRF について、すべてのノードには同じ BGP タイマーの値が使用されます。

ノード BGP タイマー値ごとの各 VRF 機能の導入により、BGP タイマーを定義し、各ノードベースの VRF ごとに関連付けることが可能です。ノードでは複数の VRF を所持することが可能で、それぞれ、`fvCtx` に対応しています。ノード設定（`l3extLNodeP`）には、BGP プロトコル プロファイル（`bgpProtP`）の設定が含まれており、希望の BGP コンテキスト ポリシーを参照します（`bgpCtxPol`）。これにより、同じ VRF 内のさまざまなノードが異なる BGP タイマーの値を含めることが可能になります。

各 VRF ではノードに `bgpDom` の具体的な MO を含みます。その名前（プライマリ キー）は、VRF `<fvTenant>:<fvCtx>` です。属性として BGP タイマーの値が含まれています（例：`holdIntvl`、`kaIntvl`、`maxAsLimit`）。

有効なレイヤ 3 アウト設定を作成するために必要なすべての手順は、ノード BPG タイマーごとの各 VRF に正常に適用する必要があります。たとえば、次のような MO は必須です：`fvTenant`、`fvCtx`、`l3extOut`、`l3extInstP`、`LNodeP`、`bgpRR`。

ノードでは、BGP タイマー ポリシーは次のアルゴリズムに基づいて選択されます。

- `BgpProtP` が指定されると、`bgpProtP` の下で参照される `bgpCtxPol` を使用します。
- それ以外の場合、指定されると対応する `fvCtx` の下で参照される `bgpCtxPol` を使用します。
- それ以外の場合、指定されるとテナントでデフォルト ポリシーを使用します。例：`uni/tn-<tenant>/bgpCtxP-default`。
- それ以外の場合、テナント `common` の下の `default` ポリシーを使用します。例：`uni/tn-common/bgpCtxP-default`。これはプログラム済みです。

設定の高度な GUI を使用して BGP タイマーのノードごとの VRF あたり

BGP タイマーが特定のノードに設定されているときに、ノードで BGP タイマー ポリシーを使用し、VRF に関連付けられている BGP ポリシー タイマーはすべて無視されます。

始める前に

テナントと VRF はすでに設定されています。

手順

ステップ 1 メニュー バーで、次のように選択します。 **テナント** > ***Tenant_name*** > **ネットワー キング** > **プロトコル ポリシー** 。**ナビゲーション**] ペインで、[展開 **ネットワー キング** > **プロトコル ポリシー** > **BGP** > **BGP タイマー** 。

ステップ 2 **BGP Timers Policy** ダイアログボックスで、次の操作を実行します:

a) **Name** フィールドに、BGP タイマー ポリシーの名前を入力します。
b) 使用可能なフィールドには、必要に応じて、適切な値を選択します。[Submit] をクリックします。`

BGP タイマー ポリシーが作成されます。

ステップ 3 移動し、 **外部ルーテッド ネットワーク** 、し、次のアクションを実行して有効になっている BGP を使用したレイヤ 3 Out 作成します。

a) [Create Routed Outside] を右クリックします。
b) **Create Routed Outside** ダイアログボックスで、Layer 3 Out の名前を指定します。
c) チェック ボックスをオンにして、**BGP** を有効にします。
d) [Nodes and Interfaces Protocol Policies] を展開します。

ステップ 4 新しいノードを作成するには、**Create Node Profile** ダイアログボックスで、次の操作を実行します:

a) [Name] フィールドに、ノード プロファイルの名前を入力します

b) **BGP タイマー**]フィールドに、ドロップダウンリストから、この特定のノードに関連付ける BGP タイマー ポリシー選択します。[Finish] をクリックします。

特定の BGP タイマー ポリシーは、ノードに適用されます。

（注） BGP タイマー ポリシーと、既存のノードのプロファイルを関連付ける、ノードのプロファイルを右クリックし、タイマー ポリシーを関連付けます。

タイマー ポリシーが具体的に選択していない場合、 **BGP タイマー** されたノードのプロファイルが存在する自動的に VRF に関連付けられている BGP タイマー ポリシーは、このノードに適用を取得し、ノードのフィールドします。

ステップ 5 設定を確認するには、**Navigation** ウィンドウで、次の手順を実行します:

a) 展開 **レイヤ 3 Out** > **外部ルーテッド** *Network_name* > **論理ノード プロファイル** > **論理ノード** *Profiles_name* > **BGP プロトコル プロファイル。**>

b) **作業**]ペインで、ノードのプロファイルに関連付けられている BGP プロトコル プロファイルが表示されます。

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` (`l3extLNodeP` の下) を設定します。`bgpProtP` の下で、目的とする関係 (`bgpRsBgpNodeCtxPol`) を設定します。これは、BGP コンテキスト ポリシー (`bgpCtxPol`) に対するものです。

手順

`node1` でノード固有の BGP タイマー ポリシーを設定し、`node2` を、ノード固有ではない BGP タイマー ポリシーで設定します。

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="protp1" >
            <bgpRsBgpNodeCtxPol tnBgpCtxPolName="pol2" />
        </bgpProtP>
      </l3extLNodeP>
      <l3extLNodeP name="node2" >
      </l3extLNodeP>
```

この例では、node1 は BGP タイマー値をポリシー pol2 から取得し、node2 は BGP タイマー値を pol1 から取得します。タイマー値は bgpDom に適用されますが、これは VRF tn1:ctx1 に対応しています。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="protp1" status="deleted" >
            <bgpRsBgpNodeCtxPol tnBgpCtxPolName="pol2" />
        </bgpProtP>
      </l3extLNodeP>
      <l3extLNodeP name="node2" >
      </l3extLNodeP>
```

上の例のコード フレーズ <bgpProtP name="protp1" status="deleted" > は、BGP タイマー ポリシーを削除します。削除後、node1 が node1 が関連付けられている VRF の BGP タイマー ポリシーのデフォルト設定になります。上の例では pol1 です。

NX-OS スタイル CLI を使用してノード BGP タイマー ポリシーあたりの VRF あたりを設定する

手順

	コマンドまたはアクション	目的

ステップ 1	タイマーポリシーを作成する前に、BGP ASN およびルート リフレクタを設定します。
	例：
	`apic1(config)#`
	`apic1(config)# bgp-fabric`
	`apic1(config-bgp-fabric)#`
	`route-reflector spine 102`
	`apic1(config-bgp-fabric)# asn 42`
	`apic1(config-bgp-fabric)# exit`
	`apic1(config)# exit`
	`apic1#`	

ステップ 2	タイマー ポリシーを作成します。
	例：
	`apic1# config`
	`apic1(config)# leaf 101`
	`apic1(config-leaf)# template bgp timers pol7 tenant tn1`
	`This template will be available on all nodes where tenant tn1 has a VRF deployment`
	`apic1(config-bgp-timers)# timers bgp 120 240`
	`apic1(config-bgp-timers)# graceful-restart stalepath-time 500`
	`apic1(config-bgp-timers)# maxas-limit 300`
	`apic1(config-bgp-timers)# exit`
	`apic1(config-leaf)# exit`
	`apic1(config)# exit`
	`apic1#`
		特定の値は、例としてのみ提供されます。

ステップ 3	設定された BGP ポリシーを表示します。
	例：
	`apic1# show run leaf 101 template bgp timers pol7`
	`# Command: show running-config leaf 101 template bgp timers pol7`
	`leaf 101`
	`template bgp timers pol7 tenant tn1`
	`timers bgp 120 240`
	`graceful-restart stalepath-time 500`	

	コマンドまたはアクション	目的
	maxas-limit 300 exit exit	
ステップ 4	ノードで特定のポリシーを参照します。 **例：** apic1# **config** apic1(config)# **leaf 101** apic1(config-leaf)# **router bgp 42** apic1(config-leaf-bgp)# **vrf member tenant tn1 vrf ctx1** apic1(config-leaf-bgp-vrf)# **inherit node-only bgp timer pol7** apic1(config-leaf-bgp-vrf)# **exit** apic1(config-leaf-bgp)# **exit** apic1(config-leaf)# **exit** apic1(config)# **exit** apic1#	
ステップ 5	特定の BGP のタイマー ポリシーのノードが表示されます。 **例：** apic1# **show run leaf 101 router bgp 42 vrf member tenant tn1 vrf ctx1** # Command: show running-config leaf 101 router bgp 42 vrf member tenant tn1 vrf ctx1 leaf 101 router bgp 42 vrf member tenant tn1 vrf ctx1 inherit node-only bgp timer pol7 exit exit exit apic1#	

不整合や障害のトラブルシューティング

特定の状況下では、次のような不整合や障害が発生する可能性があります:

異なるレイヤ 3 Out (`l3Out`) が同じ VRF (`fvCtx`) に関連付けられているか、同じノードで `bgpProtP` が異なるポリシー (`bgpCtxPol`) に関連付けられていると、障害が発生します。次の例では、同じ Layer 3 Out (`out1` と `out2`) が同じ VRF (`ctx1`) に関連付けられています。`out1` の下では、`node1` は BGP タイマー プロトコル `pol1` に関連付けられており、`out2` の下では、`node1` は別の BGP タイマー プロトコル `pol2` に関連付けられています。。この場合、障害が発生します。

```
tn1
  ctx1
  out1
    ctx1
```

```
    node1
      protp pol1

  out2
    ctx1
    node1
      protp pol2
```

このような障害が発生した場合は、設定を変更して、BGP タイマー ポリシー間の競合を削除してください。

BFD サポートの設定

双方向フォワーディング検出

双方向フォワーディング検出（BFD）を使用して、ピアリング ルータの接続をサポートするように設定された ACI ファブリック境界リーフ スイッチ間の転送パスのサブセカンド障害検出時間を提供します。

BFD は、次のような場合に特に役立ちます。

- ルータ同士の間に直接的な接続がない場合に、レイヤ 2 デバイスまたはレイヤ 2 クラウド経由でピアリング ルータが接続されているとき。転送パスに障害があっても、ピア ルータにはそれがわからない可能性があります。プロトコルの制御に利用できるメカニズムは hello タイムアウトだけですが、タイムアウトまでには数十秒、さらには数分の時間がかかる場合があります。BFD では、障害を 1 秒未満で検出することが可能です。
- 信頼できる障害検出に非対応の物理メディア（共有イーサネットなど）経由でピアリング ルータが接続されているとき。この場合も、ルーティング プロトコルは、時間のかかる hello タイマーに頼るしかありません。
- 1 組のルータの間で多くのプロトコルが実行されているとき、各プロトコルは、独自のタイムアウトでリンク障害を検出する独自の hello メカニズムを持っています。BFD は、すべてのプロトコルに均一のタイムアウトを指定し、それによってコンバージェンス時間の一貫性を保ち、予測可能にします。

次に示す BFD の設定のガイドラインおよび制限事項に従ってください。

- APIC リリース 3.1 (1) 以降、リーフおよびスパイン スイッチ間の BFD は IS-IS のファブリック インターフェイスでサポートされています。さらに、スパイン スイッチの BFD 機能は、OSPF ルートとスタティック ルートでサポートされます。
- BFD は -EX および -FX ライン カード（または新しいバージョン）のモジュラ スパイン スイッチでサポートされ、また BFD は Nexus 9364C 非モジュラ スパイン スイッチ（または新しいバージョン）でサポートされます。
- VPC ピア間の BFD はサポートされません。
- マルチホップ BFD はサポートされません。
- ループバック アドレス ピアでの iBGP 上の BFD はサポートされません。

- インターフェイス ポリシーで BFD サブインターフェイス最適化を有効化できます。このフラグを 1 つのサブインターフェイスに立てることにより、その物理インターフェイス上のすべてのサブインターフェイスの最適化が有効になります。
- BGP プレフィクス ピアの BFD はサポートされません。

（注） Cisco ACI は、IP フラグメンテーションをサポートしていません。したがって、外部ルータへのレイヤ 3 Outside（L3Out）接続、または Inter-Pod Network（IPN）を介した multipod 接続を設定する場合は、MTU が両側で適切に設定されていることが重要です。ACI、Cisco NX-OS、Cisco IOS などの一部のプラットフォームでは、設定された MTU 値は IP ヘッダーを考慮に入れています（結果として、最大パケット サイズは、ACI で 9216 バイト、NX-OS および IOS で 9000 バイトに設定されます）。ただし、IOS XR などの他のプラットフォームは、パケット ヘッダーのを除く MTU 値を設定します（結果として最大パケット サイズは 8986 バイトになります）。

各プラットフォームの適切な MTU 値については、それぞれの設定ガイドを参照してください。

CLI ベースのコマンドを使用して MTU をテストすることを強く推奨します。たとえば、Cisco NX-OS CLI で `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1` などのコマンドを使用します。

GUI を使用してリーフ スイッチの BFD をグローバルに設定する

手順

ステップ 1 メニュー バーで、**[Fabric]** > **[Access Policies]** の順に選択します。

ステップ 2 **Navigation** うぃんどうで、 **Switch Policies** > **Policies** > **BFD** を展開します。
設定を双方向フォワーディング検出 (BFD) には、使用可能な 2 つの種類があります:

- BFD IPV4
- BFD IPV6

これらの BFD 設定ごとに、デフォルト ポリシーを使用するか、特定のスイッチ(またはスイッチのセット) 用に新しいポリシーを作成できます。

（注） デフォルトでは、APIC コントローラはシステムの起動時にデフォルトのポリシーを作成します。これらのデフォルト ポリシーはグローバルなもので、双方向転送検出 (BFD) の設定ポリシーです。デフォルト グローバルポリシー内の属性は、**作業**ウィンドウで設定できます。または、これらデフォルトのポリシーの値を変更することもできます。ただし、いったんデフォルトのグローバルポリシーを変更すると、システム全体(すべてのスイッチ)に影響することに注意してください。デフォルトではありませんが、特定のスイッチ(またはスイッチの設定)の特定の設定を使用する場合は、次の手順の説明に従って、スイッチのプロファイルを作成します。

ステップ 3 デフォルトではない特定のBFDポリシーのスイッチプロファイルを作成するには、**Navigation** ウィンドウで、 **Switch Policies** > **Profiles** > **Leaf Profiles** を選択します
Work ウィンドウに **[Profiles - Leaf Profiles]** 画面が表示されます。

ステップ 4 **Work** ウィンドウの右側の、**[ACTIONS]** の下で、**[Create Leaf Profile]** を選択します。
[Create Leaf Profile] ダイアログボックスが表示されます。

ステップ 5 **Create Leaf Profile** ダイアログボックスで、次の操作を実行します:

a) **Name** フィールドに、リーフ スイッチ プロファイルの名前を入力します

b) **Description** フィールドの隣にに、プロファイルの説明を入力します。（この手順は任意です）。

c) **Switch Selectors** フィールドで、**[Name]** (スイッチの名前)、**[Blocks]** (スイッチの選択)、および **[Policy Group]** (**[Create Access Switch Policy Group]** の選択) に適切な値を入力します。
Create Access Switch Policy Group ダイアログボックスが表示されます。ここでは、ポリシー グループの識別プロパティを指定できます。

ステップ 6 **Create Access Switch Policy Group** ダイアログボックスで、次の操作を実行します:

a) [Name] フィールドにポリシー グループの名前を入力します。

b) **Description** フィールドに、ポリシーの説明を入力します。（この手順はオプションです）。

c) BFD ポリシー タイプ (**BFD IPV4 Policy** または **BFD IPV6 Policy**) を選択し、値 (**default** または **Create BFD Global Ipv4 Policy**) を特定のスイッチまたはスイッチのセットに対して選択します。

ステップ 7 **Submit** をクリックします。
BFD グローバルポリシーを作成するもう 1 つの方法は、**BFD IPV4** または **BFD IPV6** のいずれかを右クリックします (**Navigation** ウィンドウにあります)。

ステップ 8 作成した BFD グローバル設定を表示するには、**Navigation** ウィンドウで、 **Switch Policies** > **Policies** > **BFD** を展開します。

GUI を使用してスパイン スイッチで BFD のグローバル設定

手順

ステップ 1 メニュー バーで、**[Fabric]** > **[Access Policies]** の順に選択します。

ステップ 2 **Navigation** うぃんどうで、 **Switch Policies** > **Policies** > **BFD** を展開します。
設定を双方向フォワーディング検出 (BFD) には、使用可能な 2 つの種類があります:

- BFD IPV4
- BFD IPV6

これらのBFD設定ごとに、デフォルトポリシーを使用するか、特定のスイッチ(またはスイッチのセット) 用に新しいポリシーを作成できます。

（注） デフォルトでは、APIC コントローラはシステムの起動時にデフォルトのポリシーを作成します。これらのデフォルト ポリシーはグローバルなもので、双方向転送検出(BFD)の設定ポリシーです。デフォルト グローバルポリシー内の属性は、**作業**ウィンドウで設定できます。または、これらデフォルトのポリシーの値を変更することもできます。ただし、いったんデフォルトのグローバルポリシーを変更すると、システム全体(すべてのスイッチ)に影響することに注意してください。デフォルトではありませんが、特定のスイッチ(またはスイッチの設定)の特定の設定を使用する場合は、次の手順の説明に従って、スイッチのプロファイルを作成します。

ステップ 3 特定グローバル BFD ポリシー(これは、デフォルトではありません) でのスパイン スイッチ プロファイルを作成する、 **ナビゲーション**] ペインで、展開、 **スイッチ ポリシー** > **プロファイル** > **スパイン プロファイル** 。
Profiles- Spine Profiles 画面が **Work** ウィンドウに表示されます。

ステップ 4 右側にある、 **作業** ()] ペインで、 **アクション** 、select **スパイン プロファイルの作成** します。
Create Spine Profile ダイアログボックスが表示されます。

ステップ 5 **Create Spine Profile** ダイアログボックスで、次の操作を実行します:

a) **Name** フィールドに、スイッチ プロファイルの名前を入力します。
b) **Description** フィールドの隣にに、プロファイルの説明を入力します。（この手順は任意です）。
c) **スパイン セレクタ** フィールドで、適切な値を入力 **名** (スイッチの名前)、 **ブロック** (スイッチを選択し、) および **ポリシー グループ** (選択 **スパイン スイッチ ポリシー グループの作成**)。
スパイン スイッチ ポリシー グループの作成 ダイアログボックスはポリシー グループ id のプロパティを指定できますが表示されます。

ステップ 6 **Create スパイン スイッチ Policy Group** ダイアログボックスで、次の操作を実行します:

a) [Name] フィールドにポリシー グループの名前を入力します。
b) **Description** フィールドに、ポリシーの説明を入力します。（この手順はオプションです）。
c) BFD ポリシー タイプ (**BFD IPV4 Policy** または **BFD IPV6 Policy**) を選択し、値 (**default** または **Create BFD Global Ipv4 Policy**) を特定のスイッチまたはスイッチのセットに対して選択します。

ステップ 7 **Submit** をクリックします。
BFD グローバルポリシーを作成するもう 1 つの方法は、**BFD IPV4** または **BFD IPV6** のいずれかを右クリックします (**Navigation** ウィンドウにあります)。

ステップ 8 作成した BFD グローバル設定を表示するには、**Navigation** ウィンドウで、 **Switch Policies** > **Policies** > **BFD** を展開します。

NX-OS スタイル CLI を使用したリーフ スイッチでの BFD のグローバルな設定

手順

ステップ 1 NX-OS CLI を使用して BFD IPV4 グローバル設定（bfdIpv4InstPol）を設定するには：

例：

```
apic1# configure
apic1(config)# template bfd ip bfd_ipv4_global_policy
apic1(config-bfd)# [no] echo-address 1.2.3.4
apic1(config-bfd)# [no] slow-timer 2500
apic1(config-bfd)# [no] min-tx 100
apic1(config-bfd)# [no] min-rx 70
apic1(config-bfd)# [no] multiplier 3
apic1(config-bfd)# [no] echo-rx-interval 500
apic1(config-bfd)# exit
```

ステップ 2 NX-OS CLI を使用して BFD IPV6 グローバル設定（bfdIpv6InstPol）を設定するには：

例：

```
apic1# configure
apic1(config)# template bfd ipv6 bfd_ipv6_global_policy
apic1(config-bfd)# [no] echo-address 34::1/64
apic1(config-bfd)# [no] slow-timer 2500
apic1(config-bfd)# [no] min-tx 100
apic1(config-bfd)# [no] min-rx 70
apic1(config-bfd)# [no] multiplier 3
apic1(config-bfd)# [no] echo-rx-interval 500
apic1(config-bfd)# exit
```

ステップ 3 NX-OS CLI を使用してアクセス リーフ ポリシー グループ (infraAccNodePGrp) を設定し、以前に作成した BFD グローバル ポリシーを継承するには:

例：

```
apic1# configure
apic1(config)# template leaf-policy-group test_leaf_policy_group
apic1(config-leaf-policy-group)# [no] inherit bfd ip bfd_ipv4_global_policy
apic1(config-leaf-policy-group)# [no] inherit bfd ipv6 bfd_ipv6_global_policy
apic1(config-leaf-policy-group)# exit
```

ステップ 4 NX-OS CLI を使用して以前に作成したリーフ ポリシー グループを リーフに関連付けるには:

例：

```
apic1(config)# leaf-profile test_leaf_profile
apic1(config-leaf-profile)# leaf-group test_leaf_group
apic1(config-leaf-group)# leaf-policy-group test_leaf_policy_group
apic1(config-leaf-group)# leaf 101-102
apic1(config-leaf-group)# exit
```

NX-OS スタイル CLI を使用したスパイン スイッチ上の BFD のグローバル設定

次の手順を使用して、NX-OS スタイル CLI を使用してスパイン スイッチの BFD をグローバルに設定します。

手順

ステップ 1 NX-OS CLI を使用して BFD IPV4 グローバル設定（bfdIpv4InstPol）を設定するには：

例：

```
apic1# configure
apic1(config)# template bfd ip bfd_ipv4_global_policy
apic1(config-bfd)# [no] echo-address 1.2.3.4
apic1(config-bfd)# [no] slow-timer 2500
apic1(config-bfd)# [no] min-tx 100
apic1(config-bfd)# [no] min-rx 70
apic1(config-bfd)# [no] multiplier 3
apic1(config-bfd)# [no] echo-rx-interval 500
apic1(config-bfd)# exit
```

ステップ 2 NX-OS CLI を使用して BFD IPV6 グローバル設定（bfdIpv6InstPol）を設定するには：

例：

```
apic1# configure
apic1(config)# template bfd ipv6 bfd_ipv6_global_policy
apic1(config-bfd)# [no] echo-address 34::1/64
apic1(config-bfd)# [no] slow-timer 2500
apic1(config-bfd)# [no] min-tx 100
apic1(config-bfd)# [no] min-rx 70
apic1(config-bfd)# [no] multiplier 3
apic1(config-bfd)# [no] echo-rx-interval 500
apic1(config-bfd)# exit
```

ステップ 3 NX-OS CLI を使用してスパイン ポリシー グループを設定し以前作成した BFD グローバル ポリシーを継承するには：

例：

```
apic1# configure
apic1(config)# template spine-policy-group test_spine_policy_group
apic1(config-spine-policy-group)# [no] inherit bfd ip bfd_ipv4_global_policy
apic1(config-spine-policy-group)# [no] inherit bfd ipv6 bfd_ipv6_global_policy
apic1(config-spine-policy-group)# exit
```

ステップ 4 NX-OS を使用して以前作成したスパイン ポリシー グループをスパイン スイッチに関連付けるには；

例：

```
apic1# configure
apic1(config)# spine-profile test_spine_profile
apic1(config-spine-profile)# spine-group test_spine_group
apic1(config-spine-group)# spine-policy-group test_spine_policy_group
```

```
apic1(config-spine-group)# spine 103-104
apic1(config-leaf-group)# exit
```

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>
```

GUI 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

明示的な双方向フォワーディング検出 (BFD) を設定できる、3 つのサポート対象のインターフェイス（ルーテッド L3 インターフェイス、外部インターフェイス SVI とルーテッド サブインターフェイス）があります。グローバル コンフィギュレーションを使用しないで、さらに特定のインターフェイスの明示的な設定をしたい場合、特定のスイッチまたは一連のすべてのインターフェイスに適用される独自のグローバル コンフィギュレーションを作成できます。特定のインターフェイス上の特定のスイッチの粒度がさらに必要な場合、このインターフェイス オーバーライド設定を使用する必要があります。

始める前に

テナントはすでに作成されています。

手順

ステップ 1 メニュー バーで、 **Tenant** を選択します。

ステップ 2 **ナビゲーション** ウィンドウ ([クイック スタート)、作成したテナントの展開 ***Tenant_name*** > **ネットワー キング** > **外部ルーテッド ネットワーク** 。

ステップ 3 **External Routed Networks** を右クリックし、**Create Routed Outside** を選択します。
[Create Routed Outside] ダイアログボックスが表示されます。

ステップ 4 **外部ルーティング作成** ダイアログボックス未満で **ルーティング外部定義** 、既存の設定はすでにセットアップする必要があります。それ以外の場合は、外部ルーティングの設定のアイデンティティを定義する値を入力します。

ステップ 5 [**ノードとインターフェイスのプロトコル プロファイル** の下部にある、 **外部ルーティングの作成** ダイアログボックス] をクリックして、「 + 」(expand)] ボタン。
Create Node Profile ダイアログボックスが表示されます。

ステップ 6 **ノードのプロファイルを指定** 、ノードのプロファイルの名前を入力、 **名** フィールド。

ステップ 7 をクリックして、「 + 」(expand) の右側にあるボタン、 **ノード** フィールド。
Select Node ダイアログボックスが表示されます。

ステップ 8 **ノードとスタティック ルートの設定を選択** でノードを選択、 **ノード ID** フィールド。

ステップ 9 **Router ID** フィールドにルータ ID を入力します。

ステップ 10 [OK] をクリックします。
Create Node Profile ダイアログボックスが表示されます。

ステップ 11 をクリックします」 + 」(expand) の右側にあるボタン、 **インターフェイス プロファイル** フィールド。
Create Interface Profile ダイアログボックスが表示されます。

ステップ 12 インターフェイスプロファイル名を **Name** フィールドに入力します。

ステップ 13 インターフェイスのタブのいずれかをクリックして、以前に作成したノードの目的のユーザインターフェイスを選択します。

- **ルーテッド インターフェイス**
- **SVI**
- **Routed Sub-Interfaces**

ステップ 14 **BFD インターフェイス プロファイル** フィールドで、BDF の詳細を入力します。**認証タイプ** フィールドで、選択 **No authentication** または **キー SHA1** 。認証 (SHA1 のキーを選択) により、入力を選択すると、 **認証キー ID** を入力してください、 **の認証キーを** (パスワード)、再次を入力して、パスワードを確認 **キーの確認** 。

ステップ 15 BFD インターフェイス ポリシー] フィールドのいずれかを選択、 **一般的な/デフォルト** 設定 (デフォルト BFD policy) を選択して、自分 BFD ポリシーの作成または **BFD インターフェイス ポリシーの作成** します。
選択した場合 **BFD インターフェイス ポリシーの作成** 、 **BFD インターフェイス ポリシーの作成** BFD インターフェイス ポリシーの値を定義するダイアログボックスが表示されます。

ステップ 16 [Submit] をクリックします。

ステップ 17 BFD ポリシーのレベルに移動して、設定されているインターフェイスを表示する **ネットワーキング** > **プロトコル ポリシー** > **BFD** 。

NX-OS スタイルの CLI を使用して BFD インターフェイスのオーバーライドを設定する

手順

ステップ 1 NX-OS CLI を使用して BFD インターフェイス ポリシー (bfdIfPol) を設定するには:

例：

```
apic1# configure
apic1(config)# tenant t0
apic1(config-tenant)# vrf context v0
apic1(config-tenant-vrf)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant t0 vrf v0
apic1(config-leaf-vrf)# exit
apic1(config-leaf)# interface Ethernet 1/18
apic1(config-leaf-if)# vrf  member tenant t0 vrf v0
apic1(config-leaf-if)# exit
apic1(config-leaf)# template bfd bfdIfPol1 tenant t0
apic1(config-template-bfd-pol)# [no] echo-mode enable
apic1(config-template-bfd-pol)# [no] echo-rx-interval 500
apic1(config-template-bfd-pol)# [no] min-rx 70
apic1(config-template-bfd-pol)# [no] min-tx 100
apic1(config-template-bfd-pol)# [no] multiplier 5
apic1(config-template-bfd-pol)# [no] optimize subinterface
apic1(config-template-bfd-pol)# exit
```

ステップ 2 NX-OS CLI を使用して、以前に作成した BFD インターフェイス ポリシーを、IPv4 アドレスを持つ L3 インターフェイスに継承させるには:

例：

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface Ethernet 1/15
apic1(config-leaf-if)# bfd ip tenant mode
apic1(config-leaf-if)# bfd ip inherit interface-policy bfdPol1
apic1(config-leaf-if)# bfd ip authentication keyed-sha1 key 10 key password
```

ステップ 3 NX-OS CLI を使用して、以前に作成した BFD インターフェイス ポリシーを、IPv6 アドレスを持つ L3 インターフェイスに継承させるには:

例：

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface Ethernet 1/15
apic1(config-leaf-if)# ipv6 address 2001::10:1/64 preferred
apic1(config-leaf-if)# bfd ipv6 tenant mode
apic1(config-leaf-if)# bfd ipv6 inherit interface-policy bfdPol1
apic1(config-leaf-if)# bfd ipv6 authentication keyed-sha1 key 10 key password
```

ステップ 4 NX-OS CLI を使用して、IPv4 アドレスを持つ VLAN インターフェイス上の BFD を設定するには:

例：

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface vlan 15
apic1(config-leaf-if)# vrf member tenant t0 vrf v0
apic1(config-leaf-if)# bfd ip tenant mode
apic1(config-leaf-if)# bfd ip inherit interface-policy bfdPol1
apic1(config-leaf-if)# bfd ip authentication keyed-sha1 key 10 key password
```

ステップ 5 NX-OS CLI を使用して、IPv6 アドレスを持つ VLAN インターフェイス上の BFD を設定するには:

例:

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface vlan 15
apic1(config-leaf-if)# ipv6 address 2001::10:1/64 preferred
apic1(config-leaf-if)# vrf member tenant t0 vrf v0
apic1(config-leaf-if)# bfd ipv6 tenant mode
apic1(config-leaf-if)# bfd ipv6 inherit interface-policy bfdPol1
apic1(config-leaf-if)# bfd ipv6 authentication keyed-sha1 key 10 key password
```

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"/>

            <l3extLIfP 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>
            </l3extLIfP>


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

GUI を使用して BFD コンシューマ プロトコルを設定する

この手順では、BFD機能の消費者であるコンシューマプロトコル(OSPF、BGP、EIGRP、スタティック ルート、および IS-IS) での双方向フォワーディング検出 (BFD) を有効にする方法を説明します。これらのプロトコルで BFD を使用するには、それらのフラグを有効にする必要があります。

（注） これらの 4 つのコンシューマ プロトコルは、左側のナビゲーション ウィンドウの **Tenant** > **Networking** > **Protocol Policies** の下にあります。

始める前に

テナントはすでに作成されています。

手順

ステップ 1 メニュー バーで、 **Tenant** を選択します。

ステップ 2 BGP プロトコルの BFD を設定するには、**Navigation** ウインドウ (クイック スタートの下) で、作成したテナント、 ***Tenant_name*** > **Networking** > **Protocol Policies** > **BGP** > **BGP Peer Prefix** を展開します。

ステップ 3 **Work** ウィンドウの右側の **[ACTIONS]** の下で、**[Create BGP Peer Prefix Policy]** を選択します。**[Create BGP Peer Prefix Policy]** ダイアログボックスが表示されます。

（注） 左のナビゲーション ウィンドウで **[BGP Peer Prefix]** を右クリックして **[Create BGP Peer Prefix]** を選択し、ポリシーを作成することもできます。

ステップ 4 **[Name]** フィールドに名前を入力し、残りのフィールドに値を入力して BGP ピア プレフィックス ポリシーを定義します。

ステップ 5 **Submit** をクリックします。
作成した BGP ピア プレフィックス ポリシーは、左のナビゲーション ウィンドウの **[BGP Peer Prefix]** の下に表示されます。

ステップ 6 **Navigation** ウィンドウで、 **Networking** > **External Routed Networks** に戻ります。

ステップ 7 **External Routed Networks** を右クリックし、**Create Routed Outside** を選択します。
[Create Routed Outside] ダイアログボックスが表示されます。

ステップ 8 **Create Routed Outside** ダイアログボックスで、**[Name]** フィールドに名前を入力します。**[Name]** フィールドの右側で、**[BGP]** プロトコルを選択します。

ステップ 9 **[Nodes and Interfaces Protocol Profiles]** セクションで、[+] (展開) ボタンをクリックします。
[Create Node Profile] ダイアログボックスが表示されます。

ステップ 10 **BGP ピア接続** セクションで、をクリックします」 ＋ 」(expand)] ボタン。
[Create BGP Peer Connectivity Profile] ダイアログボックスが表示されます。

ステップ 11 **[Create BGP Peer Connectivity Profile]** ダイアログボックスの **[Peer Controls]** の隣で、**[Bidirectional Forwarding Detection]** を選択して BGP コンシューマ プロトコルの BFD を、次のように有効にします (またはオフにして BFD を無効にします)。

ステップ 12 OSPF プロトコルの BFD を設定するには、**Navigation** ウィンドウで、 **Networking** > **Protocol Policies** > **OSPF** > **OSPF Interface** に移動します。

ステップ 13 **Work** ウィンドウの右側の、**[ACTIONS]** の下で、**[Create OSPF Interface Policy]** を選択します。
[Create OSPF Interface Policy] ダイアログボックスが表示されます。

（注） 左のナビゲーション ウィンドウで **[OSPF Interface]** を右クリックして **[Create OSPF Interface Policy]** を選択し、ポリシーを作成することもできます。

ステップ 14 **[Name]** フィールドに名前を入力し、残りのフィールドに値を入力して OSPF インターフェイス ポリシーを定義します。

ステップ 15 このダイアログボックスの **[Interface Controls]** セクションでは、BFD の有効と無効を切り替えることができます。有効にするには、図のように、**[BFD]** の隣のボックスをオンにして OSPF コンシューマ プロトコルにフラグを追加します (またはボックスをオフにして BFD を無効にします)。

ステップ 16 **Submit** をクリックします。

ステップ 17 EIGRP プロトコルの BFD を設定するには、**Navigation** ウィンドウで、 **Networking** > **Protocol Policies** > **EIGRP** > **EIGRP Interface** に移動します。

ステップ 18 **Work** ウィンドウの右側の、**[ACTIONS]** の下で、**[Create EIGRP Interface Policy]** を選択します。
[Create EIGRP Interface Policy] ダイアログボックスが表示されます。

（注） 左のナビゲーション ウィンドウで **[EIGRP Interface]** を右クリックして **[Create EIGRP Interface Policy]** を選択し、ポリシーを作成することもできます。

ステップ 19 **[Name]** フィールドに名前を入力し、残りのフィールドに値を入力して OSPF インターフェイス ポリシーを定義します。

ステップ 20 このダイアログボックスの **[Control State]** セクションでは、BFD の有効と無効を切り替えることができます。有効にするには、図のように、**[BFD]** の隣のボックスをオンにして EIGRP コンシューマ プロトコルにフラグを追加します (またはボックスをオフにして BFD を無効にします)。

ステップ 21 **Submit** をクリックします。

ステップ 22 スタティック ルート プロトコルで BFD を設定するには、**Navigation** ウィンドウで、**Networking** > **External Routed Networks** > に戻ります。

ステップ 23 **External Routed Networks** を右クリックし、**Create Routed Outside** を選択します。
[Create Routed Outside] ダイアログボックスが表示されます。

ステップ 24 **[Define the Routed Outside]** セクションで、すべての必須フィールドに値を入力します。

ステップ 25 **[Nodes and Interfaces Protocol Profiles]** セクションで、[+] (展開) ボタンをクリックします。
[Create Node Profile] ダイアログボックスが表示されます。

ステップ 26 **[Nodes]** セクションで、[+] (展開) ボタンをクリックします。

[Select Node] ダイアログボックスが表示されます。

ステップ 27 **[Static Routes]** セクションで、[+] (展開) ボタンをクリックします。
[Create Static Route] ダイアログボックスが表示されます。このセクションで、必要なフィールドの値を入力します。

ステップ 28 **[Route Control]** の隣で、**[BFD]** の隣のボックスをオンにして有効にします (または、無効にする場合にはオフにします)。

ステップ 29 **[OK]** をクリックします。

ステップ 30 IS-IS プロトコルの BFD を設定するには、**Navigation** ペインで **Fabric** > **Fabric Policies** > **Interface Policies** > **Policies** > **L3 Interface** に移動します。

ステップ 31 **Work** ウィンドウの右側の、**[ACTIONS]** の下で、**[Create L3 Interface Policy]** を選択します。
[Create L3 Interface Policy] ダイアログボックスが表示されます。

（注） 左のナビゲーション ウィンドウで **[L3 Interface]** を右クリックして **[Create EIGRP Interface Policy]** を選択し、ポリシーを作成することもできます。

ステップ 32 **[Name]** フィールドに名前を入力し、残りのフィールドに値を入力して L3 インターフェイス ポリシーを定義します。

ステップ 33 BFD ISIS ポリシーを有効にするには、 **Enable** をクリックします。

ステップ 34 [SUBMIT] をクリックします。

NX-OS スタイルの CLI 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 NX-OS は、CLI を使用して、BGP コンシューマ プロトコルを BFD をイネーブルにします。

例：

```
apic1# configure
apic1(config)# bgp-fabric
apic1(config-bgp-fabric)# asn 200
apic1(config-bgp-fabric)# exit
apic1(config)# leaf 101
apic1(config-leaf)# router bgp 200
apic1(config-bgp)# vrf member tenant t0 vrf v0
apic1(config-leaf-bgp-vrf)# neighbor 1.2.3.4
apic1(config-leaf-bgp-vrf-neighbor)# [no] bfd enable
```

ステップ 2 NX-OS は、CLI を使用して、EIGRP コンシューマ プロトコルを BFD をイネーブルにします。

例：

```
apic1(config-leaf-if)# [no] ip bfd eigrp enable
```

ステップ 3 NX-OS は、CLI を使用して、OSPF コンシューマ プロトコルを BFD をイネーブルにします。

例：

```
apic1(config-leaf-if)# [no] ip ospf bfd enable


apic1# configure
apic1(config)# spine 103
apic1(config-spine)# interface ethernet 5/3.4
apic1(config-spine-if)# [no] ip ospf bfd enable
```

ステップ 4 NX-OS は、CLI を使用して、スタティック ルート コンシューマ プロトコルを BFD をイネーブルにします。

例：

```
apic1(config-leaf-vrf)# [no] ip route 10.0.0.1/16 10.0.0.5 bfd


apic1(config)# spine 103
apic1(config-spine)# vrf context tenant infra vrf overlay-1
apic1(config-spine-vrf)# [no] ip route 21.1.1.1/32 32.1.1.1 bfd
```

ステップ 5 NX-OS は、CLI を使用して、IS-IS コンシューマ プロトコルを BFD をイネーブルにします。

例：

```
apic1(config)# leaf 101
apic1(config-spine)# interface ethernet 1/49
apic1(config-spine-if)# isis bfd enabled
apic1(config-spine-if)# exit
apic1(config-spine)# exit


apic1(config)# spine 103
apic1(config-spine)# interface ethernet 5/2
apic1(config-spine-if)# isis bfd enabled
apic1(config-spine-if)# exit
apic1(config-spine)# exit
```

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"/>

            <l3extLIfP 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>
            </l3extLIfP>


        </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>

             <l3extLIfP 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>
             </l3extLIfP>

         </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"/>

            <l3extLIfP 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>
            </l3extLIfP>


        </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>
            <l3extLIfP 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" />

            </l3extLIfP>

        </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>

             <l3extLIfP name='portIf'>
                 <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-103/pathep-[eth5/10]"
encap='vlan-4' ifInstT='sub-interface' addr="20.3.10.1/24"/>

                <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>
```

OSPF 外部ルーテッド ネットワーク

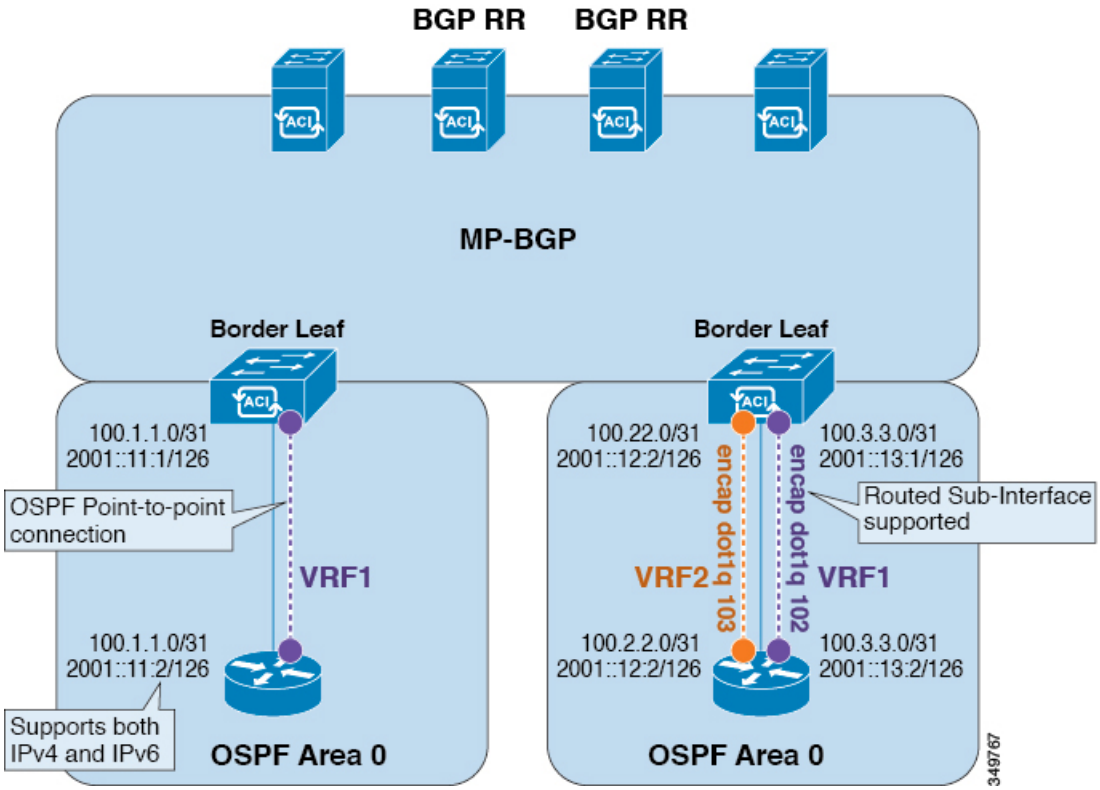
OSPF レイヤ 3 Outside 接続

OSPF レイヤ 3 Outside 接続は、標準または NSSA エリアです。バックボーン（エリア 0）エリアも、OSPF レイヤ 3 Outside 接続エリアとしてサポートされます。ACI は、IPv4 の OSPFv2 と IPv6 の OSPFv3 の両方をサポートします。OSPF レイヤ 3 Outside を作成するときに、OSPF バージョンを設定する必要はありません。インターフェイス プロファイル設定（IPv4 または IPv6 アドレッシング）に基づいて、正しい OSPF プロセスが自動的に作成されます。IPv4 と IPv6

の両方のプロトコルが同じインターフェイス（デュアル スタック）でサポートされますが、2 つの個別インターフェイス プロファイルを作成する必要があります。

レイヤ 3 Outside 接続は、ルーテッド インターフェイス、ルーテッド サブインターフェイス、および SVI でサポートされます。SVI は、L2 と L3 両方のトラフィックで物理接続を共有する必要がある場合に使用されます。SVI は、ポート、ポート チャネル、VPC ポート チャネルでサポートされます。

図 *2 : OSPF* レイヤ *3 Out* 接続



SVI がレイヤ 3 Outside 接続に使用されると、外部ブリッジ ドメインが境界リーフ スイッチに作成されます。外部ブリッジ ドメインは、ACI ファブリック上の 2 つの VPC スイッチ間の接続を可能にします。これにより、両方の VPC スイッチが、相互の、および外部 OSPF デバイスとの OSPF 隣接関係を確立できます。

ブロードキャスト ネットワークで OSPF を実行する場合、障害が発生したネイバーを検出する時間は dead 間隔（デフォルトは 40 秒）です。障害が発生した後でネイバー隣接関係を再確立する場合にも、代表ルータ（DR）の選定が原因で時間がかかる可能性があります。

（注） 1 つの VPC ノードへのリンクまたはポート チャネルに障害が発生しても、OSPF 隣接関係がダウンすることはありません。OSPF 隣接関係は、その他の VPC ノードを介してアクセスできる外部 BD によりアップ状態を維持することができます。

GUI 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。
- 詳細については、『*Cisco APIC and Transit Routing*』を参照してください。

手順

ステップ 1 メニュー バーで、**[TENANTS]** > **[mgmt]** を選択します。

ステップ 2 [Navigation] ペインで、**[Networking]** > **[External Routed Networks]** を展開します。

ステップ 3 [External Routed Networks] を右クリックし、[Create Routed Outside] をクリックします。

ステップ 4 [Create Routed Outside] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、名前（RtdOut）を入力します。
b) [OSPF] チェックボックスをオンにします。
c) [OSPF Area ID] フィールドに、エリア ID を入力します。
d) [OSPF Area Control] フィールドで、適切なチェックボックスをオンにします。
e) [OSPF Area Type] フィールドで、適切なエリア タイプを選択します。
f) [OSPF Area Cost] フィールドで、適切な値を選択します。
g) [VRF] フィールドのドロップダウン リストから、VRF（inb）を選択します。

（注） このステップでは、ルーテッド Outside をインバンド VRF に関連付けます。

h) [External Routed Domain] ドロップダウン リストから、適切なドメインを選択します。
i) [Nodes and Interfaces Protocol Profiles] 領域の [+] アイコンをクリックします。

ステップ 5 [Create Node Profile] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、ノード プロファイルの名前を入力します（borderLeaf）。
b) [Nodes] フィールドで、[+] アイコンをクリックして [Select Node] ダイアログボックスを表示します。
c) [Node ID] フィールドで、ドロップダウン リストから、最初のノードを選択します（leaf1）。
d) [Router ID] フィールドに、一意のルータ ID を入力します。
e) [Use Router ID as Loopback Address] フィールドをオフにします。

（注） デフォルトでは、ルータ ID がループバック アドレスとして使用されます。これらが異なるようにする場合は、[Use Router ID as Loopback Address] チェックボックスをオフにします。

f) [Loopback Addresses] を展開し、[IP] フィールドに IP アドレスを入力します。[Update] をクリックし、[OK] をクリックします。

希望する IPv4 または IPv6 の IP アドレスを入力します。

g) [Nodes] フィールドで、[+] アイコンを展開して [Select Node] ダイアログボックスを表示します。

（注） 2 つ目のノード ID を追加します。

h) [Node ID] フィールドで、ドロップダウン リストから、次のノードを選択します（leaf2）。

i) [Router ID] フィールドに、一意のルータ ID を入力します。

j) [Use Router ID as Loopback Address] フィールドをオフにします。

（注） デフォルトでは、ルータ ID がループバック アドレスとして使用されます。これらが異なるようにする場合は、[Use Router ID as Loopback Address] チェックボックスをオフにします。

k) [Loopback Addresses] を展開し、[IP] フィールドに IP アドレスを入力します。[Update] をクリックし、[OK] をクリックします。[OK] をクリックします。

希望する IPv4 または IPv6 の IP アドレスを入力します。

ステップ 6 [Create Node Profile] ダイアログボックスで、[OSPF Interface Profiles] 領域の [+] アイコンをクリックします。

ステップ 7 [Create Interface Profile] ダイアログボックスで、次のタスクを実行します。

a) [Name] フィールドに、プロファイルの名前（portProf）を入力します。

b) [Interfaces] 領域で、[Routed Interfaces] タブをクリックし、[+] アイコンをクリックします。

c) [Select Routed Interfaces] ダイアログボックスの [Path] フィールドで、ドロップダウン リストから、最初のポート（leaf1、ポート 1/40）を選択します。

d) [IP Address] フィールドに、IP アドレスとマスクを入力します。[OK] をクリックします。

e) [Interfaces] 領域で、[Routed Interfaces] タブをクリックし、[+] アイコンをクリックします。

f) [Select Routed Interfaces] ダイアログボックスの [Path] フィールドで、ドロップダウン リストから、2 つ目のポート（leaf2、ポート 1/40）を選択します。

g) [IP Address] フィールドに、IP アドレスとマスクを入力します。[OK] をクリックします。

（注） この IP アドレスは、前に leaf1 に入力した IP アドレスと異なっている必要があります。

h) [Create Interface Profile] ダイアログボックスで、[OK] をクリックします。

インターフェイスが OSPF インターフェイスとともに設定されます。

ステップ 8 [Create Node Profile] ダイアログボックスで、[OK] をクリックします。

ステップ 9 [Create Routed Outside] ダイアログボックスで、[Next] をクリックします。
[Step 2 External EPG Networks] 領域が表示されます。

ステップ 10 [External EPG Networks] 領域で、[+] アイコンをクリックします。

ステップ 11 [Create External Network] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、外部ネットワークの名前（extMgmt）を入力します。
b) [Subnet] を展開し、[Create Subnet] ダイアログボックスの [IP address] フィールドに、サブネットの IP アドレスとマスクを入力します。
c) [Scope] フィールドで、目的のチェックボックスをオンにします。[OK] をクリックします。
d) **[Create External Network]** ダイアログボックスで、**[OK]** をクリックします。
e) **[Create Routed Outside]** ダイアログボックスで、**[Finish]** をクリックします。

（注） [Work] ペインで、[External Routed Networks] 領域に、外部ルーテッド ネットワークのアイコン（RtdOut）が表示されるようになりました。

NX-OS CLI を使用したテナントの OSPF 外部ルーテッド ネットワークの作成

外部ルーテッド ネットワーク接続の設定には、次のステップがあります。

1. テナントの下に VRF を作成します。
2. 外部ルーテッド ネットワークに接続された境界リーフ スイッチの VRF の L3 ネットワーキング構成を設定します。この設定には、インターフェイス、ルーティング プロトコル (BGP、OSPF、EIGRP) 、プロトコル パラメータ、ルートマップが含まれています。
3. テナントの下に外部 L3 EPG を作成してポリシーを設定し、これらの EPG を境界リーフ スイッチに導入します。ACI ファブリック内で同じポリシーを共有する VRF の外部ルーテッド サブネットが、1 つの「外部 L3 EPG」または 1 つの「プレフィクス EPG」を形成します。

設定は、2 つのモードで実現されます。

- テナント モード：VRF の作成および外部 L3 EPG 設定
- リーフ モード：L3 ネットワーキング構成と外部 L3 EPG の導入

次の手順は、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、テナントを選択してからテナント用の VRF を作成する必要があります。

（注） この項の例では、テナント「exampleCorp」の「OnlineStore」アプリケーションの「web」epg に外部ルーテッド接続を提供する方法について説明します。

手順

ステップ 1 VLAN ドメインを設定します。

例：

```
apic1(config)# vlan-domain dom_exampleCorp
apic1(config-vlan)# vlan 5-1000
apic1(config-vlan)# exit
```

ステップ 2 テナント VRF を設定し、VRF のポリシーの適用を有効にします。

例：

```
apic1(config)# tenant exampleCorp
apic1(config-tenant)# vrf context
 exampleCorp_v1
apic1(config-tenant-vrf)# contract enforce
apic1(config-tenant-vrf)# exit
```

ステップ 3 テナント BD を設定し、ゲートウェイ IP を「public」としてマークします。エントリ「scope public」は、このゲートウェイ アドレスを外部 L3 ネットワークのルーティング プロトコルによるアドバタイズに使用できるようにします。

例：

```
apic1(config-tenant)# bridge-domain exampleCorp_b1
apic1(config-tenant-bd)# vrf member exampleCorp_v1
apic1(config-tenant-bd)# exit
apic1(config-tenant)# interface bridge-domain exampleCorp_b1
apic1(config-tenant-interface)# ip address 172.1.1.1/24 scope public
apic1(config-tenant-interface)# exit
```

ステップ 4 リーフの VRF を設定します。

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant exampleCorp vrf exampleCorp_v1
```

ステップ 5 OSPF エリアを設定し、ルート マップを追加します。

例：

```
apic1(config-leaf)# router ospf default
apic1(config-leaf-ospf)# vrf member tenant exampleCorp vrf exampleCorp_v1
apic1(config-leaf-ospf-vrf)# area 0.0.0.1 route-map map100 out
apic1(config-leaf-ospf-vrf)# exit
apic1(config-leaf-ospf)# exit
```

ステップ 6 VRF をインターフェイス (この例ではサブインターフェイス) に割り当て、OSPF エリアを有効にします。

例：

（注） サブインターフェイスの構成では、メイン インターフェイス (この例では、ethernet 1/11) は、「no switchport」によって L3 ポートに変換し、サブインターフェイスが使用するカプセル化 VLAN を含む vlan ドメイン (この例では dom_exampleCorp) を割り当てる必要があります。サブインターフェイス ethernet1/11.500 で、500 はカプセル化 VLAN です。

```
apic1(config-leaf)# interface ethernet 1/11
apic1(config-leaf-if)# no switchport
apic1(config-leaf-if)# vlan-domain member dom_exampleCorp
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/11.500
apic1(config-leaf-if)# vrf member tenant exampleCorp vrf exampleCorp_v1
apic1(config-leaf-if)# ip address 157.10.1.1/24
apic1(config-leaf-if)# ip router ospf default area 0.0.0.1
```

ステップ 7 外部 L3 EPG ポリシーを設定します。これは、外部サブネットを特定し、epg「web」と接続する契約を消費するために一致させるサブネットが含まれます。

例：

```
apic1(config)# tenant t100
apic1(config-tenant)# external-l3 epg  l3epg100
apic1(config-tenant-l3ext-epg)# vrf member v100
apic1(config-tenant-l3ext-epg)# match ip 145.10.1.0/24
apic1(config-tenant-l3ext-epg)# contract consumer web
apic1(config-tenant-l3ext-epg)# exit
apic1(config-tenant)#exit
```

ステップ 8 リーフ スイッチの外部 L3 EPG を導入します。

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant t100 vrf v100
apic1(config-leaf-vrf)# external-l3 epg l3epg100
```

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。
- 詳細については、『*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"/>

         <l3extLIfP 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/>
                                 </l3extLIfP>
      </l3extLNodeP>
      <l3extRsEctx tnFvCtxName="inb"/>
      <ospfExtP areaId="57" />
   </l3extOut>
</fvTenant>
```

EIGRP 外部ルーテッド ネットワーク

EIGRP レイヤ 3 Outside 接続について

この例は、Cisco APIC を使用して、拡張内部ゲートウェイ ルーティング プロトコル（EIGRP）を設定する方法を示しています。次の情報は、EIGRP を設定するときに適用されます:

- テナント、VRF、およびブリッジ ドメインがすでに作成されている必要があります。
- レイヤ 3 外部テナント ネットワークがすでに設定されている必要があります。
- 外部ルーテッドのルート制御プロファイルがすでに設定されている必要があります。
- EIGRP VRF ポリシーは EIGRP ファミリ コンテキスト ポリシーと同じです。
- EIGRP はエクスポート ルート制御プロファイルをサポートしています。ルート制御に関する設定はすべてのプロトコルで共通です。

サブネット ルートをネットワーク レベルのルートへ自動的に要約するよう（ルート要約）、EIGRP を設定できます。たとえば、192.31.7.0 のサブネットが設定されているインターフェイス上で、サブネット 131.108.1.0 が 131.108.0.0 としてアドバタイズされるように設定することができます。自動集約は、EIGRP プロセスに設定されているネットワーク ルータ設定コマン

ドが2つまたはそれ以上ある場合に実行されます。デフォルトでは、この機能は有効です。詳細については、「*Route Summarization*」を参照してください。

EIGRP プロトコルのサポート

EIGRP プロトコルは、Cisco Application Centric Infrastructure（ACI）ファブリック内の他のルーティング プロトコルと同様にモデル化されています。

サポートされる機能

サポートされる機能は次のとおりです。

- IPv4 および IPv6 ルーティング
- 各アドレス ファミリの仮想ルーティングおよび転送（VRF）とインターフェイスの制御
- ノード間の OSPF による再配布
- VRF ごとのデフォルト ルート リーク ポリシー
- パッシブ インターフェイスおよびスプリット ホライズンのサポート
- エクスポートされたルートにタグを設定するためのルート マップ制御
- EIGRP インターフェイス ポリシーの帯域幅および遅延設定オプション

サポートされない機能

次の機能はサポートされていません。

- スタブ ルーティング
- BGP 接続に使用される EIGRP
- 同じノード上の複数の EIGRP `L3extOut`
- 認証サポート
- サマリー プレフィックス
- インターフェイスごとのインポートおよびエクスポート用配布リスト

EIGRP 機能のカテゴリ

EIGRP の機能は、次のように大きく分類できます。

- プロトコル ポリシー
- `L3extOut` の設定
- インターフェイス設定
- ルート マップ サポート

- デフォルト ルート サポート
- 中継サポート

EIGRP をサポートしているプライマリ管理対象オブジェクト

次のプライマリ管理対象オブジェクトは、EIGRP サポートを提供します。

- .**EIGRP アドレス ファミリ コンテキスト ポリシー** `eigrpCtxAfPol`：`fvTenant`（テナント/プロトコル）で設定されているアドレス ファミリ コンテキスト ポリシー
- `fvRsCtxToEigrpCtxAfPol`：所定のアドレス ファミリ（IPv4 または Ipv6）についての VRF から `eigrpCtxAfPol` への関係。関係は、アドレス ファミリごとに 1 つのみ存在できます。
- `eigrpIfPol`：`fvTenant` で設定される EIGRP インターフェイス ポリシー。
- `eigrpExtP`：`L3extOut` 上で EIGRP のフラグを有効にします。
- `eigrpIfP`：`l3extLIfP` に接続された EIGRP インターフェイス プロファイル。
- `eigrpRsIfPol`：EIGRP インターフェイス プロファイルから `eigrpIfPol` への関係。
- `Defrtleak`：`l3extOut` 下のデフォルト ルート リーク ポリシー。

テナントでサポートされる EIGRP プロトコル ポリシー

テナント下では次の EIGRP プロトコル ポリシーがサポートされます。

- **EIGRP インターフェイス ポリシー（`eigrpIfPol`）**：インターフェイス上の所定のアドレス ファミリに適用される設定が含まれます。インターフェイス ポリシーでは次の設定が可能です。
  - 秒単位の *hello* 間隔
  - 分単位の *hold* 間隔
  - 次のインターフェイス制御フラグのうち 1 つ以上。
    - スプリット ホライズン
    - パッシブ
    - ネクスト ホップ セルフ

- **EIGRP アドレス ファミリ コンテキスト ポリシー（`eigrpCtxAfPol`）**：所定の VRF 内の所定のアドレス ファミリの設定が含まれます。`eigrpCtxAfPol` は、テナント プロトコル ポリシー下で設定され、テナント下の 1 つ以上の VRF に適用できます。`eigrpCtxAfPol` は、VRF-per-address ファミリの関係を通して VRF で有効にできます。所定のアドレス ファミリに関係がない場合、あるいは関係に記述されている `eigrpCtxAfPol` が存在しない場合は、`[共通]` テナント下に作成されたデフォルトの VRF ポリシーがそのアドレス ファミリに使用されます。

次の設定では、`eigrpCtxAfPol` で許可されます。

- 内部ルートのアドミニストレーティブ ディスタンス
- 外部ルートのアドミニストレーティブ ディスタンス
- 最大許容 ECMP パス数
- アクティブ タイマー間隔
- メトリック バージョン（32 ビット/64 ビット メトリック）

ガイドラインと EIGRP を設定するときの制限事項

EIGRP を設定する場合は、次の注意事項に従ってください。

- 外部同じレイヤ 3 の EIGRP および BGP を設定することはサポートされていません。
- 外部同じレイヤ 3 の EIGRP や OSPF を設定することはサポートされていません。
- 1 つ EIGRP レイヤ 3 Out VRF あたりノードごとできますがあります。ノードで複数の Vrf を導入している場合、自身レイヤ 3 Out 各 VRF ことができます。
- 複数の EIGRP ピア、1 つレイヤ 3 Out からがサポートされます。これにより、1 つレイヤ 3 Out と同じノードから複数の EIGRP デバイスに接続できます。

GUI を使用したEIGRPの設定

手順

ステップ 1 メニュー バーで、**[Tenants]** > **[All Tenants]** の順に選択します。

ステップ 2 **Work** ウィンドウで、テナントをダブルクリックします。

ステップ 3 **Navigation** ウィンドウで、***Tenant_name*** > **Networking** > **Protocol Policies** > **EIGRP** を展開します。

ステップ 4 右クリックして **EIGRP アドレス ファミリ コンテキスト**] を選択します **EIGRP アドレス ファミリ コンテキストのポリシーを作成** します。

ステップ 5 **Create EIGRP Address Family Context Policy** ダイアログボックスで、以下の操作を実行します:

a) **Name** フィールドに、コンテキスト ポリシーの名前を入力します。
b) **アクティブ間隔 (分)** フィールドで、インターバル タイマーを選択します。
c) **外部距離** 、および **内部距離** フィールドで、適切な値を選択します。
d) **パスの上限** フィールドで、[インターフェイス (ノードごと/リーフ スイッチごと) 間の値を適切なロード バランシングを選択します。
e) **メトリック スタイル** フィールドで、適切なメトリック スタイルを選択します。[Submit] をクリックします。`

Work ウィンドウに、コンテキスト ポリシーの詳細が表示されます。

ステップ 6 VRF のコンテキスト ポリシーを適用する、 **ナビゲーション**] ペインで、[展開 **ネットワー キング** > **Vrf** 。

ステップ 7 適切な VRF を選択し、[、 **作業** ペインの [**プロパティ** 、展開 **アドレス ファミリごとの EIGRP コンテキスト** 。

ステップ 8 **EIGRP アドレス ファミリ タイプ** ドロップダウンリスト、IP バージョンを選択します。

ステップ 9 **EIGRP アドレス ファミリ コンテキスト** ドロップダウンリスト、コンテキスト ポリシーを選択します。**Update** をクリックし、**Submit** をクリックします。

ステップ 10 レイヤ 3 Out、内の EIGRP を有効にする、 **ナビゲーション**] ペインで、をクリックして **ネットワー キング** > **外部ルーテッド ネットワーク** 、目的のレイヤ 3 ネットワークの外部] をクリックします。

ステップ 11 **作業** ペインの [**プロパティ** 、チェック ボックスをオンに **EIGRP** 、EIGRP 自律システム番号を入力します。[Submit] をクリックします。`

ステップ 12 EIGRP インターフェイス ポリシーの作成、 **ナビゲーション**] ペインで、をクリックして **ネットワー キング** > **プロトコル ポリシー** > **EIGRP インターフェイス** し、次のアクションを実行します。

a) 右クリックして **EIGRP インターフェイス** 、をクリックし、 **EIGRP インターフェイス ポリシーの作成** します。

b) **Create EIGRP Interface Policy** ダイアログボックスで、**Name** フィールドにポリシーの名前を入力します。

c) **制御状態** フィールドは、1 つまたは複数の制御を有効にする目的のチェック ボックスをチェックします。

d) **Helloインターバル (秒)** フィールドで、目的の間隔を選択します。

e) **保留間隔 (秒)** フィールドで、目的の間隔を選択します。[Submit] をクリックします。`

f) **Bandwidth** フィールドで、目的の帯域幅を選択します。

g) **遅延** フィールドで、10 マイクロ秒またはピコセル秒で、目的の遅延を選択します。

作業] ペインで、EIGRP インターフェイス ポリシーの詳細が表示されます。

ステップ 13 **ナビゲーション**] ペインで、適切な外部ルーテッド ネットワークの EIGRP が有効になってクリック展開 **論理ノード プロファイル** および次の操作の実行します。

a) 適切なノードとそのノードの下にインターフェイスを展開します。

b) インターフェイスを右クリックし、をクリックして **EIGRP インターフェイス プロファイルの作成** します。

c) **EIGRP インターフェイス プロファイルの作成** ダイアログボックスで、 **EIGRP ポリシー** フィールドで、目的のEIGRPインターフェイスポリシーを選択します。[Submit]をクリックします。`

（注） EIGRP の VRF ポリシーおよび EIGRP インターフェイス ポリシーは、EIGRP が有効になっているときに使用するプロパティを定義します。EIGRP の VRF ポリシーおよび EIGRP インターフェイス ポリシーは、ユーザが新しいポリシーを作成しない場合にもデフォルトポリシーとして利用できます。したがって、ユーザには、ポリシーのいずれかの選択に明示的には、デフォルトのポリシーは EIGRP が有効になっているときに利用自動的にします。

これで EIGRP の設定は完了です。

NX-OS スタイルの CLI を使用したEIGRPの設定

手順

ステップ 1 ファブリックの Application Policy Infrastructure Controller (APIC) に SSH 接続します。

例：

```
# ssh admin@node_name
```

ステップ 2 設定モードを開始します。

例：

```
apic1# configure
```

ステップ 3 テナントの設定モードを入力します。

例：

```
apic1(config)# tenant tenant1
```

ステップ 4 テナントでレイヤ 3 Outside を設定します:

例：

```
apic1(config-tenant)# show run
# Command: show running-config tenant tenant1
# Time: Tue Feb 16 09:44:09 2016
  tenant tenant1
    vrf context l3out
      exit
    l3out l3out-L1
      vrf member l3out
      exit
    l3out l3out-L3
      vrf member l3out
      exit
    external-l3 epg tenant1 l3out l3out-L3
      vrf member l3out
      match ip 0.0.0.0/0
      match ip 3.100.0.0/16
      match ipv6 43:101::/48
      contract consumer default
      exit
    external-l3 epg tenant1 l3out l3out-L1
      vrf member l3out
      match ipv6 23:101::/48
      match ipv6 13:101::/48
      contract provider default
      exit
    exit
```

ステップ 5 リーフで EIGRP の VRF を設定します:

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant tenant1 vrf l3out l3out l3out-L1
apic1(config-leaf-vrf)# show run
# Command: show running-config leaf 101 vrf context tenant tenant1 vrf l3out l3out
l3out-L1
# Time: Tue Feb 16 09:44:45 2016
  leaf 101
    vrf context tenant tenant1 vrf l3out l3out l3out-L1
      router-id 3.1.1.1
      route-map l3out-L1_in
        scope global
        ip prefix-list tenant1 permit 1:102::/48
        match prefix-list tenant1
          exit
        exit
      route-map l3out-L1_out
        scope global
        ip prefix-list tenant1 permit 3.102.10.0/23
        ip prefix-list tenant1 permit 3.102.100.0/31
        ip prefix-list tenant1 permit 3.102.20.0/24
        ip prefix-list tenant1 permit 3.102.30.0/25
        ip prefix-list tenant1 permit 3.102.40.0/26
        ip prefix-list tenant1 permit 3.102.50.0/27
        ip prefix-list tenant1 permit 3.102.60.0/28
        ip prefix-list tenant1 permit 3.102.70.0/29
        ip prefix-list tenant1 permit 3.102.80.0/30
        ip prefix-list tenant1 permit 3.102.90.0/32
        <OUTPUT TRUNCATED>
        ip prefix-list tenant1 permit ::/0
        match prefix-list tenant1
          exit
        exit
      route-map l3out-L1_shared
        scope global
        exit
      exit
    exit
```

ステップ 6 EIGRP インターフェイス ポリシーを設定します:

例：

```
apic1(config-leaf)# template eigrp interface-policy tenant1 tenant tenant1
This template will be available on all leaves where tenant tenant1 has a VRF deployment
apic1(config-template-eigrp-if-pol)# show run
# Command: show running-config leaf 101 template eigrp interface-policy tenant1 tenant
tenant1
# Time: Tue Feb 16 09:45:50 2016
  leaf 101
    template eigrp interface-policy tenant1 tenant tenant1
      ip hello-interval eigrp default 10
      ip hold-interval eigrp default 30
      ip throughput-delay eigrp default 20 tens-of-micro
      ip bandwidth eigrp default 20
      exit
    exit
```

ステップ 7 EIGRP の VRF ポリシーを設定します:

例：

```
apic1(config-leaf)# template eigrp vrf-policy tenant1 tenant tenant1
This template will be available on all leaves where tenant tenant1 has a VRF deployment
apic1(config-template-eigrp-vrf-pol)# show run
```

```
# Command: show running-config leaf 101 template eigrp vrf-policy tenant1 tenant tenant1
# Time: Tue Feb 16 09:46:31 2016
  leaf 101
    template eigrp vrf-policy tenant1 tenant tenant1
      metric version 64bit
      exit
    exit
```

ステップ 8 EIGRP VLAN インターフェイスを設定し、インターフェイスで EIGRP を有効にします:

例：

```
apic1(config-leaf)# interface vlan 1013
apic1(config-leaf-if)# show run
# Command: show running-config leaf 101 interface vlan 1013
# Time: Tue Feb 16 09:46:59 2016
  leaf 101
    interface vlan 1013
      vrf member tenant tenant1 vrf l3out
      ip address 101.13.1.2/24
      ip router eigrp default
      ipv6 address 101:13::1:2/112 preferred
      ipv6 router eigrp default
      ipv6 link-local fe80::101:13:1:2
      inherit eigrp ip interface-policy tenant1
      inherit eigrp ipv6 interface-policy tenant1
      exit
    exit
apic1(config-leaf-if)# ip summary-address ?
 eigrp  Configure route summarization for EIGRP
apic1(config-leaf-if)# ip summary-address eigrp default 11.11.0.0/16 ?
 <CR>
apic1(config-leaf-if)# ip summary-address eigrp default 11.11.0.0/16
apic1(config-leaf-if)# ip summary-address eigrp default 11:11:1::/48
apic1(config-leaf-if)# show run
# Command: show running-config leaf 101 interface vlan 1013
# Time: Tue Feb 16 09:47:34 2016
  leaf 101
    interface vlan 1013
      vrf member tenant tenant1 vrf l3out
      ip address 101.13.1.2/24
      ip router eigrp default
      ip summary-address eigrp default 11.11.0.0/16
      ip summary-address eigrp default 11:11:1::/48
      ipv6 address 101:13::1:2/112 preferred
      ipv6 router eigrp default
      ipv6 link-local fe80::101:13:1:2
      inherit eigrp ip interface-policy tenant1
      inherit eigrp ipv6 interface-policy tenant1
      exit
    exit
```

ステップ 9 物理インターフェイスに VLAN を適用します:

例：

```
apic1(config-leaf)# interface ethernet 1/5
apic1(config-leaf-if)# show run
# Command: show running-config leaf 101 interface ethernet 1 / 5
# Time: Tue Feb 16 09:48:05 2016
  leaf 101
    interface ethernet 1/5
      vlan-domain member cli
      switchport trunk allowed vlan 1213 tenant tenant13 external-svi l3out l3out-L1
```

```
      switchport trunk allowed vlan 1613 tenant tenant17 external-svi l3out l3out-L1
      switchport trunk allowed vlan 1013 tenant tenant1 external-svi l3out l3out-L1
     switchport trunk allowed vlan 666 tenant ten_v6_cli external-svi l3out l3out_cli_L1

      switchport trunk allowed vlan 1513 tenant tenant16 external-svi l3out l3out-L1
      switchport trunk allowed vlan 1313 tenant tenant14 external-svi l3out l3out-L1
      switchport trunk allowed vlan 1413 tenant tenant15 external-svi l3out l3out-L1
      switchport trunk allowed vlan 1113 tenant tenant12 external-svi l3out l3out-L1
      switchport trunk allowed vlan 712 tenant mgmt external-svi l3out inband_l1
      switchport trunk allowed vlan 1913 tenant tenant10 external-svi l3out l3out-L1
      switchport trunk allowed vlan 300 tenant tenant1 external-svi l3out l3out-L1
      exit
    exit
```

ステップ 10 ルータ EIGRP を有効にします:

例：

```
apic1(config-eigrp-vrf)# show run
# Command: show running-config leaf 101 router eigrp default vrf member tenant tenant1
vrf l3out
# Time: Tue Feb 16 09:49:05 2016
  leaf 101
    router eigrp default
      exit
    router eigrp default
      exit
    router eigrp default
      exit
    router eigrp default
      vrf member tenant tenant1 vrf l3out
        autonomous-system 1001 l3out l3out-L1
        address-family ipv6 unicast
          inherit eigrp vrf-policy tenant1
          exit
        address-family ipv4 unicast
          inherit eigrp vrf-policy tenant1
          exit
        exit
      exit
```

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/eigrpCtxAfP-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">
                <l3extLIfP 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>
                </l3extLIfP>
            </l3extLNodeP>
        </l3extOut>
    </fvTenant>
</polUni>
```

IPv6

```
<polUni>
    <fvTenant name="cisco_6">
        <l3extOut name="ext">
```

```
            <eigrpExtP asn="4001"/>
            <l3extLNodeP name="node1">
                <l3extLIfP 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>
                </l3extLIfP>
            </l3extLNodeP>
        </l3extOut>
    </fvTenant>
</polUni>
```

IPv4 および IPv6

```
<polUni>
    <fvTenant name="cisco_6">
        <l3extOut name="ext">
            <eigrpExtP asn="4001"/>
            <l3extLNodeP name="node1">
                <l3extLIfP 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>
                </l3extLIfP>

                <l3extLIfP 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>
                </l3extLIfP>
            </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>
```

`Bandwidth（bw）属性は`（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。