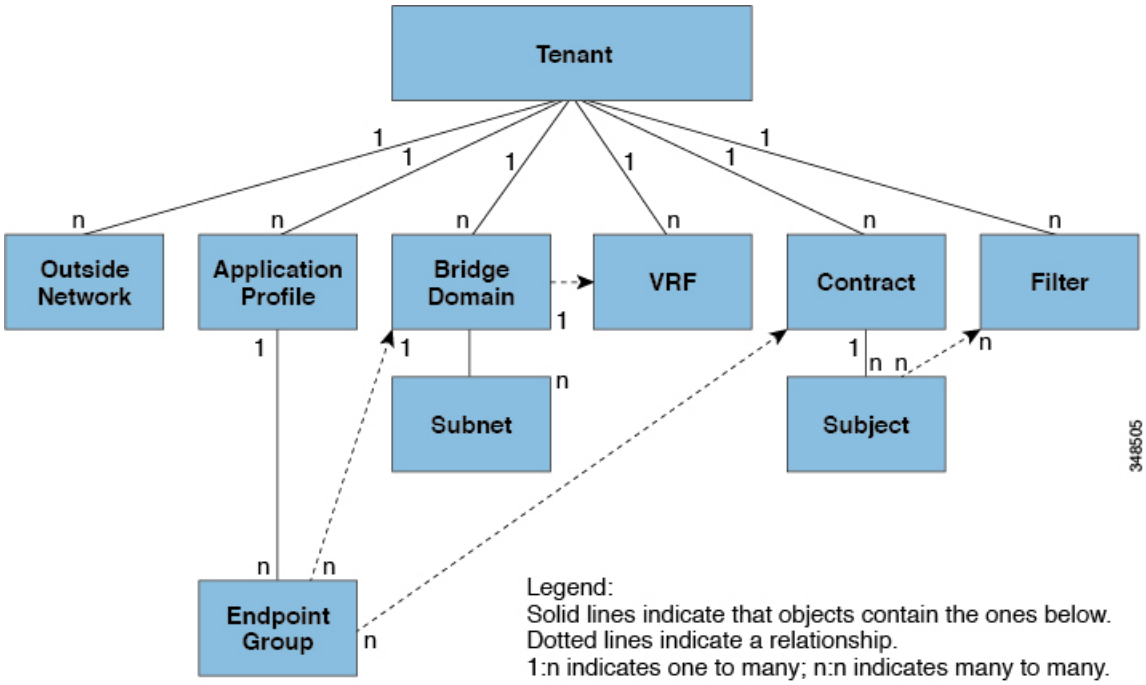
基本ユーザ テナント設定

この章は、次の内容で構成されています。

テナント

テナント (`fvTenant`) は、アプリケーション ポリシーの論理コンテナで、管理者はドメインベースのアクセス コントロールを実行できます。テナントはポリシーの観点から分離の単位を表しますが、プライベート ネットワークは表しません。テナントは、サービス プロバイダーの環境ではお客様を、企業の環境では組織またはドメインを、または単にポリシーの便利なグループ化を表すことができます。次の図は、管理情報ツリー (MIT) のテナント部分の概要を示します。

図 *1:* テナント



テナントは相互に分離することも、リソースを共有することもできます。テナントに含まれる主要な要素は、フィルタ、コントラクト、外部ネットワーク、ブリッジ ドメイン、仮想ルーティングおよび転送 (VRF) インスタンス、エンドポイント グループ (EPG) を含むアプリケーション プロファイルです。テナントのエンティティはそのポリシーを継承します。VRF はコンテキストとも呼ばれ、それぞれを複数のブリッジ ドメインに関連付けることができます。

（注） APIC GUI のテナント ナビゲーション パスでは、VRF (コンテキスト) はプライベート ネットワークと呼ばれます。

テナントはアプリケーション ポリシーの論理コンテナです。ファブリックには複数のテナントを含めることができます。レイヤ 4 ～ 7 のサービスを展開する前に、テナントを設定する必要があります。ACI ファブリックは、テナント ネットワークに対して IPv4、IPv6、およびデュアル スタック構成をサポートします。

テナント内のルーティング

アプリケーション セントリック インフラストラクチャ（ACI）のファブリックでは、テナントのデフォルト ゲートウェイ機能が提供され、ファブリックの Virtual Extensible Local Area（VXLAN）ネットワーク間のルーティングが行えます。各テナントについて、APIC でサブネットが作成されるたびに、ファブリックは仮想デフォルト ゲートウェイまたはスイッチ仮想インターフェイス（SVI）を提供します。これは、そのテナント サブネットの接続エンドポイ

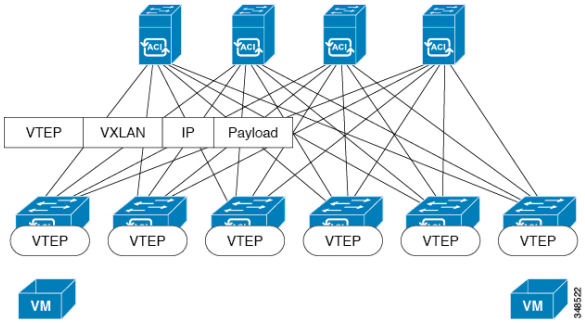
ントがあるすべてのスイッチにわたります。各入力インターフェイスはデフォルトのゲートウェイ インターフェイスをサポートし、ファブリック全体のすべての入力インターフェイスは任意のテナント サブネットに対する同一のルータの IP アドレスと MAC アドレスを共有します。

サブネット間のテナント トラフィックの転送を促進するレイヤ3VNID

ACI ファブリックは、ACI ファブリック VXLAN ネットワーク間のルーティングを実行するテナントのデフォルト ゲートウェイ機能を備えています。各テナントに対して、ファブリックはテナントに割り当てられたすべてのリーフ スイッチにまたがる仮想デフォルト ゲートウェイを提供します。これは、エンドポイントに接続された最初のリーフ スイッチの入力インターフェイスで提供されます。各入力インターフェイスはデフォルト ゲートウェイ インターフェイスをサポートします。ファブリック全体のすべての入力インターフェイスは、特定のテナント サブネットに対して同一のルータの IP アドレスと MAC アドレスを共有します。

ACI ファブリックは、エンドポイントのロケータまたは VXLAN トンネル エンドポイント (VTEP) アドレスで定義された場所から、テナント エンドポイント アドレスとその識別子を切り離します。ファブリック内の転送は VTEP 間で行われます。次の図は、ACI で切り離された ID と場所を示します。

図 *2 : ACI* によって切り離された *ID* と場所



VXLAN は VTEP デバイスを使用してテナントのエンド デバイスを VXLAN セグメントにマッピングし、VXLAN のカプセル化およびカプセル化解除を実行します。各 VTEP 機能には、次の 2 つのインターフェイスがあります。

- ブリッジングを介したローカル エンドポイント通信をサポートするローカル LAN セグメントのスイッチ インターフェイス
- 転送 IP ネットワークへの IP インターフェイス

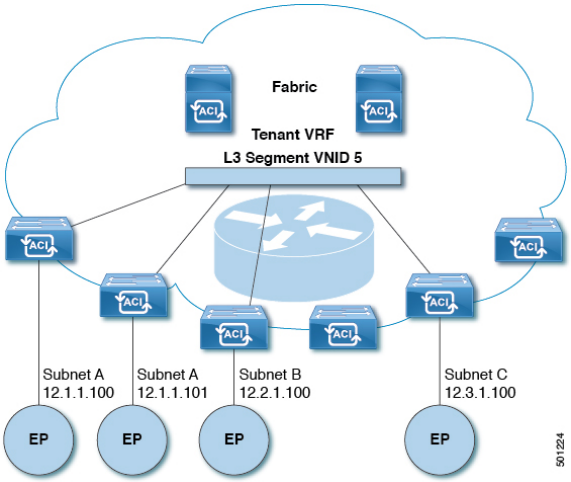
IP インターフェイスには一意の IP アドレスがあります。これは、インフラストラクチャ VLAN として知られる、転送 IP ネットワーク上の VTEP を識別します。VTEP デバイスはこの IP アドレスを使用してイーサネット フレームをカプセル化し、カプセル化されたパケットを、IP インターフェイスを介して転送ネットワークへ送信します。また、VTEP デバイスはリモート VTEP で VXLAN セグメントを検出し、IP インターフェイスを介してリモートの MAC Address-to-VTEP マッピングについて学習します。

ACI の VTEP は分散マッピング データベースを使用して、内部テナントの MAC アドレスまたはIP アドレスを特定の場所にマッピングします。VTEP はルックアップの完了後に、宛先リーフ スイッチ上の VTEP を宛先アドレスとして、VXLAN 内でカプセル化された元のデータ パケットを送信します。宛先リーフ スイッチはパケットをカプセル化解除して受信ホストに送信します。このモデルにより、ACI はスパニングツリー プロトコルを使用することなく、フル メッシュでシングル ホップのループフリー トポロジを使用してループを回避します。

VXLAN セグメントは基盤となるネットワーク トポロジに依存しません。逆に、VTEP 間の基盤となる IP ネットワークは、VXLAN オーバーレイに依存しません。これは送信元 IP アドレスとして開始 VTEP を持ち、宛先 IP アドレスとして終端 VTEP を持っており、外部 IP アドレス ヘッダーに基づいてパケットをカプセル化します。

次の図は、テナント内のルーティングがどのように行われるかを示します。

図 *3 : ACI* のサブネット間のテナント トラフィックを転送するレイヤ *3 VNID*



ACI はファブリックの各テナント VRF に単一の L3 VNID を割り当てます。ACI は、L3 VNID に従ってファブリック全体にトラフィックを転送します。出力リーフ スイッチでは、ACI によって L3 VNID からのパケットが出力サブネットの VNID にルーティングされます。

ACI のファブリック デフォルト ゲートウェイに送信されてファブリック入力に到達したトラフィックは、レイヤ 3 VNID にルーティングされます。これにより、テナント内でルーティングされるトラフィックはファブリックで非常に効率的に転送されます。このモデルを使用すると、たとえば同じ物理ホスト上の同じテナントに属し、サブネットが異なる 2 つの VM 間では、トラフィックが (最小パス コストを使用して) 正しい宛先にルーティングされる際に経由する必要があるは入力スイッチ インターフェイスのみです。

ACI ルート リフレクタは、ファブリック内での外部ルートの配布にマルチプロトコル BGP（MP-BGP）を使用します。ファブリック管理者は自律システム (AS) 番号を提供し、ルート リフレクタにするスパイン スイッチを指定します。

（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）を介したマルチポッド接続を設定する場合は、インターフェイス MTU がリンクの両端で適切に設定することを推奨します。

IGP プロトコル パケット（EIGRP、OSPFv3）は、インターフェイス MTU サイズに基づいてコンポーネントによって構築されます。 Cisco ACIでは、CPU MTU サイズがインターフェイス MTU サイズよりも小さく、構築されたパケットサイズが CPU MTU より大きい場合、パケットはカーネルによってドロップされます（特に IPv6）。このような制御パケットのドロップを回避するには、コントロールプレーンとインターフェイスの両方で常に同じ MTU 値を設定します。

Cisco ACI、Cisco NX-OS、および Cisco IOS などの一部のプラットフォームでは、設定可能な MTU 値はイーサネット ヘッダー (一致する IP MTU、14-18 イーサネット ヘッダー サイズを除く) を考慮していません。また、IOS XR などの他のプラットフォームには、設定された MTU 値にイーサネットヘッダーが含まれています。設定された値が 9000 の場合、Cisco ACI、Cisco NX-OS および Cisco IOS の最大 IP パケット サイズは 9000 バイトになりますが、IOS-XR のタグなしインターフェイスの最大 IP パケットサイズは 8986 バイトになります。

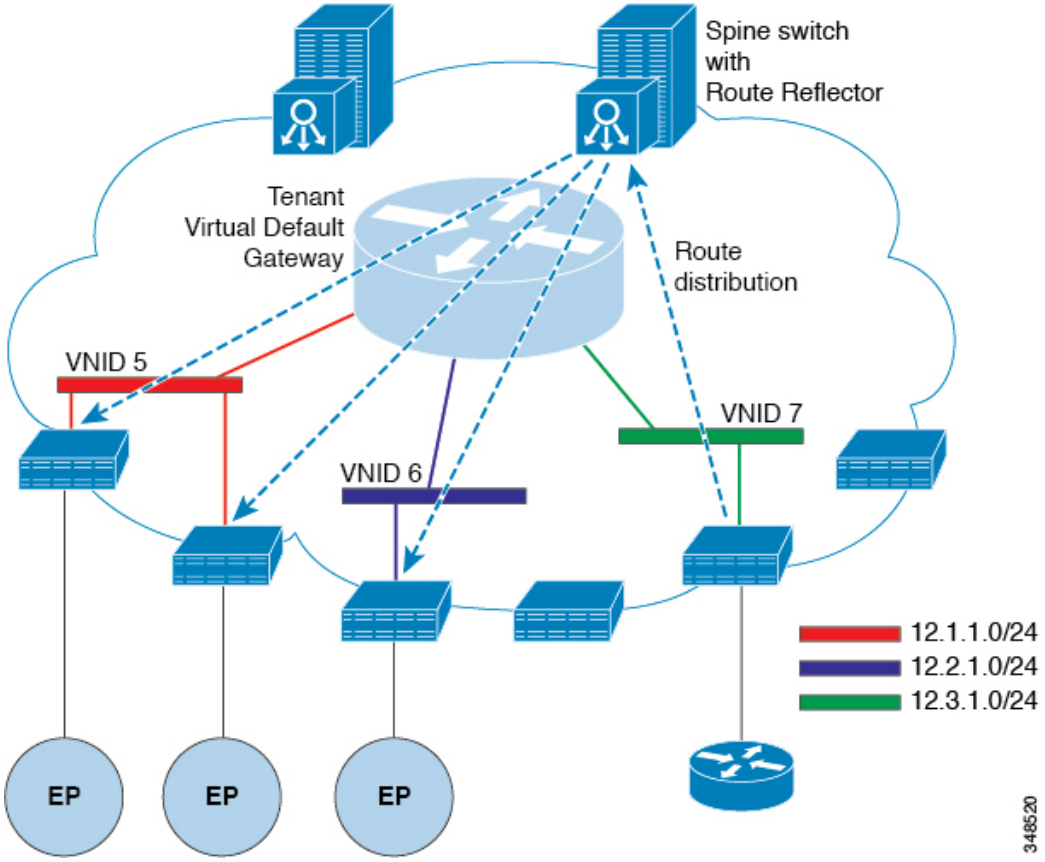
各プラットフォームの適切な MTU 値については、それぞれの設定ガイドを参照してください。

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、コマンド、ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1 を使用してください。

ルータ ピアリングおよびルート配布

次の図に示すように、ルーティング ピア モデルを使用すると、リーフ スイッチ インターフェイスが外部ルータのルーティング プロトコルとピアリングするように静的に設定されます。

図 *4:* ルータのピアリング

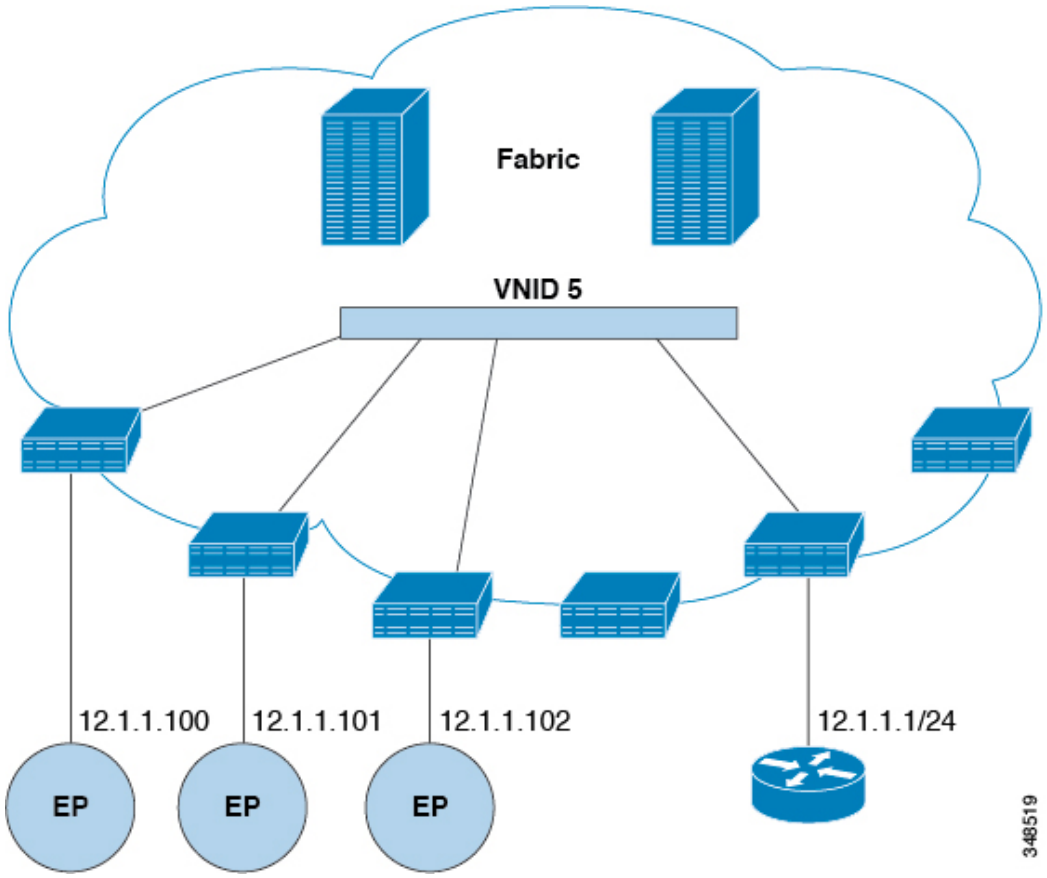


ピアリングによって学習されるルートは、スパイン スイッチに送信されます。スパイン スイッチはルート リフレクタとして動作し、外部ルートを同じテナントに属するインターフェイスを持つすべてのリーフ スイッチに配布します。これらのルートは、最長プレフィクス照合 (LPM) により集約されたアドレスで、外部ルータが接続されているリモートのリーフ スイッチの VTEP IP アドレスが含まれるリーフ スイッチの転送テーブルに配置されます。WAN ルートには転送プロキシはありません。WAN ルートがリーフ スイッチの転送テーブルに適合しない場合、トラフィックはドロップされます。外部ルータがデフォルト ゲートウェイではないため、テナントのエンドポイント (EP) からのパケットは ACI ファブリックのデフォルト ゲートウェイに送信されます。

外部ルータへのブリッジド インターフェイス

次の図に示すように、リーフ スイッチのインターフェイスがブリッジド インターフェイスとして設定されている場合、テナント VNID のデフォルト ゲートウェイが外部ルータとなります。

図 *5:* ブリッジド外部ルータ



ACI ファブリックは、外部ルータの存在を認識せず、APIC はリーフ スイッチのインターフェイスを EPG に静的に割り当てます。

ルート リフレクタの設定

ACIファブリックのルートリフレクタは、マルチプロトコルBGP（MP-BGP）を使用してファブリック内に外部ルートを配布します。ACIファブリックでルートリフレクタをイネーブルにするには、ファブリックの管理者がルート リフレクタになるスパイン スイッチを選択して、自律システム（AS）番号を提供する必要があります。冗長性を確保するために、ポッドあたり少なくとも 2 つのスパイン ノードを MP-BGP ルート リフレクタとして設定することを推奨します。

ルート リフレクタが ACI ファブリックで有効になったら、管理者は、レイヤ 3 Out（L3Out）というコンポーネントを使用してリーフ ノードを介して外部ネットワークへの接続を設定できます。L3Out で設定されたリーフ ノードは、境界リーフと呼ばれます。境界リーフは、L3Out で指定されたルーティング プロトコルを介して、接続された外部デバイスとルートを交換します。L3Out 経由でスタティック ルートを設定することもできます。

L3Out とスパイン ルート リフレクタの両方が展開されると、境界リーフ ノードは L3Out を介して外部ルートを学習し、それらの外部ルートはスパイン MP-BGP ルート リフレクタを介してファブリック内のすべてのリーフ ノードに配布されます。

リーフでサポートされるルートの最大数については、ご使用のリリースの『Cisco APICの検証済みスケーラビリティ ガイド』を参照してください。

Layer 3 Out を使用した外部接続の設定

この項では、ACI ファブリックが L3Out および MP-BGP ルート リフレクタを介して外部ルーテッド ネットワークに接続するために必要な手順を段階的に説明します。

この例では、Open Shortest Path First（OSPF）を「mgmt」テナント下の L3Out のルーティング プロトコルとして使用します。

GUI 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 メニュー バーで、**[System]** > **[System Settings]** の順に選択します。

ステップ 2 [**ナビゲーション**（**Navigation**）] ペインで、[**BGP ルート リフレクタ**（**BGP Route Reflector**）] を右クリックして、[**ルート リフレクタ ノードの作成**（**Create Route Reflector Node**）] をクリックします。

ステップ 3 **[ルート リフレクタ ノードの作成**（**Create Route Reflector Node**）] ダイアログ ボックスで、[**スパイン ノード**（**Spine Node**）] ドロップダウン リストから、適切なスパイン ノードを選択します。**Submit** をクリックします。

（注） 必要に応じてスパイン ノードを追加するには、上記の手順を繰り返してください。

スパイン スイッチがルート リフレクタ ノードとしてマークされます。

ステップ 4 **BGP Route Reflector** プロパティ エリアの **Autonomous System Number** フィールドで、適切な番号を選択します。**Submit** をクリックします。

（注） 自律システム番号は、Border Gateway Protocol（BGP）がルータに設定されている場合は、リーフが接続されたルータ設定に一致する必要があります。スタティックまたは Open Shortest Path First（OSPF）を使用して学習されたルートを使用している場合は、自律システム番号値を任意の有効な値にできます。

ステップ 5 メニュー バーで、**[ファブリック**（**Fabric**）**]** > **[ファブリック ポリシー**（**Fabric Policies**）**]** > **[ポッド**（**Pods**）**]** > **[ポリシー グループ**（**Policy Groups**）**]** をクリックします。

ステップ 6 **[ナビゲーション**（**Navigation**）] ペインで、[**ポリシー グループ**（**Policy Groups**）] を展開して右クリックし、[**POD ポリシー グループの作成**（**Create POD Policy Group**）] をクリックします。

ステップ 7 **[ポッド ポリシー グループの作成（Create Pod Policy Group）]** ダイアログ ボックスで、**[名前（Name）]** フィールドに、ポッド ポリシー グループの名前を入力します。

ステップ 8 [BGP Route Reflector Policy] ドロップダウン リストで、適切なポリシー（デフォルト）を選択します。[Submit] をクリックします。`
BGP ルート リフレクタのポリシーは、ルート リフレクタのポッド ポリシー グループに関連付けられ、BGP プロセスはリーフ スイッチでイネーブルになります。

ステップ 9 メニュー バーで、**[ファブリック（Fabric）]** > **[ファブリック ポリシー（Fabric Policies）]** > **[プロファイル（Profiles）]** > **[ポッド プロファイル デフォルト（Pod Profile default）]** > **[デフォルト（default）]**を選択します。

ステップ 10 [Work] ペインで、[Fabric Policy Group] ドロップダウン リストから、前に作成されたポッド ポリシーを選択します。[Submit] をクリックします。`
ポッド ポリシー グループが、ファブリック ポリシー グループに適用されました。

ACI ファブリックの MP-BGP ルート リフレクタの設定

ACI ファブリック内のルートを配布するために、MP-BGP プロセスを最初に実行し、スパイン スイッチを BGP ルート リフレクタとして設定する必要があります。

次に、MP-BGP ルート リフレクタの設定例を示します。

（注） この例では、BGP ファブリック ASN は 100 です。スパイン スイッチ 104 と 105 が MP-BGP ルート リフレクタとして選択されます。

```
apic1(config)# bgp-fabric
apic1(config-bgp-fabric)# asn 100
apic1(config-bgp-fabric)# route-reflector spine 104,105
```

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

<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 セレ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”>
    <fabricRsPodPGrpBGPRRP 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>
```

MP-BGP ルート リフレクタ設定の確認

手順

ステップ 1 次の操作を実行して、設定を確認します。

a) セキュア シェル（SSH）を使用して、必要に応じて各リーフ スイッチへの管理者としてログインします。

b) show processes | grep bgp コマンドを入力して、状態が S であることを確認します。

状態が NR（実行していない）である場合は、設定が正常に行われませんでした。

ステップ 2 次の操作を実行して、自律システム番号がスパイン スイッチで設定されていることを確認します。

a) SSH を使用して、必要に応じて各スパイン スイッチへの管理者としてログインします。

b) シェル ウィンドウから次のコマンドを実行します。

例：

cd /mit/sys/bgp/inst

例：

grep asn summary

設定した自律システム番号が表示される必要があります。自律システム番号の値が 0 と表示される場合は、設定が正常に行われませんでした。

GUI を使用した管理テナントの OSPF L3Out 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 L3Out を作成するためのものです。テナントの OSPF L3Out 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]** の順に選択します。

ステップ 2 [**ナビゲーション**（**Navigation**）] ペインで、**[ネットワーキング**（**Networking**）**]** > **[L3Outs]** を展開します。

ステップ 3 [L3Outs] を右クリックし、[L3Out の作成（Create L3Out）] をクリックします。

[**L3Out の作成**（**Create L3Out**）] ウィザードが表示されます。

ステップ 4 [L3Out の作成（Create L3Out）] ウィザードの [識別（Identity）] ウィンドウで、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) [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。

c) [L3 ドメイン（L3 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。
d) [OSPF] チェックボックスをオンにします。
e) [OSPF Area ID] フィールドに、エリア ID を入力します。
f) [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。
g) [OSPF Area Type] フィールドで、適切なエリア タイプを選択します。
h) [OSPF Area Cost] フィールドで、適切な値を選択します。
i) **[次へ**（**Next**）**]** をクリックします。

[**ノードとインターフェイス**（**Nodes and Interfaces**）] ウィンドウが表示されます。

ステップ 5 [ノードとインターフェイス（Nodes and Interfaces）] ウィンドウで、次の操作を実行します。

a) [デフォルトを使用（Use Defaults）] ボックスをオフにします。

これにより、[ノードプロファイル名（Node Profile Name）] フィールドを編集できます。

b) [**ノード プロファイル名**（**Node Profile 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）。

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) ループバック アドレスにルータ ID を使用しない場合は、[ループバック アドレス（Loopback Address）] フィールドで別の IP アドレスを使用するか、空のままにします。

（注） [ルータ ID（Router ID）] フィールドに入力したエントリと同じ内容が [ループバック アドレス（Loopback Address）] フィールドに自動で入力されます。これは以前のビルドでの [ループバック アドレスのルータ ID の使用（Use Router ID for Loopback Address）] と同等です。ループバック アドレスにルータ ID を使用しない場合は、別の IP アドレスを使用するか、このフィールドを空のままにします。

f) 必要に応じて、このノードの [インターフェイス（Interface）]、[IP アドレス（IP Address）]、[インターフェイス プロファイル名（Interface Profile Name）]、および [MTU] フィールドに適切な情報を入力します。
g) [ノード（Nodes）] フィールドで、[+] アイコンをクリックして、別のノードの 2 番目のフィールド セットを追加します。

（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、最初のノードを選択します（leaf1）。
i) [Router ID] フィールドに、一意のルータ ID を入力します。
j) ループバック アドレスにルータ ID を使用しない場合は、[ループバック アドレス（Loopback Address）] フィールドで別の IP アドレスを使用するか、空のままにします。

（注） [ルータ ID（Router ID）] フィールドに入力したエントリと同じ内容が [ループバック アドレス（Loopback Address）] フィールドに自動で入力されます。これは以前のビルドでの [ループバック アドレスのルータ ID の使用（Use Router ID for Loopback Address）] と同等です。ループバック アドレスにルータ ID を使用しない場合は、別の IP アドレスを使用するか、このフィールドを空のままにします。

k) 必要に応じて、このノードの [インターフェイス（Interface）]、[IP アドレス（IP Address）]、[インターフェイス プロファイル名（Interface Profile Name）]、および [MTU] フィールドに適切な情報を入力します。
l) **[次へ（Next）]** をクリックします。

[プロトコル（Protocols）] ウィンドウが表示されます。

ステップ 6 [プロトコル（Protocols）] ウィンドウの [ポリシー（Policy）] 領域で、[デフォルト（default）] をクリックし、[次（Next）] をクリックします。

[外部 EPG（External EPG）] ウィンドウが表示されます。

ステップ 7 [外部 EPG（External EPG）] ウィンドウで次のアクションを実行します。

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) [すべての外部ネットワークのデフォルト EPG（Default EPG for all external network）]フィールドをオフにします。

   [サブネット（Subnets）] 領域が表示されます。

c) [+] をクリックして [サブネットの作成（Create Subnet）] ダイアログ ボックスにアクセスします。
d) [サブネットの作成（Create Subnet）] ダイアログ ボックスで、[IP アドレス（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。
e) [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。
f) [外部 EPG（External EPG）] ダイアログ ボックスで、[完了（Finish）]をクリックします。

   （注） [作業（Work）] ペインの [L3Outs] 領域に、[L3Out] アイコン（RtdOut）が表示されます。

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
```

テナント、VRF、およびブリッジ ドメインの作成

テナントの概要

- テナントには、承認されたユーザのドメインベースのアクセス コントロールをイネーブルにするポリシーが含まれます。承認されたユーザは、テナント管理やネットワーキング管理などの権限にアクセスできます。
- ユーザは、ドメイン内のポリシーにアクセスしたりポリシーを設定するには読み取り/書き込み権限が必要です。テナント ユーザは、1 つ以上のドメインに特定の権限を持つことができます。

- マルチテナント環境では、リソースがそれぞれ分離されるように、テナントによりグループユーザのアクセス権限が提供されます(エンドポイント グループやネットワーキングなどのため)。これらの権限では、異なるユーザが異なるテナントを管理することもできます。

テナントの作成

テナントには、最初にテナントを作成した後に作成できるフィルタ、契約、ブリッジ ドメイン、およびアプリケーション プロファイルなどのプライマリ要素が含まれます。

VRF およびブリッジ ドメイン

テナントの VRF およびブリッジ ドメインを作成および指定できます。定義されたブリッジ ドメイン要素のサブネットは、対応するレイヤ 3 コンテキストを参照します。

IPv6 ネイバー探索を有効にする方法については、『*Cisco APIC Layer 3 Networking Guide*』の「*IPv6 and Neighbor Discovery*」を参照してください。

GUI を使用したテナント、VRF およびブリッジ ドメインの作成

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。

手順

ステップ 1 メニュー バーで、**[テナント（Tenants）]** > **[テナントの追加（Add Tenant）]** を選択します。

ステップ 2 [Create Tenant] 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、名前を入力します。

b) **[セキュリティ ドメイン（Security Domains）]** セクションで、**[+]** をクリックして、**[セキュリティ ドメインの作成（Create Security Domain）]** ダイアログ ボックスを開きます。

c) **[名前（Name）]** フィールドに、セキュリティ ドメインの名前を入力し、**[送信（Submit）]** をクリックします。

d) **[テナントの作成（Create Tenant）]** ダイアログ ボックスで、作成したセキュリティ ドメインの **[更新（Update）]** をクリックします。

e) 必要に応じて他のフィールドに入力します。

f) [送信（Submit）] をクリックします。

テナント名 > **[ネットワーキング（Networking）]**画面が表示されます。

ステップ 3 **[作業（Work）]** ペインで、**[VRF]** アイコンをキャンバスにドラッグして **[Create VRF]** ダイアログボックスを開き、次の操作を実行します。

a) [Name] フィールドに、名前を入力します。

b) 必要に応じて他のフィールドに入力します。
c) **[送信（Submit）]** をクリックして VRF インスタンスの設定を完了します。

ステップ 4 **[作業（Work）]** ペインで、VRF インスタンスを囲む円内のキャンバスに **[ブリッジ ドメイン（Brdige Domain）]** アイコンをドラッグして、2 つを接続します。[Create Bridge Domain] ダイアログボックスが表示されたら、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) 必要に応じて他のフィールドに入力します。
c) [次へ（Next）] をクリックします。
d) **[サブネット（Subnets）]** セクションで、**[+]** をクリックして、**[サブネットの作成（Create Subnet）]** ダイアログ ボックスを開きます。
e) **[ゲートウェイ IP**（Gateway IP）] フィールドに、IP アドレスとサブネット マスクを入力します。
f) 必要に応じて他のフィールドに入力します。
g) [OK] をクリックします。
h) **[ブリッジ ドメインの作成（Create Bridge Domain）]** ダイアログ ボックスに戻り、必要に応じて他のフィールドに入力します。
i) [次へ（Next）] をクリックします。
j) 必要に応じてフィールドに入力します。
k) **[OK]** をクリックしてブリッジ ドメインの設定を完了します。

ステップ 5 **[作業（Work）]** ペインで、VRF インスタンスを囲む円内のキャンバスに **[L3]** アイコンをドラッグして、2 つを接続します。[Create Routed Outside] ダイアログボックスが表示されたら、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) **[ノードとインターフェイス プロトコル プロファイル（Nodes And Interfaces Protocol Profiles]** セクションで、**[+]** をクリックして **[ノード プロファイルの作成（Create Node Profile）]** ダイアログ ボックスを開きます。
c) [Name] フィールドに、名前を入力します。
d) **[ノード（Nodes）]** セクションで、**[+]** をクリックして **[ノードの選択（Select Node）]** ダイアログ ボックスを開きます。
e) **[ノード ID（Node ID）]** ドロップダウン リストから、ノードを選択します。
f) [Router ID] フィールドに、ルータ ID を入力します。
g) **[スタティック ルート（Static Routes）]** セクションで、**[+]** をクリックして**[スタティック ルートの作成（Create Static Routes）]** ダイアログ ボックスを開きます。
h) [Prefix] フィールドに、IPv4 アドレスまたは IPv6 アドレスを入力します。
i) **[ネクスト ホップ アドレス（Next Hop Addresses）]** セクションで、**[+]** をクリックして **[ネクスト ホップの作成（Create Next Hop）]** ダイアログ ボックスを開きます。
j) **[ネクスト ホップ アドレス（Next Hop Addresses）]** フィールドを展開し、IPv4 アドレスまたは IPv6 アドレスを入力します。
k) **[設定（Preference）]** フィールドに、数値を入力します。
l) 必要に応じて他のフィールドに入力します。
m) [OK] をクリックします。

n) **[静的ルートの作成（Create Static Route）]** ダイアログ ボックスで、必要に応じて他のフィールドに入力します。
o) [OK] をクリックします。
p) **[ノードの選択（Select Node）]** ダイアログ ボックスで、必要に応じて他のフィールドに入力します。
q) [OK] をクリックします。
r) **[ノード プロファイルの作成（Create Node Profile）]** ダイアログ ボックスで、必要に応じて他のフィールドに入力します。
s) [OK] をクリックします。
t) 必要に応じて **[BGP]**、**[OSPF]**、または **[EIGRP]** チェックボックスをオンにします。
u) 必要に応じて他のフィールドに入力します。
v) [次へ（Next）] をクリックします。
w) 必要に応じてフィールドに入力します。
x) **[OK]** をクリックしてレイヤ 3 の設定を完了します。

レイヤ 3 の設定を確認するには、**[ナビゲーション（Navigation）]** ペインで、**[ネットワーキング（Networking）]** > **[VRF]**の順に展開します。

EPG の導入

特定のポートへの EPG の静的な導入

このトピックでは、Cisco APIC を使用しているときに特定のポートに EPG を静的に導入する一般的な方法の例を示します。

GUI を使用して特定のノードまたはポートへ EPG を導入する

始める前に

EPG を導入するテナントがすでに作成されていること。

特定のノードまたはノードの特定のポートで、EPG を作成することができます。

手順

ステップ 1 Cisco APIC にログインします。
ステップ 2 **[Tenants[** > ***[tenant]*** を選択します。
ステップ 3 左側のナビゲーション ウィンドウで、 ***tenant***、 **Application Profiles**、および ***application profile*** を展開します。
ステップ 4 **Application EPGs** を右クリックし、**Create Application EPG** を選択します。

ステップ 5 **Create Application EPG STEP 1 > Identity** 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、EPG の名前を入力します。
b) **Bridge Domain** ドロップダウンリストから、ブリッジ ドメインを選択します。
c) [Statically Link with Leaves/Paths] チェックボックスをオンにします。

このチェック ボックスを使用して、どのポートに EPG を導入するかを指定できます。

d) [Next] をクリックします。
e) [Path] ドロップダウンリストから、宛先 EPG への静的パスを選択します。

ステップ 6 **Create Application EPG STEP 2 > Leaves/Paths** ダイアログボックスで、**Physical Domain** ドロップダウンリストから物理ドメインを選択します。

ステップ 7 次のいずれかの手順を実行します。

オプション	説明
次のものに EPG を展開する場合、	次を実行します。
ノード	1. **Leaves** エリアを展開します。 2. [Node] ドロップダウン リストから、ノードを選択します。 3. **Encap** フィールドで、適切な VLAN を入力します。 4. (オプション) **Deployment Immediacy** ドロップダウンリストで、デフォルトの **On Demand** のままにするか、**Immediate**を選択します。 5. (オプション) [Mode] ドロップダウンリストで、デフォルトの [Trunk] のままにするか、別のモードを選択します。
ノード上のポート	1. **Paths** エリアを展開します。 2. **Path** ドロップダウンリストから、適切なノードおよびポートを選択します。 3. (オプション) **Deployment Immediacy** フィールドのドロップダウンリストで、デフォルトの **On Demand** のままにするか、**Immediate** を選択します。 4. (オプション) [Mode] ドロップダウンリストで、デフォルトの [Trunk] のままにするか、別のモードを選択します。 5. **Port Encap** フィールドに、導入するセカンダリ VLAN を入力します。 6. (オプション) **Primary Encap** フィールドで、展開するプライマリ VLAN を入力します。

ステップ 8 **Update** をクリックし、**Finish** をクリックします。

ステップ 9 左側のナビゲーション ウィンドウで、作成した EPG を展開します。

ステップ 10 次のいずれかの操作を実行します:

- ノードで EPG を作成した場合は、**Static Leafs** をクリックし、作業ウィンドウで、静的バインド パスの詳細を表示します。
- ノードのポートでEPGを作成した場合は、**Static Ports**をクリックし、作業ウィンドウで、静的バインド パスの詳細を表示します。

特定のポートにEPGを導入するためのドメイン、接続エンティティ プロファイル、および VLAN の作成

このトピックでは、特定のポートに EPG を導入する場合に必須である物理ドメイン、接続エンティティ プロファイル（AEP）、および VLAN を作成する方法の典型的な例を示します。

すべてのエンドポイント グループ（EPG）にドメインが必要です。また、インターフェイス ポリシー グループを接続エンティティ プロファイル（AEP）に関連付ける必要があり、AEP と EPG が同じドメインに存在する必要がある場合は、AEP をドメインに関連付ける必要があります。EPG とドメイン、およびインターフェイス ポリシー グループとドメインの関連付けに基づいて、EPG が使用するポートと VLAN が検証されます。以下のドメイン タイプが EPG に関連付けられます。

- アプリケーション EPG
- レイヤ 3 Outside 外部ネットワーク インスタンス EPG
- レイヤ 2 Outside 外部ネットワーク インスタンス EPG
- アウトオブバンドおよびインバンド アクセスの管理 EPG

APIC は、これらのドメイン タイプのうち 1 つまたは複数に EPG が関連付けられているかどうかを確認します。EPG が関連付けられていない場合、システムは設定を受け入れますが、エラーが発生します。ドメインの関連付けが有効でない場合、導入された設定が正しく機能しない可能性があります。たとえば、VLAN のカプセル化を EPG で使用することが有効でない場合、導入された設定が正しく機能しない可能性があります。

（注） スタティック バインディングを使用しない AEP との EPG アソシエーションは、一方のエンドポイントが同じ EPG の下でタギングをサポートし、もう一方のエンド ポイントが同じ EPG 内で VLAN タギングをサポートしないような AEP の下では、EPG を**トランク**として設定するシナリオで機能させることはできません。EPG で AEPを関連付ける際には、トランク、アクセス（タグ付き）、またはアクセス（タグなし）として設定できます。

GUI を使用して特定のポートに EPG を展開するためのドメインおよび VLAN の作成

始める前に

- EPG を導入するテナントがすでに作成されていること。
- EPG は特定のポートに静的に導入されます。

手順

ステップ 1 メニュー バーで、**[ファブリック（FABRIC）]** > **[アクセス ポリシー（Access Policies）]** の順に選択します。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[クイックスタート（Quick Start）]** をクリックします。

ステップ 3 **[作業（Work）]** ペインで、**[インターフェイスの設定（Configure Interfaces）]** をクリックします。

ステップ 4 **[インターフェイスの設定（Configure Interfaces）]** ダイアログで、以下のアクションを実行します。

a) **[ノード タイプ（Node Type）]** で、**[リーフ（Leaf）]** をクリックします。
b) **[ポート タイプ（Port Type）]** で、**[アクセス（Access）]** をクリックします。
c) **[インターフェイス タイプ（Interface Type）]** で、目的のタイプを選択します。
d) **[インターフェイス集約タイプ（Interface Aggregation Type）]** で、**[個別（Individual）]** を選択します。
e) **[ノード（Node）]** で、**[ノードの選択（Select Node）]** をクリックし、目的のノードのボックスにチェックを入れて、**[OK]** をクリックします。複数のノードを選択できます。
f) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。
g) **[リーフ アクセス ポート ポリシー グループ（Leaf Access Port Policy Group）]** の場合は、**[リーフ アクセス ポート ポリシー グループの選択（Select Leaf Access Port Policy Group）]** をクリックします。
h) **[リーフ アクセス ポート ポリシー グループの選択（Select Leaf Access Port Policy Group）]** ダイアログで、**[リーフ アクセス ポート ポリシー グループの作成（Create Leaf Access Port Policy Group）]** をクリックします。
i) **[リーフ アクセス ポート ポリシー グループの作成（Create Leaf Access Port Policy Group）]** ダイアログの **[リンク レベル ポリシー（Link Level Policy）]** で、**[リンク レベル ポリシーの選択（Select Link Level Policy）]** をクリックします。
j) リンク レベル ポリシーを選択して **[選択（Select）]** を選択するか、**[リンク レベル ポリシーの作成（Create Link Level Policy）]** をクリックし、必要に応じてフィールドに入力して、**[保存（Save）]** をクリックします。
k) [保存（Save）] をクリックします。

ステップ 5 以下のアクションを実行して、ドメインと VLAN プールを作成します。

a) **[ナビゲーション（Navigation）]** ペインで、**[物理ドメインと外部ドメイン（Physical and External Domains）]** を展開します。
b) **[物理ドメイン（Physical Domains）]** を右クリックし、適切な**[物理ドメインの作成（Create Physical Domain）]** を選択します。
c) **[名前（Name）]** に、ドメインの名前を入力します。
d) **[VLAN プール（VLAN Pool）]**で、**[VLAN プールの作成（Create VLAN Pool）]** を選択し、必要に応じてフィールドに入力して、**[送信（Submit）]** をクリックします。
e) 目的に応じて、残りのフィールドに入力します。
f) [送信（Submit）] をクリックします。

ステップ 6 メニュー バーで、**[テナント（Tenants）]**> > **[すべてのテナント（ALL Tenants）]** の順に選択します。

ステップ 7 **[作業（Work）]** ペインで、目的のテナントをダブルクリックします。

ステップ 8 **[ナビゲーション（Navigation）]** ペインで、**テナント名** > **[アプリケーション プロファイル（Application Profiles）]** > **プロファイル名** > **[アプリケーション EPG（Application EPGs）]** > ***EPG* 名**を展開し、以下の操作を実行します。

a) **[ドメイン（Domains）（VM またはベアメタル）]** を右クリックし、**[物理ドメインの関連付けの追加（Add Physical Domain Association）]** をクリックします。
b) **[物理ドメインの関連付けの追加（Add Physical Domain Association）]** ダイアログで、**[物理ドメインのプロファイル（Physical Domain Profile）]** ドロップダウン リストから、前に作成したドメインを選択します。
c) [Submit] をクリックします。
AEP は、ノード上の特定のポート、およびドメインに関連付けられます。物理ドメインは VLAN プールに関連付けられ、テナントはこの物理ドメインに関連付けられます。

スイッチ プロファイルとインターフェイス プロファイルが作成されます。インターフェイス プロファイルのポート ブロックにポリシー グループが作成されます。AEP が自動的に作成され、ポート ブロックおよびドメインに関連付けられます。 ドメインは VLAN プールに関連付けられ、テナントはドメインに関連付けられます。

AEP またはインターフェイス ポリシー グループを使用したアプリケーション EPG の複数のポートへの導入

APIC の拡張 GUI と REST API を使用して、接続エンティティ プロファイルをアプリケーション EPG に直接関連付けることができます。これにより、単一の構成の接続エンティティ プロファイルに関連付けられたすべてのポートに、関連付けられたアプリケーション EPG を導入します。

APIC REST API または NX-OS スタイルの CLI を使用し、インターフェイス ポリシー グループを介して複数のポートにアプリケーション EPG を導入できます。

APIC GUI を使用した AEP による複数のインターフェイスへの EPG の導入

短時間でアプリケーションを接続エンティティプロファイルに関連付けて、その接続エンティティプロファイルに関連付けられたすべてのポートにEPGを迅速に導入することができます。

始める前に

- ターゲット アプリケーション EPG が作成されている。
- AEP での EPG 導入に使用する VLAN の範囲が含まれている VLAN プールが作成されている。
- 物理ドメインが作成され、VLAN プールと AEP にリンクされている。
- ターゲットの接続エンティティ プロファイルが作成され、アプリケーション EPG を導入するポートに関連付けられている。

手順

ステップ 1 ターゲットの接続エンティティ プロファイルに移動します。

a) 使用する接続エンティティ プロファイルのページを開きます。**[ファブリック（Fabric）]**>**[アクセス ポリシー（Access Policies）]**>**[ポリシー（Policies）]**>**[グローバル（Global）]**>**[アタッチ可能なアクセス エンティティ プロファイル（Attachable Access Entity Profiles）]** に移動します。

b) ターゲットの接続エンティティ プロファイルをクリックして、[Attachable Access Entity Profile] ウィンドウを開きます。

ステップ 2 [Show Usage] ボタンをクリックして、この接続エンティティ プロファイルに関連付けられたリーフ スイッチとインターフェイスを表示します。

この接続エンティティ プロファイルに関連付けられたアプリケーション EPG が、この接続エンティティプロファイルに関連付けられたすべてのスイッチ上のすべてのポートに導入されます。

ステップ 3 [Application EPGs] テーブルを使用して、この接続エンティティ プロファイルにターゲット アプリケーション EPG を関連付けます。アプリケーション EPG エントリを追加するには、[+] をクリックします。各エントリに次のフィールドがあります。

フィールド	アクション（Action）
Application EPG	ドロップダウンを使用して、関連付けられたテナント、アプリケーション プロファイル、およびターゲット アプリケーション EPG を選択します。
Encap	ターゲット アプリケーション EPG の通信に使用される VLAN の名前を入力します。

フィールド	アクション（**Action**）
Primary Encap	アプリケーション EPG にプライマリ VLAN が必要な場合は、プライマリ VLAN の名前を入力します。
モード	ドロップダウンを使用して、データを送信するモードを指定します。 • [Trunk]：ホストからのトラフィックに VLAN ID がタグ付けされている場合に選択します。 • [Access]：ホストからのトラフィックに 802.1p タグがタグ付けされている場合に選択します。 • [Access Untagged]：ホストからのトラフィックがタグ付けされていない場合に選択します。

ステップ 4 [Submit] をクリックします。
この接続エンティティ プロファイルに関連付けられたアプリケーション EPG が、この接続エンティティ プロファイルに関連付けられたすべてのスイッチ上のすべてのポートに導入されます。

マイクロセグメント EPG

ベア メタルでのネットワークベースの属性によるマイクロセグメンテーションの使用

Cisco APIC を使用して Cisco ACI でのマイクロセグメンテーションを設定し、ネットワークベースの属性、MAC アドレス、または 1 つ以上の IP アドレスを使用した新しい属性ベースの EPG を作成できます。ネットワークベースの属性を使用して Cisco ACI でのマイクロセグメンテーションを設定し、単一のベース EPG または複数の EPG 内で VM または物理エンドポイントを分離できます。

IP ベースの属性の使用

IP ベースのフィルタを使用して、単一のマイクロセグメントで単一 IP アドレス、サブネット、または多様な非連続 IP アドレスを分離できます。ファイアウォールの使用と同様に、セキュリティ ゾーンを作成するための迅速かつ簡単な方法として、IP アドレスに基づいて物理エンドポイントを分離できます。

MAC ベースの属性の使用

MAC ベースのフィルタを使用して、単一 MAC アドレスまたは複数の MAC アドレスを分離できます。不適切なトラフィックをネットワークに送信するサーバがある場合はこの方法を推奨

します。MAC ベースのフィルタを使用してマイクロセグメントを作成することで、このサーバを分離できます。

GUI を使用したベアメタル環境でのネットワークベースのマイクロセグメント EPG の設定

Cisco APIC を使用してマイクロセグメンテーションを設定し、異なる複数のベース EPG または同一の EPG に属する物理エンドポイント デバイスを新しい属性ベースの EPG に配置できます。

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 [Tenants] を選択し、マイクロセグメントを作成するテナントを選択します。

ステップ 3 テナントのナビゲーション ウィンドウで、テナント フォルダ、[Application Profiles] フォルダ、[Profile] フォルダ、および [Application EPGs] フォルダを展開します。

ステップ 4 次のいずれかを実行します。

- 同じベース EPG の物理エンドポイント デバイスを新しい属性ベースの EPG に配置するには、物理エンドポイント デバイスを含むベース EPG をクリックします。
- 異なる複数のベース EPG の物理エンドポイント デバイスを新しい属性ベースの EPG に配置するには、物理エンドポイント デバイスを含むベース EPG の 1 つをクリックします。

ベース EPG のプロパティが作業ウィンドウに表示されます。

ステップ 5 作業ウィンドウで、画面の右上にある [Operational] タブをクリックします。

ステップ 6 [Operational] タブの下の [Client End-Points] タブがアクティブになっていることを確認します。作業ウィンドウに、ベース EPG に属するすべての物理エンドポイントが表示されます。

ステップ 7 新しいマイクロセグメントに配置するエンドポイント デバイス（複数可）の IP アドレスまたは MAC アドレスを書き留めます。

ステップ 8 異なる複数のベース EPG のエンドポイント デバイスを新しい属性ベースの EPG に配置する場合は、各ベース EPG に対してステップ 4 ～ 7 を繰り返します。

ステップ 9 テナントのナビゲーション ウィンドウで、[uSeg EPGs] フォルダを右クリックし、[Create uSeg EPG] を選択します。

ステップ 10 以下の一連の手順を実行し、エンドポイント デバイス グループの 1 つに対して属性ベースの EPG の作成を開始します。

a) [Create uSeg EPG] 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。

新しい属性ベースの EPG はマイクロセグメントであることを示す名前を選択することを推奨します。

b) [intra-EPG isolation] フィールドで [enforced] または [unenforced] を選択します。

[enforced] を選択した場合は、ACI によってこの uSeg EPG 内のエンドポイント デバイス間の通信がすべて阻止されます。

c) [Bridge Domain] エリアで、ドロップダウン リストからブリッジ ドメインを選択します。
d) [uSeg Attributes] 領域で、ダイアログボックスの右側にある [+] ドロップダウン リストから [IP Address Filter] または [MAC Address Filter] を選択します。

ステップ 11 フィルタを設定するには、次のいずれかの一連の手順を実行します。

項目	結果
IP ベースの属性	**1.** [Create IP Attribute] 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。 名前については、フィルタ機能を反映したものを選択するよう推奨します。 **2.** [IP Address] フィールドに、適切なサブネット マスクの IP アドレスまたはサブネットを入力します。 **3.** [OK] をクリックします。 **4.** （オプション）ステップ 10 c ～ 11 c を繰り返して、2 番目の IP アドレス フィルタを作成します。 この手順で、マイクロセグメントに不連続の IP アドレスを含めることができます。 **5.** [Create uSeg EPG] ダイアログボックスで、[Submit] をクリックします。
MAC ベースの属性	**1.** [Create MAC Attribute] 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。 名前については、フィルタ機能を反映したものを選択するよう推奨します。 **2.** [MAC Address] フィールドに、MAC アドレスを入力します。 **3.** [OK] をクリックします。 **4.** [Create uSeg EPG] ダイアログボックスで、[Submit] をクリックします。

ステップ 12 次の手順を実行して uSeg EPG を物理ドメインに関連付けます。

a) [Navigation] ペインで、uSeg EPG フォルダが開いていることを確認し、作成したマイクロセグメントのコンテナを開きます。
b) [Domains (VMs and Bare-Metals)] フォルダをクリックします。
c) 作業ウィンドウの右側にある [Actions] をクリックし、ドロップダウンリストから [Add Physical Domain Association] を選択します。
d) [Add Physical Domain Association] ダイアログボックスで、[Physical Domain Profile] ドロップダウン リストからプロファイルを選択します。
e) [Deploy Immediacy] エリアで、デフォルトの [On Demand] を受け入れます。
f) [Resolution Immediacy] エリアで、デフォルトの [Immediate] を受け入れます。
g) [Submit] をクリックします。`

ステップ 13 uSeg EPG を適切なリーフ スイッチに関連付けます。

a) ナビゲーション ウィンドウで、uSeg EPG フォルダが開いていることを確認して [Static Leafs] をクリックします。
b) [Static Leafs] ウィンドウで、**[Actions]** > **[Statically Link with Node]** をクリックします
c) [Statically Link With Node] ダイアログで、リーフ ノードとモードを選択します。
d) **Submit** をクリックします。

ステップ 14 作成するその他のネットワーク属性ベースの EPG すべてに対してステップ 9 ～ 13 を繰り返します。

次のタスク

属性ベースの EPG が正しく作成されたことを確認します。

IP ベースまたは MAC ベースの属性を設定する場合は、新しいマイクロセグメントに配置したエンドポイント デバイスでトラフィックが動作していることを確認します。

共有リソースとしての IP アドレスベースのマイクロセグメント EPG

IP アドレスベースのマイクロセグメント EPG を VRF（この EPG が配置されている）の内外からアクセスできるリソースとして設定できます。この場合は、既存の IP アドレスベースのマイクロセグメント EPG にサブネット（ユニキャスト IP アドレスが割り当てられている）を設定し、そのサブネットをこの EPG が属する VRF 以外の VRF にあるデバイスでアドバタイズおよび共有できるようにします。次に、EPG を共有サブネットの IP アドレスに関連付けるオプションを有効にした状態で IP 属性を定義します。

GUI を使用した共有リソースとしての IP ベースのマイクロセグメント EPG の設定

VRF および現在のファブリック外のクライアントがアクセス可能な共有サービスとして、32 ビット マスクの IP アドレスを持つマイクロセグメント EPG を設定できます。

始める前に

設定に関する次の GUI の説明では、サブネット マスクが /32 に設定された IP アドレスベースのマイクロセグメント EPG が事前設定されていることを前提としています。

（注）
- 物理環境で IP アドレス ベースの EPG を設定する手順については、次を参照してください。 ベア メタルでのネットワークベースの属性によるマイクロセグメンテーションの使用 （24 ページ）
- 仮想環境で IP アドレス ベースの EPG を設定する手順については、『*Cisco ACI Virtualization Guide*』の「*Configuring Microsegmentation with Cisco ACI*」を参照してください。

手順

ステップ 1 ターゲットとなる IP アドレスベースの EPG に移動します。

a) APIC GUI で、**[Tenant]** > **[tenant_name]** > **[uSeg EPGs]** > **[uSeg_epg_name]** をクリックして EPG の [Properties] ダイアログを表示します。

ステップ 2 ターゲット EPG では、EPG のサブネット アドレスに一致するように IP 属性を設定します。

a) [Properties] ダイアログで、[uSeg Attributes] テーブルを見つけて [+] をクリックします。プロンプトが表示されたら、**[IP Address Filter]** を選択して **[Create IP Attribute]** ダイアログを表示します。

b) [Name] フィールドに名前を入力します。

c) **[Use FV Subnet]** のチェックボックスをオンにします。

このオプションを有効にすることで、IP 属性値が共有サブネットの IP アドレスに一致することを示します。

d) **[Submit]** をクリックします。

ステップ 3 ターゲット EPG の共有サブネットを作成します。

a) ターゲットとなる IP アドレスベースの uSeg EPG のフォルダを APIC のナビゲーション ウィンドウで開いたまま、**[Subnets]** フォルダを右クリックして **[Create EPG Subnets]** を選択します。

b) **[Default Gateway]** フィールドに、IP アドレスベースのマイクロセグメント EPG の IP アドレスまたはマスクを入力します。

（注）
- いずれの場合もサブネット マスクは /32 である必要があります。
- IP アドレスベースの EPG に関しては、実際にゲートウェイのデフォルト アドレスを入力するのではなく、共有 EPG サブネットの IP アドレスを入力します。

c) **[Treat as a virtual IP address]** を選択します。

d) [Scope] で [Advertised Externally] と [Shared between VRFs] を選択します。

e) [Submit] をクリックします。

GUI を使用した共有リソースとしての IP ベースのマイクロセグメント EPG の設定解除

共有サービスとして設定された IP アドレスベースのマイクロセグメント EPG を設定解除するには、共有サブネットを削除し、さらにそのサブネットを共有リソースとして使用するオプションを無効にする必要があります。

始める前に

共有サービスとして設定された IP アドレスベースのマイクロセグメント EPG を設定解除するには、次の情報を確認しておく必要があります。

- IP アドレスベースのマイクロセグメント EPG の共有サービス アドレスとして設定されているサブネット。
- **Use FV Subnet** オプションが有効な状態で設定されている IP 属性。

手順

ステップ 1 IP アドレスベースのマイクロセグメント EPG からサブネットを削除します。

a) APIC GUI で、**[Tenant]** > **[tenant_name]** > **[Application Profiles]** > **[epg_name]** > **[uSeg EPGs]** > **[uSeg EPGs]** > **[uSeg_epg_name]** をクリックします。
b) ターゲットとなる IP アドレスベースの uSeg EPG のフォルダを APIC のナビゲーション ウィンドウで開いたまま、[Subnets] フォルダをクリックします。
c) **Subnets** ウィンドウで、アドバタイズされて他の VRF と共有されるサブネットを選択し、**Actions** > **Delete** をクリックします。
d) [Yes] をクリックして削除を確定します。

ステップ 2 [Use FV Subnet] オプションを無効にします。

a) ターゲットとなる IP アドレスベースの uSeg EPG のフォルダを APIC のナビゲーション ウィンドウで開いたまま、マイクロセグメント EPG の名前をクリックして EPG の [Properties] ダイアログを表示します。
b) [Properties] ダイアログで、[uSeg Attributes] テーブルから [Use FV Subnet] オプションが有効になっている IP 属性の項目を見つけます。
c) その項目をダブルクリックして **Edit IP Attribute** ダイアログを表示します。
d) [Edit IP Attribute] ダイアログで、[Use FV Subnet] オプションを選択解除します。
e) [IP Address] フィールドに別の IP アドレス属性を指定します。

（注） このアドレスは、32 ビット マスクのユニキャスト アドレスである必要があります（例：124.124.124.123/32）。

f) [Submit] をクリックします。

アプリケーション プロファイルと契約の導入

セキュリティ ポリシーの適用

トラフィックは前面パネルのインターフェイスからリーフ スイッチに入り、パケットは送信元 EPG の EPG でマーキングされます。リーフ スイッチはその後、テナント エリア内のパケット

の宛先 IP アドレスでフォワーディング ルックアップを実行します。ヒットすると、次のシナリオのいずれかが発生する可能性があります。

1. ユニキャスト（/32）ヒットでは、宛先エンドポイントの EPG と宛先エンドポイントが存在するローカル インターフェイスまたはリモート リーフ スイッチの VTEP IP アドレスが提供されます。
2. サブネット プレフィクス（/32 以外）のユニキャスト ヒットでは、宛先サブネット プレフィクスの EPG と宛先サブネット プレフィクスが存在するローカル インターフェイスまたはリモート リーフ スイッチの VTEP IP アドレスが提供されます。
3. マルチキャスト ヒットでは、ファブリック全体の VXLAN カプセル化とマルチキャスト グループの EPG で使用するローカル レシーバのローカル インターフェイスと外側の宛先 IP アドレスが提供されます。

（注） マルチキャストと外部ルータのサブネットは、入力リーフ スイッチでのヒットを常にもたらします。セキュリティ ポリシーの適用は、宛先 EPG が入力リーフ スイッチによって認識されるとすぐに発生します。

転送テーブルの誤りにより、パケットがスパイン スイッチの転送プロキシに送信されます。転送プロキシはその後、転送テーブル検索を実行します。これが誤りである場合、パケットはドロップされます。これがヒットの場合、パケットは宛先エンドポイントを含む出力リーフ スイッチに送信されます。出力リーフ スイッチが宛先の EPG を認識するため、セキュリティ ポリシーの適用が実行されます。出力リーフ スイッチは、パケット送信元の EPG を認識する必要があります。ファブリック ヘッダーは、入力リーフ スイッチから出力リーフ スイッチに EPG を伝送するため、このプロセスをイネーブルにします。スパイン スイッチは、転送プロキシ機能を実行するときに、パケット内の元の EPG を保存します。

出力リーフ スイッチでは、送信元 IP アドレス、送信元 VTEP、および送信元 EPG 情報は、学習によってローカルの転送テーブルに保存されます。ほとんどのフローが双方向であるため、応答パケットがフローの両側で転送テーブルに入力し、トラフィックが両方向で入力フィルタリングされます。

セキュリティ ポリシー仕様を含むコントラクト

ACI セキュリティ モデルでは、コントラクトに EPG 間の通信を管理するポリシーが含まれます。コントラクトは通信内容を指定し、EPG は通信の送信元と宛先を指定します。コントラクトは次のように EPG をリンクします。

EPG 1 ---------- コントラクト ---------- EPG 2

コントラクトで許可されていれば、EPG 1 のエンドポイントは EPG 2 のエンドポイントと通信でき、またその逆も可能です。このポリシーの構造には非常に柔軟性があります。たとえば、EPG 1 と EPG 2 間には多くのコントラクトが存在でき、1 つのコントラクトを使用する EPG が 3 つ以上存在でき、コントラクトは複数の EPG のセットで再利用できます。

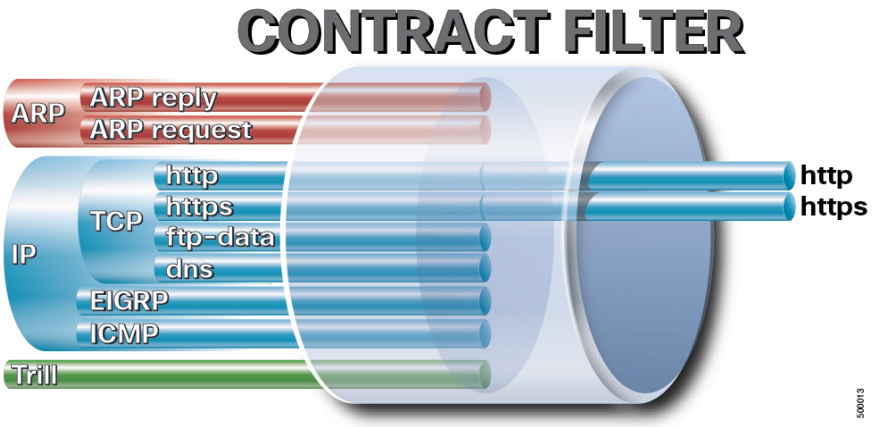
また EPG とコントラクトの関係には方向性があります。EPG はコントラクトを提供または消費できます。コントラクトを提供する EPG は通常、一連のクライアント デバイスにサービスを提供する一連のエンドポイントです。そのサービスによって使用されるプロトコルはコントラクトで定義されます。コントラクトを消費する EPG は通常、そのサービスのクライアントである一連のエンドポイントです。クライアント エンドポイント (コンシューマ) がサーバ エンドポイント (プロバイダー) に接続しようとすると、コントラクトはその接続が許可されるかどうかを確認します。特に指定のない限り、そのコントラクトは、サーバがクライアントへの接続を開始することを許可しません。ただし、EPG 間の別のコントラクトが、その方向の接続を簡単に許可する場合があります。

この提供/消費の関係は通常、EPG とコントラクト間を矢印を使って図で表されます。次に示す矢印の方向に注目してください。

EPG 1 <------- 消費 -------- コントラクト <------- 提供 -------- EPG 2

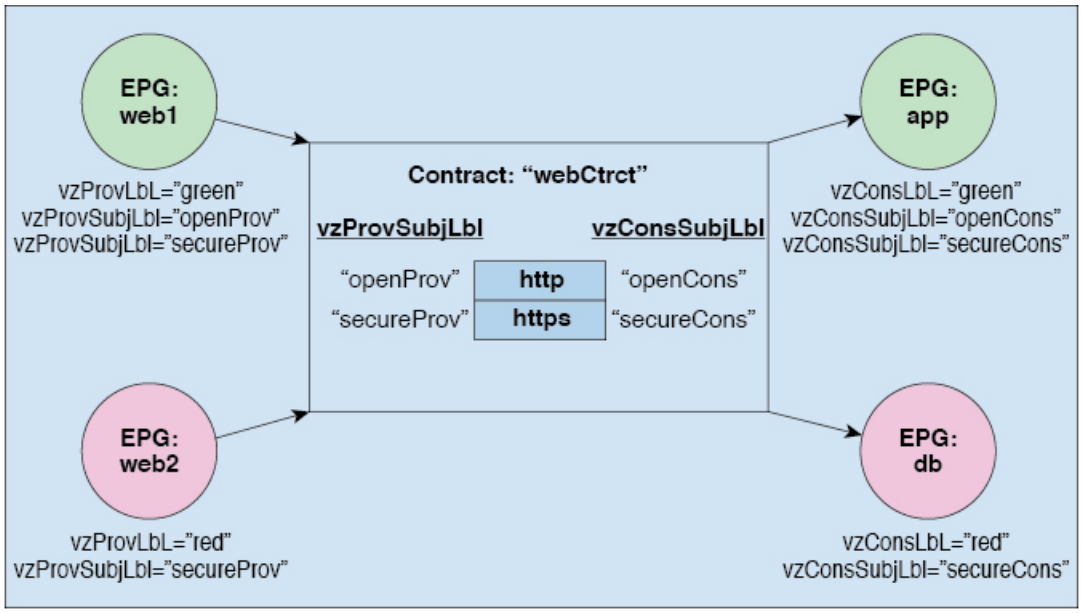
コントラクトは階層的に構築されます。1 つ以上のサブジェクトで構成され、各サブジェクトには 1 つ以上のフィルタが含まれ、各フィルタは 1 つ以上のプロトコルを定義できます。

図 *6:* コントラクト フィルタ



次の図は、コントラクトが EPG の通信をどのように管理するかを示します。

図 *7: EPG/EPG* 通信を決定するコントラクト



たとえば、TCP ポート 80 とポート 8080 を指定する HTTP と呼ばれるフィルタと、TCP ポート 443 を指定する HTTPS と呼ばれる別のフィルタを定義できます。その後、2 セットの情報カテゴリを持つ webCtrct と呼ばれるコントラクトを作成できます。openProv と openCons are が HTTP フィルタが含まれる情報カテゴリです。secureProv と secureCons は HTTPS フィルタが含まれる情報カテゴリです。この webCtrct コントラクトは、Web サービスを提供する EPG とそのサービスを消費するエンドポイントを含む EPG 間のセキュアな Web トラフィックと非セキュアな Web トラフィックの両方を可能にするために使用できます。

これらの同じ構造は、仮想マシンのハイパーバイザを管理するポリシーにも適用されます。EPG が Virtual Machine Manager（VMM）のドメイン内に配置されると、APIC は EPG に関連付けられたすべてのポリシーを VMM ドメインに接続するインターフェイスを持つリーフ スイッチにダウンロードします。VMM ドメインの完全な説明については、『*Application Centric Infrastructure Fundamentals*』の「*Virtual Machine Manager Domains*」の章を参照してください。このポリシーが作成されると、APIC は EPG のエンドポイントへの接続を可能にするスイッチを指定する VMM ドメインにそれをプッシュ（あらかじめ入力）します。VMM ドメインは、EPG 内のエンドポイントが接続できるスイッチとポートのセットを定義します。エンドポイントがオンラインになると、適切な EPG に関連付けられます。パケットが送信されると、送信元 EPG および宛先 EPG がパケットから取得され、対応するコントラクトで定義されたポリシーでパケットが許可されたかどうかが確認されます。許可された場合は、パケットが転送されます。許可されない場合は、パケットはドロップされます。

コントラクトは 1 つ以上のサブジェクトで構成されます。各サブジェクトには 1 つ以上のフィルタが含まれます。各フィルタには 1 つ以上のエントリが含まれます。各エントリは、アクセス コントロール リスト (ACL) の 1 行に相当し、エンドポイント グループ内のエンドポイントが接続されているリーフ スイッチで適用されます。

詳細には、コントラクトは次の項目で構成されます。

- 名前：テナントによって消費されるすべてのコントラクト (common テナントまたはテナント自体で作成されたコントラクトを含む) にそれぞれ異なる名前が必要です。
- サブジェクト：特定のアプリケーションまたはサービス用のフィルタのグループ。
- フィルタ：レイヤ 2 ～ レイヤ 4 の属性 (イーサネット タイプ、プロトコル タイプ、TCP フラグ、ポートなど) に基づいてトラフィックを分類するために使用します。
- アクション：フィルタリングされたトラフィックで実行されるアクション。次のアクションがサポートされます。
  - トラフィックの許可 (通常のコントラクトのみ)
  - トラフィックのマーク (DSCP/CoS) (通常のコントラクトのみ)
  - トラフィックのリダイレクト (サービス グラフによる通常のコントラクトのみ)
  - トラフィックのコピー (サービス グラフまたは SPAN による通常のコントラクトのみ)
  - トラフィックのブロック（禁止コントラクトのみ）

    Cisco APIC リリース 3.2(x) および名前が EX または FX で終わるスイッチでは、標準コントラクトで代わりに件名 [拒否] アクションまたは [コントラクトまたは件名の除外] を使用して、指定のパターンを持つトラフィックをブロックできます。
  - トラフィックのログ（禁止コントラクトと通常のコントラクト）
- エイリアス：(任意) 変更可能なオブジェクト名。オブジェクト名は作成後に変更できませんが、エイリアスは変更できるプロパティです。

このように、コントラクトによって許可や拒否よりも複雑なアクションが可能になります。コントラクトは、所定のサブジェクトに一致するトラフィックをサービスにリダイレクトしたり、コピーしたり、その QoS レベルを変更したりできることを指定可能です。具象モデルでアクセス ポリシーをあらかじめ入力すると、APIC がオフラインまたはアクセスできない場合でも、エンドポイントは移動でき、新しいエンドポイントをオンラインにでき、通信を行うことができます。APIC は、ネットワークの単一の障害発生時点から除外されます。ACI ファブリックにパケットが入力されると同時に、セキュリティ ポリシーがスイッチで実行している具象モデルによって適用されます。

Three-Tier アプリケーションの展開

フィルタは、フィルタを含むコントラクトにより許可または拒否されるデータ プロトコルを指定します。コントラクトには、複数のサブジェクトを含めることができます。情報カテゴリは、単方向または双方向フィルタの作成に使用できます。単方向フィルタは、コンシューマからプロバイダー方向（IN）またはプロバイダーからコンシューマ方向（OUT）のどちらかに対して使用されます。双方向フィルタは、両方の方向で使用されます。これは、再帰的ではありません。

コントラクトは、エンドポイント グループ間 (EPG 間) の通信をイネーブルにするポリシーです。このポリシーは、アプリケーション層間の通信を指定するルールです。コントラクトが

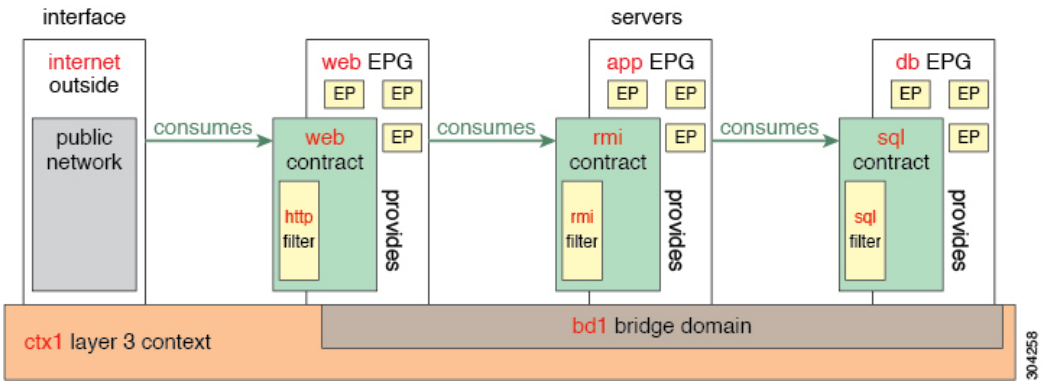
EPG に添付されていない場合は、EPG 間通信がデフォルトで無効になります。EPG 内の通信は常に許可されているので、EPG 内の通信には契約は必要ありません。

アプリケーション プロファイルでは、APIC がその後ネットワークおよびデータセンターのインフラストラクチャで自動的にレンダリングするアプリケーション要件をモデル化することができます。アプリケーションプロファイルでは、管理者がインフラストラクチャの構成要素ではなくアプリケーションの観点から、リソースプールにアプローチすることができます。アプリケーション プロファイルは、互いに論理的に関連する EPG を保持するコンテナです。EPG は同じアプリケーション プロファイル内の他の EPG および他のアプリケーション プロファイル内の EPG と通信できます。

アプリケーション ポリシーを展開するには、必要なアプリケーション プロファイル、フィルタ、および契約を作成する必要があります。通常、APIC ファブリックは、テナント ネットワーク内の Three-Tier アプリケーションをホストします。この例では、アプリケーションは 3 台のサーバ（Web サーバ、アプリケーション サーバ、およびデータベース サーバ）を使用して実行されます。Three-Tier アプリケーションの例については、次の図を参照してください。

Web サーバには HTTP フィルタがあり、アプリケーション サーバには Remote Method Invocation（RMI）フィルタがあり、データベース サーバには Structured Query Language（SQL）フィルタがあります。アプリケーション サーバは、SQL コントラクトを消費してデータベース サーバと通信します。Web サーバは、RMI コントラクトを消費して、アプリケーション サーバと通信します。トラフィックは Web サーバから入り、アプリケーション サーバと通信します。アプリケーション サーバはその後、データベース サーバと通信し、トラフィックは外部に通信することもできます。

図 *8 : Three-Tier* アプリケーションの図



http 用のフィルタを作成するパラメータ

この例での http 用のフィルタを作成するパラメータは次のとおりです。

パラメータ名	http のフィルタ
名前	http
エントリの数	2

パラメータ名	http のフィルタ
エントリ名	Dport-80 Dport-443
Ethertype	IP
プロトコル	tcp tcp
宛先ポート	http https

rmi および sql 用のフィルタを作成するパラメータ

この例での rmi および sql 用のフィルタを作成するパラメータは次のとおりです。

パラメータ名	rmi のフィルタ	sql のフィルタ
名前	rmi	sql
エントリの数	1	1
エントリ名	Dport-1099	Dport-1521
Ethertype	IP	IP
プロトコル	tcp	tcp
宛先ポート	1099	1521

アプリケーション プロファイル データベースの例

この例のアプリケーション プロファイル データベースは次のとおりです。

EPG	提供されるコントラクト	消費されるコントラクト
web	web	rmi
app	rmi	sql
db	sql	--

GUI を使用したアプリケーション プロファイルの作成

手順

ステップ 1 メニュー バーで、**[TENANTS]** を選択します。**[Navigation]** ペインで、テナントを展開し、**[Application Profiles]** を右クリックし、**[Create Application Profile]** をクリックします。

ステップ 2 **[Create Application Profile]** ダイアログボックスで、**[Name]** フィールドに、アプリケーション プロファイル名 (OnlineStore) を追加します。

GUI を使用した EPG の作成

EPG が使用するポートは、VM マネージャ (VMM) ドメインまたは EPG に関連付けられた物理ドメインのいずれか 1 つに属している必要があります。

手順

ステップ 1 メニュー バーで、[Tenants]、EPG を作成するテナントの順に選択します。

ステップ 2 ナビゲーション ペインで、テナントのフォルダ、[Application Profiles] フォルダ、アプリケーション プロファイルのフォルダの順に展開します。

ステップ 3 [Application EPG] フォルダを右クリックし、[Create Application EPG] 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、EPG の名前（db）を追加します。

b) [Bridge Domain] フィールドで、ドロップダウン リストからブリッジ ドメイン（bd1）を選択します。

c) [Associate to VM Domain Profiles] チェックボックスをオンにします。[Next] をクリックします。

d) [STEP 2 > Domains] エリアで、[Associate VM Domain Profiles] を展開し、ドロップダウン リストから対象の VMM ドメインを選択します。

e) [Deployment Immediacy] ドロップダウンリストで、デフォルト値を受け入れるか、いつポリシーが Cisco APIC から物理リーフ スイッチに展開されるかを選択します。

f) [Resolution Immediacy] ドロップダウンリストで、いつポリシーが物理リーフ スイッチから仮想リーフに展開されるかを選択します。

Cisco AVS がある場合には、**Immediate** または **On Demand** を選択します。Cisco ACI Virtual Edge または VMware VDS がある場合には、**Immediate**、 **On Demand**、または **Pre-provision** を選択します。

g) （オプション）[Delimiter] フィールドに、|、~、!、@、^、+、または = のいずれかの記号を入力します。

記号を入力しなかった場合、システムは VMware ポートグループ名のデリミタとしてデフォルトの | を使用します。

h) Cisco ACI Virtual Edge または Cisco AVS を利用している場合は、[Encap Mode] ドロップダウンリストからカプセル化モードを選択します。

次のいずれかのカプセル化モードを選択できます。

- **VXLAN**：これはドメインの VLAN 設定を上書きし、EPG は VXLAN カプセル化を使用します。ただし、ドメインでマルチキャスト プールが設定されていない場合は、EPG に対してエラーが発生します。
- **VLAN**：これはドメインの VXLAN 設定を上書きし、EPG は VLAN カプセル化を使用します。ただし、ドメインで VLAN プールが設定されていない場合は、EPG に対してエラーがトリガーされます。
- **自動**： EPG は、VMM ドメインと同じカプセル化モードを使用します。これはデフォルトの設定です。

i) Cisco ACI Virtual Edge がある場合、**Switching Mode** ドロップダウンリストで、 **native** または **AVE** を選択します。

native を選択した場合、EPGは VMware VDS を通して切り替えられます。**AVE** を選択した場合、EPG はCisco ACI Virtual Edge を通して切り替えられます。デフォルトは**native** す。

j) **Update** をクリックし、**Finish** をクリックします。

ステップ 4 **Create Application Profile** ダイアログボックスで、EPG をさらに 2 つ作成します。同じブリッジ ドメイン、同じデータセンター内に、3 つの EPG を作成します。これらは、db、app、および web です。

APIC GUI を使用したコントラクトの設定

コントラクトとフィルタの注意事項と制約事項

ファブリックが Cisco Nexus 93128TX、93120TX、9396TX、9396PX、9372PX、9372PX-E、9372TX-E などの第 1 世代の Cisco Nexus 9300 リーフ スイッチで構成されている場合、**EtherType** 一致としての **IP** のみがコントラクト フィルタでサポートされます。コントラクト フィルタの **[EtherType]** フィールドで、より詳細なオプション（**IPv4** や **IPv6** など）を照合する機能は、スイッチ名の末尾に -EX、-FX、または -FX2 が指定されたリーフ スイッチ モデルでのみサポートされます。

GUI を使用したフィルタの作成

3 つの個別のフィルタを作成します。この例では、HTTP、RMI、SQL です。このタスクでは、HTTP フィルタを作成する方法を示します。このタスクは、他のフィルタを作成するタスクと同じです。

始める前に

テナント、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を確認します。

手順

ステップ 1 メニュー バーで、[テナント] を選択します。

ステップ 2 **Navigation** ウィンドウで、***tenant-name*** > **Contracts** を選択し、**Filters** を選択し、**Create Filter** を選択します。

ステップ 3 [Create Filter] 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、フィルタ名（http）を入力します。

b) **[エントリ（Entries）]** テーブル内で+をクリックして、**[名前（Name）]** フィールドに、名前（Dport-80）を入力します。

c) [EtherType] ドロップダウン リストから、EtherType（IP）を選択します。

d) [IP Protocol] ドロップダウン リストから、プロトコル（tcp）を選択します。

e) [Destination Port/Range] ドロップダウン リストから、[From] フィールドと [To] フィールドで、[http] を選択します。（http）

f) [Update] をクリックし、[Submit] をクリックします。
新しく追加されたフィルタが、[Navigation] ペインと [Work] ペインに表示されます。

（注） **[エントリ（Entries）]** テーブルでは、**ARP フラグ**には機能がなく、設定できません。このフィールドは使用しません。

ステップ 4 [Name] フィールドの [Entries] を展開します。同じプロセスを実行して、別のエントリを**宛先**ポートとして HTTPS で追加し、[Update] をクリックします。
この新しいフィルタ ルールが追加されます。

ステップ 5 さらに 2 つのフィルタ（rmi および sql）を作成し、rmi および sql 用のフィルタを作成するパラメータ （35 ページ）に示すパラメータを使用するには、上記手順の同じプロセスを実行します。

GUI を使用した契約の作成

手順

ステップ 1 メニュー バーで **Tenants** を選択し、実行するテナント名を選択します。**Navigation** ウィンドウで、***tenant-name*** > **Contracts** を展開します。

ステップ 2 **Standard** > **Create Contract** を右クリックします。

ステップ 3 **Create Contract** 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、契約名（web）を入力します。

b) [Subjects] の横の [+] 記号をクリックし、新しいサブジェクトを追加します。
c) [Create Contract Subject] ダイアログボックスで、[Name] フィールドにサブジェクト名を入力します。（web）
d) （注） この手順では、契約のサブジェクトで前に作成されたフィルタを関連付けます。

[Filter Chain] 領域で、[Filters] の横の [+] 記号をクリックします。
e) ダイアログボックスで、ドロップダウンメニューから、フィルタ名（http）を選択し、[Update] をクリックします。

ステップ 4 [Create Contract Subject] ダイアログボックスで、[OK] をクリックします。

ステップ 5 この手順と同じステップに従って、rmi と sql 用の契約をさらに 2 つ作成します。rmi 契約の場合は rmi サブジェクトを選択し、sql の場合は sql サブジェクトを選択します。

GUI を使用した契約の消費と提供

EPG 間のポリシー関係を作成するために、前に作成した契約を関連付けることができます。

提供するコントラクトと使用するコントラクトに名前を付けるときは、提供するコントラクトと使用するコントラクトの両方に同じ名前を付けてください。

手順

ステップ 1 （注） db、app、および web EPG は、アイコンで表示されます。

APIC GUI ウィンドウをクリックして db EPG から app EPG にドラッグします。
[Add Consumed Contract] ダイアログボックスが表示されます。

ステップ 2 [Name] フィールドで、ドロップダウン リストから、**sql** 契約を選択します。[OK] をクリックします。
この手順により、db EPG は sql 契約を提供でき、app EPG は sql 契約を消費することができます。

ステップ 3 APIC GUI 画面 をクリックして、app ePG から web EPG にドラッグします。
[Add Consumed Contract] ダイアログボックスが表示されます。

ステップ 4 [Name] フィールドで、ドロップダウン リストから、**rmi** 契約を選択します。[OK] をクリックします。
この手順により、app EPG は rmi 契約を提供でき、web EPG は rmi 契約を消費することができます。

ステップ 5 web EPG のアイコンをクリックし、[Provided Contracts] 領域の [+] 記号をクリックします。
[Add Provided Contract] ダイアログボックスが表示されます。

ステップ 6 [Name] フィールドで、ドロップダウン リストから、**web** 契約を選択します。[OK] をクリックします。**[送信（Submit）]** をクリックします。
OnlineStore と呼ばれる 3 層アプリケーション プロファイルが作成されました。

ステップ 7 確認するには、[Navigation] ペインで、[Application Profiles] 下の [OnlineStore] に移動してクリックします。
[Work] ペインで、3 つの EPG app、db および web が表示されていることを確認できます。

ステップ 8 [Work] ペインで、**[Operational]** > **[Contracts]** を選択します。
消費/提供される順番で表示された EPG と契約を確認できます。

コントラクト パフォーマンスの最適化

契約のパフォーマンスの最適化

Cisco APIC、リリース 3.2 で始まるより効率的なハードウェア契約データの TCAM ストレージをサポートしている双方向契約を設定できます。最適化を有効になっている、両方向の統計情報を契約は統合します。

TCAM 最適化は、第 2 世代 Cisco Nexus 9000 シリーズのトップオブラック（TOR）スイッチでサポートされます。これは、EX、FX、および FX2 以降のサフィックスが付いたものです（たとえば、N9K-C93180LC-EX または N9K-C93180YC-FX ）。

TCAM 契約の効率的なデータ ストレージを設定するには、次のオプションが有効にします。

- プロバイダとコンシューマの間で両方向に適用されるコントラクトをマークします。
- IP TCP または UDP プロトコルを使用するフィルタの場合は、リバース ポート オプションを有効にします。
- コントラクト サブジェクトを設定する場合は、**[ポリシー圧縮の有効化（Enable Policy Compression）]** ディレクティブを選択します。これにより、`actrl : Rule` 管理対象オブジェクトの `アクション` 属性に `no_stats` オプションが追加されます。

制限事項

[ポリシー圧縮の有効化（Enable Policy Compression）]（`no_stats`）オプションを選択すると、ルールごとの統計情報が失われます。ただし、両方の方向の複合ルール統計情報は、ハードウェア統計情報に存在します。

Cisco APIC 3.2(1) にアップグレードした後、`no_stats` オプションをアップグレード前のコントラクト サブジェクト（フィルタまたはフィルタ エントリを含む）に追加するには、コントラクト サブジェクトを削除し、**Enable Policy Compression** ディレクティブで再設定する必要があります。そうしないと、圧縮は行われません。

双方向サブジェクト フィルタを使用するコントラクトごとに、Cisco NX-OS は 2 つのルールを作成します。

- `sPcTag` および `dPcTag` が含まれ、`direction=bi-dir` とマークされているルール。これはハードウェアでプログラミングされます。

- プログラミングされていない direction=uni-dir-ignore でマークされたルール

次の設定とルールは圧縮されません。

- ルールの優先順位を持つ fully_qual
- ルールの反対側 (双 dir および uni dir 無視 マーク) と同一ではないプロパティは、次のように **アクション** を含む **統制** 、 **prio、qos** または **markDscp**
- ルール 暗黙的 または implarp フィルタ
- ルール アクションで Deny 、 Redir 、 コピー 、または Deny ログ

次の月クエリ出力は、圧縮のと見なされる、契約の 2 つのルールを示します。

```
apic1# moquery -c actrlRule
Total Objects shown: 2

# actrl.Rule
scopeId            : 2588677
sPcTag             : 16388
dPcTag             : 49156
fltId              : 67
action             : no_stats,permit
actrlCfgFailedBmp  :
actrlCfgFailedTs   : 00:00:00:00.000
actrlCfgState      : 0
childAction        :
ctrctName          :
descr              :
direction          : bi-dir
dn                 : sys/actrl/scope-2588677/rule-2588677-s-16388-d-49156-f-67
id                 : 4112
lcOwn              : implicit
markDscp           : unspecified
modTs              : 2019-04-27T09:01:33.152-07:00
monPolDn           : uni/tn-common/monepg-default
name               :
nameAlias          :
operSt             : enabled
operStQual         :
prio               : fully_qual
qosGrp             : unspecified
rn                 : rule-2588677-s-16388-d-49156-f-67
status             :
type               : tenant


# actrl.Rule
scopeId            : 2588677
sPcTag             : 49156
dPcTag             : 16388
fltId              : 64
action             : no_stats,permit
actrlCfgFailedBmp  :
actrlCfgFailedTs   : 00:00:00:00.000
actrlCfgState      : 0
childAction        :
ctrctName          :
descr              :
direction          : uni-dir-ignore
```

```
dn                  : sys/actrl/scope-2588677/rule-2588677-s-49156-d-16388-f-64
id                  : 4126
lcOwn               : implicit
markDscp            : unspecified
modTs               : 2019-04-27T09:01:33.152-07:00
monPolDn            : uni/tn-common/monepg-default
name                :
nameAlias           :
operSt              : enabled
operStQual          :
prio                : fully_qual
qosGrp              : unspecified
rn                  : rule-2588677-s-49156-d-16388-f-64
status              :
type                : tenant
```

表 *1:* 圧縮マトリクス

リバース フィルタ ポートが有効	TCP または UDP 発信元ポート	TCP または UCP 宛先ポート	圧縮
はい	ポート A	ポート B	はい
はい	未指定	ポート B	はい
はい	ポート A	未指定	はい
はい	未指定	未指定	はい
非対応	ポート A	ポート B	非対応
非対応	未指定	ポート B	非対応
非対応	ポート A	未指定	非対応
非対応	未指定	未指定	対応

GUI を使用して TCAM の使用が最適化された契約を設定する

この手順はで、ハーウェア上の TCAM による契約データの保存を最適化する契約を設定する方法について説明します。

始める前に

- テナント、VRF、およびを提供し、契約を消費する Epg を作成します。
- この契約で許可または拒否されるトラフィックを定義する、1 つ以上のフィルタを作成します。

手順

ステップ 1 メニュー バーで **Tenants** を選択し、実行するテナント名を選択します。**Navigation** ウィンドウで、*tenant-name* および **Contracts** を展開します。

ステップ 2 **Standard** > **Create Contract** を右クリックします。

ステップ 3 **Create Contract** 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、契約 名を入力します。

b) **+** アイコン (**Subjects** の隣にあるもの) をクリックして、新しい情報カテゴリを追加します。

c) [Create Contract Subject] ダイアログボックスで、[Name] フィールドにサブジェクト名を入力します。

（注） この手順では、フィルタを契約の情報カテゴリに関連付けます。

d) TCAM の契約し状況強最適化機能を有効にするには、**Apply Both Directions** および **Reverse Filter Ports** が有効になっていることを確認します。

e) **+** アイコンをクリックして **Filters** を展開します。

f) ダイアログボックスで、ドロップダウンメニューから、デフォルトのフィルタを指定します。すでに設定したフィルタを選択するか、**Create Filter** で新しいフィルタを作成します。

g) **[指令（Directives）]** フィールドで、**[ポリシー圧縮の有効化（Enable Policy Compression）]** を選択します。

h) **Action** フィールドで、**Permit** または **Deny** を選択します。

（注） 現在のところ、**Deny** アクションはサポートされていません。最適化は **Permit** アクションに対してのみ行われます。

i) （任意） **Priority** フィールドで、優先度レベルを選択します。

j) **Update** をクリックします。

ステップ 4 **Create Contract Subject** ダイアログボックスで、**OK** をクリックします。

ステップ 5 **Create Contract** ダイアログボックスで、**Submit**をクリックします。

契約とサブジェクトの例外

コントラクトまたはコントラクトの件名の例外の設定

Cisco APIC リリース 3.2(1) では、EPG 間のコントラクトが拡張され、コントラクトに参加しているコントラクトプロバイダまたはコンシューマのサブネットを拒否できます。インターEPG コントラクトおよびい内部 EPG コントラクトは、この機能でサポートされます。

プロバイダ EPG の件名を有効にして、件名またがコントラクトの例外で一致基準が設定されているものを除くすべてのコンシューマ EPG との通信が可能になります。たとえば、サブセットを除く、テナントのすべての EPG にサービスを提供するために EPG を有効にする場合、これら EPG を除外できます。これを設定するには、コントラクトまたはそのコントラクトの件名のいずれかで例外を作成します。サブセットがコントラクトの提供または消費のアクセスを拒否します。

ラベル、カウンタ、許可および拒否ログは、コントラクトおよび件名の例外でサポートされています。

コントラクトのすべての件名に例外を適用するには、コントラクトに例外を追加します。コントラクトの単一の件名にのみ例外を適用する場合、件名に例外を追加します。

件名にフィルタを追加する場合、フィルタのアクションを設定できます（フィルタ条件に一致するオブジェクトを許可または拒否する）。また、**[拒否]**フィルタについては、フィルタの優先順位を設定することができます。**[許可]**フィルタは常にデフォルトの優先順位があります。自動拒否の件名-フィルタ関係をマーキングすると、件名に一致している場合、各 EPG のペアに適用されます。コントラクトと件名には、複数の件名-フィルタ関係を含むことができます。これは、フィルタに一致するオブジェクトを許可または拒否するように独自に設定できます。

例外タイプ

コントラクトと件名の例外は次のタイプに基づき、* ワイルドカードなどの正規表現を含むことができます。

例外の条件は、[コンシューマ正規表現] および [プロバイダ正規表現]のフィールドで定義されているように、これらのオブジェクトを除外します。	例	説明
テナント	`<vzException consRegex= “common” field= “Tenant” name= “excep03” provRegex= “t1” />`	この例では、`common` テナントを使用して、 EPG が `t1` テナントにより提供されるコントラクトを消費しないように除外します。
VRF	`<vzException consRegex= “ctx1” field= “Ctx” name= “excep05” provRegex= “ctx1” />`	この例では、`ctx1` のメンバーが同じ VRF から提供されるサービスを使用しないように除外します。
EPG	`<vzException consRegex= “EPgPa.*” field= “EPg” name= “excep03” provRegex= “EPg03” />`	この例では、名前が `EPGPa` から始まる複数の EPG が存在すると仮定し、`EPg03` により提供されているコントラクトのコンシューマとしてすべて拒否される必要があります。

例外の条件は、[コンシューマ正規表現] および [プロバイダ正規表現]のフィールドで定義されているように、これらのオブジェクトを除外します。	例	説明
Dn	<vzException consRegex= "uni/tn-t36/ap-customer/epg-epg193" field= "Dn" name="excep04" provRegex= "uni/tn-t36/ap-customer/epg-epg200" />	この例では、epg193 が epg200 により提供されたコントラクトを消費しないように除外します。
タグ	<vzException consRegex= "red" field= "Tag" name= "excep01" provRegex= "green" />	例では、red タグでマークされているオブジェクトが消費することと、green タグでマークされているオブジェクトがコントラクトに参加しないように除外します。

GUI を使用したコントラクトまたはサブジェクトの例外の設定

このタスクでは、EPGのほとんどに対して通信を許可するものの、その一部のアクセスは拒否するコントラクトを設定します。

始める前に

コントラクトを提供し、利用するために、テナント、VRF、アプリケーションプロファイルとEPG を設定します。

手順

ステップ 1 メニュー バーで **[テナント]** > **[すべてテナント]** をクリックします。

ステップ 2 コントラクトを作成しているテナントをダブルクリックします。

ステップ 3 ナビゲーション バーで、**[コントラクト]** を展開し、**[フィルタ]** を右クリックして、**[フィルタの作成]** を選択します。

フィルタでは、コントラクト経由のアクセスを許可または拒否するトラフィックを定義するアクセス制御リスト（ACL）に重要です。許可または拒否できるオブジェクトを定義する複数のフィルタを作成することができます。

ステップ 4 フィルタ名を入力し、許可または拒否するトラフィックを定義する条件を追加して、**[送信]** をクリックします。

ステップ 5 **[コントラクト]** を右クリックし、**[コントラクトの作成]** を選択します。

ステップ 6 コントラクト名を入力し、範囲を設定して、[+] アイコンをクリックし件名を追加します。

ステップ 7 繰り返して別の件名を追加します。

ステップ 8 [Submit] をクリックします。

ステップ 9 コントラクトのすべての件名の例外を追加する手順は、次のとおりです。

a) コントラクトをクリックし、**[コントラクトの例外]** をクリックします。
b) 件名を追加し、許可または拒否するように設定します。
c) [+] アイコンをクリックしてコントラクトを追加します。
d) 例外の名前とタイプを入力します。
e) 正規表現を **[コンシューマ Regex]** および **[プロバイダ Regex]** フィールドに追加し、コントラクトのすべての件名から除外する EPG を定義します。

ステップ 10 コントラクトの 1 つの件名の例外を追加する手順は、次のとおりです。

a) 件名をクリックし、**[件名の例外]** をクリックします。
b) [+] アイコンをクリックしてコントラクトを追加します。
c) 例外の名前とタイプを入力します。
d) 正規表現を **[コンシューマ Regex]** および **[プロバイダ Regex]** に追加し、コントラクトのすべての件名から除外する EPG を定義します。

EPG 内契約

EPG 内契約

EPG 間の通信を制御するには、契約を設定します。Cisco APIC リリース 3.0(1) 以降では、EPG 内の契約を設定できます。

EPG 内契約がない場合、EPG のエンドポイント間の通信は、完全に可能か不可能かになります。通信はデフォルトでは無制限ですが、エンドポイント間の通信を禁止するために、EPG内分離を設定することができます。

ただし、EPG 内契約を使用すれば、同じ EPG のエンドポイント間の通信を制御して、いくつかのトラフィックを許可し、残りの部分を禁止することができます。たとえば、Webトラフィックを許可し、残りの部分をブロックすることが必要な場合があるでしょう。または、すべての ICMP トラフィックと TCP ポート 22 のトラフィックを許可し、他のすべてのトラフィックをブロック中することができます。

EPG 内契約の注意事項と制約事項

EPG 内契約を計画する場合は、次の注意事項と制約事項に従ってください。

- EPG 内契約は、VMware VDS、Open vSwitch (OVS)、およびベアメタル サーバ上のアプリケーション EPG とマイクロセグメント EPG (uSeg) で設定できます。

（注） OVS は、Kubernetes 統合 Cisco Application Centric Infrastructure（ACI）機能で使用できます。Kubernetes では、EPG を作成し、それらに名前空間を割り当てることがことができます。VMware VDS またはベアメタル サーバと同様、Cisco Application Policy Infrastructure Controller（APIC）では、EPG 内ポリシーを EPG に適用することができます。

- EPG 内契約では、リーフ スイッチがプロキシによる Address Resolution Protocol（ARP）をサポートしていることが必要です。EPG 間契約が、モデル名や後発のモデルの最後に EX または EX が付く、Cisco Nexus 9000 シリーズ スイッチでサポートされています。
- EPG 内契約は、Cisco Application Virtual Switch、Cisco ACI Virtual Edge、および Microsoft ドメインではサポートされていません。EPG内契約を設定してこれらのドメインに適用しようとすると、ポートがブロック状態になる可能性があります。
- サービス グラフでの EPG 内コントラクト：
  - サービス グラフを拒否のアクションを含む EPG 内契約のサブジェクトと関連付けることはできません。
  - サービス グラフで EPG 内契約がサポートされるのは、シングル ノード ワンアーム モードのポリシーベース リダイレクトおよびコピー サービスに限られます。
- Cisco APIC リリース 5.2(1) 以降、EPG 内コントラクトは L3Out EPG でサポートされます。
  - アクションは **[許可（permit）]**、**[拒否（deny）]**、または **[リダイレクト（redirect）]** です。**リダイレクト** アクションには、ポリシーベース リダイレクト（PBR）を使用したサービス グラフが必要です。
  - IP アドレスとサブネットが 0.0.0.0/0 または 0 :: 0 の L3Out EPG は、EPG 内コントラクトも EPG 内分離も使用できません。Cisco APIC は、これらの場合に障害を発生させます。ただし、代わりに L3Out EPG の IP アドレスとサブネット 0.0.0.0/1 および 128.0.0.0/1 を使用してすべてのトラフィックを捕捉できます。
  - EPG の EPG 内コントラクトとは異なり、L3Out EPG の EPG 内コントラクトには暗黙の拒否ルールが自動的に追加されません。他のトラフィックを拒否するには、EPG内分離を有効にする必要があります。L3Out EPG の EPG 内分離は、VRF インスタンスが強制モードの場合にのみ機能します。
  - Cisco ACI では、トラフィックが L3Out 内適用の Cisco ACI 境界リーフ スイッチに到達する方法を制御できません。

GUI を使用したアプリケーション EPG への EPG 内契約の追加

コントラクトを設定した後、EPG内コントラクトとしてEPGにコントラクトを追加できます。この手順は、VMware VDS、OVS、およびベアメタル サーバと同じです。

始める前に

- アプリケーション EPG が設定済みである必要があります。
- このアプリケーション用のフィルタが設定された契約が必要です。「GUI を使用した契約の作成 （38 ページ）」を参照してください。

手順

ステップ 1 メニュー バーで、**[テナント（Tenants）]**> > **[すべてのテナント（ALL Tenants）]** の順に選択します。

ステップ 2 [Work] ペインで、テナントの名前をダブルクリックします。

ステップ 3 EPG のタイプに応じて、次の一連の手順のいずれかを実行します。

EPG 内コントラクトに適用する場合：	結果
アプリケーション EPG	1. 左のナビゲーション ペインで、***[tenant_name]*** > **[アプリケーション プロファイル（Application Profiles）]** > ***[アプリケーション プロファイル（application profile）]*** > **[アプリケーション EPG（Application EPGs）]** > ***[epg]*** を展開します。 2. **[コントラクト]** フォルダを右クリックして、**[EPG 内コントラクトの追加]** を選択します。 3. **[Ext-EPG 内コントラクトの追加（Add Intra Ext-EPG Contract）]** ダイアログ ボックスで **[コントラクト（Contract）]** ドロップダウン リストからコントラクトを選択します。 4. **[送信（Submit）]** をクリックします。`
USeg EPG	1. 左のナビゲーション ペインで、***[tenant_name]*** > **[アプリケーション プロファイル（Application Profiles）]** > ***[アプリケーション プロファイル（application profile）]*** > **[uSeg EPGs]** > ***[epg]*** を展開します。 2. **[コントラクト]** フォルダを右クリックして、**[EPG 内コントラクトの追加]** を選択します。 3. **[Ext-EPG 内コントラクトの追加（Add Intra Ext-EPG Contract）]** ダイアログ ボックスで **[コントラクト（Contract）]** ドロップダウン リストからコントラクトを選択します。 4. **[送信（Submit）]** をクリックします。`
L3Out EPG	1. [ナビゲーション（Navigation）] ペインで、***[tenant_name]*** > **[ネットワーキング（Networking）]** > **[L3Outs]** > ***[L3Out_name]*** > **[外部 EPG（External EPGs）]** > ***[ext_epg_name]*** を選択します。

EPG 内コントラクトに適用する場合：	結果
	2. [作業（Work）] ペインの **[Ext-EPG 内分離（Intra Ext-EPG Isolation）]** で、**[適用（Enforced）]** を選択します。 3. **[送信（Submit）]** をクリックします。` 4. [作業（Work）] ペインで、**[ポリシー（Policy）]** > **[コントラクト（Contracts）]**タブを選択します。 5. アクション メニューで、**[Ext-EPG 内コントラクトの追加（Add Intra Ext-EPG Contract）]** を選択します。 6. **[Ext-EPG 内コントラクトの追加（Add Intra Ext-EPG Contract）]** ダイアログ ボックスで **[コントラクト（Contract）]** ドロップダウン リストからコントラクトを選択します。 7. **[送信（Submit）]** をクリックします。` 選択した契約が、[作業（Work）] ペインの **[コントラクト タイプ**：**EPG 内コントラクト（Contract Type**：**Intra EPG Contract）]** セクションに表示されます。

NX-OS スタイル CLI を使用したアプリケーション EPG への EPG 内契約の追加

契約を設定した後、内通EPG契約として、契約を設定できます。手順はVMware VDS、OVS、およびベアメタル サーバで同じです。

始める前に

- 設定されている、EPG は必須です。
- フィルタを持つ契約を設定している必要があります。

手順

ステップ 1 コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ 2 テナントを作成または選択します。

例：

```
apic1(config)# tenant Tenant-l3out
```

ステップ 3 外部 Layer 3 EPG を作成または選択します。

例：

```
apic1(config-tenant)# external-l3 epg ext-epg
```

ステップ 4 外部 EPG を VRF インスタンスにバインドします。

例：

```
apic1(config-tenant-l3ext-epg)# vrf member vrf1
```

ステップ 5 EPG 内で分離を有効にします。

例：

```
(config-tenant-l3ext-epg)# isolation enforce
```

その後、必要に応じて、コマンドの前に no を付けて EPG 内分離を無効にできます。

例：

```
(config-tenant-l3ext-epg)# no isolation enforce
```

ステップ 6 エンドポイント間の目的のトラフィックを許可する契約を内部 EPG に割り当てます。

例：

```
apic1(config-tenant-l3ext-epg)# contract intra-epg contr-intra
```

REST API を使用したアプリケーション EPG への EPG 内契約の追加

コントラクトを設定した後、EPG内コントラクトとしてEPGにコントラクトを追加できます。この手順は、VMware VDS、OVS、およびベアメタル サーバと同じです。

始める前に

- EPG が設定済みである必要があります。
- フィルタが設定されたコントラクトが必要です。

手順

ステップ 1 次の例のような XML POST 要求を使用してセレクタを設定します。

例：

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

```
<infraAccPortP name="Ports-1-12" status="deleted"/>

<!-- VMM VLAN range -->
<fvnsVlanInstP name="test" allocMode="dynamic">
  <fvnsEncapBlk name="encap" from="vlan-10" to="vlan-100"/>
</fvnsVlanInstP>

<!-- Static VLAN range -->
<fvnsVlanInstP name="test" allocMode="static">
  <fvnsEncapBlk name="default" from="vlan-101" to="vlan-4095"/>
</fvnsVlanInstP>

<infraAttEntityP name="test">
  <infraRsDomP tDn="uni/phys-test"/>
  <infraRsDomP tDn="uni/l3dom-test"/>
  <infraRsDomP tDn="uni/vmmp-VMware/dom-test"/>
</infraAttEntityP>

<!-- Node profile -->
<infraNodeP name="test">
  <infraLeafS name="test" type="range">
    <infraNodeBlk name="default" from_="101" to_="102"/>
  </infraLeafS>
  <infraRsAccPortP tDn="uni/infra/accportprof-test"/>
</infraNodeP>

<!-- Port profile -->
<infraAccPortP name="test">
  <!-- 12 regular ports -->
  <infraHPortS name="ports1Through12" type="range">
    <infraPortBlk name="default" fromCard="1" toCard="1" fromPort="1" toPort="12"/>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-test"/>
  </infraHPortS>

  <!-- 2 ports in PC -->
  <infraHPortS name="portsForPc1" type="range">
   <infraPortBlk name="default" fromCard="1" toCard="1" fromPort="13" toPort="14"/>

    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-testPc"/>
  </infraHPortS>

  <!-- 2 ports in PC -->
  <infraHPortS name="portsForPc2" type="range">
    <infraPortBlk name="blk1" fromCard="1" toCard="1" fromPort="15" toPort="16"/>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-pc"/>
  </infraHPortS>

  <!-- 2 ports in PC for FEX -->
  <infraHPortS name="portsForFex" type="range">
    <infraPortBlk name="blk1" fromCard="1" toCard="1" fromPort="17" toPort="18"/>
    <infraRsAccBaseGrp tDn="uni/infra/fexprof-default/fexbundle-test" fexId="111"/>
  </infraHPortS>

  <!-- 2 ports in PC for VPC -->
  <infraHPortS name="portsForVpc" type="range">
    <infraPortBlk name="blk1" fromCard="1" toCard="1" fromPort="19" toPort="20"/>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-testVpc"/>
  </infraHPortS>
</infraAccPortP>

<!-- FEX profile -->
<infraFexP name="default">
  <infraFexBndlGrp name="default"/>
```

```
    <!-- 12 FEX ports -->
    <infraHPortS name="ports1Through12" type="range">
      <infraPortBlk name="default" fromCard="1" toCard="1" fromPort="1" toPort="12"/>
      <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-test"/>
    </infraHPortS>

    <!-- 3 ports in FEX PC -->
    <infraHPortS name="portsForPc" type="range">
      <infraPortBlk name="blk1" fromCard="1" toCard="1" fromPort="13" toPort="16"/>
      <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-testPcOnFex"/>
    </infraHPortS>

    <!-- 3 ports in FEX VPC -->
    <infraHPortS name="portsForVpc" type="range">
      <infraPortBlk name="blk1" fromCard="1" toCard="1" fromPort="17" toPort="19"/>
      <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-testVpcOnFex"/>
    </infraHPortS>
  </infraFexP>

  <!-- Functional profile -->
  <infraFuncP>
    <!-- Regular port group -->
    <infraAccPortGrp name="test">
      <infraRsAttEntP tDn="uni/infra/attentp-test"/>
    </infraAccPortGrp>

    <!-- PC -->
    <infraAccBndlGrp name="testPc" lagT="link">
      <infraRsLacpPol tnLacpLagPolName="testPc"/>
      <infraRsAttEntP tDn="uni/infra/attentp-test"/>
    </infraAccBndlGrp>

    <!-- VPC -->
    <infraAccBndlGrp name="testVpc" lagT="node">
      <infraRsLacpPol tnLacpLagPolName="testVpc"/>
      <infraRsAttEntP tDn="uni/infra/attentp-test"/>
    </infraAccBndlGrp>

    <!-- PC on FEX -->
    <infraAccBndlGrp name="testPcOnFex" lagT="link">
      <infraRsLacpPol tnLacpLagPolName="testPcOnFex"/>
      <infraRsAttEntP tDn="uni/infra/attentp-test"/>
    </infraAccBndlGrp>

    <!-- VPC on FEX -->
    <infraAccBndlGrp name="testVpcOnFex" lagT="node">
      <infraRsLacpPol tnLacpLagPolName="testVpcOnFex"/>
      <infraRsAttEntP tDn="uni/infra/attentp-test"/>
    </infraAccBndlGrp>
  </infraFuncP>

  <!-- Link aggregation policies -->
  <lacpLagPol name="testPc" minLinks='1' maxLinks='10'/>
  <lacpLagPol name="testVpc" minLinks='1' maxLinks='10'/>
  <lacpLagPol name="testPcOnFex" minLinks='2' maxLinks='5'/>
  <lacpLagPol name="testVpcOnFex" minLinks='2' maxLinks='10'/>
</infraInfra>

<fabricInst>
  <fabricProtPol name="testVpc">
    <fabricExplicitGEp name="testVpc" id="101">
      <fabricNodePEp id="101"/>
      <fabricNodePEp id="102"/>
    </fabricExplicitGEp>
```

```
    </fabricProtPol>
  </fabricInst>

  <physDomP name="test">
    <infraRsVlanNs tDn="uni/infra/vlanns-test-static"/>
  </physDomP>

  <l3extDomP name="test">
    <infraRsVlanNs tDn="uni/infra/vlanns-test-static"/>
  </l3extDomP>
</polUni>
```

ステップ 2 次の例のような XML POST 要求を使用してテナントを設定します。

例：

```
<?xml version="1.0" encoding="UTF-8"?>
<polUni>
   <fvTenant name="Tenant-l3out">
      <vzBrCP intent="install" name="contr-intra" scope="context">
         <vzSubj consMatchT="AtleastOne" name="subj" revFltPorts="yes">
            <vzRsSubjFiltAtt action="permit" priorityOverride="default"
              tnVzFilterName="flt-ssh" />
         </vzSubj>
      </vzBrCP>
      <vzBrCP intent="install" name="contr2" scope="context">
         <vzSubj consMatchT="AtleastOne" name="contr2-subj" revFltPorts="yes">
            <vzRsSubjFiltAtt action="permit" priorityOverride="default"
              tnVzFilterName="flt-ftp" />
         </vzSubj>
      </vzBrCP>
      <vzBrCP intent="install" name="contr1" scope="context">
         <vzSubj consMatchT="AtleastOne" name="subj-http" revFltPorts="yes">
            <vzRsSubjFiltAtt action="deny" priorityOverride="default"
              tnVzFilterName="flt-http" />
         </vzSubj>
      </vzBrCP>
      <l3extOut enforceRtctrl="export" mplsEnabled="no" name="l3out1">
         <l3extRsL3DomAtt tDn="uni/l3dom-test" />
         <l3extRsEctx tnFvCtxName="vrf1" />
         <l3extLNodeP name="l3out1_nodeProfile" tag="yellow-green">
            <l3extRsNodeL3OutAtt rtrId="172.16.0.1" rtrIdLoopBack="yes"
              tDn="topology/pod-1/node-101" />
            <l3extLIfP name="l3out1_interfaceProfile" tag="yellow-green">
               <l3extRsPathL3OutAtt addr="192.168.15.1/24" autostate="disabled"
                 encap="unknown" encapScope="local" ifInstT="l3-port" ipv6Dad="enabled"
                 isMultiPodDirect="no" llAddr="::" mac="00:22:BD:F8:19:FF"
                 mode="regular" mtu="inherit"
                 tDn="topology/pod-1/paths-101/pathep-[eth1/10]" />
            </l3extLIfP>
         </l3extLNodeP>

         <!--
           Set pcEnfPref to "enforced" to enable intra-Ext-EPG isolation.
           Set pcEnfPref to "unenforced" to disable intra-Ext-EPG isolation.
         -->
         <l3extInstP floodOnEncap="disabled" matchT="AtleastOne"
           name="l3epg1" pcEnfPref="unenforced" prefGrMemb="exclude">
            <l3extSubnet ip="172.16.0.0/16" scope="import-security" />
            <fvRsCons tnVzBrCPName="contr2" />
            <fvRsIntraEpg tnVzBrCPName="contr-intra" />
         </l3extInstP>
      </l3extOut>
      <fvCtx bdEnforcedEnable="no" ipDataPlaneLearning="enabled" knwMcastAct="permit"
```

```
  name="vrf1" pcEnfDir="egress" pcEnfPref="unenforced" vrfIndex="0">
   <fvRsVrfValidationPol />
   <vzAny matchT="AtleastOne" prefGrMemb="disabled" />
</fvCtx>
<fvBD OptimizeWanBandwidth="no" arpFlood="yes" epClear="no" hostBasedRouting="no"
  intersiteBumTrafficAllow="no" intersiteL2Stretch="no" ipLearning="yes"
  ipv6McastAllow="no" limitIpLearnToSubnets="yes" llAddr="::"
  mac="00:22:BD:F8:19:FF" mcastAllow="no" multiDstPktAct="bd-flood" name="bd-web"
  type="regular" unicastRoute="yes" unkMacUcastAct="proxy" unkMcastAct="flood"
  v6unkMcastAct="flood" vmac="not-applicable">
   <fvSubnet ip="192.168.1.254/24" ipDPLearning="enabled" preferred="no"
     scope="private" virtual="no" />
   <fvRsCtx tnFvCtxName="vrf1" />
   <fvRsBdToEpRet resolveAct="resolve" />
</fvBD>
<fvBD OptimizeWanBandwidth="no" arpFlood="yes" epClear="no" hostBasedRouting="no"
  intersiteBumTrafficAllow="no" intersiteL2Stretch="no" ipLearning="yes"
  ipv6McastAllow="no" limitIpLearnToSubnets="yes" llAddr="::"
  mac="00:22:BD:F8:19:FF" mcastAllow="no" multiDstPktAct="bd-flood" name="bd-app"
  type="regular" unicastRoute="yes" unkMacUcastAct="proxy" unkMcastAct="flood"
  v6unkMcastAct="flood" vmac="not-applicable">
   <fvSubnet ip="192.168.2.254/24" ipDPLearning="enabled" preferred="no"
     scope="private" virtual="no" />
   <fvRsCtx tnFvCtxName="vrf1" />
   <fvRsBdToEpRet resolveAct="resolve" />
</fvBD>
<vzFilter name="flt-ftp">
   <vzEntry applyToFrag="no" arpOpc="unspecified" dFromPort="ftpData"
     dToPort="ftpData" etherT="ipv4" icmpv4T="unspecified" icmpv6T="unspecified"
     matchDscp="unspecified" name="ftp" prot="tcp" sFromPort="unspecified"
     sToPort="unspecified" stateful="no" />
</vzFilter>
<vzFilter name="flt-ssh">
   <vzEntry applyToFrag="no" arpOpc="unspecified" dFromPort="ssh" dToPort="ssh"
     etherT="ipv4" icmpv4T="unspecified" icmpv6T="unspecified"
     matchDscp="unspecified" name="ssh" prot="tcp" sFromPort="unspecified"
     sToPort="unspecified" stateful="no" />
</vzFilter>
<vzFilter name="flt-http">
   <vzEntry applyToFrag="no" arpOpc="unspecified" dFromPort="http" dToPort="http"
     etherT="ipv4" icmpv4T="unspecified" icmpv6T="unspecified"
     matchDscp="unspecified" name="flt1" prot="tcp" sFromPort="unspecified"
     sToPort="unspecified" stateful="no" />
</vzFilter>
<fvAp name="ap-app1">
   <fvAEPg floodOnEncap="disabled" hasMcastSource="no" isAttrBasedEPg="no"
     matchT="AtleastOne" name="epg-app" pcEnfPref="unenforced"
     prefGrMemb="exclude" shutdown="no">
      <fvRsProv intent="install" matchT="AtleastOne" tnVzBrCPName="contr2" />
      <fvRsProv intent="install" matchT="AtleastOne" tnVzBrCPName="contr1" />
      <fvRsPathAtt encap="vlan-103" instrImedcy="immediate" mode="native"
        primaryEncap="unknown" tDn="topology/pod-1/paths-101/pathep-[eth1/3]" />
      <fvRsDomAtt bindingType="none" classPref="encap" encap="unknown"
        encapMode="auto" epgCos="Cos0" epgCosPref="disabled" instrImedcy="lazy"
       netflowDir="both" netflowPref="disabled" numPorts="0" portAllocation="none"

        primaryEncap="unknown" primaryEncapInner="unknown" resImedcy="immediate"
        secondaryEncapInner="unknown" switchingMode="native" tDn="uni/phys-test"
        untagged="no" vnetOnly="no" />
      <fvRsBd tnFvBDName="bd-app" />
   </fvAEPg>
   <fvAEPg floodOnEncap="disabled" hasMcastSource="no"
    isAttrBasedEPg="no" matchT="AtleastOne" name="epg-web" pcEnfPref="unenforced"
```

```
            prefGrMemb="exclude" shutdown="no">
             <fvRsPathAtt encap="vlan-104" instrImedcy="immediate" mode="native"
               primaryEncap="unknown" tDn="topology/pod-1/paths-101/pathep-[eth1/4]" />
             <fvRsDomAtt bindingType="none" classPref="encap" encap="unknown"
               encapMode="auto" epgCos="Cos0" epgCosPref="disabled" instrImedcy="lazy"
              netflowDir="both" netflowPref="disabled" numPorts="0" portAllocation="none"

               primaryEncap="unknown" primaryEncapInner="unknown" resImedcy="immediate"
               secondaryEncapInner="unknown" switchingMode="native" tDn="uni/phys-test"
               untagged="no" vnetOnly="no" />
             <fvRsCons intent="install" tnVzBrCPName="contr1" />
             <fvRsBd tnFvBDName="bd-web" />
           </fvAEPg>
        </fvAp>
    </fvTenant>
</polUni>
```

EPG のコントラクト継承

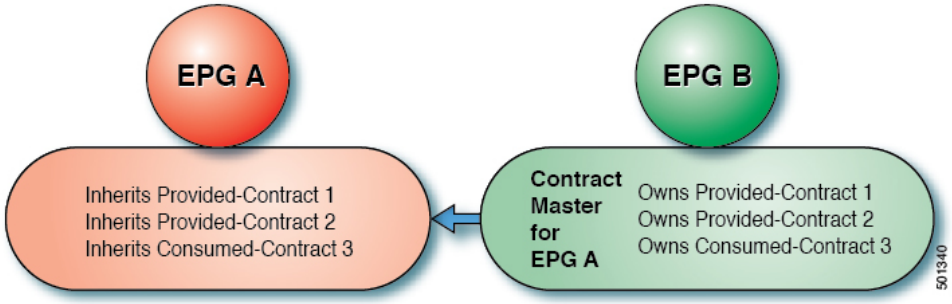
コントラクト継承について

関連する契約を新しい EPG に統合するため、EPG を有効にして同じテナントの別の EPG に直接関連する契約すべて（提供済み/消費済み）を継承できます。コントラクトの継承は、アプリケーション EPG、マイクロセグメント EPG、L2Out EPG、および L3Out EPG に設定できます。

リリース 3.x では、EPG 間の提供済み/消費済みの両方の契約に、契約を継承する設定も可能です。EPG 間契約が、モデル名や後発のモデルの最後に EX または EX が付く、Cisco Nexus 9000 シリーズ スイッチでサポートされています。

EPG を有効にし、APIC GUI、NX-OS スタイル CLI、REST API を使用して、別の EPG に直接関連する契約すべてを継承できます。

図 *9:* コントラクトの継承



上の図で、EPG A は EPG B から（EPG A の契約マスター）提供済みの契約 1 および 2、消費済みの契約 3 を継承するように設定されています。

コントラクト継承を設定する際は、次のガイドラインに従ってください。

- コントラクト継承は、アプリケーション EPG、マイクロセグメント（uSeg）EPG、外部 L2Out EPG、および外部 L3Out EPG 用に設定できます。コントラクト関係は同じタイプの EPG 間で確立する必要があります。
- 関係が確立されると、提供するコントラクトと消費するコントラクトの両方がコントラクト マスターから継承されます。
- コントラクト マスターとコントラクトを継承する EPG は同じテナント内にある必要があります。
- マスター契約への変更は、すべての継承に伝播されます。新しい契約がマスターに追加される場合、継承先にも追加されます。
- EPG は、複数のコントラクト マスターからコントラクトを継承することができます。
- コントラクト継承は単一のレベルでのみサポートされ（連結できない）、コントラクトマスターがコントラクトを継承することはできません。
- コントラクト継承のラベルがサポートされます。EPG A が EPG B からコントラクトを継承するとき、EPG A と EPG B で異なるサブジェクト ラベルが設定されている場合、APIC は EPG B から継承されたコントラクトの EPG B で設定されたラベルを使用します。 APIC は EPG A が直接関与するコントラクトに対し、EPG A の下で設定されたラベルを使用します。
- EPG が契約に直接関連付けられている、または契約を継承しているかどうかに関わらず、TCAM 内のエントリが消費されます。したがって契約スケール ガイドラインが引き続き適用されます。詳細については、お使いのリリースの「検証されたスケーラビリティ ガイド」を参照してください。
- vzAny セキュリティ コントラクトとタブー コントラクはサポートされません。
- Cisco APIC リリース 5.0(1) および 4.2(6) 以降、コントラクトと EPG が同じテナントにある場合、サービス グラフによるコントラクトの継承がサポートされます。

契約の継承設定および継承済みおよびスタンドアロン契約を表示することに関する詳細は、「*Cisco APIC* の基本設定ガイドを参照してください。

GUI を使用した EPG のコントラクト継承の設定

GUI を使用したアプリケーション EPG のコントラクト継承の設定

アプリケーション EPG のコントラクト継承を設定するには、APIC の基本または拡張モード GUI で次の手順を使用します。

始める前に

EPG が使用するテナントとアプリケーション プロファイルを設定します。

オプション。コントラクトを継承する EPG が使用するブリッジ ドメインを設定します。

EPG コントラクト マスターとして機能するように、少なくとも 1 つのアプリケーション EPG を設定します。

共有するコントラクトを設定し、コントラクト マスターに関連付けます。

手順

ステップ 1 **[Tenants]** > ***[tenant-name]*** > **[Application Profiles]** に移動して、[AP-name] を展開します。

ステップ 2 [Application EPGs] を右クリックし、[Create Application EPG] を選択します。

ステップ 3 **EPG コントラクト マスター**からコントラクトを継承する EPG の名前を入力します。

ステップ 4 [Bridge Domain] フィールドで、共通/デフォルトのブリッジ ドメインまたは以前に作成したブリッジ ドメインを選択するか、この EPG のブリッジ ドメインを作成します。

ステップ 5 [EPG Contract Master] フィールドで、+ 記号をクリックして事前に設定したアプリケーション プロファイルと EPG を選択し、[Update] をクリックします。

ステップ 6 [Finish] をクリックします。

ステップ 7 EPG に関する情報（コントラクト マスターなど）を表示するには、**[Tenants]** > ***[tenant-name]*** > **[Application Profiles]** > ***[AP-name]*** > **[Application EPGs]** > ***[EPG-name]*** に移動します。**EPG コントラクト マスター**を表示するには、[General] をクリックします。

ステップ 8 継承されるコントラクトに関する情報を表示するには、[EPG-name] を展開して [Contracts] をクリックします。

GUI を使用した uSeg EPG のコントラクト継承の設定

uSeg EPG のコントラクト継承を設定するには、APIC の基本または拡張モード GUI で次の手順を使用します。

始める前に

EPG が使用するテナントとアプリケーション プロファイルを設定します。

オプション。コントラクトを継承する EPG が使用するブリッジ ドメインを設定します。

EPG コントラクト マスターとして機能するように uSeg EPG を設定します。

共有するコントラクトを設定し、コントラクト マスターに関連付けます。

手順

ステップ 1 **[Tenants]** > ***[tenant-name]*** > **[Application Profiles]** に移動して、[AP-name] を展開します。

ステップ 2 [uSeg EPGs] を右クリックし、[Create uSeg EPG] を選択します。

ステップ 3 コントラクト マスターからコントラクトを継承する EPG の名前を入力します。

ステップ 4 [Bridge Domain] フィールドで、共通/デフォルトのブリッジ ドメインまたは以前に作成したブリッジ ドメインを選択するか、この EPG のブリッジ ドメインを作成します。

ステップ 5 [uSeg-EPG-name] をクリックします。[EPG Contract Master] フィールドで、+ 記号をクリックしてアプリケーションプロファイルと EPG（コントラクトマスターとして機能する）を選択し、[Update] をクリックします。

ステップ 6 [Finish] をクリックします。

ステップ 7 契約に関する情報を表示するには、**[Tenants]** > テナント名 > **[Application Profiles]** > ***AP*** 名 > **[uSeg EPGs]** > に移動し、EPG 名を展開して [Contracts] をクリックします。。

GUI を使用した L2Out EPG のコントラクト継承の設定

外部 L2Out EPG のコントラクト継承を設定するには、Cisco Application Policy Infrastructure Controller（APIC）GUI で次の手順を実行します。

始める前に

EPG が使用するテナントとアプリケーション プロファイルを設定します。

Layer 2 Outside（L2Out）と、**L2Out Contract Master** として機能する外部 L2Out EPG（L2extInstP）を設定します。

共有するコントラクトを設定し、コントラクト マスターに関連付けます。

手順

ステップ 1 **[テナント（Tenants）]** > ***[tenant_name]*** > **[ネットワーキング（Networking）]** > **[L2Outs]** に移動します。

ステップ 2 [L2Out-name] を展開します。

ステップ 3 **[外部 EPG（External EPGs）]** を右クリックし、**[外部 EPG の作成（Create External EPG）]** を選択します。

ステップ 4 外部ネットワークの名前を入力し、必要に応じてその他の属性を追加します。

ステップ 5 **Submit** をクリックします。

ステップ 6 **外部 EPG（External EPGs）]** を展開します。

ステップ 7 ***external-epg-name*** をクリックします。

ステップ 8 **[外部 EPG（External EPG）]** パネルで、**[L2Out コントラクト マスター（L2Out Contract Masters）]** フィールドの [+] 記号をクリックします。

ステップ 9 この外部 L2Out EPG の L2Out および L2Out コントラクト マスターを選択します。

ステップ 10 [更新（Update）] をクリックします。

ステップ 11 この外部 L2Out EPG によって継承されたコントラクトを表示するには、外部 EPG 名をクリックし、**[コントラクト（Contracts）]** > **[継承コントラクト（Inherited Contracts）]** をクリックします。

GUI を使用して外部 L3Out EPG コントラクト継承

外部 L3Out EPG のコントラクト継承を設定するには、Cisco Application Policy Infrastructure Controller（APIC）GUI で次の手順を使用します。

始める前に

EPG が使用するテナントとアプリケーション プロファイルを設定します。

外部ルーテッドネットワーク（L3Out）と、**L3Out コントラクト マスター**として機能する外部 L3Out EPG（L3extInstP）を設定します。

共有するコントラクトを設定し、コントラクト マスターに関連付けます。

手順

ステップ 1 外部 L3Out EPG のコントラクト継承を設定するには、**[テナント（Tenants）]** > ***[tenant-name]*** > **[ネットワーキング（Networking）]** > **[L3Outs]** に移動します。

ステップ 2 外部 L3Out EPG につながる [L3Out-name] を展開します。

ステップ 3 **[外部 EPG（External EPGs）]** を右クリックし、**[外部 EPG の作成（Create External EPG）]** を選択します。

ステップ 4 外部 EPG の名前を入力し、オプションでサブネットおよびその他の属性を追加します。

ステップ 5 **Submit** をクリックします。

ステップ 6 [Networks] を展開します。

ステップ 7 [network-name] をクリックします。

ステップ 8 **[外部 EPG（External EPG）]** パネルで、**[L3Out コントラクト マスター（L3Out Contract Masters）]** フィールドの [+] 記号をクリックします。

ステップ 9 この外部 L3Out EPG の L3Out コントラクト マスターとして機能する L3Out および外部 EPG を選択します。

ステップ 10 [更新（Update）] をクリックします。

ステップ 11 この外部 L3Out EPG によって継承されたコントラクトを表示するには、外部 EPG 名をクリックし、**[コントラクト（Contracts）]** > **[継承コントラクト（Inherited Contracts）]** をクリックします。

優先グループ契約

契約優先グループについて

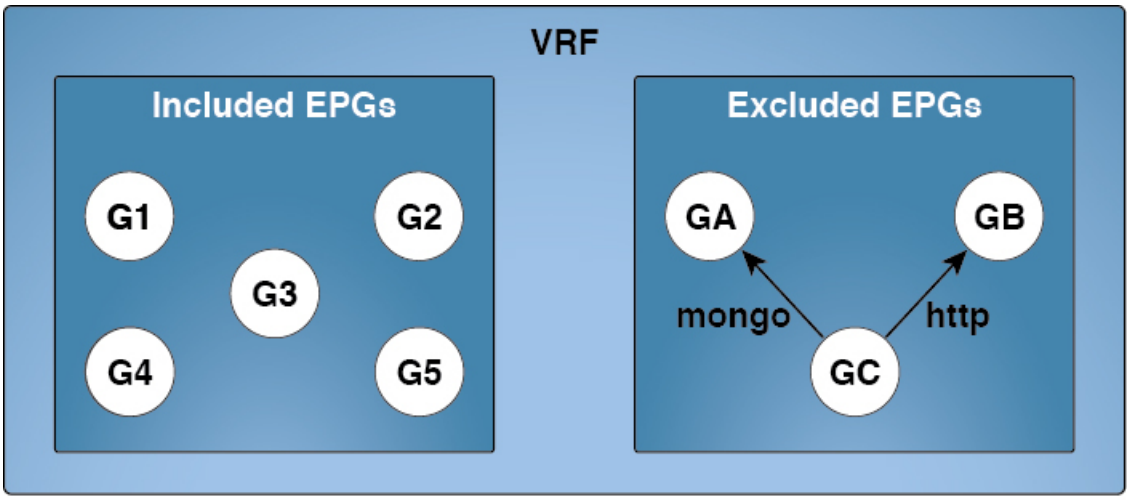
契約優先グループが設定されている VRF で、EPG に利用可能なポリシー適用には 2 種類あります。

- EPG を含む：EPG が契約優先グループのメンバーシップを持っている場合、EPG は契約をせずにお互いに自由に通信できます。これは、source-any-destination-any-permit デフォルト ルールに基づくものです。
- EPG を除外：優先グループのメンバーではない EPG は、相互に通信するために契約が必要です。そうしない場合、デフォルトの source-any-destination-any-deny ルールが適用されます。

契約優先グループ機能では、VRF で EPG 間のより高度な通信の制御が可能です。VRF の EPG のほとんどはオープン通信ですが、一部には他の EPG との制限がある場合、契約優先グループとフィルタ付きの契約の組み合わせを設定し、EPG 内の通信を正確に制御できます。

優先グループから除外されている EPG は、source-any-destination-any-deny デフォルト ルールを上書きする契約がある場合にのみ、他 EPG と通信できます。

図 *10* : 契約優先グループの概要



Source	Destination	Filter	Action
any	any	implicit	permit
GA	any	implicit	deny
any	GA	implicit	deny
GB	any	implicit	deny
any	GB	implicit	deny
GC	any	implicit	deny
any	GC	implicit	deny

Source	Destination	Filter	Action
GC	GA	mongo	permit
GA	GC	mongo	permit
GC	GB	http	permit
GB	GC	http	permit

サービス グラフ サポート

APIC リリース 4.0(1) 以降では、サービス グラフによって作成された EPG を優先契約グループに含めることができます。優先グループ メンバーシップのタイプ（include または exclude）を定義する新しいポリシー（サービス EPG ポリシー）が使用可能です。設定後は、デバイス選択ポリシーまたはサービス グラフ テンプレートのアプリケーションを通じて適用できます。

また、シャドウ EPG を優先グループに含めるか、優先グループから除外するかも設定できるようになりました。

制限事項

以下の制限が契約優先グループに適用されます。

- L3Out およびアプリケーション EPG が契約優先グループで設定されており、EPG が VPC でのみ展開されているトポロジで、VPC の 1 つのリーフ スイッチのみに L3Out のプレフィックス エントリがあることがわかります。この場合、VPC の他のリーフ スイッチにはエントリがなく、そのためトラフィックをドロップします。

  この問題を回避するには、次のいずれかを行います。

  - VRF の契約グループを無効および再度有効にします。

- L3Out EPG のプレフィックス エントリを削除し再度作成します。

- また、サービス グラフ契約のプロバイダまたはコンシューマ EPG が契約グループに含まれる場合、シャドウ EPG は契約グループから除外できません。シャドウ EPG は契約グループで許可されますが、シャドウ EPG が展開されているノードで契約グループ ポリシーの展開をトリガしません。ノードに契約グループ ポリシーをダウンロードするには、契約グループ内にダミー EPG を展開します。
- CSCvm63145 により、コントラクト優先グループの EPG は共有サービス コントラクトを使用できますが、L3Out EPG をコンシューマとして使用する共有サービス コントラクトのプロバイダになることはできません。

契約優先グループの注意事項

契約優先グループを設定する際には、次の注意事項を参照してください:

- （s、g）エントリが境界リーフ スイッチにインストールされている場合、次の条件を満たすと、ファブリックからファブリック外部の送信元に送られたユニキャスト トラフィックでドロップが生じることがあります。
  - 優先グループが L3Out EPG で使用されている
  - 送信元のユニキャスト ルーティング テーブルでデフォルト ルート 0.0.0.0/0 が使用されている

  これは予想された動作です。
- 契約優先グループに含まれる EPG は、外部 EPG (InstP) の 0/0 プレフィックスではサポートされていません。外部 EPG (InstP) からテナント EPG に対し、契約優先グループで使用するために 0/0 プレフィックスが必要な場合には、0/0 を 0/1 と 128/1 に分割することができます。
- 契約優先グループ EPG は、GOLF 機能ではサポートされていません。アプリケーション EPG と GOLF の L3Out EPG との間の通信は、明示的な契約によって制御する必要があります。

GUI を使用した契約優先グループの設定

始める前に

テナントと VRF、および契約優先グループを使用する EPG を作成します。

手順

ステップ 1 メニュー バーで、**[Tenants]** > **テナント名**をクリックします。

ステップ 2 [Navigation] ペインで、テナント、[Networking]、[VRFs] の順に展開します。

ステップ 3 コントラクト優先グループを設定する VRF 名をクリックします。

ステップ 4 **Preferred Group Member** フィールドで、**Enabled** をクリックします。

ステップ 5 **Submit** をクリックします。

ステップ 6 **Navigation** ウィンドウで、**Application Profiles** を展開し、テナント VRF のアプリケーション プロファイルを作成するか、展開します。

ステップ 7 **Application EPGs** を展開し、契約優先グループを使用する EPG をクリックします。

ステップ 8 [Policy] および [General] タブを選択します。

ステップ 9 **Preferred Group Member** フィールドで、**Include** をクリックします。

ステップ 10 **Submit** をクリックします。

次のタスク

この EPG と無制限の通信を行う、他の EPG の優先グループのメンバーシップを有効にします。また、優先グループの EPG とメンバーではないかもしれない他の EPG の間の通信を制御する、適切な契約を設定することもできます。

（注） L4-L7 サービス グラフを介して優先グループ メンバーをサポートする場合は、L4-L7 サービス EPG ポリシーを作成する必要があります。L4-L7 サービス EPG ポリシーの作成に関する詳細については、GUI を使用した L4-L7 サービス EPG ポリシーの作成 （63 ページ）を参照してください。

GUI を使用した L4-L7 サービス EPG ポリシーの作成

このタスクでは、EPG を優先グループに含めるか、優先グループから除外するかを定義するポリシーを作成します。優先グループ メンバーシップにより、エンドポイントは契約がなくても相互に通信できます。作成したポリシーは、EPG にサービス グラフ テンプレートを適用するときに選択できます。

始める前に

テナントを作成しておく必要があります。

手順

ステップ 1 メニュー バーで、[Tenant] > テナント名を選択します。

ステップ 2 [Navigation] ペインで、[Policies] > [Protocol] > [L4-L7 Service EPG Policy] を選択します。

ステップ 3 [Navigation] ペインで、[L4-L7 Service EPG Policy] を右クリックして [Create L4-L7 Service EPG Policy] を選択します。

[Create L4-L7 Service EPG Policy] ダイアログボックスが表示されます。

ステップ 4 [Name] フィールドにポリシーの一意の名前を入力します。

ステップ 5 オプション。[Description] フィールドにポリシーの説明を入力します。

ステップ 6 [Preferred Group Member] フィールドで、EPG を除外するか優先メンバーとして含めるかを選択します。

ステップ 7 [Submit] をクリックします。`

新しく作成したポリシーが [L4-L7 Service EPG Policy] 作業ウィンドウ リストに表示されます。作業ウィンドウでポリシーを編集するには、ポリシーを含む行をダブルクリックします。

次のタスク

サービス グラフを EPG に適用するときに、サービス グラフ テンプレートで新しい L4-L7 サービス EPG ポリシーを選択できるようになりました。『*Cisco APIC Layer 4 to Layer 7 Services Deployment Guide*』の「Using the GUI」の章で「Applying a Service Graph Template to Endpoint Groups Using the GUI」を参照してください。

許可ルールと拒否ルールを含む契約

許可ルールおよび拒否ルールを含む契約の概要

Cisco Application Policy Infrastructure Controller（Cisco APIC）リリース 3.2 以降では、許可だけではなく、許可と拒否の両方のアクションを含む契約を設定できます。さまざまな優先順位（デフォルト、高、中、低）の拒否アクションを設定できます。

ルールの競合は次のように解決されます。

- 暗黙の否定には、すべてのルールの中で最も低い優先順位が割り当てられます。
- VzAny 間の契約には暗黙の拒否より高い優先順位が割り当てられます。
- EPG 間の契約のルールは vzAny 間のルールより優先順位が高いため、特定の EPG ペア間の契約は vzAny の契約よりも優先されます。
- 特定の EPG ペア間の契約に含まれるデフォルト優先順位の拒否ルールは、その EPG ペアの許可ルールと優先順位レベルが同じです。同じ優先順位の許可ルールと拒否ルールの両方がトラフィックに一致する場合は、拒否ルールが優先されます。
- vzAny 間の契約に含まれるデフォルト優先順位の拒否ルールは、その vzAny ペアの許可ルールと優先順位レベルが同じです。同じ優先順位の許可ルールと拒否ルールの両方がトラフィックに一致する場合は、拒否ルールが優先されます。
- 優先順位が最も高い拒否ルールは、EPG 間の契約と同じレベルで処理されます。
- 優先順位が中の拒否ルールは、vzAny-EPG 間の契約と同じレベルで処理されます。

- 優先順位が最も低い拒否ルールは、vzAny 間の契約と同じレベルで処理されます。
- EPG 間の契約で拒否の優先順位を下げると、EPG 間の許可ルールの一致が拒否よりも優先されます。

GUI を使用してACL 契約の許可とロギングの拒否を有効にする

次の手順では、GUI を使用してACL 契約の許可とロギングの拒否を有効にする方法を表示します。

（注） 許可ロギングを含むテナントは、EPG が関連する VRF を含むテナントです。これは必ずしも EPG と同じテナントや関連する契約である必要はありません。

手順

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

ステップ 2 [Navigation] ペインで、[Contracts] を展開し、 [Standard] を右クリックして [Create Contract] を選択します。

ステップ 3 [Create Contract] ダイアログボックスで、次の作業を実行します。

a) [Name] フィールドに、契約の名前を入力します。
b) [Scope] フィールドで、そのスコープ（[VRF]、[Tenant]、または [Global]）を選択します
c) オプション。契約に適用するターゲット DSCP または QoS クラスを設定します。
d) [+] アイコンをクリックして、[Subject] を展開します。

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

ステップ 5 件名の名前と詳細な説明を入力します。

ステップ 6 オプション。ターゲット DSCP のドロップダウン リストから、件名に適用する DSCP を選択します。

ステップ 7 契約を両方向でなくコンシューマからプロバイダの方向にのみ適用するのでない限り、[Apply Both Directions] はオンにしたままにしておきます。

ステップ 8 [Apply Both Directions] をチェックしてない場合 [Reverse Filter Ports] をチェックしたままにして、ルールがプロバイダから消費者に適用されるようにレイヤ 4 ソースと宛先ポートを交換します。

ステップ 9 [+] アイコンをクリックして、[Filters] を展開します。

ステップ 10 [Name] ドロップダウン リストで、たとえば、**arp**、**default**、**est**、**icmp** などオプションを選択するか、以前設定したフィルタを選択します。

ステップ 11 [Directives] ドロップダウン リストで、[log] をクリックします。

ステップ 12 （任意） この件名で実行するアクションを [Deny] に変更します（またはアクションをデフォルトの [Permit] のままにします。

Directive：ログ有効化により、この件名のアクションが [Permit] になっている場合、ACL は件名と契約により制御されているフローとパケットを追跡します。この件名のアクションが [Deny] の場合、ACL の拒否ログはフローとパケットを追跡します。

ステップ 13 （任意） 件名の優先順位を設定します。

ステップ 14 [Update] をクリックします。

ステップ 15 [OK] をクリックします。

ステップ 16 **[送信（Submit）]** をクリックします。
ロギングがこの契約に対して有効になります。

NX-OS CLI を使用した ACL 契約許可ロギングの有効化

次の例は、NX-OS CLI を使用して契約許可ロギングを有効にする方法を示しています。

手順

ステップ 1 契約許可ルールにより送信できたパケットまたはフローのロギングを有効にするには、次のコマンドを使用します。

```
configure
tenant <tenantName>
contract <contractName> type <permit>
subject <subject Name>
access-group <access-list> <in/out/both> log
```

例：

次に例を示します。

```
apic1# configure
apic1(config)# tenant BDMode1
apic1(config-tenant)# contract Logicmp type permit
apic1(config-tenant-contract)# subject icmp
apic1(config-tenant-contract-subj)# access-group arp both log
```

ステップ 2 許可ロギングを無効にするには、**no** 形式の access-group コマンドを使用します。たとえば、`no access-group arp both log` コマンドを使用します。

REST API を使用した ACL 契約許可ロギングの有効化

次の例は、REST API を使用して許可および拒否ロギングを有効にする方法を示しています。この例では、ACL の許可を設定し、件名 Permit 設定し、設定されたアクションを拒否するには、契約のロギングを拒否します。

手順

この設定では、次の例のように XML で post を送信します。

例：

```
<vzBrCP dn="uni/tn-Tenant64/brc-C64" name="C64" scope="context">
    <vzSubj consMatchT="AtleastOne" name="HTTPSsbj" provMatchT="AtleastOne"
revFltPorts="yes" rn="subj-HTTPSsbj">
         <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes"
priorityOverride="default"
rn="rssubjFiltAtt-PerHTTPS" tDn="uni/tn-Tenant64/flt-PerHTTPS" tRn="flt-PerHTTPS"
tnVzFilterName="PerHTTPS"/>
    </vzSubj>
    <vzSubj consMatchT="AtleastOne" name="httpSbj" provMatchT="AtleastOne"
revFltPorts="yes" rn="subj-httpSbj">
        <vzRsSubjFiltAtt action="deny" directives="log" forceResolve="yes"
priorityOverride="default"
rn="rssubjFiltAtt-httpFilter" tDn="uni/tn-Tenant64/flt-httpFilter" tRn="flt-httpFilter"
 tnVzFilterName="httpFilter"/>
    </vzSubj>
   <vzSubj consMatchT="AtleastOne" name="subj64" provMatchT="AtleastOne" revFltPorts="yes"
 rn="subj-subj64">
        <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes"
priorityOverride="default"
rn="rssubjFiltAtt-icmp" tDn="uni/tn-common/flt-icmp" tRn="flt-icmp" tnVzFilterName="icmp"/>

    </vzSubj>
</vzBrCP>
```

GUI を使用した禁止契約拒否ロギングの有効化

次の手順は、GUIを使用して禁止コントラクトの拒否ロギングを有効にする方法を示しています。

手順

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

ステップ 2 [Navigation] ペインで、[Contracts] を展開します。

ステップ 3 [Taboos] を右クリックし、[Create Taboo Contract] を選択します。

ステップ 4 [Create Taboo Contract] ダイアログ ボックスで、次の操作を実行して禁止契約を指定します。

a) [Name] フィールドに、契約の名前を入力します。

b) オプション。[Description] フィールドに、禁止契約の説明を入力します。

c) [+] アイコンをクリックして、[Subject] を展開します。

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

a) [Specify Identity of Subject] 領域に、名前と説明（オプション）を入力します。

b) [+] アイコンをクリックして、[Filters] を展開します。
c) [Name] ドロップダウン リストから、**<tenant_name>/arp**、 **<tenant_name>/default**、**<tenant_name>/est**、**<tenant_name>/icmp** などのデフォルト値のいずれかを選択し、以前作成したフィルタか [Create Filter] を選択します。

（注） [Specify Filter Identity] 領域で [Create Filter] を選択した場合、次の操作を実行して、ACL 拒否ルールの基準を指定します。

1. 名前とオプションの説明を入力します。
2. [Entries] を展開し、ルールの名前を入力して、拒否するトラフィックを定義する条件を選択します。
3. [Directives] ドロップダウンリストで [log] を選択します。
4. [Update] をクリックします。
5. [OK] をクリックします。

ステップ 6 **[送信（Submit）]** をクリックします。
ロギングがこの禁止契約に対して有効になります。

NX-OS CLI を使用した禁止契約拒否ロギングの有効化

次の例は、NX-OS CLI を使用して禁止契約拒否ロギングを有効にする方法を示しています。

手順

ステップ 1 禁止契約拒否ルールのためにドロップされたパケットまたはフローのロギングを有効にするには、次のコマンドを使用します。

```
configure
tenant <tenantName>
contract <contractName> type <deny>
subject <subject Name>
access-group <access-list> <both> log
```

例：

次に例を示します。

```
apic1# configure
apic1(config)# tenant BDMode1
apic1(config-tenant)# contract dropFTP type deny
apic1(config-tenant-contract)# subject dropftp
apic1(config-tenant-contract-subj)# access-group ftp both log
```

ステップ 2 拒否ロギングを無効にするには、**no** 形式の access-group コマンドを使用します。たとえば、`no access-group https both log` コマンドを使用します。

REST API を使用した禁止契約拒否ロギングの有効化

次の例は、REST API を使用して禁止契約拒否ロギングを有効にする方法を示しています。

手順

タブー契約を設定するロギングを拒否する、次の例のように XML で post を送信します。

例：

```
<vzTaboo dn="uni/tn-Tenant64/taboo-TCtrctPrefix" name="TCtrctPrefix" scope="context">
    <vzTSubj name="PrefSubj" rn="tsubj-PrefSubj"">
        <vzRsDenyRule directives="log" forceResolve="yes" rn="rsdenyRule-default"
tCl="vzFilter"
tDn="uni/tn-common/flt-default" tRn="flt-default"/>
    </vzTSubj>
</vzTaboo>
```

GUI を使用した ACL 許可および拒否ログの表示

次の手順は、GUI を使用して、トラフィック フローの ACL 許可および拒否ログを（有効になっていれば）表示する方法を示しています。

手順

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

ステップ 2 [Navigation] ペインで、[Tenant <tenant name>] をクリックします。

ステップ 3 **Tenants <tenant name> [Work]** ペインで、**[Operational]** タブをクリックします。

ステップ 4 [Operational] タブの下で、[Flows] タブをクリックします。
[Flows] タブの下で、いずれかのタブをクリックして、レイヤ 2 許可ログ（[L2 Permit]）、レイヤ 3 許可ログ（[L3 Permit]）、レイヤ 2 拒否ログ（[L2 Drop]）、またはレイヤ 3 拒否ログ（[L3 Drop]）のログ データを表示します。各タブで、トラフィックがフローしていれば、ACL ロギング データを表示できます。データ ポイントは、ログ タイプと ACL ルールに応じて異なります。たとえば、[L3 Permit] ログおよび [L3 Deny] ログには次のデータ ポイントが含まれます。

- VRF
- Alias
- 送信元 IP アドレス
- 宛先 IP アドレス
- プロトコル
- 送信元ポート

- 宛先ポート
- 送信元 MAC アドレス
- 宛先 MAC アドレス
- Node
- 送信元インターフェイス
- VRF Encap
- 送信元 EPG
- 宛先 EPG
- 送信元 PC タグ
- 宛先 PC タグ

（注） また、[Flows] タブの横の [Packets] タブを使用して、シグニチャ、送信元、および宛先が同じであるパケットのグループ（最大 10 個）の ACL ログにアクセスできます。送信されたりドロップされたりするパケットのタイプを確認できます。

REST API を使用した ACL 許可および拒否ログ

次の例は、REST API を使用して、トラフィック フローのレイヤ 2 拒否ログ データを表示する方法を示しています。次の MO を使用してクエリを送信することができます。

- acllogDropL2Flow
- acllogPermitL2Flow
- acllogDropL3Flow
- acllogPermitL3Flow
- acllogDropL2Pkt
- acllogPermitL2Pkt
- acllogDropL3Pkt
- acllogPermitL3Pkt

始める前に

ACL 契約許可および拒否ログのデータを表示する前に、許可または拒否ロギングを有効にする必要があります。

手順

レイヤ 3 ドロップ ログ データを表示するには、REST API を使用して次のクエリを送信します。

```
GET https://apic-ip-address/api/class/acllogDropL3Flow
```

例：

次の例では、サンプル出力をいくつか示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="2">
    <acllogPermitL3Flow childAction=""
dn="topology/pod-1/node-101/ndbgs/acllog/tn-common/ctx-inb

/permitl3flow-spctag-333-dpctag-444-sepgname-unknown-depgname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]


-dip-[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-

        [port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
 dstMacAddr="00:00:12:00:00:25"
        dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn=""
protocol="udp" srcEpgName="unknown"
        srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0"
srcMacAddr="00:00:15:00:00:28" srcPcTag="333"
        srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
    <acllogPermitL3Flow childAction=""
dn="topology/pod-1/node-102/ndbgs/acllog/tn-common/ctx-inb

/permitl3flow-spctag-333-dpctag-444-sepgname-unknown-depgname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]-dip-


[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-

        [port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
 dstMacAddr="00:00:12:00:00:25"
        dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn=""
protocol="udp" srcEpgName="unknown"
        srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0"
srcMacAddr="00:00:15:00:00:28" srcPcTag="333"
        srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
</imdata>
```

NX-OS CLI を使用した ACL 許可および拒否ログの表示

次の手順は、NX-OS スタイル CLI **show acllog** コマンドを使用して ACL ログの詳細を表示する方法を示しています。

レイヤ 3 コマンドの構文は、**show acllog {permit | deny} l3 {pkt | flow} tenant <tenant_name> vrf <vrf_name> srcip <source_ip> dstip <destination_ip> srcport <source_port> dstport <destination_port> protocol <protocol> srcintf <source_interface> start-time <startTime> end-time <endTime> detail** です。

レイヤ 2 コマンドの構文は、**show acllog {permit | deny} l2 {flow | pkt} tenant <tenant_name> vrf <VRF_name> srcintf <source_interface> vlan <VLAN_number> detail** です。

（注） **show acllog** コマンドの完全な構文は、第二世代 Cisco Nexus 9000 シリーズ スイッチ（N9K-C93180LC-EX など名前の最後に EX または FX がつく。もしくはそれ以降のシリーズ）および Cisco APIC リリース 3.2 以降でのみ使用できます。第一世代のスイッチ（名前の最後に EX または FX が付かない）または 3.2 以前の Cisco APIC リリースでは、使用可能な構文は上記の通りです。

Cisco APIC 3.2 以降では、追加のキーワードが **detail** keyword:**[dstEpgName <destination_EPG_name>| dstmac <destination_MAC_address> | dstpctag <destination_PCTag> | srcEpgName <source_EPG_name> | srcmac <source_MAC_address> | srcpctag <source_PCTag>]** とともにコマンドの両方のバージョンに追加されます。

手順

ステップ 1 次の例では、**show acllog drop l3 flow tenant common vrf default detail** コマンドを使用して、共通テナントのレイヤ 3 拒否ログに関する詳細情報を表示する方法を示します。

例：

```
apic1# show acllog deny l3 flow tenant common vrf default detail
SrcPcTag  : 49153
DstPcTag  : 32773
SrcEPG    : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg6
DstEPG    : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg5
SrcIp     : 16.0.2.10
DstIp     : 19.0.2.10
Protocol  : udp
SrcPort   : 17459
DstPort   : 8721
SrcMAC    : 00:00:15:00:00:28
DstMAC    : 00:00:12:00:00:25
Node      : 101
SrcIntf   : port-channel5
VrfEncap  : VXLAN: 2097153
```

この例では第二世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 2 次の例では、**show acllog deny l2 flow tenant common vrf tsw0connctx0 detail** コマンドを使用して、共通テナントのレイヤ 3 拒否ログに関する詳細情報を表示する方法を示します。

例：

```
apic1# show acllog deny l2 flow tenant common vrf tsw0connctx0 detail
SrcPcTag DstPcTag   SrcEPG           DstEPG          SrcMAC            DstMAC
Node   SrcIntf   vlan
--------- --------- -----------------  --------------  --------------------
------------------   -----   ----------  ------
 32773    49153   uni/tn-TSW       uni/tn-TSW  00:00:11:00:00:11  11:00:32:00:00:33
101     port-     2
                  _Tenant0/ap-     _Tenant0/ap-
       channel8
```

```
                    tsw0AP0/epg-     tsw0AP0/epg-
                    tsw0ctx0BD0epg5  tsw0ctx0BD0epg6
```

この例では第二世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 3 次の例では、**show acllog permit l3 pkt tenant <tenant name> vrf <vrf name> [detail]** コマンドを使用して、送信された一般的な VRF ACL レイヤ 3 許可パケットに関する詳細情報を表示する方法を示しています。

```
apic1# show acllog permit l3 pkt tenant common vrf default detail acllog permit l3 packets
 detail:
srcIp      : 10.2.0.19
dstIp      : 10.2.0.16
protocol   : udp
srcPort    : 13124
dstPort    : 4386
srcIntf    : port-channel5
vrfEncap   : VXLAN: 2097153
pktLen     : 112
srcMacAddr : 00:00:15:00:00:28
dstMacAddr : 00:00:12:00:00:25
timeStamp  : 2015-03-17T21:31:14.383+00:00
```

この例では第一世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 4 次の例では、**show acllog permit l2 pkt tenant <tenant name> vrf <vrf name> srcintf <s interface>** コマンドを使用して、インターフェイス ポートチャネル 15 から送信されたデフォルトの VRF レイヤ 2 パケットに関する情報を表示する方法を示しています。

```
apic1# show acllog permit l2 pkt tenant common vrf default srcintf port-channel5

acllog permit L2 Packets
      Node          srcIntf       pktLen     timeStamp
 --------------  --------------  --------  --------------
                 port-channel5       1      2015-03-17T21:
                                            31:14.383+00:00
```

この例では第一世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

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。