



レイヤ4からレイヤ7のサービスの管理

- [About Layer 4 to Layer 7 Services](#) (1 ページ)
- [レイヤ7サービスにレイヤ4を管理するためのアクセス](#) (3 ページ)
- [デバイス パッケージ](#) (6 ページ)
- [トランキング](#) (8 ページ)
- [Device Selection Policies](#) (9 ページ)
- [ポリシー ベースのリダイレクトとサービス ノードのトラッキング](#) (10 ページ)
- [サービス グラフ テンプレート](#) (14 ページ)
- [レイヤ4～レイヤ7パラメータ](#) (16 ページ)
- [Copy Services](#) (20 ページ)
- [自動化の開発](#) (23 ページ)
- [Example: Configuring Layer 4 to Layer 7 Services \(Firewall\)](#) (31 ページ)
- [例: レイヤ7ルートピアリングするレイヤ4の設定](#) (41 ページ)

About Layer 4 to Layer 7 Services

アプリケーション セントリック インフラストラクチャのレイヤ4～7 サービスについて

VLANおよび仮想ルーティングおよび転送（VRF）スティーチングは従来のサービス挿入モデルによってサポートされますが、Application Policy Infrastructure Controller（APIC）はポリシー制御の中心点として機能する一方でサービス挿入を自動化できます。APIC ポリシーは、ネットワーク ファブリックとサービスアプライアンスの両方を管理します。APICは、トラフィックがサービスを通して流れるように、ネットワークを自動的に設定できます。APICは、アプリケーション要件に従ってサービスを自動的に設定することもでき、それにより組織はサービス挿入を自動化し、従来のサービス挿入の複雑な技術の管理に伴う課題を排除できます。

開始する前に次の APIC オブジェクトを設定する必要があります。

- レイヤ4～7サービスを提供/消費するテナント
- テナントのネットワーク外部のレイヤ3

- 最低でも1個のブリッジドメイン
- アプリケーションプロファイル
- 物理ドメインまたはVMMドメイン
VMMドメインについて、VMMドメインのクレデンシャルを設定し、vCenter/vShieldコントローラプロファイルを設定します。
- カプセル化ブロック範囲を持つVLANプール
- 最低でも1個の契約
- 最低でも1個のEPG

次のタスクを実行し、レイヤ4～7サービスを展開します。

1. **デバイスパッケージ**をインポートします。
プロバイダーの管理者のみがデバイスパッケージをインポートできます。
2. デバイスおよび論理インターフェイスを登録します。
また、このタスクでは、具象デバイスと具象インターフェイスを登録し、具象デバイスパラメータを設定します。
3. **論理デバイス**を作成します。
4. デバイスパラメータを設定します。
5. オプション。ASAファイアウォールサービスを設定する場合は、デバイスのトランッキングを有効にします。
6. **デバイス選択ポリシー**を設定します。
7. **サービスグラフテンプレート**を設定します。
 1. アプリケーションプロファイルからのデフォルトのサービスグラフテンプレートのパラメータを選択します。
 2. 必要に応じた追加のサービスグラフテンプレートのパラメータを設定します。
8. 契約のサービスグラフテンプレートを添付します。
9. 必要な場合は、追加の設定パラメータを設定します。

レイヤ4～レイヤ7サービスの展開の詳細については、「Cisco APIC レイヤ4～レイヤ7サービス展開ガイド」を参照してください。

レイヤ7サービスにレイヤ4を管理するためのアクセス

テナントを使用したデバイスへのインバンド接続の設定の VRF は、REST API を使用して

The following is an example of using REST APIs to configure in-band connectivity to devices using tenant's VRF:

1. 管理に使用する EPG を定義します。



(注) 適切なセレクトラ設定を使用してドメインマッピングに対してポートを開きます。

次の例では、EPG の「サービス」がテナント VRF デバイス管理に使用されるサービス デバイス/VM サブネットの管理に使用されます (3.3.3.0/24)。

```
<polUni>
  <fvTenant name="tenant1">
    <fvCtx name="mgmt_ctx1"/>
    <vnsCtrlrMgmtPol ctxDn="uni/tn-tenant1/ctx-mgmt_ctx1">
      <vnsRsMgmtAddr tDn="uni/tn-tenant1/ap-services/epg-ifc/CtrlrAddrInst-ifc"/>
    </vnsCtrlrMgmtPol>
    <fvBD name="mgmt_ServicesMgmtBD">
      <fvRsCtx tnFvCtxName="mgmt_ctx1"/>
      <fvSubnet ip="3.3.3.3/24"/>
    </fvBD>
    <fvAp name="services">
      <fvAEPg name="ifc">
        <fvRsBd tnFvBDName="mgmt_ServicesMgmtBD"/>
        <vnsAddrInst name="ifc">
          <fvnsUcastAddrBlk from="3.3.3.100/24" to="3.3.3.200/24"/>
        </vnsAddrInst>
        <fvRsDomAtt tDn="uni/vmmp-VMware/dom-mininet"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```

2. LDevVip に EPG を関連付けます。

```
<polUni>
  <fvTenant name="tenant1">
    <vnsLDevVip name="ADCCluster1"
      funcType="GoTo" devtype="VIRTUAL">
      <vnsRsMDevAtt tDn="uni/infra/mDev-Citrix-NetScaler-10.5"/>
      <vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-mininet"/>
      <vnsRsDevEpg tDn="uni/tn-tenant1/ap-services/epg-ifc"/>
    </vnsLDevVip>
    <vnsCMgmt name="devMgmt"
      host="3.3.3.180"
      port="80"/>
  </fvTenant>
</polUni>
```

```

        <vnsCCred name="username"
            value="nsroot"/>

        <vnsCCredSecret name="password"
            value="nsroot"/>
    </vnsLDevVip>
</fvTenant>
</polUni>

```

REST API を使用した管理テナント VRF によるデバイスへのインバンド接続の設定

以下は、管理テナント VRF を使用したデバイスへのインバンド接続を設定する REST API の使用例です。

1. テナント管理で EPG l4l7MgmtEpg を作成します。



(注) l4l7MgmtEpg は、tn-mgmt 内の inb コンテキスト下の bd アクセスの一部です。

contract1 はデフォルト EPG 内の tn-mgmt l4l7MgmtEpg および tn-mgmt inb 間のコントラクトです。

```

<polUni>
  <fvTenant dn="uni/tn-mgmt">
    <fvAp name="services">
      <fvAEPg name="l4l7MgmtEpg">
        <fvRsBd tnFvBDName="access" />
        <fvRsDomAtt tDn="uni/vmmp-VMware/dom-mininet" />
        <fvRsCons tnVzBrCPName='contract1'>
          </fvRsCons>
        </fvAEPg>
      </fvAp>
    <fvBD name="access">
      <fvSubnet ip="3.3.3.3/24" />
      <fvRsCtx tnFvCtxName="inb"/>
    </fvBD>
    <vzFilter name='all'>
      <vzEntry name='all' ></vzEntry>
    </vzFilter>
    <vzBrCP name="contract1" scope="tenant">
      <vzSubj name='subj1'>
        <vzInTerm>
          <vzRsFiltAtt tnVzFilterName="all" />
        </vzInTerm>
        <vzOutTerm>
          <vzRsFiltAtt tnVzFilterName="all" />
        </vzOutTerm>
      </vzSubj>
    </vzBrCP>
  </fvTenant>
</polUni>

```

2. サービス デバイス/VM にサブネット 3.3.3.0/24 で管理 IP アドレスがあることを確認します。
これは tn-mgmt アクセスBD が設定されたのと同じサブネットです。（前のステップの設定を参照してください）。
3. 次を LDevVip に追加します。



(注) これは、前のステップで作成された EPG にポイントします。

```
<vnsRsDevEpg tDn="uni/tn-mgmt/ap-services/epg-l4l7MgmtEpg"/>.
```

```
<polUni>
  <fvTenant name="mgmt">
    <vnsLDevVip name="ADCCluster1"
      funcType="GoTo" devtype="VIRTUAL">
      <vnsRsMDevAtt tDn="uni/infra/mDev-Citrix-NetScaler-10.5"/>
      <vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-mininet"/>
      <vnsRsDevEpg tDn="uni/tn-mgmt/ap-services/epg-l4l7MgmtEpg"/>
    <vnsCMgmt name="devMgmt"
      host="3.3.3.180"
      port="80"/>
    <vnsCCred name="username"
      value="nsroot"/>
    <vnsCCredSecret name="password"
      value="nsroot"/>
  </vnsLDevVip>
</fvTenant>
</polUni>
```

4. IFC インバンド ゲートウェイにポイントするようにサービスデバイス/VM にルートを追加します。

たとえば、netScaler のルート上で、ルート 3.0.0.0 255.255.255.0 3.3.3.3 を追加します。
3.0.0.0/24 は IFC インバンド サブネット、3.3.3.3 は l4l7MgmtEpg の SVI IP です。

5. 次の点を確認します。
 - IFC ルート テーブルに ifc インバンド IP のエントリがあります。
 - IFC は、リーフの l4l7MgmtEpg ゲートウェイに ping を送信できます。
 - サービス ノードは、l4l7MgmtEpg SVI ゲートウェイおよび IFC inb SVI Ip を ping できます。

デバイス パッケージ

デバイス パッケージについて

APIC (Application Policy Infrastructure Controller) は、サービス デバイスの設定およびモニタリングにデバイス パッケージを必要とします。デバイス パッケージは、単一クラスのサービス デバイスを管理し、デバイスとその機能に関する情報を APIC に提供します。

デバイス パッケージに関する詳細は、「Cisco APIC レイヤ 4～レイヤ 7 デバイス パッケージ 展開ガイド」を参照してください。

REST でのデバイス パッケージのインストール時の注意事項

- デバイス パッケージは HTTP または HTTPS POST を使用してインストールできます。
- HTTP が APIC で有効になっている場合、POST の URL は「`http://10.10.10.10/ppi/node/mo/.xml`」です。
- HTTPS が APIC で有効になっている場合、POST の URL は「`https://10.10.10.10/ppi/node/mo/.xml`」です。
- メッセージには有効なセッション Cookie が必要です。
- POST の本文にはアップロードされるデバイス パッケージを含める必要があります。POST で許可されるのは 1 つのパッケージだけです。

API を使用したデバイス パッケージ ファイルのアップロード

サービス デバイスをインストールするには、APIC にデバイス パッケージ ファイルをアップロードする必要があります。この操作用の API コマンドでは、次の URI の特別な形式を使用します。

```
{http|https}://host[:port]/ppi/node/mo/.{json|xml}
```

URI パスには、「api」の代わりに「ppi」（パッケージ プログラミング インターフェイス）が含まれ、コマンドはメッセージの本文としてデバイス パッケージ ファイルでの POST 操作として送信されます。デバイス パッケージ ファイルは zip ファイルです。

次の例では、デバイス パッケージ ファイルをアップロードする API 操作を示します。

```
POST https://192.0.20.123/ppi/node/mo/.json
```

L4～L7 のサービス デバイス パッケージのインストールの詳細については、『Cisco APIC レイヤ 4～レイヤ 7 サービス導入ガイド』を参照してください。

REST API を使用したデバイス パッケージのインストール

デバイス パッケージは HTTP または HTTPS POST を使用してインストールできます。

手順

デバイス パッケージをインストールします。

- HTTP が Application Policy Infrastructure Controller (APIC) で有効になっている場合の POST の URL は次のとおりです。

```
http://10.10.10.10/ppi/node/mo/.xml
```

- HTTPS が APIC で有効になっている場合の POST の URL は次のとおりです。

```
https://10.10.10.10/ppi/node/mo/.xml
```

メッセージには有効なセッション Cookie が必要です。

POST の本文にはアップロードされるデバイス パッケージを含める必要があります。POST で許可されるのは1つのパッケージだけです。

REST API とともにインポートされたデバイスの使用

次の REST API ではインポートされたデバイスを使用します。

```
<polUni>
  <fvTenant dn="uni/tn-tenant1" name="tenant1">
    <vnsLDevIf ldev="uni/tn-mgmt/lDevVip-ADCCluster1"/>
    <vnsLDevCtx ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any">
      <vnsRsLDevCtxToLDev tDn="uni/tn-tenant1/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]"/>

      <vnsLIIfCtx connNameOrLbl="inside">
        <vnsRsLIIfCtxToLIIf
tDn="uni/tn-tenant1/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLIIf-inside"/>
        <fvSubnet ip="10.10.10.10/24"/>
        <vnsRsLIIfCtxToBD tDn="uni/tn-tenant1/BD-tenant1BD1"/>
      </vnsLIIfCtx>
      <vnsLIIfCtx connNameOrLbl="outside">
        <vnsRsLIIfCtxToLIIf
tDn="uni/tn-tenant1/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLIIf-outside"/>
        <fvSubnet ip="70.70.70.70/24"/>
        <vnsRsLIIfCtxToBD tDn="uni/tn-tenant1/BD-tenant1BD4"/>
      </vnsLIIfCtx>
    </vnsLDevCtx>
  </fvTenant>
</polUni>
```

トランキング

トランキングの概要

レイヤ4～レイヤ7仮想ASAデバイスのトランキングを有効にでき、これはトランクポートグループを使用してエンドポイントグループのトラフィックを集約します。トランキングを使用せず、仮想サービスデバイスには各インターフェイスに1個のVLANのみ所有し、最大10個のサービスグラフを所有できます。トランキングが有効にしている状態では、仮想サービスデバイスはサービスグラフの数を無制限に設定できます。

トランクポートグループについての詳細は、『Cisco ACI Virtualization Guide』を参照してください。

トランキングは、仮想ASAデバイスでのみサポートされます。ASAデバイスパッケージは、バージョン1.2.7.8以降である必要があります。

REST Api を使用してレイヤ7仮想ASAデバイスにレイヤ4でのトランキングを有効化

次の手順では、REST Api を使用して、レイヤ7仮想のASAデバイスにレイヤ4でのトランキングを有効にする例を示します。

始める前に

- ASA デバイスの仮想レイヤ7にレイヤ4に設定した必須。

手順

名前付きレイヤ7デバイスにレイヤ4でのトランキングを有効にする InsiemeCluster :

```
<polUni>
  <fvTenant name="tenant1">
    <vnsLDevVip name="InsiemeCluster" devtype="VIRTUAL" trunking="yes">
      ...
    </vnsLDevVip>
  </fvTenant>
</polUni>
```


Device Selection Policies

デバイス選択ポリシーについて

デバイスは、コントラクト名、グラフ名、またはグラフ内の機能ノード名に基づいて選択できます。デバイスを作成した後は、デバイスに選択条件ポリシーを提供するデバイスコンテキストを作成できます。

デバイス選択ポリシー（デバイスコンテキストとも呼ばれる）は、サービスグラフテンプレートのデバイスを選択するためのポリシーを指定します。これにより、管理者は複数のデバイスを持つことができ、それらを異なるサービスグラフテンプレートに対して使用することができます。たとえば、管理者は、高いパフォーマンス ADC アプライアンスがあるデバイスと、パフォーマンスが低い ADC アプライアンスがある別のデバイスを持つことができます。高いパフォーマンスの ADC デバイス用と低いパフォーマンスの ADC デバイス用の 2 つの異なるデバイス選択ポリシーを使用して、管理者は高いパフォーマンスが必要となるアプリケーションには高いパフォーマンスの ADC デバイスを選択し、低いパフォーマンスが必要なアプリケーションには低いパフォーマンスの ADC デバイスを選択することができます。

REST API を使用してデバイス選択ポリシーの作成

次の REST API ではデバイス選択ポリシーを作成します。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevCtx ctrctNameOrLbl="webCtrct" graphNameOrLbl="G1" nodeNameOrLbl="Node1">

      <vnsRsLDevCtxToLDev tDn="uni/tn-acme/1DevVip-ADCcluster1"/>

      <!-- The connector name C4, C5, etc.. should match the
           Function connector name used in the service graph template -->

      <vnsLIfCtx connNameOrLbl="C4">
        <vnsRsLIfCtxToLIf tDn="uni/tn-acme/1DevVip-ADCcluster1/LIf-ext"/>
      </vnsLIfCtx>
      <vnsLIfCtx connNameOrLbl="C5">
        <vnsRsLIfCtxToLIf tDn="uni/tn-acme/1DevVip-ADCcluster1/LIf-int"/>
      </vnsLIfCtx>
    </vnsLDevCtx>
  </fvTenant>
</polUni>
```

REST API を使用したデバイスでの論理インターフェイスの追加

次の REST API はデバイス内に論理インターフェイスを追加します。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCcluster1">

      <!-- The LIF name defined here (such as e.g., ext, or int) should match the
```

```

vnsRsLifCtxToLif `tDn` defined in LifCtx -->

<vnsLif name="ext">

  <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-outside"/>
  <vnsRsCifAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-ext"/>
</vnsLif>
<vnsLif name="int">
  <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-inside"/>
  <vnsRsCifAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-int"/>
</vnsLif>
</vnsLDevVip>
</fvTenant>
</polUni>

```

ポリシーベースのリダイレクトとサービスノードのトラッキング

ポリシーベースのリダイレクトとサービスノードのトラッキング



- (注) この機能は、APIC リリース 2.2(3x) リリースおよび APIC リリース 3.1 (1) で使用できます。APIC リリース 3.0(x) ではサポートされていません。

Cisco APIC、リリース 2.2(3x) とポリシーベースのリダイレクトとサービスノードの追跡 (PBR) のサポートが機能します。

宛先ノードのサポートデュアル IP スタックをリダイレクトします。したがって、IPv4 と IPv6 の両方のアドレスは、同時に設定できます。

スイッチは、トラッキング PBR をサポートするのに Cisco IP SLA モニタリング機能を内部的に使用します。トラッキング機能では、サービスノードに到達できない場合に、リダイレクト宛先ノードがマークされます。トラッキング機能は、サービスノードの接続を再開するかどうかリダイレクト宛先ノードを示します。サービスノードがマークダウンときに送信または、トラフィックのハッシュを使用できません。代わりに、トラフィックを送信またはリダイレクト宛先ノードのクラスタ内の異なるサービスノードにハッシュがされます。

一方向のトラフィックのブラック holing を避けるためには、リダイレクト正常性ポリシーサービスノードの入力と出力をリダイレクト宛先ノードを関連付けることができます。これにより入力または出力のいずれかのリダイレクト宛先ノードがダウンしている場合、その他のリダイレクト宛先ノードもマークダウンされます。したがって、入力と出力トラフィックの両方は、リダイレクト宛先ノードのクラスタ内の異なるサービスノードにハッシュを取得します。

しきい値設定

サービス ノードを追跡するため PBR ポリシーを設定するとき、次のしきい値の設定を使用できます。

- しきい値の有効化または無効化：しきい値が有効になっているとき、最小および最大のしきい値のパーセンテージを指定します。リダイレクト先グループを完全に無効にして、リダイレクトを防止したい場合は、有効になっているしきい値は必須です。リダイレクトがないときに、トラフィックがコンシューマとプロバイダ間で直接送信されます。
- 最小しきい値：指定した最小しきい値のパーセンテージ。トラフィックが最小パーセンテージを下回る場合、リダイレクトではなくパケットが許可されます。デフォルト値は 0 です
- 最大しきい値：指定された最大しきい値のパーセンテージ。最小しきい値に達すると、操作状態に戻すため最大パーセンテージに最初に到達する必要があります。デフォルト値は 0 です

例として、ポリシーに 3 つのリダイレクト先があると仮定してみましょう。最小しきい値が 70% に指定されており、最大しきい値が 80% に指定されています。3 つのリダイレクト先ポリシーのいずれかがダウンすると、1/3、つまり最小しきい値以下の 33% 可用性パーセンテージが下がります。その結果、リダイレクト先グループの最小しきい値のパーセンテージがダウンし、トラフィックがリダイレクトではなく許可の取得を開始します。同じ例で続けると、最大しきい値が 80% の場合、リダイレクト ポリシー先グループを操作状態に戻すため、最大しきい値のパーセンテージ以上のパーセンテージに最初に達する必要があります。

リダイレクト トラッキングとサービス ノードに基づくポリシーの注意事項と制約事項

PBR トラッキングおよびサービス ノードを利用するときに、これらの注意事項と制約事項に従います。

- マルチポッドファブリック設定はサポートされています。マルチサイトセットアップはサポートされていません。
- コンシューマとプロバイダ Epg のレイヤ 3 Out はサポートされます。
- リダイレクト宛先ノードの追跡では、TCP または ICMP プロトコルタイプが使用されません。
- ポリシーベース リダイレクトでサポートされる追跡可能 IP アドレスの最大数は、リーフスイッチで 100、ACI ファブリックでは 200 です。
- ACI ファブリックでのグラフ インスタンスの最大数は、ファブリックあたり 1000 です。
- グラフ インスタンスの最大数は、デバイスあたり 100 です。
- PBR を設定できるサービス ノードの最大数は、ポリシーあたり 40 です。

- 1 つのサービス チェーンでサポートされるサービス ノードの最大数は 3 です。
- PBR トラッキングでは、共有サービスがサポートされています。
- 許可アクションまたは拒否アクションはサポートされています。

REST API を使用したサービスノードのトラッキングのサポートをする PBR の設定

手順

トラッキング サービス ノードをサポートする PBR を設定します。

例 :

```
<polUni>
  <fvTenant name="coke" >
    <fvIPSLAMonitoringPol name="tcp_Freq60_Pol1" slaType="tcp" slaFrequency="60"
slaPort="2222" />
    <vnsSvcCont>
      <vnsRedirectHealthGroup name="fwService1"/>
      <vnsSvcRedirectPol name="fwExt" hashingAlgorithm="sip" thresholdEnable="yes"
minThresholdPercent="20" maxThresholdPercent="80">
        <vnsRedirectDest ip="40.40.40.100" mac="00:00:00:00:00:01">
          <vnsRsRedirectHealthGroup
tDn="uni/tn-coke/svcCont/redirectHealthGroup-fwService1"/>
        </vnsRedirectDest>
        <vnsRsIPSLAMonitoringPol tDn="uni/tn-coke/ipslaMonitoringPol-tcp_Freq60_Pol1"/>
      </vnsSvcRedirectPol>
      <vnsSvcRedirectPol name="fwInt" hashingAlgorithm="sip" thresholdEnable="yes"
minThresholdPercent="20" maxThresholdPercent="80">
        <vnsRedirectDest ip="30.30.30.100" mac="00:00:00:00:00:02">
          <vnsRsRedirectHealthGroup
tDn="uni/tn-coke/svcCont/redirectHealthGroup-fwService1"/>
        </vnsRedirectDest>
        <vnsRsIPSLAMonitoringPol
tDn="uni/tn-coke/ipslaMonitoringPol-tcp_Freq60_Pol1"/>
      </vnsSvcRedirectPol>
    </vnsSvcCont>
  </fvTenant>
</polUni>
```

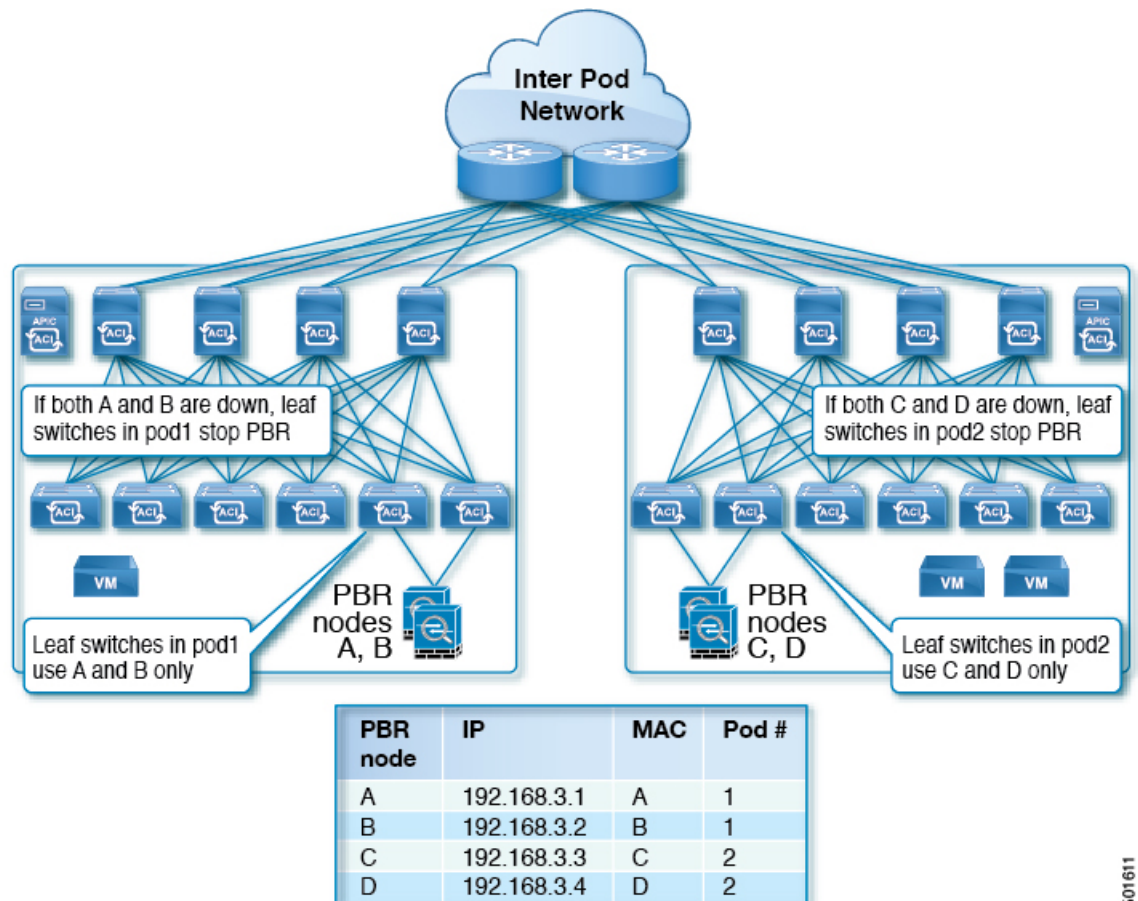
ベース リダイレクトの場所に対応したポリシーについて

ロケーション対応ポリシー ベースのリダイレクト (PBR) はサポートされています。この機能は、multipod 設定シナリオに役立ちます。ここでは、ポッド認識サポートされ、優先ローカル PBR ノードを指定できます。ロケーション対応のリダイレクトを有効にすると、ポッド Id が指定されて、レイヤ4~レイヤ7 PBR ポリシー内のすべてのリダイレクト宛先はポッド認識必

必要があります。リダイレクト宛先は、特定のポッドにあるリーフスイッチでのみプログラムされます。

次の図は、2 個のポッドの例を表示します。ポッド 1 で PBR ノード A と B、C と D PBR ノードがポッド 2 では。ポッド 1 のリーフスイッチが A、B、PBR ノードを使用する prefer し、ポッド 2 のリーフスイッチ C と D で PBR ノードの使用場所に対応した PBR 設定を有効にすると PBR ノード A と B ポッド 1 では、ダウンは、[ポッド 1 のリーフスイッチと開始 PBR ノード C と D を使用するには同様に、PBR ノード C と D ポッド 2 では、ダウンが、ポッド 2 のリーフスイッチと開始 PBR ノード A および B を使用するには

図 1: 2 個のポッドのロケーション対応 PBR 設定の例



501611

ロケーション認識型 PBR の注意事項

ロケーション認識型 PBR を活用する際はこれらの注意事項に従ってください。

- Cisco Nexus 9300 (Cisco Nexus 9300 EX および 9300 FX を除く) プラットフォーム スイッチは、ロケーション認識型 PBR 機能をサポートしていません。
- GOLF ホストアドバタイズメントと北南ファイアウォール連携にロケーション認識型 PBR を使用します。

REST API を使用して設定の場所に対応した PBR

2つ設定する必要があります項目の場所に対応した PBR を有効にして、プログラムが特定のポッドにあるリーフスイッチ内の送信先をリダイレクトします。次の例の場所に対応した PBR を有効にするよう設定されている属性が: `programLocalPodOnly` と `podId` 。

手順

ロケーション対応 PBR を設定します。

例 :

```
<polUni>
  <fvTenant name="coke" >
    <fvIPSLAMonitoringPol name="icmp_Freq60_Pol1" slaType="icmp" slaFrequency="60"/>
    <vnsSvcCont>
      <vnsRedirectHealthGroup name="fwService1"/>
      <vnsSvcRedirectPol name="fwExt" hashingAlgorithm="sip" thresholdEnable="yes"
minThresholdPercent="20" maxThresholdPercent="80" programLocalPodOnly="yes">
        <vnsRedirectDest ip="40.40.40.100" mac="00:00:00:00:00:01" podId="2">
          <vnsRsRedirectHealthGroup
tDn="uni/tn-coke/svcCont/redirectHealthGroup-fwService1"/>
        </vnsRedirectDest>
        <vnsRsIPSLAMonitoringPol
tDn="uni/tn-coke/ipslaMonitoringPol-icmp_Freq60_Pol1"/>
      </vnsSvcRedirectPol>
      <vnsSvcRedirectPol name="fwInt" hashingAlgorithm="dip" thresholdEnable="yes"
minThresholdPercent="20" maxThresholdPercent="80">
        <vnsRedirectDest ip="30.30.30.100" mac="00:00:00:00:00:02">
          <vnsRsRedirectHealthGroup
tDn="uni/tn-coke/svcCont/redirectHealthGroup-fwService1"/>
        </vnsRedirectDest>
        <vnsRsIPSLAMonitoringPol
tDn="uni/tn-coke/ipslaMonitoringPol-icmp_Freq60_Pol1"/>
      </vnsSvcRedirectPol>
    </vnsSvcCont>
  </fvTenant>
</polUni>
```

サービス グラフ テンプレート

サービス グラフ テンプレートについて

Cisco Application Centric Infrastructure (ACI) では、特定のタイプのファイアウォールとそれに続く特定のモデルおよびバージョンのロードバランサといった一連のメタデバイスを定義できます。これは、サービス グラフ テンプレートと呼ばれ、また、抽象グラフとも呼ばれます。抽象サービス グラフ テンプレートがコントラクトによって参照されると、サービス グラフ テンプレートはファブリック内に存在するファイアウォールやロードバランサなどの具象デバイスにマッピングすることでインスタンス化されます。マッピングは「コンテキスト」の概念で発

生じます。「デバイス コンテキスト」とは、ACI がサービス グラフ テンプレートにマッピングできるファイアウォールおよびロードバランサを特定できるようにするためのマッピング設定です。もう1つの重要な概念は、具象デバイスのクラスタを表す「論理デバイス」です。サービス グラフ テンプレートのレンダリングは、コントラクトによって定義されるパスに挿入可能な適切な論理デバイスの識別に基づいています。

ACIはサービスをアプリケーションの重要部分と見なします。必要とされるすべてのサービスが、Cisco Application Policy Infrastructure Controller (APIC) から ACI ファブリックでインスタンス化されるサービスグラフとして扱われます。ユーザは、アプリケーションに対してサービスを定義し、サービスグラフテンプレートはアプリケーションが必要とする一連のネットワークまたはサービス機能を識別します。グラフを APIC に設定すると、APIC はサービス グラフ テンプレートで指定されたサービス機能要件に基づいてサービスを自動的に設定します。さらに APIC は、サービス グラフ テンプレートで指定されたサービス機能のニーズに応じてネットワークを自動的に設定しますが、これによってサービスデバイスでの変更が必要になることはありません。

REST API を使用したサービス グラフ テンプレートの設定

次の REST API を使用してサービス グラフ テンプレートを設定できます。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <!--L3 Network-->
    <fvCtx name="MyNetwork"/>
    <!-- Bridge Domain for MySrvr EPG -->
    <fvBD name="MySrvrBD">
      <fvRsCtx tnFvCtxName="MyNetwork" />
      <fvSubnet ip="10.10.10.10/24">
      </fvSubnet>
    </fvBD>
    <!-- Bridge Domain for MyClnt EPG -->
    <fvBD name="MyClntBD">
      <fvRsCtx tnFvCtxName="MyNetwork" />
      <fvSubnet ip="20.20.20.20/24">
      </fvSubnet>
    </fvBD>
    <fvAp dn="uni/tn-acme/ap-MyAP" name="MyAP">
      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MyClnt" name="MyClnt">
        <fvRsBd tnFvBDName="MySrvrBD" />
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
        <fvRsProv tnVzBrCPName="webCtrct">
        </fvRsProv>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]"
encap="vlan-202"/>
        <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]"
encap="vlan-202"/>
      </fvAEPg>
      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MySRVR" name="MySRVR">
        <fvRsBd tnFvBDName="MyClntBD" />
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs" />
        <fvRsCons tnVzBrCPName="webCtrct">
        </fvRsCons>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]"
encap="vlan-203"/>
        <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]"
encap="vlan-203"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```

```

    </fvAp>
  </fvTenant>
</polUni>

```

REST API を使用したセキュリティ ポリシーの作成

次の REST API を使用してセキュリティ ポリシーを作成することができます。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vzFilter name="HttpIn">
      <vzEntry name="e1" prot="6" dToPort="80"/>
    </vzFilter>
    <vzBrCP name="webCtrct">
      <vzSubj name="http">
        <vzRsSubjFiltAtt tnVzFilterName="HttpIn"/>
      </vzSubj>
    </vzBrCP>
  </fvTenant>
</polUni>

```

レイヤ4~レイヤ7パラメータ

展開済みのサービス グラフの設定パラメータの変更について

サービス グラフを最初に展開すると、サービス グラフを正常に展開する前に、サービス グラフの設定パラメータまたは機能が拒否されます。これらの設定パラメータまたは機能には、IP アドレス、ルート プレフィックス、ネクスト ホップの情報などのデバイス ネットワーク設定や、ファイアウォールの IP アクセスリストまたはロードバランサのサーバロードバランシング設定などサービス設定を含みます。

Application Policy Infrastructure Controller (APIC) の日常業務の一部としてサービス グラフ機能を変更する必要があります。APIC の GUI または CLI を使用して、サービス グラフの設定パラメータと機能を変更できます。APIC を通じたサービス デバイスの機能変更には、サービス デバイス上の変更は必要ありません。

コンフィギュレーションパラメータを持つアプリケーション EPG の XML POST の例

次の XML の例は、デバイスパッケージ内のコンフィギュレーションパラメータを示しています。

```

<fvAEPg dn="uni/tn-acme/ap-myApp/epg-app" name="app">
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
key="Monitor"
  name="monitor1">
    <vnsRsFolderInstToMFolder
tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Monitor"/>
    <vnsParamInst name="weight" key="weight" value="10"/>
  </vnsFolderInst>
</fvAEPg>

```



```

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
key="Service"
  name="Service1">
  <vnsParamInst name="servicename" key="servicename" value="crpvgrtst02-8010"/>
  <vnsParamInst name="servicetype" key="servicetype" value="TCP"/>
  <vnsParamInst name="servername" key="servername" value="s192.168.100.100"/>
  <vnsParamInst name="serveripaddress" key="serveripaddress"
value="192.168.100.100"/>
  <vnsParamInst name="serviceport" key="serviceport" value="8080"/>
  <vnsParamInst name="svrtimeout" key="svrtimeout" value="9000" />
  <vnsParamInst name="clttimeout" key="clttimeout" value="9000" />
  <vnsParamInst name="usip" key="usip" value="NO" />
  <vnsParamInst name="useproxyport" key="useproxyport" value="" />
  <vnsParamInst name="cip" key="cip" value="ENABLED" />
  <vnsParamInst name="cka" key="cka" value="NO" />
  <vnsParamInst name="sp" key="sp" value="OFF" />
  <vnsParamInst name="cmp" key="cmp" value="NO" />
  <vnsParamInst name="maxclient" key="maxclient" value="0" />
  <vnsParamInst name="maxreq" key="maxreq" value="0" />
  <vnsParamInst name="tcpb" key="tcpb" value="NO" />
  <vnsCfgRelInst name="MonitorConfig" key="MonitorConfig" targetName="monitor1"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any"
key="Network"
  name="Network">
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any"
key="vip"
  name="vip">
  <vnsParamInst name="vipaddress1" key="vipaddress" value="10.10.10.200"/>
</vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any"
devCtxLbl="C1" key="snip" name="snip1">
  <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.200"/>
</vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G2" nodeNameOrLbl="any"
devCtxLbl="C2" key="snip" name="snip2">
</vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
key="Network"
  name="Network">
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
key="vip"
  name="vip">
  <vnsParamInst name="vipaddress1" key="vipaddress" value="10.10.10.100"/>
</vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
devCtxLbl="C1" key="snip" name="snip1">
  <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.100"/>
</vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
devCtxLbl="C2" key="snip" name="snip2">
  <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.101"/>
</vnsFolderInst>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="G1" nodeNameOrLbl="any"
devCtxLbl="C3" key="snip" name="snip3">
  <vnsParamInst name="snipaddress" key="snipaddress" value="192.168.1.102"/>
</vnsFolderInst>
</vnsFolderInst>
</vnsFolderInst>

<!-- SLB Configuration -->
<vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
key="VServer"

```

```

name="VServer">
  <!-- Virtual Server Configuration -->
  <vnsParamInst name="port" key="port" value="8010"/>
  <vnsParamInst name="vip" key="vip" value="10.10.10.100"/>
  <vnsParamInst name="vservername" key="vservername" value="crpvgrtst02-vip-8010"/>

  <vnsParamInst name="servicename" key="servicename" value="crpvgrtst02-8010"/>
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
    key="VServerGlobalConfig" name="VServerGlobalConfig">
    <vnsCfgRelInst name="ServiceConfig" key="ServiceConfig" targetName="Service1"/>

    <vnsCfgRelInst name="VipConfig" key="VipConfig" targetName="Network/vip"/>
  </vnsFolderInst>
</vnsFolderInst>
</fvAEPg>

```

デバイスパッケージ内のコンフィギュレーションパラメータのXMLの例

次のXMLの例は、デバイスパッケージ内のコンフィギュレーションパラメータを示しています。

```

<vnsMFolder key="VServer" scopedBy="epg">
  <vnsRsConnector tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external"/>
  <vnsMParam key="vservername" description="Name of VServer" mandatory="true"/>
  <vnsMParam key="vip" description="Virtual IP"/>
  <vnsMParam key="subnet" description="Subnet IP"/>
  <vnsMParam key="port" description="Port for Virtual server"/>
  <vnsMParam key="persistencetype" description="persistencetype"/>
  <vnsMParam key="servicename" description="Service bound to this vServer"/>
  <vnsMParam key="servicetype" description="Service bound to this vServer"/>
  <vnsMParam key="clttimeout" description="Client timeout"/>
  <vnsMFolder key="VServerGlobalConfig"
    description="This references the global configuration">
    <vnsMRel key="ServiceConfig">
      <vnsRsTarget tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Service"/>
    </vnsMRel>
    <vnsMRel key="ServerConfig">
      <vnsRsTarget tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Server"/>
    </vnsMRel>
    <vnsMRel key="VipConfig">
      <vnsRsTarget
        tDn="uni/infra/mDev-Acme-ADC-1.0/mDevCfg/mFolder-Network/mFolder-vip"/>
      <vnsRsConnector tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external"/>
    </vnsMRel>
  </vnsMFolder>
</vnsMFolder>

```

コンフィギュレーションパラメータを持つ抽象機能ノードに対するXML POSTの例

次のXML POSTの例は、コンフィギュレーションパラメータを持つ抽象機能ノードを示しています。

```

<vnsAbsNode name = "SLB" funcType="GoTo" >
  <vnsRsDefaultScopeToTerm tDn="uni/tn-tenant1/AbsGraph-G3/AbsTermNode-Output1/outtmn1"/>

```

```

<vnsAbsFuncConn name = "C4" direction = "input">
  <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external" />
</vnsAbsFuncConn>
<vnsAbsFuncConn name = "C5" direction = "output">
  <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-internal" />
</vnsAbsFuncConn>

<vnsAbsDevCfg>
  <vnsAbsFolder key="Network" name="Network" scopedBy="epg">
    <!-- Following scopes this folder to input terminal or Src Epg -->
    <vnsRsScopeToTerm
tDn="uni/tn-tenant1/AbsGraph-G3/AbsTermNode-Output1/outtmn1"/>

    <!-- VIP address -->
    <vnsAbsFolder key="vip" name="vip" scopedBy="epg">
      <vnsAbsParam name="vipaddress" key="vipaddress" value=""/>
    </vnsAbsFolder>

    <!-- SNIP address -->
    <vnsAbsFolder key="snip" name="snip" scopedBy="epg">
      <vnsAbsParam name="snipaddress" key="snipaddress" value=""/>
    </vnsAbsFolder>

  </vnsAbsFolder>

  <vnsAbsFolder key="Service" name="Service" scopedBy="epg" cardinality="n">
    <vnsRsScopeToTerm
tDn="uni/tn-tenant1/AbsGraph-G3/AbsTermNode-Output1/outtmn1"/>
    <vnsAbsParam name="servicename" key="servicename" value=""/>
    <vnsAbsParam name="servername" key="servername" value=""/>
    <vnsAbsParam name="serveripaddress" key="serveripaddress" value=""/>
  </vnsAbsFolder>
</vnsAbsDevCfg>

<vnsAbsFuncCfg>
  <vnsAbsFolder key="VServer" name="VServer" scopedBy="epg">
    <vnsRsScopeToTerm
tDn="uni/tn-tenant1/AbsGraph-G3/AbsTermNode-Output1/outtmn1"/>
    <!-- Virtual Server Configuration -->
    <vnsAbsParam name="vip" key="vip" value=""/>
    <vnsAbsParam name="vservername" key="vservername" value=""/>
    <vnsAbsParam name="servicename" key="servicename"/>
    <vnsRsCfgToConn tDn="uni/tn-tenant1/AbsGraph-G3/AbsNode-Node2/AbsFConn-C4"
/>
  </vnsAbsFolder>
</vnsAbsFuncCfg>
<vnsRsNodeToMFunc tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB"/>
</vnsAbsNode>

```

コンフィギュレーションパラメータを持つ抽象機能プロファイルに対するXML POSTの例

次のXML POSTの例は、コンフィギュレーションパラメータを持つ抽象機能プロファイルを示しています。

```

<vnsAbsFuncProfContr name = "NP">
  <vnsAbsFuncProfGrp name = "Grp1">
    <vnsAbsFuncProf name = "P1">
      <vnsRsProfToMFunc tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB"/>
    <vnsAbsDevCfg name="D1">

```

```

<vnsAbsFolder key="Service" name="Service-Default" cardinality="n">
  <vnsAbsParam name="servicetype" key="servicetype" value="TCP"/>
  <vnsAbsParam name="serviceport" key="serviceport" value="80"/>
  <vnsAbsParam name="maxclient" key="maxclient" value="1000"/>
  <vnsAbsParam name="maxreq" key="maxreq" value="100"/>
  <vnsAbsParam name="cip" key="cip" value="enable"/>
  <vnsAbsParam name="usip" key="usip" value="enable"/>
  <vnsAbsParam name="sp" key="sp" value=""/>
  <vnsAbsParam name="svrtimeout" key="svrtimeout" value="60"/>
  <vnsAbsParam name="clttimeout" key="clttimeout" value="60"/>
  <vnsAbsParam name="cka" key="cka" value="NO"/>
  <vnsAbsParam name="tcpb" key="tcpb" value="NO"/>
  <vnsAbsParam name="cmp" key="cmp" value="NO"/>
</vnsAbsFolder>
</vnsAbsDevCfg>
<vnsAbsFuncCfg name="SLB">
  <vnsAbsFolder key="VServer" name="VServer-Default">
    <vnsAbsParam name="port" key="port" value="80"/>
    <vnsAbsParam name="persistencetype" key="persistencetype"
      value="cookie"/>
    <vnsAbsParam name="clttimeout" key="clttimeout" value="100"/>
    <vnsAbsParam name="servicetype" key="servicetype" value="TCP"/>
    <vnsAbsParam name="servicename" key="servicename"/>
  </vnsAbsFolder>
</vnsAbsFuncCfg>
</vnsAbsFuncProf>
</vnsAbsFuncProfGrp>
</vnsAbsFuncProfContr>

```

Copy Services

コピー サービスについて

Switched Port Analyzer (SPAN;)とは異なり、すべてのトラフィックが重複するが、Cisco Application Centric Infrastructure (ACI) コピー サービス機能により、コントラクトの仕様に従って、エンドポイントグループ間のトラフィックの一部を選択的にコピーできます。コントラクト対象のブロードキャスト、不明なユニキャストおよびマルチキャスト (BUM)、およびコントロールプラントラフィックはコピーされません。対照的に、SPAN はエンドポイントグループ、アクセスポート、アップリンクポート以外のすべてをコピーします。SPANとは異なり、コピーサービスは、コピーされたトラフィックにヘッダーを追加しません。コピーサービスのトラフィックは、通常のトラフィックの転送への影響を最小限に抑えるため、スイッチ内で内部的に管理されます。

レイヤ4～レイヤ7サービスの展開の詳細については、「Cisco APIC レイヤ4～レイヤ7サービス展開ガイド」を参照してください。

REST API を使用してコピー サービスの設定

コピー デバイスは、copy ノードを作成するコピー サービス機能の一部として使用されます。コピーのノードは、トラフィックをコピーするエンドポイントグループ間のデータフローのどの時点を指定します。

この手順では、REST API を使用してコピー サービスを設定する例を提供します。



- (注) コピー デバイスを設定すると、**context aware** パラメータは使用されません。**context aware** パラメータには `single context` というデフォルト値がありますが、これは無視されます。

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 コピー デバイスを作成します。

例：

```
<vnsLDevVip contextAware="single-Context" devtype="PHYSICAL" funcType="None" isCopy="yes"
  managed="no"
  mode="legacy-Mode" name="copy0" packageModel="" svcType="COPY" trunking="no">
  <vnsRsALDevToPhysDomP tDn="uni/phys-phys_scale_copy"/>
  <vnsCDev devCtxLbl="" name="copy_Dyn_Device_0" vcenterName="" vmName="">
  <vnsCIf name="int1" vnicName="">
  <vnsRsCIfPathAtt tDn="topology/pod-1/paths-104/pathep-[eth1/15]"/>
  </vnsCIf>
  <vnsCIf name="int2" vnicName="">
  <vnsRsCIfPathAtt tDn="topology/pod-1/paths-105/pathep-[eth1/15]"/>
  </vnsCIf>
  </vnsCDev>
  <vnsLIf encap="vlan-3540" name="TAP">
  <vnsRsCIfAttN tDn="uni/tn-t22/lDevVip-copy0/cDev-copy_Dyn_Device_0/cIf-[int2]"/>

  <vnsRsCIfAttN tDn="uni/tn-t22/lDevVip-copy0/cDev-copy_Dyn_Device_0/cIf-[int1]"/>

  </vnsLIf>
</vnsLDevVip>
```

ステップ2 論理デバイス コンテキスト (デバイス選択ポリシーとも呼ばれる) を作成します。

例：

```
<vnsLDevCtx ctrctNameOrLbl="c0" descr="" graphNameOrLbl="g0" name="" nodeNameOrLbl="CP1">

  <vnsRsLDevCtxToLDev tDn="uni/tn-t22/lDevVip-copy0"/>
  <vnsLIfCtx connNameOrLbl="copy" descr="" name="">
  <vnsRsLIfCtxToLIf tDn="uni/tn-t22/lDevVip-copy0/lIf-TAP"/>
  </vnsLIfCtx>
</vnsLDevCtx>
```

ステップ3 作成し、コピーするグラフ テンプレートを適用します。

例：

```
<vnsAbsGraph descr="" name="g0" ownerKey="" ownerTag="" uiTemplateType="UNSPECIFIED">
  <vnsAbsTermNodeCon descr="" name="T1" ownerKey="" ownerTag="">
  <vnsAbsTermConn attNotify="no" descr="" name="1" ownerKey="" ownerTag=""/>
  <vnsInTerm descr="" name=""/>
  <vnsOutTerm descr="" name=""/>
</vnsAbsTermNodeCon>
```

```

<vnsAbsTermNodeProv descr="" name="T2" ownerKey="" ownerTag="">
  <vnsAbsTermConn attNotify="no" descr="" name="1" ownerKey="" ownerTag=""/>
  <vnsInTerm descr="" name=""/>
  <vnsOutTerm descr="" name=""/>
</vnsAbsTermNodeProv>
<vnsAbsConnection adjType="L2" connDir="provider" connType="external" descr=""
name="C1"
  ownerKey="" ownerTag="" unicastRoute="yes">
  <vnsRsAbsConnectionConns tDn="uni/tn-t22/AbsGraph-g0/AbsTermNodeCon-T1/AbsTConn"/>

  <vnsRsAbsConnectionConns tDn="uni/tn-t22/AbsGraph-g0/AbsTermNodeProv-T2/AbsTConn"/>

  <vnsRsAbsCopyConnection tDn="uni/tn-t22/AbsGraph-g0/AbsNode-CP1/AbsFConn-copy"/>

</vnsAbsConnection>
<vnsAbsNode descr="" funcTemplateType="OTHER" funcType="None" isCopy="yes"
managed="no"
  name="CP1" ownerKey="" ownerTag="" routingMode="unspecified" sequenceNumber="0"
  shareEncap="no">
  <vnsAbsFuncConn attNotify="no" descr="" name="copy" ownerKey="" ownerTag=""/>
  <vnsRsNodeToLDev tDn="uni/tn-t22/lDevVip-copy0"/>
</vnsAbsNode>
</vnsAbsGraph>

```

ステップ4 エンドポイントのグループに関連付けられている契約でコピー グラフに関係を定義します。

例：

```

<vzBrCP descr="" name="c0" ownerKey="" ownerTag="" prio="unspecified" scope="tenant"
targetDscp="unspecified">
  <vzSubj consMatchT="AtleastOne" descr="" name="Subject" prio="unspecified"
provMatchT="AtleastOne"
  revFltPorts="yes" targetDscp="unspecified">
  <vzRsSubjFiltAtt directives="" tnVzFilterName="default"/>
  <vzRsSubjGraphAtt directives="" tnVnsAbsGraphName="g0"/>
</vzSubj>
</vzBrCP>

```

ステップ5 エンドポイント グループを契約を接続します。

例：

```

<fvAEPg name="epg2860">
  <fvRsCons tnVzBrCPName="c0"/>
  <fvRsBd tnFvBDName="bd0"/>
  <fvRsDomAtt tDn="uni/phys-phys_scale_SB"/>
  <fvRsPathAtt tDn="topology/pod-1/paths-104/pathep-[PC_int2_g1]" encap="vlan-2860"

  instrImedcy="immediate"/>
</fvAEPg>
<fvAEPg name="epg2861">
  <fvRsProv tnVzBrCPName="c0"/>
  <fvRsBd tnFvBDName="bd0"/>
  <fvRsDomAtt tDn="uni/phys-phys_scale_SB"/>
  <fvRsPathAtt tDn="topology/pod-1/paths-105/pathep-[PC_policy]" encap="vlan-2861"

  instrImedcy="immediate"/>
</fvAEPg>

```

自動化の開発

REST API について

自動化は、Application Policy Infrastructure Controller (APIC) のノースバウンド Representational State Transfer (REST) API を使用します。APIC UI を通じて実行できる処理はすべて、ノースバウンド API を使用した XML ベースの REST POST を使用して実行できます。たとえば、これらの API 経由でのイベントのモニタ、EPG のダイナミックな有効化、およびポリシーの追加などを実行できます。

また、ノースバウンド REST API を使用して、デバイスがオンボードになったことの通知や、エラーをモニタできます。両方のケースで特定のアクションをトリガするイベントをモニタできます。たとえば、特定のアプリケーション層で発生したエラーを検出し、接続の切断がありリーフノードがダウンした場合、これらのアプリケーションを他の場所に再展開するアクションをトリガできます。パケットドロップが検出された特定のコントラクトがある場合、これらのコントラクトの複数のコピーを特定のアプリケーション上で有効化できます。また、レポートされた問題に基づいて特定のカウンタをモニタできる統計モニタリングポリシーを使用できます。

APIC ノースバウンド API にサブミットされた XML ファイルを構成する方法については、『Cisco APIC Layer 4 to Layer 7 Device Package Development Guide』を参照してください。

『Cisco APIC Management Information Model Reference』で定義されている次の Python API はノースバウンド API を使用した REST POST コールのサブミットに使用できます。

- vns:LDevVip : デバイス クラスタをアップロードします
- vns:CDev : デバイスをアップロードします
- vns:LIF : 論理インターフェイスを作成します
- vns:AbsGraph : グラフを作成します
- vz:BrCP : 契約にグラフを追加します

REST API を使用した自動化の例

ここでは、REST API を使用してタスクを自動化する例を示します。

次の REST 要求は、ブロードキャストドメインを持つテナント、レイヤ3ネットワーク、アプリケーションエンドポイントグループ、およびアプリケーションプロファイルを作成します。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">

    <!--L3 Network-->
    <fvCtx name="MyNetwork"/>

    <!-- Bridge Domain for MySrvr EPG -->
    <fvBD name="MySrvrBD">
      <fvRsCtx tnFvCtxName="MyNetwork"/>
      <fvSubnet ip="10.10.10.10/24">
    </fvSubnet>
```

```

</fvBD>

<!-- Bridge Domain for MyClnt EPG -->
<fvBD name="MyClntBD">
  <fvRsCtx tnFvCtxName="MyNetwork"/>
  <fvSubnet ip="20.20.20.20/24">
  </fvSubnet>
</fvBD>

<fvAp dn="uni/tn-acme/ap-MyAP" name="MyAP">

  <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MyClnt" name="MyClnt">
    <fvRsBd tnFvBDName="MySrvrBD"/>
    <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs"/>
    <fvRsProv tnVzBrCPName="webCtrct"> </fvRsProv>
    <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]"
      encap="vlan-202"/>
    <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]"
      encap="vlan-202"/>
  </fvAEPg>

  <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MySRVR" name="MySRVR">
    <fvRsBd tnFvBDName="MyClntBD"/>
    <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs"/>
    <fvRsCons tnVzBrCPName="webCtrct"> </fvRsCons>
    <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]"
      encap="vlan-203"/>
    <fvRsPathAtt tDn="topology/pod-1/paths-18/pathep-[eth1/21]"
      encap="vlan-203"/>
  </fvAEPg>
</fvAp>
</fvTenant>
</polUni>

```

次の REST 要求は VLAN ネームスペースを作成します。

```

<polUni>
  <infraInfra>
    <fvnsVlanInstP name="MyNS" allocMode="dynamic">
      <fvnsEncapBlk name="encap" from="vlan-201" to="vlan-300"/>
    </fvnsVlanInstP>
  </infraInfra>
</polUni>

```

次の REST 要求は VMM ドメインを作成します。

```

<polUni>
  <vmmProvP vendor="Vendor1">
    <vmmDomP name="MyVMs">
      <infraRsVlanNs tDn="uni/infra/vlanns-MyNS-dynamic"/>
      <vmmUsrAccP name="admin" usr="administrator" pwd="in$leme"/>
      <vmmCtrlrP name="vcenter1" hostOrIp="192.168.64.186">
        <vmmRsAcc tDn="uni/vmmp-Vendor1/dom-MyVMs/usracc-admin"/>
      </vmmCtrlrP>
    </vmmDomP>
  </vmmProvP>
</polUni>

```

次の REST 要求は物理ドメインを作成します。

```

<polUni>
  <physDomP name="phys">
    <infraRsVlanNs tDn="uni/infra/vlanns-MyNS-dynamic"/>
  </physDomP>
</polUni>

```


次の REST 要求は管理対象デバイス クラスタを作成します。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1" contextAware=1>
      <vnsRsMDevAtt tDn="uni/infra/mDev-Acme-ADC-1.0"/>
      <vnsRsDevEpg tDn="uni/tn-acme/ap-services/epg-ifc"/>
      <vnsRsALDevToPhysDomP tDn="uni/phys-phys"/>

      <vnsCMgmt name="devMgmt" host="42.42.42.100" port="80"/>

      <vnsCCred name="username" value="admin"/>

      <vnsCCredSecret name="password" value="admin"/>
    </vnsLDevVip>
  </fvTenant>
</polUni>
```

次の REST 要求は非管理対象デバイス クラスタを作成します。

```
<polUni>
  <fvTenant name="HA_Tenant1">

    <vnsLDevVip name="ADCCluster1" devtype="VIRTUAL" managed="no">
      <vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-mininet"/>
    </vnsLDevVip>

  </fvTenant>
</polUni>
```

次の REST 要求はデバイス クラスタ コンテキストを作成します。

```
<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevCtx ctrctNameOrLbl="webCtrct" graphNameOrLbl="G1" nodeNameOrLbl="Node1">

      <vnsRsLDevCtxToLDev tDn="uni/tn-acme/lDevVip-ADCCluster1"/>
      <vnsLIIfCtx connNameOrLbl="ssl-inside">
        <vnsRsLIIfCtxToLIIf tDn="uni/tn-acme/lDevVip-ADCCluster1/lIf-int"/>
      </vnsLIIfCtx>
      <vnsLIIfCtx connNameOrLbl="any">
        <vnsRsLIIfCtxToLIIf tDn="uni/tn-acme/lDevVip-ADCCluster1/lIf-ext"/>
      </vnsLIIfCtx>
    </vnsLDevCtx>
  </fvTenant>
</polUni>
```

次の要求は、ルーティング ピアリングに使用されるデバイス クラスタ コンテキストを作成します。

```
<polUni>
  <fvTenant dn="uni/tn-coke{{tenantId}}" name="coke{{tenantId}}">
    <vnsRtrCfg name="Dev1Ctx1" rtrId="180.0.0.12"/>
    <vnsLDevCtx ctrctNameOrLbl="webCtrct1" graphNameOrLbl="WebGraph"
      nodeNameOrLbl="FW">
      <vnsRsLDevCtxToLDev tDn="uni/tn-tenant1/lDevVip-Firewall"/>
      <vnsRsLDevCtxToRtrCfg tnVnsRtrCfgName="FwRtrCfg"/>
      <vnsLIIfCtx connNameOrLbl="internal">
        <vnsRsLIIfCtxToInstP
tDn="uni/tn-tenant1/out-OspfInternal/instP-IntInstP"
          status="created,modified"/>
        <vnsRsLIIfCtxToLIIf tDn="uni/tn-tenant1/lDevVip-Firewall/lIf-internal"/>
      </vnsLIIfCtx>
      <vnsLIIfCtx connNameOrLbl="external">
```

```

        <vnsRsLIfCtxToInstP tDn="uni/tn-common/out-OspfExternal/instP-ExtInstP"
            status="created,modified"/>
        <vnsRsLIfCtxToLIf tDn="uni/tn-tenant1/lDevVip-Firewall/lIf-external"/>
    </vnsLIfCtx>
</vnsLDevCtx>
</fvTenant>
</polUni>

```



- (注) テナント（レイヤ3 Outside）の外部接続の設定については、『Cisco APIC ベーシック コンフィギュレーション ガイド』を参照してください。

次の REST 要求はデバイス クラスタの論理インターフェイスを追加します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1">
      <vnsLIf name="C5">
        <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-outside"/>
        <vnsRsCIfAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-int"/>
      </vnsLIf>
      <vnsLIf name="C4">
        <vnsRsMetaIf tDn="uni/infra/mDev-Acme-ADC-1.0/mIfLbl-inside"/>
        <vnsRsCIfAtt tDn="uni/tn-acme/lDevVip-ADCCluster1/cDev-ADC1/cIf-ext"/>
      </vnsLIf>
    </vnsLDevVip>
  </fvTenant>
</polUni>

```

次の REST 要求は物理デバイス クラスタの具象デバイスを追加します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">
    <vnsLDevVip name="ADCCluster1">
      <vnsCDev name="ADC1" devCtxLbl="C1">
        <vnsCIf name="int">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/22]"/>
        </vnsCIf>
        <vnsCIf name="ext">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/21]"/>
        </vnsCIf>
        <vnsCIf name="mgmt">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/20]"/>
        </vnsCIf>
        <vnsCMgmt name="devMgmt" host="172.30.30.100" port="80"/>
        <vnsCCred name="username" value="admin"/>
        <vnsCCred name="password" value="admin"/>
      </vnsCDev>
      <vnsCDev name="ADC2" devCtxLbl="C2">
        <vnsCIf name="int">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/23]"/>
        </vnsCIf>
        <vnsCIf name="ext">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/24]"/>
        </vnsCIf>
        <vnsCIf name="mgmt">
          <vnsRsCIfPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/30]"/>
        </vnsCIf>
        <vnsCMgmt name="devMgmt" host="172.30.30.200" port="80"/>
        <vnsCCred name="username" value="admin"/>
      </vnsCDev>
    </vnsLDevVip>
  </fvTenant>
</polUni>

```

```

        <vnsCCred name="password" value="admin"/>
    </vnsCDev>
</vnsLDevVip>
</fvTenant>
</polUni>

```

次の REST 要求は仮想デバイス クラスタの具象デバイスを追加します。

```

<polUni>
  <fvTenant dn="uni/tn-coke5" name="coke5">
    <vnsLDevVip name="Firewall15" devtype="VIRTUAL">
      <vnsCDev name="ASA5" vcenterName="vcenter1" vmName="ifav16-ASAv-scale-05">
        <vnsCIf name="Gig0/0" vnicName="Network adapter 2"/>
        <vnsCIf name="Gig0/1" vnicName="Network adapter 3"/>
        <vnsCIf name="Gig0/2" vnicName="Network adapter 4"/>
        <vnsCIf name="Gig0/3" vnicName="Network adapter 5"/>
        <vnsCIf name="Gig0/4" vnicName="Network adapter 6"/>
        <vnsCIf name="Gig0/5" vnicName="Network adapter 7"/>
        <vnsCIf name="Gig0/6" vnicName="Network adapter 8"/>
        <vnsCIf name="Gig0/7" vnicName="Network adapter 9"/>
        <vnsCMgmt name="devMgmt" host="3.5.3.170" port="443"/>
        <vnsCCred name="username" value="admin"/>
        <vnsCCredSecret name="password" value="insieme"/>
      </vnsCDev>
    </vnsLDevVip>
  </fvTenant>
</polUni>

```

次の REST 要求は管理対象サービス グラフを作成します。

```

<polUni>
  <fvTenant name="acme">
    <vnsAbsGraph name = "G1">

      <vnsAbsTermNode name = "Input1">
        <vnsAbsTermConn name = "C1" direction = "output">
          </vnsAbsTermConn>
        </vnsAbsTermNode>

        <!-- Node1 Provides SLB functionality -->
        <vnsAbsNode name = "Node1" funcType="GoTo" >
          <vnsRsDefaultScopeToTerm
            tDn="uni/tn-acme/AbsGraph-G1/AbsTermNode-Output1/outtmnl"/>

          <vnsAbsFuncConn name = "C4" direction = "input">
            <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-external"/>

            <vnsRsConnToLIf tDn="uni/tn-acme/lDevVip-ADCcluster1/lIf-C4"/>
          </vnsAbsFuncConn>

          <vnsAbsFuncConn name = "C5" direction = "output">
            <vnsRsMConnAtt tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB/mConn-internal"/>

            <vnsRsConnToLIf tDn="uni/tn-acme/lDevVip-ADCcluster1/lIf-C5"/>
          </vnsAbsFuncConn>

          <vnsRsNodeToMFunc tDn="uni/infra/mDev-Acme-ADC-1.0/mFunc-SLB"/>
        </vnsAbsNode>

        <vnsAbsTermNode name = "Output1">
          <vnsAbsTermConn name = "C6" direction = "input">
            </vnsAbsTermConn>
          </vnsAbsTermNode>

        <vnsAbsConnection name = "CON1">

```

```

    <vnsRsAbsConnectionConns
      tDn="uni/tn-acme/AbsGraph-G1/AbsTermNode-Input1/AbsTConn"/>
    <vnsRsAbsConnectionConns
      tDn="uni/tn-acme/AbsGraph-G1/AbsNode-Node1/AbsFConn-C4"/>
  </vnsAbsConnection>

  <vnsAbsConnection name = "CON3">
    <vnsRsAbsConnectionConns
      tDn="uni/tn-acme/AbsGraph-G1/AbsNode-Node1/AbsFConn-C5"/>
    <vnsRsAbsConnectionConns
      tDn="uni/tn-acme/AbsGraph-G1/AbsTermNode-Output1/AbsTConn"/>
  </vnsAbsConnection>
</vnsAbsGraph>
</fvTenant>
</polUni>

```

次の REST 要求は非管理対象モードでサービス グラフを作成します。

```

<polUni>
  <fvTenant name="HA_Tenant1">
    <vnsAbsGraph name="g1">

      <vnsAbsTermNodeProv name="Input1">
        <vnsAbsTermConn name="C1">
          </vnsAbsTermConn>
        </vnsAbsTermNodeProv>

        <!-- Node1 Provides LoadBalancing functionality -->
        <vnsAbsNode name="Node1" managed="no">
          <vnsRsDefaultScopeToTerm
            tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsTermNodeProv-Input1/outtmn1"/>
          <vnsAbsFuncConn name="outside" attNotify="true">
            </vnsAbsFuncConn>
          <vnsAbsFuncConn name="inside" attNotify="true">
            </vnsAbsFuncConn>
          </vnsAbsNode>

          <vnsAbsTermNodeCon name="Output1">
            <vnsAbsTermConn name="C6">
              </vnsAbsTermConn>
            </vnsAbsTermNodeCon>

          <vnsAbsConnection name="CON2" adjType="L3" unicastRoute="yes">
            <vnsRsAbsConnectionConns
              tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsTermNodeCon-Output1/AbsTConn"/>
            <vnsRsAbsConnectionConns
              tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsNode-Node1/AbsFConn-outside"/>
          </vnsAbsConnection>

          <vnsAbsConnection name="CON1" adjType="L2" unicastRoute="no">
            <vnsRsAbsConnectionConns
              tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsNode-Node1/AbsFConn-inside"/>
            <vnsRsAbsConnectionConns
              tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsTermNodeProv-Input1/AbsTConn"/>
          </vnsAbsConnection>

        </vnsAbsGraph>
      </fvTenant>
    </polUni>

```

次の REST 要求はセキュリティ ポリシー (コントラクト) を作成します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">

```

```

<vzFilter name="HttpIn">
  <vzEntry name="e1" prot="6" dToPort="80"/>
</vzFilter>

<vzBrCP name="webCtrct">
  <vzSubj name="http">
    <vzRsSubjFiltAtt tnVzFilterName="HttpIn"/>
  </vzSubj>
</vzBrCP>
</fvTenant>
</polUni>

```

次の REST 要求はアプリケーション EPG からのグラフ コンフィギュレーション パラメータを提供します。

```

<polUni>
  <fvTenant dn="uni/tn-acme" name="acme">

    <!-- Application Profile -->
    <fvAp dn="uni/tn-acme/ap-MyAP" name="MyAP">

      <!-- EPG 1 -->
      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MyClnt" name="MyClnt">
        <fvRsBd tnFvBDName="MyClntBD"/>
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs"/>
        <fvRsProv tnVzBrCPName="webCtrct">
          </fvRsProv>
        <fvRsPathAtt tDn="topology/pod-1/paths-17/pathep-[eth1/20]"
encap="vlan-201"/>
        <fvSubnet name="SrcSubnet" ip="192.168.10.1/24"/>
      </fvAEPg>

      <!-- EPG 2 -->
      <fvAEPg dn="uni/tn-acme/ap-MyAP/epg-MySRVR" name="MySRVR">
        <fvRsBd tnFvBDName="MyClntBD"/>
        <fvRsDomAtt tDn="uni/vmmp-Vendor1/dom-MyVMs"/>
        <fvRsCons tnVzBrCPName="webCtrct">
          </fvRsCons>

        <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
          key="Monitor" name="monitor1">
          <vnsParamInst name="weight" key="weight" value="10"/>
        </vnsFolderInst>

        <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any"
          key="Service" name="Service1">
          <vnsParamInst name="servicename" key="servicename"
            value="crpvgrtst02-8010"/>
          <vnsParamInst name="servicetype" key="servicetype" value="TCP"/>
          <vnsParamInst name="servername" key="servername"
            value="s192.168.100.100"/>
          <vnsParamInst name="serveripaddress" key="serveripaddress"
            value="192.168.100.100"/>
          <vnsParamInst name="serviceport" key="serviceport" value="8080"/>
          <vnsParamInst name="svrtimeout" key="svrtimeout" value="9000"/>
          <vnsParamInst name="clttimeout" key="clttimeout" value="9000"/>
          <vnsParamInst name="usip" key="usip" value="NO"/>
          <vnsParamInst name="useproxyport" key="useproxyport" value=""/>
          <vnsParamInst name="cip" key="cip" value="ENABLED"/>
          <vnsParamInst name="cka" key="cka" value="NO"/>
          <vnsParamInst name="sp" key="sp" value="OFF"/>
          <vnsParamInst name="cmp" key="cmp" value="NO"/>
        </vnsFolderInst>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>

```

```

    <vnsParamInst name="maxclient" key="maxclient" value="0"/>
    <vnsParamInst name="maxreq" key="maxreq" value="0"/>
    <vnsParamInst name="tcpcb" key="tcpcb" value="NO"/>
    <vnsCfgRelInst name="MonitorConfig" key="MonitorConfig"
      targetName="monitor1"/>
  </vnsFolderInst>

  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
    nodeNameOrLbl="any" key="Network" name="Network">
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
      nodeNameOrLbl="any" key="vip" name="vip">
      <vnsParamInst name="vipaddress1" key="vipaddress"
        value="10.10.10.100"/>
    </vnsFolderInst>
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
      nodeNameOrLbl="any" devCtxLbl="C1" key="snip" name="snip1">
      <vnsParamInst name="snipaddress" key="snipaddress"
        value="192.168.1.100"/>
    </vnsFolderInst>
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
      nodeNameOrLbl="any" devCtxLbl="C2" key="snip" name="snip2">
      <vnsParamInst name="snipaddress" key="snipaddress"
        value="192.168.1.101"/>
    </vnsFolderInst>
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
      nodeNameOrLbl="any" devCtxLbl="C3" key="snip" name="snip3">
      <vnsParamInst name="snipaddress" key="snipaddress"
        value="192.168.1.102"/>
    </vnsFolderInst>
  </vnsFolderInst>

  <!-- SLB Configuration -->
  <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
    nodeNameOrLbl="any" key="VServer" name="VServer">
    <!-- Virtual Server Configuration -->
    <vnsParamInst name="port" key="port" value="8010"/>
    <vnsParamInst name="vip" key="vip" value="10.10.10.100"/>
    <vnsParamInst name="vservername" key="vservername"
      value="crpvgrtst02-vip-8010"/>
    <vnsParamInst name="servicename" key="servicename"
      value="crpvgrtst02-8010"/>
    <vnsParamInst name="servicetype" key="servicetype" value="TCP"/>
    <vnsFolderInst ctrctNameOrLbl="any" graphNameOrLbl="any"
      nodeNameOrLbl="any" key="VServerGlobalConfig"
      name="VServerGlobalConfig">
      <vnsCfgRelInst name="ServiceConfig" key="ServiceConfig"
        targetName="Service1"/>
      <vnsCfgRelInst name="VipConfig" key="VipConfig"
        targetName="Network/vip"/>
    </vnsFolderInst>
  </vnsFolderInst>
</fvAEPg>
</fvAp>
</fvTenant>
</polUni>

```

次の REST 要求はコントラクトにサービス グラフをアタッチします。

```

<polUni>
  <fvTenant name="acme">
    <vzBrCP name="webCtrct">
      <vzSubj name="http">
        <vzRsSubjGraphAtt graphName="G1" termNodeName="Input1"/>
      </vzSubj>
    </vzBrCP>
  </fvTenant>
</polUni>

```

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

Example: Configuring Layer 4 to Layer 7 Services (Firewall)

例：REST API を使用したレイヤ4～レイヤ7サービスの設定

このトピックでは、REST API を使用してレイヤ4～7サービス（ASA ファイアウォール）を設定するための手順を示します。

始める前に

- レイヤ3アウトサイドネットワークおよびブリッジドメインでレイヤ4～レイヤ7サービスを使用するためテナントを作成します。
- アプリケーションプロファイルを作成します。
- 物理ドメインまたはVMMドメインを設定します。
- デバイスパッケージをインポートおよび登録して、それらにパラメータを設定します。

手順

ステップ1 次の例のように、XMLを使用してレイヤ4～レイヤ7のASAvデバイスパッケージモデルを作成します。

例：

```
<vnsLDevVip trunking="no" svcType="FW"
packageModel="ASA" name="ASA" mode="legacy-Mode"
managed="yes" isCopy="no" funcType="GoTo"
dn="uni/tn-Tenant-test/lDevVip-ASA" devtype="VIRTUAL"
contextAware="single-Context">

<vnsCCred name="username" value="admin"/>
<vnsRsMDevAtt tDn="uni/infra/mDev-CISCO-ASA-1.2"/>
<vnsCCredSecret name="password"/>
<vnsCMgmt name="" port="443" host="172.31.184.249"/>
<vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-ACI_vDS"/>
<vnsCDev name="Device1" vmName="ASA-L3" vcenterName="vcenter" devCtxLbl="">
<vnsCCred name="username" value="admin"/>
<vnsCCredSecret name="password"/>
<vnsCMgmt name="" port="443" host="172.31.184.249"/>
<vnsCIf name="GigabitEthernet0/1" vnicName="Network adapter 3"/>
<vnsCIf name="GigabitEthernet0/0" vnicName="Network adapter 2"/>
<vnsRsCDevToCtrlrP tDn="uni/vmmp-VMware/dom-ACI_vDS/ctrlr-vcenter"/>
</vnsCDev>

<vnsLIf name="provider" encap="unknown">
<vnsRsMetaIf tDn="uni/infra/mDev-CISCO-ASA-1.2/mIfLbl-internal" isConAndProv="no"/>
<vnsRsCIfAttN tDn="uni/tn-Tenant-test/lDevVip-ASA/cDev-Device1/cIf-[GigabitEthernet0/1]"/>
</vnsLIf>
```

例：REST API を使用したレイヤ4～レイヤ7サービスの設定

```

<vnsLif name="consumer" encap="unknown">
<vnsRsMetaIf tDn="uni/infra/mDev-CISCO-ASA-1.2/mIfLbl-external" isConAndProv="no"/>
<vnsRsCifAttN tDn="uni/tn-Tenant-test/lDevVip-ASAv/cDev-Device1/cIf-[GigabitEthernet0/0]"/>
</vnsLIF>
</vnsLDevVip>

```

ステップ2 次の例のように、XML を使用してレイヤ4～レイヤ7の **FW-Graph** を設定します。

例：

```

<vnsAbsGraph uiTemplateType="UNSPECIFIED" ownerTag="" ownerKey="" name="FW-Graph"
dn="uni/tn-Tenant-test/AbsGraph-FW-Graph" descr="">

<vnsAbsTermNodeCon ownerTag="" ownerKey="" name="T1" descr="">
<vnsAbsTermConn ownerTag="" ownerKey="" name="1" descr="" attNotify="no"/>
<vnsInTerm name="" descr=""/>
<vnsOutTerm name="" descr=""/>
</vnsAbsTermNodeCon>

<vnsAbsTermNodeProv ownerTag="" ownerKey="" name="T2" descr="">
<vnsAbsTermConn ownerTag="" ownerKey="" name="1" descr="" attNotify="no"/>
<vnsInTerm name="" descr=""/>
<vnsOutTerm name="" descr=""/>
</vnsAbsTermNodeProv>

<vnsAbsConnection ownerTag="" ownerKey="" name="C1" descr="" unicastRoute="yes"
directConnect="no" connType="external" connDir="provider" adjType="L2">
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsNode-N1/AbsFConn-consumer"/>
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsTermNodeCon-T1/AbsTConn"/>
</vnsAbsConnection>

<vnsAbsConnection ownerTag="" ownerKey="" name="C2" descr="" unicastRoute="yes"
directConnect="no" connType="external" connDir="provider" adjType="L2">
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsNode-N1/AbsFConn-provider"/>
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsTermNodeProv-T2/AbsTConn"/>
</vnsAbsConnection>

<vnsAbsNode ownerTag="" ownerKey="" name="N1" descr="" shareEncap="no" sequenceNumber="0"
routingMode="unspecified" managed="yes" isCopy="no" funcType="GoTo"
funcTemplateType="FW_ROUTED">
<vnsAbsFuncConn ownerTag="" ownerKey="" name="consumer" descr="" attNotify="no">
<vnsRsMConnAtt tDn="uni/infra/mDev-CISCO-ASA-1.2/mFunc-Firewall/mConn-external"/>
</vnsAbsFuncConn>

<vnsAbsFuncConn ownerTag="" ownerKey="" name="provider" descr="" attNotify="no">
<vnsRsMConnAtt tDn="uni/infra/mDev-CISCO-ASA-1.2/mFunc-Firewall/mConn-internal"/>
</vnsAbsFuncConn>
<vnsRsNodeToAbsFuncProf
tDn="uni/infra/mDev-CISCO-ASA-1.2/absFuncProfContr/absFuncProfGrp-WebServiceProfileGroup/absFuncProf-WebPolicyForRoutedNode"/>
<vnsRsNodeToLDev tDn="uni/tn-Tenant-test/lDevVip-ASAv"/>
<vnsRsNodeToMFunc tDn="uni/infra/mDev-CISCO-ASA-1.2/mFunc-Firewall"/>
</vnsAbsNode>
</vnsAbsGraph>

```

ステップ3 次の例のように XML を使用して、デバイス選択ポリシーを作成します。

例：

```

<vnsLDevCtx nodeNameOrLbl="N1" name="" graphNameOrLbl="FW-Graph"
dn="uni/tn-Tenant-test/ldevCtx-c-Client-to-Web-g-FW-Graph-n-N1" descr=""

```



```

ctrctNameOrLbl="Client-to-Web">
<vnsRsLDevCtxToLDev tDn="uni/tn-Tenant-test/lDevVip-ASAv"/>

<vnsLIfCtx name="" descr="" connNameOrLbl="provider">
<vnsRsLIfCtxToBD tDn="uni/tn-Tenant-test/BD-BD2"/>
<vnsRsLIfCtxToLIf tDn="uni/tn-Tenant-test/lDevVip-ASAv/lIf-provider"/>
</vnsLIfCtx>

<vnsLIfCtx name="" descr="" connNameOrLbl="consumer">
<vnsRsLIfCtxToBD tDn="uni/tn-Tenant-test/BD-BD1"/>
<vnsRsLIfCtxToLIf tDn="uni/tn-Tenant-test/lDevVip-ASAv/lIf-consumer"/>
</vnsLIfCtx>
</vnsLDevCtx>

```

ステップ4 次の例のように、XML を使用して **FW グラフ** サービス グラフに関連付けられているコントラクトを設定します。

例：

```

<vzBrCP targetDscp="unspecified" scope="tenant" prio="unspecified" ownerTag=""
ownerKey="" name="Client-to-Web" dn="uni/tn-Tenant-test/brc-Client-to-Web" descr="">

<vzSubj targetDscp="unspecified" prio="unspecified" name="Subject" descr=""
revFltPorts="yes" provMatchT="AtleastOne" consMatchT="AtleastOne"
<vzRsSubjFiltAtt tnVzFilterName="default" directives=""/>
<vzRsSubjGraphAtt directives="" tnVnsAbsGraphName="FW-Graph"/>
</vzSubj>
</vzBrCP>

```

ステップ5 次の例のように、XML を使用して **クライアント EPG** を作成します。

例：

```

<fvAEPg prio="unspecified" pcEnfPref="unenforced" name="Client"
matchT="AtleastOne" isAttrBasedEPg="no" fwdCtrl=""
dn="uni/tn-Tenant-test/ap-ANP/epg-Client" descr="">
<fvRsCons prio="unspecified" tnVzBrCPName="Client-to-Web"/>
<fvRsDomAtt tDn="uni/vmmp-VMware/dom-ACI_vDS" resImedcy="lazy" primaryEncap="unknown"
instrImedcy="lazy" encap="unknown" delimiter="" classPref="encap"/>
<fvRsBd tnFvBDName="BD1"/>
<fvRsCustQosPol tnQosCustomPolName=""/>
</fvAEPg>

```

ステップ6 次の例のように、XML を使用して **Web EPG** を作成します。

例：

```

<fvAEPg prio="unspecified" pcEnfPref="unenforced" name="Web" matchT="AtleastOne"
isAttrBasedEPg="no" fwdCtrl="" dn="uni/tn-Tenant-test/ap-ANP/epg-Web" descr="">
<fvRsDomAtt tDn="uni/vmmp-VMware/dom-ACI_vDS" resImedcy="lazy" primaryEncap="unknown"
instrImedcy="lazy" encap="unknown" delimiter="" classPref="encap"/>
<fvRsBd tnFvBDName="BD2"/>
<fvRsCustQosPol tnQosCustomPolName=""/>

<vnsFolderInst name="internalIf" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="Interface"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">

<vnsFolderInst name="internalIfCfg" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="InterfaceConfig"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">
<vnsParamInst name="internal_security_level" locked="no" key="security_level"

```

例：REST API を使用したレイヤ4～レイヤ7サービスの設定

```

cardinality="unspecified"
value="100" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst name="externalIf" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="Interface"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">

<vnsFolderInst name="ExtAccessGroup" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="AccessGroup"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">
<vnsCfgRelInst name="name" locked="no" key="inbound_access_list_name"
cardinality="unspecified" mandatory="no"
targetName="access-list-inbound"/>
</vnsFolderInst>

<vnsFolderInst name="externalIfCfg" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="InterfaceConfig"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">

<vnsParamInst name="external_security_level" locked="no" key="security_level"
cardinality="unspecified" value="50" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst name="IntConfig" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="InIntfConfigRelFolder"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">
<vnsCfgRelInst name="InConfigrel" locked="no" key="InIntfConfigRel"
cardinality="unspecified" mandatory="no" targetName="internalIf"/>
</vnsFolderInst>

<vnsFolderInst name="ExtConfig" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="ExIntfConfigRelFolder"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">
<vnsCfgRelInst name="ExtConfigrel" locked="no" key="ExIntfConfigRel"
cardinality="unspecified" mandatory="no"
targetName="externalIf"/>
</vnsFolderInst>

<vnsFolderInst name="access-list-inbound" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="AccessList"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">
<vnsFolderInst name="permit-https" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="AccessControlEntry"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">
<vnsParamInst name="action-permit" locked="no" key="action" cardinality="unspecified"
value="permit"
validation="" mandatory="no"/>
<vnsParamInst name="order1" locked="no" key="order" cardinality="unspecified" value="10"
validation="" mandatory="no"/>

<vnsFolderInst name="dest-service" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
key="destination_service"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">

```

```
<vnsParamInst name="op" locked="no" key="operator" cardinality="unspecified" value="eq"
  validation="" mandatory="no"/>
<vnsParamInst name="port" locked="no" key="low_port" cardinality="unspecified"
  value="https" validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="dest-address" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
  key="destination_address"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
    validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="src-address" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
  key="source_address"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
    validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="tcp" scopedBy="epg" nodeNameOrLbl="N1" locked="no" key="protocol"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="tcp" locked="no" key="name_number" cardinality="unspecified"
    value="tcp" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst name="permit-http" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
  key="AccessControlEntry"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="action-permit" locked="no" key="action" cardinality="unspecified"
    value="permit"
    validation="" mandatory="no"/>

<vnsParamInst name="order1" locked="no" key="order" cardinality="unspecified" value="10"
  validation="" mandatory="no"/>

<vnsFolderInst name="dest-service" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
  key="destination_service"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="op" locked="no" key="operator" cardinality="unspecified" value="eq"
    validation="" mandatory="no"/>
  <vnsParamInst name="port" locked="no" key="low_port" cardinality="unspecified" value="http"
    validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="dest-address" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
  key="destination_address"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
    validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="src-address" scopedBy="epg" nodeNameOrLbl="N1" locked="no"
  key="source_address"
  graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
  cardinality="unspecified">
  <vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
    validation="" mandatory="no"/>
</vnsFolderInst>
```

例: REST API を使用したレイヤ4～レイヤ7サービスの設定

```

validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="tcp" scopedBy="epg" nodeNameOrLbl="N1" locked="no" key="protocol"
graphNameOrLbl="FW-Graph" devCtxLbl="" ctrctNameOrLbl="Client-to-Web"
cardinality="unspecified">

<vnsParamInst name="tcp" locked="no" key="name_number" cardinality="unspecified"
value="tcp" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>
</vnsFolderInst>

<fvRsProv prio="unspecified" matchT="AtleastOne" tnVzBrCPName="Client-to-Web"/>
</fvAEPg>

```

例

テナントのレイヤ4～レイヤ7のASA v ファイアウォールサービス全体を設定するには、次の例のようにXMLを使用します。

```

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

<vnsLDevCtx name="" descr="" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web"><vnsRsLDevCtxToLDev tDn="uni/tn-Tenant-test/lDevVip-ASAv"/>

<vnsLIfCtx name="" descr="" connNameOrLbl="provider">
<vnsRsLIfCtxToBD tDn="uni/tn-Tenant-test/BD-BD2"/>
<vnsRsLIfCtxToLIf tDn="uni/tn-Tenant-test/lDevVip-ASAv/lIf-provider"/>
</vnsLIfCtx>

<vnsLIfCtx name="" descr="" connNameOrLbl="consumer">
<vnsRsLIfCtxToBD tDn="uni/tn-Tenant-test/BD-BD1"/>
<vnsRsLIfCtxToLIf tDn="uni/tn-Tenant-test/lDevVip-ASAv/lIf-consumer"/>
</vnsLIfCtx>
</vnsLDevCtx>

<vzBrCP ownerTag="" ownerKey="" name="Client-to-Web" descr="" targetDscp="unspecified"
scope="tenant" prio="unspecified">

<vzSubj name="Subject" descr="" targetDscp="unspecified" prio="unspecified"
revFltPorts="yes"
provMatchT="AtleastOne" consMatchT="AtleastOne">
<vzRsSubjFiltAtt tnVzFilterName="default" directives=""/>
<vzRsSubjGraphAtt directives="" tnVnsAbsGraphName="FW-Graph"/>
</vzSubj>
</vzBrCP>

<vnsAbsGraph ownerTag="" ownerKey="" name="FW-Graph" descr="" uiTemplateType="UNSPECIFIED">
<vnsAbsTermNodeCon ownerTag="" ownerKey="" name="T1" descr="">
<vnsAbsTermConn ownerTag="" ownerKey="" name="1" descr="" attNotify="no"/>
<vnsInTerm name="" descr=""/>
<vnsOutTerm name="" descr=""/>
</vnsAbsTermNodeCon>

<vnsAbsTermNodeProv ownerTag="" ownerKey="" name="T2" descr="">
<vnsAbsTermConn ownerTag="" ownerKey="" name="1" descr="" attNotify="no"/>
<vnsInTerm name="" descr=""/>
<vnsOutTerm name="" descr=""/>
</vnsAbsTermNodeProv>

```

```

<vnsAbsConnection ownerTag="" ownerKey="" name="C1" descr="" unicastRoute="yes"
directConnect="no" connType="external" connDir="provider" adjType="L2">
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsNode-N1/AbsFConn-consumer"/>
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsTermNodeCon-T1/AbsTConn"/>
</vnsAbsConnection>

<vnsAbsConnection ownerTag="" ownerKey="" name="C2" descr="" unicastRoute="yes"
directConnect="no" connType="external" connDir="provider" adjType="L2">
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsNode-N1/AbsFConn-provider"/>
<vnsRsAbsConnectionConns
tDn="uni/tn-Tenant-test/AbsGraph-FW-Graph/AbsTermNodeProv-T2/AbsTConn"/>
</vnsAbsConnection>

<vnsAbsNode ownerTag="" ownerKey="" name="N1" descr="" shareEncap="no" sequenceNumber="0"
routingMode="unspecified" managed="yes" isCopy="no" funcType="GoTo"
funcTemplateType="FW_ROUTED">

<vnsAbsFuncConn ownerTag="" ownerKey="" name="consumer" descr="" attNotify="no">
<vnsRsMConnAtt tDn="uni/infra/mDev-CISCO-ASA-1.2/mFunc-Firewall/mConn-external"/>
</vnsAbsFuncConn>

<vnsAbsFuncConn ownerTag="" ownerKey="" name="provider" descr="" attNotify="no">
<vnsRsMConnAtt tDn="uni/infra/mDev-CISCO-ASA-1.2/mFunc-Firewall/mConn-internal"/>
</vnsAbsFuncConn>
<vnsRsNodeToAbsFuncProf
tDn="uni/infra/mDev-CISCO-ASA-1.2/absFuncProfContr/absFuncProfGrp-WebServiceProfileGroup/absFuncProf-WebPolicyForRoutedMode"/>
<vnsRsNodeToLDev tDn="uni/tn-Tenant-test/lDevVip-ASAv"/>
<vnsRsNodeToMFunc tDn="uni/infra/mDev-CISCO-ASA-1.2/mFunc-Firewall"/>
</vnsAbsNode>
</vnsAbsGraph>

<fvBD ownerTag="" ownerKey="" name="BD1" descr="" unicastRoute="yes" vmac="not-applicable"
unkMcastAct="flood" unkMacUcastAct="proxy" type="regular" multiDstPktAct="bd-flood"
mcastAllow="no"
mac="00:22:BD:F8:19:FF" llAddr="::" limitIpLearnToSubnets="no" ipLearning="yes"
epMoveDetectMode="" arpFlood="no">
<fvRsBDToNdP tnNdIfPolName=""/>
<fvRsCtx tnFvCtxName="VRF1"/>
<fvRsIgmprsn tnIgmprsnPolName=""/>
<fvRsBdToEpRet tnFvEpRetPolName="" resolveAct="resolve"/>
</fvBD>

<fvBD ownerTag="" ownerKey="" name="BD2" descr="" unicastRoute="yes" vmac="not-applicable"
unkMcastAct="flood" unkMacUcastAct="proxy" type="regular" multiDstPktAct="bd-flood"
mcastAllow="no"
mac="00:22:BD:F8:19:FF" llAddr="::" limitIpLearnToSubnets="no" ipLearning="yes"
epMoveDetectMode="" arpFlood="no">
<fvRsBDToNdP tnNdIfPolName=""/>
<fvRsCtx tnFvCtxName="VRF1"/>
<fvRsIgmprsn tnIgmprsnPolName=""/>
<fvRsBdToEpRet tnFvEpRetPolName="" resolveAct="resolve"/>
</fvBD>

<fvCtx ownerTag="" ownerKey="" name="VRF1" descr="" pcEnfPref="enforced" pcEnfDir="ingress"
knwMcastAct="permit">
<fvRsBgpCtxPol tnBgpCtxPolName=""/>

```

例：REST API を使用したレイヤ4～レイヤ7サービスの設定

```

<fvRsCtxToExtRouteTagPol tnL3extRouteTagPolName=""/>
<fvRsOspfCtxPol tnOspfCtxPolName=""/>
<vzAny name="" descr="" matchT="AtleastOne"/>
<fvRsCtxToEpRet tnFvEpRetPolName=""/>
</fvCtx>
<vnsSvcCont/>

<fvAp ownerTag="" ownerKey="" name="ANP" descr="" prio="unspecified">

<fvAEPg name="Web" descr="" prio="unspecified" pcEnfPref="unenforced"
matchT="AtleastOne" isAttrBasedEPg="no" fwdCtrl="">
<fvRsDomAtt tDn="uni/vmmp-VMware/dom-ACI_vDS" resImedcy="lazy" primaryEncap="unknown"
instrImedcy="lazy" encap="unknown" delimiter="" classPref="encap"/>
<fvRsBd tnFvBDName="BD2"/>
<fvRsCustQosPol tnQosCustomPolName=""/>

<vnsFolderInst name="internalIf" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="Interface"
devCtxLbl="" cardinality="unspecified">

<vnsFolderInst name="internalIfCfg" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="InterfaceConfig"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="internal_security_level" locked="no" key="security_level"
cardinality="unspecified"
value="100" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst name="externalIf" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="Interface"
devCtxLbl="" cardinality="unspecified">

<vnsFolderInst name="ExtAccessGroup" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="AccessGroup" devCtxLbl=""
cardinality="unspecified">
<vnsCfgRelInst name="name" locked="no" key="inbound_access_list_name"
cardinality="unspecified"
mandatory="no" targetName="access-list-inbound"/>
</vnsFolderInst>

<vnsFolderInst name="externalIfCfg" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="InterfaceConfig"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="external_security_level" locked="no" key="security_level"
cardinality="unspecified" value="50" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst name="IntConfig" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="InIntfConfigRelFolder"
devCtxLbl="" cardinality="unspecified">
<vnsCfgRelInst name="InConfigrel" locked="no" key="InIntfConfigRel"
cardinality="unspecified"
mandatory="no" targetName="internalIf"/>
</vnsFolderInst>

<vnsFolderInst name="ExtConfig" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="ExIntfConfigRelFolder"
devCtxLbl="" cardinality="unspecified">
<vnsCfgRelInst name="ExtConfigrel" locked="no" key="ExIntfConfigRel"
cardinality="unspecified"
mandatory="no" targetName="externalIf"/>

```

```
</vnsFolderInst>

<vnsFolderInst name="access-list-inbound" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="AccessList" devCtxLbl=""
cardinality="unspecified">

<vnsFolderInst name="permit-https" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="AccessControlEntry"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="action-permit" locked="no" key="action" cardinality="unspecified"
value="permit" validation="" mandatory="no"/>
<vnsParamInst name="order1" locked="no" key="order" cardinality="unspecified" value="10"
validation="" mandatory="no"/>

<vnsFolderInst name="dest-service" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="destination_service"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="op" locked="no" key="operator" cardinality="unspecified"
value="eq" validation="" mandatory="no"/>
<vnsParamInst name="port" locked="no" key="low_port" cardinality="unspecified"
value="https" validation="" mandatory="no"/>
</vnsFolderInst>
<vnsFolderInst name="dest-address" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="destination_address"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="src-address" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="source_address"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="tcp" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="protocol" devCtxLbl=""
cardinality="unspecified">
<vnsParamInst name="tcp" locked="no" key="name_number" cardinality="unspecified"
value="tcp" validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst name="permit-http" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="AccessControlEntry"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="action-permit" locked="no" key="action" cardinality="unspecified"
value="permit" validation="" mandatory="no"/>
<vnsParamInst name="order1" locked="no" key="order" cardinality="unspecified" value="10"
validation="" mandatory="no"/>

<vnsFolderInst name="dest-service" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="destination_service"
devCtxLbl="" cardinality="unspecified">
<vnsParamInst name="op" locked="no" key="operator" cardinality="unspecified" value="eq"
validation="" mandatory="no"/>
<vnsParamInst name="port" locked="no" key="low_port" cardinality="unspecified" value="http"
validation="" mandatory="no"/>
```

例: REST API を使用したレイヤ4～レイヤ7サービスの設定

```

</vnsFolderInst>

<vnsFolderInst name="dest-address" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="destination_address"
devCtxLbl=""
cardinality="unspecified">
<vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="src-address" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="source_address" devCtxLbl=""

cardinality="unspecified">
<vnsParamInst name="any" locked="no" key="any" cardinality="unspecified" value="any"
validation="" mandatory="no"/>
</vnsFolderInst>

<vnsFolderInst name="tcp" nodeNameOrLbl="N1" graphNameOrLbl="FW-Graph"
ctrctNameOrLbl="Client-to-Web" scopedBy="epg" locked="no" key="protocol" devCtxLbl=""
cardinality="unspecified">

<vnsParamInst name="tcp" locked="no" key="name_number" cardinality="unspecified"
value="tcp"
validation="" mandatory="no"/>
</vnsFolderInst>
</vnsFolderInst>
</vnsFolderInst>
<fvRsProv prio="unspecified" matchT="AtleastOne" tnVzBrCPName="Client-to-Web"/>
</fvAEPg>

<fvAEPg name="Client" descr="" prio="unspecified" pcEnfPref="unenforced"
matchT="AtleastOne"
isAttrBasedEPg="no" fwdCtrl="">
<fvRsCons prio="unspecified" tnVzBrCPName="Client-to-Web"/>
<fvRsDomAtt tDn="uni/vmmp-VMware/dom-ACI_vDS" resImedcy="lazy" primaryEncap="unknown"
instrImedcy="lazy" encap="unknown" delimiter="" classPref="encap"/>
<fvRsBd tnFvBDName="BD1"/>
<fvRsCustQosPol tnQosCustomPolName=""/>
</fvAEPg>
</fvAp>
<fvRsTenantMonPol tnMonEPGPolName=""/>

<vnsLDevVip name="ASAv" managed="yes" isCopy="no" funcType="GoTo" trunking="no"
svcType="FW" packageModel="ASAv" mode="legacy-Mode" devtype="VIRTUAL"
contextAware="single-Context">
<vnsCCred name="username" value="admin"/>
<vnsRsMDevAtt tDn="uni/infra/mDev-CISCO-ASA-1.2"/>
<vnsCCredSecret name="password"/>
<vnsCMgmt name="" port="443" host="172.31.184.249"/>
<vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-ACI_vDS"/>

<vnsCDev name="Device1" devCtxLbl="" vmName="ASAv-L3" vcenterName="vcenter">
<vnsCCred name="username" value="admin"/>
<vnsCCredSecret name="password"/>
<vnsCMgmt name="" port="443" host="172.31.184.249"/>
<vnsCIf name="GigabitEthernet0/1" vnicName="Network adapter 3"/>
<vnsCIf name="GigabitEthernet0/0" vnicName="Network adapter 2"/>
<vnsRsCDevToCtrlrP tDn="uni/vmmp-VMware/dom-ACI_vDS/ctrlr-vcenter"/>
</vnsCDev>

<vnsLIf name="provider" encap="unknown">
<vnsRsMetaIf tDn="uni/infra/mDev-CISCO-ASA-1.2/mIfLbl-internal" isConAndProv="no"/>
<vnsRsCIfAttN tDn="uni/tn-Tenant-test/lDevVip-ASAv/cDev-Device1/cIf-[GigabitEthernet0/1]"/>

```



```

</vnsLif>

<vnsLif name="consumer" encap="unknown">
<vnsRsMetaIf tDn="uni/infra/mDev-CISCO-ASA-1.2/mIfLbl-external" isConAndProv="no"/>
<vnsRsCifAttN tDn="uni/tn-Tenant-test/lDevVip-ASAv/cDev-Device1/cIf-[GigabitEthernet0/0]"/>
</vnsLif>
</vnsLDevVip>
</fvTenant>

```

例: レイヤ7ルートピアリングするレイヤ4の設定

レイヤ4からレイヤ7のルートピアリングを REST API を設定する

これらの l3extOut ポリシーは、ファブリックリーフで OSPF を有効にするために必要な OSPF 設定を指定します。外部通信に使用される l3extOut ポリシーと非常によく似ています。

l3extOut ポリシーは、ファブリック内外にどのルートを提供するかを制御するプレフィックスベースの EPG も指定します。Scope=import 属性は、どのエンドポイントプレフィックスを学習するかを制御し、外部 L4-L7 デバイスにこのルートを実装するよう指示します。Scope=export 属性は、ファブリックによってこのルートを L4-L7 デバイスにアドバタイズする必要があると指定します。

l3extOut ポリシーの2つの例を次に示します。**OspfInternal** を **eth1/23** で導入する例と、**OspfExternal** を **eth1/25** で導入する例です。

始める前に

1つ以上の l3extOut 外部ネットワーク接続を作成し、それらをサービスデバイスが接続されているファブリックリーフノードに展開します。

手順

ステップ 1 eth1/23 で OspfInternal を設定するには、次のような例と同様に XML とともにポストを送信します。

例:

```

<?xml version="1.0" encoding="UTF-8?>
<!-- /api/policymgr/mo.xml -->
<polUni>
  <fvTenant name="coke{{tenantId}}">
    {% if status is not defined %}
      {% set status = "created,modified" %}
    {% endif %}

    <l3extOut name="OspfInternal" status="{{status}}">

      <l3extLNodeP name="bLeaf-101">
        <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="180.0.0.11"/>
        <l3extLifP name='portIf'>
          <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/23]"

```

レイヤ4からレイヤ7のルートピアリングを REST API を設定する

```

        ifInstT='ext-svi' encap='vlan-3844' addr="30.30.30.100/28"
mtu='1500'/'>
        <!-- <ospfIfP authKey="tecom" authType="md5" authKeyId='1'> -->
        <ospfIfP>
            <ospfRsIfPol tnOspfIfPolName='ospfIfPol'/'>
            </ospfIfP>
        </l3extLIIfP>
    </l3extLNodeP>

    <ospfExtP areaId='111' areaType='nssa' areaCtrl='redistribute'/'>

    <l3extInstP name="OspfInternalInstP">
        <l3extSubnet ip="30.30.30.100/28" scope="import"/'>
        <l3extSubnet ip="20.20.20.0/24" scope="import"/'>
        <l3extSubnet ip="10.10.10.0/24" scope="export"/'>
    </l3extInstP>

    <l3extRsEctx tnFvCtxName="cokectx1"/'>

    </l3extOut>

    <ospfIfPol name="ospfIfPol" nwT='bcast' xmitDelay='1'
        helloIntvl='10' deadIntvl='40' status="created,modified"/'>
    </fvTenant>
</polUni>

```

ステップ2 eth1/25 で OspfExternal を設定するには、次のような例と同様に XML とともにポストを送信します。

例：

```

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

            {% if status is not defined %}
                {% set status="created,modified" %}
            {% endif %}

            <l3extOut name=OspfExternal" status="{{status}}">
                <l3extLNodeP name="bLeaf-101">
                    <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="180.0.0.8/28"/'>

                    <l3extLIIfP name='portIf'>
                        {% if intfType is not defined %}
                            {% set intfType="ext-svi" %}
                        {% endif %}
                    <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/25]"
                        ifInstT='ext-svi' encap='vlan-3843' addr="40.40.40.100/28" mtu='1500'/'>

                    <!-- ospfIfP authKey="tecom" authType="md5" authKeyId='1'> -->
                    <ospfIfP>
                        <ospfRsIfPol tnOspfIfPolName='ospfIfPol'/'>
                        </ospfIfP>
                    </l3extLIIfP>
                </l3extLNodeP>

                <ospfExtP areaId='111' areaType='nssa' areaCtrl='redistribute'/'>

```

```

<l3extInstP name="OspfExternalInstP">
  <l3extSubnet ip="40.40.40.100/28" scope="import"/>
  <l3extSubnet ip="10.10.10.0/24" scope="import"/>
  <l3extSubnet ip="20.20.20.0/24" scope="export"/>
</l3extInstP>

  <l3extRsEctx tnfvCtxName="commonctx"/>

</l3extOut>

  <ospfIfPol name="ospfIfPol" nwT='bcast' xmitDelay='1' helloIntvl='10' deadIntvl='40'
status="created,modified"/>
  </fvTenant>
</polUni>

```

l3extInstP オブジェクトはプレフィックス 40.40.40.100/28 を指定し、10.10.10.0/24 はエンドポイントの関連付けに基づいてプレフィックスに使用され、L4-L7 デバイスがこれらのルートを実行するアドバタイズすることを示します。

l3extRsPathL3OutAtt オブジェクトは、各 L3extOut を展開する場所を指定します。

(注) ルートピアリングを機能させるには、l3extRsPathL3OutAtt と、L4-L7 論理デバイス クラスタが接続されている rsCIfPathAtt が一致する必要があります。

L7 ルートピアリングするレイヤ4のl3extOutポリシーを指定します。

選択ポリシー vnsLIfCtx を使用する論理デバイス クラスタに、特定の l3extOut ポリシーを使用することができます。The vnsRsLIfCtxToInstP points the LIfCtx to the appropriate OspfInternal and OspfExternal l3extInstP EPGs. レイヤ7ルートピアリングするレイヤ4の使用 L3extOut ポリシーを設定するには、次の例などの XML で post を送信します。

```

<vnsLDevCtx ctrctNameOrLbl="webCtrct{{graphId}}" graphNameOrLbl="WebGraph"
nodeNameOrLbl="FW">
  <vnsRsLDevCtxToLDev tDn="uni/tn-solar{{tenantId}}/lDevVip-Firewall"/>
  <vnsLIfCtx connNameOrLbl="internal">
    {% if L3ExtOutInternal is not defined %}
    <fvSubnet ip="10.10.10.10/24"/>
    {% endif %}
    <vnsRsLIfCtxToBD tDn="uni/tn-solar{{tenantId}}/BD-solarBD1"/>
    <vnsRsLIfCtxToLIf tDn="uni/tn-solar{{tenantId}}/lDevVip-Firewall/lIf-internal"/>

    {% if L3ExtOutInternal is defined %}
    <vnsRsLIfCtxToInstP
tDn="uni/tn-solar{{tenantId}}/out-OspfInternal/instP-OspfInternalInstP"
status={{L3ExtOutInternal}}"/>
    {% endif %}
  </vnsLIfCtx>
  <vnsLIfCtx connNameOrLbl="external">
    {% if L3ExtOutExternal is not defined %}
    <fvSubnet ip="40.40.40.40/24"/>
    {% endif %}
    <vnsRsLIfCtxToBD tDn="uni/tn-solar{{tenantId}}/BD-solarBD4"/>
    <vnsRsLIfCtxToLIf tDn="uni/tn-solar{{tenantId}}/lDevVip-Firewall/lIf-external"/>

    {% if L3ExtOutExternal is defined %}
    <vnsRsLIfCtxToInstP

```

L7 ルートピアリングするレイヤ4の **l3extOut** ポリシーを指定します。

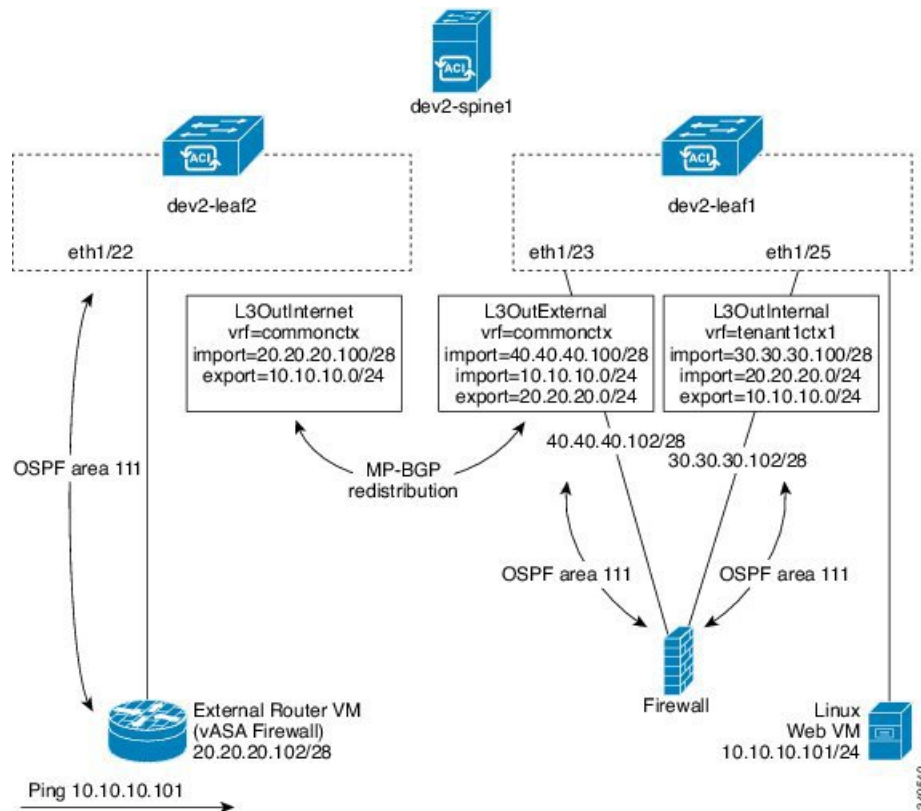
```
tDn="uni/tn-solar{{tenantId}}/out-OspfExternal/instP-OspfExternalInstP"
status={{L3ExtOutExternal}}"/>
  {% endif %}
</vnsLIfCtx>
</vnsLDevCtx>
```

次の例に示すように、関連付けられた具象デバイスには、それを同じファブリックリーフに導入する `vnsRsCifPathAtt` を設定する必要があります。

```
<vnsCDev name="ASA">
  <vnsRsLDevCtxToLDev tDn="uni/tn-solar{{tenantId}}/lDevVip-Firewall"/>
  <vnsCIf name="Gig0/0">
    <vnsRsCifPathAtt tDn="topology/pod-1/paths-101/pathep-[eht1/23]"/>
  </vnsCIf>
  <vnsCIf name="Gig0/1">
    <vnsRsCifPathAtt tDn="topology/pod-1/paths-101/pathep-[eht1/25]"/>
  </vnsCIf>
  <vnsCMgmt name="devMgmt" host="{{asaIp}}" port="443" />
  <vnsCCred name="username" value="admin" />
  <vnsCCredSecret name="password" value="insieme" />
</vnsCDev>
```

次の図に、ルートピアリングがエンドツーエンドでどのように動作するかを示します。

図 2: 導入例



2リーフ/1スパインのこのトポロジでは、Linux Web サーバが IP 10.10.10.101/24 にあり、`dev2-leaf1` に接続された ESX サーバでホストされます。サービスグラフは、同様に `dev2-leaf1` に接続されたツーアームファイアウォールから構成された状態で導入されます。The service graph is associated with a contract that binds an external `l3extOut L3OutInternet` with the provider

EPG (Web VM).L3OutExternal と L3OutInternal の2つの内部 l3extOut ポリシーも、サービスデバイスが接続されているリーフポートに導入されます。

■ L7 ルートピアリングするレイヤ4の `!3ext0ut` ポリシーを指定します。