REST API の使用

この章の内容は、次のとおりです。

APIC の準備の例について

このマニュアルのいくつかの例の手順には、パラメータ名が含まれています。これらのパラメータ名は、便宜上理解しやすいように例として提供されるもので、それらを使用する必要はありません。

APIC によるスイッチ検出

APICは、ACIファブリックの一部であるすべてのスイッチに対する自動プロビジョニングおよび管理の中心となるポイントです。単一のデータセンターには、複数の ACI ファブリックを組み込むことができます。各データセンターは、自身のAPICクラスタとファブリックの一部であるCisco Nexus 9000 シリーズ スイッチを持つことができます。スイッチが単一の APIC クラスタによって

のみ管理されるようにするには、各スイッチがファブリックを管理するその特定の APIC クラスタに登録される必要があります。

APICは、現在管理している任意のスイッチに直接接続されている新規スイッチを検出します。クラスタ内の各 APIC インスタンスは、直接接続されているリーフ スイッチのみを最初に検出します。リーフ スイッチが APIC で登録されると、APIC はリーフ スイッチに直接接続されているすべてのスパイン スイッチを検出します。各スパイン スイッチが登録されると、その APIC はそのスパインスイッチに接続されているすべてのリーフスイッチを検出します。このカスケード化された検出により、APIC は簡単なわずかな手順でファブリック トポロジ全体を検出することができます。

APIC クラスタによるスイッチ登録

（注） スイッチを登録する前に、ファブリック内のすべてのスイッチが物理的に接続され、適切な設定で起動されていることを確認します。シャーシの設置については、http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/products-installation-guides-list.htmlを参照してください。

スイッチが APIC で登録されると、そのスイッチは APIC で管理されるファブリック インベントリの一部となります。アプリケーション セントリック インフラストラクチャ ファブリック（ACI ファブリック）を使用すると、APICはインフラストラクチャ内のスイッチのプロビジョニング、管理、およびモニタリングのシングル ポイントとなります。

（注） インフラストラクチャのIPアドレス範囲は、インバンドおよびアウトオブバンドのネットワーク用の ACI ファブリックで使用する他の IP アドレスと重複してはなりません。

REST API を使用した未登録スイッチの登録

（注） インフラストラクチャのIPアドレス範囲は、インバンドおよびアウトオブバンドのネットワーク用の ACI ファブリックで使用する他の IP アドレスと重複してはなりません。

手順

スイッチを登録します。

例：

```
POST: https://<apic-ip>/api/node/mo/uni/controller.xml
<fabricNodeIdentPol>
<fabricNodeIdentP  serial="FGE173900ZD" name="leaf1" nodeId="101"/>
<fabricNodeIdentP  serial="FGE1740010A" name="leaf2" nodeId="102"/>
<fabricNodeIdentP  serial="FGE1740010H" name="spine1" nodeId="203"/>
```

```
<fabricNodeIdentP  serial="FGE1740011B" name="spine2" nodeId="204"/>
</fabricNodeIdentPol>
```

APIC からのスイッチ検出の検証とスイッチ管理

スイッチが APIC で登録された後、APIC はファブリック トポロジ ディスカバリを自動的に実行し、ネットワーク全体のビューを取得し、ファブリック トポロジ内のすべてのスイッチを管理します。

各スイッチは、個々にアクセスせずに、APIC から設定、モニタ、およびアップグレードできます。

REST API を使用した登録済みスイッチの検証

手順

REST API を使用してスイッチの登録を検証します。

例：

```
GET: https://<apic-ip>/api/node/class/topSystem.xml?

<?xml version="1.0" encoding="UTF-8"?>
<imdata>
    <topSystem address="10.0.0.1" dn="topology/pod-1/node-1/sys" fabricId="1" id="1"
name="apic1"
    oobMgmtAddr="10.30.13.44" podId="1" role="apic" serial="" state="in-service"
systemUpTime="00:00:00:02.199" .../>
    <topSystem address="10.0.0.2" dn="topology/pod-1/node-2/sys" fabricId="1" id="2"
name="apic2"
    oobMgmtAddr="10.30.13.45" podId="1" role="apic" serial="" state="in-service"
systemUpTime="00:00:00:02.199" .../>
    <topSystem address="10.0.0.3" dn="topology/pod-1/node-3/sys" fabricId="1" id="3"
name="apic3"
    oobMgmtAddr="10.30.13.46" podId="1" role="apic" serial="" state="in-service"
systemUpTime="00:00:00:02.199" .../>
   <topSystem address="10.0.98.127" dn="topology/pod-1/node-101/sys" fabricId="1" id="101"

    name="leaf1" oobMgmtAddr="0.0.0.0" podId="1" role="leaf" serial="FOX-270308"
state="in-service"
    systemUpTime="00:00:00:02.199" .../>
   <topSystem address="10.0.98.124" dn="topology/pod-1/node-102/sys" fabricId="1" id="102"

    name="leaf2" oobMgmtAddr="0.0.0.0" podId="1" role="leaf" serial="FOX-270308"
state="in-service"
    systemUpTime="00:00:00:02.199" .../>
   <topSystem address="10.0.98.125" dn="topology/pod-1/node-203/sys" fabricId="1" id="203"

    name="spine2" oobMgmtAddr="0.0.0.0" podId="1" role="spine" serial="FOX-616689"
state="in-service"
    systemUpTime="00:00:00:02.199" .../>
   <topSystem address="10.0.98.126" dn="topology/pod-1/node-204/sys" fabricId="1" id="204"

    name="spine1" oobMgmtAddr="0.0.0.0" podId="1" role="spine" serial="FOX-616689"
state="in-service"
    systemUpTime="00:00:00:02.199" .../>
</imdata>
```

ファブリック トポロジの検証

すべてのスイッチが APIC クラスタに登録された後、APIC はファブリック内のすべてのリンクおよび接続を自動的に検出し、その結果トポロジ全体を検出します。

REST API を使用したファブリック トポロジの検証

手順

REST API を使用して、ファブリック トポロジを検証します。

例：
ユーザ リファレンスの識別子は次のとおりです。

- n1 = 最初のノードの 識別子
- s1 = 最初のノードのスロット
- p1 = スロット s1 のポート
- n2 = 2 番目のノードの識別子
- s2 = 2 番目のノードのスロット
- p2 = スロット s2 のポート

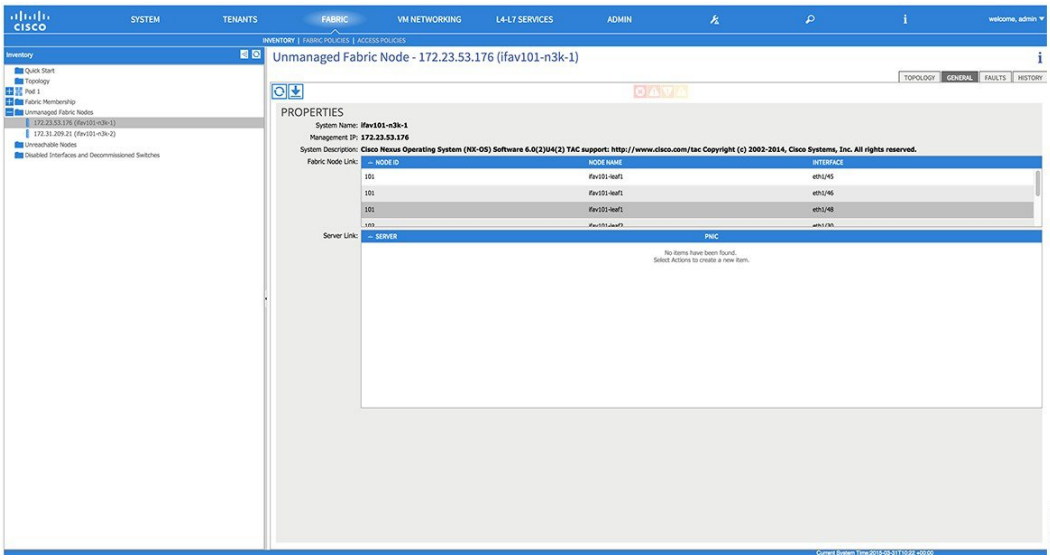
```
GET: https://<apic-ip>/api/node/class/fabricLink.xml?

<?xml version="1.0" encoding="UTF-8"?>
<imdata>
    <fabricLink dn="topology/lnkcnt-19/lnk-18-1-50-to-19-5-2" n1="18" n2="19" p1="50" p2="2"
 s1="1" s2="5" status="" .../>
    <fabricLink dn="topology/lnkcnt-20/lnk-18-1-49-to-20-5-1" n1="18" n2="20" p1="49" p2="1"
 s1="1" s2="5" status="" .../>
     <fabricLink dn="topology/lnkcnt-3/lnk-18-1-1-to-3-1-1" n1="18" n2="3" p1="1" p2="1"
s1="1" s2="1" status="" .../>
    <fabricLink dn="topology/lnkcnt-19/lnk-17-1-49-to-19-5-1" n1="17" n2="19" p1="49" p2="1"
 s1="1" s2="5" status="" .../>
    <fabricLink dn="topology/lnkcnt-20/lnk-17-1-50-to-20-5-2" n1="17" n2="20" p1="50" p2="2"
 s1="1" s2="5" status="" .../>
     <fabricLink dn="topology/lnkcnt-1/lnk-17-1-1-to-1-1-1" n1="17" n2="1" p1="1" p2="1"
s1="1" s2="1" status="" .../>
     <fabricLink dn="topology/lnkcnt-2/lnk-17-1-2-to-2-1-1" n1="17" n2="2" p1="2" p2="1"
s1="1" s2="1" status="" .../>
</imdata>
```

VM 管理でのアンマネージド スイッチの接続

VM コントローラ（たとえば、vCenter）によって管理されているホストは、レイヤ 2 スイッチを介してリーフ ポートに接続できます。必要な唯一の前提条件は、レイヤ 2 スイッチを管理アドレスで設定することです。この管理アドレスは、スイッチに接続されているポート上で Link Layer Discovery Protocol（LLDP）または Cisco Discovery Protocol（CDP）によってアドバタイズされる必要があります。レイヤ2スイッチは、APICによって自動的に検出され、管理アドレスで識別さ

れます。次の図は、[Fabric] > [Inventory] ビューにアンマネージド スイッチを表示する APIC GUI を示します。

図 *1*：*APIC* ファブリック インベントリのアンマネージド レイヤ *2* スイッチ



ネットワーク タイム プロトコルの設定

時刻同期と NTP

シスコ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）ファブリックにおいて、時刻の同期は、モニタリング、運用、トラブルシューティングなどの多数のタスクが依存している重要な機能です。クロック同期は、トラフィック フローの適切な分析にとって重要であり、複数のファブリック ノード間でデバッグとフォールトのタイムスタンプを関連付けるためにも重要です。

1 つ以上のデバイスでオフセットが生じると、多くの一般的な運用問題を適切に診断して解決する機能がブロックされる可能性があります。また、クロック同期によって、アプリケーションのヘルススコアが依存している ACI の内蔵アトミック カウンタ機能をフル活用できます。時刻同期が存在しない場合や不適切に設定されている場合でも、エラーやヘルススコアの低下が引き起こされるわけではありません。これらの機能を適切に使用できるように、ファブリックやアプリケーションを完全に展開する前に、時刻同期を設定する必要があります。デバイスのクロックを同期させる最も一般的な方法は、ネットワーク タイム プロトコル（NTP）を使用することです。

NTP を 設定する前に、どの管理 IP アドレス スキームを ACI ファブリックに配置するかを検討してください。すべての ACI ノードと Application Policy Infrastructure Controller（APIC）の管理を設定するために、インバンド管理とアウトオブバンド管理の 2 つのオプションがあります。ファブリックに対して選択した管理オプションに応じて、NTP の設定が異なります。時刻同期の展開に

関するもう 1 つの考慮事項は、時刻源の場所です。プライベート内部時刻または外部パブリック時刻の使用を決定する際は、時刻源の信頼性について慎重に検討する必要があります。

インバンドおよびアウトオブバンドの管理 NTP

（注）
- 管理 EPG が NTP サーバ用に設定されていることを確認してください。設定されていない場合、このサーバはスイッチで設定されません。
- インバンド管理アクセスおよびアウトオブバンド管理アクセスについては、本書の「管理アクセスの追加」という項を参照してください。

- アウトオブバンド管理 NTP：ACI ファブリックをアウトオブバンド管理とともに展開する場合、ファブリックの各ノードは、スパイン、リーフ、および APIC クラスタの全メンバーを含めて、ACI ファブリックの外部から管理されます。この IP 到達可能性を活用することで、各ノードは一貫した時刻源として同じ NTP サーバに個々に照会することができます。NTP を設定するには、アウトオブバンド管理のエンドポイントグループを参照する日付時刻ポリシーを作成する必要があります。日付時刻ポリシーは 1 つのポッドに限定され、ACI ファブリック内のプロビジョニングされたすべてのポッドに展開する必要があります。現在は、ACI ファブリックあたり 1 つのポッドのみが許可されます。
- インバンド管理 NTP：ACI ファブリックをインバンド管理とともに展開する場合は、ACI のインバンド管理ネットワーク内から NTP サーバへの到達可能性を検討します。ACI ファブリック内で使用されるインバンド IP アドレッシングには、ファブリックの外部から到達できません。インバンド管理されているファブリックの外部の NTP サーバを使用するには、その通信を可能にするポリシーを作成します。インバンド管理ポリシーの設定に使用される手順は、アウトオブバンド管理ポリシーの確立に使用される手順と同じです。 違いは、ファブリックが NTP サーバに接続できるようにする方法です。

NTP over IPv6

NTP over IPv6 アドレスは、ホスト名とピア アドレスでサポートされます。gai.conf も、IPv4 アドレスのプロバイダーまたはピアの IPv6 アドレスが優先されるように設定できます。ユーザは、IP アドレス（インストールまたは優先順位よって IPv4、IPv6、または両方）を提供することによって解決できるホスト名を設定できます。

REST API を使用した NTP の設定

手順

ステップ 1 NTP を設定します。

例：

```
POST url: https://APIC-IP/api/node/mo/uni/fabric/time-test.xml

<imdata totalCount="1">
    <datetimePol adminSt="enabled" authSt="disabled" descr="" dn="uni/fabric/time-CiscoNTPPol"
 name="CiscoNTPPol" ownerKey="" ownerTag="">
        <datetimeNtpProv descr="" keyId="0" maxPoll="6" minPoll="4" name="10.10.10.11"
preferred="yes">
            <datetimeRsNtpProvToEpg tDn="uni/tn-mgmt/mgmtp-default/inb-default"/>
        </datetimeNtpProv>
    </datetimePol>
</imdata>
```

ステップ 2 デフォルトの日付と時刻のポリシーをポッド ポリシー グループに追加します。

例：

```
POST url: https://APIC-IP/api/node/mo/uni/fabric/funcprof/podpgrp-calo1/rsTimePol.xml

POST payload: <imdata totalCount="1">
<fabricRsTimePol tnDatetimePolName="CiscoNTPPol">
</fabricRsTimePol>
</imdata>
```

ステップ 3 ポッド ポリシー グループをデフォルトのポッド プロファイルに追加します。

例：

```
POST url:
https://APIC-IP/api/node/mo/uni/fabric/podprof-default/pods-default-typ-ALL/rspodPGrp.xml

payload: <imdata totalCount="1">
<fabricRsPodPGrp tDn="uni/fabric/funcprof/podpgrp-calo1" status="created">
</fabricRsPodPGrp>
</imdata>
```

GUI を使用した NTP の動作の確認

手順

ステップ 1 メニュー バーで、[FABRIC] > [Fabric Policies] を選択します。

ステップ 2 [Navigation] ペインで、[Pod Policies] > [Policies] > [Date and Time] > *[ntp_policy]* > *[server_name]* の順に選択します。
ntp_policy は前に作成したポリシーです。[Host Name] フィールドまたは [IP address] フィールドではIPv6アドレスがサポートされます。入力したホスト名にIPv6アドレスが設定されている場合、IPv6 アドレスが IPv4 アドレスより優先されるように実装する必要があります。

ステップ 3 [Work] ペインで、サーバの詳細を確認します。

CLI を使用した、各ノードに導入された NTP ポリシーの確認

手順

ステップ 1 ファブリックの APIC に SSH 接続します。

ステップ 2 attach コマンドを入力して Tab キーを 2 回押し、使用可能なノードの名前をすべて表示します。

例：
```
admin@apic1:~> attach <Tab> <Tab>
```

ステップ 3 APICへのアクセスに使用したのと同じパスワードを使用して、ノードのいずれかにログインします。

例：
```
admin@apic1:~> attach node_name
```

ステップ 4 NTP ピアのステータスを表示します。

例：
```
leaf-1# show ntp peer-status
```
到達可能な NTP サーバの IPアドレスの前にはアスタリスク（*）が付き、遅延がゼロ以外の値になります。

ステップ 5 ステップ 3 および 4 を繰り返し、ファブリック内の各ノードを確認します。

ユーザ アカウントの作成

ローカル ユーザの設定

初期の設定スクリプトで、管理者アカウントが設定され、管理者はシステム起動時の唯一のユーザとなります。APIC は、きめ細かなロールベースのアクセス コントロール システムをサポートしており、そのシステムでは、権限が少ない管理者以外のユーザを含め、ユーザ アカウントをさまざまなロールで作成することができます。

リモート ユーザの設定

ローカルユーザを設定する代わりに、APICを一元化された企業クレデンシャルのデータセンターに向けることができます。APIC は、Lightweight Directory Access Protocol（LDAP）、Active Directory、RADIUS、および TACACS+ をサポートしています。

外部認証プロバイダーを通じて認証されたリモート ユーザを設定するには、次の前提条件を満たす必要があります。

- DNS 設定は、RADIUS サーバのホスト名ですでに名前解決されている必要があります。
- 管理サブネットを設定する必要があります。

REST API を使用したローカル ユーザの設定

手順

ローカル ユーザを作成します。

例：

```
URL: https://<apic-ip>/api/policymgr/mo/uni/userext.xml
POST CONTENT:
        <aaaUser name="operations" phone="" pwd="<strong_password>" >
          <aaaUserDomain childAction="" descr="" name="all" rn="userdomain-all" status="">

               <aaaUserRole childAction="" descr="" name="Ops" privType="writePriv"/>
           </aaaUserDomain>
      </aaaUser>
```

外部認証サーバの AV ペア

Cisco 属性/値（AV）ペアを既存のユーザ レコードに追加して、ユーザ権限を APIC コントローラに伝播することができます。Cisco AV ペアは、APIC ユーザに対してロールベース アクセス コントロール（RBAC）のロールと権限を指定するために使用する単一の文字列です。オープンRADIUS サーバ（/etc/raddb/users）の設定例は次のとおりです。

```
aaa-network-admin Cleartext-Password := "<password>"
Cisco-avpair = "shell:domains = all/aaa/read-all(16001)"
```

Cisco AV ペアが欠落しているか不良であるリモート ユーザのデフォルトの動作の変更

手順

ステップ 1 メニュー バーで、[ADMIN] > [AAA] の順にクリックします。

ステップ 2 [Navigation] ペインで、[AAA Authentication] をクリックします。

ステップ 3 [Work] ペインの [Properties] 領域で、[Remote user login policy] ドロップダウン リストから、[Assign Default Role] を選択します。
デフォルト値は [No Login] です。[Assign Default Role] オプションは、Cisco AV ペアが欠落しているか不良であるユーザに最小限の読み取り専用権限を割り当てます。不正なAVペアは、解析ルール適用時に問題があった AV ペアです。

AV ペアを割り当てるためのベスト プラクティス

ベストプラクティスとして、シスコは、bash シェルでユーザに割り当てられる AV ペアには 16000 ～ 23999 の範囲の一意の UNIX ユーザ ID を割り当てることを推奨します（SSH、Telnet または Serial/KVM のコンソールを使用）。Cisco AV ペアが UNIX ユーザ ID を提供しない状況が発生すると、そのユーザにはユーザ ID 23999 または範囲内の類似した番号が割り当てられます。これにより、そのユーザのホーム ディレクトリ、ファイル、およびプロセスに UNIX ID 23999 を持つリモート ユーザがアクセスできるようになってしまいます。

外部認証サーバの AV ペアの設定

属性/値（AV）のペア文字列のカッコ内の数字は、セキュア シェル（SSH）または Telnet を使用してログインしたユーザの UNIX ユーザ ID として使用されます。

手順

外部認証サーバの AV ペアを設定します。
Cisco AV ペアの定義は次のとおりです（シスコは、UNIX ユーザ ID が指定されているかどうかにかかわらず AV ペアをサポートします）。

例：

```
* shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2

         * shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2(8101)

These are the boost regexes supported by APIC:
uid_regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?),\\S+?/\\S*?/\\S*?){0,31})(\\(\\d+\\))$");
regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?),\\S+?/\\S*?/\\S*?){0,31})$");
```

次に、例を示します。

```
shell:domains = coke/tenant-admin/read-all,pepsi//read-all(16001)
```

REST API を使用したリモート ユーザの設定

手順

ステップ 1 RADIUS プロバイダーを作成します。

例：

```
 URL: https://<apic-ip>/api/policymgr/mo/uni/userext/radiusext.xml
POST Content:
<aaaRadiusProvider name="radius-auth-server.org.com" key="test123" />
```

ステップ 2 ログイン ドメインを作成します。

例：

```
 URL: https://<apic-ip>/api/policymgr/mo/uni/userext.xml
POST Content:
<aaaLoginDomain name="rad"> <aaaDomainAuth realm="radius"/> </aaaLoginDomain>
```

管理アクセスの追加

APIC コントローラには、管理ネットワークに到達するルートが 2 つあります。1 つはインバンド管理インターフェイスを使用し、もう 1 つはアウトオブバンド管理インターフェイスを使用します。

- インバンド管理アクセス：APIC および ACI ファブリックへのインバンド管理接続を設定できます。APIC がリーフ スイッチと通信するときに APIC によって使用される VLAN を最初に設定し、次に VMM サーバがリーフ スイッチとの通信に使用する VLAN を設定します。
- アウトオブバンド管理アクセス：APIC および ACI ファブリックへのアウトオブバンド管理接続を設定できます。アウトオブバンドエンドポイント グループ（EPG）に関連付けられるアウトオブバンド契約を設定し、外部ネットワーク プロファイルにその契約を接続します。

（注） APIC アウトオブバンド管理接続のリンクは、1 Gbps である必要があります。

APIC コントローラは、インバンド管理インターフェイスが設定されている場合は、アウトオブバンド管理インターフェイスを通してインバンド管理インターフェイスを常に選択します。アウトオブバンド管理インターフェイスは、インバンド管理インターフェイスが設定されていない場合、または宛先アドレスが APIC のアウトオブバンド管理サブネットと同じサブネットにある場合にのみ使用されます。この動作は、変更または再設定できません。

APIC 管理インターフェイスは IPv6 アドレスをサポートしないため、このインターフェイスを介して外部 IPv6 サーバに接続することはできません。

インバンドまたはアウトオブバンドの管理テナントで外部管理インスタンス プロファイルを設定しても、ファブリック全体の通信ポリシーで設定されているプロトコルには影響しません。外部管理インスタンス プロファイルで指定されているサブネットおよびコントラクトは、HTTP/HTTPS または SSH/Telnet には影響しません。

IPv4/IPv6 アドレスおよびインバンド ポリシー

インバンド管理アドレスは、ポリシーによってのみ（Postman REST API、NX-OS スタイル CLI、または GUI）APIC コントローラにプロビジョニングできます。また、インバンド管理アドレスは、各ノードに静的に設定する必要があります。

アウトオブバンド ポリシーの IPv4/IPv6 アドレス

アウトオブバンド管理アドレスは、ブートストラップ時に、またはポリシーを使用して（Postman REST API、NX-OS スタイル CLI、GUI）APIC コントローラにプロビジョニングできます。また、アウトオブバンド管理アドレスは、各ノードに静的にまたはクラスタ全体にアドレスの範囲（IPv4/IPv6）を指定することによって設定する必要があります。IP アドレスは、範囲からクラスタ内のノードにランダムに割り当てられます。

管理アクセスの設定

REST API を使用したインバンド管理アクセスの設定

インバンド管理アクセスでは、IPv4 アドレスと IPv6 アドレスがサポートされます。スタティック設定を使用した IPv6 設定がサポートされます（インバンドとアウトバンドの両方）。IPv4 および IPv6 のインバンドおよびアウトオブバンドのデュアル設定は、スタティック設定を使用する場合にのみサポートされます。詳細については、「*Configuring Static Management Access in Cisco APIC*」の KB 記事を参照してください。

手順

ステップ 1 VLAN ネームスペースを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/uni.xml -->
<polUni>
  <infraInfra>
    <!-- Static VLAN range -->
    <fvnsVlanInstP name="inband" allocMode="static">
      <fvnsEncapBlk name="encap" from="vlan-10" to="vlan-11"/>
    </fvnsVlanInstP>
  </infraInfra>
</polUni>
```

ステップ 2 物理ドメインを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/uni.xml -->
<polUni>
  <physDomP name="inband">
    <infraRsVlanNs tDn="uni/infra/vlanns-inband-static"/>
  </physDomP>
</polUni>
```

ステップ 3 インバンド管理用のセレクタを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <infraInfra>
    <infraNodeP name="vmmNodes">
      <infraLeafS name="leafS" type="range">
        <infraNodeBlk name="single0" from_="101" to_="101"/>
      </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-vmmPorts"/>
    </infraNodeP>

    <!-- Assumption is that VMM host is reachable via eth1/40. -->
    <infraAccPortP name="vmmPorts">
      <infraHPortS name="portS" type="range">
        <infraPortBlk name="block1"
                      fromCard="1" toCard="1"
                      fromPort="40" toPort="40"/>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-inband" />
      </infraHPortS>
    </infraAccPortP>


    <infraNodeP name="apicConnectedNodes">
      <infraLeafS name="leafS" type="range">
        <infraNodeBlk name="single0" from_="101" to_="102"/>
      </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-apicConnectedPorts"/>
    </infraNodeP>

    <!-- Assumption is that APIC is connected to eth1/1. -->
    <infraAccPortP name="apicConnectedPorts">
      <infraHPortS name="portS" type="range">
        <infraPortBlk name="block1"
                      fromCard="1" toCard="1"
                      fromPort="1" toPort="3"/>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-inband" />
      </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
      <infraAccPortGrp name="inband">
        <infraRsAttEntP tDn="uni/infra/attentp-inband"/>
      </infraAccPortGrp>
    </infraFuncP>

    <infraAttEntityP name="inband">
      <infraRsDomP tDn="uni/phys-inband"/>
    </infraAttEntityP>
  </infraInfra>
</polUni>
```

ステップ 4 インバンド ブリッジ ドメインとエンドポイント グループ（EPG）を設定します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="mgmt">
    <!-- Configure the in-band management gateway address on the
         in-band BD. -->
    <fvBD name="inb">
      <fvSubnet ip="10.13.1.254/24"/>
```

```
    </fvBD>

    <mgmtMgmtP name="default">
      <!-- Configure the encap on which APICs will communicate on the
           in-band network. -->
      <mgmtInB name="default" encap="vlan-10">
        <fvRsProv tnVzBrCPName="default"/>
      </mgmtInB>
    </mgmtMgmtP>
  </fvTenant>
</polUni>
```

ステップ 5 アドレス プールを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="mgmt">
    <!-- Adresses for APIC in-band management network -->
    <fvnsAddrInst name="apicInb" addr="10.13.1.254/24">
      <fvnsUcastAddrBlk from="10.13.1.1" to="10.13.1.10"/>
    </fvnsAddrInst>

    <!-- Adresses for switch in-band management network -->
    <fvnsAddrInst name="switchInb" addr="10.13.1.254/24">
      <fvnsUcastAddrBlk from="10.13.1.101" to="10.13.1.120"/>
    </fvnsAddrInst>
  </fvTenant>
</polUni>
```

（注） IPv6 のダイナミック アドレス プールはサポートされていません。

ステップ 6 管理グループを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <infraInfra>
    <!-- Management node group for APICs -->
    <mgmtNodeGrp name="apic">
      <infraNodeBlk name="all" from_="1" to_="3"/>
      <mgmtRsGrp tDn="uni/infra/funcprof/grp-apic"/>
    </mgmtNodeGrp>

    <!-- Management node group for switches-->
    <mgmtNodeGrp name="switch">
      <infraNodeBlk name="all" from_="101" to_="104"/>
      <mgmtRsGrp tDn="uni/infra/funcprof/grp-switch"/>
    </mgmtNodeGrp>

    <!-- Functional profile -->
    <infraFuncP>
      <!-- Management group for APICs -->
      <mgmtGrp name="apic">
        <!-- In-band management zone -->
        <mgmtInBZone name="default">
          <mgmtRsInbEpg tDn="uni/tn-mgmt/mgmtp-default/inb-default"/>
          <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-apicInb"/>
        </mgmtInBZone>
      </mgmtGrp>
```

```
      <!-- Management group for switches -->
      <mgmtGrp name="switch">
        <!-- In-band management zone -->
        <mgmtInBZone name="default">
          <mgmtRsInbEpg tDn="uni/tn-mgmt/mgmtp-default/inb-default"/>
          <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-switchInb"/>
        </mgmtInBZone>
      </mgmtGrp>
    </infraFuncP>
  </infraInfra>
</polUni>
```

（注） IPv6 のダイナミック アドレス プールはサポートされていません。

REST API を使用したアウトオブバンド管理アクセスの設定

アウトオブバンド管理アクセスでは、IPv4 アドレスと IPv6 アドレスがサポートされます。

はじめる前に

APIC アウトオブバンド管理接続のリンクは、1 Gbps である必要があります。

手順

ステップ 1 アウトオブバンド契約を作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <!-- Contract -->
        <vzOOBBrCP name="oob-default">
            <vzSubj name="oob-default">
                <vzRsSubjFiltAtt tnVzFilterName="default" />
            </vzSubj>
        </vzOOBBrCP>
    </fvTenant>
</polUni>
```

ステップ 2 アウトオブバンド契約をアウトオブバンド EPG に関連付けます。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <mgmtMgmtP name="default">
            <mgmtOoB name="default">
                <mgmtRsOoBProv tnVzOOBBrCPName="oob-default" />
            </mgmtOoB>
        </mgmtMgmtP>
    </fvTenant>
</polUni>
```

ステップ 3 アウトオブバンド契約を外部管理 EPG に関連付けます。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <mgmtExtMgmtEntity name="default">
            <mgmtInstP name="oob-mgmt-ext">
                <mgmtRsOoBCons tnVzOOBBrCPName="oob-default" />
                <!-- SUBNET from where switches are managed -->
                <mgmtSubnet ip="10.0.0.0/8" />
            </mgmtInstP>
        </mgmtExtMgmtEntity>
    </fvTenant>
</polUni>
```

ステップ 4 管理アドレス プールを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <fvnsAddrInst name="switchOoboobaddr" addr="172.23.48.1/21">
            <fvnsUcastAddrBlk from="172.23.49.240" to="172.23.49.244"/>
        </fvnsAddrInst>
    </fvTenant>
</polUni>
```

ステップ 5 ノード管理グループを作成します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <infraInfra>
        <infraFuncP>
            <mgmtGrp name="switchOob">
                <mgmtOoBZone name="default">
                    <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-switchOoboobaddr" />
                    <mgmtRsOobEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default" />
                </mgmtOoBZone>
            </mgmtGrp>
        </infraFuncP>
        <mgmtNodeGrp name="switchOob">
            <mgmtRsGrp tDn="uni/infra/funcprof/grp-switchOob" />
            <infraNodeBlk name="default" from_="101" to_="103" />
        </mgmtNodeGrp>
    </infraInfra>
</polUni>
```

REST API を使用した APIC コントローラの IP アドレスの変更

手順

APIC コントローラの IP アドレスを変更します。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- /api/policymgr/mo/.xml -->
<polUni>
  <infraInfra>
    <infraFuncP>
      <mgmtGrp name="mgmtGroupApic">
        <mgmtOoBZone name="mgmtOobZoneApic">
          <mgmtRsOobEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default"/>
          <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-oobAddrApic"/>
        </mgmtOoBZone>
      </mgmtGrp>
    </infraFuncP>
    <mgmtNodeGrp name="mgmtNodeGroupApic">
      <mgmtRsGrp tDn="uni/infra/funcprof/grp-mgmtGroupApic"/>
      <infraNodeBlk name="default" from_="1" to_="1"/>
    </mgmtNodeGrp>
  </infraInfra>
</polUni>

<?xml version="1.0" encoding="UTF-8"?>
<!-- /api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="mgmt">
    <fvnsAddrInst name="oobAddrApic" addr="172.23.48.1/21">
      <fvnsUcastAddrBlk from="172.23.48.16" to="172.23.48.16"/>
    </fvnsAddrInst>
  </fvTenant>
</polUni>
```

次の作業

- APIC コントローラに再接続するには、新しい IP アドレスを使用する必要があります。
- 新しい IP アドレスがコントローラに割り当てられたら、コントローラの古い IP アドレスを削除する必要があります。

既存の IP tables 機能をミラーリングする IPv6 の変更

すべての IPv6 は、ネットワーク アドレス変換（NAT）を除いて、既存の IP tables 機能をミラーリングします。

既存の IP tables

1 以前は、IPv6 テーブルのすべてのルールが一度に 1 つずつ実行され、すべてのルールの追加または削除に対してシステム コールが行われていました。

2 新しいポリシーが追加されるたびに、ルールが既存の IP tables ファイルに追加され、ファイルへの追加変更は行われませんでした。

3 新しい送信元ポートがアウトオブバンドポリシーで設定されると、同じポート番号で送信元と宛先のルールを追加しました。

IP tables への変更

1 IP tables が作成されると、はじめにハッシュ マップに書き込まれ、次に中間ファイル IP tables-new に書き込まれてこれが復元されます。保存すると、新しい IP tables ファイルが /etc/sysconfig/ フォルダに作成されます。これら両方のファイルは同じ場所にあります。すべてのルールにシステム コールを行う代わりに、ファイルを復元および保存している時にのみシステム コールを行う必要があります。

2 ルールを追加する代わりに新しいポリシーがファイルに追加されると、hashmaps にデフォルトポリシーをロードし、新しいポリシーを確認し、hashmaps に追加することによって、IP テーブルがゼロから作成されます。その後、中間ファイル（/etc/sysconfig/iptables-new）に書き込まれて保存されます。

3 アウトオブバンド ポリシーのルールの送信元ポートだけを設定することはできません。宛先ポートまたは送信元ポートいずれかを宛先ポートとともにルールに追加できます。

4 新しいポリシーが追加されると、新しいルールが IP tables ファイルに追加されます。このルールは、IP tables デフォルト ルールのアクセス フローを変更します。

```
-A INPUT -s <OOB Address Ipv4/Ipv6> -j apic-default
```

5 新しいルールが追加された場合、これは IP tables-new ファイルに存在して IP tables ファイルには存在せず、IP tables-new ファイルにエラーがあることを意味します。復元が正常な場合に限り、ファイルが保存され、新しいルールを IP tables ファイルで確認できます。

（注）
- IPv4 のみ有効な場合、IPv6 ポリシーを設定しないでください。
- IPv6 のみ有効な場合、IPv4 ポリシーを設定しないでください。
- IPv4 と IPv6 の両方が有効な場合にポリシーが追加されると、両方のバージョンに設定されます。したがって、IPv4 サブネットを追加すると IP tables に追加され、同様に IPv6 サブネットは IPv6 tables に追加されます。

管理接続モード

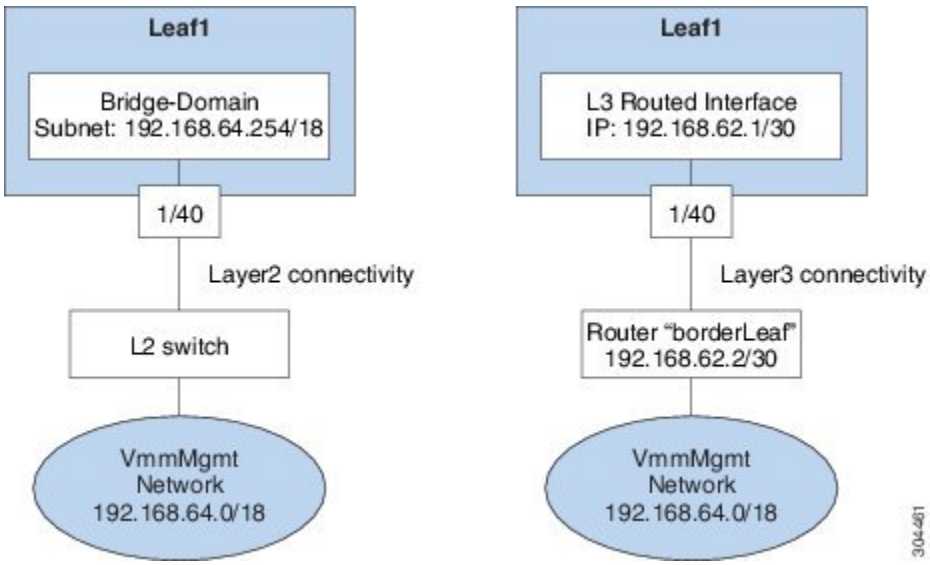
アウトオブバンドとインバンド管理接続が設定されているかどうかに応じて、アウトオブバンドまたはインバンドネットワークを使用する外部エンティティへの接続を確立します。vCenter Server などの外部エンティティへの接続を確立するには次の 2 つのモードを使用できます。

- レイヤ 2 管理接続：外部エンティティがレイヤ 2 を使用してリーフ ノードに接続されている場合は、このモードを使用します。
- レイヤ 3 管理接続：外部エンティティがレイヤ 3 を使用してルータを介してリーフ ノードに接続されている場合は、このモードを使用します。リーフは、外部エンティティに到達可能なルータに接続されます。

（注）
- インバンド IP アドレス範囲は、リーフ ノードからファブリックの外へのレイヤ 3 接続で使用される IP アドレス範囲から分離して異なっている必要があります。
- レイヤ 3 インバンド管理の設計では、トポロジ内のスパイン ファブリック ノードへのインバンド管理アクセスは提供されません。

次の図は、接続の確立に使用可能な 2 つのモードを示します。

図 *2*：レイヤ *2* およびレイヤ *3* 管理接続の例



REST API を使用したレイヤ 2 管理接続の設定

（注） 名前 vmm がこのタスクで文字列の例として使用されます。

ポリシーは、Tenant-mgmt 下で次のオブジェクトを作成します。

次のようにブリッジ ドメイン（vmm）および次の関連オブジェクトを作成します。

- このブリッジ ドメイン内で、この IP プレフィクス（192.168.64.254/18）でサブネット オブジェクトを作成します。この IP アドレス（192.168.64.254）は、従来のスイッチ設定でスイッチ仮想インターフェイス（SVI）として通常使用されるブリッジ ドメインに割り当てられます。
- インバンド ネットワーク（ctx）への関連付けを作成します。

次のように関連オブジェクトを有するアプリケーション プロファイル（vmm）と管理 EPG（vmmMgmt）を作成します。

- ブリッジ ドメイン（vmm）への関連付けを作成します。
- この EPG を leaf1 に展開するポリシーを作成します。この EPG に使用されるカプセル化は vlan-11 です。

はじめる前に

vCenter ドメインプロファイルを作成する前に、インバンド管理ネットワークを使用して外部ネットワークを確立するための接続を確立する必要があります。

管理接続ポリシーの一部として設定された IP アドレス範囲が ACI ファブリックで使用されるインフラストラクチャの IP アドレス範囲と重複していないことを確認します。

手順

リーフ ポートに接続されるルータを使用して、APIC から外部ルートへの接続を確立できます。

例：

```
POST
https://APIC-IP/api/mo/uni.xml

<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="mgmt">
    <fvBD name="vmm">
      <fvRsCtx tnFvCtxName="inb"/>
      <fvSubnet ip='192.168.64.254/18'/>
    </fvBD>

    <fvAp name="vmm">
      <fvAEPg name="vmmMgmt">
        <fvRsBd tnFvBDName="vmm" />
        <fvRsPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]" encap="vlan-11"/>
        <fvRsCons tnVzBrCPName="default"/>
        <fvRsDomAtt tDn="uni/phys-inband"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```

REST API を使用したレイヤ 3 管理接続の設定

名前 vmm がこのタスクで文字列の例として使用されます。

ポリシーは、Tenant-mgmt 下で次のオブジェクトを作成します。

- 次の手順でルーテッド外部ポリシー（vmm）を作成します。

1. レイヤ 3 外部ネットワーク インスタンス プロファイル オブジェクト（vmmMgmt）を作成します。
2. ネクストホップ ルータ 192.168.62.2 の IP アドレスでリモート ネットワーク（192.168.64.0/18）のルートを作成します。
3. leaf1 に接続されている論理ノード プロファイル オブジェクト（borderLeaf）を作成します。
4. IP アドレス 192.168.62.1/30 のルーテッド インターフェイス 1/40 でポート プロファイル（portProfile1）を作成します。
5. インバンド ネットワーク（ctx）への関連付けを作成します。

はじめる前に

管理接続ポリシーの一部として設定された IP アドレス範囲が ACI ファブリックで使用されるインフラストラクチャの IP アドレス範囲と重複していないことを確認します。

手順

リーフ ポートに接続されるルータを使用して、APIC から外部ルートへの接続を確立できます。

例：

```
<!-- api/policymgr/mo/.xml -->
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
   <fvTenant name="mgmt">
      <l3extOut name="vmm">
         <l3extInstP name="vmmMgmt">
            <l3extSubnet ip="192.168.0.0/16" />
            <fvRsCons tnVzBrCPName="default" />
         </l3extInstP>
         <l3extLNodeP name="borderLeaf">
            <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="1.2.3.4">
               <ipRouteP ip="192.168.64.0/18">
                  <ipNexthopP nhAddr="192.168.62.2" />
               </ipRouteP>
            </l3extRsNodeL3OutAtt>
            <l3extLIfP name="portProfile">
               <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
ifInstT="l3-port" addr="192.168.62.1/30" />
            </l3extLIfP>
         </l3extLNodeP>
         <l3extRsEctx tnFvCtxName="inb" />
      </l3extOut>
   </fvTenant>
</polUni>
```

管理接続の検証

この検証プロセスは、レイヤ 2 とレイヤ 3 の両方のモードに適用され、APIC GUI、REST API、または CLI を使用して確立される接続を確認するために使用できます。

管理接続を確立するための手順を完了したら、APIC コンソールにログインします。到達可能な vCenter Server の IP アドレス（たとえば、192.168.81.2）に ping を送り、その ping が機能することを確認します。この操作は、ポリシーが正常に適用されたことを示します。

VMM ドメインの設定

仮想マシン ネットワーキング ポリシーの設定

APIC は、サードパーティの VM マネージャ（VMM）（VMware vCenter および SCVMM など）と統合し、ACI の利点を仮想化されたインフラストラクチャに拡張します。APIC によって、VMM システム内の ACI ポリシーをその管理者が使用できるようになります。

ここでは、VMware vCenter および vShield を使用する VMM 統合の例を示します。シスコ ACI と VMM 統合の異なるモードに関する詳細については、『*ACI Virtualization Guide*』を参照してください。

VM マネージャについて

（注） vCenter との統合のために必要な APIC の設定に関する情報を次に示します。VMware コンポーネントの設定手順については、VMware のマニュアルを参照してください。

次は、VM マネージャの用語の詳細情報です。

- VM コントローラは、VMware vCenter や VMware vShield などの、外部仮想マシン管理エンティティです。APICは、コントローラと通信し、仮想ワークロードに適用されるネットワーク ポリシーを公開します。VM コントローラの管理者は、APIC 管理者に VM コントローラの認証クレデンシャルを提供します。同じタイプの複数のコントローラが同じクレデンシャルを使用できます。
- クレデンシャルは、VM コントローラと通信するための認証クレデンシャルを表します。複数のコントローラが同じクレデンシャルを使用できます。
- 仮想マシンのモビリティ ドメイン（vCenter のモビリティ ドメイン）は、同様のネットワーキング ポリシー要件を持つ VM コントローラのグループです。この必須コンテナは、VLAN プールなどのためのポリシー、サーバ/ネットワーク MTU ポリシー、またはサーバ/ネットワーク アクセス LACP ポリシーとともに 1 つ以上の VM コントローラを保持します。エンドポイント グループが vCenter ドメインに関連付けられると、ネットワーク ポリシーが vCenter ドメイン内のすべての VM コントローラにプッシュされます。
- プールは、トラフィックのカプセル化 ID の範囲を表します（たとえば、VLAN ID、VNID、マルチキャスト アドレスなど）。プールは共有リソースで、VMM などの複数のドメインおよびレイヤ 4 ～ レイヤ 7 のサービスで消費できます。リーフ スイッチは、重複した VLAN

プールをサポートしていません。異なる重複した VLAN プールを VMM ドメインと関連付けることはできません。VLAN ベースのポートには、次の 2 種類があります。

- ダイナミック プール：APIC によって内部的に管理され、エンドポイント グループ（EPG）の VLAN を割り当てます。vCenter ドメインはダイナミック プールのみに関連付けることができます。
- スタティック プール：EPG にはドメインとの関係があり、ドメインにはプールとの関係があります。プールには、さまざまなカプセル化された VLAN および VXLAN が含まれます。スタティック EPG 導入環境の場合、ユーザはインターフェイスとカプセル化を定義します。カプセル化は、EPG が関連付けられているドメインに関連付けられたプールの範囲内である必要があります。

- 導入する VMware vCenter では、VLAN モードまたは VXLAN モードで動作する必要があります。VMM ドメインは VLAN プールに関連付け、vShield は vCenter に関連付ける必要があります。

接続可能エンティティ プロファイルについて

接続エンティティ プロファイル

ACI ファブリックにより、リーフ ポートを通じて baremetal サーバ、ハイパーバイザ、レイヤ 2 スイッチ（たとえば、Cisco UCS ファブリック インターコネクト）、レイヤ 3 ルータ（たとえば、Cisco Nexus 7000 シリーズ スイッチ）などのさまざまな**外部エンティティ**に接続する複数の**接続ポイント**が提供されます。 これらの接続ポイントは、リーフ スイッチ上の物理ポート、ポート チャネル、または仮想ポート チャネル（vPC）にすることができます。

接続可能エンティティ プロファイル（AEP）は、同様のインフラストラクチャ ポリシー要件を持つ外部エンティティのグループを表します。インフラストラクチャ ポリシーは、物理インターフェイス ポリシーで構成され、たとえば Cisco Discovery Protocol（CDP）、Link Layer Discovery Protocol（LLDP）、最大伝送単位（MTU）、Link Aggregation Control Protocol（LACP）などがあります。

VM マネージャ（VMM）ドメインは、AEP に関連付けられたインターフェイス ポリシー グループから物理インターフェイス ポリシーを自動的に取得します。

- AEP でオーバーライド ポリシーを VMM ドメイン用の別の物理インターフェイス ポリシーを指定するために使用できます。このポリシーは、ハイパーバイザが中間レイヤ 2 ノードを介してリーフ スイッチに接続され、異なるポリシーがリーフ スイッチおよびハイパーバイザの物理ポートで要求される場合に役立ちます。たとえば、リーフ スイッチとレイヤ 2 ノード間で LACP を設定できます。同時に、AEP オーバーライド ポリシーで LACP をディセーブルにすることで、ハイパーバイザとレイヤ 2 スイッチ間の LACP をディセーブルにできます。

AEP は、リーフ スイッチで VLAN プールを展開するのに必要です。異なるリーフ スイッチ間でカプセル化プール（たとえば VLAN）を再利用することができます。AEP は、（ドメインに関連付けられた）VLAN プールの範囲を物理インフラストラクチャに暗黙的に提供します。

（注）

- AEP は、リーフ上で VLAN プール（および関連 VLAN）をプロビジョニングします。VLAN はポートでは実際にイネーブルになっていません。EPG がポートに展開されていない限り、トラフィックは流れません。
- AEP を使用して VLAN プールを展開しないと、EPG がプロビジョニングされても VLAN はリーフ ポートでイネーブルになりません。
  - リーフ ポートで静的にバインディングしている EPG イベントに基づいて、または VMware vCenter などの外部コントローラからの VM イベントに基づいて、特定の VLAN がリーフ ポート上でプロビジョニングされるかイネーブルになります。
  - EPG で VMM カプセル化を静的に設定する場合は、スタティック プールを使用する必要があります。静的割り当てと動的割り当てを組み合わせる場合は、ダイナミック プールを作成し、スタティック モードでそのプール内にブロックを追加します。
- リーフ スイッチは、重複した VLAN プールをサポートしていません。異なる重複した VLAN プールをドメインを介して関連付けられる同一の AEP に関連付けることはできません。

LLDP および CDP の設定の詳細については、本ガイドのブレード サーバとの連携に関する章を参照してください。

VMM ドメイン プロファイルを作成するための前提条件

VMM ドメイン プロファイルを設定するには、次の前提条件を満たす必要があります。

- すべてのファブリック ノードが検出され、設定されている。
- インバンド（inb）またはアウトオブバンド（oob）管理が APIC 上で設定されている。
- Virtual Machine Manager（VMM）がインストールされ、設定されて、inb/oob 管理ネットワーク（たとえば、vCenter）経由で到達可能である。
- VMM の管理者とルートのクレデンシャルがある（vCenter など）。

（注） vCenter の管理者とルートのクレデンシャルを使用しない場合は、必要な最小アクセス許可を持つカスタム ユーザ アカウントを作成できます。必要なユーザ権限のリストについては、最小 VMware vCenter 権限を持つカスタム ユーザ アカウントを参照してください。

- IP アドレスではなくホスト名で VMM を参照する予定がある場合は、APIC の DNS ポリシーを設定する必要があります。
- VMware vShield のドメイン プロファイルを作成している場合は、DHCP サーバとリレー ポリシーを設定する必要があります。

最小 VMware vCenter 権限を持つカスタム ユーザ アカウント

Cisco APIC から vCenter を設定するには、vCenter で次の最小権限セットが許可されるクレデンシャルである必要があります。

- アラーム
- 分散スイッチ
- dvPort グループ
- フォルダ
- ホスト
  - 詳細設定
  - ローカル操作.再構成済み仮想マシン
  - ネットワーク設定
- ネットワーク
- 仮想マシン
  - 仮想マシン.構成.デバイス設定の変更
  - 仮想マシン.構成.設定

これにより、APIC は vCenter に VMware API コマンドを送信して、DVS/AVS の作成、VMK インターフェイス（AVS）の作成、ポート グループの発行および必要なすべてのアラートのリレーを行うことができます。

VMM ドメイン プロファイルの作成

この項では、VMM ドメインの例は、vCenter ドメインまたは vCenter および vShield ドメインです。

REST API を使用した vCenter ドメイン プロファイルの作成

手順

ステップ 1 VMM ドメイン名、コントローラおよびユーザ クレデンシャルを設定します。

例：

```
POST URL: https://<api-ip>/api/node/mo/.xml
```

```
<polUni>
<vmmProvP vendor="VMware">
<!-- VMM Domain -->
<vmmDomP name="productionDC">
<!-- Association to VLAN Namespace -->
<infraRsVlanNs tDn="uni/infra/vlanns-VlanRange-dynamic"/>
<!-- Credentials for vCenter -->
<vmmUsrAccP name="admin" usr="administrator" pwd="admin" />
<!-- vCenter IP address -->
<vmmCtrlrP name="vcenter1" hostOrIp="<vcenter ip address>" rootContName="<Datacenter Name
in vCenter>">
<vmmRsAcc tDn="uni/vmmp-VMware/dom-productionDC/usracc-admin"/>
</vmmCtrlrP>
</vmmDomP>
</vmmProvP>
```

ステップ 2 VLAN ネームスペースの導入用の接続可能エンティティ プロファイルを作成します。

例：

```
POST URL: https://<apic-ip>/api/policymgr/mo/uni.xml
<infraInfra>
<infraAttEntityP name="profile1">
<infraRsDomP tDn="uni/vmmp-VMware/dom-productionDC"/>
</infraAttEntityP>
</infraInfra>
```

ステップ 3 インターフェイス ポリシー グループおよびセレクタを作成します。

例：

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

<infraInfra>
    <infraAccPortP name="swprofile1ifselector">
        <infraHPortS name="selector1" type="range">
            <infraPortBlk name="blk"
             fromCard="1" toCard="1" fromPort="1" toPort="3">
            </infraPortBlk>
     <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-group1" />
        </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
        <infraAccPortGrp name="group1">
            <infraRsAttEntP tDn="uni/infra/attentp-profile1" />
        </infraAccPortGrp>
    </infraFuncP>
</infraInfra>
```

ステップ 4 スイッチ プロファイルを作成します。

例：

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

<infraInfra>
    <infraNodeP name="swprofile1">
        <infraLeafS name="selectorswprofile11718" type="range">
            <infraNodeBlk name="single0" from_="101" to_="101"/>
            <infraNodeBlk name="single1" from_="102" to_="102"/>
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-swprofile1ifselector"/>
    </infraNodeP>
</infraInfra>
```

ステップ 5 VLAN プールを設定します。

例：

```
 POST URL: https://<apic-ip>/api/node/mo/.xml

<polUni>
<infraInfra>
<fvnsVlanInstP name="VlanRange" allocMode="dynamic">
   <fvnsEncapBlk name="encap" from="vlan-100" to="vlan-400"/>
</fvnsVlanInstP>
</infraInfra>
</polUni>
```

ステップ 6 設定されたすべてのコントローラとそれらの動作状態を検索します。

例：

```
GET:
https://<apic-ip>/api/node/class/compCtrlr.xml?
<imdata>
<compCtrlr apiVer="5.1" ctrlrPKey="uni/vmmp-VMware/dom-productionDC/ctrlr-vcenter1"
deployIssues="" descr="" dn="comp/prov-VMware/ctrlr-productionDC-vcenter1" domName="
productionDC"
hostOrIp="esx1" mode="default" model="VMware vCenter Server 5.1.0 build-756313"
name="vcenter1" operSt="online" port="0" pwd="" remoteOperIssues="" scope="vm"
usr="administrator" vendor="VMware, Inc." ... />
</imdata>
```

ステップ 7 「ProductionDC」という VMM ドメイン下の「vcenter1」という名前の vCenter をハイパーバイザと VM で検索します。

例：

```
GET:
https://<apic-ip>/api/node/mo/comp/prov-VMware/ctrlr-productionDC-vcenter1.xml?query-target=children

<imdata>
<compHv descr="" dn="comp/prov-VMware/ctrlr-productionDC-vcenter1/hv-host-4832" name="esx1"
state="poweredOn" type="hv" ... />
<compVm descr="" dn="comp/prov-VMware/ctrlr-productionDC-vcenter1/vm-vm-5531" name="AppVM1"
state="poweredOff" type="virt" .../>
<hvsLNode dn="comp/prov-VMware/ctrlr-productionDC-vcenter1/sw-dvs-5646" lacpEnable="yes"
lacpMode="passive" ldpConfigOperation="both" ldpConfigProtocol="lldp" maxMtu="1500"
mode="default" name="apicVswitch" .../>
</imdata>
```

REST API を使用した vCenter および vShield ドメイン プロファイルの作成

手順

ステップ 1 VLAN プールを作成します。

例：

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

<polUni>
```

```
    <infraInfra>
        <fvnsVlanInstP name="vlan1" allocMode="dynamic">
            <fvnsEncapBlk name="encap" from="vlan-100" to="vlan-400"/>
        </fvnsVlanInstP>
    </infraInfra>
</polUni>
```

ステップ 2 vCenter ドメインを作成し、VLAN プールを割り当てます。

例：

```
POST URL: https://<apic-ip>/api/policymgr/mo/uni.xml
<vmmProvP dn="uni/vmmp-VMware">
<vmmDomP name="productionDC">
<infraRsVlanNs tDn="uni/infra/vlanns-vlan1-dynamic"/>
</vmmDomP>
</vmmProvP>
```

ステップ 3 インフラストラクチャ VLAN の導入用の接続可能エンティティ プロファイルを作成します。

例：

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

<infraInfra>
    <infraAttEntityP name="profile1">
        <infraRsDomP tDn="uni/vmmp-VMware/dom-productionDC"/>
        <infraProvAcc name="provfunc"/>
    </infraAttEntityP>
</infraInfra>
```

ステップ 4 インターフェイス ポリシー グループおよびセレクタを作成します。

例：

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

<infraInfra>
    <infraAccPortP name="swprofile1ifselector">
        <infraHPortS name="selector1" type="range">
            <infraPortBlk name="blk"
             fromCard="1" toCard="1" fromPort="1" toPort="3">
            </infraPortBlk>
     <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-group1" />
        </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
        <infraAccPortGrp name="group1">
            <infraRsAttEntP tDn="uni/infra/attentp-profile1" />
        </infraAccPortGrp>
    </infraFuncP>
</infraInfra>
```

ステップ 5 スイッチ プロファイルを作成します。

例：

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

<infraInfra>
    <infraNodeP name="swprofile1">
        <infraLeafS name="selectorswprofile11718" type="range">
            <infraNodeBlk name="single0" from_="101" to_="101"/>
            <infraNodeBlk name="single1" from_="102" to_="102"/>
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-swprofile1ifselector"/>
    </infraNodeP>
</infraInfra>
```

ステップ 6 コントローラのクレデンシャルを作成します。

例：

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

<vmmProvP dn="uni/vmmp-VMware">
    <vmmDomP name="productionDC">
        <vmmUsrAccP name="vcenter_user" usr="administrator" pwd="default"/>
        <vmmUsrAccP name="vshield_user" usr="admin" pwd="default"/>
    </vmmDomP>
</vmmProvP>
```

ステップ 7 vCenter コントローラを作成します。

例：

```
<vmmProvP dn="uni/vmmp-VMware">
    <vmmDomP name="productionDC">
        <vmmCtrlrP name="vcenter1" hostOrIp="172.23.50.85" rootContName="Datacenter1">
            <vmmRsAcc tDn="uni/vmmp-VMware/dom-productionDC/usracc-vcenter_user"/>
        </vmmCtrlrP>
    </vmmDomP>
</vmmProvP>
```

ステップ 8 VXLAN プールおよびマルチキャスト アドレス範囲を作成します。

例：

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

<infraInfra>
    <fvnsVxlanInstP name="vxlan1">
        <fvnsEncapBlk name="encap" from="vxlan-6000" to="vxlan-6200"/>
    </fvnsVxlanInstP>
    <fvnsMcastAddrInstP name="multicast1">
        <fvnsMcastAddrBlk name="mcast" from="224.0.0.1" to="224.0.0.20"/>
    </fvnsMcastAddrInstP>
</infraInfra>
```

ステップ 9 vShield コントローラを作成します。

例：

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

<vmmProvP dn="uni/vmmp-VMware">
    <vmmDomP name="productionDC">
        <vmmCtrlrP name="vshield1" hostOrIp="172.23.54.62" scope="iaas">
            <vmmRsAcc tDn="uni/vmmp-VMware/dom-productionDC/usracc-vshield_user"/>
            <vmmRsVmmCtrlrP tDn="uni/vmmp-VMware/dom-productionDC/ctrlr-vcenter1"/>
            <vmmRsVxlanNs tDn="uni/infra/vxlanns-vxlan1"/>
            <vmmRsMcastAddrNs tDn="uni/infra/maddrns-multicast1"/>
        </vmmCtrlrP>
    </vmmDomP>
</vmmProvP>
```

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 ネイバー探索の有効化の詳細については、関連 KB 記事、「*KB: Creating a Tenant, VRF, and Bridge Domain with IPv6 Neighbor Discovery*」を参照してください。

REST API を使用したテナント、VRF、およびブリッジ ドメインの作成

手順

ステップ 1 テナントを作成します。

例：

```
 POST <IP>/api/mo/uni.xml
<fvTenant name="ExampleCorp"/>
```

POST が成功すると、出力に作成したオブジェクトが表示されます。

ステップ 2 VRF およびブリッジ ドメインを作成します。

（注） ゲートウェイ アドレスは、IPv4 または IPv6 アドレスにすることができます。IPv6 ゲートウェイ アドレスの詳細については、関連する KB 記事「*KB: Creating a Tenant, VRF, and Bridge Domain with IPv6 Neighbor Discovery*」を参照してください。

例：

```
 URL for POST: https://<apic-ip>/api/mo/uni/tn-ExampleCorp.xml

<fvTenant name="ExampleCorp">
   <fvCtx name="pvn1"/>
   <fvBD name="bd1">
      <fvRsCtx tnFvCtxName="pvn1"/>
      <fvSubnet ip="10.10.100.1/24"/>
   </fvBD>
</fvTenant>
```

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

サーバまたはサービス ポリシーの設定

DHCP リレー ポリシーの設定

DHCP リレー ポリシーは、DHCP クライアントとサーバが異なるサブネット上にある場合に使用できます。クライアントが配置された vShield ドメイン プロファイルとともに ESX ハイパーバイザ上にある場合は、DHCP リレー ポリシー設定を使用することが必須です。

vShield コントローラが Virtual Extensible Local Area Network（VXLAN）を展開すると、ハイパーバイザホストはカーネル（vmkN、仮想トンネルエンドポイント（VTEP））インターフェイスを作成します。これらのインターフェイスは、DHCP を使用するインフラストラクチャ テナントで IP アドレスを必要とします。したがって、APIC が DHCP サーバとして動作しこれらの IP アドレスを提供できるように、DHCP リレー ポリシーを設定する必要があります。

ACI fabricは、DHCP リレーとして動作するときに、DHCP オプション 82（DHCP Relay Agent Information Option）を、クライアントの代わりに中継する DHCP 要求に挿入します。応答（DHCP オファー）がオプション 82 なしで DHCP サーバから返された場合、その応答はファブリックによってサイレントにドロップされます。したがって、ACI fabricが DHCP リレーとして動作するときは、ACI fabricに接続されたノードを計算するために IP アドレスを提供している DHCP サーバはオプション 82 をサポートする必要があります。

REST API を使用した APIC インフラストラクチャの DHCP サーバ ポリシーの設定

- このタスクは、vShield ドメイン プロファイルを作成するユーザの前提条件です。
- アプリケーション EPG で使用されるポートおよびカプセル化は、物理または VM マネージャ（VMM）ドメインに属している必要があります。ドメインとのそのような関連付けが確立されていないと、APIC は EPG の展開を続行しますが、エラーを生成します。

- Cisco APIC は、IPv4 と IPv6 の両方のテナント サブネットで DHCP リレーをサポートします。DHCP サーバ アドレスには IPv4 または IPv6 を使用できます。DHCPv6 リレーは、ファブリック インターフェイスで IPv6 が有効になっており、1 つ以上の DHCPv6 リレー サーバが設定されている場合にのみ、発生します。

はじめる前に

レイヤ 2 またはレイヤ 3 管理接続が設定されていることを確認します。

手順

インフラストラクチャ テナントの DHCP サーバ ポリシーとして APIC を設定します。

（注） このリレー ポリシーは、接続エンティティ プロファイルの設定を使用した接続されたハイパーバイザであるすべてのリーフ ポートにプッシュされます。接続エンティティ プロファイルによる設定の詳細については、VMM ドメインプロファイルの作成に関連する例を参照してください。

例：

```
<!-- api/policymgr/mo/.xml -->
<polUni>

POST URL:
https://APIC-IP/api/mo/uni.xml


  <fvTenant name="infra">

    <dhcpRelayP name="DhcpRelayP"  owner="tenant">
      <dhcpRsProv tDn="uni/tn-infra/ap-access/epg-default" addr="10.0.0.1" />
    </dhcpRelayP>

    <fvBD name="default">
      <dhcpLbl name="DhcpRelayP" owner="tenant"/>
    </fvBD>

  </fvTenant>
</polUni>
```

DNS サービス ポリシーの設定

DNS ポリシーは、ホスト名で外部サーバ（AAA、RADIUS、vCenter、サービスなど）に接続するために必要です。DNS サービス ポリシーは共有ポリシーであるため、このサービスを使用するすべてのテナントと VRF を特定の DNS プロファイル ラベルで設定する必要があります。ACI ファブリックの DNS ポリシーを設定するには、次のタスクを完了する必要があります。

- 管理 EPG が DNS ポリシー用に設定されていることを確認してください。設定されていない場合、このポリシーはスイッチで有効になりません。
- DNS プロバイダーと DNS ドメインに関する情報が含まれる DNS プロファイル（デフォルト）を作成します。
- DNS プロファイル（デフォルトまたは別の DNS プロファイル）の名前を必要なテナントで DNS ラベルに関連付けます。

テナントごと、VRF ごとの DNS プロファイル設定を設定することができます。適切な DNS ラベルを使用して、追加の DNS プロファイルを作成して、特定のテナントの特定の VRF に適用できます。たとえば、名前が acme の DNS プロファイルを作成する場合、テナント設定で acme の DNS ラベルを適切な [Networking] > [VRF] ポリシー設定に追加できます。

インバンド DNS サービス ポリシーによる外部宛先の設定

次のように、サービスに対して外部宛先を設定します。

ソース	インバンド管理	アウトオブバンド管理	外部サーバの場所
APIC	IP アドレスまたは完全修飾ドメイン名（FQDN）	IP アドレスまたは FQDN	Anywhere
リーフ スイッチ	IP アドレス	IP アドレスまたは FQDN （注） DNS ポリシーは、DNS サーバの到達可能性に対するアウトオブバンド管理 EPG を指定する必要があります。	Anywhere
スパイン スイッチ	IP アドレス	IP アドレスまたは FQDN （注） DNS ポリシーは、DNS サーバの到達可能性に対するアウトオブバンド管理 EPG を指定する必要があります。	リーフ スイッチに直接接続されます

次に示すのは、外部サーバのリストです。

- Call Home SMTP サーバ
- Syslog サーバ
- SNMP トラップの宛先

- 統計情報のエクスポートの宛先
- エクスポートの設定の宛先
- Techsupport のエクスポートの宛先
- コア エクスポートの宛先

推奨されるガイドラインは次のとおりです。

- 外部サーバは、リーフ アクセス ポートに接続する必要があります。
- 管理ポートの追加の配線を避けるために、リーフ スイッチにはインバンド接続を使用します。
- スパイン スイッチにはアウトオブバンド管理接続を使用します。スパイン スイッチとリーフ スイッチが外部サーバの同じセットに到達できるように、スパイン スイッチのこのアウトオブバンドネットワークをインバンド管理の仮想ルーティングおよび転送（VRF）機能があるリーフ ポートの 1 つに接続します。
- 外部サーバには IP アドレスを使用します。

DNS プロファイルの IPv4 または IPv6 の優先順位のポリシー

DNS プロファイルは、IPv4 と IPv6 のバージョン優先順位の選択をサポートします。ユーザ インターフェイスを使用して、優先順位を有効にすることができます。IPv4 がデフォルトです。

次の例は、Postman REST API を使用したポリシーベースの設定を示します。

```
<?xml version="1.0" encoding="UTF-8”?>
<!— api/node/mo/uni/fabric/dnsp-default.xml —>
<dnsProfile dn="uni/fabric/dnsp-default" IPVerPreference="IPv6" childAction="" descr="" >
</dnsProfile>
```

gai.conf の設定は、宛先アドレス選択を制御します。ファイルには、ラベル テーブル、優先順位テーブル、IPv4 範囲テーブルが含まれます。IPv4 または IPv6 をもう一方よりも優先付けする変更は、優先順位テーブルのエントリに含める必要があります。Linux システムで多数のフレーバーに使用されている標準ファイルの内容例を下に示します。ファイルの precedence ラベルの一行でデフォルト設定を上書きします。

次の例は、IPv4 を IPv6 よりも優先させるための gai.conf です。

```
# Generated by APIC
label ::1/128       0
label ::/0          1
label 2002::/16     2
label ::/96         3
label ::ffff:0:0/96 4
precedence  ::1/128       50
precedence ::/0           40
precedence 2002::/16      30
precedence ::/96          20
# For APICs prefering IPv4 connections, change the value to 100.
precedence ::ffff:0:0/96  10
```

デュアル スタック IPv4 および IPv6 DNS サーバ

DNS サーバには、A レコード（IPv4）または AAAA レコード（IPv6）のプライマリ DNS レコードがあります。A および AAAA レコードは、ドメイン名を特定の IP アドレス（IPv4 または IPv6）と関連付けます。

ACI ファブリックは、IPv4 で実行する信頼できるパブリック DNS サーバを使用するように設定できます。これらのサーバは、A レコード（IPv4）または AAAA レコード（IPv6）で解決および応答できます。

純粋な IPv6 環境では、システム管理者は IPv6 DNS サーバを使用する必要があります。IPv6 DNS サーバは、/etc/resolv.conf に追加することによって有効化されます。

より一般的な環境では、デュアルスタック IPv4 および IPv6 DNS サーバを使用します。デュアルスタックの場合、IPv4 と IPv6 の両方が /etc/resolv.conf にリストされます。ただし、デュアルスタック環境で、単純に IPv6 DNS サーバをリストに追加すると、DNS 解決の大きな遅延を引き起こす可能性があります。これは、デフォルトで IPv6 プロトコルが優先されるため、IPv4 DNS サーバに接続できないためです（/etc/resolv.conf で最初にリストされている場合）。この解決法は、IPv4 DNS サーバの前に IPv6 DNS サーバをリストすることです。また、IPv4 と IPv6 両方のルックアップで同一ソケットを使用できるようにするために、「options single-request-reopen」を追加します。

IPv6 DNS サーバが最初にリストされているデュアルスタック IPv4 および IPv6 DNS サーバの resolv.conf の例を次に示します。「single-request-reopen」オプションにも注意してください。

```
options single-request-reopen
nameserver 2001:4860:4680::8888
nameserver 2001:4860:4680::8844
nameserver 8.8.8.8
nameserver 8.8.4.4
```

デュアル スタック IPv4 および IPv6 環境

ACI ファブリックの管理ネットワークが IPv4 と IPv6 の両方をサポートする場合、Linux システムアプリケーション（glibc）では、getaddrinfo() が IPv6 を最初に返すため、IPv6 ネットワークをデフォルトで使用します。

ただし、特定の条件下では IPv4 アドレスが IPv6 アドレスよりも推奨されることがあります。Linux IPv6 スタックには、IPv6 にマッピングされた IPv4 アドレス（::ffff/96）を使用して、IPv6 アドレスとしてマッピングされた IPv4 アドレスを有効にする機能があります。これは、IPv6 対応アプリケーションが IPv4 と IPv6 両方を受け入れまたは接続するためにシングル ソケットのみ使用できるようにします。これは /etc/gai.conf の getaddrinfo() の glibc IPv6 選択項目によって制御されます。

/etc/hosts を使用する場合は glibc が複数のアドレスを返すようにするために、/etc/hosts ファイルに「multi on」を追加する必要があります。追加しないと、最初に一致したものだけを返す場合があります。

アプリケーションが IPv4 と IPv6 の両方が存在するかどうかを認識していない場合、異なるアドレス ファミリを使用するフォールバック試行が実行されないことがあります。このようなアプリケーションでは、フォールバックの実装が必要な場合があります。

REST API を使用した DNS プロバイダーと接続するための DNS サービス ポリシーの設定

はじめる前に

レイヤ 2 またはレイヤ 3 管理接続が設定されていることを確認します。

手順

ステップ 1 DNS サービス ポリシーを設定します。

例：

```
POST URL :
https://apic-IP/api/node/mo/uni/fabric.xml

<dnsProfile name="default">

    <dnsProv addr="172.21.157.5" preferred="yes"/>
    <dnsProv addr="172.21.157.6"/>

    <dnsDomain name="cisco.com" isDefault="yes"/>

    <dnsRsProfileToEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default"/>

</dnsProfile>
```

ステップ 2 アウトオブバンド管理テナント下で DNS ラベルを設定します。

例：

```
POST URL: https://apic-IP/api/node/mo/uni/tn-mgmt/ctx-oob.xml
<dnsLbl name="default" tag="yellow-green"/>
```

CLI を使用して、DNS プロファイルが設定されファブリック コントローラ スイッチに適用されているかを確認する

手順

ステップ 1 デフォルトの DNS プロファイルの設定を確認します。

例：

```
admin@apic1:~> cd /aci/fabric/fabric-policies/global-policies/dns-profiles/default
admin@apic1:default> cat summary
# dns-profile
name : default
description : added via CLI by tdeleon@cisco.com
ownerkey :
ownertag :

dns-providers:
address preferred
-------------- ---------
```

```
10.44.124.122 no
10.70.168.183 no
10.37.87.157 no
10.102.6.247 yes
dns-domains:
name default description
--------- ------- -----------
cisco.com yes
management-epg : tenants/mgmt/node-management-epgs/default/out-of-band/default
```

ステップ 2 DNS ラベルの設定を確認します。

例：

```
admin@apic1:default> cd
/aci/tenants/mgmt/networking/private-networks/oob/dns-profile-labels/default
admin@apic1:default> cat summary
# dns-lbl
name : default
description :
ownerkey :
ownertag :
tag : yellow-green
```

ステップ 3 適用された設定がファブリック コントローラで動作していることを確認します。

例：

```
admin@apic1:~> cat /etc/resolv.conf
# Generated by IFC
search cisco.com
nameserver 10.102.6.247
nameserver 10.44.124.122
nameserver 10.37.87.157
nameserver 10.70.168.183
admin@apic1:~> ping www.cisco.com
PING origin-www.cisco.com (72.163.4.161) 56(84) bytes of data.
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=1 ttl=238 time=35.4 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=2 ttl=238 time=29.0 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=3 ttl=238 time=29.2 ms
```

ステップ 4 適用された設定がリーフおよびスパイン スイッチで動作していることを確認します。

例：

```
leaf1# cat /etc/resolv.conf
search cisco.com
nameserver 10.102.6.247
nameserver 10.70.168.183
nameserver 10.44.124.122
nameserver 10.37.87.157
leaf1# cat /etc/dcos_resolv.conf
# DNS enabled
leaf1# ping www.cisco.com
PING origin-www.cisco.com (72.163.4.161): 56 data bytes
64 bytes from 72.163.4.161: icmp_seq=0 ttl=238 time=29.255 ms
64 bytes from 72.163.4.161: icmp_seq=1 ttl=238 time=29.212 ms
64 bytes from 72.163.4.161: icmp_seq=2 ttl=238 time=29.343 ms
```

テナントの外部接続の設定

スタティック ルートをアプリケーション セントリック インフラストラクチャ（ACI）ファブリック上の他のリーフ スイッチに配布する前に、マルチプロトコル BGP（MP-BGP）プロセスが最初

に動作していて、スパイン スイッチが BGP ルート リフレクタとして設定されている必要があります。

ACI ファブリックを外部ルーテッド ネットワークに統合するために、管理テナントのレイヤ 3 接続に対し Open Shortest Path First（OSPF）を設定できます。

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 URL: https://apic-ip/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 URL:
https://APIC-IP/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 URL:
https://APIC-IP/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。

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

- ルータ ID と論理インターフェイス プロファイルの IP アドレスが異なっていて重複していないことを確認します。
- 次の手順は、管理テナントの OSPF 外部ルーテッド ネットワークを作成するためのものです。テナントの OSPF 外部ルーテッド ネットワークを作成するには、テナントを選択し、テナント用の VRF を作成する必要があります。
- 詳細については、トランジット ルーティングに関する KB 記事も参照してください。

手順

管理テナントの OSPF 外部ルーテッド ネットワークを作成します。

例：

```
POST: https://192.0.20.123/api/mo/uni/tn-mgmt.xml

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

   <l3extOut name="RtdOut">
      <l3extRsL3DomAtt tDn="uni/l3dom-extdom"/>
      <l3extInstP name="extMgmt">
      </l3extInstP>
      <l3extLNodeP name="borderLeaf">
         <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-101" rtrId="10.10.10.10"/>
         <l3extRsNodeL3OutAtt tDn="topology/pod-1/node-102" rtrId="10.10.10.11"/>
         <l3extLIfP name='portProfile'>
           <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
ifInstT='l3-port' addr="192.168.62.1/24"/>
           <l3extRsPathL3OutAtt tDn="topology/pod-1/paths-102/pathep-[eth1/40]"
ifInstT='l3-port' addr="192.168.62.5/24"/>
            <ospfIfP/>
                                 </l3extLIfP>
      </l3extLNodeP>
      <l3extRsEctx tnFvCtxName="inb"/>
      <ospfExtP areaId="57" />
   </l3extOut>
</fvTenant>
```

アプリケーション ポリシーの展開

Three-Tier アプリケーションの展開

フィルタは、フィルタを含む契約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。契約には、複数のサブジェクトを含めることができます。サブジェクトは、単方向または双方向のフィルタを実現するために使用できます。単方向フィルタは、コンシューマからプロバイダー（IN）のフィルタまたはプロバイダーからコンシューマ（OUT）のフィルタのどちらか一方向に使用されるフィルタです。双方向フィルタは、両方の方向で使用される同一フィルタです。これは、再帰的ではありません。

契約は、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。これらのポリシーは、アプリケーション層間の通信を指定するルールです。契約が EPG に付属していない場合、EPG 間の通信はデフォルトでディセーブルになります。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。

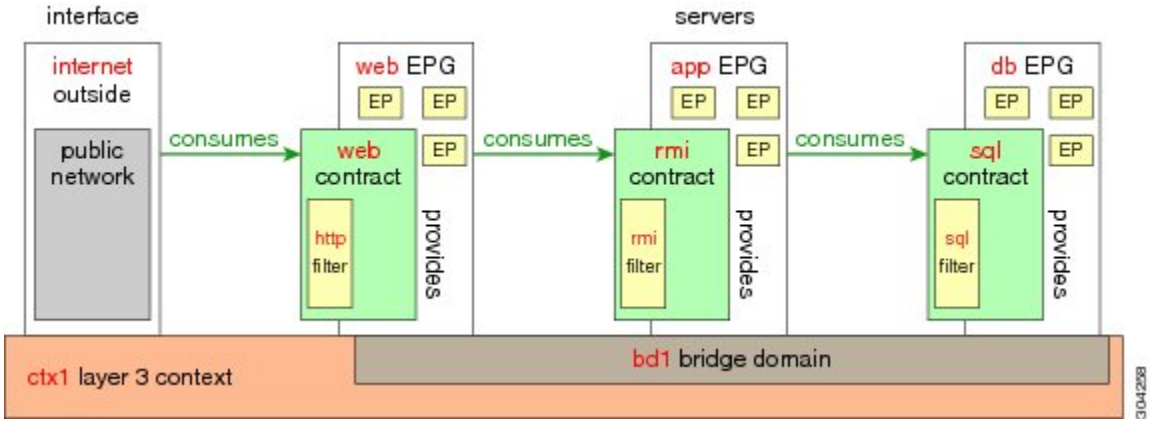
アプリケーション プロファイルでは、APIC がその後ネットワークおよびデータセンターのインフラストラクチャで自動的にレンダリングするアプリケーション要件をモデル化することができます。アプリケーション プロファイルでは、管理者がインフラストラクチャの構成要素ではなくアプリケーションの観点から、リソース プールにアプローチすることができます。アプリケーション プロファイルは、互いに論理的に関連する EPG を保持するコンテナです。EPG は同じア

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 契約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 契約を消費して、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 *3*：*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
エントリ名	Dport-80 Dport-443
Ethertype	IP

パラメータ名	http のフィルタ
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	提供される契約	消費される契約
Web	Web	rmi
app	rmi	sql
db	sql	--

REST API を使用したアプリケーション ポリシーの展開

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 XML API を使用してアプリケーションを展開するには、次の HTTP POST メッセージを送信します。

例：

```
POST
https://192.0.20.123/api/mo/uni/tn-ExampleCorp.xml
```

ステップ 2 次の XML 構造を POST メッセージの本文に含めます。

例：

```
<fvTenant name="ExampleCorp">

    <fvAp name="OnlineStore">
        <fvAEPg name="web">
            <fvRsBd tnFvBDName="bd1"/>
            <fvRsCons tnVzBrCPName="rmi"/>
            <fvRsProv tnVzBrCPName="web"/>
            <fvRsDomAtt tDn="uni/vmmp-VMware/dom-datacenter"/>
        </fvAEPg>

        <fvAEPg name="db">
            <fvRsBd tnFvBDName="bd1"/>
            <fvRsProv tnVzBrCPName="sql"/>
            <fvRsDomAtt tDn="uni/vmmp-VMware/dom-datacenter"/>
        </fvAEPg>

        <fvAEPg name="app">
            <fvRsBd tnFvBDName="bd1"/>
            <fvRsProv tnVzBrCPName="rmi"/>
            <fvRsCons tnVzBrCPName="sql"/>
            <fvRsDomAtt tDn="uni/vmmp-VMware/dom-datacenter"/>
        </fvAEPg>
    </fvAp>

<vzFilter name="http" >
<vzEntry dFromPort="80" name="DPort-80" prot="tcp" etherT="ip"/>
<vzEntry dFromPort="443" name="DPort-443" prot="tcp" etherT="ip"/>
</vzFilter>
<vzFilter name="rmi" >
<vzEntry dFromPort="1099" name="DPort-1099" prot="tcp" etherT="ip"/>
</vzFilter>
<vzFilter name="sql">
<vzEntry dFromPort="1521" name="DPort-1521" prot="tcp" etherT="ip"/>
</vzFilter>
    <vzBrCP name="web">
        <vzSubj name="web">
            <vzRsSubjFiltAtt tnVzFilterName="http"/>
        </vzSubj>
    </vzBrCP>

    <vzBrCP name="rmi">
        <vzSubj name="rmi">
            <vzRsSubjFiltAtt tnVzFilterName="rmi"/>
        </vzSubj>
```

```
    </vzBrCP>

    <vzBrCP name="sql">
        <vzSubj name="sql">
           <vzRsSubjFiltAtt tnVzFilterName="sql"/>
        </vzSubj>
    </vzBrCP>
</fvTenant>
```

XML 構造の最初の行は、ExampleCorp という名前のテナントを変更するかまたは必要に応じて作成します。

```
<fvTenant name="ExampleCorp">
```

次の行は、OnlineStore という名前のアプリケーション ネットワーク プロファイルを作成します。

```
<fvAp name="OnlineStore">
```

アプリケーション ネットワーク プロファイル内の要素は、3 つのエンドポイント グループを作成します（3 台のサーバそれぞれに 1 つずつ）。次の行は、web という名前のエンドポイント グループを作成し、bd1 という名前の既存のブリッジ ドメインに関連付けます。このエンドポイント グループは、rmi という名前のバイナリ契約で許可されたトラフィックのコンシューマまたは宛先であり、web という名前のバイナリ契約で許可されたトラフィックのプロバイダーまたは送信元です。エンドポイント グループは、datacenter という名前の VMM ドメインに関連付けられます。

```
<fvAEPg name="web">
    <fvRsBd tnFvBDName="bd1"/>
    <fvRsCons tnVzBrCPName="rmi"/>
    <fvRsProv tnVzBrCPName="web"/>
    <fvRsDomAtt tDn="uni/vmmp-VMware/dom-datacenter"/>
</fvAEPg>
```

残りの 2 つのエンドポイント グループは、アプリケーション サーバとデータベース サーバに対し、同様の方法で作成されます。

次の行は、TCP トラフィックのタイプ HTTP（ポート 80）および HTTPS（ポート 443）を指定する http という名前のトラフィック フィルタを定義します。

```
<vzFilter name="http" >
<vzEntry dFromPort="80" name="DPort-80" prot="tcp" etherT="ip"/>
<vzEntry dFromPort="443" name="DPort-443" prot="tcp" etherT="ip"/>
</vzFilter>
```

残りの2つのフィルタは、アプリケーションのデータおよびデータベース（sql）のデータに対し、同様の方法で作成されます。

次の行は、http という名前のフィルタを組み込む web という名前のバイナリ契約を作成します。

```
<vzBrCP name="web">
    <vzSubj name="web">
        <vzRsSubjFiltAtt tnVzFilterName="http"/>
    </vzSubj>
</vzBrCP>
```

残りの 2 つの契約は、rmi および sql のデータ プロトコルに対し、同様の方法で作成されます。

最後の行は、構造を閉じます。

```
</fvTenant>
```