Cisco APIC レイヤ 2 ネットワーク設定ガイド、リリース 5.2(x)

初版：2021 年 6 月 4 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

Trademarks

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS REFERENCED IN THIS DOCUMENTATION ARE SUBJECT TO CHANGE WITHOUT NOTICE. EXCEPT AS MAY OTHERWISE BE AGREED BY CISCO IN WRITING, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENTATION ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

The Cisco End User License Agreement and any supplemental license terms govern your use of any Cisco software, including this product documentation, and are located at: http://www.cisco.com/go/softwareterms.Cisco product warranty information is available at http://www.cisco.com/go/warranty. US Federal Communications Commission Notices are found here http://www.cisco.com/c/en/us/products/us-fcc-notice.html.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any products and features described herein as in development or available at a future date remain in varying stages of development and will be offered on a when-and if-available basis. Any such product or feature roadmaps are subject to change at the sole discretion of Cisco and Cisco will have no liability for delay in the delivery or failure to deliver any products or feature roadmap items that may be set forth in this document.

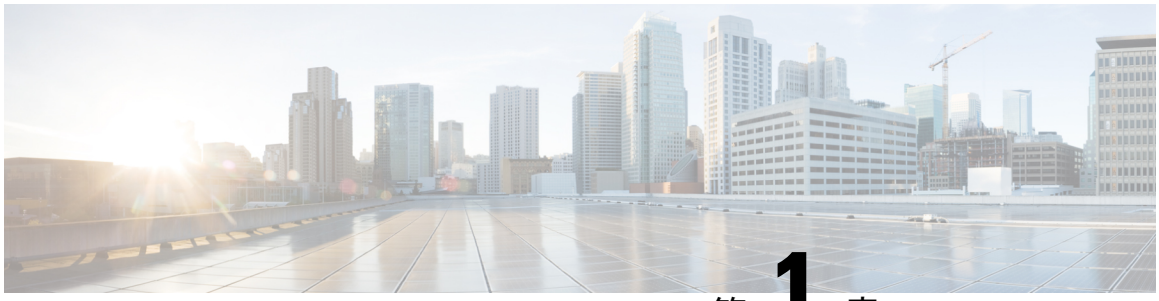
Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com go trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

Trademarks

目 次



第 1 章

新機能および変更された機能

この章は、次の内容で構成されています。

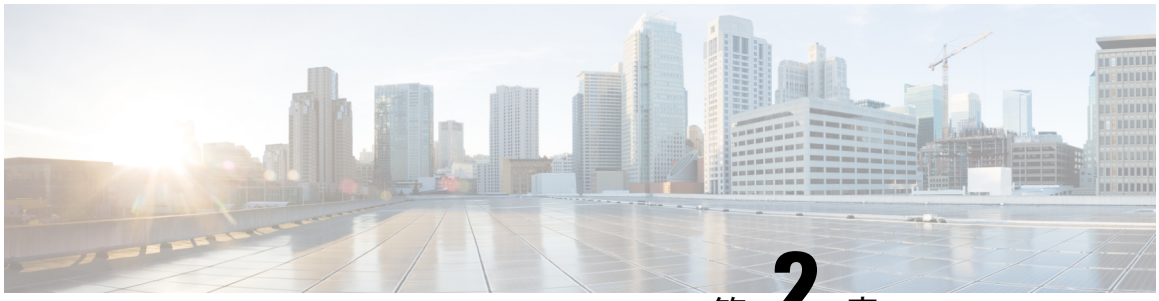
新機能および変更された機能に関する情報

次の表に、本リリースに関するこのガイドでの重要な変更点の概要を示します。ただし、今リリースまでのガイドにおける変更点や新機能の一部は表に記載されていません。

表 *1 : Cisco APIC 6.0(1)* の新機能と変更された機能に関する情報

特長	説明	参照先

新機能および変更された機能に関する情報



第 2 章

Cisco ACI 転送

この章は、次の内容で構成されています。

ACI ファブリックは現代のデータ センター トラフィック フローを最適化する

Cisco ACI アーキテクチャは、従来のデータセンター設計から来る制限を解放して、最新のデータセンターで増大する East-West トラフィックの需要に対応します。

今日のアプリケーション設計は、データセンターのアクセス レイヤを通る、サーバ間の East-West トラフィックを増大させています。このシフトを促進しているアプリケーションには、Hadoop のようなビッグデータの分散処理の設計 、VMware vMotion のようなライブの仮想マシンまたはワークロードの移行、サーバのクラスタ リング、および多層アプリケーションなどが含まれます。

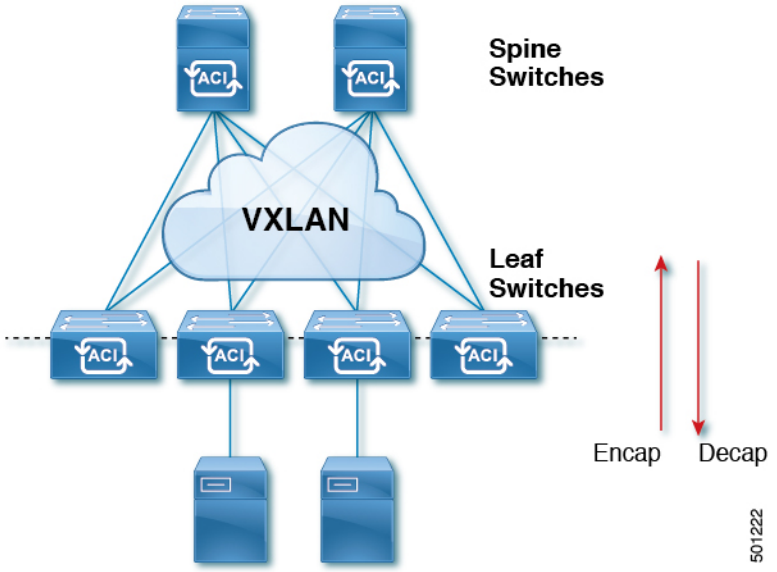
North-South トラフィックは、コア、集約、およびアクセス レイヤ、またはコラプスト コアとアクセス レイヤが重要となる、従来型のデータセンター設計を推進します。クライアントデータはWAN またはインターネットで受信され、サーバの処理を受けた後、データセンターを出ます。このような方式のため、WAN またはインターネットの帯域幅の制限により、データセンターのハードウェアは過剰設備になりがちです。ただし、スパニング ツリー プロトコルが、ループをブロックするために要求されます。これは、ブロックされたリンクにより利用可能な帯域幅を制限し、トラフィックが準最適なパスを通るように強制する可能性があります。

従来のデータセンター設計においては、、IEEE 802.1Q VLAN がレイヤ 2 境界の論理セグメンテーションまたはまたはブロードキャスト ドメインを提供します。ただし、ネットワーク リンクの VLAN の使用は効率的ではありません。データセンター ネットワークでデバイスの配置要件は柔軟性に欠け、VLAN の最大値である 4094 の VLAN が制限となり得ます。IT 部門と

クラウドプロバイダが大規模なマルチテナントデータセンターを構築するようになるにつれ、VLAN の制限は問題となりつつあります。

スパイン リーフ アーキテクチャは、これらの制限に対処します。ACI ファブリックは、外界からは、ブリッジングとルーティングが可能な単一のスイッチに見えます。レイヤ 3 のルーティングをアクセス レイヤに移動すると、最新のアプリケーションが必要としている、レイヤ 2 の到達可能性が制限されます。仮想マシン ワークロード モビリティや一部のクラスタ リングのソフトウェアのようなアプリケーションは、送信元と宛先のサーバ間がレイヤ 2 で隣接していることを必要とします。アクセス レイヤでルーティングを行えば、トランク ダウンされた同じ VLAN の同じアクセス スイッチに接続したサーバだけが、レイヤ 2 で隣接します。ACI では、VXLAN が、基盤となるレイヤ 3 ネットワーク インフラストラクチャからレイヤ 2 のドメインを切り離すことにより、このジレンマを解決します。

図 *1: ACI* ファブリック



トラフィックがファブリックに入ると、ACIがカプセル化してポリシーを適用し、必要に応じてスパイン スイッチ (最大 2 ホップ) によってファブリックを通過させ、ファブリックを出るときにカプセル化を解除します。ファブリック内では、ACIはエンドポイント間通信でのすべての転送について、Intermediate System-to-Intermediate System プロトコル (IS-IS) および Council of Oracle Protocol (COOP) を使用します。これにより、すべての ACI リンクがアクティブで、ファブリック内での等コストマルチパス (ECMP) 転送と高速再コンバージョンが可能になります。ファブリック内と、ファブリックの外部のルータ内でのソフトウェア定義ネットワーク間のルーティング情報を伝播するために、 ACI はマルチプロトコル Border Gateway Protocol (MP-BGP) を使用します。

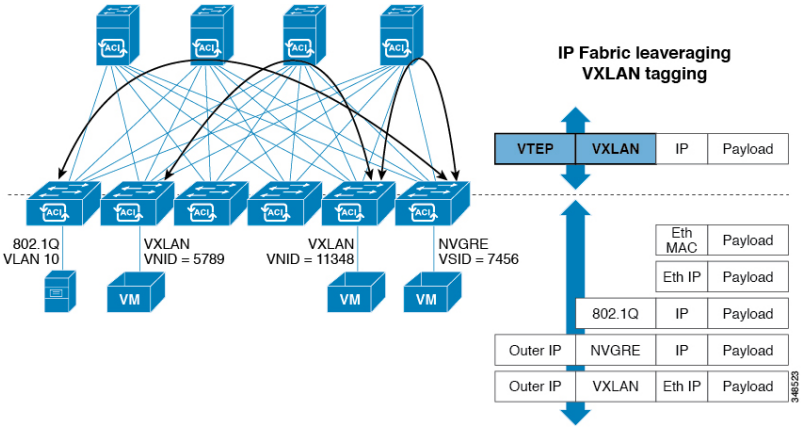
ACI で VXLAN

VXLAN は、レイヤ 2 オーバーレイの論理ネットワークを構築するレイヤ 3 のインフラストラクチャ上でレイヤ 2 のセグメントを拡張する業界標準プロトコルです。ACIインフラストラク

チャ レイヤ 2 ドメインが隔離ブロードキャストと障害ブリッジ ドメインをオーバーレイ内に存在します。このアプローチは大きすぎる、障害ドメインの作成のリスクなしで大きくなるデータセンター ネットワークを使用できます。

すべてのトラフィック、ACIファブリックはVXLANパケットとして正規化されます。入力でACI VXLANパケットで外部VLAN、VXLAN、およびNVGREパケットをカプセル化します。次の図は、ACIカプセル化の正規化を示します。

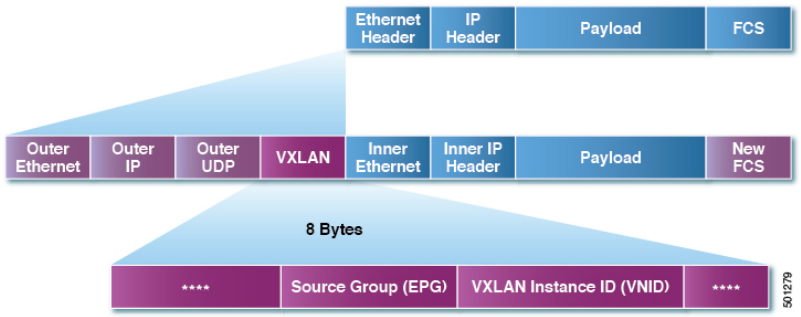
図 *2 : ACI* カプセル化の正規化



ACI ファブリックでの転送は、カプセル化のタイプまたはカプセル化のオーバーレイ ネットワークによって制限または制約されません。ACI ブリッジ ドメインのフォワーディング ポリシーは、必要な場合に標準の VLAN 動作を提供するために定義できます。

ファブリック内のすべてのパケットにACIポリシー属性が含まれているため、ACIは完全に分散された方法でポリシーを一貫して適用できます。ACIにより、アプリケーション ポリシーのEPG IDが転送から分離されます。次の図に示すように、ACI VXLANヘッダーは、ファブリック内のアプリケーション ポリシーを特定します。

図 *3 : ACI VXLAN* のパケット形式



ACIVXLAN パケットには、レイヤ 2 の MAC アドレスとレイヤ 3 IP アドレスの送信元と宛先フィールド、ファブリック内の効率的なと拡張性の転送を有効にします。ACIVXLAN パケット ヘッダーの送信元グループ フィールドは、パケットが属するアプリケーション ポリシー エンドポイント グループ (EPG) を特定します。VXLAN インスタンス ID (VNID) は、テナントの仮想ルーティングおよび転送 (VRF) ドメイン ファブリック内で、パケットの転送を有効にし

ます。VXLAN ヘッダーで 24 ビット VNID フィールドでは、同じネットワークで一意レイヤ 2 のセグメントを最大 16 個の拡張アドレス空間を提供します。この拡張アドレス空間は、大規模なマルチ テナント データセンターを構築する柔軟性 IT 部門とクラウド プロバイダーを提供します。

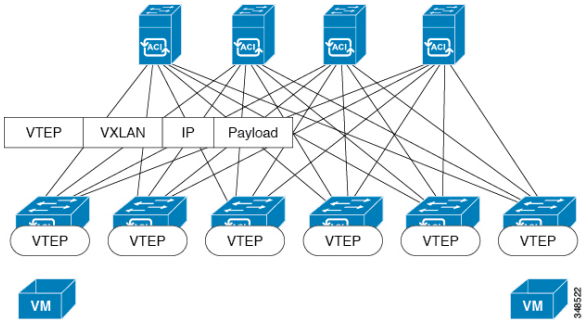
VXLAN を有効に ACI ファブリック全体にわたってスケールでの仮想ネットワーク インフラストラクチャのレイヤ 3 のアンダーレイ レイヤ 2 を展開します。アプリケーション エンドポイント ホスト柔軟に配置できます、アンダーレイ インフラストラクチャのレイヤ 3 バウンダリのリスクなしでデータセンター ネットワーク間をオーバーレイ ネットワーク、VXLAN でレイヤ 2 の隣接関係を維持します。

サブネット間のテナント トラフィックの転送を促進するレイヤ 3 VNID

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

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

図 *4: ACI* によって切り離された *ID* と場所



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

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

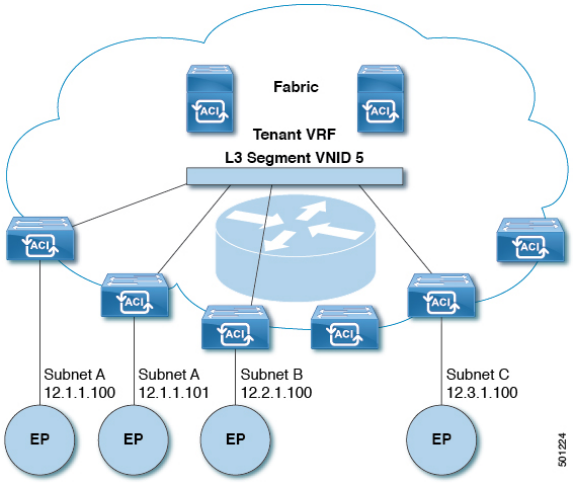
IP インターフェイスには一意の IP アドレスがあります。これは、インフラストラクチャ VLAN として知られる、転送 IP ネットワーク上の VTEP を識別します。VTEP デバイスはこの IP アドレスを使用してイーサネット フレームをカプセル化し、カプセル化されたパケットを、IP インターフェイスを介して転送ネットワークへ送信します。また、VTEP デバイスはリモート VTEP で VXLAN セグメントを検出し、IP インターフェイスを介してリモートの MAC Address-to-VTEP マッピングについて学習します。

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

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

次の図は、テナント内のルーティングがどのように行われるかを示します。

図 *5 : ACI* のサブネット間のテナント トラフィックを転送するレイヤ *3 VNID*



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

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

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

（注） Cisco ACI は IP フラグメンテーションをサポートしていません。したがって、外部ルータへのレイヤ 3 Outside（L3Out）接続、または Inter-Pod Network（IPN）を介したマルチポッド接続を設定する場合は、インターフェイス MTU がリンクの両端で適切に設定することを推奨します。Cisco ACI、Cisco NX-OS、および Cisco IOS などの一部のプラットフォームでは、設定可能な MTU 値はイーサネット ヘッダー (一致する IP MTU、14-18 イーサネット ヘッダー サイズを除く) を考慮していません。また、IOS XR などの他のプラットフォームには、設定された MTU 値にイーサネット ヘッダーが含まれています。設定された値が 9000 の場合、Cisco ACI、Cisco NX-OS および Cisco IOS の最大 IP パケット サイズは 9000 バイトになりますが、IOS-XR のタグなしインターフェイスの最大 IP パケットサイズは 8986 バイトになります。

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

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

スパニング ツリー プロトコル BPDU の送信

スパニング ツリープロトコル（STP）を実行している 2 つ以上のスイッチが EPG の Cisco Application Centric Infrastructure（ACI）に接続されており、スタティックポートが次のように割り当てられている場合：

- EPG で静的に割り当てられたすべてのポートは、タグなしでアクセスされます。STP ブリッジ プロトコル データ ユニット（BPDU）はタグなしで送受信されます。
- スタティックに割り当てられたトランクポートとスタティックに割り当てられたアクセスタグなしポートが混在している場合：トランク ポートで受信された STP BPDU は、`dot1q` タグ付きのアクセス タグなしポートに送信されます。したがって、アクセス ポートは不整合状態になります。
- EPG で静的に割り当てられたトランク ポートと静的に割り当てられたアクセス ポートの組み合わせの場合、Cisco ACI は `dot1q` タグを使用して STP BPDU を送信し、アクセスポートは `802.1p` アクセスを使用します。

  この場合、タグ付き STP パケットを受信して処理するには、レイヤ 2 スイッチで `802.1p` アクセスを使用する必要があります。

  `802.1p` がレイヤ 2 スイッチで許可されていない場合は、トランク ポート アクセスを使用します。

- Cisco ACI は全二重ハブとして機能し、BPDU が受信されたカプセル化 VLAN に関連付けられた VxLAN VNID 内でスパニング ツリー BPDU をフラッディングします。Cisco ACI は全二重メディアであるため、高速スパニングツリー プロトコル（RSTP）または高速 VLAN 単位スパニングツリー（RPVST）のバージョンを実行する外部スイッチは、デフォルトでポイントツーポイント リンク タイプになります。その結果、STP を実行し、同じカプセル化 VLAN および EPG VNID に接続する 2 つ以上の外部スイッチがある場合、コンバージェンスと不安定性の問題を回避するために、外部スイッチインターフェイスでリンク タイプを「共有」に設定する必要があります。これらの問題は、スイッチがこのカプセル化に接続されているすべてのブリッジ（または STP 対応スイッチ）から BPDU を受信するために発生する可能性があります。

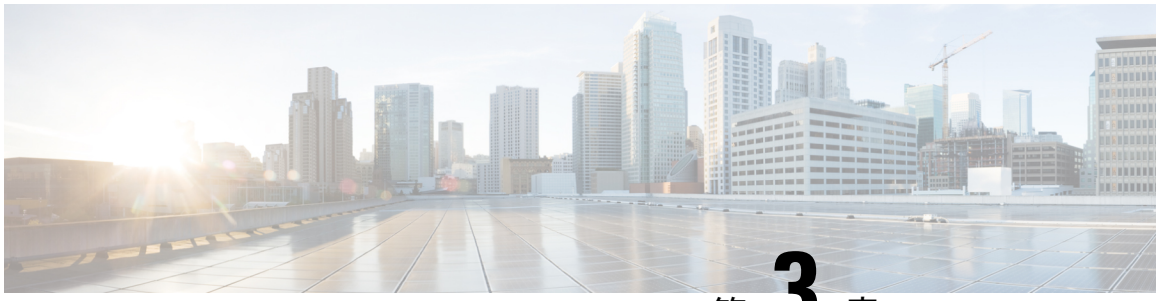
スパニング ツリー BPDU は、EPG パスで定義された特定の VLAN ID 内でフラッディングされます。この VLAN は、リーフスイッチでは `FD_VLAN` と呼ばれます。リーフスイッチ間でFD_VLAN 内のトラフィックを転送するために、Cisco ACI は、`fabric_encap` と呼ばれる VXLAN VNID を割り当てます。`fabric_encap` は、VLAN プールに属する数値ベース識別子を取得し、VLAN プールから割り当てられた VLAN ID のインデックス値を追加することによって取得されます。たとえば、VXLAN VNID 9000 は、VLAN 範囲 10 ー 20 を含む VLAN プール A に割り当てられます。VLAN プール A の VLAN 10 には VNID 9000 が割り当てられ、VLAN 11 には VNID 9001 が割り当てられます。

このため、2 つの異なる EPG が同じ VLAN ID を使用しており、同じ VLAN プールからその VLAN ID を割り当てている場合は、異なるファブリックスイッチ上の 2 つの EPG に対して同じ `fabric_encap` VNID を導出できます。これにより、2 つの EPG 間でスパニング ツリー BPDU が意図せずフラッディングされる可能性があります。

この動作を回避するには、物理ドメインなどの個別の VLAN プールを持つ異なるドメインを各 EPG に割り当て、特定の VLAN ID を個別の VLAN プールから割り当てます。これにより、ベース ID が異なるようになるので、`fabric_encap` VNIDの重複が防止されます。

`fabric_encap` の値は、次のコマンドを使用して確認できます。また、特定の 802.1q VLAN ID のリーフ スイッチの出力の「Fabric_enc」列でも確認できます。

```
vsh_lc -c "show system internal eltmc info vlan br"
```



第 3 章

レイヤ 2 ネットワーク設定の前提条件

レイヤ 2 の前提条件

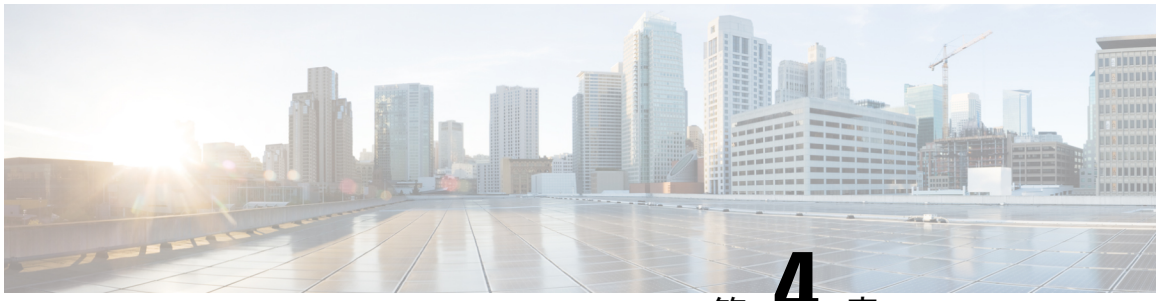
このガイドで説明するタスクを実行する前に、以下の事柄を完了しておいてください。

- ACI ファブリックをインストールして、APIC コントローラがオンラインになっており、APIC クラスタが形成されていて健全な状態であることを確認します。詳細については、『*Cisco APIC Getting Started Guide, Release 2.x*』を参照してください。
- レイヤ 2 ネットワークを設定する管理者のために、ファブリックの管理者アカウントを作成します。 詳細については、『*Cisco APIC Basic Configuration Guide*』の「*User Access, Authentication, and Accounting*」および「*Management*」の章を参照してください。
- ACI ファブリックにターゲット リーフ スイッチをインストールし、登録します。詳細については、『*Cisco APIC Getting Started Guide, Release 2.x*』を参照してください。

  仮想スイッチのインストールと登録の詳細については、『*Cisco ACI Virtualization Guide*』を参照してください。
- レイヤ 2 ネットワークを利用するテナント、VRF、および EPG を (アプリケーション プロファイルやコントラクトとともに) 設定します。詳細については、『*Cisco APIC Basic Configuration Guide*』の「*Basic User Tenant Configuration*」の章を参照してください。

⚠

注意 ファブリックのリーフ スイッチとスパイン スイッチの間に 1 ギガビット イーサネット (GE) または 10GE リンクを設置すると、帯域幅が不十分なために、パケットが転送されずにドロップされる可能性があります。これを避けるためには、リーフ スイッチとスパイン スイッチの間で 40GE または 100GE リンクを使用してください。



第 4 章

ネットワーク ドメイン

この章は、次の内容で構成されています。

ネットワーク ドメイン

ファブリック管理者は、ポート、プロトコル、VLANプール、およびカプセル化を設定するドメインポリシーを作成します。これらのポリシーは、単一テナント専用にすることも、共有することもできます。ファブリック管理者がACIファブリック内にドメインを設定すると、テナント管理者はテナント エンドポイント グループ（EPG）をドメインに関連付けることができます。

以下のネットワーク ドメイン プロファイルを設定できます。

- VMM ドメイン プロファイル（vmmDomP）は、仮想マシンのハイパーバイザ統合のために必要です。
- 物理ドメイン プロファイル（physDomP）は、ベア メタル サーバ接続と管理アクセスに使用します。
- ブリッジド外部ネットワーク ドメイン プロファイル（l2extDomP）は通常、ACI ファブリックのリーフ スイッチにブリッジド外部ネットワーク トランク スイッチを接続するために使用されます。
- ルーテッド外部ネットワーク ドメイン プロファイル（l3extDomP）は、ACI ファブリックのリーフ スイッチにルータを接続するために使用されます。
- ファイバチャネル ドメイン プロファイル(fcDomP)は、ファイバチャネルのVLANとVSANを接続するために使用されます。

ドメインは VLAN プールに関連付けられるように設定されます。その後、EPG は、ドメインに関連付けられている VLAN を使用するように設定されます。

（注） EPG ポートと VLAN の設定は、EPG が関連付けられているドメイン インフラストラクチャ設定で指定されている設定に一致する必要があります。一致しない場合、APIC でエラーが発生します。そのようなエラーが発生した場合は、ドメインインフラストラクチャ設定が EPG ポートと VLAN の設定に一致していることを確認してください。

関連資料

レイヤ 3 のネットワーキングの詳細については、『*Cisco APIC Layer 3 Networking Configuration Guide*』を参照してください。

VMM ドメインの設定の詳細については、『*Cisco ACI Virtualization Guide*』の「*Cisco ACI Virtual Machine Networking*」を参照してください。

ブリッジ ドメイン

ブリッジ ドメインについて

ブリッジ ドメイン (BD) はファブリック内のレイヤ 2 フォワーディングの構造を表します。1 つ以上のエンドポイント グループ（EPG）を 1 つのブリッジ ドメインまたはサブネットと関連付けることができます。ブリッジ ドメインには1つまたは複数のサブネットを関連付けることができます。1つまたは複数のブリッジ ドメインの組み合わせによってテナント ネットワークを形成します。2 つの EPG の間でのサービス機能を挿入するときには、それらの EPG は個別 BD の中になければなりません。2 つの EPG の間でのサービス機能を使用するには、これらの EPG は分離している必要があります。このことは、レイヤ 2 およびレイヤ 3 に基づく、レガシー サービス 挿入に従います。

VMM ドメイン

Virtual Machine Manager ドメインの主要コンポーネント

ACI ファブリック Virtual Machine Manager（VMM）ドメインにより、管理者は仮想マシン コントローラの接続ポリシーを設定できます。ACI VMM ドメイン ポリシーの基本的なコンポーネントは次のとおりです。

- **Virtual Machine Manager ドメイン プロファイル**：同様のネットワーキング ポリシー要件を持つ VM コントローラをグループ化します。たとえば、VM コントローラは VLAN プールとアプリケーション エンドポイント グループ（EPG）を共有できます。APIC はコントローラと通信し、のちに仮想ワークロードに適用されるポート グループなどのネットワー

ク設定を公開します。VMM ドメイン プロファイルには、次の基本コンポーネントが含まれます。

- **クレデンシャル**：有効な VM コントローラ ユーザ クレデンシャルを APIC VMM ドメインと関連付けます。
- **コントローラ**：ポリシーの適用ドメインの一部である VM コントローラへの接続方法を指定します。たとえば、コントローラは VMM ドメインの一部である VMware vCenter への接続を指定します。

（注） 1 つのドメインに VM コントローラの複数のインスタンスを含めることができますが、それらは同じベンダーのものである必要があります（VMware または Microsoft など）。

- **EPG の関連付け**：エンドポイント グループにより、エンドポイント間の接続と可視性が VMM ドメイン ポリシーの範囲内に規制されます。VMM ドメイン EPG は次のように動作します。
  - APIC は、これらの EPG をポート グループとして VM コントローラにプッシュします。
  - 1 つの EPG は、複数の VMM ドメインをカバーでき、1 つの VMM ドメインには複数の EPG を含めることができます。
- **接続可能エンティティ プロファイルの関連付け**：VMM ドメインを物理ネットワーク インフラストラクチャと関連付けます。接続可能エンティティ プロファイル（AEP）は、多数のリーフ スイッチ ポートで VM コントローラ ポリシーを展開するための、ネットワーク インターフェイス テンプレートです。AEP は、使用できるスイッチやポートおよびその設定方法を指定します。
- **VLAN プールの関連付け**：VLAN プールは、VMM ドメインが消費する VLAN カプセル化に使用する VLAN ID または範囲を指定します。

Virtual Machine Manager のドメイン

APIC VMM ドメイン プロファイルは、VMM ドメインを定義するポリシーです。VMM ドメイン ポリシーは APIC で作成され、リーフ スイッチにプッシュされます。

VMM ドメインは以下を提供します。

- 複数の VM コントローラ プラットフォームに対してスケーラブルな耐障害性サポートを可能にする、ACI ファブリックの共通レイヤ
- ACI ファブリック内の複数のテナントに対する VMM サポート

VMM ドメインには、VMware vCenter や Microsoft SCVMM Manager などの VM コントローラと、VM コントローラと対話するための ACI API に必要なクレデンシャルが含まれます。VMM ドメインはドメイン内の VM モビリティを実現できますが、ドメイン間は実現できません。単一の VMM ドメイン コントローラに VM コントローラの複数のインスタンスを含めることはできますが、同じタイプである必要があります。たとえば、1 つの VMM ドメインに、それぞれが複数の VM を実行する複数のコントローラを管理する多くの VMware vCenter を含めることができますが、SCVMM Manager も含めることはできません。VMM ドメインはコントローラ要素（pNIC、vNIC、VM 名など）をインベントリに含め、コントローラにポリシーをプッシュして、ポート グループなどの必要な要素を作成します。ACI VMM ドメインは VM モビリティなどのコントローラ イベントを監視し、状況に応じて応答します。

物理ドメイン設定

物理ドメインの設定

物理ドメインは、特定の VLAN ネームスペースが使用される範囲を制御します。物理ドメインと関連付けられた VLAN のネームスペースは、仮想サーバからのポート グループのスタティック マッピングに使用できますが、非仮想サーバを対象としています。物理デバイス タイプの物理ドメインを設定できます。

始める前に

- テナントを設定します。

手順

ステップ 1 メニュー バーで [Fabric] をクリックします。

ステップ 2 サブメニュー バーで [External Access Policies] をクリックします。

ステップ 3 [Navigation] ウィンドウで、[Physical and External Domains] を展開し、[Physical Domains] をクリックします。

ステップ 4 [Actions] ドロップダウン リストで [Create Physical Domain] を選択します。[Create Physical Domain] ダイアログボックスが表示されます。

ステップ 5 次のフィールドに入力します。

名前	説明
名前（**Name**）	物理ドメイン プロファイルの名前。
Associate Attachable Entity Profiles	このドメインに関連付けられる、アタッチ可能なエンティティ プロファイルを選択します。
VLAN Pool	物理ドメインが使用する VLAN プール。VLAN プールは、APIC によってこの物理ドメインを使用しているサービス グループ テンプ

名前	説明
	レートに対して割り当てられる VLAN のプールの範囲を指定します。[Dynamic] または [Static] の割り当てをクリックします。

ステップ 6 （任意）AAA のセキュリティ ドメインを追加し、[Select] チェック ボックスをオンにします。

ステップ 7 [送信（Submit）] をクリックします。

REST API を使用した物理ドメインの設定

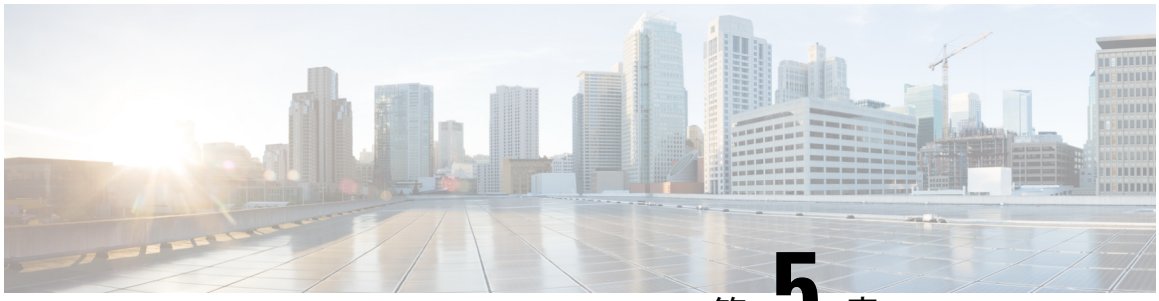
物理ドメインは、VLAN プールとアクセス エンティティ プロファイル（AEP）間のリンクとして動作します。ドメインはファブリックの設定をテナントの設定に結びつけます。ドメインを EPG に関連付けるのはテナント管理者であり、ドメインが作成されるのは [Fabric] タブだからです。この順序で設定すると、プロファイル名と VLAN プールのみが設定されます。

手順

次の例のような XML を POST を送信することによって物理ドメインを設定します。

例：

```
<physDomP dn="uni/phys-bsprint-PHY" lcOwn="local" modTs="2015-02-23T16:13:21.906-08:00"

   monPolDn="uni/fabric/monfab-default" name="bsprint-PHY" ownerKey="" ownerTag=""
status="" uid="8131">
      <infraRsVlanNs childAction="" forceResolve="no" lcOwn="local"
modTs="2015-02-23T16:13:22.065-08:00"
         monPolDn="uni/fabric/monfab-default" rType="mo" rn="rsvlanNs" state="formed"
stateQual="none"
        status="" tCl="fvnsVlanInstP" tDn="uni/infra/vlanns-[bsprint-vlan-pool]-static"
 tType="mo" uid="8131"/>
      <infraRsVlanNsDef forceResolve="no" lcOwn="local"
modTs="2015-02-23T16:13:22.065-08:00" rType="mo"
         rn="rsvlanNsDef" state="formed" stateQual="none" status="" tCl="fvnsAInstP"
         tDn="uni/infra/vlanns-[bsprint-vlan-pool]-static" tType="mo"/>
      <infraRtDomP lcOwn="local" modTs="2015-02-23T16:13:52.945-08:00"
rn="rtdomP-[uni/infra/attentp-bsprint-AEP]"
         status="" tCl="infraAttEntityP" tDn="uni/infra/attentp-bsprint-AEP"/>
</physDomP>
```



第 5 章

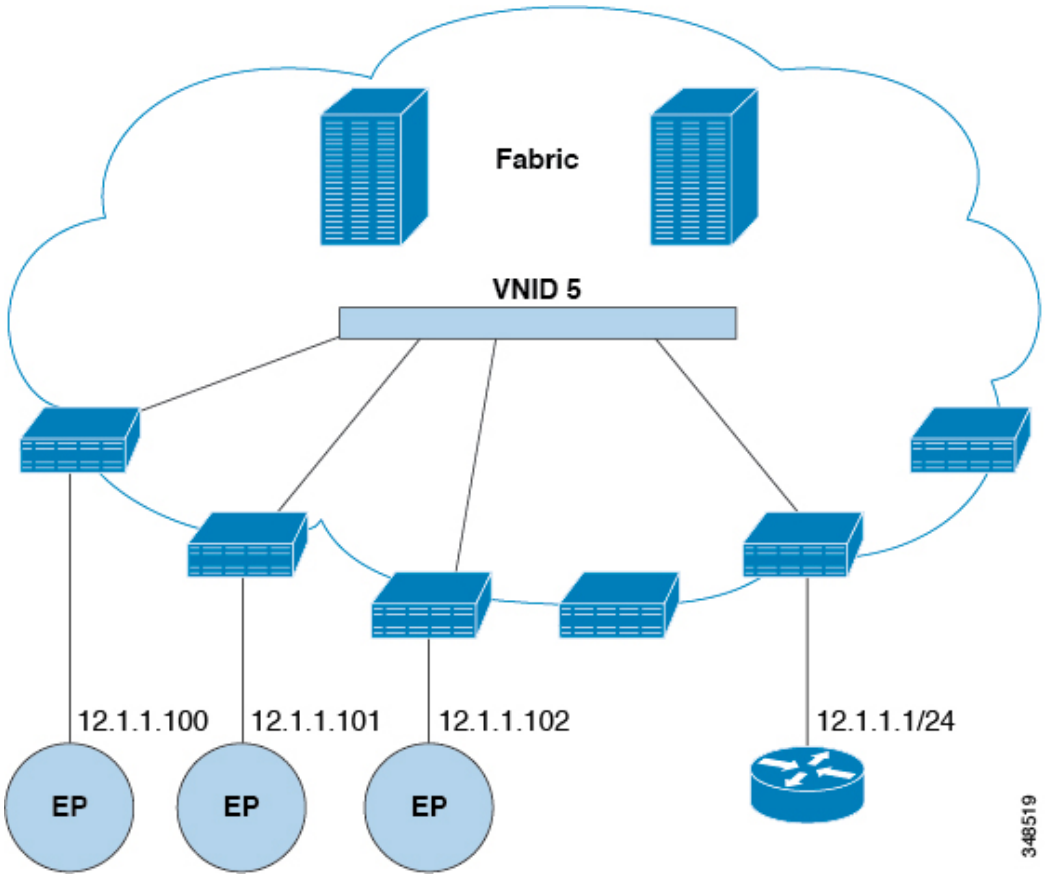
ブリッジング

この章は、次の内容で構成されています。

外部ルータへのブリッジドインターフェイス

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

図 *6:* ブリッジド外部ルータ

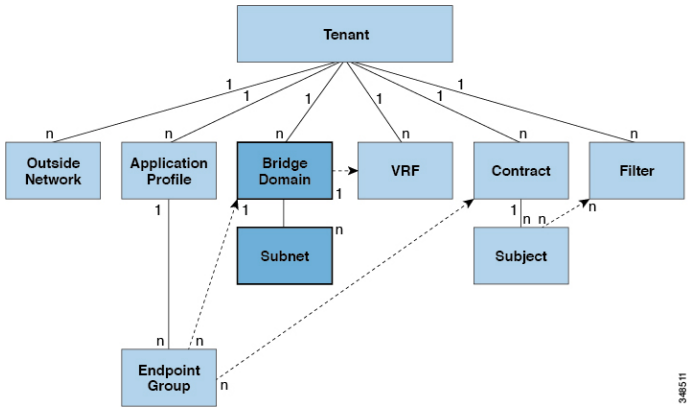


ACI ファブリックは、外部ルータの存在を認識せず、APIC はリーフ スイッチのインターフェイスを EPG に静的に割り当てます。

ブリッジ ドメインとサブネット

ブリッジ ドメイン（`fvBD`）は、ファブリック内のレイヤ 2 フォワーディングの構造を表します。次の図は、管理情報ツリー（MIT）内のブリッジ ドメイン (BD) の場所とテナントの他のオブジェクトとの関係を示します。

図 *7:* ブリッジ ドメイン



BD は、VRF (コンテキストまたはプライベート ネットワークとも呼ばれる) にリンクする必要があります。レイヤ 2 VLAN を除いて、少なくとも 1 つのサブネット（`fvSubnet`）が関連付けられている必要があります。BDは、このようなフラッディングが有効の場合に、一意のレイヤ 2 MAC アドレス空間およびレイヤ 2 フラッド ドメインを定義します。VRF が一意の IP アドレス空間を定義する一方で、そのアドレス空間は複数のサブネットで構成できます。これらのサブネットは、対応する VRF を参照する 1 つ以上のブリッジ ドメインで定義されます。

BD下または EPG 下のサブネットのオプションは次のとおりです:

- *Public*：サブネットをルーテッド接続にエクスポートできます。
- *Private*：サブネットはテナント内にのみ適用されます。
- *Shared*：共有サービスの一部として、同じテナントまたは他のテナントにわたる複数の VRF に対してサブネットの共有やエクスポートを行うことができます。共有サービスの例としては、異なるテナントの別の VRF に存在する EPG へのルーテッド接続などがあります。これにより、トラフィックが VRF 間で双方向に移動することが可能になります。共有サービスを提供する EPG のサブネットは （BD 下ではなく） その EPG 下で設定する必要があり、そのスコープは外部的にアドバタイズされ、VRF 間共有されるのように設定する必要があります。

（注） 共有サブネットは、通信に含まれる VRF 全体で一意でなければなりません。EPG 下のサブネットがレイヤ3外部ネットワーク共有サービスを提供する場合、このようなサブネットは、ACI ファブリック内全体でグローバルに一意である必要があります。

BD パケットの動作は次の方法で制御できます:

パケット タイプ	モード
ARP	**ARP フラッディング**は有効または無効にできます。フラッディングを行わない場合、ARP パケットはユニキャストで送信されます。 （注） `limitIpLearnToSubnets` を `fvBD` で設定すると、BD の設定済みサブネット内または共有サービス プロバイダーである EPG サブネット内に IP アドレスが存在する場合のみ、エンドポイントの学習が BD に限定されます。
未知のユニキャスト	**L2 Unknown Unicast** は、**Flood** または **Hardware Proxy** になり得ます。 （注） BD が **L2 Unknown Unicast** を持っており、それが **Flood** に設定されている場合、エンドポイントが削除されると、システムはそれを両方のローカル リーフ スイッチから削除します。そして、**Clear Remote MAC Entries** を選択すると、BD が展開されているリモートのリーフ スイッチからも削除されます。この機能を使用しない場合、リモート リーフは、タイマーが時間切れになるまで、学習したこのエンドポイントの情報を保持します。 **L2 Unknown Unicast** の設定を変更すると、このブリッジ ドメインに関連付けられた EPG にアタッチされているデバイスのインターフェイス上で、トラフィックがバウンスします (アップ ダウンします)。
未知の IP マルチキャスト	**L3 の不明なマルチキャスト フラッディング** **Flood** — パケットは入力および境界リーフ スイッチノードでのみフラッディングされます。N9K-93180YC-EX では、パケットは、ブリッジ ドメインが導入されているすべてのノードでフラッディングされます。 **Optimized** — 1 リーフあたり 50 のブリッジ ドメインのみサポートされます。この制限は N9K-93180YC-EX には該当しません。

パケット タイプ	モード
L2マルチキャスト、ブロードキャスト、ユニキャスト	**マルチ宛先フラッディング**、次のいずれかになり得ます。 • **Flood in BD** — ブリッジ ドメインにフラッドします。 • **Flood in Encapsulation** — カプセル化でフラッドします。 • **Drop** — パケットをドロップします。

（注） Cisco APIC リリース 3.1(1) 以降では、Cisco Nexus 9000 シリーズ スイッチで (EX と FX で終わる名前を持つものとそれ以降)、次のプロトコルのカプセル化のフラッディングまたはブリッジ ドメインにフラッディングが可能です。OSPF/OSPFv3、BGP、EIGRP、CDP、LACP、LLDP、ISIS、IGMP、PIM、ST-BPDU、ARP/GARP、RARP、ND。

ブリッジ ドメインは複数のスイッチにまたがることができます。ブリッジ ドメインには複数のサブネットを含めることができますが、サブネットは単一のブリッジ ドメイン内に含まれます。ブリッジ ドメイン（fvBD）の `limitIPLearnToSubnets` プロパティが `yes` に設定されていると、ブリッジ ドメインの設定済みサブネットのいずれかの中に IP アドレスがあるとき、または EPG が共有サービス プロバイダーである場合には EPG サブネット内に IP アドレスがあるときのみ、ブリッジ ドメイン内でエンドポイントの学習が行われます。サブネットは複数の EPG にまたがることができ、1 つ以上の EPG を 1 つのブリッジ ドメインまたはサブネットに関連付けることができます。ハードウェアのプロキシ モードでは、異なるブリッジ ドメインのエンドポイントがレイヤ3のルックアップ動作の一部として学習されると、そのエンドポイントに ARP トラフィックが転送されます。

ブリッジ ドメイン オプション

ブリッジ ドメインは、不明なユニキャスト フレームのフラッド モードで、またはこれらのフレームのフラッディングを排除する最適化されたモードで動作するように設定できます。フラッディング モードで使用する場合、レイヤ 2 の不明なユニキャスト トラフィックはブリッジ ドメイン（GIP）のマルチキャスト ツリーでフラッディングされます。最適化されたモードでブリッジ ドメインを動作するようにするには、ハードウェア プロキシに設定する必要があります。この状況では、レイヤ 2 の不明なユニキャスト フレームはスパイン プロキシ エニーキャスト VTEP アドレスに送信されます。

注意 不明なユニキャスト フラッディング モードから hw プロキシ モードに変更すると、ブリッジ ドメイン内のトラフィックが停止します。

ブリッジ ドメインで IP ルーティングが有効になっている場合、マッピング データベースは、MAC アドレスだけでなく、エンドポイントの IP アドレスを学習します。

レイヤ 3 の設定 ブリッジ ドメイン ()] パネルのタブには次のパラメータを設定するには、管理者が使用できます。

- **ユニキャスト ルーティング** ：この設定が有効になっているサブネット アドレスが設定されている場合は、ファブリックはデフォルト ゲートウェイの機能を提供して、トラフィックをルーティングします。ユニキャストルーティングを有効にすると、マッピングデータベースがこのブリッジドメインのエンドポイントに付与された IP アドレスと VTEP の対応関係を学習します。IP学習は、ブリッジドメイン内にサブネットが構成されているかどうかに左右されません。
- **サブネット アドレス** ：このオプションは、ブリッジ ドメインの SVI IP アドレス (デフォルト ゲートウェイ) を設定します。
- **制限のサブネット IP ラーニング** ：このオプションは、ユニキャスト リバース転送パス チェックに似ています。このオプションを選択すると、ファブリックはブリッジ ドメインに設定されている 1 以外のサブネットから IP アドレスを学習されません。

⚠

注意 有効化 **サブネットに制限 IP ラーニング** がブリッジ ドメイン内のトラフィックを停止します。

拡張 L2 専用モード：レガシーモード

Cisco ACI では、VLAN が異なるリーフ ノードに展開されている限り、任意の目的で同じ VLAN ID を再利用できます。これにより、Cisco ACI ファブリックは、ファブリックとしての VLAN の理論上の最大数、4094 を超えることができます。ただし、これを実現するため、および基盤となる VxLAN 実装の複雑さを隠すために、個々のリーフノードに含めることのできる VLAN の数は少なくなります。このことは、リーフノードあたりの VLAN の密度が必要な場合に問題の原因となる可能性があります。このようなシナリオでは、ブリッジ ドメインで以前はレガシーモードと呼ばれていた、拡張 L2 専用モードを有効にできます。拡張L2 専用モードのブリッジドメインでは、リーフノードごとに多数の VLAN を使用できます。ただし、このようなブリッジ ドメインにはいくつかの制限があります。

拡張L2 専用モードとそれ以外のモードで、リーフ ノードごとにサポートされる VLAN またはブリッジ ドメインの数については、ご使用のリリースの Verified Scalability Guideを参照してください。

拡張 L2 専用モードの制限事項

レガシー モードまたは拡張 L2 専用モードの制限は次のとおりです。

- ブリッジ ドメインには、1 つの EPG と 1 つの VLAN のみを含めることができます。
- ユニキャスト ルーティングはサポートされていません。
- コントラクトはサポートされていません。

- VMM 統合のダイナミック VLAN 割り当てはサポートされていません。
- サービス グラフはサポートされていません。
- QoS ポリシーはサポートされていません。
- ブリッジ ドメインは、スタンドアロン Cisco NX-OS では基本的に VLAN として動作します。

拡張 L2 専用モードの設定

次に、拡張 L2 専用 モードでブリッジ ドメインを設定する際の考慮事項を示します。

- VLAN ID はブリッジドメインで設定されます。
- EPG で設定された VLAN ID は上書きされます。
- 既存のブリッジ ドメインで拡張 L2 専用モードの有効と無効を切り替えると、サービスに影響します。

  VLAN API が変更前に使用されていたものと異なる場合、Cisco APIC は自動的にブリッジドメインの展開解除と再展開を行います。

  モード変更の前後で同じ VLAN ID が使用された場合、Cisco APIC はブリッジ ドメインの自動的な展開解除と再展開は行いません。手動でブリッジドメインを展開解除して再展開する必要があります。これは、 EPG で静的ポート設定を削除して再作成することで実行できます。
- 拡張 L2 専用モードの VLAN ID を変更する場合は、まずモードを無効にしてから、新しい VLAN ID で拡張 L2 専用モードを有効にする必要があります。

ブリッジ ドメインごとの IP 学習の無効化

2 つのホストが Cisco ACI スイッチにアクティブおよびスタンバイのホストとして接続されている場合、ブリッジ ドメインごとの IP 学習は無効になります。MAC 学習は引き続きハードウェアで発生しますが、IP 学習は ARP/GARP/ND プロセスからのみ発生します。この機能は、ファイアウォールまたはローカル ゲートウェイのような、柔軟な導入を可能にします。

ブリッジ ドメインごとに IP 学習を無効化するには、次の注意事項と制限事項を参照してください。

- remote top-of-rack（ToR）スイッチで送信元 IP アドレスが S,G 情報を入力するように学習していないため、レイヤ 3 マルチキャストはサポートされていません。
- DL ビットが iVXLAN ヘッダーで設定されているため、MAC アドレスはリモート TOR のデータ パスから学習されません。BD が展開されているファブリックで、リモート TOR からすべての TOR に不明なユニキャスト トラフィックをフラッディングします。エンドポイント データプレーン ラーニングが無効になっている場合は、この状況を克服するようにプロキシ モードで BD を設定することをお勧めします。
- ARP がフラッド モードであり、GARP ベースの検出を有効にする必要があります。

- IP ラーニングを無効にすると、対応する VRF でレイヤ 3 エンドポイントがフラッシュされません。同じ TOR を永遠に指すエンドポイントになる可能性があります。この問題を解決するには、すべての TOR のこの VRF 内ですべてのリモート IP エンドポイントをフラッシュします。

BD の設定を変更して、データプレーン学習を無効にしても、以前にローカルに学習したエンドポイントはフラッシュされません。これにより、既存のトラフィック フロー中断の影響は限られます。Cisco ACI リーフが特定の送信元 MAC を持つトラフィックをエンド ポイント保持ポリシーよりも長く見ない場合、MAC が学習したエンドポイントは通常どおりエージングします。

（注） IP データプレーン ラーニングを無効にすると、トラフィック転送の結果としてエンドポイント IP 情報が更新されることはなくなりますが、Cisco ACI は ARP/ND を使用してエンドポイント IP 情報を更新できます。つまり、ローカル エンドポイントのエージング（設定変更前に学習されたか、設定変更後に学習されたか）は、通常のエージングとは若干異なり、[システム (System)] > [システム設定 (System Settings)] > [エンドポイント制御 (Endpoint Controls)] > [IPエージング (IP Aging)] にも依存します。

IP エージングが無効の場合、すでに学習されたエンドポイント MAC と一致する送信元 MAC からのトラフィックは、エンドポイントテーブルの MAC アドレス情報を更新し、その結果、IP 情報も更新します（これは IP データプレーンの学習が有効になっている場合と同じです）。

IP エージングが有効の場合、ACI はエンドポイント IP アドレスを個別にエージング アウトします（これは IP データプレーン ラーニングが有効になっている場合と同じです）が、すでに学習したエンドポイントとマッチする既知の送信元 MAC および IP からのトラフィックにより、エンドポイント テーブルの MAC アドレス情報は更新されるのに対し、IP 情報は更新されないという点で、IP データプレーン ラーニングを有効にした設定とは異なります。

GUI を使用したテナント、VRF およびブリッジ ドメインの作成

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

手順

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

ステップ 2 [Create Tenant] ダイアログボックスで、次のタスクを実行します。

a) [Name] フィールドに、名前を入力します。

b) **[セキュリティ ドメイン（Security Domains）]** セクションで、**[+]** をクリックして、**[セキュリティ ドメインの作成（Create Security Domain）]** ダイアログ ボックスを開きます。
c) **[名前（Name）]** フィールドに、セキュリティ ドメインの名前を入力し、**[送信（Submit）]** をクリックします。
d) **[テナントの作成（Create Tenant）]** ダイアログ ボックスで、作成したセキュリティ ドメインの **[更新（Update）]** をクリックします。
e) 必要に応じて他のフィールドに入力します。
f) [送信（Submit）] をクリックします。

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

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

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

ステップ 4 **[作業（Work）]** ペインで、VRF インスタンスを囲む円内のキャンバスに **[ブリッジ ドメイン（Brdige Domain）]** アイコンをドラッグして、2 つを接続します。[Create Bridge Domain] ダイアログボックスが表示されたら、次のタスクを実行します。

a) [Name] フィールドに、名前を入力します。
b) 必要に応じて他のフィールドに入力します。
c) [次へ（Next）] をクリックします。
d) **[サブネット（Subnets）]** セクションで、**[+]** をクリックして、**[サブネットの作成（Create Subnet）]** ダイアログ ボックスを開きます。
e) **[ゲートウェイ IP**（Gateway IP）] フィールドに、IP アドレスとサブネット マスクを入力します。
f) 必要に応じて他のフィールドに入力します。
g) [OK] をクリックします。
h) **[ブリッジ ドメインの作成（Create Bridge Domain）]** ダイアログ ボックスに戻り、必要に応じて他のフィールドに入力します。
i) [次へ（Next）] をクリックします。
j) 必要に応じてフィールドに入力します。
k) **[OK]** をクリックしてブリッジ ドメインの設定を完了します。

ステップ 5 **[作業（Work）]** ペインで、VRF インスタンスを囲む円内のキャンバスに **[L3]** アイコンをドラッグして、2 つを接続します。[Create Routed Outside] ダイアログボックスが表示されたら、次のタスクを実行します。

a) [Name] フィールドに、名前を入力します。
b) **[ノードとインターフェイス プロトコル プロファイル（Nodes And Interfaces Protocol Profiles]** セクションで、**[+]** をクリックして **[ノード プロファイルの作成（Create Node Profile）]** ダイアログ ボックスを開きます。
c) [Name] フィールドに、名前を入力します。

d) **[ノード（Nodes）]** セクションで、**[+]** をクリックして **[ノードの選択（Select Node）]** ダイアログ ボックスを開きます。
e) **[ノード ID（Node ID）]** ドロップダウン リストから、ノードを選択します。
f) [Router ID] フィールドに、ルータ ID を入力します。
g) **[スタティック ルート（Static Routes）]** セクションで、**[+]** をクリックして**[スタティック ルートの作成（Create Static Routes）]** ダイアログ ボックスを開きます。
h) [Prefix] フィールドに、IPv4 アドレスまたは IPv6 アドレスを入力します。
i) **[ネクスト ホップ アドレス（Next Hop Addresses）]** セクションで、**[+]** をクリックして **[ネクスト ホップの作成（Create Next Hop）]** ダイアログ ボックスを開きます。
j) **[ネクスト ホップ アドレス（Next Hop Addresses）]** フィールドを展開し、IPv4 アドレスまたは IPv6 アドレスを入力します。
k) **[設定（Preference）]** フィールドに、数値を入力します。
l) 必要に応じて他のフィールドに入力します。
m) [OK] をクリックします。
n) **[静的ルートの作成（Create Static Route）]** ダイアログ ボックスで、必要に応じて他のフィールドに入力します。
o) [OK] をクリックします。
p) **[ノードの選択（Select Node）]** ダイアログ ボックスで、必要に応じて他のフィールドに入力します。
q) [OK] をクリックします。
r) **[ノード プロファイルの作成（Create Node Profile）]** ダイアログ ボックスで、必要に応じて他のフィールドに入力します。
s) [OK] をクリックします。
t) 必要に応じて **[BGP]**、**[OSPF]**、または **[EIGRP]** チェックボックスをオンにします。
u) 必要に応じて他のフィールドに入力します。
v) [次へ（Next）] をクリックします。
w) 必要に応じてフィールドに入力します。
x) **[OK]** をクリックしてレイヤ 3 の設定を完了します。

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

NX-OS CLI を使用した、テナント、VRF およびブリッジ ドメインの作成

ここでは、テナント、VRF およびブリッジ ドメインを作成する方法を説明します。

（注） テナントの設定を作成する前に、vlan-domain コマンドを使用して VLAN ドメインを作成し、ポートを割り当てる必要があります。

手順

ステップ 1 次のように、VLAN ドメイン（一連のポートで許可される一連の VLAN を含む）を作成し、VLAN の入力を割り当てます。

例：

次の例（exampleCorp）では、VLAN 50 ～ 500 が割り当てられることに注意してください。

```
apic1# configure
apic1(config)# vlan-domain dom_exampleCorp
apic1(config-vlan)# vlan 50-500
apic1(config-vlan)# exit
```

ステップ 2 VLANが割り当てられたら、これらのVLANを使用できるリーフ（スイッチ）およびインターフェイスを指定します。次に、「vlan-domain member」と入力し、その後に作成したドメインの名前を入力します。

例：

次の例では、これらの VLAN（50 ～ 500）は、インターフェイス イーサネット 1/2 ～ 4（1/2、1/3、1/4 を含む 3 つのポート）上の leaf 101 で有効になっています。これは、このインターフェイスを使用すると、VLAN を使用できるあらゆるアプリケーションにこのポートの VLAN 50 ～ 500 を使用できることを意味します。

```
apic1(config-vlan)# leaf 101
apic1(config-vlan)# interface ethernet 1/2-4
apic1(config-leaf-if)# vlan-domain member dom_exampleCorp
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

ステップ 3 次の例に示すように、グローバル コンフィギュレーション モードでテナントを作成します。

例：

```
apic1(config)# tenant exampleCorp
```

ステップ 4 次の例に示すように、テナント コンフィギュレーション モードでプライベート ネットワーク（VRF とも呼ばれます）を作成します。

例：

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

ステップ 5 次の例に示すように、テナントの下にブリッジ ドメイン（BD）を作成します。

例：

```
apic1(config-tenant)# bridge-domain exampleCorp_b1
apic1(config-tenant-bd)# vrf member exampleCorp_v1
apic1(config-tenant-bd)# exit
```

（注） この場合、VRF は「exampleCorp_v1」です。

ステップ 6 次の例に示すように、BD の IP アドレス（IP および ipv6）を割り当てます。

例：

```
apic1(config-tenant)# interface bridge-domain exampleCorp_b1
apic1(config-tenant-interface)# ip address 172.1.1.1/24
apic1(config-tenant-interface)# ipv6 address 2001:1:1::1/64
apic1(config-tenant-interface)# exit
```

次のタスク

次の項では、アプリケーション プロファイルを追加し、アプリケーション エンドポイント グループ（EPG）を作成し、EPG をブリッジ ドメインに関連付ける方法について説明します。

関連トピック

NX-OS スタイルの CLI を使用した VLAN ドメインの設定

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 https://apic-ip-address/api/mo/uni.xml
<fvTenant name="ExampleCorp"/>
```

POST が成功すると、作成したオブジェクトが出力に表示されます。

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：

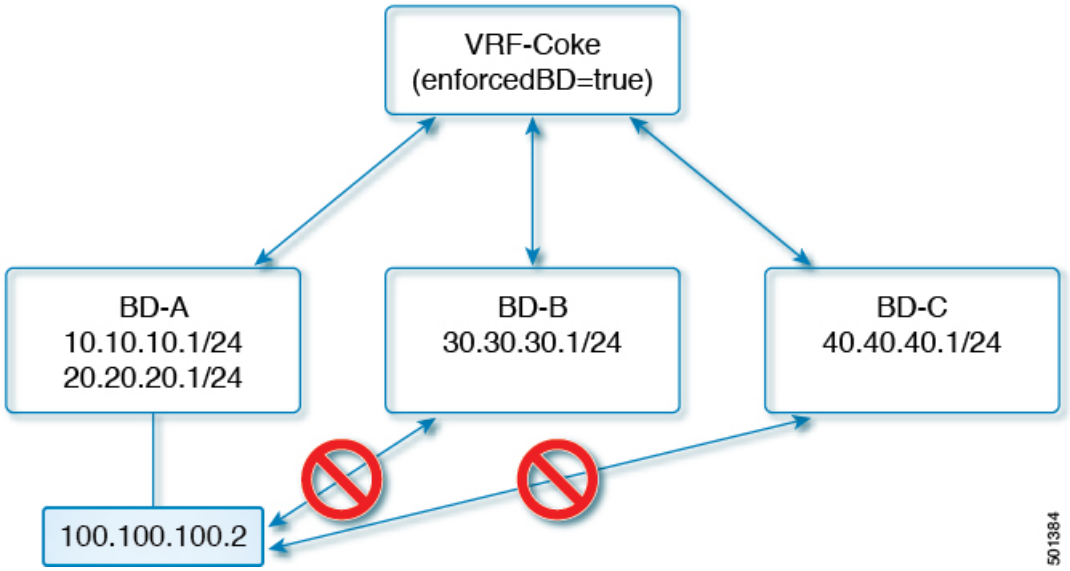
```
 URL for POST: https://apic-ip-address/api/mo/uni/tn-ExampleCorp.xml

<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。

適用されるブリッジ ドメインの設定

適用ブリッジ ドメインでは、関連付けられたブリッジ ドメイン内のサブネット ゲートウェイにしか ping を送信できない、対象のエンド ポイント グループ（EPG）内に、1 つのエンドポイントが作成されます。この設定を使用すると、任意のサブネットゲートウェイに ping を送信できる IP アドレスのグローバル例外リストを作成できます。

図 *8*：適用されるブリッジ ドメイン



（注）
- 例外 IP アドレスは、すべての VRF インスタンスのすべてのブリッジ ドメイン ゲートウェイに ping を送信できます。
- L3Out 用に設定されたループバック インターフェイスでは、対象のループバック インターフェイスに合わせて設定されたIPアドレスへの到達可能性は適用されません。
- EBGP ピアとなる IP アドレスが、L3Out インターフェイスのサブネットとは異なるサブネットに存在している場合には、許容例外サブネットにピア サブネットを追加する必要があります。そうしないと、送信元 IP アドレスが L3Out インターフェイスのサブネットとは異なるサブネットに存在するため、eBGP トラフィックがブロックされます。
- 適用ブリッジ ドメインは、VRFインスタンスがインバンドまたはアウトオブバンドであるかどうかにかかわらず、管理テナントではサポートされません。これらのVRFインスタンスへのトラフィックを制御するルールは、通常のコントラクトを使用して設定する必要があります。

NX-OS スタイル CLI を使用した適用されるブリッジ ドメインの設定

このセクションでは、NX-OS スタイル コマンドライン インターフェイス (CLI) を使用して、適用されるブリッジ ドメインを設定する方法について説明します。

手順

ステップ 1 テナントを作成し有効にします。

例：

次の例 (「cokeVrf」) が作成され有効になっています。

```
apic1(config-tenant)# vrf context cokeVrf
apic1(config-tenant-vrf)# bd-enforce enable
apic1(config-tenant-vrf)# exit
apic1(config-tenant)#exit
```

ステップ 2 例外リストに、サブネットを追加します。

例：

```
apic1(config)#bd-enf-exp-ip add1.2.3.4/24
apic1(config)#exit
```

適用されるブリッジドメインは次のようなコマンドを使用して動作可能かどうかを確認できます。

```
apic1# show running-config all | grep bd-enf
bd-enforce enable
bd-enf-exp-ip add 1.2.3.4/24
```

例

次のコマンドでは、除外リストからサブネットを削除します。

```
apic1(config)# no bd-enf-exp-ip 1.2.3.4/24
apic1(config)#tenant coke
apic1(config-tenant)#vrf context cokeVrf
```

次のタスク

適用されるブリッジ ドメインを無効にするには、次のコマンドを実行します。

```
apic1(config-tenant-vrf)# no bd-enforce enable
```

REST API を使用した、適用されるブリッジ ドメインの設定

手順

	コマンドまたはアクション	目的

ステップ 1	テナントを作成します。
	例：
	`POST https://apic-ip-address/api/mo/uni.xml`
	`<fvTenant name="ExampleCorp"/>`
		POST が成功すると、作成したオブジェクトが出力に表示されます。

ステップ 2	VRF およびブリッジ ドメインを作成します。
	例：
	`URL for POST: https://apic-ip-address/api/mo/uni/tn-ExampleCorp.xml`
	`<fvTenant name="ExampleCorp">`
	`<fvCtx name="pvn1"/>`
	`<fvBD name="bd1">`
	`<fvRsCtx tnFvCtxName="pvn1" bdEnforceEnable="yes"/>`
	`<fvSubnet ip="10.10.100.1/24"/>`
	`</fvBD>`
	`</fvTenant>`
	例外 IP を追加するには、次の POST 送信を使用します:
	https://apic-ip-address/api/node/mo/uni/infra.xml
	`<bdEnforceExceptionCont>`
		（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。

	コマンドまたはアクション	目的
	<bdEnforceExceptIp ip="11.0.1.0/24"/> </bdEnforceExceptionCont> （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。	

カプセル化によるすべてのプロトコルおよびプロキシ ARP のカプセル化のフラッディングを設定する

Cisco Application Centric Infrastructure（ACI）は、ブリッジ ドメインをレイヤ 2 ブロードキャスト境界として使用します。各ブリッジ ドメインには複数のエンドポイント グループ（EPG）を含めることができ、各 EPG は複数の仮想ドメインまたは物理ドメインにマッピングできます。各 EPG は、各ドメインで異なる VLAN カプセル化プールを使用することもできます。各 EPG は、各ドメインで異なる VLAN または VXLAN カプセル化プールを使用することもできます。

通常、ブリッジ ドメイン内に複数の EPG を配置すると、ブロードキャスト フラッディングはブリッジ ドメイン内のすべての EPG にトラフィックを送信します。EPG はエンドポイントをグループ化し、特定の機能を実行するためにトラフィックを管理するために使用されるものなので、ブリッジ ドメイン内のすべての EPG に同じトラフィックを送信することは必ずしも実用的ではありません。

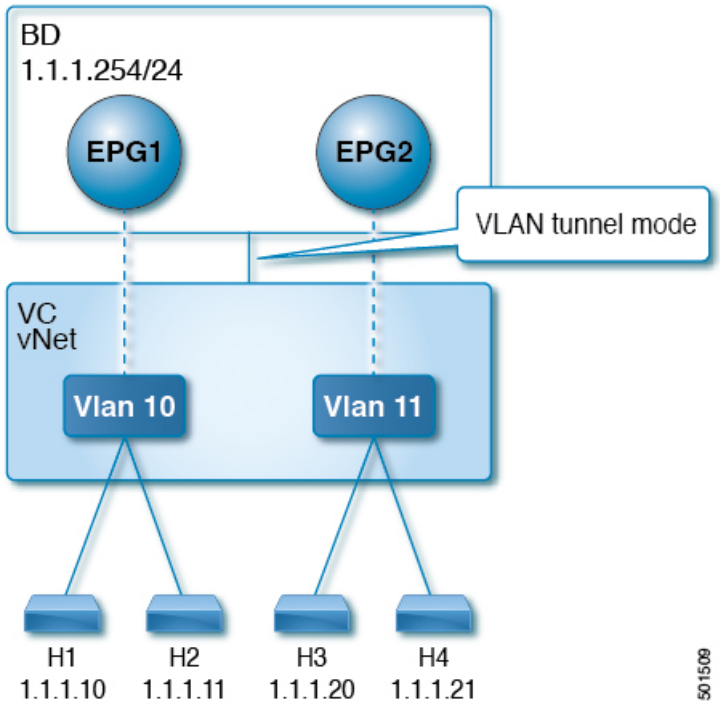
カプセル化でのフラッディングは、ネットワーク内のブリッジ ドメインを統合するのに役立ちます。この機能では、EPG が関連付けられている仮想ドメインまたは物理ドメインのカプセル化に基づいて、ブリッジ ドメイン内のエンドポイント（EP）へのブロードキャスト フラッディングを制御できます。

VLAN カプセル化を使用したカプセル化でのフラッディングの使用例

カプセル化のフラッディングは、外部デバイスが VLAN に依存しない MAC 学習のために vNet ごとに 1 つの MAC アドレスが維持される仮想接続 トンネル モードを使用している場合によく用いられます。

トンネル モードで複数の VLAN を使用すると、いくつかの課題を導入できます。次の図に示すように、単一のトンネルでCisco ACIを使用する一般的な導入では、1 つのブリッジ ドメインの下に複数の EPG があります。この場合、特定のトラフィックがブリッジ ドメイン内（つまりすべての EPG 内）でフラッディングし、MACがあいまいになって転送エラーが発生するリスクがあります。

図 *9 : VLAN*トンネルモードの*Cisco ACI*の課題



このトポロジでは、ブレードスイッチ（この例では仮想接続）に、1つのアップリンクを使用してCisco ACIリーフノードに接続する単一のトンネルネットワークが定義されています。このリンクでは、2 人のユーザの VLAN、VLAN 10 と VLAN 11 が行われます。サーバーのゲートウェイがCisco ACIクラウドの外部にあるため、ブリッジ ドメインはフラッディング モードに設定されます。次のプロセスで ARP 交渉が発生します。

- サーバは、VLAN 10 ネットワーク経由で 1 つの ARP ブロードキャスト要求を送信します。
- ARP パケットは、外部のサーバに向かってトンネル ネットワークを通過し、そのダウンリンクから学習した送信元 MAC アドレスを記録します。
- その後、サーバーはアップリンクからCisco ACIリーフスイッチにパケットを転送します。
- Cisco ACIファブリックは、アクセスポート VLAN 10に着信する ARP ブロードキャスト パケットを確認し、EPG1 にマッピングします。
- ブリッジ ドメインは ARP パケットをフラッディングするように設定されているため、パケットはブリッジ ドメイン内でフラッディングされます。したがって、両方の EPG が同じブリッジ ドメイン内にあるため、これらのポートにフラッディングされます。
- 同じ ARP ブロードキャスト パケットは、同じアップリンクで復帰します。
- ブレードスイッチは、このアップリンクからの元の送信元 MAC アドレスを認識します。

結果：ブレード スイッチは、単一の MAC 転送テーブル内のダウンリンク ポートとアップリンク ポートの両方から学習した同じMACアドレスを持ち、トラフィックが中断します。

推奨される解決策

カプセル化オプションのフラッディングは、ブリッジドメイン内のフラッディング トラフィックを単一のカプセル化に制限するために使用されます。EPG1/VLAN X and EPG2/VLAN Y が同じブリッジドメインを共有し、カプセル化でのフラッディングが有効になっている時、カプセル化フラッディング トラフィックは他の EPG/VLAN に到達しません。

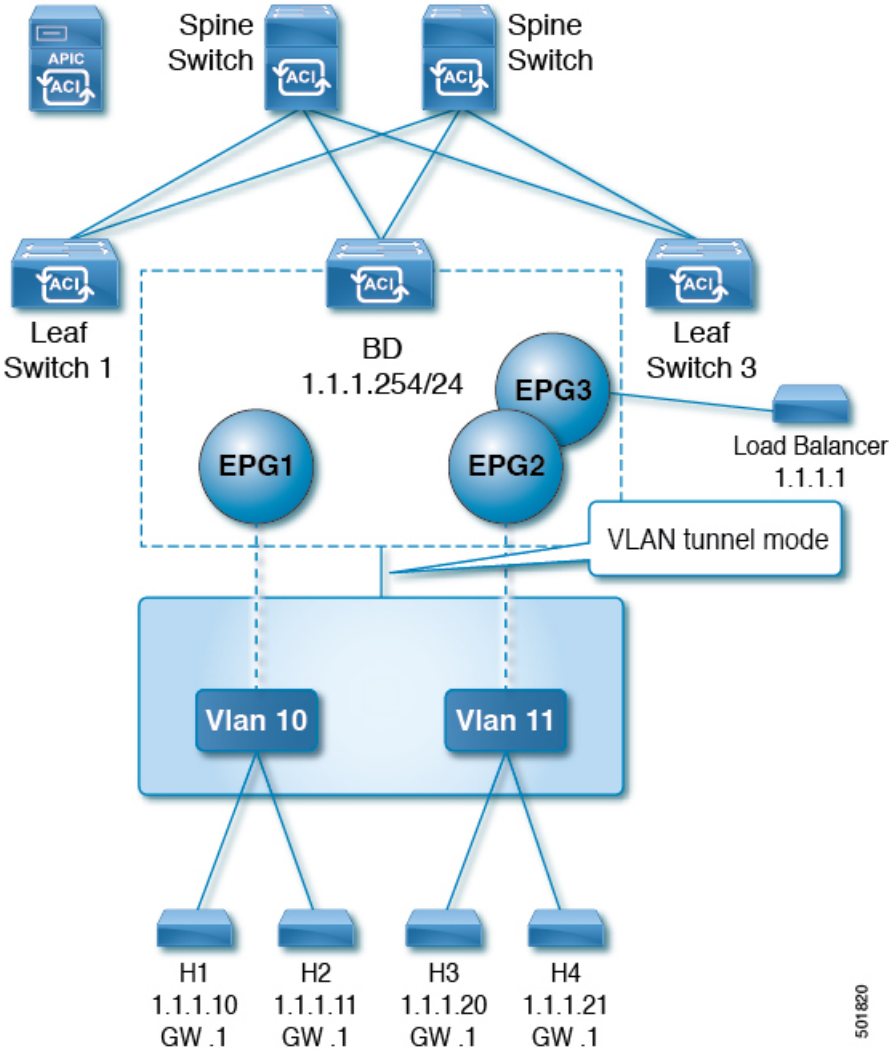
Cisco Application Policy Infrastructure Controller（APIC）リリース 3.1(1) 以降、Cisco Nexus 9000 シリーズスイッチ（名前の末尾が EX および FX 以降）では、すべてのプロトコルがカプセル化されます。また、VLAN 間のトラフィックのブリッジ ドメインでフラッディングが有効になっている場合、プロキシARPはMACフラップの問題が発生しないようにします。また、すべてのフラッディング（ARP、GARP、およびBUM）をカプセル化に制限します。この制限は、有効になっているブリッジ ドメイン下のすべての EPG に適用されます。

（注） Cisco APIC APIC リリース 3.1 (1) より前のリリースでは、これらの機能はサポートされていません（カプセル内でフラッディングするとき含まれるプロキシ ARP およびすべてのプロトコル）。以前の Cisco APIC リリースまたは以前の世代のスイッチ（名前に EX またはFXがない）では、カプセル化でフラッディングを有効にしても機能しません。情報障害は生成されませんが、Cisco APIC はヘルス スコアを 1 減らします。

（注） Cisco APIC リリース 3.2(5) 以降では、VXLAN カプセル化に関連付けられた EPG のカプセル化でフラッディングを設定できます。以前は、VLAN のみが仮想ドメインのカプセル化でのフラッディングでサポートされていました。ブリッジ ドメインまたは EPG を作成または変更するときに、カプセル化でのフラッディングを設定します。

推奨される解決策は、外部スイッチを追加して、1 つのブリッジ ドメインで複数の EPG をサポートすることです。外部のスイッチがある 1 つのブリッジ ドメイン下で複数の EPG を持つこの設計は、次の図に示されています。

図 *10* : 外部のスイッチがある *1* つのブリッジ ドメイン下で複数の *EPG* を持つ設計



同じブリッジ ドメイン内では、一部の EPG をサービス ノードにすることができ、他の EPG にはカプセル化でのフラッディングを設定できます。ロード バランサは別の EPG に存在します。ロード バランサは EPG からパケットを受信し、その他の EPG に送信します（プロキシ ARP はなく、カプセル内のフラッディングは発生しません）。

マルチ宛先プロトコル トラフィック

EPG/ブリッジ ドメイン レベルのブロードキャスト セグメンテーションは、次のネットワーク制御プロトコルでサポートされます。

- OSPF
- EIGRP
- CDP
- LACP

- LLDP
- IS-IS
- BGP
- IGMP
- PIM
- STP BPDU（EPG 内フラッディング）
- ARP/GARP（ARP プロキシによって制御）
- ND

カプセル化でのフラッディングの制限事項

すべてのプロトコルのカプセル化でのフラッディングには、次の制限が適用されます。

- カプセルのフラッディングは、ARP ユニキャスト モードでは機能しません。
- このリリースでは、ネイバー送信要求（プロキシ NS/ND）はサポートされていません。
- プロキシアドレス解決プロトコル（ARP）は暗黙的に有効にされるため、ARP トラフィックは異なるカプセル化間の通信のために CPU に送信できます。

  ARP トラフィックを処理するために異なるポートに均等に配信されるようにするには、ポート単位のコントロール プレーン ポリシング（CoPP）を有効にします。
- カプセル化でのフラッディングは、フラッド モードのブリッジ ドメインおよびフラッド モードの ARP でのみサポートされます。ブリッジ ドメイン スパイン プロキシ モードはサポートされていません。
- IPv4 レイヤ 3 マルチキャストはサポートされていません。
- カプセル化でのフラッディングが有効な場合でも、IPv6 NS/NDプロキシはサポートされません。その結果、同じ IPv6 サブネット下にあっても、カプセル化が異なる EPG に存在する 2 つのエンドポイント間の接続は、機能しないことがあります。
- 別の VLAN への VM の移行は、時間的な問題（60 秒）があります。別の VLAN または VXLAN への VM の移行の際には、一時的に（60秒）問題が発生します。
- VM の IP アドレスがファイアウォールの IP アドレスではなくゲートウェイの IP アドレスに変更された場合、ファイアウォールはバイパスされため、ファイアウォールをゲートウェイにする VM 間の通信設定は推奨されません。
- 以前のリリースではサポートされていません（以前と現在のリリース間の相互運用もサポートされていません）。
- 古い世代アプリケーション リーフ エンジン（ALE）とアプリケーション スパイン エンジン（ASE）を使用した混合モード トポロジは推奨されません。また、カプセル化でのフラッディングではサポートされません。同時に有効にすると、QoS の優先順位が適用されるのを防ぐことができます。

- 同じマルチサイト ドメインの一部であるCisco ACIファブリック全体に拡張された EPG とブリッジドメインでは、カプセル化でのフラッディングはサポートされません。ただし、Cisco ACIファブリックでローカルに定義された EPG とブリッジ ドメインでは、カプセル化でのフラッディングは引き続き機能し、完全にサポートされています。Cisco ACIファブリックと、そのファブリックに関連付けられたリモート リーフ スイッチ間でストレッチされる EPG またはブリッジ ドメインにも、同じ考慮事項が適用されます。
- マイクロセグメンテーションが設定されている EPG では、カプセル化でのフラッディングはサポートされません。
- 共通パーベイシブ ゲートウェイでは、カプセル化でのフラッディングはサポートされていません。Cisco APIC Layer 3 Networking Configuration Guide の「Common Pervasive Gateway」の章を参照してください。
- ブリッジ ドメインのすべてのEPGでカプセル化でのフラッディングを設定する場合は、ブリッジ ドメインでもカプセル化でのフラッディングを設定してください。
- IGMP スヌーピングは、カプセル化でのフラッディングではサポートされません。
- Cisco ACIにおいては、カプセル化でのフラッディングのために設定された EPG で受信されるパケットのフラッディングを、（カプセル化ではなく）ブリッジ ドメインで生じさせる条件が存在します。これは、管理者がカプセル化でのフラッディングを EPG で直接設定したか、ブリッジ ドメインで設定したかに関係なく発生します。この転送動作の条件は、入力リーフ ノードに宛先MACアドレスのリモート エンドポイントがあり、出力リーフ ノードに対応するローカル エンドポイントがない場合です。これは、インターフェイスのフラッピング、STP TCNによるエンドポイント フラッシュ、過剰な移動のためにブリッジ ドメインで学習が無効になっているなどの理由で発生する可能性があります。

  4.2(6o) 以降の 4.2(6)リリース、4.2(7m) 以降の 4.2(7) リリース、および 5.2(1g) 以降のリリースでは、この動作が拡張されました。管理者が（EPG ではなく）ブリッジ ドメインでカプセル化のフラッディングを有効にすると、Cisco ACI は非入力（出力および中継）リーフ ノード上の外部デバイスに面したダウンリンクからのカプセル化では、このようなパケットを送信しません。この新しい動作により、パケットが予期しないカプセル化に漏洩することが防止されます。カプセル化でのフラッディングがEPG レベルでのみ有効になっている場合、非入力リーフ ノードは、カプセル化ではなくブリッジ ドメインでパケットをフラッディングする可能性があります。詳細については、拡張バグ CSCvx83364 を参照してください。

カプセル化範囲限定のフラッディングの設定

NX-OS スタイルの CLI、REST API、またはCisco Application Policy Infrastructure Controller（APIC）GUI を使用して、カプセル化でフラッディングを設定します。

EPG に設定されたカプセル化のフラッディングは、ブリッジド メイン（BD）に設定されたカプセル化のフラッディングよりも優先されます。BD と EPG の両方を設定すると、動作は次に説明したようになります。

表 *2: BD* と *EPG* の両方が設定されているときの動作

設定	動作
EPG でのカプセルのフラッディングとブリッジ ドメインでのカプセルのフラッディング	カプセル化のフラッディングは、ブリッジ ドメインのすべての VLAN および VXLAN 上のトラフィックに対して発生します。
EPG でのカプセルのフラッディングが発生せずブリッジドメインでのカプセルのフラッディングが発生する	カプセル化のフラッディングは、ブリッジ ドメイン内のすべての VLAN および VXLAN のトラフィックに対して発生します。
EPG でのカプセルのフラッディングが発生しブリッジ ドメインでのカプセルのフラッディングが発生しない	カプセル化のフラッディングは、ブリッジ ドメインの EPG 内のその VLAN または VXLAN のトラフィックに対して発生します。
EPG でのカプセルのフラッディングが発生せずブリッジドメインでのカプセルのフラッディングも発生しない	ブリッジ ドメイン全体でフラッディングします。

Cisco APIC GUI を使用したカプセル化範囲限定のフラッディングの設定

ブリッジドメイン（BD）またはエンドポイントグループ（EPG）を作成または変更する場合は、Cisco Application Policy Infrastructure Controller（APIC）GUIを使用してカプセル化でフラッディングを設定します。

手順

ステップ 1 BD の作成時にカプセル化でフラッディングを設定するには、次の手順を実行します。

a) Cisco APIC にログインします。
b) **[Tenants]** > ***[tenant]*** > **[Networking]** > **[Bridge Domains]** を選択します。
c) **Bridge Domains** を右クリックして、**Create Bridge Domain** を選択します。
d) 手順 1 の **[Create Bridge Domain]** ダイアログ ボックスで、**[Multi Destination Flooding]** ドロップダウン リストから、[Flood in Encapsulation] を選択します。
e) 設定に応じてダイアログボックスの他のフィールドに入力し、**[Finish]** をクリックします。

ステップ 2 BD の変更時にカプセル化でフラッディングを設定するには、次の手順を実行します。

a) Cisco APIC にログインします。
b) **[Tenants]** > ***<tenan>]*** > **[Networking]** > **[Bridge Domains]** > ***<bridge domain>*** を選択します。
c) BD の作業ウィンドウで、**[Policy]** タブを選択し、**[General]** タブを選択します。
d) **[Multi Destination Flooding]** 領域で、**[Flood in Encapsulation]** を選択します。
e) **[送信（Submit）]** をクリックします。`

ステップ 3 EPG の作成時にカプセル化でフラッディングを設定するには、次の手順を実行します。

a) Cisco APIC にログインします。

b) **[Tenants]** > **<*tenant*>** > **[Application Profiles]** に移動します。
c) **[Application Profiles]** を右クリックし、**[Create Application EPG]** を選択します。
d) **[Create Application EPG]** ダイアログボックスの **[Flood in Encapsulation]** 領域で、**[Enabled]** を選択します。

カプセル化のフラッディングはデフォルトで無効になっています。

e) 設定に応じてダイアログボックスの他のフィールドに入力し、**[Finish]** をクリックします。

ステップ 4 EPG の変更時にカプセル化でフラッディングを設定するには、次の手順を実行します。

a) **[Tenants]** > **<*tenant*>** > **[Application Profiles]** > **[Application EPG]** > **<*application EPG*>** に移動します。
b) EPG の作業ウィンドウで、**[Policy]** タブを選択し、**[General]** タブを選択します。
c) **[Flood in Encapsulation]** 領域で、**[Enabled]** を選択します。
d) [送信（Submit）] をクリックします。

NX-OS スタイル CLI を使用したカプセル化でのフラッディングの設定

NX-OS スタイル CLI を使用して選択したエンドポイント グループ（EPG）のみに対してカプセル化でフラッディングを追加する場合は、EPG 下で **flood-on-encapsulation enable** コマンドを入力します。

すべての EPG に対してカプセル化でフラッディングを追加する場合、ブリッジ ドメインに対して **multi-destination encap-flood** CLI コマンドを使用します。

手順

ステップ 1 ブリッジドメイン（BD）のカプセル化でフラッディングを設定します。

例：

```
APIC1#configure
APIC1(config)# tenant tenant
APIC1(config-tenant)# bridge-domain BD-name
APIC1(config-tenant-bd)# multi-destination encap-flood
APIC1(config-tenant)#exit
APIC1(config)#
```

ステップ 2 EPG のカプセル化でフラッディングを設定します。

例：

```
APIC1(config)# tenant tenant
APIC1(config-tenant)# application AP1
APIC1(config-tenant-app)# epg EPG-name
APIC1(config-tenant-app-epg)# flood-on-encapsulation
APIC1(config-tenant-app-epg)#no flood-on-encapsulation
```

REST API を使用したカプセル化範囲限定のフラッディングの設定

手順

ステップ 1 REST APIを使用して、ブリッジドメイン（BD）およびエンドポイントグループ（EPG）のカプセル化でフラッディングを設定できます。

a) （BD の）カプセル化でフラッディングを設定します。

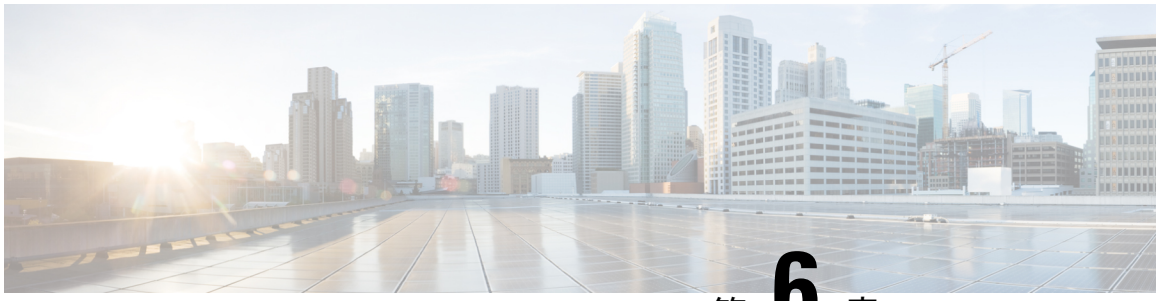
例：

```
URL for POST: https://apic-ip-address/api/mo/uni/tn-T1.xml

        <fvTenant name="T1">
           <fvCtx name="T1PN" />
           <fvBD arpFlood="yes" multiDstPktAct="encap-flood" name="T1PNBD-Web"
unkMacUcastAct="flood" unkMcastAct="flood" >
           </fvBD>
         </fvTenant>
```

ステップ 2 EPG のカプセル化でフラッディングを設定します。

例：

```
URL for POST: https://apic-ip-address/api/mo/uni/tn-T1.xml

<fvTenant name="T1">
    <fvAp name="AP1">
       <fvAEPg floodOnEncap="enabled" name="MS81Web" >
       </fvAEPg>
    </fvAp>
</fvTenant>
```



第 6 章

EPG

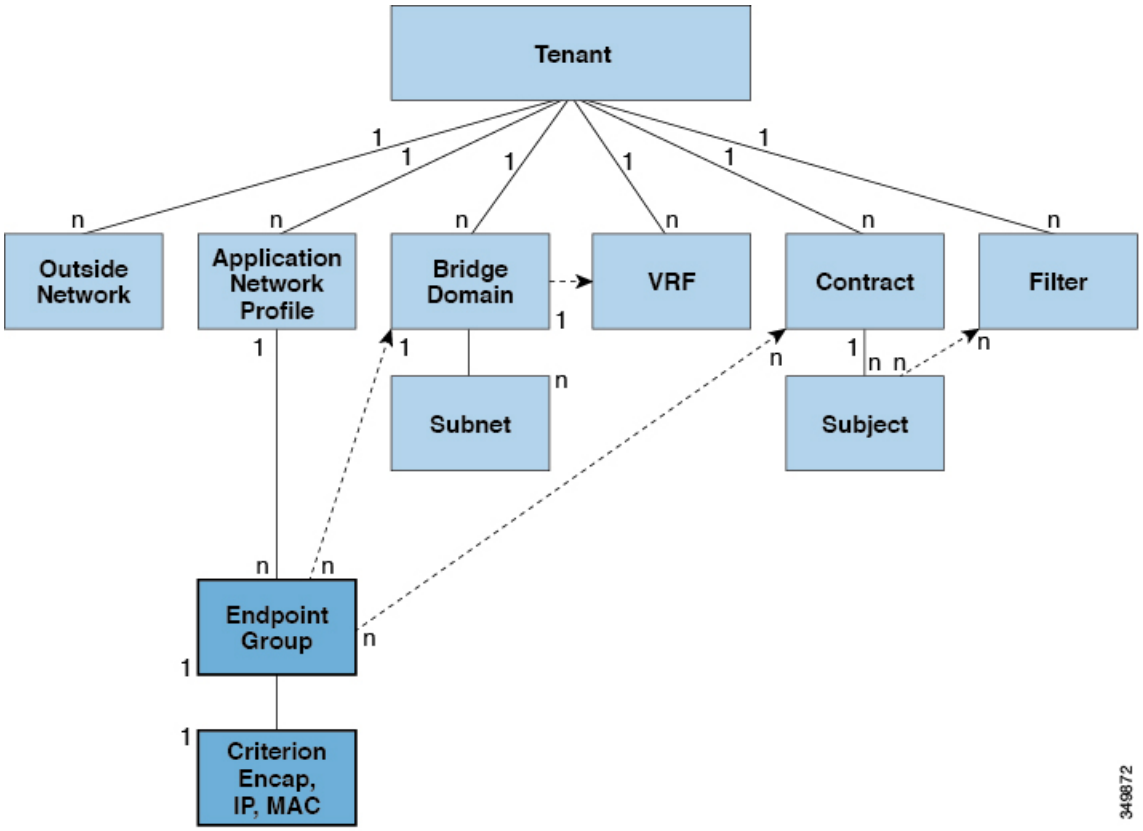
この章は、次の内容で構成されています。

エンドポイント グループについて

エンドポイント グループ

エンドポイント グループ（EPG）は、ポリシー モデルの最も重要なオブジェクトです。次の図は、管理情報ツリー（MIT）内のアプリケーションEPGの場所とテナント内の他のオブジェクトとの関係を示します。

図 *11 :* エンドポイント グループ



EPGは、エンドポイントの集合を含む名前付き論理エンティティである管理対象オブジェクトです。エンドポイントは、ネットワークに直接的または間接的に接続されるデバイスです。エンドポイントには、アドレス（ID）、ロケーション、属性（バージョンやパッチ レベルなど）があり、物理または仮想にできます。エンドポイントのアドレスを知ることで、他のすべてのID の詳細にアクセスすることもできます。EPG は、物理および論理トポロジから完全に分離されます。エンドポイントの例には、インターネット上のサーバ、仮想マシン、ネットワーク接続ストレージ、またはクライアントが含まれます。EPG 内のエンドポイント メンバシップは、ダイナミックまたはスタティックにできます。

ACI ファブリックには、次のタイプの EPG を含めることができます。

- アプリケーション エンドポイント グループ（`fvAEPg`）
- レイヤ 2 外部外側ネットワーク インスタンスのエンドポイント グループ（`l2extInstP`）
- レイヤ 3 外部外側ネットワーク インスタンスのエンドポイント グループ（`l3extInstP`）
- アウトオブバンド（`mgmtOoB`）またはインバンド（`mgmtInB`）アクセス用の管理エンドポイント グループ。

EPGには、セキュリティ、仮想マシンのモビリティ（VMM）、QoS、レイヤ4～レイヤ7サービスなどの共通のポリシー要件を持つエンドポイントが含まれます。エンドポイントは個別に設定および管理されるのではなく、EPG 内に配置され、グループとして管理されます。

ポリシーは EPG に適用されます。個々のエンドポイントに適用されることは絶対にありません。EPG は、APIC において管理者により静的に設定されるか、vCenter または OpenStack などの自動システムによって動的に設定されます。

（注） EPG がスタティック バインディング パスを使用する場合、この EPG に関連付けられるカプセル化 VLAN はスタティック VLAN プールの一部である必要があります。IPv4/IPv6 デュアルスタック設定の場合、IP アドレスのプロパティは `fvStCEp` MO の `fvStIp` 子プロパティに含まれます。IPv4 および IPv6 アドレスをサポートする複数の `fvStIp` を 1 つの `fvStCEp` オブジェクト下に追加できます。ACI を、IPv4 のみのファームウェアから、IPv6 をサポートするバージョンのファームウェアにアップグレードすると、既存の IP プロパティが `fvStIp` MO にコピーされます。

EPG の設定内容にかかわらず、含まれるエンドポイントに EPG ポリシーが適用されます。

ファブリックへの WAN ルータ接続は、スタティック EPG を使用する設定の 1 つの例です。ファブリックへの WAN ルータ接続を設定するには、関連付けられている WAN サブネット内のエンドポイントを含む `l3extInstP` EPG を管理者が設定します。ファブリックは、エンドポイントの接続ライフサイクルが経過する間に、検出プロセスを通して EPG のエンドポイントについて学習します。エンドポイントを学習すると、ファブリックは、それに基づいて `l3extInstP` EPG ポリシーを適用します。たとえば、WAN 接続クライアントがアプリケーション（`fvAEPg`）EPG 内でサーバとの TCP セッションを開始すると、`l3extInstP` EPG は、`fvAEPg` EPG Web サーバとの通信が始まる前に、そのクライアント エンドポイントにポリシーを適用します。クライアント サーバ TCP セッションが終わり、クライアントとサーバの間の通信が終了すると、そのエンドポイントはもうファブリック内に存在しません。

（注） リーフ スイッチが EPG 下の *static binding (leaf switches)* 用に設定されている場合は、次の制限が適用されます。

- スタティック バインディングをスタティック パスで上書きすることはできません。
- そのスイッチのインターフェイスをルーテッド外部ネットワーク（L3out）設定に使用することはできません。
- そのスイッチのインターフェイスに IP アドレスを割り当てることはできません。

VMware vCenter への仮想マシン管理接続は、ダイナミック EPG を使用する設定の 1 つの例です。ファブリックで仮想マシン管理ドメインが設定されると、vCenter は、必要に応じて仮想マシン エンドポイントを開始、移動、シャットダウンさせることのできる EPG の動的設定をトリガーします。

EPG シャットダウンでの ACI ポリシー設定

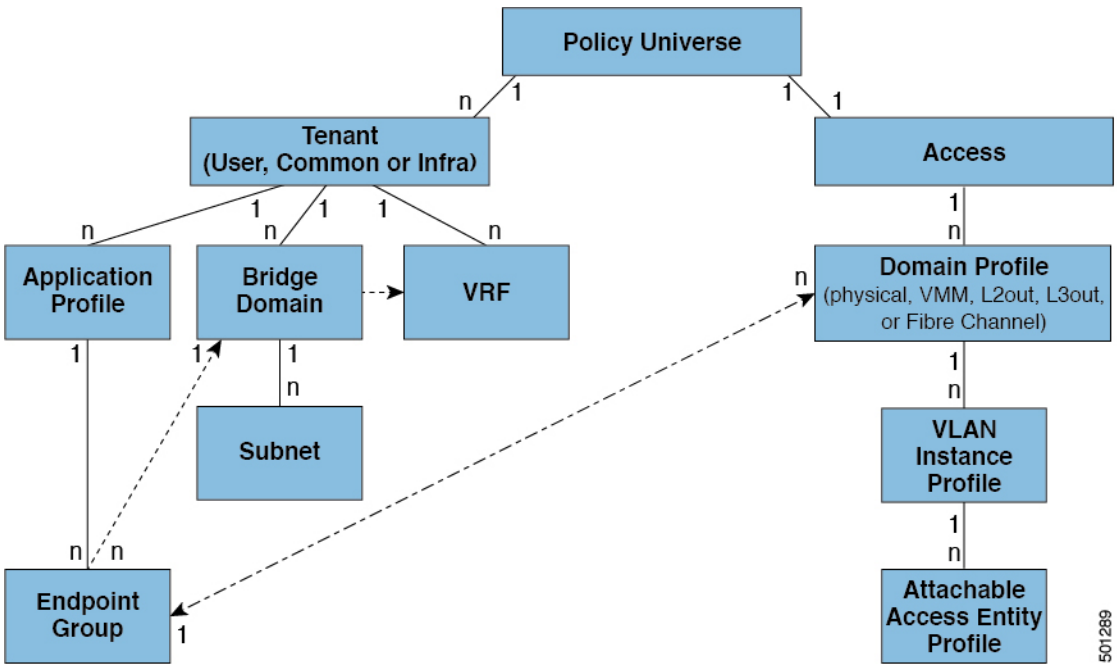
EPG がシャットダウン モードの場合、EPG に関連する ACI ポリシー設定はすべてのスイッチから削除されます。EPG はすべてのスイッチから削除されます。EPG が ACI データ ストアに存在している間は、非アクティブ モードになります。APIC GUI で、EPG をサービスから削除するチェックボックスをオンにすることができます。

（注） シャットダウン モードの EPG に接続されているホストは、EPG との間で送受信できません。

アクセス ポリシーによる VLAN から EPG への自動割り当て

テナント ネットワーク ポリシーがファブリックのアクセス ポリシーと別に設定される一方で、テナント ポリシーの基盤となるアクセス ポリシーが整わないとテナント ポリシーはアクティブ化されません。ファブリック アクセス外向きインターフェイスは、仮想マシン コントローラなどの外部デバイス、ハイパーバイザ、ホスト、ルータ、またはファブリック エクステンダ (FEX) と接続します。アクセス ポリシーにより、管理者はポート チャネルおよび仮想ポート チャネル、LLDP、CDP、LACP などのプロトコル、モニタリングや診断などの機能を設定することができます。

図 *12 :* アクセス ポリシーとエンドポイント グループの関連付け



ポリシー モデルでは、vlan の Epg 緊密に結合されています。トラフィックが流れるようにするには、物理、VMM、L2out、L3out、またはファイバ チャネル ドメイン内に VLAN を持つリーフ ポートに EPG を展開する必要があります。詳細については、ネットワーク ドメイン（13 ページ）を参照してください。

ポリシー モデルでは、EPG に関連付けられているドメイン プロファイルには、VLAN インスタンス プロファイルが含まれています。ドメイン プロファイルには、両方の VLAN インスタンス プロファイル(VLAN プール)および attacheable アクセス エンティティ プロファイル(AEP)アプリケーション Epg に直接と関連付けられているが含まれています。AEP は、すべてのポートの [接続されている、および Vlan の割り当てのタスクを自動化するに関連付けられているアプリケーション Epg を展開します。大規模なデータセンター数千の Vlan の数百のプロビジョニング仮想マシンのアクティブなは簡単に、中に ACI ファブリックは VLAN プールから、VLAN Id を自動的に割り当てることができます。これは、膨大な従来データセンターで Vlan をトランキングと比較して、時間を節約できます。

VLAN の注意事項

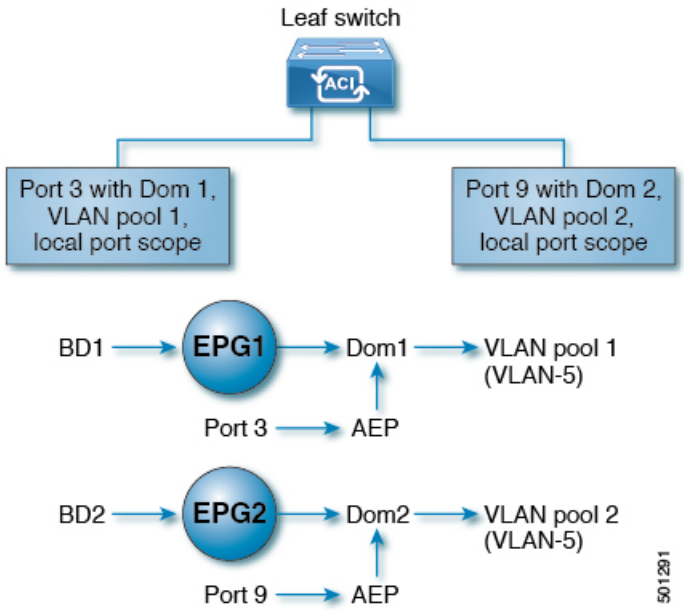
EPG トラフィックがフローは、Vlan の設定には次のガイドラインを使用します。

- 複数のドメインは、VLAN プールを共有できますが、1 つのドメインは、1 つの VLAN プールにのみ使用できます。
- 1 つのリーフ スイッチで同じ VLAN のカプセル化を複数の Epg を展開するを参照してください。 ポート単位の VLAN （47 ページ） 。

ポート単位の VLAN

v1.1 リリースより前の ACI バージョンでは、特定の VLAN カプセル化はリーフ スイッチ上の単一の EPG だけにマッピングされます。同じリーフ スイッチ上に同じ VLAN カプセル化を持つ第 2 の EPG があると、ACI でエラーが発生します。

v1.1 リリース以降では、次の図と同様、ポート単位の VLAN 設定で、特定のリーフ スイッチ(または FEX) 上に複数の EPG を同じ VLAN カプセル化で展開することができます。



単一のリーフ スイッチ上で、同じカプセル化番号を使用する複数の EPG の展開を有効にするには、次の注意事項に従ってください。

- EPG は、さまざまなブリッジ ドメインに関連付けられている必要があります。
- EPG は、さまざまなポートに展開する必要があります。
- ポートと EPG の両方が、VLAN 番号が含まれている VLAN プールに関連付けられている同じドメインに関連付けられている必要があります。
- ポートは `portLocal` VLAN スコープで設定されている必要があります。

たとえば、上の図の ポート 3 と 9 上に展開されている EPG のポート単位の VLAN で、両方が VLAN-5 を使用していれば、ポート 3 と EPG1 は Dom1 (プール 1) に、ポート 9 と EPG2 は Dom2 (プール 2) に関連付けられます。

ポート 3 からのトラフィックは EPG1 に関連付けられ、ポート 9 からのトラフィックは EPG2 に関連付けられます。

これは、外部レイヤ 3 外部接続用に設定されたポートには適用されません。

EPG に複数の物理ドメインがあり、VLAN プールが重複している場合は、EPG をポートに展開するために使用される AEP に複数のドメインを追加しないでください。これにより、トラフィック転送の問題が回避されます。

EPG に重複する VLAN プールを持つ物理ドメインが 1 つしかない場合、複数のドメインを単一の AEP に関連付けることができます。

入力および出力の両方向で個別の（ポート、VLAN）変換エントリの割り当てが可能なのは、`vlanScope` が `portlocal` に設定されているポートだけです。特定のポートで `vlanScope` が `portGlobal` (デフォルト) に設定されている場合には、EPG で使用される各 VLAN は、特定のリーフ スイッチ上で一意のものである必要があります。

（注） マルチ スパニング ツリー (MST) で設定されているインターフェイス上では、ポート単位の VLAN はサポートされていません。このツリーでは、VLAN ID が 1 つのリーフ スイッチ上で一意であること、そして VLAN の範囲がグローバルであることを必要とするからです。

同じリーフ スイッチで EPG に使用されていた VLAN 番号の再利用

以前に、リーフ スイッチのポートに展開されている EPG 用に VLAN を設定していて、同じ VLAN 番号を同じリーフ スイッチの異なるポートの異なる EPG で再利用する場合には、中断なしでセットアップできるようにするため、次の例に示すようなプロセスに従ってください。

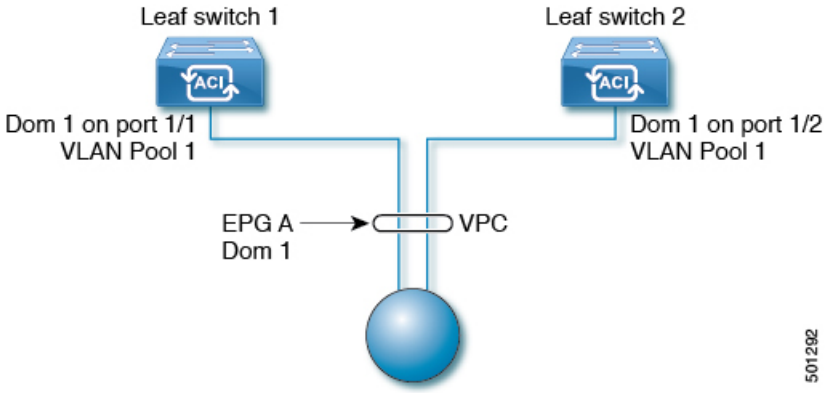
この例では、EPGは以前、9 ～ 100 の範囲の VLAN プールを含むドメインに関連付けられていたポートに展開されていました。ここで、9 ～ 20 からの VLAN カプセル化を使用する EPG を設定したいとします。

1. 異なるポート (たとえば、9 ～ 20 の範囲) で新しい VLAN プールを設定します。

2. ファイアウォールに接続されているリーフ ポートを含む新しい物理的なドメインを設定します。
3. ステップ 1 で設定した VLAN プールに物理的なドメインを関連付けます。
4. リーフ ポートの VLAN の範囲を `portLocal` として設定します。
5. 新しい EPG (この例ではファイアウォールが使用するもの) を、ステップ 2 で作成した物理ドメインに関連付けます。
6. リーフ ポートで EPG を展開します。

vPC に展開された EPG の VLAN ガイドライン

図 *13 : vPC の 2 つのレッグの VLAN*



EPG を vPC に展開する場合は、vPC の 2 つのレッグのリーフ スイッチ ポートに割り当てられた同じドメイン（同じ VLAN プール）に関連付ける必要があります。

この図では、EPG A は、リーフ スイッチ 1 およびリーフ スイッチ 2 のポートに展開されている vPC に展開されています。2 本のリーフ スイッチ ポートおよび EPG は、すべて同じ VLAN プールが含まれている同じドメインに関連付けられています。

特定のポートに EPG を導入する

GUI を使用して特定のノードまたはポートへ EPG を導入する

始める前に

EPG を導入するテナントがすでに作成されていること。

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

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 **[Tenants[** > ***[tenant]*** を選択します。

ステップ 3 左側のナビゲーション ウィンドウで、 ***tenant***、 **Application Profiles**、および ***application profile*** を展開します。

ステップ 4 **Application EPGs** を右クリックし、**Create Application EPG** を選択します。

ステップ 5 **Create Application EPG STEP 1 > Identity** ダイアログボックスで、次の操作を実行します:

a) **Name** フィールドに、EPG の名前を入力します。

b) **Bridge Domain** ドロップダウンリストから、ブリッジ ドメインを選択します。

c) [Statically Link with Leaves/Paths] チェックボックスをオンにします。

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

d) [Next] をクリックします。

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

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

ステップ 7 次のいずれかの手順を実行します。

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

オプション	説明
	5. **Port Encap** フィールドに、導入するセカンダリ VLAN を入力します。 6. (オプション) **Primary Encap** フィールドで、展開するプライマリ VLAN を入力します。

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

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

ステップ 10 次のいずれかの操作を実行します:

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

NX-OS スタイルの CLI を使用した APIC の特定のポートへの EPG の導入

手順

ステップ 1 VLAN ドメインを設定します。

例：

```
apic1(config)# vlan-domain dom1
apic1(config-vlan)# vlan 10-100
```

ステップ 2 テナントを作成します。

例：

```
apic1# configure
apic1(config)# tenant t1
```

ステップ 3 プライベート ネットワーク/VRF を作成します。

例：

```
apic1(config-tenant)# vrf context ctx1
apic1(config-tenant-vrf)# exit
```

ステップ 4 ブリッジ ドメインを作成します。

例：

```
apic1(config-tenant)# bridge-domain bd1
apic1(config-tenant-bd)# vrf member ctx1
apic1(config-tenant-bd)# exit
```

ステップ 5 アプリケーション プロファイルおよびアプリケーション EPG を作成します。

例：

```
apic1(config-tenant)# application AP1
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# bridge-domain member bd1
apic1(config-tenant-app-epg)# exit
apic1(config-tenant-app)# exit
apic1(config-tenant)# exit
```

ステップ 6 EPG を特定のポートに関連付けます。

例：

```
apic1(config)# leaf 1017
apic1(config-leaf)# interface ethernet 1/13
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf-if)# switchport trunk allowed vlan 20 tenant t1 application AP1 epg
EPG1
```

（注） 上の例に示した vlan-domain コマンドと vlan-domain member コマンドは、ポートに EPG を導入するための前提条件です。

REST API を使用した APIC の特定のポートへの EPG の導入

始める前に

EPG を導入するテナントが作成されていること。

手順

特定のポート上に EPG を導入します。

例：

```
<fvTenant name="<tenant_name>" dn="uni/tn-test1" >
    <fvCtx name="<network_name>" pcEnfPref="enforced" knwMcastAct="permit"/>
    <fvBD name="<bridge_domain_name>" unkMcastAct="flood" >
        <fvRsCtx tnFvCtxName="<network_name>"/>
    </fvBD>
    <fvAp name="<application_profile>" >
        <fvAEPg name="<epg_name>" >
           <fvRsPathAtt tDn="topology/pod-1/paths-1017/pathep-[eth1/13]" mode="regular"
 instrImedcy="immediate" encap="vlan-20"/>
        </fvAEPg>
    </fvAp>
</fvTenant>
```

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

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

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

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

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

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

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

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

始める前に

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

手順

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

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

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

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

a) **[ノード タイプ（Node Type）]** で、**[リーフ（Leaf）]** をクリックします。

b) **[ポート タイプ（Port Type）]** で、**[アクセス（Access）]** をクリックします。

c) **[インターフェイス タイプ（Interface Type）]** で、目的のタイプを選択します。

d) **[インターフェイス集約タイプ（Interface Aggregation Type）]** で、**[個別（Individual）]** を選択します。

e) **[ノード（Node）]** で、**[ノードの選択（Select Node）]** をクリックし、目的のノードのボックスにチェックを入れて、**[OK]** をクリックします。複数のノードを選択できます。

f) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。

g) **[リーフ アクセス ポート ポリシー グループ（Leaf Access Port Policy Group）]** の場合は、**[リーフ アクセス ポート ポリシー グループの選択（Select Leaf Access Port Policy Group）]** をクリックします。

h) **[リーフ アクセス ポート ポリシー グループの選択（Select Leaf Access Port Policy Group）]** ダイアログで、**[リーフ アクセス ポート ポリシー グループの作成（Create Leaf Access Port Policy Group）]** をクリックします。

i) **[リーフ アクセス ポート ポリシー グループの作成（Create Leaf Access Port Policy Group）]** ダイアログの **[リンク レベル ポリシー（Link Level Policy）]** で、**[リンク レベル ポリシーの選択（Select Link Level Policy）]** をクリックします。

j) リンク レベル ポリシーを選択して **[選択（Select）]** を選択するか、**[リンク レベル ポリシーの作成（Create Link Level Policy）]** をクリックし、必要に応じてフィールドに入力して、**[保存（Save）]** をクリックします。

k) [保存（Save）] をクリックします。

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

a) **[ナビゲーション（Navigation）]** ペインで、**[物理ドメインと外部ドメイン（Physical and External Domains）]** を展開します。

b) **[物理ドメイン（Physical Domains）]** を右クリックし、適切な**[物理ドメインの作成（Create Physical Domain）]** を選択します。

c) **[名前（Name）]** に、ドメインの名前を入力します。

d) **[VLAN プール（VLAN Pool）]**で、**[VLAN プールの作成（Create VLAN Pool）]** を選択し、必要に応じてフィールドに入力して、**[送信（Submit）]** をクリックします。

e) 目的に応じて、残りのフィールドに入力します。

f) [送信（Submit）] をクリックします。

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

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

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

a) **[ドメイン（Domains）（VM またはベアメタル）]** を右クリックし、**[物理ドメインの関連付けの追加（Add Physical Domain Association）]** をクリックします。

b) **[物理ドメインの関連付けの追加（Add Physical Domain Association）]** ダイアログで、**[物理ドメインのプロファイル（Physical Domain Profile）]** ドロップダウン リストから、前に作成したドメインを選択します。

c) [送信（Submit）] をクリックします。
AEP は、ノード上の特定のポート、およびドメインに関連付けられます。物理ドメインは VLAN プールに関連付けられ、テナントはこの物理ドメインに関連付けられます。

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

NX-OS スタイルの CLI を使用した、EPG を特定のポートに導入するための AEP、ドメイン、および VLAN の作成

始める前に

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

手順

ステップ 1 VLAN ドメインを作成し、VLAN 範囲を割り当てます。

例：

```
apic1(config)# vlan-domain domP
apic1(config-vlan)# vlan 10
apic1(config-vlan)# vlan 25
apic1(config-vlan)# vlan 50-60
apic1(config-vlan)# exit
```

ステップ 2 インターフェイス ポリシー グループを作成し、そのポリシー グループに VLAN ドメインを割り当てます。

例：

```
apic1(config)# template policy-group PortGroup
apic1(config-pol-grp-if)# vlan-domain member domP
```

ステップ 3 リーフ インターフェイス プロファイルを作成し、そのプロファイルにインターフェイス ポリシー グループを割り当てて、そのプロファイルを適用するインターフェイス ID を割り当てます。

例：

```
apic1(config)# leaf-interface-profile InterfaceProfile1
apic1(config-leaf-if-profile)# leaf-interface-group range
apic1(config-leaf-if-group)# policy-group PortGroup
apic1(config-leaf-if-group)# interface ethernet 1/11-13
apic1(config-leaf-if-profile)# exit
```

ステップ 4 リーフ プロファイルを作成し、そのリーフ プロファイルにリーフ インターフェイス プロファイルを割り当てて、そのプロファイルを適用するリーフ ID を割り当てます。

例：

```
apic1(config)# leaf-profile SwitchProfile-1019
apic1(config-leaf-profile)# leaf-interface-profile InterfaceProfile1
apic1(config-leaf-profile)# leaf-group range
apic1(config-leaf-group)# leaf 1019
apic1(config-leaf-group)#
```

REST API を使用した、EPG を特定のポートに導入するための AEP、ドメイン、および VLAN の作成

始める前に

- EPG を導入するテナントがすでに作成されていること。

• EPG は特定のポートに静的に導入されます。

手順

ステップ 1 インターフェイス プロファイル、スイッチ プロファイル、および接続エンティティ プロファイル（AEP）を作成します。

例：

```
   <infraInfra>

     <infraNodeP  name="<switch_profile_name>" dn="uni/infra/nprof-<switch_profile_name>"
 >
           <infraLeafS  name="SwitchSeletor" descr="" type="range">
                <infraNodeBlk name="nodeBlk1" descr="" to_="1019" from_="1019"/>
           </infraLeafS>
          <infraRsAccPortP tDn="uni/infra/accportprof-<interface_profile_name>"/>
      </infraNodeP>

      <infraAccPortP name="<interface_profile_name>"
dn="uni/infra/accportprof-<interface_profile_name>" >
           <infraHPortS  name="portSelector"  type="range">
               <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-<port_group_name>"
 fexId="101"/>
               <infraPortBlk name="block2"  toPort="13" toCard="1" fromPort="11"
fromCard="1"/>
          </infraHPortS>
     </infraAccPortP>

    <infraAccPortGrp  name="<port_group_name>"
dn="uni/infra/funcprof/accportgrp-<port_group_name>" >
          <infraRsAttEntP tDn="uni/infra/attentp-<attach_entity_profile_name>"/>
          <infraRsHIfPol tnFabricHIfPolName="1GHifPol"/>
    </infraAccPortGrp>

    <infraAttEntityP  name="<attach_entity_profile_name>"
dn="uni/infra/attentp-<attach_entity_profile_name>" >
         <infraRsDomP tDn="uni/phys-<physical_domain_name>"/>
    </infraAttEntityP>

<infraInfra>
```

ステップ 2 ドメインを作成する。

例：

```
<physDomP  name="<physical_domain_name>" dn="uni/phys-<physical_domain_name>">
    <infraRsVlanNs tDn="uni/infra/vlanns-[<vlan_pool_name>]-static"/>
</physDomP>
```

ステップ 3 VLAN 範囲を作成します。

例：

```
<fvnsVlanInstP  name="<vlan_pool_name>" dn="uni/infra/vlanns-[<vlan_pool_name>]-static"
  allocMode="static">
    <fvnsEncapBlk name="" descr="" to="vlan-25" from="vlan-10"/>
</fvnsVlanInstP>
```

ステップ 4 ドメインに EPG を関連付けます。

例：

```
<fvTenant  name="<tenant_name>" dn="uni/tn-" >
    <fvAEPg prio="unspecified" name="<epg_name>" matchT="AtleastOne"
dn="uni/tn-test1/ap-AP1/epg-<epg_name>" descr="">
        <fvRsDomAtt tDn="uni/phys-<physical_domain_name>" instrImedcy="immediate"
resImedcy="immediate"/>
    </fvAEPg>
</fvTenant>
```

重複する VLAN の検証

このグローバル機能は、単一の EPG での重複する VLAN プールの関連付けを防止します。APIC のいずれかの EPG で重複するプールが割り当てられている場合、この機能は有効にできません（有効にしようとするとエラーが表示されます）。既存の重複プールが存在しない場合は、この機能を有効にできます。有効にすると、EPGにドメインを割り当てることを試行し、そのドメインに、EPG にすでに関連付けられている別のドメインと重複する VLAN プールが含まれていた場合、設定はブロックされます。

重複する VLAN プールが EPG の下に存在する場合、各スイッチによって EPG に割り当てられる FD VNID は非確定的になり、異なるスイッチが異なる VNID を割り当てる場合があります。これにより、vPC ドメイン内のリーフ間で EPM 同期が失敗する可能性が生じます（EPG 内のすべてのエンドポイントの接続が断続的になります）。また、ユーザーが EPG 間で STP を拡張している場合、FDVNIDの不一致によりスイッチ間でBPDUがドロップされるため、ブリッジング ループが発生する可能性もあります。

GUI を使用した重複 VLAN の検証

この手順では、APIC GUI を使用して VLAN のオーバーラップの検証を設定する例を示します。

手順

ステップ 1 メニュー バーで、**[システム（System）] > [システム設定（System Settings）]** を選択します。

ステップ 2 ナビゲーション ペインで、**[ファブリック ワイドの設定（Fabric Wide Setting）]** を選択します。

ステップ 3 作業ウィンドウで、**[EPG VLAN 検証の適用（Enforce EPG VLAN Validation）]** を見つけてオンにします。

（注） 重複する VLAN プールがすでに存在し、このパラメータがオンになっている場合、システムはエラーを返します。この機能を選択する前に、EPG に重複しない VLAN プールを割り当てる必要があります。

このパラメータをオンにして、重複するVLANプールをEPGに追加しようとすると、エラーが返されます。

ステップ 4 [送信（Submit）] をクリックします。

REST API を使用した重複 VLAN の検証

この手順では、REST API を使用して VLAN の重複の検証を設定する例を示します。

手順

ステップ 1 XML API を使用して検証を有効にするには、この HTTP POST メッセージを送信します。

例：

```
POST https://apic-ip-address/api/mo/infra/settings.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：

```
<?xml version="1.0" encoding="UTF-8"?>
<infraSetPol validateOverlappingVlans=yes />'
```

（注） 重複する VLAN がすでに存在する場合、POST 中にエラーメッセージが表示され、重複する VLAN を持つ対応する EPG が示されます。

```
<?xml version="1.0" encoding="UTF-8"?><imdata totalCount="1">
<error code="100" text="Validation failed: Vlan ranges for an EPg cannot overlap
 Dn0=uni/tn-ag/ap-app/epg-e1,
"/></imdata>
```

添付されているエンティティ プロファイルで複数のインターフェイスに EPG を導入する

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

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

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

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

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

始める前に

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

手順

ステップ 1 ターゲットの接続エンティティ プロファイルに移動します。

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

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

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

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

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

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

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

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

NX-OS スタイルの CLI を使用したインターフェイス ポリシー グループによる複数のインターフェイスへの EPG の導入

NX-OS CLI では、接続エンティティ プロファイルを EPG に関連付けることによる迅速な導入が明示的に定義されていません。代わりにインターフェイス ポリシー グループが定義されてドメインが割り当てられます。このポリシー グループは、VLAN に関連付けられたすべてのポートに適用され、その VLAN を介して導入されるアプリケーション EPG を含むように設定されます。

始める前に

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

手順

ステップ 1 ターゲット EPG をインターフェイス ポリシー グループに関連付けます。

このコマンド シーケンスの例では、VLAN ドメイン **domain1** と VLAN **1261** に関連付けられたインターフェイス ポリシー グループ **pg3** を指定します。このポリシー グループに関連付けられたすべてのインターフェイスに、アプリケーション EPG **epg47** が導入されます。

例：

```
 apic1# configure terminal
apic1(config)# template policy-group pg3
apic1(config-pol-grp-if)# vlan-domain member domain1
apic1(config-pol-grp-if)# switchport trunk allowed vlan 1261 tenant tn10 application
pod1-AP
      epg epg47
```

ステップ 2 ターゲット ポートで、アプリケーション EPG に関連付けられたインターフェイス ポリシー グループのポリシーが導入されたことを確認します。

次の **show** コマンド シーケンスの出力例は、ポリシー グループ **pg3** がリーフ スイッチ **1017** 上のイーサネット ポート **1/20** に導入されていることを示しています。

例：

```
 apic1# show run leaf 1017 int eth 1/20
# Command: show running-config leaf 1017 int eth 1/20
# Time: Mon Jun 27 22:12:10 2016
  leaf 1017
    interface ethernet 1/20
      policy-group pg3
      exit
    exit
ifav28-ifc1#
```

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

AEP のインターフェイス セレクタを使用して、AEPg の複数のパスを設定できます。以下を選択できます。

1. ノードまたはノード グループ
2. インターフェイスまたはインターフェイス グループ

   インターフェイスは、インターフェイス ポリシー グループ（および `infra:AttEntityP`）を使用します。
3. infra:AttEntityP を AEPg に関連付けることで、使用する VLAN を指定する。

   infra:AttEntityP は、VLAN が異なる複数の AEPg に関連付けることができます。

3 のように infra:AttEntityP を AEPg に関連付けた場合、1 で選択したノード上の 2 のインターフェイスに、3 で指定した VLAN を使用して AEPg が導入されます。

この例では、AEPg `uni/tn-Coke/ap-AP/epg-EPG1` が、ノード 101 および 102 のインターフェイス 1/10、1/11、および 1/12 に vlan-102 で導入されます。

始める前に

- ターゲット アプリケーション EPG（`AEPg`）を作成する。
- 接続エンティティ プロファイル（AEP）による EPG 導入に使用する VLAN の範囲が含まれている VLAN プールを作成する。
- 物理ドメインを作成して VLAN プールおよび AEP にリンクさせる。

手順

選択したノードとインターフェイスに AEPg を導入するには、次の例のような XML を POST 送信します。

例：

```
 <infraInfra dn="uni/infra">
   <infraNodeP name="NodeProfile">
       <infraLeafS name="NodeSelector" type="range">
          <infraNodeBlk name="NodeBlok" from_="101" to_="102"/>
          <infraRsAccPortP tDn="uni/infra/accportprof-InterfaceProfile"/>
        </infraLeafS>
   </<infraNodeP>

   <infraAccPortP name="InterfaceProfile">
      <infraHPortS name="InterfaceSelector" type="range">
          <infraPortBlk name=" InterfaceBlock" fromCard="1" toCard="1" fromPort="10"
toPort="12"/>
          <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-PortGrp" />
      </infraHPortS>
   </infraAccPortP>

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

   <infraAttEntityP name="AttEntityProfile" >
       <infraGeneric name="default" >
          <infraRsFuncToEpg tDn="uni/tn-Coke/ap-AP/epg-EPG1" encap="vlan-102"/>
       </infraGeneric>
   </infraAttEntityP>
</infraInfra>
```

EPG 内の分離

EPG 内エンドポイント分離

EPG内エンドポイント分離ポリシーにより、仮想エンドポイントまたは物理エンドポイントが完全に分離されます。分離を適用した状態で稼働している EPG 内のエンドポイント間の通信は許可されません。分離を適用した EGP では、多くのクライアントが共通サービスにアクセスするときに必要な EPG カプセル化の数は低減しますが、相互間の通信は許可されません。

EPG の分離は、すべての Cisco Application Centric Infrastructure（ACI）ネットワーク ドメインに適用されるか、どれにも適用されないかの、どちらかになります。Cisco ACI ファブリックは接続エンドポイントに直接分離を実装しますが、ファブリックに接続されているスイッチはプライマリ VLAN（PVLAN）タグに従って分離規則を認識します。

（注） EPG 内エンドポイント分離を適用して EPG を設定した場合は、次の制限が適用されます。

- 分離を適用した EPG 全体のすべてのレイヤ 2 エンドポイント通信がブリッジ ドメイン内にドロップされます。
- 分離を適用した EPG 全体のすべてのレイヤ 3 エンドポイント通信が同じサブネット内にドロップされます。
- トラフィックが、分離が適用されている EPG から分離が適用されていない EPG に流れている場合、QoS CoS の優先順位設定の保持はサポートされません。

BPDU は、EPG 内分離が有効になっている EPG を介して転送されません。したがって、Cisco ACI 上の独立した EPG にマッピングされている VLAN でスパニング ツリーを実行する外部レイヤ 2 ネットワークを接続すると、Cisco ACI は外部ネットワークのスパニング ツリーがレイヤ2ループを検出できなくなる可能性があります。この問題を回避するには、これらの VLAN 内の Cisco ACI と外部ネットワーク間に単一の論理リンクのみを設定します。

ベア メタル サーバの EPG 内分離

ベア メタル サーバの EPG 内分離

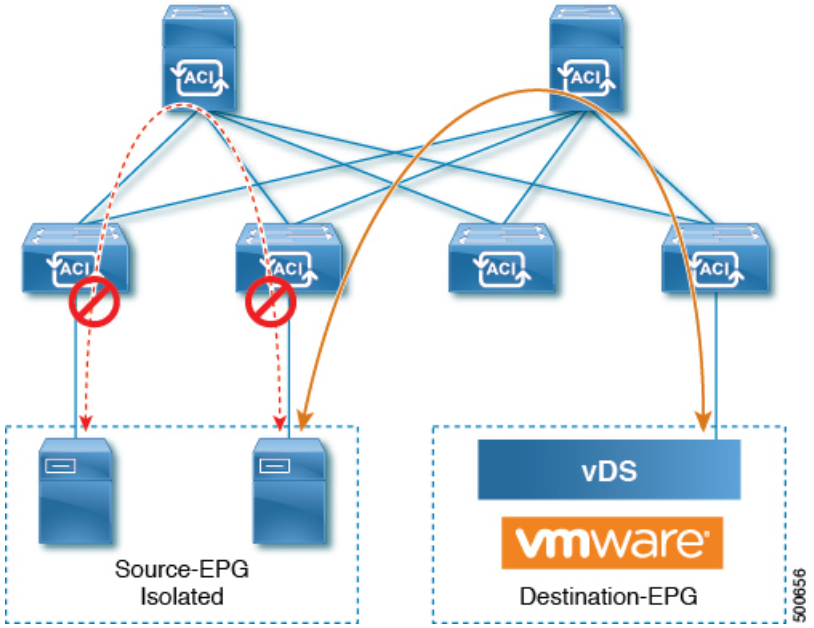
EPG 内エンドポイント分離のポリシーは、ベア メタル サーバなどの直接接続されているエンドポイントに適用できます。

次のような使用例があります。

- バックアップ クライアントは、バックアップ サービスにアクセスするための通信要件は同じですが、相互に通信する必要はありません。

- ロードバランサの背後にあるサーバの通信要件は同じですが、それらのサーバを相互に分離すると、不正アクセスや感染のあるサーバに対して保護されます。

図 *14* : ベア メタル サーバの *EPG* 内分離



ベア メタル の EPG 分離はリーフ スイッチで適用されます。ベア メタル サーバは VLAN カプセル化を使用します。ユニキャスト、マルチキャスト、およびブロードキャストのすべてのトラフィックが、分離が適用された EPG 内でドロップ（拒否）されます。ACI ブリッジ ドメインには、分離された EPG と通常の EPG を混在させることができます。分離された EPG それぞれには、VLAN 間トラフィックを拒否する複数の VLAN を指定できます。

GUI を使用したベア メタル サーバの EPG 内分離の設定

EPG が使用するポートは、リーフ スイッチにベア メタル サーバを直接接続するために使用する物理ドメイン内のベア メタル サーバと関連付ける必要があります。

手順

ステップ 1 テナントで、[Application Profile] を右クリックし、[Create Application EPG] ダイアログ ボックスを開いて次の操作を実行します。

a) [Name] フィールドに、EPG の名前（intra_EPG-deny）を追加します。
b) [Intra EPG Isolation] で、[Enforced] をクリックします。
c) [Bridge Domain] フィールドで、ドロップダウン リストからブリッジ ドメイン（bd1）を選択します。
d) [Statically Link with Leaves/Paths] チェックボックスをオンにします。
e) [Next] をクリックします。

ステップ 2 [Leaves/Paths] ダイアログボックスで、次の操作を実行します。

a) [Path] セクションで、ドロップダウン リストからトランク モードでのパス（Node-107/eth1/16）を選択します。

セカンダリ VLAN の [Port Encap]（vlan-102）を指定します。

（注） ベア メタル サーバがリーフ スイッチに直接接続されている場合、Port Encap のセカンダリ VLAN のみが指定されます。

プライマリ VLAN の [Primary Encap]（vlan-103）を指定します。

b) [Update] をクリックします。

c) **[終了]** をクリックします。

NX-OS スタイルの CLI を使用したベア メタル サーバの EPG 内分離の設定

手順

	コマンドまたはアクション	目的
ステップ 1	CLI で、EPG 内分離 EPG を作成します。 **例**： 以下に、VMM ケースを示します。	

```
ifav19-ifc1(config)# tenant
Test_Isolation
ifav19-ifc1(config-tenant)# application
 PVLAN
ifav19-ifc1(config-tenant-app)# epg
EPG1
ifav19-ifc1(config-tenant-app-epg)#
show running-config
# Command: show running-config
tenant Test_Isolation
 application PVLAN epg EPG1
  tenant Test_Isolation
     application PVLAN
      epg EPG1
        bridge-domain member BD1
        contract consumer bare-metal
        contract consumer default
        contract provider Isolate_EPG

       isolation enforce   <---- This
 enables EPG isolation mode.
        exit
    exit
ifav19-ifc1(config)# leaf ifav19-leaf3
ifav19-ifc1(config-leaf)# interface
ethernet 1/16
ifav19-ifc1(config-leaf-if)# show
running-config
ifav19-ifc1(config-leaf-if)# switchport
 trunk native vlan 101 tenant
```

	コマンドまたはアクション	目的
	Test_Isolation application PVLAN epg StaticEPG primary-vlan 100 exit	
ステップ 2	設定を確認します。 例： show epg StaticEPG detail Application EPg Data: Tenant : Test_Isolation Application : PVLAN AEPg : StaticEPG BD : BD1 uSeg EPG : no Intra EPG Isolation : **enforced** Vlan Domains : phys Consumed Contracts : bare-metal Provided Contracts : default,Isolate_EPG Denied Contracts : Qos Class : unspecified Tag List : VMM Domains: Domain Type Deployment Immediacy Resolution Immediacy State Encap Primary Encap ---------- ---------- ---------- ---------- ---------- ---------- DVS1 VMware On Demand immediate formed auto auto Static Leaves: Node Encap Deployment Immediacy Mode Modification Time ---------- ---------- ---------- ---------- Static Paths: Node Interface Encap Modification Time ---------- ---------- ---------- 1018 eth101/1/1 vlan-100 2016-02-11T18:39:02.337-08:00 1019 eth1/16 vlan-101 2016-02-11T18:39:02.337-08:00	

	コマンドまたはアクション	目的
	Static Endpoints: Node Interface Encap End Point MAC End Point IP Address Modification Time ---------- -------------------------------- ---------- ---------- -------------------------------- --------------------------------	

REST API を使用したベア メタル サーバのイントラ EPG 分離の設定

始める前に

EPG が使用するポートは、物理ドメイン内のベア メタル サーバ インターフェイスに関連付けられ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://apic-ip-address/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="Tenant_BareMetal" >
  <fvAp name="Web">
    <fvAEPg name="IntraEPGDeny" pcEnfPref="enforced">
          <!-- pcEnfPref="enforced" ENABLES ISOLATION-->
          <fvRsBd tnFvBDName="bd" />
          <fvRsDomAtt tDn="uni/phys-Dom1" />
          <!-- PATH ASSOCIATION -->
          <fvRsPathAtt tDn="topology/pod-1/paths-1017/pathep-[eth1/2]" encap="vlan-51"
primaryEncap="vlan-100" instrImedcy='immediate'/>
    </fvAEPg>
 </fvAp>
</fvTenant>
```

VMware vDS の EPG 内分離

VMware VDS または Microsoft Hyper-V 仮想スイッチの EPG 分離

EPG 内分離は、同じベース EPG またはマイクロセグメント（uSeg）EPG にある物理または仮想エンドポイントデバイスが相互に通信しないようにするオプションです。デフォルトでは、同じEPGに含まれるエンドポイントデバイスは互いに通信することができます。しかし、EPG内のエンドポイント デバイスの別のエンドポイント デバイスからの完全な分離が望ましい状況が存在します。たとえば、同じ EPG 内のエンドポイント VM が複数のテナントに属している場合、またはウイルスが広がるのを防ぐために、EPG内の分離を実行することができます。

Cisco Application Centric Infrastructure（ACI）仮想マシン マネージャ（VMM）ドメインは、EPG内分離が有効になっている EPG ごとに、VMware VDS または Microsoft Hyper-V 仮想スイッチで分離 PVLAN ポート グループを作成します。ファブリック管理者がプライマリ カプセル化を指定するか、または EPG と VMM ドメインの関連付け時にファブリックが動的にプライマリ カプセル化を指定します。ファブリック管理者が VLAN pri 値とVLAN-sec 値を静的に選択すると、VMM ドメインによって VLAN-pri と VLAN-sec がドメイン プール内のスタティック ブロックの一部であることが検証されます。

プライマリ カプセル化は、EPG VLAN ごとに定義されます。EPG 内分離にプライマリ カプセル化を使用するには、次のいずれかの方法で展開する必要があります。

- プライマリ VLAN とセカンダリ VLAN で定義されたポートを異なるスイッチに分離します。EPG VLANはスイッチごとに作成されます。ポートカプセル化があり、EPGのスイッチ上のスタティック ポートのみの場合、プライマリ カプセル化は関連付けられません。
- ポートカプセル化のみを使用するスタティック ポートには別のカプセル化を使用します。これにより、プライマリカプセル化が関連付けられていない2番目のEPG VLANが作成されます。

  次の例では、プライマリ VLAN-1103 を持つ 2 つのインターフェイス（Eth1/1、Eth1/3）の出力トラフィックを考慮します。Eth1/1ポート カプセル化が VLAN-1132 に（VLAN-1130 から）変更されたため、Eth1/3とセカンダリ VLAN を共有しません。

```
Port encap with VLAN-1130 on Eth1/1
Eth1/1: Port Encap only VLAN-1130
Eth1/6: Primary VLAN-1103 and Secondary VLAN-1130


fab2-leaf3# show vlan id 53 ext

VLAN Name                              Encap            Ports
---- --------------------------------- ---------------- -------------------------
53   JT:jt-ap:EPG1-1                   vlan-1130        Eth1/1, Eth1/3

module-1# show sys int eltmc info vlan access_encap_vlan 1130
            vlan_id:             53   :::           isEpg:              1
         bd_vlan_id:             52   :::         hwEpgId:          11278
      srcpolicyincom:             0   :::       data_mode:              0
        accencaptype:             0   :::    fabencaptype:              2
       accencapval:           1130   :::     fabencapval:          12192
             sclass:          49154   :::         sglabel:             12
         sclassprio:              1   :::     floodmetptr:             13
         maclearnen:              1   :::       iplearnen:              1
```

```
          sclasslrnen:              1   :::   bypselffwdchk:              0
             qosusetc:              0   :::       qosuseexp:              0
             isolated:              1   :::   primary_encap:           1103
            proxy_arp:              0   :::       qinq core:              0
            ivxlan_dl:              0   :::       dtag_mode:              0
       is_service_epg:              0

Port encap changed to VLAN-1132 on Eth1/1
fab2-leaf3# show vlan id 62 ext

VLAN Name                             Encap            Ports
---- -------------------------------- ---------------- -------------------------
62   JT:jt-ap:EPG1-1                  vlan-1132        Eth1/1

module-1# show sys int eltmc info vlan access_encap_vlan 1132
           [SDK Info]:
              vlan_id:             62   :::           isEpg:              1
           bd_vlan_id:             52   :::         hwEpgId:          11289
       srcpolicyincom:              0   :::       data_mode:              0
         accencaptype:              0   :::    fabencaptype:              2
         accencapval:            1132   :::     fabencapval:          11224
               sclass:           49154   :::         sglabel:             12
           sclassprio:              1   :::     floodmetptr:             13
           maclearnen:              1   :::       iplearnen:              1
          sclasslrnen:              1   :::   bypselffwdchk:              0
             qosusetc:              0   :::       qosuseexp:              0
             isolated:              1   :::   primary_encap:              0
            proxy_arp:              0   :::       qinq core:              0
            ivxlan_dl:              0   :::       dtag_mode:              0
       is_service_epg:              0

fab2-leaf3# show vlan id 53 ext

VLAN Name                             Encap            Ports
---- -------------------------------- ---------------- -------------------------
53   JT:jt-ap:EPG1-1                  vlan-1130        Eth1/3

module-1# show sys int eltmc info vlan access_encap_vlan 1130
           [SDK Info]:
              vlan_id:             53   :::           isEpg:              1
           bd_vlan_id:             52   :::         hwEpgId:          11278
       srcpolicyincom:              0   :::       data_mode:              0
         accencaptype:              0   :::    fabencaptype:              2
         accencapval:            1130   :::     fabencapval:          12192
               sclass:           49154   :::         sglabel:             12
           sclassprio:              1   :::     floodmetptr:             13
           maclearnen:              1   :::       iplearnen:              1
          sclasslrnen:              1   :::   bypselffwdchk:              0
             qosusetc:              0   :::       qosuseexp:              0
             isolated:              1   :::   primary_encap:           1103
            proxy_arp:              0   :::       qinq core:              0
            ivxlan_dl:              0   :::       dtag_mode:              0
```

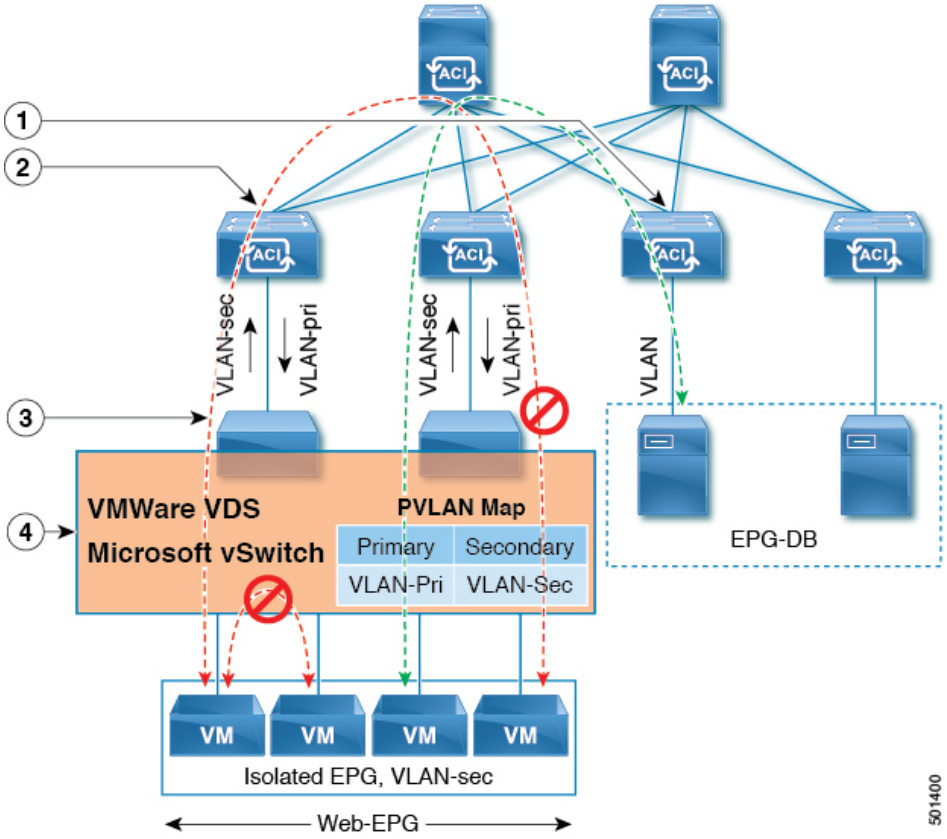
（注）
- イントラ EPG 隔離が強制されない場合、設定で指定されていても VLAN-pri 値は無視されます。
- EDM UCSM 統合を使用した VMware 分散仮想スイッチ（DVS）ドメインが失敗することがあります。ドメインに接続されているエンドポイントグループ（EPG）でEPG内分離を設定し、プライベート VLAN をサポートしない UCSM Mini 6324 を使用すると、ドメインに障害が発生します。

BPDU は、EPG 内分離が有効になっている EPG を介して転送されません。したがって、Cisco ACI 上の独立した EPG にマッピングされている VLAN でスパニング ツリーを実行する外部レイヤ 2 ネットワークを接続すると、Cisco ACI は外部ネットワークのスパニング ツリーがレイヤ2ループを検出できなくなる可能性があります。この問題を回避するには、これらのVLAN内の Cisco ACI と外部ネットワーク間に単一の論理リンクのみを設定します。

VMware VDS または Microsoft Hyper-V 仮想スイッチの VLAN-pri/VLAN-sec ペアは、EPG とドメインの関連付け中にVMM ドメインごとに選択されます。EPG内隔離EPGに作成されたポート グループは `PVLAN` に設定されたタイプでタグ付けされた VLAN-sec を使用します。VMware VDS または Microsoft Hyper-V 仮想スイッチおよびファブリックは、VLAN-pri/VLAN-sec カプセル化をスワップします。

- Cisco ACI ファブリックから VMware VDS または Microsoft Hyper-V 仮想スイッチへの通信は VLAN-pri を使用します。
- VMware VDS または Microsoft Hyper-V 仮想スイッチから Cisco ACI ファブリックへの通信は VLAN-sec を使用します。

図 *15 : VMware VDS* または *Microsoft Hyper-V* 仮想スイッチの *EPG* 分離



この図に関する次の詳細に注意してください。

1. EPG-DB は Cisco ACI リーフスイッチに VLAN トラフィックを送信します。Cisco ACI 出力リーフ スイッチは、プライマリ VLAN（PVLAN）タグを使用してトラフィックをカプセル化し、Web-EPG エンドポイントに転送します。

2. VMware VDS または Microsoft Hyper-V 仮想スイッチは、VLAN-sec を使用して Cisco ACI リーフ スイッチにトラフィックを送信します。 Web-EPG 内のすべての VLAN 内トラフィックに対して分離が適用されるため、Cisco ACI リーフ スイッチはすべての EPG 内トラフィックをドロップします。

3. Cisco ACI リーフ スイッチへの VMware VDS または Microsoft Hyper-V 仮想スイッチ VLAN-sec アップリンクが分離トランク モードです。Cisco ACI リーフ スイッチは、VMware VDS または Microsoft Hyper-V 仮想スイッチへのダウンリンク トラフィックに VLAN-pri を使用します。

4. PVLAN マップは、VMware VDS または Microsoft Hyper-V 仮想スイッチおよび Cisco ACI リーフ スイッチで設定されます。WEB-EPG からの VM トラフィックは VLAN-sec 内でカプセル化されます。VMware VDS または Microsoft Hyper-V 仮想スイッチは PVLAN タグに従ってローカルの WEB 内 EPG VM トラフィックを拒否します。すべての内部 ESXi ホストまたは Microsoft Hyper-V ホスト VM トラフィックは、VLAN-Sec を使用して Cisco ACI リーフ スイッチに送信されます。

関連情報

Cisco ACI 仮想エッジ環境での EPG 内分離の設定については、*Cisco ACI Virtual Edge Configuration Guide* の「Intra-EPG Isolation Enforcement for Cisco ACI Virtual Edge」の章を参照してください。

GUI を使用した VMware VDS または Microsoft Hyper-V の EPG 内分離の設定

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 **Tenants** > ***tenant*** を選択します。

ステップ 3 左側のナビゲーション ウィンドウで、**[アプリケーション プロファイル]** フォルダと適切なアプリケーション プロファイルを展開します。

ステップ 4 **Application EPGs** フォルダを右クリックし、**Create Application EPG** を選択します。

ステップ 5 **Create Application EPG** ダイアログ ボックスで、次の手順を実行します:

a) **Name** フィールドに EPG 名を追加します。

b) **Intra EPG Isolation** エリアで、**Enforced** をクリックします。

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

d) EPG をベア メタル/物理ドメイン インターフェイスまたは VM ドメインに関連付けます。

- VM ドメインの場合、[Associate to VM Domain Profiles] チェックボックスをオンにします。
- ベア メタルの場合、[Statically Link with Leaves/Paths] チェックボックスをオンにします。

e) [Next] をクリックします。

f) **Associated VM Domain Profiles** エリアで、+ アイコンをクリックします。

g) **Domain Profile** プロファイルのドロップダウン リストから、適切な VMM ドメインを選択します。

スタティックの場合、**Port Encap (or Secondary VLAN for Micro-Seg)** フィールドでセカンダリ VLAN を指定し、**Primary VLAN for Micro-Seg** フィールドで、プライマリ VLAN を指定します。Encap フィールドを空白のままにすると、値が動的に割り当てられます。

（注） スタティックの場合、スタティック VLAN を VLAN プールで使用できる必要があります。

ステップ 6 **Update** をクリックし、**Finish** をクリックします。

NX-OS スタイル CLI を使用した VMware VDS または Microsoft Hyper-V の EPG 内分離の設定

手順

ステップ 1 CLI で、EPG 内分離 EPG を作成します。

例：

次の例は VMware VDSの場合です:

```
apic1(config)# tenant Test_Isolation
apic1(config-tenant)# application PVLAN
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# show running-config
# Command: show running-config tenant Tenant_VMM application Web epg intraEPGDeny
  tenant Tenant_VMM
    application Web
      epg intraEPGDeny
        bridge-domain member VMM_BD
       vmware-domain member PVLAN encap vlan-2001 primary-encap vlan-2002 push on-demand

        vmware-domain member mininet
          exit
        isolation enforce
        exit
      exit
    exit
apic1(config-tenant-app-epg)#
```

例：

次の例は、Microsoft Hyper-V 仮想スイッチを示します。

```
apic1(config)# tenant Test_Isolation
apic1(config-tenant)# application PVLAN
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# show running-config
# Command: show running-config tenant Tenant_VMM application Web epg intraEPGDeny
  tenant Tenant_VMM
    application Web
      epg intraEPGDeny
        bridge-domain member VMM_BD
        microsoft-domain member domain1 encap vlan-2003 primary-encap vlan-2004
        microsoft-domain member domain2
          exit
        isolation enforce
        exit
      exit
    exit
apic1(config-tenant-app-epg)#
```

ステップ 2 設定を確認します。

例：

```
 show epg StaticEPG detail
Application EPg Data:
Tenant              : Test_Isolation
Application         : PVLAN
```

```
AEPg                : StaticEPG
BD                  : VMM_BD
uSeg EPG            : no
Intra EPG Isolation : enforced
Vlan Domains        : VMM
Consumed Contracts  : VMware_vDS-Ext
Provided Contracts  : default,Isolate_EPG
Denied Contracts    :
Qos Class           : unspecified
Tag List            :
VMM Domains:
Domain                 Type       Deployment Immediacy  Resolution Immediacy  State
      Encap       Primary
Encap
 --------------------  ---------  --------------------  --------------------
--------------  ----------  ----------
 DVS1                   VMware     On Demand             immediate             formed
       auto        auto

Static Leaves:
 Node        Encap             Deployment Immediacy  Mode                Modification
Time
----------  ----------------  --------------------  ------------------
------------------------------

Static Paths:
 Node        Interface                       Encap             Modification Time

 ----------  ------------------------------  ----------------
------------------------------
 1018        eth101/1/1                      vlan-100
2016-02-11T18:39:02.337-08:00
 1019        eth1/16                         vlan-101
2016-02-11T18:39:02.337-08:00

Static Endpoints:
 Node        Interface          Encap            End Point MAC      End Point IP Address
          Modification Time
 ----------  ------------------------------  ----------------  ------------------
------------------------------  ------------------------------

Dynamic Endpoints:
Encap: (P):Primary VLAN, (S):Secondary VLAN
 Node        Interface            Encap             End Point MAC      End Point IP
Address         Modification Time
 ----------  ------------------------------  ----------------  ------------------
------------------------------  ------------------------------
1017        eth1/3              vlan-943(P)       00:50:56:B3:64:C4  ---
         2016-02-17T18:35:32.224-08:00
                                vlan-944(S)
```

REST API を使用した VMware VDS または Microsoft Hyper-V バーチャル スイッチの EPG 内の分離の設定

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://apic-ip-address/api/mo/uni/tn-ExampleCorp.xml
```

ステップ 2 VMware VDS または Microsoft Hyper-V 仮想スイッチデプロイメントの場合は、POST メッセージの本文に次の XML 構造のいずれかを含めます。

例：

次の例は、VMware VDS の場合です。

```
<fvTenant name="Tenant_VMM" >
  <fvAp name="Web">
    <fvAEPg name="IntraEPGDeny" pcEnfPref="enforced">
          <!-- pcEnfPref="enforced" ENABLES ISOLATION-->
          <fvRsBd tnFvBDName="bd" />
          <!-- STATIC ENCAP ASSOCIATION TO VMM DOMAIN-->
          <fvRsDomAtt encap="vlan-2001" instrImedcy="lazy" primaryEncap="vlan-2002"
resImedcy="immediate" tDn="uni/vmmp-VMware/dom-DVS1”>
    </fvAEPg>
 </fvAp>
</fvTenant>
```

例：

次の例は、Microsoft Hyper-V の仮想スイッチの場合です。

```
<fvTenant name="Tenant_VMM" >
  <fvAp name="Web">
    <fvAEPg name="IntraEPGDeny" pcEnfPref="enforced">
          <!-- pcEnfPref="enforced" ENABLES ISOLATION-->
          <fvRsBd tnFvBDName="bd" />
         <!-- STATIC ENCAP ASSOCIATION TO VMM DOMAIN-->
       <fvRsDomAtt tDn="uni/vmmp-Microsoft/dom-domain1”>
 <fvRsDomAtt encap="vlan-2004" instrImedcy="lazy" primaryEncap="vlan-2003"
 resImedcy="immediate" tDn="uni/vmmp-Microsoft/dom-domain2”>
   </fvAEPg>
 </fvAp>
</fvTenant>
```

Cisco ACI 仮想エッジの EPG 内分離の設定

Cisco ACI Virtual Edge での EPG 内分離の適用

デフォルトでは、EPGに属するエンドポイントは契約が設定されていなくても相互に通信できます。ただし、相互に、EPG 内のエンドポイントを特定できます。たとえば、EPG 内でウイルスや他の問題を持つ VM が EPG の他の VM に影響を及ぼすことがないように、エンドポイント分離を適用するのが望ましい場合があります。

アプリケーション内のすべてのエンドポイントに分離を設定することも、いずれにも設定しないこともできます。一部のエンドポイントに分離を設定し、他のエンドポイントに設定しない方法は使用できません。

EPG 内のエンドポイントを分離しても、エンドポイントが別の EPG 内のエンドポイントと通信できるようにするコントラクトには影響しません。

（注） VLAN モードで Cisco ACI Virtual Edge ドメインと関連付けられている EPG での EPG内分離の適用はサポートされていません。このような EPG で EPG 内の分離を適用しようとすると、エラーがトリガーされます。

（注） Cisco ACI Virtual Edge マイクロセグメント (uSeg) EPG で EPG 内分離を使用することは現在のところサポートされていません。

（注） VXLAN カプセル化を使用し、EPG 内分離が適用されている Cisco ACI Virtual Edge EPG では、プロキシ ARP はサポートされていません従って、Cisco ACI Virtual Edge EPG 間で契約が設定されていても、EPG 内分離された EPG 間でサブネット間通信を行うことはできません。(VXLAN)。

GUI を使用した Cisco ACI Virtual Edge の EPG 内分離の設定

この手順に従って、EPG のエンドポイントが相互に分離されている EPG を作成します。

EPG が使用するポートは VM マネージャ（VMM）のいずれかに属している必要があります。

（注） この手順は、EPG の作成時に EPG 内のエンドポイントを分離することを前提としています。既存の EPG 内のエンドポイントを分離するには、Cisco APIC 内の EPG を選択し、[Properties] ペインの [Intra EPG Isolation] 領域で [Enforced] を選択して [SUBMIT] をクリックします。

始める前に

VXLAN 関連の設定が Cisco ACI Virtual Edge VMM ドメインに存在すること、特に Cisco ACI Virtual Edge ファブリック全体のマルチキャスト アドレスとマルチキャスト アドレスのプール (EPG ごとに 1 つ) が存在することを確認します。

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 [Tenants] を選択してテナントのフォルダを展開し、[Application Profiles] フォルダを展開します。

ステップ 3 アプリケーション プロファイルを右クリックし、[Create Application EPG] を選択します。

ステップ 4 [Create Application EPG] ダイアログボックスで、次の手順を実行します。

a) [Name] フィールドに EPG 名を入力します。

b) [Intra EPG Isolation] 領域で、[Enforced] をクリックします。

c) [Bridge Domain] ドロップダウン リストから、ブリッジ ドメインを選択します。

d) [Associate to VM Domain Profiles] チェックボックスをオンにします。

e) [Next] をクリックします。

f) **Associate VM Domain Profiles** エリアで、次の手順に従います:

- **+** (プラス) アイコンをクリックし、**Domain Profile** ドロップダウンリストから、対象とする Cisco ACI Virtual Edge VMM ドメインを選択します。
- **Switching Mode** ドロップダウンリストリストから、**AVE** を選択します。
- **Encap Mode** ドロップダウンリストから **VXLAN** または **Auto** を選択します。

  Auto を選択したら、Cisco ACI Virtual Edge VMM ドメインのカプセル化モードが VXLAN になっていることを確認します。
- (オプション) セットアップに適した他の設定オプションを選択します。

g) [Update] をクリックし、[Finish] をクリックします。

次のタスク

統計情報を選択して表示すると、エンドポイントが関与する問題の診断に役立ちます。このガイドの [Tenants] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を選択する （79 ページ） と [Tenents] タブの下で Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を表示する （79 ページ） を参照してください。

[Tenants] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を選択する

Cisco ACI Virtual Edge で EPG 内分離を設定した場合、拒否された接続数、受信パケット数、送信済みマルチキャストパケット数などのエンドポイントの統計情報を表示する前に、それらを選択する必要があります。その後、統計情報を表示できます。

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 **[Tenants]** > ***[tenant]*** の順に選択します。

ステップ 3 テナントのナビゲーション ウィンドウで、**Application Profiles**、***profile***、および **Application EPGs** フォルダを展開し、表示するエンドポイント統計情報を含む EPG を選択します。

ステップ 4 EPG の [Properties] 作業ペインで、[Operational] タブをクリックして EPG 内のエンドポイントを表示します。

ステップ 5 エンドポイントをダブルクリックします。

ステップ 6 エンドポイントの [Properties] ダイアログボックスで、[Stats] タブをクリックし、チェック アイコンをクリックします。

ステップ 7 **Select Stats** ダイアログボックスの **Available** ペインで、エンドポイントについて表示する統計情報を選択し、右向き矢印を使用してそれらの情報を **Selected** ペインに移動します。

ステップ 8 [送信（Submit）] をクリックします。

[Tenents] タブの下で Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を表示する

Cisco ACI Virtual Edge で EPG 内分離を設定していた場合には、エンドポイントの統計情報を選択すると、確認することができるようになります。

始める前に

分離エンドポイントについて表示する統計情報を選択しておく必要があります。手順については、このガイドの [Tenants] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を選択する （79 ページ） を参照してください。

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 **[Tenants]** > ***[tenant]*** の順に選択します。

ステップ 3 テナントのナビゲーション ウィンドウで、**Application Profiles**、***profile***、および **Application EPGs** フォルダを展開し、表示の必要な統計情報があるエンドポイントを含んでいる EPG を選択します。

ステップ 4 EPG の [Properties] 作業ペインで、[Operational] タブをクリックして EPG 内のエンドポイントを表示します。

ステップ 5 統計情報を表示するエンドポイントをダブルクリックします。

ステップ 6 エンドポイントの **Properties** 作業ウィンドウで、**Stats** タブをクリックします。

作業ウィンドウに、先ほど選択した統計情報が表示されます。作業ウィンドウの左上で、テーブル ビュー アイコンやチャート ビュー アイコンをクリックして、ビューを変更できます。

[Virtual Networking] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を選択する

Cisco ACI Virtual Edge で EPG 内分離を設定した場合、拒否された接続数、受信パケット数、送信済みマルチキャストパケット数などのエンドポイントの統計情報を表示する前に、それらを選択する必要があります。その後、統計情報を表示できます。

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 **Virtual Networking** > **Inventory** > **VMM Domains** > **VMware** > ***VMM domain*** > **Controllers** > ***controller instance name*** > ***DVS-VMM name*** > **Portgroups** > ***EPG name*** > ***Learned Point MAC address (node)*** > を選択します。

ステップ 3 [Stats] タブをクリックします。

ステップ 4 チェック マークが付いたタブをクリックします。

ステップ 5 **Select Stats** ダイアログボックスで、表示する統計情報を **Available**ペインでクリックし、右向き矢印をクリックして、それらを **Selected** ペインに移動します。

ステップ 6 (オプション)サンプリング間隔を選択します。

ステップ 7 [送信（Submit）] をクリックします。

[Virtual Networking] タブで Cisco ACI Virtual Edge の分離エンドポイント統計情報を表示する

Cisco ACI Virtual Edge で EPG 内分離を設定していた場合には、エンドポイントの統計情報を選択すると、確認することができるようになります。

始める前に

分離エンドポイントについて表示する統計情報を選択しておく必要があります。手順については、このガイドの[Tenants] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を選択する （79 ページ） を参照してください。

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 **Virtual Networking** > **Inventory** > **VMM Domains** > **VMware** > ***VMM name*** > **Controllers** > **controller instance name** > ***DVS-VMM name*** > **Portgroups** > ***EPG name*** > ***Learned Point MAC address (node)*** を選択します。

ステップ 3 [Stats] タブをクリックします。

中央のウィンドウに、先ほど選択した統計情報を表示します。作業ウィンドウの左上で、テーブル ビュー アイコンやチャート ビュー アイコンをクリックして、ビューを変更できます。

NX-OS スタイルの CLI を使用した Cisco ACI Virtual Edge の EPG 内分離の設定

始める前に

VXLAN に関連する設定に存在するかどうかを確認します Cisco ACI Virtual Edge VMM ドメイン、特に、 Cisco ACI Virtual Edge ファブリック全体のマルチキャスト アドレスと (EPG ごとに 1 つ) のマルチキャスト アドレスのプール。

手順

CLI で、EPG 内分離 EPG を作成します。

例：

```
# Command: show running-config tenant Tenant2 application AP-1 epg EPG-61
  tenant Tenant2
    application AP-1
      epg EPG-61
        bridge-domain member BD-61
        vmware-domain member D-AVE-SITE-2-3
          switching-mode AVE
          encap-mode vxlan
          exit
        isolation enforce            # This enables EPG into isolation mode.
        exit
```

```
    exit
  exit
```

次のタスク

統計情報を選択して表示すると、エンドポイントが関与する問題の診断に役立ちます。このガイドの [Tenants] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を選択する （79 ページ） と [Tenents] タブの下で Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を表示する （79 ページ） を参照してください。

REST API を使用した Cisco ACI Virtual Edge の EPG 内分離の設定

始める前に

VXLAN に関連する設定に存在するかどうかを確認します Cisco ACI Virtual Edge VMM ドメイン、特に、 Cisco ACI Virtual Edge ファブリック全体のマルチキャスト アドレスと (EPG ごとに 1 つ) のマルチキャスト アドレスのプール。

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://10.197.139.36/api/mo/uni/tn-Tenant2.xml
```

ステップ 2 VMM の導入では、POST メッセージの本文に次の例に示す XML 構造を含めます。

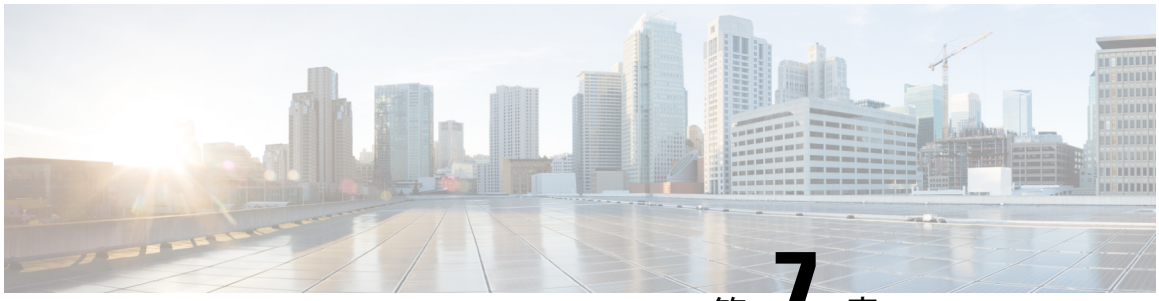
例：

```
<fvTenant name="Tenant2" >
  <fvAp name="AP-1">
    <fvAEPg name="EPG-61" pcEnfPref="enforced">
      <!-- pcEnfPref="enforced" ENABLES ISOLATION-->
      <!-- pcEnfPref="unenforced" DISABLES ISOLATION-->
        <fvRsBd tnFvBDName="BD-61" />
        <fvRsDomAtt switchingMode="AVE" encapMode="vxlan" resImedcy="immediate"
tDn="uni/vmmp-VMware/dom-D-AVE-SITE-1-XXIII" >
        </fvRsDomAtt>
    </fvAEPg>
  </fvAp>
</fvTenant>
```

次のタスク

統計情報を選択して表示すると、エンドポイントが関与する問題の診断に役立ちます。このガイドの [Tenants] タブの下で、Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を

選択する （79 ページ） と [Tenents] タブの下で Cisco ACI Virtual Edge の分離されたエンドポイントの統計情報を表示する （79 ページ） を参照してください。



第 7 章

アクセス インターフェイス

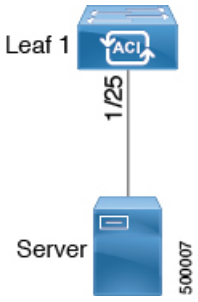
この章は、次の内容で構成されています。

物理ポート

ポリシーの関連付けを使用したリーフ スイッチ物理ポートの設定

この手順では、クイック スタート ウィザードを使用して、サーバーを Cisco Application Centric Infrastructure（ACI）リーフ スイッチ インターフェイスに接続します。手順は、Cisco ACI リーフ スイッチ インターフェイスに他の種類のデバイスを接続する場合と同じになります。

図 *16*：ベア メタル サーバのスイッチ インターフェイス設定



始める前に

- Cisco ACI ファブリックが設置され、APIC がオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが Cisco ACI ファブリックに登録され、使用可能であること。

手順

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

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

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

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

a) **[ノード タイプ（Node Type）]** で、**[リーフ（Leaf）]** をクリックします。

b) **[ポート タイプ（Port Type）]** で、**[アクセス（Access）]** をクリックします。

c) **[インターフェイス タイプ（Interface Type）]** で、目的のタイプを選択します。

d) **[インターフェイス集約タイプ（Interface Aggregation Type）]** で、**[個別（Individual）]** を選択します。

e) **[ノード（Node）]**で、**[ノードの選択（Select Node）]** をクリックし、目的のスイッチ（ノード）のボックスにチェックを入れ、**[OK]** をクリックします。複数のスイッチを選択できます。

f) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。

g) **[リーフ アクセス ポート ポリシー グループ（Leaf Access Port Policy Group）]** の場合は、**[リーフ アクセス ポート ポリシー グループの選択（Select Leaf Access Port Policy Group）]** をクリックします。

h) **[リーフ アクセス ポート ポリシー グループの選択（Select Leaf Access Port Policy Group）]** ダイアログで、**[リーフ アクセス ポート ポリシー グループの作成（Create Leaf Access Port Policy Group）]** をクリックします。

インターフェイス ポリシー グループは、選択したスイッチのインターフェイスに適用するインターフェイス ポリシーのグループを指定する名前付きポリシーです。インターフェイス ポリシーの例は、リンク レベルのポリシー（たとえば、1 gbit のポート速度）、ストーム制御インターフェイス ポリシーなどです。

i) **[リーフ アクセス ポート ポリシー グループの作成（Create Leaf Access Port Policy Group）]** ダイアログで、目的のポリシーを選択または作成します。

j) [保存（Save）] をクリックします。

次のタスク

これで、基本リーフ スイッチ インターフェイスの設定手順は完了しました。

（注） この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データトラフィックはフローできません。

ポート アソシエーションを使用してリーフ スイッチ物理ポートを構成する

この手順では、ACI リーフ スイッチ インターフェイスにサーバを接続する手順を示します。手順は、ACI リーフ スイッチ インターフェイスに他の種類のデバイスを接続する場合と同じになります。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。

手順

ステップ 1 APIC メニュー バーで、**[ファブリック]**>**[インベントリ]**>**[インベントリ]** に移動し、ポッドを選択して、**[設定]** タブに移動します。

スイッチのグラフィカル表示が表示されます。設定するポートを選択します。選択されると、強調表示されたポート設定タイプの形式で、ポート設定タイプが上部に表示されます。設定タイプを選択すると、その設定パラメータが表示されます。

ステップ 2 適切なフィールドを設定に割り当てたら、**[送信]** をクリックします。

この設定で、リーフ スイッチへのすべての変更は、ポートを選択しポリシーを適用することによって行われます。すべてのリーフ スイッチ設定は、このページの右側で行われます。

ポートを選択した し、 ポリシーを適用します。

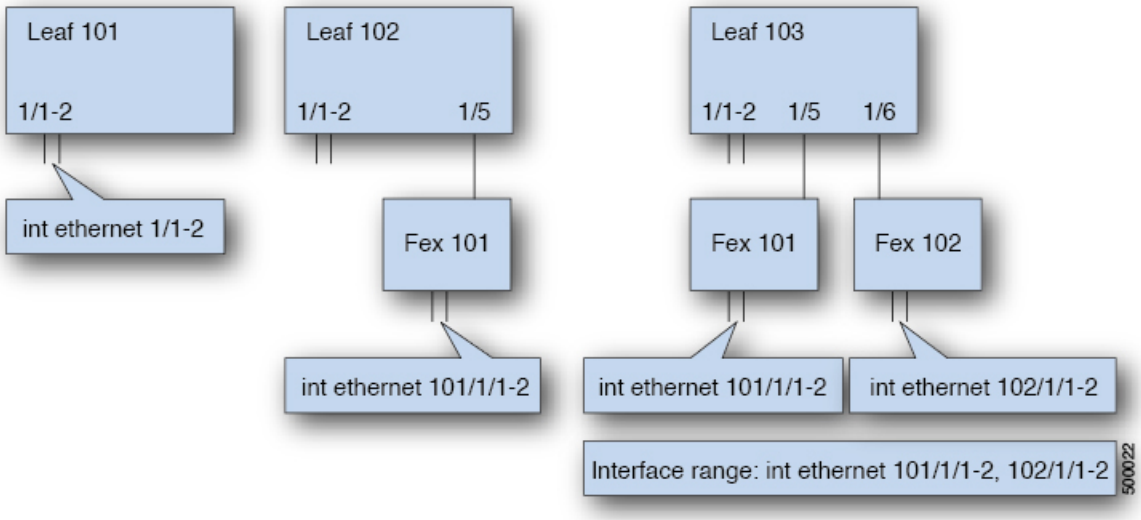
次のタスク

これで、基本リーフ インターフェイスの設定手順は完了しました。

NX-OS CLI を使用したリーフ ノードおよび FEX デバイス上の物理ポートの設定

次の例のコマンドでは、REST API/SDK および GUI と完全な互換性がある多数の管理対象オブジェクト（MO）を ACI ポリシー モデルで作成します。ただし、CLI ユーザは ACI モデル内部ではなく意図したネットワーク設定に注力できます。

次の図に、リーフ ノードに直接のイーサネット ポート、またはリーフ ノードに接続された FEX モジュールの例と、CLI でそれぞれがどのように表示されるのかを示します。FEX ポートでは、*fex-id* はポート自体の名前に **ethernet 101/1/1** として含まれます。インターフェイス範囲を記述する際は、**ethernet** キーワードを NX-OS で繰り返す必要はありません。例：**interface ethernet 101/1/1-2, 102/1/1-2**。



- リーフ ノードの ID 番号はグローバルです。
- *fex-id* 番号は各リーフにローカルです。
- キーワード **ethernet** の後のスペースに注意してください。

手順

ステップ 1 **configure**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ 2 **leaf** *node-id*

設定するリーフを指定します（複数可）。*node-id* には、設定の適用対象となる単一のノード ID、または ID の範囲を *node-id1−node-id2* という形式で指定できます。

例：

```
apic1(config)# leaf 102
```

ステップ 3 **interface** *type*

設定するインターフェイスを指定します。インターフェイス タイプと ID を指定できます。イーサネット ポートの場合は、「ethernet slot / port」を使用します。

例：

```
apic1(config-leaf)# interface ethernet 1/2
```

ステップ 4 （任意） **fex associate** *node-id*

設定するインターフェイスが FEX インターフェイスの場合、このコマンドを使用して、設定前に FEX モジュールをリーフ ノードに接続する必要があります。

（注） この手順は FEX ポートを使用してポートチャネルを作成する前に行う必要があります。

例：

```
apic1(config-leaf-if)# fex associate 101
```

ステップ 5 **speed** *speed*

ここでの速度設定は一例です。ここでは、以下の表に示す任意のインターフェイス設定を設定できます。

例：

```
apic1(config-leaf-if)# speed 10G
```

次の表に、この時点で設定できるインターフェイス設定を示します。

コマンド	目的
[no] shut	物理インターフェイスをシャットダウンします
[no] speed *speedValue*	物理インターフェイスの速度を設定します
[no] link debounce time *time*	リンク でバウンスを設定します
[no] negotiate auto	ネゴシエートを設定します
[no] cdp enable	Cisco Discovery Protocol（CDP）を無効または有効にします

コマンド	目的
[no] mcp enable	Mis-Cabling Protocol（MCP）を無効または有効にします
[no] lldp transmit	物理インターフェイスの送信を設定します
[no] lldp receive	物理インターフェイスの LLDP 受信を設定します
spanning-tree {bpduguard | bpdufilter} {enable | disable}	スパニング ツリー BPDU を設定します
[no] storm-control level *percentage* [burst-rate *percentage*]	ストーム制御（パーセント）を設定します
[no] storm-control pps *packets-per-second* burst-rate *packets-per-second*	ストーム制御（秒当たりのパケット）を設定します

例

リーフ ノードに 1 つのポートを設定します。次に、プロパティ speed、cdp、および admin state についてリーフ 101 のインターフェイス eth1/2 を設定する例を示します。

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# cdp enable
apic1(config-leaf-if)# no shut
```

複数のリーフ ノードの複数のポートを設定します。次に、リーフ ノード 101 ～ 103 のそれぞれのインターフェイス eth1/1-10 での速度設定の例を示します。

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface eth 1/1-10
apic1(config-leaf-if)# speed 10G
```

リーフ ノードに FEX を接続します。次に、リーフ ノードに FEX モジュールを接続する例を示します。NX-OS とは異なり、リーフ ポート Eth1/5 は暗黙的にファブリック ポートとして設定され、FEX ファブリック ポートチャネルは FEX アップリンク ポートで内部で作成されます。ACI では、FEX ファブリック ポートチャネルはデフォルト設定を使用し、ユーザ設定は使用できません。

（注） 次の例に示すように、この手順は FEX ポートを使用してポートチャネルを作成する前に行う必要があります。

```
apic1(config)# leaf 102
apic1(config-leaf)# interface eth 1/5
apic1(config-leaf-if)# fex associate 101
```

リーフ ノードに接続した FEX ポートを設定します。次に、リーフ ノード 102 ～ 103 のそれぞれに接続した FEX モジュール 101 のインターフェイス eth1/1-10 での速度設定の例を示します。FEX ID 101 はポート ID に含まれています。FEX ID は 101 から始まり、リーフに対してローカルです。

```
apic1(config)# leaf 102-103
apic1(config-leaf)# interface eth 101/1/1-10
apic1(config-leaf-if)# speed 1G
```

ポートのクローニング

ポート構成の複製

Cisco APIC リリース 3.2 以降では、ポート構成の複製がサポートされています。リーフ スイッチ ポートの設定後に、設定をコピーして、他のポートに適用することができます。これは、(NX-OS スタイル CLI) ではなく、APIC GUI でのみサポートされます。

ポート複製は、ファブリック アクセス ポリシーを使用して設定されたインターフェイスではなく、ファブリックの複数ノードで展開され、個別に設定された少数のリーフ スイッチ ポート (インターフェイス) で使用します。

ポート複製はレイヤ 2 の設定でのみサポートされます。

次のポリシーは、複製されたポートではサポートされません。

- 接続可能アクセス エンティティ
- ストーム制御
- DWDM
- MACsec

APIC GUI を使用した設定済みリーフ スイッチ ポートのクローニング

このタスクでは、以前に設定したリーフ スイッチ ポートを複製する方法について説明します。ポートの設定方法の詳細については、『*Cisco APIC Layer 2 Networking Configuration Guide*』を参照してください。

始める前に

Fabric > **Inventory** の下の GUI で、リーフ スイッチ ポート (サポートされるレイヤ 2 ポリシーのあるもの) を設定し、以下のいずれかを実行します:

- **Topology** > **Interface** > **Configuration Mode**
- **Pod** > **Interface** > **Configuration Mode**
- **Pod**> **Leaf** > **Interface** > **Configuration Mode**

手順

ステップ 1 メニュー バーで、**Fabric** > **Inventory** を選択します。

ステップ 2 元のポートを設定した場所に移動します。

ステップ 3 たとえば、**Pod** を展開して、**Leaf** を選択します。

ステップ 4 **Interface** をクリックし、ドロップダウンリストから **Configuration** を選択します (**Mode** にあります)。

ステップ 5 インターフェイスのメニュー バーの **+** のアイコンをクリックして、複製するポートが存在するリーフ スイッチを選択します。

ステップ 6 以前に設定したポートを右クリックし、**Copy** を選択します。

ステップ 7 設定をコピーするポートを右クリックし、**Paste** を選択します。

ポート チャネル

PC/vPC ホスト ロード バランシング アルゴリズム

次の表に、Cisco Application Centric Infrastructure（ACI）リーフ ノード ダウンリンクにわたるポートチャネル ロード バランシングで使用されるデフォルトのハッシュアルゴリズムと対称ハッシュ アルゴリズム オプションを示します。対称ハッシュ アルゴリズム オプションは、Cisco Application Policy Infrastructure Controller（APIC）リリース 2.3(1e) で導入されました。

表 *3 : PC/vPC* ホスト ロード バランシング アルゴリズム

Traffic Type	データ ポイントのハッシュ
エンド ホスト PC/vPC（デフォルト）	**レイヤ 2 トラフィック用**： • 送信元 MAC アドレス • 宛先 MAC アドレス • セグメント ID（VXLAN VNID）または VLAN ID **IP トラフィックの場合**： • 送信元 MAC アドレス • 宛先 MAC アドレス • 送信元 IP アドレス • 宛先 IP アドレス • プロトコル タイプ • 送信元レイヤ 4 ポート • 宛先レイヤ 4 ポート • セグメント ID（VXLAN VNID）または VLAN ID
PC 対称ハッシュ（構成可能）	**オプションを選択する**： • 送信元 IP アドレス • 宛先 IP アドレス • 送信元レイヤ 4 ポート • 宛先レイヤ 4 ポート

（注） 同じリーフ ノードで SIP/DIP/L4-src-port/L4-dest-port タイプを混在させないでください。次に例を示します。

以下は**サポートされています**。

- Po1：SIP のみで対称ハッシュを有効にします。
- Po2：対称ハッシュを有効にしません。デフォルトのハッシュを使用します。

以下は**サポートされていません**。

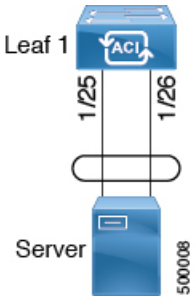
- Po1：SIP のみで対称ハッシュを有効にします。
- Po2：DIP のみで対称ハッシュを有効にします。

ポート チャネル ハッシュ アルゴリズムは、個々のリーフ ノードに個別に適用されます。アルゴリズムは、vPC ペアのリーフ ノードへのロード バランシングなど、ファブリック内のロードバランシングには影響しません。したがって、対称 EtherChannel ハッシュ機能は、vPC の場合にエンドツーエンドのトラフィックの対称性を保証しません。

GUI を使用した ACI リーフ スイッチのポート チャネルの構成

この手順では、クイック スタート ウィザードを使用して、サーバーを Cisco Application Centric Infrastructure（ACI）リーフ スイッチ インターフェイスに接続します。手順は、Cisco ACI リーフ スイッチ インターフェイスに他の種類のデバイスを接続する場合と同じになります。

図 *17:* スイッチ ポート チャネル設定



始める前に

- Cisco ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが Cisco ACI ファブリックに登録され、使用可能であること。

手順

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

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

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

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

a) **[ノード タイプ（Node Type）]** で、**[リーフ（Leaf）]** をクリックします。
b) **[ポート タイプ（Port Type）]** で、**[アクセス（Access）]** をクリックします。
c) **[インターフェイス タイプ（Interface Type）]** で、**[イーサネット（Ethernet）]** をクリックします。
d) **[インターフェイス集約タイプ（Interface Aggregation Type）]** で、**[PC]** を選択します。
e) **[ノード（Node）]**で、**[ノードの選択（Select Node）]** をクリックし、目的のスイッチ（ノード）のボックスにチェックを入れ、**[OK]** をクリックします。複数のスイッチを選択できます。
f) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。
g) **[PC/vPC インターフェイス ポリシー グループ（PC/vPC Interface Policy Group）]** の場合は、**[PC/vPC インターフェイス ポリシー グループの選択（Select PC/vPC Interface Policy Group）]** をクリックします。
h) **[PC/vPC インターフェイス ポリシー グループの選択（Select PC/vPC Interface Policy Group）]**ダイアログで、既存のポリシー グループを選択するか、**[PC/vPC インターフェイス ポリシー グループの作成（Create PC/vPC Interface Policy Group）]** をクリックして新しいポリシー グループを作成します。

インターフェイス ポリシー グループは、選択したスイッチのインターフェイスに適用するインターフェイス ポリシーのグループを指定する名前付きポリシーです。インターフェイス ポリシーの例は、リンクレベルのポリシー（たとえば、1 gbit のポート速度）、ストーム制御インターフェイス ポリシーなどです。

i) ポリシー グループの作成を選択した場合は、**[PC/vPC インターフェイス ポリシー グループの作成（Create PC/vPC Interface Policy Group）]** ダイアログで、フィールドに入力し、目的に応じてポリシーを選択または作成します。
j) [保存（Save）] をクリックします。

ステップ 5 [保存（Save）] をクリックします。

次のタスク

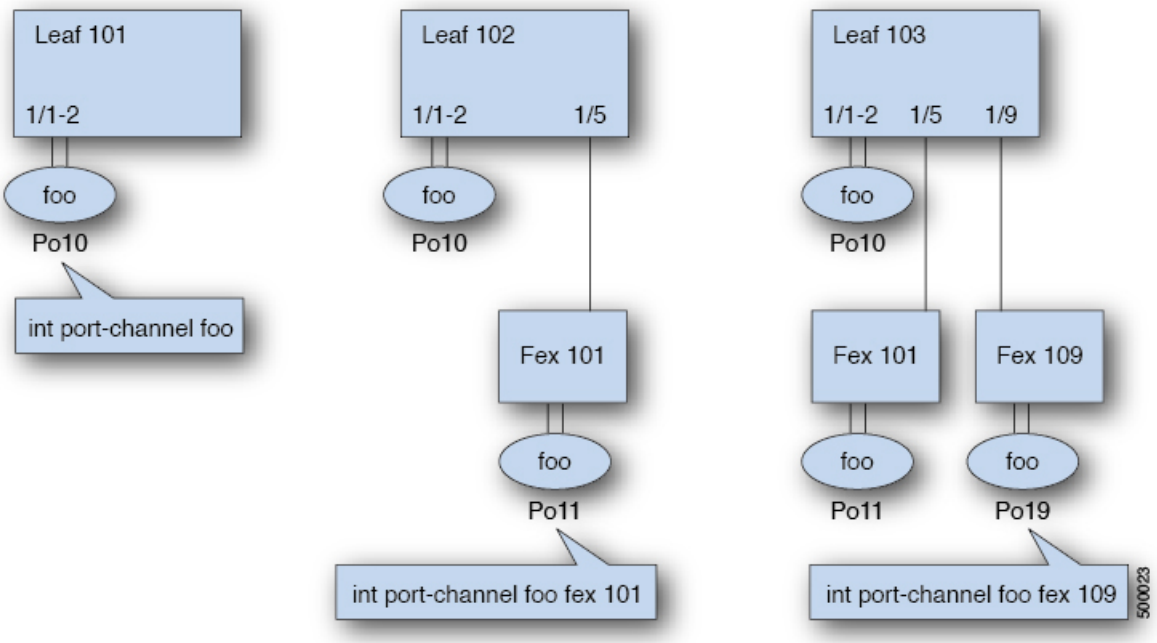
これで、ポート チャネルの設定手順は完了しました。

（注） この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データトラフィックはフローできません。

NX-OS CLI を使用したリーフ ノードおよび FEX デバイスのポートチャネルの設定

ポートチャネルは NX-OS の論理インターフェイスです。これは、複数の物理ポートのために帯域幅を集約するだけでなく、リンク障害時の冗長性を確保する目的でも使用されます。NX-OS におけるポートチャネル インターフェイスは、ノード内では一意となる、1 ～ 4096 の範囲でユーザが指定した番号によって識別されます。ポートチャネルインターフェイスは、(**interface port-channel** コマンドを使用して) 明示的に設定するか、または (**channel-group** コマンドを使用して) 暗黙的に作成します。ポートチャネル インターフェイスの設定は、ポートチャネルのすべてのメンバーポートに適用されます。特定の互換性パラメータ（速度など）は、メンバーポートでは設定できません。

ACI モデルでは、ポートチャネルは論理エンティティとして設定され、1 つ以上のリーフ ノードでポートセットに割り当てることができるポリシーのコレクションを表す名前によって識別されます。このような割り当てによって各リーフ ノードにポートチャネル インターフェイスが 1 個作成されます。これは、リーフ ノード内の 1 ～ 4096 の範囲で自動生成される番号によって識別されます。同じポートチャネル名を持つノード間では、番号を同じにすることも、分けることもできます。これらのポートチャネルのメンバーシップも同様に、同じにすることも分けることもできます。FEX ポート上にポートチャネルが作成されるときは、同じポートチャネル名を使用して、リーフ ノードに接続されている各 FEX デバイスに対して 1 つのポートチャネル インターフェイスを作成することができます。したがって、N 個の FEX モジュールに接続されている各リーフ ノードには最大で N+1 個の一意のポートチャネル インターフェイス（自動生成されるポートチャネル番号で識別される）を作成できます。これは以下の例で説明します。FEX ポートのポートチャネルは、*fex-id* とポートチャネル名を指定することによって識別されます（例：**interface port-channel foo fex 101**）。



- 各リーフが N 個の FEX ノードに接続されているときは、ポートチャネル foo のリーフごとに N+1 個のインスタンスが可能です。
- リーフ ポートおよび FEX ポートを同じポートチャネル インスタンスの一部にすることはできません。
- 各 FEX ノードはポートチャネル foo のインスタンスを 1 つだけ持つことができます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `apic1# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**template port-channel** *channel-name* 例： `apic1(config)# template port-channel foo`	新しいポートチャネルを作成するか、既存のポートチャネルを設定します（グローバル コンフィギュレーション）。
ステップ 3	[**no**] **switchport access vlan** *vlan-id* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name* 例： `apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg webEpg`	ポート チャネルを関連付けるすべてのポート上に VLAN を持つ EPG を導入します。

ステップ 4	**channel-mode active**
	例：
	`apic1(config-po-ch-if)# channel-mode active`
	（注） 対称ハッシュを有効にするには、**lacp symmetric-hash** コマンドを入力します。
	`apic1(config-po-ch-if)# lacp symmetric-hash`
		（注） **channel-mode** コマンドは、NX-OS の channel-group コマンドの mode オプションに相当します。ただし、ACI ではこれは（メンバー ポートではなく）ポートチャネルでサポートされます。
		対称ハッシュは、次のスイッチではサポートされていません。
		• Cisco Nexus 93128TX
		• Cisco Nexus 9372PX
		• Cisco Nexus 9372PX-E
		• Cisco Nexus 9372TX
		• Cisco Nexus 9372TX-E

	コマンドまたはアクション	目的
		• Cisco Nexus 9396PX • Cisco Nexus 9396TX
ステップ 5	**exit** 例： `apic1(config-po-ch-if)# exit`	設定モードに戻ります。
ステップ 6	**leaf** *node-id* 例： `apic1(config)# leaf 101`	設定するリーフ スイッチを指定します。*node-id* には、設定の適用対象となる単一のノード ID、または ID の範囲を *node-id1−node-id2* という形式で指定できます。
ステップ 7	**interface** *type* 例： `apic1(config-leaf)# interface ethernet 1/1-2`	ポートチャネルに設定するインターフェイスまたはインターフェイスの範囲を指定します。
ステップ 8	[**no**] **channel-group** *channel-name* 例： `apic1(config-leaf-if)# channel-group foo`	インターフェイスまたはインターフェイスの範囲をポートチャネルに割り当てます。ポートチャネルからインターフェイスを削除するには、キーワード **no** を使用します。インターフェイス上からポートチャネルの割り当てを変更する場合は、以前のポートチャネルからインターフェイスを最初に削除することなく **channel-group** コマンドを入力することができます。

ステップ 9	（任意） **lacp port-priority** *priority*
	例：
	`apic1(config-leaf-if)# lacp port-priority 1000`
	`apic1(config-leaf-if)# lacp rate fast`
		この設定とその他のポート単位の LACP プロパティは、この時点でポートチャネルのメンバー ポートに適用できます。
		（注） ACI モデルでは、これらのコマンドはポートがポート チャネルのメンバーになった後でのみ使用できます。ポートがポート チャネルから削除された場合、これらのポート単位のプロパティの設定も削除されます。

次の表に、ACI モデルでポート チャネル プロパティのグローバル コンフィギュレーションを行うためのさまざまなコマンドを示します。これらのコマンドは、（config-leaf-if）CLI モード

で特定のリーフのポートチャネルのオーバーライドを設定するためにも使用できます。ポートチャネル上から行った設定は、すべてのメンバー ポートに適用されます。

CLI 構文	機能
[no] speed <speedValue>	ポートチャネルの速度を設定します
[no] link debounce time <time>	ポートチャネルのリンク デバウンスを設定します
[no] negotiate auto	ポートチャネルのネゴシエートを設定します
[no] cdp enable	ポートチャネルの CDP を無効または有効にします
[no] mcp enable	ポートチャネルの MCP を無効または有効にします
[no] lldp transmit	ポートチャネルの送信を設定します
[no] lldp receive	ポートチャネルの LLDP 受信を設定します
spanning-tree <bpduguard | bpdufilter> <enable | disable>	スパニング ツリー BPDU を設定します
[no] storm-control level <percentage> [burst-rate <percentage>]	ストーム制御（パーセント）を設定します
[no] storm-control pps <packet-per-second> burst-rate <packets-per-second>	ストーム制御（秒当たりのパケット）を設定します
[no] channel-mode { active | passive | on| mac-pinning }	ポートチャネルのリンクの LACP モードを設定します
[no] lacp min-links <value>	リンクの最小数を設定します
[no] lacp max-links <value>	リンクの最大数を設定します
[no] lacp fast-select-hot-standby	ホット スタンバイ ポートの LACP 高速セレクトを設定します
[no] lacp graceful-convergence	LACP グレースフル コンバージェンスを設定します
[no] lacp load-defer	LACP ロード遅延メンバー ポートを設定します
[no] lacp suspend-individual	LACP 個別ポートの中断を設定します
[no] lacp port-priority	LACP ポート プライオリティ
[no] lacp rate	LACP レートを設定します

例

ポート チャネル（グローバル コンフィギュレーション）を設定します。速度およびチャネル モードの 2 つの設定を含むポリシーのコレクションを表す論理エンティティ「foo」を作成します。必要に応じてより多くのプロパティを設定できます。

（注） channel mode コマンドは、NX-OS の channel group コマンドの mode オプションに相当します。ただし、ACI ではこれは（メンバー ポートではなく）ポートチャネルでサポートされます。

```
apic1(config)# template port-channel foo
apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg
 webEpg
apic1(config-po-ch-if)# speed 10G
apic1(config-po-ch-if)# channel-mode active
```

FEX のポートチャネルにポートを設定します。この例では、ポート チャネル foo はリーフ ノード 102 に接続されている FEX 101 のポート イーサネット 1/1-2 に割り当てられ、ポート チャネル foo のインスタンスを作成します。リーフ ノードは番号（例えば 1002）を自動生成し、スイッチのポート チャネルを識別します。このポート チャネル番号は、作成されたポート チャネル foo のインスタンス数とは無関係で、リーフ ノード 102 に固有のものです。

（注） リーフ ノードに FEX モジュールを接続する設定は、FEX ポートを使用してポート チャネルを作成する前に実行する必要があります。

```
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group foo
```

リーフ 102 では、このポート チャネル インターフェイスを interface port-channel foo FEX 101 で呼ぶこともできます。

```
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel foo fex 101
apic1(config-leaf)# shut
```

複数のリーフ ノードでポート チャネルにポートを設定します。この例におけるポート チャネル foo は、101 ～ 103 の各リーフ ノード内にあるイーサネット 1/1-2 ポートに割り当てられます。リーフ ノードは各ノードで固有の番号（ノード間で同一または分けることができる）を自動生成し、ポートチャネル インターフェイスとして機能します。

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/1-2
```

```
apic1(config-leaf-if)# channel-group foo
```

ポートチャネルにメンバーを追加します。この例では、各リーフ ノードのポートチャネルに 2 つのメンバー eth1/3-4 を追加し、各ノードのポートチャネル foo がメンバー eth 1/1-4 を持つようにします。

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
```

ポートチャネルからメンバーを削除します。この例は、各リーフ ノードでポートチャネル foo から 2 つのメンバー eth1/2、eth1/4 を削除し、各ノードのポート チャネル foo がメンバー eth 1/1、eth1/3 を持つようにします。

```
 apic1(config)# leaf 101-103
apic1(config-leaf)# interface eth 1/2,1/4
apic1(config-leaf-if)# no channel-group foo
```

複数のリーフ ノードで異なるメンバーを持つポートチャネルを設定します。次に、同じポートチャネル foo ポリシーを使用して、リーフごとにメンバー ポートが異なる複数のリーフ ノードでポートチャネルインターフェイスを作成する例を示します。リーフ ノードのポートチャネル番号は、同じポートチャネル foo に対して同一にすることも別々にすることもできます。ただし CLI では、設定は interface port-channel foo で参照されます。FEX ポートにポートチャネルが設定されている場合は、interface port-channel foo fex <fex-id> で参照されます。

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 103
apic1(config-leaf)# interface ethernet 1/5-8
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group foo
```

LACP のポート単位のプロパティを設定します。次に、LACP のポート単位のプロパティについてポートチャネルのメンバー ポートを設定する例を示します。

（注） ACI モデルでは、これらのコマンドはポートがポート チャネルのメンバーになった後でのみ使用できます。ポートがポート チャネルから削除された場合、これらポート単位のプロパティ設定も削除されます。

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/1-2
```

```
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# lacp port-priority 1000
apic1(config-leaf-if)# lacp rate fast
```

ポート チャネルの管理状態を設定します。この例におけるポートチャネル foo は、channel-group コマンドを使用することで、101 ～ 103 の各リーフ ノードに対して設定されます。ポートチャネルの管理状態はポートチャネルインターフェイスを使用してリーフごとに設定できます。ACI モデルでは、ポートチャネルの管理状態をグローバル スコープで設定することはできません。

```
// create port-channel foo in each leaf
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo

// configure admin state in specific leaf
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel foo
apic1(config-leaf-if)# shut
```

オーバーライド設定は、他のプロパティを共有しながら各リーフのポートチャネルインターフェイスに特定の VLAN ドメインを割り当てる場合などにとても便利です。

```
// configure a port channel global config
apic1(config)# interface port-channel foo
apic1(config-if)# speed 1G
apic1(config-if)# channel-mode active

// create port-channel foo in each leaf
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo

// override port-channel foo in leaf 102
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel foo
apic1(config-leaf-if)#  speed 10G
apic1(config-leaf-if)#  channel-mode on
apic1(config-leaf-if)#  vlan-domain dom-foo
```

次の例では、channel-group コマンドを使用することで、ポートのポート チャネル割り当てを変更します。他のポート チャネルに割り当てる前にポート チャネルのメンバーシップを削除する必要はありません。

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# channel-group bar
```

REST API を使用して複数のスイッチに適用される 2 つのポート チャネルの設定

この例では、リーフ スイッチ 17 に 2 つのポート チャネル (PC) を、リーフ スイッチ 18 に別のポート チャネルを、リーフ スイッチ 20 に第 3 のチャネルを作成します。各リーフ スイッチで、同じインターフェイスが PC の一部になります (ポート チャネル 1 の場合はインターフェイス 1/10 ～ 1/15、ポート チャネル 2 の場合は 1/20 ～ 1/25) 。各スイッチ ブロックには連続するスイッチ ID のグループを 1 つしか含めることができないため、ポリシーは 2 つのスイッチ ブロックを使用します。これらの PC はすべて同じ設定になります。

（注） PC の設定が同じであっても、この例では、2 つの異なるインターフェイス ポリシー グループを使用します。各インターフェイス ポリシー グループは、スイッチ上の PC を表します。所定のインターフェイス ポリシー グループに関連付けられているインターフェイスはすべて、同じ PC の一部です。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチおよびプロトコルが設定されており、使用可能であること。

手順

2 台の PC を作成するには、次のような XML 形式の post を送信します。

例：

```
<infraInfra dn="uni/infra">

       <infraNodeP name=”test">
             <infraLeafS name="leafs" type="range">
                  <infraNodeBlk name="nblk”
                       from_=”17" to_=”18”/>
                  <infraNodeBlk name="nblk”
                       from_=”20" to_=”20”/>
             </infraLeafS>
             <infraRsAccPortP tDn="uni/infra/accportprof-test1"/>
             <infraRsAccPortP tDn="uni/infra/accportprof-test2"/>
       </infraNodeP>

       <infraAccPortP name="test1">
             <infraHPortS name="pselc" type="range">
                  <infraPortBlk name="blk1”
                       fromCard="1" toCard="1"
                       fromPort="10" toPort=”15”/>
```

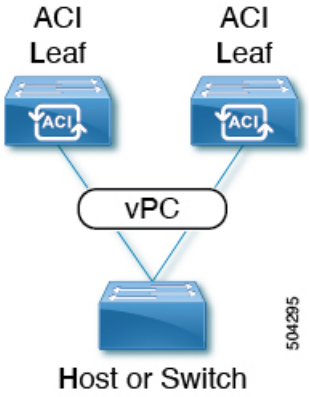
```
            <infraRsAccBaseGrp
                      tDn="uni/infra/funcprof/accbundle-bndlgrp1"/>
             </infraHPortS>
       </infraAccPortP>

       <infraAccPortP name="test2">
             <infraHPortS name="pselc" type="range">
                  <infraPortBlk name="blk1”
                       fromCard="1" toCard="1"
                       fromPort=“20" toPort=”25”/>
                  <infraRsAccBaseGrp
                       tDn="uni/infra/funcprof/accbundle-bndlgrp2" />
             </infraHPortS>
       </infraAccPortP>

       <infraFuncP>
             <infraAccBndlGrp name="bndlgrp1" lagT="link">
                   <infraRsHIfPol  tnFabricHIfPolName=“default"/>
                   <infraRsCdpIfPol tnCdpIfPolName=”default”/>
                   <infraRsLacpPol tnLacpLagPolName=”default"/>
             </infraAccBndlGrp>

             <infraAccBndlGrp name="bndlgrp2" lagT="link">
                   <infraRsHIfPol  tnFabricHIfPolName=“default"/>
                   <infraRsCdpIfPol tnCdpIfPolName=”default”/>
                   <infraRsLacpPol tnLacpLagPolName=”default"/>
             </infraAccBndlGrp>
       </infraFuncP>

</infraInfra>
```

仮想ポート チャネル

Cisco ACI の仮想ポートチャネルについて

仮想ポートチャネル（vPC）によって、2つの異なるCisco Application Centric Infrastructure（ACI）リーフ ノードに物理的に接続されたリンクを、リンク集約テクノロジーをサポートするネットワーク スイッチ、サーバー、他のネットワーク デバイスなどから単一のポートチャネル（PC）に見えるようにすることができます。vPC は、vPC のピア スイッチとして指定された 2 台の Cisco ACI リーフ スイッチから構成されます。Of the vPC peers, one is primary and one is secondary. The system formed by the switches is referred to as a vPC domain.

図 *18 : vPC* ドメイン



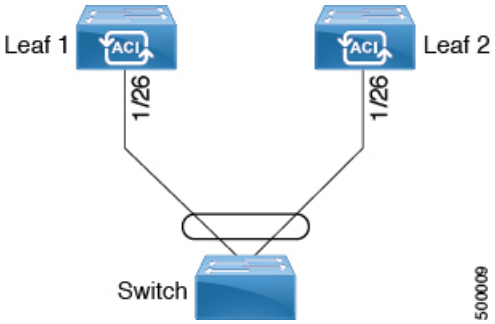
次の動作は、Cisco ACI vPC 実装に固有です。

- vPC ピア間に専用ピア リンクはありません。代わりに、ファブリック自体がマルチシャーシ トランキング（MCT）として機能します。
- ピア到達可能性プロトコル：Cisco ACI は、Cisco Fabric Services（CFS）の代わりに Zero Message Queue（ZMQ）を使用します。
  - ZMQ は、トランスポートとして TCP を使用するオープンソースの高性能メッセージング ライブラリです。
  - このライブラリは、スイッチ上では libzmq としてパッケージ化されており、vPC ピアと通信する必要がある各アプリケーションにリンクされています。
- ピアの到達可能性は、物理ピア リンクを使用して処理されません。代わりに、ルーティング トリガーを使用してピアの到達可能性を検出します。
  - vPC マネージャは、ピア ルート通知のためにユニキャスト ルーティング情報ベース（URIB）に登録します。
  - IS-IS がピアへのルートを検出すると、URIB は vPC マネージャに通知します。vPC マネージャは、ピアとの ZMQ ソケットを開こうとします。
  - ピア ルートが IS-IS によって取り消されると、URIB は vPC マネージャに再び通知し、vPC マネージャは MCT リンクをダウンします。

Cisco ACI 仮想ポートチャネルのワークフロー

このワークフローでは、仮想ポート チャネル（vPC）の設定に必要な手順の概要を示します。

図 *19 :* バーチャル ポート チャネルの設定



Cisco Application Centric Infrastructure（ACI）Cisco ACI

始める前に

- インフラ セキュリティ ドメインに読み取り/書き込みアクセス権限があることを確認します。
- 必要なインターフェイスを持つターゲット リーフ スイッチが使用できることを確認します。

2 つのリーフ スイッチ間に vPC ドメインを作成する場合は、以下のハードウェア モデルの制限が適用されます。

- 第 1 世代のスイッチは、第 1 世代の他のスイッチとのみ互換性があります。これらのスイッチモデルは、スイッチ名の末尾に「EX」、「FX」、「FX2」、「GX」またはそれ以降のサフィックスがないことで識別できます。たとえば、N9K-9312TX という名前などです。

  第 2 世代以降のスイッチは、vPC ドメインで混在させることができます。これらのスイッチモデルは、スイッチ名の末尾に「EX」、「FX」、「FX2」、「GX」またはそれ以降のサフィックスが付いていることで識別できます。たとえば、N9K-93108TC-EX や N9K-9348GC-FXP という名前などです。

互換性のある vPC スイッチ ペアの例：

- N9K-C9312TX および N9K-C9312TX
- N9K-C93108TC-EX および N9K-C9348GC-FXP
- N9K-C93180TC-FX and N9K-C93180YC-FX
- N9K-C93180YC-FX および N9K-C93180YC-FX

互換性のない vPC スイッチ ペアの例：

- N9K-C9312TX および N9K-C93108TC-EX
- N9K-C9312TX および N9K-C93180YC-FX

手順

ステップ 1 仮想ポートチャネルを設定します。

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

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

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

d) **[ノード タイプ（Node Type）]** で、**[リーフ（Leaf）]** をクリックします。

e) **[ポート タイプ（Port Type）]** で、**[アクセス（Access）]** をクリックします。

f) **[インターフェイス タイプ（Interface Type）]** で、**[イーサネット（Ethernet）]** をクリックします。

g) **[インターフェイスの集約タイプ（Interface Aggregation Type）]**で、**[vPC]** を選択します。

h) **[vPC リーフ スイッチ ペア（vPC Leaf Switch Pair）]** の場合は、**[vPC リーフ スイッチ ペアの選択（Select vPC Leaf Switch Pair）]** をクリックし、目的のスイッチ ペアのボックスにチェックを入れて、**[選択（Select）]** をクリックします。複数のスイッチを選択できます。オプションとして、**[vPC リーフ スイッチ ペアの作成（Create vPC Leaf Switch Pair）]** をクリックし、目的に応じてフィールドに入力します。

i) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。

j) **[PC/vPC インターフェイス ポリシー グループ（PC/vPC Interface Policy Group）]** の場合は、**[PC/vPC インターフェイス ポリシー グループの選択（Select PC/vPC Interface Policy Group）]** をクリックします。

k) **[PC/vPC インターフェイス ポリシー グループの選択（Select PC/vPC Interface Policy Group）]** ダイアログで、既存のポリシー グループを選択して **[選択（Select）]** をクリックするか、**[PC/vPC インターフェイス ポリシー グループの作成（Create PC/vPC Interface Policy Group）]** をクリックして新しいポリシー グループを作成し、フィールドに入力して **[保存（Save）]** をクリックしてから、そのポリシー グループを選択し、**[選択（Select）]** をクリックします。

l) [保存（Save）] をクリックします。

m) CLI コマンドの **show int** を外部スイッチが接続されている Cisco Application Centric Infrastructure（ACI）リーフ スイッチに対して使用し、スイッチと仮想ポートチャネルが適切に設定されていることを確認します。

この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データ トラフィックはフローできません。

ステップ 2 アプリケーション プロファイルを設定します。

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

b) [作業（Work）] ペインで、テナントをダブルクリックします。
c) [ナビゲーション（Navigation）] ペインで、**テナント名** > **[クイックスタート（Quick Start）]** を選択します。
d) エンドポイント グループ（EPG）、コントラクト、ブリッジ ドメイン、サブネット、およびコンテキストを設定します。
e) 以前に作成した仮想ポートチャネル スイッチのプロファイルにアプリケーション プロファイル EPG を関連付けます。

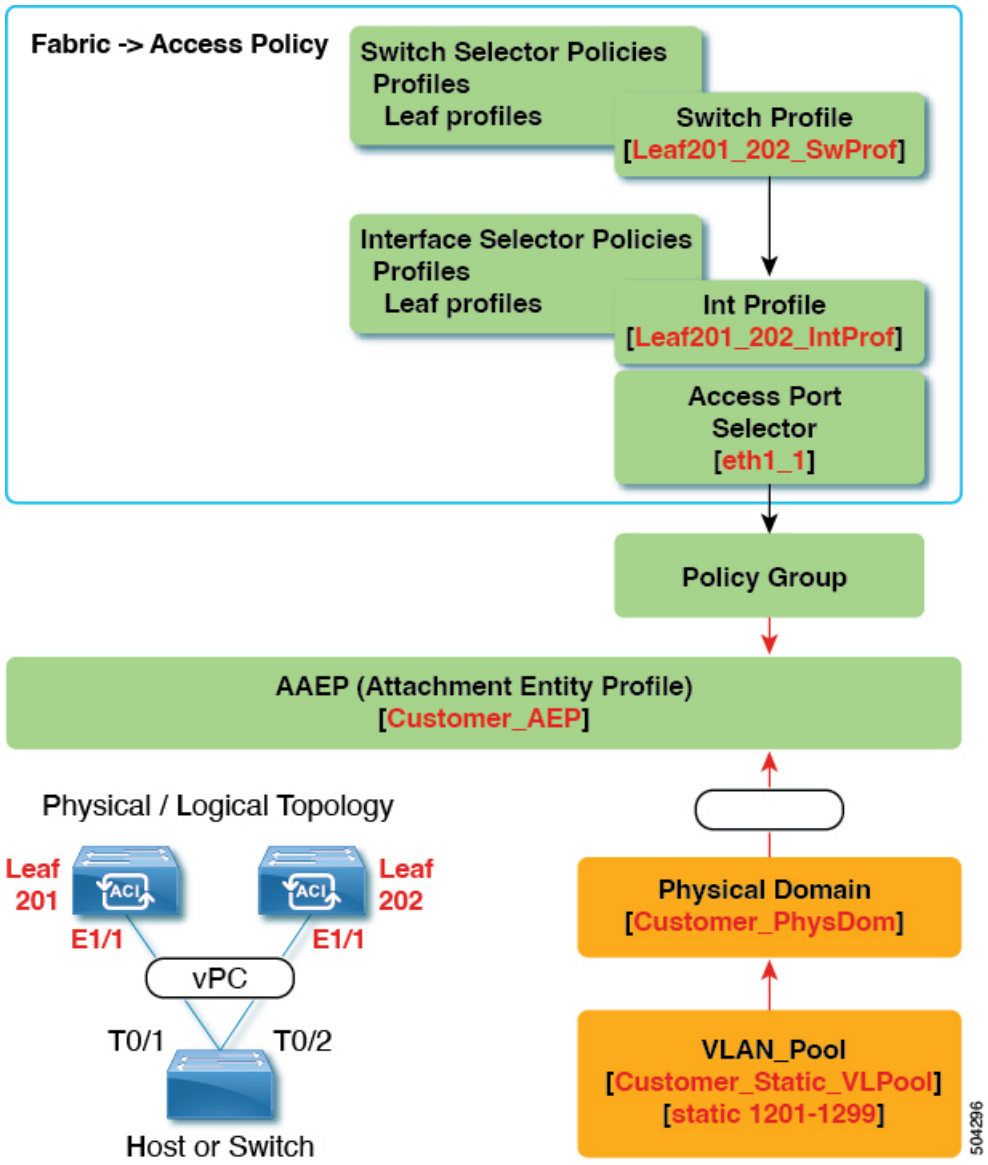
Virtual Port Channel Use Cases

結合プロファイルを持ち、2台のリーフスイッチ間で同じリーフスイッチインターフェイスを持つ vPC

このユース ケースの例では、次のことを定義します。

- `Leaf201_202_SwProf` と呼ばれる結合スイッチ プロファイル（ノード 201 およびノード 202）
- `Leaf201_202_IntProf` と呼ばれる結合インターフェース プロファイル（ノード 201 およびノード 202）
- `Eth1_1` と呼ばれるアクセス ポート セレクタ（`Leaf201_202` インターフェイス プロファイルの下）は、vPC インターフェイス ポリシー グループを指しています。
- vPC インターフェイス ポリシー グループは、`Customer_AEP` と呼ばれる AAEP を指しています。
- AEP（`Customer_AEP`）には、`Customer_PhysDom` との関連付けがあります。
- `Customer_PhysDom` には、`Customer_Static_VLPool` と呼ばれる VLAN プールとの関連付けがあります。

図 *20* : 結合プロファイルを持ち、*2* 台のリーフ スイッチ間で同じリーフ スイッチ インターフェイスを持つ *vPC*



この構成の機能

スイッチ `Leaf201` および `Leaf202` で、ポート Eth1/1 を vPC の一部として設定します。この vPC インターフェイスは、VLAN 1201 ～ 1299 にアクセスできます。インターフェイス ポリシー グループに応じて、LACP アクティブおよびその他のインターフェイス固有のポリシー設定を有効にすることができます。

この構成をいつ使用するか

たとえば、vPC 接続されたサーバーのみを備えたコンピューティング リーフ スイッチの専用ペアがある場合、これは、それらのスイッチのファブリック アクセス ポリシーの下で、結合

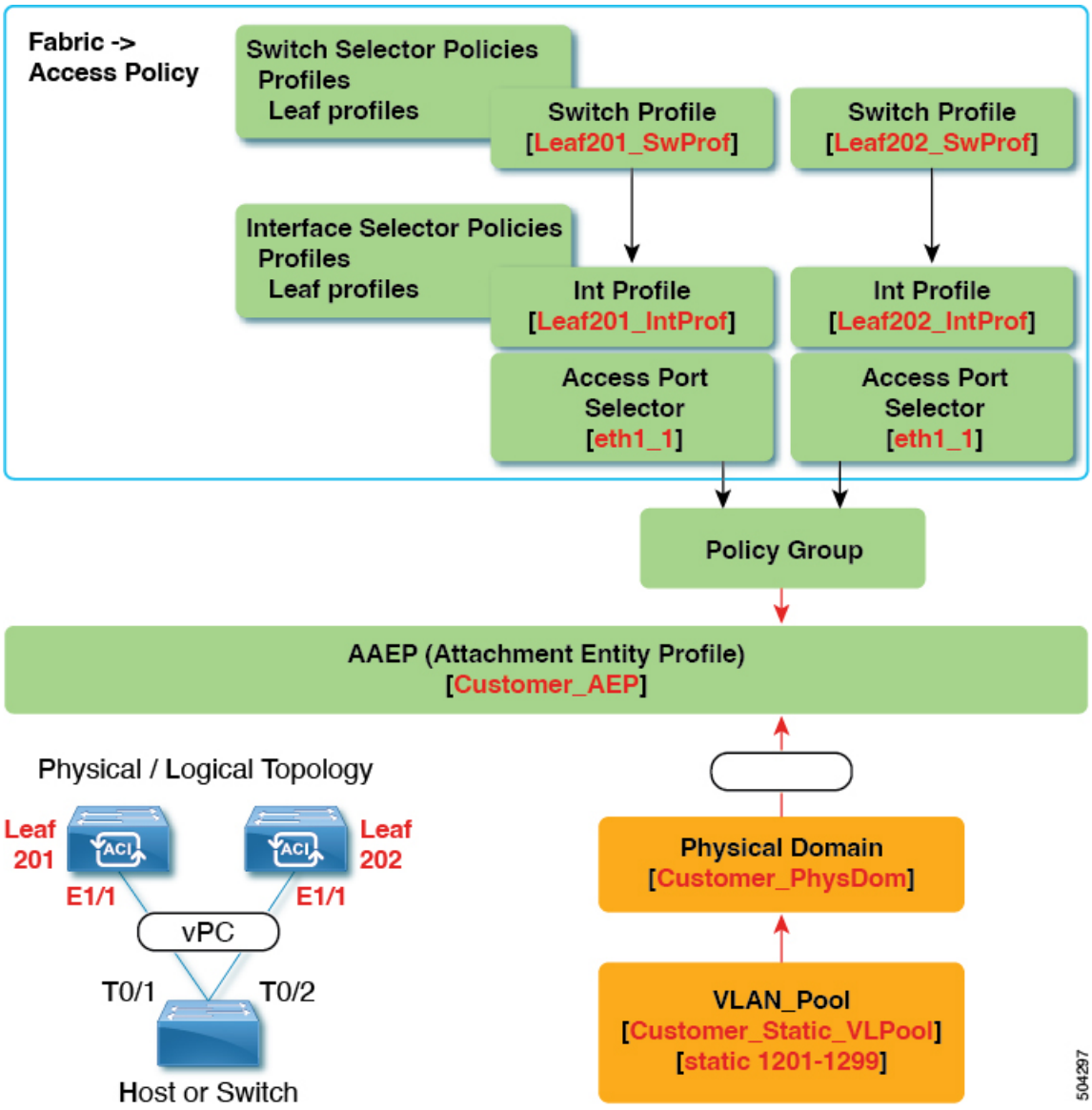
スイッチ/インターフェイス プロファイルを使用するための堅実なユース ケースになります。スイッチ、インターフェイス、アクセス ポート セレクタ、および vPC インターフェイス ポリシー グループを事前設定しておけば、最小限の労力で 48 のシャーシ タイプのサーバーを接続できるようにすることができます。

個別のプロファイルを持つ2台のリーフスイッチ間で同じリーフスイッチインターフェイスを持つ vPC

このユース ケースの例では、次のことを定義します。

- `Leaf201_SwProf` および `Leaf202_SwProf` と呼ばれる個々のスイッチ プロファイル（ノード 201 およびノード 202）。
- `Leaf201_IntProf` および `Leaf202_IntProf` と呼ばれる個々のインターフェース プロファイル（ノード 201 およびノード 202）
- Eth1_1 と呼ばれるアクセス ポート セレクタ（Leaf201 および Leaf202 インターフェイス プロファイルの下）は、同じ vPC インターフェイス ポリシー グループを指しています。
- vPC インターフェイス ポリシー グループは、`Customer_AEP` と呼ばれる AAEP を指しています。
- AEP（`Customer_AEP`）には、`Customer_PhysDom` との関連付けがあります。
- `Customer_PhysDom` には、`Customer_Static_VLPool` と呼ばれる VLAN プールとの関連付けがあります。

図 *21* : 個別のプロファイルを持つ *2* 台のリーフ スイッチ間で同じリーフ スイッチ インターフェイスを持つ *vPC*



この構成の機能

スイッチ `Leaf201` および `Leaf202` で、ポート Eth1/1 を vPC の一部として設定します。この vPC インターフェイスは、VLAN 1201 ～ 1299 にアクセスできます。インターフェイス ポリシー グループに応じて、LACP アクティブおよびその他のインターフェイス固有のポリシー設定を有効にすることができます。

この構成をいつ使用するか

コンピューティング、サービス、または Cisco Application Policy Infrastructure Controller（APIC）などの混合ワークロードをサポートするリーフ スイッチがある場合は、この構成を使用しま

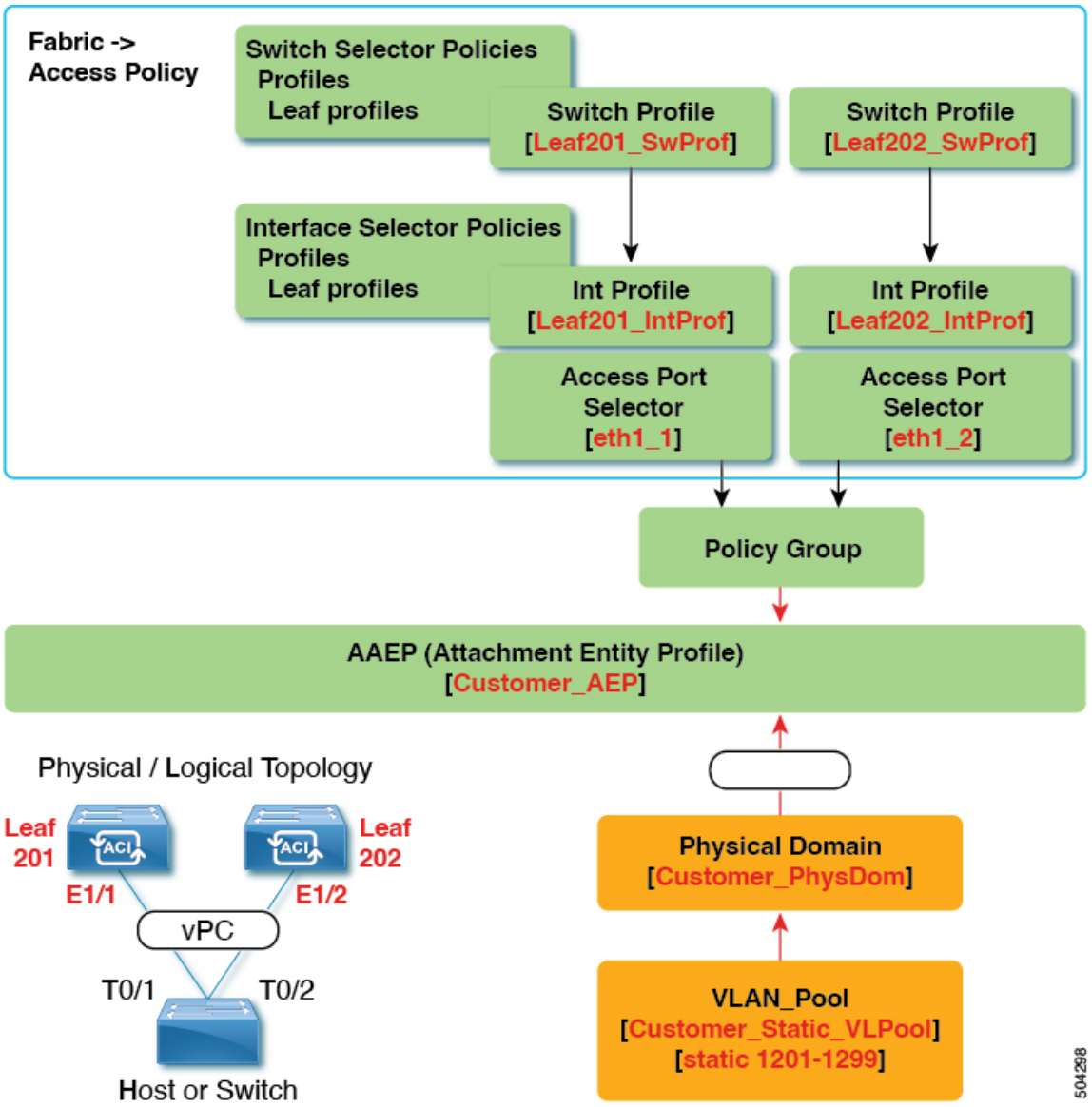
す。この場合、個別のインターフェイスプロファイルを使用すると、最大限の柔軟性が得られると同時に、**ファブリック** > **アクセス ポリシー**の設定を可能な限りクリーンで管理しやすい状態に保つことができます。

個別のプロファイルを持つ 2 つのリーフ スイッチにまたがる異なるリーフ スイッチ インターフェイスを持つ vPC

このユース ケースの例では、次のことを定義します。

- `Leaf201_SwProf` および `Leaf202_SwProf` と呼ばれる個々のスイッチ プロファイル（ノード 201 およびノード 202）。
- `Leaf201_IntProf` および `Leaf202_IntProf` と呼ばれる個々のインターフェース プロファイル（ノード 201 およびノード 202）
- `Eth1_1` と呼ばれるアクセス ポート セレクタ（`Leaf201` インターフェイス プロファイルの下）は、同じ vPC インターフェイス ポリシー グループを指しています。
- `Eth1_2` と呼ばれるアクセス ポート セレクタ（`Leaf202` インターフェイス プロファイルの下）は、同じ vPC インターフェイス ポリシー グループを指しています。
- vPC インターフェイス ポリシー グループは、`Customer_AEP` と呼ばれる AAEP を指しています。
- AEP（`Customer_AEP`）には、`Customer_PhysDom` との関連付けがあります。
- `Customer_PhysDom` には、`Customer_Static_VLPool` と呼ばれる VLAN プールとの関連付けがあります。

図 *22* : 個別のプロファイルを持つ *2* つのリーフ スイッチにまたがる異なるリーフ スイッチ インターフェイスを持つ *vPC*



この構成の機能

Leaf201 のポート Eth1/1 および Leaf202 のポート Eth 1/2 で、これらのポートが vPC の一部になるように設定します。この vPC インターフェイスは、VLAN 1201 ～ 1299 にアクセスできます。インターフェイス ポリシー グループに応じて、LACP アクティブおよびその他のインターフェイス固有のポリシー設定を有効にすることができます。

この構成をいつ使用するか

この構成は、一致するインターフェイスを使用できないラボ環境で使用します。ただし、サーバーのどこに接続したかを判断するには、常にGUIを参照する必要があります。結果として、この構成は扱いにくく、理想的ではありません。

（注） 本番環境ではこの構成を使用しないでください。

GUI を使用した vPC スイッチ ペアの定義

この手順では、GUI を使用して vPC スイッチ ペアを定義します。次の例に示すように、リーフ スイッチ ピア グループ名は単純にすることをお勧めします。

- Leaf201_202
- Leaf203_204
- Leaf205_206

名前付けと番号付けのベスト プラクティスについては、*Cisco ACI* オブジェクトの名前付けと番号付け： ベスト プラクティスドキュメントを参照してください。

https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/b-Cisco-ACI-Naming-and-Numbering.html

手順

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

ステップ 2 ナビゲーション ペインで、**[ポリシー（Policies）]** > **[スイッチ（Switch）]** > **[仮想ポートチャネルのデフォルト（Virtual Port Channel default）]**を選択します。

ステップ 3 **[明示的な vPC 保護グループ（Explicit vPC Protection Groups）]** テーブルで、**[+]** をクリックし、次のようにフィールドに入力します。

a) **[名前（Name）]** フィールドに、vPC ペアの名前を入力します。

名前の例：`Leaf201_202`。この例のような名前を使用すると、どの2つのファブリック ノードが vPC ピアであるかを簡単に識別できます。

b) **[ID]** フィールドに、vPC ペアの ID（論理ピア ID） を入力します。

ID の例：`201`。この例では、ペアの最初のノード ID 番号を使用して、ID を vPC ペアと関連付けやすくしています。

c) **[Switch 1]** および **[Switch 2]** フィールドで、vPC スイッチ ペアのリーフスイッチを選択します。

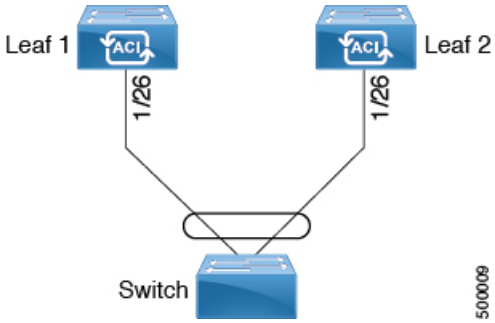
d) [送信（Submit）] をクリックします。

vPC ペアは、**[明示的な vPC 保護グループ（Explicit vPC Protection Groups）]** テーブルに追加されます。**[仮想 IP（Virtual IP）]** 値は、システム トンネル エンドポイント（TEP）プールから自動生成された IP アドレスであり、vPC スイッチ ペアの仮想共有（エニーキャスト）TEP を表します。つまり、vPC ペアの vPC 接続エンドポイント宛てのパケットは、このエニーキャスト VTEP を使用してパケットを送信します。

GUI を使用した ACI リーフ スイッチの仮想ポート チャネルの設定

この手順では、クイック スタート ウィザードを使用して、トランク スイッチを Cisco Application Centric Infrastructure（ACI）リーフ スイッチの仮想ポート チャネルに接続します。手順は、Cisco ACI リーフ スイッチ インターフェイスに他の種類のデバイスを接続する場合と同じになります。

図 *23 :* スイッチ バーチャル ポート チャネル設定



（注） ポートがピアから LACP PDU を受信しない場合、LACP はポートを中断ステートに設定します。これが、サーバが LACP にポートを論理的アップにするように要求するときに、サーバの起動に失敗する原因になることがあります。**LACP suspend individual** を無効にして、動作を個々の使用に合わせて調整します。そのためには、vPC ポリシー グループでポート チャネル ポリシーを作成し、モードを LACP アクティブに設定してから、**Suspend Individual Port** を削除します。これ以後、vPC 内のポートはアクティブなまま、LACP パケットを送信し続けます。

仮想ポート チャネル間での適応型ロード バランシング（ALB）（ARP ネゴシエーションに基づく） は、Cisco ACI ではサポートされていません。

始める前に

- Cisco ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。

- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが Cisco ACI ファブリックに登録され、使用可能であること。

（注） 2 つのリーフ スイッチ間に vPC ドメインを作成する場合は、以下のハードウェア モデルの制限が適用されます。

- 第 1 世代のスイッチは、第 1 世代の他のスイッチとのみ互換性があります。これらのスイッチ モデルは、スイッチ名の末尾に「EX」、「FX」、「FX2」、「GX」またはそれ以降のサフィックスがないことで識別できます。たとえば、N9K-9312TX という名前などです。

  第 2 世代以降のスイッチは、vPC ドメインで混在させることができます。これらのスイッチ モデルは、スイッチ名の末尾に「EX」、「FX」、「FX2」、「GX」またはそれ以降のサフィックスが付いていることで識別できます。たとえば、N9K-93108TC-EX や N9K-9348GC-FXP という名前などです。

手順

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

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

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

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

a) **[ノード タイプ（Node Type）]** で、**[リーフ（Leaf）]** をクリックします。

b) **[ポート タイプ（Port Type）]** で、**[アクセス（Access）]** をクリックします。

c) **[インターフェイス タイプ（Interface Type）]** で、**[イーサネット（Ethernet）]** をクリックします。

d) **[インターフェイスの集約タイプ（Interface Aggregation Type）]**で、**[vPC]** を選択します。

e) **[vPC リーフ スイッチ ペア（vPC Leaf Switch Pair）]** の場合は、**[vPC リーフ スイッチ ペアの選択（Select vPC Leaf Switch Pair）]** をクリックし、目的のスイッチ ペアのボックスにチェックを入れて、**[選択（Select）]** をクリックします。複数のスイッチを選択できます。オプションとして、**[vPC リーフ スイッチ ペアの作成（Create vPC Leaf Switch Pair）]** をクリックし、必要に応じてフィールドに入力し、ペアを選択して **[選択（Select）]** をクリックします。

f) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。

g) **[PC/vPC インターフェイス ポリシー グループ（PC/vPC Interface Policy Group）]** の場合は、**[PC/vPC インターフェイス ポリシー グループの選択（Select PC/vPC Interface Policy Group）]** をクリックします。

h) **[PC/vPC インターフェイス ポリシー グループの選択（Select PC/vPC Interface Policy Group）]** ダイアログで、既存のポリシー グループを選択し、**[選択（Select）]** をクリックします。オプションとして、**[PC/vPC インターフェイス ポリシー グループの作成（Create PC/vPC Interface Policy Group）]** をクリックして新しいポリシー グループを作成し、フィールドに入力して **[保存（Save）]** をクリックし、そのポリシー グループを選択して **[選択（Select）]** をクリックします。

i) **[ポート チャネル メンバー ポリシー（Port Channel Member Policy）]** で、**[ポート チャネル メンバー ポリシーの選択（Select Port Channel Member Policy）]**、をクリックし、ポリシーを選択して **[選択（Select）]** をクリックします。オプションとして、**[ポート チャネル メンバー ポリシーの作成（Create Port Channel Member Policy）]** をクリックし、必要に応じてフィールドに入力して **[保存（Save）]** をクリックし、そのポリシーを選択して **[選択（Select）]** をクリックします。

j) [保存（Save）] をクリックします。

確認：vPC が適切に設定されていることを確認するには、外部スイッチがアタッチされているリーフ スイッチ上で、CLI コマンド **show int** を使用します。

次のタスク

これで、スイッチ バーチャル ポート チャネルの設定手順は完了しました。

（注） この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データ トラフィックはフローできません。

NX-OS CLI を使用したリーフ ノードおよび FEX デバイスの仮想ポート チャネルの設定

仮想 ポート チャネル（vPC）は、ホストまたはスイッチを 2 つのアップストリーム リーフ ノードに接続して帯域幅の使用率と可用性を向上させる、ポート チャネルの拡張機能です。NX-OS では、vPC 設定は 2 つのアップストリーム スイッチのそれぞれで行われ、スイッチ間のピア リンクを使用して設定が同期されます。

（注） 2 台のリーフ スイッチ間で vPC ドメインを作成する場合、以下のいずれかの方法によって、両スイッチの世代を一致させる必要があります。

- 1 - Cisco Nexus N9K スイッチで、スイッチの名前の末尾には、「EX」なしの生成たとえば、N9K 9312TX
- 2: Cisco Nexus N9K スイッチがスイッチ モデル名の最後の「ex」の生成たとえば、N9K-93108TC-EX

これら 2 つのスイッチは互換性のある vPC ピアではありません。代わりに、同じ世代のスイッチを使用してください。

Cisco Application Centric Infrastructure（ACI）モデルでは、ピア リンクは必要なく、vPC 設定は両方のアップストリーム リーフ ノードに対してグローバルに実行できます。**vpc context** と呼ばれるグローバル コンフィギュレーション モードが Cisco ACI では導入されており、vPC インターフェイスは、両方のリーフノードにグローバルコンフィギュレーションを適用可能にする **interface vpc** というタイプを使用して表されます。

Cisco ACI モデルの vPC では、リーフポートを使用する vPC と FPC ポートを介した vPC の 2 つの異なるトポロジがサポートされます。リーフノードのペア間には多数の vPC インターフェイスを作成することができます。同様に、ストレート トポロジのリーフノード ペアに接続された FEX モジュールのペア間にも、多数の vPC インターフェイスを作成できます。

vPV に関する検討事項としては、以下のようなものがあります。

- 使用される vPC 名は、リーフノードペア間で一意です。たとえば、「corp」という vPC を作成する場合、FEX の有無にかかわらず、各リーフ ペアで作成できるのは 1 つだけです。
- リーフポートと FEX ポートを同じ vPC に含めることはできません。
- 各 FEX モジュールは、vPC corp の 1 つのインスタンスにのみ含めることができます。
- 設定を可能にする vPC コンテキスト
- vPC コンテキストモードでは、特定のリーフペアのすべての vPC を設定できます。vPC over FEX の場合、次の 2 つの代替例に示すように、vPC コンテキスト用に、または vPC インターフェイスとともに *fex-id* ペアを指定する必要があります。

```
(config)# vpc context leaf 101 102
(config-vpc)# interface vpc Reg fex 101 101
```

または

```
(config)# vpc context leaf 101 102 fex 101 101
(config-vpc)# interface vpc Reg
```

Cisco ACIモデルでは、vPC の設定は次の手順で行います（次の例に示します）。

（注） VLAN ドメインは、VLAN の範囲で必要です。ポート チャネルのテンプレートに関連付けられている必要があります。

1. VLAN の範囲で VLAN ドメイン構成 (グローバル設定)
2. vPC ドメイン設定（グローバル設定）
3. ポート チャネルのテンプレートの設定 (グローバル設定)
4. ポート チャネルのテンプレートを VLAN ドメインに関連付ける
5. vPC ポート チャネル設定（グローバル設定）
6. ポートをリーフノードの vPC に設定する
7. レイヤ 2、レイヤ 3を vPC コンテキストの vPC に設定する

手順

ステップ 1 **configure**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ 2 **vlan-domain***name*[**dynamic**] [**type** *domain-type*]

仮想ポート チャネルの VLAN ドメインの設定 (ポート チャネルのテンプレートとここ)。

例：

```
apic1(config)# vlan-domain dom1 dynamic
```

ステップ 3 **vlan***range*

VLAN ドメインの VLAN の範囲を設定し、configuration mode(設定モード、コンフィギュレーション モード) を終了します。単一の VLAN または複数の VLAN 範囲を設定できます。

例：

```
apic1(config-vlan)# vlan 1000-1999
apic1(config-vlan)# exit
```

ステップ 4 **vpc domain explicit** *domain-id* **leaf** *node-id1 node-id2*

vPC ドメインをリーフノードのペア間に設定します。リーフ ノード ペアとともに明示モードで vPC ドメイン ID を指定できます。

vPC ドメインを設定するための代替コマンドは次のとおりです。

- **vpc domain** [**consecutive** | **reciprocal**]

連続オプションおよび相互オプションを使用すると、Cisco ACI ファブリック内のすべてのリーフ ノードで vPC ドメインを自動設定できます。

- **vpc domain consecutive** *domain-start* **leaf** *start-node end-node*

  このコマンドは、リーフ ノード ペアの選択されたセットに対して連続して vPC ドメインを設定します。

例：

```
apic1(config)# vpc domain explicit 1 leaf 101 102
```

ステップ 5 **peer-dead-interval** *interval*

リーフ スイッチは、ピアから応答を受信する前に、vPC を復元するまで待機する時間の遅延を設定します。この時間内ピアから応答を受信するはないとリーフ スイッチ、ピアを停止するいると見なすをマスターとしての役割を持つvPC始動します。ピアから応答を受信するとその時点で、vPC を復元します。範囲は 5 ～ 600 秒です。デフォルトは 200 秒です。

例：

```
apic1(config-vpc)# peer-dead-interval 10
```

ステップ 6 **exit**

グローバル コンフィギュレーション モードに戻ります。

例：

```
apic1(config-vpc)# exit
```

ステップ 7 **template port-channel** *channel-name*

新しいポートチャネルを作成するか、既存のポートチャネルを設定します（グローバル コンフィギュレーション）。

すべての vPC は、各リーフ ペアのポートチャネルとして設定されます。同じ vPC のリーフ ペアでは、同じポートチャネル名を使用する必要があります。このポートチャネルは、リーフ ノードの 1 つ以上のペア間で vPC を作成するために使用できます。各リーフ ノードには、この vPC のインスタンスが 1 つだけあります。

例：

```
apic1(config)# template port-channel corp
```

ステップ 8 **vlan-domain member***vlan-domain-name*

以前に設定された VLAN ドメインには、ポート チャネルのテンプレートを関連付けます。

例：

```
vlan-domain member dom1
```

ステップ 9 **switchport access vlan** *vlan-id* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

ポート チャネルを関連付けるすべてのポート上に VLAN を持つ EPG を導入します。

例：

```
apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg
 webEpg
```

ステップ 10 **channel-mode active**

（注） vPC のポートチャネルはアクティブ チャネルモードである必要があります。

例：

```
apic1(config-po-ch-if)# channel-mode active
```

ステップ 11 **exit**

設定モードに戻ります。

例：

```
apic1(config-po-ch-if)# exit
```

ステップ 12 **leaf** *node-id1 node-id2*

設定するリーフ スイッチのペアを指定します。

例：

```
apic1(config)# leaf 101-102
```

ステップ 13 **interface** *typeleaf/interface-range*

ポート チャネルに設定するインターフェイスまたはインターフェイスの範囲を指定します。

例：

```
apic1(config-leaf)# interface ethernet 1/3-4
```

ステップ 14 [**no**] **channel-group** *channel-name* **vpc**

インターフェイスまたはインターフェイスの範囲をポートチャネルに割り当てます。ポートチャネルからインターフェイスを削除するには、キーワード **no** を使用します。インターフェイス上からポートチャネルの割り当てを変更する場合は、以前のポートチャネルからインターフェイスを最初に削除することなく **channel-group** コマンドを入力することができます。

（注） このコマンドの **vpc** キーワードは、ポートチャネルを vPC にします。vPC がまだ存在しない場合は、vPC ID が自動的に生成され、すべてのメンバー リーフ ノードに適用されます。

例：

```
apic1(config-leaf-if)# channel-group corp vpc
```

ステップ 15 **exit**

例：

```
apic1(config-leaf-if)# exit
```

ステップ 16 **exit**

例：

```
apic1(config-leaf)# exit
```

ステップ 17 **vpc context leaf** *node-id1 node-id2*

vPC コンテキスト モードでは、vPC の設定を両方のリーフ ノード ペアに適用できます。

例：

```
apic1(config)# vpc context leaf 101 102
```

ステップ 18 **interface vpc** *channel-name*

例：

```
apic1(config-vpc)# interface vpc blue fex 102 102
```

ステップ 19 （任意）[**no**] **shutdown**

vPC コンテキストでの管理状態の設定では、両方のリーフ ノードに対して 1 つのコマンドで vPC の管理状態を変更できます。

例：

```
apic1(config-vpc-if)# no shut
```

例

次に、基本的な vPC を設定する例を示します。

```
apic1# configure
apic1(config)# vlan-domain dom1 dynamic
apic1(config-vlan)# vlan 1000-1999
apic1(config-vlan)# exit
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config-vpc)# peer-dead-interval 10
apic1(config-vpc)# exit
apic1(config)# template port-channel corp
apic1(config-po-ch-if)# vlan-domain member dom1
apic1(config-po-ch-if)# channel-mode active
apic1(config-po-ch-if)# exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group corp vpc
apic1(config-leaf-if)# exit
apic1(config)# vpc context leaf 101 102
```

次に、FEX ポートを使用して vPC を設定する例を示します。

```
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group Reg vpc
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc corp
apic1(config-vpc-if)# exit
apic1(config-vpc)# interface vpc red fex 101 101
apic1(config-vpc-if)# switchport
apic1(config-vpc-if)# exit
apic1(config-vpc)# interface vpc blue fex 102 102
apic1(config-vpc-if)# shut
```

REST API を使用して 2 つのスイッチ全体で単一のバーチャル ポート チャネルを設定する

2 つのスイッチ間で仮想ポート チャネルを作成するための 2 つの手順は次のとおりです。

- `fabricExplicitGEp` を作成します。このポリシーは、仮想ポート チャネルを形成するためにペアになるリーフ スイッチを指定します。
- インフラ セレクタを使用してインターフェイス コンフィギュレーションを指定します。

APIC は、`fabricExplicitGEp` の複数の検証を実行し、これらの検証のいずれかが失敗すると、障害が発生します。1 つのリーフは、他の 1 つのリーフのみとペアにできます。APIC は、このルールに違反する設定を拒否します。`fabricExplicitGEp` を作成する際、管理者はペアにするリーフ スイッチの両方の ID を提供する必要があります。APIC は、このルールに違反する設定を拒否します。両方のスイッチを `fabricExplicitGEp` の作成時に起動する必要があります。片方のスイッチが起動していない場合、APIC は設定を受け入れますが、障害を発生させます。両方のスイッチをリーフ スイッチにする必要があります。片方または両方のスイッチ ID がスパインに一致すると、APIC は設定を受け入れますが、障害を発生させます。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチおよびプロトコルが設定されており、使用可能であること。

手順

`fabricExplicitGEp` ポリシーを作成し、インターフェイスを指定する内部セレクタを使用するには、次の例のように XML とともにポストを送信します。

例：

```
<fabricProtPol pairT="explicit">
<fabricExplicitGEp name="tG" id="2">
 <fabricNodePEp id=”18”/>
 <fabricNodePEp id=”25"/>
        </fabricExplicitGEp>
   </fabricProtPol>
```

REST API を使用して 2 つのスイッチの選択したポート ブロックでバーチャル ポート チャネルを設定する

このポリシーは、リーフ 18 ではインターフェイス 1/10 ～ 1/15 を使用し、リーフ 25 ではインターフェイス 1/20 ～ 1/25 を使用して、リーフ スイッチ 18 および 25 で単一の仮想ポート チャネル（vPC）を作成します。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチおよびプロトコルが設定されており、使用可能であること。

（注） 2 台のリーフ スイッチ間で vPC ドメインを作成する場合、以下のいずれかの方法によって、両スイッチの世代を一致させる必要があります。

- 1 - Cisco Nexus N9K スイッチで、スイッチの名前の末尾には、「EX」なしの生成たとえば、N9K 9312TX
- 2: Cisco Nexus N9K スイッチがスイッチ モデル名の最後の「ex」の生成たとえば、N9K-93108TC-EX

これら 2 つのスイッチは互換性のある vPC ピアではありません。代わりに、同じ世代のスイッチを使用してください。

手順

vPC を作成するには、次の例のように XML でポストを送信します。

例：

```
<infraInfra dn="uni/infra">

       <infraNodeP name=”test1">
           <infraLeafS name="leafs" type="range">
                <infraNodeBlk name="nblk”
                     from_=”18" to_=”18”/>
           </infraLeafS>
           <infraRsAccPortP tDn="uni/infra/accportprof-test1"/>
       </infraNodeP>

       <infraNodeP name=”test2">
           <infraLeafS name="leafs" type="range">
                <infraNodeBlk name="nblk”
                     from_=”25" to_=”25”/>
```

```
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-test2"/>
    </infraNodeP>

    <infraAccPortP name="test1">
        <infraHPortS name="pselc" type="range">
            <infraPortBlk name="blk1"
                fromCard="1" toCard="1"
                fromPort="10" toPort="15"/>
            <infraRsAccBaseGrp
                tDn="uni/infra/funcprof/accbundle-bndlgrp" />
        </infraHPortS>
    </infraAccPortP>

    <infraAccPortP name="test2">
        <infraHPortS name="pselc" type="range">
            <infraPortBlk name="blk1"
                fromCard="1" toCard="1"
                fromPort="20" toPort="25"/>
        <infraRsAccBaseGrp
            tDn="uni/infra/funcprof/accbundle-bndlgrp" />
        </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
        <infraAccBndlGrp name="bndlgrp" lagT="node">
            <infraRsHIfPol  tnFabricHIfPolName="default"/>
            <infraRsCdpIfPol tnCdpIfPolName="default"/>
            <infraRsLacpPol tnLacpLagPolName="default"/>
        </infraAccBndlGrp>
    </infraFuncP>

</infraInfra>
```

仮想ポートチャネル移行：第一世代スイッチから第二世代スイッチへのノードの移行

最初にファブリックは、2つの第2世代スイッチ間のvPCを使用して設定されます。トラフィック フローは、これらの vPC のみがデータ トラフィックに使用されるように設計されます。第一世代のスイッチを第二世代のスイッチに移行するには、次の手順が必要です。

この手順では、vPC プライマリおよび vPC セカンダリが vPC ペアの第一世代のスイッチであり、前述のようにトラフィックを送信します。

このスイッチでサポートされるトランシーバ、アダプタ、およびケーブルを確認するには、『Cisco トランシーバ モジュール互換性情報』を参照してください。

トランシーバの仕様と取り付けに関する情報を確認するには、『Cisco トランシーバ モジュール インストール ガイド』を参照してください。

始める前に

仮想ポートチャネル（vPC）を構成する 2 つの第 2 世代 Cisco Nexus 9000 シリーズスイッチがあります。同じケーブルを使用して 2 つの第二世代 Cisco Nexus 9000 シリーズ スイッチに移行しようとしています。

第 1 世代 Cisco Nexus 9000 シリーズスイッチには、PID（製品 id）に EX または FX が含まれていないスイッチが含まれています。

第 2 世代 Cisco Nexus 9000 シリーズスイッチには、PID に EX または FX があるスイッチが含まれています。

移行している vPC 第 1 世代スイッチに接続されているすべての APIC コントローラをファブリック内の他のスイッチに移動し、APIC クラスタが「完全に適合」になるまで待ちます。

手順

ステップ 1 APIC GUI から、vPC セカンダリの**コントローラからの削除**操作を実行します。スイッチは APIC によってクリーン リブートされます。操作が完了するまで約 10 分待ちます。このアクションでは、すべてのトラフィックでデータ トラフィックにその他の第一世代スイッチを使用するように促します。vPC セカンダリからケーブルを外します。

ステップ 2 スイッチ固有のハードウェア取り付けガイドにある「スイッチ シャーシの取り付け」セクションに記載されている手順の順序を逆にして、第一世代のスイッチを取り外します。

ステップ 3 スイッチ固有のハードウェア取り付けガイドの「スイッチ シャーシの取り付け」セクションに記載されている手順に従って、第二世代スイッチを取り付けます。

ステップ 4 第一世代のスイッチから取り外した緩んでいないケーブルを、第二世代スイッチの同じポートに接続します。

ステップ 5 新しい第二世代スイッチを APIC に登録します。新しいノードを同じノード名およびノード ID に登録します。このスイッチはファブリックの一部になります。ポリシーは新しいスイッチにプッシュされ、生成スイッチの不一致があるため、vPC レッグはダウンしたままになります。この時点で、vPC プライマリは引き続きデータトラフィックを送信します。

ステップ 6 APIC GUI から、vPC プライマリの**コントローラからの削除**操作を実行します。このスイッチは、APIC によってクリーンにリブートされます。

操作が完了するまで約 10 分待ちます。第二世代スイッチの vPC レッグは、以前にダウン状態になっています。このアクションにより、すべてのトラフィックが新しい第二世代スイッチに移動するように求められます。新しい第二世代スイッチの vPC ポートは、リモート デバイス上で展開された VLAN に対して STP が無効になっている場合、約 10 ～ 22 秒で起動し、ファブリック内のフローに応じて 10 ～ 40 秒の範囲でトラフィックがドロップすることに注意してください。STP がリモート デバイスの VLAN で有効になっている場合、ファブリック内のフローに応じて、トラフィック損失は 40 ～ 75 秒の範囲になります。

ステップ 7 その他の第一世代スイッチからケーブルを外します。

ステップ 8 手順 2 で行ったように、第一世代スイッチを取り外します。

ステップ 9 手順 3 で行ったように、第二世代スイッチを取り付けます。

ステップ 10 手順 4 で行ったように、緩んだケーブルを接続します。

ステップ 11 新しい第二世代スイッチをAPICに登録します。新しいノードを同じノード名およびノードID に登録します。このスイッチはファブリックの一部になります。ポリシーが新しいスイッチにプッシュされ、vPC レッグが起動し、トラフィックの通過を開始します。

反射性リレー

リフレクティブ リレー (802.1Qbg)

リフレクティブ リレーでは、Cisco APIC リリース 2.3(1) でスイッチング オプションの開始時刻です。リフレクティブ リレー: IEEE 標準 802.1Qbg のタグのないアプローチ: ポリシーを適用し、必要に応じて、宛先またはターゲット VM サーバ上にトラフィックを送信する外部のスイッチへのすべてのトラフィックを転送します。ローカルスイッチングはありません。ブロードキャストまたはマルチキャスト トラフィックは、リフレクティブ リレーは、各 VM サーバでローカルにパケットのレプリケーションを提供します。

リフレクティブ リレーの利点の 1 つは、スイッチング機能および管理機能、Vm をサポートするサーバ リソースを解放するための外部スイッチを活用しています。リフレクティブ リレーでは、ポリシー、同じサーバ上の Vm の間のトラフィックに適用する Cisco APIC で設定することもできます。

Cisco ACI、入ってきたの同じポートからオンに戻すにトラフィックを許可する、リフレクティブ リレーを有効にできます。APIC GUI、NX-OS CLI または REST API を使用して、レイヤ 2 インターフェイス ポリシーとして individual ports(個々のポート、個別ポート)、ポート チャネルまたは仮想ポート チャネルでリフレクティブ リレーを有効にすることができます。この機能はデフォルトではディセーブルになっています。

用語 仮想イーサネット ポートのためのアグリゲータ 802.1Qbg を説明する (VEPA) が使用されるも機能します。

リフレクティブ リレーのサポート

リフレクティブ リレーには、次のサポートされています。

- IEEE 標準 802.1Qbg タグのないアプローチ、リフレクティブ リレーとも呼ばれます。

  Cisco APIC 2.3(1) リリースのリリースは IEE 標準 802.1Qbg をサポートしていませんマルチチャネル テクノロジーと S タグ付きアプローチです。

- 物理ドメイン。

  仮想ドメインはサポートしていません。

- 物理ポート、ポート チャネル（Pc）と仮想ポートチャネル（vPC）

  シスコ ファブリック エクステンダ (FEX) とブレード サーバはサポートされていません。リフレクティブ リレーはサポートされていないインターフェイスで有効になっていると、

障害が発生すると、最後の有効な設定が保持されます。ポートでリフレクティブ リレーを無効にすると、障害をクリアします。

- Cisco Nexus 9000 シリーズのスイッチと *EX* または *FX* 、モデル名の最後にします。

高度な GUI を使用したリフレクティブ リレーの有効化

; By default(デフォルトで、デフォルトでは) リフレクティブ リレーが無効になっていますただし、スイッチのレイヤ 2 インターフェイス ポリシーとして、ポート、またはポート チャネルまたは仮想ポート チャネルでこれを有効にできます。最初にポリシーを設定し、ポリシー グループとポリシーを関連付けます。

（注） 高度なモードの GUI でのみ次の手順を実行できます。

始める前に

この手順では、Cisco Application Centric Infrastructure（ACI）ファブリックを設定し、物理スイッチをインストールしてあることを前提としています。

手順

ステップ 1 Cisco APIC にログインして、[Advanced] モードを選択します。

ステップ 2 **[ファブリック]** > **[外部アクセス ポリシー]** > > **[インターフェイス ポリシー]** を選択し、**[ポリシー]** フォルダを開きます。

ステップ 3 **[L2 インターフェイス]** フォルダを右クリックして、**[L2 インターフェイス ポリシーの作成]** を選択します。

ステップ 4 **[L2 インターフェイス ポリシーの作成]** ダイアログ ボックスで、**[名前]** フィールドに名前を入力します。

ステップ 5 **[リフレクティブ リレー （802.1Qbg)]** エリアで、**[有効]** をクリックします。

ステップ 6 必要に応じて、ダイアログ ボックスのその他のオプションを選択します。

ステップ 7 [Submit] をクリックします。

ステップ 8 **[ポリシー]** ナビゲーション ペインで、**[ポリシー グループ]** フォルダを開いて、**[リーフ ポリシー グループ]** フォルダをクリックします。

ステップ 9 **[リーフ ポリシー グループ]** 中央ペインで、**[ACTIONS]** ドロップダウン リストを展開し、**[Create Leaf Access Port Policy Group]**、**[Create PC Interface Policy Group]**、**[Create vPC Interface Policy Group]**、または **[Create PC/vPC Override Policy Group]** を選択します。

ステップ 10 ポリシーグループ ダイアログ ボックスで、**[Name field]** フィールドに名前を入力します。

ステップ 11 **[L2 インターフェイス ポリシー]** ドロップダウン リストで、リフレクティブ リレーを有効にするために作成したポリシーを選択します。

ステップ 12 [SUBMIT] をクリックします。

NX-OS は、CLI を使用してリフレクティブ リレーの有効化

; By default(デフォルトで、デフォルトでは) リフレクティブ リレーが無効になっていますただし、スイッチのレイヤ 2 インターフェイス ポリシーとして、ポート、またはポート チャネルまたは仮想ポート チャネルでこれを有効にできます。CLI では、NX-OS テンプレートを使用して、複数のポートでリフレクティブ リレーの有効化または individual ports(個々のポート、個別ポート) で有効にすることができます。

始める前に

この手順では、Cisco Application Centric Infrastructure（ACI）ファブリックを設定し、物理スイッチをインストールしてあることを前提としています。

手順

リフレクティブ リレー 1 つまたは複数のポートで有効にします。

例：

この例では、1 つのポートでリフレクティブ リレーが有効にします。

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# switchport vepa enabled
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

例：

この例では、リフレクティブ リレー、テンプレートを使用して複数のポートで有効にします。

```
apic1(config)# template policy-group grp1
apic1(config-pol-grp-if)# switchport vepa enabled
apic1(config-pol-grp-if)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2-4
apic1(config-leaf-if)# policy-group grp1
```

例：

この例では、ポート チャネルでリフレクティブ リレーが有効にします。

```
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel po2
apic1(config-leaf-if)# switchport vepa enabled
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)#
```

例：

この例では、複数のポート チャネルでリフレクティブ リレーが有効にします。

```
apic1(config)# template port-channel po1
apic1(config-if)# switchport vepa enabled
apic1(config-if)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group po1
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

例：

この例では、仮想ポート チャネルでリフレクティブ リレーが有効にします。

```
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config-vpc)# exit
apic1(config)# template port-channel po4
apic1(config-if)# exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface eth 1/11-12
apic1(config-leaf-if)# channel-group po4 vpc
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc po4
apic1(config-vpc-if)# switchport vepa enabled
```

REST API を使用してリフレクティブ リレーの有効化

; By default(デフォルトで、デフォルトでは) リフレクティブ リレーが無効になっていますただし、スイッチのレイヤ 2 インターフェイス ポリシーとして、ポート、またはポート チャネルまたは仮想ポート チャネルでこれを有効にできます。

始める前に

この手順では、Cisco Application Centric Infrastructure（ACI）ファブリックを設定し、物理スイッチをインストールしてあることを前提としています。

手順

ステップ 1 リフレクティブ リレーを有効になっていると、レイヤ 2 インターフェイス ポリシーを設定します。

例：

```
<l2IfPol name=“VepaL2IfPol” vepa=“enabled”  />
```

ステップ 2 リーフ アクセス ポートのポリシー グループにレイヤ 2 インターフェイス ポリシーを適用します。

例：

```
<infraAccPortGrp name=“VepaPortG">
    <infraRsL2IfPol tnL2IfPolName=“VepaL2IfPol”/>
</infraAccPortGrp>
```

ステップ 3 インターフェイス セレクタとインターフェイス プロファイルを設定します。

例：

```
<infraAccPortP name="vepa">
        <infraHPortS name="pselc" type="range">
             <infraPortBlk name="blk"
                 fromCard="1" toCard="1" fromPort="20" toPort="22">
             </infraPortBlk>
             <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-VepaPortG" />
         </infraHPortS>
 </infraAccPortP>
```

ステップ 4 ノード セレクタとノードのプロファイルを設定します。

例：

```
<infraNodeP name="VepaNodeProfile">
        <infraLeafS name="VepaLeafSelector" type="range">
                 <infraNodeBlk name="VepaNodeBlk" from_="101" to_="102"/>
        </infraLeafS>
               <infraRsAccPortP tDn="uni/infra/accportprof-vepa"/>
    </infraNodeP>
```

FEX インターフェイス

FEX デバイスへのポート、PC、および vPC 接続の設定

FEX 接続とそれらの設定に使用されるプロファイルは、GUI、NX-OS スタイルの CLI または REST API を使用して作成できます。

Cisco APIC、リリース 3.0(1k) 以降 FEX 接続の設定のインターフェイスのプロファイルがサポートされます。

NX-OS スタイルの CLI を使用してこれらを設定する方法については、NX-OS スタイルの CLI を使用したポート、PC、および vPC の設定に関するトピックを参照してください。

ACI FEX のガイドライン

FEX を展開するときは、次のガイドラインに従ってください。

- リーフ スイッチ前面パネル ポートが EPG および VLAN を展開するように設定されていないと仮定して、最大 10,000 個のポート EPG が FEX を使用して展開することをサポートします。
- メンバーとして FEX ポートを含む各 FEX ポートまたは vPC では、各 VLAN で最大 20 個の EPG がサポートされます。

FEX 仮想ポート チャネル

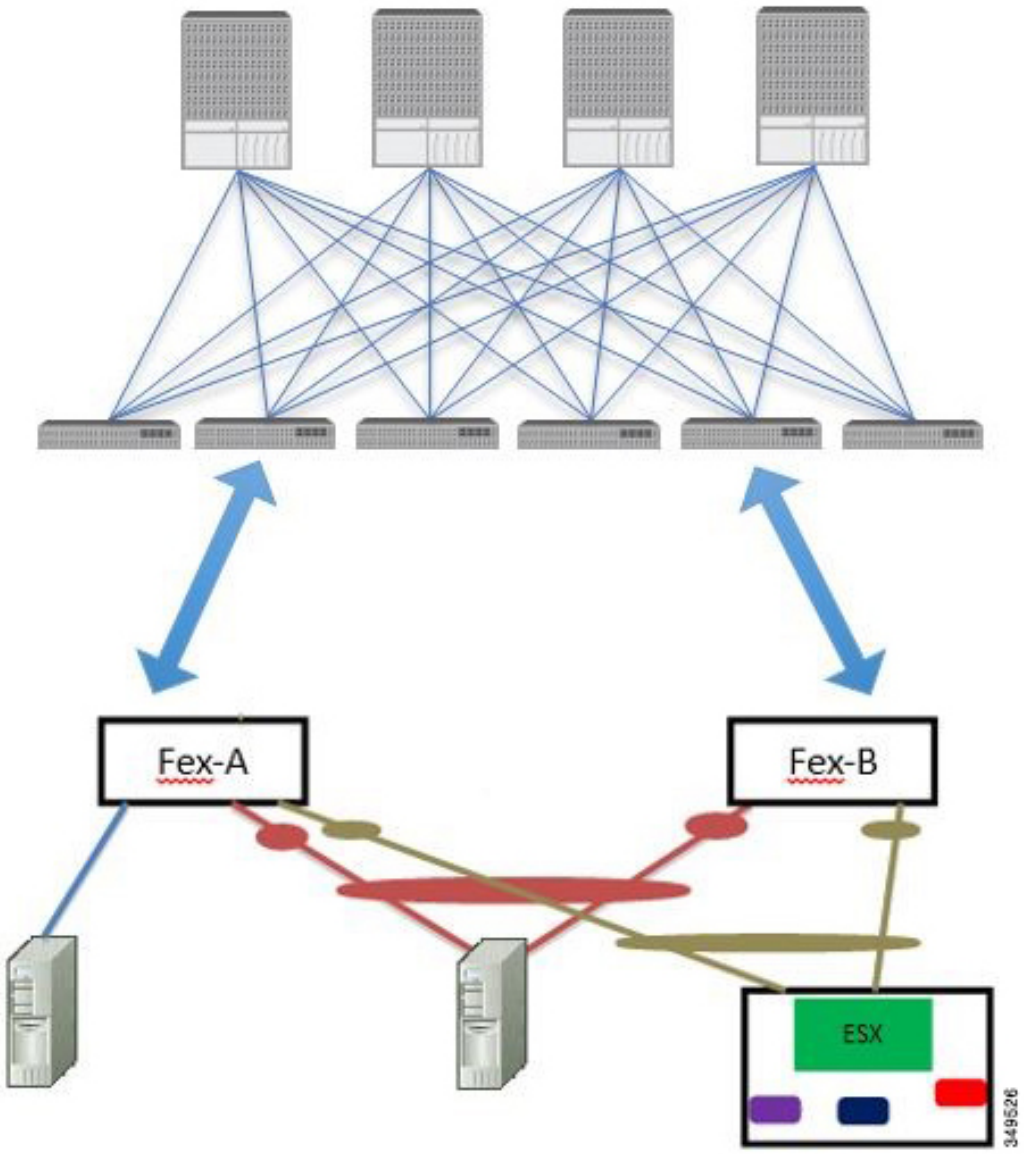
ACI ファブリックは、FEX ストレート vPC とも呼ばれる Cisco Fabric Extender（FEX）サーバ側仮想ポート チャネル（vPC）をサポートします。

（注） 2 台のリーフ スイッチ間で vPC ドメインを作成する場合、以下のいずれかの方法によって、両スイッチの世代を一致させる必要があります。

- 1: なしで Cisco Nexus N9K スイッチの生成」EX」または「FX」、スイッチ名前末尾にたとえば、N9K 9312TX
- 2: Cisco Nexus N9K スイッチ間での生成」EX」または「FX」スイッチ モデルの名前の末尾にたとえば、N9K-93108TC-EX

これら 2 つのスイッチは互換性のある vPC ピアではありません。代わりに、同じ世代のスイッチを使用してください。

図 *24* : サポートされる *FEX vPC* トポロジ



サポートされる FEX vPC ポート チャネル トポロジは次のとおりです。

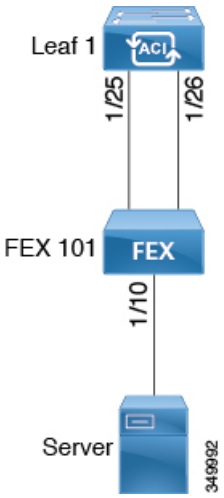
- FEX の背後にある VTEP および非 VTEP の両方のハイパーバイザ。
- ACI ファブリックに接続された 2 つの FEX に接続された仮想スイッチ（AVS や VDS など）（物理 FEX ポートに直接接続された vPC はサポートされません。vP Cはポート チャネルでのみサポートされます）。

（注） GAAP を、同じ FEX 上の異なるインターフェイスで IP から MAC バインディングへ変更する際の n.jpy へのプロトコルとして使用する場合、ブリッジ ドメインは **[ARP フラッディング (ARP Flooding)]** に設定し、**[EP 移動検出モード (EP Mode Detection Mode)]** : **[GARP ベースの検出 (GRAP-based Detection)]** を、ブリッジ ドメイン ウィザードの **[L3 設定 (L3 Configuration)]** ページで有効にする必要があります。この回避策は、のみ生成 1 スイッチで必要です。第 2 世代のスイッチで、または以降では、この問題ではありません。

GUI を使用した基本 FEX 接続の設定

次の手順では、FEX 導入に必要ないくつかのポリシーを自動的に作成するクイック スタート ウィザードを使用します。この手順では、FEXにサーバを接続する手順を示します。手順は、Cisco Application Centric Infrastructure（ACI）が接続されたFEXにデバイスを接続する場合と同じになります。

図 *25 :* 基本的な *FEX* 設定



（注） FEX ID 165 ～ 199 の FEX 接続の設定は、APIC GUI ではサポートされていません。これらの FEX ID のいずれかを使用するには、NX-OS スタイル CLI を使用してプロファイルを設定します。詳細については、「*NX-OS* スタイル *CLI* のインターフェイス プロファイルを使用して *FEX* 接続を設定する」を参照してください。

始める前に

- Cisco ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。

- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチ、インターフェイス、およびプロトコルが設定されており、使用可能であること。
- FEX に電源が入っていて、ターゲット リーフ スイッチのインターフェイスに接続されていること。

（注） FEX に接続されているファブリックポートチャネルでは、最大 8 つのメンバーがサポートされます。

手順

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

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

ステップ 3 **[作業（Work）]** ペインで、**[ファブリック エクステンダ（Fabric Extender）]** をクリックします。

ステップ 4 **[ファブリック エクステンダ（Fabric Extender）]** ダイアログボックスで、次の操作を実行します。

a) **[ノード（Node）]** で、**[ノードの選択（Select Node）]** をクリックし、目的のノードのボックスにチェックを入れて、**[OK]** をクリックします。複数のノードを選択できます。

b) **[すべてのスイッチのインターフェイス（Interfaces For All Switches）]** で、目的のインターフェイスの範囲を入力します。

c) **[接続先 FEX の ID（Connected FEX ID）]** には、FEX の ID を入力します。

NX-OS スタイル CLI を使用して、FEX ID 165 ～ 199 を設定する必要があります。『*Configuring FEX Connections Using Interface Profiles with the NX-OS Style CLI*』を参照してください。

d) [保存（Save）] をクリックします。
APIC によって、必要な FEX プロファイル（<*switch policy name*>*_FexP*<*FEX ID*>) およびセレクタ (<*switch policy name*>*_ifselctor*）が自動的に生成されます。

確認：FEX がオンラインであることを確認するには、FEX が接続されているスイッチに対して CLI コマンド show fex を使用します。

ステップ 5 サーバを単一 FEX ポートに接続できるようにするために、自動生成された FEX プロファイルをカスタマイズします。

a) [Navigation] ペインで、ポリシー リストで作成したスイッチ ポリシーを見つけます。また、自動生成された FEX <*switch policy name*>*_FexP*<*FEX ID*> プロファイルもあります。

b) *<switch policy name>_FexP<FEX ID>* プロファイルの作業ウィンドウで、+ をクリックして、新しいエントリを *Interface Selectors For FEX* リストに追加します。
[Create Access Port Selector] ダイアログが開きます。
c) セレクタの名前を指定します。
d) 使用する FEX インターフェイス ID を指定します。
e) リストから既存のインターフェイス ポリシー グループを選択するか、アクセス ポート ポリシー グループを作成します。

アクセス ポート ポリシー グループは、選択した FEX のインターフェイスに適用するインターフェイス ポリシーのグループを指定する名前付きポリシーです。インターフェイス ポリシーの例は、リンクレベル ポリシー（たとえば、1 gbit ポート速度）、接続エンティティ プロファイル、ストーム制御インターフェイス ポリシーなどです。

（注） インターフェイス ポリシー グループ内で、FEX ポート セレクタで指定されているインターフェイスを EPG が使用できるようにするために、[Attached Entity Profile] は必須です。

f) **Submit** をクリックして FEX プロファイルを APICに送信します。
APIC によって FEX プロファイルが更新されます。

確認：FEX インターフェイスが適切に設定されていることを確認するには、FEX が接続されているスイッチに対して CLI コマンド show int を使用します。

これで、基本 FEX の設定手順は完了しました。

次のタスク

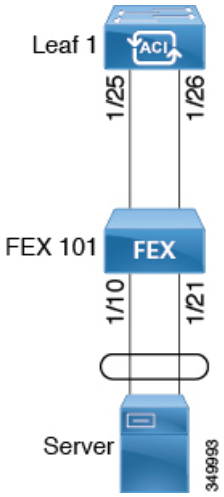
（注） この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データトラフィックはフローできません。

GUI を使用した FEX ポート チャネル接続の設定

主な手順は次のとおりです。

1. ポート チャネルの形成に FEX ポートを使用するように FEX プロファイルを設定します。
2. サーバに接続できるようにポート チャネルを設定します。

図 *26 : FEX* ポート チャネル



（注） この手順では、FEX ポート チャネルにサーバを接続する手順を示します。手順は、ACI が接続された FEX にデバイスを接続する場合と同じになります。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチ、インターフェイス、およびプロトコルが設定されており、使用可能であること。
- FEX が設定されており、電源が入っていて、ターゲット リーフ インターフェイスに接続されていること。

手順

ステップ 1 APIC で、FEX プロファイルにポート チャネルを追加します。

a) APIC メニューバーで、**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Profiles]** に移動します。

b) [Navigation] ペインで、FEX プロファイルを選択します。

APIC で自動生成された FEX プロファイル名の形式は、*<switch policy name>_FexP<FEX ID>* です。

c) **FEX Profile** 作業エリアで、+ をクリックして新しいエントリを *Interface Selectors For FEX* リストに追加します。
[Create Access Port Selector] ダイアログが開きます。

ステップ 2 FEX ポート チャネルにサーバを接続できるように、[Create Access Port Selector] をカスタマイズします。

a) セレクタの名前を指定します。
b) 使用する FEX インターフェイス ID を指定します。
c) リストから既存のインターフェイス ポリシー グループを選択するか、PC インターフェイス プロファイル グループを作成します。

ポート チャネル インターフェイス ポリシー グループは、選択した FEX のインターフェイスに適用するポリシーのグループを指定します。インターフェイス ポリシーの例は、リンク レベル ポリシー（たとえば、1 gbit ポート速度）、接続エンティティ プロファイル、ストーム制御インターフェイス ポリシーなどです。

（注） インターフェイス ポリシー グループ内で、FEX ポート セレクタで指定されているインターフェイスを EPG が使用できるようにするために、[Attached Entity Profile] は必須です。

d) [Port Channel Policy] オプションで、設定の要件に従って静的または動的な LACP を選択します。
e) [Submit] をクリックし、更新された FEX プロファイルを APIC に送信します。
APIC によって FEX プロファイルが更新されます。

確認: ポート チャネルが適切に設定されていることを確認するには、FEX が接続されているスイッチに対して CLI コマンド **show port-channel summary** を使用します。

次のタスク

これで、FEX ポート チャネルの設定手順は完了しました。

（注） この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データ トラフィックはフローできません。

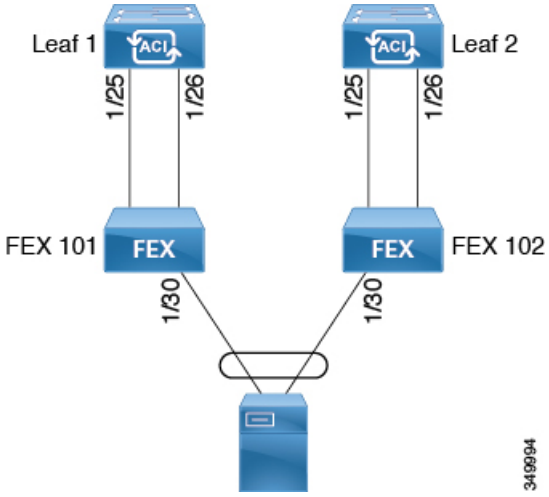
GUI を使用した FEX vPC 接続の設定

主な手順は次のとおりです。

1. バーチャル ポート チャネルを形成するように、2 つの既存 FEX プロファイルを設定します。

2. FEX ポート チャネルにサーバを接続できるように、バーチャル ポート チャネルを設定します。

図 *27 : FEX* バーチャル ポート チャネル



（注） この手順では、FEX バーチャル ポート チャネルにサーバを接続する手順を示します。手順は、ACI が接続された FEX にデバイスを接続する場合と同じになります。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチ、インターフェイス、およびプロトコルが設定されており、使用可能であること。
- FEX が設定されており、電源が入っていて、ターゲット リーフ インターフェイスに接続されていること。

（注） 2 台のリーフ スイッチ間で vPC ドメインを作成する場合、以下のいずれかの方法によって、両スイッチの世代を一致させる必要があります。

- 1 - Cisco Nexus N9K スイッチで、スイッチの名前の末尾には、「EX」なしの生成たとえば、N9K 9312TX
- 2: Cisco Nexus N9K スイッチがスイッチ モデル名の最後の「ex」の生成たとえば、N9K-93108TC-EX

これら 2 つのスイッチは互換性のある vPC ピアではありません。代わりに、同じ世代のスイッチを使用してください。

手順

ステップ 1 APIC で、2 つの FEX プロファイルにバーチャル ポート チャネルを追加します。

a) APIC メニューバーで、**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Profiles]** に移動します。

b) [Navigation] ペインで、最初の FEX プロファイルを選択します。

APIC で自動生成された FEX プロファイル名の形式は、<*switch policy name*>_FexP<*FEX ID*> です。

c) **FEX Profile** 作業エリアで、**+** をクリックして新しいエントリを *Interface Selectors For FEX* リストに追加します。
[Create Access Port Selector] ダイアログが開きます。

ステップ 2 FEX バーチャル ポート チャネルにサーバを接続できるように、[Create Access Port Selector] をカスタマイズします。

a) セレクタの名前を指定します。

b) 使用する FEX インターフェイス ID を指定します。

通常、各 FEX に同じインターフェイス ID を使用してバーチャル ポート チャネルを形成します。

c) リストから既存のインターフェイス ポリシー グループを選択するか、VPC インターフェイス プロファイル グループを作成します。

バーチャル ポート チャネル インターフェイス ポリシー グループは、選択した FEX のインターフェイスに適用するポリシーのグループを指定します。インターフェイス ポリシーの例は、リンク レベル ポリシー（たとえば、1 gbit ポート速度）、接続エンティティ プロファイル、ストーム制御インターフェイス ポリシーなどです。

（注） インターフェイス ポリシー グループ内で、FEX ポート セレクタで指定されているインターフェイスを EPG が使用できるようにするために、[Attached Entity Profile] は必須です。

d) [Port Channel Policy] オプションで、設定の要件に従って静的または動的な LACP を選択します。

e) [Submit] をクリックし、更新された FEX プロファイルを APIC に送信します。
APIC によって FEX プロファイルが更新されます。

確認: ポート チャネルが適切に設定されていることを確認するには、FEX が接続されているスイッチに対して CLI コマンド **show port-channel summary** を使用します。

ステップ 3 最初の FEX に指定したものと同じインターフェイス ポリシー グループを使用するように 2 番目の FEX を設定します。

a) 2 番目の FEX プロファイルの **FEX Profile** 作業エリアで、**+** をクリックして *Interface Selectors For FEX* リストに新しいエントリを追加します。
[Create Access Port Selector] ダイアログが開きます。

b) セレクタの名前を指定します。

c) 使用する FEX インターフェイス ID を指定します。

通常、各 FEX に同じインターフェイス ID を使用してバーチャル ポート チャネルを形成します。

d) ドロップダウン リストから、最初の FEX プロファイルで使用したものと同じバーチャル ポート チャネル インターフェイス ポリシー グループを選択します。

バーチャル ポート チャネル インターフェイス ポリシー グループは、選択した FEX のインターフェイスに適用するポリシーのグループを指定します。インターフェイス ポリシーの例は、リンクレベル ポリシー（たとえば、1 gbit ポート速度）、接続エンティティ プロファイル、ストーム制御インターフェイス ポリシーなどです。

（注） インターフェイス ポリシー グループ内で、FEX ポート セレクタで指定されているインターフェイスを EPG が使用できるようにするために、[Attached Entity Profile] は必須です。

e) [Submit] をクリックし、更新された FEX プロファイルを APIC に送信します。
APIC によって FEX プロファイルが更新されます。

確認：バーチャル ポート チャネルが適切に設定されていることを確認するには、いずれかの FEX が接続されているスイッチに対して CLI コマンド **show vpc extended** を使用します。

次のタスク

これで、FEX バーチャル ポート チャネルの設定手順は完了しました。

（注） この設定はハードウェア接続を有効にしますが、このハードウェア設定に関連付けられた有効なアプリケーション プロファイル、EPG、およびコントラクトがないと、データ トラフィックはフローできません。

REST API を使用した FEXVPC ポリシーの設定

このタスクにより、FEX 仮想ポート チャネル（VPC）ポリシーを作成します。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチ、インターフェイス、およびプロトコルが設定されており、使用可能であること。
- FEX が設定されており、電源が入っていて、ターゲット リーフ インターフェイスに接続されていること。

（注） 2 つのリーフ スイッチ間での VPC ドメインを作成するとき、同じスイッチの生成を次のいずれかのどちらのスイッチも必要があります。

- 1 - Cisco Nexus N9K スイッチで、スイッチの名前の末尾には、「EX」なしの生成たとえば、N9K 9312TX
- 2: Cisco Nexus N9K スイッチがスイッチ モデル名の最後の「ex」の生成たとえば、N9K-93108TC-EX

スイッチなど、これらの 2 つが互換性のある VPC ピアではありません。代わりに、同じ世代のスイッチを使用します。

手順

2 つのスイッチへの VPC を介して FEX のリンク ポリシーを作成するには、次の例などと XML post を送信します。

例：

```
<polUni>
<infraInfra dn="uni/infra">

<infraNodeP name="fexNodeP105">
    <infraLeafS name="leafs" type="range">
        <infraNodeBlk name="test" from_="105" to_="105"/>
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-fex116nif105" />
</infraNodeP>

<infraNodeP name="fexNodeP101">
    <infraLeafS name="leafs" type="range">
        <infraNodeBlk name="test" from_="101" to_="101"/>
```

```
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-fex113nif101" />
</infraNodeP>

<infraAccPortP name="fex116nif105">
   <infraHPortS name="pselc" type="range">
   <infraPortBlk name="blk1"
        fromCard="1" toCard="1" fromPort="45" toPort="48" >
   </infraPortBlk>
   <infraRsAccBaseGrp tDn="uni/infra/fexprof-fexHIF116/fexbundle-fex116" fexId="116" />
</infraHPortS>
</infraAccPortP>

<infraAccPortP name="fex113nif101">
   <infraHPortS name="pselc" type="range">
   <infraPortBlk name="blk1"
        fromCard="1" toCard="1" fromPort="45" toPort="48" >
   </infraPortBlk>
   <infraRsAccBaseGrp tDn="uni/infra/fexprof-fexHIF113/fexbundle-fex113" fexId="113" />
</infraHPortS>
</infraAccPortP>

<infraFexP name="fexHIF113">
    <infraFexBndlGrp name="fex113"/>
    <infraHPortS name="pselc-fexPC" type="range">
        <infraPortBlk name="blk"
            fromCard="1" toCard="1" fromPort="15" toPort="16" >
        </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexPCbundle" />
    </infraHPortS>
        <infraHPortS name="pselc-fexVPC" type="range">
        <infraPortBlk name="blk"
            fromCard="1" toCard="1" fromPort="1" toPort="8" >
        </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexvpcbundle" />
    </infraHPortS>
    <infraHPortS name="pselc-fexaccess" type="range">
        <infraPortBlk name="blk"
            fromCard="1" toCard="1" fromPort="47" toPort="47">
        </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-fexaccport" />
     </infraHPortS>

</infraFexP>

<infraFexP name="fexHIF116">
    <infraFexBndlGrp name="fex116"/>
    <infraHPortS name="pselc-fexPC" type="range">
        <infraPortBlk name="blk"
            fromCard="1" toCard="1" fromPort="17" toPort="18" >
        </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexPCbundle" />
    </infraHPortS>
        <infraHPortS name="pselc-fexVPC" type="range">
        <infraPortBlk name="blk"
            fromCard="1" toCard="1" fromPort="1" toPort="8" >
        </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexvpcbundle" />
      </infraHPortS>
      <infraHPortS name="pselc-fexaccess" type="range">
        <infraPortBlk name="blk"
            fromCard="1" toCard="1" fromPort="47" toPort="47">
        </infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-fexaccport" />
```

```
     </infraHPortS>


</infraFexP>

<infraFuncP>
<infraAccBndlGrp name="fexPCbundle" lagT="link">
    <infraRsLacpPol tnLacpLagPolName='staticLag'/>
    <infraRsHIfPol tnFabricHIfPolName="1GHIfPol" />
    <infraRsAttEntP tDn="uni/infra/attentp-fexvpcAttEP"/>
</infraAccBndlGrp>

<infraAccBndlGrp name="fexvpcbundle" lagT="node">
    <infraRsLacpPol tnLacpLagPolName='staticLag'/>
    <infraRsHIfPol tnFabricHIfPolName="1GHIfPol" />
    <infraRsAttEntP tDn="uni/infra/attentp-fexvpcAttEP"/>
</infraAccBndlGrp>
</infraFuncP>


<fabricHIfPol name="1GHIfPol" speed="1G" />
<infraAttEntityP name="fexvpcAttEP">
    <infraProvAcc name="provfunc"/>
    <infraRsDomP tDn="uni/phys-fexvpcDOM"/>
</infraAttEntityP>

<lacpLagPol dn="uni/infra/lacplagp-staticLag"
    ctrl="susp-individual,graceful-conv"
    minLinks="2"
    maxLinks="16">
</lacpLagPol>
```

NX-OS スタイル CLI とプロファイルを使用して FEX 接続の設定

NX-OS スタイル CLI を使用してリーフ ノードへの接続を FEX を設定するには、次の手順を使用します。

（注） FEX Id を持つ FEX 接続を構成する 165 に 199 APIC GUI ではサポートされていません。これらの FEX Id のいずれかを使用するには、次のコマンドを使用して、プロファイルを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `apic1# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**leaf-interface-profile** *name* 例：	設定するリーフ インターフェイス プロファイルを指定します。

	コマンドまたはアクション	目的
	apic1(config)# **leaf-interface-profile fexIntProf1**	
ステップ 3	**leaf-interface-group** *name* 例： apic1(config-leaf-if-profile)# **leaf-interface-group leafIntGrp1**	設定するインターフェイス グループを指定します。
ステップ 4	**fex associate** *fex-id* [**template** *template-type fex-template-name*] 例： apic1(config-leaf-if-group)# **fex associate 101**	リーフ ノードに FEX モジュールを接続します。使用するテンプレートを指定するのにオプションのテンプレートのキーワードを使用します。存在しない場合、システムは、名前とタイプが指定したで、テンプレートを作成します。

例

このマージの例では、ID 101 で FEX 接続のリーフ インターフェイス プロファイルを設定します。

```
apic1# configure
apic1(config)# leaf-interface-profile fexIntProf1
apic1(config-leaf-if-profile)# leaf-interface-group leafIntGrp1
apic1(config-leaf-if-group)# fex associate 101
```

アップリンクからダウンリンクまたはダウンリンクからアップリンクにポートを変更するためのポート プロファイルの設定

ポート プロファイルの設定

アップリンクおよびダウンリンク変換は、名前の末尾が EX か FX、またはそれ以降の Cisco Nexus 9000 シリーズ スイッチでサポートされます（たとえば、N9K-C9348GC-FXP または N9K-C93240YC-FX2）。変換後のダウンリンクに接続されている FEX もサポートされています。

この機能は次の Cisco スイッチでサポートされています。

- N9K-C9348GC-FXP（FEX をサポートしていません）
- N9K-C93180LC-EX
- N9K-C93180YC-FX および N9K-93180YC-EX

- N9K-C93108TC-EXおよびN9K-C93108TC-FX（アップリンクからダウンリンクへの変換のみサポート）
- N9K-C9336C-FX2
- N9K-C9336C-FX2-E
- N9K-C93240YC-FX2
- N9K-C93216TC-FX2
- N9K-C93360YC-FX2
- N9K-C93180YC-FX3（5.1(3) リリース以降）
- N9K-C93108TC-FX3P（5.1(3) リリース以降）
- N9K-C9364C-GX

アップリンクポートがダウンリンクポートに変換されると、他のダウンリンクポートと同じ機能を持つようになります。

制約事項

- FAST リンク フェールオーバー ポリシーとポート プロファイルは、同じポートではサポートされていません。ポート プロファイルが有効になっている場合、FAST リンク フェールオーバーを有効にすることはできません。その逆も同様です。
- サポートされているリーフ スイッチの最後の 2 つのアップリンク ポートは、ダウンリンク ポートに変換することはできません（これらはアップリンク接続用に予約されています）。
- ダイナミック ブレークアウト（100Gbと40Gbの両方）は、N9K-C93180YC-FX スイッチのプロファイルされたQSFPポートでサポートされます。ブレイクアウトおよびポートプロファイルでは、ポート 49-52 でアップリンクからダウンリンクへの変換が一緒にサポートされています。ブレークアウト（**10g-4x** オプションと **25g-4x** オプションの両方）は、ダウンリンク プロファイル ポートでサポートされます。
- N9K-C9348GC-FXP は FEX をサポートしていません。
- ブレークアウトはダウンリンク ポートでのみサポートされます。他のスイッチに接続されているファブリック ポートではサポートされません。
- Cisco ACI リーフスイッチは、56 を超えるファブリック リンクを持つことはできません。

ガイドライン

アップリンクをダウンリンクに変換したり、ダウンリンクをアップリンクに変換したりする際は、次のガイドラインにご注意ください。

サブジェクト	ガイドライン
ポート プロファイルを使用したノードのデコミッション	デコミッションされたノードがポートプロファイル機能を展開している場合、ポート変換はノードのデコミッション後も削除されません。ポートをデフォルト状態に戻すには、デコミッション後に手動で設定を削除する必要があります。これを行うには、スイッチにログインし、`setup-clean-config.sh -k` スクリプトを実行して、実行完了を待ちます。それから、`リロード` コマンドを入力します。`-k` スクリプトオプションを使用すると、ポートプロファイルの設定がリロード後も維持され、追加のリブートが不要になります。
最大アップリンク ポートの制限	最大アップリンク ポートの制限に達し、ポート 25 および 27 がアップリンクからダウンリンクへ返還されるとき、Cisco 93180LC EX スイッチのアップリンクに戻ります。 Cisco N9K-93180LC-EX スイッチでは、ポート 25 および 27 がオリジナルのアップリンク ポートです。ポート プロファイルを使用して、ポート 25 および 27 をダウンリンク ポートに変換する場合でも、ポート 29、30、31、および 32 は引き続き 4 つの元のアップリンク ポートとして使用できます。変換可能なポート数のしきい値のため（最大 12 ポート）、8 個以上のダウンリンク ポートをアップリンク ポートに変換できます。たとえば、ポート 1、3、5、7、9、13、15、17 はアップリンク ポートに変換されます。ポート 29、30、31、および 32 は、4 つの元からのアップリンク ポートです（Cisco 93180LC-EX スイッチでの最大アップリンク ポートの制限）。 スイッチがこの状態でポート プロファイル設定がポート 25 および 27 で削除される場合、ポート 25 および 27 はアップリンク ポートへ再度変換されますが、前述したようにスイッチにはすでに 12 個のアップリンク ポートがあります。ポート 25 および 27 をアップリンク ポートとして適用するため、ポート範囲 1、3、5、7、9、13、15、17 からランダムで 2 個のポートがアップリンクへの変換を拒否されます。この状況はユーザにより制御することはできません。 そのため、リーフ ノードをリロードする前にすべての障害を消去し、ポートタイプに関する予期しない問題を回避することが必須です。ポートプロファイルの障害を消去せずにノードをリロードすると、特に制限超過に関する障害の場合、ポートは予想される動作状態になることに注意する必要があります。

ブレークアウト制限

スイッチ	リリース	制限事項
N9K-C93180LC-EX	Cisco APIC 3.1(1) 以降	• 40 Gb と 100 Gb のダイナミック ブレークアウトは、ポート 1 ~ 24 の奇数ポート上でサポートされます。 • 上位ポート（奇数ポート）ブレークアウトされると、下部ポート（偶数ポート）はエラーが無効になります。 • ポート プロファイルおよびブレークアウトは、同じポートでサポートされていません。ただし、ポート プロファイルを適用してファブリック ポートをダウンリンクに変換してからであれば、ブレークアウト設定を適用できます。
N9K-C9336C-FX2-E	Cisco APIC 5.2(4) 以降	• 40Gb および 100Gb のダイナミック ブレークア ウトは、ポート 1 ー 34 でサポートされます。 • ポート プロファイルは、ブレークアウトが有効になっているポートには適用できません。ただし、ポート プロファイルを適用してファブリック ポートをダウンリンクに変換してからであれば、ブレークアウト設定を適用できます。 • 34 ポートすべてをブレークアウトポートとして設定できます。 • 34 のポートにブレークアウト設定を適用する場合は、34 のダウンリンク ポートを持つようにポートのポート プロファイルを設定してから、リーフ スイッチをリブートする必要があります。 • 複数のポートのリーフ スイッチにブレークアウト設定を同時に適用する場合、34 ポートのハードウェアがプログラムされるまでに最大 10 分かかります。プログラミングが完了するまで、ポートはダウンしたままになります。新しい設定の場合、クリーン リブート後、またはスイッチの検出中に遅延が発生する可能性があります。

スイッチ	リリース	制限事項
N9K-C9336C-FX2	Cisco APIC 4.2(4) 以降	• 40Gb および 100Gb のダイナミック ブレークア ウトは、ポート 1 ー 34 でサポートされます。 • ポート プロファイルは、ブレークアウトが有効になっているポートには適用できません。ただし、ポート プロファイルを適用してファブリック ポートをダウンリンクに変換してからであれば、ブレークアウト設定を適用できます。 • 34 ポートすべてをブレークアウトポートとして設定できます。 • 34 のポートにブレークアウト設定を適用する場合は、34 のダウンリンク ポートを持つようにポートのポート プロファイルを設定してから、リーフ スイッチをリブートする必要があります。 • 複数のポートのリーフ スイッチにブレークアウト設定を同時に適用する場合、34 ポートのハードウェアがプログラムされるまでに最大 10 分かかります。プログラミングが完了するまで、ポートはダウンしたままになります。新しい設定の場合、クリーン リブート後、またはスイッチの検出中に遅延が発生する可能性があります。
N9K-C9336C-FX2	Cisco APIC 3.2(1) 以降、ただし 4.2(4) は含まない	• ポート 1 ~ 30 では、40 Gb と 100 Gb のダイナミック ブレークがサポートされています。 • ポート プロファイルおよびブレークアウトは、同じポートでサポートされていません。ただし、ポート プロファイルを適用してファブリック ポートをダウンリンクに変換してからであれば、ブレークアウト設定を適用できます。 • 最大 20 のポートをブレークアウト ポートとして設定できます。

スイッチ	リリース	制限事項
N9K-C93180YC-FX	Cisco APIC 3.2(1) 以降	• 40 Gb と 100 Gb のダイナミック ブレークは、52、上にあるときにプロファイリング QSFP ポートがポート 49 でサポートされます。ダイナミック ブレークアウトを使用するには、次の手順を実行します。 • ポート 49~52 を前面パネル ポート（ダウンリンク）に変換します。 • 次の方法のいずれかを使用して、ポート プロファイルのリロードを実行します。 • APIC GUI で、**[ファブリック]** > **[インベントリ]** > **[ポッド]** > **[リーフ]** に移動し、**[シャーシ]** クリックしてから **[リロード]** を選択します。 • NX-OS スタイル CLI で、**setup-clean-config.sh -k** スクリプトを入力し、実行を待機し、**reload** コマンドを入力します。 • プロファイルされたポート 49 - 52 のブレーク アウトを適用します。 • ポート 53 および 54 では、ポート プロファイルまたはブレークアウトをサポートしていません。
N9K-C93240YC-FX2	Cisco APIC 4.0(1) 以降	ブレークアウトは変換後のダウンリンクではサポートされていません。

ポート プロファイルの設定のまとめ

次の表に、アップリンクからダウンリンク、およびダウンリンクからアップリンクへのポートプロファイル変換をサポートするスイッチでサポートされるアップリンクおよびダウンリンクをまとめます。

スイッチ モデル	デフォルト リンク	最大アップリンク (ファブリック ポート)	最大ダウンリンク (サーバのポート)	サポートされているリリース
N9K-C9348GC-FXP	48 x 100 M/1 G BASE-T ダウンリンク 4 x 10/25 Gbps SFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	48 x 100 M/1 G BASE-T ダウンリンク 4 x 10/25 Gbps SFP28 アップリンク 2 x 40/100 Gbps QSFP28 アップリンク	デフォルトのポート設定と同じ	3.1(1)
N9K-C93180LC-EX	24 X 40 Gbps QSFP28 ダウンリンク（1 ー 24） 2 x 40/100 Gbps QSFP28 アップリンク（25、27） 4 x 40/100 Gbps QSFP28 アップリンク（29 ー 32） または 12 X 100 Gbps QSFP28 ダウンリンク（1 ー 24 の範囲の奇数） 2 x 40/100 Gbps QSFP28 アップリンク（25、27） 4 x 40/100 Gbps QSFP28 アップリンク（29 ー 32）	18 X 40 Gbps QSFP28 ダウンリンク（1 ー 24） 6 X 40 Gbps QSFP28 アップリンク（1 ー 24） 2 x 40/100 Gbps QSFP28 アップリンク（25、27） 4 x 40/100 Gbps QSFP28 アップリンク（29 ー 32） または 6 x 100 Gbps QSFP28 ダウンリンク（1 ー 24 の範囲の奇数） 6 x 100 Gbps QSFP28 アップリンク（1 ー 24 の範囲の奇数） 2 x 40/100 Gbps QSFP28 アップリンク（25、27） 4 x 40/100 Gbps QSFP28 アップリンク（29 ー 32）	24 X 40 Gbps QSFP28 ダウンリンク（1 ー 24） 2 x 40/100 Gbps QSFP28 ダウンリンク（25、27） 4 x 40/100 Gbps QSFP28 アップリンク（29 ー 32） または 12 X 100 Gbps QSFP28 ダウンリンク（1 ー 24 の範囲の奇数） 2 x 40/100 Gbps QSFP28 ダウンリンク（25、27） 4 x 40/100 Gbps QSFP28 アップリンク（29 ー 32）	3.1(1)

スイッチ モデル	デフォルト リンク	最大アップリンク (ファブリック ポート)	最大ダウンリンク (サーバのポート)	サポートされているリリース
N9K-C93180YC-EX N9K-C93180YC-FX N9K-C93180YC-FX3	48 x 10/25 Gbps ファイバ ダウンリンク 6 x 40/100 Gbps QSFP28 アップリンク	デフォルトのポート設定と同じ	48 x 10/25 Gbps ファイバ ダウンリンク 4 x 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	3.1(1)
		48 X 10/25 Gbps ファイバ アップリンク 6 x 40/100 Gbps QSFP28 アップリンク		4.0(1) 5.1(3)
N9K-C93108TC-EX N9K-C93108TC-FX N9K-C93108TC-FX3	48 x 10GBASE T ダウンリンク 6 x 40/100 Gbps QSFP28 アップリンク	デフォルトのポート設定と同じ	48 x 10/25 Gbps ファイバ ダウンリンク 4 x 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	3.1(1) 4.0(1) 5.1(3)

スイッチ モデル	デフォルト リンク	最大アップリンク (ファブリック ポート)	最大ダウンリンク (サーバのポート)	サポートされているリリース
N9K-C9336C-FX2	30 x 40/100 Gbps QSFP28 ダウンリンク 6 x 40/100 Gbps QSFP28 アップリンク	18 x 40/100 Gbps QSFP28 ダウンリンク 18 x 40/100 Gbps QSFP28 アップリンク	デフォルトのポート設定と同じ	3.2(1)
		18 x 40/100 Gbps QSFP28 ダウンリンク 18 x 40/100 Gbps QSFP28 アップリンク	34 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	3.2(3)
		36 x 40/100-Gbps QSFP28 アップリンク	34 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	4.1(1)
N9K-C9336C-FX2-E	30 x 40/100 Gbps QSFP28 ダウンリンク 6 x 40/100 Gbps QSFP28 アップリンク	36 x 40/100-Gbps QSFP28 アップリンク	34 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	5.2(4)
N9K-93240YC-FX2	48 x 10/25 Gbps ファイバ ダウンリンク 12 x 40 / 100Gbps QSFP28 アップリンク	デフォルトのポート設定と同じ	48 x 10/25 Gbps ファイバ ダウンリンク 10 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	4.0(1)
		48 X 10/25 Gbps ファイバ アップリンク 12 x 40 / 100Gbps QSFP28 アップリンク		4.1(1)

スイッチ モデル	デフォルト リンク	最大アップリンク (ファブリック ポート)	最大ダウンリンク (サーバのポート)	サポートされているリリース
N9K-C93216TC-FX2	96 X 10G BASE-T ダウンリンク 12 x 40 / 100Gbps QSFP28 アップリンク	デフォルトのポート設定と同じ	96 X 10G BASE-T ダウンリンク 10 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	4.1(2)
N9K-C93360YC-FX2	96 X 10/25 Gbps SFP28 ダウンリンク 12 x 40 / 100Gbps QSFP28 アップリンク	44 x 10 / 25Gbps SFP28 ダウンリンク 52 x 10 / 25Gbps SFP28 アップリンク 12 x 40 / 100Gbps QSFP28 アップリンク	96 X 10/25 Gbps SFP28 ダウンリンク 10 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	4.1(2)
N9K-C93600CD-GX	28 X 40/100 Gbps QSFP28 ダウンリンク 8 x 40/100/400 Gbps QSFP-DD アップリンク	28 X 40/100 Gbps QSFP28 アップリンク 8 x 40/100/400 Gbps QSFP-DD アップリンク	28 X 40/100 Gbps QSFP28 ダウンリンク 6 X 40/100/400 Gbps QSFP-DD ダウンリンク 2 x 40/100/400 Gbps QSFP-DD アップリンク	4.2(2)
N9K-C9364C-GX	48/40/100 Gbps QSFP28 ダウンリンク 16 X 40/100 Gbps QSFP28 アップリンク	64 X 40/100 Gbps QSFP28 アップリンク	62 X 40/100 Gbps QSFP28 ダウンリンク 2 x 40/100 Gbps QSFP28 アップリンク	4.2(3)

スイッチ モデル	デフォルト リンク	最大アップリンク (ファブリック ポート)	最大ダウンリンク (サーバのポート)	サポートされているリリース
N9K-C9316D-GX	12 X 40/100/400 Gbps QSFP-DD ダウンリンク 4 x 40/100/400 Gbps QSFP-DD アップリンク	16 X 40/100/400 Gbps QSFP-DD アップリンク	14 x 40/100/400 Gbps QSFP-DD ダウンリンク	5.1(4)
N9K-C9332D-GX2B	2 X 1/10 Gbps SFP+ ダウンリンク 24/40/100/400 Gbps QSFP-DD ダウンリンク 8 x 40/100/400 Gbps QSFP-DD アップリンク	2 X 1/10 Gbps SFP+ ダウンリンク 32 X 40/100/400 Gbps QSFP-DD アップリンク	2 X 1/10 Gbps SFP+ ダウンリンク 30 X 40/100/400 Gbps QSFP-DD ダウンリンク 2 x 40/100/400 Gbps QSFP-DD アップリンク	5.2(3)
N9K-C9348D-GX2A	2 X 1/10 Gbps SFP+ ダウンリンク 36 X 40/100/400 Gbps QSFP-DD ダウンリンク 12 x 40/100/400 Gbps QSFP-DD アップリンク	2 X 1/10 Gbps SFP+ ダウンリンク 48 x 40/100/400 Gbps QSFP-DD アップリンク	2 X 1/10 Gbps SFP+ ダウンリンク 46 X 40/100/400 Gbps QSFP-DD ダウンリンク 2 x 40/100/400 Gbps QSFP-DD アップリンク	5.2(5)
N9K-C9364D-GX2A	2 X 1/10 Gbps SFP+ ダウンリンク 48 X 40/100/400 Gbps QSFP-DD ダウンリンク 16 X 40/100/400 Gbps QSFP-DD アップリンク	2 X 1/10 Gbps SFP+ ダウンリンク 64 x 40/100/400 Gbps QSFP-DD アップリンク	2 X 1/10 Gbps SFP+ ダウンリンク 62 X 40/100/400 Gbps QSFP-DD ダウンリンク 2 x 40/100/400 Gbps QSFP-DD アップリンク	5.2(5)

GUI を使用したポート プロファイルの設定

この手順では、ポート タイプ (アップリンクまたはダウンリンク) を決定するポート プロファイルを設定する方法について説明します。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成または変更できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。

手順

ステップ 1 [Fabric] メニューから、[Inventory] を選択します。

ステップ 2 [Inventory] 画面左側のナビゲーション ウィンドウで、[Topology] を選択します。

ステップ 3 [Topology] タブで、右側ナビゲーション ウィンドウの[Interface] タブを選択します。

ステップ 4 [Configuration] モードを選択します。

ステップ 5 テーブル メニューの [+] アイコン ([Add Switches]) をクリックして、リーフ スイッチを追加します。

ステップ 6 [Add Switches] テーブルで、 [Switch ID] を選択し、 [Add Selected] をクリックします。

ポートを選択すると、使用可能なオプションが強調表示されます。

ステップ 7 ポートを選択し、新しいポート タイプとして [Uplink] または [Downlink] を選択します。

最後の2つのポートはアップリンク用に予約されます。これらをダウンリンク ポートに変換することはできません。

ステップ 8 アップリンクまたはダウンリンクをクリックしてから、 [Submit] (後ほど自分でスイッチをリロードする場合) または [Submit and Reload Switch] をクリックします。

（注） ダウンリンクをアップリンクに、またはアップリンクをダウンリンクに変換した後、GUI または CLI の `reload` コマンドを使用してスイッチをリロードする必要があります。スイッチの電源の再投入では不十分です。

NX-OS スタイル CLI を使用したポート プロファイルの設定

NX-OS スタイルの CLI を使用したポート プロファイルの設定をするには、次の手順を実行します。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。

- 必要なファブリック インフラストラクチャ設定を作成または変更できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。

手順

ステップ 1 **configure**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ 2 **leaf** *node-id*

設定するリーフまたはリーフ スイッチを指定します。

例：

```
apic1(config)# leaf 102
```

ステップ 3 **interface** *type*

設定するインターフェイスを指定します。インターフェイス タイプと ID を指定できます。イーサネット ポートの場合は、ethernet *slot* / *port* を使用します。

例：

```
apic1(config-leaf)# interface ethernet 1/2
```

ステップ 4 **port-direction** {**uplink** | **downlink**}

ポートの方向を決定するか変更します。この例ではダウンリンクにポートを設定します。

（注） N9K-C9336C-FX スイッチでは、アップリンクからダウンリンクへの変更はサポートされていません。

例：

```
apic1(config-leaf-if)# port-direction downlink
```

ステップ 5 ポートがあるリーフ スイッチにログインし、**setup-clean-config.sh -k** コマンドを入力してから **reload** コマンドを入力します。

REST API を使用したポート プロファイルの設定

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。

- 必要なファブリック インフラストラクチャ設定を作成または変更できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。

手順

ステップ 1 ダウンリンクからアップリンクへ変換するポートプロファイルを作成するには、次の例のように XML で POST 送信します。

```
<!-- /api/node/mo/uni/infra/prtdirec.xml -->
<infraRsPortDirection tDn="topology/pod-1/paths-106/pathep-[eth1/7]" direc="UpLink" />
```

ステップ 2 アップリンクからダウンリンクへ変換するポートプロファイルを作成するには、次のように、XML で post を送信します。

例：

```
<!-- /api/node/mo/uni/infra/prtdirec.xml -->
<infraRsPortDirection tDn="topology/pod-1/paths-106/pathep-[eth1/52]" direc="DownLink"
/>
```

NX-OS スタイル CLI を使用したポート プロファイルの設定と変換の確認

show interface brief CLI コマンドを使用して、ポートの設定と変換を確認することができます。

（注） ポート プロファイルは、Cisco N9K-C93180LC EX スイッチのトップ ポートにのみ展開されます。たとえば、1、3、5、7、9、11、13、15、17、19、21、および 23 となります。ポートプロファイルを使用してトップ ポートを変換すると、ボトム ポートはハードウェア的に無効になります。たとえば、ポート プロファイルを使用して Eth 1/1 を変換すると、Eth 1/2 はハードウェア的に無効になります。

手順

ステップ 1 この例では、アップリンク ポートをダウンリンク ポートに変換する場合の出力を示しています。アップリンク ポートをダウンリンク ポートに変換変換する前に、この例での出力が表示されます。**routed** というキーワードは、ポートがアップリンク ポートであることを示しています。

例：

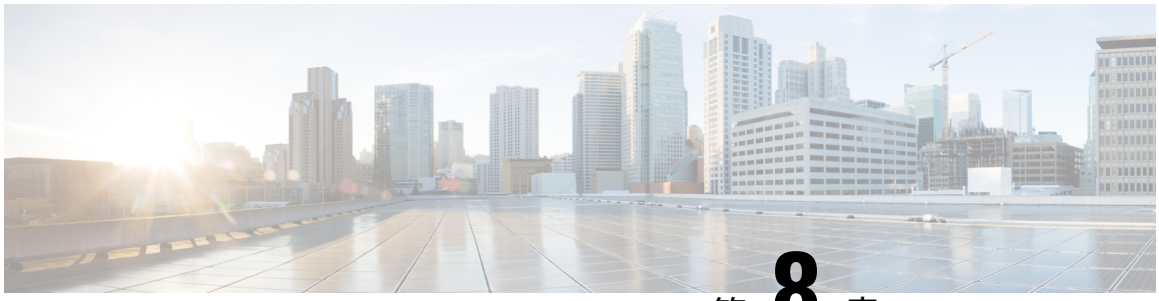
```
switch# show interface brief
```

```
<snip>
Eth1/49          --      eth  routed  down   sfp-missing              100G(D)   --
Eth1/50          --      eth  routed  down   sfp-missing              100G(D)   --
<snip>
```

ステップ 2 ポートプロファイルを設定して、スイッチのリロード、後に、例では、出力が表示されます。キーワード **トランク** ダウンリンク ポートとしてポートを示します。

例：

```
switch# show interface brief
<snip>
Eth1/49          0       eth  trunk   down   sfp-missing              100G(D)   --
Eth1/50          0       eth  trunk   down   sfp-missing              100G(D)   --
<snip>
```



第 8 章

FCoE 接続

この章は、次の内容で構成されています。

Cisco ACI ファブリックでの Fibre Channel over Ethernet トラフィックのサポート

Cisco Application Centric Infrastructure（ACI）では、Cisco ACI ファブリック上の Fibre Channel over Ethernet（FCoE）に対するサポートを設定して、管理することができます。

FCoE は、ファイバ チャネル パケットをイーサネット パケット内にカプセル化するプロトコルです。これにより、ストレージ トラフィックをファイバ チャネル SAN とイーサネット ネットワーク間でシームレスに移動できます。

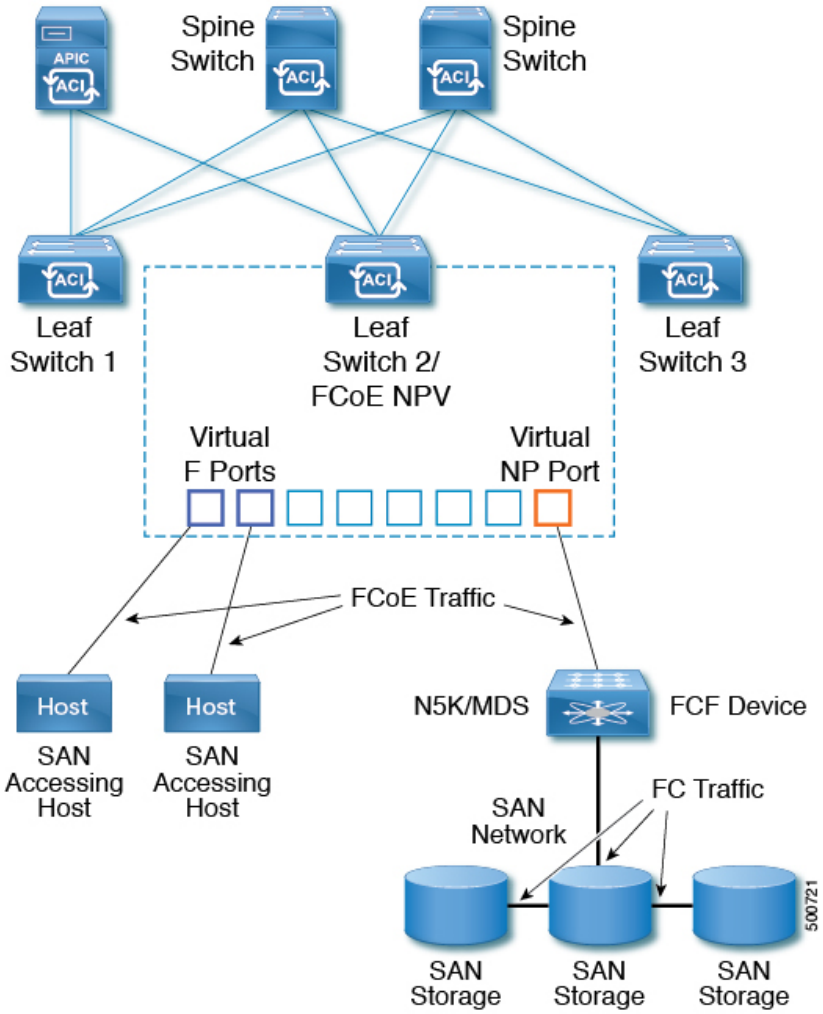
Cisco ACI ファブリックで FCoE プロトコルのサポートを標準実装することにより、イーサネットベースの Cisco ACI ファブリックに配置されているホストが、ファイバ チャネル ネットワークに配置されている SAN ストレージデバイスと通信できます。ホストは、Cisco ACI リーフスイッチに展開された仮想 F ポートを介して接続しています。SAN ストレージ デバイスとファイバ チャネル ネットワークは、ファイバチャネルフォワーディング（FCF）ブリッジおよび仮想 NP ポートを介して Cisco ACIファブリックに接続されます。このポートは、仮想 F ポートと同じ Cisco ACI リーフスイッチに導入されます。仮想 NP ポートおよび仮想 F ポートも汎用的に仮想ファイバ チャネル (vFC) ポートに呼ばれます。

（注） FCoE トポロジにおける Cisco ACI リーフ スイッチの役割は、ローカル接続された SAN ホストとローカル接続された FCF デバイスの間で、 FCoE トラフィックのパスを提供することです。リーフ スイッチでは SAN ホスト間のローカル スイッチングは行われず、FCoE トラフィックはスパイン スイッチに転送されません。

Cisco ACI を介した FCoE トラフィックをサポートするトポロジ

Cisco ACIファブリック経由のFCoEトラフィックをサポートする一般的な設定のトポロジは、次のコンポーネントで構成されます。

図 *28 : Cisco ACI FCoE* トラフィックをサポートするトポロジ



- NPV バックボーンとして機能するようにファイバ チャネル SAN ポリシーを通して設定されている 1 つ以上の Cisco ACI リーフ スイッチ。

- 仮想 F ポートとして機能するように設定された NPV 設定リーフ スイッチ上で選択されたインターフェイス。SAN管理アプリケーションまたはSANを使用しているアプリケーションを実行しているホストとの間を往来する FCoE トラフィックの調整を行います。
- 仮想 NP ポートとして機能するように設定された NPV 設定リーフ スイッチ上で選択されたインターフェイス。ファイバ チャネル転送（FCF）ブリッジとの間を往来する FCoE トラフィックの調整を行います。

FCF ブリッジは、通常 SAN ストレージ デバイスを接続しているファイバ チャネルリンクからファイバ チャネル トラフィックを受信し、ファイバ チャネル パケットを FCoE フレームにカプセル化して、Cisco ACI ファブリック経由で SAN 管理ホストまたは SAN データ消費ホストに送信します。FCoE トラフィックを受信し、ファイバ チャネルに再パッケージしてファイバ チャネル ネットワーク経由で伝送します。

（注） 前掲の Cisco ACI トポロジでは、FCoE トラフィックのサポートには、ホストと仮想 F ポート間の直接接続、および、FCF デバイスと仮想 NP ポート間の直接接続が必要です。

Cisco Application Policy Infrastructure Controller（APIC）サーバーは、Cisco APIC GUI、NX-OS スタイルの CLI、または REST API へのアプリケーション コールを使用して、FCoE トラフィックを設定およびモニタできます。

FCoE の初期化をサポートするトポロジ

FCoE トラフィック フローが説明の通り機能するためには、別の VLAN 接続を設定する必要があります。SAN ホストこの接続を経由して、FCoE 初期化プロトコル（FIP）パケットをブロードキャストし、F ポートとして有効にされているインターフェイスを検出します。

vFC インターフェイス設定ルール

Cisco APIC GUI、NX-OS スタイル CLI、または REST API のいずれを使用して vFC ネットワークと EPG の導入を設定する場合でも、次の一般的なルールがプラットフォーム全体に適用されます。

- F ポート モードは、vFC ポートのデフォルト モードです。NP ポート モードは、インターフェイス ポリシーで具体的に設定する必要があります。
- デフォルトのロード バランシング モードはリーフ スイッチ、またはインターフェイス レベル vFC 設定が src dst ox id。
- ブリッジ ドメインごとに 1 つの VSAN 割り当てがサポートされます。
- VSAN プールおよび VLAN プールの割り当てモードは、常にスタティックである必要があります。
- vFC ポートでは、VLAN にマッピングされている VSAN を含む VSAN ドメイン（ファイバ チャネル ドメインとも呼ばれます）との関連付けが必要です。

Fibre Channel over Ethernet のガイドラインと制限事項

FCoE に使用する VLAN の vlanScope を `Global` に設定する必要があります。vlanScope を `portLocal` に設定することは、FCoE ではサポートされていません。値は、レイヤ 2 インターフェイス ポリシー（l2IfPol）を使用して設定されます。

Fibre Channel over Ethernet（FCoE）をサポートするハードウェア

FCoE は、次のスイッチでサポートされます。

- N9K-C93180LC-EX

  40 ギガビットイーサネット（GE）ポートが FCoE F または NP ポートとして有効になっている場合、40GE ポートブレークアウトを有効にすることはできません。FCoE は、ブレークアウト ポートではサポートされません。

- N9K-C93108TC-FX
- N9K-C93108TC-EX（FCoE NPVのみ）
- N9K-C93180YC-EX
- N9K-C93180LC-EX

  FEX ポートでの FCoE がサポートされます。

- N9K-C93180YC-FX

  サポート対象は、10/25G ポート（1ー48）、40G ポート（1/49ー54）、4x10G ブレークアウト ポート（1/49ー54）、および FEX ポート上の FCoE です。

FCoE は、次の Nexus FEX デバイスでサポートされます。

- 10 ギガ-ビット C2348UPQ N2K
- 10 ギガ-ビット C2348TQ N2K
- N2K-C2232PP-10GE
- N2K-B22DELL-P
- N2K-B22HP-P
- N2K-B22IBM-P
- N2K B22DELL P FI

APIC GUI を使用した FCoE の設定

FCoE GUI の設定

FCoE ポリシー、プロファイル、およびドメインの設定

[Fabric Access Policies] タブで APIC GUI を使用すれば、ポリシー、ポリシー グループ、およびプロファイルを設定して、ACI リーフ スイッチ上の F および NP ポートをサポートする FCoE のカスタマイズされ、スケールアウトした展開と割り当てを行うことが可能になります。次に、APIC の [Tenant] タブで、では、これらのポートへの EPG アクセスを設定できます。

ポリシーおよびポリシー グループ

FCoE のサポートのために作成または設定する APIC ポリシーとポリシー グループには、次のものが含まれます:

アクセス スイッチ ポリシー グループ

ACI リーフ スイッチを通して FCoE トラフィックをサポートする、スイッチ レベルのポリシーの組み合わせです。

このポリシー グループをリーフ プロファイルと関連付けて、指定された ACI リーフ スイッチでの FCoE サポートを有効にすることができます。

このポリシー グループは、次のポリシーで構成されています:

- **ファイバ チャネル SAN ポリシー**

  NPV リーフが使用する、EDTOV、RATOV、および MAC アドレス プレフィックス (FC マップとも呼ばれる) の値を指定します。

- **ファイバ チャネル ノード ポリシー**

  このポリシーグループに関連付けられる FCoE トラフィックに適用される、ロードバランス オプションと FIP キープ アライブ間隔を指定します。

インターフェイス ポリシー グループ

ACI リーフ スイッチのインターフェイスを通して FCoE トラフィックをサポートする、インターフェイス レベルのポリシーの組み合わせです。

このポリシー グループを FCoE のサポート的インターフェイス プロファイルと関連付けて、指定したインターフェイスでの FCoE サポートを有効にすることができます。

2 つのインターフェイス ポリシー グループを設定できます。F ポートの 1 つのポリシー グループと、 NP ポートの 1 つのポリシー グループです。

インターフェイス ポリシー グループの以下のポリシーは、FCoE の有効化およびトラフィックに適用されます:

- 優先順位フロー制御ポリシー

  このポリシー グループが適用されているインターフェイスの優先順位フロー制御(PFC)の状態を指定します。

  このポリシーは、どのような状況で QoS レベルの優先順位フロー制御が FCoE トラフィックに適用されるかを指定します。

- **Fibre Channel Interface Policy**

  このポリシー グループが適用されているインターフェイスが F ポートまたは NP ポートとして設定されるかどうかを指定します。

- 低速ドレイン ポリシー

  ACI ファブリックでトラフィックの輻輳の原因となる FCoE パケットを処理するためのポリシーを指定します。

グローバル ポリシー

設定により、ACI ファブリックの FCoE トラフィックのパフォーマンス特性に影響を及ぼす APIC グローバル ポリシーです。

グローバル **QOS クラス ポリシー**（**Level1**、**Level2**、**Level4**、**Level5**、または**Level6** 接続に対応するもの）には、ACI ファブリック上の FCoE トラフィックに影響する次の設定が含まれます。

- **[PFC Admin State] は Auto に設定することが必要**

  FCoE トラフィックのこのレベルで優先順位フロー制御を有効にするかどうかを指定します (デフォルト値は false です)。

- **No Drop COS**

  特定のサービス クラス (CoS) レベルで指定された FCoE トラフィックのこのレベルに対し、no-drop ポリシーを有効にするかどうかを指定します。

  注: PFC および FCoE ノードロップに対して有効にされている QoS レベルは、CNA 上の PFC に対して有効にされている優先順位グループ ID と一致している必要があります。

  注: ノードロップおよび PFC に対して有効にできるのは、ただ 1 つの QoS レベルです。そして同じ QoS レベルが FCoE Epg に関連付けられている必要があります。

- QoS クラス — 優先順位フロー制御は、CoS レベルがファブリックに対してグローバルに有効にされていること、そしてFCoE トラフィックを生成するアプリケーションのプロファイルに割り当てられていることを必要とします。

  CoS保存も有効にする必要があります。**[ファブリック (Fabric)]** > **[アクセス ポリシー (Access Policies)]** > **[ポリシー (Policies)]** > **[グローバル (Global)]** > **[QoS クラス (QoS Class)]**荷移動して、**[COS Dot1P Preserve を保存 (Preserve COS Dot1p Preserve)]** を有効にします

（注） 一部のレガシー CNA も、**レベル 2** グローバル QoS ポリシーが、**ノードロップ** PFC、FCoE (Fibre Channel over Ethernet) QoS ポリシーで使用されていることを必要とする場合があります。使用しているコンバージド ネットワーク アダプタ （CNA） がファブリックにロギングしておらず、CNA から FCoE Initiation Protocol (FIP) フレームが送信されていないことがわかった場合には、 **レベル 2** を FCoE QoSポリシーとして有効にしてみてください。**Level2** ポリシーは、使用中の FCoE EPG にアタッチする必要があり、PFC no-drop に対して 1 つの QoS レベルのみを有効にできます。

プロファイル

FCoE をサポートするために作成または設定ができる APIC プロファイルとしては、次のものがあります:

リーフ プロファイル

FCoE トラフィックのサポートが構成される、ACI ファブリック リーフ スイッチを指定します。

アクセス スイッチ ポリシー グループに含まれるポリシーの組み合わせは、このプロファイルに含まれるリーフ スイッチに適用できます。

インターフェイス プロファイル

F ポートまたは NP ポートが展開される一連のインターフェイスを指定します。

少なくとも 2 つのリーフ インターフェイス プロファイルを設定します。一方は F ポートのインターフェイスプロファイルで、もう一方はNPポートのインターフェイスプロファイルです。

F ポートのインターフェイス ポリシー グループに含まれるポリシーの組み合わせは、F ポートのインターフェイスプロトコルに含まれている一連のインターフェイスに適用できます。

NPポートのインターフェイス ポリシー グループに含まれるポリシーの組み合わせは、NP ポートのインターフェイスプロトコルに含まれている一連のインターフェイスに適用できます。

アタッチ エンティティ プロファイル

インターフェイス ポリシ - グループの設定をファイバチャネル ドメイン マッピングにバインドします。

ドメイン

FCoEをサポートするために作成または設定ができるドメインとしては、次のものがあります:

物理ドメイン

FCoE VLAN ディスカバリのための LANをサポートするため作成された仮想ドメイン。物理ドメインは、FCoE VLAN ディスカバリをサポートするための VLAN プールを指定します。

ファイバ チャネル ドメイン

FCoE 接続のための仮想 SAN をサポートするため作成された仮想ドメイン。

ファイバ チャネル ドメインは、FCoE トラフィックが搬送される VSAN プール、VLAN プールおよび VSAN 属性を指定します。

- **VSAN プール** - 既存の VLAN に関連付けられた仮想 SAN のセット。個々 の VSAN は、VLANをイーサネット接続のためのインターフェイスに割り当てるのと同じ方法で、関連付けられた FCoE 対応のインターフェイスに割り当てることができます。
- **VLAN プール** - 個々 の VSAN に関連付けることができる VLAN のセット。
- **VSAN 属性** - VSAN から VLAN へのマッピング。

テナント エンティティ

[テナント] タブでは、ブリッジ ドメインおよびEPGエンティティを、FCoE ポートにアクセスし、FCoE トラフィックを交換するように設定します。

エンティティには、次のものがあります:

ブリッジ ドメイン (FCoE サポートのために設定されたもの)

テナントの下で、FCoE 接続を使用するアプリケーションのために FCoE トラフィックを送るように作成され、設定されたブリッジ ドメイン。

アプリケーション EPG

同じテナントの下で FCoE ブリッジ ドメインと関連付けられる EPG。

ファイバ チャネル パス

FCoE F ポートまたは NP ポートとして有効にされ、選択した EPG に関連付けられるインターフェイスを指定します。ファイバチャネルのパスをEPGに関連付けると、FCoEインターフェイスはが指定された VSAN に展開されます。

APIC GUI を使用した FCoE vFC ポートの展開

APIC GUI では、カスタマイズされたノード ポリシー グループ、リーフ プロファイル、インターフェイス ポリシー グループ、インターフェイス プロファイル、仮想 SAN ドメインを作成し、システム管理者が F ポートまたは NP ポートとして指定するすべてのインターフェイスを再利用して、整合性のある FCoE 関連ポリシーが適用されている FCoE トラフィックを処理できます。

始める前に

- ACI ファブリックがインストールされています。
- ポート チャネル (PC) トポロジ上で導入する場合、ポート チャネルはGUI を使用した ACI リーフ スイッチのポート チャネルの構成 （94 ページ）の説明に従ってセットアップします。
- 仮想ポートチャネル（vPC）トポロジを介して展開する場合は、GUI を使用した ACI リーフ スイッチの仮想ポート チャネルの設定 （115 ページ）の説明に従って vPC が設定されます。

手順

ステップ 1 FCoE 補助スイッチ ポリシー グループを作成し、FCoE 設定をサポートするすべてのリーフ スイッチ ポリシーを指定して組み合わせます。

このポリシー グループは、NPV ホストとして機能するリーフ スイッチに適用されます。

a) APIC GUI で、APIC のメニュー バーから **[Fabric]** > **[Access Policies]** > **[Switches]** > **[Leaf Switches]** > **[Policy Groups]** の順にクリックします。
b) **Policy Groups** を右クリックして、**Create Access Switch Policy Group** をクリックします。
c) [Create Access Switch Policy Group] ダイアログボックスで、以下で説明する設定を指定して、[Submit] をクリックします。

ポリシー	説明
名前	スイッチ ポリシー グループを識別します。 このスイッチ ポリシー グループの FCoE 補助機能を示す名前を入力します。たとえば、 **`fcoe_switch_policy_grp`** のようにします。
ファイバ チャネル SAN ポリシー	次の SAN ポリシーの値を指定します: • FC プロトコルの EDTOV (デフォルト: 2000) • FC プロトコルの RATOV (デフォルト: 10000) • リーフ スイッチが使用する MAC アドレスのプレフィックス (FC マップとも呼ばれます)。この値は、同じポートに接続されているピア デバイスの値と一致する必要があります。通常、デフォルト値の OE:FC:00 が使用されます。 ドロップダウン オプション ボックスをクリックします。 • デフォルトの EDTOV、RATOV、および MAC アドレスのプレフィックス値を使用するには、**default** をクリックします。 • 既存のポリシーで指定した値を使用するには、そのポリシーをクリックします。

ポリシー	説明
	• カスタマイズした新しいMACアドレスプレフィックスを指定する新しいポリシーを作成するには、[Create Fibre Channel SAN Policy] をクリックして、プロンプトに従います。

ステップ 2 FCoE トラフィックをサポートするリーフ スイッチのリーフ プロファイルを作成します。

このプロファイルは、前の手順で設定されたスイッチ ポリシー グループを割り当てるスイッチまたはリーフ スイッチの設定を指定します。この関連付けにより、事前定義されたポリシー設定で FCoE トラフィックをサポートするスイッチの設定を有効にします。

a) APIC メニュー バーから、**[Fabric]** > **[Access Policies]** > **[Switches]** > **[Leaf Switches]** > **[Profiles]** の順にクリックします。
b) **[リーフ プロファイル]** を右クリックし、**[リーフ プロファイルの作成]** をクリックします。
c) **[リーフ プロファイルの作成]** ダイアログで、リーフ プロファイルを作成し名前を付けます（例：NPV 1）
d) また、**Create Leaf Profile** ダイアログの **Leaf Selectors** テーブルで、+ をクリックしてテーブルで新しい行を作成し、NPV デバイスとして動作するリーフ スイッチを指定します。
e) テーブルの新しい行で、リーフ名とブロックを選択し、前のステップで作成したスイッチ ポリシー グループを割り当てます。
f) **[Next（次へ）]** をクリックし、さらに **[Finish（終了）]** をクリックします。

ステップ 3 少なくとも 2 個の FCoE 補助インターフェイス ポリシー グループの作成：1 個は FCoE F ポート インターフェイスをサポートするすべてのポリシーを組み合わせ、1 個は FCoE NP ポートをサポートしているすべてのポリシーを組み合わせるためのものです。

これらのインターフェイス ポリシー グループは、F ポートおよびNP ポートとして使用されるインターフェイスに適用されるインターフェイスのプロファイルに適用します。

a) APIC メニュー バーで、**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Policy Groups]** の順にクリックします。
b) **[Policy Groups]** を右クリックし、ポート アクセスの設定方法に応じて、**[Create Leaf Access Port Policy Group]**、**[Create PC Interface Port Policy]**、または **[Create vPC Interface Port Policy Group]** のいずれかのオプションをクリックします。

（注）
- PC インターフェイスで展開する場合、追加情報については GUI を使用した ACI リーフ スイッチのポート チャネルの構成 （94 ページ） を参照してください。
- vPC インターフェイスを介して展開する場合は、GUI を使用した ACI リーフ スイッチの仮想ポート チャネルの設定 （115 ページ）で詳細を確認してください。

c) ポリシー グループ ダイアログで、設定するファイバチャネルインターフェイス ポリシー、低速ドレイン ポリシー、優先順位フロー制御ポリシーを含むように指定します。

ポリシー	説明
名前	このポリシー グループの名前。 このリーフアクセスポートのポリシーグループとポートタイプ（FまたはNP）の補助機能を示す、サポートを意図した名前を入力します。**`fcoe_f_port_policy`** または **`fcoe_np_port_policy`**。
優先順位フロー制御ポリシー	このポリシー グループが適用されているインターフェイスの優先順位 フロー制御（PFC）の状態を指定します。 オプションには、次のものが含まれます。 • **[自動]**（デフォルト値）DCBXによってアドバタイズされ、ピアとの交渉が正常に行われた値を条件として、設定されている非ドロップ CoS のローカル ポートで、優先順位フロー制御（PFC）を有効にします。障害により、非ドロップ CoS 上で優先順位フロー制御が無効になります。 • **[オフ]** 機能によりあらゆる状況下で、ローカル ポートの FCoE 優先順位フロー制御を無効にします。 • **[オン]** 機能によりあらゆる状況下で、ローカル ポートの FCoE 優先順位フロー制御を有効にします。 ドロップダウン オプション ボックスをクリックします。 • デフォルト値を使用するには、**[デフォルト]** をクリックします。 • 既存のポリシーで指定した値を使用するには、そのポリシーをクリックします。 • 別の値を指定する新しいポリシーを作成するには、**[優先順位フロー制御ポリシーの作成]** をクリックし、指示に従います。 （注） PFC では、サービス クラス（CoS）レベルがファブリックに対してグローバルに有効になり、FCoE トラフィックを生成するアプリケーションのプロファイルに割り当てられている必要があります。また、CoS 保持が有効になっている必要があります。有効にするには、**[Fabric]** > **[Access Policies]** > **[Policies]** > **[Global]** > **[QoS Class]** に移動して、[Preserve COS Dot1p Preserve] を有効にします。
低速ドレインポリシー	ACI ファブリックでトラフィック輻輳を引き起こすFCoEパケットを処理する方法を指定します。オプションには、次のものが含まれます。 • 輻輳クリア アクション（デフォルト：無効） FCoE トラフィックの輻輳時に実行するアクション。次のオプションがあります。 • エラー：無効：ポートを無効にします。 • ログ：イベントログの輻輳を記録します。

ポリシー	説明
	• 無効：実行しません。 • 輻輳検出乗数（デフォルト：10） FCoE トラフィック輻輳に対処するため輻輳クリア アクションをトリガするポート上で受信した一時停止フレーム数。 • フラッシュ管理状態 • 有効：バッファをフラッシュします。 • 無効：バッファをフラッシュしません。 • フラッシュのタイムアウト（デフォルト：500 ミリ秒単位） 輻輳時にバッファのフラッシュをトリガするしきい値（ミリ秒）。 • デフォルト値を使用するには、**[デフォルト]** をクリックします。 • 既存のポリシーで指定した値を使用するには、そのポリシーをクリックします。 • 別の値を指定する新しいポリシーを作成するには、**[低速ドレイン ポリシーの作成]** をクリックしてプロンプトに従います。

ステップ 4 少なくとも 2 個のインターフェイス プロファイルの作成：1 個は F ポート接続をサポートするプロファイル、1 個は NP ポート接続をサポートするプロファイル、追加ポート ポリシーの変数に関連付けるオプションの追加プロファイル。

a) APIC バー メニューで、**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Profiles]** をクリックします。
b) **Profiles** を右クリックし、**Create Leaf Interface Profile** を選択します。
c) [Create Leaf Interface Profile] ダイアログで、たとえば「FCoE_F_port_Interface_profile-1」など、プロファイルを説明する名前を入力します。
d) インターフェイスの [Interface Selectors] テーブルで、[+] をクリックして [Create Access Port Selector] ダイアログを表示します。このダイアログを使用すると、インターフェイスの範囲を表示し、次の表に記載されたフィールドに設定を適用できます。

オプション	説明
名前	このポート セレクタを説明する名前。
Interface IDs	この範囲が適用されるインターフェイスの設定を指定します。 • スイッチにすべてのインターフェイスを含むには、**[すべて]** を選択します。 • この範囲に個々のインターフェイスを含めるには、たとえば **1/20** など単一のインターフェイス ID を指定します。

オプション	説明
	• この範囲にインターフェイスの範囲を含めるには、たとえば **1/10 - 1/15** など、ハイフンで区切られた最低値と最大値を入力します。 （注） F ポートおよび NP ポートのインターフェイスのプロファイルを設定する際に、重複しない別の範囲をインターフェイスに指定します。
インターフェイス ポリシー グループ	前の手順で設定した F ポート インターフェイス ポリシー グループまたは NP ポート ポリシー グループの名前。 • F ポートとしてこのプロファイルに含まれるインターフェイスを指定するには、F ポート用に設定されているインターフェイス ポリシー グループを選択します。 • NP ポートとしてプロファイルに含まれるインターフェイスを指定するには、NP ポート用に設定されているインターフェイス ポリシー グループを選択します。

ステップ 5 [Submit] をクリックします。前の手順を繰り返しすと、F ポートおよび NP ポートの両方にインターフェイス ポリシーを有することができます。

ステップ 6 FCoE トラフィックにグローバル QoS ポリシーを適用するかどうかを設定します。

さまざまなレベル（1、2、4，5、6）の FCoE トラフィックにさまざまな QoS ポリシーを指定することができます。

a) APIC バー メニューから、**[Fabric]** > **[Access Policies]** > **[Policies]** > **[Global]** > **[QoS Class]** の順にクリックし、[QoS Class] ペインで [Preserve CoS] フラグを有効にします。

b) **[QoS Class - Level 1]**、**[QoS Class - Level 2]**、**[QoS Class - Level 4]**、**[QoS Class - Level 5]**、または **[QoS Class - Level 6]** ダイアログで、次のフィールドを編集して PFC と no-drop CoS を指定します。それから **Submit.** をクリックします。

（注） PFC とノードロップ CoS で設定できるのは 1 レベルだけです。

ポリシー	説明
PFC 管理状態	FCoE トラフィックのこのレベルに優先順位フロー制御を有効にするかどうか（デフォルト値は false です）。 優先順位フロー制御を有効にすると、FCoE トラフィックのこのレベルの [輻輳アルゴリズム] が [ノードロップ] に設定されます。
No-Drop-CoS	FCoE トラフィックの輻輳の場合でも FCoE パケット処理をドロップしない CoS レベル。

ステップ 7 ファイバ チャネル ドメインを定義します。仮想 SAN (VSAN) のセットを作成し、それらを既存の VLAN の設定にマップします。

a) APIC バー メニューで、**[Fabric]** > **[Access Policies]** > **[Physical and External Domains]** > **[Fibre Channel Domains]** の順にクリックします。
b) [Fibre Channel Domains] を右クリックし、[Create Fibre Channel Domain] をクリックします。
c) [Fibre Channel Domain] ダイアログで、次の設定を指定します。

オプション	説明/処理
Name	作成する VSAN ドメインに割り当てる名前またはラベルを指定します。(たとえば vsan-dom2 など)
VSAN Pool	このドメインに割り当てられる VSAN プール。 • 既存の VSAN プールを選択するには、ドロップダウンをクリックしてリストから選択します。変更する場合は、編集アイコンをクリックします。 • VSAN プールを作成するには、**Create a VSAN Pool** をクリックします。 VSAN プールを作成するダイアログで、プロンプトに従って以下を設定します: • FCoE をサポートするには、**静的**リソース割り当て方法が用いられます。 • FCoE F ポート インターフェイスと NP ポート インターフェイスを割り当てる際に利用できる VSAN の範囲です。 （注） 最小値は 1 です。最大値は 4078 です。 必要であれば、複数の範囲の VSAN を設定できます。
VLAN プール	VSAN プールのメンバーがマッピングで使用できる VLAN のプール。 VLAN プールは、このドメインの FCoE 接続をサポートする際に利用する、VLAN の数値範囲を指定します。指定した範囲内の VLAN が、VSAN がマップを行う際に利用できます。 • 既存の VLAN プールを選択するには、ドロップダウンをクリックしてリストから選択します。変更する場合は、編集アイコンをクリックします。 • VLAN プールを作成するには、**Create a VLAN Pool** をクリックします。 VLAN プールを作成するダイアログで、プロンプトに従って以下を設定します: • FCoE をサポートするには、**静的**リソース割り当て方法が用いられます。 • VSAN でマッピングを行う際に利用できる VLAN の範囲です。 （注） 最小値は 1 です。最大値は 4094 です。 必要であれば、複数の範囲の VLAN を設定できます。
VSAN Attr	このドメインの VSAN 属性マップ VSAN 属性は、VSAN プールの VSAN を VLAN プールの VLAN にマップします。

オプション	説明/処理
	• 既存の VSAN 属性マップを選択するには、ドロップダウンをクリックしてリストから選択します。変更する場合は、編集アイコンをクリックします。 • VSAN 属性マップを作成するには、**Create VSAN Attributes** をクリックします。 VSAN 属性を構成するダイアログで、プロンプトに従って以下を設定します: • 適切なロード バランシング オプション (src-dst-ox-id or src-dst-id)。 • 個々 の VSAN から個々の VLAN へのマッピング。たとえば vsan-8 を vlan 10 にマッピングします （注） このドメインのために指定した範囲の VSAN と VLAN だけが、相互にマッピングできます。

ステップ 8 接続済みエンティティ プロファイルを作成し、ファイバ チャネル ドメインをインターフェイス ポリシー グループにバインドします。

a) APIC メニューバー で、**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Policy Groups]** > ***[interface_policy_group_name]*** の順にクリックします。

この手順の *interface_policy_group_name* は、手順 3 で定義したインターフェイス ポリシ - グループです。

b) インターフェイス ポリシー グループのダイアログ ボックスで、[Attached Entity Profile] ドロップダウンをクリックし、既存のアタッチ エンティティ プロファイルを選択するか、**Create Attached Entity Profile** をクリックして、新しいものを作成します。

c) [Attached Entity Profile] ダイアログでは、以下の設定を指定します:

フィールド	説明
名前	この接続済みエンティティ プロファイルの名前
Domains To Be Associated To Interfaces	インターフェイス ポリシー グループに関連付けられるドメインが一覧表示されます。 ここでは、手順 7 で設定したファイバ チャネル ドメインを選択します。 [Submit] をクリックします。

ステップ 9 リーフ プロファイルおよび F ポートと NP ポート インターフェイス プロファイルを関連付けます。

a) APIC メニュー バーから、**[Fabric]** > **[Access Policies]** > **[Switches]** > **[Leaf Switches]** > **[Profiles]** をクリックし、手順 2 で設定したリーフ プロファイルの名前をクリックします。

b) [Create Leaf Profile] ダイアログで、[Associated Interface Selector Profiles] 表を探し、[+] をクリックして新しい表の行を作成し、手順4で作成したFポートインターフェイスプロファイルを選択します。
c) もう一度 **Associated Interface Selector Profiles** テーブルで、+をクリックしてテーブルの新しい行を作成し、手順4で作成したNPポートインターフェイスプロファイルを選択します。
d) [Submit] をクリックします。

次のタスク

ACI ファブリックのインターフェイスに仮想 F ポートおよび NP ポートを正常に展開した後、次の手順でシステム管理者がこれらのインターフェイスを介して EGP アクセスと接続が可能になります。

詳細については、「APIC GUI を使用した vFC ポートへの EPG アクセスの展開 （176 ページ）」を参照してください。

APIC GUI を使用した vFC ポートへの EPG アクセスの展開

ACI ファブリック エンティティを、FCoE トラフィックおよび指定したインターフェイスの F ポートおよび NP ポートをサポートするように設定したら、次の手順はこれらのポートへの EPG アクセスを設定することです。

始める前に

- ACI ファブリックがインストールされていること。
- FC ネットワーク（SAN ストレージなど）に接続しているファイバ チャネル転送（FCF）スイッチは、イーサネットによって ACI リーフ スイッチポートに物理的に接続しています。
- FC ネットワークにアクセスする必要があるホスト アプリケーションは、同じ ACI リーフ スイッチのポートにイーサネットで物理的に接続されていること。
- リーフ ポリシー グループ、リーフ プロファイル、インターフェイス ポリシー グループ、インターフェイス プロファイルとファイバ チャネル ドメインのすべてが、FCoE トラフィックをサポートするように設定されていること。

手順

ステップ 1 適切なテナントの下で、既存のブリッジ ドメインを FCoE をサポートするように設定するか、FCoE をサポートするブリッジ ドメインを作成します。

オプション:	アクション:
FCoE の既存のブリッジドメインを設定するには	1. **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > ***bridge_domain_name*** をクリックします。 2. **タイプ** ブリッジ ドメインのフィールド **プロパティ** パネルにある、クリックして **fc** 。 3. [Submit] をクリックします。
FCoE の新しいブリッジドメインを作成するには	1. **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > **Actions** > **Create a Bridge Domain** をクリックします。 2. **Name** フィールド (**Specify Bridge Domain for the VRF** ダイアログ) で、ブリッジ ドメインの名前を入力します。 3. [Specify Bridge Domain for the VRF] ダイアログの [Type] フィールドで、[fc] をクリックします。 4. [VRF] フィールドで、ドロップダウンから VRF を選択するか、**Create VRF** をクリックし、新しい VRF を作成して設定します。 5. ブリッジ ドメインの設定を終了します。 6. [Submit] をクリックします。

ステップ 2 同じテナントでの下で、既存の EPG を設定するか、新しい EPG を作成して、FCoE が設定されたブリッジ ドメインと関連付けます。

オプション:	アクション:
既存の EPG を関連付ける	1. **[Tenant]** > ***[<テナント名>]*** > **[Application Profiles]** > ***[<アプリケーション プロファイル名>]*** > **[Application EPGs]** > ***[<EPG 名>]*** の順にクリックします。 2. **[QoS class]** フィールドで、このEPGによって生成されたトラフィックに割り当てる Quality of Service（**Level1**、**Level2**、**Level4**、**Level5**、または **Level6**）を選択します。 優先順位フロー制御のドロップ輻輳なしハンドリングでQoS レベルのいずれかを設定する場合、そしてドロップなしパケット優先順位で FCoE トラフィックを処理する必要がある場合には、この EPG にその QoS レベルを割り当てます。 3. **Bridge Domain** フィールド (EPG の **Properties** パネル) で、ドロップダウンリストをクリックして、タイプに合わせて設定したドメインの名前を選択します。ここでは fcoe です。 4. [Submit] をクリックします。

オプション:	アクション:
	（注） [Bridge Domain] フィールドを変更した場合には、変更後 30 ～ 35 秒待機する必要があります。[Bridge Domain] フィールドの変更を急ぎすぎると、NPV スイッチの vFC インターフェイスが障害を起こし、スイッチのリロードが必要になります。
新しい EPG を作成して関連付ける	**1.** **[Tenant]** > ***[<テナント名>]*** > **[Application Profiles]** > ***[<アプリケーション プロファイル名>]*** > **[Application EPGs]** の順にクリックします。 **2.** **Application EPGs** を右クリックし、**Create Application EPG** をクリックします。 **3.** **[QoS class]** フィールドで、このEPGによって生成されたトラフィックに割り当てる Quality of Service（**Level1**、**Level2**、**Level4**、**Level5**、または **Level6**）を選択します。 優先順位フロー制御のドロップ輻輳なしハンドリングで QoS レベルのいずれかを設定する場合、そしてドロップなしパケット優先順位で FCoE トラフィックを処理する必要がある場合には、この EPG にその QoS レベルを割り当てます。 **4.** **Bridge Domain** フィールド (**Specify the EPG Identity** ダイアログ) フィールドで、ドロップダウンリストをクリックして、タイプに合わせて設定したドメインの名前を選択します。ここでは fcoe です。 （注） [Bridge Domain] フィールドを変更した場合には、変更後 30 ～ 35 秒待機する必要があります。[Bridge Domain] フィールドの変更を急ぎすぎると、NPV スイッチの vFC インターフェイスが障害を起こし、スイッチのリロードが必要になります。 **5.** ブリッジ ドメインの設定を終了します。 **6.** **Finish** をクリックします。

ステップ 3 ファイバ チャネル ドメインと EPG の関連付けを追加します。

a) **[Tenant]** > ***[<テナント名>]*** > **[Application Profiles]** > ***[<アプリケーション プロファイル名>]*** > **[Application EPGs]** > ***[<EPG 名>]*** > **[Domains（VMs and Bare Metal）]** の順にクリックします。

b) [Domains（VMs and Bare Metal）] を右クリックし、[Add Fibre Channel Domain Association] をクリックします。

c) [Add Fibre Channel Domain Association] ダイアログで、[Fibre Channel Domain Profile] フィールドを探します。

d) ドロップダウンリスト をクリックし、以前に設定したファイバ チャネル ドメインの名前を選択します。

e) [Submit] をクリックします。

ステップ 4 関連する EPG の下で、ファイバ チャネルのパスを定義します。

ファイバ チャネルのパスでは、FCoE F ポートまたは NP ポートとして有効にされたインターフェイスを指定して、選択した EPG に関連付けます。

a) **[Tenant]** > ***[<テナント名>]*** > **[Application Profiles]** > ***[<アプリケーション プロファイル名>]*** > **[Application EPGs]** > ***[<EPG 名>]*** > **[Fibre Channel（Paths）]** の順にクリックします。
b) [Fibre Channel（Paths）] を右クリックし、[Deploy Fibre Channel] をクリックします。
c) [Deploy Fibre Channel] ダイアログで、次の設定を行います。

オプション:	アクション:
Path Type	FCoE トラフィックを送受信するためにアクセスされるインターフェイスのタイプです (ポート、ダイレクト ポート チャネル、または仮想ポート チャネル)。
Path	選択した EPG に関連付けられている FCoE トラフィックが流れるノードインターフェイスのパスです。 ドロップダウン リストをクリックして、リスト表示されたインターフェイスの中から選択します。。 （注） 以前に F ポートまたは NP ポートとして設定されているインターフェイスのみを選択します。設定されていないインターフェイスを選択すると、これらのインターフェイスにはデフォルト値だけが適用されます。 （注） FCoE over FEX を展開するには、以前に設定した FEX ポートを選択します。
VSAN	**Path** フィールドで選択したインターフェイスを使用する VSAN です。 （注） 指定する VSAN は、VSAN プールとして指定した VSAN の範囲になければなりません。 ほとんどの場合、この EPG がアクセスするために設定されているすべてのインターフェイスは、同じ VSAN に割り当てられている必要があります。ただし、仮想ポートチャネル (VPC) 接続上にファイバ チャネル パスを指定する場合を除きます。その場合には、2 つの VSAN を指定し、接続のレッグごとに 1 つを使用します。
VSAN Mode	選択した VSAN が選択したインターフェイスにアクセスするモードです (**Native** または **Regular**)。 FCoE サポート用に設定された各インターフェイスでは、ネイティブ モードに設定された VSAN が 1 つだけ必要です。同じインターフェイスに割り当てられる追加の VSAN は、通常モードでアクセスする必要があります。
Pinning label	(オプション) このオプションは、アクセスを F ポートへマッピングする場合にのみ適用されます。そしてこの F ポートは、特定のアップリンク NP ポートにバインドする必要があります。これは、ピニング ラベル (ピニング ラベル 1 またはピニング ラベル 2) を特定の NP ポートに関連付けます。それから、ピニング ラベ

オプション:	アクション:
	ルをターゲットFポートに割り当てます。この関連づけを行うと、関連付けられたNPポートは、すべての場合に、ターゲットFポートへのアップリンク ポートとしての役割を果たします。 ピニング ラベルを選択し、それをNPポートとして設定されたインターフェイスに関連付けます。 このオプションは、「トラフィック-マッピング」とも呼ばれるものを実装します。 （注） Fポートと、関連付けられているピニング ラベルのNPポートは、同一のリーフ スイッチ上に存在する必要があります。

ステップ 5 [Submit] をクリックします。

ステップ 6 EPG アクセスをマッピングする、FCoE 対応のインターフェイスごとに、手順 4 と 5 を繰り返します。

ステップ 7 正常に導入できたかどうかは、次のように確認します。

a) **Fabric** > **Inventory** > ***Pod_name*** > ***leaf_name*** > **Interfaces** > **VFC interfaces** をクリックします。

ポートを展開したインターフェイスが、VFC インターフェイス下にリスト表示されます。

次のタスク

vFC インターフェイスへの EPG アクセスをセットアップした後の最後の手順は、FCoE 初期化プロトコル (FIP) をサポートするネットワークをセットアップすることです。これによって、それらのインターフェイスの検出が有効になります。

詳細については、「FCoE Initiation Protocol をサポートする EPG の導入 （180 ページ）」を参照してください。

FCoE Initiation Protocol をサポートする EPG の導入

FCoE EPG からサーバのポートへのアクセスを設定した後も、FCoE Initiation Protocol (FIP) をサポートするように EPG のアクセスを設定する必要があります。

始める前に

- ACI ファブリックがインストールされています。
- FC ネットワークにアクセスする必要があるホスト アプリケーションは、同じ ACI Leaf スイッチのポートにイーサネットで物理的に接続されます。

- リーフ ポリシー グループ、リーフ プロファイル、インターフェイス ポリシー グループ、インターフェイスのプロファイルとファイバ チャネル ドメインはすべて、APIC GUI を使用した vFC ポートへの EPG アクセスの展開 （176 ページ）のトピックで説明されているように、FCoE トラフィックをサポートするように設定されています。
- EPG から vFC ポートへのアクセスは、「APIC GUI を使用した vFC ポートへの EPG アクセスの展開 （176 ページ）」のトピックで説明しているように、有効になっています。

手順

ステップ 1 同じテナントの下で、FIP をサポートするように既存のブリッジ ドメインを設定するか、FIP をサポートする通常のブリッジ ドメインを作成します。

オプション:	アクション:
FCoE の既存のブリッジ ドメインを設定するには	1. **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > ***bridge_domain_name*** をクリックします。 2. **Type** フィールド (ブリッジ ドメインの **Properties** パネル) で、**Regular** をクリックします。 3. [Submit] をクリックします。
FCoE の新しいブリッジ ドメインを作成するには	1. **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > **Actions** > **Create a Bridge Domain** をクリックします。 2. **Name** フィールド (**Specify Bridge Domain for the VRF** ダイアログ) で、ブリッジ ドメインの名前を入力します。 3. [Specify Bridge Domain for the VRF] ダイアログの [Type] フィールドで、[Regular] をクリックします。 4. [VRF] フィールドで、ドロップダウンから VRF を選択するか、**Create VRF** をクリックし、新しい VRF を作成して設定します。 5. ブリッジ ドメインの設定を終了します。 6. [Submit] をクリックします。

ステップ 2 同じテナントで、既存の EPG を設定するか、または通常型のブリッジ ドメインと関連付ける新しい EPG を作成します。

オプション:	アクション:
既存の EPG を関連付ける	1. **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs** > ***epg_name*** をクリックします。

オプション:	アクション:
	2. **Bridge Domain** フィールド (EPG の **Properties** パネル) で、ドロップダウンリストをクリックして、先ほど FIP をサポートするように設定した通常型のブリッジ ドメインの名前を入力します。 3. [Submit] をクリックします。
新しい EPG を作成して関連付けるには、	1. **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs** をクリックします。 2. **Application EPGs** を右クリックし、**Create Application EPG** をクリックします。 3. **Bridge Domain** フィールド (**Specify the EPG Identity** ダイアログ) で、ドロップダウンリストをクリックして、先ほど FIP をサポートするように設定した通常型のブリッジ ドメインの名前を選択します。 4. ブリッジ ドメインの設定を終了します。 5. **Finish** をクリックします。

ステップ 3 EPG と物理ドメインの関連付けを追加します。

a) **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs** > ***epg_name*** > **Domains & Bare Metal** をクリックします。

b) **Domains & Bare Metal** を右クリックし、**Add Physical Domain Association** をクリックします。

c) **Add Physical Domain Association** ダイアログの [Physical Domain Profile Field] を操作します。

d) ドロップダウンリストをクリックし、FIP のサポートで使用する LAN を含む物理ドメインの名前を選択します。

e) [Submit] をクリックします。

ステップ 4 関連する EPG でパスを定義します。

FCoE F ポートまたは NP ポートとして有効にされ、選択した EPG に関連付けられるインターフェイスを指定します。

a) **[Tenant]** > ***[<テナント名>]*** > **[Application Profiles]** > **[ap1]** > **[Application EPGs]** > ***[<EPG 名>]*** > **[Static Ports]** の順にクリックします。

b) [Static Ports] を右クリックし、[Deploy Static EPG on PC, VPC, or Interface] をクリックします。

c) **Path Type** フィールドで、F モード vFC を展開するポート タイプ (ポート、直接ポート チャネル、または仮想ポート チャネル) を指定します。

d) **Path** フィールドで、F ポートを展開するすべてのパスを指定します。

e) FCoE VLAN ディスカバリとして、およびポート モードとして 802.1p (アクセス) のために使用する [VLAN Encap] を選択します。

f) [Submit] をクリックします。

FCoE コンポーネントは、FCoE ネットワークの動作を開始するために、ディスカバリ プロセスを開始します。

APIC GUI を使用した FCoE 接続のアンデプロイ

ACI ファブリック上のリーフ スイッチ インターフェイスの FCoE イネーブルメントを取り消すには、APIC GUI を使用した FCoE vFC ポートの展開（168 ページ）で定義したファイバ チャネル パスとファイバ チャネル ドメインとその要素を削除します。

（注） クリーンアップ中に vFC ポートのイーサネット設定オブジェクト (infraHPortS) を削除した場合 (たとえば、GUI の **Leaf Interface Profiles** ページの **Interface Selector** テーブル)、デフォルトの vFC プロパティはそのインターフェイスに関連付けられたままになります。たとえば、vFC NP ポート 1/20 のインターフェイス設定が削除され、そのポートは vFC ポートのままですが、デフォルト以外の NP ポート設定が適用されるのではなく、デフォルトの F ポート設定が使用されます。

始める前に

FCoE の展開中に指定した関連する VSAN プール、VLAN プール、および VSAN 属性マップを含む、ファイバ チャネル パスとファイバ チャネル ドメインの名前を知っている必要があります。

手順

ステップ 1 関連するファイバチャネルパスを削除して、この配置でパスが指定されたポート/vsan から vFC をアンデプロイします。

この操作では、この展開でパスが指定されたポート/vsan から vFC 展開が削除されます。

a) **[Tenant]** > ***[<テナント名>]*** > **[Application Profiles]** > ***[<アプリケーション プロファイル名>]*** > **[Application EPGs]** > ***[<アプリケーション EPG 名>]*** > **[Fibre Channel（Paths）]** の順にクリックします。次に、ターゲットのファイバチャネルパスの名前を右クリックし、**[Delete]** を選択します。

b) [Yes] をクリックして削除を確定します。

ステップ 2 ファイバ チャネル ドメインを定義したときに設定した VLAN 対 VSAN マップを削除します。

この操作は、マップに定義されているすべての要素から vFC の展開を削除します。

a) **[Fabric]** > **[Access Policies]** > **[Pools]** > **[VSAN Attributes]** をクリックします。次に、ターゲット マップの名前を右クリックし、**[Delete]** を選択します。

b) **[Yes]** をクリックして削除を確定します。

ステップ 3 ファイバ チャネル ドメインを定義したときに定義した VLAN プールと VSAN プールを削除します。

これにより、ACI ファブリックからのすべての vFC 展開が不要になります。

a) **[Fabric]** > **[Access Policies]** > **[Pools]** > **[VSAN]** をクリックし、ターゲットVSANプール名を右クリックして、**[Delete]** を選択します。
b) [Yes] をクリックして削除を確定します。
c) **[Fabric]** > **[Access Policies]** > **[Pools]** > **[VLAN]** をクリックし、ターゲット VLAN プール名を右クリックして、**[Delete]** を選択します。
d) [Yes] をクリックして削除を確定します。

ステップ 4 削除したばかりのVSANプール、VLANプール、およびマップエレメントを含むファイバチャネル ドメインを削除します。

a) **[Tenants]** > **[<テナント名>]** > **[Application Profiles]** > **[Fibre Channel Domains]** をクリックします。次に、ターゲットのファイバ チャネル ドメインの名前を右クリックし、**[Delete]** を選択します。
b) **[Yes]** をクリックして削除を確定します。

ステップ 5 テナント/EPG/App とセレクタは、必要がない場合は削除できます。

オプション	Action
関連するアプリケーションEPGを削除するが、関連するテナントとアプリケーションプロファイルを保存する場合は、次のようにします。	**[Tenants]** > ***[tenant_name]*** > **[Application Profiles]** > ***[app_profile_name]*** > **[Application EPGs]** をクリックし、ターゲット アプリケーション EPG の名前を右クリックして **[Delete]** を選択し、**[Yes]** をクリックして削除を確認します。
関連するアプリケーションプロファイルを削除するが関連するテナントを保存する場合は、次のようにします。	**[Tenants]** > ***[tenant_name]*** > **[Application Profiles]** をクリックし、ターゲット アプリケーション プロファイルの名前を右クリックし、**[Delete]** を選択してから **[Yes]** をクリックして削除を確認します。
関連するテナントを削除する場合:	**[Tenants]** > をクリックし、ターゲット テナントの名前を右クリックして**[Delete]**を選択し、**[Yes]** をクリックして削除を確認します。

NX-OS スタイルの CLI を使用した FCoE の設定

FCoE NX-OS スタイル CLI 設定

NX-OS スタイル CLI を使用したポリシーまたはプロファイルのない FCoE 接続の設定

次の例の NX-OS スタイル CLI シーケンス EPG の FCoE 接続を設定する **e1** テナントで **t1** 設定またはスイッチ レベルとインターフェイス レベル ポリシーとプロファイルを適用せず。

手順

	コマンドまたはアクション	目的

ステップ 1	ターゲット テナントの下には、FCoE トラフィックをサポートするブリッジ ドメインを設定します。
	例：
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# vrf context v1`
	`apic1(config-tenant-vrf)# exit`
	`apic1(config-tenant)# bridge-domain b1`
	`apic1(config-tenant-bd)# fc`
	`apic1(config-tenant-bd)# vrf member v1`
	`apic1(config-tenant-bd)# exit`
	`apic1(config-tenant)# exit`
		サンプル コマンド シーケンスはブリッジ ドメインを作成 **b1** テナントで **t1** FCoE 接続をサポートするように設定します。

ステップ 2	同じのテナントの下には、FCoE に設定されたブリッジ ドメインとターゲット EPG を関連付けます。
	例：
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# application a1`
	`apic1(config-tenant-app)# epg e1`
	`apic1(config-tenant-app-epg)#`
	`bridge-domain member b1`
	`apic1(config-tenant-app-epg)# exit`
	`apic1(config-tenant-app)# exit`
	`apic1(config-tenant)# exit`
		サンプル コマンド シーケンス作成 EPG **e1** し、FCoE に設定されたブリッジ ドメインにその EPG を関連付けます **b1**
		。

ステップ 3	VLAN マッピングに VSAN ドメイン、VSAN プール、VLAN プール、VSAN を作成します。
	例：
	A
	`apic1(config)# vsan-domain dom1`
	`apic1(config-vsan)# vsan 1-10`
	`apic1(config-vsan)# vlan 1-10`
		例 A 、サンプル コマンド シーケンスは、VSAN ドメインを作成 **dom1** VSAN プールと VLAN プール、VSAN 1 を VLAN 1 にマッピングされ、VLAN 2 に VSAN 2 をマップ
		例 B 、代替サンプル コマンド シーケンスは再利用可能な VSAN 属性テンプレートを作成 **pol1** VSAN ドメインを作成

	コマンドまたはアクション	目的
	apic1(config-vsan)# fcoe vsan 1 vlan 1 loadbalancing src-dst-ox-id apic1(config-vsan)# fcoe vsan 2 vlan 2 例： **B** apic1(config)# template vsan-attribute pol1 apic1(config-vsan-attr)# fcoe vsan 2 vlan 12 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# fcoe vsan 3 vlan 13 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# exit apic1(config)# vsan-domain dom1 apic1(config-vsan)# vsan 1-10 apic1(config-vsan)# vlan 1-10 apic1(config-vsan)# inherit vsan-attribute pol1 apic1(config-vsan)# exit	し、 **dom1** 、そのテンプレートから属性とマッピングを継承します。
ステップ 4	FCoE Initialization (FIP) プロセスをサポートする物理ドメインを作成します。 例： apic1(config)# vlan-domain fipVlanDom apic1(config-vlan)# vlan 120 apic1(config-vlan)# exit	例では、コマンド シーケンスは、通常の VLAN ドメインを作成 **fipVlanDom** 、VLAN を含む **120** FIP プロセスをサポートします。
ステップ 5	ターゲット テナントの下には、定期的なブリッジ ドメインを設定します。 例： apic1(config)# tenant t1 apic1(config-tenant)# vrf context v2 apic1(config-tenant-vrf)# exit apic1(config-tenant)# bridge-domain fip-bd apic1(config-tenant-bd)# vrf member v2 apic1(config-tenant-bd)# exit apic1(config-tenant)# exit	コマンド シーケンスがブリッジ ドメインを作成例では、 **fip bd** 。
ステップ 6	同じのテナントの下には、設定されている定期的なブリッジドメインでこのEPGを関連付けます。 例： apic1(config)# tenant t1 apic1(config-tenant)# application a1 apic1(config-tenant-app)# epg epg-fip apic1(config-tenant-app-epg)# bridge-domain member fip-bd apic1(config-tenant-app-epg)# exit apic1(config-tenant-app)# exit apic1(config-tenant)# exit	例では、コマンド シーケンス関連付けます EPG **epg fip** ブリッジ ドメインを **fip bd** 。

	コマンドまたはアクション	目的

ステップ 7	VFC インターフェイスを F モードで設定します。
	例：
	A
	`apic1(config)# leaf 101`
	`apic1(config-leaf)# interface ethernet 1/2`
	`apic1(config-leaf-if)# vlan-domain member fipVlanDom`
	`apic1(config-leaf-if)# switchport trunk native vlan 120 tenant t1 application a1 epg epg-fip`
	`apic1(config-leaf-if)# exit`
	
	`apic1(config-leaf)# exit`
	`apic1(config-leaf)# interface vfc 1/2`
	`apic1(config-leaf-if)# switchport mode f`
	`apic1(config-leaf-if)# vsan-domain member dom1`
	`apic1(config-leaf-if)# switchport vsan 2 tenant t1 application a1 epg e1`
	`apic1(config-leaf-if)# switchport trunk allowed vsan 3 tenant t1 application a1 epg e2`
	`apic1(config-leaf-if)# exit`
	例：
	B
	`apic1(config)# vpc context leaf 101 102`
	`apic1(config-vpc)# interface vpc vpc1`
	`apic1(config-vpc-if)# vlan-domain member vfdom100`
	`apic1(config-vpc-if)# vsan-domain member dom1`
	`apic1(config-vpc-if)# #For FIP discovery`
	`apic1(config-vpc-if)# switchport trunk native vlan 120 tenant t1 application a1 epg epg-fip`
	`apic1(config-vpc-if)# switchport vsan 2 tenant t1 application a1 epg e1`
	`apic1(config-vpc-if)# exit`
	`apic1(config-vpc)# exit`
	`apic1(config)# leaf 101-102`
	`apic1(config-leaf)# interface ethernet 1/3`
	`apic1(config-leaf-if)# channel-group vpc1 vpc`
	`apic1(config-leaf-if)# exit`
	`apic1(config-leaf)# exit`
	例：
	C
		例では **A** コマンド シーケンスは、インターフェイスを有効に **1/2** リーフ スイッチで **101** として機能する、 **F** ポートおよびインターフェイスの VSAN のドメインに関連 **dom1** 。
		ネイティブ モードで 1 つ (と 1 つだけ) の VSAN 対象のインターフェイスの各割り当てる必要があります。各インターフェイスには、通常モードで1つ以上の追加 Vsan を割り当てることができます。
		サンプル コマンド シーケンスは、対象のインターフェイスを関連付けます **1/2** と。
		• VLAN **120** FIP ディスカバリの EPG に関連付けます **epg fip** およびアプリケーション **a1** テナントで **t1** 。
		• VSAN **2** ネイティブ VSAN として、EPG に関連付けます **e1** およびアプリケーション **a1** テナントで **t1** 。
		• VSAN **3** 定期的な VSAN として。
		例 **B** では、コマンド シーケンスは、両方のレッグに同じ VSAN を持つ vPC を介して vFC を設定します。CLI からログごとに異なる Vsan を指定することはできません。代替設定は、GUI を高度な apic 内で実行できます。

	コマンドまたはアクション	目的
	apic1(config)# leaf 101 apic1(config-leaf)# interface vfc-po pc1 apic1(config-leaf-if)# vsan-domain member dom1 apic1(config-leaf-if)# switchport vsan 2 tenant t1 application a1 epg e1 apic1(config-leaf-if)# exit apic1(config-leaf)# interface ethernet 1/2 apic1(config-leaf-if)# channel-group pc1 apic1(config-leaf-if)# exit apic1(config-leaf)# exit	
ステップ 8	VFCインターフェイスをNPモードで設定します。 例： apic1(config)# leaf 101 apic1(config-leaf)# interface vfc 1/4 apic1(config-leaf-if)# switchport mode np apic1(config-leaf-if)# vsan-domain member dom1	サンプル コマンド シーケンスは、インターフェイスを有効に **1/4** リーフ スイッチで **101** として機能する、 **NP** ポートおよびインターフェイスのVSANのドメインに関連 **dom1** 。
ステップ 9	VSANを対象となるFCoE対応インターフェイスに割り当てます。 例： apic1(config-leaf-if)# switchport trunk allowed vsan 1 tenant t1 application a1 epg e1 apic1(config-leaf-if)# switchport vsan 2 tenant t4 application a4 epg e4	ネイティブ モードで 1 つ (と 1 つだけ) の VSAN 対象のインターフェイスの各割り当てる必要があります。各インターフェイスには、通常モードで1つ以上の追加 Vsan を割り当てることができます。 サンプル コマンド シーケンスは、ターゲット インターフェイスを VSAN **1** に割り当て、それを EPG **e1** とアプリケーション **a1** にテナント **t1** の下で関連付けます。「trunk allowed」は、VSAN 1 に通常モードのステータスを割り当てます。コマンド シーケンスも割り当てます、インターフェイス、必要な **ネイティブ モード** VSAN **2** 。次の例に示すは、同一のインターフェイスを異なるテナント アクセスで実行されているさまざまな Epg を提供するためにさまざまな Vsan の動作を渡します。

NX-OSスタイルCLIを使用したポリシーまたはプロファイルがあるFCoE接続の設定

次の例 NX-OS スタイル CLI のシーケンスを作成し、EPG の FCoE 接続を設定するポリシーを使用して **e1** テナントで **t1** 。

手順

	コマンドまたはアクション	目的

ステップ 1	ターゲット テナントの下には、FCoE トラフィックをサポートするブリッジ ドメインを設定します。
	例：
	`apic1# configure`
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# vrf context v1`
	`apic1(config-tenant-vrf)# exit`
	`apic1(config-tenant)# bridge-domain b1`
	`apic1(config-tenant-bd)# fc`
	`apic1(config-tenant-bd)# vrf member v1`
	`apic1(config-tenant-bd)# exit`
	`apic1(config-tenant)# exit`
	`apic1(config)#`
		サンプル コマンド シーケンスはブリッジ ドメインを作成 **b1** テナントで **t1** FCoE接続をサポートするように設定します。

ステップ 2	同じのテナントの下には、設定されている FCoE ブリッジ ドメインと、ターゲット EPG を関連付けます。
	例：
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# application a1`
	`apic1(config-tenant-app)# epg e1`
	`apic1(config-tenant-app-epg)# bridge-domain member b1`
	`apic1(config-tenant-app-epg)# exit`
	`apic1(config-tenant-app)# exit`
	`apic1(config-tenant)# exit`
	`apic1(config)#`
		サンプル コマンド シーケンス作成EPG **e1** その EPG の FCoE に設定されたブリッジ ドメイン関連付け **b1** 。

ステップ 3	VLAN マッピングに VSAN ドメイン、VSAN プール、VLAN プール、VSAN を作成します。
	例：
	A
	`apic1(config)# vsan-domain dom1`
	`apic1(config-vsan)# vsan 1-10`
	`apic1(config-vsan)# vlan 1-10`
	`apic1(config-vsan)# fcoe vsan 1 vlan`
		例 A 、サンプル コマンド シーケンスは、VSAN ドメインを作成 **dom1** VSAN プールと VLAN プール、マップ VSAN 1 VLAN 1 と VLAN 2 に VSAN 2 をマップ
		例 B 、代替サンプル コマンド シーケンスは再利用可能な vsan 属性テンプレートを作成 **pol1** VSAN ドメインを

	コマンドまたはアクション	目的
	1 loadbalancing src-dst-ox-id apic1(config-vsan)# fcoe vsan 2 vlan 2 **例：** **B** apic1(config)# template vsan-attribute pol1 apic1(config-vsan-attr)# fcoe vsan 2 vlan 12 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# fcoe vsan 3 vlan 13 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# exit apic1(config)# vsan-domain dom1 apic1(config-vsan)# inherit vsan-attribute pol1 apic1(config-vsan)# exit	作成し、 **dom1** 、そのテンプレートから属性とマッピングを継承します。
ステップ 4	FCoE Initialization (FIP) プロセスをサポートする物理ドメインを作成します。 **例：** apic1(config)# vlan-domain fipVlanDom apic1(config)# vlan-pool fipVlanPool	
ステップ 5	ファイバ チャネル SAN ポリシーを設定します。 **例：** apic1# apic1# configure apic1(config)# template fc-fabric-policy ffp1 apic1(config-fc-fabric-policy)# fctimer e-d-tov 1111 apic1(config-fc-fabric-policy)# fctimer r-a-tov 2222 apic1(config-fc-fabric-policy)# fcoe fcmap 0E:FC:01 apic1(config-fc-fabric-policy)# exit	サンプル コマンド シーケンスは、SAN のファイバ チャネル ポリシーを作成 **ffp1** の組み合わせを指定するエラー検出タイムアウト値 (EDTOV)、resource allocation(リソース割り当て、リソースの割り当て) タイムアウト値 (RATOV)、およびターゲット リーフ上の FCoE 対応のインターフェイスのデフォルト FC マップ値スイッチです。
ステップ 6	ファイバ チャネル ノード ポリシーを作成します。 **例：** apic1(config)# template fc-leaf-policy flp1 apic1(config-fc-leaf-policy)# fcoe fka-adv-period 44 apic1(config-fc-leaf-policy)# exit	サンプル コマンド シーケンスは、ファイバ チャネル ノードのポリシーを作成 **flp1** を中断のロード バランシングの有効化と FIP キープア ライブ値の組み合わせを指定します。これらの値は、ターゲット リーフ スイッチ上のすべて

	コマンドまたはアクション	目的
		の FCoE 対応インターフェイスにも適用されます。

ステップ 7	ノード ポリシー グループを作成します。
	例：
	`apic1(config)# template leaf-policy-group lpg1`
	`apic1(config-leaf-policy-group)# inherit fc-fabric-policy ffp1`
	`apic1(config-leaf-policy-group)# inherit fc-leaf-policy flp1`
	`apic1(config-leaf-policy-group)# exit`
	`apic1(config)# exit`
	`apic1#`
		サンプル コマンド シーケンスはノード ポリシー グループを作成 **lpg1** 、SAN のファイバ チャネル ポリシーの値を結合する **ffp1** とファイバ チャネル ノードのポリシー、 **flp1** 。このノード ポリシー グループの合計値は、後で設定されているノードのプロファイルに適用できます。

ステップ 8	ノード プロファイルを作成します。
	例：
	`apic1(config)# leaf-profile lp1`
	`apic1(config-leaf-profile)# leaf-group lg1`
	`apic1(config-leaf-group)# leaf 101`
	`apic1(config-leaf-group)# leaf-policy-group lpg1`
		サンプル コマンド シーケンスがノードのプロファイルを作成 **lp1** ノード ポリシー グループと関連付けます **lpg1** 、ノード グループ **lg1** 、およびリーフ スイッチ **101** 。

ステップ 9	F ポート インターフェイスのインターフェイス ポリシー グループを作成します。
	例：
	`apic1(config)# template policy-group ipg1`
	`apic1(config-pol-grp-if)# priority-flow-control mode auto`
	`apic1(config-pol-grp-if)# switchport mode f`
	`apic1(config-pol-grp-if)# slow-drain pause timeout 111`
	`apic1(config-pol-grp-if)# slow-drain congestion-timeout count 55`
	`apic1(config-pol-grp-if)# slow-drain congestion-timeout action log`
		サンプル コマンド シーケンスは、インターフェイス グループのポリシーを作成 **ipg1** し、プライオリティ フロー制御の有効化、F ポートの有効化、およびこのポリシー グループに適用されているすべてのインターフェイスに対して低速ドレイン ポリシーの値を決定する値の組み合わせを割り当てます。

ステップ 10	NP ポート インターフェイスのインターフェイス ポリシー グループを作成します。
	例：
	`apic1(config)# template policy-group ipg2`
	`apic1(config-pol-grp-if)# priority-flow-control mode auto`
	`apic1(config-pol-grp-if)# switchport mode np`
		サンプル コマンド シーケンスは、インターフェイス グループ ポリシー **ipg2** を作成し、このポリシー グループに適用されているすべてのインターフェイスに対して、優先順位フロー制御の有効化、NP ポートの有効化、低速ドレイン ポリシーの値を決定する値の組み合わせを割り当てます。

	コマンドまたはアクション	目的
	apic1(config-pol-grp-if)# slow-drain pause timeout 111 apic1(config-pol-grp-if)# slow-drain congestion-timeout count 55 apic1(config-pol-grp-if)# slow-drain congestion-timeout action log	
ステップ 11	F ポート インターフェイスのインターフェイス プロファイルを作成します。 **例**： apic1# configure apic1(config)# leaf-interface-profile lip1 apic1(config-leaf-if-profile)# description 'test description lip1' apic1(config-leaf-if-profile)# leaf-interface-group lig1 apic1(config-leaf-if-group)# description 'test description lig1' apic1(config-leaf-if-group)# policy-group ipg1 apic1(config-leaf-if-group)# interface ethernet 1/2-6, 1/9-13	サンプル コマンド シーケンスは、インターフェイス プロファイルを作成 **lip1** F ポートのインターフェイスの F ポートの特定のインターフェイス ポリシー グループ プロファイルを関連付けます **ipg1** 、このインターフェイスを指定しプロファイルとその関連するポリシー。適用されます。
ステップ 12	NP ポート インターフェイスのインターフェイス プロファイルを作成します。 **例**： apic1# configure apic1(config)# leaf-interface-profile lip2 apic1(config-leaf-if-profile)# description 'test description lip2' apic1(config-leaf-if-profile)# leaf-interface-group lig2 apic1(config-leaf-if-group)# description 'test description lig2' apic1(config-leaf-if-group)# policy-group ipg2 apic1(config-leaf-if-group)# interface ethernet 1/14	サンプル コマンド シーケンスは、インターフェイス プロファイルを作成 **lip2** NP ポート インターフェイス、NP ポートの特定のインターフェイス ポリシー グループ プロファイルに関連付けます **ipg2** 、このインターフェイスを指定し、プロファイルとその関連するポリシー適用されます。
ステップ 13	レベル 1 の QoS クラス ポリシーを設定します。 **例**： apic1(config)# qos parameters level1 apic1(config-qos)# pause no-drop cos 3	サンプル コマンド シーケンスは、FCoE トラフィック プライオリティ フロー制御ポリシーを適用することがおよび非ドロップ パケットのクラスのサービス レベル 3 の処理を一時停止の QoS レベルを指定します。

NX-OS スタイル CLI を使用して FCoE オーバー FEX の設定

FEX ポートは、ポート Vsan として設定されます。

手順

ステップ 1 テナントと VSAN のドメインを設定します。

例：

```
apic1# configure
apic1(config)#  tenant t1
apic1(config-tenant)#   vrf context v1
apic1(config-tenant-vrf)#     exit
apic1(config-tenant)#   bridge-domain b1
apic1(config-tenant-bd)#       fc
apic1(config-tenant-bd)#       vrf member v1
apic1(config-tenant-bd)#       exit
apic1(config-tenant)#   application a1
apic1(config-tenant-app)#      epg e1
apic1(config-tenant-app-epg)# bridge-domain member b1
apic1(config-tenant-app-epg)# exit
apic1(config-tenant-app)#      exit
apic1(config-tenant)# exit

apic1(config)#  vsan-domain dom1
apic1(config-vsan)# vlan 1-100
apic1(config-vsan)# vsan 1-100
apic1(config-vsan)# fcoe vsan 2 vlan 2 loadbalancing src-dst-ox-id
apic1(config-vsan)# fcoe vsan 3 vlan 3 loadbalancing src-dst-ox-id
apic1(config-vsan)# fcoe vsan 5 vlan 5
apic1(config-vsan)# exit
```

ステップ 2 FEX をインターフェイスに関連付けます。

例：

```
apic1(config)#  leaf 101
apic1(config-leaf)#  interface ethernet 1/12
apic1(config-leaf-if)#  fex associate 111
apic1(config-leaf-if)#  exit
```

ステップ 3 ポート、ポート チャネル、および VPC あたり FEX を介して FCoE を設定します。

例：

```
apic1(config-leaf)#  interface vfc 111/1/2
apic1(config-leaf-if)#  vsan-domain member dom1
apic1(config-leaf-if)#  switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-leaf-if)# exit

apic1(config-leaf)#  interface vfc-po pc1 fex 111
apic1(config-leaf-if)# vsan-domain member dom1
apic1(config-leaf-if)# switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 111/1/3
apic1(config-leaf-if)# channel-group pc1
apic1(config-leaf-if# exit
apic1(config-leaf)# exit
```

```
apic1(config)# vpc domain explicit 12 leaf 101 102
apic1(config-vpc)# exit
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)#     interface vpc vpc1 fex 111 111
apic1(config-vpc-if)#       vsan-domain member dom1
apic1(config-vpc-if)#       switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-vpc-if)#       exit
apic1(config-vpc)#     exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# fex associate 111
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 111/1/2
apic1(config-leaf-if)# channel-group vpc1 vpc
apic1(config-leaf-if)# exit
```

ステップ 4 設定を確認するには、次のコマンドを実行します。

例：

```
apic1(config-vpc)# show vsan-domain detail
vsan-domain : dom1

vsan : 1-100

vlan : 1-100

Leaf          Interface          Vsan  Vlan  Vsan-Mode    Port-Mode  Usage
 Operational State
------------  -----------------  ----  ----  -----------  ---------
---------------------------------------
101           vfc111/1/2       2       2    Native                  Tenant: t1
  Deployed
                                                                     App: a1
                                                                     Epg: e1

101           PC:pc1           5       5    Native                Tenant: t1
  Deployed
                                                                     App: a1
                                                                     Epg: e1

101           vfc111/1/3       3       3    Native       F          Tenant: t1
  Deployed
                                                                     App: a1
                                                                     Epg: e1
```

NX-OS スタイルの CLI を使用した FCoE 設定の検証

次 **show** コマンドは、リーフ スイッチ ポートで FCoE の設定を確認します。

手順

使用して、 **vsan ドメインを表示** コマンドをターゲット スイッチで FCoE が有効になっていることを確認します。

コマンドの例では、FCoEがリストされているリーフ スイッチおよび接続の詳細をFCF で有効になっていることを確認します。

例：

```
ifav-isim8-ifc1# show vsan-domain detail
vsan-domain : iPostfcoeDomP1

vsan : 1-20  51-52  100-102  104-110  200  1999  3100-3101  3133
      2000

vlan : 1-20  51-52  100-102  104-110  200  1999  3100-3101  3133
      2000

                                      Vsan    Port                    Operational
 Leaf  Interface       Vsan  Vlan     Mode    Mode  Usage             State
 ----  ---------       ----  ----  -------- ----  ----------------  ------------
 101   vfc1/11         1     1     Regular  F     Tenant: iPost101  Deployed
                                                  App: iPost1
                                                  Epg: iPost1


 101   vfc1/12         1     1     Regular  NP    Tenant: iPost101  Deployed
                                                  App: iPost1
                                                  Epg: iPost1


 101   PC:infraAccBndl 4     4     Regular  NP    Tenant: iPost101  Deployed
          Grp_pc01                                App: iPost4
                                                  Epg: iPost4


 101   vfc1/30         2000        Native        Tenant: t1        Not deployed
                                                  App: a1           (invalid-path)
                                                  Epg: e1
```

NX-OS スタイル CLI を使用した FCoE 要素の展開解除

ACI ファブリックから FCoE 接続を導入解除に移動してもでは、いくつかのレベルで FCoE コンポーネントを削除することが必要です。

手順

ステップ 1 リーフ ポート インターフェイスの属性のリスト、そのモードの設定をデフォルトに設定し、その EPG の導入とドメインの関連付けを削除します。

インターフェイス vfc のポート モードの設定を設定する例 **1/2** のデフォルトに [EPG の導入を削除 **e1** と VSAN ドメインに関連付け **dom1** そのインターフェイスから。

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# interface vfc 1/2
apic1(config-leaf-if)# show run
# Command: show running-config leaf 101 interface vfc 1 / 2
# Time: Tue Jul 26 09:41:11 2016
  leaf 101
    interface vfc 1/2
      vsan-domain member dom1
      switchport vsan 2 tenant t1 application a1 epg e1
      exit
    exit
apic1(config-leaf-if)# no switchport mode
apic1(config-leaf-if)# no switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-leaf-if)# no vsan-domain member dom1
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

ステップ 2 VSAN/VLAN マッピング、および VLAN と VSAN のプールを一覧表示して削除します。

この例では、**vsan 2** の VSAN/VLAN マッピング、VLAN プール **1-10**、および VSAN プール **1-10** を、VSAN ドメイン **dom1** から削除します。

例：

```
apic1(config)# vsan-domain dom1
apic1(config-vsan)# show run
# Command: show running-config vsan-domain dom1
# Time: Tue Jul 26 09:43:47 2016
  vsan-domain dom1
    vsan 1-10
    vlan 1-10
    fcoe vsan 2 vlan 2
    exit
apic1(config-vsan)# no fcoe vsan 2
apic1(config-vsan)# no vlan 1-10
apic1(config-vsan)# no vsan 1-10
apic1(config-vsan)# exit

################################################################################
NOTE: To remove a template-based VSAN to VLAN mapping use an alternate sequence:
################################################################################

apic1(config)# template vsan-attribute <template_name>
apic1(config-vsan-attr)# no fcoe vsan 2
```

ステップ 3 VSAN ドメインを削除します。

例は、ドメインの VSAN を削除する **dom1** 。

例：

```
apic1(config)# no vsan-domain dom1
```

ステップ 4 必要はないかどうかは、関連付けられているテナント、EPG、およびセレクタを削除できます。

REST API を使用した FCoE の設定

Configuring FCoE Connectivity Using the REST API

REST API で FCoE を使用したインターフェイスにアクセスする、FCoE が有効なインターフェイスと EPG を設定できます。

手順

ステップ 1 VSAN プールを作成するには、次の例のように XML で post を送信します。

例では VSAN プール **vsanPool1** を作成し、含まれている VSAN の範囲を指定します。

例：

```
https://apic-ip-address/api/mo/uni/infra/vsanns-[vsanPool1]-static.xml

<!-- Vsan-pool -->
<fvnsVsanInstP name="vsanPool1" allocMode="static">
     <fvnsVsanEncapBlk name="encap" from="vsan-5" to="vsan-100"/>
</fvnsVsanInstP>
```

ステップ 2 VLAN プールを作成するには、次の例のように XML で post を送信します。

例では VLAN プール **vlanPool1** を作成し、含まれている VLAN の範囲を指定します。

例：

```
https://apic-ip-address/api/mo/uni/infra/vlanns-[vlanPool1]-static.xml

<!-- Vlan-pool -->
<fvnsVlanInstP name="vlanPool1" allocMode="static">
 <fvnsEncapBlk name="encap" from="vlan-5" to="vlan-100"/>
</fvnsVlanInstP>
```

ステップ 3 VSAN 属性ポリシーを作成するには、次の例のように XML で post を送信します。

例では VSAN 属性ポリシー **vsanattri1** を作成し、**vsan 10** を **vlan 43** にマップし、**vsan 11** を **vlan 44** にマップします。

例：

```
https://apic-ip-address/api/mo/uni/infra/vsanattrp-[vsanattr1].xml

<fcVsanAttrP name="vsanattr1">

    <fcVsanAttrPEntry vlanEncap="vlan-43" vsanEncap="vsan-10"/>
```

```
    <fcVsanAttrPEntry vlanEncap="vlan-44" vsanEncap="vsan-11"
     lbType="src-dst-ox-id"/>
</fcVsanAttrP>
```

ステップ 4 ファイバ チャネル ドメインを作成するには、次の例のように XML で post を送信します。

この例では、VSAN ドメイン **vsanDom1** を作成します。

例：

```
https://apic-ip-address/api/mo/uni/fc-vsanDom1.xml
<!-- Vsan-domain -->
<fcDomP name="vsanDom1">
     <fcRsVsanAttr tDn="uni/infra/vsanattrp-[vsanattr1]"/>
     <infraRsVlanNs tDn="uni/infra/vlanns-[vlanPool1]-static"/>
     <fcRsVsanNs tDn="uni/infra/vsanns-[vsanPool1]-static"/>
</fcDomP>
```

ステップ 5 テナント、アプリケーション プロファイル、EPG を作成し、FCoE ブリッジ ドメインを EPG に関連付けるには、次の例のように XML で post を送信します。

この例では、ブリッジ ドメイン **bd1** を、FCoE およびアプリケーション EPG **epg1** をサポートするように設定されたターゲット テナントの下に作成します。これは EPG を VSAN ドメイン vsanDom1 に、そしてファイバ チャネル パス (インターフェイス **1/39** に向かう、リーフ スイッチ **101** 上にあるもの) に関連付けます。これは、<fvRsFcPathAtt> オブジェクトを "deleted" ステータスに割り当てることにより、インターフェイス **1/40** へのファイバ チャネル パスを削除します。各インターフェイスは、VSAN に関連付けられます。

（注） その他の 2 つの代替可能な vFC 展開も表示されます。1 つの例では、ポート チャネルで vFC を展開します。その他の例では、仮想ポート チャネルで vFC を展開します。

例：

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant
name="tenant1">
  <fvCtx name="vrf1"/>

  <!-- bridge domain -->
  <fvBD name="bd1" type="fc" >
    <fvRsCtx tnFvCtxName="vrf1" />
  </fvBD>

  <fvAp name="app1">
    <fvAEPg name="epg1">
      <fvRsBd tnFvBDName="bd1" />
      <fvRsDomAtt tDn="uni/fc-vsanDom1" />
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/39]"
       vsan="vsan-11" vsanMode="native"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
       vsan="vsan-10" vsanMode="regular" status="deleted"/>
    </fvAEPg>

<!-- Sample deployment of vFC on a port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDN="topology/pod-1/paths 101/pathep-pc01"/>

<!-- Sample deployment of vFC on a virtual port channel -->
```

```
      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-101/pathep-vpc01"/>
      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-102/pathep-vpc01"/>

  </fvAp>
</fvTenant>
```

ステップ 6 ポート ポリシー グループおよび AEP を作成するには、次の例のように XML で POST を送信します。

この例では、次のリクエストを実行します:

- ポリシー グループ **portgrp1** を作成します。これはFCインターフェイス ポリシー **fcIfPol1**、プライオリティ フロー制御ポリシー **pfcIfPol1** およびスロードレイン ポリシー **sdIfPol1** を含んでいます。
- アタッチ エンティティ プロファイル (AEP) **AttEntP1** を作成します。これは、VSAN ドメイン **vsanDom1** 内のポートを、**fcIfPol1**、 **pfcIfPol1**、および **sdIfPol1** のために指定される設定と関連付けます。

例：

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
  <infraInfra>
    <infraFuncP>
      <infraAccPortGrp name="portgrp1">
        <infraRsFcIfPol tnFcIfPolName="fcIfPol1"/>
        <infraRsAttEntP tDn="uni/infra/attentp-AttEntP1" />
        <infraRsQosPfcIfPol tnQosPfcIfPolName="pfcIfPol1"/>
        <infraRsQosSdIfPol tnQosSdIfPolName="sdIfPol1"/>
      </infraAccPortGrp>
    </infraFuncP>

    <infraAttEntityP name="AttEntP1">
      <infraRsDomP tDn="uni/fc-vsanDom1"/>
    </infraAttEntityP>
    <qosPfcIfPol dn="uni/infra/pfc-pfcIfPol1" adminSt="on">
    </qosPfcIfPol>
    <qosSdIfPol dn="uni/infra/qossdpol-sdIfPol1" congClearAction="log"
     congDetectMult="5" flushIntvl="100" flushAdminSt="enabled">
    </qosSdIfPol>
    <fcIfPol dn="uni/infra/fcIfPol-fcIfPol1" portMode="np">
    </fcIfPol>

 </infraInfra>
</polUni>
```

ステップ 7 ノードセレクタおよびポートセレクタを作成するには、次の例のように XML で POST を送信します。

この例では、次のリクエストを実行します:

- ノードセレクタ **leafsel1** を作成します。これはリーフ ノード **101** を指定します。
- ポートセレクタ **portsel1** を作成します。これはポート **1/39** を指定します。

例：

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
  <infraInfra>
    <infraNodeP name="nprof1">
      <infraLeafS name="leafsel1" type="range">
        <infraNodeBlk name="nblk1" from_="101" to_="101"/>
      </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-pprof1"/>
    </infraNodeP>

    <infraAccPortP name="pprof1">
      <infraHPortS name="portsel1" type="range">
        <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="39" toPort="39">
        </infraPortBlk>

        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-portgrp1" />
      </infraHPortS>

    </infraAccPortP>
 </infraInfra>
</polUni>
```

ステップ 8 vPC を作成するには、次の例のような XML を POST 送信します。

例：

```
https://apic-ip-address/api/mo/uni.xml
<polUni>
  <fabricInst>

     <vpcInstPol name="vpc01" />

     <fabricProtPol pairT="explicit" >
        <fabricExplicitGEp name="vpc01" id="100" >
           <fabricNodePEp id="101"/>
           <fabricNodePEp id="102"/>
           <fabricRsVpcInstPol tnVpcInstPolName="vpc01" />
           <!-- <fabricLagId accBndlGrp="infraAccBndlGrp_{{pcname}}" /> -->
        </fabricExplicitGEp>
     </fabricProtPol>

  </fabricInst>
</polUni>
```

REST API を使用した FEX で FCoE の設定

始める前に

- 説明されている 1 ~ 4 の手順に従います Configuring FCoE Connectivity Using the REST API（197 ページ）

手順

ステップ 1 FEX（セレクタ）上の FCoE の設定：ポート：

例：

```
<infraInfra dn="uni/infra">
  <infraNodeP name="nprof1">
    <infraLeafS name="leafsel1" type="range">
      <infraNodeBlk name="nblk1" from_="101" to_="101"/>
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-pprof1" />
  </infraNodeP>

    <infraAccPortP name="pprof1">
      <infraHPortS name="portsel1" type="range">
        <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="17" toPort="17"></infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/fexprof-fexprof1/fexbundle-fexbundle1"
fexId="110" />
      </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
    <infraAccPortGrp name="portgrp1">
      <infraRsAttEntP tDn="uni/infra/attentp-attentp1" />
    </infraAccPortGrp>
    </infraFuncP>

  <infraFexP name="fexprof1">
    <infraFexBndlGrp name="fexbundle1"/>
    <infraHPortS name="portsel2" type="range">
      <infraPortBlk name="blk2"
        fromCard="1" toCard="1" fromPort="20" toPort="20"></infraPortBlk>
      <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-portgrp1"/>
    </infraHPortS>
  </infraFexP>

 <infraAttEntityP name="attentp1">
<infraRsDomP tDn="uni/fc-vsanDom1"/>
 </infraAttEntityP>
</infraInfra>
```

ステップ 2 テナント設定：

例：

```
fvTenant name="tenant1">
<fvCtx name="vrf1"/>

<!-- bridge domain -->
    <fvBD name="bd1" type="fc" >
    <fvRsCtx tnFvCtxName="vrf1" />
    </fvBD>

<fvAp name="app1">
    <fvAEPg name="epg1">
        <fvRsBd tnFvBDName="bd1" />
       <fvRsDomAtt tDn="uni/fc-vsanDom1" />
<fvRsFcPathAtt tDn="topology/pod-1/paths-101/extpaths-110/pathep-[eth1/17]" vsan="vsan-11"
 vsanMode="native"/>
        </fvAEPg>
```

```
    </fvAp>
</fvTenant>
```

ステップ 3 FEX（セレクタ）上の FCoE の設定：ポート チャネル：

例：

```
<infraInfra dn="uni/infra">
  <infraNodeP name="nprof1">
    <infraLeafS name="leafsel1" type="range">
      <infraNodeBlk name="nblk1" from_="101" to_="101"/>
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-pprof1" />
  </infraNodeP>

    <infraAccPortP name="pprof1">
      <infraHPortS name="portsel1" type="range">
        <infraPortBlk name="blk1"
        fromCard="1" toCard="1" fromPort="18" toPort="18"></infraPortBlk>
        <infraRsAccBaseGrp tDn="uni/infra/fexprof-fexprof1/fexbundle-fexbundle1"
fexId="111" />
      </infraHPortS>
    </infraAccPortP>

  <infraFexP name="fexprof1">
    <infraFexBndlGrp name="fexbundle1"/>
    <infraHPortS name="portsel1" type="range">
      <infraPortBlk name="blk1"
        fromCard="1" toCard="1" fromPort="20" toPort="20"></infraPortBlk>
      <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-pc1"/>
    </infraHPortS>
  </infraFexP>

  <infraFuncP>
    <infraAccBndlGrp name="pc1">
      <infraRsAttEntP tDn="uni/infra/attentp-attentp1" />
    </infraAccBndlGrp>
  </infraFuncP>

 <infraAttEntityP name="attentp1">
<infraRsDomP tDn="uni/fc-vsanDom1"/>
 </infraAttEntityP>
</infraInfra>
```

ステップ 4 テナント設定：

例：

```
<fvTenant name="tenant1">
<fvCtx name="vrf1"/>

<!-- bridge domain -->
    <fvBD name="bd1" type="fc" >
    <fvRsCtx tnFvCtxName="vrf1" />
    </fvBD>

<fvAp name="app1">
    <fvAEPg name="epg1">
        <fvRsBd tnFvBDName="bd1" />
       <fvRsDomAtt tDn="uni/fc-vsanDom1" />
<fvRsFcPathAtt tDn="topology/pod-1/paths-101/extpaths-111/pathep-[pc1]" vsan="vsan-11"
vsanMode="native" />
        </fvAEPg>
    </fvAp>
</fvTenant>
```

ステップ 5 FCoE over FEX の設定（セレクタ）：vPC：

例：

```
<polUni>
<fabricInst>
<vpcInstPol name="vpc1" />
<fabricProtPol pairT="explicit" >
<fabricExplicitGEp name="vpc1" id="100" >
<fabricNodePEp id="101"/>
<fabricNodePEp id="102"/>
<fabricRsVpcInstPol tnVpcInstPolName="vpc1" />
</fabricExplicitGEp>
</fabricProtPol>
</fabricInst>
</polUni>
```

ステップ 6 テナント設定：

例：

```
<fvTenant name="tenant1">
<fvCtx name="vrf1"/>

<!-- bridge domain -->
    <fvBD name="bd1" type="fc" >
    <fvRsCtx tnFvCtxName="vrf1" />
    </fvBD>

<fvAp name="app1">
    <fvAEPg name="epg1">
        <fvRsBd tnFvBDName="bd1" />
       <fvRsDomAtt tDn="uni/fc-vsanDom1" />
<fvRsFcPathAtt vsanMode="native" vsan="vsan-11"
tDn="topology/pod-1/protpaths-101-102/extprotpaths-111-111/pathep-[vpc1]" />
</fvAEPg>
    </fvAp>
</fvTenant>
```

ステップ 7 セレクタ設定：

例：

```
<polUni>
<infraInfra>
<infraNodeP name="nprof1">
<infraLeafS name="leafsel1" type="range">
<infraNodeBlk name="nblk1" from_="101" to_="101"/>
</infraLeafS>
<infraRsAccPortP tDn="uni/infra/accportprof-pprof1" />
</infraNodeP>

<infraNodeP name="nprof2">
<infraLeafS name="leafsel2" type="range">
<infraNodeBlk name="nblk2" from_="102" to_="102"/>
</infraLeafS>
<infraRsAccPortP tDn="uni/infra/accportprof-pprof2" />
</infraNodeP>

<infraAccPortP name="pprof1">
<infraHPortS name="portsel1" type="range">
<infraPortBlk name="blk1"
fromCard="1" toCard="1" fromPort="18" toPort="18">
</infraPortBlk>
<infraRsAccBaseGrp tDn="uni/infra/fexprof-fexprof1/fexbundle-fexbundle1" fexId="111" />
```

```
</infraHPortS>
</infraAccPortP>
<infraAccPortP name="pprof2">
<infraHPortS name="portsel2" type="range">
<infraPortBlk name="blk2"
fromCard="1" toCard="1" fromPort="18" toPort="18">
</infraPortBlk>
<infraRsAccBaseGrp tDn="uni/infra/fexprof-fexprof2/fexbundle-fexbundle2" fexId="111" />
</infraHPortS>
</infraAccPortP>

<infraFexP name="fexprof1">
<infraFexBndlGrp name="fexbundle1"/>
<infraHPortS name="portsel1" type="range">
<infraPortBlk name="blk1"
fromCard="1" toCard="1" fromPort="20" toPort="20">
</infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-vpc1"/>
</infraHPortS>
</infraFexP>

<infraFexP name="fexprof2">
<infraFexBndlGrp name="fexbundle2"/>
<infraHPortS name="portsel2" type="range">
<infraPortBlk name="blk2"
fromCard="1" toCard="1" fromPort="20" toPort="20">

</infraPortBlk>
<infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-vpc1"/>
</infraHPortS>
</infraFexP>

<infraFuncP>
<infraAccBndlGrp name="vpc1" lagT="node">
    <infraRsAttEntP tDn="uni/infra/attentp-attentp1" />
</infraAccBndlGrp>
</infraFuncP>

<infraAttEntityP name="attentp1">
<infraRsDomP tDn="uni/fc-vsanDom1"/>
</infraAttEntityP>
</infraInfra>
</polUni>
```

REST API を使用した FCoE vPC の設定

この手順では、仮想ポート チャネル（vPC）を作成します。

手順

ステップ 1 vPC ドメインを作成します。

このステップでは、グループ ポリシー（fabric:ExplicitGEp）が含まれている仮想ポート チャネル セキュリティ ポリシー（fabric:ProtPol）を作成します。グループ ポリシーには、「101」お

よび「102」という名前の 2 つのノード ポリシー エンドポイント（fabric:NodePEp）が含まれています。

例：

```
POST https://apic-ip-address/api/node/mo/uni/fabric/protpol.xml

<fabricProtPol>
    <fabricExplicitGEp name="vpc-explicitGrp1101102" id="100" >
        <fabricNodePEp id="101" />
        <fabricNodePEp id="102" />
    </fabricExplicitGEp>
</fabricProtPol>
```

ステップ 2 ファイバ チャネル インターフェイス ポリシーを作成します。

このステップでは、トランク モードが有効になっている、「vpc1」という名前のファイバチャネル インターフェイス ポリシー（fc:IfPol）を作成します。

例：

```
POST https://apic-ip-address/api/node/mo/uni/infra/fcIfPol-vpc1.xml

<fcIfPol name="vpc1" trunkMode="trunk-on" />
```

ステップ 3 LACP ポート チャネル ポリシーを作成します。

このステップでは、LACPアクティブモードが有効になっている、「vpc1」という名前のLACPポート チャネル ポリシー（lacp:LagPol）を作成します。suspend-individual-port コントロールはポート チャネルから無効にされています。そうでない場合、ホストからLACP BPDUが受信されないときに物理インターフェイスが中断されます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/infra/lacplagp-vpc1.xml

<lacpLagPol name="vpc1" mode="active" ctrl="graceful-conv,fast-sel-hot-stdby" />
```

ステップ 4 vPC の作成

例：

```
POST https://apic-ip-address/api/node/mo/uni/infra.xml

<infraInfra>
  <infraAccPortP
      name="Switch101-102_Profile_ifselector"
      descr="GUI Interface Selector Generated PortP Profile: Switch101-102_Profile">
    <infraHPortS name="Switch101-102_1-ports-49" type="range">
      <infraPortBlk name="block1" fromPort="49" toPort="49" />
      <infraRsAccBaseGrp
          tDn="uni/infra/funcprof/accbundle-Switch101-102_1-ports-49_PolGrp" />
    </infraHPortS>
  </infraAccPortP>
  <infraFuncP>
    <infraAccBndlGrp name="Switch101-102_1-ports-49_PolGrp" lagT="node">
```

```
      <infraRsAttEntP tDn="uni/infra/attentp-fcDom_AttEntityP" />
      <infraRsFcIfPol tnFcIfPolName="vpc1" />
      <infraRsLacpPol tnLacpLagPolName="vpc1" />
    </infraAccBndlGrp>
  </infraFuncP>
  <infraNodeP
      name="Switch101-102_Profile"
      descr="GUI Interface Selector Generated Profile: Switch101-102_Profile">
    <infraLeafS name="Switch101-102_Profile_selector_101102" type="range">
      <infraNodeBlk name="single0" from_="101" to_="101" />
      <infraNodeBlk name="single1" from_="102" to_="102" />
    </infraLeafS>
    <infraRsAccPortP
        tDn="uni/infra/accportprof-Switch101-102_Profile_ifselector" />>
  </infraNodeP>
</infraInfra>
```

ステップ 5 ネイティブ VLAN を作成します。

a) ブリッジ ドメインを作成し、VRF に関連付けます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/tn-newtenant/BD-BDnew1.xml

<fvBD name="BDnew1" mac="00:22:BD:F8:19:FF" >
  <fvRsCtx tnFvCtxName="vrf" />
</fvBD>
```

b) EPG アプリケーションを作成し、ブリッジ ドメインに関連付けます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/tn-newtenant/ap-AP1/epg-epgNew.xml

<fvAEPg name="epgNew" >
  <fvRsBd tnFvBDName="BDnew1" />
</fvAEPg>
```

c) スタティック パスを作成し、VLAN に関連付けます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/tn-newtenant/ap-AP1/epg-epgNew.xml

<fvRsPathAtt
  encap="vlan-1"
  instrImedcy="immediate"
  mode="native"
  tDn="topology/pod-1/protpaths-101-102/pathep-[Switch101-102_1-ports-49_PolGrp]" />
```

ステップ 6 vFC を作成します。

a) ブリッジ ドメインを作成し、VRF に関連付けます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/tn-newtenant/BD-BD3.xml

<fvBD
    name="BD3"
    mac="00:22:BD:F8:19:FF"
    type="fc"
    unicastRoute="false" >
  <fvRsCtx tnFvCtxName="vrf" />
</fvBD>
```

b) EPG アプリケーションを作成し、ブリッジ ドメインに関連付けます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/tn-newtenant/ap-AP1/epg-epg3.xml

<fvAEPg name="epg3" >
  <fvRsBd tnFvBDName="BD3" />
</fvAEPg>
```

c) スタティック パスを作成し、VSAN に関連付けます。

例：

```
POST https://apic-ip-address/api/node/mo/uni/tn-newtenant/ap-AP1/epg-epg3.xml

<fvRsFcPathAtt
    vsan="vsan-3"
    vsanMode="native"
    tDn="topology/pod-1/paths-101/pathep-[eth1/49]" />
```

REST API または SDK 経由の FCoE 接続の解除

APIC REST API または SDK で FCoE 接続を展開解除するには、展開に関連付けられている次のオブジェクトを削除します。

オブジェクト	説明
<fvRsFcPathAtt>（ファイバ チャネル パス）	ファイバ チャネル パスは、実際のインターフェイスに vFC パスを指定します。このタイプの各オブジェクトを削除すると、そのオブジェクトの関連付けられているインターフェイスから展開が削除されます。
<fcVsanAttrpP>（VSAN/VLAN マップ）	VSAN/VLAN マップは VSAN を関連付けられる VLAN にマップします。このオブジェクトを削除することで、FCoE 接続をサポートする VSAN 間の関連付けと VSAN の基盤を削除します。

オブジェクト	説明
<fvnsVsanInstP>（VSAN プール）	VSAN プールは、FCoE 接続をサポートする利用可能な VSANのセットを指定します。このプールを削除すると、それらの VSAN が削除されます。
<fvnsVlanIsntP>（(VLAN プール）	VLAN プールは、VSAN マッピングの VLAN セットを指定します。関連杖けられている VLAN プールを削除するとFCoEの展開解除後にクリーンアップされ、VSANエンティティが実行された VLAN エンティティの基盤を削除します。
<fcDomP> (VSAN またはファイバ チャネル ドメイン)	ファイバ チャネル ドメインには、すべての VSAN とそのマッピングが含まれています。このオブジェクトを削除すると、このドメインに関連付けられているすべてのインターフェイスから vFC の展開を解除します。
<fvAEPg>（アプリケーション EPG）	FCoE接続に関連付けられているEPGアプリケーション。アプリケーションEPGの目的がFCoEに関連するアクティビティをサポートするためだけの場合、このオブジェクトの削除を検討できます。
<fvAp>（アプリケーション プロファイル）	FCoE 接続に関連付けられているアプリケーション プロファイル。アプリケーションプロファイルの目的がFCoEに関連するアクティビティをサポートするためだけの場合、このオブジェクトの削除を検討できます。
<fvTenant>（テナント）	FCoE接続に関連付けられているテナント。テナントの目的がFCoEに関連するアクティビティをサポートするためだけの場合、このオブジェクトの削除を検討できます。

（注） クリーンアップ中に vFC ポートのイーサネット設定オブジェクトを削除する場合（infraHPortS）、デフォルトの vFC プロパティはそのインターフェイスに関連付けられて残ります。たとえば、vFC NP ポート 1/20 のインターフェイス設定が削除される場合、そのポートは vFC ポートのままですが、デフォルト NP 設定以外のデフォルト F にポート設定が適用された状態です。

次の大中で、FCoE プロトコルを使用するインターフェイスにアクセスする FCoE が有効なインターフェイスおよび EPG を展開解除します。

手順

ステップ 1 関連付けられたファイバ チャネル パス オブジェクトを削除するには、次の例のように XML で POST を送信します。

例では、ファイバ チャネル パス オブジェクト<fvRsFcPathAtt> のすべてのインスタンスを削除します。

（注） ファイバ チャネル パスを削除すると、使用されたポート/VSAN から vFC を展開解除します。

例：

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant
name="tenant1">
  <fvCtx name="vrf1"/>

  <!-- bridge domain -->
  <fvBD name="bd1" type="fc" >
    <fvRsCtx tnFvCtxName="vrf1" />
  </fvBD>

  <fvAp name="app1">
    <fvAEPg name="epg1">
      <fvRsBd tnFvBDName="bd1" />
      <fvRsDomAtt tDn="uni/fc-vsanDom1" />
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/39]"
       vsan="vsan-11" vsanMode="native" status="deleted"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
       vsan="vsan-10" vsanMode="regular" status="deleted"/>
    </fvAEPg>

<!-- Sample undeployment of vFC on a port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDN="topology/pod-1/paths 101/pathep-pc01" status="deleted"/>

<!-- Sample undeployment of vFC on a virtual port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-101/pathep-vpc01" status="deleted"/>
      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-102/pathep-vpc01" status="deleted"/>

  </fvAp>
</fvTenant>
```

ステップ 2 関連付けられている VSAN/VLAN マップを削除するには、次の例のように post を送信します。

例では、VSAN/VLANマップ **vsanattri1** と、関連づけられた <fcVsanAttrpP> オブジェクトを削除します。

例：

```
https://apic-ip-address/api/mo/uni/infra/vsanattrp-[vsanattr1].xml

<fcVsanAttrP name="vsanattr1" status="deleted">

    <fcVsanAttrPEntry vlanEncap="vlan-43" vsanEncap="vsan-10" status="deleted"/>
    <fcVsanAttrPEntry vlanEncap="vlan-44" vsanEncap="vsan-11"
     lbType="src-dst-ox-id" status="deleted" />
</fcVsanAttrP>
```

ステップ 3 関連付けられている VSAN プールを削除するには、次の例のように post を送信します。

この例では、VSAN プール **vsanPool1** と、関連づけられた<fvnsVsanInstP>オブジェクトを削除します。

例：

```
https://apic-ip-address/api/mo/uni/infra/vsanns-[vsanPool1]-static.xml

<!-- Vsan-pool -->
<fvnsVsanInstP name="vsanPool1" allocMode="static" status="deleted">
 <fvnsVsanEncapBlk name="encap" from="vsan-5" to="vsan-100" />
</fvnsVsanInstP>
```

ステップ 4 関連付けられている VLAN プールを削除するには、次の例にように XML で post を送信します。

この例では、VLAN プール **vlanPool1** と、関連づけられた<fvnsVlanIsntP>オブジェクトを削除します。

例：

```
https://apic-ip-address/api/mo/uni/infra/vlanns-[vlanPool1]-static.xml

<!-- Vlan-pool -->
<fvnsVlanInstP name="vlanPool1" allocMode="static" status="deleted">
 <fvnsEncapBlk name="encap" from="vlan-5" to="vlan-100" />
</fvnsVlanInstP>
```

ステップ 5 関連づけられたファイバ チャネル ドメインを削除するには、次の例のように XML で post を送信します。

例では、VSAN ドメイン **vsanDom1** と、関連づけられた<fcDomP>オブジェクトを削除します。

例：

```
https://apic-ip-address/api/mo/uni/fc-vsanDom1.xml
<!-- Vsan-domain -->
<fcDomP name="vsanDom1" status="deleted">
     <fcRsVsanAttr tDn="uni/infra/vsanattrp-[vsanattr1]"/>
     <infraRsVlanNs tDn="uni/infra/vlanns-[vlanPool1]-static"/>
     <fcRsVsanNs tDn="uni/infra/vsanns-[vsanPool1]-static"/>
</fcDomP>
```

ステップ 6 **オプション**：適切な場合、関連づけられた EPG、関連づけられたアプリケーション プロファイル、または関連づけられたテナントを削除できます。

例：

次の例では、関連付けられているアプリケーション EPG **epg1** と、関連づけられた<fvAEPg>オブジェクトが削除されます。

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant
name="tenant1"/>
  <fvCtx name="vrf1"/>

  <!-- bridge domain -->
  <fvBD name="bd1" type="fc" >
    <fvRsCtx tnFvCtxName="vrf1" />
  </fvBD>

  <fvAp name="app1">
```

```
    <fvAEPg name="epg1" status= "deleted">
      <fvRsBd tnFvBDName="bd1" />
      <fvRsDomAtt tDn="uni/fc-vsanDom1" />
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/39]"
       vsan="vsan-11" vsanMode="native" status="deleted"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
       vsan="vsan-10" vsanMode="regular" status="deleted"/>
    </fvAEPg>

<!-- Sample undeployment of vFC on a port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDN="topology/pod-1/paths 101/pathep-pc01" status="deleted"/>

<!-- Sample undeployment of vFC on a virtual port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-101/pathep-vpc01" status="deleted"/>
      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-102/pathep-vpc01" status="deleted"/>

  </fvAp>
</fvTenant>
```

例：

次の例では、関連付けられているアプリケーション プロファイル **app1** と、関連づけられた <fvAp>オブジェクトが削除されます。

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant
name="tenant1">
  <fvCtx name="vrf1"/>

  <!-- bridge domain -->
  <fvBD name="bd1" type="fc">
    <fvRsCtx tnFvCtxName="vrf1" />
  </fvBD>

  <fvAp name="app1" status="deleted">
    <fvAEPg name="epg1" status= "deleted">
      <fvRsBd tnFvBDName="bd1" />
      <fvRsDomAtt tDn="uni/fc-vsanDom1" />
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/39]"
       vsan="vsan-11" vsanMode="native" status="deleted"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
       vsan="vsan-10" vsanMode="regular" status="deleted"/>
    </fvAEPg>

<!-- Sample undeployment of vFC on a port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDN="topology/pod-1/paths 101/pathep-pc01" status="deleted"/>

<!-- Sample undeployment of vFC on a virtual port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-101/pathep-vpc01" status="deleted"/>
      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-102/pathep-vpc01" status="deleted"/>

  </fvAp>
</fvTenant>
```

例：

次の例では、全体のテナント **tenant1** と関連づけられた<fvTenant>オブジェクトが削除されます。

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant
name="tenant1" status="deleted">
  <fvCtx name="vrf1"/>

  <!-- bridge domain -->
  <fvBD name="bd1" type="fc" status="deleted">
    <fvRsCtx tnFvCtxName="vrf1" />
  </fvBD>

  <fvAp name="app1">
    <fvAEPg name="epg1" status= "deleted">
      <fvRsBd tnFvBDName="bd1" />
      <fvRsDomAtt tDn="uni/fc-vsanDom1" />
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/39]"
       vsan="vsan-11" vsanMode="native" status="deleted"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/40]"
       vsan="vsan-10" vsanMode="regular" status="deleted"/>
    </fvAEPg>

<!-- Sample undeployment of vFC on a port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDN="topology/pod-1/paths 101/pathep-pc01" status="deleted"/>

<!-- Sample undeployment of vFC on a virtual port channel -->

      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-101/pathep-vpc01" status="deleted"/>
      <fvRsFcPathAtt vsanMode="native" vsan="vsan-10"
       tDn="topology/pod-1/paths-102/pathep-vpc01" status="deleted"/>

  </fvAp>
</fvTenant>
```

vPC による SAN ブート

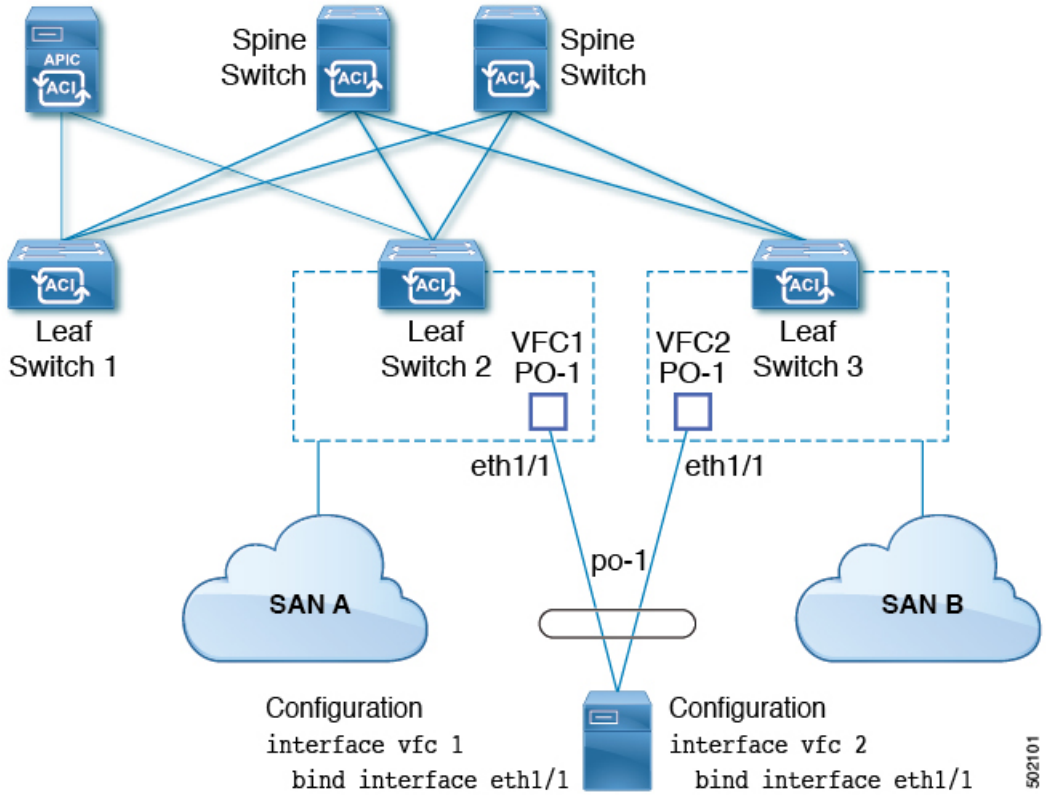
Cisco ACI は、Link Aggregation Control Protocol（LACP）ベースの vPC におけるイニシエータの SAN ブートをサポートしています。この制限事項は、LACP ベースのポート チャネルに固有です。

通常のホスト-vPC トポロジでは、ホストに接続している vFC インターフェイスは vPC にバインドされており、vFCインターフェイスをアップする前にvPCを論理的にアップする必要があります。このトポロジでは、vPC で LACP が設定されている場合、ホストは SAN からブートできません。これは、ホストのLACPは通常はアダプタのファームウェアで実装されているのではなく、ホスト ドライバで実装されているためです。

SAN ブートについては、ホストに接続している vFC インターフェイスは、 ポート チャネル自体ではなく、ポート チャネルのメンバーにバインドされています。このバインディングによ

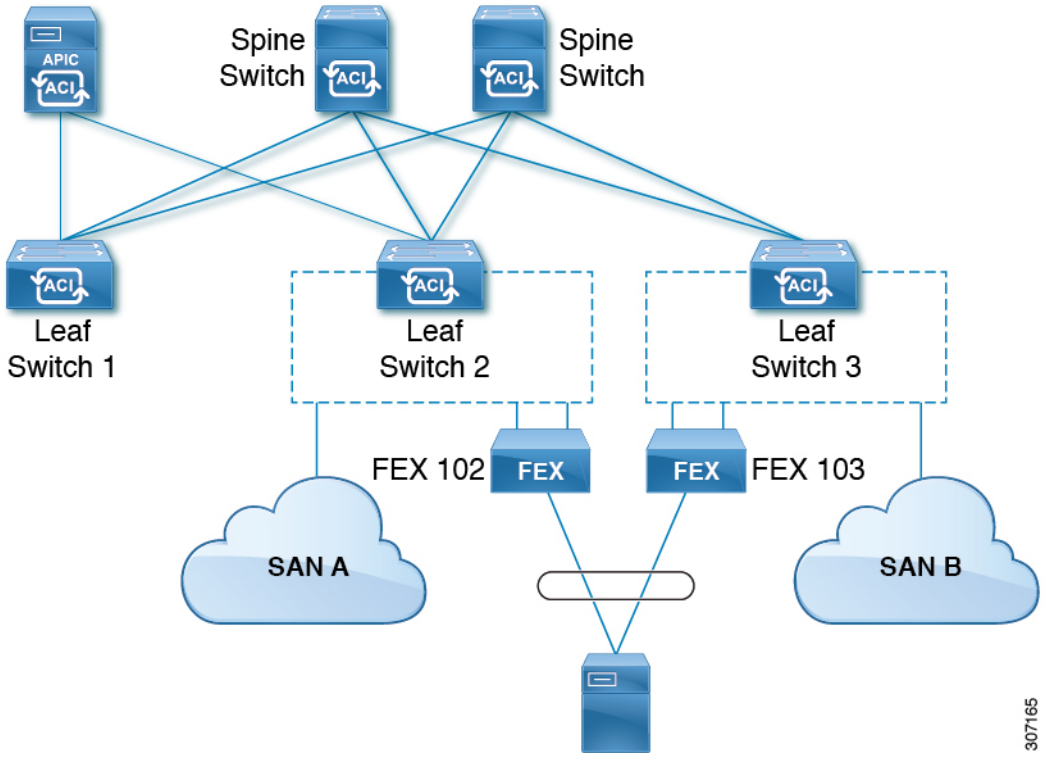
り、最初の構成でLACPベースのポートチャネルに依存することなく、CNA/ホストバスアダプタ（HBA）のリンクがアップした時点で、SANブート中にホスト側のvFCがアップするようになります。

図 *29 : vPC* による *SAN* ブートのトポロジ



Cisco APIC リリース 4.0(2) 以降、次の図に示すように、SANブートはFEXホストインターフェイス（HIF）ポートvPCを介してサポートされます。

図 *30 : FEX* ホストインターフェイス（*HIF*）ポート *vPC* を使用した *SAN* ブート トポロジ



vPC による SAN ブートのガイドラインと制約事項

- 複数のメンバーのポート チャネルはサポートされていません。
- vFC がメンバー ポートにバインドされている場合、ポート チャネルに複数のメンバーを持たせることはできません。
- vFC がポート チャネルにバインドされている場合、ポート チャネルには 1 つのメンバー ポートしか持たせることはできません。

GUI を使用した vPC による SAN ブートの設定

設定を簡単に行うため、この手順では [Configure Interface, PC, and vPC] ウィザード（**[Fabric] > [Access Policies] > [Quickstart]**）を使用します。

始める前に

この手順では、次の項目がすでに設定済みであることを前提としています。

- VSAN Pool
- VLAN Pool
- VSAN の属性、 VSAN プール内の VSAN の VLAN へのマッピング

- ファイバ チャネル ドメイン（VSAN ドメイン）
- テナント、アプリケーション プロファイル
- アタッチ エンティティ プロファイル

手順

ステップ 1 APIC メニュー バーで、[Fabric] > [Access Policies] > [Quick Start] に移動し、[Configure an interface, PC, and VPC] をクリックします。

ステップ 2 [Configure an interface, PC, and VPC] 作業領域の [vPC Switch Pairs] ツールバーで、[+] をクリックしてスイッチ ペアを作成します。次のアクションを実行します。

a) [vPC Domain ID] テキスト ボックスで、スイッチ ペアを指定する番号を入力します。
b) [Switch 1] ドロップダウンリストで、リーフ スイッチを選択します。

同じ vPC ポリシー グループ内のインターフェイスを持つスイッチのみをペアリングできます。

c) [Switch 2] ドロップダウンリストで、リーフ スイッチを選択します。
d) [Save] をクリックしてこのスイッチ ペアを保存します。

ステップ 3 [Configure an interface, PC, and vPC] 作業領域で、緑色の大きい[+] をクリックし、スイッチを選択します。
[Select Switches To Configure Interfaces] 作業領域が開き、[Quick] オプションがデフォルトで選択されます。

ステップ 4 [Switches] ドロップダウンリストから 2 つのスイッチ ID を選択し、スイッチ プロファイルに名前を付けます。

ステップ 5 再び緑色の大きい [+] をクリックし、スイッチ インターフェイスを設定します。

ステップ 6 [Interface Type] コントロールで、[vPC] を選択します。

ステップ 7 [Interfaces] には、両方のスイッチでvPCメンバーとして使用される1つのポート番号（1/49 など）を入力します。

この操作によってインターフェイス セレクタ ポリシーが作成されます。[Interface Selector Name] テキスト ボックスで、ポリシーの名前を受け入れるか変更できます。

ステップ 8 [Interface Policy Group] コントロールで、[Create One] を選択します。

ステップ 9 [Fibre Channel Interface Policy] テキストボックスから、[Create Fibre Channel Interface Policy] を選択し、次の操作を実行します。

a) [Name] フィールドに、ファイバ チャネル インターフェイス ポリシーの名前を入力します。
b) [Port Mode] セレクタで、[F] を選択します。
c) [Trunk Mode] セレクタで、[trunk-on] を選択します。
d) [Submit] をクリックします。

ステップ 10 [Port Channel Policy] テキスト ボックスで、[Create Port Channel Policy] を選択し、次の操作を実行します。

a) [Name] フィールドに、ポート チャネル ポリシーの名前を入力します。
b) [Mode] ドロップダウンリストで、[LACP Active] を選択します。
c) [Control] セレクタから [Suspend Individual Port] を削除します。

[Suspend Individual Port] はポート チャネルから削除する必要があります。削除しないと、ホストからの LACP BPDU が受信されない場合に物理インターフェイスが中断されます。

d) [Submit] をクリックします。

ステップ 11 [Attached Device Type] ドロップダウンリストで、[Fibre Channel] を選択します。

ステップ 12 [Fibre Channel Domain] ドロップダウンリストで、ファイバ チャネル ドメイン（VSAN ドメイン）を選択します。

ステップ 13 [保存（Save）] をクリックして、この vPC 設定を保存します。

ステップ 14 [Save] をクリックして、このインターフェイス設定を保存します。

ステップ 15 [Submit] をクリックします。

ステップ 16 [Tenants] > [<テナント名>] > [Application Profiles] > [<名前>] > [Application EPGs] の順に展開します。

ステップ 17 [Application EPGs] を右クリックし、[Create Application EPG] を選択して、次の操作を実行します。

この EPG がネイティブ EPG になり、ネイティブ VLAN が設定されます。

a) [Name] フィールドに、EPG の名前を入力します。
b) [Bridge Domain] ドロップダウンリストで、[Create Bridge Domain] を選択します。
c) [Name] フィールドに、ブリッジ ドメインの名前を入力します。
d) [Type] コントロールで、[regular] を選択します。
e) [VRF] ドロップダウンリストで、テナント VRF を選択します。VRF がまだ存在しない場合は、[Create VRF] を選択し、VRF に名前を付けて、[Submit] をクリックします。
f) [Next]、[Next]、[Finish] の順にクリックして [Create Application EPG] に戻ります。
g) [Finish] をクリックします。

ステップ 18 前のステップで作成したネイティブ EPG を展開します。

ステップ 19 [Static Ports] を右クリックし、[Deploy Static EPG On PC, VPC, or Interface] をクリックして、次の操作を実行します。

a) [Path Type] コントロールで、[Virtual Port Channel] を選択します。
b) [Path]ドロップダウンリストから、vPC 用に作成されたポート チャネル ポリシーを選択します。
c) [Port Encap] ドロップダウンリストから [VLAN] を選択し、イーサネット VLAN の番号を入力します。
d) [Deployment Immediacy] コントロールで、[Immediate] を選択します。
e) [Mode] コントロールで、[Access（802.1P）] を選択します。
f) [Submit] をクリックします。

ステップ 20 [Application EPGs] を右クリックし、[Create Application EPG] を選択して、次の操作を実行します。

この EPG は、SAN ごとに 2 つの EPG のうちの 1 番目になります。

a) [Name] フィールドに、EPG の名前を入力します。
b) [Bridge Domain] ドロップダウンリストで、[Create Bridge Domain] を選択します。
c) [Name] フィールドに、ブリッジ ドメインの名前を入力します。
d) [Type] コントロールで、[fc] を選択します。
e) [VRF] ドロップダウンリストで、テナント VRF を選択します。VRF がまだ存在しない場合は、[Create VRF] を選択し、VRF に名前を付けて、[Submit] をクリックします。
f) [Next]、[Next]、[Finish] の順にクリックして [Create Application EPG] に戻ります。
g) [Finish] をクリックします。

ステップ 21 前の手順を繰り返して、2 番目のアプリケーション EPG を作成します。

この 2 番目の EPG は 2 番目の SAN に使用されます。

ステップ 22 2 つの SAN EPG のうちいずれか 1 つを展開し、[Fibre Channel（Paths）] を右クリックし、[Deploy Fibre Channel] を選択して、次の操作を実行します。

a) [Path Type] コントロールで、[Port] を選択します。
b) [Node] ドロップダウンリストで、スイッチ ペアの一方のリーフを選択します。
c) [Path] ドロップダウンリストで、VPC のイーサネット ポート番号を選択します。
d) [VSAN] テキスト ボックスで、「vsan-」で始まる VSAN 番号を入力します。

たとえば、VSAN 番号が 300 の場合は「vsan-300」と入力します。

e) [VSAN Mode] コントロールで、[Native] を選択します。
f) [Submit] をクリックします。

ステップ 23 2 つの SAN EPG のうちもう一方を展開し、前の手順を繰り返してスイッチ ペアのもう一方のリーフを選択します。

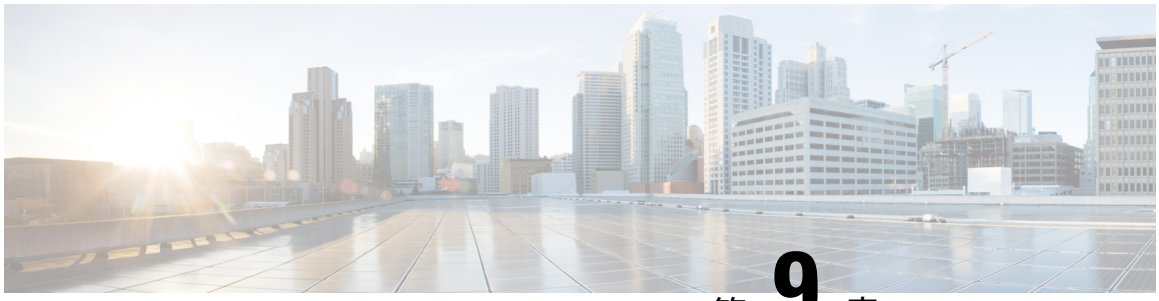
CLI を使用した vPC による SAN ブートの設定

この例では、次の項目がすでに設定されていると仮定しています。

- VLAN ドメイン
- テナント、アプリケーション プロファイル、アプリケーション EPG
- ポート チャネル テンプレート「Switch101-102_1-ports-49_PolGrp」

この例では、VSAN 200 はリーフ 101 上の物理イーサネット インターフェイス 1/49 にバインドされていて、VSAN 300 はリーフ 102 上の物理イーサネット インターフェイス 1/49 にバインドされています。2 つのインターフェイスは、仮想ポート チャネル Switch101-102_1-ports-49_PolGrp のメンバーです。

```
apic1(config-leaf)# show running-config
# Command: show running-config leaf 101
# Time: Sat Sep  1 12:51:23 2018
  leaf 101

    interface ethernet 1/49
      # channel-group Switch101-102_1-ports-49_PolGrp vpc
      switchport trunk native vlan 5 tenant newtenant application AP1 epg epgNative
      port-direction downlink
      exit

    # Port-Channel inherits configuration from "template port-channel
Switch101-102_1-ports-49_PolGrp"
    interface port-channel Switch101-102_1-ports-49_PolGrp
      exit

    interface vfc 1/49
      # Interface inherits configuration from "channel-group
Switch101-102_1-ports-49_PolGrp" applied to interface ethernet 1/49
      switchport vsan 200 tenant newtenant application AP1 epg epg200
      exit

apic1(config-leaf)# show running-config
# Command: show running-config leaf 102
# Time: Sat Sep  1 13:28:02 2018
  leaf 102
    interface ethernet 1/49
      # channel-group Switch101-102_1-ports-49_PolGrp vpc
      switchport trunk native vlan 1 tenant newtenant application AP1 epg epgNative
      port-direction downlink
      exit

    # Port-Channel inherits configuration from "template port-channel
Switch101-102_1-ports-49_PolGrp"
    interface port-channel Switch101-102_1-ports-49_PolGrp
      exit

    interface vfc 1/49
      # Interface inherits configuration from "channel-group
Switch101-102_1-ports-49_PolGrp" applied to interface ethernet 1/49
      switchport vsan 300 tenant newtenant application AP1 epg epg300
```



第 9 章

ファイバ チャネル NPV

この章は、次の内容で構成されています。

ファイバ チャネル接続の概要

Cisco ACI では、N ポート仮想化（NPV）モードを使用したリーフ スイッチでのファイバ チャネル（FC）接続がサポートされています。NPV により、スイッチにおいて、ローカル接続されたホスト ポート（N ポート）からの FC トラフィックをノード プロキシ（NP ポート）アップリンクに集約して、コア スイッチに送ることができます。

スイッチは、NPV を有効にした後は NPV モードになります。NPV モードはスイッチ全体に適用されます。NPV モードのスイッチに接続するエンド デバイスはそれぞれ、この機能を使用するために N ポートとしてログインする必要があります（ループ接続デバイスはサポートされていません）。（NPV モードの）エッジ スイッチから NPV コア スイッチへのすべてのリンクは、（E ポートではなく）NP ポートとして確立されます。このポートは、通常のスイッチ間リンクに使用されます。

（注） FC NPV アプリケーションにおける ACI リーフ スイッチの役割は、ローカル接続された SAN ホストとローカル接続されたコア スイッチ間の FC トラフィックのパスを提供することです。リーフ スイッチでは SAN ホスト間のローカル スイッチングは行われず、FC トラフィックはスパイン スイッチに転送されません。

FC NPV の利点

FC NPV では次の機能を提供します。

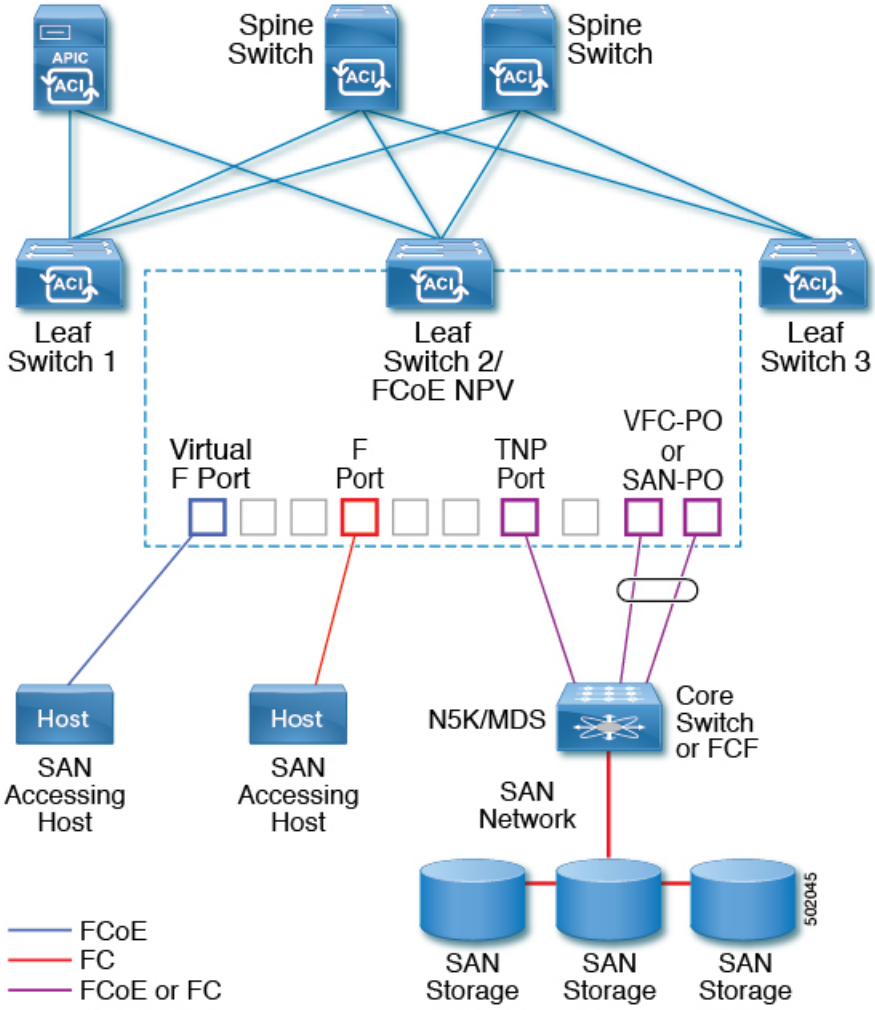
- ファブリックでドメイン ID を追加しなくても、ファブリックに接続するホスト数が増加します。NPV のコア スイッチのドメイン ID は、複数の NPV スイッチ間で共有されます。
- FC ホストと FCoE ホストは、ネイティブの FC インターフェイスを使用して SAN ファブリックに接続します。
- トラフィックの自動マッピングによるロード バランシング。NPV に接続しているサーバを新しく追加した場合に、トラフィックが現在のトラフィック負荷に基づいて、外部のアップリンク間で自動的に分散されます。
- トラフィックの静的マッピング。NPV に接続しているサーバを、外部のアップリンクに静的にマッピングすることができます。

FC NPV モード

ACI の Feature-set fcoe-npv は、最初に FCoE/FC 設定がプッシュされるときに、デフォルトで自動的に有効になります。

FC トポロジ

ACI ファブリック経由の FC トラフィックをサポートするさまざまな設定のトポロジを、次の図に示します。



- ACI リーフ スイッチ上のサーバー/ストレージ ホスト インターフェイスは、ネイティブの FC ポートか仮想 FC（FCoE）ポートのどちらかとして機能するように設定できます。
- FC コア スイッチへのアップリンク インターフェイスは、次のいずれかのポート タイプとして設定できます。
  - ネイティブ FC NP ポート
  - SAN-PO NP ポート
- FCF スイッチへのアップリンク インターフェイスは、次のいずれかのポート タイプとして設定できます。
  - 仮想（vFC）NP ポート
  - vFC-PO NP ポート

- N ポート ID 仮想化（NPIV）がサポートされており、デフォルトで有効になっています。そのため、単一のリンクを経由して N ポートに複数の N ポート ID またはファイバ チャネル ID（FCID）を割り当てることが可能です。
- コア スイッチへの NP ポートでは、トランキングを有効にすることができます。トランキングにより、ポートで複数の VSAN をサポートできます。トランク モードが有効になった NP ポートのことを、TNP ポートと呼びます。
- 複数の FC NP ポートを結合してコア スイッチへの SAN ポート チャネル（SAN-PO）とすることができます。トランキングは SAN ポート チャネルでサポートされます。
- FC F ポートでは 4/16/32 Gbps および自動速度設定がサポートされますが、ホスト インターフェイスでは 8Gbps はサポートされません。デフォルトの速度は「auto」です。
- FC NP ポートでは、4/8/16/32 Gbps および自動速度設定がサポートされます。デフォルトの速度は「auto」です。
- Flogi に続く複数の FDISC（ネスト NPIV）は、FC/FCoE ホストと FC/FCoE NP リンクによってサポートされます。
- FEX の背後にある FCoE ホストは、FCoE NP/アップリンクを介してサポートされます。
- APIC 4.1(1) リリース以降、FEX の背後にある FCoE ホストは、ファイバ チャネル NP/アップリンクを介してサポートされます。
- 1 つの FEX の背後にあるすべての FCoE ホストは、複数の vFC および vFC-PO アップリンク間、または単一のファイバ チャネル/SAN ポート チャネル アップリンクを通じてロード バランシングできます。
- SAN ブートは、FEX で FCoE アップリンク経由でサポートされます。
- APIC 4.1(1) リリース以降、SAN ブートは FC/SAN-PO アップリンクでもサポートされます。
- SAN ブートは、FEX を介して接続された FCoE ホストの vPC を介してサポートされます。

NPV トラフィック管理

通常は、すべてのトラフィックにおいて、すべての使用可能なアップリンクの使用を許可することをお勧めします。NPV トラフィック管理は、自動トラフィック エンジニアリングがネットワーク要件を満たさない場合にだけ使用してください。

自動アップリンク選択

NPV は、外部 NP アップリンク インターフェイスの自動選択をサポートしています。サーバ（ホスト）インターフェイスがアップになると、サーバ インターフェイスと同じ VSAN 内で利用可能な外部インターフェイスから、負荷が最も少ない外部インターフェイスが選択されます。

新しい外部インターフェイスが動作可能になっても、新たに利用可能になったアップリンクを含めるために既存の負荷は自動的に再分散されません。外部インターフェイスが新しいアップリンクを選択できるようになってから、サーバ インターフェイスが動作します。

トラフィック マップ

FC NPVは、トラフィック マップをサポートしています。トラフィック マップにより、サーバ（ホスト）インターフェイスがコア スイッチに接続するために使用可能な外部（NP アップリンク）インターフェイスを指定できます。

Note FC NPV トラフィック マップがサーバインターフェイスに設定されると、サーバインターフェイスはそのトラフィック マップ内の外部インターフェイスからのみ選択する必要があります。指定された外部インターフェイスがいずれも動作していない場合、サーバは非動作状態のままになります。

FC NPV トラフィック マップ機能を使用すると、次のようなメリットが得られます。

- 特定のサーバインターフェイス（またはサーバインターフェイスの範囲）に外部インターフェイスの事前設定された設定を割り当てることによって、トラフィック エンジニアリングが容易になります。
- インターフェイスの再初期化またはスイッチの再起動後に、サーバインターフェイスは同じ トラフィック パスを提供することで、常に同じ外部インターフェイス（または指定された外部インターフェイスのセットのいずれか）に接続するので、永続的なFC ID 機能の適切な動作が確保されます。

複数の NP リンクにまたがるサーバ ログインの破壊的自動ロード バランシング

FC NPVは、サーバ ログインの中断的ロードバランシングをサポートしています。中断的ロードバランシングが有効の場合、新しいNPアップリンクが動作すると、FC NPVによって、サーバインターフェイスがすべての利用可能なNPアップリンクにわたって再分配されます。サーバ インターフェイスを一方の NP アップリンクからの他方の NP アップリンクに移動するために、FC NPVはサーバインターフェイスを強制的に再初期化して、サーバがコア スイッチへのログインを新たに実行するようにします。

別のアップリンクに移されたサーバインターフェイスだけが再初期化されます。移されたサーバ インターフェイスごとにシステム メッセージが生成されます。

Note サーバ インターフェイスを再配布すると、接続されたエンド デバイスへのトラフィックが中断されます。既存のポート チャネルにメンバーを追加しても、中断的自動ロード バランシングはトリガーされません。

サーバ トラフィックの中断を避けるために、新しいNPアップリンクを追加してから、この機能をイネーブルし、サーバインターフェイスが再配布されてからこの機能を再度ディセーブルにしてください。

ディスラプティブ ロード バランシングがイネーブルでない場合、サーバ インターフェイスの一部またはすべてを手動で再初期化して、新しいNPアップリンク インターフェイスにサーバ トラフィックを分散することができます。

FC NPV トラフィック管理のガイドライン

FC NPV トラフィック管理を導入する際には、次の注意事項に従ってください。

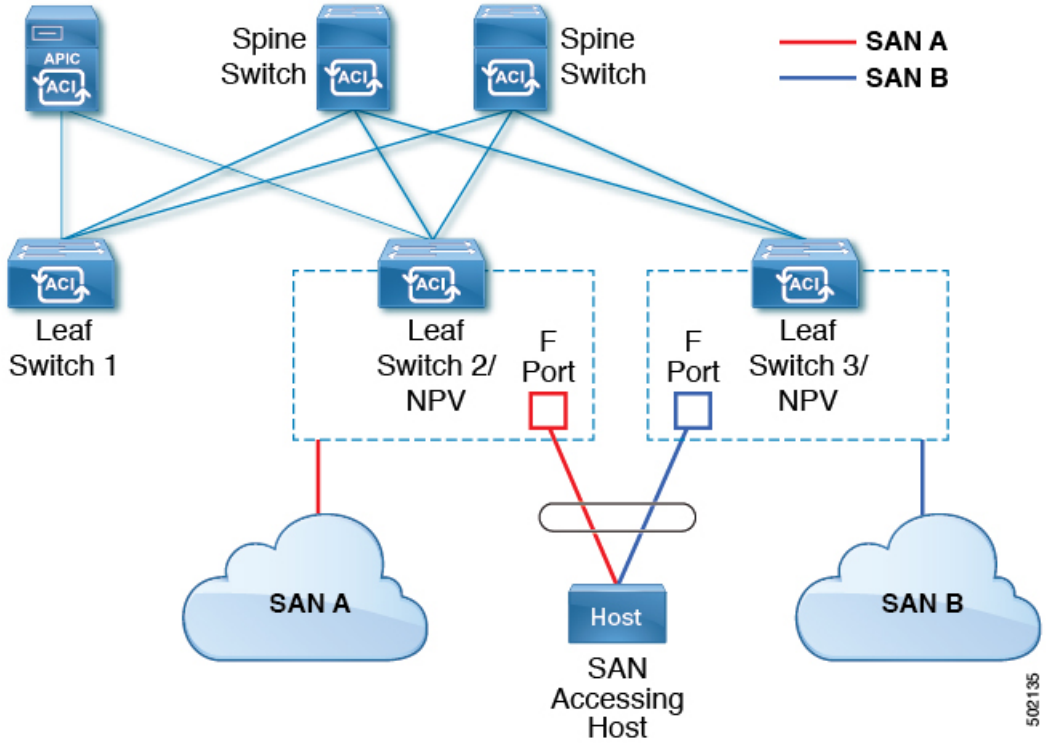
- NPV トラフィック管理は、自動トラフィック エンジニアリングがネットワーク要件を満たさない場合にだけ使用してください。
- すべてのサーバ インターフェイスにトラフィック マップを設定する必要はありません。FC NPV はデフォルトで自動トラフィック管理を使用します。
- NPアップリンク インターフェイスのセットを使用するように設定されたサーバインターフェイスは、利用可能なNPアップリンク インターフェイスがなくても、他の利用可能なNP アップリンク インターフェイスを使用できません。
- ディスラプティブ ロード バランシングがイネーブルになると、サーバ インターフェイスは 1 つの NP アップリンクから別の NP アップリンクに移動される場合があります。NP アップリンク インターフェイス間を移動する場合、FC NPV ではコア スイッチに再度ログインする必要があり、トラフィックの中断が生じます。
- サーバのセットを特定のコア スイッチにリンクするには、サーバインターフェイスをNP アップリンク インターフェイスのセット（すべてこのコア スイッチに接続されている）に関連付けてください。
- コア スイッチに永続的な FC ID を設定し、トラフィック マップ機能を使用してサーバ インターフェイスのトラフィックを NP アップリンクに送ります（すべてのアップリングが関連付けられたコア スイッチに接続しています）。
- トラフィック マップの固定を初めて設定する際は、最初のトラフィック マップを設定する前に、サーバのホスト ポートをシャットダウンする必要があります。
- トラフィックのマッピングを複数のアップリンクに設定していて、ホストへのログインに使用されるトラフィック マップを削除する場合は、先にホストをシャットダウンする必要があります。
- FEX の背後にある FCoE ホストのトラフィック マップを設定する場合、1 つのホストを複数の FCoE NP/アップリンク（VFC または VFC-PO）または単一のファイバ チャネル/SAN ポート チャネル NP/アップリンクにマッピングできます。

Note サーバが外部インターフェイスに静的にマッピングされている場合は、外部インターフェイスが何らかの理由でダウンする事態に備えて、サーバ トラフィックが再分配されません。

SAN A/B の分離

SAN A と SAN B の分離により、いずれかのファブリック コンポーネントが障害を起こしても SAN 接続が使用できることが保証されます。SAN A と SAN B の分離は、ファブリック全体で導入されている VSAN を分割することで、物理的または論理的に実現できます。

図 *31 : SAN A/B* の分離



SAN ポート チャネル

SAN ポート チャネルについて

- SAN ポート チャネルは、同じファイバ チャネル ノードに接続された一組の FC インターフェイスを結合して 1 つのリンクとして動作させる論理インターフェイスです。
- SAN ポート チャネルにより、帯域の利用率と可用性がサポートされます。

- Cisco ACI スイッチの SAN ポート チャネルは、FC コア スイッチに接続するため、そして VSAN のアップリンク間で最適な帯域幅利用と透過型のフェールオーバーを実現するために使用されます。

SAN ポート チャネルのガイドラインと制限事項

- Cisco ACI スイッチのアクティブ ポート チャネルの最大数（SAN ポート チャネルと VFC アップリンク/NP ポート チャネルの合計）は 7 です。追加で設定されたポート チャネルはすべて、既存のいずれかのアクティブ ポート チャネルをシャットダウンまたは削除するまで、 **errdisabled** 状態のままです。既存のアクティブ ポート チャネルをシャットダウンまたは削除してから、**errdisabled** のポート チャネルを shut/no shut してアップします。
- SAN ポート チャネルに結合できる FC インターフェイスの最大数は 16 個に制限されます。
- SAN ポート チャネルの Cisco ACI スイッチでのデフォルトのチャネル モードは**アクティブ**です。これは変更できません。
- SAN ポート チャネルがコア スイッチとして Cisco FC コア スイッチに接続されている場合は、アクティブなチャネル モードだけがサポートされます。Cisco FC コア スイッチでアクティブなチャネルモードを設定する必要があります。

SAN ポート チャネル モードについて

SAN ポート チャネルは、デフォルトではチャネル モードがアクティブの状態で設定されています。アクティブの場合、ピア ポートのチャネルグループ モードに関係なく、メンバー ポートはピア ポートとのポートチャネル プロトコルのネゴシエーションを開始します。チャネル グループで設定されているピア ポートがポートチャネル プロトコルをサポートしていない場合、またはネゴシエーション不可能を示すステータスを返す場合、ポート チャネルは無効になります。アクティブのポートチャネルモードでは、片側でポートチャネルメンバーのポートの有効化および無効化を明示的に行わなくても、自動回復が可能です。

ファイバチャネル N ポート仮想化のガイドラインと制限事項

ファイバ チャネル N ポート仮想化（NPV） を設定する場合、次の注意事項および制限事項に注意してください。

- ファイバ チャネル NP ポートはトランク モードをサポートしますが、ファイバ チャネル F ポートはサポートしません。
- トランク ファイバ チャネル ポートでは、最も高い VSAN により内部ログインが行われます。
- コア スイッチで次の機能を有効にする必要があります。

```
feature npiv
feature fport-channel-trunk
```

- 8G のアップリンク速度を使用する場合は、コア スイッチで IDLE フィル パターンを設定する必要があります。

（注） Cisco MDS スイッチでの IDLE フィル パターンの設定例を次に示します。

```
Switch(config)# int fc2/3
Switch(config)# switchport fill-pattern IDLE speed 8000
Switch(config)# show run int fc2/3

interface fc2/3
switchport speed 8000
switchport mode NP
switchport fill-pattern IDLE speed 8000
no shutdown
```

- Cisco APIC 4.1(1) リリース以降では、ファイバ チャネル NPV のサポートは Cisco N9K-C93180YC-FX スイッチに限定されます。
- ファイバ チャネル設定にはポート 1 ー 48 を使用できます。ポート 49 ー 54 をファイバ チャネル ポートにすることはできません。
- ポートをイーサネットからファイバチャネルに、またはその逆に変換する場合は、スイッチをリロードする必要があります。Currently, you can convert only one contiguous range of ports to Fibre Channel ports, and this range must be a multiple of 4, ending with a port number that is a multiple of 4.現時点で変換できるのは、ファイバチャネル ポートの連続した範囲のポートだけです。そしてこの範囲は 4 の倍数である必要があり、最後のポート番号は 4 の倍数になっている必要があります。たとえば、1 ～ 4、1 ～ 8、21 ～ 24 などです。
- Brocade ポート ブレード ファイバ チャネル 16 ～ 32 へのファイバ チャネル アップリンク（NP）接続は、Cisco N9K-93180YC-FX リーフ スイッチ ポートが 8G の速度で設定されている場合はサポートされません。
- 選択したポートの速度が SFP によってサポートされている必要があります。たとえば、32G の SFP は 8/16/32G をサポートするため、4G のポート速度には 8G または 16G の SFP が必要です。16G の SFP のサポートは 4/8/16G であるため、32G のポート速度には 32G の SFP が必要です。
- 速度の自動ネゴシエーションがサポートされています。デフォルトの速度は「auto」です。
- 40G およびブレークアウト ポートではファイバ チャネルを使用できません。
- FEX を FC ポートに直接接続することはできません。
- FEX HIF ポートを FC に変換することはできません。

ファイバチャネルNポート仮想化でサポートされるハードウェア

ファイバ チャネル N ポート仮想化（FC NPV）は、次のスイッチでサポートされます。

- N9K-C93108TC-FX
- N9K-C93180YC-FX

次のファイバ チャネル Small Form-Factor Pluggable（SFP）トランシーバはサポートされています。

- DS-SFP-FC8G-SW：2/4/8G（2G の FC NPV ポート速度はサポート外）
- DS-SFP-FC16G-SW：4/8/16G（FC NPV ポート速度が 32G の場合は非互換）
- DS-SFP-FC32G-SW：8/16/32G（FC NPVポート速度が 4G の場合は非互換）

サポートされている NPIV コア スイッチは、Cisco Nexus 5000 シリーズ、Nexus 6000 シリーズ、Nexus 7000 シリーズ（FCoE）、および Cisco MDS 9000 シリーズ マルチレイヤ スイッチです。

ファイバ チャネル N ポート仮想化の相互運用性

次の表に、Cisco Application Policy Infrastructure Controller（APIC）のファイバ チャネル N ポート仮想化（FC NPV）機能の相互運用性がテストされたサード パーティ製品を示します。

表 *4: FC NPV* でサポートされるサード パーティ製品

サード パーティ スイッチ ベンダー	Brocade
サード パーティ ハードウェア モデル	DS-6620B
サード パーティ ソフトウェア リリース	8.2.1a
Cisco NX-OS リリース	14.1(1) 以降
Cisco Nexus 9000 モデル	N9K-C93180YC-FX
相互運用性モード	NA（NPV）
Cisco SFP モジュール	DS-SFP-FC32G-SW
サード パーティ **SFP** モジュール	Brocade-32G

ファイバ チャネル NPV GUI の設定

GUI を使用したネイティブ ファイバ チャネル ポート プロファイルの設定

この手順では、ファイバチャネルのホスト（サーバなど）に接続するための一連のネイティブ ファイバ チャネル（FC）F ポートの設定を行います。

設定を簡単に行うため、この手順では **[Configure Interface, PC, and vPC]** ウィザードを使用します。

手順

ステップ 1 APIC メニュー バーで、**[Fabric] > [Access Policies] > [Quickstart]** に移動し、*[Configure an interface, PC, and vPC]* をクリックします。

ステップ 2 [Configured Switch Interfaces] ツールバーで、[+] をクリックしてスイッチ プロファイルを作成します。次のアクションを実行します。

このスイッチ プロファイルでは、サーバ ホスト ポートを設定します。別のスイッチ プロファイルでは、アップリンク ポートを設定します。

a) [Switches] ドロップダウンリストで、NPV リーフ スイッチを選択します。

この操作によって、自動的にリーフ スイッチ プロファイルが作成されます。[Switch Profile Name] テキスト ボックスで、リーフ スイッチ プロファイルの名前を受け入れるか変更できます。

b) さらにインターフェイス設定を開くには、ポートで大きな緑色の [+] をクリックします。

c) [Interface Type] で、[FC] を選択して、ファイバ チャネル ホスト インターフェイス ポート（F ポート）を指定します。

d) [Interfaces] で、FC ポートのポート範囲を入力します。

FC ポートに変換できるポートの連続範囲は 1 つだけです。この範囲は 4 の倍数にする必要があり、4 の倍数のポート番号で終わる必要があります（たとえば、1 ～ 4、1 ～ 8、21 ～ 24 は有効な範囲です）。

この操作によってインターフェイス セレクタ ポリシーが作成されます。[Interface Selector Name] テキスト ボックスで、ポリシーの名前を受け入れるか変更できます。

（注） イーサネットから FC へのポートの変換には、スイッチのリロードが必要です。インターフェイス ポリシーを適用すると、スイッチをリロードするよう求める通知アラームが GUI に表示されます。スイッチのリロード中はスイッチへの通信が中断され、スイッチにアクセスしようとするとタイムアウトになります。

e) [Policy Group Name] ドロップダウンリストで、[Create FC Interface Policy Group] を選択します。

f) [Create FC Interface Policy Group] ダイアログボックスで、[Name] フィールドに名前を入力します。

g) [Fibre Channel Interface Policy] ドロップダウンリストで、[Create Fibre Channel Interface Policy] を選択します。

h) [Create Fibre Channel Interface Policy] ダイアログボックスで、[Name] フィールドに名前を入力し、次の設定を行います。

フィールド	設定
ポート モード	ホスト インターフェイスの場合、[F] を選択します。
Trunk Mode	ホストインターフェイスの場合、[trunk-off] を選択します。
速度	[auto]（デフォルト）を選択します。
[自動最大速度 (Auto Max Speed)]	Auto Max Speed設定は、速度が **auto** の場合にのみ適用されます。 [Auto Max Speed]は、速度が自動モードのときに最大速度を制限します。
Receive Buffer Credit	[64] を選択します。

i) [Submit]をクリックして、ファイバチャネルインターフェイス ポリシーを保存し、[Create FC Interface PolicyGroup] ダイアログボックスに戻ります。

j) [Attached Entity Profile] ドロップダウンリストで、[Create Attachable Access Entity Profile] を選択します。

アタッチ可能なエンティティ プロファイルのオプションでは、リーフ アクセス ポート ポリシーを展開するインターフェイスを指定します。

k) [Name] フィールドに、アタッチ可能なエンティティのポリシーの名前を入力します。

l) [Domains（VMM, Physical, or External）To Be Associated To Interfaces] ツールバーで、[+] をクリックしてドメイン プロファイルを追加します。

m) [Domain Profile] ドロップダウンリストで、[Create Fibre Channel Domain] を選択します。

n) [Name] フィールドに、ファイバ チャネル ドメインの名前を入力します。

o) [VSAN Pool] ドロップダウンリストで、[Create VSAN Pool] を選択します。

p) [Name] フィールドに、VSAN プールの名前を入力します。

q) [Encap Blocks] ツールバーで、[+] をクリックして VSAN 範囲を追加します。

r) [Create VSAN Ranges] ダイアログボックスで、[From] および [To] の VSAN 番号を入力します。

s) [Allocation Mode] で、[Static Allocation] を選択し、[OK] をクリックします。

t) [Create VSAN Ranges] ダイアログボックスで、[Submit] をクリックします。

u) [Create Fibre Channel Domain] ダイアログボックスで、[Submit] をクリックします。

（注） ファイバ チャネル ドメインでは、FCoE ではなくネイティブ FC ポートを使用する場合、VLAN プールや VSAN 属性を設定する必要はありません。

v) [Create Attachable Access Entity Profile] ダイアログボックスで、[Update] をクリックしてファイバ チャネル ドメイン プロファイルを選択し、[Submit] をクリックします。

w) [Create FC Policy Group] ダイアログボックスで、[Submit] をクリックします。

x) **[Configure Interface、PC、and vPC]** ダイアログ ボックスで、**[Save]** をクリックして、サーバー ホスト ポートのこのスイッチ プロファイルを保存します。

（注） イーサネットから FC へのポートの変換には、スイッチのリロードが必要です。インターフェイス ポリシーを適用すると、スイッチをリロードするよう求める通知アラームが GUI に表示されます。スイッチのリロード中はスイッチへの通信が中断され、スイッチにアクセスしようとするとタイムアウトになります。

（注） たとえば、アップリンクをダウンリンクとして再設定し、スイッチをリロードするなど、スイッチのポート プロファイルを変更すると、スイッチが Cisco APIC から設定を取得するまで、スイッチとの通信が中断されます。

[Fabric] > [Access Policies] > [Switches] > [Leaf Switches] > [Profiles] > [<名前>] で、[Leaf Profiles] 作業ペインの [Associated Interface Selector Profiles] リストにファイバ チャネル ポート プロファイルが表示されます。

次のタスク

- ファイバ チャネル アップリンク接続プロファイルを設定します。
- テナント内のサーバ ポートとアップリンク ポートを展開し、ファイバ チャネルのコア スイッチに接続します。

GUI を使用したネイティブ FC ポート チャネル プロファイルの設定

この手順では、ファイバ チャネルのコア スイッチへのアップリンク接続に使用するネイティブ ファイバ チャネル ポート チャネル（FC PC）プロファイルを設定します。

（注） この手順は、**[Configure Interface, PC, and vPC]** ウィザードを使用して実行することもできます。

始める前に

アタッチ可能なエンティティ プロファイルを含む、アップリンク接続を設定します。

手順

ステップ 1 [Fabric] > [Access Policies] > [Interfaces] > [Leaf Interfaces] > [Profiles] の順に展開します。

ステップ 2 [Profiles] を右クリックし、[Create Leaf Interface Profile] をクリックします。

ステップ 3 [Create Leaf Interface Profile] ダイアログボックスで、次の手順を実行します。

a) [Name] フィールドに、リーフ インターフェイス プロファイルの名前を入力します。

b) [Interface Selectors] ツールバーで、[+] をクリックして [Create Access Port Selector] ダイアログボックスを表示します。

c) [Name] フィールドに、ポート セレクタの名前を入力します。

d) [Interface IDs] フィールドで、FC PC ポートのポート範囲を入力します。

ポート チャネルには最大 16 個のポートを持たせることができます。

FC ポートに変換できるポートの連続範囲は 1 つだけです。この範囲は 4 の倍数にする必要があり、4 の倍数のポート番号で終わる必要があります（たとえば、1 ～ 4、1 ～ 8、21 ～ 24 は有効な範囲です）。

（注） イーサネットから FC へのポートの変換には、スイッチのリロードが必要です。インターフェイス ポリシーを適用すると、スイッチを手動でリロードするよう求める通知アラームが GUI に表示されます。スイッチのリロード中はスイッチへの通信が中断され、スイッチにアクセスしようとするとタイムアウトになります。

e) [Interface Policy Group] ドロップダウンリストで、[Create FC PC Interface Policy Group] を選択します。

f) [Name] フィールドに、FC PC インターフェイス ポリシー グループの名前を入力します。

g) [Fibre Channel Interface Policy] ドロップダウンリストで、[Create Fibre Channel Interface Policy] を選択します。

h) [Name] フィールドに、FC PC インターフェイス ポリシーの名前を入力します。

i) [Create Interface FC Policy] ダイアログボックスで、[Name] フィールドに名前を入力し、次の設定を行います。

フィールド	設定
ポート モード	アップリンク インターフェイスの場合、[NP] を選択します。
Trunk Mode	アップリンク インターフェイスの場合、[trunk-on] を選択します。

j) [Submit] をクリックして、FC PC インターフェイス ポリシーを保存し、[Create FC PC Interface Policy Group] ダイアログボックスに戻ります。

k) **Port Channel Policy** ドロップで、**Create Port Channel Policy** を選択します。

l) [Name] フィールドに、ポート チャネル ポリシーの名前を入力します。

このメニューにある他の設定は無視できます。

m) [Submit] をクリックして、ポート チャネル ポリシーを保存し、[Create FC PC Interface Policy Group] ダイアログボックスに戻ります。
n) [Attached Entity Profile] ドロップダウンリストで、既存のアタッチ可能なエンティティ プロファイルを選択します。
o) [Submit] をクリックして [Create Access Port Selector] ダイアログボックスに戻ります。
p) [OK] をクリックして [Create Leaf Interface Profile] ダイアログボックスに戻ります。
q) [OK] をクリックして [Leaf Interfaces - Profiles] 作業ペインに戻ります。

ステップ 4 [Fabric] > [Access Policies] > [Switches] > [Leaf Switches] > [Profiles] の順に展開します。

ステップ 5 作成したリーフ スイッチ プロファイルを右クリックし、[Create Interface Profile] をクリックします。

ステップ 6 [Create Interface Profile] ダイアログボックスで、次の手順を実行します。

a) [Interface Select Profile] ドロップダウンリストで、ポート チャネル用に作成したリーフ インターフェイス プロファイルを選択します。
b) [Submit] をクリックして [Leaf Interfaces - Profiles] 作業ペインに戻ります。

（注） イーサネットから FC へのポートの変換には、スイッチのリロードが必要です。インターフェイス ポリシーを適用すると、スイッチをリロードするよう求める通知アラームが GUI に表示されます。スイッチのリロード中はスイッチへの通信が中断され、スイッチにアクセスしようとするとタイムアウトになります。

[Fabric] > [Access Policies] > [Switches] > [Leaf Switches] > [Profiles] > [<名前>] で、作業ペインの [Associated Interface Selector Profiles] リストに FC ポート チャネル プロファイルが表示されます。

次のタスク

テナント内のサーバ ポートとアップリンク ポートを展開し、ファイバ チャネルのコア スイッチに接続します。

ファイバ チャネル ポートの展開

この手順では、ファイバ チャネル サーバ ホスト ポートとアップリンク ポートをアクティブにします。

始める前に

- ファイバ チャネル（FC）サーバ ホスト ポート プロファイル（F ポート）を設定します。
- FC アップリンク ポート プロファイル（NP または TNP ポート）を設定します。
- 関連付けられている 2 つのインターフェイス セレクタ プロファイル（1 つはホスト ポート用、1 つはアップリンク ポート用）を含むリーフ スイッチ プロファイルを設定します。

手順

ステップ 1 [Tenants] > [<テナント名>] > [Application Profiles] の順に展開します。

テナントが存在しない場合は、テナントを作成する必要があります。

ステップ 2 [Application Profiles] を右クリックし、[Create Application Profile] をクリックして、次の操作を実行します。

a) [Name] フィールドに、アプリケーション プロファイルの名前を入力します。
b) [Submit] をクリックします。

ステップ 3 [Tenants] > [<テナント名>] > [Application Profiles] > [<名前>] > [Application EPGs] の順に展開します。

ステップ 4 [Application EPGs] を右クリックし、[Create Application EPG] をクリックして、次の操作を実行します。

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

a) [Name] フィールドに、アプリケーション EPG の名前を入力します。
b) 次を設定します。

フィールド	設定
Intra EPG Isolation	[Unenforced] を選択します。
Preferred Group Member	[Exclude] を選択します。
カプセル化のフラッディング	[Disabled] を選択します。

c) [Bridge Domain] ドロップダウンリストで、[Create Bridge Domain] を選択します。
d) [Name] フィールドに、ブリッジ ドメインの名前を入力します。
e) [Type] で、[fc] を選択してファイバ チャネル ブリッジ ドメインを指定します。
f) [VRF] ドロップダウンリストで、[Create VRF] を選択します。
g) [Name] フィールドに、VRF の名前を入力します。
h) [Submit] をクリックして [Create Bridge Domain] ダイアログボックスに戻ります。
i) [Next]、[Next]、[Finish] の順にクリックして [Create Application EPG] ダイアログボックスに戻ります。
j) [Finish] をクリックします。

ステップ 6 [Tenants] > [<テナント名>] > [Application Profiles] > [<名前>] > [Application EPGs] > [<名前>] > [Domains（VMs and Bare-Metals）] の順に展開します。

ステップ 7 [Domains（VMs and Bare-Metals）] を右クリックし、[Add Fibre Channel Domain Association] をクリックして、次の操作を実行します。

a) [Fibre Channel Domain Profile] ドロップダウンリストで、ホスト ポートの設定時に作成したファイバ チャネル ドメインを選択します。
b) [Submit] をクリックします。

ステップ 8 [Tenants] > [<テナント名>] > [Application Profiles] > [<名前>] > [Application EPGs] > [<名前>] > [Fibre Channel（Paths）] の順に展開し、次の操作を実行します。

このステップでは、サーバ ホスト ポートを展開します。

a) [Fibre Channel（Paths）] を右クリックし、[Deploy Fibre Channel] をクリックします。
b) [Path Type] コントロールで、[Port] をクリックします。
c) [Node] ドロップダウンリストで、リーフ スイッチを選択します。
d) [Path] ドロップダウンリストで、サーバ ホスト ポートとして設定されているリーフ スイッチ ポートを選択します。
e) [VSAN] フィールドに、ポートの VSAN を入力します。
f) [VSAN Mode] コントロールで、[Native] をクリックします。
g) [Type] が fcoe であることを確認します。
h) （オプション）トラフィック マップを必要とする場合は、[Pinning Label] ドロップダウンリストを使用します。

（注） 複数のアップリンク ポートが使用可能で、ホスト ポートにおいて常にその FLOGI を特定のアップリンクに送るようにする場合は、固定プロファイル（トラフィック マップ）を作成してホスト ポートをアップリンク ポートに関連付けることができます。そのようにしない場合は、使用可能なアップリンク ポート間でホストがロードバランスされます。

i) [Submit] をクリックします。
j) ファイバ チャネル ポート ホストごとに**ステップ a** から繰り返します。

ステップ 9 [Tenants] > [<テナント名>] > [Application Profiles] > [<名前>] > [Application EPGs] > [<名前>] > [Fibre Channel（Paths）] の順に展開し、次の操作を実行します。

このステップでは、アップリンク ポート チャネルを展開します。

a) [Fibre Channel（Paths）] を右クリックし、[Deploy Fibre Channel] をクリックします。
b) [Path Type] コントロールで、[Direct Port Channel] をクリックします。
c) [Port Type] ドロップダウンリストで、アップリンク ポートチャネルを選択します。
d) [VSAN] フィールドに、ポートのデフォルトの VSAN を入力します。
e) [VSAN Mode] コントロールで、ポートの VSAN の場合は [Native] をクリックし、トランクの VSAN の場合は [Regular] をクリックします。
f) [Type] が fcoe であることを確認します。
g) [Submit] をクリックします。`
h) ファイバ チャネル アップリンク ポートまたはポート チャネルごとに**ステップ a** から繰り返します。

ファイバ チャネル ポートのトラフィック マップの設定

複数のアップリンク ポートが使用可能なアプリケーションでは、デフォルトで、サーバ トラフィックが使用可能なアップリンク ポート間でロードバランスされます。場合によっては、1

つ以上の特定のアップリンク ポートまたはポート チャネルにログイン要求（FLOGI）を送信するようサーバを設定する必要があります。このような場合、固定プロファイル（トラフィック マップ）を作成して、それらのアップリンク ポートまたはポート チャネルにサーバ ポートを関連付けることができます。

この手順では、1 つ以上のサーバ ポートと 1 つ以上のアップリンク ポートまたはポート チャネルがすでに設定済みであると仮定します。サーバ ポートがすでに設定済みであるため、最初に、アップリンクにマッピングするすべてのサーバ ポートをシャットダウン（無効化）する必要があります。トラフィック マップを設定した後で、再度ポートを有効にします。

始める前に

この手順では、次の項目がすでに設定済みであることを前提としています。

- サーバ ポート（F ポート）およびアップリンク ポートまたはポート チャネル（NP ポート）
- テナント（アプリケーション プロファイルおよびアプリケーション EPG を含む）

（注） 固定プロファイル（トラフィック マップ）を作成する前に、アップリンクにマッピングするサーバ ポートをシャットダウンする必要があります。

手順

ステップ 1 [Fabric] > [Inventory] > [Pod <n>] > [Leaf <n>] > [Interfaces] > [FC Interfaces] 作業ウィンドウを選択し、アップリンクにマッピングするサーバ インターフェイス ポートを選択して無効にします。

ステップ 2 [Tenants] > [<テナント名>] > [Application Profiles] > [<アプリケーション プロファイル名>] > [Application EPGs] > [<EPG 名>] > [Fibre Channel（Paths）] の順に展開し、次の操作を実行します。

a) [Fibre Channel（Paths）] を右クリックし、[Deploy Fibre Channel] をクリックします。
b) [Path Type] コントロールで、[Port] をクリックします。
c) [Node] ドロップダウンリストで、リーフ スイッチを選択します。
d) [Path] ドロップダウンリストで、特定のアップリンク ポートにマッピングするサーバ ポートを選択します。
e) [VSAN] フィールドに、ポートのデフォルトの VSAN を入力します。
f) [VSAN Mode] コントロールで、[Native] をクリックします。
g) [Type] が fcoe であることを確認します。
h) [Pinning Label] ドロップダウンリストで、[Create Pinning Profile] を選択します。
i) [Name] フィールドに、トラフィック マップの名前を入力します。
j) [Path Type] コントロールで、[Port] をクリックして単一の NP アップリンク ポートに接続するか、[Direct Port Channel] をクリックして FC ポート チャネルに接続します。

パスの種類で [Port] を選択した場合は、表示される [Node] ドロップダウンリストでリーフ スイッチを選択する必要もあります。

パスの種類で [Direct Port Channel] を選択した場合は、インターフェイス ポリシー グループで定義した FC PC を選択する必要もあります。

k) [Path] ドロップダウンリストで、サーバ ポートをマッピングするアップリンク ポートまたはポート チャネルを選択します。

l) [Submit] をクリックして [Deploy Fibre Channel] ダイアログボックスに戻ります。

m) [Submit] をクリックします。`

ステップ 3 [Fabric] > [Inventory] > [Pod <n>] > [Leaf <n>] > [Interfaces] > [FC Interfaces] 作業ウィンドウを選択し、アップリンクにマッピングするサーバ インターフェイス ポートを選択して再び有効にします。

ファイバ チャネル NPV NX-OS スタイル CLI の設定

CLI を使用したファイバ チャネル インターフェイスの設定

NPV 対応リーフ スイッチでは、ユニバーサル ポートをファイバ チャネル（FC）ポートに変換することができます。FC ポートは F ポートまたは NP ポートのどちらかにすることができ、NP ポートではポート チャネルを形成できます。

手順

ステップ 1 ポートの範囲をイーサネットからファイバ チャネルに変換します。

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# slot 1
apic1(config-leaf-slot)# port 1 12 type fc
```

この例では、リーフ 101 のポート 1/1-12 をファイバ チャネル ポートに変換します。**[no]** 形式の **port type fc** コマンドで、ポートをファイバ チャネルから再びイーサネットに変換します。

（注） ポートの変換はリーフ スイッチのリブート後にのみ行われます。

現在のところ、FC ポートに変換できるポートの連続範囲は 1 つだけです。この範囲は 4 の倍数にする必要があり、4 の倍数のポート番号で終わる必要があります（例：1 ～ 4、1 ～ 8、21 ～ 24）。

ステップ 2 すべてのファイバ チャネル インターフェイスを設定します。

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# interface fc 1/1
apic1(config-leaf-fc-if)# switchport mode [f | np]
apic1(config-leaf-fc-if)# switchport rxbbcredit <16-64>
apic1(config-leaf-fc-if)# switchport speed [16G | 32G | 4G | 8G | auto | unknown]
apic1(config-leaf-fc-if)# switchport trunk-mode [ auto | trunk-off | trunk-on | un-init]
apic1(config-leaf-fc-if)# switchport [trunk allowed] vsan <1-4093> tenant <name>  \
                              application <name> epg <name>
```

（注） FC ホスト インターフェイス（F ポート）は、8Gbps の速度設定をサポートしていません。

FC インターフェイスは、アクセス モードまたはトランク モードで設定できます。FC ポートをアクセス モードに設定するには、次のコマンド形式を使用します。

例：

```
apic1(config-leaf-fc-if)# switchport vsan 2 tenant t1 application a1 epg e1
```

FC ポートをトランク モードに設定するには、次のコマンド形式を使用します。

例：

```
apic1(config-leaf-fc-if)# switchport trunk allowed vsan 4 tenant t1 application a1 epg
e1
```

FC ポート チャネルを設定するには、FC ポート インターフェイス テンプレートを設定し、FC ポートチャネルのメンバーになる FC インターフェイスに適用します。

ポート チャネルには最大 16 個のメンバーを持たせることができます。

例：

```
apic1(config)# template fc-port-channel my-fc-pc
apic1(config-fc-po-ch-if)# lacp max-links 4
apic1(config-fc-po-ch-if)# lacp min-links 1
apic1(config-fc-po-ch-if)# vsan-domain member dom1
apic1(config-fc-po-ch-if)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface fc 1/1-2
apic1(config-leaf-fc-if)# fc-channel-group my-fc-pc
apic1(config-leaf-fc-if)# exit
apic1(config-leaf)# interface fc-port-channel my-fc-pc
apic1(config-leaf-fc-pc)# switchport mode [f | np]
apic1(config-leaf-fc-pc)# switchport rxbbcredit <16-64>
apic1(config-leaf-fc-pc)# switchport speed [16G | 32G | 4G | 8G | auto | unknown]
apic1(config-leaf-fc-pc)# switchport trunkmode [ auto | trunk-off | trunk-on | un-init]
```

CLI を使用したファイバ チャネル NPV ポリシーの設定

始める前に

NPVアプリケーションで使用するリーフ スイッチ ポートをファイバチャネル（FC）ポートに変換した。

手順

ステップ 1 ファイバ チャネル F ポート ポリシー グループのテンプレートを作成します。

例：

```
apic1(config)# template fc-policy-group my-fc-policy-group-f-ports
apic1(config-fc-pol-grp-if)# vsan-domain member dom1
apic1(config-fc-pol-grp-if)# switchport mode f
apic1(config-fc-pol-grp-if)# switchport trunk-mode trunk-off
```

速度など、他のスイッチ ポート設定を行うことができます。

ステップ 2 ファイバ チャネル NP ポート ポリシー グループのテンプレートを作成します。

例：

```
apic1(config)# template fc-policy-group my-fc-policy-group-np-ports
apic1(config-fc-pol-grp-if)# vsan-domain member dom1
apic1(config-fc-pol-grp-if)# switchport mode np
apic1(config-fc-pol-grp-if)# switchport trunk-mode trunk-on
```

速度など、他のスイッチ ポート設定を行うことができます。

ステップ 3 ファブリック全体のファイバ チャネル ポリシーを作成します。

例：

```
apic1(config)# template fc-fabric-policy my-fabric-fc-policy
apic1(config-fc-fabric-policy)# fctimer e-d-tov 1000
apic1(config-fc-fabric-policy)# fctimer r-a-tov 5000
apic1(config-fc-fabric-policy)# fcoe fcmap 0E:FC:01
```

ステップ 4 ファイバ チャネル ポート チャネル ポリシーを作成します。

例：

```
apic1(config)# template fc-port-channel my-fc-pc
apic1(config-fc-po-ch-if)# lacp max-links 4
apic1(config-fc-po-ch-if)# lacp min-links 1
apic1(config-fc-po-ch-if)# vsan-domain member dom1
```

ステップ 5 リーフ全体のファイバ チャネル ポリシー グループを作成します。

例：

```
apic1(config)# template fc-leaf-policy my-fc-leaf-policy
apic1(config-fc-leaf-policy)# npv auto-load-balance disruptive
apic1(config-fc-leaf-policy)# fcoe fka-adv-period 10
```

（注） ここに示すポリシー コマンドは単なる例であり、必須の設定ではありません。

ステップ 6 リーフ ポリシー グループを作成します。

```
apic1(config)# template leaf-policy-group lpg1
apic1(config-leaf-policy-group)# inherit fc-fabric-policy my-fabric-fc-policy
apic1(config-leaf-policy-group)# inherit fc-leaf-policy my-fc-leaf-policy
```

FC 関連のポリシーを継承することによって、リーフ ポリシー グループが作成されます。

ステップ 7 リーフ プロファイルを作成し、リーフポリシーグループをリーフグループに適用します。

例：

```
apic1(config)# leaf-profile my-leaf-profile
apic1(config-leaf-profile)# leaf-group my-leaf-group
apic1(config-leaf-group)# leaf 101
apic1(config-leaf-group)# leaf-policy-group lpg1
```

この例では、リーフ ポリシー グループ lpg1 にグループ化された、ファブリック全体の FC ポリシーとリーフ全体の FC ポリシーを、リーフ 101 に適用します。

ステップ 8 リーフ インターフェイス プロファイルを作成し、fc ポリシーグループを一組の FC インターフェイスに適用します。

例：

```
apic1(config)# leaf-interface-profile my-leaf-interface-profile
apic1(config-leaf-if-profile)# leaf-interface-group my-leaf-interface-group
apic1(config-leaf-if-group)# fc-policy-group my-fc-policy-group-f-ports
apic1(config-leaf-if-group)# interface fc 1/1-10
```

CLI を使用した NPV トラフィック マップの設定

この手順では、FC/FCoE サーバ（ホスト）インターフェイスから NP モードに設定された FC/FCoE 外部（アップリンク）インターフェイスに送信されるトラフィックをマッピングします。

始める前に

すべてのサーバ インターフェイスが F ポートである必要があり、すべてのアップリンク インターフェイスが NP ポートである必要があります。

手順

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# npv traffic-map server-interface  \
    { vfc <slot/port> | vfc-po <po-name> |fc <slot/port> } \
    label <name> tenant <tn> app <ap> epg <ep>
apic1(config-leaf)# npv traffic-map external-interface \
    { vfc <slot/port> | vfc-po <po-name> |fc <slot/port> } \
    tenant <tn> label <name>
```

例：

```
apic1(config)# leaf 101
apic1(config-leaf)# npv traffic-map server-interface vfc 1/1 label serv1 tenant t1 app
ap1 epg epg1
apic1(config-leaf)# npv traffic-map external-interface vfc-po my-fc-pc tenant t1 label
ext1
```

ファイバ チャネル NPV REST API の設定

REST API を使用した FC 接続の設定

FC が有効なインターフェイスと Epg REST API を使用して、FC プロトコルを使用してこれらのインターフェイスへのアクセスを設定することができます。

手順

ステップ 1 VSAN プールを作成するには、次の例などと XML post を送信します。この例では、VSAN プール myVsanPool1 を作成し、vsan-50 から vsan-60 までを含むように VSAN の範囲を指定します。

例：

```
https://apic-ip-address/api/mo/uni/infra/vsanns-[myVsanPool1]-static.xml

<fvnsVsanInstP allocMode="static" name="myVsanPool1">
  <fvnsVsanEncapBlk from="vsan-50" name="encap" to="vsan-60"/>
</fvnsVsanInstP>
```

ステップ 2 ファイバ チャネル ドメインを作成するには、次の例のように XML で post を送信します。この例では、ファイバチャネル ドメイン（VSAN ドメイン）myFcDomain1 を作成し、VSAN プール myVsanPool1 に関連付けます。

例：

```
https://apic-ip-address/api/mo/uni/fc-myFcDomain1.xml

<fcDomP name="myFcDomain1">
  <fcRsVsanNs tDn="uni/infra/vsanns-[myVsanPool1]-static"/>
</fcDomP>
```

ステップ 3 FC ポートのアタッチ エンティティ ポリシー（AEP）を作成するには、次の例のように XML で POST を送信します。この例では、AEP myFcAEP1 を作成し、ファイバ チャネル ドメイン myFcDomain1 に関連付けます。

例：

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
<infraInfra>
    <infraAttEntityP name="myFcAEP1">
        <infraRsDomP tDn="uni/fc-myFcDomain1"/>
    </infraAttEntityP>
</infraInfra>
</polUni>
```

ステップ 4 サーバ ホスト ポートの FC インターフェイス ポリシーとポリシー グループを作成するには、XML で POST を送信します。この例は次の要求を実行します。

- サーバ ホスト ポートの FC インターフェイス ポリシー myFcHostIfPolicy1 を作成します。これらは、トランキングのない F ポートです。
- FC ホスト インターフェイス ポリシー myFcHostIfPolicy1 を含む FC インターフェイス ポリシー グループ myFcHostPortGroup1 を作成します。
- ポリシー グループを FC インターフェイス ポリシーに関連付けて、これらのポートを FC ポートに変換します。
- ホスト ポート プロファイル myFcHostPortProfile を作成します。
- ポートを 5 ～ 8 の範囲で指定するポート セレクタ myFcHostSelector を作成します。
- リーフ ノード 104 を指定するノード セレクタ myFcNode1 を作成します。
- リーフ ノード 104 を指定するノード セレクタ myLeafSelector を作成します。
- ホスト ポートをリーフ ノードに関連付けます。

例：

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
    <infraInfra>
       <fcIfPol name="myFcHostIfPolicy1" portMode="f" trunkMode="trunk-off" speed="auto"/>

        <infraFuncP>
            <infraFcAccPortGrp name="myFcHostPortGroup1">
                <infraRsFcL2IfPol tnFcIfPolName="myFcHostIfPolicy1" />
            </infraFcAccPortGrp>
        </infraFuncP>
        <infraAccPortP name="myFcHostPortProfile">
```

```
            <infraHPortS name="myFcHostSelector" type="range">
                <infraPortBlk name="myHostPorts" fromCard="1" toCard="1" fromPort="1"
toPort="8" />
                <infraRsAccBaseGrp tDn="uni/infra/funcprof/fcaccportgrp-myFcHostPortGroup1"
 />
            </infraHPortS>
        </infraAccPortP>
        <infraNodeP name="myFcNode1">
            <infraLeafS name="myLeafSelector" type="range">
                <infraNodeBlk name="myLeaf104" from_="104" to_="104" />
            </infraLeafS>
            <infraRsAccPortP tDn="uni/infra/accportprof-myHostPorts" />
        </infraNodeP>
    </infraInfra>
</polUni>
```

（注） この設定を適用する場合は、ポートを FC ポートとしてアップするためにスイッチのリロードが必要になります。

現在のみ FC ポートに変換できるポートの 1 つの連続した範囲と、この範囲にする必要がありますが 4 の倍数で終わるポート番号 4 の倍数ことです。たとえば、1 ～ 4、1 ～ 8、21 ～ 24 などです。

ステップ 5 アップリンク ポート チャネルの FC アップリンク ポート インターフェイス ポリシーとポリシー グループを作成するには、XML で POST を送信します。この例は次の要求を実行します。

- アップリンク ポートの FC インターフェイス ポリシー myFcUplinkIfPolicy2 を作成します。これらは、トランキングが有効になっている NP ポートです。
- FC アップリンク インターフェイス ポリシー myFcUplinkIfPolicy2 を含む FC インターフェイス バンドル ポリシー グループ myFcUplinkBundleGroup2 を作成します。
- ポリシー グループを FC インターフェイス ポリシーに関連付けて、これらのポートを FC ポートに変換します。
- アップリンク ポート プロファイル myFcUplinkPortProfile を作成します。
- ポートを 1/9 ～ 12 の範囲で指定するポート セレクタ myFcUplinkSelector を作成します。
- ホスト ポートをリーフ ノード 104 に関連付けます。

例：

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
    <infraInfra>
        <fcIfPol name="myFcUplinkIfPolicy2" portMode="np" trunkMode="trunk-on"
speed="auto"/>
        <infraFuncP>
            <infraFcAccBndlGrp name="myFcUplinkBundleGroup2">
                <infraRsFcL2IfPol tnFcIfPolName="myFcUplinkIfPolicy2" />
            </infraFcAccBndlGrp>
        </infraFuncP>
        <infraAccPortP name="myFcUplinkPortProfile">
            <infraHPortS name="myFcUplinkSelector" type="range">
                <infraPortBlk name="myUplinkPorts" fromCard="1" toCard="1" fromPort="9"
 toPort="12" />
```

```
                <infraRsAccBaseGrp
tDn="uni/infra/funcprof/fcaccportgrp-myFcUplinkBundleGroup2" />
            </infraHPortS>
        </infraAccPortP>
        <infraNodeP name="myFcNode1">
            <infraLeafS name="myLeafSelector" type="range">
                <infraNodeBlk name="myLeaf104" from_="104" to_="104" />
            </infraLeafS>
            <infraRsAccPortP tDn="uni/infra/accportprof-myUplinkPorts" />
        </infraNodeP>
    </infraInfra>
</polUni>
```

（注） この設定を適用する場合は、ポートを FC ポートとしてアップするためにスイッチのリロードが必要になります。

現在のみ FC ポートに変換できるポートの 1 つの連続した範囲と、この範囲にする必要がありますが 4 の倍数で終わるポート番号 4 の倍数ことです。たとえば、1 ～ 4、1 ～ 8、21 ～ 24 などです。

ステップ 6 テナント、アプリケーションプロファイル、EPG を作成し、FC ブリッジ ドメインを EPG に関連付けるするには、次の例などと XML post を送信します。例では、FC およびアプリケーション EPG epg1 をサポートするように設定されたターゲット テナントの下に、ブリッジ ドメイン myFcBD1 を作成します。これにより、ファイバ チャネル ドメイン myFcDomain1 とファイバ チャネル パスが、リーフ スイッチ 104 のインターフェイス 1/7 に関連付けられます。各インターフェイスは、VSAN に関連付けられます。

例：

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant name="tenant1">
  <fvCtx name="myFcVRF"/>
  <fvBD name="myFcBD1" type="fc">
    <fvRsCtx tnFvCtxName="myFcVRF"/>
  </fvBD>
  <fvAp name="app1">
    <fvAEPg name="epg1">
      <fvRsBd tnFvBDName="myFcBD1"/>
      <fvRsDomAtt tDn="uni/fc-myFcDomain1"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-104/pathep-[fc1/1]" vsan="vsan-50"
vsanMode="native"/>
      <fvRsFcPathAtt tDn="topology/pod-1/paths-104/pathep-[fc1/2]" vsan="vsan-50"
vsanMode="native"/>
    </fvAEPg>
  </fvAp>
</fvTenant>
```

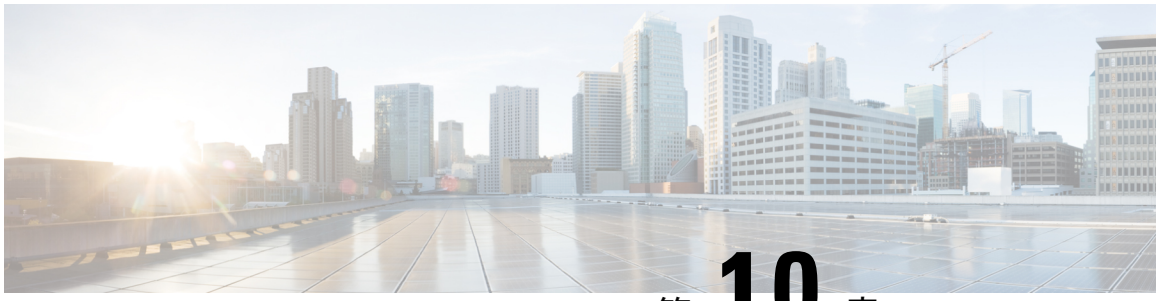
ステップ 7 サーバ ポートをアップリンク ポートに固定するトラフィック マップを作成するには、次の例のように XML で POST を送信します。この例では、サーバ ポート vFC 1/47 をアップリンク ポート FC 1/7 に固定するトラフィック マップを作成します。

例：

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant name="tenant1">
  <fvAp name="app1">
```

```
    <fvAEPg name="epg1">
      <fvRsFcPathAtt tDn="topology/pod-1/paths-104/pathep-[eth1/47]" vsan="vsan-50"
vsanMode="native">
        <fcPinningLbl name="label1"/>
      </fvRsFcPathAtt>
    </fvAEPg>
  </fvAp>
</fvTenant>

https://apic-ip-address/api/mo/uni/tn-vfc_t1.xml

<fvTenant name="tenant1">
  <fcPinningP name="label1">
    <fcRsPinToPath tDn="topology/pod-1/paths-104/pathep-[fc1/7]"/>
  </fcPinningP>
</fvTenant>
```

（注） トラフィック マップの固定を初めて設定する場合は、最初のトラフィック マップを設定する前にサーバ ホスト ポートをシャットダウンする必要があります。



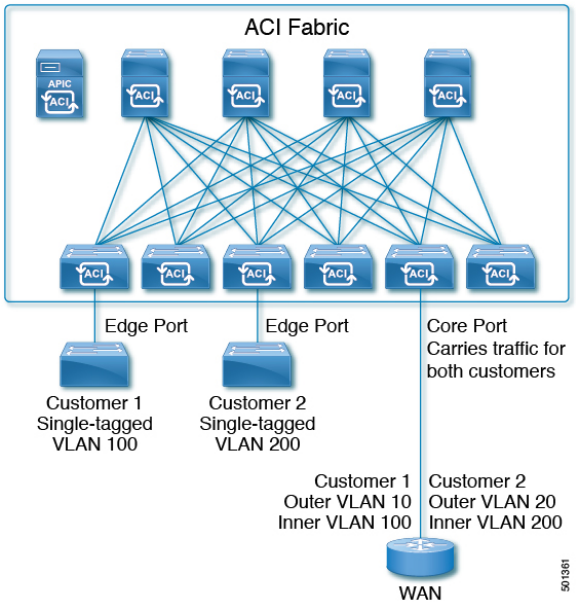
第 10 章

802.1 q トンネリング

この章は、次の内容で構成されています。

ACI 802.1 q トンネルについて

図 *32 : ACI 802.1 q* トンネル



エッジ（トンネル）ポートで 802.1Q トンネルを設定して、Quality of Service（QoS）の優先順位設定とともに、ファブリックのイーサネット フレームの point-to-multi-point トンネリングを有効にできます。Dot1q トンネルは、タグなし、802.1Q タグ付き、802.1ad 二重タグ付きフレームを、ファブリックでそのまま送信します。各トンネルでは、単一の顧客からのトラフィック

を伝送し、単一のブリッジ ドメインに関連付けられています。Cisco Application Centric Infrastructure（ACI）の前面パネル ポートは、Dot1q トンネル の一部とすることができます。レイヤ 2 スイッチングは宛先 MAC（DMAC）に基づいて行われ、通常の MAC ラーニングはトンネルで行われます。エッジ ポート Dot1q トンネルは、スイッチ モデル名の最後に「EX」またはそれ以降のサフィックスが付く、Cisco Nexus 9000 シリーズ スイッチでサポートされます。

同じコア ポートで複数の 802.1Q トンネルを設定することができ、複数の顧客からの二重タグ付きトラフィックを伝送できます。それぞれは、802.1Q トンネルごとに設定されたアクセスのカプセル化で識別されます。802.1Q トンネルでは、MAC アドレス学習を無効にすることもできます。エッジ ポートとコア ポートの両方を、アクセス カプセル化が設定され、MAC アドレス学習が無効にされた 802.1Q トンネルに所属させることができます。エッジ ポートとコア ポートの Dot1q トンネル は、スイッチ モデル名の最後に「FX」またはそれ以降のサフィックスが付く、Cisco Nexus 9000 シリーズ スイッチでサポートされます。

IGMP および MLD パケットは、802.1Q トンネルを介して転送できます。

このドキュメントで使用する用語は、**Cisco Nexus 9000 シリーズ** のドキュメントとは異なっている場合があります。

表 *5 : 802.1Q* トンネルの用語

ACI のドキュメント	Cisco Nexus 9000 シリーズのドキュメント
エッジ ポート	トンネル ポート
コア ポート	トランク ポート

次の注意事項および制約事項が適用されます:

- VTP、CDP、LACP、LLDP、および STP プロトコルのレイヤ 2 トンネリングは、次の制限付きでサポートされます。
  - リンク集約制御プロトコル (LACP) トンネリングは、個々のリーフ インターフェイスを使用する、ポイントツーポイント トンネルでのみ、予想通りに機能します。ポート チャネル（PC）または仮想ポートチャネル（vPC）ではサポートされていません。
  - PC または vPC を持つ CDP および LLDP トンネリングは確定的ではありません。これは、トラフィックの宛先として選択するリンクによって異なります。
  - レイヤ 2 プロトコル トンネリングに VTP を使用するには、CDP をトンネル上で有効にする必要があります。
  - レイヤ 2 プロトコルのトンネリングが有効になっており、Dot1q トンネルのコア ポートにブリッジ ドメインが展開されている場合、STP は 802.1Q トンネル ブリッジ ドメインではサポートされません。
  - Cisco ACI リーフ スイッチは、トンネル ブリッジ ドメインのエンドポイントでフラッシングを行い、ブリッジ ドメインでフラッディングすることにより、STP TCN パケットに反応します。

- 2 個上のインターフェイスを持つ CDP および LLDP トンネリングが、すべてのインターフェイスでパケットをフラッディングします。
- エッジ ポートからコア ポートにトンネリングしているレイヤ 2 プロトコル パケットの宛先 MAC アドレスは、01-00-0c-cd-cd-d0 に書き換えられ、コア ポートからエッジ ポートにトンネリングしているレイヤ 2 プロトコル パケットの宛先 MAC アドレスは、プロトコルに対して標準のデフォルト MAC アドレスに書き換えられます。

- PC または vPC が Dot1q Tunnel 内の唯一のインターフェイスであり、削除してから再設定した場合には、PC/VPC の Dot1q トンネル への関連付けを削除して、再設定してください。
- 製品 ID に EX が含まれるスイッチに導入された 802.1Q トンネルでは、最初の 2 つの VLAN タグの 0x8100 + 0x8100、0x8100 + 0x88a8、0x88a8 + 0x88a8のEthertype の組み合わせはサポートされません。

  トンネルが EX と FX またはそれ以降のスイッチの組み合わせに導入されている場合は、この制限が適用されます。

  製品 ID に FX 以降が含まれるスイッチにのみトンネルが導入されている場合、この制限は適用されません。
- コア ポートについては、二重タグつきフレームのイーサタイプは、0x8100 の後に 0x8100 が続く必要があります。
- 複数のエッジ ポートおよびコア ポートを（リーフ スイッチ上のものであっても）Dot1q トンネル に含めることができます。
- エッジ ポートは 1 つのトンネルの一部にのみ属することが可能ですが、コア ポートは複数の Dot1q トンネルに属することができます。
- 通常の EPG を 802.1Q で使用されるコア ポートに展開できます。
- L3Outs は、Dot1q トンネルで有効になっているインターフェイスではサポートされていません。
- FEX インターフェイスは Dot1q トンネル のメンバーとしてはサポートされていません。
- インターフェイス レベルの統計情報は Dot1q トンネル のインターフェイスでサポートされていますが、トンネル レベルの統計情報はサポートされていません。

GUI を使用した802.1Q トンネルの設定

APIC GUI を使用した 802. 1 Q トンネル インターフェイスの設定

次の手順で、トンネルを使用するインターフェイスを設定します:

始める前に

トンネルを使用するテナントを作成します。

手順

ステップ 1 メニュー バーで、**[Fabric]** > **[Access Policies]** の順にクリックします。

ステップ 2 [ナビゲーション] バーで、**[ポリシー]** > **[インターフェイス]** > **[L2 インターフェイス]** をクリックします。

ステップ 3 **[L2 インターフェイス]** を右クリックし、**[L2 インターフェイス ポリシーの作成]** を選択して、次の操作を実行します。

a) **Name** フィールドに、レイヤ 2 インターフェイス ポリシーの名前を入力します。

b) オプション。ポリシーの説明を追加します。L2 インターフェイス ポリシーの目的を説明することをお勧めします。

c) **Dot1q トンネル**で、エッジ ポートとして使用するインターフェイスを有効にするインターフェイス ポリシーを作成するために、**QinQ** フィールドで、**edgePort** をクリックします。

d) **Dot1q トンネル**でコア ポートとして使用するインターフェイスを有効にするインターフェイス ポリシーを作成するために、**QinQ** フィールドで、**corePort** をクリックします。

ステップ 4 次の手順で、L2 インターフェイス ポリシーをポリシー グループに適用します。

a) **[ファブリック]** > **[アクセス ポリシー]** > **[インターフェイス]** > **[リーフ インターフェイス]** をクリックして、**[ポリシー グループ]** を展開します。

b) **[リーフ アクセス ポート]**、**[PC インターフェイス]** または **[VPC インターフェイス]** を右クリックし、トンネルに設定しているインターフェイスのタイプに応じて、次のいずれかを選択します。

- **リーフ アクセス ポート ポリシー グループの作成**
- **PC ポリシー グループの作成**
- **VPC ポリシー グループの作成**

c) 表示されるダイアログボックスで、以下のアクションを実行します:

- **Name** フィールドに、ポリシー グループの名前を入力します。

  オプション。ポリシー グループについての説明を追加します。ポリシー グループの目的を説明することをお勧めします。

- **L2 Interface Policy** フィールドで、下向き矢印をクリックし、前に作成した L2 インターフェイス ポリシーを選択します。

- CDP レイヤ 2 トンネリング プロトコルでトンネルを作成する場合は、[CDP Policy] 下向き矢印をクリックし、ポリシーダイアログボックスでポリシーの名前を追加し、管理状態を無効にして、[Submit] をクリックします。.

- LLDP レイヤ 2 トンネリング プロトコルでトンネルを作成する場合には、[LLDP Policy] 下向き矢印をクリックし、ポリシーダイアログボックスでポリシーの名前を追加し、送信状態を無効にして [submit] をクリックします。
- [Submit] をクリックします。

ステップ 5 次の手順に従ってリーフ インターフェイス プロファイルを作成します:

a) **[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Profiles]** をクリックします。

b) **Profiles** プロファイルを右クリックし、**Create Leaf Interface Profile** を選択し、次の手順に従います:

- **Name** フィールドに、**Leaf Interface Profile** の名前を入力します。

  オプション。説明を追加します。

- **Interface Selectors** フィールドで、**+** をクリックし、以下の情報を入力します:
  - **[名前]** フィールドに、インターフェイス セレクタの名前を入力します。

    オプション。説明を追加します。
  - **Interface IDs** フィールドに、このトンネルに含められる **Dot1q Tunnel** インターフェイス、または複数のインターフェイスの名前を入力します。
  - **Interface Policy Group** フィールドで、下向き矢印をクリックして、前に作成したインターフェイス ポリシー グループを選択します。

ステップ 6 トンネル設定のポートへのスタティック バインディングを作成するには、**[Tenant]** > **[Networking]** > **[Dot1Q Tunnels]** の順にクリックします。[Dot1Q Tunnels] を展開し、前に作成した **Dot1Q Tunnels <ポリシー名>** をクリックして、次の操作を実行します。

a) [Static Bindings] テーブルを展開して [Create Static Binding] ダイアログボックスを開きます。

b) [Port] フィールドで、ポートの種類を選択します。

c) [Node] フィールドで、ドロップダウンリストからノードを選択します。

d) [Path] フィールドで、ドロップダウンリストからインターフェイス パスを選択し、[Submit] をクリックします。

NX-OS スタイルの CLI を使用した 802.1Q トンネルの設定

NX-OS スタイル CLI を使用した802.1Q トンネルの設定

（注） **Dot1q トンネル** に含まれるインターフェイスのポート、ポート チャネル、仮想ポート チャネルを使用できます。手順の詳細にはポートの設定が含まれます。エッジおよびコア ポート チャネルと仮想ポート チャネルを設定するコマンドについては、下の例を参照してください。

次の手順で、**Dot1q トンネル** を作成し、NX-OS スタイル CLI を使用してトンネルで使用するインターフェイスを設定します。

（注） **Dot1q トンネル** には2 個以上のインターフェイスを含める必要があります。手順を繰り返し（または 2 個のインターフェイスをまとめて設定)、**Dot1q トンネル** で使用する各インターフェイスをマークします。この例で、2 個のインターフェイスは単一の顧客で使用されているエッジ スイッチ ポートとして設定されます。

次の手順を使用して、設定を次の手順を使用して、NX-OS スタイル CLI を使用して **Dot1q トンネル** を設定します。

1. トンネルで使用するインターフェイスを最低 2 個設定します。
2. **Dot1q トンネル** を作成します。
3. トンネルとすべてのインターフェイスを関連付けます。

始める前に

Dot1q トンネル を使用するテナントを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `apic1# configure`	コンフィギュレーション モードに入ります。
ステップ 2	次の手順により 802.1Q で使用するための 2 個のインターフェイスを設定します。	

	コマンドまたはアクション	目的
ステップ 3	**leaf** *ID* 例： `apic1(config)# leaf 101`	**Dot1q トンネル** のインターフェイスが配置されるリーフを特定します。
ステップ 4	**interface ethernet** *slot/port* 例： `apic1(config-leaf)# interface ethernet 1/13-14`	トンネルのポートとしてマークされるインターフェイスを特定します。

ステップ 5	**switchport mode dot1q-tunnel** {**edgePort** | **corePort**}
	例：
	`apic1(config-leaf-if)# switchport mode dot1q-tunnel edgePort`
	`apic1(config-leaf-if)# exit`
	`apic1(config-leaf)# exit`
	`apic1(config)# exit`
		802.1Q トンネルで使用するインターフェイスをマークして、設定モードをそのままにします。
		この例では、エッジポートを使用するためにいくつかのインターフェイス設定を示します。トンネルに複数のインターフェイスを設定するには、手順 3 ～ 5 を繰り返します。

ステップ 6	次の手順で802.1q トンネルを作成します。	
ステップ 7	**leaf** *ID* 例： `apic1(config)# leaf 101`	インターフェイスが配置されているリーフに戻ります。
ステップ 8	**interface ethernet***slot/port* 例： `apic1(config-leaf)# interface ethernet 1/13-14`	トンネルに含まれるインターフェイスに戻ります。

ステップ 9	**switchport tenant***tenant-name***dot1q-tunnel** *tunnel-name*
	例：
	`apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_edgetunnel`
	`apic1(config-leaf-if)# exit`
		トンネルにインターフェイスに関連付け、設定モードを終了します。

ステップ 10	トンネルとその他のインターフェイスを関連付けるには、ステップ 7 ～ 10 を繰り返します。	

例：NX-OS スタイル CLI でポートを使用する 802.1Q トンネルを設定する

この例では、2 つのポートを **Dot1q トンネル**で使用されるエッジ ポート インターフェイスとしてマークし、 さらに 2 つの ポートをコア ポート インターフェイスで使用されるものととしてマークし、トンネルを作成して、ポートをトンネルに関連付けます。

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/13-14
apic1(config-leaf-if)# switchport mode dot1q-tunnel edgePort
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)leaf 102
apic1(config-leaf)# interface ethernet 1/10, 1/21
apic1(config-leaf-if)# switchport mode dot1q-tunnel corePort
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# tenant tenant64
apic1(config-tenant)# dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# l2protocol-tunnel cdp
apic1(config-tenant-tunnel)# l2protocol-tunnel lldp
apic1(config-tenant-tunnel)#  access-encap 200
apic1(config-tenant-tunnel)# mac-learning disable
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/13-14
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 1/10, 1/21
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

例：NX-OS スタイル CLI でポート チャネルを使用する 802.1Q トンネルを設定する

例では、このエッジ ポート 8021 q インターフェイスとして 2 つのポート チャネルにマークし、2 つ以上のポート チャネルをコア ポート 802.1 q インターフェイスとしてマークして、**Dotq トンネル** を作成し、トンネルとポート チャネルを関連付けます。

```
apic1# configure
apic1(config)# tenant tenant64
apic1(config-tenant)# dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# l2protocol-tunnel cdp
apic1(config-tenant-tunnel)# l2protocol-tunnel lldp
apic1(config-tenant-tunnel)#  access-encap 200
apic1(config-tenant-tunnel)# mac-learning disable
apic1(config-tenant-tunnel)# exit
```

```
apic1(config-tenant)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel pc1
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/2-3
apic1(config-leaf-if)# channel-group pc1
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface port-channel pc1
apic1(config-leaf-if)# switchport mode dot1q-tunnel edgePort
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel pc2
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/4-5
apic1(config-leaf-if)# channel-group pc2
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface port-channel pc2
apic1(config-leaf-if)# switchport mode dot1q-tunnel corePort
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
```

例：NX-OS スタイル CLI で仮想ポート チャネルを使用する 802.1Q トンネルを設定する

この例では、2 つの仮想ポート チャネル (vPC) を **Dot1q トンネル**の エッジポート 802.1Q インターフェイスとしてマークし、さらに2つのVPCをトンネルのためのコアポートインターフェイスとしてマークし、トンネルを作成して、仮想ポートチャネルをトンネルに関連付けています。

```
apic1# configure
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc vpc1
apic1(config-vpc-if)# switchport mode dot1q-tunnel edgePort
apic1(config-vpc-if)# exit
apic1(config-vpc)# exit
apic1(config)# vpc domain explicit 1 leaf 103 104
apic1(config)# vpc context leaf 103 104
apic1(config-vpc)# interface vpc vpc2
apic1(config-vpc-if)# switchport mode dot1q-tunnel corePort
apic1(config-vpc-if)# exit
apic1(config-vpc)# exit
apic1(config)# tenant tenant64
apic1(config-tenant)# dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# l2protocol-tunnel cdp
apic1(config-tenant-tunnel)# l2protocol-tunnel lldp
apic1(config-tenant-tunnel)#  access-encap 200
apic1(config-tenant-tunnel)# mac-learning disable
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 103
apic1(config-leaf)# interface ethernet 1/6
apic1(config-leaf-if)# channel-group vpc1 vpc
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 104
```

```
apic1(config-leaf)# interface ethernet 1/6
apic1(config-leaf-if)# channel-group vpc1 vpc
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config-vpc)# interface vpc vpc1
apic1(config-vpc-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-vpc-if)# exit
```

REST API を使用した 802.1Q トンネルの設定

REST API を使用してポートを持つトンネル 802.1 q の設定

作成、 **Dot1q トンネル** 、ポートを使用して、次の例などの手順でのインターフェイスを設定します。

始める前に

Dot1q トンネル を使用するテナントを設定します。

手順

ステップ 1 次の例のように XML で REST API を使用して **Dot1q トンネル** を作成します。

例では、LLDP レイヤ 2 トンネリング プロトコルでトンネルを設定し、アクセス カプセル化 VLAN を追加し、トンネルで MAC ラーニングを無効にします。

例：

```
<fvTnlEPg name="VRF64_dot1q_tunnel" qiqL2ProtTunMask="lldp" accEncap="vlan-10"
fwdCtrl="mac-learn-disable"  >
    <fvRsTnlpathAtt  tDn="topology/pod-1/paths-101/pathep-[eth1/13]"/>
</fvTnlEPg>
```

ステップ 2 次の例のように XML で静的にバインディングするレイヤ 2 インターフェイス ポリシーを設定します。

この例では、エッジ スイッチ ポートにレイヤ 2 インターフェイス ポリシーを設定します。コア スイッチ ポートのポリシーを設定するには、`l2IfPol` MO で `edgePort` の代わりに `corePort` を使用します。

例：

```
<l2IfPol name="VRF64_L2_int_pol" qinq="edgePort"  />
```

ステップ 3 次の例のように、XML でリーフ アクセス ポート ポリシー グループにレイヤ 2 インターフェイス ポリシーを適用します。

例：

```
<infraAccPortGrp name="VRF64_L2_Port_Pol_Group" >
    <infraRsL2IfPol tnL2IfPolName="VRF64_L2_int_pol"/>
</infraAccPortGrp>
```

ステップ 4 次の例のように、XML でインターフェイス セレクタとともにリーフ プロファイルを設定します。

例：

```
<infraAccPortP name="VRF64_dot1q_leaf_profile" >
    <infraHPortS name="vrf64_access_port_selector" type="range">
        <infraPortBlk name="block2" toPort="15" toCard="1" fromPort="13" fromCard="1"/>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-VRF64_L2_Port_Pol_Group"
/>
    </infraHPortS>
</infraAccPortP>
```

例

次の例のように、2 個の POST 上のエッジ ポートのポート設定を示します。

Post 1 の XML：

```
<polUni>
  <infraInfra>
    <l2IfPol name="testL2IfPol" qinq="edgePort"/>
      <infraNodeP name="Node_101_phys">
      <infraLeafS name="phys101" type="range">
        <infraNodeBlk name="test" from_="101" to_="101"/>
      </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-phys21"/>
    </infraNodeP>
    <infraAccPortP name="phys21">
      <infraHPortS name="physHPortS" type="range">
        <infraPortBlk name="phys21" fromCard="1" toCard="1" fromPort="21" toPort="21"/>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-21"/>
      </infraHPortS>
    </infraAccPortP>
    <infraFuncP>
      <infraAccPortGrp name="21">
        <infraRsL2IfPol tnL2IfPolName="testL2IfPol"/>
        <infraRsAttEntP tDn="uni/infra/attentp-AttEntityProf1701"/>
      </infraAccPortGrp>
    </infraFuncP>
    <l2IfPol name='testL2IfPol' qinq=‘edgePort'/>
    <infraAttEntityP name="AttEntityProf1701">
      <infraRsDomP tDn="uni/phys-dom1701"/>
    </infraAttEntityP>
  </infraInfra>
</polUni>
```

Post 2 の XML：

```
<polUni>
  <fvTenant dn="uni/tn-Coke" name="Coke">
    <fvTnlEPg name="WEB5" qiqL2ProtTunMask="lldp" accEncap="vlan-10"
fwdCtrl="mac-learn-disable" >
        <fvRsTnlpathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/21]"/>
      </fvTnlEPg>
  </fvTenant>
</polUni>
```

REST API を使用した PC を持つトンネル 802.1Q の設定

PC を使用して **Dot1q トンネル** を作衛して、次の例のような手順でインターフェイスを設定します。

始める前に

Dot1q トンネル を使用するテナントを設定します。

手順

ステップ 1 次の例のように XML で REST API を使用して **Dot1q トンネル** を作成します。

例では、LLDP レイヤ 2 トンネリング プロトコルでトンネルを設定し、アクセス カプセル化 VLAN を追加し、トンネルで MAC ラーニングを無効にします。

例：

```
    <fvTnlEPg name="WEB" qiqL2ProtTunMask=lldp accEncap="vlan-10"
fwdCtrl="mac-learn-disable" >
        <fvRsTnlpathAtt tDn="topology/pod-1/paths-101/pathep-[po2]"/>
      </fvTnlEPg>
```

ステップ 2 次の例のように XML で静的にバインディングするレイヤ 2 インターフェイス ポリシーを設定します。

この例では、エッジ スイッチ ポートにレイヤ 2 インターフェイス ポリシーを設定します。コア スイッチ ポートのレイヤ 2 インターフェイス ポリシーを設定するには、`l2IfPol` MO の `edgePort` 代わりに `corePort` を使用します。

例：

```
<l2IfPol name="testL2IfPol" qinq="edgePort"/>
```

ステップ 3 次の例のように、XML で PC インターフェイス ポリシー グループにレイヤ 2 インターフェイス ポリシーを適用します。

例：

```
<infraAccBndlGrp name="po2" lagT="link">
   <infraRsL2IfPol tnL2IfPolName="testL2IfPol"/>
    </infraAccBndlGrp>
```

ステップ 4 次の例のように、XML でインターフェイス セレクタを持つリーフ プロファイルを設定します。

例：

```
<infraAccPortP name="PC">
   <infraHPortS name="allow" type="range">
    <infraPortBlk name="block2" fromCard="1" toCard="1" fromPort="10" toPort="11" />
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-po2"/>
   </infraHPortS>
  </infraAccPortP>
```

例

次の例では、2 個のポストの PC 設定を示します。

この例では、エッジ ポートとして PC ポートを設定します。コア ポートとして設定を使用するには、Post 1 にある `l2IfPol` MO の `edgePort` 代わりに `corePort` を使用します。

Post 1 の XML：

```
<infraInfra dn="uni/infra">
    <infraNodeP name="bLeaf3">
        <infraLeafS name="leafs3" type="range">
            <infraNodeBlk name="nblk3" from_="101" to_="101">
            </infraNodeBlk>
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-shipping3"/>
    </infraNodeP>
    <infraAccPortP name="shipping3">
    <infraHPortS name="pselc3" type="range">
        <infraPortBlk name="blk3" fromCard="1" toCard="1" fromPort="24" toPort="25"/>

        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-accountingLag3" />
    </infraHPortS>
  </infraAccPortP>
<infraFuncP>
   <infraAccBndlGrp name="accountingLag3" lagT='link'>
    <infraRsAttEntP tDn="uni/infra/attentp-default"/>
        <infraRsLacpPol tnLacpLagPolName='accountingLacp3'/>
        <infraRsL2IfPol tnL2IfPolName="testL2IfPol3"/>
   </infraAccBndlGrp>
</infraFuncP>
<lacpLagPol name='accountingLacp3' ctrl='15' descr='accounting' maxLinks='14' minLinks='1'
 mode='active' />
<l2IfPol name='testL2IfPol3' qinq='edgePort'/>
<infraAttEntityP name="default">
 </infraAttEntityP>
</infraInfra>
```

Post 2 の XML：

```
<polUni>
  <fvTenant dn="uni/tn-Coke" name="Coke">
    <!-- bridge domain -->
    <fvTnlEPg name="WEB6" qiqL2ProtTunMask="lldp" accEncap="vlan-10"
fwdCtrl="mac-learn-disable" >
        <fvRsTnlpathAtt tDn="topology/pod-1/paths-101/pathep-[accountingLag1]"/>
      </fvTnlEPg>
  </fvTenant>
</polUni>
```

REST API を使用した vPC での 802.1 Q トンネルの設定

vPC を使用して **Dot1q トンネル**を作成し、次の例のような手順でインターフェイスを設定します。

始める前に

Dot1q トンネル を使用するテナントを設定します。

手順

ステップ 1 次の例のように、XML とともに REST API を使用して 802.1 q トンネルを作成します。

例ではレイヤ 2 トンネル プロトコルとともにトンネルを設定し、アクセス カプセル化 VLAN を追加し、トンネルで MAC 学習を無効にします。

例：

```
<fvTnlEPg name="WEB" qiqL2ProtTunMask=lldp accEncap="vlan-10" fwdCtrl="mac-learn-disable"
 >
        <fvRsTnlpathAtt tDn="topology/pod-1/protpaths-101-102/pathep-[po4]" />
      </fvTnlEPg>
```

ステップ 2 次の例のように、XML とともに静的バインディングでレイヤ 2 インターフェイス ポリシーを設定します。

この例では、エッジ スイッチ ポートのレイヤ 2 インターフェイス ポリシーを設定します。コア スイッチ ポートのレイヤ 2 インターフェイス ポリシーを設定するには、`qinq ="corePort"` ポート タイプを使用します。

例：

```
<l2IfPol name="testL2IfPol" qinq="edgePort"/>
```

ステップ 3 次の例のように、XML を持つ VPC インターフェイス ポリシー グループにレイヤ 2 インターフェイス ポリシーを適用します。

例：

```
<infraAccBndlGrp name="po4" lagT="node">
   <infraRsL2IfPol tnL2IfPolName="testL2IfPol"/>
   </infraAccBndlGrp>
```

ステップ 4 次の例のように、XML を持つインターフェイス セレクタでリーフ プロファイルを設定します。

例：

```
  <infraAccPortP name="VPC">
   <infraHPortS name="allow" type="range">
    <infraPortBlk name="block2" fromCard="1" toCard="1" fromPort="10" toPort="11" />
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-po4"/>
   </infraHPortS>
  </infraAccPortP>
```

例

次の例は、3つのポストでのvPC設定を示しています。

この例では、vPC ポートをエッジ ポートとして設定します。コア ポートとして設定するには、ポスト 2 で `corePort` を `edgePort` の代わりに `l2IfPol` MO で使用します。

Post 1 の XML：

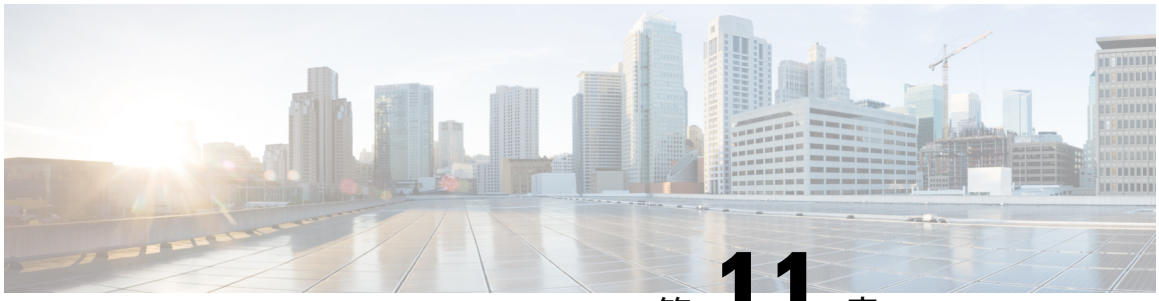
```
<polUni>
    <fabricInst>
        <fabricProtPol pairT="explicit">
            <fabricExplicitGEp name="101-102-vpc1" id="30">
                <fabricNodePEp id="101"/>
                <fabricNodePEp id="102"/>
            </fabricExplicitGEp>
        </fabricProtPol>
    </fabricInst>
</polUni>
```

Post 2 の XML：

```
<infraInfra dn="uni/infra">
    <infraNodeP name="bLeaf1">
        <infraLeafS name="leafs" type="range">
            <infraNodeBlk name="nblk" from_="101" to_="101">
            </infraNodeBlk>
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-shipping1"/>
    </infraNodeP>

    <infraNodeP name="bLeaf2">
        <infraLeafS name="leafs" type="range">
            <infraNodeBlk name="nblk" from_="102" to_="102">
            </infraNodeBlk>
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-shipping2"/>
    </infraNodeP>

  <infraAccPortP name="shipping1">
    <infraHPortS name="pselc" type="range">
        <infraPortBlk name="blk" fromCard="1" toCard="1" fromPort="4" toPort="4"/>

        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-accountingLag1" />
    </infraHPortS>
  </infraAccPortP>

  <infraAccPortP name="shipping2">
    <infraHPortS name="pselc" type="range">
        <infraPortBlk name="blk" fromCard="1" toCard="1" fromPort="2" toPort="2"/>

        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-accountingLag2" />
    </infraHPortS>
  </infraAccPortP>

<infraFuncP>
   <infraAccBndlGrp name="accountingLag1" lagT='node'>
    <infraRsAttEntP tDn="uni/infra/attentp-default"/>
        <infraRsLacpPol tnLacpLagPolName='accountingLacp1'/>
        <infraRsL2IfPol tnL2IfPolName="testL2IfPol"/>
   </infraAccBndlGrp>
   <infraAccBndlGrp name="accountingLag2" lagT='node'>
    <infraRsAttEntP tDn="uni/infra/attentp-default"/>
         <infraRsLacpPol tnLacpLagPolName='accountingLacp1'/>
         <infraRsL2IfPol tnL2IfPolName="testL2IfPol"/>
   </infraAccBndlGrp>
</infraFuncP>
<lacpLagPol name='accountingLacp1' ctrl='15' descr='accounting' maxLinks='14' minLinks='1'
 mode='active' />
<l2IfPol name='testL2IfPol' qinq='edgePort'/>

 <infraAttEntityP name="default">
 </infraAttEntityP>
</infraInfra>
```

Post 3 の XML：

```
<polUni>
  <fvTenant dn="uni/tn-Coke" name="Coke">
    <!-- bridge domain -->
    <fvTnlEPg name="WEB6" qiqL2ProtTunMask="lldp" accEncap="vlan-10"
fwdCtrl="mac-learn-disable" >
       <fvRsTnlpathAtt tDn="topology/pod-1/protpaths-101-102/pathep-[accountingLag2]"/>

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



第 11 章

Epg の Q-で-Q カプセル化のマッピング

Epg の Q-で-Q カプセル化のマッピング

Cisco Application Policy Infrastructure Controller（APIC）を使用すれば、通常のインターフェイス、PC、または vPC で入力される二重タグ付き VLAN トラフィックを EPG にマッピングできます。この機能が有効で、二重タグ付きトラフィックが EPG のネットワークに入ると、両方のタグがファブリック内で個別に処理され、Cisco Application Centric Infrastructure（ACI）スイッチの出力時に二重タグに復元されます。単一タグおよびタグなしのトラフィックの入力はドロップします。

次の注意事項および制約事項が適用されます。

- この機能は、Cisco Nexus 9300-FX プラットフォーム スイッチでのみサポートされています。
- 外側と内側の両方のタグは、EtherType 0x8100 である必要があります。
- MAC ラーニングとルーティングは、アクセスのカプセル化ではなく、EPG ポート、sclass、および VRF インスタンスに基づいています。
- QoS 優先度設定がサポートされ、入力の外側のタグから派生し、出力の両方のタグに書き換えられます。
- EPG はリーフ スイッチの他のインターフェイスに同時に関連付けることができ、単一タグの VLAN に設定されます。
- サービス グラフは、Q-in-Q カプセル化したインターフェイスにマッピングされているプロバイダとコンシューマ EPG をサポートしています。サービス ノードの入力および出力ト

ラフィックが単一タグのカプセル化フレームにある限り、サービス グラフを挿入することができます。

- vPC ポートが Q-in-Q カプセル化モードに対して有効になっている場合、VLAN 整合性チェックは実行されません。

この機能では、次の機能とオプションがサポートされていません。

- ポート単位の VLAN 機能
- FEX 接続
- Mixed mode

  たとえば、Q-in-Q カプセル化モードのインターフェイスでは、通常の VLAN のカプセル化ではなく、二重タグ付きカプセルのみを持つ EPG にバインディングされている静的パスを有します。
- STP と「カプセル化でのフラッディング」オプション
- タグなしおよび 802.1p モード
- マルチポッドと複数サイト
- レガシ ブリッジ ドメイン
- L2Out および L3Out 接続
- VMM の統合
- ポート モードをルーテッドから Q-in-Q カプセル化モードに変更する
- Q-in-Q カプセル化モードのポートでの VLAN 単位の誤配線プロトコル

GUI を使用した EPG の Q-in-Q カプセル化マッピングの設定

GUI を使用して、特定のリーフ スイッチ インターフェイス上で Q-in-Q カプセル化を有効にします

リーフスイッチポート、PC、または vPC は、APIC GUI の次のいずれかの場所の **[インターフェイス (Interface)]** タブで Q-in-Q カプセル化モードを有効にします。

- **[Fabric]** > **[Inventory]** > **[Topology]**
- **[Fabric]** > **[Inventory]** > **[Pod]**
- **[Fabric]** > **[Inventory]** > **[Pod]** > ***[leaf-name]***

[Topology] タブまたは **[Pod Interface]** タブで **VPC** を設定します。

始める前に

Q-in-Qモードに設定されたインターフェイスでマッピングされるテナント、アプリケーションプロファイル、およびアプリケーション EPG を作成する必要があります。

手順

ステップ 1 メニューバーで **[Fabric** > **Inventory]** を選択し、**[Topology]**、**[Pod]** をクリックするか、**[Pod]** を展開してリーフを選択します。

ステップ 2 [Topology] タブ、または [Pod] パネルの [Interface] タブを選択します。

ステップ 3 **[Operation**/**Configuration]** トグル ボタンをクリックして、設定パネルを表示します。

ステップ 4 **[+]** をクリックしてリーフスイッチの図を追加し、1 つ以上のスイッチを選択して**[Add Selected]** をクリックします。

[<リーフ名>] パネルの [Interface] タブで、[Operation]/[Configuration] トグルボタンをクリックすると、自動的にスイッチのダイアグラムが表示されます。

ステップ 5 Q-in-Q カプセル化モードを有効にするインターフェイスをクリックします。

ステップ 6 ポートを設定するには、次の手順を実行します。

a) 左上の **L2** をクリックします。
b) L2 タブの **[L2 QinQ State]** フィールドで **[Double Q Tag Port]** をクリックし、**[Submit]** をクリックします。

ステップ 7 PCを設定するには、次の手順を実行します。

a) 左上の **PC** をクリックします。
b) **[Physical Interface]** タブで、**[Policy Group Name]** を入力します。
c) L2 タブの **[L2 QinQ State]** フィールドで **[Double Q Tag Port]** をクリックし、**[Submit]** をクリックします。

ステップ 8 vPC を設定するには、次のステップを実行します。

a) 2 つのリーフ スイッチ ダイアグラムで、VPC の 2 つのレッグのインターフェイスをクリックします。
b) **[vPC]** をクリックします。
c) **[Physical Interface]** タブで、**[Logical Pair ID]** (自動保護グループの識別子) を入力します。各保護グループには、固有の ID があります。ID は1～1000 の範囲です) および **[Policy Group Name]**。
d) L2 タブの **[L2 QinQ State]** フィールドで **[Double Q Tag Port]** をクリックし、**[Submit]** をクリックします。

GUIを使用したファブリックインターフェイスポリシーでリーフインターフェイスのQ-in-Qカプセル化の有効化

リーフ インターフェイス プロファイルを使用して、Q-in-Q カプセル化のリーフ インターフェイス、PC、および vPC を有効にします。

始める前に

Q-in-Qモードに設定されたインターフェイスでマッピングされるテナント、アプリケーションプロファイル、およびアプリケーション EPG を作成する必要があります。

手順

ステップ 1 メニュー バーで、**Fabric** > **External Access Policies** を選択します。

ステップ 2 [ナビゲーション] バーで、**[ポリシー]** > **[インターフェイス]** > **[L2 インターフェイス]** をクリックします。

ステップ 3 **[L2 インターフェイス]** を右クリックし、**[L2 インターフェイス ポリシーの作成]** を選択して、次の操作を実行します。

a) **[名前]** フィールドに、レイヤ 2 インターフェイス ポリシーの名前を入力します。

b) オプション。ポリシーの説明を追加します。L2 インターフェイス ポリシーの目的を説明することをお勧めします。

c) Q-in-Qカプセル化を有効にするインターフェイスポリシーを作成するには、**[QinQ]** フィールドで **[doubleQtagPort]** をクリックします。

d) [Submit] をクリックします。

ステップ 4 次の手順で、ポリシー グループに L2 インターフェイス ポリシーを適用されます。

a) **[ファブリック]** > **[外部アクセス ポリシー]** > **[インターフェイス]** > **[リーフ インターフェイス]** をクリックし、**[ポリシー グループ]** を展開します。

b) **[リーフ アクセス ポート]**、**[PC インターフェイス]**、または **[vPC インターフェイス]** を右クリックし、トンネルに設定するインターフェイスのタイプに応じて、次のいずれかを選択します。

- **リーフ アクセス ポート ポリシー グループの作成**
- **PC ポリシー グループの作成**
- **vPC ポリシー グループの作成**

c) 結果のダイアログボックスでポリシーグループ名を入力し、以前作成したL2インターフェイス ポリシーを選択し、**[送信]** をクリックします。

ステップ 5 次の手順で、リーフ インターフェイス プロファイルを作成します。

a) **[ファブリック]** > **[外部アクセス ポリシー]** > **[インターフェイス]** > **[リーフ インターフェイス]** > **[プロファイル]** の順にクリックします。

b) **[リーフ プロファイル]** を右クリックして、**[リーフ インターフェイス ポリシーの作成]** を選択し、次の手順を実行します。

- **Name** フィールドに、**Leaf Interface Profile** の名前を入力します。

  オプション。説明を追加します。

- **[インターフェイス セレクタ]** フィールドで、**[+]** をクリックし、次の情報を入力します。
  - **[名前]** フィールドに、インターフェイス セレクタの名前を入力します。

    オプション。説明を追加します。
  - セレクタの名前とし、任意で説明を入力します。
  - インターフェイス ID フィールドに、プロファイルに含む単一または複数のインターフェイスを入力します。
  - **[インターフェイス ポリシー グループ]** フィールドで、以前作成したインターフェイス ポリシー グループを選択します。

GUI を使用して EPG から Q-in-Q カプセル化が有効なインターフェイスにマッピングする

EPF は、次のモデルのいずれかで Q-in-Q カプセルが有効なインターフェイスに関連付けることができます:

- 特定の Q-in-Q カプセル化が有効なインターフェイス上に静的な EPG を展開します。
- EPG を Q-in-Q カプセル化が有効なリーフ スイッチに静的にリンクします。
- EPG を Q-in-Q カプセル化が有効なエンドポイント (スタティック MAC アドレスを持つもの) に関連付けます

APIC GUI の同じエリアに 3 つすべてのタスクが実行されます。

始める前に

- Q-in-Q モードで構成されたインターフェイスにマッピングされるテナント、アプリケーション プロファイル、おおびアプリケーション EPG を作成します。
- ターゲット インターフェイスは Q-in-Q カプセル化で構成されている必要があります。

手順

ステップ 1 メニュー バーで、**Tenants** > ***tenant-name*** の順にクリックします。

ステップ 2 ナビゲーション ウィンドウで、**Application Profiles** > > ***application-profile-name*** > **Application EPGs** > ***application-EPG-name*** を展開します。

ステップ 3 Q-in-Q モードが有効になっているインターフェイス、PC、または vPC にスタティック EPG を展開するには、次の手順を実行します。

a) アプリケーション EPG の下で、**[スタティック ポート (Static Ports)]** を右クリックし、**[スタティック EPG を PC、vPC、またはインターフェイスに展開 (Deploy Static EPG on PC, vPC, or Interface)]** を選択します。

b) パスのタイプ、ノード、およびQ-in-Qが有効になっているインターフェイスのパスを選択します。

c) **Port Encap (or Secondary VLAN for Micro-Seg)** フィールドで、 **QinQ** を選択し、EPG にマップされるトラフィックの外部および内部 VLAN タグを入力します。

d) [Submit] をクリックします。

ステップ 4 EPG をQ-in-Q モードが有効なノードに静的にリンクするには、次の手順を実行します:

a) アプリケーション EPG で、**Static Leafs** を右クリックして、**Statically Link With Node** を選択します。

b) [Node] フィールドで、リストから Q-in-Q が有効なスイッチを選択します。

c) [Encap] フィールドで、 **QinQ** を選択し、EPG の外部および内部 VLAN タグを入力します。

d) [Submit] をクリックします。

ステップ 5 EPG と静的エンドポイントを関連付けるには、次の手順を実行します:

a) アプリケーション EPG で、**Static EndPoints** を右クリックし、**Create Static EndPoint** を選択します。

b) インターフェイスの MAC アドレスを入力します。

c) パスのタイプ、ノード、およびQ-in-Qカプセル化が有効になっているインターフェイスのパスを選択します。

d) オプション。エンドポイントの IP アドレスを追加します。

e) **Encap** フィールドで、**QinQ** を選択し、外部および内部 VLAN タグを入力します。

f) [送信（Submit）] をクリックします。

NX-OS スタイル CLI を使用した Q-in-Q カプセル化リーフ インターフェイスへの EPG のマッピング

Q-in-Qカプセル化のインターフェイスを有効にし、EPGにインターフェイスを関連付けます。

始める前に

Q-in-Q モードに設定されているインターフェイスでマッピングされるテナント、アプリケーション プロファイル、アプリケーション EPG を作成します。

手順

	コマンドまたはアクション	目的
ステップ 1	**Configure** 例： `apic1# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**leaf** *number* 例： apic1(config)# leaf 101	設定するリーフを指定します。
ステップ 3	**interface ethernet***slot/port* 例： `apic1 (config-leaf)# interface ethernet 1/25`	設定するインターフェイスを指定します。
ステップ 4	**switchport mode dot1q-tunnel** doubleQtagPort 例： `apic1(config-leaf-if)# switchport mode dot1q-tunnel doubleQtagPort`	Q-in-Q カプセル化のインターフェイスを有効にします。
ステップ 5	**switchport trunkqinq outer-vlan***vlan-number* **inner-vlan** *vlan-number* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name* 例： `apic1(config-leaf-if)# switchport trunk qinq outer-vlan 202 inner-vlan 203 tenant tenant64 application AP64 epg EPG64`	インターフェイスを EPG に関連付けます。

例

次の例では、リーフ インターフェイス 101/1/25 で Q-in-Q カプセル化を有効にして（VLAN ID 201 外部および VLAN ID 203 内部）、EPG64 にインターフェイスを関連付けます。

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/25
apic1(config-leaf-if)#switchport mode dot1q-tunnel doubleQtagPort
apic1(config-leaf-if)# switchport trunk qinq outer-vlan 202 inner-vlan 203 tenant tenant64
 application AP64 epg EPG64
```

REST API を使用した Q-in-Q カプセル化対応インターフェイスに EPG をマッピングする

始める前に

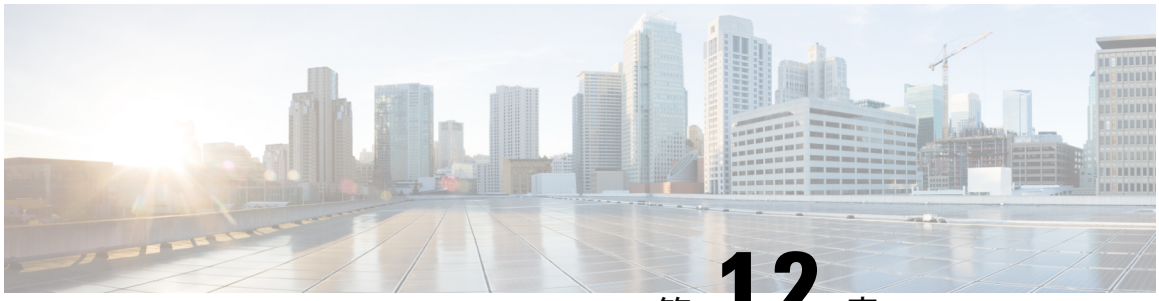
Q-in-Q モードに設定されたインターフェイスでマッピングされるテナント、アプリケーションプロファイル、およびアプリケーション EPG を作成します。

手順

次の例のように、Q-in-Q カプセル化のインターフェイスを有効にし、XML で EPG のインターフェイスを関連付けます。

例：

```
<polUni>
  <fvTenant dn="uni/tn-tenant64" name="tenant64">
    <fvCtx name="VRF64"/>
    <fvBD name="BD64_1">
      <fvRsCtx tnFvCtxName="VRF64"/>
      <fvSubnet ip="20.0.1.2/24"/>
    </fvBD>
    <fvAp name="AP64">
      <fvAEPg name="WEB7">
        <fvRsBd tnFvBDName="BD64_1"/>
        <fvRsQinqPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/25]"
encap="qinq-202-203"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```



第 12 章

ブレークアウト ポート

この章は、次の項で構成されています。

ダイナミック ブレークアウト ポートの設定

ブレークアウト ケーブルは非常に短いリンクに適しており、コスト効率の良いラック内および隣接ラック間を接続する方法を提供します。

ブレークアウトでは、40 ギガビット（Gb）ポートを独立して 4 分割し、10Gb または 100Gb ポートを独立した状態で論理 25 Gb ポートに 4 分割できます。

ブレークアウト ポートを設定する前に、次のケーブルのいずれかを使用して 40 Gb ポートを 4 つの 10 Gb ポートまたは 100 Gb ポートを 4 つの 25 Gb ポートに接続します。

- Cisco QSFP-4SFP10G
- Cisco QSFP-4SFP25G
- Cisco QSFP-4X10G-AOC
- MPO から、両端に QSFP-40G-SR4 および 4 X SFP-10G-SR を備えたブレークアウト スプリッタ ケーブルへ
- MPO から、両端に QSFP-100G-SR4-S と 4 X SFP-25G-SR-S を備えたブレークアウト スプリッタ ケーブルへ

（注） サポートされている光ファイバとケーブルについては、『*Cisco Optics-to-Device Compatibility Matrix*』を参照してください。

https://tmgmatrix.cisco.com/

40Gb から 10Gb へのダイナミック ブレークアウト機能は、次のスイッチのアクセス側ポートでサポートされます。

- N9K-C93180LC-EX
- N9K-C93180YC-FX
- N9K-C9336C-FX2
- N9K-C93360YC-FX2
- N9K-C93216TC-FX2
- N9K-C93108TC-FX3P（5.1(3) リリース以降）
- N9K-C93180YC-FX3（5.1(3) リリース以降）
- N9K-C93600CD-GX（5.1(3) リリース以降）
- N9K-C9364C-GX（5.1(3) リリース以降）

100 Gb から 25 Gb までの ブレークアウト機能は、次のスイッチのポートが面しているアクセスでサポートされています。

- N9K-C93180LC-EX
- N9K-C9336C-FX2
- N9K-C93180YC-FX
- N9K-C93360YC-FX2
- N9K-C93216TC-FX2
- N9K-C93108TC-FX3P（5.1(3) リリース以降）
- N9K-C93180YC-FX3（5.1(3) リリース以降）
- N9K-C93600CD-GX（5.1(3) リリース以降）
- N9K-C9364C-GX（5.1(3) リリース以降）

次に示すガイドラインおよび制限事項に従ってください。

- ブレイクアウトポートは、ダウンリンクと変換されたダウンリンクでのみサポートされます。
- 次のスイッチは、プロファイルされたQSFPポートでダイナミックブレークアウト（100Gbと 40Gb の両方）をサポートします。

- Cisco N9K-C93180YC-FX
- Cisco N9K-C93216TC-FX2
- Cisco N9K-C93360YC-FX2
- Cisco N9K-C93600CD-GX

  これは、ポート 1/25 ー 34 にのみ適用されます。ポートをダウンリンクに変換する場合、ポート 1/29 ー 34 はダイナミック ブレークアウトに使用できます。

- Cisco N9K-C9336C-FX2

  最大 34 のダイナミック ブレークアウトを構成できます。

- Cisco N9K-C9364C-GX（5.1(3) リリース以降）

  1/1 ～ 59 の奇数番号のプロファイリングされた QSFP ポートで、最大 30 のダイナミック ブレークアウトを設定できます。

- Cisco N9K-93600CD-GX（5.1(3) リリース以降）

  40/100G ポート x 24 から最大 12 のダイナミック ブレークアウトを設定でき、ポート 25 ～ 34 から最大 10 のダイナミック ブレークアウトを設定できます。ポートをダウンリンクに変換する場合、ポート 29 ー 34 はダイナミック ブレークアウトに使用できます。最後の 2 つのポート（ポート 35 と 36）は、ファブリック リンク用に予約されています。

- Cisco N9K-C9336C-FX2 スイッチは、ブレークアウト サブポートで LACP fast hello をサポートします。
- ブレークアウト ポートはCisco Application Policy Infrastructure Controller（APIC）接続には使用できません。
- ファスト リンク フェールオーバー ポリシーは、ダイナミック ブレークアウト機能と同一ポートではサポートされていません。
- ブレークアウトのサポートは、ポリシー モデルが使用されているその他のポート タイプと同じ方法で使用できます。
- ポートがダイナミック ブレークアウトに対して有効になっている場合、親ポートのその他のポート（モニタリング ポリシー以外）は無効になります。
- ポートがダイナミック ブレークアウトに対して有効になっている場合、親ポートのその他の EPG 展開が無効になります。
- ブレークアウト サブポートは、ブレークアウト ポリシー グループを使用してもこれ以上分割することはできません。
- ブレイクアウト サブポートはLACP をサポートします。デフォルトでは、「デフォルト」ポート チャネル メンバー ポリシーで定義された LACP 送信レート設定が使用されます。LACP 送信レートは、「デフォルト」ポート チャネル メンバー ポリシーを変更するか、各 PC/vPC インターフェイス ポリシー グループでのオーバーライド ポリシー グループを使用すれば、変更できます。

- ブレイクアウト サブポートを持つポート チャネルの LACP 送信レートを変更する必要がある場合、ブレークアウト サブポートを含むすべてのポート チャネルで同じ LACP 送信レート設定を使用することが必要です。オーバーライド ポリシーを設定して、次のように送信レートを設定できます。

  1. デフォルトのポート チャネル メンバー ポリシーを設定/変更して、Fast Transmit Rate を含めます（**[Fabric]** > **[Access Policies]** > **[Policies]** > **[Interface]** > **[Port Channel Member]**）。
  2. すべての PC/vPC インターフェイス ポリシー グループを設定して、上記のデフォルト ポート チャネル メンバー ポリシーをオーバーライド ポリシー グループに含めます（**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Policy Groups]** > **[PC/vPC Interface]**）。

- 次の注意事項および制約事項が Cisco N9K-C9364C-GX スイッチに適用されます。
  - 奇数番号のポート（行 1 および行 3）は、ブレークアウトをサポートします。隣接する偶数ポート（行 2 または行 4）は無効になります（「hw-disabled」）。これは、ポート 1/1 ー 60 に適用されます。
  - 最後の 2 つのポート（1/63 と 64）は、ファブリック リンク用に予約されています。
  - ポート 1/61 と 62 はダウンリンク ポートに変換できますが、ブレークアウトはサポートされていません。ブレークアウト ポートと 40/100G の非ブレークアウト ポートは、1/1 ー 4 または1/5 ー 8など、1/1 から始まる 4 つのポートのセットに混在させることはできません。

    たとえば、ポート 1/1 がブレークアウト対応の場合、ポート 1/3 はブレークアウト対応またはネイティブ 10Gで使用できます。ポート 1/3 が 40/100G の場合、error-disabled 状態になります。
  - ダウンリンクの最大数は、30 x 4ポート 10/25（ブレークアウト）+ 2 ポート（1/61 と 62）= 122ポートです。ポート 1/63 および 64 はファブリック リンク用に予約されており、1/2 ー 60の偶数番号のポートは error-disabled になっています。
  - このスイッチは、すべてのポートで 10G with QSA をサポートします。ネイティブ 10G には QSA が必要です。

- 次の注意事項および制約事項が Cisco N9K-93600CD-GX スイッチに適用されます。
  - 奇数番号のポート（行 1 のすべてのポート）はブレークアウトをサポートします。行 2 の偶数番号のポートは無効になります（「hw-disabled」）。これは、ポート 1 ー 24 にのみ適用されます。
  - ブレークアウトと 40/100G 非ブレークアウトは、1/1 ー 4 または 1/5 ー 8 など、1/1 から 1/24 までの 4 つのポートのセットに混在させることはできません。次に例を示します。
    - ポート 1/1 ー 24 の場合、セットごとに 4 つのポートを使用できます。

たとえば、ポート 1/1 がブレークアウト対応の場合、ポート 1/3 はブレークアウト対応またはネイティブ 10Gで使用できます。ポート 1/3 が 40/100G の場合、error-disabled 状態になります。

- ポート 1/25 ー 28 では、セットごとに 2 つのポートを使用できます。

  たとえば、ポート 1/25 がブレークアウト対応の場合でも、ポート 1/27 は 40/100G で使用できます。

- ダウンリンクの最大数は、12 x 4 ポート 10/25G（ブレークアウト）+ 10 x 4 ポート 10/25G（ブレークアウト）= 88 ポートです。ポート 35 および 36 はファブリックリンク用に予約されており、12 個のポートは無効になっています。
- このスイッチは、すべてのポートで 10G with QSA をサポートします。ネイティブ 10G には QSA が必要です。

ダウンリンクのダイナミック ブレークアウト ポートの注意事項と制約事項

40Gb から 10Gb へのダイナミック ブレークアウト機能は、次のスイッチのアクセス側ポートでサポートされます。

- N9K-C93180LC-EX
- N9K-C93180YC-FX
- N9K-C9336C-FX2
- N9K-C93360YC-FX2
- N9K-C93216TC-FX2
- N9K-C93108TC-FX3P
- N9K-C93180YC-FX3
- N9K-C93600CD-GX
- N9K-C9364C-GX

100Gb から 25Gb へのブレークアウト機能は、次のスイッチのアクセスポートでサポートされます。

- N9K-C93180LC-EX
- N9K-C9336C-FX2
- N9K-C93180YC-FX
- N9K-C93360YC-FX2
- N9K-C93216TC-FX2

- N9K-C93108TC-FX3P
- N9K-C93180YC-FX3
- N9K-C93600CD-GX
- N9K-C9364C-GX

400Gbから100Gbへのブレークアウト機能は、次のスイッチのアクセスポートでサポートされます。

- N9K-C9348D-GX2A
- N9K-C9364D-GX2A
- N9K-C9332D-GX2B
- N9K-C93600CD-GX
- N9K-C9316D-GX
- QDD-4X100G-FR-S、QDD-4X100G-LR-S、およびQDD-400G-DR4-Sオプティクスは、400Gbポートでサポートされます。
- 100Gb 速度のピア ノードは、次のオプティクスを使用できます。
  - QSFP-100G-FR-S
  - QSFP-100G-DR-S
  - QSFP-100G-LR-S

ブレークアウト ポートを設定する前に、次のいずれかのケーブルを使用して、40Gb ポートを 4 つの 10Gb ポートに、100Gb ポートを 4 つの 25 Gb ポートに、または 400Gb ポートを 4 つの 100Gb ポートに接続します。

- Cisco QSFP-4SFP10G
- Cisco QSFP-4SFP25G
- Cisco QSFP-4X10G-AOC
- MPO から、両端に QSFP-40G-SR4 および 4 X SFP-10G-SR を備えたブレークアウト スプリッタ ケーブルへ
- MPO から、両端に QSFP-100G-SR4-S と 4 X SFP-25G-SR-S を備えたブレークアウト スプリッタ ケーブルへ
- MPO から、両端にQDD-4X100G-FR-S、 QDD-4X100G-LR-S またはQDD-400G-DR4-S、およびQSFP-100G-FR-S x 4 またはQSFP-100G-DR-S x 4 を備えたブレークアウト スプリッタ ケーブルへ

（注） サポートされている光ファイバとケーブルについては、『*Cisco Optics-to-Device Compatibility Matrix*』を参照してください。

https://tmgmatrix.cisco.com/

次に示すガイドラインおよび制限事項に従ってください。

- ブレークアウトポートは、ダウンリンクと変換ダウンリンクの両方でサポートされます。
- 次のスイッチは、プロファイルされたQSFPポートでダイナミック ブレークアウト（100Gbと 40Gb の両方）をサポートします。
  - Cisco N9K-C93180YC-FX
  - Cisco N9K-C93216TC-FX2
  - Cisco N9K-C93360YC-FX2
  - Cisco N9K-C93600CD-GX

    これは、ポート 1/25 ー 34 にのみ適用されます。ポートをダウンリンクに変換する場合、ポート 1/29 ー 34 はダイナミック ブレークアウトに使用できます。
  - Cisco N9K-C9336C-FX2

    最大 34 のダイナミック ブレークアウトを構成できます。
  - Cisco N9K-C9364C-GX

    1/1 ～ 59 の奇数番号のプロファイリングされた QSFP ポートで、最大 30 のダイナミック ブレークアウトを設定できます。
  - Cisco N9K-93600CD-GX

    40/100G ポート x 24 から最大 12 のダイナミック ブレークアウトを設定でき、ポート 25 ～ 34 から最大 10 のダイナミック ブレークアウトを設定できます。ポートをダウンリンクに変換する場合、ポート 29 ー 34 はダイナミック ブレークアウトに使用できます。最後の 2 つのポート（ポート 35 と 36）は、ファブリック リンク用に予約されています。
- Cisco N9K-C9336C-FX2 スイッチは、ブレークアウト サブポートで LACP fast hello をサポートします。
- ブレークアウト ポートはCisco Application Policy Infrastructure Controller（APIC）接続には使用できません。
- ファスト リンク フェールオーバー ポリシーは、ダイナミック ブレークアウト機能と同一ポートではサポートされていません。
- ブレークアウトのサポートは、ポリシー モデルが使用されているその他のポート タイプと同じ方法で使用できます。

- ポートでダイナミック ブレークアウトが有効になっている場合、親ポート上の他のポリシー（モニタリング ポリシーを除く）は無効になります。
- ポートがダイナミック ブレークアウトに対して有効になっている場合、親ポートのその他の EPG 展開が無効になります。
- ブレークアウト サブポートは、ブレークアウト ポリシー グループを使用してもこれ以上分割することはできません。
- Cisco APIC ポリシーを使用して構成された、ダイナミック ブレークアウトまたは 400Gb ポートの 100Gb ポート x 4 への ブレークアウトは、QDD-4X100G-FR-S、QDD-4X100G-LR-S、および QDD-400G-DR4-S オプティクスでサポートされています。
- ブレイクアウトサブポートはLACPをサポートします。デフォルトでは、「デフォルト」ポート チャネル メンバー ポリシーで定義された LACP 送信レート設定が使用されます。LACP 送信レートは、「デフォルト」ポート チャネル メンバー ポリシーを変更するか、各 PC/vPC インターフェイス ポリシー グループでのオーバーライド ポリシー グループを使用すれば、変更できます。
- ブレイクアウト サブポートを持つポート チャネルの LACP 送信レートを変更する必要がある場合、ブレークアウト サブポートを含むすべてのポート チャネルで同じ LACP 送信レート設定を使用することが必要です。オーバーライドポリシーを設定して、次のように送信レートを設定できます。
  1. デフォルトのポート チャネル メンバー ポリシーを設定/変更して、Fast Transmit Rate を含めます（**[Fabric]** > **[Access Policies]** > **[Policies]** > **[Interface]** > **[Port Channel Member]**）。
  2. すべての PC/vPC インターフェイス ポリシー グループを設定して、上記のデフォルト ポート チャネル メンバー ポリシーをオーバーライド ポリシー グループに含めます（**[Fabric]** > **[Access Policies]** > **[Interfaces]** > **[Leaf Interfaces]** > **[Policy Groups]** > **[PC/vPC Interface]**）。
- 次の注意事項および制約事項が Cisco N9K-C9364C-GX スイッチに適用されます。
  - 奇数番号のポート（行 1 および行 3）は、ブレークアウトをサポートします。隣接する偶数ポート（行2または行4）は無効になります（「hw-disabled」）。これは、ポート 1/1 ー 60 に適用されます。
  - 最後の 2 つのポート（1/63 と 64）は、ファブリック リンク用に予約されています。
  - ポート 1/61 と 62 はダウンリンク ポートに変換できますが、ブレークアウトはサポートされていません。ブレークアウトポートと40/100Gの非ブレークアウトポートは、1/1 ー 4 または1/5 ー 8など、1/1 から始まる 4 つのポートのセットに混在させることはできません。

    たとえば、ポート 1/1 がブレークアウト対応の場合、ポート 1/3 はブレークアウト対応またはネイティブ 10Gで使用できます。ポート 1/3 が 40/100G の場合、error-disabled 状態になります。

- ダウンリンクの最大数は、30 x 4ポート 10/25（ブレークアウト）+ 2 ポート（1/61 と 62）= 122ポートです。ポート 1/63 および 64 はファブリック リンク用に予約されており、1/2 ー 60の偶数番号のポートは error-disabled になっています。
- このスイッチは、すべてのポートで 10G with QSA をサポートします。ネイティブ 10G には QSA が必要です。

- 次の注意事項および制約事項が Cisco N9K-93600CD-GX スイッチに適用されます。
  - 奇数番号のポート（行 1 のすべてのポート）はブレークアウトをサポートします。行 2 の偶数番号のポートは無効になります（「hw-disabled」）。これは、ポート 1 ー 24 にのみ適用されます。
  - ブレークアウトと 40/100G 非ブレークアウトは、1/1 ー 4 または 1/5 ー 8 など、1/1 から 1/24 までの 4 つのポートのセットに混在させることはできません。次に例を示します。
    - ポート 1/1 ー 24 の場合、セットごとに 4 つのポートを使用できます。

      たとえば、ポート 1/1 がブレークアウト対応の場合、ポート 1/3 はブレークアウト対応またはネイティブ 10Gで使用できます。ポート 1/3 が 40/100G の場合、error-disabled 状態になります。
    - ポート 1/25 ー 28 では、セットごとに 2 つのポートを使用できます。

      たとえば、ポート 1/25 がブレークアウト対応の場合でも、ポート 1/27 は 40/100G で使用できます。
  - ダウンリンクの最大数は、12 x 4 ポート 10/25G（ブレークアウト）+ 10 x 4 ポート 10/25G（ブレークアウト）= 88 ポートです。ポート 35 および 36 はファブリックリンク用に予約されており、12 個のポートは無効になっています。
  - このスイッチは、すべてのポートで 10G with QSA をサポートします。ネイティブ 10G には QSA が必要です。

ファブリック リンクの自動ブレークアウトポートの注意事項と制約事項

ブレークアウトがサポートされているラインカードにトランシーバを挿入すると、ポートは自動的にブレークアウトします。ブレークアウトを手動で設定する必要はありません。

400Gb から 100Gbへのブレークアウト機能は、次のライン カードのファブリック ポートでサポートされます。

- N9K-X9716D-GX と QDD-4X100G-FR-S トランシーバ

400Gb から 100Gb へのブレークアウト機能は、次のスイッチのファブリック ポートでサポートされます。

- N9K-C9348D-GX2A
- N9K-C9364D-GX2A
- N9K-C9332D-GX2B
- N9K-C93600CD-GX
- N9K-C9316D-GX
- QDD-4X100G-FR-S および QDD-4X100G-LR-S オプティクスは、400Gb ポートでサポートされます。
- 100Gb 速度のピア ノードは、次のトランシーバを使用できます。
  - QSFP-100G-DR-S
  - QSFP-100G-FR-S
  - QSFP-100G-LR-S

次のケーブルを使用して、400Gb ポートを 4 つの 100Gb ポートに接続します。

- MPO から、両端に QDD-4X100G-FR-S または QDD-4X100G-LR-S、および 4 x QSFP-100G-FR-S または 4 x QSFP-100G-DR-S を備えた 4xLC ブレークアウト スプリッタ ケーブル

ファブリック リンクでの 400G から 4x100G へのブレイクアウトに関する次のガイドラインと制限事項に従ってください。

- GX2 スイッチは、次のスイッチからスイッチへの接続をサポートします。
  - スパインスイッチからリーフスイッチへ
  - リーフスイッチからスパインスイッチへ
  - リーフスイッチからリーフスイッチ（多層）
- GX ライン カードは、次のスイッチからスイッチへの接続をサポートします。
  - スパインスイッチからリーフスイッチへ
- 次の構成はサポートされていません。
  - スパインスイッチからスパインスイッチ ブレークアウトへ
  - スパインスイッチから IPN ブレークアウトへ
- 特定のハードウェアおよびポートのブレークアウトをサポートしていないリリースにダウングレードすると、ブレークアウト ポートはブレークアウトされず、リンクがダウンします。スパインとリーフスイッチ間のすべての接続がブレークアウトのみの場合、ブレーク

アウトをサポートしていないリリースにダウングレードすると、リンクはダウンし、ノードはファブリック外になります。

- Cisco Nexus 9300 GX2 シリーズまたは Cisco N9K-X9716D-GX ライン カードでは、ラインカードの電源がオフの状態で光ファイバを交換しても、ポートは起動しません。次に例を示します。

  1. スロット 4 に Cisco N9K-X9716D-GX ラインカードがあり、4x100-FR-S トランシーバがポート（たとえば、ポート 8）に挿入されている。ポート 8は、4x100-FR-S トランシーバが挿入されたときに自動的にアクティブになる自動ブレークアウト機能により、4 つのポート（Eth4/8/1-4）に分割されます。
  2. スロット 4 のラインカードの電源をオフにします。
  3. ラインカードの電源がオフになっている間に、ポート 8 から 4x100G-FR-S 光ファイバを取り外し、4x100G-FR-S 以外の光ファイバを挿入します。
  4. スロット 4 のラインカードの電源をオンにします。ポート Eth4/8は、ピア エンドで互換性のあるポートとトランシーバの組み合わせに接続した後でも起動しません。

APIC GUI を使用したダイナミック ブレイクアウト ポートの設定

次の手順でリーフ インターフェイス プロファイルでブレークアウト リーフ ポートを設定し、スイッチとプロファイルを関連付け、サブ ポートを設定します。

（注） APIC GUI でブレイクアウトのためのポートを設定することもできます。**[Fabric]** > **[Inventory]** に移動し、[Topology] または [Pod] をクリックするか、 [Pod] を展開して、[Leaf] をクリックします。次に設定を有効にし、[Interface] タブをクリックします。

手順

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。
- 40GE または 100GE リーフ スイッチ ポートは、ダウンリンク ポートに Cisco ブレイク アウト ケーブルを接続します。

手順

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

ステップ 2 ナビゲーション ウィンドウで、**Interfaces** および **Leaf Interfaces** および **Profiles** を展開します。

ステップ 3 **Profiles** を右クリックして **Create Leaf Interface Profile** を選択します。

ステップ 4 名前と説明 (オプション) を入力して、**Interface Selectors** の [+] 記号をクリックします。

ステップ 5 次の手順を実行します。

a) **Access Port Selector** の名前と説明（オプション）を入力します。

b) **Interface IDs** フィールドで、ブレイクアウト ポートのスロットとポートを入力します。

c) **Interface Policy Group** フィールドで、下矢印をクリックして **Create Leaf Breakout Port Group** を選択します。

d) **Leaf Breakout Port Group** の名前 (およびオプションとして説明) を入力します。

e) **Breakout Map** フィールドで、**10g-4x** または **25g-4x** を選択します。

ブレイク アウトをサポートするスイッチについては、ダイナミック ブレークアウト ポートの設定 （271 ページ） を参照してください。

f) [Submit] をクリックします。

ステップ 6 ブレークアウト ポートを EPG に割り当てるには、次の手順を実行します。

メニュー バーで、**[Tenant]** > **[Application Profiles]** > **[Application EPG]** の順に選択します。[Application EPGs] を右クリックして [Create Application EPG] ダイアログボックスを開き、次の手順を実行します。

a) [Statically Link with Leaves/Paths] チェックボックスをオンにして、ダイアログボックスの [Leaves/Paths] タブにアクセスします。

b) 次のいずれかの手順を実行します。

オプション	説明
次のものに EPG を展開する場合、	次を実行します。
ノード	1. **Leaves** エリアを展開します。 2. [Node] ドロップダウン リストから、ノードを選択します。 3. **Encap** フィールドで、適切な VLAN を入力します。 4. (オプション) **Deployment Immediacy** ドロップダウンリストで、デフォルトの **On Demand** のままにするか、**Immediate**を選択します。 5. (オプション) [Mode] ドロップダウンリストで、デフォルトの [Trunk] のままにするか、別のモードを選択します。
ノード上のポート	1. **Paths** エリアを展開します。

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

ステップ 7 リーフ インターフェイス プロファイルをリーフ スイッチに関連付けるため、次の手順に従います。

a) **Switches** と **Leaf Switches**、および **Profiles** を展開します。
b) **Profiles** を右クリックして **Create Leaf Profiles** を選択します。
c) リーフ プロファイルの名前と、オプションとして説明を入力します。
d) **+** 記号 (**Leaf Selectors** エリア) をクリックします。
e) リーフ セレクタの名前と、オプションとして説明を入力します。
f) **Blocks** フィールドの下向き矢印をクリックして、ブレイクアウト インターフェイス プロファイルと関連付けるスイッチを選択します。
g) **Policy Group** フィールドの下向き矢印をクリックし、**Create Access Switch Policy Group** を選択します。
h) アクセス スイッチ ポリシー グループの名前と、オプションとして説明を入力します。
i) オプション。その他のポリシーを有効にします。
j) [Submit] をクリックします。
k) **Update** をクリックします。
l) [Next] をクリックします。
m) **Associations Interface Selector Profiles** エリアで、ブレイクアウト ポート用に以前に作成したインターフェイス セレクタ プロファイルを選択します。
n) **Finish** をクリックします。

ステップ 8 ブレイクアウト ポートが 4 つのサブ ポートに分割されたことを確認するために、次の手順に従います:

a) メニュー バーで、**Fabric** > **Inventory** をクリックします。
b) ナビゲーションバーで、ブレイクアウトポートがあるポッドとリーフをクリックします。
c) **Interfaces** および **Physical Interfaces** を展開します。
ブレイクアウト ポートが設定された場所に 4 つのポートが表示されます。たとえば、1/10 をブレイクアウト ポートとして設定した場合、次のように表示されます:

- **eth1/10/1**

- **eth1/10/2**
- **eth1/10/3**
- **eth1/10/4**

ステップ 9 サブ ポートを設定するには、次の手順を実行します:

a) メニュー バーで、**[Fabric]** > **[Access Policies]** をクリックします。
b) ナビゲーションバーで、**Interfaces**、**Leaf Interfaces**、**Profiles**、および前に作成したブレイクアウト リーフ インターフェイス プロファイルを展開します。

ブレークアウト ケーブルが付属するポートのセレクタが表示されます。既存のポートのセレクタでサブポート ブロックを定義する代わりに、新しいアクセス ポート セレクタで定義する必要があります。

c) ナビゲーションバーで、上位レベルのインターフェイス プロファイルを右クリックし、**[Create Access Port Selector]** を選択します。
d) **[Name]** フィールドで、サブ ポートの名前を入力します。
e) **Interface IDs** フィールドに、4 つのサブ ポートの ID を、`1/10/1-4` のフォーマットで入力します。
f) **[Interface Policy Group]** フィールドで、**[Create Leaf Access Port Policy Group]** を選択します。
g) **[送信（Submit）]** をクリックします。`

ステップ 10 AAEP をポートにリンクする個々のインターフェイスにポリシーグループを適用するには、次の手順を実行します。

a) **[Name]** フィールドに、リーフ アクセス ポートのグループ ポリシー名を入力します。
b) **[Link Level Policy]** フィールドで、**[link-level_auto]** を選択します。
c) **[CDP Policy]** フィールドで、**[cdp_enabled]** を選択します。
d) **[LLDP Policy]** フィールドで、**[default]** を選択します。
e) **[Attached Entity Profile]** フィールドで、ポリシー グループにアタッチする AAEP プロファイルを選択します。
f) [送信（Submit）] をクリックします。

NX-OS スタイルの CLI を使用したダイナミック ブレイクアウト ポートの設定

ブレークアウトポートを設定、設定を確認および NX-OS スタイル CLI を使用してサブ ポートで、EPG を設定するには、次の手順を使用します。

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。
- 40GE または 100GE リーフ スイッチ ポートは、ダウンリンク ポートに Cisco ブレイク アウト ケーブルを接続します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `apic1# configure`	コンフィギュレーション モードに入ります。
ステップ 2	**leaf** *ID* 例： `apic1(config)# leaf 101`	ブレークアウトポートが配置され、リーフ configuration mode(設定モード、コンフィギュレーション モード) を開始リーフ スイッチを選択します。
ステップ 3	**interface ethernet***slot/port* 例： `apic1(config-leaf)# interface ethernet 1/16`	40 ギガビット イーサネット (GE) ブレークアウトポートとして有効にするインターフェイスを識別します。
ステップ 4	**breakout10g-4x**|**25g-4x** 例： `apic1(config-leaf-if)# breakout 10g-4x`	ブレイク アウトを選択したインターフェイスを有効にします。 （注） ダイナミック ブレイク アウトポート機能は、スイッチのサポートを参照してください。 ダイナミック ブレークアウト ポートの設定 （271 ページ） 。

ステップ 5	**show run**
	例：
	`apic1(config-leaf-if)# show run`
	`# Command: show running-config leaf 101 interface ethernet 1 / 16`
	`# Time: Fri Dec 2 18:13:39 2016`
	`leaf 101`
	`interface ethernet 1/16`
	`breakout 10g-4x`
		インターフェイスの実行コンフィギュレーションを表示することによって、設定を確認し、グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
	`apic1(config-leaf-if)# exit` `apic1(config-leaf)# exit`	
ステップ 6	**tenant** *tenant-name* 例： `apic1(config)# tenant tenant64`	選択またはブレイクアウトポートで消費され、テナント configuration mode(設定モード、コンフィギュレーション モード) を開始するテナントを作成します。

ステップ 7	**vrf context** *vrf-name*
	例：
	`apic1(config-tenant)# vrf context vrf64`
	`apic1(config-tenant-vrf)# exit`
		作成またはテナントに関連付けられている Virtual Routing and Forwarding (VRF) インスタンスを識別し、configuration mode(設定モード、コンフィギュレーション モード) を終了します。

ステップ 8	**bridge-domain** *bridge-domain-name* 例： `apic1(config-tenant)# bridge-domain bd64`	作成またはテナントに関連付けられているブリッジ ドメインを識別し、BD configuration mode(設定モード、コンフィギュレーション モード) を開始します。

ステップ 9	**vrf member** *vrf-name*
	例：
	`apic1(config-tenant-bd)# vrf member vrf64`
	`apic1(config-tenant-bd)# exit`
		ブリッジドメイン、VRF の関連付け、configuration mode(設定モード、コンフィギュレーション モード) を終了します。

ステップ 10	**application** *application-profile-name* 例： `apic1(config-tenant)# application app64`	作成またはテナントと EPG に関連付けられているアプリケーションプロファイルを識別します。
ステップ 11	**epg** *epg-name* 例： `apic1(config-tenant)# epg epg64`	作成または EPG を識別し、EPG configuration mode(設定モード、コンフィギュレーション モード) に入力します。

ステップ 12	**bridge-domain member** *bridge-domain-name*
	例：
	`apic1(config-tenant-app-epg)# bridge-domain member bd64`
	`apic1(config-tenant-app-epg)# exit`
	`apic1(config-tenant-app)# exit`
	`apic1(config-tenant)# exit`
		EPG をブリッジドメインに関連付け、グローバル設定モードをに戻ります。
		たとえば、必要に応じて、サブポートを設定コマンドを使用して、速度リーフインターフェイス モードでサブポートを設定します。

	コマンドまたはアクション	目的

ステップ 13	**leaf** *leaf-name*
	例：
	`apic1(config)# leaf 1017`
	`apic1(config-leaf)# interface ethernet 1/13`
	`apic1(config-leaf-if)# vlan-domain member dom1`
	`apic1(config-leaf-if)# switchport trunk allowed vlan 20 tenant t1 application AP1 epg EPG1`
	（注） 上の例に示した vlan-domain コマンドと vlan-domain member コマンドは、ポートに EPG を導入するための前提条件です。
		EPG をブレークアウト ポートに関連付けます。

ステップ 14	**speed** *interface-speed*
	例：
	`apic1(config)# leaf 101`
	`apic1(config-leaf)# interface ethernet 1/16/1`
	`apic1(config-leaf-if)# speed 10G`
	`apic1(config-leaf-if)# exit`
		リーフ インターフェイス モードを開始し、[インターフェイスの速度を設定 configuration mode(設定モード、コンフィギュレーション モード) を終了します。

ステップ 15	**show run** 例： `apic1(config-leaf)# show run`	サブ ポートを設定した後にリーフ configuration mode(設定モード、コンフィギュレーション モード) で次のコマンドを入力して、サブ ポートの詳細が表示されます。

サブ ポート 1/16/1、2/1/16、1/16/3 および 4/1/16 ブレイク アウトを有効になっているリーフ インターフェイス 1/16 で 101 上のポートを確認します。

例

この例では、ブレイク アウト ポートで設定します。

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/16
apic1(config-leaf-if)# breakout 10g-4x
```

この例では、サブインターフェイス ポートの EPG で設定します。

```
apic1(config)# tenant tenant64
apic1(config-tenant)# vrf context vrf64
apic1(config-tenant-vrf)# exit
apic1(config-tenant)# bridge-domain bd64
```

```
apic1(config-tenant-bd)# vrf member vrf64
apic1(config-tenant-bd)# exit
apic1(config-tenant)# application app64
apic1(config-tenant-app)# epg epg64
apic1(config-tenant-app-epg)# bridge-domain member bd64
apic1(config-tenant-app-epg)# end
```

この例では、10 G に、ブレイク アウトの速度サブ ポートを設定します。

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/16/1
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit

apic1(config-leaf)# interface ethernet 1/16/2
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/16/3
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/16/4
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit
```

この例では、リーフ 101、インターフェイス 1/16 に接続されている、4 つのアシスタント的なポートを示します。

```
apic1#(config-leaf)# show run
# Command: show running-config leaf 101
# Time: Fri Dec  2 00:51:08 2016
  leaf 101
    interface ethernet 1/16/1
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16/2
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16/3
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16/4
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16
      breakout 10g-4x
      exit
    interface vfc 1/16
```

REST API を使用したダイナミック ブレークアウト ポートの設定

次の手順でリーフ インターフェイス プロファイルでブレークアウト リーフ ポートを設定し、スイッチとプロファイルを関連付け、サブ ポートを設定します。

ブレークアウト機能のスイッチ サポートについては、ダイナミック ブレークアウト ポートの設定 （271 ページ） を参照してください。

手順

始める前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要なファブリック インフラストラクチャ設定を作成できる APIC ファブリック管理者アカウントが使用可能であること。
- ターゲット リーフ スイッチが ACI ファブリックに登録され、使用可能であること。
- 40GE または 100GE リーフ スイッチ ポートは、ダウンリンク ポートに Cisco ブレークアウト ケーブルを接続します。

手順

ステップ 1 次の例のように、JSON でブレークアウトポートのブレークアウト ポリシー グループを設定します。

例：

この例では、ポート セレクタの下の唯一のポート 44 でインターフェイス プロファイル「brkout44」を作成します。ポート セレクタは、ブレークアウト ポリシー グループ「new-brkoutPol」を指しています。

```
{
            "infraAccPortP": {
              "attributes": {
                "dn":"uni/infra/accportprof-brkout44",
                "name":"brkout44",
                "rn":"accportprof-brkout44",
                "status":"created,modified"
                },
                "children":[ {
                     "infraHPortS": {
                        "attributes": {

"dn":"uni/infra/accportprof-brkout44/hports-new-brekoutPol-typ-range",
                           "name":"new-brkoutPol",
                           "rn":"hports-new-brkoutPol-typ-range",
                           "status":"created,modified"
```

```
                        },
                        "children":[ {
                           "infraPortBlk": {
                             "attributes": {

"dn":"uni/infra/accportprof-brkout44/hports-new-brkoutPol-typ-range/portblk-block2",
                                "fromPort":"44",
                                "toPort":"44",
                                "name":"block2",
                                "rn":"portblk-block2",
                                "status":"created,modified"
                              },
                              "children":[] }
                     }, {
               "infraRsAccBaseGrp": {
                 "attributes":{
                    "tDn":"uni/infra/funcprof/brkoutportgrp-new-brkoutPol",
                    "status":"created,modified"
                 },
                 "children":[]
               }
             }
           ]
         }
      }
   ]
 }
}
```

ステップ 2 次の例のように JSON で以前作成された新しいスイッチ プロファイルを作成し、ポート プロファイルに関連付けます。

例：

この例で、唯一のノードとしてスイッチ 1017 で新しいプロファイル「'leaf1017」を作成します。この新しいスイッチプロファイルを、上で作成されたポートプロファイル「brkout44」に関連付けます。この後、スイッチ 1017 上のポート 44 には 4 個のサブ ポートがあります。

例：

```
{
   "infraNodeP": {
     "attributes": {
       "dn":"uni/infra/nprof-leaf1017",
       "name":"leaf1017","rn":"nprof-leaf1017",
       "status":"created,modified"
      },
      "children": [ {
        "infraLeafS": {
          "attributes": {
            "dn":"uni/infra/nprof-leaf1017/leaves-1017-typ-range",
            "type":"range",
            "name":"1017",
            "rn":"leaves-1017-typ-range",
            "status":"created"
      },
      "children": [ {
        "infraNodeBlk": {
          "attributes": {
           "dn":"uni/infra/nprof-leaf1017/leaves-1017-typ-range/nodeblk-102bf7dc60e63f7e",

            "from_":"1017","to_":"1017",
            "name":"102bf7dc60e63f7e",
```

```
            "rn":"nodeblk-102bf7dc60e63f7e",
            "status":"created"
           },
           "children": [] }
          }
        ]
       }
     }, {
       "infraRsAccPortP": {
         "attributes": {
           "tDn":"uni/infra/accportprof-brkout44",
           "status":"created,modified"
      },
      "children": [] }
       }
     ]
   }
}
```

ステップ 3 サブポートを設定します。

例：

この例では、スイッチ 1017 のサポート 1/44/1、2/1/44、1/44/3、1/44/4 を設定します。下の例では、インターフェイス 3/1/44 を設定します。また、`infraPortBlk` オブジェクトの代わりに `infraSubPortBlk` オブジェクトを作成します。

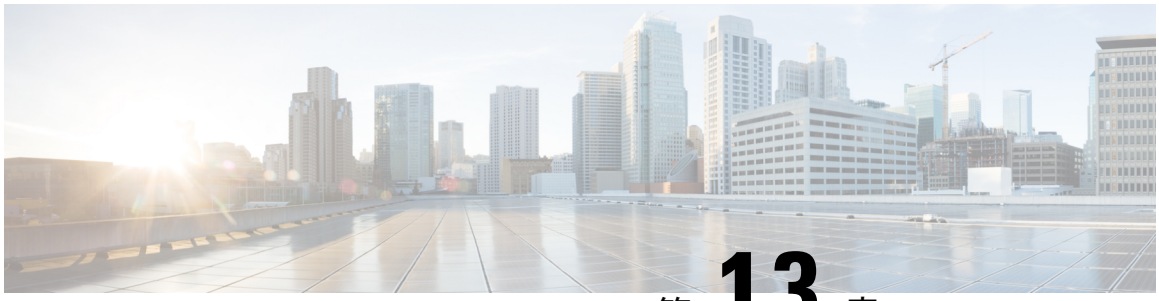
```
{
   "infraAccPortP": {
     "attributes": {
       "dn":"uni/infra/accportprof-brkout44",
       "name":"brkouttest1",
       "rn":"accportprof-brkout44",
       "status":"created,modified"
     },
     "children": [{
       "infraHPortS": {
         "attributes": {
           "dn":"uni/infra/accportprof-brkout44/hports-sel1-typ-range",
           "name":"sel1",
           "rn":"hports-sel1-typ-range",
           "status":"created,modified"
         },
         "children": [{
           "infraSubPortBlk": {
             "attributes": {

"dn":"uni/infra/accportprof-brkout44/hports-sel1-typ-range/subportblk-block2",
              "fromPort":"44",
              "toPort":"44",
              "fromSubPort":"3",
              "toSubPort":"3",
              "name":"block2",
              "rn":"subportblk-block2",
              "status":"created"
             },
             "children":[]}
             },
            {
              "infraRsAccBaseGrp": {
                  "attributes": {
                    "tDn":"uni/infra/funcprof/accportgrp-p1",
                    "status":"created,modified"
                  },
```

```
                "children":[]}
                }
              ]
            }
          }
        ]
      }
    }
```

ステップ 4 特定のポート上に EPG を導入します。

例：

```
<fvTenant name="<tenant_name>" dn="uni/tn-test1" >
    <fvCtx name="<network_name>" pcEnfPref="enforced" knwMcastAct="permit"/>
    <fvBD name="<bridge_domain_name>" unkMcastAct="flood" >
        <fvRsCtx tnFvCtxName="<network_name>"/>
    </fvBD>
    <fvAp name="<application_profile>" >
        <fvAEPg name="<epg_name>" >
           <fvRsPathAtt tDn="topology/pod-1/paths-1017/pathep-[eth1/13]" mode="regular"
 instrImedcy="immediate" encap="vlan-20"/>
        </fvAEPg>
    </fvAp>
</fvTenant>
```



第 13 章

プロキシ ARP

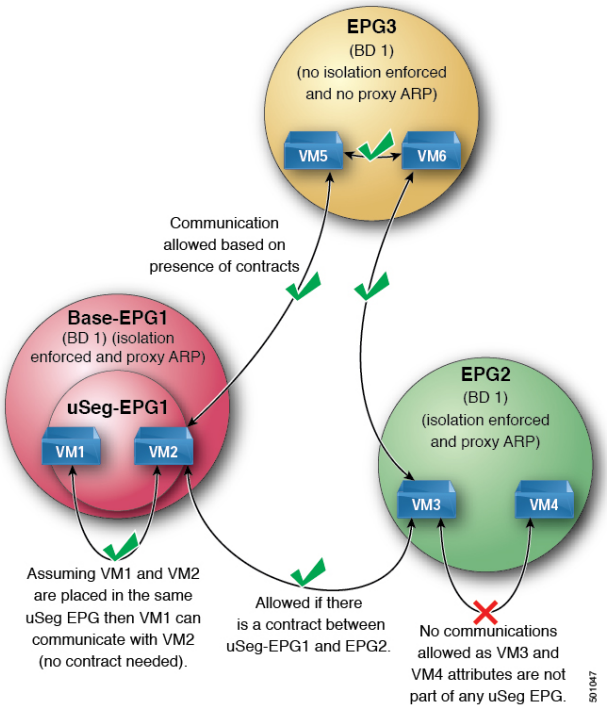
この章は、次の内容で構成されています。

プロキシ ARP について

Cisco ACI のプロキシ ARP は、ネットワークまたはサブネット内のエンドポイントが、別のエンドポイントの MAC アドレスを知らなくても、そのエンドポイントと通信できるようにします。プロキシ ARP はトラフィックの宛先場所を知っており、代わりに、最終的な宛先として自身の MAC アドレスを提供します。

プロキシ ARP を有効にするには、EPG 内エンドポイント分離を EPG で有効にする必要があります。詳細については、次の図を参照してください。EPG 内エンドポイント分離と Cisco ACI の詳細については、「*Cisco ACI* 仮想化ガイド」を参照してください。

図 *33* : プロキシ *ARP* および *Cisco APIC*



Cisco ACI ファブリック内のプロキシ ARP は従来のプロキシ ARP とは異なります。通信プロセスの例として、プロキシ ARP が EPG で有効になっているとき、エンドポイント A が ARP 要求をエンドポイント B に送信し、エンドポイント B がファブリック内で学習される場合、エンドポイント A はブリッジ ドメイン（BD）MAC からプロキシ ARP 応答を受信します。エンドポイント A が B、エンドポイントの ARP 要求を送信し、エンドポイント B はすでに ACI ファブリック内で学習しない場合は、ファブリックはプロキシ ARP の BD 内で要求を送信します。エンドポイント B は、ファブリックに戻る要求、このプロキシ ARP に応答します。この時点では、ファブリックはプロキシ ARP エンドポイント A への応答を送信しませんが、エンドポイント B は、ファブリック内で学習します。エンドポイント A は、エンドポイント B に別の ARP 要求を送信する場合、ファブリックはプロキシ ARP 応答から送信 BD mac です。

次の例ではプロキシ ARP 解像度がクライアント VM1 と VM2 間の通信の手順します。

1. VM2 通信を VM1 が必要です。

図 *34 : VM2* 通信を *VM1* が必要です。

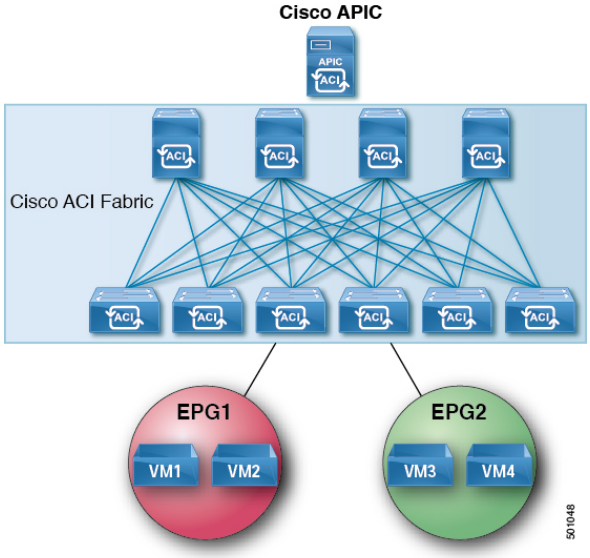


表 *6 : ARP* 表の説明

デバイス	状態
VM1	IP = * MAC = *
ACI ファブリック	IP = * MAC = *
VM2	IP = * MAC = *

2. VM1 は、ブロードキャスト MAC アドレスとともに ARP 要求を VM2 に送信します。

図 *35 : VM1* はブロードキャスト *MAC* アドレスとともに *ARP* 要求を *VM2* に送信します

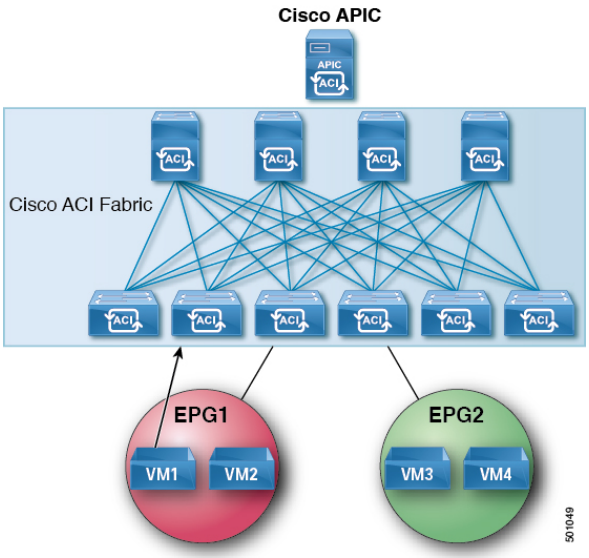


表 *7 : ARP* 表の説明

デバイス	状態
VM1	IP = VM2 IP; MAC = ?
ACI ファブリック	IP = VM1 IP; MAC = VM1 MAC
VM2	IP = * MAC = *

3. ACI ファブリックは、ブリッジ ドメイン（BD）内のプロキシ ARP 要求をフラッディングします。

図 *36 : ACI* ファブリックは *BD* 内のプロキシ *ARP* 要求をフラッディングします

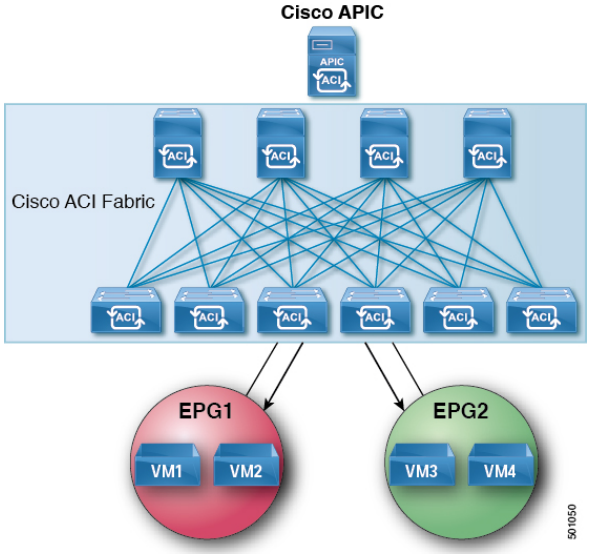


表 *8 : ARP* 表の説明

デバイス	状態
VM1	IP = VM2 IP; MAC = ?
ACI ファブリック	IP = VM1 IP; MAC = VM1 MAC
VM2	IP = VM1 IP; MAC = BD MAC

4. VM2 は、ARP 応答を ACI ファブリックに送信します。

図 *37 : VM2* は *ARP* 応答を *ACI* ファブリックに送信します

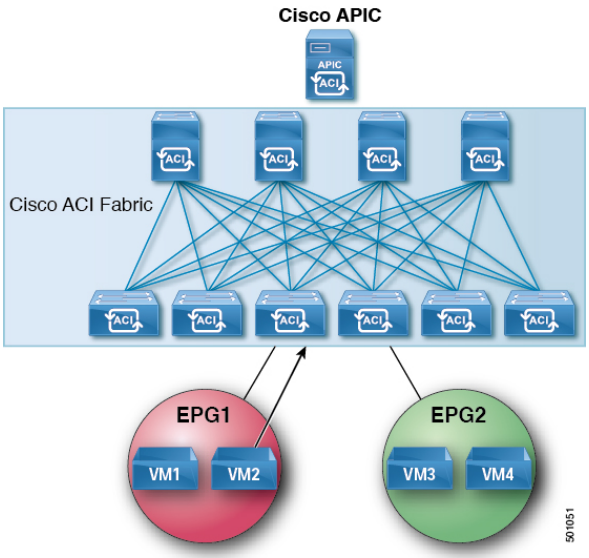


表 *9 : ARP* 表の説明

デバイス	状態
VM1	IP = VM2 IP; MAC = ?
ACI ファブリック	IP = VM1 IP; MAC = VM1 MAC
VM2	IP = VM1 IP; MAC = BD MAC

5. VM2 が学習されます。

図 *38 : VM2* が学習されます

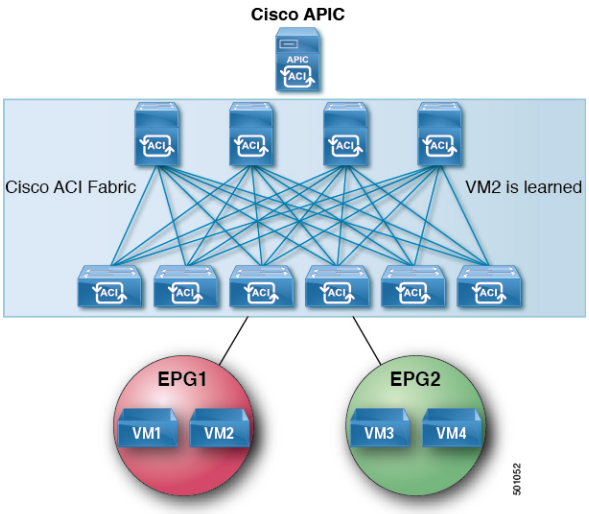


表 *10* : *ARP* 表の説明

デバイス	状態
VM1	IP = VM2 IP; MAC = ?
ACI ファブリック	IP = VM1 IP; MAC = VM1 MAC IP = VM2 IP; MAC = VM2 MAC
VM2	IP = VM1 IP; MAC = BD MAC

6. VM1 は、ブロードキャスト MAC アドレスとともに ARP 要求を VM2 に送信します。

図 *39* : *VM1* はブロードキャスト *MAC* アドレスとともに *ARP* 要求を *VM2* に送信します

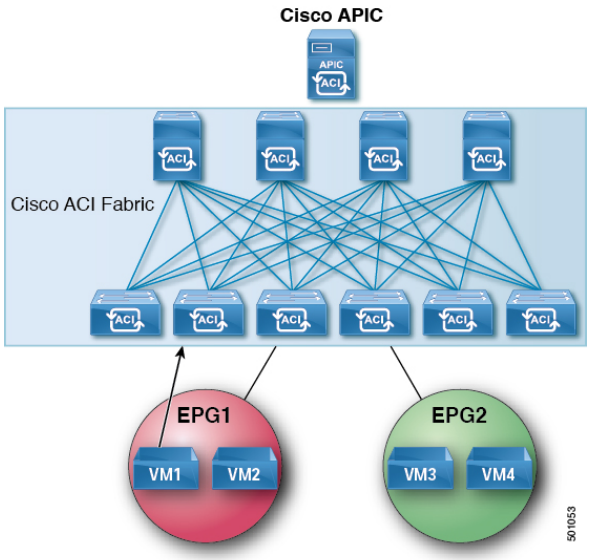


表 *11* : *ARP* 表の説明

デバイス	状態
VM1	IP = VM2 IP MAC = ?
ACI ファブリック	IP = VM1 IP; MAC = VM1 MAC IP = VM2 IP; MAC = VM2 MAC
VM2	IP = VM1 IP; MAC = BD MAC

7. ACI ファブリックは、プロキシ ARP VM1 への応答を送信します。

図 *40 : ACI* ファブリック *VM1* にプロキシ *ARP* 応答を送信します。

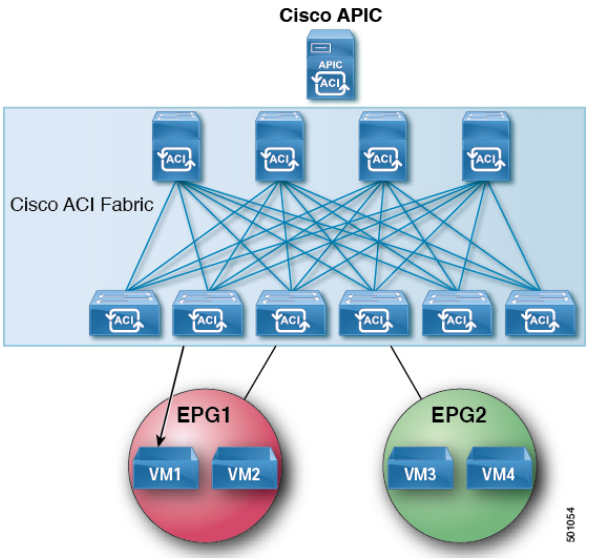


表 *12 : ARP* 表の説明

デバイス	状態
VM1	IP = VM2 IP;MAC = BD MAC
ACI ファブリック	IP = VM1 IP; MAC = VM1 MAC IP = VM2 IP; MAC = VM2 MAC
VM2	IP = VM1 IP; MAC = BD MAC

ガイドラインと制約事項

プロキシ ARP を使用すると、次のガイドラインと制限事項を考慮してください。

- プロキシ ARP は、隔離 Epg でのみサポートされます。EPG が隔離ではない場合、障害が発生します。プロキシ ARP が有効になっていると隔離 Epg 内で発生する通信では、uSeg Epg を設定する必要があります。たとえば、隔離の EPG 内で別の IP アドレスを持つ複数の Vm がある可能性があり、これらの Vm の IP address range(IP アドレス範囲、IP アドレスの範囲) に一致する IP の属性を持つ uSeg EPG を設定することができます。
- 隔離されたエンドポイントを通常のエンドポイントと、定期的なエンドポイントを隔離のエンドポイントからの ARP 要求には、プロキシ ARP は使用しないでください。このような場合は、エンドポイントは、接続先の Vm の実際の MAC アドレスを使用して通信します。

プロキシ ARP がサポートされている組み合わせ

次のプロキシ ARP 表では、サポートされている組み合わせを示します。

ARP 送信元/宛先	定期的な EPG	プロキシ ARP に適用される EPG の隔離
定期的な EPG	ARP	ARP
プロキシ ARP に適用される EPG の隔離	ARP	プロキシ ARP

拡張 GUI を使用したプロキシ ARP の設定

始める前に

- 適切なテナント、VRF、ブリッジ ドメイン、アプリケーションプロファイルおよび EPG を作成する必要があります。
- プロキシ ARP が有効にするのにが EPG で内通 EPG の分離を有効にする必要があります。

手順

ステップ 1 メニュー バーで、**Tenant** > **Tenant_name** をクリックします。

ステップ 2 **ナビゲーション**] ペインで、展開、 **Tenant_name** > **アプリケーション プロファイル** > **Application_Profile_name** > **アプリケーション Epg** 、右クリックして **アプリケーション EPG の作成** を実行するダイアログボックス、次のアクションに、 **アプリケーション EPG の作成** ダイアログボックス:

a) **Name** フィールドに EPG 名を追加します。

ステップ 3 **Intra EPG Isolation** フィールドで、**Enforced** を選択します。
内通 EPG 分離が適用されるときに、 **転送制御** フィールドは使用可能になります。

ステップ 4 **Forwarding Control** フィールドで、**proxy-arp** チェック ボックスをオンにします。
proxy-arp が有効になります。

ステップ 5 **Bridge Domain** フィールドで、ドロップダウン リストから、関連付ける適切なブリッジ ドメインを選択します。

ステップ 6 必要に応じて、ダイアログボックスの残りのフィールドを選択し、をクリックして **終了** 。

プロキシ ARP は、Cisco NX-OS スタイル CLI を使用しての設定

始める前に

- 適切なテナント、VRF、ブリッジ ドメイン、アプリケーションプロファイルおよび EPG を作成する必要があります。
- プロキシ ARP が有効にするのにが EPG で内通 EPG の分離を有効にする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `apic1# configure`	コンフィギュレーション モードに入ります。
ステップ 2	**tenant** *tenant-name* 例： `apic1(config)# tenant Tenant1`	テナント コンフィギュレーション モードを開始します。
ステップ 3	**application** *application-profile-name* 例： `apic1(config-tenant)# application Tenant1-App`	アプリケーション プロファイルを作成し、アプリケーション モードを開始します。
ステップ 4	**epg** *application-profile-EPG-name* 例： `apic1(config-tenant-app)# epg Tenant1-epg1`	EPG を作成し、EPG モードに入ります。
ステップ 5	**proxy-arp enable** 例： `apic1(config-tenant-app-epg)# proxy-arp enable`	プロキシ ARP を有効にします。 （注） プロキシ arp をディセーブルにできます、 **no プロキシ arp** コマンド。
ステップ 6	**exit** 例： `apic1(config-tenant-app-epg)# exit`	ポート アプリケーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**exit** 例： `apic1(config-tenant-app)# exit`	テナント コンフィギュレーション モードに戻ります。
ステップ 8	**exit** 例： `apic1(config-tenant)# exit`	グローバル コンフィギュレーション モードに戻ります。

例

次に、プロキシ ARP を設定する例を示します。

```
apic1# conf t
apic1(config)# tenant Tenant1
apic1(config-tenant)# application Tenant1-App
apic1(config-tenant-app)# epg Tenant1-epg1
apic1(config-tenant-app-epg)# proxy-arp enable
apic1(config-tenant-app-epg)#
apic1(config-tenant)#
```

プロキシ ARP REST API を使用しての設定

始める前に

- プロキシ ARP が有効にするのにが EPG で内通 EPG の分離を有効にする必要があります。

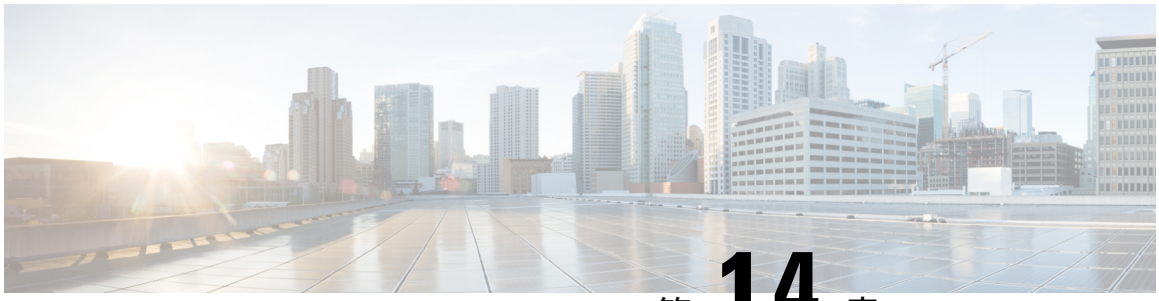
手順

プロキシ ARP を設定します。

例：

```
<polUni>
  <fvTenant name="Tenant1" status="">
    <fvCtx name="EngNet"/>
    <!-- bridge domain -->
    <fvBD name="BD1">
        <fvRsCtx tnFvCtxName="EngNet" />
        <fvSubnet ip="1.1.1.1/24"/>
    </fvBD>
    <fvAp name="Tenant1_app">
        <fvAEPg  name="Tenant1_epg" pcEnfPref-"enforced" fwdCtrl="proxy-arp">
            <fvRsBd tnFvBDName="BD1" />
            <fvRsDomAtt tDn="uni/vmmp-VMware/dom-dom9"/>
        </fvAEPg>
    </fvAp>
```

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



第 14 章

トラフィック ストーム制御

この章は、次の項で構成されています。

トラフィック ストーム制御について

トラフィック ストームは、パケットが LAN でフラッディングする場合に発生するもので、過剰なトラフィックを生成し、ネットワークのパフォーマンスを低下させます。トラフィック ストーム制御ポリシーを使用すると、物理インターフェイス上におけるブロードキャスト、未知のマルチキャスト、または未知のユニキャストのトラフィック ストームによって、レイヤ 2 ポート経由の通信が妨害されるのを防ぐことができます。

デフォルトでは、ストーム制御はACIファブリックでは有効になっていません。ACIブリッジドメイン（BD）レイヤ 2 の未知のユニキャストのフラッディングは BD 内でデフォルトで有効になっていますが、管理者が無効にすることができます。その場合、ストーム制御ポリシーはブロードキャストと未知のマルチキャストのトラフィックにのみ適用されます。レイヤ2の未知のユニキャストのフラッディングが BD で有効になっている場合、ストーム制御ポリシーは、ブロードキャストと未知のマルチキャストのトラフィックに加えて、レイヤ2の未知のユニキャストのフラッディングに適用されます。

トラフィック ストーム制御（トラフィック抑制ともいいます）を使用すると、着信するブロードキャスト、マルチキャスト、未知のユニキャストのトラフィックのレベルを1秒間隔でモニタできます。この間に、トラフィック レベル（ポートで使用可能な合計帯域幅のパーセンテージ、または特定のポートで許可される1秒あたりの最大パケット数として表されます）が、設定したトラフィック ストーム制御レベルと比較されます。入力トラフィックが、ポートに設定したトラフィック ストーム制御レベルに到達すると、トラフィック ストーム制御機能によっ

てそのインターバルが終了するまでトラフィックがドロップされます。管理者は、ストーム制御しきい値を超えたときにエラーを発生させるようにモニタリング ポリシーを設定できます。

ストーム制御の注意事項と制約事項

以下のガイドラインと制約事項に従って、トラフィック ストーム制御レベルを設定してください。

- 通常、ファブリック管理者は以下のインターフェイスのファブリック アクセス ポリシーでストーム制御を設定します。
  - 標準トランク インターフェイス。
  - 単一リーフ スイッチ上のダイレクト ポート チャネル。
  - バーチャル ポート チャネル（2 つのリーフ スイッチ上のポート チャネル）。
- リリース 4.2(1) 以降では、ストーム制御のしきい値に達した場合に、次の制約事項に従って、SNMP トラップをCisco Application Centric Infrastructure（ACI）からトリガーできるようになりました。
  - ストーム制御に関連するアクションには、ドロップとシャットダウンの 2 つがあります。シャットダウン アクションでは、インターフェイス トラップが発生しますが、ストームがアクティブまたはクリアであることを示すためのストーム制御トラップは、シャットダウンアクションによっては決定されません。したがって、ポリシーでシャットダウンアクションが設定されているストーム制御トラップは無視する必要があります。
  - ストーム制御ポリシーがオンの状態でポートがフラップすると、統計情報の収集時にクリア トラップとアクティブ トラップが一緒に表示されます。通常、クリア トラップとアクティブ トラップは一緒に表示されませんが、この場合は予期される動作です。
- ポート チャネルおよびバーチャル ポート チャネルでは、ストーム制御値（1 秒あたりのパケット数またはパーセンテージ）はポートチャネルのすべての個別メンバーに適用されます。ポートチャネルのメンバーであるインターフェイスには、ストーム制御を設定しないでください。

（注） Cisco Application Policy Infrastructure Controller（APIC）リリース 1.3(1) およびスイッチ リリース 11.3(1) 以降のスイッチ ハードウェアの場合、ポート チャネル設では、集約ポートのトラフィック抑制は設定値の最大2倍になることがあります。新しいハードウェア ポートは slice-0 と slice-1 の 2 つのグループに内部的にさらに分割されています。スライスマップを確認するには、`vsh_lc` コマンドの `show platform internal hal l2 port gpd` を使用して、`Sl` カラムで `slice 0` または `slice 1` を探します。ポート チャネル メンバーがスライス 0 とスライス 1 の両方に該当する場合、式は各スライスに基づいて計算されるため、許可されるストーム制御トラフィックが設定値の 2 倍になることがあります。

- 使用可能な帯域幅のパーセンテージで設定する場合、値100はトラフィック ストーム制御を行わないことを意味し、値 0.01 はすべてのトラフィックを抑制します。
- ハードウェアの制限およびさまざまなサイズのパケットのカウント方式が原因で、レベルのパーセンテージは概数になります。着信トラフィックを構成するフレームのサイズに応じて、実際に適用されるパーセンテージ レベルと設定したパーセンテージ レベルの間には、数パーセントの誤差がある可能性があります。1 秒あたりのパケット数（PPS）の値は、256 バイトに基づいてパーセンテージに変換されます。
- 最大バーストは、通過するトラフィックがないときに許可されるレートの最大累積です。トラフィックが開始されると、最初の間隔では累積レートまでのすべてのトラフィックが許可されます。後続の間隔では、トラフィックは設定されたレートまでのみ許可されます。サポートされる最大数は65535 KB です。設定されたレートがこの値を超えると、PPS とパーセンテージの両方についてこの値で制限されます。
- 累積可能な最大バーストは 512 MB です。
- 最適化されたマルチキャスト フラッディング（OMF）モードの出力リーフ スイッチでは、トラフィック ストーム制御は適用されません。
- OMF モードではない出力リーフ スイッチでは、トラフィック ストーム制御が適用されます。
- FEX のリーフ スイッチでは、ホスト側インターフェイスにはトラフィック ストーム制御を使用できません。
- Cisco Nexus C93128TX、C9396PX、C9396TX、C93120TX、C9332PQ、C9372PX、C9372TX、C9372PX-E、C9372TX-E の各スイッチでは、トラフィック ストーム制御のユニキャスト/マルチキャストの差別化がサポートされていません。
- Cisco Nexus C93128TX、C9396PX、C9396TX、C93120TX、C9332PQ、C9372PX、C9372TX、C9372PX-E、C9372TX-E の各スイッチでは、トラフィック ストーム制御の SNMP トラップがサポートされていません。

- Cisco Nexus C93128TX、C9396PX、C9396TX、C93120TX、C9332PQ、C9372PX、C9372TX、C9372PX-E、C9372TX-Eの各スイッチでは、トラフィック ストーム制御トラップがサポートされていません。
- ストーム制御アクションは、物理イーサネット インターフェイスおよびポート チャネル インターフェイスでのみサポートされます。

  リリース4.1(1)以降では、ストーム制御**シャットダウン**オプションがサポートされています。デフォルトのSoak Instance Countを持つインターフェイスに対して**シャットダウン**アクションが選択されると、しきい値を超えるパケットは3秒間ドロップされ、ポートは3秒間シャットダウンされます。デフォルトのアクションは、**ドロップ**です。**シャットダウン**アクションを選択すると、ユーザーはソーキング間隔を指定するオプションを使用できます。デフォルトのソーキング間隔は3秒です。設定可能な範囲は3 ～ 10秒です。
- インターフェイスに設定されたデータ プレーン ポリシング（DPP）ポリサーの値がストーム ポリサーの値よりも低い場合、DPPポリサーが優先されます。DPPポリサーとストーム ポリサーの間に設定されている低い方の値が、設定されたインターフェイスで適用されます。
- リリース4.2(6)以降、ストーム ポリサーは、DHCP、ARP、ND、HSRP、PIM、IGMP、およびEIGRPプロトコルに対応する、リーフ スイッチのすべての転送制御トラフィックに強制されます。このことは、ブリッジ ドメインが**BDでのフラッディング**または**カプセル化でのフラッディング**のどちらに設定されているかには関係しません。この動作の変更は、EX以降のリーフスイッチにのみ適用されます。
  - EXスイッチでは、プロトコルの1つに対し、スーパーバイザ ポリサーとストーム ポリサーの両方を設定できます。この場合、サーバーが設定されたスーパーバイザ ポリサー レート（制御プレーン ポリシング、CoPP）よりも高いレートでトラフィックを送信すると、ストーム ポリサーはストーム ポリサー レートとして設定されているよりも多くのトラフィックを許可します。着信トラフィック レートがスーパーバイザ ポリサー レート以下の場合、ストーム ポリサーは設定されたストーム トラフィック レートを正しく許可します。この動作は、設定されたスーパーバイザ ポリサーおよびストーム ポリサーのレートに関係なく適用されます。
  - ストーム ポリサーが、指定されたプロトコルのリーフ スイッチで転送されるすべての制御トラフィックに適用されるようになった結果、リーフ スイッチで転送される制御トラフィックがストーム ポリサー ドロップの対象になります。以前のリリースでは、この動作の変更の影響を受けるプロトコルでは、このようなストーム ポリサーのドロップは発生しません。
- トラフィック ストーム制御は、PIMが有効になっているブリッジ ドメインまたはVRFインスタンスのマルチキャスト トラフィックをポリシングできません。
- ストーム コントロール ポリサーがポートチャネル インターフェイスに適用されている場合、許可されるレートが設定されているレートを超えることがあります。ポートチャネルのメンバーリンクが複数のスライスにまたがる場合、許可されるトラフィック レートは、構成されたレートにメンバーリンクがまたがるスライスの数を掛けたものに等しくなります。

ポートからスライスへのマッピングは、スイッチ モデルによって異なります。

例として、ストーム ポリサー レートが 10Mbps のメンバー リンク port1、port2、および port3 を持つポートチャネルがあるとします。

- port1、port2、port3 が slice1 に属している場合、トラフィックは 10Mbps にポリシングされます。
- port1 と port2 が slice1 に属し、port3 が slice2 に属している場合、トラフィックは 20Mbps にポリシングされます。
- port1 が slice1 に属し、port2が slice2 に属し、port3 が slice3 に属している場合、トラフィックは 30Mbps にポリシングされます。

GUI を使用したトラフィック ストーム制御ポリシーの設定

手順

ステップ 1 メニュー バーで、[Fabric] をクリックします。

ステップ 2 サブメニュー バーで、[Access Policies] をクリックします。

ステップ 3 **Navigation** ウィンドウで **Policies** を展開します。

ステップ 4 **Interface** を展開します。

ステップ 5 [Storm Control] を右クリックし、[Create Storm Control Interface Policy] を選択します。

ステップ 6 [Create Storm Control Interface Policy] ダイアログボックスで、[Name] フィールドにポリシーの名前を入力します。

ステップ 7 **Configure Storm Control** フィールドで、**All Types** または **Unicast, Broadcast, Multicast** のいずれかのオプション ボタンをクリックします。

（注） **Unicast, Broadcast, Multicast** オプション ボタンを選択すると、それぞれのトラフィック タイプで個別にストーム制御を設定することができます。

ステップ 8 [Specify Policy In] フィールドで、[Percentage] または [Packets Per Second] いずれかのオプション ボタンをクリックします。

ステップ 9 [Percentage] を選択した場合は、次の手順を実行します。

a) [Rate] フィールドに、トラフィック レートのパーセンテージを入力します。

ポートで使用可能な合計帯域幅のパーセンテージを指定する 0 ～ 100 の数値を入力します。1 秒の間隔中に入力トラフィックがこのレベルに達するか、それを超えると、トラフィック ストーム制御により、その間隔の残りのトラフィックはドロップされます。値 100 は、トラフィック ストーム制御を行わないことを意味します。値 0 の場合、すべてのトラフィックが抑制されます。

b) [Max Burst Rate] フィールドに、バースト トラフィック レートのパーセンテージを入力します。

ポートで使用可能な合計帯域幅のパーセンテージを指定する 0 ～ 100 の数値を入力します。入力トラフィックがこれ以上になると、トラフィック ストーム制御が開始してトラフィックをドロップされるようになります。

（注） **Max Burst Rate** は、**Rate** の値以上でなければなりません。

ステップ 10 [Packets Per Second] を選択した場合は、次の手順を実行します。

a) [Rate] フィールドに、トラフィック レートを 1 秒あたりのパケット数で入力します。

この間、トラフィック レベル（1 秒あたりにポートを通過するパケット数として表される）が、設定したトラフィック ストーム制御レベルと比較されます。入力トラフィックが、ポートに設定したトラフィック ストーム制御レベルに達するかそれを超えると、トラフィック ストーム制御機能によってそのインターバルが終了するまでトラフィックがドロップされます。

b) [Max Burst Rate] フィールドに、バースト トラフィック レートを 1 秒あたりのパケット数で入力します。

この間、トラフィック レベル（1 秒あたりにポートを通過するパケット数として表される）が、設定したバースト トラフィック ストーム制御レベルと比較されます。入力トラフィックが、ポートに設定したトラフィック ストーム制御レベルに達するかそれを超えると、トラフィック ストーム制御機能によってそのインターバルが終了するまでトラフィックがドロップされます。

ステップ 11 **[Storm Control Action]** で [shutdown]を選択し、**[Storm Control Soak Count]** フィールドでデフォルトを調整することで、ポリシー アクションをデフォルトから変更できます。

（注） デフォルトの Soak Instance Count を持つインターフェイスに対して**シャットダウン** アクションが選択されると、しきい値を超えるパケットは3秒間ドロップされ、ポートは 3 秒間シャットダウンされます。

ステップ 12 [Submit] をクリックします。

ステップ 13 ストーム制御インターフェイス ポリシーをインターフェイス ポートに適用します。

a) メニュー バーで、[Fabric] をクリックします。
b) サブメニュー バーで、[Access Policies] をクリックします。
c) **Navigation** ウィンドウで **Interfaces** を展開します。
d) **Leaf Interfaces** を展開します。
e) **Policy Groups** を展開します。
f) **Leaf Policy Groups** を選択します。

（注） APIC バージョンが 2.x よりも前の場合は、**[Policy Groups]** を選択します。

g) リーフ アクセス ポート ポリシー グループ、PC インターフェイス ポリシー グループ、vPC インターフェイス ポリシー グループ、またはストーム制御ポリシーを適用する PC/vPC オーバーライド ポリシー グループを選択します。

h) [Work] ペインで、[Storm Control Interface Policy] のドロップダウンをクリックし、作成したトラフィック ストーム制御ポリシーを選択します。
i) [送信（Submit）] をクリックします。

NX-OS スタイルの CLI を使用したトラフィック ストーム制御ポリシーの設定

手順

	コマンドまたはアクション	目的

ステップ 1	次のコマンドを入力して PPS ポリシーを作成します。
	例：
	`(config)# template policy-group pg1`
	`(config-pol-grp-if)# storm-control pps 10000 burst-rate 10000`	

ステップ 2	次のコマンドを入力してパーセント ポリシーを作成します。
	例：
	`(config)# template policy-group pg2`
	`(config-pol-grp-if)# storm-control level 50 burst-rate 60`	

ステップ 3	物理ポート、ポート チャネルまたは仮想ポート チャネルでストーム制御を設定します。
	例：
	`[no] storm-control [unicast|multicast|broadcast] level <percentage> [burst-rate <percentage>]`
	`[no] storm-control [unicast|multicast|broadcast] pps <packet-per-second> [burst-rate <packet-per-second>]`
	
	`sd-tb2-ifc1# configure terminal`
	
	`sd-tb2-ifc1(config)# leaf 102`
	
	`sd-tb2-ifc1(config-leaf)# interface ethernet 1/19`
	`sd-tb2-ifc1(config-leaf-if)# storm-control unicast level 35 burst-rate 45`
	`sd-tb2-ifc1(config-leaf-if)#`	

	コマンドまたはアクション	目的
	storm-control broadcast level 36 burst-rate 36 sd-tb2-ifc1(config-leaf-if)# storm-control broadcast level 37 burst-rate 38 sd-tb2-ifc1(config-leaf-if)# sd-tb2-ifc1# configure terminal sd-tb2-ifc1(config)# leaf 102 sd-tb2-ifc1(config-leaf)# interface ethernet 1/19 sd-tb2-ifc1(config-leaf-if)# storm-control broadcast pps 5000 burst-rate 6000 sd-tb2-ifc1(config-leaf-if)# storm-control unicast pps 7000 burst-rate 7000 sd-tb2-ifc1(config-leaf-if)# storm-control unicast pps 8000 burst-rate 10000 sd-tb2-ifc1(config-leaf-if)#	
ステップ 4	ポリシー アクションを変更するには、次の手順を実行します。 **例：** apic1(config-leaf-if)# storm-control action ? drop drop shutdown shutdown	
ステップ 5	ポート **シャットダウン** アクションにのみ適用される soak-instance カウントを設定します。 **例：** apic-ifc1(config-leaf)# int eth 1/27 apic-ifc1(config-leaf-if)# storm-control soak-instance-count ? <3-10> Storm Control SI-Count Instances	

REST API を使用したトラフィック ストーム制御ポリシーの設定

トラフィック ストーム制御ポリシーを設定するには、希望するプロパティを使用して `stormctrl:IfPol` オブジェクトを作成します。

MyStormPolicy というポリシーを作成するには、次の HTTP POST メッセージを送信します。

```
POST https://192.0.20.123/api/mo/uni/infra/stormctrlifp-MyStormPolicy.json
```

使用可能な帯域幅のパーセンテージでポリシーを指定するには、POST メッセージの本文に次の JSON ペイロード構造を含めます。

```
{"stormctrlIfPol":
    {"attributes":
        {"dn":"uni/infra/stormctrlifp-MyStormPolicy",
        "name":"MyStormPolicy",
        "rate":"75",
        "burstRate":"85",
        "rn":"stormctrlifp-MyStormPolicy",
        "status":"created"
        },
        "children":[]
    }
}
```

1 秒あたりのパケット数でポリシーを指定するには、POST メッセージの本文に次の JSON ペイロード構造を含めます。

```
{"stormctrlIfPol":
    {"attributes":
        {"dn":"uni/infra/stormctrlifp-MyStormPolicy",
        "name":"MyStormPolicy",
        "ratePps":"12000",
        "burstPps":"15000",
        "rn":"stormctrlifp-MyStormPolicy",
        "status":"created"
        },
        "children":[]
    }
}
```

トラフィック ストーム制御インターフェイス ポリシーをインターフェイス ポートに適用します。

```
[A] Interface Ethernet 1/1:

Storm-control Action
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraHPathS name='__ui_l101_eth1--1' status='created,modified'>
<infraRsPathToAccBaseGrp tDn='uni/infra/funcprof/accportgrp-__ui_l101_eth1--1'
status='created,modified'>
</infraRsPathToAccBaseGrp>
<infraRsHPathAtt tDn='topology/pod-1/paths-101/pathep-[eth1/1]' status='created,modified'>
</infraRsHPathAtt>
</infraHPathS>
<infraFuncP status='created,modified'>
<infraAccPortGrp name='__ui_l101_eth1--1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_l101_eth1--1'>
</infraRsStormctrlIfPol>
</infraAccPortGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlAction='shutdown'
name='__ui_l101_eth1--1'>
</stormctrlIfPol>
</infraInfra>

Storm-control Soak-instance-count
<?xml version="1.0" encoding="UTF-8"?>
```

```
<infraInfra status='created,modified'>
<infraHPathS name='__ui_l101_eth1--1'>
<infraRsPathToAccBaseGrp tDn='uni/infra/funcprof/accportgrp-__ui_l101_eth1--1'
status='created,modified'>
</infraRsPathToAccBaseGrp>
</infraHPathS>
<infraFuncP status='created,modified'>
<infraAccPortGrp name='__ui_l101_eth1--1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_l101_eth1--1'>
</infraRsStormctrlIfPol>
</infraAccPortGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlSoakInstCount='4'
name='__ui_l101_eth1--1' >
</stormctrlIfPol>
</infraInfra>

CLI:
Last login: 2019-03-12T22:13:52.000+00:00 UTC
apic1# config
apic1(config)# leaf 101
apic1(config-leaf)# in eth 1/1
apic1(config-leaf-if)# storm-control action shutdown
apic1(config-leaf-if)# storm-control soak-instance-count 4



[B] Port Channel pc1:

Storm-control Action
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraHPathS name='__ui_l101_pc1'>
<infraRsPathToAccBaseGrp tDn='uni/infra/funcprof/accbundlepolgrp-__ui_l101_pc1'
status='created,modified'>
</infraRsPathToAccBaseGrp>
</infraHPathS>
<infraFuncP status='created,modified'>
<infraAccBndlPolGrp name='__ui_l101_pc1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_l101_pc1'>
</infraRsStormctrlIfPol>
</infraAccBndlPolGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlAction='shutdown' name='__ui_l101_pc1'>
</stormctrlIfPol>
</infraInfra>

Storm-control soak-instance-count
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraHPathS name='__ui_l101_pc1'>
<infraRsPathToAccBaseGrp tDn='uni/infra/funcprof/accbundlepolgrp-__ui_l101_pc1'
status='created,modified'>
</infraRsPathToAccBaseGrp>
</infraHPathS>
<infraFuncP status='created,modified'>
<infraAccBndlPolGrp name='__ui_l101_pc1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_l101_pc1'>
</infraRsStormctrlIfPol>
</infraAccBndlPolGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlSoakInstCount='9' name='__ui_l101_pc1'>
</stormctrlIfPol>
</infraInfra>
```

```
CLI:
Last login: 2019-03-12T22:21:48.000+00:00 UTC
apic1# config
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel pc1
apic1(config-leaf-if)# storm-control action shutdown
apic1(config-leaf-if)# storm-control soak-instance-count 9
```

```
[C] Template Port Channel

Strom-control Action
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraFuncP status='created,modified'>
<infraAccBndlGrp name='tPC1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_po_tPC1'>
</infraRsStormctrlIfPol>
</infraAccBndlGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlAction='shutdown' name='__ui_po_tPC1'>
</stormctrlIfPol>
</infraInfra>

Storm-control Soak-instance-count
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraFuncP status='created,modified'>
<infraAccBndlGrp name='tPC1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_po_tPC1'>
</infraRsStormctrlIfPol>
</infraAccBndlGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlSoakInstCount='9' name='__ui_po_tPC1'>
</stormctrlIfPol>
</infraInfra>

CLI
Last login: 2019-03-12T22:21:48.000+00:00 UTC
apic1# config
apic1(config)# template port-channel tPC1
apic1(config-po-ch-if)# storm-control action shutdown
apic1(config-po-ch-if)# storm-control soak 9
```

```
[D] Template Policy-group

Storm-control Action
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraFuncP status='created,modified'>
<infraAccPortGrp name='pg1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_pg_pg1'>
</infraRsStormctrlIfPol>
</infraAccPortGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlAction='shutdown' name='__ui_pg_pg1'>
</stormctrlIfPol>
</infraInfra>

Storm-control Soak-instance-count
```

```
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraFuncP status='created,modified'>
<infraAccPortGrp name='pg1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_pg_pg1'>
</infraRsStormctrlIfPol>
</infraAccPortGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlSoakInstCount='8' name='__ui_pg_pg1'>
</stormctrlIfPol>
</infraInfra>

CLI
Last login: 2019-03-12T22:36:36.000+00:00 UTC
apic1# config
apic1(config)# template policy-group pg1
apic1(config-pol-grp-if)# storm-control action shutdown
apic1(config-pol-grp-if)# storm-control soak-instance-count 8



[E] VPC

Storm-control Action
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraHPathS name='__ui_l101_102_vpc1' status='created,modified'>
<infraRsPathToAccBaseGrp tDn='uni/infra/funcprof/accbundlepolgrp-__ui_l101_102_vpc1'
status='created,modified'>
</infraRsPathToAccBaseGrp>
<infraRsHPathAtt tDn='topology/pod-1/protpaths-101-102/pathep-[vpc1]'
status='created,modified'>
</infraRsHPathAtt>
</infraHPathS>
<infraFuncP status='created,modified'>
<infraAccBndlPolGrp name='__ui_l101_102_vpc1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_l101_102_vpc1'>
</infraRsStormctrlIfPol>
</infraAccBndlPolGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlAction='shutdown'
name='__ui_l101_102_vpc1'>
</stormctrlIfPol>
</infraInfra>

Storm-control Soak-instance-count
<?xml version="1.0" encoding="UTF-8"?>
<infraInfra status='created,modified'>
<infraHPathS name='__ui_l101_102_vpc1'>
</infraHPathS>
<infraFuncP status='created,modified'>
<infraAccBndlPolGrp name='__ui_l101_102_vpc1' status='created,modified'>
<infraRsStormctrlIfPol status='created,modified' tnStormctrlIfPolName='__ui_l101_102_vpc1'>
</infraRsStormctrlIfPol>
</infraAccBndlPolGrp>
</infraFuncP>
<stormctrlIfPol status='created,modified' stormCtrlSoakInstCount='8'
name='__ui_l101_102_vpc1'>
</stormctrlIfPol>
</infraInfra>

CLI:
Last login: 2019-03-12T23:54:13.000+00:00 UTC
apic1# configure
```

```
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc vpc1
apic1(config-vpc-if)# storm-control action shutdown
apic1(config-vpc-if)# storm-control soak-instance-count 8
```

ストーム制御 SNMP トラップの設定

ここでは、リーフ スイッチでトラフィック ストーム制御 SNMP トラップを設定する方法について説明します。

MIB 定義のトラップ名を使用して、SNMP トラップのストーム制御を設定することができます。インターフェイスのMIBイベントと、いつストームが検出されてクリアされたかにより、同じリーフのトラップをフィルタリングして、ストームを設定します。ストームは次の2つの方法で設定できます。

- 詳細な設定：ユニキャスト、マルチキャスト、ブロードキャストなどのトラフィックのタイプを設定します。
- 詳細でない設定：すべてのタイプのトラフィックを設定します。

ストーム制御のしきい値に達した場合に Cisco ACI から SNMP トラップをトリガーする際の制限の詳細については、ストーム制御の注意事項と制約事項（306ページ）を参照してください。トラフィック ストーム制御トラップでサポートされていない Cisco Nexus スイッチの詳細については、ストーム制御のガイドラインを参照してください。

ストーム トラップ

ストーム トラップは、イベントが発生し、ストームがアクティブまたはクリアされるたびにトリガーされます。

```
cpscEventRev1 NOTIFICATION-TYPE
        OBJECTS   { cpscStatus }
        STATUS     current
        DESCRIPTION
```

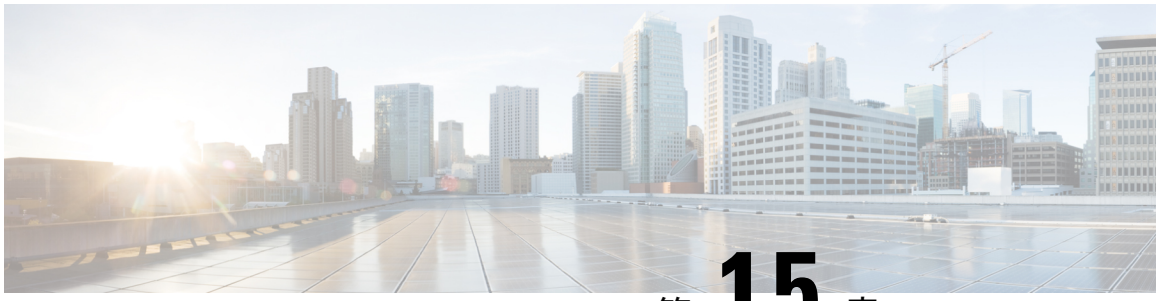
実装では、特定のトラフィック タイプに関してインターフェイスでストーム イベントが発生したときに、この通知を送信することになります。

ストーム ステータスは、それぞれブロードキャスト、ユニキャスト、マルチキャスト、および非詳細ラフィック タイプのフィールドである `[bcDropIncreased]`、`[uucDropIncreased]`、`[mcDropIncreased]`、および `[dropIncreased]` で更新されます。これらは `dbgIfStorm` MO のフィールドです。詳細設定と非詳細設定では、フラグを使用してストームを設定します。ストームがアクティブな場合、フラグは 1 に設定され、ストームがクリアされると、フラグは 2 に設定されます。次のコマンドのフラグにより、SNMPトラップ トリガーに必要なイベントが生成されます。

```
cat / mit / sys / phys-\ [eth--1 \] / dbgIfStorm / summary
```

```
# Interface Storm Drop Couters
bcDropBytes   :0
bcDropIncreased   :2
childAction    :
```

```
dn        :sys/phys-[eth/1]/dbgIfStorm
dropBytes      :0
dropIncreased  :2
mcDropBytes    :0
mcDropIncreased :2
modTs          :never
monPoIDn       :uni/infra/moninfra-default
m              :dbgIfStorm
status         :
uucDropBytes   :0
uucDropIncreased :2
```



第 15 章

MACsec

この章は、次の内容で構成されています。

MACsec について

MACsec は、IEEE 802.1AE 規格ベースのレイヤ 2 ホップバイホップ暗号化であり、これにより、メディア アクセス非依存プロトコルに対してデータの機密性と完全性を確保できます。

MACsec は、暗号化キーにアウトオブバンド方式を使用して、有線ネットワーク上で MAC レイヤの暗号化を提供します。MACsec Key Agreement（MKA）プロトコルでは、必要なセッション キーを提供し、必要な暗号化キーを管理します。

802.1 ae MKA と暗号化はリンク、つまり、リンク (ネットワーク アクセス デバイスと、PC か IP 電話機などのエンドポイント デバイス間のリンク) が直面しているホストのすべてのタイプでサポートされますかにリンクが接続されている他のスイッチまたはルータ。

MACsec は、イーサネット パケットの送信元および宛先 MAC アドレスを除くすべてのデータを暗号化します。ユーザは、送信元と宛先の MAC アドレスの後に最大 50 バイトの暗号化をスキップするオプションもあります。

WAN またはメトロ イーサネット上に MACsec サービスを提供するために、サービス プロバイダーは、Ethernet over Multiprotocol Label Switching（EoMPLS）および L2TPv3 などのさまざまなトランスポート レイヤ プロトコルを使用して、E-Line や E-LAN などのレイヤ 2 透過サービスを提供しています。

EAP-over-LAN（EAPOL）プロトコルデータ ユニット（PDU）のパケット本体は、MACsec Key Agreement PDU（MKPDU）と呼ばれます。3 回のハートビート後（各ハートビートは 2 秒）に

参加者から MKPDU を受信しなかった場合、ピアはライブ ピア リストから削除されます。たとえば、クライアントが接続を解除した場合、スイッチ上の参加者はクライアントから最後の MKPDU を受信した後、3 回のハートビートが経過するまで MKA の動作を継続します。

APIC ファブリック MACsec

APIC はまたは責任を負う MACsec キーチェーン ディストリビューションのポッド内のすべてのノードに特定のポートのノードになります。サポートされているMACsecキーチェーンし、apic 内でサポートされている MACsec ポリシー ディストリビューションのとおりです。

- 単一ユーザ提供キーチェーンと 1 ポッドあたりポリシー
- ユーザが提供されるキーチェーンとファブリック インターフェイスごとのユーザが提供されるポリシー
- 自動生成されたキーチェーンおよび 1 ポッドあたりのユーザが提供されるポリシー

ノードは、複数のポリシーは、複数のファブリック リンクの導入を持つことができます。これが発生すると、ファブリック インターフェイスごとキーチェーンおよびポリシーが優先して指定の影響を受けるインターフェイス。自動生成されたキーチェーンと関連付けられている MACsec ポリシーでは、最も優先度から提供されます。

APIC MACsec では、2 つのセキュリティ モードをサポートしています。MACsec **セキュリティで保護する必要があります** 中に、リンクの暗号化されたトラフィックのみを許可する **セキュリティで保護する必要があります** により、両方のクリアし、リンク上のトラフィックを暗号化します。MACsec を展開する前に **セキュリティで保護する必要があります** モードでのキーチェーンは影響を受けるリンクで展開する必要がありますまたはリンクがダウンします。たとえば、ポートをオンにできますで MACsec **セキュリティで保護する必要があります** モードがピアがしているリンクでのキーチェーンを受信する前にします。MACsec を導入することが推奨されて、この問題に対処する **セキュリティで保護する必要があります** モードとリンクの 1 回すべてにセキュリティ モードを変更 **セキュリティで保護する必要があります** 。

（注） MACsec インターフェイスの設定変更は、パケットのドロップになります。

MACsec ポリシー定義のキーチェーンの定義に固有の設定と機能の機能に関連する設定で構成されています。キーチェーン定義と機能の機能の定義は、別のポリシーに配置されます。MACsec 1 ポッドあたりまたはインターフェイスごとの有効化には、キーチェーン ポリシーおよび MACsec 機能のポリシーを組み合わせることが含まれます。

（注） 内部を使用して生成キーチェーンは、ユーザのキーチェーンを指定する必要はありません。

APIC アクセス MACsec

MACsec はリーフ スイッチ L3out インターフェイスと外部のデバイス間のリンクを保護するために使用します。APIC GUI および CLI のユーザを許可するで、MACsec キーとファブリック L3Out インターフェイスの設定を MacSec をプログラムを提供する物理/pc/vpc インターフェイスごと。ピアの外部デバイスが正しい MacSec 情報を使用してプログラムすることを確認するには、ユーザの責任です。

スイッチ プロファイルの注意事項および制約事項

MACsec は次のスイッチでサポートされます。

- N9K-C93108TC-FX3P
- N9K-C93108TC-FX
- N9K-C93180YC-FX3
- N9K-C93180YC-FX
- N9K-C93216TC-FX2
- N9K-C93240YC-FX2
- N9K-C9332C
- N9K-C93360YC-FX2
- N9K-C9336C-FX2
- N9K-C9348GC-FXP、10G +のみ
- N9K-C9364C
- N9K-C9332D-GX2B（5.2(3) リリース以降）

MACsec は次のライン カードでサポートされます。

- N9K-X9716D-GX（5.2(2) リリース以降）
- N9K-X9736C-FX

次の注意事項および制約事項に従って、スイッチで MACsec を設定します。

- MACsec は10G QSA モジュールではサポートされていません。
- MACsec は Cisco ACI リーフ スイッチの 1G の速度ではサポートされていません。
- MACsec は、L3Out が有効になっているリーフ スイッチ ポートでのみサポートされます。たとえば、Cisco ACI リーフ スイッチとコンピュータ ホスト間の MACsec はサポートされていません。スイッチ間モードのみがサポートされます。
- 銅線ポートを使用する場合、銅線ケーブルは 10G モードでピア デバイス（スタンドアロン N9k）に直接接続する必要があります。

- ピアの 10G 銅線 SFP モジュールはサポートされません。
- Cisco Application Policy Infrastructure Controller（APIC）リリース 4.0 以降では、MACsec はリモート リーフ スイッチでサポートされています。
- FEX ポートは MACsec ではサポートされません。
- **must-secure** モードは、ポッド レベルではサポートされていません。
- 「default」という名前の MACsec ポリシーはサポートされていません。
- 自動キー生成は、ファブリック ポートのポッド レベルでのみサポートされます。
- そのノードのファブリック ポートが **[必須セキュア]** モードの MACsec で実行されている場合、ノードの再起動をクリアしないでください。
- MACsec を実行しているポッドに新しいノードを追加する、またはポッド内のノードのステートレス リブートを行うには、ノードをポッドに参加させるために、**must-secure** モードを **should-secure** に変更する必要があります。
- ファブリック リンクが **should-secure** モードである場合にのみ、アップグレードまたはダウングレードを開始します。アップグレードまたはダウングレードが完了したら、モードを **must-secure** に変更できます。**must-secure** モードでアップグレードまたはダウングレードすると、ノードがファブリックへの接続を失います。失われた接続を回復するには、**should-secure** モードで、Cisco APIC に表示されるノードのファブリック リンクを設定する必要があります。ファブリックが MACsec をサポートしていないバージョンにダウングレードされた場合、ファブリック外のノードがクリーン リブートされる必要があります。
- PC または vPC インターフェイスの場合、MACsec は PC または vPC インターフェイスごとのポリシー グループを使用して展開できます。ポート セレクタは、特定のポートのセットにポリシーを展開するために使用されます。したがって、L3Out インターフェイスに対応する正しいポート セレクタを作成する必要があります。
- 設定をエクスポートする前に、**should-secure** モードで MACsec ポリシーを設定することを推奨します。
- スパイン スイッチ上のすべてのリンクは、ファブリック リンクと見なされます。ただし、スパイン スイッチ リンクを IPN 接続のために使用している場合、そのリンクはアクセス リンクとして扱われます。これらのリンクで MACsec を展開するには、MACsec アクセス ポリシーを使用する必要があります。
- リモート リーフ ファブリック リンクを IPN 接続に使用する場合、そのリンクはアクセス リンクとして扱われます。これらのリンクで MACsec を展開するには、MACsec アクセス ポリシーを使用する必要があります。
- リモート リーフ スイッチのファブリック リンクに **must-secure** モードを不適切に導入すると、ファブリックへの接続が失われる可能性があります。こうした問題を防ぐため、「must-secure モードの展開 （323 ページ）」で説明している手順に従ってください。
- 新しいキーが空のキーチェーンに追加されるか、アクティブなキーがキーチェーンから削除された場合、MACsec セッションの形成または切断に最大で 1 分かかります。

- スパインスイッチのラインカードまたはファブリックモジュールをリロードする前に、すべての **must-secure** リンクを **should-secure** モードに変更する必要があります。リロードが完了し、セッションが **should-secure** モードになったら、モードを **must-secure** に変更します。
- 暗号スイート AES 128 または Extended Packet Numbering（XPN）のない AES 256 を選択する場合は、Security Association Key（SAK）の有効期限を明示的に指定する必要があります。SAK の有効期限値をデフォルト（「無効」）のままにすると、インターフェイスがランダムにアウトオブサービスになる可能性があります。
- フレームの順序が変更されるプロバイダーネットワーク上で MACsec の使用をサポートするには、リプレイウィンドウが必要です。ウィンドウ内のフレームは順不同で受信できますが、リプレイ保護されません。デフォルトのウィンドウ サイズは 64 です。Cisco APIC GUI または CLI を使用する場合、リプレイ ウィンドウのサイズは、0 ー 2^{32}- 1 の範囲で設定できます。XPN 暗号スイートの場合、最大リプレイ ウィンドウ サイズは 2^{30}-1 です。これより大きなウィンドウ サイズを設定しても、ウィンドウ サイズは 2^{30}-1 に制限されます。暗号スイートを非 XPN 暗号スイートに変更した場合、制限はなく、設定されたウィンドウ サイズが使用されます。
- 5.2(2) リリース以降で Cisco N9K-X9716D-GX ライン カード ファブリック ポートで MACsec を使用していて、それを 5.2(2) より前のリリースにダウングレードした場合、そのような以前のリリースではこのラインカードで MACsec はサポートされません。ただし、MACsec がサポートされていないことによる障害は発生しません。このシナリオでは、ピア リーフ スイッチが MACsec をサポートしている場合、セッションはセキュアな状態で起動します。ただし、スパイン側では、セッションが保留中として表示されます。
- リンク レベル フロー制御（LLFC）およびプライオリティ フロー制御（PFC）は、MACsec ではサポートされません。

must-secure モードの展開

must-secure モードに設定されているポリシーを誤って展開すると、接続が失われる可能性があります。そのような問題を避けるため次の手順に従う必要があります。

- MACsec **must-secure** モードを有効にする前に、各リンク ペアにキーチェーンがあることを確認する必要があります。確実を期すため、ポリシーを **should-secure** モードで展開し、MACsec セッションが想定されるリンクでアクティブになったら、モードを **must-secure** に変更することをお勧めします。
- **[必須セキュア]** に設定されている MACsec ポリシーでキーチェーンの交換を試行すると、リンクがダウンする原因となる可能性があります。この場合は、次の手順に従います。
  1. 新しいキーチェーンを使用している MACsec ポリシーを **[should-secure]** モードに変更します。
  2. 影響を受けるインターフェイスが **[should-secure]** モードを使用しているか確認します。
  3. 新しいキーチェーンを使用するように MACsec ポリシーを更新します。

4. アクティブな MACsec セッションと関連するインターフェイスが新しいキーチェーンを使用していることを確認します。
5. MACsec ポリシーを **[必須セキュア]** モードに変更します。

- **must-secure** モードで展開された MACsec ポリシーを無効化/削除するには、次の手順を実行します。
  1. MACsec ポリシーを **[should-secure]** に変更します。
  2. 影響を受けるインターフェイスが **[should-secure]** モードを使用しているか確認します。
  3. MACsec ポリシーを無効/削除します。

キーチェーンの定義

- 開始時刻が **[現在]** のキーチェーンに 1 個のキーが存在します。**must-secure** を、即座にアクティブになるキーを持たないキーチェーンで展開した場合、キーの時刻が来て MACsec セッションが開始されるまで、トラフィックはリンク上でブロックされます。**should-secure** モードが使用されている場合、キーが現在になり、MACsec セッションが開始されるまでトラフィックが暗号化されます。
- 終了時刻が **infinite** のキーチェーンに 1 個のキーが存在する必要があります。キーチェーンの期限が切れると、**must-secure** モードに設定されている影響を受けるインターフェイスでトラフィックがブロックされます。設定されたインターフェイス **はセキュア** モード暗号化されていないトラフィック送信します。
- 終了時刻のオーバーラップし、キーの間に移行すると、MACsecセッションを順番に使用されるキーの開始時刻が残っています。

GUI を使用したファブリック リンクの MACsec の設定

手順

ステップ 1 メニュー バーで、**Fabric** > **Fabric Policies** > **Policies** > **MACsec** > **Interfaces** をクリックします。**Navigation** ウィンドウで、**Interfaces** を右クリックして **Create MACsec Fabric Interface Policy** を開き、次の手順を実行します:

a) **Name** フィールドに、MACsec ファブリック インターフェイス ポリシーの名前を入力します。
b) **MACsec Parameters** フィールドで、以前に設定した MACsec パラメータ ポリシーを選択するか、新しいものを作成します。
c) **MACsec Keychain Policy** フィールドで、以前に設定した MACsec パラメータ ポリシーを選択するか、新しいものを作成して、**Submit** を作成します。

MACsec Keychain Policy を作成するには、GUI を使用した MACsec キーチェーン ポリシーの設定 （326 ページ）を参照してください。

ステップ 2 **MACsec Fabric Interface Policy** をファブリック リーフまたはスパイン ポート ポリシー グループに適用するには、**Interfaces** > ***Leaf/Spine*** **Interfaces** > **Policy Groups** > ***Spine/Leaf*** **Port Policy Group_*name*** をクリックします。**Work** ウィンドウで、今作成した **MACsec Fabric Interface Policy** を選択します。

ステップ 3 **MACsec Fabric Interface Policy** をポッド ポリシー グループに適用するには、ナビゲーション ウィンドウで **Pods** > **Policy Groups** > **Pod Policy Group_*name*** をクリックします。**Work** ウィンドウで、今作成した **MACsec Fabric Interface Policy** を選択します。

GUI を使用したアクセス リンクの MACsec の設定

手順

ステップ 1 メニュー バーで、**[ファブリック]** > **[外部アクセス ポリシー]** をクリックします。**Navigation** ウィンドウで、**Policies** > **Interface** > **MACsec** > **Interfaces** をクリックし、**Interfaces** を右クリックして **Create MACsec Fabric Interface Policy** を開き、次の手順を実行します:

a) **Name** フィールドに、MACsec アクセス インターフェイス ポリシーの名前を入力します。
b) **MACsec Parameters** フィールドで、以前に設定した MACsec パラメータ ポリシーを選択するか、新しいものを作成します。
c) **MACsec Keychain Policy** フィールドで、以前に設定した MACsec パラメータ ポリシーを選択するか、新しいものを作成して、**Submit** を作成します。

MACsec Keychain Policy を作成するには、GUI を使用した MACsec キーチェーン ポリシーの設定 （326 ページ）を参照してください。

ステップ 2 **MACsec Access Interface Policy** をファブリック リーフまたはスパイン ポート ポリシー グループに適用するには、**Interfaces** > ***Leaf/Spine*** **Interfaces** > **Policy Groups** > ***Spine/Leaf*** **Policy Group_*name*** をクリックします。**Work** ウィンドウで、今作成した **MACsec Fabric Interface Policy** を選択します。

APIC GUI を使用した MACsec パラメータの設定

手順

ステップ 1 メニュー バーで、**[Fabric]** > **[Access Policies]** の順にクリックします。**ナビゲーション**] ペインで、[をクリックする **インターフェイス ポリシー** > **ポリシー**] を右クリックし、 **MACsec ポリシー** を開く **MACsec アクセス パラメータ ポリシーの作成** し、次のアクションを実行します。

a) **Name** フィールドに、MACsec アクセス パラメータ ポリシーの名前を入力します。
b) **セキュリティ ポリシー** フィールドで、暗号化されたトラフィックのモードを選択し、をクリックして **Submit** 。

（注） MACsec を展開する前に **セキュア モードをする必要があります** キーチェーンは、影響を受けるインターフェイスに導入する必要があります、またはインターフェイスがダウンします。

ステップ 2 適用する、 **MACsec アクセス パラメータ ポリシー** リーフまたはナビゲーション ペインで、スパイン ポートのポリシー グループをクリックして **インターフェイス ポリシー** > **ポリシー グループ** > **スパイン リーフ/ポリシー Group_ 名** 。**作業**] ペインで、[、 **MACsec アクセス インターフェイス ポリシー** だけを作成します。

GUI を使用した MACsec キーチェーン ポリシーの設定

手順

ステップ 1 メニュー バーで **Fabric** > **Fabric Policies** > **Policies** > **MACsec** > **KeyChains** をクリックします。**Navigation** ウィンドウで、**KeyChains** を右クリックして **Create MACsec Keychain Policy** を開き、次の手順を実行します:

a) **Name** フィールドに、MACsec ファブリック インターフェイス ポリシーの名前を入力します。
b) **MACsec キー ポリシー** テーブルを展開して、キー ポリシーを作成します。

ステップ 2 **MACsec Policy** ダイアログボックスで次の操作を実行します。

a) **Name** フィールドに、MACsec キー ポリシーの名前を入力します。
b) **Key Name** フィールドにキーの名前を入力します (64 文字までの 16 進数)。

（注） キーチェーンあたり最大 64 のキーがサポートされています。

c) **Pre-shared Key** フィールドに、事前共有キーの情報を入力します。

（注） • 128 ビットの暗号スイートでは、32 文字の PSK だけが許可されます。

• 256 ビットの暗号スイートでは、64 文字の PSK だけが許可されます。

d) **Start Time** フィールドで、キーが有効になる日付を選択します。
e) **End Time** フィールドで、キーの有効期限が切れる日付を選択します。**Ok** と **Submit** をクリックします。

（注） キーチェーンで複数のキーを定義する場合には、古いキーから新しいキーへのスムーズな移行を確実にするために、キーの有効期間をオーバーラップさせて定義する必要があります。古いキーの endTime と新しいキーの startTime をオーバーラップさせてください。

アクセス ポリシーでキーチェーン ポリシーを設定するには、メニュー バーで**Fabric** > **External Access Policies** をクリックします。**Navigation** ウィンドウで **Policies** > **Interface** > **MACsec** > **MACsec KeyChain Policies** をクリックし、**Create MACsec Keychain Policy** を右クリックして開き、上記の手順を実行します。

NX-OS スタイルの CLI を使用したMACsecの設定

手順

ステップ 1 アクセス インターフェイスの MACsec セキュリティ ポリシーの設定

例：

```
apic1# configure
apic1(config)#   template macsec access security-policy accmacsecpol1
apic1(config-macsec-param)#     cipher-suite gcm-aes-128
apic1(config-macsec-param)#     conf-offset offset-30
apic1(config-macsec-param)#     description 'description for mac sec parameters'
apic1(config-macsec-param)#     key-server-priority 1
apic1(config-macsec-param)#     sak-expiry-time 110
apic1(config-macsec-param)#     security-mode must-secure
aapic1(config-macsec-param)#     window-size 1
apic1(config-macsec-param)#     exit
apic1(config)#
```

ステップ 2 アクセス インターフェイスの MACsec キー チェーンを設定します。

PSK は、2 通りの方法で設定できます:

（注）
- 下のキー 12ab に示すように、**psk-string** コマンドを使用してインラインで設定します。PSK は、ログに記録され、公開されるため、安全ではありません。
- キー ab12 で示すように、新しいコマンド **Enter PSK string** を **psk-string** コマンドの後で使用し、個別に入力して設定します。ローカルにエコーされるだけで、ログには記録されないため、PSK は安全です。

例：

```
apic1# configure
apic1(config)#   template macsec access keychain acckeychainpol1
apic1(config-macsec-keychain)#     description 'macsec key chain kc1'
apic1(config-macsec-keychain)#     key 12ab
apic1(config-macsec-keychain-key)#       life-time start 2017-09-19T12:03:15 end
2017-12-19T12:03:15
apic1(config-macsec-keychain-key)#       psk-string 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)#       exit
apic1(config-macsec-keychain)#     key ab12
apic1(config-macsec-keychain-key)#       life-time start now end infinite
apic1(config-macsec-keychain-key)#       life-time start now end infinite
apic1(config-macsec-keychain-key)# psk-string
Enter PSK string: 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)# exit
apic1(config-macsec-keychain)# exit
apic1(config)#
```

ステップ 3 アクセス インターフェイスの MACsec インターフェイス ポリシーを設定します:

例：

```
apic1# configure
apic1(config)#   template macsec access interface-policy accmacsecifpol1
apic1(config-macsec-if-policy)#     inherit macsec security-policy accmacsecpol1 keychain
 acckeychainpol1
apic1(config-macsec-if-policy)#     exit
apic1(config)#
```

ステップ 4 MACsec インターフェイス ポリシーをリーフ (またはスパイン) 上のアクセス ンターフェイスに関連付けます:

例：

```
apic1# configure
apic1(config)#   template macsec access interface-policy accmacsecifpol1
apic1(config-macsec-if-policy)#     inherit macsec security-policy accmacsecpol1 keychain
 acckeychainpol1
apic1(config-macsec-if-policy)#     exit
apic1(config)
```

ステップ 5 ファブリック インターフェイス用に MACsec セキュリティ ポリシーを設定します:

例：

```
apic1# configure
apic1(config)#   template macsec fabric security-policy fabmacsecpol1
apic1(config-macsec-param)#     cipher-suite gcm-aes-xpn-128
apic1(config-macsec-param)#     description 'description for mac sec parameters'
apic1(config-macsec-param)#     window-size 1
apic1(config-macsec-param)#     sak-expiry-time 100
apic1(config-macsec-param)#     security-mode must-secure
apic1(config-macsec-param)#     exit
apic1(config)#
```

ステップ 6 ファブリック インターフェイス用に MACsec キー チェーンを設定します:

PSK は、2 通りの方法で設定できます:

（注）
- 下のキー 12ab に示すように、**psk-string** コマンドを使用してインラインで設定します。PSK は、ログに記録され、公開されるため、安全ではありません。
- キー ab12 で示すように、新しいコマンド **Enter PSK string** を **psk-string** コマンドの後で使用し、個別に入力して設定します。ローカルにエコーされるだけで、ログには記録されないため、PSK は安全です。

例：

```
apic1# configure
apic1(config)#   template macsec fabric security-policy fabmacsecpol1
apic1(config-macsec-param)#     cipher-suite gcm-aes-xpn-128
apic1(config-macsec-param)#     description 'description for mac sec parameters'
apic1(config-macsec-param)#     window-size 1
apic1(config-macsec-param)#     sak-expiry-time 100
apic1(config-macsec-param)#     security-mode must-secure
apic1(config-macsec-param)#     exit
apic1(config)#   template macsec fabric keychain fabkeychainpol1
apic1(config-macsec-keychain)#     description 'macsec key chain kc1'
apic1(config-macsec-keychain)#     key 12ab
apic1(config-macsec-keychain-key)#       psk-string 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)#       life-time start 2016-09-19T12:03:15 end
2017-09-19T12:03:15
apic1(config-macsec-keychain-key)#       exit
apic1(config-macsec-keychain)#     key cd78
apic1(config-macsec-keychain-key)# psk-string
Enter PSK string: 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)# life-time start now end infinite
apic1(config-macsec-keychain-key)# exit
apic1(config-macsec-keychain)# exit
apic1(config)#
```

ステップ 7 MACsec インターフェイス ポリシーをリーフ (またはスパイン) 上のファブリック ンターフェイスに関連付けます:

例：

```
apic1# configure
apic1(config)#   leaf 101
apic1(config-leaf)#     fabric-interface ethernet 1/52-53
apic1(config-leaf-if)#       inherit macsec interface-policy fabmacsecifpol2
apic1(config-leaf-if)#       exit
apic1(config-leaf)#
```

REST API を使用した MACsec の設定

MACsec ファブリック ポリシーをファブリックのすべてのポッドに適用します。

例：

```
<fabricInst>
 <macsecFabPolCont>
```

```
        <macsecFabParamPol name="fabricParam1"  secPolicy="should-secure"
replayWindow="120" >
        </macsecFabParamPol>
        <macsecKeyChainPol name="fabricKC1">
            <macsecKeyPol name="Key1"
preSharedKey="0102030405060708090A0B0C0D0E0F10010203040506070809 0A0B0C0D0E0F10"
keyName="A1A2A3A0" startTime="now" endTime="infinite"/>
        </macsecKeyChainPol>
    </macsecFabPolCont>

    <macsecFabIfPol name="fabricPodPol1" useAutoKeys="0">
        <macsecRsToParamPol tDn="uni/fabric/macsecpcontfab/fabparamp-fabricParam1"/>
        <macsecRsToKeyChainPol tDn="uni/fabric/macsecpcontfab/keychainp-fabricKC1"/>
    </macsecFabIfPol>

    <fabricFuncP>
  <fabricPodPGrp name = "PodPG1">
   <fabricRsMacsecPol tnMacsecFabIfPolName="fabricPodPol1"/>
     </fabricPodPGrp>
 </fabricFuncP>

    <fabricPodP name="PodP1">
  <fabricPodS name="pod1" type="ALL">
   <fabricRsPodPGrp tDn="uni/fabric/funcprof/podpgrp-PodPG1"/>
     </fabricPodS>
 </fabricPodP>

</fabricInst>
```

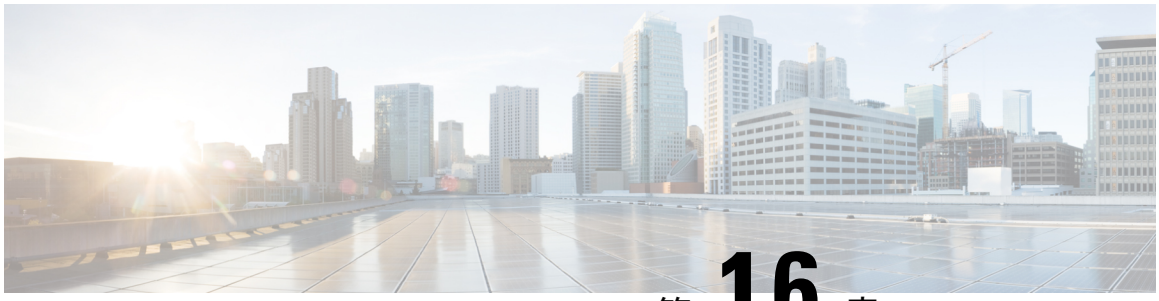
リーフ 101 の eth1/4 上で MACsec アクセス ポリシーを適用します。

例：

```
<infraInfra>
 <macsecPolCont>
       <macsecParamPol name="accessParam1"  secPolicy="should-secure" replayWindow="120"
 >
        </macsecParamPol>
        <macsecKeyChainPol name="accessKC1">
            <macsecKeyPol name="Key1"
preSharedKey="0102030405060708090A0B0C0D0E0F10010203040506070809 0A0B0C0D0E0F10"
keyName="A1A2A3A0" startTime="now" endTime="infinite"/>
        </macsecKeyChainPol>
    </macsecPolCont>

    <macsecIfPol name="accessPol1">
        <macsecRsToParamPol tDn="uni/infra/macsecpcont/paramp-accessParam1"/>
        <macsecRsToKeyChainPol tDn="uni/infra/macsecpcont/keychainp-accessKC1"/>
    </macsecIfPol>

    <infraFuncP>
  <infraAccPortGrp name = "LeTestPGrp">
   <infraRsMacsecIfPol tnMacsecIfPolName="accessPol1"/>
     </infraAccPortGrp>
 </infraFuncP>

 <infraHPathS name="leaf">
  <infraRsHPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/4]" />
  <infraRsPathToAccBaseGrp tDn="uni/infra/funcprof/accportgrp-LeTestPGrp" />
 </infraHPathS>

</infraInfra>
```

リーフ 101 の Eth1/49 およびスパイン 102 の eth 5/1 上で MACsec ファブリック ポリシーを適用します。

```
<fabricInst>
 <macsecFabPolCont>
        <macsecFabParamPol name="fabricParam1"  secPolicy="should-secure"
replayWindow="120" >
        </macsecFabParamPol>
        <macsecKeyChainPol name="fabricKC1">
            <macsecKeyPol name="Key1"
preSharedKey="0102030405060708090A0B0C0D0E0F10010203040506070809 0A0B0C0D0E0F10"
keyName="A1A2A3A0" startTime="now" endTime="infinite"/>
        </macsecKeyChainPol>
    </macsecFabPolCont>

    <macsecFabIfPol name="fabricPol1" useAutoKeys="0">
        <macsecRsToParamPol tDn="uni/fabric/macsecpcontfab/fabparamp-fabricParam1"/>
        <macsecRsToKeyChainPol tDn="uni/fabric/macsecpcontfab/keychainp-fabricKC1"/>
    </macsecFabIfPol>

    <fabricFuncP>
  <fabricLePortPGrp name = "LeTestPGrp">
   <fabricRsMacsecFabIfPol tnMacsecFabIfPolName="fabricPol1"/>
     </fabricLePortPGrp>

  <fabricSpPortPGrp name = "SpTestPGrp">
   <fabricRsMacsecFabIfPol tnMacsecFabIfPolName="fabricPol1"/>
     </fabricSpPortPGrp>
 </fabricFuncP>

 <fabricLFPathS name="leaf">
  <fabricRsLFPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/49]" />
  <fabricRsPathToLePortPGrp tDn="uni/fabric/funcprof/leportgrp-LeTestPGrp" />
 </fabricLFPathS>

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

 <fabricSpineP name="SpNode" >
  <fabricRsSpPortP tDn="uni/fabric/spportp-spine_profile" />
  <fabricSpineS name="spsw" type="range">
   <fabricNodeBlk name="node102" to_="102" from_="102" />
  </fabricSpineS>
 </fabricSpineP>
</fabricInst>
```



第 16 章

ファブリック ポート トラッキング

ファブリック ポート トラッキングについて

ポート トラッキング機能は、ファブリック ポートのステータスに基づいて、各リーフ ノードのダウンリンクポートのステータスを管理します。ファブリック ポートはリーフとスパイン ノード間のリンクです。多層トポロジ内の階層 1 と階層 2 のリーフ ノード間のリンク、およびリモート リーフ ノード間のリンク（バックツーバック リンク）も、ファブリック リンクと見なされます。

この機能が有効にされていて、特定のリーフ ノードで動作しているファブリック ポートの数が設定されたしきい値以下になると、外部ノードが他の正常なリーフ ノードにスイッチ オーバーできるように、リーフ ノードのダウンリンク ポートはダウンにされます。動作中のファブリック ポートの数が設定されたしきい値を超えて回復すると、ダウンリンク ポートは回復します。この時点で、ダウンリンク ポートの起動を遅延させるための待機時間が設定されています。リーフ ノードが vPC ピアの一部であり、インフラ ISIS の隣接関係がない場合（ノードが他の vPC ピア リーフ ノードと通信できない場合）、すべてのファブリック ポートがダウンした場合など、ポート トラッキングがトリガーされた場合、ステータスの復元後に vPC ダウンリンク ポートが起動するまでの時間は、vPC 遅延タイマーまたはポート トラッキングで設定された遅延のいずれか長い方になります。非 vPC ダウンリンクポートは、常にポート トラッキングで設定された遅延タイマーに従います。

Cisco Application Centric Infrastructure（ACI）スイッチ リリース 14.2(1) 以降、ファブリック インフラ ISIS 隣接のステータス（`aggFabAdjOperSt` で表されます。これは管理対象オブジェクト クラス `isisDom` の属性 です）も、ダウンリンク ポートのシャットダウンをトリガーするための代替条件としてチェックされます。このチェックは、特定のリーフ スイッチのファブリック ポートがアップしているもののが、別の理由でリーフ ノードが他の Cisco ACI ノードへの到達可能性を失った場合を考慮に入れて、行われます。この条件は、動作可能なファブリック ポートの最小数などの他のパラメータに関係なく、機能が有効になっている場合は常にチェックされます。ただし、これはリモート リーフ ノードには適用できません。そのようなノードはファブリック インフラの到達可能性について ISIS に依存していないためです。

Cisco ACI スイッチ リリース 15.0(1) 以降、[APIC ポートを含める（Include APIC ports）] オプションがサポートされています。このオプションは、デフォルトで無効です。このオプションが無効になっている場合、ポート トラッキングは、ユーザー トラフィック用に設定されたダウンリンク ポート （つまり、EPG または L3Out によって使用されているポート）のみをダウン状態にしますが、Cisco Application Policy Infrastructure Controller（APIC）に接続されているダウンリンク ポートまたは未使用のポートはダウン状態にしません。このオプションを有効にすると、ポート トラッキングによってリーフ ノードのすべてのダウンリンク ポートがダウン状態にされます。リリース 15.0(1) より前のリリースでは、ユーザー トラフィック用に構成されたダウンリンク ポートは、Cisco APIC に接続されたポートがダウン状態になっていなかったときにダウン状態にされます。ポート トラッキングを構成して、Cisco APIC に接続されたポートをダウン状態にすることはできませんでした。

（注） ポート トラッキングは、各リーフノードでポートを停止または起動する条件をチェックします。

FEX ファブリック ポート（FEX と FEX の親リーフノードを接続するネットワーク インターフェイス、NIF）は、ポート トラッキングの影響を受けません。

GUI を使用したファブリック ポート トラッキングの設定

この手順では、Cisco Application Policy Infrastructure Controller（APIC）GUIを使用してポートトラック機能を設定します。

手順

ステップ 1 メニュー バーで、**[システム（System）] > [システム設定（System Settings）]** の順に選択します。

ステップ 2 [ナビゲーション（Navigation）] ペインで、**[ポート トラッキング（Port Tracking）]** を選択します。

ステップ 3 **[ポート トラッキングの状態（Port tracking state）]** パラメータで **[on]** を選択して、ファブリック ポート トラッキングを有効にします。

ステップ 4 **[遅延復元タイマー（Delay restore timer）]** パラメータには、時間を秒単位で指定します。

このパラメータは、ファブリック ポートの状態とインフラ ISIS 隣接関係が復元された後、リーフノードがダウンリンク ポートを起動するまでの時間を決定します。

ステップ 5 **[ポートトラッキングをトリガーするアクティブなファブリック ポートの数（Number of active fabric ports that triggers port tracking）]** パラメータを設定します。

リーフ ノード上の動作可能なファブリック ポートの数が設定された数以下になると、リーフノードはダウンリンク ポートをダウンさせます。

ステップ 6 （任意） [Include APIC ports when port tracking is trigger]チェックボックスをオンにします。

このパラメータを有効にすると、ポート トラッキングがトリガーされたときに Cisco APIC に接続されているダウンリンク ポートとユーザー トラフィックのダウンリンク ポートがダウンします。このオプションは、Cisco APIC が高可用性のために 2 つの異なるリーフ ノードに接続されている場合を除き、オンにしないでください。

REST API を使用したファブリック ポート トラッキングの設定

この手順では、REST API を使用してポート トラック機能を設定します。

次の例のように REST API POST を送信します。

POST: *apic_ip_address*/api/mo/uni.xml

```
<infraPortTrackPol
    dn="uni/infra/trackEqptFabP-default"    # Fixed DN. Do not change.
    adminSt="on"                            # 'on' to enable, 'off' to disable
    delay="120"                             # The delay timer (sec) to bring up the
                                            #   downlink ports
    minlinks="0"                            # The minimum required number of operational
                                            #   fabric ports
/>
```

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。

リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。

あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

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