Cisco Nexus 9000 シリーズ NX-OS VXLAN 構成ガイド、リリース 10.4(x)

最終更新：2024 年 10 月 24 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/

【注意】シスコ製品をご使用になる前に、安全上の注意（ **www.cisco.com/jp/go/safety_warning/** ）をご確認ください。本書は、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。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

目 次

はじめに

この前書きは、次の項で構成されています。

対象読者

このマニュアルは、Cisco Nexus スイッチの設置、設定、および維持に携わるネットワーク管理者を対象としています。

表記法

コマンドの説明には、次のような表記法が使用されます。

表記法	説明
bold	太字の文字は、表示どおりにユーザが入力するコマンドおよびキーワードです。
italic	イタリック体の文字は、ユーザが値を指定する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角かっこで囲んで示しています。
[x | y]	いずれか1つを選択できる省略可能なキーワードや引数は、角かっこで囲み、縦棒で区切って示しています。
{x | y}	必ずいずれか1つを選択しなければならない必須キーワードや引数は、波かっこで囲み、縦棒で区切って示しています。

表記法	説明
[x {y | z}]	角かっこまたは波かっこが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。角かっこ内の波かっこと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
`variable`	ユーザが値を入力する変数であることを表します。イタリック体が使用できない場合に使用されます。
string	引用符を付けない一組の文字。string の前後には引用符を使用しないでください。引用符を使用すると、その引用符も含めて string と見なされます。

例では、次の表記法を使用しています。

表記法	説明
`screen フォント`	スイッチが表示する端末セッションおよび情報は、スクリーンフォントで示しています。
`太字の screen フォント`	ユーザが入力しなければならない情報は、太字の screen フォントで示しています。
イタリック体の *screen* フォント	ユーザが値を指定する引数は、イタリック体の screen フォントで示しています。
< >	パスワードのように出力されない文字は、山カッコ（< >）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

Cisco Nexus 9000 シリーズ スイッチの関連資料

Cisco Nexus 9000 シリーズ スイッチ全体のマニュアル セットは、次の URL にあります。

http://www.cisco.com/en/US/products/ps13386/tsd_products_support_series_home.html

マニュアルに関するフィードバック

このマニュアルに関する技術的なフィードバック、または誤りや記載もれなどお気づきの点がございましたら、HTML ドキュメント内のフィードバック フォームよりご連絡ください。ご協力をよろしくお願いいたします。

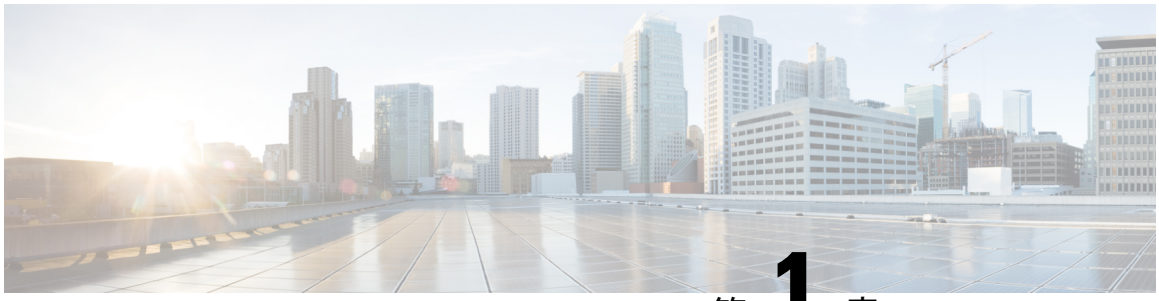
通信、サービス、およびその他の情報

- シスコからタイムリーな関連情報を受け取るには、Cisco Profile Manager でサインアップしてください。
- 重要な技術によりビジネスに必要な影響を与えるには、Cisco Services [英語] にアクセスしてください。
- サービスリクエストを送信するには、Cisco Support [英語] にアクセスしてください。
- 安全で検証済みのエンタープライズクラスのアプリケーション、製品、ソリューション、およびサービスを探して参照するには、Cisco Marketplace にアクセスしてください。
- 一般的なネットワーク、トレーニング、認定関連の出版物を入手するには、Cisco Press [英語] にアクセスしてください。
- 特定の製品または製品ファミリの保証情報を探すには、Cisco Warranty Finder にアクセスしてください。

Cisco バグ検索ツール

Cisco バグ検索ツール（BST）は、シスコ製品とソフトウェアの障害と脆弱性の包括的なリストを管理する Cisco バグ追跡システムへのゲートウェイとして機能する、Web ベースのツールです。BST は、製品とソフトウェアに関する詳細な障害情報を提供します。

通信、サービス、およびその他の情報



第 1 章

新機能および変更された機能に関する情報

新機能および変更された機能に関する情報

表 *1*: 新機能および変更された機能

機能	説明	変更が行われたリリース	参照先
VXLAN EVPN トラフィック エンジニアリング	VXLAN EVPN Tトラフィック エンジニアリング - マルチサイト出力ロードバランシング機能のサポートを追加して、トラフィック ステアリングを改善し、複数のサイト間の DC 間リンクをより有効に活用できるようになりました。	10.4(3)F	VXLAN EVPN トラフィック エンジニアリング：マルチサイト出力ロードバランシングの構成 （363 ページ）
GPO	Cisco Nexus N9K スイッチでのグループ ポリシー オプションのサポートが追加されました。	10.4(3)F	グループポリシーオプションを使用した VXLAN ファブリックのマイクロセグメンテーション（GPO） （527 ページ）

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
VXLAN QoS	X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチで BGW スパインを使用する際に、VXLAN QoS ポリシーのサポートが追加されました。	10.4(3)F	VXLAN QoS の注意事項および制約事項 （460 ページ）
MPLS SR QoS	X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチで、MPLS SR QoS が追加されました。	10.4(3)F	に関する注意事項と制限事項 EVPN と L3VPN (MPLS SR) のシームレスな統合の設定 （286 ページ）

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
VXLAN	Cisco Nexus 9364C-H1 スイッチに VXLAN サポートが追加されました。	10.4(3)F	

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
			VXLAN の注意事項と制約事項 （55 ページ） VXLAN 展開の考慮事項（63 ページ） VXLAN 静的トンネルの注意事項と制約事項 （93 ページ） アンダーレイ（VXLANv6）での IPv6 を使用した VXLAN の注意事項と制限事項 （102 ページ） VXLAN BGP EVPN の注意事項と制約事項 （125 ページ） ダウンストリーム VNI を使用する VXLAN EVPN の注意事項と制約事項 （133 ページ） vPC ファブリック ピアリングの注意事項と制約事項（214 ページ） ESI を使用した EVPN マルチホーミングとの相互運用性 （227 ページ） ESI を使用した EVPN マルチホーミングとの相互運用性に関する注意事項と制限事項 （228 ページ）に関する注意事項と制限事項 EVPN と L3VPN (MPLS SR) のシームレスな統合の設定（286 ページ） EVPN から L3VPN SRv6 へのハンドオフの注意事項と制限事項 （308 ページ） マルチサイトでの VXLAN

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
			EVPN の注意事項と制限事項 （331 ページ） マルチサイトでの TRM のガイドラインと制限事項 （356 ページ） テナント ルーテッド マルチキャストに関する注意事項と制限事項 （377 ページ） レイヤ 3 テナント ルーテッド マルチキャストの注意事項と制約事項 （379 ページ） レイヤ 2/レイヤ 3 テナント ルーテッド マルチキャスト（混合モード）の注意事項と制約事項 （381 ページ） TRM データ MDT の注意事項と制約事項 （423 ページ） VXLAN NGOAM の注意事項と制約事項 （435 ページ） VXLAN EVPN ループの検出と緩和のガイドラインと制限事項 （436 ページ） VXLAN QoS の注意事項および制約事項 （460 ページ） ポート VLAN マッピングに関する注意事項と制限事項： （510 ページ） トランク ポートでの内部 VLAN および外部 VLAN マッピングの設定 （516 ページ）

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
			ポート マルチ VLAN マッピングに関する注意事項と制限事項：（519 ページ） ポリシーベースのリダイレクトの注意事項と制約事項（578 ページ） VNF の比例マルチパスのガイドラインと制限事項（590 ページ） VXLAN クロス コネクトの注意事項と制限事項 （648 ページ） Q-in-VNI の注意事項と制約事項 （656 ページ） L2PT を搭載した Q-in-VNI の注意事項と制約事項（664 ページ） QinQ-QinVNI の注意事項と制約事項 （674 ページ） CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの注意事項と制約事項 （689 ページ） VXLAN にわたるプライベート VLAN に関する注意事項および制約事項 （732 ページ） VXLAN BGP EVPN 中の DHCP スヌーピングの概要（735 ページ） VXLAN 上の DHCP スヌーピングの注意事項および制約事項 （737 ページ）

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
VXLAN 送信元ポートの機能拡張	VXLAN UDP 送信元ポートは、VXLAN カプセル化パケットのポート番号範囲を設定する新しい構成オプションで拡張された。	10.4(3)F	VXLAN UDP 送信元 ポートの設定 （143 ページ）
レイヤ 3 インターフェイスの NGOAM-SLD	レイヤ 3 イーサネットおよびレイヤ 3 ポートチャネルインターフェイスでの NGOAM サウスバウンドループ検出（SLD）のサポートが追加されました。	10.4(3)F	VXLAN NGOAM の注意事項と制約事項 （435 ページ） レイヤ 3 インターフェイスの NGOAM ループ検出の設定 （443 ページ）
マルチサイト エニーキャスト ボーダー ゲートウェイのサポート	Cisco Nexus X9836DM-A および X98900CD-A ラインカードを搭載した Cisco Nexus 9808/9804 スイッチでの VXLAN マルチサイト エニーキャスト BGW のサポートを追加。	10.4(3)F	マルチサイトでの VXLAN EVPN の注意事項と制限事項 （331 ページ）
マルチサイト エニーキャスト BGW の TRM サポート	Cisco Nexus X9836DM-A および X98900CD-A ラインカードを搭載した Cisco Nexus 9808/9804 スイッチで、マルチサイト エニーキャスト BGW を使用した TRM のサポートを追加。	10.4(3)F	レイヤ 3 テナント ルーテッド マルチキャストの注意事項と制約事項 （379 ページ） レイヤ 3 テナント ルーテッド マルチキャストの設定 （400 ページ）
Cisco Nexus 9800 スイッチの NGOAM	Cisco Nexus 9800 スイッチで NGOAM ping、traceroute、および pathtrace のサポートが追加されましたが、Xconnect およびサウスバウンド ループ検出（SLD）はサポートされません。	10.4(3)F	VXLAN NGOAM の注意事項と制約事項 （435 ページ）

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
Cisco Nexus 9800 スイッチのボーダー スパイン サポート	Cisco Nexus 9800 スイッチのボーダースパインとしての VXLAN 機能のサポートが追加されました。	10.4(3)F	VXLAN の注意事項と制約事項 （55 ページ）

機能	説明	変更が行われたリリース	参照先
ファブリックおよび IPv6 DCI IR での IPv6 マルチキャストアンダーレイによるマルチサイトのサポート	ファブリック側のプロトコル独立マルチキャスト（PIMv6）エニーソースマルチキャスト（ASM）と DCI 側の入力複製（IPv6）を使用した VXLAN EVPN および TRM マルチサイトのサポートが追加されました。	10.4(3)F	#unique_48 マルチキャストアンダーレイで IPv6 を使用した VXLAN EVPN および TRM の注意事項および制限事項 IPv6 アンダーレイを使用した VXLAN EVPN マルチサイトについて IPv6 アンダーレイを使用した VXLAN EVPN マルチサイトでの注意事項と制限事項 IPv6 マルチキャスト アンダーレイを使用した VXLAN EVPN マルチサイトの有効化 ファブリック/DCI リンクトラッキングの設定 （341 ページ） ファブリック外部ネイバーの設定 （342 ページ） vPC サポートを使用したマルチサイトの注意事項と制限事項 （345 ページ） IPv6 アンダーレイを使用した TRM マルチサイトの設定についての情報 IPv6 アンダーレイを使用した TRM マルチサイトでの注意事項と制限事項 IPv6 アンダーレイを使用した TRM マルチサイトの構成 マルチサイト設定による TRM の確認 （361 ページ）

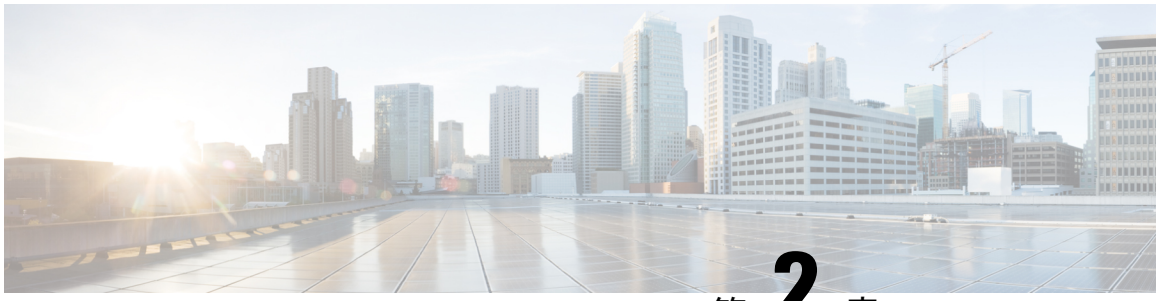
新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
VXLAN PIM BiDir アンダーレイのサポート	Cisco Nexus 9300-FX3/GX/GX2/H2R/H1 スイッチ、9700-GX ラインカードを搭載した 9500 スイッチでの PIM BiDir のサポートが追加されました。	10.4(3)F	アンダーレイの考慮事項（19 ページ） VXLAN アンダーレイでのマルチキャストルーティング （39 ページ）
VXLAN	Cisco Nexus 93400LD-H1 スイッチに VXLAN サポートを追加	10.4(2)F	VXLAN の注意事項と制約事項 （55 ページ）
ファブリック内の TRMv4 マルチサイトエニーキャスト アンダーレイと DCI を介した IR	IPv6 マルチキャストアンダーレイを使用した VXLAN EVPN および TRM のサポートが追加されました。	10.4(2)F	#unique_48 マルチキャストアンダーレイで IPv6 を使用した VXLAN EVPN および TRM の注意事項および制限事項 IPv6 マルチキャスト アンダーレイを使用した VXLAN EVPN および TRM の構成 IPv6 マルチキャスト アンダーレイを使用した VXLAN EVPN および TRM の設定例 IPv6 マルチキャスト アンダーレイを使用した VXLAN EVPN および TRM の確認
プライベート VLAN	Cisco Nexus C9348GCFX3 および Cisco C9348GC-FX3PH のプライベート VLAN のサポートを追加。	10.4(1)F	VXLAN にわたるプライベート VLAN に関する注意事項および制約事項 （732 ページ）

機能	説明	変更が行われたリリース	参照先
VXLAN EVPN ファースト ホップ セキュリティ	IPv4 ファースト ホップ セキュリティのサポートが EVPN VXLAN 環境で提供され、ある VTEP で認証されたホストを別の VTEP に移動できるようになりました。	10.4(1)F	初期ホップセキュリティの構成 （735 ページ）
VTEP とシングルアクティブ ESI の共存	Rx シングルアクティブモードの ESI マルチホーミング サポートを追加。	10.4(1)F	ESI を使用した EVPN マルチホーミングとの相互運用性 （227 ページ） ESI を使用した EVPN マルチホーミングの例 （229 ページ）
入力 VTEP での VXLAN カプセル化パケットの外部 DSCP の設定	入力 VTEP の外部 DSCP フィールドを設定するための、 **tunnel** キーワードを追加。	10.4(1)F	外部 DSCP を使用した IP から VXLAN （457 ページ） VXLAN QoS の注意事項および制約事項 （460 ページ） 入力 VTEP での外部 DSCP の構成 （467 ページ）
出力 VTEP で外部 DSCP に基づいてパケットを分類し、書き換える	入力サービス ポリシーを使用して出力 VTEP の外部 DSCP 値を照合するための、 **tunnel** キーワードを追加。	10.4(1)F	VXLAN から IP （458 ページ） VXLAN QoS の注意事項および制約事項 （460 ページ） 出力 VTEP でのタイプ QoS の設定 （465 ページ）

新機能および変更された機能に関する情報

機能	説明	変更が行われたリリース	参照先
レイヤ 2 の VXLAN QoS 外部ヘッダー ポリシー	VXLAN パケットの外部 DSCP で照合を行い、出力 VTEP のカプセル化解除されたイーサネットパケットで CoS を書き換えるための、新しい **default-vxlan-in-tnl-dscp-policy** QoS ポリシーマップテンプレートを追加。	10.4(1)F	CoS の保持 （459 ページ） VXLAN QoS の注意事項および制約事項 （460 ページ） CoS 保存の設定 （470 ページ）
VXLAN	Cisco Nexus 9332D-H2R プラットフォーム スイッチに VXLAN サポートを追加。	10.4(1)F	VXLAN の注意事項と制約事項 （55 ページ）
VXLAN 送信元ポートの機能拡張	VXLAN UDP 送信元ポートは、VXLAN カプセル化パケットのポート番号範囲を設定する新しい構成オプションで拡張された。	10.4(1)F	VXLAN UDP 送信元 ポートの設定 （143 ページ）
VXLAN マルチサイト BGW 展開のスプリット ループバック	NVE インターフェイス ループバックの構成に関する詳細を追加	10.4(1)F	NVE インターフェイス ループバックの作成および構成 （79 ページ）単一の NVE 送信元ループバック インターフェイスから別の送信元ループバックへの移行 （82 ページ）



第 2 章

概要

この章は、次の内容で構成されています。

ライセンス要件

Cisco NX-OS ライセンス方式の推奨の詳細と、ライセンスの取得および適用の方法については、『*Cisco NX-OS* ライセンス ガイド』および『*Cisco NX-OS* ライセンス オプション ガイド』を参照してください。

サポートされるプラットフォーム

Cisco NX-OS リリース 7.0(3)I7(1) 以降、「Nexus スイッチ プラットフォーム サポート マトリクス」を使用して、選択した機能をサポートするさまざまな Cisco Nexus 9000 および 3000 スイッチのリリース元である Cisco NX-OS を知ることができます。

VXLAN の概要

Virtual Extensible LAN（VXLAN）MAC-in-UDP のカプセル化とトンネリングを使用して、レイヤ 3 インフラストラクチャを越えてレイヤ 2 ネットワークを拡張する方法を提供します。この機能により、共有される共通の物理インフラストラクチャにおいて、仮想化され、マルチテナントのデータセンター デザインを可能にすることができます。

VXLAN には、次の利点があります。

- データセンター ファブリック全体でのワークロードの柔軟な配置。

  これは、テナントのワークロードが単一のデータセンター内の物理ポッド全域に配置されるように、基盤となる共有ネットワーク インフラストラクチャでレイヤ2セグメントを拡張する方法を提供します。または、地理的に多様な複数のデータセンターにまたがる場合もあります。

- より多くのレイヤ 2 セグメントに対応するための高度なスケーラビリティ。

  VXLAN は 24 ビットのセグメント ID、つまり VXLAN ネットワーク ID（VNID）を使用します。これにより、最大 1600 万個の VXLAN セグメントを同じ管理ドメイン内で共存させることができます比較すると、従来の VLAN は最大 4096個の VLAN をサポートできる 12 ビットのセグメント ID を使用します。

- 基盤となるインフラストラクチャにおける、有効なネットワーク パスの使用率。

  VXLAN パケットは、レイヤ 3 ヘッダーに基づいて、基盤となるネットワークを介して転送されます。これは、等コストマルチパス（ECMP）ルーティングおよびをリンク集約プロトコルを使用して、有効なすべてのパスを使用します。対照的に、レイヤ2ネットワークは、ループを回避するために有効な転送パスをブロックすることがあります。

ハードウェア ベースの VXLAN ゲートウェイとしての Cisco Nexus 9000

Cisco Nexus 9000 シリーズ スイッチは、ハードウェア ベースの VXLAN のゲートウェイとして機能することが可能です。これは、レイヤ 3 の境界を越えた 1 つの転送ドメインとして転送のパフォーマンスを低下させずに、VXLAN セグメントと VLAN セグメントをシームレスに接続します。Cisco Nexus 9000 Series ハードウェア ベース VXLAN のカプセル化およびカプセル化解除により、すべてのフレーム サイズに対してライン レート パフォーマンスを提供します。

VXLAN のカプセル化およびパケット形式

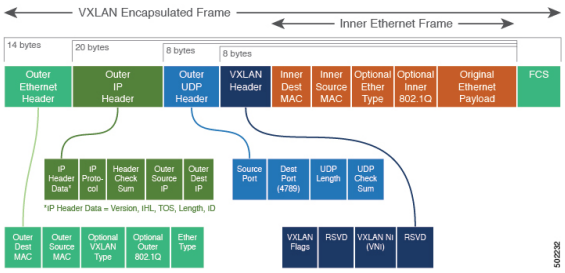
VXLAN は、レイヤ 3 ネットワーク上のレイヤ 2 オーバーレイ方式です。VXLAN は MAC Address-in-User Datagram Protocol（MAC-in-UDP）のカプセル化を使用して、データセンター ネットワークでレイヤ 2 セグメントを拡張する方法を提供します。VXLAN は、共有される共

通の物理インフラストラクチャにおいて、柔軟で大規模なマルチテナント環境をサポートするためのソリューションです。物理データセンター ネットワークでの転送プロトコルは IP と UDP です。

VXLAN は MAC-in-UDP のカプセル化方式を定義します。この方式において、元のレイヤ 2 フレームに VXLAN ヘッダーが追加され、UDP-IP パケットに置かれます。この MAC-in-UDP のカプセル化によって、VXLAN はレイヤ 3 ネットワーク上でレイヤ 2 ネットワークをトンネルします。

VXLAN は、24 ビット VNID といくつかの予約ビットで構成される 8 バイト VXLAN ヘッダーを使用します。VXLAN ヘッダーおよび元のイーサネット フレームは、UDP ペイロードに入ります。24 ビット VNID は、レイヤ 2 セグメントを識別し、セグメント間でレイヤ 2 の分離を維持するために使用されます。VNID のすべての 24 ビットを使用して、VXLAN は 1600 万個の LAN セグメントをサポートできます。

図 *1 :*



VXLAN トンネル

内部イーサネット フレームをカプセル化およびカプセル化解除する2つのデバイス間の VXLAN カプセル化通信は、VXLAN トンネルと呼ばれます。VXLAN トンネルは UDP カプセル化されているため、ステートレスです。

VXLAN トンネル エンドポイント

VXLAN トンネル エンドポイント（VTEP）は、VXLAN トンネルを終端するデバイスです。VXLAN カプセル化と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。1 つは、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 セグメントのレイヤ 2 インターフェイスです。もう 1 つは、IP トランスポート ネットワーク上のレイヤ 3 インターフェイスです。

IP インターフェイスには、トランスポート IP ネットワークの VTEP を識別する一意の IP アドレスがあります。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。VTEP は、ローカルに接続されている同じVNIを共有する他の VTEP デバイスを検出します。ローカルに接続された MAC アドレスをピアにアドバタイズします。また、IP インターフェイスを介してリモート MAC アドレスから VTEP へのマッピングも学習します。

アンダーレイ ネットワーク

VXLAN セグメントは、基盤となる物理ネットワーク トポロジに依存しません。逆に、アンダーレイ ネットワークとも呼ばれる基盤となる IP ネットワークは、VXLAN オーバーレイから独立しています。アンダーレイ ネットワークは、外部 IP アドレス ヘッダーに基づいて VXLANカプセル化パケットを転送します。カプセル化されたパケットは、発信元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ルーティングされます。

VXLANファブリックのアンダーレイの主な目的は、仮想トンネルエンドポイント（VTEP）の到達可能性をアドバタイズすることです。アンダーレイは、VXLANトラフィックの高速で信頼性の高い転送も提供します。

オーバーレイネットワーク

ブロードキャストの用語では、オーバーレイはアンダーレイ ネットワーク インフラストラクチャ上に構築される仮想ネットワークです。VXLANファブリックでは、オーバーレイ ネットワークはコントロール プレーンと VXLAN トンネルで構築されます。コントロール プレーンは、MAC アドレスの到達可能性をアドバタイズするために使用されます。VXLAN トンネルは、VTEP 間でイーサネット フレームを転送します。

分散型エニーキャスト ゲートウェイ

分散型エニーキャスト ゲートウェイとは、VNI の一部であるすべてのリーフで同じ IP アドレスと MAC アドレスを使用するデフォルト ゲートウェイ アドレッシングの使用を指します。そのため、直接接続されているワークロードのデフォルト ゲートウェイとしてすべてのVTEPが機能します。分散エニーキャスト ゲートウェイ機能は、ワークロード配置の柔軟化および VXLAN ファブリック全体でのトラフィックの最適化を促進するために使用されます。

コントロール プレーン

VXLAN で使用される、広く採用されている 2 つのコントロール プレーンがあります。

フラッディングおよび学習マルチキャスト ベースのラーニング コントロール プレーン

Cisco Nexus 9000 シリーズ スイッチは、フラッディングおよびマルチキャスト ベースのコントロール プレーン方式をサポートします。

- マルチキャスト ベースのコントロールプレーンでVXLANを設定すると、特定のVXLAN VNI で設定されたすべての VTEP が同じマルチキャスト グループに参加します。各 VNI が独自のマルチキャスト グループを持つことも、複数の VNI が同じグループを共有することもできます。

- マルチキャストは、VNIに対して、ブロードキャスト、Unknownユニキャスト、およびマルチキャスト（BUM）トラフィックを転送するために使用されます。
- マルチキャスト設定は、Any-Source Multicast（ASM）またはPIM BiDirをサポートする必要があります。
- 最初、VTEPは、直接接続されているデバイスのMACアドレスのみを学習します。
- リモートMACアドレスからVTEPへのマッピングは、会話型学習によって学習されます。

VXLAN BGP EVPN コントロール プレーン

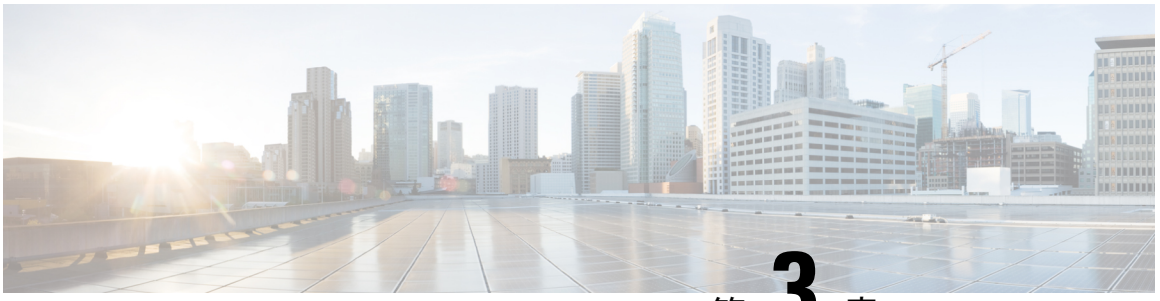
Cisco Nexus 9000シリーズ スイッチは、Multiprotocol Border Gateway Protocol（MPBGP）イーサネットVPN（EVPN）コントロールプレーンを提供するように設定できます。コントロールプレーンは、レイヤ2およびレイヤ3 VXLANオーバーレイ ネットワークを備えた分散型エニーキャスト ゲートウェイを使用します。

MPBGP EVPNコントロール プレーンでは、データセンター ネットワークについて、次のものが提供できます。

- データセンターネットワークの物理トポロジに制限されない、柔軟なワークロード配置。
  - データセンター ファブリック内の任意の場所に仮想マシンを配置します。
- データセンター内部およびデータセンター間における最適なサーバ間East-Westトラフィック。
  - サーバ/仮想マシン間のEast-Westトラフィックは、ファースト ホップ ルータでのほぼ特定されたルーティングで達成されます。ファースト ホップ ルーティングはアクセス レイヤで行われます。ホスト ルートの交換は、サーバまたはホストへの流入と送出に関するルーティングがほぼ特定されるようにする必要があります。仮想マシン（VM）モビリティは、新しいMACアドレス/IPアドレスがローカル スイッチに直接接続されている場合に、新しいエンドポイント接続を検出することでサポートされます。ローカルスイッチは新しいMAC/IPを検出すると、ネットワークの残りの部分に新しいロケーションを通知します。
- データセンターでのフラッディングの解消または削減。
  - フラッディングの削減は、MAC 到達可能性情報をMP-BGP EVPN経由で配信してL2不明ユニキャスト トラフィックに関連したフラッディングを最適化することで行われます。ARP/IPv6ネイバー要請に関連するブロードキャストの削減の最適化は、MPBGP EVPNを介して必要な情報を配信することによって実現されます。情報はアクセス スイッチでキャッシュされます。アドレス送信要求は、ファブリックの他の部分にブロードキャストを送信せずにローカルで応答できます。
- 特定のファブリック コントローラから独立して展開可能な標準ベースのコントロール プレーン。
  - MPBGP EVPN コントロール プレーンのアプローチで得られるもの：

- 特定のトンネルエンドポイントの背後にあるホストおよびセグメントに関連付けられたトンネル エンドポイントへの IP 到達可能性情報。
- ホスト MAC への到達可能性の配信による不明ユニキャスト フラッディングの削減/削除。
- ホスト IP/MAC バインディングの配信によるローカル ARP の抑制。
- ホスト モビリティ。
- シングル アドレス ファミリ（MPBGP EVPN）による L2 と L3 の両方のルート到達可能性情報の配信。

- レイヤ 2 およびレイヤ 3 トラフィックのセグメンテーション。
  - VXLAN カプセル化を使用したトラフィック セグメンテーションが行われ、そこでは VNI がセグメント識別子として機能します。



第 3 章

アンダーレイの設定

この章は、次の内容で構成されています。

IP ファブリック アンダーレイ

アンダーレイの考慮事項

ユニキャスト アンダーレイ：

VXLAN EVPN ファブリックのアンダーレイの主な目的は、仮想トンネル エンドポイント（VTEP）および BGP ピアリング アドレスの到達可能性をアドバタイズすることです。アンダーレイ プロトコルを選択する主な基準は、ノード障害時の高速コンバージェンスです。その他の基準は次のとおりです。

- 設定の簡素化。
- 起動時にネットワークへのノードの展開を遅らせる機能。

このドキュメントでは、Cisco でサポートおよびテストされている 2 つの主要なプロトコルである IS-IS と OSPF について詳しく説明します。また、VXLAN EVPN ファブリックのアンダーレイとしての eBGP プロトコルの使用についても説明します。

アンダーレイ/オーバーレイの観点から見ると、サーバから Virtual Extensible LAN（VXLAN）ファブリック上の別のサーバへのパケット フローは、次のように説明されます。

1. サーバーは、送信元 VXLAN トンネル エンドポイント（VTEP）にトラフィックを送信します。VTEP は、宛先 MAC に基づいてレイヤ 2 またはレイヤ 3 通信を実行し、ネクストホップ（宛先 VTEP）を取得します。

（注） パケットがブリッジされると、ターゲット エンド ホストの MAC アドレスが内部フレームの DMAC フィールドにスタンプされます。パケットがルーティングされると、デフォルト ゲートウェイの MAC アドレスが内部フレームの DMAC フィールドにスタンプされます。

2. VTEP はトラフィック（フレーム）を VXLAN パケットにカプセル化し（オーバーレイ機能。図 1 を参照）、アンダーレイ IP ネットワークに信号を送ります。
3. アンダーレイ ルーティング プロトコルに基づいて、パケットは IP ネットワークを介して送信元 VTEP から宛先 VTEP に送信されます（アンダーレイ機能。アンダーレイの概要図を参照）。
4. 宛先 VTEP は VXLAN カプセル化（オーバーレイ機能）を削除し、目的のサーバにトラフィックを送信します。

VTEP は、アンダーレイネットワークの一部でもあります。これは、IP アンダーレイ ネットワークを介して VXLAN カプセル化トラフィックを送信するために、VTEP が相互に到達可能である必要があるためです。

*[*オーバーレイの概要（*Overlay Overview*）*]* と *[*アンダーレイの概要（*Underlay Overview*）*]* の画像（下記）は、オーバーレイとアンダーレイの大きな違いを示しています。VTEP に焦点が当てられているため、スパイン スイッチはバックグラウンドでのみ表示されます。リアルタイムでは、VTEP から VTEP へのパケット フローがスパイン スイッチを通過することに注意してください。

図 *2:* オーバーレイの概要

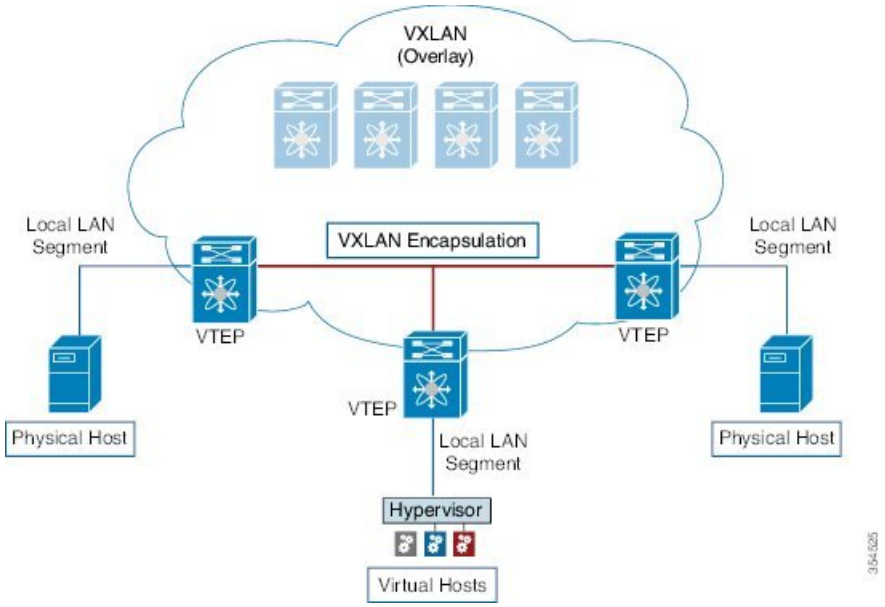
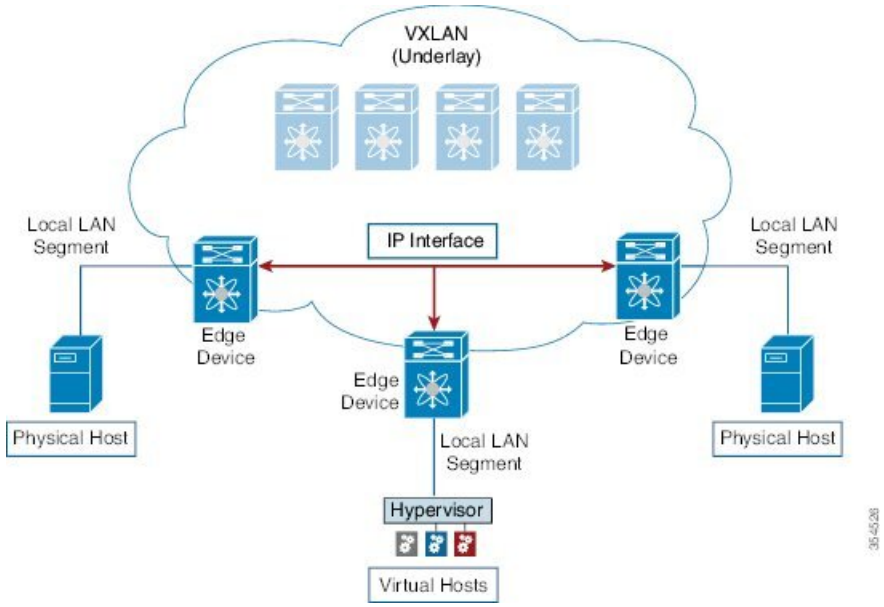


図 *3:* アンダーレイの概要



VXLAN EVPN プログラマブル ファブリックのアンダーレイ *IP* ネットワークの導入に関する考慮事項

VXLAN EVPN プログラマブル ファブリックのアンダーレイ IP ネットワークの導入に関する考慮事項は次のとおりです。

- 最大伝送ユニット（MTU）：VXLAN のカプセル化により、MTU の要件が大きくなるので、潜在的なフラグメンテーションを回避する必要があります。
  - VTEP 間のパス上の各インターフェイスで 9216 バイトの MTU を使用すると、サーバの最大 MTU + VXLAN オーバーヘッドに対応できます。ほとんどのデータセンターサーバ NIC は最大 9000 バイトをサポートします。したがって、VXLAN トラフィックにフラグメンテーションは必要ありません。
  - VXLAN IP ファブリック アンダーレイは、IPv4 アドレス ファミリをサポートします。
- ユニキャスト ルーティング：任意のユニキャスト ルーティング プロトコルを VXLAN IP アンダーレイに使用できます。VTEP 間のルーティングには、OSPF、IS-IS、または eBGP を実装できます。

（注） ベストプラクティスとして、シンプルな IGP（OSPF または IS-IS）を使用して、オーバーレイ情報交換用の iBGP を使用した VTEP 間のアンダーレイ到達可能性を確認します。

- IP アドレッシング：ポイントツーポイント（P2P）または IP アンナンバード リンク。リーフ スイッチ ノードとスパイン スイッチ ノード間の例として、ポイントツーポイントリンクごとに、通常 /30 IP マスクを割り当てる必要があります。オプションで、/31 マスクま

たは IP アンナンバード リンクを割り当てることができます。IP アンナンバード アプローチは、アドレッシングの観点から見ると、より少ない IP アドレスを使用します。OSPF または IS-IS プロトコル アンダーレイの IP アンナンバード オプションは、IP アドレスの使用を最小限に抑えます。

/31 ネットワーク：OSPF または IS-IS のポイントツーポイントの番号付きネットワークは、2 つのスイッチ（インターフェイス）間にのみ存在し、ブロードキャストまたはネットワーク アドレスは必要ありません。したがって、このネットワークには /31 ネットワークで十分です。このネットワーク上のネイバーは隣接関係を確立し、ネットワークの指定ルータ（DR）はありません。

（注） VXLAN アンダーレイの IP アンナンバードは、Cisco NX-OS リリース 7.0(3)I7(2) 以降でサポートされます。同じデバイス間の単一のアンナンバード リンク（たとえば、スパインからリーフ）だけがサポートされます。複数の物理リンクが同じリーフとスパインを接続している場合は、アンナンバード リンクを持つ単一の L3 ポートチャネルを使用する必要があります。

- マルチ宛先（BUM）トラフィック用のマルチキャスト プロトコル：VXLAN には BGP EVPN コントロール プレーンがありますが、VXLAN ファブリックにはブロードキャスト/不明なユニキャスト/マルチキャスト（BUM）トラフィックを転送するためのテクノロジーが必要です。
- PIM Bidir は、Cisco Nexus 9300-EX/FX/FX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、PIM Bidir は、Cisco Nexus 9300-FX3/GX/GX2/H2R/H1 プラットフォーム スイッチ、および 9700-GX ライン カードを備えた 9500 スイッチでサポートされます。
- vPC 構成：これは、 *Cisco Nexus 9000* シリーズ *NX-OS* インターフェイス構成ガイドの **vPC の構成**に記載されています。

ユニキャスト ルーティングおよび IP アドレッシング オプション

各ユニキャスト ルーティング プロトコル オプション（OSPF、IS-IS、および eBGP）と設定例を次に示します。セットアップの要件に合わせてオプションを使用します。

重要 すべてのルーティング設定サンプルは IP アンダーレイの観点からのものであり、包括的なものではありません。ルーティング プロセス、認証、双方向フォワーディング検出（BFD）情報などの詳細な構成情報については、*Cisco Nexus 9000* シリーズ *NX-OS* ユニキャスト ルーティング構成ガイドを参照してください。

OSPF アンダーレイ IP ネットワーク

いくつかの考慮事項を次に示します。

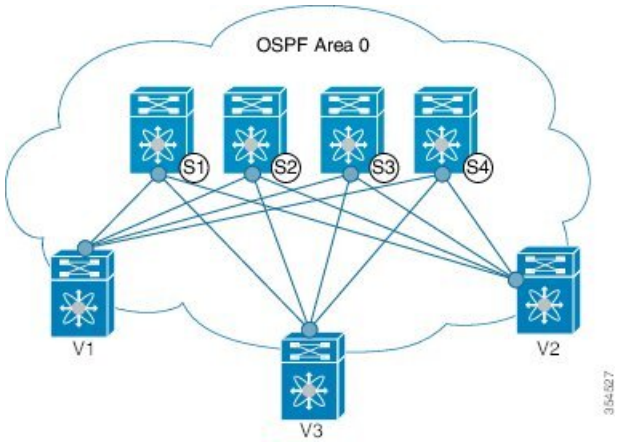
- IP アドレッシングには、P2P リンクを使用します。2 つのスイッチだけが直接接続されているため、指定ルータ/バックアップ指定ルータ（DR/BDR）の選択を回避できます。
- ポイントツーポイントネットワーク タイプ オプションを使用します。ルーテッドインターフェイスまたはポートに最適であり、リンク ステート アドバタイズメント（LSA）の観点から最適です。
- ブロードキャスト タイプのネットワークは使用しないでください。LSA データベースの観点からは最適ではなく（LSA タイプ 1：ルータ LSA および LSA タイプ 2：ネットワーク LSA）、DR/BDR の選択が必要になるため、追加の選択とデータベース オーバーヘッドが発生します。

（注） ルーティング ドメインのサイズに多数のルータや IP プレフィックスが含まれている場合は、OSPF ネットワークをエリアに分割できます。規模と設定に関する一般的な OSPF のベスト プラクティスのルールは、VXLAN アンダーレイにも適用できます。たとえば、LSA タイプ 1 およびタイプ 2 はエリア外にフラッディングされません。複数のエリアがある場合、OSPF LSA データベースのサイズを縮小して、CPU とメモリの消費を最適化できます。

（注）
- 使いやすくするために、各設定の最初に、タスクの設定を開始する必要がある設定モードが記載されています。
- イメージのトポロジの一部について、設定タスクと対応する show コマンドの出力が表示されます。たとえば、リーフ スイッチと接続されたスパイン スイッチの設定例が示されている場合、その設定の show コマンド出力には対応する設定が表示されます。

OSPF の設定例：P2P および IP アンナンバード ネットワークのシナリオ

図 *4:* アンダーレイ ルーティング プロトコルとしての *OSPF*



OSPF – /31マスクを使用したP2Pリンクシナリオ

上の図では、リーフ スイッチ（V1、V2、V3）が画像の下部にあります。これらは、画像の上部に示されている 4 つのスパイン スイッチ（S1、S2、S3、および S4）に接続されています。リーフ スイッチ（VTEP機能もある）と各スパイン間のP2P接続の場合、リーフ スイッチV1、V2、および V3 を各スパイン スイッチに接続する必要があります。

V1 では、各スパイン スイッチに接続するように P2P インターフェイスを設定する必要があります。

リーフ スイッチ（V1）インターフェイスとスパイン スイッチ（S1）インターフェイス間のサンプル P2P 設定を次に示します。

リーフ スイッチ V1 の OSPF グローバル設定

(config) #

```
feature ospf
router ospf UNDERLAY
 router-id 10.1.1.54
```

OSPF リーフ スイッチ V1 P2P インターフェイスの設定

(config) #

```
interface Ethernet 1/41
   description Link to Spine S1
   no switchport
   ip address 198.51.100.1/31
   mtu 9192
   ip router ospf UNDERLAY area 0.0.0.0
   ip ospf network point-to-point
```

ip ospf network point-to-point コマンドは、OSPF ネットワークをポイントツーポイント ネットワークとして設定します。

OSPFインスタンスは、リコールを改善するためにUNDERLAY としてタグ付けされています。

OSPF ループバック インターフェイス コンフィギュレーション（リーフ スイッチ V1）

リーフ スイッチ V1 の OSPF ルータ ID として使用できるように、ループバック インターフェイスを設定します。

(config) #

```
interface loopback 0
   ip address 10.1.1.54/32
   ip router ospf UNDERLAY area 0.0.0.0
```

インターフェイスは、OSPF インスタンスの UNDERLAY および OSPF エリア 0.0.0.0 に関連付けられます。

スパイン スイッチ S1 の OSPF グローバル設定

(config) #

```
feature ospf
router ospf UNDERLAY
 router-id 10.1.1.53
```

（対応する）OSPFスパイン スイッチ S1 P2P インターフェイス設定

(config) #

```
interface Ethernet 1/41
   description Link to VTEP V1
   ip address 198.51.100.2/31
   mtu 9192
   ip router ospf UNDERLAY area 0.0.0.0
   ip ospf network point-to-point
   no shutdown
```

（注） リンクの両端の MTU サイズは同じに設定する必要があります。

OSPF ループバック インターフェイスの設定（スパイン スイッチ S1）

スパイン スイッチ S1 の OSPF ルータ ID として使用できるように、ループバック インターフェイスを設定します。

(config) #

```
interface loopback 0
   ip address 10.1.1.53/32
   ip router ospf UNDERLAY area 0.0.0.0
```

インターフェイスは、OSPF インスタンスの UNDERLAY および OSPF エリア 0.0.0.0 に関連付けられます。

.

.

「アンダーレイ ルーティング プロトコルとして *OSPF*」イメージの *OSPF* トポロジ設定を完了するには、次のように設定します。

- 残りの *3* つのスパイン スイッチへの *3* つの *V1* インターフェイス（または *3* つの *P2P* リンク）。
- *V2*、*V3*、*V4* とスパイン スイッチ間の *P2P* リンクを接続する手順を繰り返します。

OSPF-IP アンナンバード シナリオ

次に、OSPF IP アンナンバード設定の例を示します。

OSPF リーフ スイッチ V1 の設定

リーフ スイッチ V1 の OSPF グローバル設定

(config) #

```
feature ospf
router ospf UNDERLAY
  router-id 10.1.1.54
```

OSPFインスタンスは、リコールを改善するためにUNDERLAYとしてタグ付けされています。

OSPF リーフ スイッチ V1 P2P インターフェイスの設定

(config) #

```
interface Ethernet1/41
  description Link to Spine S1
  mtu 9192
  ip ospf network point-to-point
  ip unnumbered loopback0
  ip router ospf UNDERLAY area 0.0.0.0
```

ip ospf network point-to-point コマンドは、OSPF ネットワークをポイントツーポイント ネットワークとして設定します。

OSPF ループバック インターフェイスの設定

リーフ スイッチ V1 の OSPF ルータ ID として使用できるように、ループバック インターフェイスを設定します。

(config) #

```
interface loopback0
  ip address 10.1.1.54/32
  ip router ospf UNDERLAY area 0.0.0.0
```

インターフェイスは、OSPF インスタンスの UNDERLAY および OSPF エリア 0.0.0.0 に関連付けられます。

OSPF スパイン スイッチ S1 の設定：

スパイン スイッチ S1 の OSPF グローバル設定

(config) #

```
feature ospf
```

```
router ospf UNDERLAY
  router-id 10.1.1.53
```

（対応する）OSPF スパイン スイッチ S1 P2P インターフェイス設定

(config) #

```
interface Ethernet1/41
  description Link to VTEP V1
  mtu 9192
  ip ospf network point-to-point
  ip unnumbered loopback0
  ip router ospf UNDERLAY area 0.0.0.0
```

OSPF ループバック インターフェイス設定（スパイン スイッチ S1）

スパイン スイッチ S1 の OSPF ルータ ID として使用できるように、ループバック インターフェイスを設定します。

(config) #

```
interface loopback0
  ip address 10.1.1.53/32
  ip router ospf UNDERLAY area 0.0.0.0
```

インターフェイスは、OSPF インスタンスの UNDERLAY および OSPF エリア 0.0.0.0 に関連付けられます。

.

.

「アンダーレイ ルーティング プロトコルとしての OSPF」イメージの OSPF トポロジ設定を完了するには、次のように設定します。

- 残りの *3* つのスパイン スイッチへの *3* つの *VTEP V1* インターフェイス（または *3* つの *IP* アンナンバード リンク）。
- *VTEP V2*、*V3*、および *V4* とスパイン スイッチ間の *IP* アンナンバード リンクを接続する手順を繰り返します。

OSPF 検証

OSP F設定を確認するには、次のコマンドを使用します。

```
Leaf-Switch-V1# show ip ospf

 Routing Process UNDERLAY with ID 10.1.1.54 VRF default
 Routing Process Instance Number 1
 Stateful High Availability enabled
 Graceful-restart is configured
   Grace period: 60 state: Inactive
   Last graceful restart exit status: None
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Administrative distance 110
 Reference Bandwidth is 40000 Mbps
 SPF throttling delay time of 200.000 msecs,
   SPF throttling hold time of 1000.000 msecs,
   SPF throttling maximum wait time of 5000.000 msecs
```

```
 LSA throttling start time of 0.000 msecs,
   LSA throttling hold interval of 5000.000 msecs,
   LSA throttling maximum wait time of 5000.000 msecs
 Minimum LSA arrival 1000.000 msec
 LSA group pacing timer 10 secs
 Maximum paths to destination 8
 Number of external LSAs 0, checksum sum 0
 Number of opaque AS LSAs 0, checksum sum 0
 Number of areas is 1, 1 normal, 0 stub, 0 nssa
 Number of active areas is 1, 1 normal, 0 stub, 0 nssa
 Install discard route for summarized external routes.
 Install discard route for summarized internal routes.
   Area BACKBONE(0.0.0.0)
        Area has existed for 03:12:54
        Interfaces in this area: 2 Active interfaces: 2
        Passive interfaces: 0  Loopback interfaces: 1
        No authentication available
        SPF calculation has run 5 times
         Last SPF ran for 0.000195s
        Area ranges are
        Number of LSAs: 3, checksum sum 0x196c2
```

```
Leaf-Switch-V1# show ip ospf interface

loopback0 is up, line protocol is up
    IP address 10.1.1.54/32
    Process ID UNDERLAY VRF default, area 0.0.0.0
    Enabled by interface configuration
    State LOOPBACK, Network type LOOPBACK, cost 1
    Index 1
 Ethernet1/41 is up, line protocol is up
    Unnumbered interface using IP address of loopback0 (10.1.1.54)
    Process ID UNDERLAY VRF default, area 0.0.0.0
    Enabled by interface configuration
    State P2P, Network type P2P, cost 4
    Index 2, Transmit delay 1 sec
    1 Neighbors, flooding to 1, adjacent with 1
    Timer intervals: Hello 10, Dead 40, Wait 40, Retransmit 5
      Hello timer due in 00:00:07
    No authentication
    Number of opaque link LSAs: 0, checksum sum 0
```

```
Leaf-Switch-V1# show ip ospf neighbors

OSPF Process ID UNDERLAY VRF default
 Total number of neighbors: 1
 Neighbor ID     Pri State            Up Time  Address         Interface
 10.1.1.53         1 FULL/ -          06:18:32 10.1.1.53        Eth1/41
```

コマンドの詳細なリストについては、『Configuration and Command Reference』ガイドを参照してください。

IS-IS アンダーレイ IP ネットワーク

考慮事項を次に示します。

- IS-ISはConnectionless Network Service（CLNS）を使用し、IP から独立しているため、リンクが変更されたときに完全な SPF 計算が回避されます。
- ネット ID：各 IS-IS インスタンスには、エリア内の IS-IS インスタンスを一意に識別するネットワーク エンティティ タイトル（NET）ID が関連付けられています。NET ID は、そ

の IS-IS インスタンスをエリア内で一意に特定する IS-IS システム ID とエリア ID からなります。たとえば、NET ID が 49.0001.0010.0100.1074.00 の場合、システム ID は 0010.0100.1074 で、エリア ID は 49.0001 です。

重要

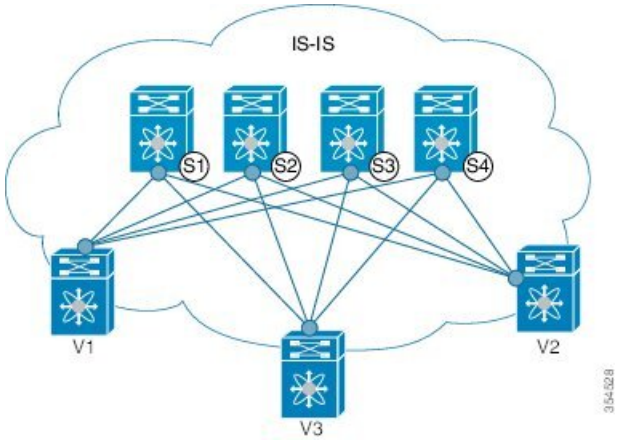
ファブリック内のレベル 1 IS-IS：シスコは、プログラマブルファブリック内のすべてのノードで、IS-IS レベル 1 のみの設定と IS-IS レベル 2 のみの設定の使用を検証しています。ファブリックは、すべてのノードがファブリック内の他のすべてのノードへの最適パスを必要とするスタブ ネットワークと見なされます。Cisco NX-OS IS-IS の実装は、ファブリック内の多数のノードをサポートするように拡張できます。したがって、ファブリックを複数の IS-IS ドメインに分割する必要はありません。

（注）

- 使いやすくするために、各設定の最初に、タスクの設定を開始する必要がある設定モードが記載されています。
- イメージのトポロジの一部について、設定タスクと対応する show コマンドの出力が表示されます。たとえば、リーフ スイッチと接続されたスパイン スイッチの設定例が示されている場合、その設定の show コマンド出力には対応する設定が表示されます。

IS-IS の設定例：P2P および IP アンナンバード ネットワークのシナリオ

図 *5:* アンダーレイ ルーティング プロトコルとしての *IS-IS*



上記の図では、リーフスイッチ（V1、V2、およびV3、VTEP機能）が画像の下部にあります。これらは、イメージの上部に示されている4つのスパインスイッチ（S1、S2、S3、およびS4）に接続されています。

IS-IS – /31マスクを使用した P2P リンク シナリオ

V1 とスパイン スイッチ S1 間の P2P の設定例を次に示します。

リーフ スイッチと各スパイン スイッチ間の P2P 接続の場合、V1、V2、および V3 を各スパイン スイッチに接続する必要があります。

V1 では、S1 に接続するためにループバック インターフェイスと P2P インターフェイスを設定する必要があります。リーフ スイッチ（V1）インターフェイスとスパイン スイッチ（S1）インターフェイス間のサンプル P2P 設定を次に示します。

リーフ スイッチ V1 の IS-IS 設定

IS-IS グローバル設定

(config) #

```
feature isis
router isis UNDERLAY
  net 49.0001.0010.0100.1074.00
  is-type level-1
  set-overload-bit on-startup 60
```

過負荷ビットの設定：最短パス優先（SPF）の計算で中間ホップとしてこのルータを使用しないことを他のルータに通知するように、Cisco Nexus スイッチを設定できます。任意で、起動時に一時的に過負荷ビットを設定することもできます。上記の例では、**set-overload-bit** コマンドを使用して、起動時の過負荷ビットを 60 秒に設定しています。

IS-IS P2P インターフェイス コンフィギュレーション（リーフスイッチ V1）

(config) #

```
interface Ethernet 1/41
   description Link to Spine S1
   mtu 9192
   ip address 209.165.201.1/31
   ip router isis UNDERLAY
```

IS-IS ループバック インターフェイスの設定（リーフ スイッチ V1）

ループバック インターフェイスを、リーフ スイッチ V1 の IS-IS ルータ ID として使用できるように設定します。

(config) #

```
interface loopback 0
   ip address 10.1.1.74/32
   ip router isis UNDERLAY
```

IS-IS インスタンスは、より良いリコールのために UNDERLAY としてタグ付けされます。

（対応する）IS-IS スパインスイッチ S1 の設定

IS-IS グローバル コンフィギュレーション

(config) #

```
feature isis
router isis UNDERLAY
 net 49.0001.0010.0100.1053.00
 is-type level-1
 set-overload-bit on-startup 60
```

IS-IS P2P インターフェイス コンフィギュレーション（スパイン スイッチ S1）

(config) #

```
interface Ethernet 1/1
   description Link to VTEP V1
   ip address 209.165.201.2/31
   mtu 9192
   ip router isis UNDERLAY
```

IS-IS ループバック インターフェイスの設定（スパイン スイッチ S1）

(config) #

```
interface loopback 0
   ip address 10.1.1.53/32
   ip router isis UNDERLAY
.
.
```

上記のイメージの *IS-IS* トポロジ設定を完了するには、次のように設定します。

- さらに 3 つのリーフ スイッチ V1 のインターフェイス（または 3 つの P2P リンク）。
- リーフ スイッチ V2、V3、V4 とスパイン スイッチ間の P2P リンクを接続する手順を繰り返します。

IS-IS-IP アンナンバード シナリオ

リーフ スイッチ V1 の IS-IS 設定

IS-IS グローバル設定

(config)#

```
feature isis
router isis UNDERLAY
  net 49.0001.0010.0100.1074.00
  is-type level-1
  set-overload-bit on-startup 60
```

IS-IS インターフェイス設定（リーフ スイッチ V1）

(config) #

```
interface Ethernet1/41
  description Link to Spine S1
  mtu 9192
  medium p2p
```

```
  ip unnumbered loopback0
  ip router isis UNDERLAY
```

IS-IS ループバック インターフェイスの設定（リーフ スイッチ V1）

(config)

```
interface loopback0
  ip address 10.1.1.74/32
  ip router isis UNDERLAY
```

スパインスイッチ S1 の IS-IS 設定

IS-IS グローバル設定

(config)#

```
feature isis
router isis UNDERLAY
  net 49.0001.0010.0100.1053.00
  is-type level-1
  set-overload-bit on-startup 60
```

IS-ISインターフェイス設定（スパイン スイッチ S1）

(config)#

```
interface Ethernet1/41
  description Link to V1
  mtu 9192
  medium p2p
  ip unnumbered loopback0
  ip router isis UNDERLAY
```

IS-IS ループバック インターフェイスの設定（スパイン スイッチ S1）

(config)#

```
interface loopback0
  ip address 10.1.1.53/32
  ip router isis UNDERLAY
```

IS-IS 検証

リーフ スイッチ V1 の IS-IS 設定を確認するには、次のコマンドを使用します。

```
Leaf-Switch-V1# show isis

ISIS process : UNDERLAY
 Instance number :  1
 UUID: 1090519320
 Process ID 20258
VRF: default
  System ID : 0010.0100.1074  IS-Type : L1
  SAP : 412  Queue Handle : 15
  Maximum LSP MTU: 1492
  Stateful HA enabled
  Graceful Restart enabled. State: Inactive
  Last graceful restart status : none
  Start-Mode Complete
```

```
  BFD IPv4 is globally disabled for ISIS process: UNDERLAY
  BFD IPv6 is globally disabled for ISIS process: UNDERLAY
  Topology-mode is base
  Metric-style : advertise(wide), accept(narrow, wide)
  Area address(es) :
    49.0001
 Process is up and running
  VRF ID: 1
  Stale routes during non-graceful controlled restart
  Interfaces supported by IS-IS :
    loopback0
    loopback1
    Ethernet1/41
  Topology : 0
  Address family IPv4 unicast :
    Number of interface : 2
    Distance : 115
  Address family IPv6 unicast :
    Number of interface : 0
    Distance : 115
  Topology : 2
  Address family IPv4 unicast :
    Number of interface : 0
    Distance : 115
  Address family IPv6 unicast :
    Number of interface : 0
    Distance : 115
   Level1
  No auth type and keychain
  Auth check set
  Level2
  No auth type and keychain
  Auth check set
  L1 Next SPF: Inactive
  L2 Next SPF: Inactive

Leaf-Switch-V1# show isis interface

IS-IS process: UNDERLAY VRF: default
loopback0, Interface status: protocol-up/link-up/admin-up IP address: 10.1.1.74, IP
subnet: 10.1.1.74/32
IPv6 routing is disabled Level1
No auth type and keychain Auth check set
Level2
No auth type and keychain Auth check set
Index: 0x0001, Local Circuit ID: 0x01, Circuit Type: L1 BFD IPv4 is locally disabled for
 Interface loopback0 BFD IPv6 is locally disabled for Interface loopback0 MTR is disabled
Level Metric 1  1
2 1
Topologies enabled:
    L  MT  Metric  MetricCfg  Fwdng IPV4-MT  IPV4Cfg  IPV6-MT  IPV6Cfg
    1  0        1       no   UP    UP       yes      DN       no
    2  0        1       no   DN    DN       no       DN       no

loopback1, Interface status: protocol-up/link-up/admin-up
  IP address: 10.1.2.74, IP subnet: 10.1.2.74/32
  IPv6 routing is disabled
  Level1
    No auth type and keychain
    Auth check set
  Level2
    No auth type and keychain
    Auth check set
  Index: 0x0002, Local Circuit ID: 0x01, Circuit Type: L1
  BFD IPv4 is locally disabled for Interface loopback1
```

```
  BFD IPv6 is locally disabled for Interface loopback1
  MTR is disabled
  Passive level: level-2
  Level      Metric
  1               1
  2               1
  Topologies enabled:
    L  MT  Metric  MetricCfg  Fwdng IPV4-MT  IPV4Cfg  IPV6-MT  IPV6Cfg
    1  0        1       no   UP    UP       yes      DN       no
    2  0        1       no   DN    DN       no       DN       no

Ethernet1/41, Interface status: protocol-up/link-up/admin-up
  IP unnumbered interface (loopback0)
  IPv6 routing is disabled
    No auth type and keychain
    Auth check set
  Index: 0x0002, Local Circuit ID: 0x01, Circuit Type: L1
  BFD IPv4 is locally disabled for Interface Ethernet1/41
  BFD IPv6 is locally disabled for Interface Ethernet1/41
  MTR is disabled
  Extended Local Circuit ID: 0x1A028000, P2P Circuit ID: 0000.0000.0000.00
  Retx interval: 5, Retx throttle interval: 66 ms
  LSP interval: 33 ms, MTU: 9192
  P2P Adjs: 1, AdjsUp: 1, Priority 64
  Hello Interval: 10, Multi: 3, Next IIH: 00:00:01
  MT    Adjs   AdjsUp  Metric   CSNP  Next CSNP  Last LSP ID
  1          1        1       4     60  00:00:35   ffff.ffff.ffff.ff-ff
  2          0        0       4     60  Inactive   ffff.ffff.ffff.ff-ff
  Topologies enabled:
    L  MT  Metric  MetricCfg  Fwdng IPV4-MT  IPV4Cfg  IPV6-MT  IPV6Cfg
    1  0        4       no   UP    UP       yes      DN       no
    2  0        4       no   UP    DN       no       DN       no

Leaf-Switch-V1# show isis adjacency

IS-IS process: UNDERLAY VRF: default
IS-IS adjacency database:
Legend: '!': No AF level connectivity in given topology
System ID          SNPA    Level  State Hold Time  Interface
Spine-Switch-S1    N/A     1      UP    00:00:23   Ethernet1/41
```

コマンドの詳細なリストについては、『Configuration and Command Reference』ガイドを参照してください。

eBGP アンダーレイ IP ネットワーク

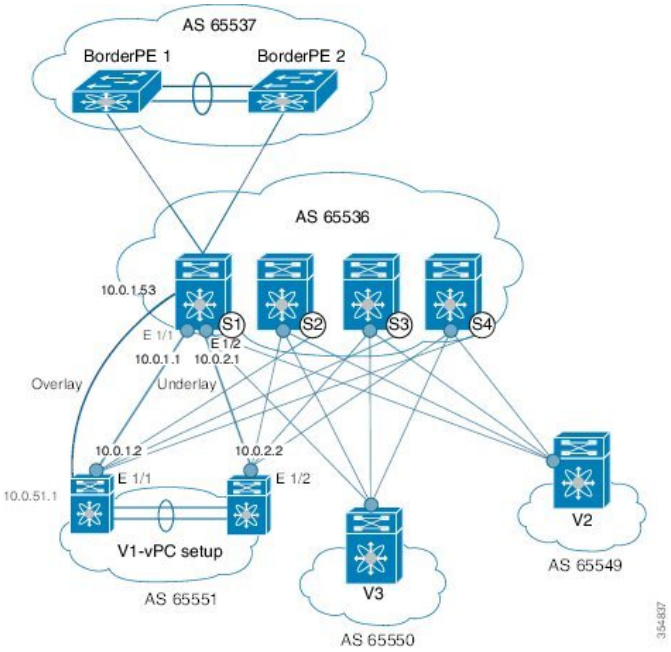
一部のお客様は、ネットワークでのサポートが必要なプロトコルの数を含めるために、アンダーレイとオーバーレイに同じプロトコルを使用したいと考えています。

eBGP ベースのアンダーレイを設定するには、さまざまな方法があります。この項で説明する設定は、機能とコンバージェンスについて検証済みです。eBGP に基づく IP アンダーレイは、次に説明する設定で構築できます。（参考：以下の画像を参照）

- 次の設計は、マルチ AS モデルに従っています。
- eBGP アンダーレイでは、リーフ ノードとスパイン ノードの間に番号付きインターフェイスが必要です。ピアの到達可能性を配布する他のプロトコルがないため、アンダーレイ BGP セッションには番号付きインターフェイスが使用されます。

- オーバーレイ セッションはループバック アドレスで設定されます。これは、リンクまたはノードの障害が発生した場合の復元力を向上させるためです。
- スパイン層の BGP スピーカーは、すべてのリーフ ノード eBGP ネイバーを個別に設定します。これは、ダイナミック BGP でカバーできる IBGP ベースのピアリングとは異なります。
- ファブリック内の複数の AS 番号のポインタを次に示します。
  - BGP スピーカーとして設定されたすべてのスパイン ノードは、1 つの AS 内にあります。
  - すべてのリーフ ノードには、スパイン層の BGP スピーカーとは異なる一意の AS 番号があります。
  - vPC リーフ スイッチ ノードのペアは、同じ AS 番号を持ちます。
  - ファブリックを表すためにグローバルに一意の AS 番号が必要な場合は、ボーダー リーフまたはボーダー PE スイッチで設定できます。他のすべてのノードは、プライベート AS 番号範囲を使用できます。
  - BGP 連合は活用されていません。

図 *6:* アンダーレイとしての *eBGP*



eBGP 設定例

スパイン スイッチとリーフ スイッチの設定例を次に示します。コンテキストを提供するための完全な設定が示されており、eBGP アンダーレイ専用に追加された設定が強調表示され、さらに説明されています。

ネイバーごとに 1 つの BGP セッションがあり、アンダーレイを設定します。これは、グローバル IPv4 アドレス ファミリ内で行われます。このセッションは、VTEP、ランデブー ポイント（RP）のループバックアドレス、およびオーバーレイ eBGP セッションのeBGP ピア アドレスを配布するために使用されます。

スパイン スイッチ S1 の設定：スパインスイッチ（この例では S1）では、すべてのリーフノードが eBGP ネイバーとして設定されます。

(config) #

```
router bgp 65536
   router-id 10.1.1.53
   address-family ipv4 unicast
   redistribute direct route-map DIRECT-ROUTES-MAP
```

redistribute direct コマンドは、BGP および VTEP ピアリングのループバック アドレスをアドバタイズするために使用されます。グローバルアドレス空間内の他の直接ルートをアドバタイズするために使用できます。ルートマップは、eBGP ピアリングおよび VTEP ループバック アドレスのみを含めるようにアドバタイズメントをフィルタリングできます。

```
   maximum-paths 2
   address-family l2vpn evpn
     retain route-target all
```

スパインスイッチの BGP スピーカーには VRF 設定がありません。したがって、ルートを保持し、リーフスイッチVTEPに送信するには、**retain route-target all** コマンドが必要です。**maximum-paths** コマンドは、アンダーレイの ECMP パスに使用されます。

リーフスイッチ V1 へのアンダーレイ セッション（vPC セットアップ）：前述のように、アンダーレイ セッションはスパインとリーフスイッチ ノード間の番号付きインターフェイスで設定されます。

(config) #

```
neighbor 10.0.1.2 remote-as 65551
   address-family ipv4 unicast
      disable-peer-as-check
      send-community both
```

スイッチの vPC ペアは、同じ AS 番号を持ちます。**disable-peer-as-check** コマンドは、ルートタイプ 5 ルートの場合など、同じ AS で設定されている vPC スイッチ間のルート伝播を可能にするために追加されました。vPC スイッチの AS 番号が異なる場合、このコマンドは必要ありません。

ボーダー リーフ スイッチへのアンダーレイ セッション：リーフとボーダー リーフスイッチへのアンダーレイ設定は同じで、IP アドレスと AS 値の変更はありません。

リーフ スイッチ V1 へのスパイン スイッチ S1 のオーバーレイ セッション

(config) #

```
route-map UNCHANGED permit 10
   set ip next-hop unchanged
```

（注） route-map UNCHANGED はユーザ定義ですが、キーワード **unchanged** は **set ip next-hop** コマンド内のオプションです。eBGP では、ある eBGP ネイバーから別の eBGP ネイバーにルートを送信するときに、ネクスト ホップが self に変更されます。ルートマップの UNCHANGED が追加され、オーバーレイ ルートの場合、元のリーフ スイッチがスパイン スイッチではなくネクストホップとして設定されます。これにより、VTEP はネクスト ホップであり、スパイン スイッチ ノードではありません。eBGP ピアへの BGP 更新でネクスト ホップ属性を変更しないことを指定するには、オプションの **unchanged** キーワードを使用します。

オーバーレイ セッションはループバック アドレスで設定されます。

(config) #

```
neighbor 10.0.51.1 remote-as 65551
   update-source loopback0
   ebgp-multihop 2
   address-family l2vpn evpn
     rewrite-evpn-rt-asn
       disable-peer-as-check
       send-community both
       route-map UNCHANGED out
```

これでスパイン スイッチの設定は完了です。*Route target auto* 機能設定は、参照のために以下に示します。

(config) #

```
vrf context coke
    vni 50000
    rd auto
    address-family ipv4 unicast
      route-target both auto
      route-target both auto evpn
    address-family ipv6 unicast
      route-target both auto
      route-target both auto evpn
```

rewrite-evpn-rt-asn コマンドは、*Route target auto* 機能を使用して EVPN RT ルートターゲットを設定する場合に必要です。

Route target auto は、スイッチで設定されたローカル AS 番号と VRF のレイヤ 3 VNID、つまりローカル AS:VNID から取得されます。マルチ AS トポロジでは、このガイドに示すように、各リーフ ノードは異なるローカル AS として表され、同じ VRF に対して生成されるルート ターゲットはスイッチごとに異なります。**rewrite-evpn-rt-asn** コマンドは、BGP アップデート メッセージのルート ターゲットの ASN 部分をローカル AS 番号に置き換えます。たとえば、VTEP V1 にローカル AS 65551、VTEP V2 にローカル AS 65549 があり、スパインスイッチ S1 にローカル AS 65536 がある場合、V1、V2、および S1 のルートターゲットは次のようになります。

- V1—65551:50000
- V2—65549:50000
- S1—65536:50000

このシナリオでは、V2 は RT 65549:50000 を使用してルートをアドバタイズし、スパインスイッチ S1 は RT 65536:50000 を使用してルートをアドバタイズし、最後に V1 が更新を取得すると、更新のルート ターゲットを 65551:50000 に置き換えます。これは、V1 でローカルに設定された RT と一致します。このコマンドを使用するには、ファブリック内のすべての BGP スピーカーで設定する必要があります。

Route Target auto 機能が使用されていない場合、つまり、一致する RT をすべてのスイッチで手動で設定する必要がある場合は、このコマンドは不要です。

リーフ スイッチの VTEP V1 設定：次の設定例では、VTEP V1 のインターフェイスが BGP ネイバーとして指定されています。ボーダー リーフ スイッチ ノードを含むすべてのリーフ スイッチ VTEP には、スパイン スイッチネイバー ノードに対する次の設定があります。

(config) #

```
router bgp 65551
  router-id 10.1.1.54
  address-family ipv4 unicast
     maximum-paths 2
     address-family l2vpn evpn
```

maximum-paths コマンドは、アンダーレイの ECMP パスに使用されます。

リーフ スイッチ VTEP V1 のスパイン スイッチ S1 へのアンダーレイ セッション

(config) #

```
neighbor 10.0.1.1 remote-as 65536
   address-family ipv4 unicast
     allowas-in
     send-community both
```

allowas-in コマンドは、リーフ スイッチ ノードに同じ AS がある場合に必要です。特に、シスコの検証済みトポロジでは、スイッチの vPC ペアが AS 番号を共有していました。

スパイン スイッチ S1 へのオーバーレイ セッション

(config) #

```
neighbor 10.1.1.53 remote-as 65536
   update-source loopback0
   ebgp-multihop 2
   address-family l2vpn evpn
   rewrite-evpn-rt-asn
     allowas-in
     send-community both
```

オーバーレイのピアリングがループバック アドレス上にあるため、**ebgp-multihop 2** コマンドが必要です。NX-OS は、ネイバーが 1 ホップ離れている場合でも、マルチホップと見なします。

vPCバックアップ セッション

(config) #

```
route-map SET-PEER-AS-NEXTHOP permit 10
  set ip next-hop peer-address

neighbor 192.168.0.1 remote-as 65551
    update-source Vlan3801
    address-family ipv4 unicast
      send-community both
      route-map SET-PEER-AS-NEXTHOP out
```

（注） このセッションは、vPC リーフ スイッチ ノード間のバックアップ SVI で設定されます。

上記のイメージの設定を完了するには、次を設定します。

- 他のスパイン スイッチの *BGP* ネイバーとしての *V1*。
- 他のリーフ スイッチに対してこの手順を繰り返します。

BGP 確認

BGP 設定を確認するには、次のコマンドを使用します。

```
show bgp all
show bgp ipv4 unicast neighbors
show ip route bgp
```

コマンドの詳細なリストについては、『Configuration and Command Reference』ガイドを参照してください。

VXLAN アンダーレイでのマルチキャスト ルーティング

VXLAN EVPN プログラマブル ファブリックは、BUM（ブロードキャスト、不明なユニキャスト、マルチキャスト）トラフィックを転送するためのマルチキャスト ルーティングをサポートします。

Cisco Nexus スイッチがサポートするマルチキャスト プロトコルについては、次の表を参照してください。

Cisco Nexus シリーズ スイッチの組み合わせ	マルチキャスト ルーティング オプション
Cisco Nexus 9000 シリーズ スイッチを搭載した Cisco Nexus 7000/7700シリーズ スイッチ	PIM ASM（スパース モード）

Cisco Nexus 9000 シリーズ	PIM ASM（スパース モード）または PIM BiDir （注） PIM BiDir は、Cisco Nexus 9300-EX および 9300-FX/FX2 プラットフォーム スイッチでサポートされます。 Cisco NX-OS リリース 10.4(3)F 以降、PIM Bidir は、Cisco Nexus 9300-FX3/GX/GX2/H2R/H1 スイッチ、および 9700-GX ライン カードを備えた 9500 スイッチでサポートされます。

入力レプリケーション を使用して、マルチキャストなしで BUM トラフィックを転送できます。入力レプリケーションは、現在 Cisco Nexus 9000 シリーズ スイッチで使用できます。

PIM ASM および PIM Bidir アンダーレイ IP ネットワーク

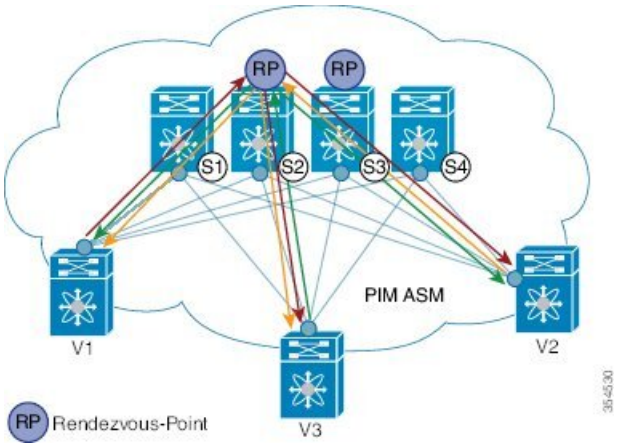
マルチキャスト トポロジの設計ポインタを次に示します。

- ランデブーポイント ロケーションとしてスパイン/集約スイッチを使用します。
- さまざまなマルチキャスト グループ（宛先グループ/DGroup）を予約して、オーバーレイを処理し、多様な VNI に最適化します。
- リーン スパインを使用したスパイン リーフ トポロジでは、
  - 複数のスパイン スイッチで複数のランデブーポイントを使用します。
  - 冗長ランデブーポイントを使用します。
  - 異なる VNI を異なるマルチキャスト グループにマッピングします。これは、ロードバランシングのために異なるランデブーポイントにマッピングされます。

重要 次の設定例は、IPアンダーレイの観点からのものであり、包括的なものではありません。PIM認証、BIM for BIM などの機能はここには示されていません。詳細については、それぞれの Cisco Nexus シリーズ スイッチ マルチキャスト コンフィギュレーション ガイドを参照してください。

PIM スパース モード（Any-Source マルチキャスト [ASM]）

図 *7: IP* マルチキャスト ルーティング プロトコルとしての *PIM ASM*



PIM ASM は、アンダーレイ マルチキャスト プロトコルとして Cisco Nexus 9000 シリーズでサポートされています。

上の図では、リーフスイッチ（VT1 設定を持つ V1、V2、および V3）がイメージの下部にあります。これらは、イメージの上部に示されている 4 つのスパイン スイッチ（S1、S2、S3、および S4）に接続されています。

2 つのマルチキャスト ランデブーポイント（S2 および S3）が設定されます。2 番目のランデブーポイントは、ロード シェアリングと冗長性のために追加されます。エニーキャスト *RP* は、*PIM ASM* トポロジ イメージに表示されます。エニーキャスト RP は、2 つのランデブーポイント間の冗長性とロード シェアリングを保証します。エニーキャスト RP を使用するには、RP として機能する複数のスパインが同じ IP アドレス（エニーキャスト RP アドレス）を共有します。一方、各 RP には、RP として機能するすべてのスパイン間の送信元に関する情報を同期するために、RP 用に設定された固有の IP アドレスがあります。

共有マルチキャスト ツリーは単方向で、パケットの転送にランデブーポイントを使用します。

PIM ASM の概要：各リーフ スイッチのマルチキャスト グループごとに 1 つの送信元ツリー。

プログラマブル ファブリック固有のポインタは次のとおりです。

- VNI にサービスを提供するすべての VTEP は、共有マルチキャスト ツリーに参加します。VTEP V1、V2、および V3 には単一のテナント（x など）から接続されたホストがあり、これらの VTEP は個別のマルチキャスト（送信元、グループ）ツリーを形成します。
- VTEP（V1 など）には、他のテナントに属するホストもあります。各テナントには、異なるマルチキャスト グループが関連付けられている場合があります。テナントがマルチキャスト グループを共有しない場合、VTEP に存在する各テナントに対してソース ツリーが作成されます。

PIM ASM の設定

（注） 使いやすくするために、各設定の最初に、タスクの設定を開始する必要がある設定モードが記載されています。

イメージのトポロジの一部について、設定タスクと対応する show コマンドの出力が表示されます。たとえば、リーフ スイッチと接続されたスパイン スイッチの設定例が示されている場合、その設定の show コマンド出力には対応する設定のみが表示されます。

リーフ スイッチ V1 の設定：リーフ スイッチで RP の到達可能性を設定します。

リーフ スイッチ V1 での PIM エニーキャスト ランデブーポイント アソシエーション

(config) #

```
feature pim
ip pim rp-address 198.51.100.220 group-list 224.1.1.1
```

198.51.100.220 は、エニーキャスト ランデブーポイントの IP アドレスです。

リーフ スイッチ V1 のループバック インターフェイス PIM 設定

(config) #

```
interface loopback 0
  ip address 209.165.201.20/32
  ip pim sparse-mode
```

リーフスイッチ V1 からスパイン スイッチ S2 へのポイントツーポイント（P2P）インターフェイス PIM 設定

(config) #

```
interface Ethernet 1/1
  no switchport
  ip address 209.165.201.14/31
  mtu 9216
  ip pim sparse-mode
.
.
```

V1 と冗長エニーキャスト ランデブーポイントとして機能するスパイン スイッチ（*S3*）間の *P2P* リンクに対して、上記の設定を繰り返します。

また、VTEP は、ランデブーポイントではないスパイン スイッチ（S1 および S4）と接続する必要があります。設定例を次に示します。

リーフ スイッチ V1 から非ランデブーポイント スパイン スイッチ（S1）へのポイントツーポイント（P2P）インターフェイス設定

(config) #

```
interface Ethernet 2/2
  no switchport
```

```
  ip address 209.165.201.10/31
  mtu 9216
  ip pim sparse-mode
```

V1 と非ランデブーポイント スパイン スイッチ間のすべての *P2P* リンクに対して上記の設定を繰り返します。

他のすべてのリーフ スイッチを設定するには、上記の手順全体を繰り返します。

スパイン スイッチのランデブー ポイントの設定

スパイン スイッチ S2 の PIM 設定

(config) #

```
feature pim
```

ループバック インターフェイス設定（RP）

(config) #

```
interface loopback 0
 ip address 10.10.100.100/32
 ip pim sparse-mode
```

ループバック インターフェイス コンフィギュレーション（エニーキャスト RP）

(config) #

```
interface loopback 1
  ip address 198.51.100.220/32
  ip pim sparse-mode
```

スパイン スイッチ S2 のエニーキャスト RP 設定

スパイン スイッチをランデブーポイントとして設定し、スイッチ S2 と S3 のループバック IP アドレスに関連付けて冗長性を確保します。

(config) #

```
feature pim
ip pim rp-address 198.51.100.220 group-list 224.1.1.1
ip pim anycast-rp 198.51.100.220 10.10.100.100
ip pim anycast-rp 198.51.100.220 10.10.20.100
.
.
```

（注） 上記の設定は、RP の役割を実行する他のスパイン スイッチ（S3）にも実装する必要があります。

非 RP スパイン スイッチの設定

ランデブーポイントとして指定されていないスパイン スイッチ（S1 と S4）に PIM ASM を設定する必要もあります。

以前、リーフスイッチ（VTEP）V1 は、非 RP スパイン スイッチへの P2P リンク用に設定されていました。非 RP スパイン スイッチの設定例を次に示します。

スパイン スイッチ S1 の PIM ASM グローバル設定（非 RP）

(config) #

```
feature pim
ip pim rp-address 198.51.100.220 group-list 224.1.1.1
```

ループバック インターフェイス設定（非RP）

(config) #

```
interface loopback 0
  ip address 10.10.100.103/32
  ip pim sparse-mode
```

スパイン スイッチ S1 からリーフ スイッチ V1 への接続のポイント 2 ポイント（P2P）インターフェイス設定

(config) #

```
interface Ethernet 2/2
  no switchport
  ip address 209.165.201.15/31
  mtu 9216
  ip pim sparse-mode
.
.
```

非ランデブーポイント スパイン スイッチと他のリーフ スイッチ（*VTEP*）間のすべての *P2P* リンクに対して、上記の設定を繰り返します。

PIM ASM の検証

PIM ASM の設定を確認するには、次のコマンドを使用します。

```
Leaf-Switch-V1# show ip mroute 224.1.1.1

IP Multicast Routing Table for VRF "default"

(*, 224.1.1.1/32), uptime: 02:21:20, nve ip pim
  Incoming interface: Ethernet1/1, RPF nbr: 10.10.100.100
  Outgoing interface list: (count: 1)
    nve1, uptime: 02:21:20, nve

(10.1.1.54/32, 224.1.1.1/32), uptime: 00:08:33, ip mrib pim
  Incoming interface: Ethernet1/2, RPF nbr: 209.165.201.12
  Outgoing interface list: (count: 1)
    nve1, uptime: 00:08:33, mrib

(10.1.1.74/32, 224.1.1.1/32), uptime: 02:21:20, nve mrib ip pim
  Incoming interface: loopback0, RPF nbr: 10.1.1.74
```

```
  Outgoing interface list: (count: 1)
    Ethernet1/6, uptime: 00:29:19, pim

Leaf-Switch-V1# show ip pim rp

PIM RP Status Information for VRF "default"
BSR disabled
Auto-RP disabled
BSR RP Candidate policy: None
BSR RP policy: None
Auto-RP Announce policy: None
Auto-RP Discovery policy: None

RP: 198.51.100.220, (0), uptime: 03:17:43, expires: never,
  priority: 0, RP-source: (local), group ranges:
      224.0.0.0/9

Leaf-Switch-V1# show ip pim interface

PIM Interface Status for VRF "default"
Ethernet1/1, Interface status: protocol-up/link-up/admin-up
  IP address: 209.165.201.14, IP subnet: 209.165.201.14/31
  PIM DR: 209.165.201.12, DR's priority: 1
  PIM neighbor count: 1
  PIM hello interval: 30 secs, next hello sent in: 00:00:11
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x33d53dc1
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: no
  PIM VPC SVI: no
  PIM Auto Enabled: no
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 423/425 (early: 0), JPs: 37/32, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 4/6, DF-Winners: 0/197, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 0
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0
loopback0, Interface status: protocol-up/link-up/admin-up
  IP address: 209.165.201.20, IP subnet: 209.165.201.20/32
  PIM DR: 209.165.201.20, DR's priority: 1
  PIM neighbor count: 0
  PIM hello interval: 30 secs, next hello sent in: 00:00:07
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
```

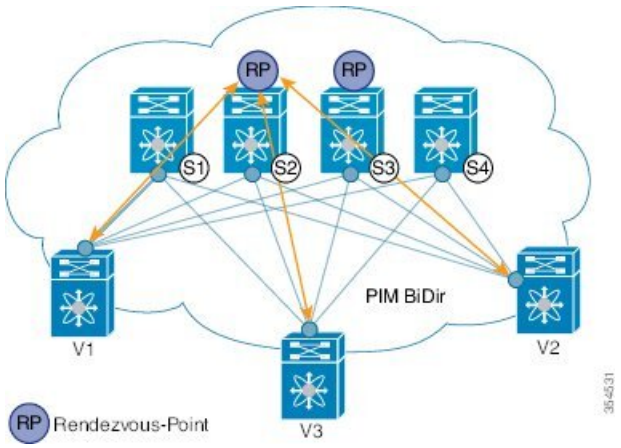
```
  PIM GenID sent in Hellos: 0x1be2bd41
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: no
  PIM VPC SVI: no
  PIM Auto Enabled: no
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 419/0 (early: 0), JPs: 2/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 3/0, DF-Winners: 0/0, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
     Packets from non-neighbors: 0
          Packets received on passiveinterface: 0
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0
```

```
Leaf-Switch-V1# show ip pim neighbor

PIM Neighbor Status for VRF "default"

Neighbor        Interface      Uptime    Expires    DR          Bidir-        BFD
                                                    Priority    Capable       State
10.10.100.100   Ethernet1/1    1w1d      00:01:33   1           yes           n/a
```

コマンドの詳細なリストについては、『Configuration and Command Reference』ガイドを参照してください。

PIM 双方向（BiDir）

図 *8 : IP* マルチキャスト ルーティング プロトコルとしての *PIM BiDir*



VXLAN BiDir アンダーレイは、Cisco Nexus 9300-EX および 9300-FX/FX2/FX3 プラットフォーム スイッチでサポートされます。

上の図では、リーフ スイッチ（V1、V2、V3）が画像の下部にあります。これらは、画像の上部に示されている 4 つのスパイン スイッチ（S1、S2、S3、および S4）に接続されています。ファントム RP メカニズムを使用する 2 つの PIM ランデブーポイントは、ロード シェアリングと冗長性のために使用されます。

（注） ロード シェアリングは、それぞれ異なる VNI の異なるマルチキャスト グループを介してのみ行われます。

双方向 PIM では、RP をルートとする 1 つの双方向共有ツリーがマルチキャスト グループごとに構築されます。送信元固有の状態はファブリック内で維持されないため、よりスケーラブルなソリューションが提供されます。

プログラマブル ファブリック固有のポインタは次のとおりです。

- 3つの VTEP は同じ VNI とマルチキャスト グループ マッピングを共有して、単一のマルチキャスト グループ ツリーを形成します。

PIM BiDir の概要：マルチキャスト グループごとに *1* つの共有ツリー。

PIM BiDir の設定

次に、冗長性とロード シェアリングのためにファントム RP を使用して、2 つのスパイン スイッチ S2 と S3 を RP として機能させる設定例を示します。ここで、S2 はグループ リスト 227.2.2.0/26 のプライマリ RP、グループ リスト 227.2.2.64/26 のセカンダリ RP です。S3 は、グループ リスト 227.2.2.64/26 のプライマリ RP およびグループ リスト 227.2.2.0/26 のセカンダリ RP です。

（注） ファントム RP は、プライマリ ルータとセカンダリ ルータで異なるマスク長のループ バック ネットワークを使用して RP の冗長性が設計されている PIM BiDir 環境で使用されます。これらのループバック インターフェイスは、RPアドレスと同じサブネット内にありますが、RPアドレスとは異なるIPアドレスを持ちます。（RPアドレスとしてアドバタイズされたIPアドレスはどのルータでも定義されていないため、「ファントム」という用語が使用されます）。ループバックのサブネットは、内部ゲートウェイ プロトコル（IGP）でアドバタイズされます。RP の到達可能性を維持するには、RP へのルートが存在することを確認するだけです。

ユニキャスト ルーティングの最長一致アルゴリズムは、セカンダリ ルータよりもプライマリ ルータを選択するために使用されます。

プライマリ ルータは最長一致ルート（たとえば、RPアドレスの/30ルート）をアナウンスし、セカンダリ ルータによってアナウンスされた特定度の低いルート（同じRPアドレスの/29ルート）よりも優先されます。プライマリ ルータは RP の /30 ルートをアドバタイズし、セカンダリ ルータは/29ルートをアドバタイズします。後者は、プライマリ ルータがオフラインになった場合にのみ選択されます。ルーティング プロトコルのコンバージェンスの速度でプライマリ RP からセカンダリ RP に切り替えることができます。

使いやすくするために、各設定の最初に、タスクの設定を開始する必要がある設定モードが記載されています。

イメージのトポロジの一部について、設定タスクと対応する show コマンドの出力が表示されます。たとえば、リーフ スイッチと接続されたスパイン スイッチの設定例が示されている場合、その設定の show コマンド出力には対応する設定のみが表示されます。

リーフ スイッチ V1 の設定

リーフスイッチ V1 でのファントム ランデブーポイント アソシエーション

(config) #

```
feature pim
ip pim rp-address 10.254.254.1 group-list 227.2.2.0/26 bidir
ip pim rp-address 10.254.254.65 group-list 227.2.2.64/26 bidir
```

リーフ スイッチ V1 のループバック インターフェイス PIM 設定

(config) #

```
interface loopback 0
  ip address 10.1.1.54/32
  ip pim sparse-mode
```

リーフ スイッチ V1 の IP アンナンバード P2P インターフェイス設定

(config) #

```
interface Ethernet 1/1
  no switchport
  mtu 9192
```

```
  medium p2p
  ip unnumbered loopback 0
  ip pim sparse-mode

interface Ethernet 2/2
  no switchport
  mtu 9192
  medium p2p
  ip unnumbered loopback 0
  ip pim sparse-mode
```

ランデブーポイントの設定（RP として動作する 2 つのスパイン スイッチ S2 および S3）

スパイン スイッチ S2 でのファントム RP の使用

(config) #

```
feature pim
ip pim rp-address 10.254.254.1 group-list 227.2.2.0/26 bidir
ip pim rp-address 10.254.254.65 group-list 227.2.2.64/26 bidir
```

スパイン スイッチ S2/RP1 のループバック インターフェイス PIM 設定（RP）

(config) #

```
interface loopback 0
  ip address 10.1.1.53/32
  ip pim sparse-mode
```

スパイン スイッチ S2/RP1 からリーフ スイッチ V1 への IP アンナンバード P2P インターフェイス設定

(config) #

```
interface Ethernet 1/1
  no switchport
  mtu 9192
  medium p2p
  ip unnumbered loopback 0
  ip pim sparse-mode
```

スパイン スイッチ S2/RP1 のループバック インターフェイス PIM 設定（ファントム RP 用）

(config) #

```
interface loopback 1
  ip address 10.254.254.2/30
  ip pim sparse-mode
```

(config) #

```
interface loopback 2
  ip address 10.254.254.66/29
  ip pim sparse-mode
```

スパイン スイッチ S3 でのファントム RP の使用

(config) #

```
feature pim
ip pim rp-address 10.254.254.1 group-list 227.2.2.0/26 bidir
ip pim rp-address 10.254.254.65 group-list 227.2.2.64/26 bidir
```

スパイン スイッチ S3/RP2 のループバック インターフェイス PIM 設定（RP）

(config) #

```
interface loopback 0
  ip address 10.10.50.100/32
  ip pim sparse-mode
```

スパイン スイッチ S3/RP2 からリーフ スイッチ V1 への IP アンナンバード P2P インターフェイス設定

(config) #

```
interface Ethernet 2/2
  no switchport
  mtu 9192
  medium p2p
  ip unnumbered loopback 0
  ip pim sparse-mode
```

スパイン スイッチ S3/RP2 のループバック インターフェイス PIM 設定（ファントム RP 用）

(config) #

```
interface loopback 1
  ip address 10.254.254.66/30
  ip pim sparse-mode


interface loopback 2
  ip address 10.254.254.2/29
  ip pim sparse-mode
```

PIM BiDir Verification

PIM BiDir の設定を確認するには、次のコマンドを使用します。

```
Leaf-Switch-V1# show ip mroute

IP Multicast Routing Table for VRF "default"

(*, 227.2.2.0/26), bidir, uptime: 4d08h, pim ip
  Incoming interface: Ethernet1/1, RPF nbr: 10.1.1.53
  Outgoing interface list: (count: 1)
    Ethernet1/1, uptime: 4d08h, pim, (RPF)

(*, 227.2.2.0/32), bidir, uptime: 4d08h, nve ip pim
  Incoming interface: Ethernet1/1, RPF nbr: 10.1.1.53
```

```
  Outgoing interface list: (count: 2)
    Ethernet1/1, uptime: 4d08h, pim, (RPF)
    nve1, uptime: 4d08h, nve

(*, 227.2.2.64/26), bidir, uptime: 4d08h, pim ip
  Incoming interface: Ethernet1/5, RPF nbr: 10.10.50.100/32
  Outgoing interface list: (count: 1)
    Ethernet1/5, uptime: 4d08h, pim, (RPF)

(*, 232.0.0.0/8), uptime: 4d08h, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)


Leaf-Switch-V1# show ip pim rp

PIM RP Status Information for VRF "default"
BSR disabled
Auto-RP disabled
BSR RP Candidate policy: None
BSR RP policy: None
Auto-RP Announce policy: None
Auto-RP Discovery policy: None

RP: 10.254.254.1, (1),
 uptime: 4d08h   priority: 0,
 RP-source: (local),
 group ranges:
 227.2.2.0/26  (bidir)
RP: 10.254.254.65, (2),
 uptime: 4d08h   priority: 0,
 RP-source: (local),
 group ranges:
 227.2.2.64/26  (bidir)


Leaf-Switch-V1# show ip pim interface

PIM Interface Status for VRF "default"
loopback0, Interface status: protocol-up/link-up/admin-up
  IP address: 10.1.1.54, IP subnet: 10.1.1.54/32
  PIM DR: 10.1.1.54, DR's priority: 1
  PIM neighbor count: 0
  PIM hello interval: 30 secs, next hello sent in: 00:00:23
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x12650908
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: no
  PIM VPC SVI: no
  PIM Auto Enabled: no
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 13158/0 (early: 0), JPs: 0/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 0/0, DF-Winners: 0/0, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
```

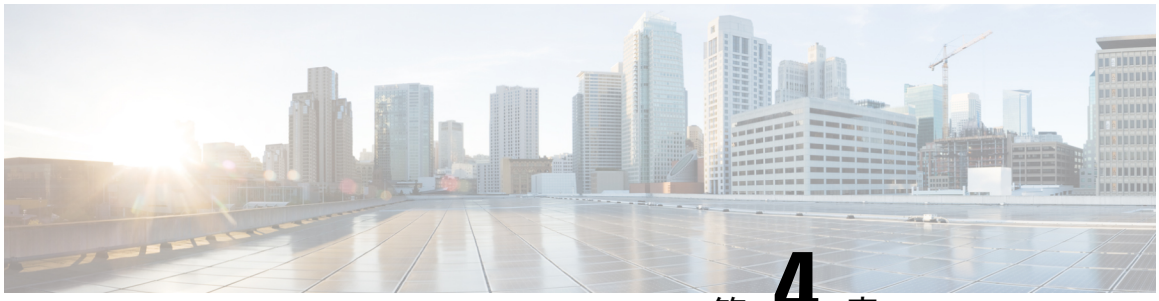
```
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 0
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0

Ethernet1/1, Interface status: protocol-up/link-up/admin-up
  IP unnumbered interface (loopback0)
  PIM DR: 10.1.1.54, DR's priority: 1
  PIM neighbor count: 1
  PIM hello interval: 30 secs, next hello sent in: 00:00:04
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x2534269b
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: no
  PIM VPC SVI: no
  PIM Auto Enabled: no
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 13152/13162 (early: 0), JPs: 2/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 9/5, DF-Winners: 6249/6254, DF-Backoffs: 0/1, DF-Passes: 0/1
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 0
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0


Leaf-Switch-V1# show ip pim neighbor

PIM Neighbor Status for VRF "default"

Neighbor     Interface      Uptime    Expires      DR          Bidir-     BFD
                                                  Priority    Capable    State
10.1.1.53    Ethernet1/1    1w1d      00:01:33     1             yes      n/a
10.10.50.100 Ethernet2/2    1w1d      00:01:33     1             yes      n/a
```

コマンドの詳細なリストについては、設定とコマンド リファレンス ガイドを参照してください。

マルチキャストを使用しないアンダーレイ導入（入力レプリケーション）

入力レプリケーションは Cisco Nexus 9000 シリーズ スイッチでサポートされます。。

NX-OS リリース 9.3(3) 以降、入力レプリケーションは Cisco Nexus 9300-GX スイッチでサポートされます。



第 4 章

VXLAN の設定

この章は、次の内容で構成されています。

VXLAN の注意事項と制約事項

VXLAN には、次の注意事項と制限事項があります。

表 *2 : Cisco Nexus 92300YC、92160YC-X、93120TX、9392PQ*、および *9348GC-FXP* スイッチの *VXLAN* トラフィックの *ACL* オプション

ACL の方向	ACL タイプ	VTEP タイプ	ポート タイプ	フローの方向	トラフィック タイプ	サポート対象
入力	PACL	入力 VTEP	L2 ポート	ネットワークにアクセス [GROUP : encap direction]	ネイティブ L2 トラフィック [GROUP : inner]	YES

ACL の方向	ACL タイプ	VTEP タイプ	ポート タイプ	フローの方向	トラフィック タイプ	サポート対象
	VACL	入力 VTEP	VLAN	ネットワークにアクセス [GROUP : encap direction]	ネイティブ L2 トラフィック [GROUP : inner]	YES
入力	RACL	入力 VTEP	テナント L3 SVI	ネットワークにアクセス [GROUP : encap direction]	ネイティブ L3 トラフィック [GROUP : inner]	YES
出力	RACL	入力 VTEP	アップリンク L3/L3-PO/SVI	ネットワークにアクセス [GROUP : encap direction]	VXLAN encap [GROUP : outer]	NO
入力	RACL	出力 VTEP	アップリンク L3/L3-PO/SVI	ネットワークにアクセス [GROUP : decap direction]	VXLAN encap [GROUP : outer]	NO
出力	PACL	出力 VTEP	L2 ポート	ネットワークにアクセス [GROUP : decap direction]	ネイティブ L2 トラフィック [GROUP : inner]	NO
	VACL	出力 VTEP	VLAN	ネットワークにアクセス [GROUP : decap direction]	ネイティブ L2 トラフィック [GROUP : inner]	NO

ACL の方向	ACL タイプ	VTEP タイプ	ポート タイプ	フローの方向	トラフィック タイプ	サポート対象
出力	RACL	出力 VTEP	テナント L3 SVI	ネットワークにアクセス [GROUP：decap direction]	Post-decap L3 トラフィック [GROUP：inner]	YES

- Cisco NX-OS リリース 10.3(1)F 以降、ノンブロッキング マルチキャスト（NBM）機能と VXLAN は、同じボックスで 2 つの異なる VRF で共存できます。

（注） アンダーレイが実行されるデフォルトの VRF で NBM が有効になっていないことを確認してください。

- スケール環境では、VRF およびレイヤ 3 VNI（L3VNI）に関連する VLAN ID を **system vlan nve-overlay id** コマンドで予約する必要があります。
- ユニキャスト、マルチキャスト、および IGMP マルチキャスト モードの NLB は、Cisco Nexus 9000 スイッチ VXLAN VTEP ではサポートされていません。回避策は、（それぞれのモードで NLB をサポートする）中間デバイスの背後に NLB クラスタを移動し、VXLAN ファブリックに外部プレフィックスとしてクラスタ IP アドレスを挿入することです。
- MultiAuth 認可変更（CoA）のサポートが追加されました。詳細については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide、Release 9.3(x)』を参照してください）。
- **lacp vpc-convergence** コマンドは、LACP をサポートするホストへの vPC ポート チャネルがある VXLAN および非 VXLAN 環境で設定できます。
- vPC あり/なしの VXLAN アンダーレイの PIM BiDir がサポートされます。

  VXLAN アンダーレイの PIM BiDir が設定されている場合、次の機能はサポートされません。

  - VXLAN のフラッディング アンド ラーニング
  - テナント ルーテッド マルチキャスト（TRM）
  - VXLAN EVPN マルチサイト
  - VXLAN EVPN マルチホーミング
  - vPC 接続 VTEP

  冗長 RP の場合は、Phantom RP を使用します。

  PIM ASM から PIM BiDir に、または PIM BiDir から PIM ASM アンダーレイに移行する場合は、次の手順例を使用することをお勧めします。

```
no ip pim rp-address 192.0.2.100 group-list 230.1.1.0/8
clear ip mroute *
clear ip mroute date-created *
clear ip pim route *
clear ip igmp groups *
clear ip igmp snooping groups * vlan all
```

すべてのテーブルがクリーンアップされるまで待ちます。

```
ip pim rp-address 192.0.2.100 group-list 230.1.1.0/8 bidir
```

- **no feature pim** コマンドを入力しても、ルートの NVE 所有権は削除されないため、ルートは維持され、トラフィックは流れ続けます。エージングはPIMによって実行されます。PIM は VXLAN encap フラグを持つエントリをエージング アウトしません。
- Fibre Channel over Ethernet（FCoE）N ポート仮想化（NPV）は、異なるファブリック アップリンクで VXLAN と共存できますが、Cisco Nexus 93180YC-EX および 93180YC-FX スイッチの同じまたは異なる前面パネル ポートにあります。

  ファイバ チャネル N ポート仮想化（NPV）は、異なるファブリック アップリンク上の VXLAN と共存できますが、Cisco Nexus 93180YC-FX スイッチの同じまたは異なる前面パネル ポート上にあります。VXLANは、イーサネット前面パネルポートにのみ存在し、FC 前面パネルポートには存在しません。
- VXLAN は Cisco Nexus 9348GC-FXP スイッチではサポートされています。
- VXLAN は Cisco Nexus 92348GC スイッチではサポートされません。
- SVI が VTEP（フラッドアンドラーニング、または EVPN）で有効になっている場合は、**hardware access-list tcam region arp-ether 256** コマンドを使用して ARP-ETHER TCAM が切り分けられていることを確認します。この要件は、Cisco Nexus 9200、9300-EX、9300-FX/FX2/FX3、および 9300-GX プラットフォーム スイッチ、および 9700-EX ライン カードを搭載した Cisco 9500 シリーズ スイッチには適用されません。
- VXLAN での PBR の **load-share** キーワードの使用方法については、『Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide、Release 9.3(x)』の「*Guidelines and Limitations for Policy-Based Routing*」セクションを参照してください。
- Cisco NX-OS リリース 9.3(3) 以降、ARP 抑制は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 9.3(5) 以降、ARP 抑制は Cisco Nexus 9364C、9300-EX、9300-FX/FX2/FXP、および 9300-GX プラットフォーム スイッチのリフレクション リレーでサポートされます。リフレクティブ リレーについては、『*Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide*』を参照してください。
- Cisco NX-OS リリース 9.3(5) 以降、Cisco Nexus 9332C、9364C、9300-EX、9300-FX/FX2/FXP、および 9300-GX プラットフォーム スイッチと -EX/FX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチの非 VXLAN L3 IP トラフィックを伝送する VXLAN アップリンクのサブインターフェイスの機能があります。この機能は、VXLAN フラッドアンドラーニング、VXLAN EVPN、VXLAN EVPN マルチサイト、および DCI でサポートされます。

- Cisco NX-OS リリース 9.3(6) 以降では、VXLAN フラッドアンドラーニング モードが Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.1(1) 以降、VXLAN フラッド アンド ラーニング モードは N9K-C9316D-GX、N9K-C93600CD-GX、および N9K-C9364C-GX TOR スイッチでサポートされます。
- -R ライン カードを備えた Cisco Nexus 9504 および 9508 スイッチの場合、VXLAN レイヤ 2 ゲートウェイは 9636C-RX ライン カードでサポートされます。Cisco Nexus 9508 スイッチで VXLAN と MPLS を同時に有効にすることはできません。
- -R ライン カードを備えた Cisco Nexus 9504 および 9508 スイッチでは、9636C-RX 以外のライン カードがある場合、レイヤ 2 ゲートウェイは有効にできません。
- -R ライン カードを搭載した Cisco Nexus 9504 および 9508 スイッチの場合、PIM/ASM はアンダーレイ ポートでサポートされます。PIM/Bidir はサポートされていません。詳細については、『*Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide*、*Release 9.3(x)*』を参照してください。
- -R ライン カードを使用する Cisco Nexus 9504 および 9508 スイッチでは、オーバーレイでの IPv6 ホスト ルーティングがサポートされます。
- -R ライン カードを搭載した Cisco Nexus 9504 および 9508 スイッチでは、ARP 抑制がサポートされています。
- Cisco NX-OS リリース 10.1(1) 以降では、ITX および ePBR over VXLAN 機能が N9K-X9716D-GX TOR および N9K-C93180YC-FX3S プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.1(1) 以降、PBR over VXLAN 機能は N9K-C9316D-GX、N9K-C93600CD-GX、および N9K-C9364C-GX TOR スイッチでサポートされます。
- PBR over VXLAN 機能のルート ポリシーの設定手順に **load-share** キーワードが追加されました。

  詳細については、『Cisco Nexus 9000 Series NX_OS Unicast Routing Configuration Guide、Release 9.x』を参照してください。
- レイヤ 2 EVPN VXLAN のコンバージェンスを向上させるために、 **lacp vpc-convergence** コマンドが追加されました。

```
interface port-channel10
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 1001-1200
  spanning-tree port type edge trunk
  spanning-tree bpdufilter enable
  lacp vpc-convergence
  vpc 10

interface Ethernet1/34  <- The port-channel member-port is configured with LACP-active
 mode (for example, no changes are done at the member-port level.)
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 1001-1200
```

```
channel-group 10 mode active
no shutdown
```

- VXLAN を使用したポート VLAN は、次の例外を除き、Cisco Nexus 9300-EX および 9500-EX ライン カードを搭載した 9500 シリーズ スイッチでサポートされます。
  - これらのスイッチでは、VXLAN を使用するポート VLAN でレイヤ 2（ルーティングなし）のみがサポートされます。
  - 内部 VLAN マッピングがサポートされていません。
- **system nve ipmc** CLI コマンドは、9700-EX ライン カードを搭載した Cisco 9200 および 9300-EX プラットフォーム スイッチには適用されません。
- NVE を、レイヤ 3 プロトコルで必要な他のループバック アドレスとは別のループバック アドレスにバインドします。VXLAN に対して専用のループバック アドレスを使用することがベスト プラクティスです。このベスト プラクティスは、vPC VXLAN 展開だけでなく、すべての VXLAN 展開にも適用できます。
- NVE インターフェイスから設定を削除するには、**default interface nve** コマンドを使用するのではなく、各設定を手動で削除することを推奨します。
- **show** コマンドは **internal** キーワード付きでサポートされていません。
- FEX ポートは、VXLAN VLAN で IGMP スヌーピングをサポートしません。
- VXLAN がサポートされるのは、Cisco Nexus 93108TC-EX と 93180YC-EX スイッチおよび Cisco Nexus 9500 シリーズ スイッチで X9732C-EX ライン カードを装備したものです。
- DHCP スヌーピング（Dynamic Host Configuration Protocol スヌーピング）は VXLAN VLAN ではサポートされません。
- RACL は VXLAN トラフィックのレイヤ 3 のアップリンクでサポートされません。出力 VACL のサポートは、ネットワークのカプセル化解除されたパケットが内部ペイロードでディレクションにアクセスするためには使用できません。

  ベスト プラクティスとして、ネットワーク ディレクションへのアクセスに対して、PACL/VACL を使用します。
- QoS バッファ ブースト機能は、VXLAN トラフィックには適用できません。
- Cisco NX-OS リリース 9.3(5) よりも前のリリースには、次の制限事項が適用されます。
  - VTEP は、VRF 参加または IEEE 802.1Q カプセル化に関係なく、サブインターフェイスを介した VXLAN カプセル化トラフィックをサポートしません。
  - VRF の参加に関係なく、サブインターフェイスが設定されている場合、VTEP は親インターフェイス上の VXLAN カプセル化トラフィックをサポートしません。
  - VXLAN VLAN と非 VXLAN VLAN のサブインターフェイスの混在はサポートされていません。

- Cisco NX-OS リリース 10.1(1) 以降、サブインターフェイスを伝送する親インターフェイスを介した VXLAN カプセル化トラフィックは、Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS Release 9.3(5) 以降では、サブインターフェイスが設定されている場合、VTEP は親インターフェイス上で VXLAN カプセル化トラフィックをサポートします。この機能は、VXLAN フラッドアンドラーニング、VXLAN EVPN、VXLAN EVPN マルチサイト、および DCI でサポートされます。次の設定例に示すように、VXLAN トラフィックはデフォルト VRF の親インターフェイス（eth1/1）で転送され、L3 IP（非 VXLAN）トラフィックはテナント VRF のサブインターフェイス（eth1/1.10）で転送されます。

```
interface ethernet 1/1
 description VXLAN carrying interface
 no switchport
 ip address 10.1.1.1/30

interface ethernet 1/1.10
 description NO VXLAN
 no switchport
 vrf member Tenant10
 encapsulation dot1q 10
  ip address 10.10.1.1/30
```

- テナント VRF（VNI を含む VRF）は、VNI がバインドされていない SVI（アンダーレイ VRF）では使用できません。
- ポイントツーマルチポイントのレイヤ 3 および SVI のアップリンクは、サポートされません。
- アップリンクとしての SVI およびサブインターフェイスはサポートされていません。
- FEX HIF（FEX ホスト インターフェイス ポート）は、VXLAN で拡張された VLAN ではサポートされています。
- 入力複製 VPC セットアップでは、vPC ピア デバイス間でレイヤ 3 接続が必要です。
- ポート VLAN マッピング機能が設定された VXLAN VLAN で、ロールバックはサポートされません。
- VXLAN UDP ポート番号は VXLAN カプセル化に使用されます。Cisco Nexus NX-OS では、UDP ポート番号は 4789 です。これは IETF 標準に準拠しており、変更できません。
- VXLAN は Cisco Nexus 9500 プラットフォーム スイッチで次のライン カードを使用してサポートされています。
  - 9500-R
  - 9564PX
  - 9564TX
  - 9536PQ
  - 9700-EX
  - 9700-FX

- Cisco Nexus 9300 シリーズ スイッチで 100G アップリンクを備えたものは、VXLAN スイッチング/ブリッジングのみをサポートします

  Cisco Nexus 9200、Cisco Nexus 9300-EX、および Cisco Nexus 9300-FX、および Cisco Nexus 9300-FX2 プラットフォーム スイッチには、この制限はありません。

（注） VXLAN ルーティングのサポートについては、40G アップリンク モジュールが必要です。

- MDP は VXLAN 設定ではサポートされません。
- 整合性チェッカは、VXLAN テーブルではサポートされません。
- ARP 抑制は、VTEP がこの VNI のファーストホップゲートウェイ（Distributed Anycast Gateway）をホストしている場合にのみ、VNI でサポートされます。この VLAN の VTEP および SVI は、分散型エニーキャスト ゲートウェイ動作用に適切に設定する必要があります（たとえば、グローバル エニーキャスト ゲートウェイ MAC アドレスと、SVI の仮想 IP アドレスを持つエニーキャスト ゲートウェイ）。
- ARP 抑制は、VXLAN ファブリックでの L2VNI ごとのファブリック全体の設定です。ファブリック内のすべての VTEP で一貫してこの機能を有効または無効にします。VTEP 間での一貫性のない ARP 抑制設定はサポートされていません。
- VXLAN ネットワーク ID（VNID）16777215 が予約済みであり、明示的に設定しないでください。
- VXLAN はインサービス ソフトウェア アップグレード（ISSU）をサポートします。ただし、VXLAN ISSU は Cisco Nexus 9300-GX プラットフォーム スイッチではサポートされません。
- VXLAN は、GRE トンネル機能または MPLS（静的またはセグメント ルーティング）機能との共存を、サポートしません。
- FEX ホスト インターフェイス ポートに接続されている VTEP はサポートされていません。
- 複数の VTEP がアンダーレイ マルチキャストに同じマルチキャスト グループ アドレスを使用しているが、VNI が異なる場合は、VTEP に少なくとも 1 つの共通の VNI が必要です。これにより、NVE ピアの検出が行われ、アンダーレイ マルチキャスト トラフィックが正しく転送されます。たとえば、リーフ L1 と L4 は VNI 10 を持ち、リーフ L2 と L3 は VNI 20 を持つことができ、両方の VNI が同じグループ アドレスを共有できます。リーフ L1 がリーフ L4 にトラフィックを送信すると、トラフィックはリーフ L2 または L3 を通過できます。NVE ピア L1 はリーフ L2 または L3 で学習されないため、トラフィックはドロップされます。したがって、グループ アドレスを共有する VTEP には、ピア ラーニングが発生し、トラフィックがドロップされないように、少なくとも 1 つの共通の VNI が必要です。この要件は、VXLAN バッドノード トポロジに適用されます。
- VXLAN は、-R ライン カードを使用した Cisco Nexus 9504 および 9508 の MVR および MPLS との共存をサポートしません。

- 復元力のあるハッシュ（ポート チャネル ロードバランシング復元力）および VXLAN 設定は、ALE アップリンク ポートを使用した VTEP と互換性がありません。

（注） 復元力のあるハッシュはデフォルトではディセーブルになっています。

- -R ライン カードを使用する Cisco Nexus 9504 および 9508 スイッチの場合、L3VNI の VLAN を vPC ピアリンク トランクの許可 VLAN リストに追加する必要があります。
- VXLAN のネイティブ VLAN はサポートされません。VXLAN のレイヤ 2 トランク上のすべてのトラフィックには、タグが設定される必要があります。この制限は、95xx ライン カードを搭載した Cisco Nexus 9300 および 9500 スイッチに適用されます。この制限は、-EX または -FX ライン カードを備えた Cisco Nexus 9200、9300-EX、9300-FX、および 9500 プラットフォーム スイッチには適用されません。
- ファブリック転送中に凍結された複製ホストを更新するには、「**fabric forwarding dup-host-recovery-timer**」コマンドのみを使用し、「**fabric forwarding dup-host-unfreeze-timer**」コマンドは非推奨であるため使用しないでください。
- L3VNI を使用する場合の VXLAN ファブリックを介した traceroute の場合、次のシナリオが想定される動作です。

  L3VNI が VRF および SVI に関連付けられている場合、関連付けられた SVI には構成されている L3 アドレスがありませんが、代わりに「ip forward」構成コマンドがあります。このインターフェイスのセットアップにより、独自の SVI アドレスで traceroute に応答することはできません。代わりに、L3VNI を含む traceroute がファブリックを介して実行されると、報告される IP アドレスは、対応するテナント VRF に属する SVI の最小の IP アドレスになります。
- エニーキャスト ゲートウェイ SVI を使用したルーティング プロトコル隣接関係はサポートされません。
- Cisco NX-OS リリース 10.3(3)F 以降、新しい L3VNI モードの MHBFD は VXLAN ではサポートされません。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN は Cisco Nexus 9332D-H2R プラットフォーム スイッチでサポートされます。

VXLAN 展開の考慮事項

- スケール環境では、VRF およびレイヤ 3 VNI（L3VNI）に関連する VLAN ID を **system vlan nve-overlay id** コマンドで予約する必要があります。

  これは、次のプラットフォームを拡張するために VXLAN リソース割り当てを最適化するために必要です。

- Cisco Nexus 9300 プラットフォーム スイッチ
- 9500 ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ

次に、VRF およびレイヤ 3 VNI に関連する VLAN ID を予約する例を示します。

```
system vlan nve-overlay id 2000

        vlan 2000
          vn-segment 50000

        interface Vlan2000
          vrf member MYVRF_50000
          ip forward
          ipv6 forward

        vrf context MYVRF_50000
          vni 50000
```

（注） **system vlan nve-overlay id** コマンドは、VRF またはレイヤ 3 VNI（L3VNI）にのみ使用してください。通常の VLAN またはレイヤ 2 VNI（L2VNI）にはこのコマンドを使用しないでください。

- VXLAN BGP EVPN を構成する場合、「システム ルーティング モード：デフォルト」が次のハードウェア プラットフォームに適用されます。
  - Cisco Nexus 9200 プラットフォーム スイッチ
  - Cisco Nexus 9300 プラットフォーム スイッチ
  - Cisco Nexus 9300-EX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX/FX2/FX3 プラットフォーム スイッチ
  - Cisco Nexus 9300-GX プラットフォーム スイッチ
  - X9500 ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ
  - X9700-EX/FX ラインカードを搭載したCisco Nexus 9500プラットフォームスイッチ
- 「System Routing Mode: template-vxlan-scale」は適用されません。
- Cisco NX-OS リリース 7.0(3)I4(x) またはNX-OSリリース 7.0(3)I5(1) と組み合わせて VXLAN BGP EVPN を使用する場合は、次のハードウェア プラットフォームでは「System Routing Mode: template-vxlan-scale」が必要です。
  - Cisco Nexus 9300-EX スイッチ
  - X9700-EX ラインカードを搭載したCisco Nexus 9500 スイッチ
- Cisco NX-OS リリース 10.3(1)F 以降、Cisco Nexus 9300-FX3/GX/GX2B ToR スイッチの ARP、ND、および MAC に対して、拡張された dual-stack-host-scale テンプレートのサポートが提供されます。

- Cisco NX-OS リリース 10.4(1)F 以降、Cisco Nexus 9332D-H2R スイッチの ARP、ND、および MAC に対して、拡張された dual-stack-host-scale テンプレートのサポートが提供されます。
- Cisco NX-OS リリース 10.4(2)F 以降、Cisco Nexus 93400LD-H1 スイッチの ARP、ND、および MAC に対して、拡張された dual-stack-host-scale テンプレートのサポートが提供されます。
- Cisco NX-OS リリース 10.4(3)F 以降、Cisco Nexus 9364C-H1 スイッチの ARP、ND、および MAC に対して、拡張された dual-stack-host-scale テンプレートのサポートが提供されます。
- ARP および ND をスケーリングするには、system routing template-dual-stack-host-scale コマンドを使用します。スケーリング制限については、*Cisco Nexus 9000* シリーズ *NX-OS* 検証済みスケーラビリティ ガイドを参照してください。
- 「システム ルーティング モード」を変更するには、スイッチをリロードする必要があります。
- **source-interface config**を使用する場合は、ループバック アドレスが必要です。ループバック アドレスは、ローカル VTEP IP を表します。
- スイッチの起動時に、 **source-interface hold-down-time** ホールドダウン時間 を使用できます。 コマンドを使用すると、オーバーレイが収束し終わるまで、NVE ループバック アドレスのアドバタイズメントを抑制することができます。*hold-down-time* の範囲は 0 ～ 2147483647 秒です。デフォルトは 300 秒です。

（注） ループバックはまだダウンしていますが、トラフィックはカプセル化されてファブリックに送信されます。

- コアで IP マルチキャストのルーティングを確立するには、IP マルチキャストの設定、PIM の設定、および RP の設定が必要です。
- VTEP to VTEP ユニキャストの到達可能性は、いずれかの IGP プロトコルを介して設定できます。
- VXLAN のフラッディングおよび学習モードでは、VXLAN VLAN のデフォルト ゲートウェイを vPC デバイスのペアにある集中型ゲートウェイとして、両者の間で FHRP（First Hop Redundancy Protocol）を実行することを推奨します。

  BGP EVPN では、すべての VTEP でエニーキャスト ゲートウェイ機能を使用することを推奨します。
- フラッディングおよび学習モードでは、集中型レイヤ 3 ゲートウェイのみがサポートされています。エニーキャスト ゲートウェイはサポートされません。推奨されるレイヤ 3 ゲートウェイの設計は、vPC 中のスイッチ ペアをレイヤ 3 の集中型ゲートウェイにして、FHRP プロトコルを SVI で動作させることです。同じサブネットで使用されている異なる IP アドレスを使う場合であっても、同じ SVI のものを複数の VTEP でスパンさせることはできません。

（注） 一元化されたゲートウェイ リーフでのSVIのフラッディングおよび学習モードの設定時は、**hardware access-list tcam region arp-ether** *size* **double-wide** を設定することが必要ですがあります（このコマンドを使用する前に既存のTCAMリージョンのサイズを小さくする必要があります）。

次に例を示します。

```
hardware access-list tcam region arp-ether 256 double-wide
```

（注） Cisco Nexus 9200シリーズスイッチでは、サイズの設定は不要です。**hardware access-list tcam region arp-ether double-wide**

- BGP-EVPN で ARP 抑制を設定する場合は、**hardware access-list tcam region arp-ether** *size* **double-wide** を使用します。 コマンドを使用して ARP をこのリージョンに対応させます（このコマンドを使用する前に既存の TCAM リージョンのサイズを小さくする必要があります）。

（注） この手順は、N9K-X9564PX、N9K-X9564TX、およびN9K-X9536PQ ライン カードを搭載した Cisco Nexus 9300 スイッチ（NFE/ALE）およびCisco Nexus 9500 スイッチに必要です。Cisco Nexus 9200 スイッチ、Cisco Nexus 9300-EX スイッチ、または N9K-X9732C-EX ライン カードを搭載した Cisco Nexus 9500 スイッチでは、この手順は不要です。

- VXLAN トンネルでは、特定のアンダーレイ ポートで複数のアンダーレイ ネクスト ホップを持つことはできません。たとえば特定の出力アンダーレイ ポートでは、1 つの宛先 MAC アドレスだけが、特定の出力ポートでの外部 MAC として利用できます。

  これは、ポート単位の制限であり、トンネル単位の制限ではありません。このことは、同じアンダーレイ ポートを介して到達可能な 2 つのトンネルにおいて、2 つの外部 MAC アドレスを利用できないことを意味します。

- VTEP デバイスの IP アドレスを変更する場合は、IP アドレスの変更前に NVE インターフェイスをシャットダウンしておきます。

- ベスト プラクティスとして、VTEP のセットをマルチサイト BGW に移行する場合、この移行が実行されているすべての VTEP で NVE インターフェイスをシャットダウンする必要があります。移行が完了し、マルチサイトに必要なすべての設定がVTEPに適用されたら、NVE インターフェイスを再起動する必要があります。

- ベスト プラクティスとして、マルチキャスト グループの RP は、スパイン レイヤ上でのみ設定する必要があります。RP のロード バランシングと冗長性のために、エニーキャスト RP を使用します。

  次に、スパインでのエニーキャスト RP 設定の例を示します。

```
ip pim rp-address 1.1.1.10 group-list 224.0.0.0/4
ip pim anycast-rp 1.1.1.10 1.1.1.1
ip pim anycast-rp 1.1.1.10 1.1.1.2
```

（注）
- 1.1.1.10 は、エニーキャスト RP セットに参加しているすべての RP で設定されたエニーキャスト RP の IP アドレスです。
- 1.1.1.1 は、ローカル RP IP です。
- 1.1.1.2 は、ピア RP IP です

- 静的入力複製および BGP EVPN 入力複製は、アンダーレイでの IP マルチキャスト ルーティングを必要としません。

VXLAN 展開に対する vPC の考慮事項

- ベスト プラクティスとして、機能 vPC が VTEP に追加または VTEP から削除される場合、変更を行う前に、vPC プライマリと vPC セカンダリの両方の NVE インターフェイスをシャットダウンする必要があります。
- NVE を、レイヤ 3 プロトコルで必要な他のループバック アドレスとは別のループバック アドレスにバインドします。VXLAN に対して専用のループバック アドレスを使用することがベスト プラクティスです。
- VPC VXLAN の場合、SVI 数のスケールアップ時は、vPC 設定の **delay restore interface-vlan** タイマーの値を大きくすることを推奨します。たとえば、1000 VNI で 1000 SVI が存在する場合は、 **delay restore interface-vlan** タイマーを 45 秒に増やすことを推奨します。
- vPC VTEP ノードから VXLAN VLAN 上の接続されたホストに対して ping が開始された場合、デフォルトで使用される送信元 IP アドレスは、SVI で設定されているエニーキャスト IP です。この ping は、応答が vPC ピア ノードにハッシュされる場合、ホストからの応答を取得できません。この問題は、一意の送信元 IP アドレスを使用せずに、VXLAN vPC ノードから接続されたホストに対して ping が開始された場合に発生する可能性があります。この状況の回避策として、VXLAN OAM を使用するか、各 vPC VTEP に一意のループバックを作成し、バックドア パスを介して一意のアドレスをルーティングします。
- NVE で使用されるループバック アドレスは、プライマリ IP アドレスとセカンダリ IP アドレスを持つように設定する必要があります。

セカンダリ IP アドレスは、VXLAN のすべてのトラフィック（マルチキャストおよびユニキャスト カプセル化トラフィックを含む）に使用されます。

- vPC ピアは同じ設定にする必要があります。
  - VLAN から vn-segment への一貫したマッピング。
  - 同じループバック インターフェイスへの一貫した NVE1 バインディング
    - 同じセカンダリ IP アドレスを使用する。
    - 異なるプライマリ IP アドレスを使用する。
  - グループへの一貫した VNI マッピング。
- マルチキャストでは、RP（ランデブー ポイント）から（S, G）join を受け取る vPC ノードが DF（指定フォワーダ）になります。DF のノードでは、マルチキャストに対してカプセル化のルートがインストールされます。

  カプセル化解除のルートは、vPC プライマリ ノードと vPC セカンダリ ノードの間でのカプセル化解除ノードの選択に基づいてインストールされます。カプセル化解除の選択で優先されるのは、RP へのコストが最小のノードです。ただし、RP へのコストが両方のノードで同じである場合は、vPC プライマリ ノードが選択されます。

  カプセル化解除の選択で優先されるノードに、カプセル化解除マルチキャスト ルートがインストールされます。他のノードには、カプセル化解除のルートはインストールされません。
- vPC デバイスで、ホストからの BUM トラフィック（ブロードキャスト、未知のユニキャスト、およびマルチキャスト トラフィック）がピア リンクに複製されます。各ネイティブ パケットからコピーが作成され、各ネイティブ パケットは、ピア vPC スイッチに接続されたオーファン ポートを提供するピア リンクを介して送信されます。

  VXLAN ネットワークでのトラフィック ループを防止するために、ピア リンクに入力されるネイティブ パケットは、アップリンクに送信できません。ただし、ピア スイッチがカプセル化ノードである場合は、コピーされたパケットがピア リンクを通過してアップリンクに送信されます。

（注） コピーされた各パケットは、特別な内部 VLAN（VLAN 4041 または VLAN 4046）に送信されます。

- ピア リンクが shut の場合、vPC セカンダリにある NVE で使用されるループバック インターフェイスは停止し、ステータスは **Admin Shut** になります。これは、アップストリーム上でループバックへのルートが取り消され、アップストリームがすべてのトラフィックを vPC プライマリへ転送できるようにするために行われます。

（注） vPC セカンダリに接続されているオーファンでは、ピア リンクが shut である間にトラフィックの損失が発生します。これは、従来の vPC セットアップのセカンダリ vPC におけるレイヤ 2 オーファンに類似しています。

- vPC ドメインがシャットダウンされるとき、シャットダウンされる vPC のある VTEP 上の NVE で使用されているループバック インターフェイスは停止し、ステータスは Admin Shut になります。これは、アップストリーム上でループバックへのルートが取り消され、アップストリームがすべてのトラフィックを他の vPC VTEP へ転送できるようにするために行われます。
- ピア リンクが no-shut の場合、NVE ループバック アドレスが再度提示されます。ルートはアドバタイズされたアップストリームとなり、トラフィックを誘導します。
- vPC の場合、ループバック インターフェイスには、プライマリ IP アドレスとセカンダリ IP アドレスの 2 つの IP アドレスがあります。

  プライマリ IP アドレスは一意で、レイヤ 3 プロトコルで使用されます。

  インターフェイス NVE は VTEP IP アドレスにセカンダリ IP アドレスを使用するため、ループバック上のセカンダリ IP アドレスは必須です。セカンダリ IP アドレスは、vPC の両方のピアで同じにする必要があります。
- vPC ピアゲートウェイ機能は、両方のピアで NVE RMAC/VMAC プログラミングを容易にするために有効にする必要があります。ピア ゲートウェイ機能のために、少なくとも 1 つのバックアップ ルーティング SVI がピア リンクで有効にされ、PIM によって設定される必要があります。これにより、VTEP がスパインへの接続を完全に失ったときに、バックアップ ルーティング パスが提供されます。この場合、リモート ピアの到達可能性は、ピア リンクを介して再ルーティングされます。バド ノード トポロジにおいて、バックアップ SVI は、個々のアンダーレイ マルチキャスト グループに対してスタティック OIF として追加する必要があります。

```
switch# sh ru int vlan 2

interface Vlan2
 description backupl_svi_over_peer-link
 no shutdown
 ip address 30.2.1.1/30
 ip router ospf 1 area 0.0.0.0
 ip pim sparse-mode
 ip igmp static-oif route-map match-mcast-groups

 route-map match-mcast-groups permit 1
  match ip multicast group 225.1.1.1/32
```

（注） バド ノード トポロジにおいて、バックアップ SVI は、個々のアンダーレイ マルチキャスト グループに対してスタティック OIF として追加する必要があります。

SVI は両方の vPC ピアで設定し、PIM を有効にする必要があります。

- NVE またはループバックが vPC 設定で shut の場合：
  - プライマリ vPC スイッチでのみ NVE またはループバックが shut の場合、グローバル VXLAN vPC 整合性チェッカはエラーになります。その後、NVE、ループバック、および vPC がセカンダリ vPC スイッチでダウンになります。
  - セカンダリ vPC スイッチでのみ NVE またはループバックが shut の場合、グローバル VXLAN vPC 整合性チェッカはエラーになります。その後、NVE、ループバック、およびセカンダリ vPC がセカンダリ vPC スイッチでダウンになります。トラフィックのフローは、プライマリ vPC スイッチを介して継続されます。
  - ベスト プラクティスとして、プライマリとセカンダリの両方の vPC スイッチで NVE とループバックの両方がアップの状態を維持する必要があります。
- マルチキャスト ロードバランシングおよびRPの冗長性のためにネットワークで設定される冗長エニーキャスト RP は、vPC VTEP トポロジでサポートされます。
- ベスト プラクティスとして、エニーキャスト vPC VTEP のセカンダリ IP アドレスの変更時には、vPC プライマリと vPC セカンダリの両方にある NVE インターフェイスが、IP の変更前に shut である必要があります。
- ARP 抑制に関係なく、VTEP（フラッド アンド ラーニング、または EVPN）で SVI が有効になっている場合は、**hardware access-list tcam region arp-ether 256 double-wide** コマンドを使用してARP-ETHER TCAMが切り分けられるようにします。この要件は、Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチ、および 9700-EX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチには適用されません。
- **internal** キーワードが付いているコ **show** マンドはサポートされていません。
- DHCP スヌーピング（Dynamic Host Configuration Protocol スヌーピング）は VXLAN VLAN ではサポートされません。
- RACL は VXLAN トラフィックのレイヤ 3 のアップリンクでサポートされません。出力 VACL のサポートは、ネットワークのカプセル化解除されたパケットが内部ペイロードでディレクションにアクセスするためには使用できません。

  ベスト プラクティスとして、ネットワーク ディレクションへのアクセスに対して、PACL/VACL を使用します。

VXLAN ACL 機能のその他のガイドラインと制限事項については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide、Release 9.3(x)』を参照してください。

- QoS 分類は、レイヤ 3 アップリンク インターフェイス上でディレクションにアクセスするための、ネットワーク内の VXLAN トラフィックではサポートされません。

  VXLAN QoS機能のその他のガイドラインと制限事項については、『Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide、Release 9.3(x)』を参照してください。

- QoS バッファ ブースト機能は、VXLAN トラフィックには適用できません。
- Cisco NX-OS Release 9.3(5) 以降では、サブインターフェイスが設定されている場合、VTEP は親インターフェイス上で VXLAN カプセル化トラフィックをサポートします。
- VTEP は、サブインターフェイス上の VXLAN カプセル化トラフィックをサポートしません。これは、VRF 参加または IEEE802.1Q カプセル化に関係ありません。
- VXLAN VLAN と非 VXLAN VLAN のサブインターフェイスの混在はサポートされていません。
- ポイントツーマルチポイントのレイヤ 3 および SVI のアップリンクは、サポートされません。
- **ip forward** コマンドを使用すると、VXLAN のカプセル化解除されたパケットでルータ IP 宛てのものを、VTEP が SUP/CPU に転送できるようになります。
- SVI として設定する前に、バックアップ VLAN は、**system nve infra-vlans** コマンドでインフラ VLAN として Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3 および 9300-GX プラットフォームで設定する必要があります。
- VXLAN は Cisco Nexus 9500 プラットフォーム スイッチで次のライン カードを使用してサポートされています。
  - 9564PX
  - 9564TX
  - 9536PQ
  - 9732C-EX
- Cisco Nexus 9500 プラットフォーム スイッチを VTEP として使用する場合、100G ライン カードは Cisco Nexus 9500 プラットフォーム スイッチではサポートされません。この制限は、9700-EX または -FX ライン カードを搭載した Cisco Nexus 9500 スイッチには適用されません。
- Cisco Nexus 9300 プラットフォーム スイッチで 100G アップリンクを備えたものは、VXLAN スイッチング/ブリッジングのみをサポートしますCisco Nexus 9200 および Cisco Nexus 9300-EX/ FX/ FX2 プラットフォーム スイッチには、この制限はありません。

（注） VXLAN ルーティングのサポートについては、40G アップリンク モジュールが必要です。

- VXLAN UDP ポート番号は VXLAN カプセル化に使用されます。Cisco Nexus NX-OS では、UDP ポート番号は 4789 です。これは IETF 標準に準拠しており、変更できません。
- Application Spine Engine（ASE2）を搭載した Cisco Nexus 9200 プラットフォーム スイッチの場合。レイヤ 3 VXLAN（SVI）スループットの問題が存在します。サイズ 99 ～ 122 のパケットではデータ損失が生じます
- VXLAN ネットワーク ID（VNID）16777215 が予約済みであり、明示的に設定しないでください。
- VRRP はインサービス ソフトウェア アップグレード（ISSU）をサポートします。
- VXLAN ISSU は、 Cisco Nexus 9300-GX プラットフォーム スイッチ。
- VXLAN は、GRE トンネル機能または MPLS（静的またはセグメント ルーティング）機能との共存を、サポートしません。
- FEX ホスト インターフェイス ポートに接続されている VTEP はサポートされていません。
- 復元力のあるハッシュ（ポート チャネル ロードバランシング復元力）および VXLAN 設定は、ALE アップリンク ポートを使用した VTEP と互換性がありません。

（注） 復元力のあるハッシュはデフォルトではディセーブルになっています。

- ARP 抑制が vPC 設定で有効または無効になっている場合、グローバル VXLAN vPC 整合性チェッカが失敗し、ARP 抑制が片側だけで無効または有効になっていると、VLAN が一時停止するため、ダウン タイムが必要です。

（注） VXLAN BGP EVPN のスケーラビリティについては、『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*、*Release 9.3(x)*』を参照してください。

VXLAN 展開に対するネットワークの考慮事項

- 転送ネットワークの MTU サイズ

  MAC-to-UDP のカプセル化に起因して、VXLAN は元のフレームに 50 バイトのオーバーヘッドを導入しています。このため、転送ネットワークの最大転送単位（MTU）は 50 バイト増やす必要があります。オーバーレイで 1500 バイトの MTU を使用する場合、転送

ネットワークは、最低でも 1550 バイトのパケットに対応できるように設定する必要があります。オーバーレイ アプリケーションで 1500 バイトを超えるフレーム サイズを頻繁に使用する場合は、転送ネットワークでジャンボ フレームのサポートが必要になります。

- 転送ネットワークの ECMP および LACP ハッシュ アルゴリズム

  前のセクションで説明したように、Cisco Nexus 9000 シリーズ スイッチは、転送ネットワークの ECMP および LACP ハッシュに対する送信元 UDP ポートのエントロピー レベルを導入しています。この実装を強化する方法として、転送ネットワークは ECMP または LACP のハッシュ アルゴリズムを使用します。これらのアルゴリズムはハッシュの入力として UDP 送信元ポートを使用し、これにより VXLAN のカプセル化されたトラフィックに対して最適なロード シェアリングを実現します。

- マルチキャスト グループの拡張

  Cisco Nexus 9000 シリーズ スイッチの VXLAN の実装では、ブロードキャスト、未知のユニキャスト、およびマルチキャスト トラフィックの転送に対してマルチキャスト トンネルを使用します。マルチキャスト転送を提供するには、1 つの VXLAN セグメントを 1 つの IP マルチキャスト グループにマッピングする方法が理想的です。ただし、複数の VXLAN セグメントは、コア ネットワーク内で 1 つの IP マルチキャスト グループを共有することが可能です。VXLAN は、ヘッダーの 24 ビット VNID フィールドを使用して最大 1600 万個の論理レイヤ 2 セグメントをサポートできます。VXLAN セグメントと IP マルチキャスト グループ間の 1 対 1 マッピングにより、VXLAN のセグメント数の増加に起因して、必要なマルチキャスト アドレス空間とコア ネットワーク デバイスのフォワーディング ステートの量がパラレルに増加します。ある時点で、転送ネットワークにおけるマルチキャスト スケーラビリティが問題になることがあります。この場合には、複数の VXLAN セグメントを 1 つのマルチキャスト グループにマッピングすると、コア デバイス上のマルチキャスト コントロール プレーンのリソースが節約され、目的の VXLAN のスケーラビリティを実現できるようになります。ただしこのマッピングは、次善のマルチキャスト転送を犠牲にして実現されます。1 つのテナントのマルチキャスト グループに転送されたパケットは、同じマルチキャスト グループを共有する他のテナントの VTEP に送信されます。このため、マルチキャスト データのプレーン リソースの使用が非効率的になります。したがってこのソリューションは、コントロール プレーンのスケーラビリティとデータ プレーンの効率性との二者択一になります。

  次善のマルチキャスト複製と転送を実現しているにも関わらず、複数テナントの VXLAN ネットワークで 1 つのマルチキャスト グループを共有することで、テナント ネットワーク間のレイヤ 2 分離に影響をもたらすことはありません。マルチキャスト グループからカプセル化されたパケットを受信すると、VTEP はパケットの VXLAN ヘッダー内の VNID をチェックし、検証します。VTEP は、不明な VNID が見つかるとパケットを廃棄します。VNID が VTEP のローカル VXLAN VNID のいずれかに一致する場合のみ、パケットを VXLAN セグメントに転送します。別のテナントのネットワークはパケットを受信しません。したがって、VXLAN セグメント間の分離は低下しません。

転送ネットワークの考慮事項

転送ネットワークの設定に関する考慮事項は次のとおりです。

- VTEP デバイス：
  - /32 IP アドレスで、ループバック インターフェイスを作成および設定します。

    （vPC VTEP では、プライマリおよびセカンダリの/32 IP アドレスを設定する必要があります）

  - 転送ネットワークで実行されるルーティング プロトコル（スタティック ルート）を通じて、ループバック インターフェイス /32 アドレスをアドバタイズします。

- 転送ネットワーク全体：

Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチ の場合は、 **system nve infra-vlans** コマンドを使用する必要があります。それ以外の場合、VXLAN トラフィック（IP/UDP 4789）はスイッチによってアクティブに処理されます。次のシナリオは、完全なリストではありませんが、 **system nve infra-vlans** の定義が必要な場合に最もよく見られます。

VNI（vn-segment）に関連付けられていないすべての VLAN は、次の場合に **system nve infra-vlans** として設定する必要があります。

VXLAN フラッドアンドラーニングおよび VXLAN EVPN の場合、非 VXLAN VLAN の存在は次のことに関連する可能性があります。

- 非 VXLAN VLAN に関連する SVI は、vPC ピアリンクを介した vPC ピア間のバックアップアンダーレイ ルーティング（バックアップ ルーティング）に使用されます。
- ダウンストリーム ルータ（外部接続、vPC 経由のダイナミック ルーティング）を接続するには、非 VXLAN VLAN に関連する SVI が必要です。
- 非 VXLAN VLAN に関連する SVI は、テナント VRF ピアリング（L3 ルート同期およびテナント VRF 内の vPC VTEP 間のトラフィック）に必要です。
- 非 VXLAN VLAN に関連する SVI は、エンドポイント（Bud-Node）へのファーストホップ ルーティングに使用されます。

VXLAN フラッド アンド ラーニングの場合、非 VXLAN VLAN の存在は次のことに関連している可能性があります。

- 非 VXLAN VLAN に関連する SVI は、スパイン（コアポート）へのアンダーレイ アップリンクに使用されます。

system nve infra-vlans としてVLAN を定義するルールは、次のような特殊なケースでは緩和できます。

- VXLAN トラフィックを転送しない非VXLAN VLANに関連する SVI（IP/UDP 4789）。

- SVI に関連付けられていない、または VXLAN トラフィックを転送しない非 VXLAN VLAN（IP/UDP 4789）。

（注） インフラ VLAN の特定の組み合わせを設定しないでください。たとえば、2 と 514、10 と 522 は 512 離れています。これは、VXLAN フラッド アンド ラーニングで説明されている「コア ポート」シナリオに限定されません。

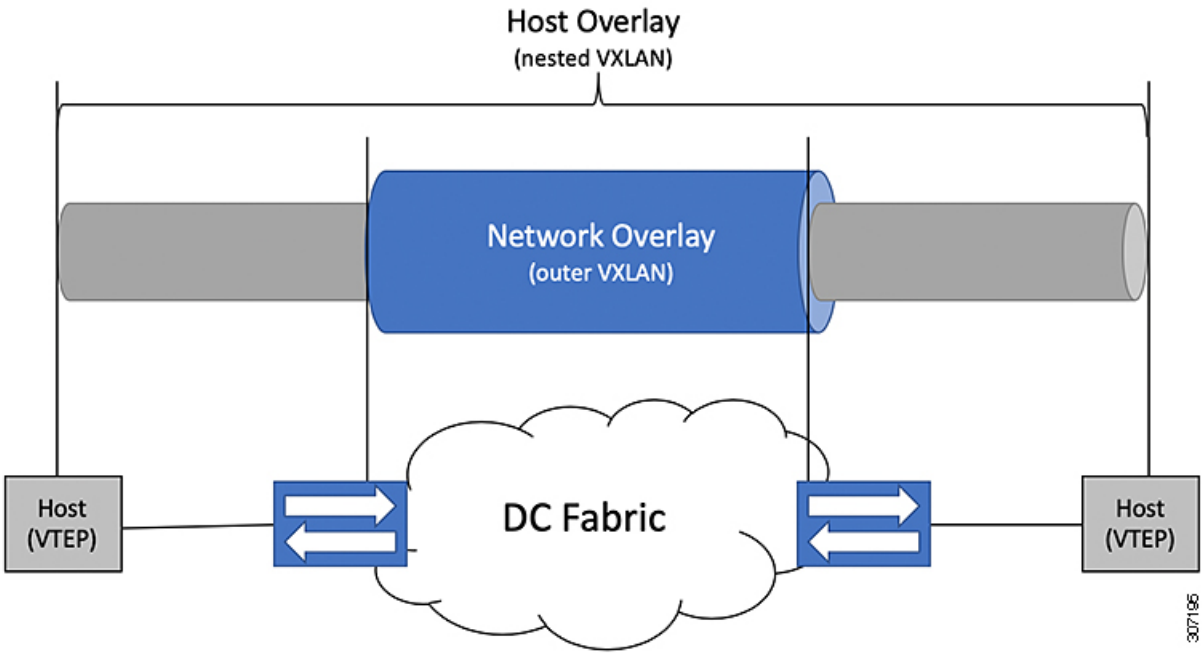
VXLAN のトンネリングに関する考慮事項

VXLAN BGP EVPN を使用する DC ファブリックは、オーバーレイのトランスポート インフラストラクチャになりつつあります。これらのオーバーレイは、多くの場合、サーバ（ホスト オーバーレイ）で生成され、既存のトランスポートインフラストラクチャ（ネットワーク オーバーレイ）の上部での統合またはトランスポートが必要です。

Cisco Nexus 9200、9300-EX、9300-FX、9300-FX2、9500-EX、9500-FX プラットフォーム スイッチ上の Cisco NX-OS リリース 7.0(3)I7(4) および Cisco NX-OS リリース 9.2(2) から、ネストされた VXLAN（Host Overlay over Network Overlay）のサポートが追加されました。また、Cisco NX-OS リリース 9.3(5) 以降の Cisco Nexus 9300-FX3 プラットフォーム スイッチでもサポートされます。

ネストされた VXLAN は、Cisco NX-OS リリース 9.3（4）以前のリリースでは、レイヤ 3 インターフェイスまたはレイヤ 3 ポート チャネル インターフェイスではサポートされません。Cisco NX-OS リリース 9.3（5）以降のレイヤ 3 インターフェイスまたはレイヤ 3 ポート チャネル インターフェイスでサポートされます。

図 *9:* ホスト オーバーレイ



ネストされた VXLAN サポートを提供するには、スイッチのハードウェアとソフトウェアが 2 つの異なる VXLAN プロファイルを区別する必要があります。

- VXLAN は、VXLAN BGP EVPN（ネストされた VXLAN）を介した転送のために、ハードウェア VTEP の背後で発信されました。
- VXLAN は、ハードウェア VTEP の背後で発生し、VXLAN BGP EVPN（BUD ノード）と統合されました。

2 つの異なる VXLAN プロファイルの検出は自動的に行われ、ネストされた VXLAN に特定の設定は必要ありません。VXLAN でカプセル化されたトラフィックが VXLAN 対応の VLAN に到着するとすぐに、トラフィックは VXLAN BGP EVPN 対応の DC ファブリックを介して転送されます。

ネストされた VXLAN では、次の接続モードがサポートされています。

- タグなしトラフィック（トランクポートまたはアクセス ポートのネイティブ VLAN）
- タグ付きトラフィック レイヤ 2 ポート（IEEE 802.1Q トランクポート上のタグ付き VLAN）
- vPC ドメインに接続されているタグなしおよびタグ付きトラフィック
- レイヤ 3 ポート チャネル インターフェイスまたはレイヤ 3 インターフェイス上のタグなしトラフィック
- レイヤ 3 ポート チャネル インターフェイスまたはレイヤ 3 インターフェイス上のタグなしトラフィック

VXLAN の設定

VXLAN のイネーブル化

手順の概要

1. **configure terminal**
2. [**no**] **feature nv overlay**
3. [**no**] **feature vn-segment-vlan-based**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル設定モードを開始します。
ステップ 2	[**no**] **feature nv overlay**	VXLAN 機能をイネーブルにします。
ステップ 3	[**no**] **feature vn-segment-vlan-based**	すべての VXLAN ブリッジ ドメインにグローバル モードを設定します。
ステップ 4	（任意） **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

VLAN から VXLAN VNI へのマッピング

手順の概要

1. **configure terminal**
2. **vlan** *vlan-id*
3. **vn-segment** *vnid*
4. **exit**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vlan** *vlan-id*	VLAN を指定します。
ステップ 3	**vn-segment** *vnid*	VXLAN VNID（仮想ネットワーク ID）を指定します
ステップ 4	**exit**	コンフィギュレーション モードを終了します。

NVE インターフェイスと関連 VNI の作成および設定

NVE インターフェイスは、VXLAN トンネルの終端となるオーバーレイ インターフェイスです。

次のように、NVE（オーバーレイ）インターフェイスを作成および設定できます。

手順の概要

1. **configure terminal**
2. **interface nve** *x*
3. **source-interface** *src-if*
4. **member vni** *vni*
5. **mcast-group** *start-address* [*end-address*]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**interface nve** *x*	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。

	コマンドまたはアクション	目的
ステップ 3	**source-interface** *src-if*	送信元インターフェイスは、有効な/32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。
ステップ 4	**member vni** *vni*	VXLAN VNI（仮想ネットワーク ID）を NVE インターフェイスに関連付けます。
ステップ 5	**mcast-group** *start-address* [*end-address*]	VNI にマルチキャスト グループを割り当てます。 （注） BUM トラフィックだけに使用します。

NVE インターフェイス ループバックの作成および構成

従来、単一のループバック インターフェイスは NVE 送信元 インターフェイスとして設定され、vPC コンプレックスの PIP と VIP の両方が構成されます。CloudSec 対応の vPC BGW に個別のループバックを設定できます。Cisco では、MLAG 展開でのコンバージェンスを向上させるために、NVE の下で送信元とエニーキャスト IP アドレスに個別のループバック インターフェイスを使用することをお勧めします。送信元インターフェイスに構成されている IP アドレスは vPC ノードの PIP であり、エニーキャスト インターフェイスに構成されている IP アドレスはその vPC コンプレックスの VIP です。NVE エニーキャスト インターフェイスも構成されている場合、NVE ソース インターフェイスで構成されたセカンダリ IP は効果がありません。

個別のループバックを使用すると、DCI 側を宛先とするデュアル接続 EVPN タイプ 2 およびタイプ 5 トラフィックのコンバージェンスが改善されます。

Cisco NX-OS リリース 10.4(1)F 以降、タイプ 2 ルートは、vMCT に固有のネクストホップとして PIP を使用してアドバタイズされます。ホールド ダウン タイマーが期限切れになる前に、PIP が NVE インターフェイスでアップ状態になります。したがって、PIP ネクストホップを持つすべてのルートは、ホールドダウン タイマーが期限切れになる前にアドバタイズします。ルートには、vMCT の孤立したタイプ 2 ルートと、redist HMM を介して学習されたローカル タイプ 5 ルート、vPC/vMCT の直接ルートまたは接続ルートが含まれます。

孤立したルートまたはローカルに接続されたルートをアドバタイズできるタイミングを示すために、ファブリック対応タイマーが vPC に追加されます。タイマーは、孤立したルートまたはローカルに接続されたルートのコンバージェンスを強化するのに役立ちます。

（注） ファブリック コンバージェンス タイマーを設定しなかった場合。デフォルトでは、タイマーは NVE ホールドダウン タイマーの 75% に設定されます。

手順の概要

1. **configure terminal**
2. **interface nve x**
3. **source-interface***loopback-interface-identifier*
4. （任意） **source-interface** [**loopback-interface-identifier**]**anycast loopback**[**loopback-interface-identifier**]
5. **show nve interface nve1 detail**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**interface nve x** 例： `switch(config-if-nve)#`	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。
ステップ 3	**source-interface***loopback-interface-identifier* 例： `switch(config-if-nve)# source-interface loopback 1`	ループバック インターフェイスを VTEP の送信元インターフェイスとして設定します。
ステップ 4	（任意） **source-interface** [**loopback-interface-identifier**]**anycast loopback**[**loopback-interface-identifier**] 例： `switch(config-if-nve)# source-interface loopback 1 anycast loopback2`	エニーキャスト ループバック インターフェイスを構成します。 （注） この構成は、以前のリリースから IPv6 アンダーレイに存在します。このリリースから、IPv4 アンダーレイの構成が追加されました。
ステップ 5	**show nve interface nve1 detail**	構成されたエニーキャスト ループバック インターフェイスに関する情報を表示します。

例

次の構成例は、エニーキャスト ループバック インターフェイスの構成を示しています。

```
switch# configure terminal
switch(config)# interface nve 1
switch(config-if-nve)# source-interface loopback 1
switch (config-if-nve)# source-interface loopback 1 anycast loopback 4
```

次の例は、スイッチに構成されたループバック インターフェイスの show コマンドを示しています。この show コマンドは、エニーキャスト ループバック インターフェイス、エニーキャスト インターフェイスに関連付けられた IP、インターフェイスの状態、ファブリック コンバージェンス タイマーなどの詳細を表示します。

（注） ファブリック コンバージェンス タイマーのデフォルト値は 135 秒です。

```
switch(config-if-nve)# show nve interface nve1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
VPC Capability: VPC-VIP-Only [notified]
Local Router MAC: e41f.7b2e.977f
Host Learning Mode: Control-Plane
Source-Interface: loopback1 (primary: 20.1.0.15)
Anycast-Interface: loopback4 (secondary: 20.1.0.145)
Source Interface State: Up
Anycast Interface State: Up
Virtual RMAC Advertisement: Yes
NVE Flags:
Interface Handle: 0x49000001
Source Interface hold-down-time: 120
Source Interface hold-up-time: 30
Remaining hold-down time: 0 seconds
Virtual Router MAC: 0200.1401.0091
Interface state: nve-intf-add-complete
Fabric convergence time: 90 seconds
Fabric convergence time left: 0 seconds
```

（注） スプリット ループバック機能がサポートされていない下位バージョンにスイッチをダウングレードすることはできません。MLAG 構成からダウングレードが開始する場合にのみ、MLAG展開でスプリットループバックをサポートするバージョンにスイッチをダウングレードできます。

単一のNVE送信元ループバックインターフェイスから別の送信元ループバックへの移行

単一の NVE 送信元ループバック インターフェイスを持つ既存の vPC 展開を、VIP および PIP の別の送信元ループバックに移動できます。この移行は、トラフィック損失への影響が少なく、既存のループバック展開をスプリットループバック展開に移行するのに役立ちます。

単一の NVE をスプリット ループバック展開に移行するには、次の手順を実行します。

1. vPC セカンダリを分離します。これは、トラフィックがプライマリのみを通過するようにするためです。

   vPC セカンダリで、次を実行します。

   1. ip pimisolate
   2. router bgp 2
   3. 分離
   4. router ospf underlay
   5. 分離
   6. sleep instance 2 20
   7. vPC domain 100
   8. shutdown

2. vPC セカンダリ上

   1. プライマリ インターフェイスのセカンダリ IP を削除します。
   2. 前のセカンダリと同じ IP アドレスを使用してエニーキャストインターフェイスを構成します。この新しい動作により、vPC CC の障害は発生せず、NVE は稼働します。

3. vPC セカンダリを接続します。ホールドダウン タイマーの期限切れを許可します。
4. vPC ロールを変更します。
5. 新しい vPC セカンダリに対してステップ 1 ～ 3 を繰り返します。これにより、構成が変更され、新しい vPC セカンダリと vPC ボックスの両方の新しい構成で更新されます。

vPC での VXLAN VTEP の設定

vPC で VXLAN VTEP を設定できます。

手順の概要

1. 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。
2. デバイスの vPC 機能を有効にします。

3. デバイスのインターフェイス VLAN 機能を有効にします。
4. デバイスの LACP 機能を有効にします。
5. デバイスの PIM 機能を有効にします。
6. デバイスの OSPF 機能を有効にします。
7. アンダーレイ マルチキャスト グループ範囲の PIM RP アドレスを定義します
8. バックアップ ルーテッド パスとして非 VXLAN 対応 VLAN を定義します。
9. インフラ VLAN として使用する VLAN を作成します。
10. vPC ピアリンク上のバックアップ ルーテッド パスに使用する SVI を作成します。
11. プライマリおよびセカンダリ IP アドレスを作成します。
12. ループバック インターフェイスにプライマリ IP アドレスを作成します。
13. vPC ドメインを作成します。
14. vPC ピア キープアライブ リンクのリモート エンドの IPv4 アドレスを設定します。
15. vPC ドメインでピアゲートウェイを有効にします。
16. vPC ドメインでピアスイッチを有効にします。
17. vPC ドメインで IP ARP 同期を有効にして、デバイスのリロード後の ARP テーブルの生成を高速化します。
18. （任意） vPC ドメインで IPv6 nd 同期を有効にして、デバイスのリロード後の nd テーブルの設定を高速化します。
19. vPC ピアリンク ポート チャネル インターフェイスを作成し、2 つのメンバー インターフェイスを追加します。
20. STP hello-time、forward-time、および max-age time を変更します。
21. （任意） SVI の遅延復元タイマーを有効にします。

手順の詳細

手順

ステップ 1 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。

```
switch# configure terminal
```

ステップ 2 デバイスの vPC 機能を有効にします。

```
switch(config)# feature vpc
```

ステップ 3 デバイスのインターフェイス VLAN 機能を有効にします。

```
switch(config)# feature interface-vlan
```

ステップ 4 デバイスの LACP 機能を有効にします。

```
switch(config)# feature lacp
```

ステップ 5 デバイスの PIM 機能を有効にします。

```
switch(config)# feature pim
```

ステップ 6 デバイスの OSPF 機能を有効にします。

```
switch(config)# feature ospf
```

ステップ 7 アンダーレイ マルチキャスト グループ範囲の PIM RP アドレスを定義します

```
switch(config)# ip pim rp-address 192.168.100.1 group-list 224.0.0/4
```

ステップ 8 バックアップ ルーテッド パスとして非 VXLAN 対応 VLAN を定義します。

```
switch(config)# system nve infra-vlans 10
```

ステップ 9 インフラ VLAN として使用する VLAN を作成します。

```
switch(config)# vlan 10
```

ステップ 10 vPC ピアリンク上のバックアップ ルーテッド パスに使用する SVI を作成します。

```
switch(config)# interface vlan 10
switch(config-if)# ip address 10.10.10.1/30
switch(config-if)# ip router ospf UNDERLAY area 0
switch(config-if)# ip pim sparse-mode
switch(config-if)# no ip redirects
switch(config-if)# mtu 9216
(Optional)switch(config-if)# ip igmp static-oif route-map match-mcast-groups
switch(config-if)# no shutdown
(Optional)switch(config)# route-map match-mcast-gropus permit 10
(Optional)switch(config-route-map)# match ip multicast group 225.1.1.1/32
```

ステップ 11 プライマリおよびセカンダリ IP アドレスを作成します。

```
switch(config)# interface loopback 0
switch(config-if)# description Control_plane_Loopback
switch(config-if)# ip address x.x.x.x/32
switch(config-if)# ip address y.y.y.y/32 secondary
switch(config-if)# ip router ospf process tag area area id
switch(config-if)# ip pim sparse-mode
switch(config-if)# no shutdown
```

ステップ 12 ループバック インターフェイスにプライマリ IP アドレスを作成します。

```
switch(config)# interface loopback 1
switch(config-if)# description Data_Plane_loopback
switch(config-if)# ip address z.z.z.z/32
switch(config-if)# ip router ospf process tag area area id
switch(config-if)# ip pim sparse-mode
switch(config-if)# no shutdown
```

ステップ 13 vPC ドメインを作成します。

```
switch(config)# vpc domain 5
```

ステップ 14 vPC ピア キープアライブ リンクのリモート エンドの IPv4 アドレスを設定します。

```
switch(config-vpc-domain)# peer-keepalive destination 172.28.230.85
```

（注） vPC ピアキープアライブ リンクを設定するまで、vPC ピア リンクは構成されません。

管理ポートと VRF がデフォルトです。

（注） 独立した VRF を設定し、vPC ピアキープアライブ リンクのための VRF 内の各 vPC ピア デバイスからのレイヤ 3 ポートを使用することを推奨します。VRF の作成および設定の詳細については、『Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide』を参照してください。

ステップ 15 vPC ドメインでピアゲートウェイを有効にします。

```
switch(config-vpc-domain)# peer-gateway
```

（注） この機能を正常に動作させるために、この vPC ドメインのすべてのインターフェイス VLAN 上で IP リダイレクトをディセーブルにします。

ステップ 16 vPC ドメインでピアスイッチを有効にします。

```
switch(config-vpc-domain)# peer-switch
```

（注） この機能を正常に動作させるために、この vPC ドメインのすべてのインターフェイス VLAN 上で IP リダイレクトをディセーブルにします。

ステップ 17 vPC ドメインで IP ARP 同期を有効にして、デバイスのリロード後の ARP テーブルの生成を高速化します。

```
switch(config-vpc-domain)# ip arp synchronize
```

ステップ 18 （任意） vPC ドメインで IPv6 nd 同期を有効にして、デバイスのリロード後の nd テーブルの設定を高速化します。

```
switch(config-vpc-domain)# ipv6 nd synchronize
```

ステップ 19 vPC ピアリンク ポート チャネル インターフェイスを作成し、2 つのメンバー インターフェイスを追加します。

```
switch(config)# interface port-channel 1
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan 1,10,100-200
switch(config-if)# mtu 9216
switch(config-if)# vpc peer-link
switch(config-if)# no shutdown
switch(config-if)# interface Ethernet 1/1 , 1/21
switch(config-if)# switchport
switch(config-if)# mtu 9216
switch(config-if)# channel-group 1 mode active
switch(config-if)# no shutdown
```

ステップ 20 STP hello-time、forward-time、および max-age time を変更します。

ベスト プラクティスとして、vPC ロールの変更が発生したときに不要な TCN 生成を回避するために、**hello-time** を 4 秒に変更することを推奨します。**hello-time** を変更した結果、**max-age** と **forward-time** を適宜変更することも推奨されます。

```
switch(config)# spanning-tree vlan 1-3967 hello-time 4
switch(config)# spanning-tree vlan 1-3967 forward-time 30
switch(config)# spanning-tree vlan 1-3967 max-age 40
```

ステップ 21 （任意） SVI の遅延復元タイマーを有効にします。

SVI または VNI スケールが大きい場合は、この値を調整することをお勧めします。たとえば、SVI カウントが 1000 の場合、interface-vlan の delay restore を 45 秒に設定することを推奨します。

```
switch(config-vpc-domain)# delay restore interface-vlan 45
```

VXLAN VTEP でのスタティック MAC の設定

VXLAN VTEP のスタティック MAC は、フラッディングおよび学習を行う Cisco Nexus 9300 シリーズ スイッチでサポートされます。この機能により、ピア VTEP でのスタティック MAC アドレス設定が可能になります。

（注） スタティック MAC は、BGP EVPN 対応 VNI のコントロール プレーンには設定できません。

手順の概要

1. **configure terminal**
2. **mac address-table static** *mac-address* **vni** *vni-id* **interface nve** *x* **peer-ip** *ip-address*
3. **exit**
4. （任意） **copy running-config startup-config**
5. （任意） **show mac address-table static interface nve** *x*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**mac address-table static** *mac-address* **vni** *vni-id* **interface nve** *x* **peer-ip** *ip-address*	リモート VTEP をポイントする MAC アドレスを指定します。
ステップ 3	**exit**	グローバル コンフィギュレーション モードを終了します。
ステップ 4	（任意） **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。
ステップ 5	（任意） **show mac address-table static interface nve** *x*	リモート VTEP をポイントするスタティック MAC アドレスを表示します。

例

次に示すのは、VXLAN VTEP に設定されたスタティック MAC アドレスの出力例です。

```
switch# show mac address-table static interface nve 1

Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link,
        (T) - True, (F) - False
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------
*  501     0047.1200.0000   static   -         F      F    nve1(33.1.1.3)
*  601     0049.1200.0000   static   -         F      F    nve1(33.1.1.4)
```

VXLAN のディセーブル化

手順の概要

1. **configure terminal**
2. **no feature vn-segment-vlan-based**
3. **no feature nv overlay**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**no feature vn-segment-vlan-based**	すべての VXLAN ブリッジ ドメインのグローバル モードをディセーブルにします。
ステップ 3	**no feature nv overlay**	VXLAN 機能をディセーブルにします。
ステップ 4	（任意） **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

BGP EVPN 入力複製の設定

次の設定では、ピアの入力複製をする BGP EVPN をイネーブルにします。

手順の概要

1. **configure terminal**
2. **interface nve** *x*
3. **source-interface** *src-if*
4. **member vni** *vni*
5. **ingress-replication protocol bgp**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**interface nve** *x*	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。
ステップ 3	**source-interface** *src-if*	送信元インターフェイスは、有効な/32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。
ステップ 4	**member vni** *vni*	VXLAN VNI（仮想ネットワーク ID）を NVE インターフェイスに関連付けます。
ステップ 5	**ingress-replication protocol bgp**	VNI の入力複製をする BGP EVPN をイネーブルにします。

静的入力複製の設定

次の設定では、ピアの静的入力複製をイネーブルにします。

手順の概要

1. **configuration terminal**
2. **interface nve** *x*
3. **member vni** [*vni-id* | *vni-range*]

4. **ingress-replication protocol static**
5. **peer-ip** *n.n.n.n*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configuration terminal**	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	**interface nve** *x*	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。
ステップ 3	**member vni** [*vni-id* | *vni-range*]	VXLAN VNI を NVE インターフェイスにマッピングします。
ステップ 4	**ingress-replication protocol static**	VNI の静的入力複製を有効にします。
ステップ 5	**peer-ip** *n.n.n.n*	ピア IP を有効にします。

VXLAN および IP-in-IP トンネリング

Cisco NX-OS リリース 9.3(6) 以降のリリースでは、VXLAN と IP-in-IP トンネリングの共存がサポートされています。

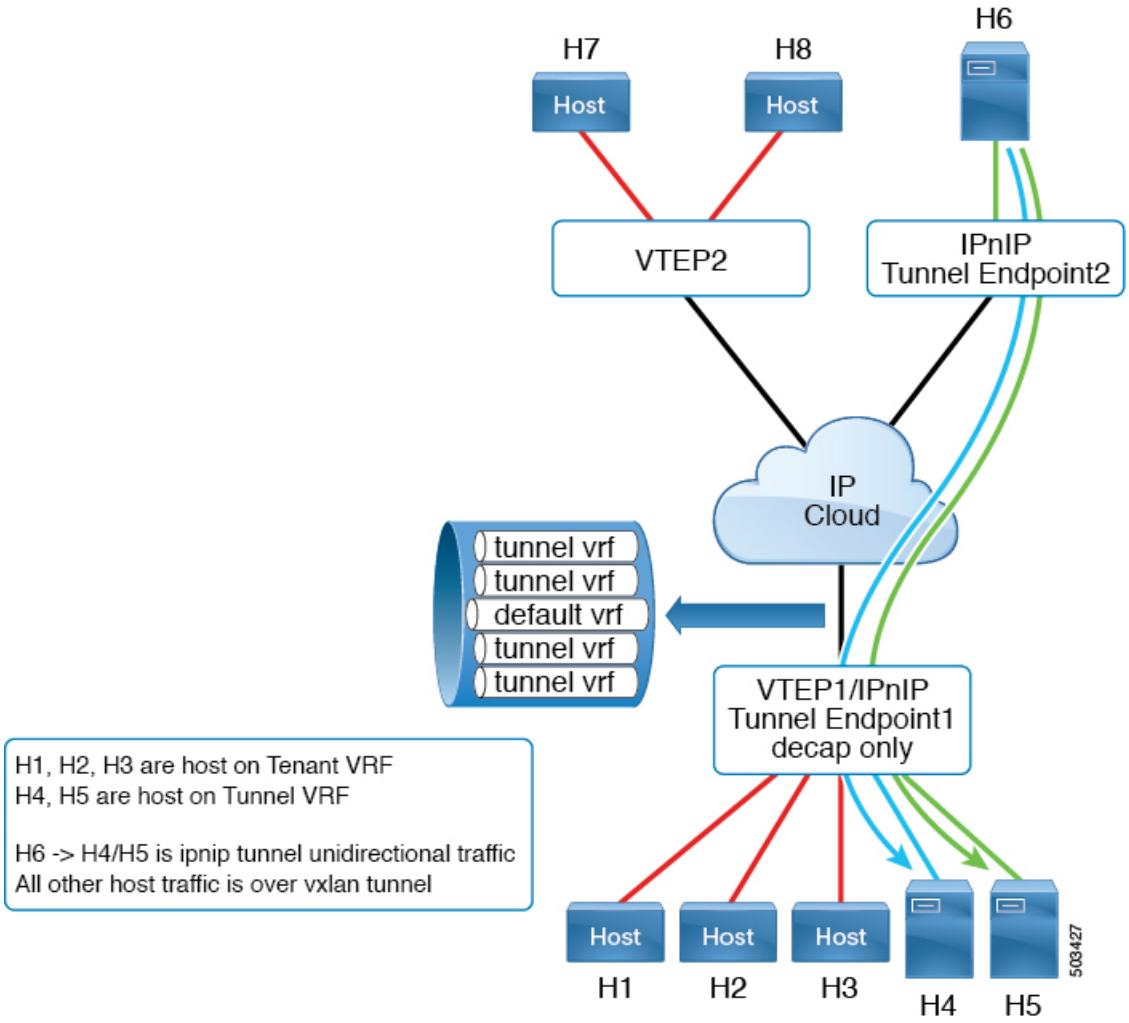
これらの機能を共存させるには、独自の VRF 内で IP-in-IP トンネルと VXLAN を分離する必要があります。VRF を分離することで、VXLAN とトンネルの両方が独立して動作します。VXLAN トンネル終端は、同じまたは異なる VRF 上で IP-in-IP トンネルとして（またはその逆に）再カプセル化されません。

インターフェイスの下にサブインターフェイスを設定して VRF を分離することで、同じアップリンクを使用して VXLAN と IP-in-IP トンネル トラフィックの両方を伝送できます。親ポートはデフォルト VRF に、サブインターフェイスはデフォルト以外の VRF に設定できます。

ポート チャネル サブインターフェイスで受信した IP-in-IP カプセル化パケットを終了するには、これらのサブインターフェイスをトンネルインターフェイスと同じ非デフォルト VRF で設定する必要があり、*** 1 *** 非デフォルト VRF のメンバーにのみなれます。。

異なる親 PC からの複数のポート チャネル サブインターフェイスは、IP-in-IP カプセル化を終了するために、同じデフォルト以外の VRF で引き続き設定できます。この制限は、1 つのポート チャネルのサブインターフェイスにのみ適用されます。この制限は、L3 ポートには適用されません。

次の例に示すように、VXLAN トラフィックはデフォルト VRF の親インターフェイス（eth1/1）で転送され、IP-in-IP（非 VXLAN）トラフィックはトンネル VRF のサブインターフェイス（eth 1/1.10）で転送されます。



Cisco Nexus 9300-FX2 プラットフォーム スイッチは、VXLAN と IP-in-IP トンネリングの共存をサポートしますが、次の制限があります。

- VXLAN はデフォルト VRF で設定する必要があります。
- 共存は、VXLAN と EVPN コントロール プレーンでサポートされます。
- IP-in-IP トンネリングは、デフォルト以外の VRF で設定する必要があり、decapsulate-any モードでのみサポートされます。

（注） デフォルト VRF でカプセル化解除トンネルが設定されているときに VXLAN を有効にしようとすると、エラー メッセージが表示されます。VXLAN と IP-in-IP トンネリングは、デフォルト以外の VRF 内の decapsulate-any トンネルに対してのみ共存でき、設定を削除できることが示されています。

- ポイントツーポイント GRE トンネルはサポートされません。ポイントツーポイント トンネルを設定しようとすると、VXLAN と IP-in-IP トンネリングが decapsulate-any トンネルに対してのみ共存できることを示すエラー メッセージが表示されます。
- 通常、トンネルを設定するには、2 つのエンドポイントを提供する必要があります。ただし、decapsulate-any は受信専用トンネルであるため、送信元 IP アドレスまたは送信元インターフェイス名のみを指定する必要があります。トンネルは、同じ VRF 内の任意の IP インターフェイスで終端します。
- トンネル統計情報は出力カウンタをサポートしていません。
- VXLAN トンネルと IP-in-IP トンネルは、同じ送信元ループバック インターフェイスを共有できません。各トンネルには、独自の送信元ループバック インターフェイスが必要です。

次の例は、設定サンプルを示しています。

```
feature vn-segment-vlan-based
feature nv overlay
feature tunnel
nv overlay evpn

interface ethernet 1/1
    description VXLAN carrying interface
    no switchport
    ip address 10.1.1.1/30

interface ethernet 1/1.10
    description IPinIP carrying interface
    no switchport
    vrf member tunnel
    encapsulation dot1q 100
    ip address 10.10.1.1/30

interface loopback 0
    description VXLAN-loopback
    ip address 125.125.125.125/32

interface loopback 100
    description Tunnel_loopback
    vrf member tunnel
    ip address 5.5.5.5/32

interface Tunnel1
    vrf member tunnel
    ip address 55.55.55.1/24
    tunnel mode ipip decapsulate-any ip
    tunnel source loopback100
    tunnel use-vrf tunnel
```

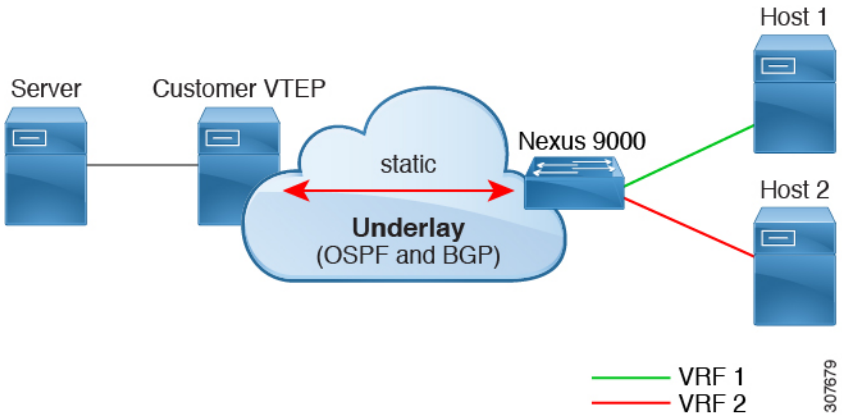
```
    no shutdown

interface nve1
    host-reachability protocol bgp
    source-interface loopback0
    global mcast-group 224.1.1.1 L2
    global mcast-group 225.3.3.3 L3
    member vni 10000
    suppress-arp
    ingress-replication protocol bgp
    member vni 55500 associate-vrf
```

VXLAN 静的トンネルの設定

VXLAN 静的トンネルについて

Cisco NX-OS リリース9.3(3) 以降では、一部の Cisco Nexus スイッチは、静的トンネルを介して顧客提供のソフトウェア VTEP に接続できます。静的トンネルはカスタマー定義であり、BGP EVPN などのコントロール プレーン プロトコルを必要とせずにホスト間の VXLAN カプセル化トラフィックをサポートします。静的トンネルは、Nexus スイッチから手動で設定することも、アンダーレイの NETCONF クライアントを介してプログラムで設定することもできます。

図 *10 : VXLAN* 静的トンネル接続ソフトウェア *VTEP*



静的トンネルは VRF ごとにサポートされます。各 VRF は専用の L3VNI を持ち、スイッチとソフトウェアVTEP（静的ピア）で適切にカプセル化およびカプセル化解除されたパケットを転送できます。通常、静的ピアは、1 つ以上の VNI を終端する 1 つ以上の VM を備えた Cisco Nexus 1000V またはベアメタル サーバです。ただし、静的ピアは、RFC 7348の「*Virtual eXtensible Local Area Network*（*VXLAN*）：仮想化レイヤ *2* ネットワークをレイヤ *3* ネットワーク上にオーバーレイするためのフレームワーク」に準拠した、お客様が開発したデバイスです。顧客が静的ピアを提供し、コントロール プレーン プロトコルが存在しないため、静的ピアが VXLAN 関連の設定を転送し、正しいホストにルーティングすることを確認する必要があります。

Cisco NX-OS Release 9.3(5) 以降では、この機能はトンネルを出入りするパケットの処理をサポートします。具体的には、Nexus スイッチがトンネルを介してホストまたは他のスイッチに

パケットを送信できるようにします。Cisco NX-OS リリース 9.3(3) および 9.3(4) では、VXLAN スタティック トンネルは、ローカル ホストからリモート ホストへの通信のみをサポートします。

VXLAN 静的トンネルの注意事項と制約事項

VXLAN 静的トンネル機能には、次の注意事項と制約事項があります。

- Cisco Nexus 9332C、9334C、9300-EX、および9300-FX/FX2/FX3, 9300-GX 、および 9300-FX3 プラットフォーム スイッチは、VXLAN 静的トンネルをサポートします。
- Cisco NX-OS リリース10.1（1）以降、VXLAN 静的トンネルは Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN スタティック トンネルは Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN スタティック トンネルは Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、VXLAN スタティック トンネルは Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN スタティック トンネルは Cisco Nexus 9364C-H1 スイッチでサポートされます。
- ソフトウェア VTEP には次のような注意事項が適用されます。
  - VNI からのトラフィックの転送方法を決定するために、必要に応じてソフトウェア VTEP を設定する必要があります。
  - ソフトウェア VTEP は RFC 7348 に準拠している必要があります。
- アンダーレイには、OSPFv2、BGP、IS-IS、または IPv4 を使用できます。
- オーバーレイは IPv4 のみです。
- 追加の VXLAN 機能（TRM、マルチサイト、OAM、クロスコネクト、VXLAN QoS など）、IGMP スヌーピング、MPLS ハンドオフ、スタティック MPLS、SR、SRv6 はサポートされていません。
- ローカル テナント VRF ループバックからソフトウェア VTEP の背後にあるホストへのオーバーレイでの ping はサポートされていません。
- 静的トンネルは ECMP 設定をサポートしません。
- 静的トンネルは、従来のフラッド アンド ラーニングまたは BGP EVPN ファブリックと同じファブリックでは設定できません。
- ローカル ホストは、VNI 対応 VLAN ではサポートされません。したがって、VNI を設定したのと同じ VLAN にホストを配置することはできません。

- ファブリック フォワーディングは、静的トンネルでサポートされます。ファブリック転送が有効になっている場合は、SVI と MAC アドレスの使用方法に影響することに注意してください。次の設定例を考えます。

```
feature fabric forwarding
fabric forwarding anycast-gateway-mac 0000.0a0a.0a0a

interface Vlan802
no shutdown
vrf member vrfvxlan5201
ip address 103.33.1.1/16
fabric forwarding mode anycast-gateway
```

ファブリック転送が有効の場合：

- **fabric forwarding mode anycast-gateway** が設定されているすべての SVI（たとえば、Vlan802）が使用されます。
- **fabric forwarding anycast-gateway-mac anycast-mac-address**（0000.0a0a.0a0a）で設定された MAC アドレスが使用されます。

VXLAN 静的トンネルの有効化

VXLAN 静的トンネルを有効にするには、次の機能を有効にします。

手順の概要

1. **config terminal**
2. **feature vn-segment**
3. **feature ofm**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		コンフィギュレーション モードを入力します。

ステップ 2	**feature vn-segment**
	例：
	`switch(config)# feature vn-segment`
	`switch(config)#`
		VLAN ベースの VXLAN を有効にします。

ステップ 3	**feature ofm** 例：	静的 VXLANトンネルを有効にします。

	コマンドまたはアクション	目的
	`switch(config)# feature ofm` `switch(config)#`	

次のタスク

静的トンネルを介した VXLAN ルーティング用の VRF オーバーレイ VLANを設定します。

静的トンネルの VRF オーバーレイの設定

VXLAN 静的トンネル用に VRF オーバーレイを設定する必要があります。

手順の概要

1. **vlan** *number*
2. **vn-segment** *number*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**vlan** *number*
	例：
	`switch(config)# vlan 2001`
	`switch(config-vlan)#`
		VLAN を指定します。

ステップ 2	**vn-segment** *number*
	例：
	`switch(config-vlan)# vn-segment 20001`
	`switch(config-vlan)#`
		VN セグメントを指定します。

次のタスク

静的トンネルを介した VXLAN ルーティングの VRF を設定します。

VXLAN ルーティングの VRF の設定

テナント VRF を設定します。

手順の概要

1. **vrf context** *vrf-name*
2. **vni** *number*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**vrf context** *vrf-name*
	例：
	`switch(config-vlan)# vrf context cust1`
	`switch(config-vrf)#`
		テナント VRF を設定します。

ステップ 2	**vni** *number*
	例：
	`switch(config-vrf)# vni 20001`
	`switch(config-vrf)#`
		テナント VRF の VNI を指定します。

次のタスク

ホストの L3 VNI を設定します。

静的トンネルの L3 VNI の設定

VTEP の L3 VNI を設定します。

始める前に

VLANインターフェイス機能を有効にする必要があります。必要に応じて **feature interface-vlan** を使用します。

手順の概要

1. **vlan** *number*
2. **interface** *vlan-number*
3. **vrf member** *vrf-name*
4. **ip forward**
5. **no shutdown**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**vlan** *number* 例：	VLAN 番号を指定します

	コマンドまたはアクション	目的
	switch(config-vrf)# **vlan 2001** switch(config-vlan)#	
ステップ 2	**interface** *vlan-number* 例： switch(config)# **interface vlan2001** switch(config-if)#	VLAN インターフェイスを指定します。
ステップ 3	**vrf member** *vrf-name* 例： switch(config-if)# **vrf member cust1** Warning: Deleted all L3 config on interface Vlan2001 switch(config-if)#	テナント VRF に VLAN インターフェイスを接続します。
ステップ 4	**ip forward** 例： switch(config-if)# **ip forward** switch(config-if)#	インターフェイスで IPv4 トラフィックを有効にします。
ステップ 5	**no shutdown** 例： switch(config-if)# **no shutdown** switch(config-if)#	インターフェイスを有効にします。

次のタスク

トンネル プロファイルを設定します。

トンネル プロファイルの設定

スタティック トンネルを設定するには、Nexus スイッチのインターフェイス、スタティック ピアの MAC アドレス、およびスタティック ピアのインターフェイスを指定するトンネル プロファイルを作成します。

始める前に

VXLAN スタティック トンネルを設定するには、アンダーレイが完全に設定され、正しく動作している必要があります。

手順の概要

1. **tunnel-profile** *profile-name*
2. **encapsulation {***VXLAN* | *VXLAN-GPE* | *SRv6***}**
3. **source-interface loopback** *virtual-interface-number*
4. **route vrf** *tenant-vrf destination-host-prefix destination-vtep-ip-address* **next-hop-vrf** *destination-vtep-vrf* **vni** *vni-number* **dest-vtep-mac** *destination-vtep-mac-address*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**tunnel-profile** *profile-name*
	例：
	`switch(config)# tunnel-profile test`
	`switch(config-tnl-profile)#`
		トンネル プロファイルを作成し、名前を指定します。

ステップ 2	**encapsulation {***VXLAN* | *VXLAN-GPE* | *SRv6***}**
	例：
	`switch(config-tnl-profile)# encapsulation vxlan`
	`switch(config-tnl-profile)#`
		トンネルプロファイルの適切なカプセル化タイプを設定します。
		（注） NX-OSリリース9.3(3) では、カプセル化タイプ **vxlan** のみがサポートされます。

ステップ 3	**source-interface loopback** *virtual-interface-number*
	例：
	`switch(config-tnl-profile)# source-interface loopback 1`
	`switch(config-tnl-profile)#`
		ループバック インターフェイスをトンネルプロファイルの送信元インターフェイスとして設定します。仮想インターフェイス番号は 0ー1023 です。

ステップ 4	**route vrf** *tenant-vrf destination-host-prefix destination-vtep-ip-address* **next-hop-vrf** *destination-vtep-vrf* **vni** *vni-number* **dest-vtep-mac** *destination-vtep-mac-address*
	例：
	`switch(tunnel-profile)# route vrf cust1 101.1.1.2/32 7.7.7.1 next-hop-vrf default vni 20001 dest-vtep-mac f80f.6f43.036c`
	`switch(tunnel-profile)#`
		宛先ソフトウェア VTEP を指定し、VNI および宛先 VTEP MAC アドレスのルート情報を入力して、トンネル ルートを作成します。
		（注） **route vrf** コマンドは、すべてのルートで *destination-vtep-ip-address* ごとに 1 つの *destination-vtep-mac-address* を受け入れます。追加のルートを設定すると、それらのルートはエラー ルートとしてキャッシュされ、それぞれに対してエラー syslog が生成されます。

VXLAN 静的トンネルの検証

トンネルの一端がダウンしても、VXLAN 静的トンネルは設定されたままになります。トンネルの一方の端がダウンしている間は、そのVTEPに到達できないため、パケットはドロップされます。ダウンしたVTEPがオンラインに戻ると、アンダーレイが接続を再学習した後、トラフィックはトンネルを介して再開できます。

show コマンドを使用して、トンネル プロファイルとトンネル ルートの状態を確認できます。

始める前に

手順の概要

1. **show tunnel-profile**
2. **show ip route** *tenant-vrf-name*
3. **show running-config ofm**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show tunnel-profile**	ソフトウェアのトンネルプロファイルに関する情報を表示します。
ステップ 2	**show ip route** *tenant-vrf-name*	ソフトウェア VTEP に接続している VRF のルート情報を表示します。たとえば、VRF のトンネルにルートが存在することを確認するために、ルート到達不能エラーが発生した場合にこのコマンドを使用できます。
ステップ 3	**show running-config ofm**	OFM 機能および静的トンネルの実行設定を表示します。ルート到達不能エラーが発生したときにこのコマンドを使用すると、宛先 VTEP のルート情報が存在するかどうかを確認できます。

次のタスク

VXLAN の検証に加えて、SPAN を使用して、スイッチを通過するパケットのポートと送信元 VLAN を確認できます。

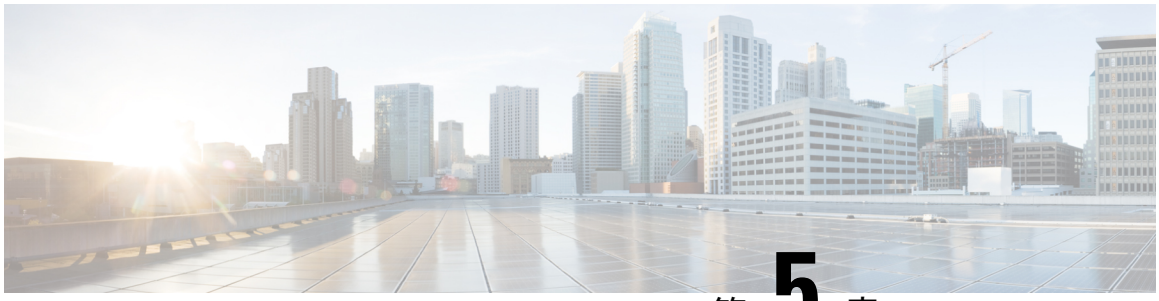
VXLAN 静的トンネルの設定例

次の設定例は、サポートされる方式による VXLAN 静的トンネル設定を示しています。

NX-OS CLI

```
vlan 2001
vlan 2001
  vn-segment 20001

interface Vlan2001
  no shutdown
  vrf member cust1
  ip forward

vrf context cust1
  vni 20001
```

```
feature ofm

tunnel-profile test
  encapsulation vxlan
  source-interface loopback1
  route vrf cust1 101.1.1.2/32 7.7.7.1 next-hop-vrf default vni 20001 dest-vtep-mac
f80f.6f43.036c
```



第 5 章

アンダーレイ（VXLANv6）でのIPv6を使用したVXLANの設定

この章は、次の内容で構成されています。

VXLANv6 の構成に関する情報

VXLAN BGP EVPN は、IPv4 アンダーレイと IPv4 VTEP で展開されます。オーバーレイ内のホストは、IPv4 または IPv6 にできます。IPv6 VTEP で アンダーレイの IPv6 を使用する VXLAN（VXLANv6） のサポートが追加されました。これには、ユニキャストルーティング プロトコルの IPv6 バージョンが必要です。

このソリューションは、VTEP が IPv6 のみでアンダーレイが IPv6 の展開を対象としています。リーフとスパイン間の BGPセッションも IPv6 です。オーバーレイ ホストは、IPv4 または IPv6 のいずれかです。

VXLANv6 機能は、アンダーレイで BGP アンナンバード ピアリングをサポートします。

アンダーレイでは、次のプロトコルがサポートされています。

- IS-IS
- OSPFv3

- eBGP

アンダーレイ（VXLANv6）での IPv6 を使用した VXLAN の注意事項と制限事項

アンダーレイ（VXLANv6）での IPv6 を使用した VXLAN の注意事項と制限事項：

- デュアル スタック（IPv4 および IPv6）は、VXLAN アンダーレイではサポートされません。IPv4 または IPv6 のいずれかである必要があります。
- VTEP の NVE 送信元インターフェイス ループバックは、IPv4（VXLANv4）または IPv6（VXLANv6）のいずれかです。
- オーバーレイのネクスト ホップ アドレス（bgp l2vpn evpn アドレス ファミリの更新）は、アンダーレイ URIB で同じアドレス ファミリに解決される必要があります。たとえば、ファブリックでの VTEP（NVE 送信元ループバック）IPv4 アドレスの使用には、IPv4 アドレスを介した BGP l2vpn evpn ピアリングのみが必要です。
- IPv6 LLA を使用するには、 **ing-sup** の TCAM リージョンをデフォルト値の 512 から 768 に再分割する必要があります。この手順では、コピー実行の開始とリロードが必要です。

次の Cisco Nexus プラットフォームは、VTEP 機能（リーフおよびボーダー）を提供するためにサポートされています。BGP ルート リフレクタは、IPv6 MP-BGP ピアリングを介して EVPN **address-family** コマンドをサポートする Cisco Nexus プラットフォームで提供できます。

- Cisco Nexus 9332C
- Cisco Nexus 9364C
- Cisco Nexus 9300-EX
- Cisco Nexus 9300-FX
- Cisco Nexus 9300-FX2
- Cisco Nexus 9300-FX3
- Cisco Nexus 9300-FXP
- Cisco Nexus 9300-GX
- Cisco Nexus 9300-GX2
- Cisco Nexus 9332D-H2R
- Cisco Nexus 93400LD-H1
- Cisco Nexus 9364C-H1

アンダーレイで IPv6 を使用する VXLAN（VXLANv6）は、次の機能をサポートします。

- オーバーレイでの Address Resolution Protocol（ARP）抑制

- アクセス コントロール リスト（ACL）と Quality of Service（QoS）
- VRF-Lite を使用したボーダー ノード
- ダイナミック ホスト コンフィギュレーション プロトコル（DHCP）
- ゲストシェルのサポート
- オーバーレイのインターネット グループ管理プロトコル（IGMP）スヌーピング
- Virtual Extensible Local Area Network（VXLAN）運用、管理、およびメンテナンス（OAM）
- ホスト ポートのストーム制御（アクセス側）
- 仮想ポート チャネル（vPC）の VIP および PIP サポート
- VXLAN ポリシーベース ルーティング（PBR）
- vPC ファブリック ピアリング
- VXLAN アクセス機能
  - プライベート VLAN（PVLAN）
  - 802.1x
  - ポート セキュリティ
  - ポート VLAN 変換
  - QinVNI
  - SelQinVNI
  - QinQ QinVNI

アンダーレイ（VXLANv6）で IPv6 を使用する VXLAN は、次の機能をサポートしていません。

- ダウンストリーム VNI
- 双方向フォワーディング検出（BFD）
- 中央集中型ルート リーク
- Cisco Data Center Network Manager（DCNM）の統合
- クロス コネクト
- イーサネット セグメント（ES）を使用した EVPN マルチホーミング
- VXLAN 対応スイッチに接続されたファブリック エクステンダ（FEX）。
- VXLAN のフラッディングおよび学習
- MACsec

- マルチプロトコル ラベル スイッチング（MPLS）および Locator/ID Separation Protocol（LISP）ハンドオフ
- マルチキャスト アンダーレイ（PIM-BiDir、Protocol Independent Multicast（PIM）Any Source Multicast（ASM）、スヌーピング）
- NetFlow
- オーバーレイ IGMP スヌーピング
- **peer vtep** コマンド
- サンプリングされたフロー（sFlow）
- 静的入力複製（IR）
- テナント ルーテッド マルチキャスト（TRM）
- 仮想ネットワーク機能（VNF）マルチパス
- VXLAN マルチサイト

Cisco NX-OS リリース 10.1(1) 以降、IPv6 アンダーレイは N9K-C9316D-GX、N9K-C93600CD-GX、および N9K-C9364C-GX TOR スイッチでサポートされています。

Cisco NX-OS リリース 10.2(3)F 以降、IPv6 アンダーレイは Cisco Nexus 9700-EX/FX/GX ラインカードでサポートされています。

Cisco NX-OS リリース 10.3(2)F 以降、IPv6 アンダーレイを使用する vPC ファブリック ピアリングは、Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 スイッチでサポートされています。

Cisco NX-OS リリース 10.4(1)F 以降、IPv6 アンダーレイを使用する vPC ファブリック ピアリングは、Cisco Nexus 9332D-H2R スイッチでサポートされています。

Cisco NX-OS リリース 10.4(2)F 以降、IPv6 アンダーレイを搭載した vPC ファブリック ピアリングは、Cisco Nexus 93400LD-H1 スイッチでサポートされます。

Cisco NX-OS リリース 10.4(3)F 以降、IPv6 アンダーレイを搭載した vPC ファブリック ピアリングは、Cisco Nexus 9364C-H1 スイッチでサポートされます。

Cisco NX-OS リリース 10.2(3)F 以降、VTEP 機能（リーフと境界）は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされています。

Cisco NX-OS リリース 10.4(1)F 以降、VTEP 機能（リーフと境界）は Cisco Nexus 9332D-H2R スイッチでサポートされています。

Cisco NX-OS リリース 10.4(2)F 以降、VTEP 機能（リーフと境界）は Cisco Nexus 93400LD-H1 スイッチでサポートされています。

Cisco NX-OS リリース 10.4(3)F 以降、VTEP 機能（リーフと境界）は Cisco Nexus 9364C-H1 スイッチでサポートされています。

Cisco NX-OS リリース 10.2(3)F 以降、VXLAN PBR は Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 プラットフォーム、N9K-C9364C、および N9K-C9332C ToR スイッチの VXLAN v6 アンダーレイでサポートされています。

Cisco NX-OS リリース 10.4(1)F 以降、VXLAN v6 アンダーレイを使用する VXLAN PBR は、Cisco Nexus 9332D-H2R スイッチでサポートされています。

Cisco NX-OS リリース 10.4(2)F 以降、VXLAN v6 アンダーレイを使用する VXLAN PBR は、Cisco Nexus 93400LD-H1 スイッチでサポートされています。

Cisco NX-OS リリース 10.4(3)F 以降、VXLAN v6 アンダーレイを使用する VXLAN PBR は、Cisco Nexus 9364C-H1 スイッチでサポートされています。

Cisco NX-OS リリース 10.2(3)F 以降、IPv6 アンダーレイは Cisco Nexus 9300-GX2 スイッチでサポートされています。

Cisco NX-OS リリース 10.4(1)F 以降、IPv6 アンダーレイは Cisco Nexus 9332D-H2R スイッチでサポートされています。

Cisco NX-OS リリース 10.4(2)F 以降、IPv6 アンダーレイは Cisco Nexus 93400LD-H1 スイッチでサポートされています。

Cisco NX-OS リリース 10.4(3)F 以降、IPv6 アンダーレイは Cisco Nexus 9364C-H1 スイッチでサポートされています。

IPv6 アンダーレイは、VXLAN EVPN の次の機能でサポートされています。

- Cisco Nexus 9300-EEX/FX/FX2/FX3/GX/GX2/H2R/H1、および Nexus 9700-EX/FX/GX ラインカードを搭載した Cisco Nexus 9500 スイッチ上のプライベート VLAN（PVLAN）。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1、および Nexus 9700-EX/FX/GX ラインカードを搭載した Cisco Nexus 9500 スイッチの 802.1x。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1、および Nexus 9700-EX/FX/GX ラインカードを搭載した Cisco Nexus 9500 スイッチのポート セキュリティ。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1、および Nexus 9700-EX/FX/GX ラインカードを搭載した Cisco Nexus 9500 スイッチでのポート VLAN 変換。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1 プラットフォーム スイッチでの QinVNI。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1 プラットフォーム スイッチでの SelQinVNI。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1 プラットフォーム スイッチでの QinQ-QinVNI。

その他の注意事項と制約事項：

- VXLAN/ファイバ チャネルの共存

vPC とアンダーレイの IPv6 を使用する VXLAN（VXLANv6）に関する情報

vPC VTEP は VIP/PIP 機能を備えた vMAC（仮想 MAC）を使用します。 vMAC は VIP で使用され、システム MAC は PIP で使用されます。

IPv4 アンダーレイでは、vMAC は IPv4 VIP アドレスから取得されます。

VMAC = 0x02 + 4 バイトの IPv4 VIP アドレス。

IPv6 アンダーレイでは、VIP は IPv6（128 ビット）であり、競合のない一意の vMAC（48 ビット）の生成には使用できません。デフォルトの方法では、IPv6 VIP から最後の 48 ビットを選択して vMAC を自動生成します。

自動生成された vMAC = 0x06 + IPv6 VIP アドレスの最後の4バイト。

異なる VIP を持ち、VIP 内の IPv6 アドレスの最後の 4 バイトが同じである 2 つの vPC コンプレックスがある場合、両方とも同じ vMAC を自動生成します。リモート VTEP の場合、2 つの異なる VIP 間で vMAC のフッピングが発生します。これは、VXLAN IPv6 をサポートする Cisco Nexus 9000 シリーズ スイッチでは問題になりません。

他のベンダーのボックスでは、これが相互運用性の問題である場合、Cisco Nexus 9000 シリーズ スイッチで vMAC を手動で設定して、自動生成された vMAC を上書きできます。アンダーレイの IPv6 を使用する VXLAN（VXLANv6） のデフォルトの動作は、VMAC の自動生成です。VMAC が手動で設定されている場合は、手動で設定された VMAC が優先されます。

```
interface nve1
    virtual-rmac <48-bit mac address>
```

VMACは、VIP/PIP と同様に管理者が管理し、ファブリック内で一意である必要があります。上記のすべての動作は、アンダーレイの IPv6 を使用する VXLAN（VXLANv6） のみと VMAC の作成およびアンダーレイでの VXLAN IPv4 のアドバタイズメントに関する変更のみです。

デフォルトの動作では、vMAC は設定された VIP から自動生成され、アドバタイズされます。相互運用性の場合を除き、前述の **virtual-rmac** コマンドを使用する必要はありません。アンダーレイの IPv6 を使用する VXLAN（VXLANv6） に対して既存の **advertise virtual-rmac** コマンドを使用する必要はありません。

vPC ピア キープアライブおよびアンダーレイの IPv6 を使用する VXLAN（VXLANv6） に関する情報

vPC の変更により、ピア キープアライブ リンクに IPv6 アドレスを使用できるようになりました。リンクは、管理インターフェイスまたはその他のインターフェイス上に配置できます。キープアライブ リンクは、両方のピアが IPv4 または IPv6 アドレスで正しく設定され、それら

のアドレスが各ピアから到達可能である場合にのみ動作可能になります。ピア キープアライブは、インバンドおよびアウトオブバンド インターフェイスで設定できます。

（注） ピア キープアライブはグローバル ユニキャスト アドレスである必要があります。

peer-keepalive のコンフィギュレーションコマンドは、IPv6 アドレスを受け入れます。

```
vpc domain 1
peer-keepalive destination 001:002::003:004 source 001:002::003:005 vrf management
```

VTEP IPアドレスの設定

手順の概要

1. **configure terminal**
2. **interface nve1**
3. **source-interface loopback** *src-if*
4. **exit**
5. **interface loopback** *loopback_number*
6. **ipv6 address** *ipv6_format*
7. **exit**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**interface nve1** 例： `switch(config)# interface nve1`	NVE インターフェイスを設定します。
ステップ 3	**source-interface loopback** *src-if* 例： `switch(config-if-nve)# source interface loopback 1`	送信元インターフェイスは、有効な/128 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この /128 IP アドレスは、転送ネットワークの中間デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルー

	コマンドまたはアクション	目的
		ティングプロトコルを介してアドレスを通知することによって、実現されます。 （注） **loopback1** の IPv6 アドレスは /128 アドレスである必要があります。 VTEP IP アドレスはリンクのローカル IPv6 アドレスに設定できません。
ステップ 4	**exit** 例： `switch(config-if-nve)# exit`	コンフィギュレーション モードを終了します。
ステップ 5	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 1`	ループバック インターフェイスを設定します。
ステップ 6	**ipv6 address** *ipv6_format* 例： `switch(config-if)# ipv6 address 2001:db8:0:0:1:0:0:1/128`	インターフェイスの IPv6 アドレスを設定します。
ステップ 7	**exit** 例： `switch(config-if)# exit`	コンフィギュレーション モードを終了します。

アンダーレイの IPv6 を使用する VXLAN（VXLANv6） の vPC の設定

アンダーレイで IPv4 を使用する VXLAN は、vPC で使用されるセカンダリ IP アドレス（VIP）の概念を活用しました。IPv6 には、IPv4 のようなセカンダリ アドレスの概念はありません。ただし、1 つのインターフェイスに複数の IPv6 グローバル アドレスを設定できます。これらのアドレスは同じ優先順位で扱われます。

VIP 設定の CLI が拡張され、アンダーレイの IPv6 を使用する VXLAN（VXLANv6） vPCがある場合に VIP を伝送するループバック インターフェイスを指定できるようになりました。IPv6 プライマリ IP アドレス（PIP）と VIP は、2 つの別々のループバック インターフェイスにあります。

IPv4 と同様に、いずれかのループバックで複数の IPv6 アドレスが指定されている場合は、それぞれに最も小さい IP が選択されます。

次の手順では、vPC セットアップで必要な VTEP IP（VIP / PIP）の設定の概要を示します。

（注） MVPN VRI ID は、vPC の TRM に構成する必要があります。この同じ VRI ID は、同じ vPC コンプレックスの一部である両方の vPC ノードで設定する必要があります。ただし、各 VRI ID はネットワーク内で一意である必要があります。これは、正しいルーティングを確保し、競合を回避するために、2 つの異なる vPC ペアが異なる VRI ID 設定を持つ必要があることを意味します。

anycast loopback コマンドは アンダーレイの IPv6 を使用する VXLAN（VXLANv6） にのみ使用されます。

手順の概要

1. **configure terminal**
2. **interface nve1**
3. **source-interface loopback** *src-if* **anycast loopback** *any-if*
4. **exit**
5. **interface loopback** *loopback_number*
6. **ipv6 address** *ipv6_format*
7. **exit**
8. **interface loopback** *loopback_number*
9. **ipv6 address** *ipv6_format*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**interface nve1** 例： `switch(config)# interface nve1`	NVE インターフェイスを設定します。
ステップ 3	**source-interface loopback** *src-if* **anycast loopback** *any-if* 例： `switch(config-if-nve)# source interface loopback 1 anycast loopback 2`	送信元インターフェイスは、有効な/128 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この /128 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。

	コマンドまたはアクション	目的
		（注） loopback1 の IPv6 アドレス（プライマリ IP アドレス（PIP）、loopback2、セカンダリ IP アドレス()VIP）は、/128 アドレスである必要があります。 VTEP IP アドレスはリンクのローカル IPv6 アドレスに設定できません。
ステップ 4	**exit** 例： `switch(config-if-nve)# exit`	コンフィギュレーション モードを終了します。
ステップ 5	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 1`	ループバック インターフェイスを設定します。
ステップ 6	**ipv6 address** *ipv6_format* 例： `switch(config-if)# ipv6 address 2001:db8:0:0:1:0:0:1/128`	インターフェイスの IPv6 アドレスを設定します。
ステップ 7	**exit** 例： `switch(config-if)# exit`	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	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 2`	ループバック インターフェイスを設定します。
ステップ 9	**ipv6 address** *ipv6_format* 例： `switch(config-inf)# ipv6 address 2001:db8:0:0:1:0:0:2/128`	インターフェイスの IPv6 アドレスを設定します。

アンダーレイの IPv6 を使用する VXLAN（VXLANv6） の設定例

アンダーレイの IPv6 を使用する VXLAN（VXLANv6） の設定例は次のとおりです。

ネクスト ホップで IPv6 アドレスを設定/照合する場合、BGP はルート タイプ 2（MAC-IP）およびルート タイプ 5（IP プレフィックス）で IPv6 ネクスト ホップ アドレスを設定/照合する必要があります。

ルートマップの下：

```
set ipv6 next-hop <vtep address>
match ipv6 next-hop  <vtep address>
```

BGP アンダーレイ

（注） BGP IPv6 ネイバーは L2VPN EVPN アドレス ファミリ セッションをサポートする必要があります。

（注） アンダーレイの IPv6 を使用する VXLAN（VXLANv6） のルータ ID は IPv4 アドレスにする必要がある。

BGP ルータ ID は 32 ビット値であり、IPv4 アドレスで表すことがよくあります。デフォルトでは、Cisco NX-OS によって、ルータのループバック インターフェイスの IPv4 アドレスにルータ ID が設定されます。アンダーレイの IPv6 を使用する VXLAN（VXLANv6） の場合、どのループバックも IPv4 アドレスを持つ必要はありません。この場合、ルータ ID のデフォルト選択は正しく行われません。ルータ ID を IPv4 アドレスに手動で設定できます。

64 ビット長の BGP RD（ルート識別子）は、4 バイトの IP アドレスの自律システム番号を使用して設定できます。アンダーレイの IPv6 を使用する VXLAN（VXLANv6） の場合、RD の設定に IP アドレスを使用するときは、VXLAN IPv4 の場合と同様に IPv4 を使用する必要があります。

```
feature bgp
nv overlay evpn

router bgp 64496
   ! IPv4 router id
   router-id  35.35.35.35
   ! Redistribute the igp/bgp routes
   address-family ipv6 unicast
     redistribute direct route-map allow

   ! For IPv6 session, directly connected peer interface
   neighbor 2001:DB8:0:1::55
     remote-as 64496
     address-family ipv6 unicast
```

OSPFv3 アンダーレイ

```
feature ospfv3

router ospfv3 201
router-id 290.0.2.1

interface ethernet 1/2
ipv6 address 2001:0DB8::1/48
```

```
ipv6 ospfv3 201 area 0.0.0.10
```

IS-IS アンダーレイ

```
router isis Enterprise
is-type level-1
net 49.0001.0000.0000.0003.00

interface ethernet 2/1
ipv6 address 2001:0DB8::1/48
isis circuit-type level-1
ipv6 router isis Enterprise
```

アンダーレイの IPv6 を使用する VXLAN（VXLANv6） の確認

アンダーレイの IPv6 を使用する VXLAN（VXLANv6） 設定のステータスを表示するには、次のコマンドを入力します。

表 *3:* アンダーレイの *IPv6* を使用する *VXLAN*（*VXLANv6*） 検証コマンド

コマンド	目的
show running-config interface nve 1	設定情報を実行するインターフェイス NVE 1 を表示します。
show nve interface 1 detail	NVE インターフェイスの詳細を表示します。
show nve peers	VTEP ピアのピアリング時間と VNI 情報を表示します。
show nve vni ingress-replication	NVE VNI 入力複製情報を表示します。
show nve peers 2018:1015::abcd:1234:3 int nv1 counters	NVE ピア カウンタ情報を表示します。
show bgp l2vpn evpn 1012.0383.9600	ルート タイプ 2 の BGP L2VPN 情報を表示します。
show bgp l2vpn evpn 303:304::1	ルート タイプ 3 の BGP L2VPN EVPN を表示します。
show bgp l2vpn evpn 5.116.204.0	ルート タイプ 5 の BGP L2VPN EVPN を表示します。
show l2route peerid	L2route peerid を表示します。
show l2route topology detail	L2route トポロジの詳細を表示します。

コマンド	目的
show l2route evpn imet all detail	L2route EVPN imet の詳細を表示します。
show l2route fl all	L2route フラッドリストの詳細を表示します。
show l2route mac all detail	L2route MAC の詳細を表示します。
show l2route mac-ip all detail	MAC アドレスとホスト IP アドレスを表示します。
show ip route 1.191.1.0 vrf vxlan-10101	VRF のルート テーブルを表示します。
show forwarding ipv4 route 1.191.1.0 detail vrf vxlan-10101	転送情報を表示します。
show ipv6 route vrf vxlan-10101	IPv6 ルーティング テーブルを表示します。
show bgp l2vpn evpn	BGP の更新されたルートを表示します。
show bgp evi *evi-id*	BGP EVI 情報を表示します。
show forwarding distribution peer-id	転送情報を表示します。
show forwarding nve l2 ingress-replication-peers	入力複製の転送情報を表示します。
show forwarding nve l3 peers	nv3 Layer 3 ピア情報を表示します。
show forwarding ecmp platform	転送 ECMP プラットフォーム情報を表示します。
show forwarding ecmp platform	転送 ECMP プラットフォーム情報を表示します。
show forwarding nve l3 ecmp	転送 NVE Layer 3 ECMP 情報を表示します。

の例 **show running-config interface nve 1**

コマンド

```
switch# show running-config interface nve 1
interface nve1
  no shutdown
  source-interface loopback1 anycast loopback2
  host-reachability protocol bgp
  member vni 10011
    ingress-replication protocol bgp
  member vni 20011 associate-vrf
```

の例 **show nve interface 1 detail**

コマンド

```
switch# show nve interface nve 1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
```

```
VPC Capability: VPC-VIP-Only [notified]
Local Router MAC: a093.51cf.78f7
Host Learning Mode: Control-Plane
Source-Interface: loopback1 (primary: 30:3:1::2)
Anycast-Interface: loopback2 (secondary: 303:304::1)
Source Interface State: Up
Anycast Interface State: Up
Virtual RMAC Advertisement: Yes
NVE Flags:
Interface Handle: 0x49000001
Source Interface hold-down-time: 745
Source Interface hold-up-time: 30
Remaining hold-down time: 0 seconds
Virtual Router MAC: 0600.0000.0001
Interface state: nve-intf-add-complete
```

show nve peers コマンドの例

```
switch# show nve peers
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- ---------------  ----- --------- -------- -----------------
nve1      1:1::1:1          Up    CP        00:44:09   5087.89d4.6bb7
```

アップ

の例 **show nve vni ingress-replication**

コマンド

```
switch# show nve vni ingress-replication
Interface VNI      Replication List  Source  Up Time
--------- -------- ----------------- ------- -------

nve1      10011    1:1::1:1           BGP-IMET   00:46:55
```

show nve peers *ipv6-address* **int nv1 counters** コマンドの例。

```
switch# show nve peers 2018:2015::abcd:1234:3 int nve 1 counters
Peer IP: 2018:1015::abcd:1234:3
TX
        0 unicast packets 0 unicast bytes
        0 multicast packets 0 multicast bytes
RX
        0 unicast packets 0 unicast bytes
        0 multicast packets 0 multicast bytes
```

ルートタイプ 2 の **show bgp l2vpn evpn** コマンドの例

```
switch# show bgp l2vpn evpn 1012.0383.9600
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 30.3.1.1:34067    (L2VNI 2001300)
BGP routing table entry for [2]:[0]:[0]:[48]:[1012.0383.9600]:[0]:[0.0.0.0]/216, version
 1051240
Paths: (1 available, best #1)
Flags: (0x000102) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    303:304::1 (metric 0) from 0:: (30.3.1.1)
```

```
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 2001300
      Extcommunity: RT:2:2001300 ENCAP:8

  Path-id 1 advertised to peers:
    2::21             2::66
BGP routing table entry for [2]:[0]:[0]:[48]:[1012.0383.9600]:[32]:[4.231.115.2]/272,
version 1053100
Paths: (1 available, best #1)
Flags: (0x000102) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    303:304::1 (metric 0) from 0:: (30.3.1.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 2001300 3003901
      Extcommunity: RT:2:2001300 RT:2:3003901 ENCAP:8 Router MAC:0600.0000.0001

  Path-id 1 advertised to peers:
    2::21             2::66
```

ルートタイプ 3 の **show bgp l2vpn evpn** コマンドの例

```
switch# show bgp l2vpn evpn 303:304::1
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 30.3.1.1:32769    (L2VNI 2000002)
BGP routing table entry for [3]:[0]:[128]:[303:304::1]/184, version 1045060
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    303:304::1 (metric 0) from 0:: (30.3.1.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: RT:2:2000002 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 2000002, Tunnel Id: 303:304::1

  Path-id 1 advertised to peers:
    2::21             2::66
```

ルート タイプ 5 の **show bgp l2vpn evpn** コマンドの例

```
switch# show bgp l2vpn evpn 5.116.204.0
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 2.0.0.52:302
BGP routing table entry for [5]:[0]:[0]:[24]:[5.116.204.0]/224, version 119983
Paths: (2 available, best #2)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: iBGP

  Path type: internal, path is valid, not best reason: Neighbor Address, no labeled
nexthop
  Gateway IP: 0.0.0.0
  AS-Path: 65001 5300 , path sourced external to AS
    3::52 (metric 200) from 2::66 (2.0.0.66)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 3003301
```

```
      Extcommunity: RT:2:3003301 ENCAP:8 Router MAC:f80b.cb53.4897
      Originator: 2.0.0.52 Cluster list: 2.0.0.66

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 2 destination(s)
             Imported paths list: evpn-tenant-0301 default
  Gateway IP: 0.0.0.0
  AS-Path: 65001 5300 , path sourced external to AS
    3::52 (metric 200) from 2::21 (2.0.0.21)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 3003301
      Extcommunity: RT:2:3003301 ENCAP:8 Router MAC:f80b.cb53.4897
      Originator: 2.0.0.52 Cluster list: 2.0.0.21

  Path-id 1 not advertised to any peer
```

show l2route peerid コマンドの例

```
switch# show l2route peerid
NVE Ifhdl    IP Address                                PeerID       Ifindex      Num of
MAC's Num of NH's
------------ ---------------------------------------- ------------ ------------ ------------
 ------------
1224736769   4999:1::1:1:1                             4            1191182340   23377
     0
```

show l2route topology detail コマンドの例

```
switch# show l2route topology detail
Flags:(L2cp)=L2 Ctrl Plane; (Dp)=Data Plane; (Imet)=Data Plane BGP IMET; (L3cp)=L3 Ctrl
 Plane; (Bfd)=BFD over Vxlan; (Bgp)=BGP EVPN; (Of)=Open Flow mode; (Mix)=Open Flow IR
mixed mode; (Acst)=Anycast GW on spine;
Topology ID   Topology Name   Attributes
-----------   -------------   ----------
101           Vxlan-10101     VNI: 10101
                              Encap:1 IOD:0 IfHdl:1224736769
                              VTEP IP: 5001:1::1:1:7
                              Emulated IP: ::
                              Emulated RO IP: 0.0.0.0
                              TX-ID: 2004 (Rcvd Ack: 0)
                              RMAC: 00fe.c83e.84a7, VRFID: 3
                              VMAC: 00fe.c83e.84a7
                              VMAC RO: 0000.0000.0000
                              Flags: L3cp, Sub_Flags: --, Prev_Flags: -
```

show l2route evpn imet all detail コマンドの例

```
switch# show l2route evpn imet all detail
Flags- (F): Originated From Fabric, (W): Originated from WAN

Topology ID  VNI   Prod  IP Addr        Eth Tag PMSI-Flags Flags   Type Label(VNI)
Tunnel ID     NFN Bitmap
-----------  ----- ----- -------------- ------- ---------- ------- ---- -----------
------------- ----------
901          10901 BGP   4999:1::1:1:1  0       0          -       6    10901
4999:1::1:1:1
```

show l2route fl all コマンドの例

```
switch# show l2route fl all
Topology ID Peer-id     Flood List                                Service Node
----------- ----------- ----------------------------------------- ------------
901         4           4999:1::1:1:1                             no
```

show l2route mac all detail コマンドの例

```
switch# show l2route mac all detail

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv (AD):Auto-Delete (D):Del Pending
(S):Stale (C):Clear, (Ps):Peer Sync (O):Re-Originated (Nho):NH-Override
(Pf):Permanently-Frozen, (Orp): Orphan

Topology    Mac Address    Prod   Flags         Seq No     Next-Hops

----------- -------------- ------ ------------- ----------
--------------------------------------
901         0016.0901.0001 BGP    SplRcv        0          6002:1::1:1:1

            Route Resolution Type: Regular
            Forwarding State: Resolved (PeerID: 2)
            Sent To: L2FM
          Encap: 1
```

show l2route mac-ip all detail コマンドの例

```
switch# show l2route mac-ip all detail
 Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv(D):Del Pending (S):Stale (C):Clear
(Ps):Peer Sync (Ro):Re-Originated (Orp):Orphan
Topology    Mac Address    Host IP                                   Prod   Flags
 Seq No     Next-Hops
----------- -------------- ----------------------------------------- ------ ----------
---------- ---------------
901         0016.0901.0001 46.1.1.101                                BGP    --
 0         6002:1::1:1:1
            Sent To: ARP
            encap-type:1
```

show ip route 1.191.1.0 vrf vxlan-10101 コマンドの例

```
switch# show ip route 1.191.1.0 vrf vxlan-10101
IP Route Table for VRF "vxlan-10101"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

1.191.1.0/29, ubest/mbest: 6/0
    *via fe80::2fe:c8ff:fe09:8fff%default, Po1001, [200/0], 00:56:21, bgp-4002, internal,
 tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:1, underlay_vrf: 1) encap: VXLAN

*via fe80::2fe:c8ff:fe09:8fff%default, Po1002, [200/0], 00:56:21, bgp-4002, internal,
tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:1, underlay_vrf: 1) encap: VXLAN

    *via fe80::2fe:c8ff:fe09:8fff%default, Po1001, [200/0], 00:56:32, bgp-4002, internal,
 tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:2, underlay_vrf: 1) encap: VXLAN
```

```
   *via fe80::2fe:c8ff:fe09:8fff%default, Po1002, [200/0], 00:56:32, bgp-4002, internal,
 tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:2, underlay_vrf: 1) encap: VXLAN
```

show forwarding ipv4 route 1.191.1.0 detail vrf vxlan-10101 コマンドの例

```
switch# show forwarding ipv4 route 1.191.1.0 detail vrf vxlan-10101

slot  1
=======
Prefix 1.191.1.0/29, No of paths: 2, Update time: Mon Apr 15 15:38:17 2019

   5001:1::1:1:1      nve1
   5001:1::1:1:2      nve1
```

show ipv6 route vrf vxlan-10101 コマンドの例

```
switch# show ipv6 route vrf vxlan-10101
IPv6 Routing Table for VRF "vxlan-10101"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]

2:2:2::101/128, ubest/mbest: 1/0
    *via 5001:1::1:1:1/128%default, [200/0], 00:55:31, bgp-4002, internal, tag 4002
(evpn) segid 10101
VTEP:(5001:1::1:1:1, underlay_vrf: 1) encap: VXLAN
```

の例 show forwarding distribution peer-id

コマンド

```
switch# show forwarding distribution peer-id
UFDM Peer-id allocations: App id 0
App: VXLAN   Vlan: 1     Id: 4999:1::1:1:1 0x49030001 Peer-id: 0x6
App: VXLAN   Vlan: 1     Id: 5001:1::1:1:1 0x49030001 Peer-id: 0x2
App: VXLAN   Vlan: 1     Id: 5001:1::1:1:2 0x49030001 Peer-id: 0x1
App: VXLAN   Vlan: 1     Id: 5001:1::1:1:7 0x49030001 Peer-id: 0x7
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:101 0x49030001 Peer-id: 0x8
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:102 0x49030001 Peer-id: 0x5
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:103 0x49030001 Peer-id: 0x9
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:104 0x49030001 Peer-id: 0xa
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:105 0x49030001 Peer-id: 0xb
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:106 0x49030001 Peer-id: 0xc
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:107 0x49030001 Peer-id: 0xd
```

の例 show forwarding nve l2 ingress-replication-peers

コマンド

```
switch# show forwarding nve l2 ingress-replication-peers
slot  1
=======

Total count of VLANS with ingr-repl peers: 1950
VLAN 1024 VNI 0 Vtep Ifindex 0x0 plt_space : 0x1ca75e14
         peer :  6002:1::1:1:1
         peer :  5001:1::1:1:7
         peer :  4999:1::1:1:1

PSS VLAN:1024, VNI:0, vtep:0x0x0, peer_cnt:3
```

```
        peer :  6002:1::1:1:1  marked : 0
        peer :  5001:1::1:1:7  marked : 0
        peer :  4999:1::1:1:1  marked : 0
 VLAN 1280 VNI 0 Vtep Ifindex 0x0 plt_space : 0x1ca75e14
        peer :  6002:1::1:1:1
        peer :  5001:1::1:1:7
        peer :  4999:1::1:1:1

PSS VLAN:1280, VNI:0, vtep:0x0x0, peer_cnt:3
        peer :  6002:1::1:1:1  marked : 0
        peer :  5001:1::1:1:7  marked : 0
        peer :  4999:1::1:1:1  marked : 0
```

の例 **show forwarding nve l3 peers**

コマンド

```
switch# show forwarding nve l3 peers
slot  1
=======

EVPN configuration state: disabled, PeerVni Adj enabled
NVE cleanup transaction-id 0
tunnel_id    Peer_id  Peer_address   Interface       rmac         origin state del count
-------------------------------------------------------------------------------------------
0x0        1225261062 4999:1::1:1:1       nve1      0600.0001.0001 URIB       merge-done
 no    100
0x0        1225261058 5001:1::1:1:1       nve1      2cd0.2d51.9f1b NVE        merge-done
 no    100
0x0        1225261057 5001:1::1:1:2       nve1      00a6.cab6.bbbb NVE        merge-done
 no    100
0x0        1225261063 5001:1::1:1:7       nve1      00fe.c83e.84a7 URIB       merge-done
 no    100
0x0        1225261064 5001:1::1:2:101     nve1      0000.5500.0001 URIB       merge-done
 no    100
0x0        1225261061 5001:1::1:2:102     nve1      0000.5500.0002 URIB       merge-done
 no    100
0x0        1225261065 5001:1::1:2:103     nve1      0000.5500.0003 URIB       merge-done
 no    100
0x0        1225261066 5001:1::1:2:104     nve1      0000.5500.0004 URIB       merge-done
 no    100
0x0        1225261067 5001:1::1:2:105     nve1      0000.5500.0005 URIB       merge-done
 no    100
```

の例 **show forwarding ecmp platform**

コマンド

```
switch# show forwarding ecmp platform
slot  1
=======



ECMP Hash: 0x198b8aae, Num Paths: 2, Hw index: 0x17532
Partial Install: No
Hw ecmp-index: unit-0:1073741827 unit-1:0 unit-2:0, cmn-index: 95538
Hw NVE ecmp-index: unit-0:0 unit-1:0 unit-2:0, cmn-index: 95538
Refcount: 134, Holder: 0x0,   Intf: Ethernet1/101, Nex-Hop: fe80:7::1:2
    Hw adj: unit-0:851977 unit-1:0 unit-2:0, cmn-index: 500010 LIF:4211
  Intf: Ethernet1/108, Nex-Hop: fe80:8::1:2
    Hw adj: unit-0:851978 unit-1:0 unit-2:0, cmn-index: 500012 LIF:4218
   VOBJ count: 0, VxLAN VOBJ count: 0, VxLAN: 0
```

```
ECMP Hash: 0x2bb2905e, Num Paths: 3, Hw index: 0x17533
Partial Install: No
Hw ecmp-index: unit-0:1073741828 unit-1:0 unit-2:0, cmn-index: 95539
Hw NVE ecmp-index: unit-0:0 unit-1:0 unit-2:0, cmn-index: 95539
Refcount: 16, Holder: 0x0,   Intf: Ethernet1/101, Nex-Hop: fe80:7::1:2
    Hw adj: unit-0:851977 unit-1:0 unit-2:0, cmn-index: 500010 LIF:4211
  Intf: Ethernet1/108, Nex-Hop: fe80:8::1:2
    Hw adj: unit-0:851978 unit-1:0 unit-2:0, cmn-index: 500012 LIF:4218
  Intf: port-channel1003, Nex-Hop: fe80:9::1:2
    Hw adj: unit-0:851976 unit-1:0 unit-2:0, cmn-index: 500011 LIF:4106
   VOBJ count: 0, VxLAN VOBJ count: 0, VxLAN: 0
```

の例 **show forwarding ecmp recursive**

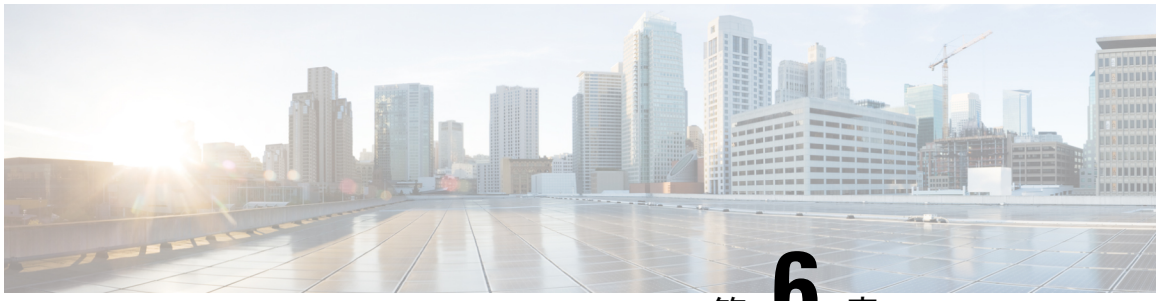
コマンド

```
switch# show forwarding ecmp recursive
slot  1
=======

Virtual Object 17 (vxlan):
    Hw vobj-index (0): unit-0:851976 unit-1:0 unit-2:0, cmn-index: 99016
    Hw NVE vobj-index (0): unit-0:0 unit-1:0 unit-2:0, cmn-index: 99016
    Hw vobj-index (1): unit-0:0 unit-1:0 unit-2:0, cmn-index: 0
    Hw NVE vobj-index (1): unit-0:0 unit-1:0 unit-2:0 cmn-index: 0
    Num prefixes : 1
Partial Install: No
    Active paths:
        Recursive NH 5001:1::1:2:10a/128 , table 0x80000001
    CNHs:
        fe80:9::1:2, port-channel1003
        Hw adj: unit-0:851976 unit-1:0 unit-2:0, cmn-index: 500011, LIF:4106
        Hw NVE adj: unit-0:0 unit-1:0 unit-2:0, cmn-index: 500011, LIF:4106
    Hw instance new : (0x182c8, 99016) ls count new 1
    FEC: fec_type 0
        VOBJ Refcount : 1
Virtual Object 167 (vxlan): ECMP-idx1:0x17536(95542), ECMP-idx2:0x0(0),
    Hw vobj-index (0): unit-0:1073741832 unit-1:0 unit-2:0, cmn-index: 99166
    Hw NVE vobj-index (0): unit-0:3 unit-1:0 unit-2:0, cmn-index: 99166
    Hw vobj-index (1): unit-0:0 unit-1:0 unit-2:0, cmn-index: 0
    Hw NVE vobj-index (1): unit-0:0 unit-1:0 unit-2:0 cmn-index: 0
    Num prefixes : 1
Partial Install: No
    Active paths:
        Recursive NH 5001:1::1:3:125/128 , table 0x80000001
    CNHs:
        fe80:7::1:2, Ethernet1/101
        Hw adj: unit-0:851977 unit-1:0 unit-2:0, cmn-index: 500010, LIF:4211
        Hw NVE adj: unit-0:0 unit-1:0 unit-2:0, cmn-index: 500010, LIF:4211
        fe80:8::1:2, Ethernet1/108
        Hw adj: unit-0:851978 unit-1:0 unit-2:0, cmn-index: 500012, LIF:4218
        Hw NVE adj: unit-0:0 unit-1:0 unit-2:0, cmn-index: 500012, LIF:4218
    Hw instance new : (0x1835e, 99166) ls count new 2
    FEC: fec_type 0
        VOBJ Refcount : 1
```

の例 **show forwarding nve l3 ecmp**

コマンド

```
switch# show forwarding nve l3 ecmp
slot  1
=======


ECMP Hash: 0x70a50e4, Num Paths: 2, Hw Index: 0x17534
table_id: 403, flags: 0x0, adj_flags: 0x0, Ref-ct: 101
  tunnel_id: 5001:1::1:1:1, segment_id: 10101
  tunnel_id: 5001:1::1:1:2, segment_id: 10101
Hw ecmp-index: unit0: 1073741830 unit1: 0 unit2: 0

ECMP Hash: 0x1189f35e, Num Paths: 2, Hw Index: 0x17535
table_id: -2147483245, flags: 0x0, adj_flags: 0x0, Ref-ct: 50
  tunnel_id: 5001:1::1:1:1, segment_id: 10101
  tunnel_id: 5001:1::1:1:2, segment_id: 10101
Hw ecmp-index: unit0: 1073741831 unit1: 0 unit2: 0
```



第 6 章

VXLAN BGP EVPN の設定

この章は、次の内容で構成されています。

VXLAN BGP EVPN について

RD Auto について

自動派生ルート識別子（rd auto）は、IETF RFC 4364 セクション 4.2 で説明されているタイプ 1 エンコーディング形式に基づいています。https://tools.ietf.org/html/rfc4364#section-4.2タイプ 1 エンコーディングでは、4 バイトの管理フィールドと 2 バイトの番号フィールドを使用できます。Cisco NX-OS 内では、自動導出 RD は、4 バイトの管理フィールド（RID）としての BGP ルータ ID の IP アドレスと、2 バイトの番号フィールド（VRF ID）の内部 VRF ID を使用して構築されます。

2 バイトの番号付けフィールドは常に VRF から取得されますが、IP-VRF または MAC-VRF での使用に応じて異なる番号付け方式になります。

- IP-VRF の 2 バイトの番号付けフィールドは、1 から始まる内部 VRF ID を使用します。VRF ID 1 および 2 は、それぞれデフォルト VRF および管理 VRF 用に予約されています。最初のカスタム定義 IP VRF は VRF ID 3 を使用します。
- MAC-VRF の 2 バイトの番号付けフィールドは、VLAN ID + 32767 を使用します。その結果、VLAN ID 1 は 32768 になります。

例：自動取得ルート識別子（RD）

- BGP ルータ ID 192.0.2.1 および VRF ID 6-RD 192.0.2.1:6 の IP-VRF

• BGP ルータ ID 192.0.2.1 および VLAN 20-RD 192.0.2.1:32787 の MAC-VRF

Route-Target Auto について

自動派生Route-Target（route-target import/export/both auto）は、IETF RFC 4364 セクション 4.2（https://tools.ietf.org/html/rfc4364#section-4.2）で説明されているタイプ 0 エンコーディング形式に基づいています。IETF RFC 4364 セクション 4.2 ではルート識別子形式について説明し、IETF RFC 4364 セクション 4.3.1では、Route-Target に同様の形式を使用することが望ましいとしています。タイプ 0 エンコーディングでは、2 バイトの管理フィールドと 4 バイトの番号フィールドを使用できます。Cisco NX-OS 内では、自動派生 Route-Target は、2 バイトの管理フィールドとして自律システム番号（ASN）、4 バイトの番号フィールドのサービス識別子（VNI）で構成されます。

2 バイト ASN

タイプ 0 エンコーディングでは、2 バイトの管理フィールドと 4 バイトの番号フィールドを使用できます。Cisco NX-OS 内では、自動派生 Route-Target は、2 バイトの管理フィールドとしての自律システム番号（ASN）と、4 バイトの番号フィールドのサービス識別子（VNI）で構成されます。

自動派生 Route-Target（RT）の例：

- ASN 65001 と L3VNI 50001 内の IP-VRF： Route-Target 65001:50001
- ASN 65001 と L2VNI 30001 内のMAC-VRF：Route-Target 65001:30001

Multi-AS 環境では、Route-Target を静的に定義するか、Route-Target の ASN 部分と一致するように書き換える必要があります。

https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/7-x/command_references/configuration_commands/b_N9K_Config_Commands_703i7x/b_N9K_Config_Commands_703i7x_chapter_010010.html#wp4498893710

4 バイト ASN

タイプ 0 エンコーディングでは、2 バイトの管理フィールドと 4 バイトの番号フィールドを使用できます。Cisco NX-OS 内では、自動派生 Route-Target は、2 バイトの管理フィールドとしての自律システム番号（ASN）と、4 バイトの番号フィールドのサービス識別子（VNI）で構成されます。4 バイト長の ASN 要求と 24 ビット（3 バイト）を必要とする VNI では、拡張コミュニティ内のサブフィールド長が使い果たされます（2 バイト タイプと 6 バイト サブフィールド）。長さと形式の制約、およびサービス識別子（VNI）の一意性の重要性の結果、4 バイトの ASN は、IETF RFC 6793 セクション 9（https://tools.ietf.org/html/rfc6793#section-9）で説明されているように、AS_TRANS という名前の 2 バイトの ASN で表されます。2 バイトの ASN 23456 は、4 バイトの ASN をエイリアスする特別な目的の AS 番号であるAS_TRANSとして IANA（https://www.iana.org/assignments/iana-as-numbers-special-registry/iana-as-numbers-special-registry.xhtml）によって登録されます。

4 バイトの ASN（AS_TRANS）を使用した自動派生 Route-Target（RT）の例：

- ASN 65656 と L3VNI 50001 内の IP-VR：Route-Target 23456:50001

- ASN 65656 とL2VNI 30001 内の MAC-VRF：Route-Target 23456:30001

（注） Cisco NX-OS リリース 9.2(1) 以降、4 バイト ASN の自動派生 Route-Target がサポートされます。

VXLAN BGP EVPN の注意事項と制約事項

VXLAN BGP EVPN には、次の注意事項と制約事項があります。

- BGP EVPN を使用する VXLAN/VTEP には、次の注意事項と制約事項が適用されます。
  - SPAN 送信元または宛先は、任意のポートでサポートされます。

  詳細については、『Cisco Nexus 9000 シリーズ NX-OS システム管理設定ガイド、リリース 9.3(x)』を参照してください。

- ARP 抑制に関係なく、VTEP（フラッドアンドラーニング、または EVPN）で SVI が有効になっている場合は、**hardware access-list tcam region arp-ether 256 double-wide** コマンドを使用して ARP-ETHER TCAM が切り分けられるようにします。この要件は、Cisco Nexus 9200、9300-EX、9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチ、および 9700-EX/FX ラインカードを備えた Cisco Nexus 9500 プラットフォーム スイッチには適用されません。

- R シリーズ ライン カードを搭載した Cisco Nexus 9504 および 9508 では、VXLAN EVPN（レイヤ 2 およびレイヤ 3）は 9636C-RX および 96136YC-R ライン カードでのみサポートされます。

- VXLAN は N9K-C92348GC-X スイッチではサポートされていません。

- セグメントルーティングまたはMPLSを介してEVPNを設定できます。詳細については、『Cisco Nexus 9000 Series NX-OS Label Switching Configuration Guide、Release 9.3(x)』を参照してください）。

- 新しい CLI encapsulation mpls コマンドを使用して MPLS トンネル カプセル化を使用できます。EVPN アドレス ファミリのラベル割り当てモードを設定できます。詳細については、『Cisco Nexus 9000 Series NX-OS Label Switching Configuration Guide、Release 9.3(x)』を参照してください。

- 2K VNI スケール設定を持つ VXLAN EVPN セットアップでは、コントロール プレーンのダウンタイムに 200 秒以上かかる場合があります。潜在的な BGP フラップを回避するには、グレースフル リスタート時間を 300 秒に延長します。

- 特定のインターフェイスでコマンド「clear ip arp <interface> vrf <vrf-name> force-delete」 を実行すると、通常そのインターフェイスに属する ARP からエントリが削除され、トラフィックが再学習されます。ただし、同じ IP の ARP がすべての ECMP パスで解決されている場合、ECMP インターフェイスの 1 つに属する ARP エントリを強制的に削除すると、そのリンクがダウンしていない限り、そのエントリが自動的に再学習されます。

- EVPN アンダーレイの IP アンナンバードは ECMP をサポートします。複数の IP アンナンバード リンクが、同じスイッチ間で背中合わせに接続されています。ARP は接続されたすべてのインターフェイスで解決されるため、ECMP が提供されます。
- Cisco NX-OS リリース 10.2(2)F 以降、次のスケール制限が強化されています — レイヤ 2 VNI、拡張レイヤ 2 VNI、レイヤ 3 VNI、分散エニーキャスト ゲートウェイを使用する SVI、インターネット ピアリング モードの IPv4 および IPv6 ホスト ルート、および ECMP パス。VXLAN スケール制限情報については、『Cisco Nexus 9000 シリーズ NX-OS 検証済みスケーラビリティ ガイド、リリース 10.2(2)F』を参照してください。
- Cisco NX-OS リリース 10.2(1q)F 以降、VXLAN EVPN は Cisco Nexus N9K-C9332D-GX2B プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN EVPN は Cisco Nexus 9364D-GX2A および 9348D-GX2A プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 9.3(5) 以降、新しい VXLAN アップリンク機能が導入されています。
  - デフォルト VRF の物理インターフェイスは、VXLAN アップリンクとしてサポートされます。
  - VRF および dot1q タグを持つサブインターフェイスを伝送するデフォルト VRF の親インターフェイスは、VXLAN アップリンクとしてサポートされます。
  - VRF 内および dot1q タグ付きのサブインターフェイスは、VXLAN アップリンクとしてサポートされません。
  - VRF の SVI は、VXLAN アップリンクとしてサポートされません。
  - 物理ピアリンクを使用する vPC では、SVI を vPC メンバー（infra-VLAN、system nve infra-vlan）間でのみバックアップ アンダーレイ、デフォルト VRF として利用できます。
  - vPC ペアでは、vPC ノードの 1 つで NVE または NVE ループバックをシャットダウンする構成はサポートされていません。これは、片側 NVE シャットまたは片側ループバック シャットでのトラフィック フェイルオーバーがサポートされていないことを意味します。
  - FEX ホストインターフェイスは VXLAN アップリンクとしてサポートされないため、VTEP を接続できません（BUD ノード）。
- vPC ボーダー ゲートウェイの起動プロセス中に、NVE ソース ループバック インターフェイスはホールド ダウン タイマーを 1 回だけではなく 2 回実行します。これは day-1 であり予期された動作です。
- NVE インターフェイスの遅延タイマーの値は、マルチサイトの遅延復元タイマーよりも小さい値に設定する必要があります。
- VXLAN セットアップでパス最大伝送ユニット（MTU）検出（PMTUD）を有効にするには、VXLAN アップリンクを **ip unreachables** で構成する必要があります。PMTUD は、パケットの発信元から宛先へのパスに沿って最小 MTU を動的に決定することで、2 つのエ

ンドポイント間のパスのフラグメンテーションを防ぎます。12-04-2022 12:35SYSTEM:USER-AUTO-STEP

- VXLAN EVPN セットアップでは、できれば **auto rd** コマンドを使用して、ボーダー ノードに一意のルート識別子を設定する必要があります。すべてのボーダーノードで一意のルート識別子を使用しないことはサポートされていません。ファブリックのすべてのVTEPに対して、一意のルート識別子を使用することを強く推奨します。
- ARP 抑制は、VTEP がこの VNI のファーストホップ ゲートウェイ（Distributed Anycast Gateway）をホストしている場合にのみ、VNI でサポートされます。この VLAN の VTEP と SVI は、分散型エニーキャスト ゲートウェイ動作用に適切に設定する必要があります。たとえば、グローバル エニーキャストゲートウェイ MAC アドレスが設定され、エニーキャスト ゲートウェイ機能が SVI の仮想 IP アドレスに設定されている必要があります。
- ローカルで発信されたタイプ2ルート（MAC/MAC-IP）のモビリティシーケンス番号は、1 つの vTEP がシーケンス番号 K を持ち、同じコンプレックス内の他の vTEP はシーケンス番号 0 の同じルートを持つことができるため、vPC ピア間で不一致になる可能性があります。これは機能上の影響はなく、ホストが移動した後でもトラフィックには影響しません。
- DHCP スヌーピング（Dynamic Host Configuration Protocol スヌーピング）は VXLAN VLAN ではサポートされません。
- RACL は、VXLAN アップリンク インターフェイスではサポートされません。VACL は、出力方向の VXLAN カプセル化解除トラフィックではサポートされません。これは、ネットワーク（VXLAN）からアクセス（イーサネット）に向かう内部トラフィックに適用されます。

  ベスト プラクティスとして、ネットワーク ディレクションへのアクセスに対して、PACL/VACL を使用します。VXLAN ACL 機能のその他のガイドラインと制限事項については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide、Release 9.3(x)』を参照してください。
- Cisco Nexus 9000 QoS バッファ ブースト機能は、VXLAN トラフィックには適用できません。
- EBGP を使用した VXLAN BGP EVPN ファブリックには、次の推奨事項が適用されます。
  - EBGP EVPN ピアリング セッション（オーバーレイ コントロール プレーン）にはループバックを使用することをお勧めします。
  - EBGP IPv4/IPv6 ピアリング セッション（アンダーレイ）に物理インターフェイスを使用することをお勧めします。
- NVE ソースインターフェイスを専用ループバック インターフェイスにバインドし、このループバックをレイヤ 3 プロトコルの機能またはピアリングと共有しないでください。VXLAN VTEP に対して専用のループバック アドレスを使用することがベスト プラクティスです。

- NVE を、レイヤ 3 プロトコルで必要な他のループバック アドレスとは別のループバック アドレスにバインドします。同じループバックを使用する NVE およびその他のレイヤ 3 プロトコルはサポートされません。
- NVE ソースインターフェイス ループバックは、デフォルト VRF に存在する必要があります。
- VTEP と外部ノード（エッジ ルータ、コアルータ、または VNF）間の EBGP ピアリングのみがサポートされます。
  - 物理インターフェイスまたはサブインターフェイスを使用した VTEP から外部ノードへの EBGP ピアリングが推奨されます。これはベスト プラクティスです（外部接続）。
  - VTEP から外部ノードへの EBGP ピアリングは、デフォルト VRF またはテナント VRF（外部接続）に存在できます。
  - VXLAN を介した VTEP から外部ノードへの EBGP ピアリングは、テナント VRF 内に存在し、ループバック インターフェイスの更新ソースを使用する必要があります（VXLAN を介したピアリング）。
  - VTEP から外部ノードへの EBGP ピアリングに SVI を使用するには、VLAN がローカルである必要があります（VXLAN 拡張ではありません）。
- VXLAN BGP EVPN を設定する場合、「システム ルーティング モード：デフォルト」のみが次のハードウェア プラットフォームに適用されます。
  - Cisco Nexus 9200 プラットフォーム スイッチ
  - Cisco Nexus 9300 プラットフォーム スイッチ
  - Cisco Nexus 9300-EX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX/FX2/FX3 プラットフォーム スイッチ
  - Cisco Nexus 9300-GX プラットフォーム スイッチ
  - X9500 ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ
  - X9700-EX および X9700-FX ラインカードを搭載した Cisco Nexus 9500 プラットフォーム スイッチ
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN BGP EVPN を構成する場合、「システム ルーティング モード: デフォルト」のみが Cisco Nexus 9300-GX2 プラットフォーム スイッチに適用されます。
- 「システム ルーティング モード」を変更するには、スイッチをリロードする必要があります。
- Cisco Nexus 9516 プラットフォームは、VXLAN EVPN ではサポートされません。

- VXLAN は Cisco Nexus 9500 プラットフォーム スイッチで次のライン カードを使用してサポートされています。
  - 9500-R
  - 9564PX
  - 9564TX
  - 9536PQ
  - 9700-EX
  - 9700-FX
- 9700-EX または -FX ライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチは、VXLAN アップリンクで 1G、10G、25G、40G、100G、および 400G をサポートします。
- Cisco Nexus 9200 および 9300-EX/FX/FX2/FX3 および -GX は、VXLAN アップリンクで 1G、10G、25G、40G、100G、および 400G をサポートします。
- Cisco NX-OS リリース 10.2(3)F 以降、Cisco Nexus 9300-GX2 プラットフォーム スイッチは、VXLAN アップリンクで10G、25G、40G、100G、および 400G をサポートします。
- Cisco Nexus 9000 プラットフォーム スイッチは、VXLAN カプセル化に UDP ポート番号 4789 に準拠する標準を使用します。この値は設定可能です。
- Application Spine Engine（ASE2）を搭載した Cisco Nexus 9200 プラットフォーム スイッチでは、パケット サイズが 99ー122 バイトに制限されています。パケット ドロップが発生する可能性があります。
- VXLAN ネットワーク ID（VNID）16777215 が予約済みであり、明示的に設定しないでください。
- Non-Disruptive In Service Software Upgrade（ND-ISSU）は、VXLAN が有効になっている Nexus 9300 でサポートされます。例外は、Cisco Nexus 9300-FX3 および 9300-GX プラットフォーム スイッチの ND-ISSU サポートです。
- VXLAN to MPLS（LDP）、VXLAN to MPLS-SR（セグメント ルーティング）、および VXLAN to SRv6 のゲートウェイ機能は、同じ Cisco Nexus 9000 シリーズ プラットフォームで動作できます。
  - VXLAN to MPLS（LDP）ゲートウェイは、Cisco Nexus 3600-R および R シリーズ ラインカードを搭載した Cisco Nexus 9500 でサポートされます。
  - VXLAN to MPLS-SR Gateway は、CR-Series ライン カードを搭載した Cisco Nexus 9300-FX2/FX3/GX および Cisco Nexus 9500 でサポートされます。
  - Cisco NX-OS Release 10.2(3)F 以降、VXLAN から MPLS-SR へのゲートウェイは、Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
  - VXLAN は、Cisco Nexus 9300-GX プラットフォームのみでサポートされます。

- Cisco NX-OS Release 10.2(3)F 以降、VXLAN から SRv6 へは、Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN と GRE の共存は、Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 スイッチ、および N9K-C93108TC-FX3P、N9K-C93180YC-FX3、N9K-X9716D-GX スイッチでサポートされます。GRE RX パス (カプセル化解除) のみがサポートされます。GRE TX パス（カプセル化）はサポートされていません。
- 複数のトンネル カプセル化（VXLAN、GRE および/または MPLS、静的ラベルまたはセグメント ルーティング）は、同じ Cisco Nexus 9000 シリーズ スイッチ上でネットワーク フォワーディング エンジン（NFE）と共存できません。

- 復元力のあるハッシュは、VXLAN VTEP が設定された次のスイッチ プラットフォームでサポートされます。
  - Cisco Nexus 9300-EX/FX/FX2/FX3/GX は ECMP 復元力のあるハッシュをサポートします。
  - ALE アップリンク ポートを備えた Cisco Nexus 9300 は、復元力のあるハッシュをサポートしていません。

（注） 復元力のあるハッシュはデフォルトではディセーブルになっています。

- Cisco NX-OS Release 10.2(3)F 移行、ECMP レジリエント ハッシュは Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- vPC VTEP として動作する Cisco Nexus 9000 プラットフォーム スイッチ上の単一の接続デバイスまたはルーテッド デバイスに **vpc orphan-ports suspend** コマンドを使用することをお勧めします。
- Cisco NX-OS リリース 10.3(2)F 以降、BGP EVPN のスタティック MAC は Cisco Nexus 9300-EX/FX/FXP/FX2/FX3/GX/GX2 シリーズ スイッチでサポートされています。
- **mac address-table static mac-address vlan vlan-id {[drop | interface {type slot/port} | port-channel number]}** コマンドは、BGP EVPN でサポートされています。
- Cisco Nexus は、SMET フラグ フィールドがオプションとして設定されている以前のバージョンの **draft-ietf-bess-evpn-igmp-mld-proxy** ドラフトに基づいて、タイプ 6 EVPN ルート（IPv4 用）をサポートします。
- エニーキャスト ゲートウェイ SVI を使用したルーティング プロトコル隣接関係はサポートされていません。

（注） VXLAN BGP EVPN のスケーラビリティについては、『Cisco Nexus 9000 Series NX-OS Verified Scalability Guide』を参照してください。

ダウンストリーム VNI を使用した VXLAN EVPN に関する

Cisco NX-OS リリース 9.3(5) では、ダウンストリーム VNI を備えた VXLAN EVPN が導入されています。以前のリリースでは、VXLAN EVPN ネットワーク内のすべてのノード間で通信を有効にするには、VNI の設定が一貫している必要があります。

VXLAN EVPN とダウンストリーム VNI は、次のソリューションを提供します。

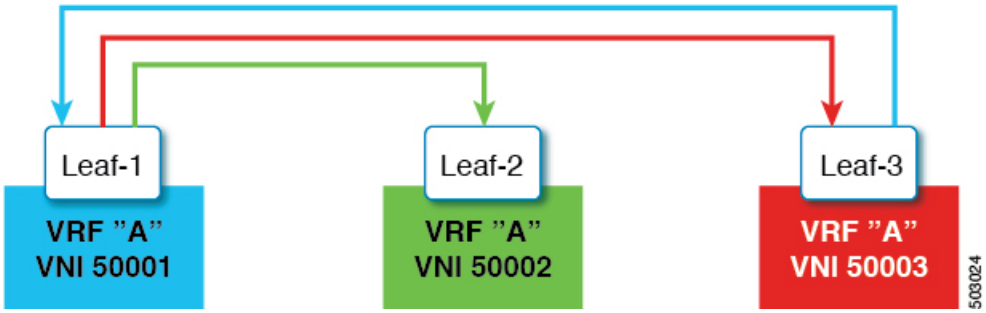
- VXLAN EVPN ネットワークのノード間での非対称 VNI 通信を有効にします。
- 顧客がドメイン外の共通の共有サービス（テナント VRF）にアクセスできるようにします。
- VNI の異なるセットを持つ分離された VXLAN EVPN サイト間の通信をサポートします。

非対称 VNI

ダウンストリーム VNI を使用する VXLAN EVPN は、非対称 VNI 割り当てをサポートします。

次の図に、非対称 VNI の例を示します。3 つの VTEP にはすべて、同じ IP VRF または MAC VRF に対して異なる VNI が設定されています。

図 *11* : 非対称 *VNI*



共有サービス VRF

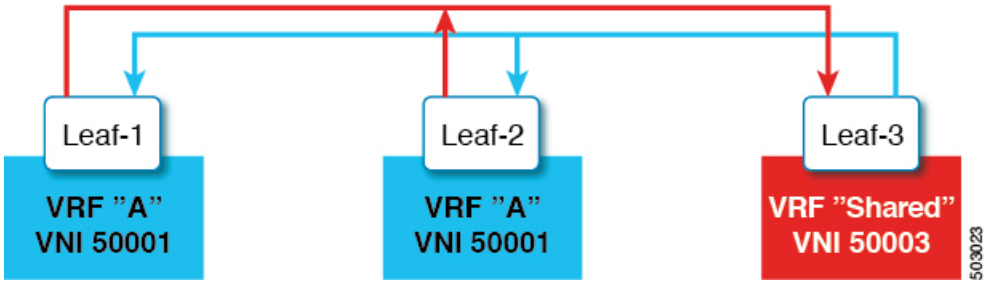
ダウンストリーム VNI を使用する VXLAN EVPN は、共有サービス VRF をサポートします。これは、複数の L3VRF を単一のローカル L3VRF にインポートし、ピア単位でダウンストリーム L3VNI の異なる値をサポートすることによって行われます。

たとえば、DNS サーバは、ホストが存在するテナント VRF に関係なく、データセンター内の複数のホストにサービスを提供する必要があります。DNS サーバは、L3VNI に接続されてい

る共有サービス VRF に接続されています。いずれかのテナント VRF からこのサーバにアクセスするには、共有サービス VRF に関連付けられた L3VNI がテナント VRF に関連付けられた L3VNI とは異なる場合でも、スイッチは共有サービス VRF からテナント VRF にルートをインポートする必要があります。

次の図では、リーフ 1 のテナント VRF A がリーフ 2 のテナント VRF A と通信できます。ただし、テナント VRF A は、リーフ 3 の背後にある共有サービスにアクセスする必要があります。

図 *12* : 共有サービス *VRF*

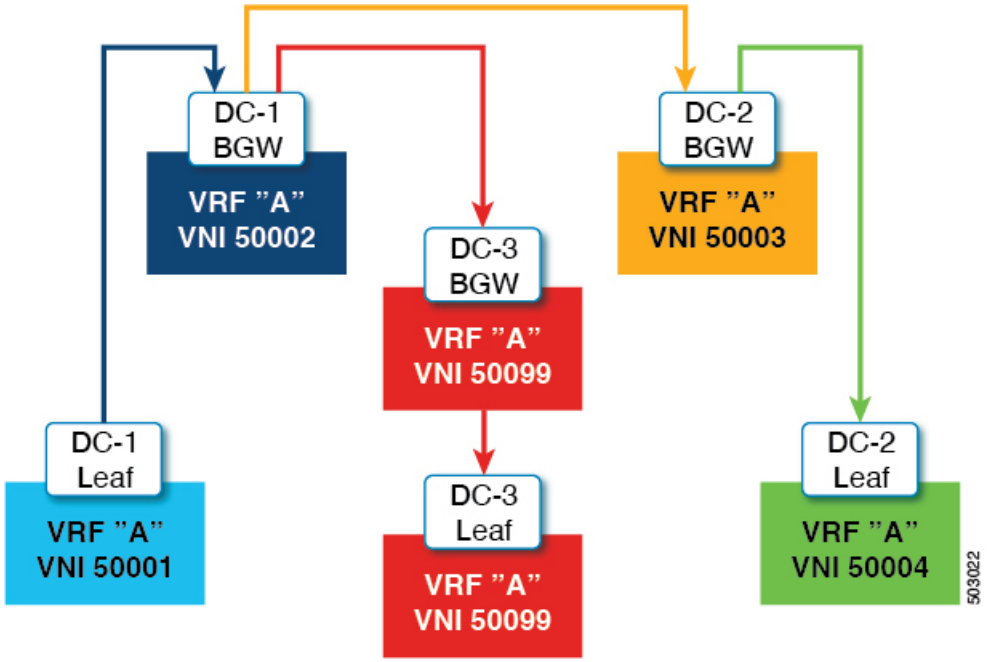


非対称 VNI を使用するマルチサイト

ダウンストリーム VNI を使用する VXLAN EVPN では、異なる VNI セットを持つサイト間の通信が可能です。これは、ボーダー ゲートウェイで非対称 VNI をスティッチングすることによって行われます。

次の図では、DC-1 と DC-2 は非対称サイトであり、DC-3 は対称サイトです。各サイトは、サイト内の異なる VNI を使用して通信します。

図 *13* : 非対称 *VNI* を使用するマルチサイト



ダウンストリーム VNI を使用する VXLAN EVPN の注意事項と制約事項

ダウンストリーム VNI をもつ VXLAN EVPN には、次の注意事項と制約事項があります。

- Cisco Nexus 9332C、9364C、9300-EX、および 9300-FX/FX2/FXP プラットフォーム スイッチと、-EX/FX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチは、ダウンストリーム VNI で VXLAN EVPN をサポートします。
- Cisco NX-OS リリース9.3(7) 以降、Cisco Nexus 9300-GX プラットフォーム スイッチは、ダウンストリーム VNI でVXLAN EVPN をサポートします。
- Cisco NX-OS リリース 10.2(3)F 以降、ダウンストリーム VNI をもつ VXLAN EVPN は Cisco Nexus 9300-FX3/GX2 プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、ダウンストリーム VNI を使用する VXLAN EVPN は Cisco Nexus 9332D-H2R スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、ダウンストリーム VNI を使用する VXLAN EVPN は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、ダウンストリーム VNI を使用する VXLAN EVPN は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- ダウンストリーム VNI を使用する VXLAN EVPN は、IPv4 アンダーレイでのみサポートされます。
- ダウンストリーム VNI は、ルート ターゲットのエクスポートおよびインポートに基づいて設定されます。ダウンストリーム VNI を活用するには、次の条件を満たす必要があります。
  - ダウンストリーム VNI では、異なる VRF（MAC-VRF または IP-VRF）を使用する必要があり、各VRFには異なる VNI（非対称 VNI）が必要です。
  - 外部 VRF（MAC-VRF または IP-VRF）のルートをインポートするには、ローカル VRF へのインポートに適したルートターゲットを構成する必要があります。
  - 自動派生ルートターゲットのみを構成すると、ダウンストリーム VNI にはなりません。
  - VRF プレフィックスのエクスポートは、静的または自動派生ルートターゲット構成によって実行できます。
  - 外部 VRF 自動派生ルートターゲットのインポートがサポートされています。
  - 外部 VRF の静的に構成されたルートターゲットのインポートがサポートされています。
- ダウンストリーム VNI は、次のアンダーレイ コンスタレーションでサポートされます。

  - レイヤ 3 VNI を使用するダウンストリーム VNI の場合、アンダーレイは入力レプリケーションまたはマルチキャスト ベースにすることができます。
  - レイヤ 2 VNI を使用するダウンストリームVNIの場合、アンダーレイは入力複製内にある必要があります。マルチキャストベースのアンダーレイは、レイヤ 2 VNI のダウンストリーム VNI ではサポートされません。

- ダウンストリーム VNI には一貫した設定が必要です。
  - サイト内のすべてのマルチサイト ボーダー ゲートウェイ（BGW）には、一貫した設定が必要です。
  - vPC ドメイン内のすべての vPC メンバーに一貫した設定が必要です。
- マルチサイトでダウンストリーム VNI を使用するには、少なくとも Cisco NX-OS リリース 9.3(5) を実行するために、すべてのサイトですべての BGW が必要です。
- 既存の中央集中型 VRF ルートリーク展開では、Cisco NX-OS リリース9.3(5) 以降への ISSU 中に短時間のトラフィック損失が発生する可能性があります。
- Cisco NX-OS リリース 9.3(5) から以前のリリースに正常にダウングレードするには、非対称 VNI 設定が削除されていることを確認します。ダウンストリーム VNI は Cisco NX-OS リリース 9.3(5) よりも前ではサポートされていないため、トラフィック転送に影響があります。
- レイヤ 3 VNI（IP-VRF）は、ピアごとに VNI 間で柔軟にマッピングできます。
  - VTEP1 上の VNI 50001 は、VNI 50001 との対称 VNI と、VTEP2 上の VNI 50002 との非対称 VNI を同時に実行できます。
  - VTEP1 の VNI 50001 は、VTEP2 の VNI 50002 および VTEP3 の VNI 50003 と非対称 VNI を実行できます。
  - VTEP1 上の VNI 50001 は、VTEP2 上の VNI 50002 および VNI5003 と非対称 VNI を同時に実行できます。
- レイヤ 2 VNI（MAC-VRF）は、ピアごとに 1 つの VNI にのみマッピングできます。
  - VTEP1 の VNI 30001 は、VTEP2 の VNI 30002 および VTEP3 の VNI 30003 と非対称 VNI を実行できます。
  - VTEP1 上の VNI 30001 は、VTEP2 上の VNI 30002 および VNI 3003 と非対称 VNI を同時に実行できません。
- VRF 内の vPC ピア ノード間の iBGP セッションはサポートされていません。
- VXLAN およびダウンストリーム VNI での BGP ピアリングは、次のコンスタレーションをサポートします。
  - 対称 VNI 間の BGP ピアリングは、ループバックを使用してサポートされます。

VXLAN BGP EVPN の設定

  - 非対称 VNI 間の BGP ピアリングは、VNI が直接メッセージ関係にある場合にサポートされます。VNI 50001（VTEP1）からのループバックは、VNI 50002（VTEP2）のループバックとピアリングできます。
  - 非対称 VNI 間の BGP ピアリングは、VNI が異なる VTEP での直接メッセージ関係にある場合にサポートされます。VNI 50001（VTEP1）からのループバックは、VNI 50002（VTEP2およびVTEP3）のループバックとピアリングできます。
  - VNI が 1:N の関係にある場合、非対称 VNI 間の BGP ピアリングはサポートされません。VNI 50001（VTEP1）のループバックは、VNI 50002（VTEP2）および VNI 50003（VTEP3）のループバックと同時にピアすることはできません。

- VXLAN 整合性チェッカは、ダウンストリーム VNI を使用する VXLAN EVPN ではサポートされません。
- ダウンストリーム VNI を使用する VXLAN EVPN は、現在、次の機能の組み合わせではサポートされていません。
  - VXLAN 静的トンネル
  - TRM およびマルチサイトでの TRM
  - CloudSec VXLAN EVPN トンネル暗号化
  - ESI ベースのマルチホーミング
  - L3VPN (MPLS SR) を備えた EVPN のシームレスな統合
  - ポリシーベース ルーティング（PBR）
- マルチサイト環境で DSVNI MAC-IP レイヤ 3 ラベル変換を有効にするには、エニーキャスト BGW で L2VNI SVI を構成してください。DSVNI の機能は、L2VNI と VRF 間の関連付けを必要とする再発信ルートに対して制限されます。L2VNI SVI で VRF member コマンドを使用して関連付けることができます。

VXLAN BGP EVPN の設定

VXLAN のイネーブル化

VXLAN および EVPN をイネーブルにします。

手順の概要

1. **feature vn-segment**
2. **feature nv overlay**
3. **feature vn-segment-vlan-based**
4. **feature interface-vlan**

5. **nv overlay evpn**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**feature vn-segment**	VLAN ベースの VXLAN をイネーブルにします。
ステップ 2	**feature nv overlay**	VXLAN をイネーブルにします。
ステップ 3	**feature vn-segment-vlan-based**	VLAN の VN-Segment を有効にします。
ステップ 4	**feature interface-vlan**	Switch Virtual Interface (SVI) を有効にします。
ステップ 5	**nv overlay evpn**	EVPN コントロール プレーンを VXLAN 用にイネーブルにします。

VLAN および VXLAN VNI の設定

（注） ステップ 3 からステップ 6 は、VXLAN VNI の VLAN を設定するためのオプションであり、カスタム ルート識別子またはルート ターゲット要件（自動派生を使用しない）の場合にのみ必要です。

手順の概要

1. **vlan** *number*
2. **vn-segment** *number*
3. **evpn**
4. **vni** *number* **l2**
5. **rd auto**
6. **route-target both** {**auto** | *rt*}

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**vlan** *number*	VLAN を指定します。
ステップ 2	**vn-segment** *number*	VXLAN VLAN でのレイヤ 2 VNI を設定するために VLAN を VXLAN VNI にマッピングします。

	コマンドまたはアクション	目的
ステップ 3	**evpn**	EVI（EVPN 仮想インスタンス）設定モードを開始します。
ステップ 4	**vni** *number* **l2**	EVI のサービスインスタンス（VNI）を指定します。
ステップ 5	**rd auto**	MAC-VRF のルート識別子（RD）を指定します。
ステップ 6	**route-target both** {**auto** | *rt*}	MAC プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、MAC-VRF ごとのプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。 （注） auto オプションの指定は IBGP のみに適用されます。 EBGP と非対称 VNI では手動で設定されたルート ターゲットが必要です。

新しい L3VNI モードの構成

新しい L3VNI モードの注意事項と制限事項

新しい L3VNI の PBR/NAT 構成の注意事項と制限事項：

- Cisco NX-OS リリース 10.2(3)F 以降、新しい L3VNI モードが Cisco Nexus 9300-X クラウド スケール スイッチでサポートされます。
- **interface vni** 構成はオプションです（PBR/NAT 機能が必要ない場合は不要です）。
- VRF-VNI-L3 の新しい構成は、暗黙的に L3VNI インターフェイスを作成します。デフォルトでは、show running コマンドには表示されません。

（注） **interface vni** を構成する前に、VRF-VNI-L3 が構成されていることを確認します。

- 次の構成は、**interface vni** で許可されます。
  - PBR/NAT
  - no interface vni
  - デフォルトのインターフェイス vni（これが存在する場合は、PBR/NAT 構成は削除されます）

- **interface vni** では **shut/no shut** コマンドは許可されていません。VRF で **shut/no shut** コマンドを実行すると、L3VNI で shut/no shut が実行されます。
- 新しい L3VNI 構成で **no feature nv overlay** を実行すると、VRF の下のすべての vrf-vni-l3 設定が削除され、PBR/NAT 設定があればクリーンアップされます。既存の VRF 設定は削除されません。
- VBU 構成の注意事項および制約事項：
  - 古い L3VNI モード構成と新しい L3VNI モード構成の両方を同じスイッチに共存させることができます。
  - VPC/VMCT システムの場合、ピア間で同じ VNI 構成モードが一貫している必要があります。
  - アップグレード後も、古い L3VNI 設定が有効です。
  - Cisco NX-OS リリース 10.3(1)F 以降、新しい L3VNI の TRM サポートが Cisco Nexus 9300-X クラウド スケール スイッチで提供されます。
  - 構成置換とロールバックがサポートされています。
  - ISSU（ND）は、新しい L3VNI でサポートされています。
- 新しい L3VNI の PBR/NAT 設定には、次の注意事項と制限事項があります。
  - NAT 構成は、新しい **interface vni** に適用できます。
  - PBR カプセル化サイド ポリシーは、カプセル化ノード インターフェイス SVI で既存のものとして設定されたままです。
  - 新しい L3VNI の PBR デキャップ サイド ポリシーが、対応する L3VNI の **interface vni** に適用されるようになりました。
  - 新しい L3VNI の PBR 構成構文は、SVI インターフェイスに似ています。
  - **no interface vni** は、最初に PBR/NAT 構成を削除してから、**interface vni** を削除します。
  - **no interface vni** は、VRF-VNI-L3 設定がまだ存在している限り、設定から CLI を削除するだけで、**interface vni** はバックエンドにまだ存在します。
- 新しい L3VNI モードでは、次の機能がサポートされています。
  - L3VNI を使用するリーフ/VTEP 機能
    - VXLAN EVPN
      - IR とマルチキャスト。
      - IGMP スヌーピング
      - vPC
      - 分散型エニーキャスト ゲートウェイ

- MCT のない vPC
- VXLAN マルチサイト
  - ボーダー リーフ、ボーダー スパイン、マルチサイト ボーダー ゲートウェイに関連した既存のすべてのシナリオに対応
  - エニーキャスト BGW および vPC BGW
- DSVNI
- VxLAN NGOAM

- VXLAN でサポートされる機能：PBR、NAT、および QoS
- VXLAN アクセス機能（QinVNI、SQinVNI、NIA、BUD-Node など）
- VXLAN ポート VLAN マッピング VXLAN 機能の 4K スケール L2VNI。

- L3VNI 構成の移行の注意事項および制約事項：
  - L3VNI 構成を古いものから新しいものに移行するには、次の手順を実行します。
    1. VLAN、vlan-vnsegment および SVI 構成を削除します。
    2. インターフェイス nve1 member-vni-associate 構成は保持します。
    3. 新しい VRF-VNI-L3 構成を追加します。詳細については、新しい L3VNI モードの構成 （140 ページ） を参照してください。
  - L3VNI 設定を新しいものから古いものに移行するには、次の手順を実行します。
    1. 新しい VRF-VNI-L3 構成を削除します。
    2. VLAN および vlan-vnsegment 構成を作成します。
    3. インターフェイス nve1 member-vni-associate 構成を保持します。
    4. L3VNI の SVI構成を作成します。
    5. VRF 構成の下に member-vni を追加します。

- アップグレードとダウンロードの注意事項と制約事項：
  - アップグレード：
    - 既存の L3VNI 設定はそのままで、機能し続けます。
    - VLAN の関連付けなしで、新しいキーワード **L3** を使用して追加の L3VNI を設定できます。
    - VLAN の関連付けなしで、既存の L3VNI 設定を新しい L3VNI に 1 つずつ移行することを選択できます。

- 必要に応じて、新しい L3VNI 構成から古い L3VNI 構成に戻すことができます（VLAN 関連付けあり）。
- ND ISSU は、新しい L3VNI の将来のリリースでサポートされます。

- ダウングレード：
  - 新しい L3 VNI が設定されている場合は、ダウングレードを実行する前に、新しい L3VNI 設定を確認して無効にします。
  - ダウングレードは、すべての新しい L3VNI 設定を削除した後にのみ許可されます。

新しい L3VNI モードの構成

この手順により、スイッチで新しい L3VNI モードが有効になります：

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **vni** *number* **l3**
4. **member vni** *vni id* **associate-vrf**
5. （任意） **{ip | ipv6} policy route-map** *map-name*
6. （任意） **ip nat outside**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**vrf context** *vrf-name* 例： `switch(config)# vrf context vxlan-501`	VRF を設定します。
ステップ 3	**vni** *number* **l3** 例： `switch(config)# vni 500001  L3`	VNI を指定します。 **L3** は、新しい L3VNI モードを示す新しいキーワードです。

	コマンドまたはアクション	目的

ステップ 4	**member vni** *vni id* **associate-vrf**
	例：
	`switch(config)# interface nve1`
	`switch(config-intf)# no shutdown`
	`switch(config-intf)# member vni 500001 associate-vrf`
		L3VNI を VRF に関連付けます。

ステップ 5	（任意） **{ip | ipv6} policy route-map** *map-name*
	例：
	`switch(config)# interface vni 500001`
	例：
	インターネットユーザに商品やサービスを提供する IPv4
	`switch(config-intf)# ip policy route-map IPV4_PBR_Appgroup`
	例：
	IPv6 の場合
	`switch(config-intf)# ipv6 policy route-map IPV6_PBR_Appgroup`
		IPv4 または IPv6 ポリシーベース ルーティング用のルート マップを L3VNI インターフェイスに割り当てます。

ステップ 6	（任意） **ip nat outside**
	例：
	`switch(config)# interface vni 500001`
	`switch(config-intf)# ip nat outside`
		NAT のルート マップを L3VNI インターフェイスに割り当てます。

新しい L3VNI モードの構成の確認

新しい L3VNI モード構成情報を表示するには、次のタスクを実行します。

コマンド	目的
Show nve vni	対応する新しい l3vni 状態を表示します

VXLAN ルーティングの VRF の設定

テナント VRF を設定します。

（注） ステップ 3ーステップ 6 は、VXLAN ルーティング用の VRF を設定するためのオプションであり、カスタム ルート識別子またはルート ターゲット要件（自動導出を使用しない）の場合にのみ必要です。

手順の概要

1. **vrf context** *vrf-name*
2. **vni** *number*
3. **rd auto**
4. **address-family** {**ipv4** | **ipv6**} **unicast**
5. **route-target both** {**auto** | *rt*}
6. **route-target both** {**auto** | *rt*} **evpn**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**vrf context** *vrf-name*	VRF を設定します。
ステップ 2	**vni** *number*	VNI を指定します。
ステップ 3	**rd auto**	IP-VRF のルート識別子（RD）を指定します。
ステップ 4	**address-family** {**ipv4** | **ipv6**} **unicast**	IPv4 または IPv6 ユニキャスト アドレス ファミリを設定します。
ステップ 5	**route-target both** {**auto** | *rt*}	IPv4 または IPv6 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、IP-VRF プレフィックス単位のインポート/エクスポートポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。 （注） **auto** オプションの指定は IBGP のみに適用されます。 EBGP と非対称 VNI では手動で設定されたルート ターゲットが必要です。
ステップ 6	**route-target both** {**auto** | *rt*} **evpn**	IPv4 または IPv6 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。 （注） **auto** オプションの指定は IBGP のみに適用されます。 EBGP と非対称 VNI では手動で設定されたルート ターゲットが必要です。

VXLAN UDP 送信元 ポートの設定

VXLAN UDP 送信元ポートを設定します。

手順の概要

1. [**no**] **vxlan udp src-port** [**high** |**rfc** |**low**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **vxlan udp src-port** [**high** |**rfc** |**low**]	VXLAN カプセル化パケットの VXLAN UDP 送信元ポート番号範囲を選択できます。 **high**：このオプションは、ポート番号の範囲を 0x8000 ～ 0xFFFF に設定します。 **rfc**：Cisco NX-OS リリース 10.4(1)F 以降では、ポート番号の範囲を 0xC000 ～ 0xFFFF に設定する **rfc** オプションが提供されています。 （注） **rfc** オプションは、Cisco Nexus 9332D-H2R、9364C-H1、および 93400LD-H1 スイッチでのみ使用できます。 **low**：Cisco NX-OS リリース 10.4(1)F 以降では、ポート番号の範囲をデフォルト値（1024 ～ 32K-1）に設定する **low** オプションが提供されています。これがデフォルトのオプションです。**high** および **rfc** コマンドの **no** フォームは、**low** コマンドと同等です。 （注） **low** オプションは、すべての Cisco Nexus 9000 シリーズプラットフォーム スイッチすべてで利用可能です。

コア向け VXLAN ルーティングの SVI の設定

コア側の SVI VRF を設定します。

手順の概要

1. **vlan** *number*
2. **vn-segment** *number*

3. **interface** *vlan-number*
4. **mtu** *vlan-number*
5. **vrf member** *vrf-name*
6. **no** {**ip** |**ipv6**} **redirects**
7. **ip forward**
8. **ipv6 address use-link-local-only**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**vlan** *number*	VLAN を指定します。
ステップ 2	**vn-segment** *number*	VXLAN VLAN でのレイヤ 3 VNI を設定するために VLAN を VXLAN VNI にマッピングします。
ステップ 3	**interface** *vlan-number*	VLAN インターフェイスを指定します。
ステップ 4	**mtu** *vlan-number*	MTU サイズ（バイト単位） <68-9216>.。
ステップ 5	**vrf member** *vrf-name*	VRF に割り当てます。
ステップ 6	**no** {**ip** |**ipv6**} **redirects**	IPv4 および IPv6 の IP リダイレクト メッセージの送信を無効にします。
ステップ 7	**ip forward**	これは、インターフェイス VLAN に定義された IP アドレスがない場合であっても、スイッチによる IPv4 ベースのルックアップを有効にします。
ステップ 8	**ipv6 address use-link-local-only**	IPv6 転送を有効にします。 （注） IPv6 アドレスの use-link-local-only は、IPv4 の IP FORWARD と同じ役割を果たします。これは、インターフェイス VLAN に定義された IP アドレスがない場合であっても、スイッチによる IP ベースのルックアップを可能にします。

コア向け VXLAN ルーティングの SVI の設定

分散デフォルト ゲートウェイとして機能するホストの SVI を設定します。

手順の概要

1. **fabric forwarding anycast-gateway-mac** *address*

2. **vlan** *number*
3. **vn-segment** *number*
4. **interface** *vlan-number*
5. **vrf member** *vrf-name*
6. **ip address** *address*
7. **fabric forwarding mode anycast-gateway**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**fabric forwarding anycast-gateway-mac** *address*	分散ゲートウェイの仮想 MAC アドレスを設定します。 （注） VTEP ごとの仮想 MAC は 1 つです。 （注） すべての VTEP が同じ仮想 MAC アドレスを持っている必要があります。
ステップ 2	**vlan** *number*	VLAN を指定します。
ステップ 3	**vn-segment** *number*	vn-segment を指定します。
ステップ 4	**interface** *vlan-number*	VLAN インターフェイスを指定します。
ステップ 5	**vrf member** *vrf-name*	VRF に割り当てます。
ステップ 6	**ip address** *address*	IP アドレスを指定します。
ステップ 7	**fabric forwarding mode anycast-gateway**	VLAN コンフィギュレーション モードで SVI をエニーキャスト ゲートウェイと関連付けます。

マルチキャストを使用する NVE インターフェイスと VNI の設定

手順の概要

1. **interface** *nve-interface*
2. **source-interface loopback1**
3. **host-reachability protocol bgp**
4. **global mcast-group** *ip-address* **{L2 | L3}**
5. **member vni** *vni*
6. **mcast-group** *ip address*
7. **member vni** *vni* **associate-vrf**
8. **mcast-group** *address*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**interface** *nve-interface*	NVE インターフェイスを設定します。
ステップ 2	**source-interface loopback1**	NVE送信元インターフェイスを専用のループバックインターフェイスにバインドします。
ステップ 3	**host-reachability protocol bgp**	これはホスト到達可能性のアドバタイズメント機構として BGP を定義します。
ステップ 4	**global mcast-group** *ip-address* **{L2 | L3}**	NVE インターフェイスごとに mcast グループをグローバルに（すべての VNI に対して）設定します。これは、すべてのレイヤ 2 またはレイヤ 3 VNI に適用され、継承されます。 （注） レイヤ 3 macst グループは、テナント ルーテッドマルチキャスト（TRM）にのみ使用されます。
ステップ 5	**member vni** *vni*	レイヤ 2 VNI をトンネル インターフェイスに追加します。
ステップ 6	**mcast-group** *ip address*	mcast group を VNI 単位で設定します。レイヤ 2 VNI 固有の mcast グループを追加し、グローバル セットの設定を上書きします。 （注） mcast グループの代わりに、入力レプリケーションを設定できます。
ステップ 7	**member vni** *vni* **associate-vrf**	レイヤ 3 VNI を、テナント VRF ごとに 1 つずつ、オーバーレイに追加します。 （注） VXLAN ルーティングのみで必要です。
ステップ 8	**mcast-group** *address*	mcast group を VNI 単位で設定します。レイヤ 3 VNI 固有の mcast グループを追加し、グローバル セットの設定を上書きします。

NVE インターフェイスでの遅延タイマーの設定

NVEインターフェイスで遅延タイマーを構成すると、BGP はVRF ピアへのファブリック ルートアドバタイズメントおよびファブリックへの VRF ピア ルートを遅延させることができるた

め、スイッチのリロード後にボーダー リーフ ノードが起動したときに一時的なトラフィック ドロップが発生しません。NX-OS ボーダー リーフおよび AnyCast ボーダー ゲートウェイでこのタイマーを構成します。

NVE インターフェイスの遅延タイマーの値は、NVE ピア、VNI、ルートなどのスケール値に依存します。構成するタイマー値を把握するには、リロード後に最後の NVE ピアをプログラムするのにかかった時間を調べ、それに 100 秒のバッファ時間を追加します。このバッファ時間は、ルート アドバタイズメントの時間も提供します。コマンドを使用して、インストールされている各 NVE ピアのタイム スタンプを表示します。**show forwarding internal trace nve-peer-history**

また、このタイマーが構成されている場合でも、NX-OS ボーダー リーフでのファブリック分離のコンバージェンスは改善されません。

手順の概要

1. **configure terminal**
2. **interface nve** *nve-interface*
3. **fabric-ready time** *seconds*
4. **show nve interface nve1 detail**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル設定モードを開始します。
ステップ 2	**interface nve** *nve-interface*	NVE インターフェイスを設定します。
ステップ 3	**fabric-ready time** *seconds*	NVEインターフェイスの遅延タイマー値を指定します。デフォルト値は 135 秒です。
ステップ 4	**show nve interface nve1 detail**	構成されたタイマー値を表示します。

VXLAN EVPN 入力複製の設定

VXLAN EVPN 入力複製において、VXLAN VTEP はネットワークにある他の VTEP の IP アドレスのリストを使用して、BUM（ブロードキャスト、未知のユニキャスト、およびマルチキャスト）トラフィックを送信します。これらの IP アドレスは、BGP EVPN コントロール プレーンを通じて VTEP 間で交換されます。

（注） VXLAN EVPN 入力複製は次のものでサポートされます。

- Cisco Nexus シリーズ 9300 シリーズ スイッチ（7.0(3)I1(2) 以降）。
- Cisco Nexus シリーズ 9500 シリーズ スイッチ（7.0(3)I2(1) 以降）。

開始する前: 次の要件は、VXLAN EVPN 入力複製の設定前に課されるものです（7.0(3)I1(2) 以降）。

- VXLAN をイネーブル化します。
- VLAN および VXLAN VNI を設定します。
- VTEP で BGP を設定します。
- VXLAN ブリッジングのルート ターゲットおよび RD を設定します。

手順の概要

1. **interface** *nve-interface*
2. **host-reachability protocol bgp**
3. **global ingress-replication protocol bgp**
4. **member vni** *vni* **associate-vrf**
5. **member vni** *vni*
6. **ingress-replication protocol bgp**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**interface** *nve-interface*	NVE インターフェイスを設定します。
ステップ 2	**host-reachability protocol bgp**	これはホスト到達可能性のアドバタイズメント機構として BGP を定義します。
ステップ 3	**global ingress-replication protocol bgp**	ローカルとリモート VTEP の IP アドレスを VNI で交換して入力複製リストを作成するため、VTEP をグローバルに（すべての VNIに）イネーブル化にします。これにより VNI の BUM トラフィックの送受信が行えるようになります。 （注） ingress-replication プロトコルを使用して、bgp はアンダーレイの設定に必要となる可能性のあるマルチキャストのニーズがなくなります。

	コマンドまたはアクション	目的
ステップ 4	**member vni** *vni* **associate-vrf**	レイヤ 3 VNI を、テナント VRF ごとに 1 つずつ、オーバーレイに追加します。 （注） VXLAN ルーティングのみで必要です。
ステップ 5	**member vni** *vni*	レイヤ 2 VNI をトンネル インターフェイスに追加します。
ステップ 6	**ingress-replication protocol bgp**	ローカルとリモートの IP アドレスを VNI で交換して入力複製リストを作成するため、VTEP をイネーブルにします。これにより VNI の BUM トラフィックの送受信が行えるようになり、グローバル設定をオーバーライドします。 （注） 入力複製の代わりに、mcast グループを設定できます。 （注） 確認するために **ingress-replication protocol bgp** アンダーレイの設定に必要となる可能性のあるマルチキャストは、すべて設定不要になります。

VTEP での BGP の設定

手順の概要

1. **router bgp** *number*
2. **router-id** *address*
3. **neighbor** *address* **remote-as** *number*
4. **address-family l2vpn evpn**
5. （任意） **Allowas-in**
6. **send-community extended**
7. **vrf** *vrf-name*
8. **address-family ipv4 unicast**
9. **advertise** *l2vpn* **evpn**
10. **maximum-paths path** {**ibgp**}
11. **address-family ipv6 unicast**
12. **advertise** *l2vpn* **evpn**
13. **maximum-paths path** {**ibgp**}

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**router bgp** *number*	BGP を設定します。
ステップ 2	**router-id** *address*	ルータ アドレスを指定します。
ステップ 3	**neighbor** *address* **remote-as** *number*	MPBGP ネイバーを定義します。各ネイバーの下に L2VPN EVPN を定義します。
ステップ 4	**address-family l2vpn evpn**	BGP ネイバーにある VPN EVPN アドレス ファミリのレイヤ 2 を設定します。 （注） VXLAN ホスト ベースのルーティング用のアドレス ファミリ IPv4 EVPN
ステップ 5	（任意） **Allowas-in**	EBGP 展開の場合のみ：AS パスで重複する自律システム（AS）番号を許可します。すべてのリーフが同じ AS を使用しているが、スパインがリーフと異なる AS を使用している場合、このパラメータを eBGP 用のリーフに設定します。
ステップ 6	**send-community extended**	BGP ネイバーのコミュニティを設定します。
ステップ 7	**vrf** *vrf-name*	VRF を指定します。
ステップ 8	**address-family ipv4 unicast**	IPv4 のアドレス ファミリを設定します。
ステップ 9	**advertise** *l2vpn* **evpn**	EVPN ルートのアドバタイジングをイネーブルにします。 （注） Cisco NX-OS リリース 9.2(1) 以降、**advertise l2vpn evpn** コマンドは有効になりません。EVPN に対する VRF のアドバタイズメントを無効にするには、インターフェイス nve1 で **no member vni** *vni* **associate-vrf** コマンドを入力して、NVE で VNI を無効にします。*vni*は、その特定の VRF に関連付けられた VNI です。
ステップ 10	**maximum-paths path** {**ibgp**}	それぞれの VRF の IPv6 アドレス ファミリ内の EVPN 転送 IP プレフィックスに対して ECMP を有効にします。
ステップ 11	**address-family ipv6 unicast**	IPv6 のアドレス ファミリを設定します。

	コマンドまたはアクション	目的
ステップ 12	**advertise** *l2vpn* **evpn**	EVPNルートのアドバタイジングをイネーブルにします。 （注） EVPN に対する VRF のアドバタイズメントを無効にするには、インターフェイス nve1 で **no member vni** *vni* **associate-vrf** コマンドを入力して、NVE で VNI を無効にします。*vni*は、その特定の VRF に関連付けられた VNI です。
ステップ 13	**maximum-paths path** {**ibgp**}	それぞれの VRF の IPv6 アドレス ファミリ内の EVPN 転送 IP プレフィックスに対して ECMP を有効にします。

スパインでの EVPN の iBGP の設定

手順の概要

1. **router bgp** *autonomous system number*
2. **neighbor** *address* **remote-as** *number*
3. **address-family l2vpn evpn**
4. **send-community extended**
5. **route-reflector-client**
6. **retain route-target all**
7. **address-family l2vpn evpn**
8. **disable-peer-as-check**
9. **route-map permitall out**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**router bgp** *autonomous system number*	BGP を指定します。
ステップ 2	**neighbor** *address* **remote-as** *number*	ネイバーを定義します。
ステップ 3	**address-family l2vpn evpn**	BGP ネイバーにある VPN EVPN アドレス ファミリのレイヤ 2 を設定します。
ステップ 4	**send-community extended**	BGP ネイバーのコミュニティを設定します。
ステップ 5	**route-reflector-client**	ルートリフレクタとしてスパインを有効にします。

	コマンドまたはアクション	目的
ステップ 6	**retain route-target all**	アドレス ファミリのレイヤ 2 VPN EVPN で、すべてのルート ターゲットの保持を [global] で設定します。 （注） eBGP では必須です。インポート ルート ターゲットに一致するように設定されたローカル VNI が存在しない場合、スパインがすべての EVPN ルートを保持およびアドバタイズできるようにします。
ステップ 7	**address-family l2vpn evpn**	BGP ネイバーにある VPN EVPN アドレス ファミリのレイヤ 2 を設定します。
ステップ 8	**disable-peer-as-check**	ルートアドバタイズメント時のピア AS 番号のチェックをディセーブルにします。すべてのリーフが同じ AS を使用しているが、スパインがリーフと異なる AS を使用している場合、このパラメータを eBGP 用のスパインに設定します。 （注） eBGP では必須です。
ステップ 9	**route-map permitall out**	ルートマップを適用してネクストホップを変更しないまま保持します。 （注） eBGP では必須です。

スパインでの EVPN の eBGP 設定

手順の概要

1. **route-map NEXT-HOP-UNCH permit 10**
2. **set ip next-hop unchanged**
3. **router bgp** *autonomous system number*
4. **address-family l2vpn evpn**
5. **retain route-target all**
6. **neighbor** *address* **remote-as** *number*
7. **address-family l2vpn evpn**
8. **disable-peer-as-check**
9. **send-community extended**
10. **route-map NEXT-HOP-UNCH out**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**route-map NEXT-HOP-UNCH permit 10**	ルートマップでは、EVPN ルート用にネクストホップを変更しないまま保持します。
ステップ 2	**set ip next-hop unchanged**	ネクストホップ アドレスを設定します。 （注）2 つのネクストホップがイネーブルの場合、ネクスト ホップの順序は維持されません。 ネクスト ホップの 1 つが VXLAN ネクスト ホップであり、他のネクスト ホップが FIB/AM/Hmm 経由でローカルに到達可能な場合、FIB/AM/Hmm 経由で到達可能なローカルネクストホップは、順序に関係なく常に取得されます。 直接/ローカル接続ネクスト ホップは、常にリモート接続ネクスト ホップよりも優先されます。
ステップ 3	**router bgp** *autonomous system number*	BGP を指定します。
ステップ 4	**address-family l2vpn evpn**	BGP ネイバーにある VPN EVPN アドレス ファミリのレイヤ 2 を設定します。
ステップ 5	**retain route-target all**	アドレス ファミリのレイヤ 2 VPN EVPN で、すべてのルート ターゲットの保持を [global] で設定します。 （注） eBGP では必須です。インポート ルート ターゲットに一致するように設定されたローカル VNI が存在しない場合、スパインがすべての EVPN ルートを保持およびアドバタイズできるようにします。
ステップ 6	**neighbor** *address* **remote-as** *number*	ネイバーを定義します。
ステップ 7	**address-family l2vpn evpn**	BGP ネイバーにある VPN EVPN アドレス ファミリのレイヤ 2 を設定します。
ステップ 8	**disable-peer-as-check**	ルート アドバタイズメント時のピア AS 番号のチェックをディセーブルにします。すべてのリーフが同じ AS を使用しているが、スパインがリーフと

	コマンドまたはアクション	目的
		異なる AS を使用している場合、このパラメータを eBGP 用のスパインに設定します。
ステップ 9	**send-community extended**	BGP ネイバーのコミュニティを設定します。
ステップ 10	**route-map NEXT-HOP-UNCH out**	ルート マップを適用してネクストホップを変更しないまま保持します。

ARP の抑制

ARP 抑制には、ハードウェアの ACL Ternary Content Addressable Memory（TCAM）リージョンのサイズ変更も含まれます。

（注） ACL TCAM リージョン設定の詳細については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「*Configuring IP ACLs*」の章を参照してください。

手順の概要

1. **hardware access-list tcam region arp-ether** *size* **double-wide**
2. **interface nve 1**
3. **global suppress-arp**
4. **member vni** *vni-id*
5. **suppress-arp**
6. **suppress-arp disable**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**hardware access-list tcam region arp-ether** *size* **double-wide**	ARP を抑制するための TCAM リージョンを設定します。 *tcam-size* —TCAM サイズ。サイズは 256 の倍数にする必要があります。サイズが 256 より大きい場合は、512 の倍数でなければなりません。 （注） TCAM 設定を有効にするには、リロードが必要です。

	コマンドまたはアクション	目的
		（注） **hardware access-list tcam region arp-ether** *size* **double-wide** コマンドの設定は、Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3 および 9300-GX プラットフォームスイッチでは必要ありません。
ステップ 2	**interface nve 1**	ネットワーク仮想化エンドポイント（NVE）インターフェイスを作成します。
ステップ 3	**global suppress-arp**	NVE インターフェイス内のすべてのレイヤ 2 VNI に対して ARP をグローバルに抑制するように設定します。
ステップ 4	**member vni** *vni-id*	VNI ID を指定します。
ステップ 5	**suppress-arp**	レイヤ2 VNI で ARP を抑制するように設定し、グローバル設定のデフォルトを上書きします。
ステップ 6	**suppress-arp disable**	特定の VNI での ARP 抑制のグローバル設定を無効にします。

VXLAN のディセーブル化

手順の概要

1. **configure terminal**
2. **no nv overlay evpn**
3. **no feature vn-segment-vlan-based**
4. **no feature nv overlay**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	コンフィギュレーション モードに入ります。
ステップ 2	**no nv overlay evpn**	EVPN コントロール プレーンをディセーブルにします。
ステップ 3	**no feature vn-segment-vlan-based**	すべての VXLAN ブリッジ ドメインのグローバル モードをディセーブルにします。

	コマンドまたはアクション	目的
ステップ 4	**no feature nv overlay**	VXLAN 機能をディセーブルにします。
ステップ 5	（任意） **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

IP アドレスと MAC アドレスの重複データ検出

IP アドレスの場合：

Cisco NX-OS は、IP アドレスの重複データ検出をサポートしています。これにより、2 つの VTEP の下で同時にホストが表示される場合、特定の期間（秒）内での移動回数に基づいた、IP アドレスの重複検出が行えます。

2 つの VTEP の下でのホストの同時可用性は、IPv4 ホストの場合は 600 ミリ秒のリフレッシュ タイムアウトで、IPv6 アドレスの場合はデフォルトのリフレッシュ タイムアウト ロジック（デフォルトは 3 秒）のホスト モビリティ ロジックによって検出されます。

デフォルトは 180 秒以内に 5 つの移動です（移動数のデフォルトは 5 つです。タイム インターバルのデフォルトは 180 秒です）。

180 秒以内に 5 つ目の移動が行われると、重複がまだ残っているかをチェックする前に、スイッチが 30 秒のロック（ホールド ダウン タイマー）をスタートさせます（シーケンス ビット増加の防止措置）。こうした 30 秒ロックの実施は 24 時間以内に最大 5 回までで（つまり 180 秒以内に 5 つの移動を 5 回分）、これを超えるとスイッチは重複エントリを恒久的にロックまたはフリーズさせます。（**show fabric forwarding ip local-host-db vrf abc**）。

ホスト IP アドレスが永続的に固定されている場合は常に、HMM によって書き込まれた syslog メッセージ。

```
2021 Aug 26 01:08:26 leaf hmm: (vrf-name) [IPv4] Freezing potential duplicate host
20.2.0.30/32, reached recover count (5) threshold
```

次に示すのは、重複 IP 検出用に特定のタイム インターバル（秒）内での VM 移動回数を設定する場合に参考になるコマンドの例です。

コマンド	説明

`switch(config)# fabric forwarding ?`
`anycast-gateway-mac`
`dup-host-ip-addr-detection`
	使用可能なサブコマンド：
	• スイッチのエニーキャスト ゲートウェイ MAC。
	• n 秒以内の重複するホスト アドレスを検出。

コマンド	説明
`switch(config)# fabric forwarding dup-host-ip-addr-detection ?` `<1-1000>`	n秒以内に許可されるホストの移動回数。指定できる移動回数の範囲は 1 ～ 1000 です。デフォルトは、5 回です。
`switch(config)# fabric forwarding dup-host-ip-addr-detection 100 ?` `<2-36000>`	ホストの移動回数における重複データ検出のタイムアウトの秒数。指定できる範囲は 2 ～ 36000 秒で、デフォルトは 180 秒です。
`switch(config)# fabric forwarding dup-host-ip-addr-detection 100 10`	10 秒間以内での重複するホスト アドレスを検出（100 個の移動までに制限）。

MAC アドレスの場合：

Cisco NX-OS は、MAC アドレスの重複データ検出をサポートしています。これによって、特定の時間間隔（秒）での移動回数に基づいて、重複した MAC アドレスを検出できます。

デフォルトは 180 秒以内に 5 つの移動です（移動数のデフォルトは 5 つです。タイム インターバルのデフォルトは 180 秒です）。

180 秒以内に 5 つ目の移動が行われると、重複がまだ残っているかをチェックする前に、スイッチが 30 秒のロック（ホールドダウン タイマー）をスタートさせます（シーケンス ビット増加の防止措置）。こうした 30 秒ロックの実施は最大 3 回までで（つまり 180 秒以内に 5 つの移動を 3 回分）、これを超えるとスイッチは重複エントリを恒久的にロックまたはフリーズさせます。（**show l2rib internal permanently-frozen-list**）。

MAC アドレスが永続的に固定されている場合は常に、L2RIB によって書き込まれた syslog メッセージ。

```
2017 Jul  5 10:27:34 leaf %$ VDC-1 %$  %USER-2-SYSTEM_MSG: Unfreeze limit (3) hit, MAC
0000.0033.3333in topo: 200 is permanently frozen - l2rib
2017 Jul  5 10:27:34 leaf %$ VDC-1 %$  %USER-2-SYSTEM_MSG: Detected duplicate host
0000.0033.3333, topology 200, during Local update, with host located at remote VTEP
1.2.3.4, VNI 2 - l2rib
2017 Jul  5 10:27:34 leaf %$ VDC-1 %$  %USER-2-SYSTEM_MSG: Unfreeze limit (3) hit, MAC
0000.0033.3334in topo: 200 is permanently frozen - l2rib
2017 Jul  5 10:27:34 leaf %$ VDC-1 %$  %USER-2-SYSTEM_MSG: Detected duplicate host
0000.0033.3334, topology 200, during Local update, with host l
```

MAC アドレスは、ローカル エントリとリモート エントリの両方が存在するまで、永久に凍結されたリストに残ります。

以下のコマンドの設定を解除しても、永久に凍結された機能が無効になることはなく、パラメーターがデフォルト値に変更されます。

- **l2rib dup-host-mac-detection**
- **l2rib dup-host-recovery**

次に示すのは、重複MAC検出用に特定のタイムインターバル（秒）内でのVM移動回数を設定する場合に参考になるコマンドの例です。

コマンド	説明

`switch(config)# l2rib dup-host-mac-detection ?`
`<1-1000>`
`default`
	L2RIBで利用可能なサブコマンド：
	• n秒以内に許可されるホストの移動回数。有効な移動回数の範囲は1 ～ 1000です。
	• デフォルト設定（180秒以内に5つの移動）。

`switch(config)# l2rib dup-host-mac-detection 100 ?` `<2-36000>`	ホストの移動回数における重複データ検出のタイムアウトの秒数。指定できる範囲は2 ～ 36000秒で、デフォルトは180秒です。
`switch(config)# l2rib dup-host-mac-detection 100 10`	10秒間以内での重複するホストアドレスを検出（100個の移動までに制限）。

L2RIB のイベント履歴サイズの設定

L2RIB コンポーネントのイベント履歴サイズを設定するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **l2rib event-history** { **mac** | **mac-ip** | **loop-detection** } **size** { **default** | **medium** | **high** | **very-high** }
3. **l2rib event-history** { **fl** | **imet** | **dme-oper** } **size** { **default** | **medium** | **high** | **very-high** }
4. **clear l2rib event-history** { **mac** | **mac-ip** | **loop-detection** } **size** { **default** | **medium** | **high** | **very-high** }

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ1	**configure terminal** 例： `switch# configure terminal`	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。

	コマンドまたはアクション	目的
ステップ 2	**l2rib event-history** { **mac** | **mac-ip** | **loop-detection** } **size** { **default** | **medium** | **high** | **very-high** } 例： `switch(config)# l2rib event-history mac size low`	L2RIB コンポーネントのイベント履歴サイズを設定します。
ステップ 3	**l2rib event-history** { **fl** | **imet** | **dme-oper** } **size** { **default** | **medium** | **high** | **very-high** } 例： `switch(config)# l2rib event-history fl size very-high`	指定された L2RIB オブジェクトのイベント ログを生成します。 • **fl**：L2RIB VXLAN フラッドリスト • **imet**：L2RIB IMET • **dme-oper**： L2RIB DME OPER （注） 大規模なスケーリング環境では、バッファサイズを **very-high** に設定してください。
ステップ 4	**clear l2rib event-history** { **mac** | **mac-ip** | **loop-detection** } **size** { **default** | **medium** | **high** | **very-high** } 例： `switch(config)# clear l2rib event-history mac size low`	L2RIB コンポーネントの設定されたイベント履歴サイズをクリアします。

VXLAN BGP EVPN 設定の確認

VXLAN BGP EVPN の設定情報を表示するには、次のいずれかのコマンドを入力します。

コマンド	目的
show nve vrf	VRF および関連する VNI を表示します。
show bgp l2vpn evpn	ルーティング テーブルの情報を表示します。
show ip arp suppression-cache [**detail** | **summary** | **vlan** *vlan* | **statistics**]	ARP 抑制情報を表示します。
show vxlan interface	VXLAN インターフェイス ステータスを表示します。

コマンド	目的
show vxlan interface | count	VXLAN VLAN 論理ポート VP カウントを表示します。 （注） VP はポート単位、VLAN 単位で割り当てられます。すべての VXLAN 対応レイヤ 2 ポートについての全 VP の合計が、論理ポート VP カウントの合計になります。たとえば、レイヤ 2 トランクインターフェイスが 10 個で、それぞれ 10 個の VXLAN VLAN がある場合、トータルの VXLAN VLAN 論理ポート VP カウントは 10 X 10 = 100 です。
show l2route evpn mac [**all** | **evi** *evi* [**bgp** | **local** | **static** | **vxlan** | **arp**]]	レイヤ 2 ルート情報を表示します。
show l2route evpn fl all	すべての fl ルートを表示します。
show l2route evpn imet all	すべての imet ルートを表示します。
show l2route evpn mac-ip all **show l2route evpn mac-ip all detail**	すべての MAC IP ルートを表示します。
show l2route topology	レイヤ 2 ルートのトポロジを表示します。

（注） BGP 設定の確認には **show ip bgp** コマンドが利用可能ですが、ベストプラクティスとして好ましいのは、その代わりに **show bgp** コマンドを使用することです。

ダウンストリーム VNI 設定による VXLAN EVPN の確認

ダウンストリーム VNI 設定情報で VXLAN EVPN を表示するには、次のいずれかのコマンドを入力します。

コマンド	目的
show bgp evi *l2-evi*	L2VNI に関連付けられている VRF を表示します。
show forwarding adjacency nve platform	対称および非対称 NVE 隣接の両方を、対応する DestInfoIndex とともに表示します。
show forwarding route vrf *vrf*	各ネクストホップの出力 VNI またはダウンストリーム VNI を表示します。

コマンド	目的
show ip route detail vrf *vrf*	各ネクストホップの出力 VNI またはダウンストリーム VNI を表示します。
show l2route evpn mac-ip all detail	リモート MAC ルートに存在するラベル付きネクストホップを表示します。
show l2route evpn imet all detail	リモート ピアに関連付けられた出力 VNI を表示します。
show nve peers control-plane-vni peer-ip *ip-address*	各 NVE 隣接の出力 VNI またはダウンストリーム VNI を表示します。

次の例は、**show bgp evi** *l2-evi* コマンドのサンプル出力を示しています。

```
switch# show bgp evi 100
-----------------------------------------------
  L2VNI ID                      : 100 (L2-100)
  RD                            : 3.3.3.3:32867
  Secondary RD                  : 1:100
  Prefixes (local/total)        : 1/6
  Created                       : Jun 23 22:35:13.368170
  Last Oper Up/Down             : Jun 23 22:35:13.369005 / never
  Enabled                       : Yes
  Associated IP-VRF             : vni100
  Active Export RT list         :
        100:100
  Active Import RT list         :
        100:100
```

次の例は、**show forwarding adjacency nve platform** コマンドのサンプル出力を示しています。

```
switch# show forwarding adjacency nve platform
slot  1
=======
IPv4 NVE adjacency information

next_hop:12.12.12.12   interface:nve1 (0x49000001)  table_id:1
  Peer_id:0x49080002  dst_addr:12.12.12.12  src_addr:13.13.13.13  RefCt:1  PBRCt:0
Flags:0x440800
cp : TRUE, DCI peer: FALSE is_anycast_ip FALSE dsvni peer: FALSE
  HH:0x7a13f  DstInfoIndex:0x3002
    tunnel init: unit-0:0x3 unit-1:0x0

next_hop:12.12.12.12   interface:nve1 (0x49000001)  table_id:1
  Peer_id:0x49080002  dst_addr:12.12.12.12  src_addr:13.13.13.13  RefCt:1  PBRCt:0
Flags:0x10440800
cp : TRUE, DCI peer: FALSE is_anycast_ip FALSE dsvni peer: TRUE
  HH:0x7a142  DstInfoIndex:0x3ffd
    tunnel init: unit-0:0x6 unit-1:0x0
...
```

次の例は、**show forwarding route vrf** *vrf* コマンドのサンプル出力を示します。

```
switch# show forwarding route vrf vrf1000

slot  1
```

```
=======

IPv4 routes for table vrf1000/base

--------------+-------------------+--------------+-----------------+-----------------
Prefix        | Next-hop          | Interface    | Labels          | Partial Install
--------------+-------------------+--------------+-----------------+-----------------
…..
10.1.1.11/32    12.12.12.12         nve1        dsvni: 301000
10.1.1.20/32    123.123.123.123     nve1        dsvni: 301000
10.1.1.21/32    30.30.30.30         nve1        dsvni: 301000
10.1.1.30/32    10.1.1.30           Vlan10
```

次の例は、**show ip route detail vrf** *vrf* コマンドのサンプル出力を示します。

```
switch# show ip route detail vrf default
IP Route Table for VRF "default"
  '*' denotes best ucast next-hop
  '**' denotes best mcast next-hop
  '[x/y]' denotes [preference/metric]
  '%<string>' in via output denotes VRF <string>

193.0.1.0/24, ubest/mbest: 4/0
      *via 30.1.0.2, Eth1/1, [100/0], 00:00:05, urib_dt6-client1 segid: 6544, tunnelid:
 0x7b9 encap: VXLAN

     *via 30.1.1.2, Eth1/1, [100/0], 00:00:05, urib_dt6-client1 segid: 6545, (Asymmetric)
 tunnelid: 0x7ba encap: VXLAN

     *via 30.1.2.2, Eth1/1, [100/0], 00:00:05, urib_dt6-client1 segid: 6546, (Asymmetric)
 tunnelid: 0x7bb encap: VXLAN
```

次の例は、**show l2route evpn mac-ip all detail** コマンドのサンプル出力を示しています。

```
switch# show l2route evpn mac-ip all
Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv(D):Del Pending (S):Stale (C):Clear
(Ps):Peer Sync (Ro):Re-Originated (Orp):Orphan
Topology Mac Address    Host IP   Prod   Flags Seq No  Next-Hops
-------- -------------- --------  ------ ----- ------- ----------------------------
5        0000.0005.1301 1.3.13.1  BGP    --     0      102.1.13.1 (Label: 2000005)
5        0000.0005.1401 1.3.14.1  BGP    --     0      102.1.145.1 (Label: 2000005)
```

次の例は、**show l2route evpn imet all detail** コマンドのサンプル出力を示しています。

```
switch# show l2route evpn imet all

Flags- (F): Originated From Fabric, (W): Originated from WAN

Topology ID VNI         Prod  IP Addr     Flags
----------- ----------- ----- ----------- ---------
3           2000003     BGP   102.1.13.1    -
3           2000003     BGP   102.1.31.1    -
3           2000003     BGP   102.1.32.1    -
3           2000003     BGP   102.1.145.1   -
```

次の例は、**show nve peers control-plane-vni** コマンドのサンプル出力を示しています。この例では、3000003 がダウンストリーム VNI です。

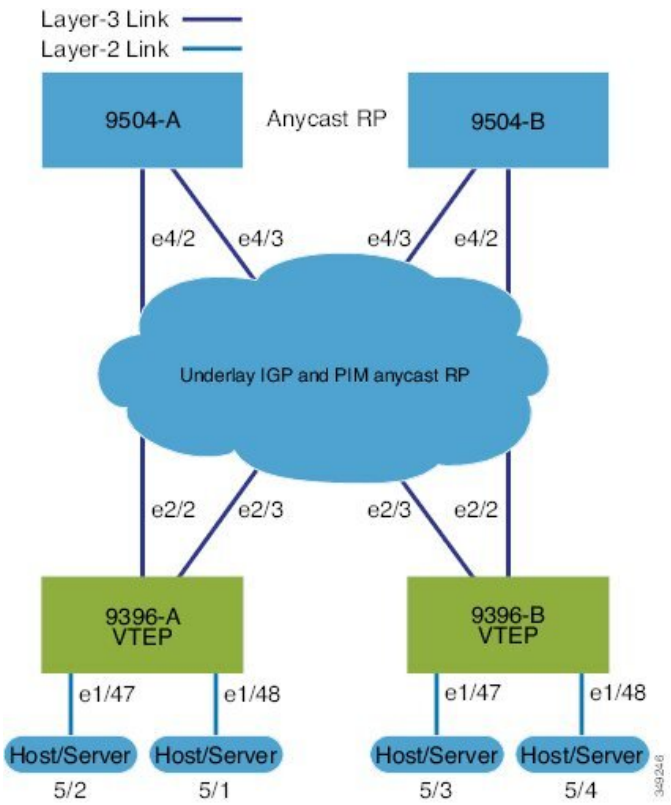
```
switch# show nve peers control-plane-vni peer-ip 203.1.1.1
Peer       VNI     Learn-Source Gateway-MAC     Peer-type  Egress-VNI SW-BD  State
```

```
---------  -----   ------------ --------------- ---------- ---------- -----
----------------------
203.1.1.1  2000003 BGP          f40f.1b6f.f8db   FAB        3000003      3005
peer-vni-add-complete
```

VXLAN BGP EVPN の例（IBGP）

VXLAN BGP EVPN の例（IBGP）。

図 *14 : VXLAN BGP EVPN* のトポロジ（*IBGP*）



スパインとリーフ間の IBGP

- スパイン（9504-A）
  - EVPN コントロール プレーンを有効にします。

    ```
    nv overlay evpn
    ```

  - 関連するプロトコルを有効にします。

    ```
    feature ospf
    feature bgp
    feature pim
    ```

  - ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 10.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

- エニーキャスト RP のループバックを設定します。

```
interface loopback1
  ip address 100.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

- エニーキャスト RP を設定します。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
ip pim anycast-rp 100.1.1.1 10.1.1.1
ip pim anycast-rp 100.1.1.1 20.1.1.1
```

- アンダーレイ ルーティング用の OSPF を有効にします。

```
router ospf 1
```

- スパインとリーフの相互接続用のインターフェイスを設定します。

```
interface Ethernet4/2
  ip address 192.168.1.42/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet4/3
  ip address 192.168.2.43/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown
```

- BGP を設定します。

```
router bgp 65535
router-id 10.1.1.1
  neighbor 30.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-reflector-client
  neighbor 40.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-reflector-client
```

- スパイン（9504-B）
  - EVPN コントロール プレーンを有効にします。

```
nv overlay evpn
```

- 他のプロトコルを有効にします

```
feature ospf
feature bgp
feature pim
```

- ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 20.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

- AnycastRP のループバックを設定します

```
interface loopback1
  ip address 100.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

- エニーキャスト RP を設定します。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
ip pim anycast-rp 100.1.1.1 10.1.1.1
ip pim anycast-rp 100.1.1.1 20.1.1.1
```

- アンダーレイ ルーティングの OSPF を有効にします

```
router ospf 1
```

- スパインとリーフの相互接続用のインターフェイスを設定します。

```
interface Ethernet4/2
  ip address 192.168.3.42/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet4/3
  ip address 192.168.4.43/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown
```

- BGP を設定します。

```
router bgp 65535
  router-id 20.1.1.1
  neighbor 30.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-reflector client
  neighbor 40.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
```

```
        send-community both
        route-reflector client
```

- リーフ（9396-A）
  - EVPN コントロール プレーンを有効にします。

```
nv overlay evpn
```

  - 関連するプロトコルを有効にします。

```
feature ospf
feature bgp
feature pim
feature interface-vlan
```

  - BGP EVPN を使用して分散型エニーキャスト ゲートウェイの配置された VxLAN を有効にします

```
feature vn-segment-vlan-based
feature nv overlay
fabric forwarding anycast-gateway-mac 0000.2222.3333
```

  - アンダーレイ ルーティング用の OSPF を有効にします。

```
router ospf 1
```

  - ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 30.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

  - ローカル VTEP IP のループバックを設定します。

```
interface loopback1
  ip address 33.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

  - スパインとリーフの相互接続用のインターフェイスを設定します。

```
interface Ethernet2/2
  no switchport
  ip address 192.168.1.22/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet2/3
  no switchport
  ip address 192.168.3.23/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  shutdown
```

• ホスト SVI（サイレント ホスト）を再配布するためのルートマップを設定します

```
route-map HOST-SVI permit 10
  match tag 54321
```

• PIM RP を設定します。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
```

• VLAN の作成

```
vlan 1001-1002
```

• オーバーレイ VRF VLAN を作成し、vn-segment を設定します。

```
vlan 101
  vn-segment 900001
```

• オーバーレイ VRF VLAN を作成し、vn-segment を設定します。

```
vlan 101
  vn-segment 900001
```

• VXLAN ルーティングのコア向け SVI を設定します

```
interface vlan101
 no shutdown
  vrf member vxlan-900001
  ip forward
  no ip redirects
  ipv6 address use-link-local-only
  no ipv6 redirects
```

• VLAN を作成し、VXLAN のマッピングを割り当てます。

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

• VRF を作成し、VNI を設定します。

```
vrf context vxlan-900001
  vni 900001
  rd auto
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target** コマンドは自動的に設定されます。

```
\
address-family ipv4 unicast
```

```
      route-target both auto
      route-target both auto evpn
    address-family ipv6 unicast
      route-target both auto
      route-target both auto evpn
```

• サーバ側 SVI を作成し、分散型エニーキャスト ゲートウェイを有効にします。

```
interface vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24 tag 54321
  ipv6 address 4:1:0:1::1/64 tag 54321
  fabric forwarding mode anycast-gateway

interface vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24 tag 54321
  ipv6 address 4:2:0:1::1/64 tag 54321
  fabric forwarding mode anycast-gateway
```

• ARP 抑制用の ACL TCAM リージョンを設定します。

（注） **hardware access-list tcam region arp-ether 256 double-wide** コマンドは、Cisco Nexus 9300-EX および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチでは必要ありません。

```
hardware access-list tcam region arp-ether 256 double-wide
```

•

（注） NVE インターフェイスを作成するには、次の 2 つのオプションのいずれかを選択できます。少数の VNI にはオプション 1 を使用します。簡易設定モードを活用するには、オプション 2 を使用します。

ネットワーク仮想化エンドポイント（NVE）インターフェイスを作成します。

オプション 1

```
interface nve1
no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2001001
    mcast-group 239.0.0.1
  member vni 2001002
    mcast-group 239.0.0.1
```

オプション 2

```
interface nve1
  source-interface loopback1
  host-reachability protocol bgp
  global mcast-group 239.0.0.1 L2
  member vni 2001001
  member vni 2001002
  member vni 2001007-2001010
```

- ホスト/サーバのインターフェイスを設定します。

```
interface Ethernet1/47
  switchport
  switchport access vlan 1002

interface Ethernet1/48
  switchport
  switchport access vlan 1001
```

- BGP を設定します。

```
router bgp 65535
  router-id 30.1.1.1
  neighbor 10.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  neighbor 20.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  vrf vxlan-900001
    address-family ipv4 unicast
      redistribute direct route-map HOST-SVI
    address-family ipv6 unicast
      redistribute direct route-map HOST-SVI
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target auto** コマンドは自動的に設定されます。

```
rd auto
    route-target import auto
    route-target export auto
```

（注） **rd auto** および **route-target** コマンドは、 **import** または **export** オプションを上書きするために使用しない限り、自動的に設定されます。

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

- リーフ（9396-B）
  - EVPN コントロール プレーンを有効にします。

```
nv overlay evpn
```

  - 関連するプロトコルを有効にします。

```
feature ospf
feature bgp
feature pim
feature interface-vlan
```

  - BGP EVPN を使用して分散エニーキャスト ゲートウェイの配置された VxLAN を有効にします。

```
feature vn-segment-vlan-based
feature nv overlay
fabric forwarding anycast-gateway-mac 0000.2222.3333
```

  - アンダーレイルーティングの OSPF の有効化

```
router ospf 1
```

  - ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 40.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

  - ローカル VTEP IP のループバックを設定します。

```
interface loopback1
  ip address 44.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

- スパインとリーフの相互接続用のインターフェイスを設定します。

```
interface Ethernet2/2
  no switchport
  ip address 192.168.3.22/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet2/3
  no switchport
  ip address 192.168.4.23/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  shutdown
```

- ホスト SVI（サイレント ホスト）を再配布するためのルートマップを設定します

```
route-map HOST-SVI permit 10
  match tag 54321
```

- PIM RP を設定します。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
```

- VLAN の作成

```
vlan 1001-1002
```

- オーバーレイ VRF VLAN を作成し、vn-segment を設定します。

```
vlan 101
  vn-segment 900001
```

- VXLAN ルーティングのコア向け SVI を設定します

```
interface vlan101
 no shutdown
  vrf member vxlan-900001
  ip forward
  no ip redirects
  ipv6 address use-link-local-only
  no ipv6 redirects
```

- VLAN を作成し、VXLAN のマッピングを割り当てます。

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

- VRF を作成し、VNI を設定します。

```
vrf context vxlan-900001
  vni 900001
  rd auto
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target** コマンドは自動的に設定されます。

```
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn
```

- サーバ側 SVI を作成し、分散エニーキャスト ゲートウェイを有効にします。

```
interface vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24
  ipv6 address 4:1:0:1::1/64
  fabric forwarding mode anycast-gateway

interface vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24
  ipv6 address 4:2:0:1::1/64
  fabric forwarding mode anycast-gateway
```

- ARP 抑制用の ACL TCAM リージョンを設定します。

（注） **hardware access-list tcam region arp-ether 256 double-wide** コマンドは、Cisco Nexus 9300-EX および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチでは必要ありません。

```
hardware access-list tcam region arp-ether 256 double-wide
```

-

（注） NVE インターフェイスを作成するには、次の 2 つのコマンド プロシージャのいずれかを選択できます。少数の VNI にはオプション 1 を使用します。簡易設定モードを活用するには、オプション 2 を使用します。

ネットワーク仮想化エンドポイント（NVE）インターフェイスを作成します。

オプション 1

```
interface nve1
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2001001
    mcast-group 239.0.0.1
  member vni 2001002
    mcast-group 239.0.0.1
```

オプション 2

```
interface nve1
  interface nve1
  source-interface loopback1
  host-reachability protocol bgp
  global mcast-group 239.0.0.1 L2
  member vni 2001001
  member vni 2001002
  member vni 2001007-2001010
```

- ホスト/サーバのインターフェイスを設定します。

```
interface Ethernet1/47
  switchport
  switchport access vlan 1002

interface Ethernet1/48
  switchport
  switchport access vlan 1001
```

- BGP を設定します。

```
router bgp 65535
  router-id 40.1.1.1
  neighbor 10.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  neighbor 20.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  vrf vxlan-900001
  vrf vxlan-900001
    address-family ipv4 unicast
      redistribute direct route-map HOST-SVI
    address-family ipv6 unicast
      redistribute direct route-map HOST-SVI
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target** コマンドは自動的に設定されます。

```
rd auto
    route-target import auto
    route-target export auto
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

- ボーダーゲートウェイ（BGW）でインターフェイスVLANを設定します。

```
interface vlan101
  no shutdown
  vrf member evpn-tenant-3103101
  no ip redirects
  ip address 101.1.0.1/16
  ipv6 address cafe:101:1::1/48
  no ipv6 redirects
  fabric forwarding mode anycast-gateway
```

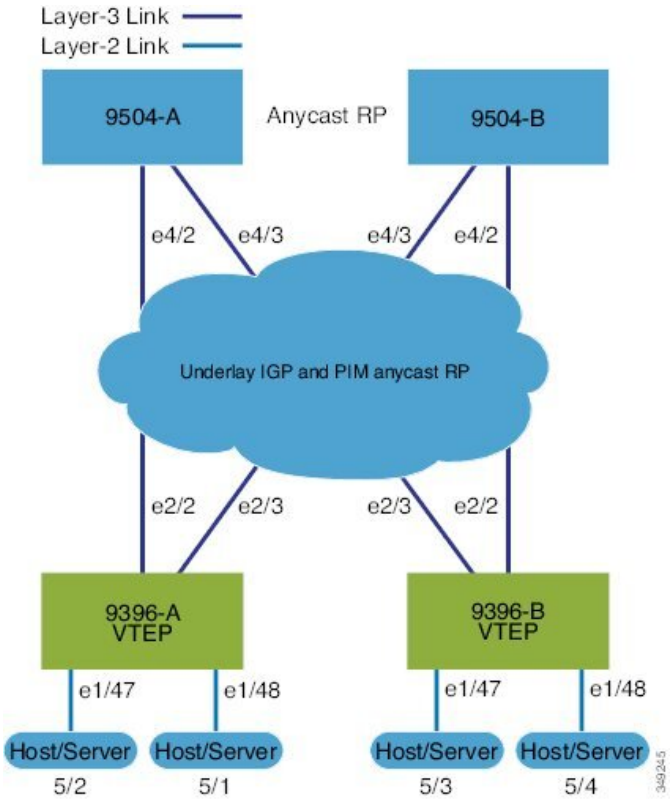
（注） BGW間にIBGPセッションがあり、EBGPファブリックが使用されている場合は、ローカルVIPまたはVIP_Rが（リロードまたはファブリックリンクフラップが原因で）ダウンしているときに、より高いAS-PATHでVIPまたはVIP_Rルートアドバタイズメントを作成するようにルートマップを設定する必要があります。次に route-map 設定例を示します。この例では、192.0.2.1がVIPアドレスで、198.51.100.1が同じBGWサイトから学習したBGP VIPルートのネクストホップです。

```
ip prefix-list vip_ip seq 5 permit 192.0.2.1/32
ip prefix-list vip_route_nh seq 5 permit 198.51.100.1/32

route-map vip_ip permit 5
  match ip address prefix-list vip_ip
  match ip next-hop prefix-list vip_route_nh
  set as-path prepend 5001 5001 5001
route-map vip_ip permit 10
```

VXLAN BGP EVPN の例（EBGP）

VXLAN BGP EVPN の例（EBGP）。

図 *15 : VXLAN BGP EVPN* のトポロジ（*EBGP*）



スパインとリーフ間の EBGP

- スパイン（9504-A）
  - EVPN コントロール プレーンを有効にします。

```
nv overlay evpn
```

  - 関連するプロトコルを有効にします。

```
feature bgp
feature pim
```

  - ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 10.1.1.1/32 tag 12345
  ip pim sparse-mode
```

  - エニーキャスト RP のループバックを設定します。

```
interface loopback1
  ip address 100.1.1.1/32 tag 12345
  ip pim sparse-mode
```

  - エニーキャスト RP を設定します。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
ip pim anycast-rp 100.1.1.1 10.1.1.1
ip pim anycast-rp 100.1.1.1 20.1.1.1
```

  - スパインで EBGP が使用する route-map を設定します。

```
route-map NEXT-HOP-UNCH permit 10
  set ip next-hop unchanged
```

  - ループバックを再配布するためのルートマップの設定

```
route-map LOOPBACK permit 10
  match tag 12345
```

  - スパインとリーフの相互接続用のインターフェイスを設定します。

```
interface Ethernet4/2
  ip address 192.168.1.42/24
  ip pim sparse-mode
  no shutdown

interface Ethernet4/3
  ip address 192.168.2.43/24
  ip pim sparse-mode
  no shutdown
```

  - EVPN アドレス ファミリ用の BGP オーバーレイを設定します。

```
router bgp 100
  router-id 10.1.1.1
  address-family l2vpn evpn
    nexthop route-map NEXT-HOP-UNCH
```

```
    retain route-target all
  neighbor 30.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
  neighbor 40.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
```

- IPv4 ユニキャスト アドレス ファミリの BGP アンダーレイを設定します。

```
 address-family ipv4 unicast
    redistribute direct route-map LOOPBACK
  neighbor 192.168.1.22 remote-as 200
    update-source ethernet4/2
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
  neighbor 192.168.2.23 remote-as 200
    update-source ethernet4/3
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
```

- スパイン（9504-B）
  - EVPN コントロール プレーンを有効にします。

```
nv overlay evpn
```

  - 関連するプロトコルを有効にします。

```
feature bgp
feature pim
```

  - ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 20.1.1.1/32 tag 12345
  ip pim sparse-mode
```

  - AnycastRP のループバックを設定します

```
interface loopback1
  ip address 100.1.1.1/32 tag 12345
  ip pim sparse-mode
```

  - エニーキャスト RP を設定します。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
ip pim anycast-rp 100.1.1.1 10.1.1.1
ip pim anycast-rp 100.1.1.1 20.1.1.1
```

- スパインで EBGP が使用する route-map を設定します。

```
route-map NEXT-HOP-UNCH permit 10
  set ip next-hop unchanged
```

- ループバックを再配布するためのルートマップの設定

```
route-map LOOPBACK permit 10
  match tag 12345
```

- スパインとリーフの相互接続用のインターフェイスを設定します。

```
interface Ethernet4/2
  no switchport
  ip address 192.168.3.42/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet4/3
  no switchport
  ip address 192.168.4.43/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  shutdown
```

- EVPN アドレス ファミリ用の BGP オーバーレイを設定します。

```
router bgp 100
  router-id 20.1.1.1
  address-family l2vpn evpn
    nexthop route-map NEXT-HOP-UNCH
    retain route-target all
  neighbor 30.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
  neighbor 40.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
```

- IPv4 ユニキャスト アドレス ファミリの BGP アンダーレイを設定します。

```
  address-family ipv4 unicast
    redistribute direct route-map LOOPBACK
```

```
  neighbor 192.168.3.22 remote-as 200
    update-source ethernet4/2
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
  neighbor 192.168.4.43 remote-as 200
    update-source ethernet4/3
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
```

- リーフ（9396-A）
  - EVPN コントロール プレーンを有効にします。

    ```
    nv overlay evpn
    ```

  - 関連プロトコルを有効にします。

    ```
    feature bgp
    feature pim
    feature interface-vlan
    ```

  - BGP EVPN を使用して分散エニーキャスト ゲートウェイの配置された VxLAN を有効にします。

    ```
    feature vn-segment-vlan-based
    feature nv overlay
    fabric forwarding anycast-gateway-mac 0000.2222.3333
    ```

  - アンダーレイ ルーティング用の OSPF を有効にします。

    ```
    router ospf 1
    ```

  - ローカル ルータ ID、PIM、および BGP のループバックを設定します。

    ```
    interface loopback0
      ip address 30.1.1.1/32
      ip pim sparse-mode
    ```

  - VTEP のループバックを設定します。

    ```
    interface loopback1
      ip address 33.1.1.1/32
      ip pim sparse-mode
    ```

  - Spine-leafi nterconnect のインターフェイスを設定します。

    ```
    interface Ethernet2/2
      no switchport
      ip address 192.168.1.22/24
      ip pim sparse-mode
      no shutdown

    interface Ethernet2/3
      no switchport
      ip address 192.168.4.23/24
    ```

```
  ip pim sparse-mode
  shutdown
```

- Host-SVI（サイレントホスト）を再配布するようにルートマップを設定します。

```
route-map HOST-SVI permit 10
  match tag 54321
```

- PIM RP を有効にします。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
```

- VLAN を作成します。

```
vlan 1001-1002
```

- オーバーレイ VRF VLAN を作成し、vn-segment を設定します。

```
vlan 101
  vn-segment 900001
```

- VXLAN ルーティングのコア向け SVI を設定します。

```
interface vlan101
 no shutdown
  vrf member vxlan-900001
  ip forward
  no ip redirects
  ipv6 address use-link-local-only
  no ipv6 redirects
```

- VLAN を作成し、VXLAN のマッピングを割り当てます。

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

- VRF を作成し、VNI を設定します。

```
vrf context vxlan-900001
  vni 900001
  rd auto
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target** コマンドは自動的に設定されます。

```
    address-family ipv4 unicast
      route-target both auto
      route-target both auto evpn
    address-family ipv6 unicast
      route-target both auto
      route-target both auto evpn
```

• サーバ側 SVI を作成し、分散エニーキャスト ゲートウェイを有効にします。

```
interface vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24 tag 54321
  ipv6 address 4:1:0:1::1/64 tag 54321
  fabric forwarding mode anycast-gateway

interface vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24 tag 54321
  ipv6 address 4:2:0:1::1/64 tag 54321
  fabric forwarding mode anycast-gateway
```

• ARP 抑制用の ACL TCAM リージョンを設定します。

（注） **hardware access-list tcam region arp-ether 256 double-wide** コマンドは、Cisco Nexus 9300-EX および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチでは必要ありません。

```
hardware access-list tcam region arp-ether 256 double-wide
```

•

（注） NVEインターフェイスを作成するには、次の2つのオプションのいずれかを選択できます。少数のVNIにはオプション1を使用します。簡易設定モードを活用するには、オプション2を使用します。

ネットワーク仮想化エンドポイント（NVE）インターフェイスを作成します。

オプション 1

```
interface nve1
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2001001
    mcast-group 239.0.0.1
  member vni 2001002
    mcast-group 239.0.0.1
```

オプション 2

```
interface nve1
  source-interface loopback1
  host-reachability protocol bgp
  global mcast-group 239.0.0.1 L2
  member vni 2001001
  member vni 2001002
  member vni 2001007-2001010
```

- ホスト/サーバのインターフェイスを設定します。

```
interface Ethernet1/47
  switchport
  switchport access vlan 1002

interface Ethernet1/48
  switchport
  switchport access vlan 1001
```

- IPv4 ユニキャスト アドレス ファミリの BGP アンダーレイを設定します。

```
router bgp 200
  router-id 30.1.1.1
  address-family ipv4 unicast
    redistribute direct route-map LOOPBACK
  neighbor 192.168.1.42 remote-as 100
    update-source ethernet2/2
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
  neighbor 192.168.4.43 remote-as 100
    update-source ethernet2/3
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
```

- EVPN アドレス ファミリ用の BGP オーバーレイを設定します。

```
  address-family l2vpn evpn
    nexthop route-map NEXT-HOP-UNCH
    retain route-target all
  neighbor 10.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
  neighbor 20.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
  vrf vxlan-900001
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target auto** コマンドは自動的に設定されます。

```
rd auto
route-target import auto
route-target export auto
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

- リーフ（9396-B）
  - EVPN コントロール プレーンを有効にします。

    ```
    nv overlay evpn
    ```

  - 関連プロトコルを有効にします。

    ```
    feature bgp
    feature pim
    feature interface-vlan
    ```

  - BGP EVPN を使用して分散エニーキャスト ゲートウェイの配置された VxLAN を有効にします。

    ```
    feature vn-segment-vlan-based
    feature nv overlay
    fabric forwarding anycast-gateway-mac 0000.2222.3333
    ```

  - アンダーレイ ルーティング用の OSPF を有効にします。

    ```
    router ospf 1
    ```

- ローカル ルータ ID、PIM、および BGP のループバックを設定します。

```
interface loopback0
  ip address 40.1.1.1/32
  ip pim sparse-mode
```

- VTEP のループバックを設定します。

```
interface loopback1
  ip address 44.1.1.1/32
  ip pim sparse-mode
```

- Spine-leafi nterconnect のインターフェイスを設定します。

```
interface Ethernet2/2
  no switchport
  ip address 192.168.3.22/24
  ip pim sparse-mode
  no shutdown

interface Ethernet2/3
  no switchport
  ip address 192.168.2.23/24
  ip pim sparse-mode
  shutdown
```

- Host-SVI（サイレントホスト）を再配布するようにルートマップを設定します。

```
route-map HOST-SVI permit 10
  match tag 54321
```

- PIM RP をイネーブルにします。

```
ip pim rp-address 100.1.1.1 group-list 224.0.0.0/4
```

- VLAN の作成

```
vlan 1001-1002
```

- オーバーレイ VRF VLAN を作成し、vn-segment を設定します。

```
vlan 101
  vn-segment 900001
```

- VXLAN ルーティングのコア向け SVI を設定します。

```
interface vlan101
 no shutdown
  vrf member vxlan-900001
  ip forward
  no ip redirects
  ipv6 address use-link-local-only
  no ipv6 redirects
```

- VLAN を作成し、VXLAN のマッピングを割り当てます。

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

• VRF を作成し、VNI を設定します。

```
vrf context vxlan-900001
  vni 900001
  rd auto
```

（注） 次のコマンドは、1 つ以上がオーバーライドとして入力されない限り、自動的に設定されます。

```
    address-family ipv4 unicast
      route-target both auto
      route-target both auto evpn
    address-family ipv6 unicast
      route-target both auto
      route-target both auto evpn
```

• サーバ側 SVI を作成し、分散型エニーキャスト ゲートウェイを有効にします。

```
interface vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24 tag 54321
  ipv6 address 4:1:0:1::1/64 tag 54321
  fabric forwarding mode anycast-gateway

interface vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24 tag 54321
  ipv6 address 4:2:0:1::1/64 tag 54321
  fabric forwarding mode anycast-gateway
```

• ARP 抑制用の ACL TCAM リージョンを設定します。

（注） **hardware access-list tcam region arp-ether 256 double-wide** コマンドは、Cisco Nexus 9300-EX および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチでは必要ありません。

```
hardware access-list tcam region arp-ether 256 double-wide
```

•

（注） NVE インターフェイスを作成するには、次の 2 つの手順のいずれかを選択できます。少数の VNI にはオプション 1 を使用します。簡易設定モードを活用するには、オプション 2 を使用します。

ネットワーク仮想化エンドポイント（NVE）インターフェイスを作成します。

オプション 1

```
interface nve1
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2001001
    mcast-group 239.0.0.1
  member vni 2001002
    mcast-group 239.0.0.1
```

オプション 2

```
interface nve1
  source-interface loopback1
  host-reachability protocol bgp
  global mcast-group 239.0.0.1 L2
  member vni 2001001
  member vni 2001002
  member vni 2001007-2001010
```

- ホスト/サーバのインターフェイスを設定します。

```
interface Ethernet1/47
  switchport
  switchport access vlan 1002

interface Ethernet1/48
  switchport
  switchport access vlan 1001
```

- IPv4 ユニキャスト アドレス ファミリの BGP アンダーレイを設定します。

```
router bgp 200
  router-id 40.1.1.1
  address-family ipv4 unicast
    redistribute direct route-map LOOPBACK
  neighbor 192.168.3.42 remote-as 100
    update-source ethernet2/2
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
  neighbor 192.168.2.43 remote-as 100
    update-source ethernet2/3
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
```

- EVPN アドレス ファミリ用の BGP オーバーレイを設定します。

```
  address-family l2vpn evpn
    nexthop route-map NEXT-HOP-UNCH
```

```
    retain route-target all
  neighbor 10.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
  neighbor 20.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      send-community both
      disable-peer-as-check
      route-map NEXT-HOP-UNCH out
  vrf vxlan-900001
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

（注） オーバーライドとして 1 つ以上を入力しない限り、 **rd auto** および **route-target auto** コマンドは自動的に設定されます。

```
rd auto
route-target import auto
route-target export auto
```

（注） EVPN モードで次のコマンドを入力する必要はありません。

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

show コマンドの例

- **show nve peers**

```
9396-B# show nve peers
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- ---------------  ----- --------- -------- -----------------
```

```
nve1      30.1.1.1         Up    CP        00:00:38 6412.2574.9f27
```

- **show nve vni**

```
9396-B# show nve vni
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured

Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      900001   n/a               Up    CP   L3 [vxlan-900001]
nve1      2001001  225.4.0.1         Up    CP   L2 [1001]
nve1      2001002  225.4.0.1         Up    CP   L2 [1002]
```

- **show ip arp suppression-cache detail**

```
9396-B# show ip arp suppression-cache detail

Flags: + - Adjacencies synced via CFSoE
       L - Local Adjacency
       R - Remote Adjacency
       L2 - Learnt over L2 interface

Ip Address      Age      Mac Address    Vlan Physical-ifindex    Flags

4.1.1.54        00:06:41 0054.0000.0000 1001 Ethernet1/48        L
4.1.1.51        00:20:33 0051.0000.0000 1001 (null)              R
4.2.2.53        00:06:41 0053.0000.0000 1002 Ethernet1/47        L
4.2.2.52        00:20:33 0052.0000.0000 1002 (null)              R
```

（注） **show vxlan interface** コマンドは、Cisco Nexus 99300-EX、9300-FX/FX2/FX3、および9300-GXプラットフォーム スイッチではサポートされません。

- **show vxlan interface**

```
9396-B# show vxlan interface
Interface       Vlan    VPL Ifindex     LTL             HW VP
=========       ====    ===========     ===             =====
Eth1/47         1002    0x4c07d22e      0x10000         5697
Eth1/48         1001    0x4c07d02f      0x10001         5698
```

- **show bgp l2vpn evpn summary**

```
leaf3# show bgp l2vpn evpn summary
BGP summary information for VRF default, address family L2VPN EVPN
BGP router identifier 40.0.0.4, local AS number 10
BGP table version is 60, L2VPN EVPN config peers 1, capable peers 1
21 network entries and 21 paths using 2088 bytes of memory
BGP attribute entries [8/1152], BGP AS path entries [0/0]
BGP community entries [0/0], BGP clusterlist entries [1/4]

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
40.0.0.1        4    10    8570    8565       60    0    0    5d22h 6
leaf3#
```

- **show bgp l2vpn evpn**

```
leaf3# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 60, local router ID is 40.0.0.4
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid,
>-best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist,
I-injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 40.0.0.2:32868
*>i[2]:[0]:[10001]:[48]:[0000.8816.b645]:[0]:[0.0.0.0]/216
                      40.0.0.2                           100          0 i
*>i[2]:[0]:[10001]:[48]:[0011.0000.0034]:[0]:[0.0.0.0]/216
                      40.0.0.2                           100          0 i
```

- **show l2route evpn mac all**

```
leaf3# show l2route evpn mac all
Topology    Mac Address    Prod   Next Hop (s)
----------- -------------- ------ ---------------
101         0000.8816.b645 BGP    40.0.0.2
101         0001.0000.0033 Local  Ifindex 4362086
101         0001.0000.0035 Local  Ifindex 4362086
101         0011.0000.0034 BGP    40.0.0.2
```

- **show l2route evpn mac-ip all**

```
leaf3# show l2route evpn mac-ip all
Topology ID Mac Address    Prod Host IP                 Next Hop (s)
----------- -------------- ---- ------------------------------------------------------
101         0011.0000.0034 BGP  5.1.3.2                      40.0.0.2
102         0011.0000.0034 BGP  5.1.3.2                      40.0.0.2
```

ND 抑制の構成

オーバーレイの ND 抑制

ホストが 2 つの異なる VXLAN ピアの背後にある場合、ホストから別のホストへのマルチキャスト ネイバー要請パケットは、BGP/EVPN VXLAN コアを介してフラッディングされます。

ND 抑制キャッシュは、以下によって構築されます。

- ホストで NS 要求をスヌーピングし、要求のソース IP および MAC バインディングを ND 抑制キャッシュに取り込みます。
- BGP EVPN MAC ルート アドバタイズメントによる IPv6-Host または MAC アドレス情報の学習

ND 抑制を使用すると、2 つの異なる VXLAN ピアの背後にあるホスト間通信の場合、リモート ホストが抑制キャッシュで最初に学習されない場合、NS パケットは BGP/EVPN VXLAN コ

アを介してフラッディングされます。ただし、スイッチ S1 の ND 抑制キャッシュにリモートホストが読み込まれると、S1 の背後にあるホストのリモート ホストに対する後続のすべての近隣要請要求パケットがスイッチ S1 によってプロキシされ、BGP-EVPN/VXLAN コア上の近隣要請パケットのフラッディングが防止されます。

ND 抑制キャッシュ スケール値については、『*Cisco Nexus 9000* シリーズ *NX-OS* 検証済みスケーラビリティ ガイド』を参照してください。

ND 抑制の注意事項および制限事項

ND 抑制には以下のような構成の注意事項および制限事項があります。

- Cisco NX-OS リリース 10.3（1）F 以降、Cisco Nexus 9300-X クラウド スケール スイッチは、プレーン BGP EVPN でのみ ND 抑制機能をサポートします。
- ND 抑制は、マルチサイト、仮想 MCT、IRB、集中型ゲートウェイ、ファイアウォール クラスタリング、vPC などの BGP-EVPN 機能バリアントではサポートされていません。
- ホストのリンク ローカルアドレスの場合、ND 抑制はサポートされておらず、代わりにホストのリンク ローカル アドレスのマルチキャスト NS が BGP EVPN VXLAN ネットワークのコアにフラッディングされます。
- ND 抑制は、suppress-arp が有効になっているすべての VNI で有効になります。
- ND Suppression CLI ノブは、次の条件下でのみ有効にする必要があります。
  - suppress-arp は VNI で有効にする必要があり、この VNI/VLAN に関連付けられた SVI が存在する必要があります。また、この SVI はアップ状態である必要があり、IPv4 と IPv6 の両方のアドレスが有効になっている必要があります。
  - ND 抑制は、次の条件では機能しません。
    - SVI が、suppress-arp/suppress nd が有効になっている VLAN/VNI に存在しない場合。
    - suppress-arp/suppress-nd が有効になっている VLAN VNI に関連付けられた SVI がダウンしている場合。
    - suppress-arp/suppress-nd が有効になっている VLAN/VNI に関連付けられた SVI に IPv4 アドレスのみがあり、IPv6 アドレスがない場合。
    - suppress-arp/suppress-nd が有効になっている VLAN/VNI に関連付けられた SVI に IPv6 アドレスのみがあり、IPv4 アドレスがない場合。

      上記のすべての条件では、ホスト間のトラフィックがドロップされる可能性があります。
- ND 抑制 VACL を機能させるには、**hardware access-list tcam region sup-tcam 768** コマンドを使用して、SUP TCAM サイズを 768 以上に増やします。

ND 抑制の構成

この手順では、NVEインターフェイスでND抑制機能を有効または無効にする方法について説明します。

始める前に

ARP 抑制が有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region ing-sup 768**
3. **copy running-config startup-config**
4. **reload**
5. **configure terminal**
6. **interface nve 1**
7. **[no]suppress nd**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル構成モードを開始します。
ステップ 2	**hardware access-list tcam region ing-sup 768** 例： `switch# hardware access-list tcam region ing-sup 768`	入力 SUP TCAM サイズを 768 に分割します。
ステップ 3	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。
ステップ 4	**reload** 例： `switch# reload`	スイッチをリロードします。
ステップ 5	**configure terminal** 例： `switch# configure terminal`	グローバル構成モードを開始します。

	コマンドまたはアクション	目的

ステップ 6	**interface nve 1**
	例：
	`switch(config)# interface nve 1`
	`switch(config-if-nve)#`
		interface nve 構成モードを開始します。

ステップ 7	**[no]suppress nd** 例： `switch(config-if-nve)# suppress nd`	すべての ARP 対応 VNI の ND 抑制を構成します。 オプション no は、すべての ARP 対応 VNI の ND 抑制を無効にします。

（注）
- グローバル **suppress arp** コマンドを構成すると、すべての VNI で ND 抑制が有効になります。
- グローバル **suppress arp** コマンドが構成されておらず、代わりに VNI ごとに **suppress arp** コマンドが構成されている場合、ARP 抑制が構成されているすべての VNI で ND 抑制が有効になります。
- vPC ペアで suppress arp コマンドを有効にする場合は、機能を有効にする前に、両方のピアで手順 1 ～ 4 が完了していることを確認してください。

ND 抑制構成の確認

ND 抑制構成情報を表示するには、次のコマンドのいずれかを入力します。

コマンド	目的
show run nv overlay	ND 抑制構成ステータスを表示します。
show nve vni	ARP が有効な VNI に対して ND 抑制構成が有効になっているかどうかを表示します。
show nve internal export nve	SDB で ND 抑制構成が有効になっているかどうかを表示します。
show nve internal export vni	SDB の VNI ごとの ND 抑制状態を表示します。
show ipv6 nd suppression-cache detail コマンド。	ローカルに存在する ICMPv6 キャッシュ エントリを表示します。
show ipv6 nd suppression-cache remote	リモートに存在する ICMPv6 キャッシュ エントリを表示します。

コマンド	目的
show ipv6 nd suppression-cache summary	ローカルとリモートの両方の IPv6 キャッシュ エントリの概要を表示します。
show ipv6 nd suppression-cache statistics	IPv6 ND 抑制キャッシュの統計情報を表示します。
show ipv6 nd suppression-cache vlan "vlan_id"	特定の VLAN の IPv6 ND 抑制キャッシュ エントリの詳細を表示します。

次の例は、**show run nv overlay** コマンドのサンプル出力を示しています。

```
switch(config-if-nve)# sh run nv overlay
!Command: show running-config nv overlay
!Running configuration last done at: Sat Mar 19 01:07:49 2022
!Time: Sat Mar 19 01:10:00 2022

version 10.2(3) Bios:version 07.68
feature nv overlay

vlan 101-110,200-203,500-501

interface nve1
  no shutdown
  host-reachability protocol bgp
  suppress nd
  global suppress-arp
```

次の例は、**show nve vni** コマンドのサンプル出力を示しています。

```
switch(config-if-nve-vni)# sh nve vni
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP
       S-ND Suppress ND
       SU - Suppress Unknown Unicast
       Xconn - Crossconnect
       MS-IR - Multisite Ingress Replication
       HYB - Hybrid IRB mode

Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      5000     239.2.0.2         Up    CP   L2 [500]           SA S-ND
```

次の例は、**show nve internal export nve** コマンドのサンプル出力を示しています。

```
switch(config-if-nve-vni)# sh nve internal export nve

NVE Interface information.
+-----------------------------------------------------------+
Interface: nve1, Admin State: Up,
   State: nve-intf-add-complete, Encap: vxlan
   Source interface: loopback3, VRF: default,
   Anycast-interface: <none>
   Mcast-routing src intf <none>
   Primary IP: 4.4.4.4, Secondary IP: 0.0.0.0,
   VNI-VRF: default, Allow-Src-Lpbk-Down: No,
   Advertise MAC route: No,
   Virtual-rMAC: 0000.0000.0000,
   Mcast-routing Primary IP: 0.0.0.0
   Suppress ND: 1
```

```
   Host-reachability: CP
   unknown-peer-forwarding-mode: disable
   VNI assignment mode: n/a
   Multisite bgw-if: <none> (ip: 0.0.0.0, admin/oper state: Down/Down)
    src-node-last-notify: None
    anycast-node-last-notify: None
    mcast-src-node-last-notify: None
    multi-src-node-last-notify: None

+-----------------------------------------------------------+
```

次の例は、**show nve internal export vni** コマンドのサンプル出力を示しています。

```
switch(config-if-nve-vni)# sh nve internal export vni

NVE VNI Information.
+-----------------------------------------------------------+
 VNI: 5000 [500] Mgroup: 239.2.0.2 Provision-State: vni-add-complete
  Primary: 4.4.4.4 Secondary: 0.0.0.0 SRC-VRF: default
  Encap: vxlan Repl-mode: Mcast
  Suppress ARP: SP Suppress ND: Enabled Mode: CP, VNI-VRF: <FALSE>  [vrf-id 0] [vrf flags
 0x0]
  Suppress Unknown-Unicast: FALSE
  X-connect : Disabled
  [VNI local configs] SA : TRUE, Mcast-group : TRUE, IR proto BGP: FALSE
  Config Src: CLI, VNI flags: 0x0
  Spine-AGW: Disabled, HYBRID: Disabled
  Multisite optimized IR: Disabled
  Multisite DCI Group Unknown Address

+-----------------------------------------------------------+
```

次の例は、**show ipv6 nd suppression-cache detail** コマンドのサンプル出力を示しています。

```
switch(config)# show ipv6 nd suppression-cache detail

Flags: + - Adjacencies synced via CFSoE
       L - Local Adjacency
       R - Remote Adjacency
       L2 - Learnt over L2 interface
       PS - Added via L2RIB, Peer Sync
       RO - Dervied from L2RIB Peer Sync Entry

IPv6 Address      Age      Mac Address    Vlan Physical-ifindex    Flags    Remote Vtep
 Addrs

172:11:1:1::51  00:00:18 acf2.c5f6.7641   11 Ethernet1/51        L
172:11:1:1::201 00:06:14 0000.0011.1111   11 (null)              R        30.100.1.1
172:11:1:1::101 00:06:14 74a0.2f1d.d481   11 (null)              R        10.10.11.11
```

次の例は、**show ipv6 nd suppression-cache local** コマンドのサンプル出力を示しています。

```
switch(config)# show ipv6 nd suppression-cache local

Flags: + - Adjacencies synced via CFSoE
       L - Local Adjacency
       R - Remote Adjacency
       L2 - Learnt over L2 interface

Ip Address      Age      Mac Address    Vlan Physical-ifindex    Flags

172:11:1:1::51  00:00:23 acf2.c5f6.7641   11 Ethernet1/51        L
```

次の例は、**show ipv6 nd suppression-cache remote** コマンドのサンプル出力を示しています。

```
switch(config)# show ipv6 nd suppression-cache remote

Flags: + - Adjacencies synced via CFSoE
       L - Local Adjacency
       R - Remote Adjacency
       L2 - Learnt over L2 interface
       PS - Added via L2RIB, Peer Sync
       RO - Dervied from L2RIB Peer Sync Entry

IPv6 Address      Age      Mac Address    Vlan Physical-ifindex    Flags    Remote Vtep
 Addrs

172:11:1:1::201 00:06:24 0000.0011.1111   11 (null)              R        30.100.1.1
172:11:1:1::101 00:06:24 74a0.2f1d.d481   11 (null)              R        10.10.11.11
```

次の例は、**show ipv6 nd suppression-cache statistics** コマンドのサンプル出力を示しています。

```
switch(config)# show ipv6 nd suppression-cache statistics

ND packet statistics for suppression-cache

Suppressed:

Total: 1
L3 mode :       Requests 1, Replies 1
                Flood ND Probe 0

Received:
Total: 1
 L3 mode:       NS 1, Non-local NA 0
                Non-local NS 0

Mobility Requests:
Total: 0
 L3 mode:       Remote-to-local 0, Local-to-remote 0
                Remote-to-remote 0

RARP Signal Refresh: 0

ND suppression-cache Local entry statistics
Adds 3, Deletes 0
```

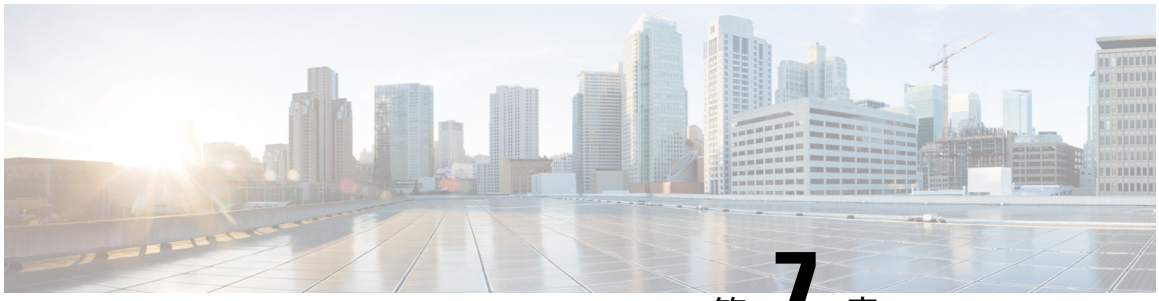
次の例は、**show ipv6 nd suppression-cache summary** コマンドのサンプル出力を示しています。

```
switch(config)# show ipv6 nd suppression-cache summary

IPV6 ND suppression-cache Summary
Remote              :2
Local               :1
Total               :3
```

次の例は、**show ipv6 nd suppression-cache vlan "vlan_id"** コマンドのサンプル出力を示しています。

```
switch(config)# show ipv6 nd suppression-cache vlan 11

Flags: + - Adjacencies synced via CFSoE
       L - Local Adjacency
       R - Remote Adjacency
       L2 - Learnt over L2 interface
       PS - Added via L2RIB, Peer Sync
       RO - Dervied from L2RIB Peer Sync Entry

IPv6 Address      Age      Mac Address    Vlan Physical-ifindex    Flags    Remote Vtep
 Addrs
```

```
172:11:1:1::51  00:00:40 acf2.c5f6.7641   11 Ethernet1/51        L
172:11:1:1::201 00:06:36 0000.0011.1111   11 (null)              R        30.100.1.1
172:11:1:1::101 00:06:36 74a0.2f1d.d481   11 (null)              R        10.10.11.11
```



第 7 章

EVPN ハイブリッド IRB モード

EVPN ハイブリッド IRB モード

EVPNハイブリッドIRBモードに関する情報

Cisco NX-OSリリース10.2（1）Fでは、EVPNハイブリッドIRBモードがサポートされています。この機能により、対称IRBモードで動作するNX-OS VTEPデバイスは、同じファブリック内の非対称IRB VTEPとシームレスに統合できます。

EVPN IRBモデル

EVPN VXLAN は VXLAN ネットワーク内の VTEP がサブネット内トラフィックをブリッジしサブネット間トラフィックをルートすることができるようにする Integrated Routing and Bridging（IRB）機能をサポートしています。EVPN-IRBオーバーレイネットワークのサブネット間ルーティングは、ファブリックVTEP全体で次の2つの方法で実装されます。

- 非対称 IRB
- 対称 IRB

非対称 IRB

非対称IRBは純粋にレイヤ2 VPNオーバーレイとしてEVPNを使用し、サブネット間トラフィックは入力VTEPでのみルーティングされます。結果として、入力 VTEP はルーティングとブリッジングの両方を実行しますが、出力 VTEP はブリッジングのみを実行します。入力VTEPでは、パケットは送信元サブネットのデフォルトゲートウェイに向けてブリッジされ、入力VTEPの宛先サブネットローカルにルーティングされます。その入力ルーティング動作から、トラフィックはレイヤ2 VPN（VNI）トンネル経由でブリッジされます。出力VTEPでの受信およびカプセル化解除後、パケットは単に宛先エンドポイントにブリッジされます。本質的に、サブネット間転送のセマンティクスに関連付けられたすべてのパケット処理は、入力VTEPに制限されます。このモデルでは、すべてのレイヤ2 VPNが、ファブリック全体で一貫したARP / NDを持つIP VRFのサブネット間手順に関係するすべてのIRB VTEP上に存在する必要があります。

対称 IRB

EVPN ハイブリッド IRB モード

対称IRBはレイヤ2およびレイヤ3 VPNオーバーレイとしてEVPNを使用し、分散型サブネット間トラフィックは任意のVTEP、入力、および出力でルーティングされます。その結果、入力および出力VTEPは、ルーティングとブリッジングの両方を実行します。入力VTEPでは、パケットは送信元サブネットのデフォルトゲートウェイに向けてブリッジされ、入力VTEP上の宛先VRFローカルにルーティングされます。この入力ルーティング動作から、トラフィックはレイヤ3 VPN（VNI）トンネルを介してルーティングされます。出力VTEPでの受信およびカプセル化解除後、パケットは最初にルーティングされ、宛先エンドポイントにブリッジされます。本質的に、サブネット間転送のセマンティクスに関連付けられたすべてのパケット処理は、すべてのVTEPに分散されます。このモデルでは、IP VRFのサブネット間手順に関係するIRB VTEPにローカルに接続されたレイヤ2 VPNだけが存在できます。 ARP / NDの消費は、エンドポイントが接続されている場所に対してローカルです。

非対称および対称相互運用

NX-OSは、対称IRBモードを使用してEVPN-IRBをサポートします。サブネット内ブリッジングを有効にするにはコントロールプレーンとデータプレーンが必要ですが、手順は対称および非対称IRBモードで同じです。サブネット内アプローチは同じですが、2つのIRBモード間のサブネット間手順には互換性がありません。その結果、同じファブリック内の対称IRB VTEPと非対称IRB VTEP間のサブネット間ルーティングはできません。

シスコのハイブリッドIRBモードでは、対称IRB VTEPは、同じファブリック内で非対称IRBモードで実行されているVTEPとシームレスに相互運用できる増分拡張をサポートします。このハイブリッドモードで有効になっているNX-OS VTEPは、ハイブリッドまたは対称IRB VTEPと通信する場合は常に、よりスケーラブルな対称IRBモードで動作します。また、ハイブリッドIRBは、非対称IRB VTEP（同じファブリック内に存在する場合）と相互運用します。

EVPNハイブリッド機能は、Cisco Nexus 9300（EX、FX、FX2、FX3、GX、N9K-9364C、N9K-9332C、N9K-C9236C、N9K-C9504.TOR、およびモジュラプラットフォーム）でサポートされています。

相互運用性コントロールプレーン

非対称と対称のIRBコントロールプレーンの主な違いは、ホストMAC + IPルート（EVPNルートタイプ2）のフォーマット方法です。非対称IRBでは、MAC + IPホストルートは、レイヤ2 VNIカプセル化およびMAC VRFルートターゲット（RT）のみでアドバタイズされます。対称IRBでは、MAC + IPホストルートは「追加の」レイヤ3 VNIおよび「追加の」IP VRF RTでアドバタイズされ、サブネット間ルーティングが可能になります。

- ハイブリッドモードでプロビジョニングされたNX-OS VTEPは、追加のL3 VNI情報とIP VRF RTを使用して、対称IRBルートタイプ2形式を使用してローカルMAC + IPルートをアドバタイズし続けます。これにより、ハイブリッドモードNX-OS VTEPは引き続きそれらの間で対称ルーティングを使用できます。

- 非対称モードで動作するVTEPは、これらの追加のL3 VNIおよびIP VRF RTフィールドを単に無視し、レイヤ3隣接関係をインストールすることによって非対称ルート手順を使用してこれらのルートを処理し、IP VRFでこれらの隣接を介してルートをホストします。レイヤ3隣接はARP / NDエントリです。

- ハイブリッドモードでプロビジョニングされたNX-OS VTEPは、非対称ルート処理を使用して非対称VTEPから受信したMAC+IPルートを処理します。その結果、レイヤ3隣接関係

EVPN ハイブリッド IRB モード

がインストールされ、非対称VTEPからアドバタイズされたリモートホストのこれらの隣接関係を介してルートがホストされます。

- その結果、NX-OSハイブリッドVTEPでは、レイヤ3隣接関係は、非対称VTEPの背後にあるホストにのみインストールされ、他のNX-OSハイブリッドVTEPの背後にあるホストにはインストールされないことに注意してください。

相互運用プロビジョニングの要件

- NX-OS対称IRB VTEPは、ファブリック内の非対称VTEPに拡張されたIP VRF内のすべてのサブネットでプロビジョニングする必要があります。
- NX-OS対称IRB VTEPは、サブネットSVIインターフェイスで「ファブリック転送モードエニーキャストゲートウェイハイブリッド」CLIを使用して「ハイブリッド」モードで非対称VTEPに拡張されたIP VRF内のサブネットでプロビジョニングする必要があります。
- 各ファブリックで非対称VTEPと相互運用する場合は、すべての対称IRB VTEPでハイブリッドモードを有効にする必要があります。

相互運用データプレーン

上記の要件の結果：

- NX-OS VTEPは、両方向で他のNX-OSハイブリッドVTEPとの対称ルーティングデータパスに従い続けます。トラフィックは、送信元サブネットでブリッジされ、L3 VNIカプセル化を使用して入力VTEPのIP VRFでルーティングされ、次にIP VRFでルーティングされ、出力VTEPの宛先サブネットでブリッジされます。
- NX-OS VTEPは、非対称VTEPの背後にあるホストへの非対称ルーティングデータパスおよびカプセル化に従います。トラフィックは、送信元サブネットでブリッジされ、ホストMAC書き換えを使用してIP VRFでルーティングされ、送信元VTEPの宛先サブネットでブリッジされますが、出力VTEPの宛先サブネットでブリッジされるだけです。

Supported Features

- ハイブリッドモードは、L3インターフェイスごとに有効にできます。
- IPv4およびIPv6オーバーレイエンドポイント
- ホストモビリティはハイブリッドモードでサポートされます。
- 入力レプリケーションとマルチキャスト アンダーレイの両方がサポートされます。
- マルチキャストと IR アンダーレイの共存は、異なる VLAN 間でサポートされます。
- 分散型エニーキャスト ゲートウェイ
- vPC

ガイドラインと制約事項

- ハイブリッドモードはDCIボーダーゲートウェイではサポートされません。

EVPN ハイブリッド IRB モード

- 分散型エニーキャストゲートウェイモードでは、非対称IRBも同じエニーキャストゲートウェイMACおよびIPでプロビジョニングする必要があります。

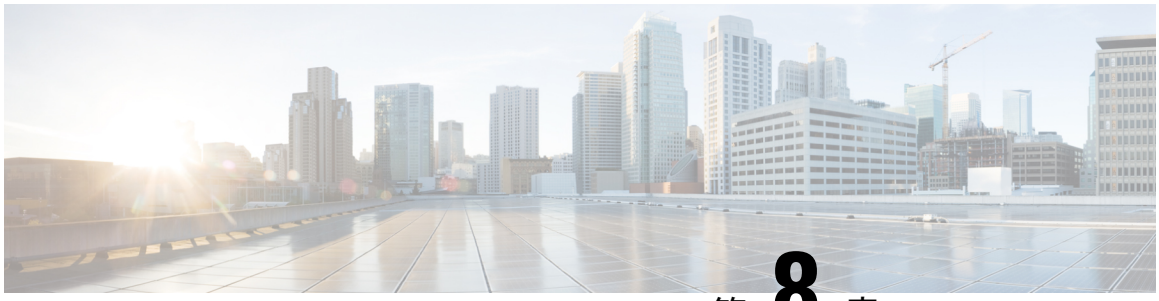
設定例：EVPNハイブリッドIRBモード

次に、EVPNハイブリッドIRBモードの設定例を示します。

```
vlan 201
vn-segment 20001
interface vlan201
no shutdown
vrf member vrf_30001
ip address 10.1.1.1/16
fabric forwarding mode anycast-gateway hybrid
```

次に、VNIとハイブリッドIRBモードを表示する例を示します。

```
switch# show nve vni
Codes: CP - Control Plane DP - Data Plane
UC - Unconfigured SA - Suppress ARP
SU - Suppress Unknown Unicast
Xconn - Crossconnect
MS-IR - Multisite Ingress Replication
HYB - Hybrid IRB Mode
Interface VNI Multicast-group State Mode Type [BD/VRF] Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1 5001 234.1.1.1 Up CP L2 [1001]
nve1 5002 234.1.1.1 Up CP L2 [1002]
nve1 5010 225.1.1.1 Up CP L2 [3003] HYB
nve1 6010 n/a Up CP L3 [vni_6010]
nve1 10001 n/a Up CP L3 [vni_10001]
nve1 30001 234.1.1.1 Up CP L2 [3001] HYB
nve1 30002 234.1.1.1 Up CP L2 [3002] HYB
```



第 8 章

HSRP とエニーキャスト ゲートウェイのデフォルト ゲートウェイの共存（VXLAN EVPN）

この章は、次の内容で構成されています。

HSRP とエニーキャスト ゲートウェイのデフォルト ゲートウェイの共存（VXLAN EVPN）

この機能は、ファースト ホップ ゲートウェイ プロトコル（HSRP がこのリリースでサポートされているモード）を使用する従来のデフォルト ゲートウェイと、VXLAN EVPN ファブリック用の分散エニーキャスト ゲートウェイ（DAG）との間の共存を提供します。中断を伴うカットオーバーや非効率的なヘア ピニングの代わりに、HSRP を使用するデフォルト ゲートウェイは、共通のデフォルト ゲートウェイの MAC および IP が構成されている限り、VXLAN EVPN の DAG と同時にアクティブにできるようになりました。この特徴の一部としての機能により、クラシック イーサネットClassic Ethernet / FabricPath と VXLAN EVPN ファブリック間の移行と共存が容易になります。この機能は、VXLAN EVPN 側、より具体的にはクラシック イーサネット/ FabricPath ネットワークに隣接するボーダー ノードでのみ有効になります。この機能により、クラシック イーサネット/FabricPath 側でソフトウェアまたはハードウェアのアップグレードを必要とせずに、より効率的なルーティングと中断の少ない移行が可能になります。

クラシック イーサネット/FabricPath HSRP ゲートウェイで事前移行手順が実行された後、DAG が VXLAN ネットワークで機能し、HSRP ゲートウェイが同じ VLAN の クラシック イーサネッ

HSRP とエニーキャスト ゲートウェイのデフォルト ゲートウェイの共存（VXLAN EVPN）

ト/FabricPath ネットワークで機能している場合でも、トラフィックへの影響を最小限に抑えて移行を実行できるようになりました。 詳細については、クラシック イーサネット/FabricPath から VXLAN への移行の構成 （205 ページ） の事前以降手順を参照してください。

以前は、移行前の手順が実行された後でも、同じ VLAN に対して DAG と HSRP ゲートウェイの両方を共存させることはできませんでした。この共存により、移行中に VXLAN ネットワークに移行されるレイヤ 3 ワークロードの最適なルーティングが可能になります。

レイヤ 2 インターコネクト

- レイヤ 2 を介して 2 つのネットワークをインターコネクトすることは、クラシック イーサネット/FabricPath から VXLAN へのシームレスなワークロード移行を促進するために重要です。
- VXLAN ネットワークの境界リーフは、レイヤ 2 インターフェイスを介してクラシック イーサネット/FabricPath ネットワークに接続されます。
- レイヤ 2 リンクは、ポート チャネル トランクまたは物理イーサネット トランクにすることができます。
- VXLAN 境界リーフ スイッチは、vPC または NX-OS スイッチにすることができ、スイッチは TOR または EOR にすることができます。同様に、従来のイーサネット/FabricPath 境界エッジ スイッチは、vPC または NX-OS スイッチにすることができます。スイッチは、従来のイーサネット/FabricPath ネットワークの HSRP ゲートウェイをホストすることもできます。

移行の場合、VXLAN 境界リーフで次を構成する必要があります。

- 2 つのインフラストラクチャに接続しているレイヤ 2 ポートは、**port-type external** として構成する必要があります。これらのポートは、外部インターフェイスと呼ばれます。
- VLAN の移行中に、IPv4 および IPv6 の固有の Burned In Address（BIA）アドレスを各 VXLAN 境界リーフの SVI で構成する必要があります。
- VXLAN 境界リーフが vPC 構成にある場合、SVI の BIA アドレスは両方のスイッチで異なる必要があります。

次の表に、レイヤ 2 相互接続のいくつかの組み合わせを示します。

表 *4:* レイヤ *2* インターコネクトの組み合わせ

VXLAN 境界リーフ	クラシック イーサネット/FabricPath 境界エッジ スイッチ
VPC	VPC
NX-OS スイッチ	NX-OS スイッチ
NX-OS スイッチ	VPC
VPC	NX-OS スイッチ

クラシック イーサネット/FabricPath から VXLAN へに移行に関する注意事項および制限事項

- VXLAN ボーダー リーフ ノードとして展開された EX/FX/FX2 プラットフォームのワークロードの移行を構成する前に、入力 PACL 領域を切り分けて使用可能にする必要があります。

  例: VXLAN およびクラシック イーサネット/FabricPath ネットワークを接続するポートで **port-type external** コマンドを設定する前に、PACL リージョンが分割されているかどうかを確認する必要があります。コマンドを使用して、入力 PACL リージョンが構成されているかどうかを確認できます。**show hardware access-list tcam region** リージョンが使用できない場合は、**hardware access-list tcam region ing-ifacl 512** コマンドを使用してリージョンを構成します。PACL リージョンが構成された後、スイッチをリロードしてください。

- 移行前に、外部インターフェイスに入力 PACL ポリシーが構成されていないことを確認してください。それらが構成されている場合は、**port-type external** コマンドを構成する前にそれらを削除する必要があります。

- この移行では、vPC ファブリック ピアリング、出力 CNTACL、VRRP、および VXLAN フラッドおよび学習はサポートされていません。また、この移行は、マルチキャストの送信元または受信者であるワークロードの移動をサポートしていません。

- 最大 6 個の外部インターフェイスのみを設定することをお勧めします。

- **hardware access-list tcam label ing-ifacl 6** 移行の場合、コマンドを使用して拡張 *IFACL* 機能が構成されていないことを確認してください。

- IPv4 および IPv6 アプリケーションの移行は、以下のように順番に実行する必要があります。

  1. 特定の VLAN の IPv4 ゲートウェイ IP の HSRP ゲートウェイで、事前移行手順を実行する必要があります。詳細については、クラシック イーサネット/FabricPath から VXLAN への移行の構成 （205 ページ） の事前以降手順を参照してください。
  2. IPv4 の BIA アドレスを使用した SVI の構成に関する移行手順は、従来のイーサネット/FabricPath ネットワークに接続された各 VXLAN ボーダー リーフ ノードで実行する必要があります。
  3. すべての IPv4 ホストを従来のイーサネット/FabricPath から VXLAN 側に移行します。
  4. すべての VLAN のすべての IPv4 ホストがクラシック イーサネット/FabricPath から VXLAN に移行されたら、移行前の手順と移行手順を IPv6 に対して繰り返す必要があります。

（注） 同時ホストの移行を最大 1000 ホストに制限することをお勧めします。ホストの前の移行が完了した後にのみ、次の移行を開始します。

- この機能は、N9K-C92348GC ではサポートされていません。
- vPC VXLAN ボーダー リーフが構成されている場合は、レイヤ 3 ピア ルータを有効にする必要があります。
- クラシック イーサネット/FabricPath から VXLAN への移行中に VXLAN ネットワークで Suppress ARP または Suppress ND 機能が有効になっている場合、VXLAN ボーダー リーフの対応するそれぞれの ARP または ND テーブルでホストを学習する必要があります。ホストを VXLAN に移動する前に GARP/ND を送信できます。

  VXLAN に移動されたホストの隣接関係が学習されていない場合、クラシック イーサネット/FabricPath ネットワークの背後にあるホストからこのホストへのトラフィックは、クラシック イーサネット/FabricPath ネットワークで失敗する可能性があります。

  次に例を示します。

  - ホスト 10.10.1.8 が VXLAN に移動されている場合、最初は次のように学習されません。

```
switch# sh ip arp 10.10.1.8 vrf vrf1501

IP ARP Table
Total number of entries: 0
Address         Age       MAC Address     Interface       Flags
switch#

switch(config)# sh ip route 10.10.1.8 vrf vrf1501

10.10.1.0/24, ubest/mbest: 2/0, attached
    *via 10.10.1.1, Vlan1001, [0/0], 22:55:42, direct
    *via 10.10.1.4, Vlan1001, [0/0], 22:55:42, direct
```

  - ホスト 10.10.1.8 から GARP を送信した後、境界リーフ スイッチの ARP テーブル出力は次のようになります。

```
switch# sh ip arp 10.10.1.8 vrf vrf1501

Flags: * - Adjacencies learnt on non-active FHRP router
       + - Adjacencies synced via CFSoE
       # - Adjacencies Throttled for Glean
       CP - Added via L2RIB, Control plane Adjacencies
       PS - Added via L2RIB, Peer Sync
       RO - Re-Originated Peer Sync Entry
       D - Static Adjacencies attached to down interface

IP ARP Table
Total number of entries: 1
Address         Age       MAC Address     Interface       Flags
10.10.1.8       00:00:04  0000.8aa9.79d3  Vlan1001

switch(config)# sh ip route 10.10.1.8 vrf vrf1501
```

```
10.10.1.8/32, ubest/mbest: 1/0, attached
    *via 10.10.1.8, Vlan1001, [190/0], 00:00:14, hmm
```

- GARP の後、ホストは次のように VXLAN ネットワークのリーフに移動します。

```
switch(config)# sh ip route 10.10.1.8 vrf vrf1501

10.10.1.8/32, ubest/mbest: 1/0
    *via 2.2.2.5%default, [200/0], 00:00:23, bgp-200, internal, tag 200, segid:

11501 tunnelid: 0x2020205 encap: VXLAN
```

クラシック イーサネット/FabricPath から VXLAN への移行の構成

ワークロードをクラシック イーサネット/FabricPath から VXLAN に移行するには、次の手順を実行します。

（注） EX/FX/FX2 プラットフォームの **show hardware access-list tcam region** コマンドを使用して、PACL リージョンが切り分けられたかどうかを確認します。そうでない場合は、ワークロードの移行を構成する前に、PACL リージョンが分割されて使用可能になっていることを確認してください。

手順

ステップ 1 VXLAN とクラシック イーサネット/FabricPath ネットワークの間にレイヤ 2 相互接続があることを確認します。表 4 : レイヤ 2 インターコネクトの組み合わせ （202 ページ） で指定されているように、これは VXLAN ボーダー リーフ（vPC 設定の有無にかかわらず）とクラシック イーサネット/FabricPath エッジ スイッチ（vPC 設定の有無にかかわらず）の間で行うことができます。このインターフェイスは、物理イーサネット レイヤ 2 ポートまたはレイヤ 2 ポート チャネルにすることができます。詳細については、VXLAN BGP EVPN の設定 （123 ページ）を参照してください。

ステップ 2 vPC VXLAN ボーダー リーフがある場合は、**peer-gateway** と **layer3 peer-router** コマンドが設定されていることを確認します。

ステップ 3 移行前の手順の一環として、HSRP の下で **mac-address** *address* {**ipv4** | **ipv6**} を使用して、クラシック イーサネット/FabricPath ネットワークの特定の VLAN に対して、エニーキャスト ゲートウェイ MAC アドレス（HSRP に VXLAN ファブリック）を構成します。

この事前移行手順を構成すると、GARP がトリガーされ、エニーキャスト ゲートウェイの MAC アドレスで VLAN 内のすべてのホストが更新されます。

ステップ 4 2 つのファブリックを接続するレイヤ 2 ポートに対して **port-type external** を使用して、VXLAN ボーダー リーフのポートを外部ポートとして設定します。

ステップ 5 移行する VLAN の SVI が、境界リーフを含むすべての VXLAN リーフで設定されていることを確認します。この手順は、VLAN にルーティングされたトラフィックがある場合に必要です。SVI をシャットダウン状態に保つようにしてください。

ステップ 6 VXLAN 境界リーフで、SVI が IPv4 および/または IPv6 BIA アドレスで設定されていることを確認します。

この構成は、クラシック イーサネット/FabricPath ネットワークへの外部インターフェイスを介してこの BIA IP アドレスを送信元 IP アドレスとして使用し、VDC-MAC を送信元 MAC として使用して、プロキシ ARP または ND 要求を送信できるようにするために必要です。この設定により、通常のゲートウェイ IP およびエニーキャスト ゲートウェイ MAC を使用しないようになります。この構成により、移行前の手順後の MAC の衝突が防止されます。

ステップ 7 IPv4 または IPv6 BIA アドレスは、VXLAN 境界リーフの SVI の送信元アドレスと同じサブネットにある必要があります。

ステップ 8 ボーダー リーフを含む VXLAN のすべてのリーフで **no shut svi** コマンドを実行します。

この構成で、VLAN 上のワークロードがクラシック イーサネット/FabricPath から VXLAN に移動すると、VXLAN 分散エニーキャスト ゲートウェイ（DAG）パラダイムに従ってソース VXLAN リーフ上でルーティングされます。

ステップ 9 クラシック イーサネット/FabricPath 側に存在し続ける VLAN のホストは、HSRP ゲートウェイでルーティングされます。これにより、DAG と HSRP の両方が共存し、VLAN に対して機能します。

ステップ 10 特定の VLAN のすべてのホストをクラシック イーサネット/FabricPath から VXLAN に移動します。

ステップ 11 他のアドレス ファミリを移行する前に、1 つのアドレス ファミリ（IPv4 または IPv6）のすべてのホストが完全に移行されていることを確認します。

ステップ 12 VLAN のすべてのホストがクラシック イーサネット/FabricPath から VXLAN に移動したら、HSRP ゲートウェイ SVI を VLAN のクラシック イーサネット/FabricPath 側から削除できます。

ステップ 13 すべての VLAN が両方のアドレス ファミリ（IPv4 および IPv6）のクラシック イーサネット/FabricPath から VXLAN に移行されたら、2 つのファブリックを接続するレイヤ 2 インターフェイスで **no port-type external** コマンドを実行します。BIA アドレスは不要になり、ボーダー リーフの SVI から削除できます。

移行が完了します。

移行用に境界リーフ上の外部ポートを設定する

アプリケーションまたはワークロードを従来のイーサネット/FabricPath から VXLAN に移行するには、境界リーフのポートをレイヤ 2 相互接続用の外部ポートとして構成する必要があります。

始める前に

VLAN 内のホストを従来のイーサネット/FabricPath から VXLAN に移行する場合は、FabricPath 側で VLAN の事前移行手順を完了してください。このために、VLAN の従来のイーサネット/FabricPath ネットワークの HSRP に AnyCast ゲートウェイの MAC アドレスを構成します。

手順の概要

1. **configure terminal**
2. **interface port-channel** *number*
3. **port-type external**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		コンフィギュレーション モードに入ります。

ステップ 2	**interface port-channel** *number*
	例：
	`switch(config)# interface port-channel 40`
	`switch(config-if)#`
		コンフィギュレーション モードを開始し、ポート チャネル インターフェイスを設定します。

ステップ 3	**port-type external**
	例：
	`switch(config-if)# port-type external`
	`switch(config-if)#`
		インターフェイスを、従来のイーサネット/FabricPath ネットワークに接続する外部インターフェイスとして構成します。

次のタスク

手順で説明したように、VLAN ホストが従来のイーサネット/FabricPath から VXLAN に移動する SVI で、IPv4 または IPv6 の BIA アドレスを構成する必要があります。この構成については、移行用の外部 IP アドレスの構成 （207 ページ） を参照してください。

移行用の外部 IP アドレスの構成

手順の概要

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **vrf member** *vrf-name*
4. **ip address** *address netmask*
5. **ip address** *address netmask* **secondary use-bia**
6. **ipv6 address** *address netmask*
7. **ipv6 address** *address netmask* **use-bia**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		コンフィギュレーション モードに入ります。

ステップ 2	**interface vlan** *vlan-id*
	例：
	`switch(config)# interface vlan 1100`
	`switch(config-if)#`
		VLAN インターフェイスを作成し、インターフェイス構成モードを開始します。

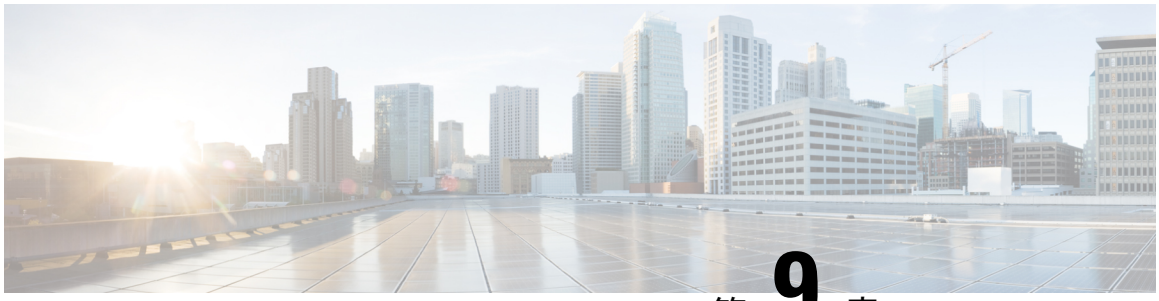
ステップ 3	**vrf member** *vrf-name*
	例：
	`switch(config-if)# vrf member vrf50`
	`switch(config-if)#`
		このインターフェイスを VRF に追加します。

ステップ 4	**ip address** *address netmask*
	例：
	`switch(config-if)# ip address 192.168.1.1/24`
	`switch(config-if)#`
		インターフェイスに IPv4 アドレスを割り当てます。

ステップ 5	**ip address** *address netmask* **secondary use-bia**
	例：
	`switch(config-if)# ip address 192.168.1.10/24 secondary use-bia`
	`switch(config-if)#`
		外部 IPv4 アドレスを設定します。

ステップ 6	**ipv6 address** *address netmask*
	例：
	`switch(config-if)# ipv6 address 2001:DB8:1::1/64`
	`switch(config-if)#`
		インターフェイスに IPv6 アドレスを割り当てます。

ステップ 7	**ipv6 address** *address netmask* **use-bia**
	例：
	`switch(config-if)# ip address 2001:DB8:1::10/64 use-bia`
	`switch(config-if)#`
		外部 IPv6 アドレスを設定します。



第 9 章

vPC マルチホーミングの構成

この章は、次の内容で構成されています。

プライマリ IP アドレスのアドバタイズ

vPC 対応リーフまたはボーダー リーフ スイッチでは、デフォルトで、すべてのレイヤ 3 ルートがリーフ スイッチ VTEP のセカンダリ IP アドレス（VIP）を BGP ネクスト ホップ IP アドレスとしてアドバタイズされます。プレフィックス ルートとリーフ スイッチで生成されたルートは、vPC リーフ スイッチ間で同期されません。これらのタイプのルートの BGP ネクスト ホップとして VIP を使用すると、トラフィックが誤った vPC リーフまたはボーダー リーフ スイッチに転送され、ブラック ホールになる可能性があります。vPC 対応リーフまたはボーダー リーフ スイッチで BGP のプレフィックス ルートまたはループバック インターフェイス ルートをアドバタイズするときにネクスト ホップとしてプライマリ IP アドレス（PIP）を使用するようにプロビジョニングすると、これらのタイプのアドバタイズ時に、BGP ネクスト ホップとして PIP を選択できます。これにより、トラフィックは常に正しい vPC 対応リーフまたはボーダー リーフ スイッチに転送されます。

PIP をアドバタイズするための設定コマンドは **advertise-pip** です。

以下に設定サンプルを示します。

```
switch(config)# router bgp 65536
  address-family 12vpn evpn
    advertise-pip
interface nve 1
    advertise virtual-rmac
```

advertise-pip コマンドは、外部的に学習したルートをアドバタイズするときに、または vPC が有効な場合に再配布される直接ルートのため、BGP がネクストホップとして PIP を使用するようにします。

VIP で VMAC（仮想 MAC）が使用され、VIP/PIP 機能が有効になっている場合は、システム MAC が PIP で使用されます。

advertise-pip および **advertise virtual-rmac** コマンドをイネーブルにすると、タイプ 5 ルートは PIP でアドバタイズされ、タイプ 2 ルートは引き続き VIP でアドバタイズされます。さらに、VMAC は VIP で使用され、システム MAC は PIP で使用されます。

（注） この機能を正しく動作させるには、**advertise-pip** および **advertise-virtual-rmac** コマンドを同時に有効または無効にする必要があります。一方を有効または無効にすると、無効な設定と見なされます。

vPC セットアップでの BorderPE スイッチ

2つの BorderPE スイッチは vPC として設定されます。VXLAN vPC 展開では、共通の仮想 VTEP IP アドレス（セカンダリ ループバック IP アドレス）が通信に使用されます。共通の仮想 VTEP は、システム固有のルータ MAC アドレスを使用します。ボーダー PE スイッチからのレイヤ 3 プレフィックスまたはデフォルト ルートは、この共通の仮想 VTEP IP（セカンダリ IP）とシステム固有のルータ MAC アドレスをネクスト ホップとしてアドバタイズされます。

advertise-pip および **advertise virtual-rmac** コマンドを入力すると、レイヤ 3 プレフィックスまたはデフォルトがプライマリ IP およびシステム固有のルータ MAC アドレスでアドバタイズされ、MAC アドレスがセカンダリ IP でアドバタイズされ、ルータの MAC アドレスがセカンダリ IP アドレスから取得されます。

vPC セットアップでの DHCP 設定

DHCP または DHCPv6 リレー機能が vPC 設定のリーフスイッチで設定され、DHCP サーバがデフォルト以外の非管理 VRF にある場合は、vPC リーフスイッチで **advertise-pip** コマンドを設定します。これにより、BGP EVPN は VTEP インターフェイスのプライマリ IP アドレスを使用して、ネクスト ホップでルート タイプ 5 のルートをアドバタイズできます。

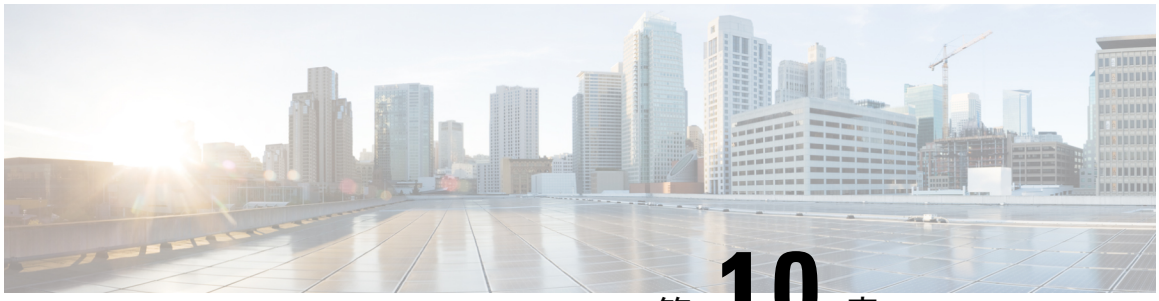
以下に設定例を示します。

```
switch(config)# router bgp 100
  address-family 12vpn evpn
    advertise-pip
interface nve 1
  advertise virtual-rmac
```

vPC セットアップでの IP プレフィックス

BGP EVPN でアドバタイズできるレイヤ 3 ルートには 3 つのタイプがあります。その内容は次のとおりです。

- ローカル ホスト ルート：これらのルートは、接続されているサーバまたはホストから学習されます。
- プレフィックス ルート：これらのルートは、リーフ、ボーダー リーフ、およびボーダー スパイン スイッチで他のルーティング プロトコルを介して学習されます。
- リーフ スイッチで生成されたルート：これらのルートには、インターフェイス ルートと静的ルートが含まれます。



第 10 章

vPC ファブリック ピアリングの設定

この章は、次の内容で構成されています。

vPC ファブリック ピアリングの詳細

vPC ファブリック ピアリング は、vPC ピア リンクの物理ポートを無駄にすることなく、拡張デュアル ホーミング アクセス ソリューションを提供します。この機能は、従来の vPC のすべての特性を保持します。

vPC ファブリック ピアリング ソリューションを次に示します。

- 仮想メンバー（トンネル）を含む vPC ファブリック ピアリング ポートチャネル。
- vPC ファブリック ピアリング （トンネル）、物理ピアリンク要件の削除。
- vPC ファブリック ピアリング アップ/ダウン イベントは、ルートの更新とファブリックのアップ/ダウンに基づいてトリガーされます。
- 拡張障害カバレッジのアップリンク トラッキング。
- vPC ファブリック ピアリング ルーティングされたネットワーク（スパインなど）を介した到達可能性。
- vPC コントロール プレーン over TCP-IP（CFSoIP）の復元力の向上。
- VXLAN トンネル上のデータ プレーン トラフィック。
- vPC メンバー スイッチ間の通信では、VXLAN カプセル化が使用されます。
- ノード上のすべてのアップリンクに障害が発生すると、そのスイッチの vPC ポートがダウンします。このシナリオでは、vPC ピアがプライマリ ロールを引き受け、トラフィックを転送します。

- vPC のステート依存性とアップ/ダウンシグナリングによるアップリンク トラッキング。
- ポジティブ アップリンク ステート トラッキングにより、vPC プライマリ ロールの選択が促進されます。
- ボーダー リーフおよびスパインの場合、ネットワーク通信はファブリックを使用するため、VRF 単位のピアリングは必要ありません。
- VIP/PIP 機能をタイプ 2 ルートに拡張することにより、孤立したホストへの転送を強化します。
- インフラ VLAN は、vPC ファブリック ピアリングには必要ありません。

（注） 1 つの VTEP としてカウントされる通常の vPC とは異なり、vPC ファブリック ピアリング は 3 つの VTEP としてカウントされます。

vPC ファブリック ピアリングの注意事項と制約事項

次に、vPC ファブリック ピアリングの注意事項と制限事項を示します。

- Cisco Nexus 9332C、9364C、および 9300-EX/FX/FXP/FX2/FX3/GX/GX2 プラットフォーム スイッチは、vPC ファブリック ピアリングをサポートします。Cisco Nexus 9200 および 9500 プラットフォーム スイッチは、vPC ファブリック ピアリングをサポートしていません。

（注） Cisco Nexus 9300-EX スイッチでは、混合モードのマルチキャストと入力レプリケーションはサポートされていません。VNI はマルチキャストまたは IR アンダーレイのいずれかで設定する必要があります。

- vPC ファブリック ピアリングでは、region ing-flow-redirect の TCAM カービングが必要です。TCAM カービングでは、機能を使用する前に設定を保存し、スイッチをリロードする必要があります。（この要件は、Cisco Nexus 9300-GX プラットフォーム スイッチには適用されません）。
- vPC ファブリック ピアリングの送信元および宛先 IP を再設定する前に、vPC ドメインをシャットダウンする必要があります。vPC ファブリック ピアリングの送信元と宛先の IP を調整したら、vPC ドメインを有効にできます（**no shutdown**）。
- **virtual peer-link destination** コマンドでサポートされる送信元および接続先 IP は、クラス A、B、および C です。クラス D および E は、vPC ファブリック ピアリングではサポートされません。

- vPC ファブリック ピアリング ピアリンクは、トランスポート ネットワーク（ファブリックのスパイン層）を介して確立されます。vPC ピア間の通信がこのように行われると、ポート ステート情報、VLAN 情報、VLAN-to-VNI マッピング、ホスト MAC アドレスの同期に使用されるコントロール プレーン情報 CFS メッセージがファブリック経由で送信されます。CFS メッセージは、トランスポート ネットワークで保護する必要がある適切な DSCP 値でマーキングされます。次の例は、Cisco Nexus 9000 シリーズ スイッチのスパイン レイヤでの QoS 設定の例を示しています。

  DSCP 値を照合してトラフィックを分類します（DSCP 56 がデフォルト値です）。

```
class-map type qos match-all CFS
  match dscp 56
```

  適切なスパイン スイッチの完全プライオリティ キューに対応する qos-group にトラフィックを設定します。この例では、スイッチは完全プライオリティ キュー（キュー 7）に対応する qos-group 7にトラフィックを送信します。異なる Cisco Nexus プラットフォームでは、キューイング構造が異なる場合があることに注意してください。

```
policy-map type qos CFS
  class CFS
    Set qos-group 7
```

  VTEP（ネットワークのリーフ層）に向かうすべてのインターフェイスに分類サービス ポリシーを割り当てます。

```
interface Ethernet 1/1
  service-policy type qos input CFS
```

- Cisco NX-OS リリース 10.1（1）以降、FEX サポートは Cisco Nexus 9300-EX/FX/FX2/FX3 プラットフォーム スイッチ IPv4 アンダーレイのために vMCT と一緒に提供されてます。
- Cisco NX-OS リリース 10.2（2）F 以降、この機能は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- vPC ファブリック ピアリング ドメインは、マルチサイト vPC BGW のロールではサポートされません。
- VIP/PIP 機能をタイプ 2 ルートに拡張して、孤立ホストへの転送を強化します。
- レイヤ 3 テナント ルーテッド マルチキャスト（TRM）はサポートされていません。レイヤ 2/レイヤ 3 TRM（混合モード）はサポートされていません。
- この機能でタイプ 5 ルートを使用する場合、この **advertise-pip** コマンドは必須設定です。
- vPC ポートの背後にある VTEP はサポートされません。これは、仮想ピアリンクピアが vPC ポートの背後にある VTEP の中継ノードとして機能できないことを意味します。
- SVI およびサブインターフェイス アップリンクはサポートされていません。
- 孤立したタイプ 2 ホストは、PIP を使用してアドバタイズされます。vPC タイプ 2 ホストは、VIP を使用してアドバタイズされます。これはタイプ 2 ホストのデフォルトの動作です。

vPC ファブリック ピアリングの設定

PIP を使用して孤立したタイプ 5 ルートをアドバタイズするには、BGP で PIP をアドバタイズする必要があります。

- リモート VTEP から孤立したホストへのトラフィックは、孤立した実際のノードに到達します。トラフィックのバウンスが回避されます。

（注） vPC レッグがダウンしている場合でも、vPC ホストは VIP IP でアドバタイズされます。

- 中断のない ISSU NX-OS ソフトウェアアップグレードは、vPC ファブリック ピアリング機能が設定されたスイッチではサポートされません。
- Cisco NX-OS リリース 10.2（F）以降、ND-ISSU と LXC-ISSU は Cisco Nexus 9300-EX/FX/FXP/FX2/FX3/GX/GX2 ToR スイッチ上の IPv4 アンダーレイのために vMCT と一緒にサポートされています。
- Cisco NX-OS リリース 10.3(2)F 以降、vPC ファブリック ピアリングは Cisco Nexus 9300-EX/FX/FXP/FX2/FX3/GX/GX2 ToR スイッチの IP6 アンダーレイに対してサポートされます。
- Cisco NX-OS リリース 10.3(2)F 以降、ND-ISSU と LXC-ISSU は Cisco Nexus 9300-EX/FX/FXP/FX2/FX3/GX/GX2 ToR スイッチ上の IPv6 アンダーレイのために vMCT と一緒にサポートされています。
- IPv6 アンダーレイの vMCT は、FEX の接続をサポートしていません。

vPC ファブリック ピアリングの設定

両方の vPC メンバー スイッチで vPC ファブリック ピアリング DSCP 値が一致していることを確認します。対応する QoS ポリシーが vPC ファブリック ピアリング DSCP マーキングと一致することを確認します。

vPC ファブリック ピアリング を通過する通信を必要とするすべての VLAN は、VXLAN を有効にする必要があります（vn-segment）。これにはネイティブ VLAN が含まれます。

（注） MSTP では、ピアリンクと vPC レッグにデフォルトのネイティブ VLAN 設定がある場合、VLAN 1 は vPC ファブリック ピアリング全体に拡張する必要があります。この動作は、VLAN 1 を VXLAN（vn-segment）経由で拡張することで実現できます。ピアリンクおよび vPC レッグにデフォルト以外のネイティブ VLAN がある場合は、VLAN を VXLAN（vn-segment）に関連付けることによって、それらの VLAN を vPC ファブリック ピアリング全体に拡張する必要があります。

vPC ファブリック ピアリングの設定

show vpc virtual-peerlink vlan consistency コマンドを使用して、vPC ファブリック ピアリングに使用する既存の VLAN-to-VXLAN マッピングを確認します。

vPC ファブリック ピアリングの **peer-keepalive** コマンドは、次のいずれかの設定でサポートされます。

- 管理インターフェイス
- デフォルトまたは非デフォルト VRF の専用レイヤ 3 リンク
- スパイン経由で到達可能なループバック インターフェイス。

機能の設定

例では、アンダーレイ ルーティング プロトコルとして OSPF を使用しています。

```
configure terminal
nv overlay evpn
feature ospf
feature bgp
feature pim
feature interface-vlan
feature vn-segment-vlan-based
feature vpc

feature nv overlay
```

vPC の設定

（注） vPC ファブリック ピアリング 送信元または宛先 IP を変更するには、変更前に vPC ドメインをシャットダウンする必要があります。vPC ドメインは、**no shutdown** コマンドを使用して変更後に動作に戻すことができます。

TCAM カービングの設定

```
hardware access-list tcam region ing-racl 0
hardware access-list tcam region ing-sup 768
hardware access-list tcam region ing-flow-redirect 512
```

（注）
- ファブリック vPC ピアリングを設定する場合、Ingress-Flow-redirect TCAM リージョン サイズの最小サイズは 512 です。また、TCAM リージョン サイズが常に 512 の倍数で構成されていることを確認します。
- TCAM カービングは、Cisco Nexus 9300-GX/GX2/H2R/H1 プラットフォーム スイッチでサポートされません。
- TCAM カービングを有効にするには、スイッチのリロードが必要です。

vPC ドメインの設定

vPC ファブリック ピアリングの設定

インターネット ユーザに商品やサービスを提供する IPv4

```
vpc domain 100
peer-keepalive destination 192.0.2.1
virtual peer-link destination 192.0.2.100 source 192.0.2.20/32 [dscp <dscp-value>]
Warning: Appropriate TCAM carving must be configured for virtual peer-link vPC
peer-switch
peer-gateway
ip arp synchronize
ipv6 nd synchronize
exit
```

IPv6 の場合

```
vpc domain 100
peer-keepalive destination 192:0:2::1
virtual peer-link destination 192:0:2::100 source 192:0:2::20/32 [dscp <dscp-value>]
Warning: Appropriate TCAM carving must be configured for virtual peer-link vPC
peer-switch
peer-gateway
ipv6 arp synchronize
ipv6 nd synchronize
exit
```

（注） オプションの **dscp** キーワード。範囲は 1 ～ 63 です。デフォルト値は 56 です。

vPC ファブリック ピアリング ポート チャネルの設定

次のポート チャネルのメンバーを設定する必要はありません。

```
interface port-channel 10
switchport
switchport mode trunk
vpc peer-link

interface loopback0
```

（注） このループバックは、NVE 送信元インターフェイス ループバック（VTEP IP アドレスに使用されるインターフェイス）ではありません。

インターネット ユーザに商品やサービスを提供する IPv4

```
interface loopback 0
ip address 192.0.2.20/32
ip router ospf 1 area 0.0.0.0
```

IPv6 の場合

```
interface loopback 0
ipv6 address 192:0:2::20/32
ipv6 router ospfv3 1 area 0.0.0.0
```

（注） BGP ピアリングまたは専用ループバックにループバックを使用できます。このルックバックは、ピアのキープ アライブとは異なる必要があります。

アンダーレイ インターフェイスの設定

L3 物理チャネルと L3 ポート チャネルの両方がサポートされます。SVI およびサブインターフェイスはサポートされていません。

インターネット ユーザに商品やサービスを提供する IPv4

```
router ospf 1
interface Ethernet1/16
ip address 192.0.2.2/24
ip router ospf 1 area 0.0.0.0
no shutdown
interface Ethernet1/17
port-type fabric
ip address 192.0.2.3/24
ip router ospf 1 area 0.0.0.0
no shutdown
interface Ethernet1/40
port-type fabric
ip address 192.0.2.4/24
ip router ospf 1 area 0.0.0.0
no shutdown
interface Ethernet1/41
port-type fabric
ip address 192.0.2.5/24
ip router ospf 1 area 0.0.0.0
no shutdown
```

IPv6 の場合

```
router ospfv3 1
interface Ethernet1/16
ipv6 address 192:0:2::2/24
ipv6 router ospfv3 1 area 0.0.0.0
no shutdown
interface Ethernet1/17
port-type fabric
ipv6 address 192:0:2::3/24
ipv6 router ospfv3 1 area 0.0.0.0
no shutdown
interface Ethernet1/40
port-type fabric
ipv6 address 192:0:2::4/24
ipv6 router ospfv3 1 area 0.0.0.0
no shutdown
interface Ethernet1/41
port-type fabric
ipv6 address 192:0:2::5/24
ipv6 router ospfv3 1 area 0.0.0.0
no shutdown
```

（注） スパインに接続されるすべてのポートは、ポートタイプのファブリックである必要があります。

vPC ファブリック ピアリングの設定

VXLAN 設定

（注） **advertise virtual-rmac**（NVE）と **advertise-pip**（BGP）の設定は必須の手順です。詳細については、vPC マルチホーミングの構成 （209 ページ） の章を参照してください。

SVI および VLAN の設定

```
vlan 10
vn-segment 10010
vlan 101
vn-segment 10101
interface Vlan101
no shutdown
mtu 9216
vrf member vxlan-10101
no ip redirects
ip forward
ipv6 address use-link-local-only
no ipv6 redirects
interface vlan10
no shutdown
mtu 9216
vrf member vxlan-10101
no ip redirects
ip address 192.0.2.102/24
ipv6 address 2001:DB8:0:1::1/64
no ipv6 redirects
fabric forwarding mode anycast-gateway
```

仮想ポート チャネルの設定

```
interface Ethernet1/3
switchport
switchport mode trunk
channel-group 100
no shutdown
exit
interface Ethernet1/39
switchport
switchport mode trunk
channel-group 101
no shutdown
interface Ethernet1/46
switchport
switchport mode trunk
channel-group 102
no shutdown
interface port-channel100
vpc 100
interface port-channel101
vpc 101
interface port-channel102
vpc 102
exit
```

vPCから vPC ファブリック ピアリング への移行

この手順には、通常の vPC から vPC ファブリック ピアリング への移行手順が含まれています。

vPC ピア間の直接レイヤ 3 リンクは、ピアキープアライブにのみ使用する必要があります。このリンクは、vPC ファブリック ピアリング ループバックのパスをアドバタイズするために使用しないでください。

（注） この移行は中断を伴います。

始める前に

移行前に、vPC ピア間のすべての物理レイヤ 2 リンクをシャットダウンすることを推奨します。また、移行前または移行後に VLAN を vn-segment にマッピングすることを推奨します。

手順の概要

1. **configure terminal**
2. **show vpc**
3. **show port-channel summary**
4. **interface ethernet** *slot/port*
5. **no channel-group**
6. インターフェイスごとにステップ 4 と 5 を繰り返します。
7. **show running-config vpc**
8. **vpc domain** *domain-id*
9. **virtual peer-link destination** *dest-ip* **source** *source-ip*
10. **interface {ethernet | port-channel}** *value*
11. **port-type fabric**
12. （任意） **show vpc fabric-ports**
13. **virtual peer-link destination** *dest-ip* | *dest_ipv6* **source** *source-ip* | *source_ipv6* **dhcp** *dhcp_val*
14. **hardware access-list tcam region ing-flow-redirect** *tcam-size*
15. **copy running-config startup-config**
16. **reload**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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。

	コマンドまたはアクション	目的
	`switch# configure terminal`	
ステップ 2	**show vpc** 例： `switch(config)# show vpc`	ポート チャネルのメンバー数を決定します。
ステップ 3	**show port-channel summary** 例： `switch(config)# show port-channel summary`	メンバーの数を決定します。
ステップ 4	**interface ethernet** *slot/port* 例： `switch(config)# interface ethernet 1/4`	設定するインターフェイスを指定します。 （注） これは、ピアリンク ポート チャネルです。
ステップ 5	**no channel-group** 例： `switch(config-if)# no channel-group`	vPC ピアリンク ポート チャネル メンバーを削除します。 （注） このステップの後に中断が発生します。
ステップ 6	インターフェイスごとにステップ 4 と 5 を繰り返します。 例：	
ステップ 7	**show running-config vpc** 例： `switch(config-if)# show running-config vpc`	vPC ドメインを決定します。
ステップ 8	**vpc domain** *domain-id* 例： `switch(config-if)# vpc domain 100`	vPC ドメイン コンフィギュレーション モードを入力します。
ステップ 9	**virtual peer-link destination** *dest-ip* **source** *source-ip* 例： `switch(config-vpc-domain)# virtual peer-link destination 192.0.2.1 source 192.0.2.100`	vPC ファブリック ピアリングの宛先および送信元 IP アドレスを指定します。
ステップ 10	**interface {ethernet | port-channel}** *value* 例： `switch(config-if)# interface Ethernet1/17`	構成する L3 アンダーレイ インターフェイスを指定します。
ステップ 11	**port-type fabric** 例：	アンダーレイ インターフェイスのポート タイプ ファブリックを設定します。

	コマンドまたはアクション	目的
	`switch(config-if)# port-type fabric`	（注） スパインに接続されるすべてのポートは、ポートタイプのファブリックである必要があります。
ステップ 12	（任意） **show vpc fabric-ports** 例： `switch# show vpc fabric-ports`	スパインに接続されているファブリック ポートを表示します。

ステップ 13	**virtual peer-link destination** *dest-ip* | *dest_ipv6* **source** *source-ip* | *source_ipv6* **dhcp** *dhcp_val*
	例：
	インターネット ユーザに商品やサービスを提供する IPv4
	`switch(config-vpc-domain)# virtual peer-link destination 192.0.2.1 source 192.0.2.100 dhcp 56`
	例：
	IPv6 の場合
	`switch(config-vpc-domain)# virtual peer-link destination 6001:aaa::11 source 6001:aaa::22 dhcp 56`
		vPC ファブリック ピアリングの宛先および送信元 IPv4/IPv6 アドレスを指定します。
		（注） IPv4 vPC ファブリック ピアリング構成は IPv4 VXLAN アンダーレイでのみ機能し、IPv6 vPC ファブリック ピアリング構成は IPv6 VXLAN アンダーレイでのみ機能します。

ステップ 14	**hardware access-list tcam region ing-flow-redirect** *tcam-size* 例： `switch(config-vpc-domain)# hardware access-list tcam region ing-flow-redirect 512`	TCAM カービングを実行します。 入力フローリダイレクト TCAM リージョン サイズの最小サイズは 512 です。また、512 の倍数で構成されていることを確認します。
ステップ 15	**copy running-config startup-config** 例： `switch(config-vpc-domain)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。
ステップ 16	**reload** 例： `switch(config-vpc-domain)# reload`	スイッチをリブートします。

vPC ファブリック ピアリング 設定の確認

vPC ファブリック ピアリング設定のステータスを表示するには、次のコマンドを入力します。

表 *5: vPC* ファブリック ピアリング 検証コマンド

コマンド	目的
show vpc fabric-ports	ファブリック ポートの状態を表示します。
show vpc	vPC ファブリック ピアリング モードに関する情報を表示します。
show vpc virtual-peerlink vlan consistency	vn-segment に関連付けられていない VLAN を表示します。

show vpc fabric-ports コマンドの例

```
switch# show vpc fabric-ports
Number of Fabric port : 9
Number of Fabric port active : 9

Fabric Ports State
-------------------------------------
Ethernet1/9 UP
Ethernet1/19/1 ( port-channel151 ) UP
Ethernet1/19/2 ( port-channel151 ) UP
Ethernet1/19/3 UP
Ethernet1/19/4 UP
Ethernet1/20/1 UP
Ethernet1/20/2 ( port-channel152 ) UP
Ethernet1/20/3 ( port-channel152 ) UP
Ethernet1/20/4 ( port-channel152 ) UP
```

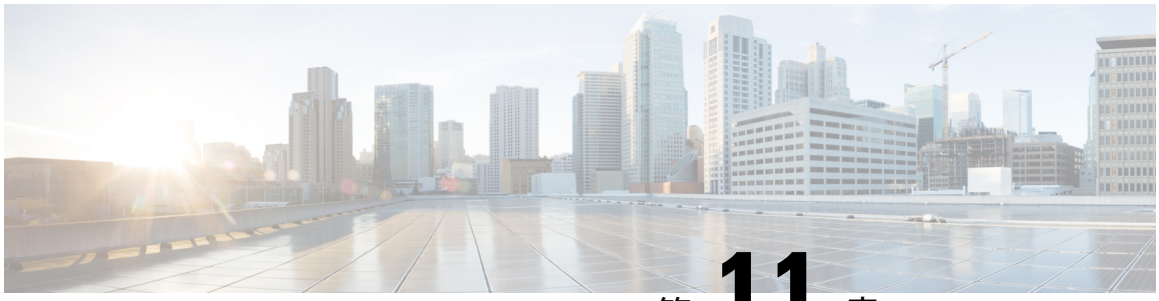
show vpc コマンドの例

```
switch# show vpc
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 3
Peer status                       : peer adjacency formed ok
vPC keep-alive status             : peer is alive
Configuration consistency status  : success
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : primary
Number of vPCs configured         : 1
Peer Gateway                      : Enabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Enabled, timer is off.(timeout = 240s)
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled
Virtual-peerlink mode             : Enabled

vPC Peer-link status
---------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ --------------------------------------------------
1     Po100  up     1,56,98-600,1001-3401,3500-3525
```

```
vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
101   Po101         up     success     success               98-99,1001-280
                                                             0

Please check "show vpc consistency-parameters vpc <vpc-num>" for the
consistency reason of down vpc and for type-2 consistency reasons for
any vpc.

ToR_B1#
```

show vpc virtual-peerlink vlan 整合性コマンドの例

```
switch# show vpc virtual-peerlink vlan consistency
Following vlans are inconsistent
23
switch#
```



第 11 章

ESI を使用した EVPN マルチホーミングとの相互運用性

この章は、次の内容で構成されています。

第 2 世代の Cisco Nexus 9000 スイッチ（EX モデル以降）は、EVPN マルチホーミングを完全にはサポートしていません。

（注） EVPN マルチホーミング機能の詳細については、「マルチホーミングの構成」の章を参照してください。

ただし、次のセクションで説明するように、Cisco Nexus 9000 スイッチは、EVPN マルチホーミング機能を完全にサポートするスイッチと同じ VXLAN EVPN ファブリックに統合できます。

ESI を使用した EVPN マルチホーミングとの相互運用性

Cisco NX-OS リリース 10.2(2)F以降、予約されていない ESI（0 または MAX-ESI）値と予約されている ESI（0 または MAX-ESI）値を持つ EVPN MAC/IP ルート（タイプ 2）は、転送（機能は通常 ESI RX と呼ばれます）のために評価されます。EVPN MAC/IP ルート解決の定義は、RFC 7432 Section 9.2.2 で定義されています。

EVPN MAC/IP ルート（タイプ 2）：

- 予約されている ESI 値（0 または MAX-ESI）は、MAC/IP ルート単独（タイプ 2 内の BGP ネクストホップ）によって単独で解決されます。
- 予約されていない ESI 値は、適合する ES イーサネット自動検出ルート（タイプ 1、ES EAD ごと）が存在する場合、単独で解決されます。

予約されていない ESI 値を使用した EVPN MAC/IP ルート解決は、Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチでサポートされます。

つまり、これらのスイッチは、ローカルに接続されたデバイスに vPC マルチホーミングを使用しながら（前の vPC マルチホーミングの構成 （209 ページ） およびvPC ファブリック ピアリングの設定 （213 ページ）セクションで説明したように）、ローカルデバイスの接続に EVPN マルチホーミングを使用する他のスイッチと VXLAN EVPN ファブリック内で共存できます。リモート エンドポイントの MAC アドレスと IP アドレスは、上記の EVPN コントロール プレーン メッセージを使用してこれらのリモート スイッチから学習され、複数のネクストホップ IP アドレス（EVPN マルチホーミングを実装する各スイッチを識別する一意の VTEP アドレス）が割り当てられます。

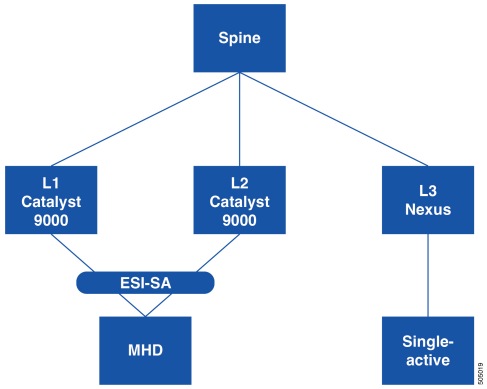
ESI を使用した EVPN マルチホーミングとの相互運用性に関する注意事項と制限事項

- Cisco Nexus-9300 スイッチは、ローカル デバイスへの EVPN マルチホーミング接続をサポートしていません（all-active モードと single-active モードの両方）。この機能は「ESI TX」と呼ばれます。
- Cisco NX-OS リリース 10.4(1)F まで、Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 スイッチおよび 9700-EX/FX/GX ライン カードを搭載した 9500 スイッチは、ESI Single-Active モードで ESI マルチホーミングをサポートするスイッチと共存できます。ただし、Single-active モードはサポートされていません。
- Cisco NX-OS リリース 10.4(1)F 以降、Single-active モードの ESI マルチホーミングをサポートするスイッチとの共存は、9700-EX/FX/GX ライン カードを搭載した Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 スイッチおよび 9500 スイッチに導入されます。
- Cisco NX-OS リリース 10.4(2)F 以降では、All-active モードと Single-active モードの両方で ESI マルチホーミングをサポートするスイッチとの共存は、Cisco Nexus 9332D-H2R および 93400LD-H1 スイッチでも使用できます。
- Cisco NX-OS リリース 10.4(3)F 以降では、All-active モードと Single-active モードの両方で ESI マルチホーミングをサポートするスイッチとの共存は、Cisco Nexus 9364C-H1 スイッチでも使用できます。
- リモート ノードとしての Cisco NX-OS デバイスは、ESI アクティブ ノードからの MAC ルートと、ESI アクティブ ノードとスタンバイ ノードの両方からの EAD-ES および EAD-EVI ルートを受け入れます。Cisco NX-OS デバイスは、これらのルートを使用して、特定のエンドポイントの MAC アドレスまたは IP アドレスのプライマリ パスとバックアップ パスを計算します。定常状態では、L2 トラフィックはプライマリパスを使用して転送されますが、プライマリで障害が発生した場合、トラフィックはバックアップパスに切り替えられます。

ESI を使用した EVPN マルチホーミングの例

EVPN ルート タイプの例

図 *16 : ESI* シングルアクティブ マルチホーミング



このトポロジでは、リーフ 3 は、ローカル デバイスへの ESI マルチホーミング接続をサポートする Cat9k（リーフ 1、リーフ 2）デバイスへのリモート VTEP として機能する Cisco Nexus 9000 デバイスです。このアプリには次の機能があります。

- ESI アクティブ ノードからの MAC、EAD per ES、EAD per EVI ルート、および ESI スタンバイ ノードからの EAD per ES、EAD per EVI ルートを受け入れます。
- ES ルートごとに EAD で設定されたフラグに基づいて、ESI がシングルアクティブかどうかを定義します。
- ES ごとの EAD および EVI ごとの EAD でいくつのノードから受信したかに基づいて、ESI シングルアクティブが双方向接続か n 方向接続かを定義します。

次に、BGP L2 EVPN Route-Type-1（EAD/ES または EAD/EVI）のリーフ 3 デバイスからの出力例を示します。Cisco Nexus 9000 ノードの EVPN アドレスファミリで **maximum-path** を構成する必要があります。これにより、BGP は、ES ごとの EAD、EVI ごとの EAD ルートのベストパスまたはマルチパスとしてすべてのパスを選択し、すべてのネクストホップを L2RIB にダウンロードできます。

```
show bgp l2vpn evpn route-type 1
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 51.51.51.51:3907 (EAD-ES [03de.affe.ed00.0b00.0000 3907])
BGP routing table entry for [1]:[03de.affe.ed00.0b00.0000]:[0xffffffff]/152, version 71
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn

Advertised path-id 1
Path type: local, path is valid, is best path, no labeled nexthop, has esi_gw
AS-Path: NONE, path locally originated
51.51.51.51 (metric 0) from 0.0.0.0 (51.51.51.51)
Origin IGP, MED not set, localpref 100, weight 32768
Received label 0
Extcommunity: RT:12000:1000002 RT:12000:1000003 RT:12000:1000012
```

```
RT:12000:1000013 ENCAP:8 ESI:1:000000

Path-id 1 advertised to peers:
111.111.46.1 111.111.47.1
```

ESI:1:000000 –> 1 フィールドでは、値はモードを示します。1 は**シングルアクティブを表し**、0 は**オールアクティブ**を表します。

シングルアクティブ MAC エントリの例

次に、シングルアクティブ MAC エントリを表示するように拡張された MAC アドレス テーブル コマンドのリーフ 3 デバイスの出力例を示します。

単一のアクティブ ESI MAC エントリの場合、**ポート**値には 2 つの VTEP が表示され、**A** はアクティブ ESI パスを表し、**S** はスタンバイ ESI パスを表します。

例：nve1（**A**:11.11.11.11 **S**:22.22.22.22）

```
switch# show mac address-table
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link,
        (T) - True, (F) - False, C - ControlPlane MAC, ~ - vsan,
        (NA)- Not Applicable, A – Active ESI Path, S – Standby ESI Path
  VLAN     MAC Address      Type    age  Secure NTFY Ports
-------+-----------------+-------+-----+------+----+------------------
C  100    0000.6666.6661  dynamic  NA    F      F   nve1(A:11.11.11.11 S:22.22.22.22)
C  101    0000.6666.6662  dynamic  NA    F      F   nve1(A:11.11.11.11 S:22.22.22.22)
C  101    0000.6666.6663  dynamic  NA    F      F   nve1(A:11.11.11.11 S:22.22.22.22)
C  102    0000.6666.6664  dynamic  NA    F      F   nve1(A:22.22.22.22 S:11.11.11.11)
C  103    0000.6666.6665  dynamic  NA    F      F   nve1(33.33.33.33 44.44.44.44)
C  104    0000.6666.6666  dynamic  NA    F      F   nve1(33.33.33.33 44.44.44.44)
C  105    0000.6666.6667  dynamic  NA    F      F   nve1(33.33.33.33 44.44.44.44)
G    -    0091.f3e7.1b08  static   -     F      F   sup-eth1(R)
switch#
```

L2 ルート パス リストの例

次の例は、**show l2route evpn path-list all detail** コマンドのリーフ 3 デバイスからの出力例です。これは、以下で強調しているように、シングルアクティブ モード フラグとバックアップ ネクストホップの詳細をキャプチャするように拡張されています。

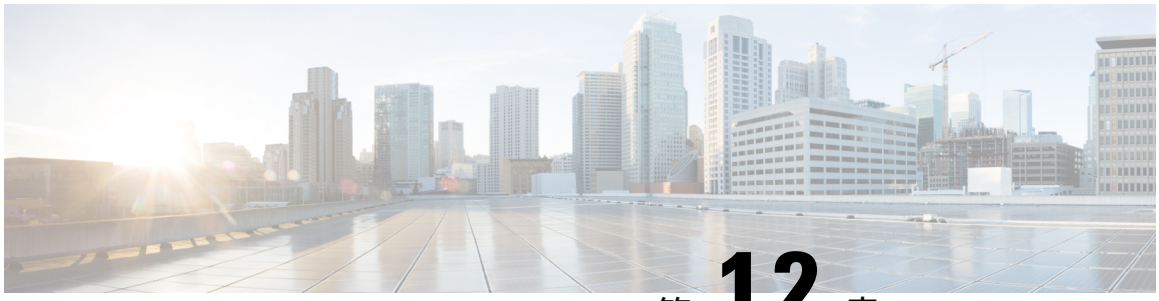
```
switch# S1# show l2route evpn path-list all detail
(R) = Remote Global EAD NH Peerid resolved,
(UR) = Remote Global EAD NH Peerid unresolved
Flags - (A):All-Active (Si):Single-Active

Topology ID  Prod   ESI                       ECMP Label Flags  Client Ctx  MACs  NFN
Bitmap
------------ ------ ------------------------- ---------- ------ ----------- ----------
------
1162         None   aaaa.aaaa.aaaa.aaaa.99aa  1          Si      0           1       8
                     CP Next-Hops:
                     Gbl EAD Next-Hops:  11.11.11.11(11,R), 22.22.22.22(22,R)
                     Res Next-Hops:  22.22.22.22
                     Bkp Next-Hops:  11.11.11.11
                     Res Next-Hops from UFDM:  22.22.22.22
                     Bkp Next-Hops from UFDM:  11.11.11.11
1162         UFDM   aaaa.aaaa.aaaa.aaaa.99aa  1          -      1493172225  0       2
                     CP Next-Hops:
```

```
Gbl EAD Next-Hops:
Res Next-Hops:  22.22.22.22
Bkp Next-Hops:  11.11.11.11
```

L2 ルート EVPN EAD の例

次の例は、**show l2route evpn ead all detail** コマンドの出力例です。これは、以下で強調しているように、シングルアクティブ モード フラグとバックアップ ネクストホップの詳細をキャプチャするように拡張されています。

```
switch# show l2route evpn ead all detail

Flags -(A):All-Active (Si):Single-Active (V):Virtual ESI (D):Del Pending(S):Stale

Topology ID   Prod   ESI                       NFN Bitmap  Num PLs Flags
------------ ------ ------------------------- ------------ ------ ------
1162          BGP    aaaa.aaaa.aaaa.aaaa.99aa   0           1      -
                Next-Hops: 11.11.11.11, 22.22.22.22
4294967294    BGP    aaaa.aaaa.aaaa.aaaa.99aa   0           1      Si
                Next-Hops: 11.11.11.11, 22.22.22.22
```



第 12 章

外部 VRF 接続とルート リークの設定

この章は、次の内容で構成されています。

外部 VRF 接続の設定

VXLAN BGP EVPN ファブリックの外部レイヤ 3 接続について

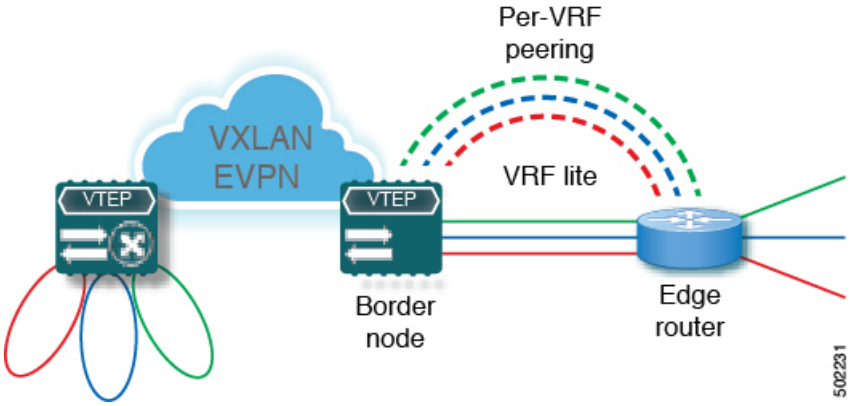
VXLAN BGP EVPN ファブリックは、外部接続を実現するために VRF 単位の IP ルーティングを使用して拡張できます。レイヤ 3 拡張に使用されるアプローチは一般に VRF Lite と呼ばれ、機能自体はより正確に Inter-AS オプション A またはバックツーバック VRF 接続として定義されます。

VXLAN BGP EVPN - VRF-lite brief

いくつかのポインタを次に示します。

- VXLAN BGP EVPN ファブリックを次の図の左側に示します。
- ファブリック内のルートは、すべてのエッジデバイス（VTEP）とルート リフレクタの間で交換されます。使用されるコントロール プレーンは、EVPN アドレス ファミリを持つ MP-BGP です。
- ボーダーノードとして機能するエッジ デバイス（VTEP）は、外部ルータ（ER）にプレフィックスを渡すように設定されます。これは、MP-BGP EVPN から IPv4/IPv6 VRF ピアリングにプレフィックスをエクスポートすることによって実現されます。
- VRF 単位のピアリングには、さまざまなルーティング プロトコルを使用できます。eBGP は最適なプロトコルですが、OSPF、IS-IS、EIGRPなどのIGP は活用できますが、再配布が必要です。

図 *17 : VRF-Lite* を使用したレイヤ *3* 外部接続



外部 VRF 接続とルート リークの注意事項と制約事項

VXLAN BGP EVPN ファブリックの外部レイヤ 3 接続には、次のガイドラインと制限事項が適用されます。

- Cisco Nexus 96136YC-R および 9636C-RX ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチのサポートが追加されました。
- 物理レイヤ 3 インターフェイス（親インターフェイス）は、外部レイヤ 3 接続（つまり、VRF デフォルト）に使用できます。
- 複数のサブインターフェイスへの親インターフェイスは、外部レイヤ 3 接続（つまり、VRFデフォルトの Ethernet1/1）には使用できません。代わりにサブインターフェイスを使用できます。
- Cisco NX-OS Release 9.3(5) 以降では、サブインターフェイスが設定されている場合、VTEP は親インターフェイス上で VXLAN カプセル化トラフィックをサポートします。
- VTEP は、VRF 参加または IEEE 802.1Q カプセル化に関係なく、サブインターフェイスを介した VXLAN カプセル化トラフィックをサポートしません。
- VXLAN VLAN と非 VXLAN VLAN のサブインターフェイスの混在はサポートされていません。
- address-family ipv4 unicast で適用される **import map** コマンドは、EVPN テーブル L3VNI の対応物に何がインポートされるかを制御しません。
- TRM が構成されている場合は、外部ルータへのインターコネクトに SVI を使用しないでください。

VRF-Lite 用 eBGP を使用した VXLAN BGP EVPN の設定

BGP を使用した VXLAN ルーティングおよび外部接続用の VRF の設定

ボーダー ノードで VRF を設定します。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **vni** *number*
4. **rd** {**auto** | *rd*}
5. **address-family** {**ipv4** | **ipv6**} **unicast**
6. **route-target both** {**auto** | *rt*}
7. **route-target both** {**auto** | *rt*} **evpn**
8. すべての L3VNI に対してステップ 1ー7 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vrf context** *vrf-name*	VRF を設定します。
ステップ 3	**vni** *number*	VNI を指定します。VRF に関連付けられた VNI は、多くの場合、レイヤ 3 VNI、L3VNI、または L3VPN と呼ばれます。L3VNI は、参加する VTEP 間で共通の識別子として設定されます。
ステップ 4	**rd** {**auto**	*rd*}	VRF のルート識別子（RD）を指定します。RD は、L3VNI 内の VTEP を一意に識別します。RD を入力する場合は、以下の形式がサポートされています。ASN2:NN、ASN4:NN、または IPV4:NN。
ステップ 5	**address-family** {**ipv4**	**ipv6**} **unicast**	IPv4 または IPv6 ユニ キャスト アドレス ファミリを設定します。
ステップ 6	**route-target both** {**auto**	*rt*}	IPv4 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 7	**route-target both** {**auto**	*rt*} **evpn**	IPv4 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形

	コマンドまたはアクション	目的
		式がサポートされます。非対称VNIをサポートするには、手動で設定された RT が必要です。
ステップ 8	すべての L3VNI に対してステップ 1ー7 を繰り返します。	

ボーダー ノードでの L3VNI のファブリック側 VLAN および SVI の設定

手順の概要

1. **configure terminal**
2. **vlan** *number*
3. **vn-segment** *number*
4. **interface** *vlan-number*
5. **mtu** *value*
6. **vrf member** *vrf-name*
7. **ip forward**
8. **no ip redirects**
9. **ipv6** *ip-address*
10. **no ipv6 redirects**
11. すべての L3VNI に対してステップ 2ー10 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	コンフィギュレーション モードを入力します。
ステップ 2	**vlan** *number*	L3VNI に使用される VLAN ID を指定します。
ステップ 3	**vn-segment** *number*	L3VNI を VXLAN EVPN ルーティング用の VLAN にマッピングします。
ステップ 4	**interface** *vlan-number*	VXLAN EVPN ルーティングの SVI（スイッチ仮想インターフェイス）を指定します。
ステップ 5	**mtu** *value*	L3VNI の MTU を指定します。
ステップ 6	**vrf member** *vrf-name*	一致する VRF コンテキストに SVI をマッピングします。
ステップ 7	**ip forward**	L3VNI の IPv4 転送を有効にします。
ステップ 8	**no ip redirects**	ICMP リダイレクトを無効化します。

	コマンドまたはアクション	目的
ステップ 9	**ipv6** *ip-address*	L3VNI の IPv6 転送を有効にします。
ステップ 10	**no ipv6 redirects**	ICMPv6 リダイレクトを無効化します。
ステップ 11	すべての L3VNI に対してステップ 2ー10 を繰り返します。	

ボーダー ノードでの VTEP の設定

手順の概要

1. **configure terminal**
2. **interface nve1**
3. **member vni** *vni* **associate-vrf**
4.

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**interface nve1**	NVE インターフェイスを設定します。
ステップ 3	**member vni** *vni* **associate-vrf**	レイヤ 3 VNI を、テナント VRF ごとに 1 つずつ、オーバーレイに追加します。
ステップ 4		すべての L3VNI に対してステップ 3 を繰り返します。

IPv4 VRF ごとのピアリングのためのボーダー ノードでの BGP VRF インスタンスの設定

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **advertise l2vpn evpn**
6. **maximum-paths ibgp** *number*
7. **maximum-paths** *number*
8. **neighbor** *address* **remote-as** *number*
9. **update-source** *type/id*

10. **address-family ipv4 unicast**
11. IPv4 の外部接続を必要とするすべての L3VNI に対して、ステップ 3ー10 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP を設定します。*autonomous-system-number*の範囲は 1ー4294967295 です。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 のアドレス ファミリを設定します。
ステップ 5	**advertise l2vpn evpn**	IPv4 アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。
ステップ 6	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。*1ー64* の数値の範囲。デフォルトは 1 です。
ステップ 7	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 8	**neighbor** *address* **remote-as** *number*	eBGP ネイバーの IPv4 アドレスおよびリモート自律システム（AS）番号を定義します。
ステップ 9	**update-source** *type/id*	eBGP ピアリングのインターフェイスを定義します。
ステップ 10	**address-family ipv4 unicast**	IPv4 プレフィックス交換の IPv4 アドレス ファミリをアクティブにします。
ステップ 11	IPv4 の外部接続を必要とするすべての L3VNI に対して、ステップ 3ー10 を繰り返します。	

IPv6 VRF ごとのピアリングのためのボーダー ノードでの BGP VRF インスタンスの設定

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*

4. **address-family ipv6 unicast**
5. **advertise l2vpn evpn**
6. **maximum-paths ibgp** *number*
7. **maximum-paths** *number*
8. **neighbor** *address* **remote-as** *number*
9. **update-source** *type/id*
10. **address-family ipv6 unicast**
11. IPv6 の外部接続を必要とするすべての L3VNI に対して、ステップ 3ーステップ 10 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv6 unicast**	IPv4 のアドレス ファミリを設定します。
ステップ 5	**advertise l2vpn evpn**	IPv6 アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。
ステップ 6	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 7	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 8	**neighbor** *address* **remote-as** *number*	eBGP ネイバーの IPv6 アドレスおよびリモート自律システム（AS）番号を定義します。
ステップ 9	**update-source** *type/id*	eBGP ピアリングのインターフェイスを定義します。
ステップ 10	**address-family ipv6 unicast**	IPv6 のアドレス ファミリを設定します。
ステップ 11	IPv6 の外部接続を必要とするすべての L3VNI に対して、ステップ 3ーステップ 10 を繰り返します。	

VRFごとのピアリングのボーダーノードでのサブインターフェイスインスタンスの設定-バージョン1

手順の概要

1. **configure terminal**
2. interface *type/id*
3. **no switchport**
4. **no shutdown**
5. **exit**
6. interface *type/id*
7. **encapsulation dot1q** *number*
8. **vrf member** *vrf-name*
9. **ip address** *address*
10. **no shutdown**
11. VRF 単位のピアリングごとに、ステップ 5ー9 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	interface *type/id*	親インターフェイスを設定します。
ステップ 3	**no switchport**	インターフェイスでレイヤ 2 スイッチング モードを無効にします。
ステップ 4	**no shutdown**	親インターフェイスを起動します。
ステップ 5	**exit**	インターフェイス コンフィギュレーション モードを終了します。
ステップ 6	interface *type/id*	サブインターフェイスインスタンスを定義します。
ステップ 7	**encapsulation dot1q** *number*	サブインターフェイスの VLAN ID を設定します。*number* 引数には、1ー3967の値を指定できます。
ステップ 8	**vrf member** *vrf-name*	一致する VRF コンテキストにサブインターフェイスをマッピングします。
ステップ 9	**ip address** *address*	サブインターフェイスに IP アドレスを設定する。
ステップ 10	**no shutdown**	サブインターフェイスを起動します。
ステップ 11	VRF 単位のピアリングごとに、ステップ 5ー9 を繰り返します。	

VXLAN BGP EVPN - デフォルト接続、外部接続のルート フィルタリング

外部接続のデフォルト ルーティングの設定について

VXLAN BGP EVPN ファブリックへのデフォルト ルート アドバタイズメントでは、ファブリックにアドバタイズされるデフォルト ルートがファブリックの外部に同時にアドバタイズされないようにする必要があります。この場合、このような事態を防ぐルート フィルタリングが必要です。

ボーダー ノード VRF でのデフォルト ルートの設定

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **ip route 0.0.0.0/0** *next-hop*
4. **ipv6 route 0::/0** *next-hop*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vrf context** *vrf-name*	VRF を設定します。
ステップ 3	**ip route 0.0.0.0/0** *next-hop*	IPv4 デフォルト ルートを設定します。
ステップ 4	**ipv6 route 0::/0** *next-hop*	IPv6 デフォルト ルートを設定します。

IPv4/IPv6 デフォルト ルート アドバタイズメントのボーダー ノードでの BGP VRF インスタンスの設定

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **network 0.0.0.0/0**
6. **address-family ipv6 unicast**
7. **network 0::/0**
8. **neighbor** *address***remote-as** *number*
9. **update-source** *type/id*
10. **address-family {ipv4 | ipv6} unicast**

11. **route-map** *name* **out**
12. デフォルト ルート フィルタリングによる外部接続を必要とするすべての L3VNI に対して、ステップ 3 ーステップ 11 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 ユニ キャスト アドレス ファミリを設定します。IPv4 アンダーレイを使用した IPv6 over VXLAN に必要です。
ステップ 5	**network 0.0.0.0/0**	IPv4 デフォルト ルート ネットワーク ステートメントを作成しています。
ステップ 6	**address-family ipv6 unicast**	IPv6 ユニ キャスト アドレス ファミリを設定します。
ステップ 7	**network 0::/0**	IPv6 デフォルト ルート ネットワーク ステートメントを作成しています。
ステップ 8	**neighbor** *address***remote-as** *number*	eBGP ネイバーの IPv4 アドレスおよびリモート自律システム（AS）番号を定義します。
ステップ 9	**update-source** *type/id*	eBGP ピアリングのインターフェイスを定義する
ステップ 10	**address-family {ipv4 | ipv6} unicast**	IPv4/IPv6 プレフィックス交換の IPv4 または IPv6 アドレス ファミリをアクティブにします。
ステップ 11	**route-map** *name* **out**	出力ルート フィルタリング用のルート マップを付加します。
ステップ 12	デフォルト ルート フィルタリングによる外部接続を必要とするすべての L3VNI に対して、ステップ 3 ーステップ 11 を繰り返します。	

IPv4 デフォルト ルート アドバタイズメントのルート フィルタリングの設定

IPv4 デフォルト ルート アドバタイズメントのルート フィルタリングを設定できます。

手順の概要

1. **configure terminal**
2. **ip prefix-list** *name* **seq 5 permit 0.0.0.0/0**
3. **route-map** *name* **deny 10**
4. **match ip address prefix-list** *name*
5. **route-map** *name* **permit 1000**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル構成モードを開始します。
ステップ 2	**ip prefix-list** *name* **seq 5 permit 0.0.0.0/0**	デフォルトルートフィルタリングの IPv4 プレフィックス リストを設定します。
ステップ 3	**route-map** *name* **deny 10**	外部接続を介してアドバタイズされるデフォルトルートを防止するために、先行する deny ステートメントを使用してルートマップを作成します。
ステップ 4	**match ip address prefix-list** *name*	default-route を含む IPv4 プレフィックスリストと照合します。
ステップ 5	**route-map** *name* **permit 1000**	外部接続を介して一致しないルートをアドバタイズする末尾の allow ステートメントを使用してルートマップを作成します。

IPv6 デフォルト ルート アドバタイズメントのルート フィルタリングの設定

手順の概要

1. **configure terminal**
2. **ipv6 prefix-list** *name* **seq 5 permit 0::/0**
3. **route-map** *name* **deny 10**
4. **match ipv6 address prefix-list** *name*
5. **route-map** *name* **permit 1000**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル構成モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**ipv6 prefix-list** *name* **seq 5 permit 0::/0**	デフォルトルートフィルタリングのIPv6プレフィックス リストを設定します。
ステップ 3	**route-map** *name* **deny 10**	外部接続を介してアドバタイズされるデフォルトルートを防止するために、先行するdenyステートメントを使用してルートマップを作成します。
ステップ 4	**match ipv6 address prefix-list** *name*	default-routeを含むIPv6プレフィックス リストと照合します。
ステップ 5	**route-map** *name* **permit 1000**	外部接続を介して一致しないルートをアドバタイズする末尾のallowステートメントを使用してルートマップを作成します。

デフォルト ルート配布およびホスト ルート フィルタの設定について

デフォルトでは、VXLAN BGP EVPN ファブリックは外部接続を介してすべての既知のルートを常にアドバタイズします。すべての状況でIPv4/32またはIPv6/128のホストルートをアドバタイズすることは有益ではないため、それぞれのルート フィルタリング アプローチが必要になることがあります。

IPv4/IPv6 ホスト ルート フィルタリングのためのボーダー ノードでの BGP VRF インスタンスの設定

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*
4. **neighbor** *address* **remote-as** *number*
5. **update-source** *type/id*
6. **address-family {ipv4 | ipv6} unicast**
7. **route-map** *name* **out**
8. ホストルートフィルタリングを使用した外部接続を必要とするすべてのL3VNIに対して、ステップ 3ー7 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP 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	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**neighbor** *address* **remote-as** *number*	eBGP ネイバーの IPv4/IPv6 アドレスとリモート自律システム（AS）番号を定義します。
ステップ 5	**update-source** *type/id*	eBGP ピアリングのインターフェイスを定義します。
ステップ 6	**address-family {ipv4 | ipv6} unicast**	IPv4/IPv6 プレフィックス交換の IPv4 または IPv6 アドレス ファミリをアクティブにします。
ステップ 7	**route-map** *name* **out**	出力ルート フィルタリング用のルート マップを付加します。
ステップ 8	ホスト ルート フィルタリングを使用した外部接続を必要とするすべての L3VNI に対して、ステップ 3 ー7 を繰り返します。	

IPv4 ホスト ルート アドバタイズメントのルート フィルタリングの設定

手順の概要

1. **configure terminal**
2. **ip prefix-list** *name* **seq 5 permit 0.0.0.0/0 eq 32**
3. **route-map** *name* **deny 10**
4. **match ip address prefix-list** *name*
5. **route-map** *name* **permit 1000**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル構成モードを開始します。
ステップ 2	**ip prefix-list** *name* **seq 5 permit 0.0.0.0/0 eq 32**	ホスト ルート フィルタリング用の IPv4 プレフィックス リストを設定します。
ステップ 3	**route-map** *name* **deny 10**	外部接続を介してアドバタイズされるデフォルト ルートを防止するために、先行する deny ステートメントを使用してルートマップを作成します。
ステップ 4	**match ip address prefix-list** *name*	host-route を含む IPv4 プレフィックス リストと照合します。

	コマンドまたはアクション	目的
ステップ 5	**route-map** *name* **permit 1000**	外部接続を介して一致しないルートをアドバタイズする末尾の allow ステートメントを使用してルートマップを作成します。

IPv6 ホスト ルート アドバタイズメントのルート フィルタリングの設定

手順の概要

1. **configure terminal**
2. **ipv6 prefix-list** *name* **seq 5 permit 0::/0 eq 128**
3. **route-map** *name* **deny 10**
4. **match ipv6 address prefix-list** *name*
5. **route-map** *name* **permit 1000**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル構成モードを開始します。
ステップ 2	**ipv6 prefix-list** *name* **seq 5 permit 0::/0 eq 128**	ホスト ルート フィルタリング用の IPv4 プレフィックス リストを設定します。
ステップ 3	**route-map** *name* **deny 10**	外部接続を介してアドバタイズされるデフォルトルートを防止するために、先行する deny ステートメントを使用してルートマップを作成します。
ステップ 4	**match ipv6 address prefix-list** *name*	host-route を含む IPv4 プレフィックス リストと照合します。
ステップ 5	**route-map** *name* **permit 1000**	外部接続を介して一致しないルートをアドバタイズする末尾の allow ステートメントを使用してルートマップを作成します。

例：VRF-Lite の eBGP を使用した VXLAN BGP EVPN の設定

VXLAN BGP EVPN から VRF-Lite を使用した外部ルータへの外部接続の例。

VXLAN BGP EVPN ボーダー ノードの設定

VXLAN BGP EVPN ボーダー ノードは、外部ルータのネイバー デバイスとして機能します。VRF 名は純粋にローカライズされており、外部ルータの VRF 名と異なる場合があります。重要な点は、L3VNI が VXLAN BGP EVPN ファブリック全体で一貫している必要があることです。読みやすくするために、VRF とインターフェイスの列挙が一貫して使用されます。

設定例は、IPv4 と IPv6 のデュアルスタック アプローチを表しています。 IPv4 または IPv6 は相互に置き換えることができます。

```
vrf context myvrf_50001
  vni 50001
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn
!
vlan 2000
  vn-segment 50001
!
interface Vlan2000
  no shutdown
  mtu 9216
  vrf member myvrf_50001
  no ip redirects
  ip forward
  ipv6 address use-link-local-only
  no ipv6 redirects
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 50001 associate-vrf
!
router bgp 65002
  vrf myvrf_50001
    router-id 10.2.0.6
    address-family ipv4 unicast
      advertise l2vpn evpn
      maximum-paths ibgp 2
      maximum-paths 2
    address-family ipv6 unicast
      advertise l2vpn evpn
      maximum-paths ibgp 2
      maximum-paths 2
    neighbor 10.31.95.95
      remote-as 65099
      address-family ipv4 unicast
    neighbor 2001::95/64
      remote-as 65099
      address-family ipv4 unicast
!
interface Ethernet1/3
  no switchport
  no shutdown
interface Ethernet1/3.2
  encapsulation dot1q 2
  vrf member myvrf_50001
  ip address 10.31.95.31/24
  ipv6 address 2001::31/64
  no shutdown
```

外部接続でのデフォルト ルート、ルート フィルタリングの設定

VXLAN BGP EVPN ボーダー ノードは、ファブリック内で IPv4 および IPv6 デフォルトルートをアドバタイズできます。VXLAN BGP EVPN ファブリックから外部ルータにホスト ルートを

アドバタイズすることが有益でない場合は、これらの IPv4/32 および IPv6/128 を外部接続ピアリング設定でフィルタリングできます。

```
ip prefix-list default-route seq 5 permit 0.0.0.0/0 le 1
ipv6 prefix-list default-route-v6 seq 5 permit 0::/0
!
ip prefix-list host-route seq 5 permit 0.0.0.0/0 eq 32
ipv6 prefix-list host-route-v6 seq 5 permit 0::/0 eq 128
!
route-map extcon-rmap-filter deny 10
  match ip address prefix-list default-route
route-map extcon-rmap-filter deny 20
  match ip address prefix-list host-route
route-map extcon-rmap-filter permit 1000
!
route-map extcon-rmap-filter-v6 deny 10
  match ipv6 address prefix-list default-route-v6
route-map extcon-rmap-filter-v6 deny 20
  match ip address prefix-list host-route-v6
route-map extcon-rmap-filter-v6 permit 1000
!
vrf context myvrf_50001
  ip route 0.0.0.0/0 10.31.95.95
  ipv6 route 0::/0 2001::95/64
!
router bgp 65002
  vrf myvrf_50001
    address-family ipv4 unicast
      network 0.0.0.0/0
    address-family ipv6 unicast
      network 0::/0


    neighbor 10.31.95.95
      remote-as 65099
      address-family ipv4 unicast
        route-map extcon-rmap-filter out
    neighbor 2001::95/64
      remote-as 65099
      address-family ipv4 unicast
        route-map extcon-rmap-filter-v6 out
```

外部ルータの設定

外部ルータは、VXLAN BGP EVPN ボーダー ノードのネイバー デバイスとして機能します。VRF 名は純粋にローカライズされており、VXLAN BGP EVPN ファブリックの VRF 名とは異なる場合があります。読みやすくするために、VRF とインターフェイスの列挙が一貫して使用されます。

設定例は、IPv4 と IPv6 のデュアルスタック アプローチを表しています。 IPv4 または IPv6 は相互に置き換えることができます。

```
vrf context myvrf_50001
!
router bgp 65099
  vrf myvrf_50001
    address-family ipv4 unicast
      maximum-paths 2
    address-family ipv6 unicast
      maximum-paths 2
    neighbor 10.31.95.31
      remote-as 65002
```

```
      address-family ipv4 unicast
    neighbor 2001::31/64
      remote-as 65002
      address-family ipv4 unicast
!
interface Ethernet1/3
  no switchport
  no shutdown
interface Ethernet1/3.2
  encapsulation dot1q 2
  vrf member myvrf_50001
  ip address 10.31.95.95/24
  Ipv6 address 2001::95/64
  no shutdown
```

VRF-Lite 用の OSPF を使用した VXLAN BGP EVPN の設定

OSPF を使用した VXLAN ルーティングおよび外部接続用の VRF の設定

OSPF VRF ごとのピアリング用に、ボーダー ノードで BGP VRF インスタンスを設定します。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **advertise l2vpn evpn**
6. **maximum-paths ibgp** *number*
7. **redistribute ospf** *name* **route-map** *name*
8. VRF 単位のピアリングごとに、ステップ 3ー7 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 アドレス ファミリを設定します。
ステップ 5	**advertise l2vpn evpn**	アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。
ステップ 6	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。

	コマンドまたはアクション	目的
ステップ 7	**redistribute ospf** *name* **route-map** *name*	OSPF から BGP への再配布を定義します。
ステップ 8	VRF 単位のピアリングごとに、ステップ 3ー7 を繰り返します。	

BGP から OSPF への再配布のルートマップの設定

手順の概要

1. **configure terminal**
2. **route-map** *name* **permit 10**
3. **match route-type internal**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**route-map** *name* **permit 10**	BGPからOSPFへの再配布のためのルートマップの作成
ステップ 3	**match route-type internal**	VXLAN BGP EVPN ファブリックで iBGP が使用されている場合は、再配布ルート マップで BGP 内部ルート タイプの一致を許可する必要があります。

VRF 単位のピアリングのためのボーダー ノードでの OSPF の設定

手順の概要

1. **configure terminal**
2. **router ospf** *instance*
3. **vrf** *vrf-name*
4. **redistribute bgp** *autonomous-system-number* **route-map** *name*
5. VRF 単位のピアリングごとに、ステップ 3ー4 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router ospf** *instance*	OSPF を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**redistribute bgp** *autonomous-system-number* **route-map** *name*	BGP から OSPF への再配布を定義します。
ステップ 5	VRF 単位のピアリングごとに、ステップ 3ー4 を繰り返します。	

VRFごとのピアリングのボーダーノードでのサブインターフェイスインスタンスの設定-バージョン2

手順の概要

1. **configure terminal**
2. **interface** *type/id*
3. **no switchport**
4. **no shutdown**
5. **exit**
6. **interface** *type/id*
7. **encapsulation dot1q** *number*
8. **vrf member** *vrf-name*
9. **ip address** *address*
10. **ip ospf network point-to-point**
11. **ip router ospf** *name* **area** *area-id*
12. **no shutdown**
13. VRF 単位のピアリングごとに、ステップ 5ー12 を繰り返します。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**interface** *type/id*	親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	**no switchport**	インターフェイスでレイヤ 2 スイッチング モードを無効にします。
ステップ 4	**no shutdown**	親インターフェイスを起動します。
ステップ 5	**exit**	インターフェイス コンフィギュレーション モードを終了します。
ステップ 6	**interface** *type/id*	サブインターフェイスインスタンスを定義します。
ステップ 7	**encapsulation dot1q** *number*	サブインターフェイスの VLAN ID を設定します。範囲は 2 ～ 4093 です。
ステップ 8	**vrf member** *vrf-name*	一致する VRF コンテキストにサブインターフェイスをマッピングします。
ステップ 9	**ip address** *address*	サブインターフェイスに IP アドレスを設定する。
ステップ 10	**ip ospf network point-to-point**	サブインターフェイスの OSPF ネットワーク タイプを定義します。
ステップ 11	**ip router ospf** *name* **area** *area-id*	OSPF インスタンスを設定します。
ステップ 12	**no shutdown**	サブインターフェイスを起動します。
ステップ 13	VRF 単位のピアリングごとに、ステップ 5ー12 を繰り返します。	

例：VRF-Lite の OSPF を使用した VXLAN BGP EVPN の設定

VXLAN BGP EVPN から VRF-Lite を使用した外部ルータへの外部接続の例。

OSPF を使用した VXLAN BGP EVPN ボーダー ノードの設定

VXLAN BGP EVPN ボーダー ノードは、外部ルータのネイバー デバイスとして機能します。VRF 名は純粋にローカライズされており、外部ルータの VRF 名と異なる場合があります。重要な点は、L3VNI が VXLAN BGP EVPN ファブリック全体で一貫している必要があることです。読みやすくするために、VRF とインターフェイスの列挙が一貫して使用されます。

設定例は、OSPFv2 を使用した IPv4 アプローチを示しています。

```
route-map extcon-rmap-BGP-to-OSPF permit 10
  match route-type internal
route-map extcon-rmap-OSPF-to-BGP permit 10
!
vrf context myvrf_50001
  vni 50001
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
!
```

```
vlan 2000
  vn-segment 50001
!
interface Vlan2000
  no shutdown
  mtu 9216
  vrf member myvrf_50001
  no ip redirects
  ip forward
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 50001 associate-vrf
!
router bgp 65002
  vrf myvrf_50001
    router-id 10.2.0.6
    address-family ipv4 unicast
      advertise l2vpn evpn
      maximum-paths ibgp 2
      maximum-paths 2
      redistribute ospf EXT route-map extcon-rmap-OSPF-to-BGP
!
router ospf EXT
  vrf myvrf_50001
    redistribute bgp 65002 route-map extcon-rmap-BGP-to-OSPF
!
interface Ethernet1/3
  no switchport
  no shutdown
interface Ethernet1/3.2
  encapsulation dot1q 2
  vrf member myvrf_50001
  ip address 10.31.95.31/24
  ip ospf network point-to-point
  ip router ospf EXT area 0.0.0.0
  no shutdown
```

ルート リークの設定

VXLAN BGP EVPN ファブリックの一元管理型 VRF ルート リークについて

VXLAN BGP EVPN は、MP-BGP とそのルート ポリシーの概念を使用して、プレフィックスをインポートおよびエクスポートします。この非常に広範なルート ポリシー モデルの機能により、ある VRF から別の VRF へ、またはその逆にルートをリークできます。カスタム VRF または VRF デフォルトの任意の組み合わせを使用できます。VRF ルート リークは、クロス VRF ルート ターゲットのインポート/エクスポート設定が行われる（リークポイント）ネットワーク内の特定の場所でのスイッチ ローカル機能です。異なる VRF 間の転送は、コントロール プレーン、つまり、ルートリークの設定が実行される場所、つまり集中型 VRF ルートリークに従います。VXLAN BGP EVPN の追加により、漏出ポイントはクロス VRF インポート/エクス

ポートされたルートをアドバタイズし、それらをリモートVTEPまたは外部ルータにアドバタイズする必要があります。

中央集中型 VRF ルート リークの利点は、リーク ポイントとして機能する VTEP だけが必要な特別な機能を必要とすることです。一方、ネットワーク内の他のすべてのVTEPはこの機能に対して中立です。

集中管理型 VRF ルート リークの注意事項と制約事項

次に、集中管理型VRFルートリークのガイドラインと制限事項を示します。

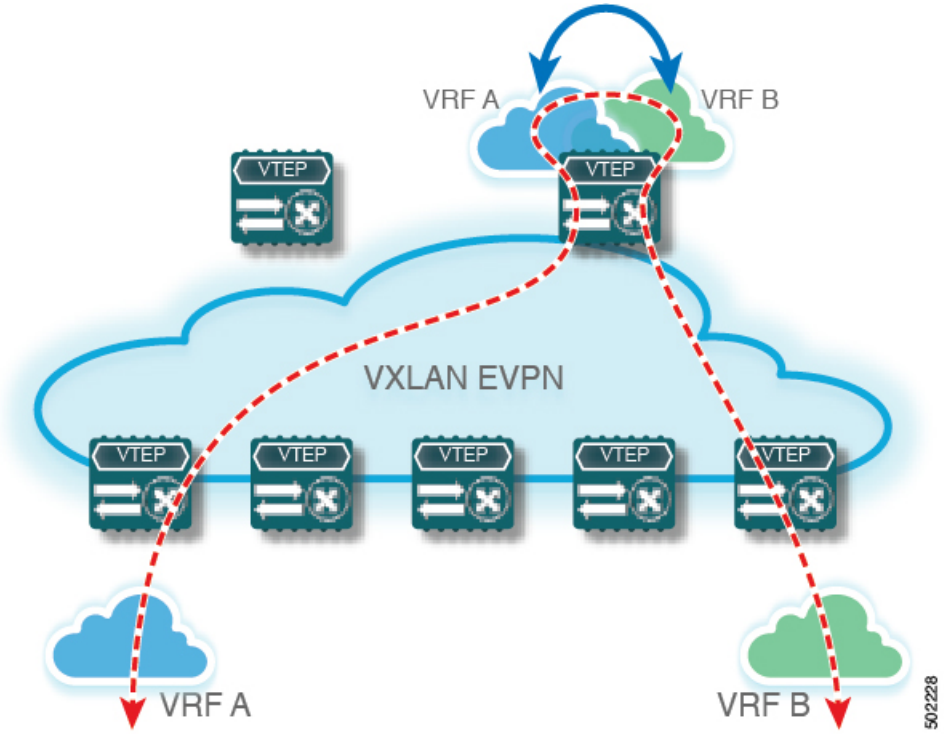
- 完全なクロス VRF 到達可能性を得るには、各プレフィックスを各 VRF にインポートする必要があります。
- **feature bgp** コマンドには **export vrf default** コマンドが必要です。
- VTEP の VRF に特定性の低いローカル プレフィックスがある場合、VTEP は異なる VRF の特定性の高いプレフィックスに到達できない可能性があります。
- ハードウェアでの VXLAN ルーティングおよびVTEP でのパケット再カプセル化は、BGP EVPN を使用した集中管理型 VRF ルートリークに必要です。
- Cisco NX-OS Release 9.3(5) 以降では、非対称 VNI を使用して集中管理型 VRF ルート リークをサポートします。詳細については、ダウンストリーム VNI を使用した VXLAN EVPN に関する （131 ページ） を参照してください。

一元管理型 VRF ルート リーク ブリーフ：カスタム VRF 間の特定のプレフィックス

いくつかのポインタを次に示します。

- VXLAN BGP EVPN ファブリックの中央集中型 VRF ルート リークを図2に示します。
- BGP EVPN プレフィックスは、VRF Red にインポートして VRF Blue からエクスポートしたり、その逆にエクスポートしたりすると、クロス VRF リークが発生します。中央集中型 VRF ルートリークは中央集中型ルーティング ブロック（RBL）で実行され、任意のまたは複数の VTEP になります。
- 設定された特定性の低いプレフィックス（集約）は、ルーティングブロックからそれぞれの宛先 VRF の残りの VTEP にアドバタイズされます。
- BGPEVPNは、ルーティングループの発生を防ぐために以前にインポートされたプレフィックスをエクスポートしません。

図 *18 :* 中央集中型 *VRF* ルートリーク：カスタム *VRF* による特定のプレフィックス



一元管理型 VRF ルート リークの設定：カスタム VRF 間の特定のプレフィックス

ルーティング ブロック VTEP での VRF コンテキストの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **vni** *number*
4. **rd auto**
5. **address-family ipv4 unicast**
6. **route-target both** {**auto** | *rt*}
7. **route-target both** {**auto** | *rt*} **evpn**
8. **route-target import** *rt-from-different-vrf*
9. **route-target import** *rt-from-different-vrf* **evpn**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vrf context** *vrf-name*	VRF を設定します。
ステップ 3	**vni** *number*	VNI を指定します。 VRF に関連付けられている VNI は、多くの場合、レイヤ 3 VNI、L3VNI、または L3VPN と呼ばれます。L3VNI は、参加する VTEP 間で共通の ID として設定されます。
ステップ 4	**rd auto**	VRF のルート識別子（RD）を指定します。RD は、L3VNI 内の VTEP を一意に識別します。
ステップ 5	**address-family ipv4 unicast**	IPv4 ユニキャスト アドレス ファミリを設定します。
ステップ 6	**route-target both** {**auto** | *rt*}	IPv4 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 7	**route-target both** {**auto** | *rt*} **evpn**	IPv4 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 8	**route-target import** *rt-from-different-vrf*	leaked-from VRF から IPv4 プレフィックスをインポートするように RT を設定します。サポートされる形式：ASN2:NN、ASN4:NN、または IPV4:NN
ステップ 9	**route-target import** *rt-from-different-vrf* **evpn**	leaked-from VRF から IPv4 プレフィックスをインポートするように RT を設定します。サポートされる形式：ASN2:NN、ASN4:NN、または IPV4:NN

ルーティング ブロックでの BGP VRF インスタンスの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **advertise l2vpn evpn**
6. **aggregate-address** *prefix/mask*
7. **maximum-paths ibgp** *number*
8. **maximum-paths** *number*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 のアドレス ファミリの設定
ステップ 5	**advertise l2vpn evpn**	IPv4 アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。
ステップ 6	**aggregate-address** *prefix/mask*	宛先 VRF に特定性の低いプレフィックス集約を作成します。
ステップ 7	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 8	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化

例：一元管理型 VRF ルート リークの設定：カスタム VRF 間の特定のプレフィックス

VXLAN BGP EVPN ルーティング ブロックの設定

VXLAN BGP EVPN ルーティング ブロックは、集中型ルート リーク ポイントとして機能します。漏洩設定は、コントロールプレーンの漏洩とデータパスの転送が同じパスをたどるように

ローカライズされます。最も重要なのは、ルーティング ブロックの VRF 設定と、それぞれの宛先 VRF への特定性の低いプレフィックス（集約）のアドバタイズメントです。

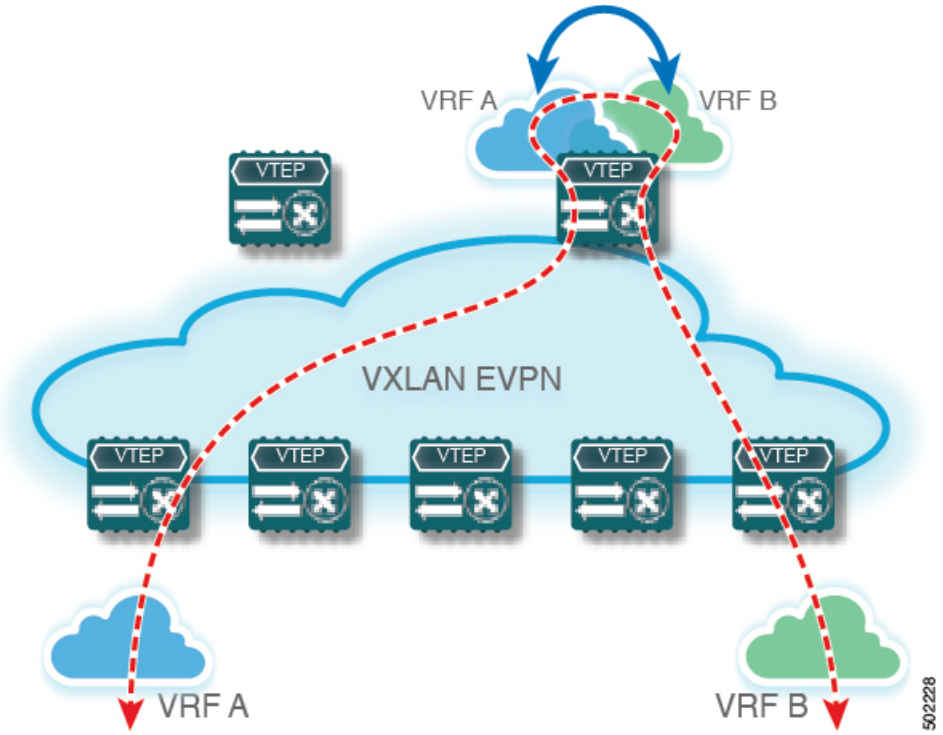
```
vrf context Blue
  vni 51010
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    route-target import 65002:51020
    route-target import 65002:51020 evpn
!
vlan 2110
  vn-segment 51010
!
interface Vlan2110
  no shutdown
  mtu 9216
  vrf member Blue
  no ip redirects
  ip forward
!
vrf context Red
  vni 51020
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    route-target import 65002:51010
    route-target import 65002:51010 evpn
!
vlan 2120
  vn-segment 51020
!
interface Vlan2120
  no shutdown
  mtu 9216
  vrf member Blue
  no ip redirects
  ip forward
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 51010 associate-vrf
  member vni 51020 associate-vrf
!
router bgp 65002
  vrf Blue
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 10.20.0.0/16
      maximum-paths ibgp 2
      Maximum-paths 2
  vrf Red
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 10.10.0.0/16
      maximum-paths ibgp 2
      Maximum-paths 2
```

中央集中型 VRF ルート リーク ブリーフ：カスタム VRF による共有インターネット

次に、いくつかのポイントを示します。

- VXLAN BGP EVPN ファブリックの VRF ルート リークを使用した共有インターネットを次の図に示します。
- デフォルト ルートは共有インターネット VRF からエクスポートされ、ボーダー ノードの VRF Blue および VRF Red 内で再アドバタイズされます。
- VRF Blue および VRF Red のデフォルト ルートが共有インターネット VRF にリークされていないことを確認します。
- VRF Blue および VRF Red の限定的でないプレフィックスは、共有インターネット VRF にエクスポートされ、必要に応じて再アドバタイズされます。
- 境界ノードから残りの VTEP に宛先 VRF（青または赤）にアドバタイズされる、より具体性の低いプレフィックス（集約）。
- BGP EVPN は、ルーティングループの発生を防ぐために以前にインポートされたプレフィックスをエクスポートしません。

図 *19*：中央集中型 *VRF* ルートリーク：カスタム *VRF* による共有インターネット



一元管理型 VRF ルートリークの設定：カスタム VRF による共有インターネット

ボーダー ノードでのインターネット VRF の設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **vni** *number*
4. **ip route 0.0.0.0/0** *next-hop*
5. **rd auto**
6. **address-family ipv4 unicast**
7. **route-target both** {**auto** | *rt*}
8. **route-target both** *shared-vrf-rt* **evpn**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vrf context** *vrf-name*	VRF を設定します。
ステップ 3	**vni** *number*	VNI を指定します。 VRF に関連付けられている VNI は、多くの場合、レイヤ 3 VNI、L3VNI、または L3VPN と呼ばれます。L3VNI は、参加する VTEP 間で共通の ID として設定されます。
ステップ 4	**ip route 0.0.0.0/0** *next-hop*	外部ルータへの共有インターネット VRF のデフォルト ルートを設定します。
ステップ 5	**rd auto**	VRF のルート識別子（RD）を指定します。RD は、L3VNI 内の VTEP を一意に識別します。
ステップ 6	**address-family ipv4 unicast**	IPv4 ユニキャスト アドレス ファミリを設定します。この設定は、IPv4 アンダーレイを使用した IPv4 over VXLAN に必要です。

	コマンドまたはアクション	目的
ステップ 7	**route-target both** {**auto** | *rt*}	EVPN および IPv4 プレフィックスのインポートおよびエクスポートのルート ターゲット（RT）を設定します。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 8	**route-target both** *shared-vrf-rt* **evpn**	共有 IPv4 プレフィックスのインポートおよびエクスポート用の特別なルートターゲット（RT）を設定します。さらなる認定のための追加のインポート/エクスポート マップがサポートされます。

ボーダー ノードでの共有インターネット BGP インスタンスの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **advertise l2vpn evpn**
6. **aggregate-address** *prefix/mask*
7. **maximum-paths ibgp** *number*
8. **maximum-paths** *number*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 のアドレス ファミリの設定
ステップ 5	**advertise l2vpn evpn**	IPv4 アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。

	コマンドまたはアクション	目的
ステップ 6	**aggregate-address** *prefix/mask*	宛先 VRF に特定性の低いプレフィックス集約を作成します。
ステップ 7	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 8	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。

ボーダー ノードでのカスタム VRF の設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **ip prefix-list** *name* **seq 5 permit 0.0.0.0/0**
3. **route-map** *name* **deny 10**
4. **match ip address prefix-list** *name*
5. **route-map** *name* **permit 20**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル構成モードを開始します。
ステップ 2	**ip prefix-list** *name* **seq 5 permit 0.0.0.0/0**	デフォルトルート フィルタリングの IPv4 プレフィックス リストを設定します。
ステップ 3	**route-map** *name* **deny 10**	default-route がリークされるのを防ぐために、先行する deny ステートメントを使用してルートマップを作成します。
ステップ 4	**match ip address prefix-list** *name*	default-route を含む IPv4 プレフィックスリストと照合します。
ステップ 5	**route-map** *name* **permit 20**	ルートリークを介して一致しないルートをアドバタイズする後続の allow ステートメントを使用してルートマップを作成します。

ボーダーノードでのカスタム VRF コンテキストの設定 - 1

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **vni** *number*
4. **rd auto**
5. **ip route 0.0.0.0/0 Null0**
6. **address-family ipv4 unicast**
7. **route-target both** {**auto** | *rt*}
8. **route-target both** {**auto** | *rt*} **evpn**
9. **import map** *name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vrf context** *vrf-name*	VRF を設定します。
ステップ 3	**vni** *number*	VNI を指定します。VRF に関連付けられている VNI は、多くの場合、レイヤ 3 VNI、L3VNI、または L3VPN と呼ばれます。L3VNI は、参加する VTEP 間で共通の識別子として設定されます。
ステップ 4	**rd auto**	VRF のルート識別子（RD）を指定します。RD は、L3VNI 内の VTEP を一意に識別します。
ステップ 5	**ip route 0.0.0.0/0 Null0**	共通 VRF でデフォルト ルートを設定し、共有インターネット VRF を持つボーダーノードにトラフィックを引き付けます。
ステップ 6	**address-family ipv4 unicast**	IPv4 アドレス ファミリを設定します。この設定は、IPv4 アンダーレイを使用した IPv4 over VXLAN に必要です。
ステップ 7	**route-target both** {**auto**	*rt*}	IPv4 アドレス ファミリ内の IPv4 プレフィックスのインポートおよびエクスポート用のルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。

	コマンドまたはアクション	目的
ステップ 8	**route-target both** {**auto** | *rt*} **evpn**	IPv4 アドレス ファミリ内の IPv4 プレフィックスのインポートおよびエクスポート用のルート ターゲット（RT）を設定します。RT は、VRF 単位のプレフィックス インポート/エクスポート ポリシーに使用されます。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 9	**import map** *name*	このルーティング テーブルにインポートされるルートにルート マップを適用します。

ボーダー ノードでの BGP でのカスタム VRF インスタンスの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **advertise l2vpn evpn**
6. **network 0.0.0.0/0**
7. **maximum-paths ibgp** *number*
8. **maximum-paths** *number*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 のアドレス ファミリを設定します。
ステップ 5	**advertise l2vpn evpn**	IPv4 アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。

	コマンドまたはアクション	目的
ステップ 6	**network 0.0.0.0/0**	IPv4 デフォルト ルート ネットワーク ステートメントを作成しています。
ステップ 7	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 8	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。

例：一元管理型 VRF ルート リークの設定：カスタム VRF による共有インターネット

共有インターネット VRF による中央集中型 VRF ルート リークの例

共有インターネット VRF の VXLAN BGP EVPN ボーダー ノードの設定

VXLAN BGP EVPN ボーダー ノードは、集中型共有インターネット VRF を提供します。漏出設定は、コントロール プレーンの漏出とデータ パス転送が同じパスをたどるようにローカライズされます。最も重要な点は、ボーダー ノードの VRF 設定と、デフォルト ルートと特定性の低いプレフィックス（集約）をそれぞれの宛先 VRF にアドバタイズすることです。

```
vrf context Shared
  vni 51099
  ip route 0.0.0.0/0 10.9.9.1
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    route-target both 99:99
    route-target both 99:99 evpn
!
vlan 2199
  vn-segment 51099
!
interface Vlan2199
  no shutdown
  mtu 9216
  vrf member Shared
  no ip redirects
  ip forward
!
ip prefix-list PL_DENY_EXPORT seq 5 permit 0.0.0.0/0
!
route-map RM_DENY_IMPORT deny 10
 match ip address prefix-list PL_DENY_EXPORT
route-map RM_DENY_IMPORT permit 20
!
vrf context Blue
  vni 51010
  ip route 0.0.0.0/0 Null0
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    route-target both 99:99
    route-target both 99:99 evpn
    import map RM_DENY_IMPORT
```

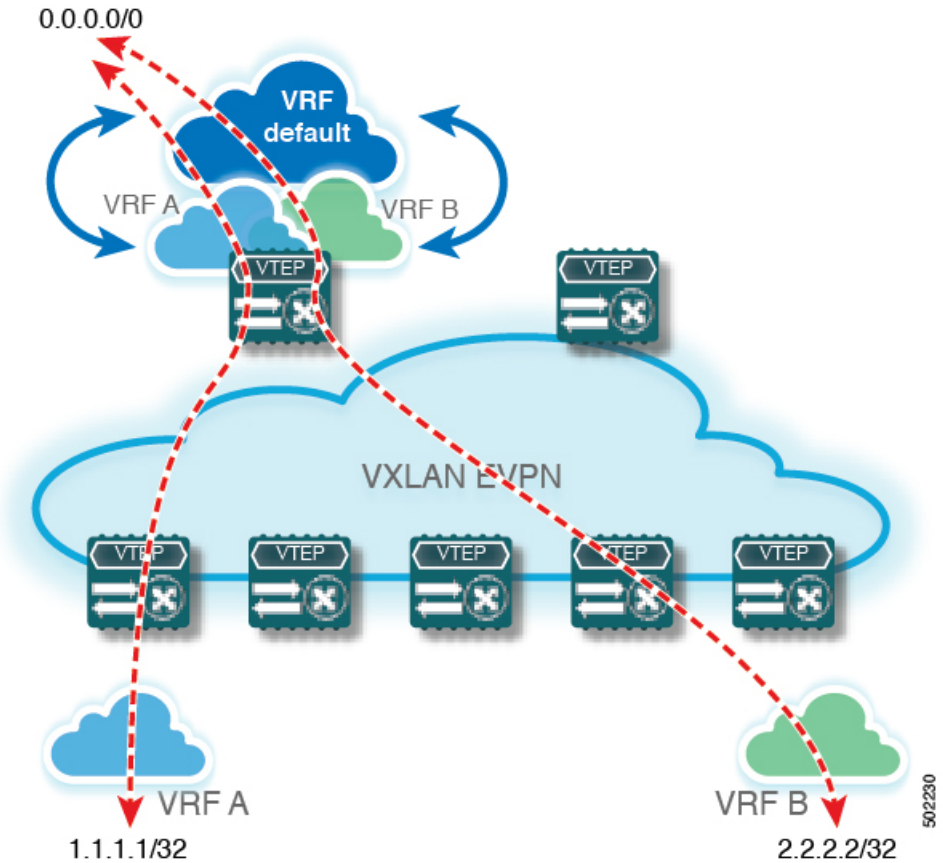
```
!
vlan 2110
  vn-segment 51010
!
interface Vlan2110
  no shutdown
  mtu 9216
  vrf member Blue
  no ip redirects
  ip forward
!
vrf context Red
  vni 51020
  ip route 0.0.0.0/0 Null0
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    route-target both 99:99
    route-target both 99:99 evpn
    import map RM_DENY_IMPORT
!
vlan 2120
  vn-segment 51020
!
interface Vlan2120
  no shutdown
  mtu 9216
  vrf member Blue
  no ip redirects
  ip forward
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 51099 associate-vrf
  member vni 51010 associate-vrf
  member vni 51020 associate-vrf
!
router bgp 65002
  vrf Shared
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 10.10.0.0/16
      aggregate-address 10.20.0.0/16
      maximum-paths ibgp 2
      maximum-paths 2
  vrf Blue
    address-family ipv4 unicast
      advertise l2vpn evpn
      network 0.0.0.0/0
      maximum-paths ibgp 2
      maximum-paths 2
  vrf Red
    address-family ipv4 unicast
      advertise l2vpn evpn
      network 0.0.0.0/0
      maximum-paths ibgp 2
      maximum-paths 2
```

一元管理型 VRF ルート リーク ブリーフ：VRF デフォルトでの共有インターネット

いくつかのポインタを次に示します。

- VXLAN BGP EVPN ファブリックの VRF ルート漏洩を伴う共有インターネットを図 4 に示します。
- default-route は VRF default からエクスポートされ、ボーダーノードの VRF Blue および VRF Red 内で再アドバタイズされます。
- VRF Blue および VRF Red のデフォルト ルートが共有インターネット VRF にリークされていないことを確認します。
- VRF Blue および VRF Red の限定的でないプレフィックスは、VRF デフォルトにエクスポートされ、必要に応じて再アドバタイズされます。
- 境界ノードから残りの VTEP に宛先 VRF（青または赤）にアドバタイズされる、より具体性の低いプレフィックス（集約）。
- BGP EVPN は、ルーティングループの発生を防ぐために以前にインポートされたプレフィックスをエクスポートしません。

図 *20* : 中央集中型 *VRF* ルートリーク：*VRF* デフォルトでの共有インターネット



一元管理型 VRF ルート リークの設定：VRF デフォルトでの共有インターネット

ボーダー ノードでの VRF デフォルトの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **ip route 0.0.0.0/0** *next-hop*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**ip route 0.0.0.0/0** *next-hop*	VRF のデフォルト ルートを外部ルータに設定する（例）

ボーダー ノードでの VRF デフォルトの BGP インスタンスの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system number*
3. **address-family ipv4 unicast**
4. **aggregate-address** *prefix/mask*
5. **maximum-paths** *number*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system number*	BGP 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	**address-family ipv4 unicast**	IPv4 のアドレス ファミリを設定します。
ステップ 4	**aggregate-address** *prefix/mask*	VRF のデフォルトで、より限定的なプレフィックス集約を作成します。
ステップ 5	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。

ボーダー ノードでのカスタム VRF の設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **ip prefix-list** *name* **seq 5 permit 0.0.0.0/0**
3. **route-map** *name* **deny 10**
4. **match ip address prefix-list** *name*
5. **route-map** *name* **permit 20**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル構成モードを開始します。
ステップ 2	**ip prefix-list** *name* **seq 5 permit 0.0.0.0/0**	デフォルトルート フィルタリングの IPv4 プレフィックス リストを設定します。
ステップ 3	**route-map** *name* **deny 10**	default-route がリークされるのを防ぐために、先行する deny ステートメントを使用してルートマップを作成します。
ステップ 4	**match ip address prefix-list** *name*	default-route を含む IPv4 プレフィックスリストと照合します。
ステップ 5	**route-map** *name* **permit 20**	ルートリークを介して一致しないルートをアドバタイズする後続の allow ステートメントを使用してルートマップを作成します。

ボーダー ノードでの VRF デフォルトから許可されるプレフィックスのフィルタの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **route-map** *name* **permit 10**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**route-map** *name* **permit 10**	allow ステートメントを使用してルートマップを作成し、カスタマー VRF およびその後のリモート VTEP にルートリークを介してルートをアドバタイズします。

ボーダーノードでのカスタム VRF コンテキストの設定 - 2

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **vni** *number*
4. **rd auto**
5. **ip route 0.0.0.0/0 Null0**
6. **address-family ipv4 unicast**
7. **route-target both** {**auto** | *rt*}
8. **route-target both** {**auto** | *rt*} **evpn**
9. **route-target both** *shared-vrf-rt*
10. **route-target both** *shared-vrf-rt* **evpn**
11. **import vrf default map** *name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**vrf context** *vrf-name*	VRF 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	**vni** *number*	VNI を指定します。VRF に関連付けられている VNI は、多くの場合、レイヤ 3 VNI、L3VNI、または L3VPN と呼ばれます。L3VNI は、参加する VTEP 間で共通の識別子として設定されます。
ステップ 4	**rd auto**	VRF のルート識別子（RD）を指定します。RD は、L3VNI 内の VTEP を一意に識別します。
ステップ 5	**ip route 0.0.0.0/0 Null0**	共通 VRF でデフォルト ルートを設定し、共有インターネット VRF を持つボーダーノードにトラフィックを引き付けます。
ステップ 6	**address-family ipv4 unicast**	IPv4 アドレス ファミリを設定します。この設定は、IPv4 アンダーレイを使用した IPv4 over VXLAN に必要です。
ステップ 7	**route-target both** {**auto** | *rt*}	IPv4 アドレス ファミリ内の EVPN および IPv4 プレフィックスのインポートおよびエクスポート用のルート ターゲット（RT）を設定します。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 8	**route-target both** {**auto** | *rt*} **evpn**	IPv4 アドレス ファミリ内の EVPN および IPv4 プレフィックスのインポートおよびエクスポート用のルート ターゲット（RT）を設定します。RT を入力する場合は、ASN2:NN、ASN4:NN、または IPV4:NN の形式がサポートされます。非対称 VNI をサポートするには、手動で設定された RT が必要です。
ステップ 9	**route-target both** *shared-vrf-rt*	共有 IPv4 プレフィックスのインポート/エクスポート用の特別なルート ターゲット（RT）を設定します。さらなる認定のための追加のインポート/エクスポート マップがサポートされます。
ステップ 10	**route-target both** *shared-vrf-rt* **evpn**	共有 IPv4 プレフィックスのインポート/エクスポート用の特別なルート ターゲット（RT）を設定します。さらなる認定のための追加のインポート/エクスポート マップがサポートされます。
ステップ 11	**import vrf default map** *name*	VRF デフォルトからのすべてのルートが、特定のルート マップに従ってカスタム VRF にインポートされることを許可します。

ボーダー ノードでの BGP でのカスタム VRF インスタンスの設定

この手順は、IPv6 にも同様に適用されます。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **vrf** *vrf-name*
4. **address-family ipv4 unicast**
5. **advertise l2vpn evpn**
6. **network 0.0.0.0/0**
7. **maximum-paths ibgp** *number*
8. **maximum-paths** *number*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**router bgp** *autonomous-system-number*	BGP を設定します。
ステップ 3	**vrf** *vrf-name*	VRF を指定します。
ステップ 4	**address-family ipv4 unicast**	IPv4 のアドレス ファミリを設定します。
ステップ 5	**advertise l2vpn evpn**	IPv4 アドレス ファミリ内の EVPN ルートのアドバタイズメントを有効にします。
ステップ 6	**network 0.0.0.0/0**	IPv4 デフォルト ルート ネットワーク ステートメントを作成しています。
ステップ 7	**maximum-paths ibgp** *number*	iBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。
ステップ 8	**maximum-paths** *number*	eBGP プレフィックスの等コスト マルチパス（ECMP）の有効化。

例：一元管理型 VRF ルート リークの設定：カスタム VRF を使用した VRF デフォルト

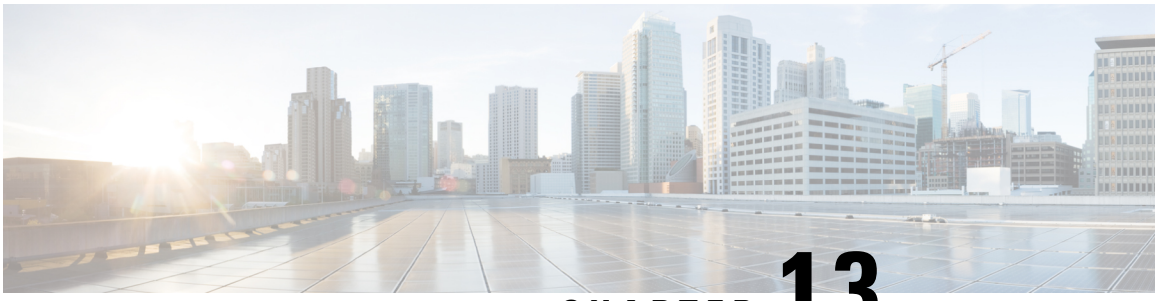
VRF デフォルトによる中央集中型 VRF ルート リークの例

VRF デフォルトの VXLAN BGP EVPN ボーダー ノードの設定

VXLAN BGP EVPN ボーダー ノードは、VRF デフォルトへの集中型アクセスを提供します。漏出設定は、コントロール プレーンの漏出とデータ パス転送が同じパスをたどるようにローカライズされます。最も重要な点は、ボーダー ノードの VRF 設定と、デフォルト ルートと特定性の低いプレフィックス（集約）をそれぞれの宛先 VRF にアドバタイズすることです。

```
ip route 0.0.0.0/0 10.9.9.1
!
ip prefix-list PL_DENY_EXPORT seq 5 permit 0.0.0.0/0
!
route-map permit 10
match ip address prefix-list PL_DENY_EXPORT
route-map RM_DENY_EXPORT permit 20
route-map RM_PERMIT_IMPORT permit 10
!
vrf context Blue
  vni 51010
  ip route 0.0.0.0/0 Null0
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    import vrf default map RM_PERMIT_IMPORT
    export vrf default 100 map RM_DENY_EXPORT allow-vpn
!
vlan 2110
  vn-segment 51010
!
interface Vlan2110
  no shutdown
  mtu 9216
  vrf member Blue
  no ip redirects
  ip forward
!
vrf context Red
  vni 51020
  ip route 0.0.0.0/0 Null0
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    import vrf default map RM_PERMIT_IMPORT
    export vrf default 100 map RM_DENY_EXPORT allow-vpn
!
vlan 2120
  vn-segment 51020
!
interface Vlan2120
  no shutdown
  mtu 9216
  vrf member Blue
  no ip redirects
  ip forward
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 51010 associate-vrf
  member vni 51020 associate-vrf
!
```

```
router bgp 65002
  address-family ipv4 unicast
      aggregate-address 10.10.0.0/16
      aggregate-address 10.20.0.0/16
      maximum-paths 2
      maximum-paths ibgp 2
  vrf Blue
    address-family ipv4 unicast
      advertise l2vpn evpn
      network 0.0.0.0/0
      maximum-paths ibgp 2
      maximum-paths 2
  vrf Red
    address-family ipv4 unicast
      advertise l2vpn evpn
      network 0.0.0.0/0
      maximum-paths ibgp 2
      maximum-paths 2
```



CHAPTER 13

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

この章は、次の内容で構成されています。

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定の詳細

データセンターの展開では、EVPN コントロール プレーン ラーニング、マルチテナンシー、シームレスなモビリティ、冗長性、POD の追加が容易になるなどの利点から、VXLAN EVPN を採用しています。同様に、コアは LDP ベースの MPLS L3VPN ネットワークであるか、従来の MPLS L3VPN LDP ベースのアンダーレイからセグメントルーティング（SR）のようなより高度なソリューション（SR）に移行するかのいずれかです。セグメント ルーティングは、ユニファイド IGP および MPLS コントロール プレーン、シンプルなトラフィック エンジニアリング方式、簡単な設定、SDN の採用などの利点のために採用されています。

データセンター内 DCI ノードとして動作するボーダー リーフまたは共有 PE ルータの 2 つの異なるテクノロジーにより、VXLAN から DCI ノードで MPLS ベースのコアにハンドオフするのは自然なことです。これらのノードは、DC ドメインのエッジにあり、コア エッジ ルータとインターフェイスします。

に関する注意事項と制限事項 EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定 の注意事項と制限事項は次のとおりです。

サポートされる機能は次のとおりです。

- -R および -RX ライン カードを備えた Cisco Nexus 9504 および 9508 スイッチ。
- レイヤ 3 オーファン
- VXLAN DC ドメイン内の 256 ピア/ノード
- -RX ライン カードでは、24,000 ECMP ルートがサポートされます。

（注） **no hardware profile mpls extended-ecmp** コマンドを入力すると、モードは4 K ECMP ルートに切り替わります。これは、ラインカードが -RX で、ECMP グループに正確に 2 つのパスがある場合にのみ適用されます。

- 出力 RACL（e-RACL）TCAM 機能と MPLS 拡張 ECMP 機能は相互に排他的です。Cisco Nexus N9K-X9636C-RX ライン カードで MPLS 拡張 ECMP（**hardware profile mpls extended-ecmp**） を有効にするには、e-RACL TCAM カービングを 0 に設定します。
- Cisco NX-OS リリース 10.3(3)F 以降では、MPLS LDP ユーザー パスワードのタイプ 6 暗号化が Cisco NX-OS スイッチでサポートされています。

次の機能はサポートされていません。

- サブネットが DC ドメイン全体に拡大する
- vPC
- SVI/サブインターフェイス

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

これらの設定手順は、VXLAN ドメインから MPLS ドメインにルートをインポートして再発信し、VXLAN ドメインに戻すためにボーダー リーフ スイッチで必要です。

手順の概要

1. **configure terminal**
2. **[no] install feature-set mpls**

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

3. **[no] feature-set mpls**
4. **feature mpls l3vpn**
5. **feature mpls ldp**
6. **mpls ip**
7. **nv overlay evpn**
8. **router bgp** *number*
9. **address-family ipv4 unicast**
10. **redistribute direct route-map** *route-map-name*
11. **exit**
12. **address-family l2vpn evpn**
13. **exit**
14. **neighbor** *address* **remote-as** *number*
15. **update-source** *type/id*
16. **ebgp-multihop** *ttl-value*
17. **address-family ipv4 unicast**
18. **send-community extended**
19. **exit**
20. **address-family ipv4 labeled-unicast**
21. **send-community extended**
22. **address-family vpnv4 unicast**
23. **send-community extended**
24. **import l2vpn evpn reoriginate**
25. **neighbor** *address* **remote-as** *number*
26. **address-family ipv4 unicast**
27. **send-community extended**
28. **address-family ipv6 unicast**
29. **send-community extended**
30. **address-family l2vpn evpn**
31. **send-community extended**
32. **import vpn unicast reoriginate**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**[no] install feature-set mpls** 例： `switch# install feature-set mpls`	MPLS 機能セットを有効化します。 このコマンドの no 形式は、MPLS 機能セットをアンインストールします。

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

	コマンドまたはアクション	目的
ステップ 3	**[no] feature-set mpls** 例： `switch# feature-set mpls`	MPLS 機能セットを有効化します。 このコマンドの no 形式は、MPLS 機能セットをアンインストールします。
ステップ 4	**feature mpls l3vpn** 例： `switch# feature mpls l3vpn`	MPLS レイヤ 3 VPN 機能をイネーブルにします。
ステップ 5	**feature mpls ldp** 例： `switch# feature mpls ldp`	MPLS ラベル配布プロトコル（LDP）をイネーブルにします。

ステップ 6	**mpls ip**
	例：
	`switch# interface Ethernet1/1`
	`switch(config-if)# mpls ip`
		MPLS リンクである指定されたインターフェイスで MPLS を有効にします。

ステップ 7	**nv overlay evpn** 例： `switch(config)# nv overlay evpn`	EVPN コントロール プレーンを VXLAN にイネーブルにします。
ステップ 8	**router bgp** *number* 例： `switch(config)# router bgp 100`	BGP を設定します。この引数の値の範囲は 1 ～ 4294967295 です。
ステップ 9	**address-family ipv4 unicast** 例： `switch(config-router)# address-family ipv4 unicast`	IPv4 のアドレス ファミリを設定します。
ステップ 10	**redistribute direct route-map** *route-map-name* 例： `switch(config-router-af)# redistribute direct route-map passall`	直接接続されたルート マップを設定します。
ステップ 11	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 12	**address-family l2vpn evpn** 例： `switch(config-router)# address-family l2vpn evpn`	L2VPN アドレス ファミリを設定します。

	コマンドまたはアクション	目的
ステップ 13	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 14	**neighbor** *address* **remote-as** *number* 例： `switch(config-router)# neighbor 108.108.108.108 remote-as 22`	BGP ネイバーを設定します。引数 *number* の範囲は、1 ～ 65535 です。
ステップ 15	**update-source** *type/id* 例： `switch(config-router-neighbor)# update-source loopback100`	BGP セッションの送信元を指定し、更新します。
ステップ 16	**ebgp-multihop** *ttl-value* 例： `switch(config-router-neighbor)# ebgp-multihop 10`	リモート ピアにマルチホップ TTL を指定します *ttl-value* の範囲は 2 ～ 255 です。
ステップ 17	**address-family ipv4 unicast** 例： `switch(config-router-neighbor)# address-family ipv4 unicast`	ユニキャスト サブアドレス ファミリを設定します。
ステップ 18	**send-community extended** 例： `switch(config-router-neighbor-af)# send-community extended`	このネイバーのコミュニティ属性を設定します。
ステップ 19	**exit** 例： `switch(config-router-neighbor-af)# exit`	コマンド モードを終了します。
ステップ 20	**address-family ipv4 labeled-unicast** 例： `switch(config-router-neighbor)# address-family ipv4 labeled-unicast`	RFC 3107 で指定されているように、ラベル付き IPv4 ユニキャスト ルートをアドバタイズします。
ステップ 21	**send-community extended** 例： `switch(config-router-neighbor-af)# send-community extended`	拡張コミュニティ属性を送信します。
ステップ 22	**address-family vpnv4 unicast** 例：	IPv4 のアドレス ファミリを設定します。

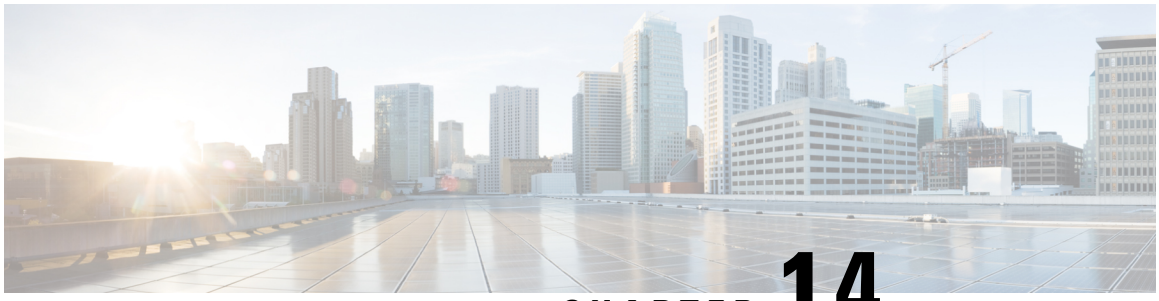
EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

	コマンドまたはアクション	目的
	`switch(config-router-neighbor)# address-family vpnv4 unicast`	
ステップ 23	**send-community extended** 例： `switch(config-router)# send-community extended`	拡張コミュニティ属性を送信します。
ステップ 24	**import l2vpn evpn reoriginate** 例： `switch(config-router)# import l2vpn evpn reoriginate`	新しい RT でルートを再発信します。
ステップ 25	**neighbor** *address* **remote-as** *number* 例： `switch(config-router)# neighbor 175.175.175.2 remote-as 1`	ネイバーを定義します。
ステップ 26	**address-family ipv4 unicast** 例： `switch(config-router)# address-family ipv4 unicast`	IPv4 のアドレス ファミリを設定します。
ステップ 27	**send-community extended** 例： `switch(config-router)# send-community extended`	BGP ネイバーのコミュニティを設定します。
ステップ 28	**address-family ipv6 unicast** 例： `switch(config-router)# address-family ipv6 unicast`	IPv4 ユニキャスト アドレス ファミリを設定します。これは、IPv4 アンダーレイを使用した IPv6 over VXLAN に必要です。
ステップ 29	**send-community extended** 例： `switch(config-router)# send-community extended`	BGP ネイバーのコミュニティを設定します。
ステップ 30	**address-family l2vpn evpn** 例： `switch(config-router)# address-family l2vpn evpn`	L2VPN アドレス ファミリを設定します。
ステップ 31	**send-community extended** 例： `switch(config-router)# send-community extended`	BGP ネイバーのコミュニティを設定します。
ステップ 32	**import vpn unicast reoriginate** 例：	新しい RT でルートを再発信します。

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定

	コマンドまたはアクション	目的
	`switch(config-router)# import vpn unicast reoriginate`	

EVPN と L3VPN (MPLS LDP) のシームレスな統合の設定



CHAPTER 14

EVPN と L3VPN (MPLS SR) のシームレスな統合の設定

この章は、次の内容で構成されています。

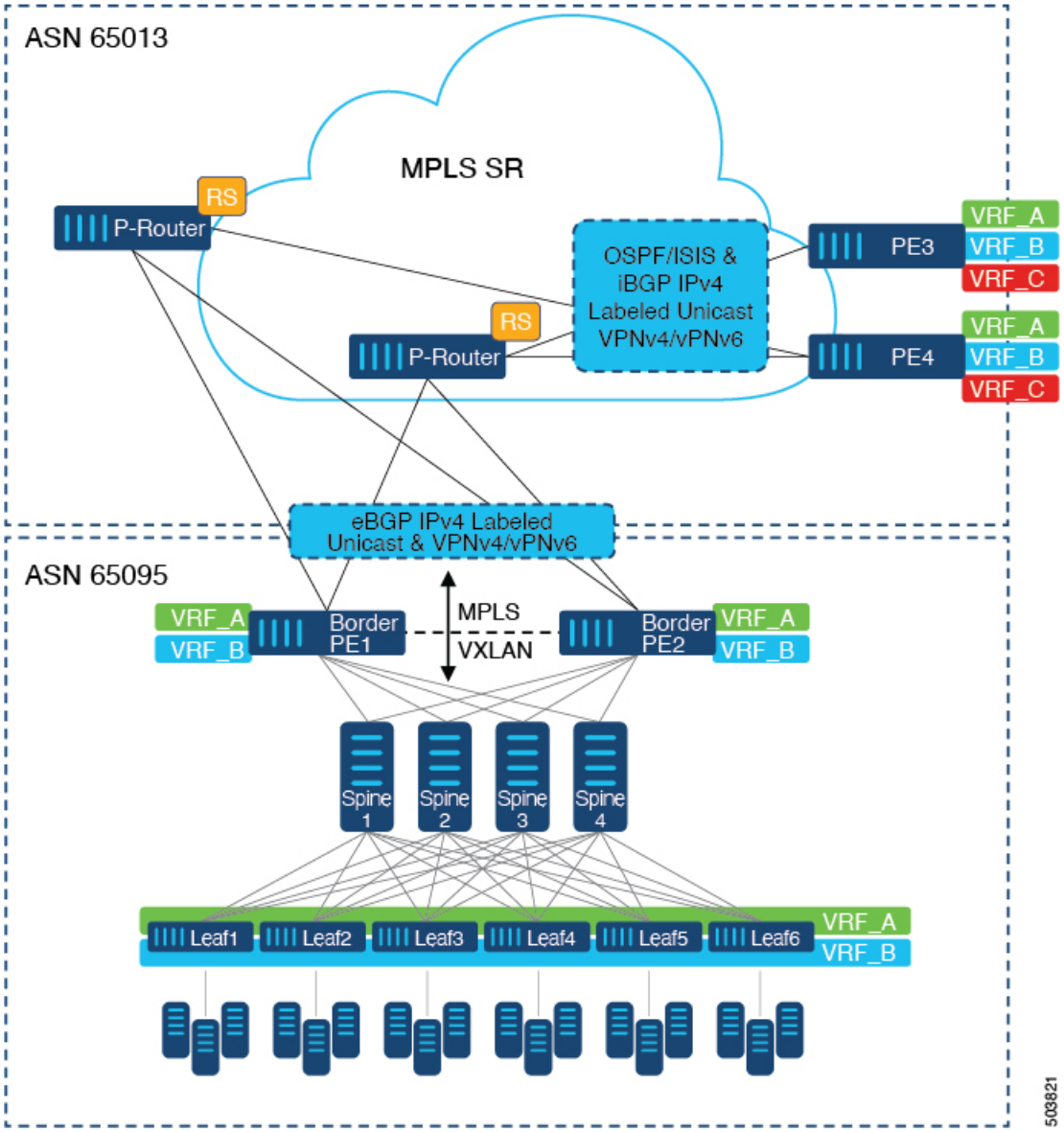
EVPN と L3VPN (MPLS SR) のシームレスな統合の設定の詳細

データセンター（DC）展開では、EVPN コントロール プレーン ラーニング、マルチテマルチテナント、シームレスモビリティ、冗長性、水平スケーリングが容易になるなどの利点から、VXLAN EVPN を採用しています。同様に、コアネットワークはそれぞれの機能を持つさまざまなテクノロジーに移行します。ラベル配布プロトコル（LDP）およびレイヤ3 VPN（L3VPN）を備えたMPLSは、データセンターを相互接続する多くのコアネットワークに存在します。テクノロジーの進化により、LDPベースのアンダーレイを使用した従来のMPLS L3VPNからL3VPNを使用したMPLSベースのセグメントルーティング（SR）への変換が可能になりました。セグメント ルーティングは、次のような利点のために採用されています。

- Unified IGP および MPLS コントロール プレーン
- よりシンプルなトラフィック エンジニアリング手法

VXLAN EVPNにデータセンター（DC）が確立され、マルチテナント対応のトランスポートを必要とするコアネットワークでは、シームレスな統合が自然に必要になります。さまざまなコントロールプレーンプロトコルとカプセル化（ここではVXLANからMPLSベースのコアネットワークまで）をシームレスに統合するために、Cisco Nexus 9000シリーズスイッチは、データセンターとコアルータ（プロバイダールータまたはプロバイダーエッジルータ）。

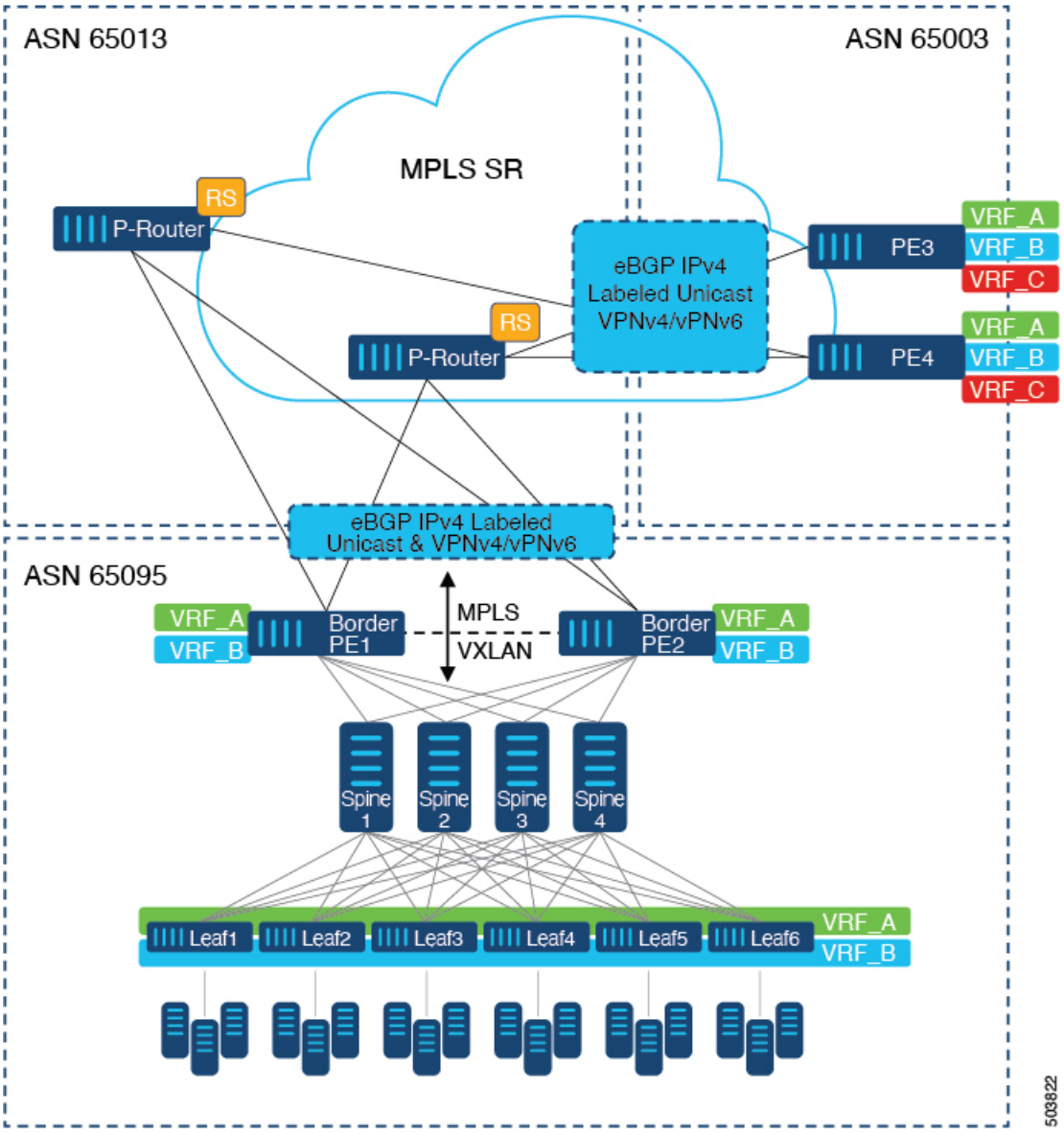
図 *21 : DC*からコアネットワークドメインへの分離を使用したトポロジ



上の図では、VXLAN EVPNを実行する単一のデータセンターファブリックが示されています。データセンターに存在するVRF（VRF_A、VRF_B）は、MPLSベースのセグメントルーティング（MPLS-SR）を実行するWAN /コア上で拡張する必要があります。データセンターファブリックボーダースイッチは、VXLAN BGP EVPNをMPLS-SRとL3VPN（VPNv4 / VPNv6）で相互接続するボーダープロバイダーエッジ（ボーダーPE1、ボーダーPE2）として機能します。BPEは、IPv4ラベル付きユニキャストとVPNv4 / VPNv6アドレスファミリ（AF）を使用して、eBGPを介してプロバイダールータ（P-Router）と相互接続されます。P-Routerは、前述のAFのBGPルートリフレクタとして機能し、iBGPを介してMPLS-SRプロバイダーエッジ（PE3、PE4）に必要なルートをリレーします。コントロールプレーンとしてのBGPの使用に加えて、同じ自律システム（AS）内のMPLS-SRノード間では、ラベル配布にIGP（OSPFまたはISIS）が使用されます。上の図に示すPE（PE3、PE4）から、Inter-ASオプションAを使用して、データセンター

またはコアネットワークVRFを別の外部ネットワークに拡張できます。この図では1つのデータセンターのみを示していますが、MPLS-SRネットワークを使用して複数のデータセンターファブリックを相互接続できます。

図 *22*：コアネットワーク内の複数の管理ドメイン



別の導入シナリオは、コアネットワークが複数の管理ドメインまたは自律システム（AS）に分かれている場合です。上の図では、VXLAN EVPNを実行する単一のデータセンターファブリックが示されています。データセンターに存在するVRF（VRF_A、VRF_B）は、MPLSベースのセグメントルーティング（MPLS-SR）を実行するWAN /コア上で拡張する必要があります。データセンターファブリックボーダースイッチは、VXLAN BGP EVPNをMPLS-SRとL3VPN（VPNv4 / VPNv6）で相互接続するボーダープロバイダーエッジ（ボーダーPE1、ボーダーPE2）として機能します。BPEは、IPv4ラベル付きユニキャストとVPNv4 / VPNv6アドレスファ

ミリ（AF）を使用して、eBGPを介してプロバイダールータ（P-Router）と相互接続されます。Pルータは前述のAFのBGPルートサーバとして機能し、eBGPを介してMPLS-SRプロバイダーエッジ（PE3、PE4）に必要なルートをリレーします。MPLS-SRノード間では、他のコントロールプレーンプロトコルは使用されません。前のシナリオと同様に、PE（PE3、PE4）はInter-ASオプションAで動作して、データセンターまたはコアネットワークVRFを外部ネットワークに拡張できます。この図では1つのデータセンターのみを示していますが、MPLS-SRネットワークを使用して複数のデータセンターファブリックを相互接続できます。

Cisco NX-OS リリース 10.3(1)F 以降、境界 PE で DSCP ベースの SRTE トラフィック ステアリングがサポートされます。詳細については、DSCP ベースの SR-TE フロー ステアリングの構成を参照してください。このシナリオは、L3VPN（MPLS SR）でのみサポートされます。ボーダー PE（ボーダー リーフ）シナリオを表す上の図では、次の点に注意してください。

1. 着信 VXLAN トラフィックは終端し、PE3 または PE4 への標準ルーティングの最適パスに従うため、L3VPN (MPLS SR) に送信されます。
2. PE1 に入る着信 VXLAN トラフィックは終端し、L3 VNI に適用される SRTE トラフィック ステアリング ポリシーは、標準ルーティングの最適パスを上書きし、SRTE フロー ステアリング ポリシーに基づいて PE3 または PE4 への代替パスを選択するようにステアリングします。

MPLS SR の追加情報については、『*Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide*』を参照してください。

に関する注意事項と制限事項 EVPN と L3VPN (MPLS SR) のシームレスな統合の設定

機能	Cisco Nexus 9300-FX2、FX3、GX、GX2、H2R、H1 プラットフォーム スイッチ	-R ラインカードを搭載した Cisco Nexus 9504 および 9508 スイッチ	注
VXLAN EVPN から SR-L3VPN へ	はい	はい	異なる DC ポッド間のレイヤ 3 接続を拡張します。SR 拡張を使用して IGP/BGP のアンダーレイを設定します。
VXLAN EVPN から SR-L3VPN へ	はい	はい	VXLAN を実行する DC POD と SR を実行する任意のドメイン（DC または CORE）間のレイヤ 3 接続を拡張します。

機能	Cisco Nexus 9300-FX2、FX3、GX、GX2、H2R、H1 プラットフォーム スイッチ	-R ラインカードを搭載した Cisco Nexus 9504 および 9508 スイッチ	注
VXLAN EVPN から MPLS L3VPN（LDP）	非対応	はい	アンダーレイは LDP です。

次の Cisco Nexus プラットフォーム スイッチは、EVPN と L3VPN（MPLS SR）のシームレスな統合をサポートします。

- 9336C-FX2 スイッチ
- 93240YC-FX2 スイッチ
- 9300-FX3 プラットフォーム スイッチ
- 9300-GX プラットフォーム スイッチ
- 96504YC-R および 9636C-RX ライン カードを搭載した 9504 および 9508 プラットフォーム スイッチ（9636C-R および 9636Q-R ライン カードはサポートされません）

Cisco NX-OS リリース 10.2(3)F 以降、EVPN と L3VPN（MPLS SR）のシームレスな統合が Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされています。

Cisco NX-OS リリース 10.4(1)F 以降、EVPN と L3VPN（MPLS SR）のシームレスな統合が Cisco Nexus 9332D-H2R プラットフォーム スイッチでサポートされています。

Cisco NX-OS リリース 10.4(2)F 以降、EVPN と L3VPN（MPLS SR）のシームレスな統合が Cisco Nexus 93400LD-H1 プラットフォーム スイッチでサポートされています。

Cisco NX-OS リリース 10.4(3)F 以降、EVPN と L3VPN（MPLS SR）のシームレスな統合が Cisco Nexus 9364C-H1、X9836DM-A および X98900CD-A ライン カードを搭載した 9808/9804 スイッチでサポートされています。

EVPN と L3VPN（MPLS SR）のシームレスな統合により、次の機能がサポートされます。

- Host Facing（Downlinks to）
  - 個々のレイヤ 3 インターフェイス（孤立ポート）
  - レイヤ 3 ポート チャネル
  - レイヤ 3 サブインターフェイス
  - Inter-ASオプションA（VRF-lite とも呼ばれる）
- コアフェーシング（VXLANへのアップリンク）
  - 個々のレイヤ 3 インターフェイス
  - レイヤ3ポートチャネル
- コアフェーシング（MPLS SRへのアップリンク）

- 個々のレイヤ 3 インターフェイス
- VRF 単位のラベル
- VPN ラベル統計情報

- エンドツーエンド Time to Live (TTL) と明示的輻輳通知（ECN）、パイプモードでのみ。
- Cisco Nexus 96136YC-RおよびCisco Nexus 9636C-RXラインカードを搭載したCisco Nexus 9504および9508プラットフォームスイッチでは、MPLS SegmentRoutingとMPLS LDPを同時に設定することはできません。

VXLAN-to-SR ハンドオフ QoS 値は、ハンドオフ中に保持され、Cisco Nexus 9336C-FX2、93240YC-FX2、9300-FX3、および 9300-GX プラットフォーム スイッチの VXLAN トンネルパケットから SR トンネル パケットに伝播されます。

Cisco NX-OS リリース 10.2(3)F 以降、VXLAN-to-SR ハンドオフ QoS 値は、ハンドオフ中に保持され、Cisco Nexus 9300-GX2 プラットフォーム スイッチの VXLAN トンネル パケットから SR トンネル パケットに伝達されます。

Cisco NX-OS リリース 10.4(1)F 以降、VXLAN-to-SR ハンドオフ QoS 値は、ハンドオフ中に保持され、Cisco Nexus 9300-GX2 プラットフォーム スイッチの VXLAN トンネル パケットから SR トンネル パケットに伝達されます。

Cisco NX-OS リリース 10.4(2)F 以降、VXLAN-to-SR ハンドオフ QoS 値は、ハンドオフ中に保持され、Cisco Nexus 9300-GX2 プラットフォーム スイッチの VXLAN トンネル パケットから SR トンネル パケットに伝達されます。

Cisco NX-OS リリース 10.4(3)F 以降、VXLAN-to-SR ハンドオフ QoS 値は、ハンドオフ中に保持され、Cisco Nexus 9300-GX2 プラットフォーム スイッチの VXLAN トンネル パケットから SR トンネル パケットに伝達されます。

次の機能は、EVPN と L3VPN（MPLS SR）のシームレスな統合ではサポートされていません。

- 分散型エニーキャストゲートウェイまたはHSRP、VRRP、GLBPなどのファーストホップ冗長プロトコル。
- 冗長ホストまたはネットワーク サービス接続用の vPC。
- コア方向のアップリンク（MPLSまたはVXLAN）のSVI /サブインターフェイス。
- 設定済みの MAC アドレスをもつ SVI/サブインターフェイス。
- MPLS セグメント ルーティングおよびボーダー ゲートウェイ（VXLAN Multi-SiteのBGW）は同時に構成できません。
- MPLS-SR ドメイン全体にわたる拡張サブネットのレイヤ 2
- Cisco Nexus 9336C-FX2、93240YC-FX2、および9300-FX3 プラットフォーム スイッチ用の VXLAN/SR および SR/VXLAN ハンドオフのドロップなし
- 統計、96136YC-R および 9636C-RX ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチ

- Cisco Nexus 9336C-FX2、93240YC-FX2、9300-FX3、および9300-GX プラットフォーム スイッチのプライオリティ フロー制御（PFC）
- Cisco NX-OS リリース 10.3（1）F 以降、DSCP ベースの SRTE トラフィック ステアリング機能により、IP ヘッダーの DSCP フィールドを使用して照合され、SRTE パスに誘導される VXLAN パケットの送信元ルーティングが可能になります。以下はこの機能の注意事項と制限事項です。
  - このフィーチャは、Cisco Nexus 9300-FX2、9300-FX3、9300-GX、9300-GX2 ToR スイッチでのみサポートされます。
  - 境界リーフまたは境界 PE の場合、ACL フィルタは内部パケットに適用されます（IPv4 パケットの場合は IPv4 アクセス リスト、IPv6 パケットの場合は IPv6 アクセス リスト）。この機能は、L3VPN ではサポートされていません。MPLS EVPN は、VXLAN ではサポートされていません。
- Cisco NX-OS リリース 10.3(2)F 以降、EVPN と L3VPN（MPLS SR）のシームレスな統合が Cisco Nexus 9300-FX プラットフォーム スイッチおよび Cisco Nexus 9700-FX と 9700-GX ライン カードでサポートされています。以下はこの機能の注意事項と制限事項です。
  - Cisco Nexus 9500 プラットフォーム スイッチがハンドオフ モードで、MPLS カプセル化パケットが L2 ポートで転送される場合、dot1q ヘッダーは追加されません。
  - Cisco Nexus 9500 プラットフォーム スイッチが EVPN から MPLS SR L3VPN へのハンドオフ モードとして設定されている場合、SVI/サブインターフェイスは、コアに面したアップリンク (MPLS または VXLAN) ではサポートされません。
  - DSCP から MPLS EXP へのプロモーションは、DCI モードの FX TOR/ラインカードでは機能しません。MPLS EXP への内部 DSCP 値のコピーは、このハンドオフ モードの FX TOR/ライン カードでは機能しません。MPLS EXP は 0x7 に設定されます。
- Cisco NX-OS リリース 10.3(2)F 以降、DSCP ベースの SRTE フロー ステアリング機能は、Cisco Nexus 9300-FX プラットフォームおよび Cisco Nexus 9700-FX と 9700-GX ライン カードでサポートされます。以下はこの機能の注意事項と制限事項です。
  - Cisco Nexus 9500 プラットフォーム スイッチがハンドオフ モードで、MPLS カプセル化パケットが L2 ポートで転送される場合、dot1q ヘッダーは追加されません。
  - Cisco Nexus 9500 プラットフォーム スイッチが EVPN から MPLS SR L3VPN へのハンドオフ モードとして設定されている場合、SVI/サブインターフェイスは、コアに面したアップリンク (MPLS または VXLAN) ではサポートされません。
  - DSCP から MPLS EXP へのプロモーションは、DCI モードの FX TOR/ラインカードでは機能しません。MPLS EXP への内部 DSCP 値のコピーは、このハンドオフ モードの FX TOR/ライン カードでは機能しません。MPLS EXP は 0x7 に設定されます。
- Cisco NX-OS リリース 10.4(3)F 以降、X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチは、システム レベル QoS でのみ MPLS SR QoS 機能

EVPN と L3VPN (MPLS SR) のシームレスな統合の設定

をサポートし、インターフェイス レベル QoS はサポートしません。さらに次の制限があります。

- デフォルトのパイプ モードがサポートされているため、内部パケットの DSCP または優先順位が保持されます。
- システム レベル QoS ポリシーマップで MPLS エクスペリメンタル ビットを設定する場合、次の一致基準がサポートされます。
  - 一致の DSCP
  - 一致の優先順位
- システム レベルの QoS では、次の機能はサポートされていません。
  - ポリシング
  - ポリシーマップ統計情報
  - MPLS EXP から DSCP へのプロモーション
- インターフェイス レベルの QoS では、MPLS カプセル化を使用したポリシーはサポートされません。
- インターフェイス レベルの QoS ポリシーは、システム レベルの QoS ポリシーよりも優先されます。インターフェイス ポリシーのどの基準にも一致しないトラフィックは、システム レベル QoS のデフォルト プロファイルによって処理されます。
- MPLS インターフェイスのキューイング統計情報で、誤って「UC ECN Mark pkts」と表示される場合があります。

EVPN と L3VPN (MPLS SR) のシームレスな統合の設定

Border Provider Edge (Border PE) の次の手順では、VXLAN ドメインから MPLS ドメインへのルートをインポートして、他の方向へのルートを再開始します。

手順の概要

1. **configure terminal**
2. **feature-set mpls**
3. **nv overlay evpn**
4. **feature bgp**
5. **feature mpls l3vpn**
6. **feature mpls segment-routing**
7. **feature interface-vlan**
8. **feature vn-segment-vlan-based**
9. **feature nv overlay**
10. **router bgp** *autonomous-system-number*

EVPN と L3VPN (MPLS SR) のシームレスな統合の設定

11. **address-family ipv4 unicast**
12. **network** *address*
13. **allocate-label all**
14. **exit**
15. **neighbor** *address* **remote-as** *number*
16. **update-source** *type/id*
17. **address-family l2vpn evpn**
18. **send-community both**
19. **import vpn unicast reoriginate**
20. **exit**
21. **neighbor** *address* **remote-as** *number*
22. **update-source** *type/id*
23. **address-family ipv4 labeled-unicast**
24. **send-community both**
25. **exit**
26. **neighbor** *address* **remote-as** *number*
27. **update-source** *type/id*
28. **ebgp-multihop** *number*
29. **address-family vpnv4 unicast**
30. **send-community both**
31. **import l2vpn evpn reoriginate**
32. **exit**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**feature-set mpls** 例： `switch(config)# feature-set mpls`	MPLS フィーチャ セットをイネーブルにします。
ステップ 3	**nv overlay evpn** 例： `switch(config)# nv overlay evpn`	VXLAN を有効にします。
ステップ 4	**feature bgp** 例： `switch(config)# feature bgp`	BGP を有効にします。

EVPN と L3VPN (MPLS SR) のシームレスな統合の設定

	コマンドまたはアクション	目的
ステップ 5	**feature mpls l3vpn** 例： `switch(config)# feature mpls l3vpn`	レイヤ 3 VPN を有効にします。 （注） 機能 mpls l3vpn は機能 mpls segment-routing を必要とします。
ステップ 6	**feature mpls segment-routing** 例： `switch(config)# feature mpls segment-routing`	セグメント ルーティングを有効にします。
ステップ 7	**feature interface-vlan** 例： `switch(config)# feature interface-vlan`	VLAN インターフェイスを有効にします。
ステップ 8	**feature vn-segment-vlan-based** 例： `switch(config)# feature vn-segment-vlan-based`	VLAN ベースの VN セグメントを有効にします
ステップ 9	**feature nv overlay** 例： `switch(config)# feature nv overlay`	VXLAN を有効にします。
ステップ 10	**router bgp** *autonomous-system-number* 例： `switch(config)# router bgp 65095`	BGP を設定します。*autonomous-system-number* の値は 1ー4294967295 です。
ステップ 11	**address-family ipv4 unicast** 例： `switch(config-router)# address-family ipv4 unicast`	IPv4 のアドレス ファミリを設定します。
ステップ 12	**network** *address* 例： `switch(config-router-af)# network 10.51.0.51/32`	MPLS-SR ドメイン向けに BGP にプレフィックスを挿入します。 （注） Border PE での MPLS-SR トンネルデポジションのすべての実行可能なネクストホップは、network ステートメントを介してアドバタイズする必要があります（/32 のみ）。
ステップ 13	**allocate-label all** 例： `switch(config-router-af)# allocate-label all`	network ステートメントによって挿入されたすべてのプレフィックスのラベル割り当てを設定します。

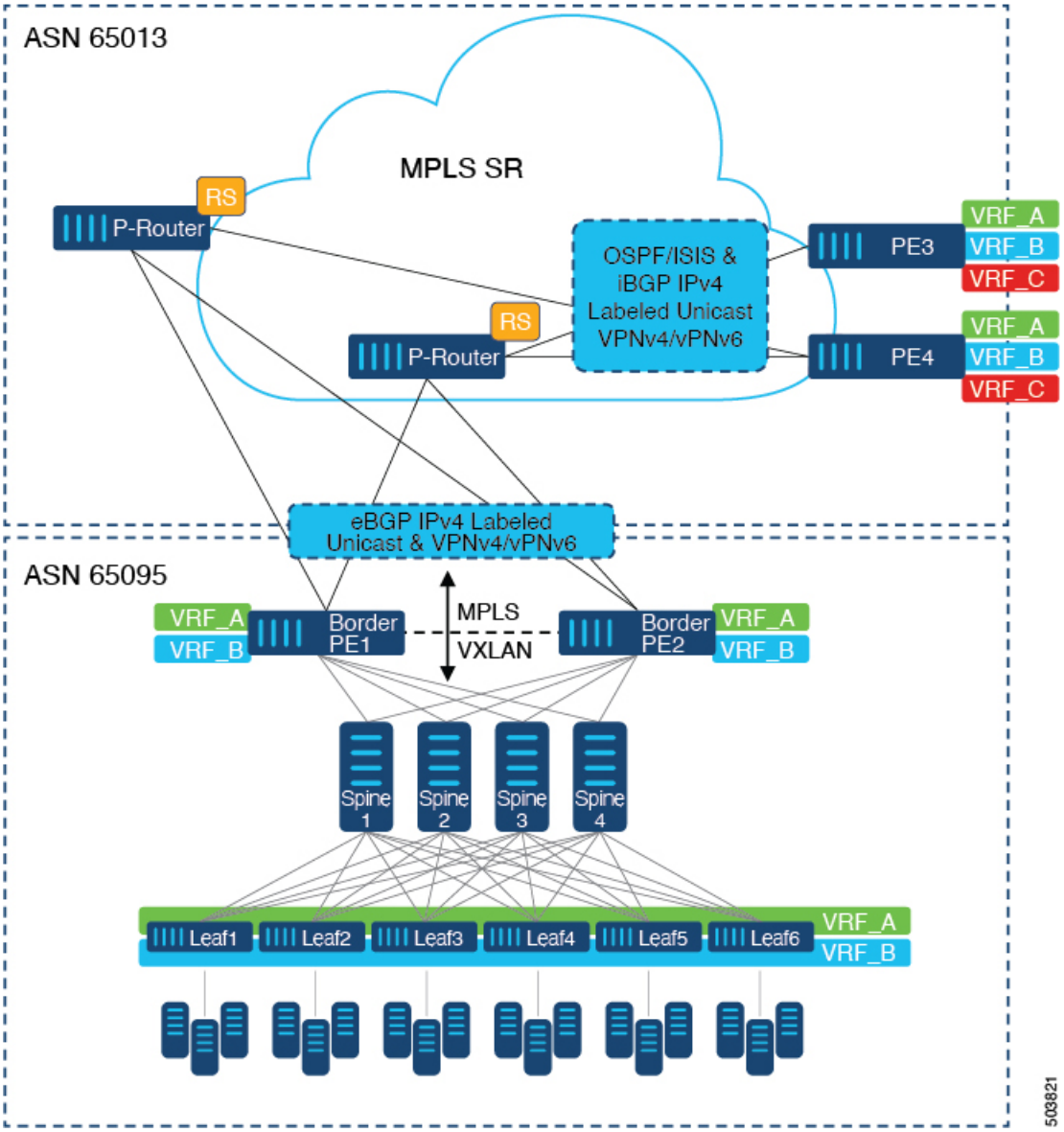
	コマンドまたはアクション	目的
ステップ 14	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 15	**neighbor** *address* **remote-as** *number* 例： `switch(config-router)# neighbor 10.95.0.95 remote-as 65095`	ルート リフレクター—に対して iBGP ネイバーの IPv4 アドレスおよびリモート自律システム（AS）番号を定義します。
ステップ 16	**update-source** *type/id* 例： `switch(config-router)# update-source loopback0`	eBGP ピアリングのインターフェイスを定義します。
ステップ 17	**address-family l2vpn evpn** 例： `switch(config-router)# address-family l2vpn evpn`	L2VPN EVPN キャスト アドレス ファミリを設定します。
ステップ 18	**send-community both** 例： `switch(config-router-af)# send-community both`	BGP ネイバーのコミュニティを設定します。
ステップ 19	**import vpn unicast reoriginate** 例： `switch(config-router-af)# import vpn unicast reoriginate`	新しい Route-Target でルートを再発信します。オプションのルートマップを使用するように拡張できます。
ステップ 20	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 21	**neighbor** *address* **remote-as** *number* 例： `switch(config-router)# neighbor 10.51.131.131 remote-as 65013`	P ルーター—に対して eBGP ネイバーの IPv4 アドレスおよびリモート自律システム（AS）番号を定義します。
ステップ 22	**update-source** *type/id* 例： `switch(config-router)# update-source Ethernet1/1`	eBGP ピアリングのインターフェイスを定義します。
ステップ 23	**address-family ipv4 labeled-unicast** 例： `switch(config-router)# address-family ipv4 labeled-unicast`	IPv4 ラベル付きユニキャストのアドレス ファミリを設定します。

	コマンドまたはアクション	目的
ステップ 24	**send-community both** 例： `switch(config-router-af)# send-community both`	BGP ネイバーのコミュニティを設定します。
ステップ 25	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 26	**neighbor** *address* **remote-as** *number* 例： `switch(config-router)# neighbor 10.131.0.131 remote-as 65013`	eBGP ネイバーの IPv4 アドレスおよびリモート自律システム（AS）番号を定義します。
ステップ 27	**update-source** *type/id* 例： `switch(config-router)# update-source loopback0`	eBGP ピアリングのインターフェイスを定義します。
ステップ 28	**ebgp-multihop** *number* 例： `switch(config-router)# ebgp-multihop 5`	リモート ピアにマルチホップ TTL を指定します。*number* の範囲は 2 ～ 255 です。
ステップ 29	**address-family vpnv4 unicast** 例： `switch(config-router)# address-family vpnv4 unicast`	VPNv4 または VPNv6 のアドレス ファミリを設定します。
ステップ 30	**send-community both** 例： `switch(config-router-af)# send-community both`	BGP ネイバーのコミュニティを設定します。
ステップ 31	**import l2vpn evpn reoriginate** 例： `switch(config-router-af)# import l2vpn evpn reoriginate`	新しい Route-Target でルートを再発信します。オプションのルートマップを使用するように拡張できます。
ステップ 32	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。

EVPN と L3VPN (MPLS SR) のシームレスな統合の設定 の設定例

シナリオ：**DC to Core Network Domain Separation**および**IGP with MPLS-SR network**

図 *23 : DC*からコアネットワークドメインへの分離を使用したトポロジ



次に示すのは、VXLAN ドメインから MPLS ドメインへ、および逆方向にルートをインポートおよび再発信するために必要な CLI 設定の例です。サンプルCLI設定は、それぞれのロールに必要な設定のみを示しています。

ボーダー PE

```
hostname BL51-N9336FX2
install feature-set mpls

feature-set mpls

feature bgp
feature mpls l3vpn
feature mpls segment-routing
feature ospf
feature interface-vlan
feature vn-segment-vlan-based
feature nv overlay

nv overlay evpn

mpls label range 16000 23999 static 6000 8000

segment-routing
  mpls
    connected-prefix-sid-map
      address-family ipv4
        10.51.0.51/32 index 51

vlan 2000
  vn-segment 50000

vrf context VRF_A
  vni 50000
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
    route-target import 50000:50000
    route-target export 50000:50000
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn
    route-target import 50000:50000
    route-target export 50000:50000

interface Vlan2000
  no shutdown
  vrf member VRF_A
  no ip redirects
  ip forward
  ipv6 address use-link-local-only
  no ipv6 redirects

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 50000 associate-vrf

interface Ethernet1/1
  description TO_P-ROUTER
  ip address 10.51.131.51/24
  mpls ip forwarding
  no shutdown

interface Ethernet1/36
```

```
  description TO_SPINE
  ip address 10.95.51.51/24
  ip router ospf 10 area 0.0.0.0
  no shutdown

interface loopback0
  description ROUTER-ID & SR-LOOPBACK
  ip address 10.51.0.51/32
  ip router ospf UNDERLAY area 0.0.0.0

interface loopback1
  description NVE-LOOPBACK
  ip address 10.51.1.51/32
  ip router ospf UNDERLAY area 0.0.0.0

router ospf UNDERLAY
  router-id 10.51.0.51

router bgp 65095
  address-family ipv4 unicast
    network 10.51.0.51/32
    allocate-label all
!
  neighbor 10.95.0.95
    remote-as 65095
    update-source loopback0
    address-family l2vpn evpn
      send-community
      send-community extended
      import vpn unicast reoriginate
!
  neighbor 10.51.131.131
    remote-as 65013
    update-source Ethernet1/1
    address-family ipv4 labeled-unicast
      send-community
      send-community extended
!
  neighbor 10.131.0.131
    remote-as 65013
    update-source loopback0
    ebgp-multihop 5
    address-family vpnv4 unicast
      send-community
      send-community extended
      import l2vpn evpn reoriginate
    address-family vpnv6 unicast
      send-community
      send-community extended
      import l2vpn evpn reoriginate

!
  vrf VRF_A
    address-family ipv4 unicast
      redistribute direct route-map fabric-rmap-redist-subnet
```

P ルーター

```
hostname P131-N9336FX2
install feature-set mpls

feature-set mpls

feature bgp
feature isis
```

```
feature mpls l3vpn
feature mpls segment-routing

mpls label range 16000 23999 static 6000 8000

segment-routing
  mpls
    connected-prefix-sid-map
      address-family ipv4
        10.131.0.131/32 index 131

route-map RM_NH_UNCH permit 10
  set ip next-hop unchanged

interface Ethernet1/1
  description TO_BORDER-PE
  ip address 10.51.131.131/24
  ip router isis 10
  mpls ip forwarding
  no shutdown

interface Ethernet1/11
  description TO_PE
  ip address 10.52.131.131/24
  ip router isis 10
  mpls ip forwarding
  no shutdown

interface loopback0
  description ROUTER-ID & SR-LOOPBACK
  ip address 10.131.0.131/32
  ip router isis 10

router isis 10
  net 49.0000.0000.0131.00
  is-type level-2
  address-family ipv4 unicast
    segment-routing mpls

router bgp 65013
  event-history detail
  address-family ipv4 unicast
    allocate-label all
!
  neighbor 10.51.131.51
    remote-as 65095
    update-source Ethernet1/1
    address-family ipv4 labeled-unicast
      send-community
      send-community extended
!
  neighbor 10.51.0.51
    remote-as 65095
    update-source loopback0
    ebgp-multihop 5
    address-family vpnv4 unicast
      send-community
      send-community extended
      route-map RM_NH_UNCH out
    address-family vpnv6 unicast
      send-community
      send-community extended
      route-map RM_NH_UNCH out
!
```

```
  neighbor 10.52.131.52
    remote-as 65013
    update-source Ethernet1/11
    address-family ipv4 labeled-unicast
      send-community
      send-community extended
!
  neighbor 10.52.0.52
    remote-as 65013
    update-source loopback0
    address-family vpnv4 unicast
      send-community
      send-community extended
      route-reflector-client
      route-map RM_NH_UNCH out
    address-family vpnv6 unicast
      send-community
      send-community extended
      route-reflector-client
      route-map RM_NH_UNCH out
```

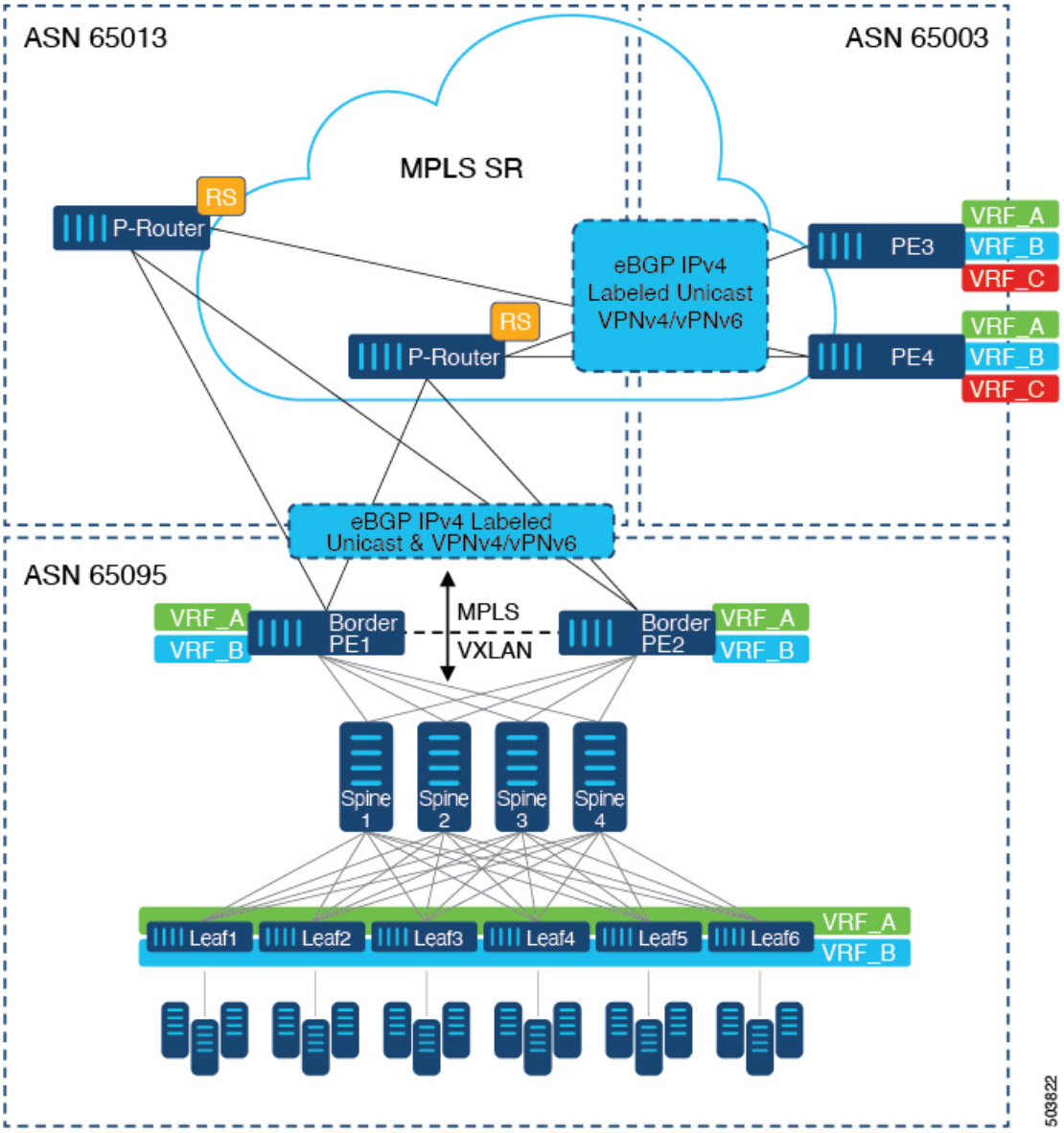
プロバイダー エッジ（PE）

```
hostname L52-N93240FX2
install feature-set mpls

feature-set mpls

feature bgp
feature isis
feature mpls l3vpn
feature mpls segment-routing

mpls label range 16000 23999 static 6000 8000

segment-routing
  mpls
    connected-prefix-sid-map
      address-family ipv4
        10.52.0.52/32 index 52

vrf context VRF_A
  rd auto
  address-family ipv4 unicast
    route-target import 50000:50000
    route-target export 50000:50000
  address-family ipv6 unicast
    route-target import 50000:50000
    route-target export 50000:50000

interface Ethernet1/49
  description TO_P-ROUTER
  ip address 10.52.131.52/24
  ip router isis 10
  mpls ip forwarding
  no shutdown

interface loopback0
  description ROUTER-ID & SR-LOOPBACK
  ip address 10.52.0.52/32
  ip router isis 10

router isis 10
  net 49.0000.0000.0052.00
  is-type level-2
```

```
  address-family ipv4 unicast
    segment-routing mpls

router bgp 65013
  address-family ipv4 unicast
    network 10.52.0.52/32
    allocate-label all
!
  neighbor 10.52.131.131
    remote-as 65013
    update-source Ethernet1/49
    address-family ipv4 labeled-unicast
      send-community
      send-community extended
!
  neighbor 10.131.0.131
    remote-as 65013
    update-source loopback0
    address-family vpnv4 unicast
      send-community
      send-community extended
    address-family vpnv6 unicast
      send-community
      send-community extended
!
  vrf VRF_A
    address-family ipv4 unicast
      redistribute direct route-map fabric-rmap-redist-subnet
```

シナリオ：DCからコアへ、およびコアネットワークドメイン分離内（MPLS-SRネットワーク内のeBGP）。

図 *24* : コアネットワーク内の複数の管理ドメイン



次に示すのは、VXLAN ドメインから MPLS ドメインへ、および逆方向にルートをインポートおよび再発信するために必要な CLI 設定の例です。サンプル CLI 構成は、シナリオ 1 とは異なるノード（P-Router ロールと Provider Edg（PE）ロール）のみを示しています。ボーダーPEは両方のシナリオで同じままです。

P ルーター

```
hostname P131-N9336FX2
install feature-set mpls

feature-set mpls

feature bgp
feature mpls l3vpn
```

```
feature mpls segment-routing

mpls label range 16000 23999 static 6000 8000

segment-routing
  mpls
    connected-prefix-sid-map
      address-family ipv4
        10.131.0.131/32 index 131

route-map RM_NH_UNCH permit 10
  set ip next-hop unchanged

interface Ethernet1/1
  description TO_BORDER-PE
  ip address 10.51.131.131/24
  mpls ip forwarding
  no shutdown

interface Ethernet1/11
  description TO_PE
  ip address 10.52.131.131/24
  mpls ip forwarding
  no shutdown

interface loopback0
  description ROUTER-ID & SR-LOOPBACK
  ip address 10.131.0.131/32
  ip router isis 10

router bgp 65013
  event-history detail
  address-family ipv4 unicast
    network 10.131.0.131/32
    allocate-label all
!
  address-family vpnv4 unicast
    retain route-target all
  address-family vpnv6 unicast
    retain route-target all
!
  neighbor 10.51.131.51
    remote-as 65095
    update-source Ethernet1/1
    address-family ipv4 labeled-unicast
      send-community
      send-community extended
!
  neighbor 10.51.0.51
    remote-as 65095
    update-source loopback0
    ebgp-multihop 5
    address-family vpnv4 unicast
      send-community
      send-community extended
      route-map RM_NH_UNCH out
    address-family vpnv6 unicast
      send-community
      send-community extended
      route-map RM_NH_UNCH out
!
  neighbor 10.52.131.52
    remote-as 65003
    update-source Ethernet1/11
```

```
    address-family ipv4 labeled-unicast
      send-community
      send-community extended
!
  neighbor 10.52.0.52
    remote-as 65003
    update-source loopback0
    ebgp-multihop 5
    address-family vpnv4 unicast
      send-community
      send-community extended
      route-map RM_NH_UNCH out
    address-family vpnv6 unicast
      send-community
      send-community extended
      route-map RM_NH_UNCH out
```

プロバイダー エッジ（PE）

```
hostname L52-N93240FX2
install feature-set mpls

feature-set mpls

feature bgp
feature mpls l3vpn
feature mpls segment-routing

mpls label range 16000 23999 static 6000 8000

segment-routing
  mpls
    connected-prefix-sid-map
      address-family ipv4
        10.52.0.52/32 index 52

vrf context VRF_A
  rd auto
  address-family ipv4 unicast
    route-target import 50000:50000
    route-target export 50000:50000
  address-family ipv6 unicast
    route-target import 50000:50000
    route-target export 50000:50000

interface Ethernet1/49
  description TO_P-ROUTER
  ip address 10.52.131.52/24
  mpls ip forwarding
  no shutdown

interface loopback0
  description ROUTER-ID & SR-LOOPBACK
  ip address 10.52.0.52/32
  ip router isis 10

router bgp 65003
  address-family ipv4 unicast
    network 10.52.0.52/32
    allocate-label all
!
  neighbor 10.52.131.131
    remote-as 65013
    update-source Ethernet1/49
    address-family ipv4 labeled-unicast
```

```
      send-community
      send-community extended
!
  neighbor 10.131.0.131
    remote-as 65013
    update-source loopback0
    ebgp-multihop 5
    address-family vpnv4 unicast
      send-community
      send-community extended
    address-family vpnv6 unicast
      send-community
      send-community extended
!
  vrf VRF_A
    address-family ipv4 unicast
      redistribute direct route-map fabric-rmap-redist-subnet
```

DSCP ベースの SR-TE フロー ステアリングの構成

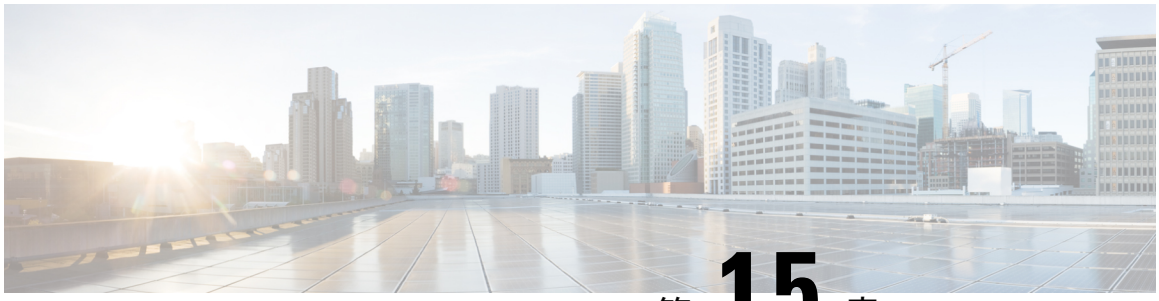
DSCP ベースの SR-TE フロー ステアリングを構成するには、まず境界 PE または境界リーフを構成して、EVPN と L3VPN をシームレスに統合します。EVPN と L3VPN (MPLS SR) のシームレスな統合の設定 （283 ページ） を参照してください。次に、トラフィックを誘導するには、次の構成を実行します。

1. SRTE ポリシーを構成します。Cisco ポータルにある *Cisco Nexus 9000* シリーズ *NX-OS* ラベル スイッチング構成ガイドのセグメント ルーティングの構成の章記載の、構成プロセス：*SRTE* フローベース トラフィック ステアリングのセクションを参照してください。
2. L3 VNI インターフェイスを構成します。新しい L3VNI モードの構成を参照してください。
3. **ip/ipv6 policy route-map** *srte-policy* コマンドを使用して、L3 VNI インターフェイスにポリシーを適用します。

DSCP ベースの SR-TE フロー ステアリングの構成例

```
segment-routing
  traffic-engineering
    segment-list name PATH1
      index 50 mpls label 16100
    segment-list name PATH2
      index 50 mpls label 16500
      index 100 mpls label 16100

    policy blue
      color 202 endpoint 21.1.1.1
      candidate-paths
        preference 100
          explicit segment-list PATH2
    policy red
      color 201 endpoint 21.1.1.1
      candidate-paths
        preference 100
          explicit segment-list PATH1
ip access-list flow-1
```

```
  statistics per-entry
  5 permit ip any any dscp af11
ip access-list flow-2
  statistics per-entry
  5 permit ip any any dscp af12

route-map srte-flow1 permit 10
  match ip address flow-1
  set ip next-hop 61.1.1.1 srte-policy name red

route-map srte-flow1 permit 20
  match ip address flow-2
  set ip next-hop 61.1.1.1 srte-policy name blue

vrf context 501
  vni 90001 l3

interface vni90001
  ip policy route-map srte-flow1
```



第 15 章

L3VPN SRv6 を備えた EVPN のシームレスな統合の設定

この章は、次の項で構成されています。

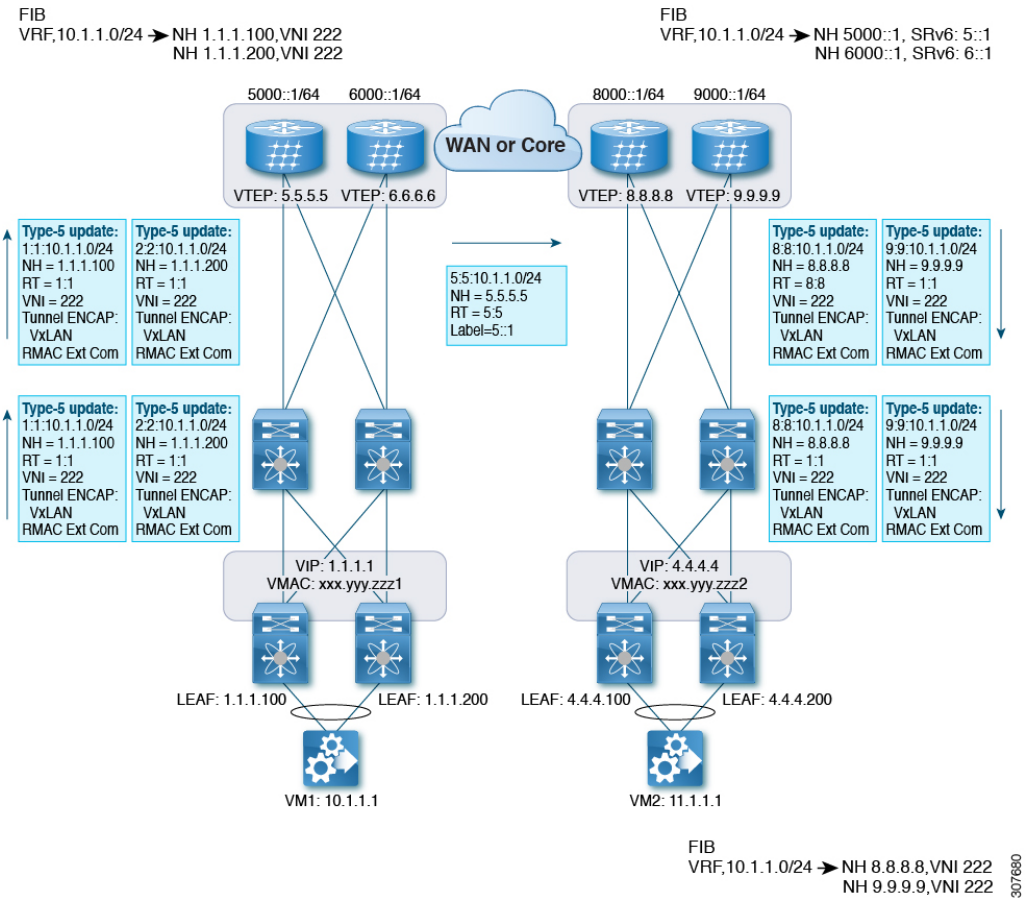
L3VPN を備えた EVPN のハンドオフのシームレスな統合について

データセンター（DC）導入では、EVPN コントロール プレーン ラーニング、マルチテナンシー、シームレス モビリティ、冗長性、POD の追加が容易になるなどの利点から、VXLAN EVPN を採用しています。同様に、コアは IP ベースの L3VPN SRv6 ネットワークであるか、IPv6 ベースの L3VPN アンダーレイから IPv6 用の IPv6 セグメント ルーティング（SRv6）のようなより高度なソリューションに移行しています。SRv6 には次のような利点があります。

- よりシンプルなトラフィック エンジニアリング（TE）方式
- より簡単に行えるクライアント設定
- SDN の採用

データセンター（DC）内とコア内の 2 つの異なるテクノロジーにより、VXLAN から SRv6 コアへのトラフィック ハンドオフがあり、これは DCI ノードで必要になり、DC ドメインのエッジにあり、コア エッジ ルータとインターフェイスします。

図 *25 : BGP EVPN VXLAN* から *L3VPN SRv6* へのハンドオフ



EVPN-VxLAN ファブリックに入るトラフィックの場合、BGP EVPN ルートは VRF の RD を含むローカル VRF にインポートされます。最適パスが計算され、VRF の RIB にインストールされた後、L3VPN SRv6 テーブルに挿入されます。最適パスとともに、VRF の RD および VRF ごとの SRv6 SID が含まれます。L3VPN SRv6 ルート ターゲットは、L3VPN SRv6 ピアにアドバタイズされるルートとともに送信されます。

EVPN VxLAN ファブリックから出力されるトラフィックの場合、BGP L3VPN SRv6 ルートは、VRF の RD を含むローカル VRF にインポートされます。最適パスが計算されて VRF の RIB にインストールされ、EVPN テーブルに挿入されます。最適パスとともに、VRF の RD および VNI が含まれます。EVPN-VXLAN ルート ターゲットはルートとともに送信され、EVPN-VxLAN ピアにアドバタイズされます。

EVPN から L3VPN SRv6 へのハンドオフの注意事項と制限事項

この機能には、次の注意事項と制約事項があります。

- 同じ RD インポートが L3VPN SRV6 ファブリックでサポートされます。
- 同じ RD インポートは、EVPN VXLAN ファブリックではサポートされません。
- ハンドオフ デバイスでは、EVPN VXLAN 側で同じ RD インポートを使用しないでください。
- Cisco NX-OS Release 9.3(3) 以降では、次のスイッチのサポートが追加されています。
  - Cisco Nexus C93600CD-GX
  - Cisco Nexus C9364C-GX
  - Cisco Nexus C9316D-GX
- Cisco NX-OS リリース 10.2(1q)F 以降、SRv6 DCI ハンドオフは Cisco Nexus 9332D-GX2B プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、SRv6 DCI ハンドオフは Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、SRv6 DCI ハンドオフは Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、SRv6 DCI ハンドオフは Cisco Nexus 9364C-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.2(3)F 以降、EVPN から L3VPN SRv6 へのハンドオフは、Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、 EVPN からの L3VPN SRv6 ハンドオフは Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、 EVPN からの L3VPN SRv6 ハンドオフは Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、 EVPN からの L3VPN SRv6 ハンドオフは Cisco Nexus 9364C-H1 スイッチでサポートされます。

EVPN VXLAN への L3VPN SRv6 ルートのインポート

L3VPN SRv6 ドメインから EVPN VXLAN ファブリックにルートを渡すプロセスでは、L3VPN SRv6 ルートのインポート条件を設定する必要があります。ルートは IPv4 または IPv6 のいずれかです。このタスクでは、EVPN VXLAN ファブリックへの単方向ルート アドバタイズメントを設定します。双方向アドバタイズメントの場合、L3VPN SRv6 ドメインのインポート条件を明示的に設定する必要があります。

始める前に

L3VPN SRv6 ファブリックが完全に設定されていることを確認します。詳細については、『*Cisco Nexus 9000 Series NX-OS SRv6 Configuration Guide*』を参照してください。

手順の概要

1. **config terminal**
2. **router bgp** *as-number*
3. **neighbor bgp** *ipv6-address* **remote-as** *as-number*
4. **address family vpnv4 unicast** または **address family vpnv6 unicast**
5. **import l2vpn evpn route-map** *name* **[reoriginate]**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch-1# config terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch-1(config)#`
		コンフィギュレーション モードを入力します。

ステップ 2	**router bgp** *as-number*
	例：
	`switch-1(config)# router bgp 100`
	`switch-1(config-router)#`
		BGP ルータ 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	**neighbor bgp** *ipv6-address* **remote-as** *as-number*
	例：
	`switch-1(config-router)# neighbor 1234::1 remote-as 200`
	`switch-1(config-router-neighbor)#`
		BGP ルータ コンフィギュレーション モードを開始します。

ステップ 4	**address family vpnv4 unicast** または **address family vpnv6 unicast**
	例：
	`switch-1(config-router-neighbor)# address-family vpnv4 unicast`
	`switch-1(config-router-neighbor-af)#`
	例：
	`switch-1(config-router-neighbor)# address-family vpnv6 unicast`
	`switch-1(config-router-neighbor-af)#`
		EVPN VXLAN が L3VPN SRv6 にハンドオフするユニキャスト トラフィックの IPv4 または IPv6 アドレス ファミリを設定します。

	コマンドまたはアクション	目的

ステップ 5	**import l2vpn evpn route-map** *name* **[reoriginate]**
	例：
	`switch-1(config-router-neighbor-af)# import l2vpn evpn route-map test reoriginate`
	`switch-1(config-router-neighbor-af)#`
		EVPN VXLAN が L3VPN SRv6 にハンドオフするユニキャスト トラフィックの IPv4 または IPv6 アドレス ファミリを設定します。このコマンドは、L3VPN SRv6 ドメインから学習したルートを EVPN VXLAN ドメインにアドバタイズできるようにします。オプションの **reoriginate** キーワードを使用すると、ドメイン固有の RT だけがアドバタイズされます。

次のタスク

双方向ルートアドバタイズメントでは、EVPN VXLAN ルートを L3VPN SRv6 ドメインにインポートするように設定します。

L3VPN SRv6 への EVPN VXLAN ルートのインポート

EVPN VXLAN ファブリックから L3VPN SRv6 ドメインにルートを渡すプロセスでは、EVPN VXLAN ルートのインポート条件を設定する必要があります。ルートは IPv4 または IPv6 のいずれかです。このタスクでは、L3VPN SRv6 ファブリックへの単方向ルート アドバタイズメントを設定します。双方向アドバタイズメントの場合、EVPN VXLAN ドメインのインポート条件を明示的に設定する必要があります。

始める前に

L3VPN SRv6 ファブリックが完全に設定されていることを確認します。詳細については、『*Cisco Nexus 9000 Series NX-OS SRv6 Configuration Guide*』を参照してください。

手順の概要

1. **config terminal**
2. **router bgp** *as-number*
3. **neighbor** *ipv6-address* **remote-as** *as-number*
4. **address-family l2vpn evpn**
5. **import vpn unicast route-map** *name* **[reoriginate]**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**config terminal** 例：	コンフィギュレーション モードを入力します。

	コマンドまたはアクション	目的

	`switch-1# config terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch-1(config)#`	

ステップ 2	**router bgp** *as-number*
	例：
	`switch-1(config)# router bgp 200`
	`switch-1(config-router)#`
		BGP ルータ 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	**neighbor** *ipv6-address* **remote-as** *as-number*
	例：
	`switch-1(config-router)# neighbor 1234::1 remote-as 100`
	`switch-1(config-router-neighbor)#`
		BGP ルータ コンフィギュレーション モードを開始します。

ステップ 4	**address-family l2vpn evpn**
	例：
	`switch(config-router-neighbor)# address-family l2vpn evpn`
	`switch(config-router-neighbor-af)#`
		EVPN VXLAN が L3VPN SRv6 にハンドオフするユニキャスト トラフィックのアドレス ファミリを設定します。

ステップ 5	**import vpn unicast route-map** *name* **[reoriginate]**
	例：
	`switch-1(config-router-neighbor-af)# import vpn unicast route-map test reoriginate`
	`switch-1(config-router-neighbor-af)#`
		EVPN VXLAN が L3VPN SRv6 にハンドオフするユニキャスト トラフィックの IPv4 または IPv6 アドレス ファミリを設定します。このコマンドは、EVPN VXLAN ドメインから学習したルートを L3VPN SRv6 ドメインにアドバタイズできるようにします。オプションの **reoriginate** キーワードを使用すると、ドメイン固有の RT だけがアドバタイズされます。

次のタスク

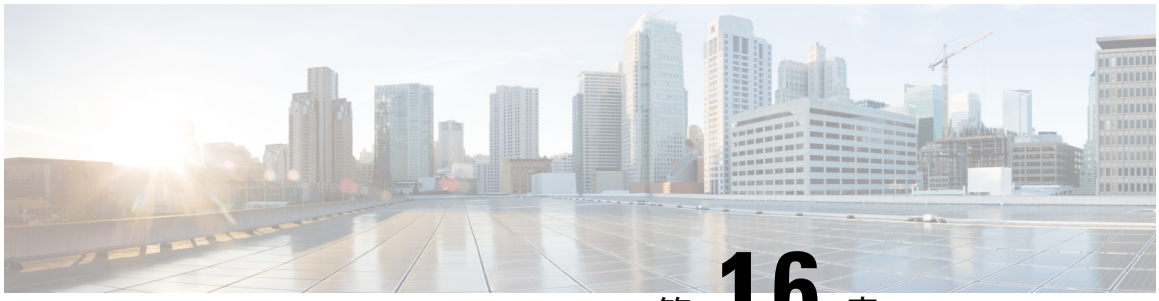
双方向ルートアドバタイズメントの場合、EVPN VXLAN ファブリックへの L3VPN SRv6 ルートのインポートを設定します。

VXLAN EVPN から L3VPN SRv6 へのハンドオフの設定例

```
feature vn-segment-vlan-based
feature nv overlay
feature interface-vlan
nv overlay evpn
feature srv6

vrf context customer1
    vni 10000
    rd auto
    address-family ipv4 unicast
     route-target both 1:1
```

```
     route-target both auto evpn
    address-family ipv6 unicast
     route-target both 1:1
     route-target both auto evpn

segment-routing
  srv6
    encapsulation
     source-address loopback1
    locators
     locator DCI_1
      prefix café:1234::/64

interface loopback0
  ip address 1.1.1.0/32

interface loopback1
  ip address 1.1.1.1/32
  ipv6 address 4567::1/128

interface nve1
  source-interface loopback0
  member vni 10000 associate-vrf
  host-reachability protocol bgp

vlan 100
  vn-segment 10000

interface vlan 100
 ip forward
 ipv6 address use-link-local-only
 vrf member customer1

router bgp 65000
   segment-routing srv6
    locator DCI_1
   neighbor 2.2.2.2 remote-as 200
      remote-as 75000
      address-family l2vpn evpn
      import vpn route-map | reoriginate
   neighbor 1234::1 remote-as 100
      remote-as 65000
      address-family vpnv4 unicast
      import l2vpn evpn route-map | reoriginate
      address-family vpnv6 unicast
      import l2vpn evpn route-map | reoriginate

vrf customer
  segment-routing srv6
  alloc-mode per-vrf
  address-family ipv4 unicast
  address-family ipv6 unicast
```

（注） **vni number** コマンドでは、VRF での VNI の構成中に **L3** キーワードを使用しないでください。新しい L3 VNI 構成は、ダイナミックに割り当てられる VNI の VLAN-BD ではサポートされないからです。



第 16 章

EVPN (TRM) の MVPN とのシームレスな統合の設定

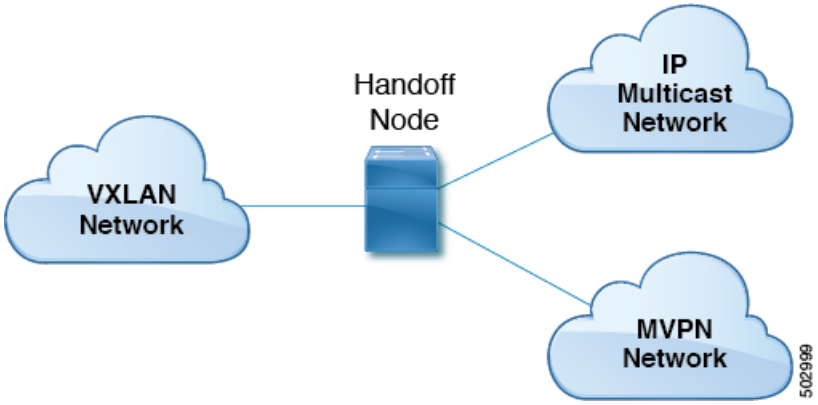
この章は、次の項で構成されています。

EVPN（TRM）の MVPN（Rosen ドラフト）とのシームレスな統合について

EVPN（TRM）と MVPN（ドラフトローゼン）のシームレスな統合により、VXLAN ネットワーク（TRM または TRM マルチサイト）と MVPN ネットワークの間でパケットをハンドオフできます。この機能をサポートするには、VXLAN TRM と MVPN が Cisco Nexus デバイス ノード（ハンドオフ ノード）でサポートされている必要があります。

ハンドオフ ノードは、MVPN ネットワークの PE および VXLAN ネットワークの VTEP です。次の図に示すように、VXLAN、MVPN、および IP マルチキャスト ネットワークに接続します。

図 *26 : VXLAN* : *MVPN* ハンドオフ ネットワーク



送信元と受信者は、3 つのネットワーク（VXLAN、MVPN、または IP マルチキャスト）のいずれかに存在できます。

すべてのマルチキャスト トラフィック（つまり、VXLAN、MVPN、またはマルチキャスト ネットワークからのテナント トラフィック）は、あるドメインから別のドメインにルーティングされます。ハンドオフ ノードは中央ノードとして機能します。必要なパケット転送、カプセル化、およびカプセル化解除を実行して、それぞれの受信者にトラフィックを送信します。

サポートされる RP の位置

カスタマー（オーバーレイ）ネットワークのランデブー ポイント（RP）は、3 つのネットワーク（VXLAN、MVPN、または IP マルチキャスト）のいずれかに配置できます。

表 *6 :* サポートされる *RP* の場所

RP の場所	説明
IP ネットワークの RP	• RP は MVPN PE にのみ接続でき、ハンドオフ ノードには接続できません。 • RPはVXLAN ハンドオフ ノードにのみ接続できます。 • RP は、MVPN PE と VXLAN の両方に接続できます。
VXLAN ファブリック内部の RP	すべての VTEP は、VXLAN ファブリック内の RP です。すべての MVPN PE は、VXLAN ファブリックに設定された RP を使用します。
VXLAN MVPN ハンドオフ ノード上の RP	RP は VXLAN MVPN ハンドオフ ノードです。

RP の場所	説明
MVPN ネットワークの RP	RP は VXLAN ネットワークの外部にあります。これは、ハンドオフ ノード以外の MPLS クラウド内のノードの 1 つで設定されます。
RP Everywhere（PIM エニーキャスト RP または MSDP ベースのエニーキャスト RP）	エニーキャスト RP は VXLAN リーフで設定できます。RP セットは、ハンドオフ ノードまたは任意の MVPN PE で設定できます。

EVPN（TRM）と MVPN とのシームレスな統合に関する注意事項と制約事項

この機能には、次の注意事項と制約事項があります。

- N9K-X9636C-RX ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチのみが、EVPN（TRM）と MVPN とのシームレスな統合をサポートします。その他の -R シリーズ ライン カードは、ハンドオフ ノードとして機能できません。
- ハンドオフ ノードは、カスタマー ネットワークのローカル（直接接続）マルチキャスト送信元または受信者を持つことができます。
- MVPN 用の ASM/SSM や TRM 用の ASM などの既存のアンダーレイ プロパティは、ハンドオフ ノードでサポートされます。
- ハンドオフ ノードは、オーバーレイの PIM SSM および ASM をサポートします。
- Inter-AS オプション A は、IP マルチキャスト ネットワークへのハンドオフ ノードでサポートされます。
- MDT 送信元ループバック IP アドレスと NVE ループバック IP アドレスの数が最大制限を超えると、トラフィックがドロップされる可能性があります。
- 次の機能は、EVPN（TRM）と MVPN のシームレスな統合ではサポートされていません。
  - ハンドオフノードの vPC
  - VXLAN EVPN 入力複製
  - MVPN のコア方向インターフェイスとしての SVI およびサブインターフェイス
  - MVPN ノードの Inter-AS オプション B および C
  - VXLAN アンダーレイとしての PIM SSM
  - アンダーレイまたはオーバーレイとしての双方向 PIM
  - MPLS パスと IP パスが混在する ECMP

- VXLAN、TRM、およびMVPNの既存の制限は、EVPN（TRM）とMVPNのシームレスな統合にも適用されます。

EVPN（TRM）とMVPNとのシームレスな統合のためのハンドオフ ノードの設定

このセクションでは、ハンドオフノードで必要な設定について説明します。他のノード（VXLAN リーフおよびスパイン、MVPN PE、RS/RR など）の設定は、以前のリリースと同じです。

ハンドオフ ノードの PIM/IGMP 設定

ハンドオフ ノードの PIM/IGMP を設定する場合は、次のガイドラインに従ってください。

- 次の例に示すように、ランデブー ポイント（RP）が TRM と MVPN アンダーレイで異なることを確認します。

```
ip pim rp-address 90.1.1.100 group-list 225.0.0.0/8 --- TRM Underlay
ip pim rp-address 91.1.1.100 group-list 233.0.0.0/8 --- MVPN Underlay
```

- オーバーレイ マルチキャスト トラフィックに共通の RP を使用します。
- RP は、静的、PIM エニーキャスト、または PIM MSDP モードにできます。次に、内部 VRF 設定モードを開始する例を示します。

```
vrf context vrfVxLAN5001
  vni 5001
  ip pim rp-address 111.1.1.1 group-list 226.0.0.0/8
  ip pim rp-address 112.2.1.1 group-list 227.0.0.0/8
```

- **ip igmp snooping vxlan** コマンドを使用して、VXLAN トラフィックの IGMP スヌーピングを有効にします。
- すべてのソース インターフェイスおよび PIM トラフィックの伝送に必要なインターフェイスで PIM スパース モードを有効にします。

ハンドオフ ノードの BGP 設定

ハンドオフ ノードの BGP の設定時には、次の注意事項に従ってください。

- すべての VXLAN リーフを L2EVPN および TRM ネイバーとして追加します。冗長ハンドオフ ノードを含めます。ルート リフレクタを使用する場合は、RR だけをネイバーとして追加します。
- すべての MVPN PE を VPN ネイバーとして追加します。MDT モードでは、MVPN PE を MDT ネイバーとして追加します。

- L2EVPN ネイバーから VPN ネイバーにユニキャスト ルートをアドバタイズするための設定をインポートします。
- BGP 送信元識別子は、VTEP 識別子（NVE インターフェイスで設定）/MVPN PE 識別子に使用される送信元インターフェイスとは異なる場合も、同じ場合もあります。

```
feature bgp
address-family ipv4 mdt
address-family ipv4 mvpn

neighbor 2.1.1.1
  address-family ipv4 mvpn
    send-community extended
  address-family l2vpn evpn
    send-community extended
    import vpn unicast reoriginate

neighbor 30.30.30.30
  address-family vpnv4 unicast
    send-community
    send-community extended
    next-hop-self
    import l2vpn evpn reoriginate
  address-family ipv4 mdt
    send-community extended
    no next-hop-third-party
```

- MVPN ピア間で Inter-AS オプション B を使用しないでください。代わりに、VPNv4 ユニキャスト アドレス ファミリで **no allocate-label option-b** コマンドを設定します。

```
address-family vpnv4 unicast
    no allocate-label option-b
```

- 最大パスの設定は EBGP モードで設定する必要があります。

```
address-family l2vpn evpn
    maximum-paths 8
vrf vrfVxLAN5001
    address-family ipv4 unicast
        maximum-paths 8
```

- ハンドオフ ノードがデュアル モードで展開されている場合は、**route-map** コマンドを使用して、VPN アドレス ファミリで孤立したホストに関連付けられているプレフィックスをアドバタイズします。

```
ip prefix-list ROUTES_CONNECTED_NON_LOCAL seq 2 premit 15.14.0.15/32

route-map ROUTES_CONNECTED_NON_LOCAL deny
    match ip address prefix-list ROUTES_CONNECTED_NON_LOCAL

neighbor 8.8.8.8
    remote-as 100
    update-source loopback1
    address-family vpnv4 unicast
      send-community
      send-community extended
      route-map ROUTES_CONNECTED_NON_LOCAL out
```

ハンドオフ ノードの VXLAN 設定

ハンドオフ ノードの VXLAN の設定時には、次の注意事項に従ってください。

- 次の機能をイネーブル化します。

```
feature nv overlay
feature ngmvpn
feature interface-vlan
feature vn-segment-vlan-based
```

- 必要な L3 VNI を設定します。

```
L3VNIs are mapped to tenant VRF.
vlan 2501
  vn-segment 5001 <-- Associate VNI to a VLAN.
```

- NVE インターフェイスを設定します。

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1 <-- This interface should not be the same as the MVPN
source interface.
  global suppress-arp
member vni 5001 associate-vrf <-- L3VNI
  mcast-group 233.1.1.1 <-- The underlay multicast group for VXLAN should be different
 from the MVPN default/data MDT.
```

- テナント VRF を設定します。

```
vrf context vrfVxLAN5001
  vni 5001 <-- Associate VNI to VRF.
  rd auto
address-family ipv4 unicast
    route-target both auto
    route-target both auto mvpn
    route-target both auto evpn

interface Vlan2501 <-- SVI interface associated with the L3VNI
  no shutdown
  mtu 9216 <-- The overlay header requires 58 byes, so the max tenant traffic is
(Configured MTU - 58).
  vrf member vrfVxLAN5001
  no ip redirects
  ip forward
  ipv6 forward
  no ipv6 redirects
  ip pim sparse-mode <-- PIM is enabled.

interface Vlan2 <-- SVI interface associated with L2 VNI
  no shutdown
  vrf member vrfVxLAN5001
  no ip redirects
  ip address 100.1.1.1/16
  no ipv6 redirects
  ip pim sparse-mode <-- PIM enabled on L2VNI
 fabric forwarding mode anycast-gateway
```

ハンドオフ ノードの MVPN 設定

ハンドオフ ノードの MVPN の設定時には、次の注意事項に従ってください。

- 次の機能をイネーブル化します。

```
install feature-set mpls
allow feature-set mpls
feature-set mpls
feature mpls l3vpn
feature mvpn
feature mpls ldp
```

- MPLS LDP 設定
  - MPLS リンクであるすべてのインターフェイスで MPLS LDP（**mpls ip**）を有効にします。
  - VXLAN に使用されるループバック インターフェイスを MPLS プレフィックスとしてアドバタイズしないでください。
    - MVPN PE ノードを識別する IP アドレスを含むプレフィックス リストを設定します。

```
ip prefix-list LDP-LOOPBACK seq 51 permit 9.1.1.10/32
ip prefix-list LDP-LOOPBACK seq 52 permit 9.1.2.10/32
```

    - MVPN PE 識別子に対してのみラベル割り当てを設定します。

```
mpls ldp configuration
  explicit-null
  advertise-labels for LDP-LOOPBACK
  label allocate global prefix-list LDP-LOOPBACK
```

- テナント VRF 設定：
  - デフォルトの MDT モードでは、VRF のすべてのテナント マルチキャスト トラフィックでアンダーレイ マルチキャスト グループを同じにします。

```
vrf context vrfVxLAN5001
  vni 5001
  mdt default 225.1.100.1
  mdt source loopback100 <-- If the source interface is not configured, the BGP
 identifier is used as the source interface.
  mdt asm-use-shared-tree <-- If the underlay is configured in ASM mode
  no mdt enforce-bgp-mdt-safi <-- Enabled by befault but should be negated if
BGP MDT should not be used for discovery.
  mdt mtu <mtu-value> <-- Overlay ENCAP Max MTU value
```

  - データ MDT モードでは、テナント マルチキャスト トラフィックのサブセットまたはすべてに一意のマルチキャスト グループ セットを設定します。

```
mdt data 229.1.100.2/32 immediate-switch
mdt data 232.1.10.4/24 immediate-switch
route-map DATA_MDT_MAP permit 10
  match ip multicast group 237.1.1.1/32
```

```
mdt data 235.1.1.1/32 immediate-switch route-map DATA_MDT_MAP
```

- MVPN トンネル統計情報を有効にします。

```
hardware profile mvpn-stats module all
```

ハンドオフノードの CoPP 設定

TRM と MVPN はどちらも、コントロールプレーンに大きく依存しています。トポロジに従って CoPP ポリシー帯域幅を設定してください。

次の CoPP クラスは、TRM および MVPN トラフィックに使用されます。

- **copp-system-p-class-multicast-router**（デフォルトの帯域幅は 3000 pps です）。
- **copp-system-p-class-l3mc-data**（デフォルトの帯域幅は 3000 pps です）。
- **copp-system-p-class-l2-default**（デフォルトの帯域幅は 50 pps です）。
- **copp-class-normal-igmp**（デフォルトの帯域幅は 6000 pps です）。

次の設定例は、マルチキャスト ルート スケールによる制御パケット ドロップを回避するように設定できる CoPP ポリシーを示しています。

（注） この例のポリサー値は概算値であり、すべてのトポロジまたはトラフィック パターンに最適とは限りません。MVPN/TRM トラフィック パターンに従って CoPP ポリシーを設定します。

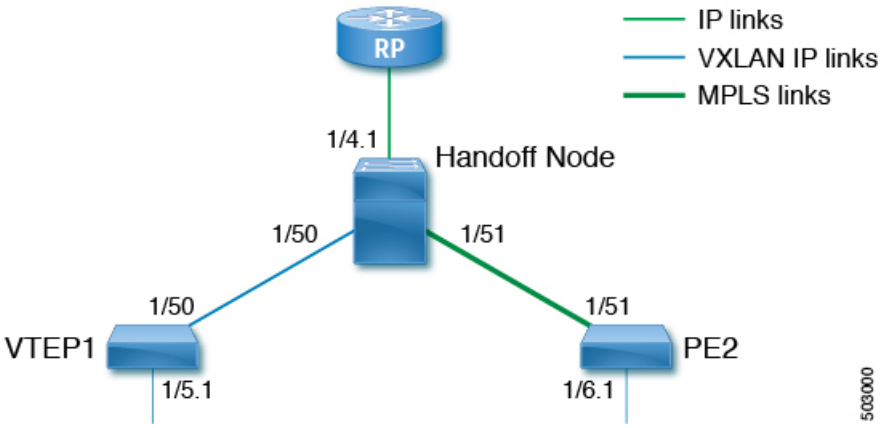
```
copp copy profile strict prefix custom
  policy-map type control-plane custom-copp-policy-strict
    class custom-copp-class-normal-igmp
      police cir 6000 pps bc 512 packets conform transmit violate drop
  control-plane
    service-policy input custom-copp-policy-strict

copp copy profile strict prefix custom
  policy-map type control-plane custom-copp-policy-strict
    class custom-copp-class-multicast-router
      police cir 6000 pps bc 512 packets conform transmit violate drop
  control-plane
    service-policy input custom-copp-policy-strict

copp copy profile strict prefix custom
  policy-map type control-plane custom-copp-policy-strict
    class copp-system-p-class-l3mc-data
      police cir 3000 pps bc 512 packets conform transmit violate drop
  control-plane
    service-policy input custom-copp-policy-strict

copp copy profile strict prefix custom
  policy-map type control-plane custom-copp-policy-strict
    class custom-copp-class-l2-default
      police cir 9000 pps bc 512 packets conform transmit violate drop
  control-plane
```

```
service-policy input custom-copp-policy-strict
```

EVPN（TRM）と MVPN とのシームレスな統合の設定例

次の図は、左側に VXLAN ネットワーク、右側に MVPN ネットワーク、中央集中型ハンドオフノードを持つサンプル トポロジを示しています。

図 *27 : EVPN*（*TRM*）と *MVPN* のシームレスな統合のサンプル トポロジ



次に、このトポロジの VTEP、ハンドオフ ノード、および PE の設定例を示します。

VTEP1 の設定：

```
feature ngmvpn
feature interface-vlan
feature vn-segment-vlan-based
feature nv overlay
feature pim
nv overlay evpn
ip pim rp-address 90.1.1.100 group-list 225.0.0.0/8
ip pim ssm range 232.0.0.0/8

vlan 555
  vn-segment 55500

route-map ALL_ROUTES permit 10
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback2
  member vni 55500 associate-vrf
    mcast-group 225.3.3.3

interface loopack1
  ip address 196.196.196.196/32

interface loopback2
  ip address 197.197.197.197/32
  ip pim sparse-mode

feature bgp
router bgp 1
```

```
    address-family l2vpn evpn
        maximum-paths 8
        maximum-paths ibgp 8
    neighbor 2.1.1.2
        remote-as 1
        update-source loopback 1
        address-family ipv4 unicast
          send-community extended
        address-family ipv6 unicast
          send-community extended
        address-family ipv4 mvpn
          send-community extended
        address-family l2vpn evpn
          send-community extended
    vrf vrfVxLAN5023
        address-family ipv4 unicast
          advertise l2vpn evpn
          redistribute direct route-map ALL_ROUTES
          maximum-paths 8
          maximum-paths ibgp 8

vrf context vpn1
  vni 55500
  ip pim rp-address 27.27.27.27 group-list 224.0.0.0/4
  ip pim ssm range 232.0.0.0/8
  ip multicast multipath s-g-hash next-hop-based
rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto mvpn
    route-target both auto evpn

interface Vlan555
  no shutdown
  vrf member vpn1
  ip forward
  ip pim sparse-mode

interface Ethernet 1/50
  ip pim sparse-mode

interface Ethernet1/5.1
  encapsulation dot1q 90
  vrf member vpn1
  ip address 10.11.12.13/24
  ip pim sparse-mode
  no shutdown
```

ハンドオフ ノードの設定：

```
install feature-set mpls
  allow feature-set mpls
feature-set mpls
feature ngmvpn
feature bgp
feature pim
feature mpls l3vpn
feature mvpn
feature mpls ldp
feature interface-vlan
feature vn-segment-vlan-based
feature nv overlay
nv overlay evpn
```

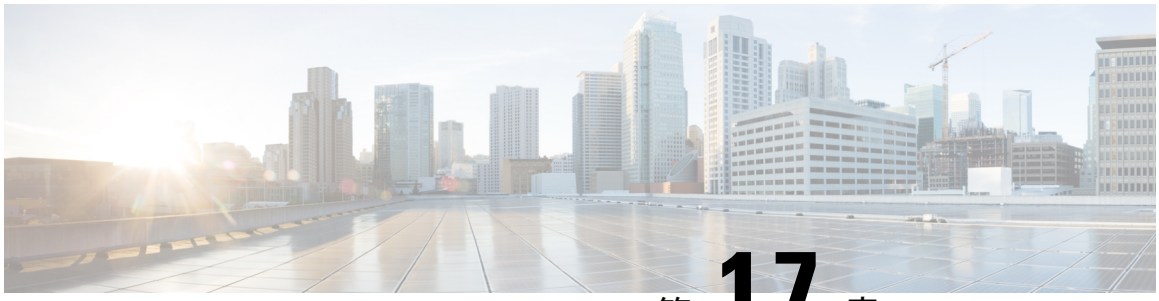
```
ip pim rp-address 90.1.1.100 group-list 225.0.0.0/8
ip pim rp-address 91.1.1.100 group-list 232.0.0.0/8

interface loopback1
  ip address 90.1.1.100 /32
  ip pim sparse-mode

interface loopback2
  ip address 91.1.1.100 /32
  ip pim sparse-mode

ip prefix-list LDP-LOOPBACK seq 2 permit 20.20.20.20/32
ip prefix-list LDP-LOOPBACK seq 3 permit 30.30.30.30/32
mpls ldp configuration
    advertise-labels for LDP-LOOPBACK
    label allocate label global prefix-list LDP-LOOPBACK

interface Ethernet 1/50
    ip pim sparse-mode

interface Ethernet 1/51
    ip pim sparse-mode
    mpls ip

interface Ethernet1/4.1
  encapsulation dot1q 50
  vrf member vpn1
  ip pim sparse-mode
  no shutdown

interface loopback0
  ip address 20.20.20.20/32
  ip pim sparse-mode

vlan 555
  vn-segment 55500

route-map ALL_ROUTES permit 10

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback3
  member vni 55500 associate-vrf
    mcast-group 225.3.3.3

interface loopback3
  ip address 198.198.198.198/32
  ip pim sparse-mode

vrf context vpn1
  vni 55500
  ip pim rp-address 27.27.27.27 group-list 224.0.0.0/4
  ip pim ssm range 232.0.0.0/8
  ip multicast multipath s-g-hash next-hop-based
  mdt default 232.1.1.1
  mdt source loopback 0
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto mvpn
    route-target both auto evpn

interface Vlan555
```

```
  no shutdown
  vrf member vpn1
  ip forward
  ip pim sparse-mode

router bgp 1
    address-family l2vpn evpn
        maximum-paths 8
        maximum-paths ibgp 8
    address-family vpnv4 unicast
        no allocate-label option-b
    address-family ipv4 mdt
    address-family ipv4 mvpn
        maximum-paths 8
        maximum-paths ibgp 8
    neighbor 196.196.196.196
        remote-as 1
        address-family ipv4 unicast
          send-community extended
        address-family ipv6 unicast
          send-community extended
        address-family ipv4 mvpn
          send-community extended
        address-family l2vpn evpn
          send-community extended
          import vpn unicast reoriginate

router bgp 1
    neighbor 30.30.30.30
        remote-as 100
        update-source loopback0
        ebgp-multihop 255
        address-family ipv4 unicast
          send-community extended
        address-family vpnv4 unicast
          send-community
          send-community extended
          next-hop-self
          import l2vpn evpn reoriginate
        address-family ipv4 mdt
          send-community extended
          no next-hop-third-party
```

PE2 の設定：

```
install feature-set mpls
  allow feature-set mpls
feature-set mpls
feature bgp
feature pim
feature mpls l3vpn
feature mpls ldp
feature interface-vlan

ip pim rp-address 91.1.1.100 group-list 232.0.0.0/8
ip prefix-list LDP-LOOPBACK seq 2 permit 20.20.20.20/32
ip prefix-list LDP-LOOPBACK seq 3 permit 30.30.30.30/32
mpls ldp configuration
    advertise-labels for LDP-LOOPBACK
    label allocate label global prefix-list LDP-LOOPBACK

interface Ethernet 1/51
    ip pim sparse-mode
    mpls ip
```

```
interface Ethernet1/6.1
  encapsulation dot1q 50
  vrf member vpn1
  ip pim sparse-mode
  no shutdown

interface loopback0
  ip address 30.30.30.30/32
  ip pim sparse-mode

vrf context vpn1
  ip pim rp-address 27.27.27.27 group-list 224.0.0.0/4
  ip pim ssm range 232.0.0.0/8
  ip multicast multipath s-g-hash next-hop-based
  mdt default 232.1.1.1
  mdt source loopback 0
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto mvpn
    route-target both auto evpn

router bgp 100
      router-id 30.30.30.30
      address-family vpnv4 unicast
            additional-paths send
            additional-paths receive
            no allocate-label option-b
      neighbor 20.20.20.20
            remote-as 1
            update-source loopback0
            address-family vpnv4 unicast
                send-community
                send-community extended
            address-family ipv4 mdt
                send-community extended
                no next-hop-third-party
```



第 17 章

VXLAN EVPN マルチサイトの構成

この章は、次の内容で構成されています。

VXLAN EVPN マルチサイト

VXLAN EVPN マルチサイト ソリューションは、2 つ以上の BGP ベース イーサネット VPN（EVPN）サイト/ファブリック（オーバーレイ ドメイン）を IP 専用ネットワーク上でスケーラブルに相互接続します。このソリューションでは、エニーキャストまたはvPCモードでボーダー ゲートウェイ（BGW）を使用して、2つのサイトを終端し、相互接続します。BGW は、トラフィックの適用と障害の封じ込め機能に必要なネットワーク制御境界を提供します。

Cisco NX-OS Release 9.3(5) よりも前のリリースの BGP コントロール プレーンでは、BGW 間の BGP セッションによって EVPN ルートのネクスト ホップ情報が書き換えられ、再発信されます。Cisco NX-OS Release 9.3(5) 以降では、再発信は常に（シングルまたはデュアルルート識別子を使用して）有効になり、書き換えは実行されません。詳細については、マルチサイトのデュアル RD サポート （330 ページ）を参照してください。

VXLAN トンネル エンドポイント（VTEP）は、BGW を含むオーバーレイ ドメインの内部ネイバーだけを認識します。ファブリック外部のすべてのルートには、レイヤ 2 およびレイヤ 3 トラフィック用の BGW 上にネクスト ホップがあります。

BGW は、サイト内のノードおよびサイトの外部にあるノードと対話するノードです。たとえば、リーフ スパイン データセンター ファブリックでは、リーフ、スパイン、またはサイトを相互接続するゲートウェイとして機能する別のデバイスを使用できます。

VXLAN EVPN マルチサイト 機能は、単一の共通 EVPN 制御および IP 転送ドメインを介して相互接続された複数のサイト ローカル EVPN コントロール プレーンおよび IP 転送ドメインとして概念化できます。すべての EVPN ノードは、一意のサイト スコープ識別子で識別されます。サイトローカル EVPN ドメインは、同じサイト識別子を持つ EVPN ノードで構成されます。BGW は一方ではサイト固有の EVPN ドメインの一部であり、他方では他のサイトからの BGW と相互接続するための共通 EVPN ドメインの一部です。特定のサイトに対して、これらの BGW はサイト固有のノードを促進し、他のすべてのサイトがそれらを介してのみ到達可能であることを可視化します。これは、以下を意味します。

- サイト ローカル ブリッジング ドメインは、他のサイトからのブリッジング ドメインと BGW を介してのみ相互接続されます。
- サイト ローカル ルーティング ドメインは、BGW を介してのみ、他のサイトからのルーティング ドメインと相互接続されます。
- サイト ローカル フラッド ドメインは、BGW を介してのみ、他のサイトからのフラッド ドメインと相互接続されます。

選択的アドバタイズメントは、BGW のテナントごとの情報の設定として定義されます。具体的には、IP VRF または MAC VRF（EVPN インスタンス）を意味します。外部接続（VRF-Lite）と EVPN マルチサイトが同じ BGW に共存する場合、アドバタイズメントは常に有効になります。

（注） サイト ID が 2 バイトを超える場合は、MVPN VRI ID をエニーキャスト BGW の TRM に設定する必要があります。同じサイトの一部であるすべてのエニーキャスト BGW で同じ VRI ID を設定する必要があります。ただし、VRI ID はネットワーク内で一意である必要があります。つまり、他のエニーキャスト BGW または vPC リーフは異なる VRI ID を使用する必要があります。

マルチサイトのデュアル RD サポート

Cisco NX-OS リリース 9.3(5) 以降では、VXLAN EVPN マルチサイトはデュアル ルート識別子（RD）を使用したルート再生成をサポートしています。この動作は自動的に有効になります。

各 VRF または L2VNI は、プライマリ RD（一意）とセカンダリ RD（BGW 間で同じ）という 2 つの RD を追跡します。再発信されたルートは、セカンダリ タイプ 0 RD（site-id：VNI）で

アドバタイズされます。他のすべてのルートは、プライマリ RD でアドバタイズされます。ルータがマルチサイト BGW モードになると、セカンダリ RD が自動的に割り当てられます。

サイトIDが2バイトを超える場合、セカンダリ RD はマルチサイト BGW で自動的に生成されず、次のメッセージが表示されます。

```
%BGP-4-DUAL_RD_GENERATION_FAILED: bgp- [12564] Unable to generate dual RD on EVPN multisite
 border gateway. This may increase memory consumption on other BGP routers receiving
re-originated EVPN routes. Configure router bgp <asn> ; rd dual id <id> to avoid it.
```

この場合、セカンダリ RD 値を手動で設定するか、デュアル RD を無効にすることができます。詳細については、マルチサイトのデュアル RD サポートの設定 （338 ページ） を参照してください。

マルチサイト BGW の ESI を使用した EVPN マルチホーミングとの相互運用性

Cisco NX-OS リリース 10.2(2)F以降、予約されていない ESI（0 または MAX-ESI）値と予約されている ESI（0 または MAX-ESI）値を持つ EVPN MAC/IP ルート（タイプ 2）は、転送（ESI RX）のために評価されます。EVPN MAC/IP ルート解決の定義は、 RFC 7432 Section 9.2.2 で定義されています。

EVPN MAC/IP ルート（タイプ 2）-

- 予約されている ESI 値（0 または MAX-ESI）は、MAC/IP ルート単独（タイプ 2 内の BGP ネクストホップ）によって単独で解決されます。
- 予約されていない ESI 値は、適合する ES イーサネット自動検出ルート（タイプ 1、ES EAD ごと）が存在する場合、単独で解決されます。

上記の MAC/IP ルート解決に加えて、マルチサイト BGW は、予約済みおよび予約されていない ESI 値を持つ MAC/IP ルートの転送、書き換え、および再発信をサポートします。これらすべての場合において、ES ごとの EAD ルートはマルチサイト BGW によって再発信されます。

異なる ESI 値を持つ EVPN MAC/IP ルート解決は、エニーキャストおよび vPC ボーダー ゲートウェイ モードの Cisco Nexus 9300-EX、-FX、-FX2、-FX3、および -GX プラットフォーム スイッチでサポートされます。

vPC BGW はサポートされていません。

マルチサイトでの VXLAN EVPN の注意事項と制限事項

VXLAN EVPN 設定時の注意事項と制約事項は次のとおりです。

- 次のスイッチは VXLAN EVPN マルチサイトをサポートします。
  - Cisco Nexus 9300-EX および 9300-FX プラットフォーム スイッチ（Cisco Nexus 9348GC-FXP プラットフォーム スイッチを除く）

- Cisco Nexus 9300-FX2 プラットフォーム スイッチ
- Cisco Nexus 9300-FX3 プラットフォーム スイッチ
- Cisco Nexus 9300-GX プラットフォーム スイッチ
- Cisco Nexus 9300-GX2 プラットフォーム スイッチ
- Cisco Nexus 9332D-H2R スイッチ
- Cisco Nexus 93400LD-H1 スイッチ
- Cisco Nexus 9364C-H1 スイッチ
- X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9800 プラットフォーム スイッチ
- -EX または FX または -GX ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ

（注） -R/RX ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチは VXLAN EVPN マルチサイトをサポートしていません。

- **evpn multisite dci-tracking** は、エニーキャスト BGW と vPC BGW DCI リンクに必須です。

  evpn multisite fabric-tracking は、エニーキャスト BGW にのみ必須です。vPC ベースの BGW の場合、このコマンドは必須ではありません。NVEインターフェイスは、アップ状態のdciトラッキング対象リンクだけで起動します。

- Cisco Nexus 9332C および 9364C プラットフォーム スイッチは BGW にすることができます。
- VXLAN EVPN マルチサイト展開では、ttag 機能を使用する場合、クラウドに接続する BGW の DCI インターフェイスで ttag が削除されていることを確認します（**ttag-strip**）。もう少し詳しく説明すると、ttag が、Ether Type 0x8905 をサポートしない、Nexus 9000 以外のデバイスにアタッチされている場合、ttag の除去が必要になります。ただし、DCIのBGW バックツーバックモデルではttagの削除は必要ありません。
- VXLAN EVPN マルチサイトおよびテナント ルーテッド マルチキャスト（TRM）は、異なるサイトに展開された送信元と受信者の間でサポートされます。
- マルチサイト BGW では、マルチサイト拡張（レイヤ 2 ユニキャスト/マルチキャストおよびレイヤ 3 ユニキャスト）と、レイヤ 3 ユニキャストおよびマルチキャスト外部接続を共存させることができます。
- マルチサイト展開を使用した TRM では、すべての BGW がファブリックからトラフィックを受信します。ただし、指定フォワーダ（DF）BGWだけがトラフィックを転送します。他のすべての BGW は、デフォルトのドロップ ACL を介してトラフィックをドロップします。この ACL は、すべての DCI トラッキング ポートでプログラムされます。DCI アッ

プリンク ポートから **evpn multisite dci-tracking** 設定を削除しないでください。この場合、ACL を削除します。これにより、1 つの BGW（DF）だけでパケットを確定的に転送するのではなく、パケットをドロップまたは複製できる非確定的なトラフィック フローが作成されます。

- エニーキャスト モードは、サイトあたり最大 6 つの BGW をサポートできます。
- vPC トポロジの BGW がサポートされます。
- サイト間/ファブリック BGW 間のマルチキャスト フラッド ドメインはサポートされていません。
- 異なるファブリック/サイトの BGW 間での iBGP EVPN ピアリングはサポートされていません。
- **peer-type fabric-external** コマンド構成は、VXLAN マルチサイト BGW にのみ必要です (このコマンドは、Cisco 以外の機器とピアリングする場合は使用しないでください)。

（注） **peer-type fabric-external** コマンド構成は、疑似 BGW で不要です。

- エニーキャスト モードは、ローカル インターフェイスに接続されたレイヤ 3 サービスのみをサポートします。
- エニーキャスト モードでは、BUM は各ボーダー リーフに複製されます。特定のサイトのボーダー リーフ間の DF 選定により、そのサイトのサイト間トラフィック（ファブリックから DCI へ、およびその逆）を転送するボーダー リーフが決定されます。
- エニーキャスト モードでは、すべてのレイヤ 3 サービスが、物理 IP をネクスト ホップとして EVPN タイプ 5 ルートを介して BGP でアドバタイズされます。
- vPC モードは 2 つの BGW のみをサポートします。
- vPC モードでは、ローカル インターフェイスでレイヤ 2 ホストとレイヤ 3 サービスの両方をサポートできます。
- vPC モードでは、BUM は外部サイトからのトラフィックのいずれかの BGW に複製されます。したがって、両方の BGW はサイト外部からサイト内部（DCI からファブリック）方向のフォワーダです。
- vPC モードでは、BUM は入力レプリケーション（IR）アンダーレイを使用して、VLAN のローカル サイト リーフから着信するトラフィックのいずれかの BGW に複製されます。両方の BGW は、IR アンダーレイを使用する VLAN のサイト内部からサイト外部（ファブリックから DCI）方向のフォワーダです。
- vPC モードでは、BUM は、マルチキャスト アンダーレイを使用して VLAN のローカル サイト リーフから着信するトラフィックの両方の BGW に複製されます。したがって、デキャッパ/フォワーダの選択が行われ、カプセル化解除の勝者/フォワーダは、マルチキャ

ストアンダーレイを使用して、サイトローカルトラフィックをVLANの外部サイトBGWにのみ転送します。

- NX-OS 10.2(2)F より前では、コア全体の DCI ピア間で入力レプリケーションのみがサポートされていました。Cisco NX-OS リリース 10.2（2）F 以降では、コア全体の DCI ピア間で入力レプリケーションとマルチキャストの両方がサポートされています。
- vPC モードでは、すべてのレイヤ 3 サービス/アタッチメントが、仮想 IP をネクスト ホップとして EVPN タイプ 5 ルートを介して BGP でアドバタイズされます。VIP/PIP 機能が設定されている場合は、ネクスト ホップとして PIP でアドバタイズされます。
- サイト間で異なるエニーキャスト ゲートウェイ MAC アドレスが設定されている場合は、拡張されたすべての VLAN に対して ARP 抑制を有効にします。
- NVE を、レイヤ 3 プロトコルで必要なループバック アドレスとは別のループバック アドレスにバインドします。ベスト プラクティスは、NVE 送信元インターフェイス（PIP VTEP）およびマルチサイト送信元インターフェイス（エニーキャストおよび仮想IP VTEP）に専用のループバック アドレスを使用することです。
- PIM BiDir は、VXLAN マルチサイトでのファブリック アンダーレイ マルチキャスト レプリケーションではサポートされません。
- PIM はマルチサイト VXLAN DCI リンクではサポートされません。
- FEX は vPC BGW およびエニーキャスト BGW ではサポートされません。
- Cisco NX-OS Release 9.3(5) 以降では、サブインターフェイスが設定されている場合、VTEP は親インターフェイス上でVXLANカプセル化トラフィックをサポートします。この機能は、VXLAN EVPN マルチサイトおよび DCI でサポートされます。DCI トラッキングは、親インターフェイスでのみ有効にできます。
- Cisco NX-OSリリース 9.3(5) 以降、VXLAN EVPN マルチサイトは非対称 VNI をサポートします。詳細については、「Multi-Site with Asymmetric VNIs and 非対称 VNI を使用するマルチサイトの設定例 （352 ページ）」を参照してください。
- 次の注意事項および制約事項がマルチサイトのデュアル RD サポートに適用されます。
  - デュアル RD は Cisco NX-OS リリース 9.3(5) 以降でサポートされます。
  - デュアル RD は、Cisco Nexus 9332C、9364C、9300-EX、および 9300-FX/FX2 プラットフォーム スイッチと、VXLAN EVPN マルチサイトが有効になっている -EX/FX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチで自動的に有効になります。
  - マルチサイトの再発信ルートに PIP アドバタイズメントを必要とする CloudSec またはその他の機能を使用するには、BGW でデュアルRDが有効になっている場合はルート サーバで BGP の追加パスを構成するか、デュアル RD を無効にします。
  - BGW ノードでのセカンダリ RD 追加パスの送信はサポートされていません。
  - ISSU 中に、すべての BGW がアップグレードされている間、リーフ ノードのパス数が一時的に 2 倍になることがあります。

- Cisco NX-OS リリース 9.3(5) 以降では、VXLAN EVPN マルチサイト トポロジの NVE インターフェイスで **host-reachability protocol bgp** コマンドを無効にすると、NVE インターフェイスは運用上ダウンしたままになります。
- Cisco NX-OS リリース 9.3(5) 以降、マルチサイト ボーダー ゲートウェイは、サイトのローカル スパイン/リーフ スイッチにアドバタイズするときに、着信リモートルートを再発信します。これらの再発信されたルートは、次のフィールドを変更します。
  - RD値が[Multisite Site ID：L3 VNID]に変更されます。
  - 特定の VRF に参加しているすべての VTEP でルート ターゲットが定義されていることが必須です。これには、BGW が特定の VRF を拡張することが含まれ、明示的に要求されます。Cisco NX-OS リリース 9.3(5) より前では、サイト内 VTEP からのルート ターゲットは、BGW で定義されていない場合でも、サイト境界を越えて誤って保持されていました。Cisco NX-OS リリース 9.3(5) 以降、必須の動作が適用されます。必要なルート ターゲットを BGW に追加することで、意図しないルート ターゲットのアドバタイズメントから明示的なルート ターゲットのアドバタイズメントへの変更を実行できます。
  - パスタイプが外部からローカルに変更されます。
- Cisco NX-OS リリース 10.2(3)F 移行、VXLAN EVPN マルチサイトは Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN EVPN マルチサイトは Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、VXLAN EVPN マルチサイトは Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN EVPN マルチサイトは Cisco Nexus 9364C-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 移行、マルチサイトのデュアル RD は Cisco Nexus 9300-FX 3 プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN マルチサイト エニーキャスト BGW は、X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチでもサポートされます。
  - VXLAN マルチサイト エニーキャスト BGW は、次の機能をサポートします。
    - VXLAN BGP EVPN ファブリックおよびマルチサイト インターコネクト
    - VXLAN レイヤ 2 VNI および VLAN ベースではない新しいレイヤ 3 VNI
    - IPv4アンダーレイ
    - ファブリック側と DCI 側の入力レプリケーション
    - ファブリックのマルチキャスト アンダーレイ
    - Bud ノード

- TRMv4
- NGOAM
- VXLAN カウンタ
  - VXLAN ピア ベースの合計パケット/バイト カウンタがサポートされます。
  - VNI ベースの合計パケット数/バイト数カウンタがサポートされます。

- VXLAN マルチサイト エニーキャスト BGW は、次の機能をサポートしていません。
  - IPv6 アンダーレイ
  - vPC BGW
  - ダウンストリーム VNI とルートリーク
  - ファブリックまたは DCI リンクとしての L3 ポート チャネル
  - DCI 側でのマルチキャスト アンダーレイ
  - VXLAN アクセス機能
  - IGMP スヌーピング
  - ブロードキャスト、マルチキャスト、およびユニキャスト トラフィック用の個別の VXLAN カウンタ
  - Data MDT
  - TRMv6
  - EVPN ストーム制御

- ファブリック リンクに障害が発生した場合のコンバージェンスを改善し、ファブリック リンクのフラッピングが発生した場合の問題を回避するには、スパインと BGW のループバック間にマルチホップ BFD を構成してください。

  すべてのファブリック リンクに障害が発生したために BGW ノードがファブリックから完全に分離される特定のシナリオでは、マルチホップ BFD を使用すると、構成された BGP ホールド時間値にはかかわりなく、スパインと分離された BGW 間の BGP セッションをすぐにダウンできます。

- VXLAN マルチサイト環境では、VXLAN オーバーレイと L3 プレフィックスの両方を介したルーティングに ECMP を使用してリモート サイト サブネットにアクセスするボーダー ゲートウェイ デバイスで、これらのルートのいずれかで隣接関係解決エラーが発生する可能性があります。スイッチがこの未解決のプレフィックスを使用しようとすると、トラフィックはドロップされます。

VXLAN EVPN マルチサイトを有効にする

この手順は、VXLAN EVPN マルチサイトの機能を有効にしてください。マルチサイトはBGWでのみ有効になります。site-id は、ファブリック/サイト内のすべての BGW で同じである必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**evpn multisite border-gateway** *ms-id* 例： `switch(config)# evpn multisite border-gateway 100`	サイト/ファブリックのサイト ID を設定します。*ms-id* の値の範囲は、1ー2,814,749,767,110,655 です。 *ms-id* は、同じファブリック/サイト内のすべての BGW で同じである必要があります。
ステップ 3	**split-horizon per-site** 例： `switch(config-evpn-msite-bgw)# split-horizon per-site`	同じサイトの別のボーダー ゲートウェイから DCI グループでカプセル化されたパケットを受信できるようにし、パケットの重複を回避します。 （注） このコマンドは、エニーキャストボーダー ゲートウェイを備えたサイトで DCI マルチキャスト アンダーレイが設定されている場合に使用します。
ステップ 4	**interface nve 1** 例： `switch(config-evpn-msite-bgw)# interface nve 1`	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。
ステップ 5	**source-interface loopback** *src-if* 例： `switch(config-if-nve)# source-interface loopback 0`	送信元インターフェイスは、有効な/32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。このれは、転送ネットワークのダイナミック ルーティング プロトコルを介してそれをアドバタイズすることによって、この要件を達成します。
ステップ 6	**host-reachability protocol bgp** 例：	これはホスト到達可能性のアドバタイズメント機構として BGP を定義します。

	コマンドまたはアクション	目的
	`switch(config-if-nve)# host-reachability protocol bgp`	
ステップ 7	**multisite border-gateway interface loopback** *vi-num* 例： `switch(config-if-nve)# multisite border-gateway interface loopback 100`	BGW 仮想 IP アドレス（VIP）に使用されるループバック インターフェイスを定義します。 border-gateway インターフェイスは、有効な /32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。このれは、転送ネットワークのダイナミック ルーティング プロトコルを介してそれをアドバタイズすることによって、この要件を達成します。このループバックは、送信元インターフェイスのループバックとは異なる必要があります。 *vi-num* の範囲は、0 ～ 1023 です。
ステップ 8	**no shutdown** 例： `switch(config-if-nve)# no shutdown`	**shutdown** コマンドを無効にします。
ステップ 9	**exit** 例： `switch(config-if-nve)# exit`	NVE 設定モードを終了します。
ステップ 10	**interface loopback** *loopback-number* 例： `switch(config)# interface loopback 0`	ループバック インターフェイスを設定します。
ステップ 11	**ip address** *ip-address* 例： `switch(config-if)# ip address 198.0.2.0/32`	ループバック インターフェイスの IP アドレスを設定します。

マルチサイトのデュアル RD サポートの設定

セカンダリ RD 値を手動で設定するか、デュアル RD を無効にする必要がある場合は、次の手順に従います。

始める前に

VXLAN EVPN マルチサイトを有効にします。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**router bgp** *as-num*
	例：
	`switch(config)# router bgp 100`
	`switch(config-router)#`
		自律システム番号を設定する。*as-num* の範囲は 1 ～ 4,294,967,295 です。

ステップ 3	[**no**] **rd dual id** [*2-bytes*] 例： `switch(config-router)# rd dual id 1`	セカンダリ RD の最初の 2 バイトを定義します。ID は、マルチサイト BGW 間で同じである必要があります。有効な範囲は 1 ～ 65535 です。 （注） 必要に応じて、**no rd dual** コマンドを使用してデュアル RD を無効にし、単一の RD にフォール バックできます。
ステップ 4	（任意） **show bgp evi** *evi-id* 例： `switch(config-router)# show bgp evi 100`	指定した EVI の **rd dual id**[*2-bytes*] コマンドの一部として設定されたセカンダリ RD を表示します。

例

次の例は、**show bgp evi** *evi-id* コマンドのサンプル出力を示しています。

```
switch# show bgp evi 100
------------------------------------------------
  L2VNI ID                      : 100 (L2-100)
  RD                            : 3.3.3.3:32867
  Secondary RD                  : 1:100
  Prefixes (local/total)        : 1/6
  Created                       : Jun 23 22:35:13.368170
  Last Oper Up/Down             : Jun 23 22:35:13.369005 / never
  Enabled                       : Yes

  Active Export RT list         :
        100:100
  Active Import RT list         :
        100:100
```

VNI デュアル モードの設定

この手順では、特定の VLAN の BUM トラフィック ドメインの設定について説明します。ファブリック/サイト内のマルチキャストまたは入力レプリケーションと、異なるファブリック/サイト間での入力レプリケーションの使用がサポートされています。

（注） 複数の VRF があり、1 つだけがすべてのリーフ スイッチに拡張されている場合は、その 1 つの拡張 VRF にダミーのループバックを追加し、BGP を介してアドバタイズできます。それ以外の場合は、拡張されている VRF の数とどのスイッチに拡張されているかを確認してから、それぞれの VRF にダミーのループバックを追加し、それらもアドバタイズする必要があります。したがって、 **advertise-pip** コマンドを使用して、今後発生する可能性のあるユーザー エラーを防止します。

多数の VNI のマルチキャストまたは入力レプリケーションの設定の詳細については、VXLAN BGP EVPN の例（EBGP） （175 ページ） を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**interface nve 1** 例： `switch(config)# interface nve 1`	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。
ステップ 3	**member vni** *vni-range* 例： `switch(config-if-nve)# member vni 200`	仮想ネットワーク識別子（VNI）を設定します。*vni-range* の範囲は 1 ～ 16,777,214 です。*vni-range* の値は、5000 などの単一の値または 5001ー5008 などの範囲です。 （注） ステップ 4 またはステップ 5 のいずれかのコマンドを入力します。
ステップ 4	**mcast-group** *ip-addr* 例： `switch(config-if-nve-vni)# mcast-group 255.0.4.1`	ファブリック内の NVE マルチキャスト グループ IP プレフィックスを設定します。

	コマンドまたはアクション	目的
ステップ 5	**ingress-replication protocol bgp** 例： `switch(config-if-nve-vni)# ingress-replication protocol bgp`	VNI の入力複製をする BGP EVPN を有効にします。
ステップ 6	**multisite ingress-replication** 例： `switch(config-if-nve-vni)# multisite ingress-replication`	レイヤ 2 VNI を拡張するためのマルチサイト BUM レプリケーション方式を定義します。

ファブリック/DCI リンク トラッキングの設定

この手順では、すべての DCI 側インターフェイスとサイトの内部/ファブリック側インターフェイスを追跡するための設定について説明します。トラッキングは必須で、すべての DCI/ファブリック リンクがダウンした場合に、サイトからまたはサイトへの EVPN ルートの再発信を無効にするために使用されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル構成モードを開始します。
ステップ 2	**interface ethernet** *port* 例： `switch(config)# interface ethernet1/1`	指定したインターフェイスのインターフェイス設定モードを開始します。 （注） ステップ 3 またはステップ 4 で、次のいずれかのコマンドを入力します。
ステップ 3	**evpn multisite dci-tracking** 例： `switch(config-if)# evpn multisite dci-tracking`	DCI インターフェイス トラッキングを設定します。
ステップ 4	（任意） **evpn multisite fabric-tracking** 例： `switch(config-if)# evpn multisite fabric-tracking`	EVPN マルチサイト ファブリック トラッキングを設定します。 **evpn multisite fabric-tracking** は、エニーキャスト BGW と vPC BGW ファブリック リンクに必須です。
ステップ 5	**ip address** *ip-addr* | **ipv6 address** *ipv6-addr* 例：	IP または IPv6 アドレスを設定します。

	コマンドまたはアクション	目的

	インターネットユーザに商品やサービスを提供する IPv4
	`switch(config-if)# ip address 192.1.1.1`
	例：
	IPv6 の場合
	`switch(config-if)# ipv6 address 2001:DB8::192:1:1:1`	

ステップ 6	**no shutdown** 例： `switch(config-if)# no shutdown`	**shutdown** コマンドを無効にします。

ファブリック外部ネイバーの設定

この手順では、他のサイト/ファブリック BGW と通信するためのファブリック外部/DCI ネイバーの設定について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**router bgp** *as-num* 例： `switch(config)# router bgp 100`	自律システム番号を設定する。*as-num* の範囲は 1 ～ 4,294,967,295 です。

ステップ 3	**neighbor** [*ip-addr* | *ipv6-addr*]
	例：
	インターネットユーザに商品やサービスを提供する IPv4
	`switch(config-router)# neighbor 100.0.0.1`
	例：
	IPv6 の場合
	`switch(config-router)# neighbor 2001:DB8::100:0:0:1`
		BGP ネイバーを設定します。

	コマンドまたはアクション	目的
ステップ 4	**remote-as** *value* 例： `switch(config-router-neighbor)# remote-as 69000`	リモート ピアの自律システム番号を構成します。
ステップ 5	**peer-type fabric-external** 例： `switch(config-router-neighbor)# peer-type fabric-external`	マルチサイトのネクスト ホップ リライトを有効にします。EVPN 交換のサイト外部 BGP ネイバーを定義します。**peer-type** のデフォルトは、**fabric-internal** です。 （注） **peer-type fabric-external** コマンドは、VXLAN マルチサイト BGW にのみ必要です。擬似 BGW には必要ありません。
ステップ 6	**address-family l2vpn evpn** 例： `switch(config-router-neighbor)# address-family l2vpn evpn`	BGP ネイバーにあるアドレス ファミリのレイヤ 2 VPN EVPN を設定します。
ステップ 7	**rewrite-evpn-rt-asn** 例： `switch(config-router-neighbor)# rewrite-evpn-rt-asn`	ルート ターゲット（RT）情報を書き換えて、MAC-VRF および IP-VRF 設定を簡素化します。BGP はルートを受信し、RT 属性を処理するときに、そのルートを送信しているピア AS と AS 値が一致するかどうかを確認し、置き換えます。具体的には、このコマンドは、BGP が設定されたネイバーのリモート AS 番号と一致するように着信ルート ターゲットの AS 番号を変更します。レシーバ ルータで変更された RT 値を確認できます。

VXLAN EVPN マルチサイト ストーム制御の設定

VXLAN EVPN マルチサイト ストーム制御により、マルチサイト BGW のマルチデスティネーション（BUM）トラフィックのレート制限が可能になります。入力方向のファブリック リンクのポリサーを使用して、DCI リンクを介して送信される BUM トラフィックを制御できます。

リモート ピアの到達可能性は、DCI リンクを介してのみ行う必要があります。適切なルーティング構成により、リモート サイト ルートがファブリック リンク上でアドバタイズされないようにする必要があります。

Cisco NX-OS リリース 9.3(6) 以降のリリースでは、レートの精度と精度が最適化されています。帯域幅は累積 DCI アップリンク帯域幅に基づいて計算され、DCI トラッキングでタグ付けされたインターフェイスのみが考慮されます。（以前のリリースには、ファブリック タグ付きインターフェイスも含まれています）。さらに、小数点以下 2 桁をサポートすることで精度が向上

します。これらの拡張機能は、Cisco Nexus 9300-EX、9300-FX/FX2/FX3、および 9300-GX プラットフォーム スイッチに適用されます。

（注） VLAN の詳細については、『Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide』を参照してください。

手順の概要

1. **configure terminal**
2. [**no**] **evpn storm-control** {**broadcast** | **multicast** | **unicast**} {**level** *level*}

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	[**no**] **evpn storm-control** {**broadcast**	**multicast**	**unicast**} {**level** *level*}
			例：
			`switch(config)# evpn storm-control unicast level 10`
			例：
			`switch(config)# evpn storm-control unicast level 10.20`
				ストーム抑制レベルを 0ー100 の数値で設定します。
				0 はすべてのトラフィックがドロップされることを意味し、100 はすべてのトラフィックが許可されることを意味します。中間の値の場合、不明なユニキャスト トラフィック レートは使用可能な帯域幅のパーセンテージに制限されます。たとえば、値 10 は、トラフィック レートが使用可能な帯域幅の 10% に制限され、そのレートを超えるものはすべてドロップされることを意味します。
				Cisco NX-OS Release 9.3(6) 以降では、小数点の後に 2 桁の数字を追加することで、レベルを小数値として設定できます。たとえば、10.20 の値を入力できます。

VXLAN EVPN マルチサイト ストーム制御の確認

EVPN ストーム制御設定情報を表示するには、次のコマンドを入力します。

コマンド	目的
slot 1 show hardware vxlan storm-control	EVPN ストーム制御設定のステータスを表示します。

（注） ストーム制御がしきい値に達すると、次のようにメッセージがログに記録されます。

```
BGWY-1 %ETHPORT-5-STORM_CONTROL_ABOVE_THRESHOLD: Traffic in port Ethernet1/32 exceeds
the configured threshold , action - Trap (message repeated 38 times)
```

vPC をサポートするマルチサイト

vPC をサポートするマルチサイトについて

BGW は vPC コンプレックスに配置できます。この場合、二重接続されたファイアウォールまたはサービス接続だけでなく、ブリッジ接続またはルーティングされる二重接続で直接接続されたホストもサポートできます。vPC BGW には vPC 固有のマルチホーミング技術があり、DF 選択またはスプリット ホライズンの EVPN タイプ 4 ルートに依存しません。

vPC サポートを使用したマルチサイトの注意事項と制限事項

vPC サポートを使用したマルチサイトは、次の注意事項と制約事項があります。

- vPCの 4000 VNI はサポートされていません。
- VIP を継続的に使用する BUM では、MCT リンクはコア分離またはファブリック分離時のトランスポートとして使用され、ファブリック分離ではユニキャスト トラフィックに使用されます。
- Cisco NX-OS リリース 10.1(2) 以降では、vPC BGW を使用した TRM マルチサイトがサポートされています。
- リモート マルチサイト BGW ループバック アドレスへのルートは、バックアップ SVI を使用して構成された vPC ボーダー ゲートウェイ スイッチ間の iBGP プロトコルよりも常に DCI リンク パスを優先する必要があります。バックアップ SVI は、DCI リンク障害が発生した場合にのみ使用する必要があります。
- vPC BGW は、IPv6 マルチキャスト アンダーレイでサポートされません。

vPC サポートによるマルチサイトの設定

この手順では、vPC をサポートするマルチサイトの設定について説明します。

- VPC ドメインの設定

- ポート チャネルを設定します。
- vPC ピア リンクを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**feature vpc** 例： `switch(config)# feature vpc`	デバイス上で vPC をイネーブルにします。
ステップ 3	**feature interface-vlan** 例： `switch(config)# feature interface-vlan`	デバイスのインターフェイス VLAN 機能をイネーブルにします。
ステップ 4	**feature lacp** 例： `switch(config)# feature lacp`	デバイスの LACP 機能をイネーブルにします。
ステップ 5	**feature pim** 例： `switch(config)# feature pim`	デバイスの PIM 機能をイネーブルにします。
ステップ 6	**feature ospf** 例： `switch(config)# feature ospf`	デバイスの OSPF 機能をイネーブルにします。
ステップ 7	**ip pim rp-address** *address* **group-list** *range* 例： `switch(config)# ip pim rp-address 100.100.100.1 group-list 224.0.0/4`	アンダーレイ マルチキャスト グループ範囲に、PIM RP アドレスを設定します。
ステップ 8	**vpc domain** *domain-id* 例： `switch(config)# vpc domain 1`	デバイス上に vPC ドメインを作成し、設定目的で vpc-domain 設定モードを開始します。デフォルトはありません。範囲は 1 ～ 1000 です。
ステップ 9	**peer switch** 例： `switch(config-vpc-domain)# peer switch`	ピア スイッチを定義します。

	コマンドまたはアクション	目的
ステップ 10	**peer gateway** 例： `switch(config-vpc-domain)# peer gateway`	vPC のゲートウェイ MAC アドレスを宛先とするパケットに対してレイヤ 3 転送をイネーブルにします。
ステップ 11	**peer-keepalive destination** *ip-address* 例： `switch(config-vpc-domain)# peer-keepalive destination 172.28.230.85`	vPC ピアキープアライブ リンクのリモート エンドの IPv4 アドレスを設定します。 （注） vPC ピアキープアライブ リンクを設定するまで、vPC ピア リンクは構成されません。 管理ポートと VRF がデフォルトです。
ステップ 12	**ip arp synchronize** 例： `switch(config-vpc-domain)# ip arp synchronize`	vPCドメインでIP ARP 同期を有効にして、デバイスのリロード後の ARP テーブルの生成を高速化します。
ステップ 13	**ipv6 nd synchronize** 例： `switch(config-vpc-domain)# ipv6 nd synchronize`	vPCドメインでIPv6 ND 同期を有効にして、デバイスのリロード後のNDテーブルの設定を高速化します。

ステップ 14	vPC ピアリンクを作成します。
	例：
	`switch(config)# interface port-channel 1`
	`switch(config)# switchport`
	`switch(config)# switchport mode trunk`
	`switch(config)# switchport trunk allowed vlan 1,10,100-200`
	`switch(config)# mtu 9216`
	`switch(config)# vpc peer-link`
	`switch(config)# no shut`
	
	`switch(config)# interface Ethernet 1/1, 1/21`
	`switch(config)# switchport`
	`switch(config)# mtu 9216`
	`switch(config)# channel-group 1 mode active`
	`switch(config)# no shutdown`
		vPC ピアリンク ポート チャネル インターフェイスを作成し、2つのメンバーインターフェイスを追加します。

ステップ 15	**system nve infra-vlans** *range* 例： `switch(config)# system nve infra-vlans 10`	バックアップ ルーテッド パスとして非 VXLAN 対応 VLAN を定義します。
ステップ 16	**vlan** *number* 例： `switch(config)# vlan 10`	インフラ VLAN として使用する VLAN を作成します。

	コマンドまたはアクション	目的

ステップ 17	SVI を作成します。
	例：
	`switch(config)# interface vlan 10`
	`switch(config)# ip address 10.10.10.1/30`
	`switch(config)# ip router ospf process UNDERLAY area 0`
	`switch(config)# ip pim sparse-mode`
	`switch(config)# no ip redirects`
	`switch(config)# mtu 9216`
	`switch(config)# no shutdown`
		vPC ピアリンク上のバックアップルーテッドパスに使用される SVI を作成します。

ステップ 18	（任意） **delay restore interface-vlan** *seconds* 例： `switch(config-vpc-domain)# delay restore interface-vlan 45`	SVI の遅延復元タイマーをイネーブルにします。SVI/VNI スケールが大きい場合は、この値を調整することを推奨します。たとえば、SCI カウントが 1000 の場合、遅延復元を 45 秒に設定することを推奨します。
ステップ 19	**evpn multisite border-gateway** *ms-id* 例： `switch(config)# evpn multisite border-gateway 100`	サイト/ファブリックのサイト ID を設定します。*ms-id* の値の範囲は 1ー281474976710655 です。*ms-id* は、同じファブリック/サイト内のすべての BGW で同じである必要があります。
ステップ 20	**interface nve 1** 例： `switch(config-evpn-msite-bgw)# interface nve 1`	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。 （注） スイッチでは 1 つの NVE インターフェイスのみ使用できます。
ステップ 21	**source-interface loopback** *src-if* 例： `switch(config-if-nve)# source-interface loopback 0`	送信元インターフェイスは、有効な /32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。
ステップ 22	**host-reachability protocol bgp** 例： `switch(config-if-nve)# host-reachability protocol bgp`	これはホスト到達可能性のアドバタイズメント機構として BGP を定義します。
ステップ 23	**multisite border-gateway interface loopback** *vi-num* 例： `switch(config-if-nve)# multisite border-gateway interface loopback 100`	BGW 仮想 IP アドレス（VIP）に使用されるループバック インターフェイスを定義します。送信元インターフェイスは、有効な /32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレ

	コマンドまたはアクション	目的
		スは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。このループバックは、送信元インターフェイスのループバックとは異なる必要があります。*vi-num* の範囲は、0 ～ 1023 です。
ステップ 24	**no shutdown** 例： `switch(config-if-nve)# no shutdown`	**shutdown** コマンドを無効にします。
ステップ 25	**exit** 例： `switch(config-if-nve)# exit`	NVE 設定モードを終了します。
ステップ 26	**interface loopback** *loopback-number* 例： `switch(config)# interface loopback 0`	ループバック インターフェイスを設定します。
ステップ 27	**ip address** *ip-address* 例： `switch(config-if)# ip address 198.0.2.0/32`	ループバック インターフェイスのプライマリ IP アドレスを設定します。
ステップ 28	**ip address** *ip-address* **secondary** 例： `switch(config-if)# ip address 198.0.2.1/32 secondary`	ループバック インターフェイスのセカンダリ IP アドレスを設定します。
ステップ 29	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	ループバック インターフェイスで PIM スパース モードを設定します。

リンク障害発生時のトランスポートとしてのピアリンクの設定

この手順では、バックアップ リンクとしてのみ使用されるように、IGP コストが高く設定された SVI インターフェイスの設定について説明します。

（注） この設定は、ファブリックや DCI リンクの障害時にピアリンクをバックアップ リンクとして使用するために必要です。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**system nve infra-vlans** *vlan-range* 例： `switch(config)# system nve infra-vlans 10`	VXLAN のアップリンクおよび vPC ピアリンクのすべての SVI インターフェイスで使用される VLAN をインフラ VLAN として指定します。インフラ VLAN の特定の組み合わせを設定しないでください。たとえば、2 と 514、10 と 522 は 512 離れています。
ステップ 3	**interface** *vlan-id* 例： `switch(config)# interface vlan10`	インターフェイスを設定します。
ステップ 4	**no shutdown** 例： `switch(config-if)# no shutdown`	**shutdown** コマンドを無効にします。
ステップ 5	**mtu** *value* 例： `switch(config-if)# mtu 9216`	最大伝送単位（MTU）を設定します。
ステップ 6	**no ip redirects** 例： `switch(config-if)# no ip redirects`	デバイスがリダイレクトを送信しないようにします。
ステップ 7	**ip address** *ip-address/length* 例： `switch(config-if)# ip address 35.1.1.2/24`	このインターフェイスの IP アドレスを設定します。
ステップ 8	**no ipv6 redirects** 例： `switch(config-if)# no ipv6 redirects`	ICMP のリダイレクト メッセージが BFD 対応インターフェイスでディセーブルであることを確認します。
ステップ 9	**ip ospf cost** *cost* 例： `switch(config-if)# ip ospf cost 100`	このインターフェイスの OSPF コスト メトリックを設定します。
ステップ 10	**ip ospf network point-to-point** 例：	OSPF ポイントツーポイント ネットワークを指定します。

	コマンドまたはアクション	目的
	`switch(config-if)# ip ospf network point-to-point`	
ステップ 11	**ip router ospf** *instance* **area** *area-number* 例： `switch(config-if)# ip router ospf 1 area 0.0.0.0`	インターフェイス上で IP のルーティング プロセスを設定して、エリアを指定します。
ステップ 12	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスにスパース モード PIM を設定します。

vPC を使用したマルチサイト サポート設定の確認

Multi-Site with vPC サポート情報を表示するには、次のいずれかのコマンドを入力します。

show vpc brief	一般的な vPC および CC のステータスを表示します。
show vpc consistency-parameters global	すべての vPC インターフェイス全体で一貫している必要があるパラメータのステータスを表示します。
show vpc consistency-parameters vni	両方の vPC ピアで一貫している必要がある NVE インターフェイス下の VNI の設定情報を表示します。

show vpc brief コマンドの出力例：

```
switch# show vpc brief
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 1
Peer status                       : peer adjacency formed ok     (<--- peer up)
vPC keep-alive status             : peer is alive
Configuration consistency status  : success (<----- CC passed)
Per-vlan consistency status      : success                     (<---- per-VNI CCpassed)
Type-2 consistency status         : success
vPC role                          : secondary
Number of vPCs configured         : 1
Peer Gateway                      : Enabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Enabled, timer is off.(timeout = 240s)
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled
[...]
```

show vpc consistency-parameters global コマンドの出力例：

```
switch# show vpc consistency-parameters global

    Legend:
        Type 1 : vPC will be suspended in case of mismatch

Name                        Type  Local Value            Peer Value
-------------               ----  ---------------------- -----------------------
[...]
Nve1 Adm St, Src Adm St,    1     Up, Up, 2.1.44.5, CP,  Up, Up, 2.1.44.5, CP,
Sec IP, Host Reach, VMAC          TRUE, Disabled,        TRUE, Disabled,
Adv, SA,mcast l2, mcast           0.0.0.0, 0.0.0.0,      0.0.0.0, 0.0.0.0,
l3, IR BGP,MS Adm St, Reo         Disabled, Up,          Disabled, Up,
                                  200.200.200.200        200.200.200.200
[...]
```

show vpc consistency-parameters vni コマンドの出力例：

```
switch(config-if-nve-vni)# show vpc consistency-parameters vni

    Legend:
        Type 1 : vPC will be suspended in case of mismatch

Name                        Type  Local Value            Peer Value
-------------               ----  ---------------------- -----------------------
Nve1 Vni, Mcast, Mode,      1     11577, 234.1.1.1,      11577, 234.1.1.1,
Type, Flags                       Mcast, L2, MS IR       Mcast, L2, MS IR
Nve1 Vni, Mcast, Mode,      1     11576, 234.1.1.1,      11576, 234.1.1.1,
Type, Flags                       Mcast, L2, MS IR       Mcast, L2, MS IR
[...]
```

非対称 VNI を使用するマルチサイトの設定例

次の例は、異なる VNI セットを持つ 2 つのサイトが同じ MAC VRF または IP VRF に接続する方法を示しています。1 つのサイトは VNI 200 を内部で使用し、もう 1 つのサイトは VNI 300 を内部で使用します。VNI 値が異なるため、ルート ターゲット auto は一致しなくなりました。したがって、ルート ターゲット値は手動で設定する必要があります。この例では、値 222:333 は異なるサイトからの 2 つの VNI をつなぎ合わせます。

サイト 1 の BGW には L2VNI 200 と L3VNI 201 があります。

サイト 2 の BGW には L2VNI 300 と L3VNI 301 があります。

（注） この設定例では、基本的なマルチサイト設定がすでに行われていることを前提としています。

（注） BGW で VLAN から VRF へのマッピングが必要です。この要件は、BGW での MAC-IP ルートの再生成に必要な L2VNI-to-L3VNI マッピングを維持するために必要です。

レイヤ 3 の設定

サイト 1 の BGW ノードで、L3VNI 201 と L3VNI 301 を使用して 2 つのサイトをつなぐ共通 RT 201:301 を設定します。

```
vrf context vni201
  vni 201
  address-family ipv4 unicast
    route-target both auto evpn
    route-target import 201:301 evpn
    route-target export 201:301 evpn
```

サイト 2の BGW ノードで、L3VNI 201 と L3VNI 301 を使用して 2 つのサイトをつなぐ共通の RT 201:301 を設定します。

```
vrf context vni301
  vni 301
  address-family ipv4 unicast
    route-target both auto evpn
    route-target import 201:301 evpn
    route-target export 201:301 evpn
```

レイヤ 2 の設定

サイト 1 の BGW ノードで、L2VNI 200 と L2VNI 300 を使用して 2 つのサイトをつなぐ共通の RT 222:333 を設定します。

```
evpn
  vni 200 l2
    rd auto
    route-target import auto
    route-target import 222:333
    route-target export auto
    route-target export 222:333
```

MAC-IP ルートの L3 ラベルを適切に再生成するには、VRF（L3VNI）を L2VNI に関連付けます。

```
interface Vlan 200
  vrf member vni201
```

サイト 2 の BGW ノードで、L2VNI 200と L2VNI 300 を使用して 2 つのサイトをつなぐ共通 RT 222:333 を設定します。

```
evpn
  vni 300 l2
    rd auto
    route-target import auto
    route-target import 222:333
    route-target export auto
    route-target export 222:333
```

MAC-IP ルートの L3 ラベルを適切に再生成するには、VRF（L3VNI）を L2VNI に関連付けます。

```
interface vlan 300
  vrf member vni301
```

マルチサイトでの TRM

ここでは、次の内容について説明します。

マルチサイトでの TRM の設定に関する情報

マルチサイトを使用したテナント ルーテッド マルチキャスト（TRM）は、マルチサイト経由で接続された複数の VXLAN EVPN ファブリック間でのマルチキャスト転送を可能にします。この機能は、さまざまなサイトの送信元と受信者に、レイヤ 3 マルチキャスト サービスを提供します。サイト間の東西マルチキャスト トラフィックの要件に対応します。

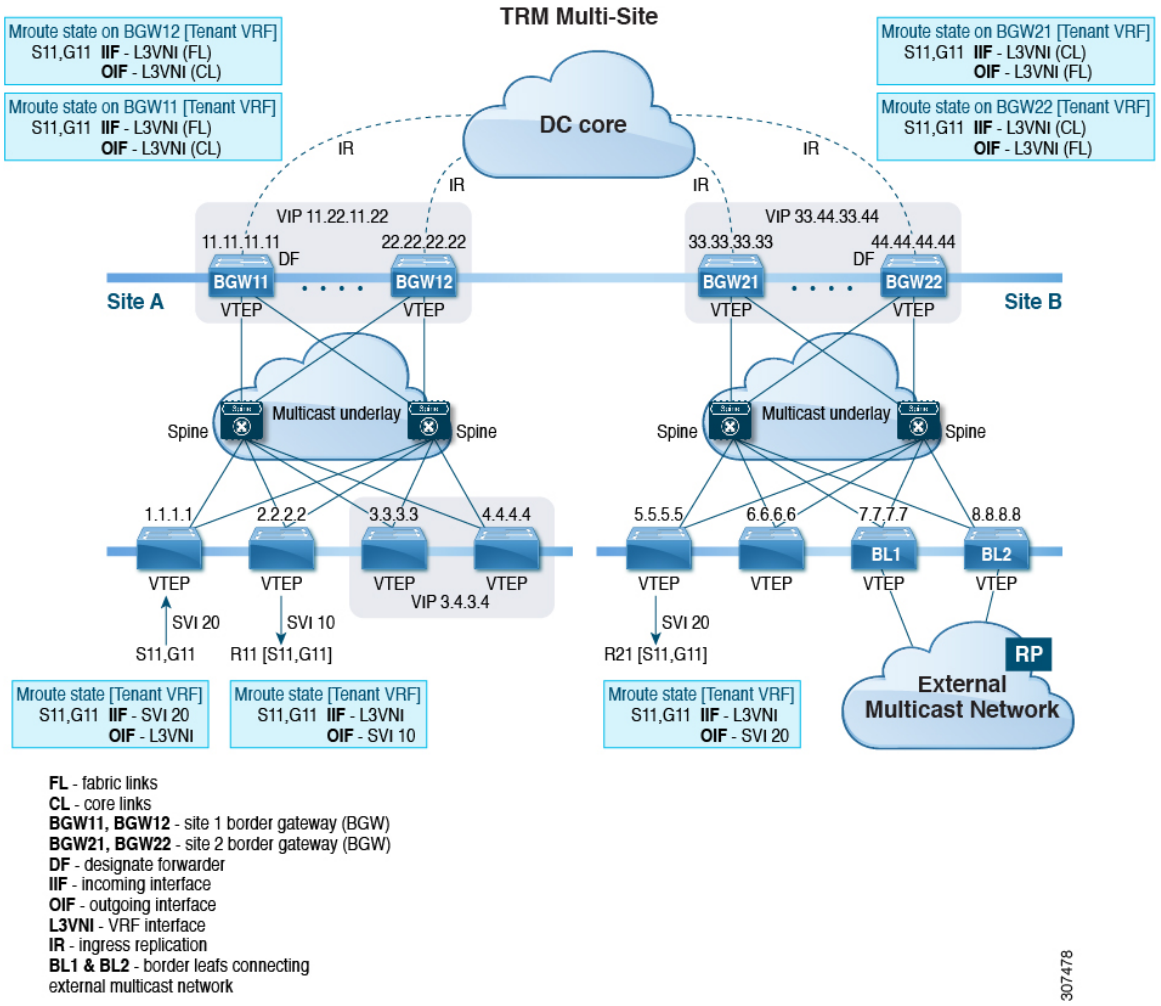
各 TRM サイトは独立して動作しています。各サイトのボーダー ゲートウェイでは、サイト間でスティッチングが可能です。サイトごとに複数のボーダー ゲートウェイを設定できます。サイト間のマルチキャスト送信元および受信者情報は、TRM が設定されたボーダー ゲートウェイ上の BGP によって伝播されます。各サイトのボーダー ゲートウェイはマルチキャスト パケットを受信し、ローカル サイトに送信する前にパケットを再カプセル化します。Cisco NX-OS リリース 10.1(2) 以降、マルチサイト対応 TRM は、エニーキャスト ボーダー ゲートウェイと vPC ボーダー ゲートウェイの両方をサポートします。

L3VNIのDesignated Forwarder（DF）として選択されたボーダー ゲートウェイは、ファブリックからコア側にトラフィックを転送します。TRM Multicast-Anycast Gateway モデルでは、VIP-R ベースのモデルを使用してリモートサイトにトラフィックを送信します。IR宛先IPは、リモート サイトの VIP-R です。受信者が存在する各サイトは、送信元サイトから 1 つのコピーのみを取得します。DF 転送は、エニーキャスト ボーダー ゲートウェイでのみ適用できます。

（注） リモート サイトにトラフィックを送信するのは DF だけです。

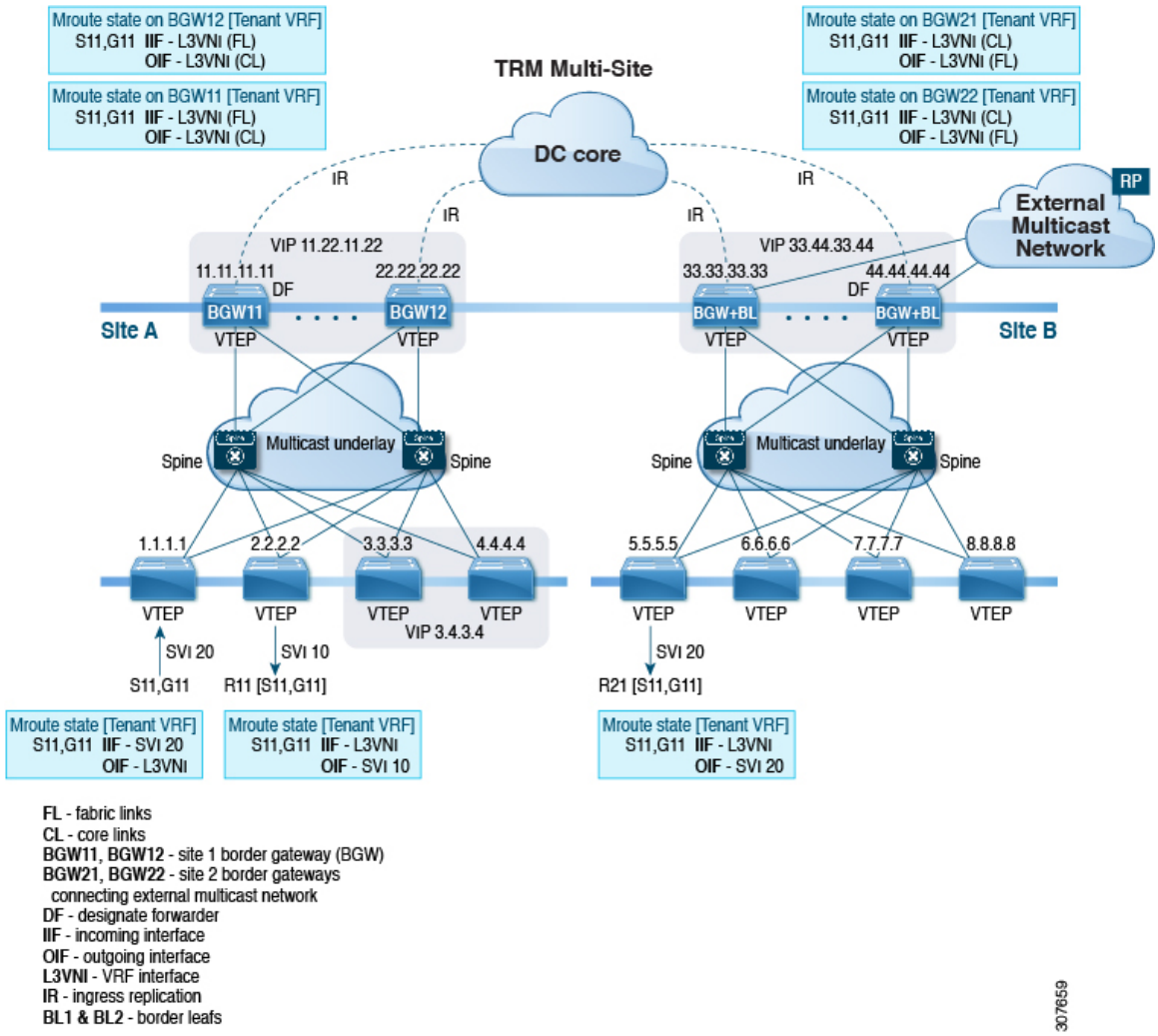
リモート サイトでは、コアからサイト間マルチキャスト トラフィックを受信する BGW がトラフィックをファブリック側に転送します。非 DF も送信元サイトから VIP-R コピーを受信できるため、コアからファブリック方向への DF チェックは行われません。

図 *28* : マルチサイト トポロジの ***TRM***、***BL*** 外部マルチキャスト接続



Cisco NX-OS リリース 9.3(3) 以降では、マルチサイト対応 TRM は、以前のリリースでサポートされていたBL 接続に加えて、外部マルチキャスト ネットワークへの BGW 接続をサポートします。転送は前の例で説明したように行われますが、外部マルチキャスト ネットワークへの出口点はオプションで BGW を介して提供できます。

図 *29 :* マルチサイト トポロジ、*BGW* 外部マルチキャスト接続を備えた *TRM*



マルチサイトでの TRM のガイドラインと制限事項

マルチサイトでは TRM には、次の注意事項と制約事項があります。

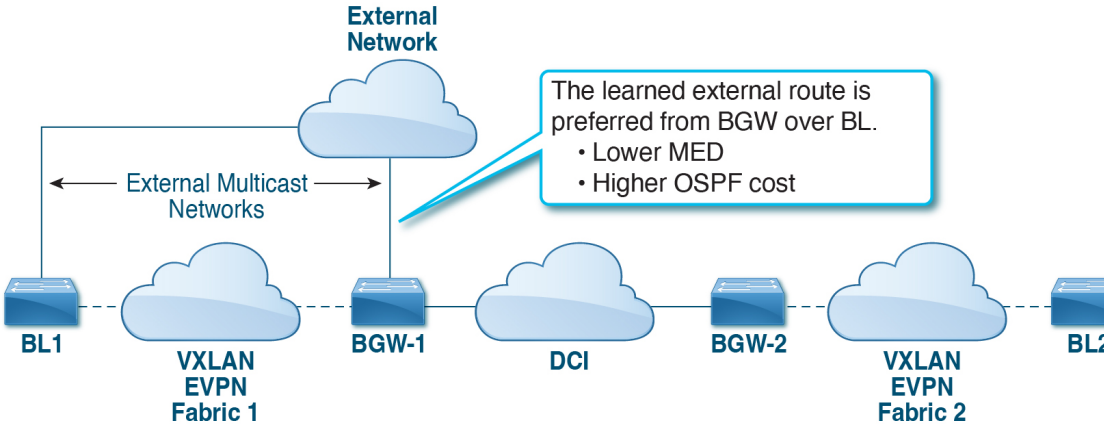
- 次のプラットフォームは、マルチサイトでの TRM をサポートしています。
  - Cisco Nexus 9300-EX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX/FX2/FX3 プラットフォーム スイッチ
  - Cisco Nexus 9300-GX プラットフォーム スイッチ
  - -EX/FX ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ
- Cisco NX-OS リリース 9.3(3) 以降では、マルチキャスト トラフィック用にボーダー リーフとマルチサイト ボーダー ゲートウェイを同じノードに共存させることができます。

- Cisco NX-OS Release 9.3(3) 以降では、特定のサイトのすべてのボーダー ゲートウェイで同じ Cisco NX-OS 9.3(x) イメージを実行する必要があります。
- Cisco NX-OS リリース 10.1(2) には、次の注意事項と制約事項があります。
  - vPC プライマリおよびセカンダリ ピアに接続された L3 ホストをサポートするために、vPC ピア間に VRF Lite リンクを（テナント VRF ごとに）追加する必要があります。
  - 2 つの vPC ピア間でバックアップ SVI が必要です。
  - L2 および L3 に接続された孤立ポートは、vPC BGW でサポートされます。
  - vPC BGW を使用した TRM マルチサイトは、vMCT ではサポートされません。

  TRM および vPC サポートによる TRM の設定の詳細については、「テナント ルーテッド マルチキャストの設定」を参照してください。
- vPC BGW および Anycast BGW を使用した TRM マルチサイトは、Cisco Nexus 9300-EX、FX、FX2、および FX3 ファミリスイッチでサポートされます。Cisco NX-OS リリース 10.2(1)F 以降、vPC BGW および Anycast BGW を使用した TRM は、Cisco Nexus 9300-GX ファミリスイッチでサポートされます。
- Cisco NX-OS リリース 10.2(1q)F 以降、マルチサイトで TRM は N9K-C9332D-GX2B プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(1q)F 以降、vPC BGW およびエニーキャスト GBW で TRM マルチサイトは N9K-C9332D-GX2B プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(2)F 以降、マルチキャスト グループ設定を使用して、**multisite mcast-group** *dci-core-group* コマンドを使用して DCI コアで TRM および L2 BUM パケットをカプセル化します。
- Cisco NX-OS リリース 10.2(3)F 以降、TRM マルチサイトは Cisco Nexus N9K-C9364D-GX2A および N9K-C9348D-GX2A プラットフォーム スイッチでサポートされています。
- マルチサイトを使用した TRM は、次の機能をサポートしています。
  - vPC ボーダー ゲートウェイを使用した TRM マルチサイト。
  - VXLAN ファブリックの PIM ASM マルチキャスト アンダーレイ
  - マルチサイト レイヤ 3 モードのみの TRM
  - エニーキャスト ゲートウェイを使用したマルチサイトでの TRM
  - 境界リーフでの VRF-Lite の終端
  - TRM マルチサイトを使用する次の RP モデル：
    - 外部 RP
    - RP Everywhere
    - 内部 RP

- 1 つのサイトで設定できる vPC BGW のペアは 1 つだけです。
- vPC BGW とエニーキャスト BGW のペアを同じサイトに共存させることはできません。
- NX-OS 10.2（2）F 以前には、コア全体の DCI ピア間では入力レプリケーションのみがサポートされてます。Cisco NX-OS リリース 10.2（2）F 以降では、コア全体の DCI ピア間で入力レプリケーションとマルチキャストの両方がサポートされています。
- ボーダールータは、ファブリックからコア、およびコアからファブリックへの MVPN ルートを再生成します。
- 異なるサイトのボーダー ゲートウェイ間の eBGP ピアリングだけがサポートされます。
- 各サイトには、TRM アンダーレイ用のローカル RP が必要です。
- 各サイトのアンダーレイ ユニキャスト ルーティングを、別のサイトのアンダーレイ ユニキャスト ルーティングから分離します。この要件は、マルチサイトにも適用されます。
- MVPN アドレス ファミリは、BGW 間で有効にする必要があります。
- 外部マルチキャスト ファブリックへの BGW 接続を設定する場合は、次の点に注意してください。
  - サイトのファブリック サイトにリーフがない場合でも、マルチキャスト アンダーレイはファブリック側のすべての BGW 間で設定する必要があります。
  - 単一サイトの BGW-BL ノードに VRF-Lite リンクを介してレイヤ 3 接続されている送信元と受信者は、外部レイヤ 3 ネットワークを介して到達可能である必要があります。同じサイトの BGBL-Node1 にレイヤ 3 接続された送信元があり、BGBL-Node2 にレイヤ 3 接続されたレシーバがある場合、これらの 2 つのエンドポイント間のトラフィックは、ファブリックを経由せずに外部のレイヤ 3 ネットワークを経由します。
  - 外部マルチキャスト ネットワークは、BGW または BL を介してのみ接続する必要があります。展開に同じサイトの BGW と BL の両方からの外部マルチキャスト ネットワーク接続が必要な場合は、BGW から学習した外部ルートが BL よりも優先されることを確認します。そのためには、BGW の BL よりも MED が低く、OSPF コストが（外部リンク上で）高くなる必要があります。

    次の図は、BGW-BL と内部リーフ（BL1）を介した外部ネットワーク接続を持つサイトを示しています。外部ソースへのパスは、リモート サイトの受信側での重複を避けるために、（BL2 ではなく）BGW-1 または BGW-2 を経由する必要があります。



- MED は iBGP でのみサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、VXLAN マルチサイト エニーキャスト BGWの以下の機能は、Cisco Nexus 9808/9804、9800-SUP-A、9808-FM-A/9804-FM-A スイッチおよび Cisco Nexus X9836DM-A と X98900CD-A ライン カードでもサポートされます。
  - TRMv4 がサポートされています。ただし、TRMv6 はサポートされていません。
  - DCI ピアの入力レプリケーション
  - ファブリック ピアのマルチキャスト アンダーレイ
  - DCI 側のマルチキャスト アンダーレイはサポートされていません。
  - ルートリーク、DSVNI、および共有サービスはサポートされていません。

マルチサイトでの TRM の設定

始める前に

次を設定する必要があります。

- VXLAN TRM
- VXLAN マルチサイト

このセクションは、TRM を使用するエニーキャスト BGW の設定手順を示します。TRM を使用する vPC BGW の場合、vxLAN TRM および VxLAN マルチサイトとともに vPC を設定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 terminal** 例： `switch# configure terminal`	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	**interface nve1** 例： `switch(config)# interface nve1`	NVE インターフェイスを設定します。
ステップ 3	**no shutdown** 例： `switch(config-if-nve)# no shutdown`	NVE インターフェイスを呼び出します。
ステップ 4	**host-reachability protocol bgp** 例： `switch(config-if-nve)# host-reachability protocol bgp`	これはホスト到達可能性のアドバタイズメント機構として BGP を定義します。
ステップ 5	**source-interface loopback** *src-if* 例： `switch(config-if-nve)# source-interface loopback 0`	送信元インターフェイスは、有効な /32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。
ステップ 6	**multisite border-gateway interface loopback** *vi-num* 例： `switch(config-if-nve)# multisite border-gateway interface loopback 1`	ボーダー ゲートウェイの仮想 IP アドレス（VIP）に使用されるループバック インターフェイスを定義します。border-gateway インターフェイスは、有効な /32 IP アドレスを持つスイッチ上に設定されているループバック インターフェイスにする必要があります。この/32 IP アドレスは、転送ネットワークの一時デバイスおよびリモート VTEP によって認識される必要があります。これは、転送ネットワークのダイナミック ルーティング プロトコルを介してアドレスを通知することによって、実現されます。このループバックは、送信元インターフェイスのループバックとは異なる必要があります。*vi-num* の範囲は、0 ～ 1023 です。

	コマンドまたはアクション	目的
ステップ 7	**member vni** *vni-range* **associate-vrf** 例： `switch(config-if-nve)# member vni 10010 associate-vrf`	仮想ネットワーク識別子（VNI）を設定します。 *vni-range* の範囲は 1ー16,777,214 です。*vni-range* の値は、5000 などの単一の値または 5001ー5008 などの範囲です。
ステップ 8	**mcast-group** *ip-addr* 例： `switch(config-if-nve-vni)# mcast-group 225.0.0.1`	ファブリック内の NVE マルチキャスト グループ IP プレフィックスを設定します。
ステップ 9	**multisite mcast-group** *dci-core-group address* 例： `switch(config-if-nve-vni)# multisite mcast-group 226.1.1.1`	DCI コアで TRM および L2 BUM パケットをカプセル化するために使用されるマルチキャスト グループを設定します。
ステップ 10	**multisite ingress-replication optimized** 例： `switch(config-if-nve-vni)# multisite ingress-replication optimized`	レイヤ 2 VNI を拡張するためのマルチサイト BUM レプリケーション方式を定義します。

マルチサイト設定による TRM の確認

マルチサイト設定の TRM のステータスを表示するには、次のコマンドを入力します。

コマンド	目的
show nve vni *virtual-network-identifier*	L3VNI を表示します。 （注） この機能では、Multi-Site 拡張 L3VNI のデフォルト設定は最適化された IR です。MS-IR フラグは本質的に、MS-IR が最適化されていることを意味します。

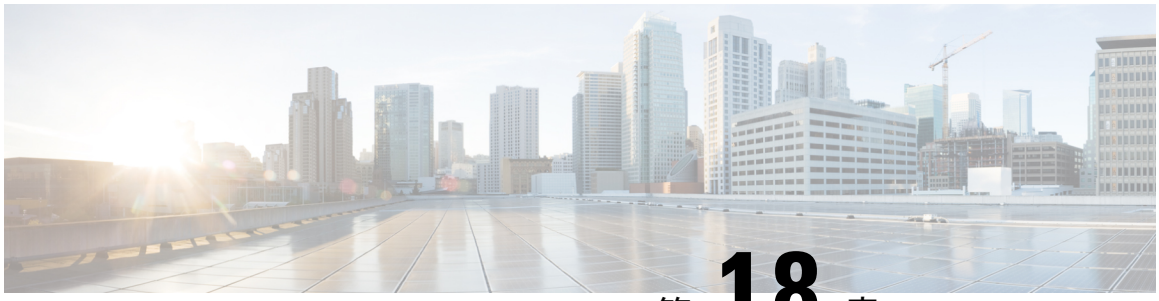
show nve vni コマンドの例：

インターネット ユーザに商品やサービスを提供する IPv4

```
switch(config)# show nve vni 51001
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP
       SU - Suppress Unknown Unicast
       Xconn - Crossconnect
       MS-IR - Multisite Ingress Replication

Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      51001    226.0.0.1         Up    CP   L3 [cust_1]        MS-IR
```

IPv6 の場合

```
switch(config)# show nve vni 90001
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP
       S-ND - Suppress ND
       SU - Suppress Unknown Unicast
       Xconn - Crossconnect
       MS-IR - Multisite Ingress Replication
       HYB - Hybrid IRB mode

Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      90001    ff03:ff03::101:1  Up    CP   L3 [v1]            MS-IR

switch(config)#
```



第 18 章

VXLAN EVPN トラフィック エンジニアリング：マルチサイト出力ロードバランシングの構成

この章では、Cisco NX-OS デバイスでVXLAN EVPN トラフィック エンジニアリング（TE）：マルチサイト出力ロードバランシング機能を構成する方法を説明します。

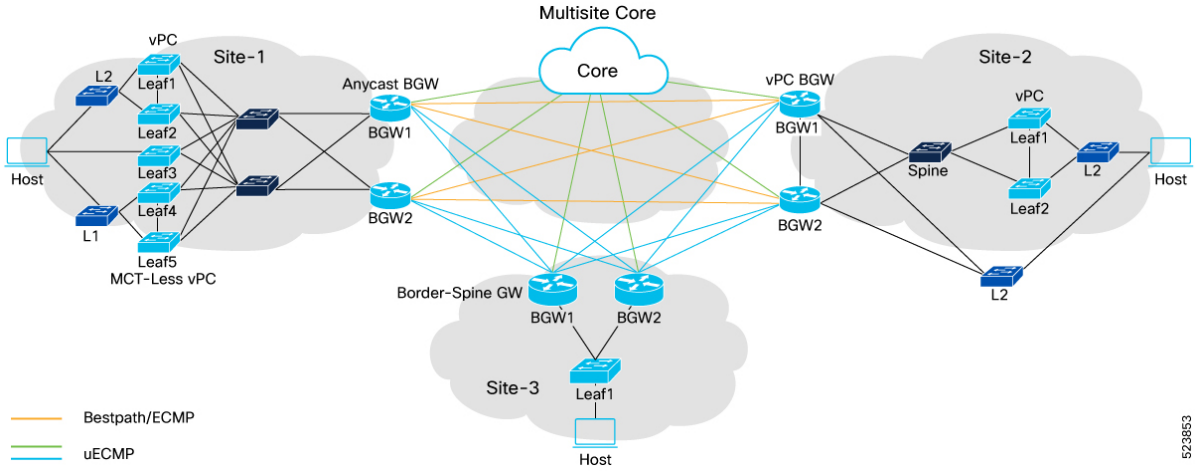
この章は、次の項で構成されています。

VXLAN EVPN TE - マルチサイト出力ロードバランシングの概要

VXLAN EVPN TE - マルチサイト出力ロードバランシング機能は、トラフィック ステアリングを促進し、マルチサイト リンクを介して異なるファブリック間で送信されるデータのロードバランシングを可能にします。

トラフィック エンジニアリングおよびロードバランシング機能は、通常サイト間ネットワーク（ISN）と呼ばれる IP ベースのアンダーレイ ネットワーク全体で動作します。したがって、これは基本的に、アンダーレイを介して送信される任意のオーバーレイ カプセル化トラフィックに適用できる IP トラフィック エンジニアリングとして提供されます。

VXLAN EVPN TE - マルチサイト出力ロードバランシングは、通常、データセンター（DC）間リンクの使用率を向上させます。



上記のトポロジは、同じマルチサイト ドメインの 3 つの VXLAN EVPN ファブリック部分を示しています。各ファブリックは、基本的にネットワーク インフラストラクチャの残りの部分とのファブリックのインターフェイスを表すボーダー ゲートウェイ デバイス（BGW）のローカル階層を介してリモート サイトに接続します。エニーキャスト BGW、vPC BGW、ボーダー ゲートウェイ スピンなど、さまざまなタイプの BGW は、この展開内で共存できます。これらは、BGW 間の直接リンクや汎用コア インフラストラクチャ（ISN）など、さまざまな接続オプションを介して相互接続できます。

- 通常、オレンジ色で示されたパスは、サイト 1 とサイト 2 に属するエンドポイント間のサイト間通信のベスト パスまたはマルチパスとして使用されます。
- VXLAN EVPN TE - マルチサイト出力ロードバランシング機能を有効にすることで、トラフィック分散用の追加パスをベストパス以外に使用できます。これには、サイト 3 と呼ばれる中間ロケーションを経由するルートなどの代替ルート（青色で強調表示）や、一般的なコアインフラストラクチャを通過するパス（緑色で強調表示）が含まれます。これらの代替パスは、不等コスト マルチパス（uECMP）または重み付け不等コスト マルチパス（wuECMP）の一部として使用できます。

VXLAN EVPN TE - マルチサイト出力ロードバランシングの注意事項および制限事項

- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN EVPN TE - マルチサイト出力ロードバランシング機能が Cisco Nexus 9300 FX/FX2/FX3/GX/GX2 スイッチ、および 9700-FX/GX ライン カードでサポートされます。ただし、現在サポートされているのは BGP ベースのアンダーレイ ルーティングのみです。
- VXLAN EVPN TE - マルチサイト出力ロードバランシング機能は、9500-FM-E ファブリック モジュールを搭載した Cisco Nexus 9500 モジュラ スイッチではサポートされていません。
- VXLAN EVPN TE - マルチサイト出力ロードバランシングは、次の機能をサポートします。

- リモート サイトへのすべてがベスト パスではない可能性があるアンダーレイ パス間での出力トラフィックのロード バランシング（LB）。
- 特定の「重み」（または負荷）を各マルチサイト パスに関連付けるための明示的および自動 LB ポリシー。これには、重み付け等コスト マルチパス（wECMP）と 重み付け不等コスト マルチパス（wuECMP）の 2 つのオプションが付属しています。
- BGP ベース アンダーレイ ルーティング構
- AS-Path ベースの uECMP パス選択。
- アンダーレイのレイヤ 3 ユニキャスト uECMP/wuECMP。
- レイヤ 3（EVPN タイプ 5）およびレイヤ 2（EVPN タイプ 2）オーバーレイ ユニキャスト ECMP/wuECMP。
- BUM 転送：
  - BUM トラフィックは、出力ロードバランシングの対象になりません。
  - ただし、入力複製がサイト間での BUM 転送（DCI IR）に使用される場合、アンダーレイのネクストホップ パスの選択は、マルチパス セットの出力ロードバランシング パスの一部から 1 つを使用できるため、アンダーレイの uECMP の利点があります。
- ソフトウェア転送：ソフトウェア転送のベースライン ケースでは、レイヤ2とレイヤ 3 の両方のトラフィックが EVPN VXLAN マルチサイト設定で uECMP/wuECMP をサポートします。
- CloudSec（PIP ネクストホップ）。
- ファイアウォール クラスタリング（PIP ネクストホップ）。
- VNI をダウンストリームします。ただし、 **dci-advertise-pip** を使用した DSVNI はサポートされていません。

（注） ネクストホップとして（マルチサイト VIP ではなく）PIP を使用してタイプ 2 およびタイプ 5 EVPN プレフィックスのアドバタイズを開始するには、BGW で **dci-advertise-pip** コマンドが必要です。オーバーレイとアンダーレイの動的重み（wuECMP）、およびアンダーレイの AIGP セクションでは、不等コスト ネクストホップ PIP アドレス宛てのオーバーレイ トラフィックに対して wuECMP ロードバランシングを実行する必要がある理由について詳しく説明します。

- VXLAN OAM。
- ポリシーベース ルーティング（PBR）

- VXLAN EVPN TE - マルチサイト出力ロードバランシングは、次の機能をサポートしていません。
  - IPv6 アンダーレイ を使用した VXLAN。
  - 重み付き ECMP/uECMP を使用した **hardware profile ecmp resilient** 構成。
  - マルチキャスト オーバーレイ トラフィック。
  - wuECMP は、9700-FX ライン カードを搭載した Cisco Nexus 9500 シリーズ スイッチではサポートされません。
- Cisco NX-OS リリース 10.4(3)F から以前のリリースにダウングレードする場合、VXLAN EVPN TE - マルチサイト出力ロードバランシング構成が削除されていることを確認します。

VXLAN EVPN TE：マルチサイト出力ロードバランシングの構成

次に、マルチサイト出力ロードバランシングを有効にするための3つの主な構成手順を示します。

（注） 次の構成を BGW にのみ適用します。

1. **フィルタ ポリシーの作成**：これは、同じマルチパス セットのアンダーレイ パス部分でトラフィックを分散するリモート宛先オーバーレイ ネクストホップ IP アドレスを識別するために必要です。このアドレスは、リモート BGW の共通のマルチサイト VIP または一意の PIP である可能性があります（**dci-advertise-pip** が構成されている場合）。
2. **マルチパス ポリシーの作成**：このポリシーは、アンダーレイ パスをマルチパス セットの一部として分類する基準を定義するように構成されます。この定義により、静的または動的な重みを持つ uECMP や wuECMP など、複数のユースケースのプロビジョニングが可能になります。具体的には、複数のオーバーレイ ネクストホップが存在する場合（BGW の PIP アドレスなど）、ネクストホップ アドレスの選択を含めるように wuECMP を拡張して、「マルチレベル」の負荷分散効果を作成できます。
3. **ELB VRF での解決の有効化**：デフォルト VRF のルーティング テーブルではなく、**egress-loadbalance-resolution- VRF** のアンダーレイ テーブルを使用して、宛先オーバーレイのネクストホップ IP アドレスに到達するためのアンダーレイ パスの解決を有効にします。

egress-loadbalance-resolution- VRF は、自動的に作成される新しい内部 VRF です。この VRF は構成できず、削除できません。

VXLAN EVPN TE - マルチサイト出力ロードバランシング機能が構成されている場合：

- アンダーレイ プロトコル（現在は BGP のみ）ルートが追加でインポートされ、このテーブルにインストールされます。
- オーバーレイのネクストホップ解決は、デフォルトのテーブルではなく、このテーブルを介して実行されます。
- これにより、デフォルト VRF にインストールされたベスト パスに加えて、サイト間通信により多くのアンダーレイ パスを使用できます。

アンダーレイの出力ロード バランス自動マルチパス ポリシーの作成

計算するアンダーレイ パスの最大数と、それらのアンダーレイ パスを同じマルチパス セットに割り当てる基準を指定するルートマップを使用して、自動マルチパス ポリシーを構成できます。このポリシーは、ベストパスと比較した場合に、AIGP メトリックや AS-Path の差など、1 つ以上の構成されたしきい値を照合できます。自動マルチパス ポリシーがない場合は、ベストパスのみがインストールされます。

アンダーレイの出力ロード バランシング自動マルチパス ポリシーを構成するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **ip prefix-list** *prefix-list-name* **seq** *value* **permit** *nexthop-ipaddress*
3. **route-map** *route-map-name*
4. **exit**
5. **router bgp** *as-number*
6. **address-family ipv4 unicast**
7. **[no] load-balance egress multipath auto-policy route-map** *route-map-name*

手順の詳細

手順

ステップ 1 configure terminal

例：

```
switch# config terminal
switch(config)#
```

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。

（注） プレフィックス リストが作成されていない場合にのみ、ステップ 2 に進みます。

ステップ 2 ip prefix-list *prefix-list-name* seq *value* permit *nexthop-ipaddress*

例：

```
switch(config)# ip prefix-list remote_nexthop seq 5 permit 10.10.112.1/32
```

リモート ネクストホップと一致するようにプレフィックス リストを構成します。

ステップ 3 **route-map** *route-map-name*

このルートマップは、アンダーレイ パスがベスト パスに対して等しくない（下位の）場合でも、同じマルチパス セットの一部としてグループ化します。これは、ベスト パスのこれらの値と比較したときに、これらのアンダーレイ パスの AIGP メトリックまたは AS パス長の構成された差に基づいて実行できます。

（注） 「match」および「set」コマンドを使用して、指定された要件に従ってシステムを構成します。

出力ロード バランシングの自動マルチパス ポリシーは、以下のように有効にできます：

例：

```
switch(config-route-map)# route-map ROUTE_MAP_2
```

a) **set maximum-paths** *max-path-value*

例：

```
switch(config-route-map)# set maximum-paths 5
```

出力ロードバランシングのために計算およびインストールされるマルチパスの最大数を構成します。範囲は、1 ～ 64 です。

および

b) **set as-path-length difference** *as-path-diff-value*

例：

```
switch(config-route-map)# set as-path-length difference 5
```

不等コスト ロードバランスに対してアンダーレイ パスで考慮する必要があるベスト パスと比較して、AS-Path-length の差を構成します。範囲は 1 ～ 255 です。

c) **set aigp-metric difference** *value*

例：

```
switch(config-route-map)# set aigp-metric difference 100
```

不等コスト ロードバランスに対してアンダーレイ パスで考慮する必要があるベスト パスと比較して、AIGP メトリック値の差を構成します。範囲は 1 ～ 4294967295 です。

（注） AIGP メトリック情報を構成および使用する方法の詳細については、「VXLAN EVPN TE の構成例：マルチサイト出力ロードバランシング」 を参照してください。

ステップ 4 **exit**

例：

```
switch(config-route-map)# exit
switch(config)#
```

BGP ルータ コンフィギュレーション モードを開始します。

ステップ 5 **router bgp** *as-number*

例：

```
switch(config)# router bgp 65001
switch(config-router)#
```

BGP ルータ コンフィギュレーション モードを開始します。

ステップ 6 **address-family ipv4 unicast**

例：

```
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)#
```

IPv4 ユニ キャスト アドレス ファミリを構成します。

ステップ 7 **[no] load-balance egress multipath auto-policy route-map** *route-map-name*

例：

```
switch(config-router-af)# load-balance egress multipath auto-policy route-map ROUTE_MAP_2
```

BGP での自動マルチパス選択とロード シェアリングを制御するパラメータ構成します。

パラメータ構成を削除するには、このコマンドの **no** 形式を使用します。

自動マルチパス ポリシーがない場合は、ベスト パスのみがインストールされます。

（注） コミュニティの一致を使用してプレフィックスを選択する場合、このプレフィックスのすべてのパスは、BGP への発信中にプレフィックスをタグ付けすることによって、出力 BGW をアドバタイズすることによって同じコミュニティでタグ付けされる必要があります。

オーバーレイの出力ロード バランシングの有効化

アンダーレイで **egress-loadbalance-resolution-vrf** を使用してオーバーレイ（EVPN）プレフィックスのネクストホップ解決を有効にするには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. **address-family l2vpn evpn**
4. **[no] nexthop load-balance egress multisite**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# config terminal`
	`switch(config)#`
		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	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 100`
	`switch(config-router)#`
		BGP ルータ 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	**address-family l2vpn evpn**
	例：
	`switch(config-router)# address-family l2vpn evpn`
	`switch((config-router-af)#`
		L2VPN アドレス ファミリを設定します。

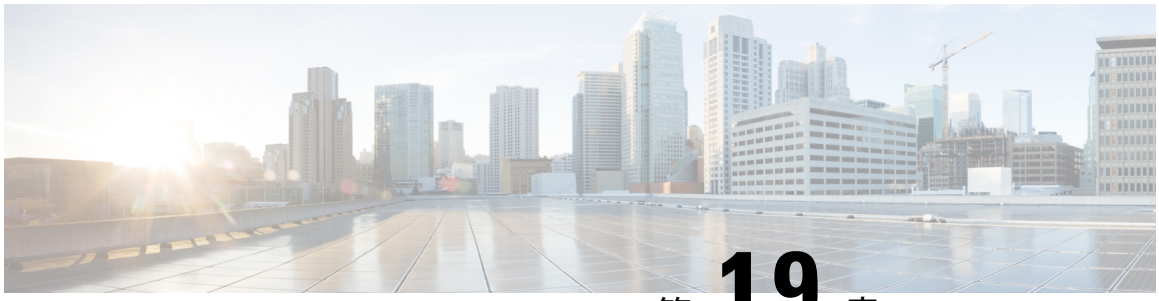
ステップ 4	**[no] nexthop load-balance egress multisite** 例： `switch(config-router-af)# nexthop load-balance egress multisite`	**egress-loadbalance-resolution- VRF** の対応する IPv4 または IPv6 テーブルを使用して、オーバーレイ（EVPN）ネクストホップ解決を有効にします。出力ロードバランス解決 VRF のテーブルの詳細については、VXLAN EVPN TE：マルチサイト出力ロードバランシングの構成（366 ページ）を参照してください。 オーバーレイの出力ロードバランシング構成を削除するには、このコマンドの **no** 形式を使用します。 **multisite** オプションは、この機能をマルチサイト ネットワークから学習された EVPN ネクストホップのみに制限します。これは、さまざまな VRF にインポートされたタイプ 2 とタイプ 5 の両方のルートに適用されます。 （注） この構成は、アンダーレイ テーブルで出力ロード バランスの計算を有効にした後に有効にする必要があります。

VXLAN EVPN TE の確認：マルチサイト出力ロードバランシング構成の確認

VXLAN EVPN TE：マルチサイト出力ロードバランシングの構成情報を表示するには、次のいずれかのコマンドを入力します。

コマンド	目的
show ip|ipv6 route [detail] vrf egress-loadbalance-resolution-	自動的に作成される特別な内部 VRF を表示します。この VRF は、BGP で出力ロード バランス構成が有効になっているときに、内部で暗黙的に使用されます。 BGP が ELB フィルタポリシーと自動マルチパス ポリシーで構成されている場合、デフォルト テーブルからルートのベスト パスを継承し、ELB ポリシーに基づいて追加の ELB パスを含めます。 **detail** オプションが有効になっている場合、wuECMP が構成されている場合、BGP が RIB に送信する重みが表示されます。 （注） Cisco NX-OS リリース 10.4(3)F 以降、**egress-loadbalance-resolution-** VRF のテーブル ID は、値 4089/0x0ff9 でスタティックに割り当てられ、割り当ての「limit-resource vrf」プールの外になります。 既存のユーザー構成には影響しません。
show ip|ipv6 route [detail] vrf tenant_vrf	デフォルト テーブルの代わりに **egress-loadbalance-resolution-table** を介してネクストホップが解決される EVPN-VXLAN テナント VRF のオーバーレイ プレフィックスを表示します。 **detail** オプションを有効にすると、wuECMP が構成されている場合、BGP から RIB に送信されるネクストホップに割り当てられた重みが表示されます。

コマンド	目的
show bgp ipv4 unicast *ipaddress* **vrf egress-loadbalance-resolution-**	AIGP がアンダーレイで構成されている場合は、派生 AIGP メトリックを含む、アンダーレイ BGP ルートとネクストホップを表示します。wuECMP の場合、ダイナミックに（つまり、AIGP メトリックまたはスタティックに構成された負荷分散重みから）導出される重みが表示されます。uECMP または ECMP の場合、重みは表示されません。
show l2route evpn mac all detail	wuECMP が構成されている場合、MAC ルートのネクストホップと重みを表示します。
show l2route evpn ead es detail	wuECMP が構成されている場合、EAD/ES ルートのネクストホップに関連付けられた重みを表示します。



第 19 章

テナント ルーテッド マルチキャストの設定

この章は、次の内容で構成されています。

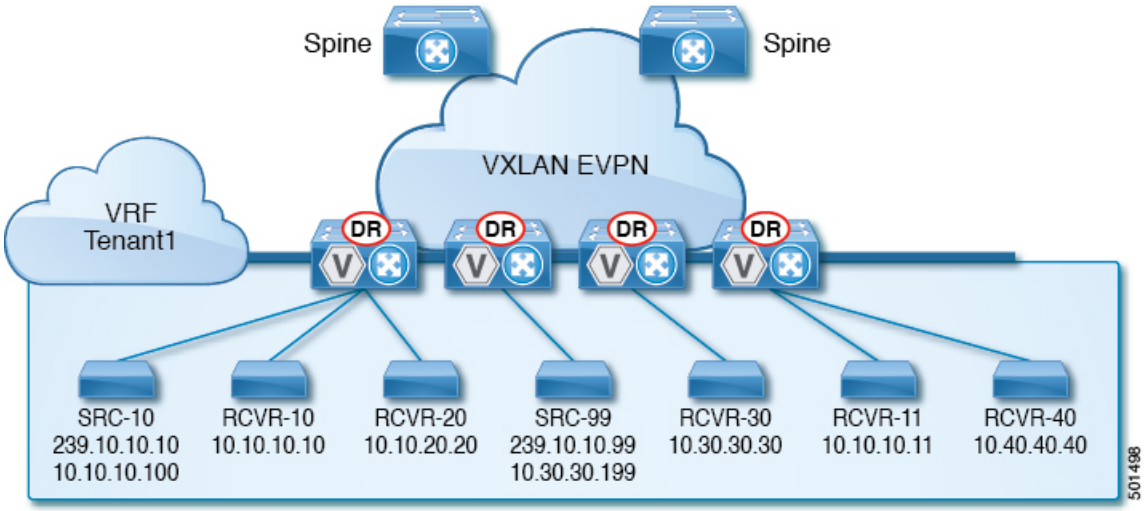
テナント ルーテッド マルチキャストについて

テナント ルーテッド マルチキャスト（TRM）は、BGP ベースの EVPN コントロール プレーンを使用する VXLAN ファブリック内でのマルチキャスト転送を有効にします。TRM は、ローカルまたは VTEP 間で同じサブネット内または異なるサブネット内の送信元と受信側の間にマルチテナント対応のマルチキャスト転送を実装します。

この機能により、VXLAN オーバーレイへのマルチキャスト配信の効率が向上します。これは、IETF RFC 6513、6514 で説明されている標準ベースの次世代コントロールプレーン（ngMVPN）に基づいています。TRM は、効率的かつ復元力のある方法で、マルチテナント ファブリック内で顧客の IP マルチキャストトラフィックを配布できるようにします。TRM の配布により、ネットワーク内のレイヤ 3 オーバーレイ マルチキャスト機能が向上します。

BGP EVPN はユニキャスト ルーティングのコントロール プレーンを提供しますが、ngMVPN はスケーラブルなマルチキャストルーティング機能を提供します。これは、ユニキャスト用の分散型 IP エニーキャスト ゲートウェイを持つすべてのエッジデバイス（VTEP）がマルチキャスト用の指定ルータ（DR）になる「常時ルート」アプローチに従います。ブリッジ型マルチキャスト転送は、エッジ デバイス（VTEP）にのみ存在し、IGMP スヌーピングは該当する受信者へのマルチキャスト転送を最適化します。ローカル配信以外のすべてのマルチキャスト トラフィックは効率的にルーティングされます。

図 *30 : VXLAN EVPN TRM*



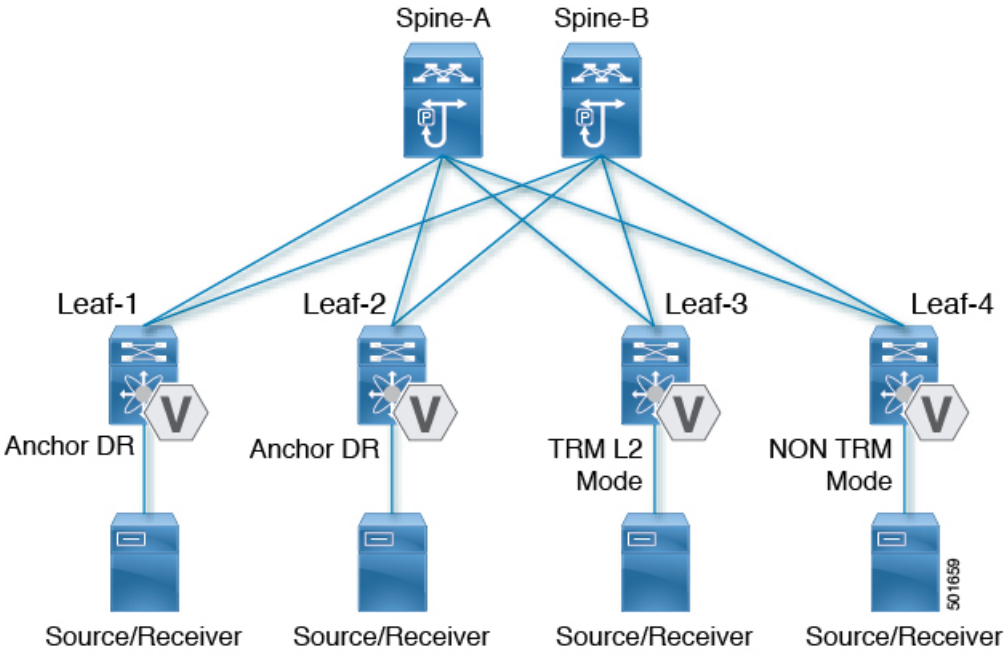
TRM を有効にすると、アンダーレイでのマルチキャスト転送が活用され、VXLAN でカプセル化されたルーテッド マルチキャストトラフィックが複製されます。デフォルト マルチキャスト配信ツリー（デフォルト MDT）は、VRF ごとに構築されます。これは、レイヤ 2 仮想ネットワーク インスタンス（VNI）のブロードキャストおよび不明ユニキャスト トラフィック、およびレイヤ 2 マルチキャスト複製グループの既存のマルチキャスト グループに追加されます。オーバーレイ内の個々のマルチキャスト グループ アドレスは、複製および転送のためにそれぞれのアンダーレイ マルチキャスト アドレスにマッピングされます。BGP ベースのアプローチを使用する利点は、TRM を備えた BGP EVPN VXLAN ファブリックが、すべてのエッ

ジ デバイスまたは VTEP に RP が存在する完全な分散型オーバーレイ ランデブー ポイント（RP）として動作できることです。

マルチキャスト対応のデータセンターファブリックは、通常、マルチキャストネットワーク全体の一部です。マルチキャスト送信元、受信側、およびマルチキャスト ランデブー ポイントはデータセンター内に存在する可能性がありますが、キャンパス内にある場合や WAN 経由で外部から到達可能である場合もあります。TRM を使用すると、既存のマルチキャスト ネットワークをシームレスに統合できます。ファブリック外部のマルチキャスト ランデブー ポイントを活用できます。さらに、TRM では、レイヤ 3 物理インターフェイスまたはサブインターフェイスを使用したテナント対応外部接続が可能です。

テナントルーテッドマルチキャスト混合モードについて

図 *31 : TRM* レイヤ *2*/レイヤ *3* 混合モード



Ipv6オーバーレイを使用するテナントルーテッドマルチキャストについて

Cisco NX-OSリリース10.2（1）以降、テナントルーテッドマルチキャスト（TRM）はオーバーレイでIPv6をサポートします。

IPv6 オーバーレイの TRM のガイドラインと制限事項

次は、IPv6オーバーレイを使用したTRMでサポートされます。

- ファブリック内のマルチキャストIPv4アンダーレイ。BidirおよびSSMはサポートされていません。
- マルチサイトのデータセンターコアのIPv4アンダーレイ。
- IPv4オーバーレイのみ、IPv6オーバーレイのみ、IPv4オーバーレイとIPv6オーバーレイの組み合わせ
- 境界リーフロールを持つエニーキャストボーダーゲートウェイ
- ボーダーゲートウェイおよびリーフでのvPCサポート
- リーフ上の仮想MCT
- エニーキャストRP（内部、外部、およびRP-everywhere）
- マルチサイト ボーダー ゲートウェイは、Cisco Nexus 9300 -FX3,、GX、GX2、-H2R、および -H1 TOR でサポートされます。
- エニーキャストRPによるRP-everywhereがサポートされます。
- TRMv6は、デフォルトのシステムルーティングモードでのみサポートされます。
- TRMを使用したVxLAN VLANによるMLDスヌーピング
- VLANでのPIM6 SVIおよびMLDスヌーピング設定はサポートされていません。
- IPv6 オーバーレイを使用する TRM は、Cisco Nexus 9300 -EX、-FX、- FX2、-FX3、-GX、- GX2、-H2R、-H1 TORでサポートされます。

次は、IPv6オーバーレイを使用したTRMではサポートされていません。

- L2 TRM
- L3TRMを使用したL2 VLANでのVXLANフラッドモードはサポートされません。
- L2-L3 TRM混合モード
- 単一サイト内のVXLAN入力レプリケーション
- アンダーレイのIPv6
- TRMなしのVXLAN VLANを使用したMLDスヌーピング
- MLDスヌーピングを使用しないPIM6 SVI設定
- MSDP

TRM フローのマルチキャスト フロー パスの可視性について

Cisco NX-OS リリース 10.3(2)F 以降、TRM フローのマルチキャスト フローパス可視化（FPV）機能は、すでにサポートされているマルチキャスト フローとともに、TRM L3 モードおよびアンダーレイ マルチキャストでサポートされます。この機能により、Cisco Nexus 9000 シリーズ スイッチのすべてのマルチキャスト ステートをエクスポートできます。これは、送信元から受信者までのフロー パスの完全で信頼性の高い追跡性を確保するのに役立ちます。Cisco Nexus 9000 シリーズ スイッチでマルチキャスト フローパス データ エクスポートを有効にするには、**multicast flow-path export** コマンドを使用します。

テナントルーテッドマルチキャストに関する注意事項と制限事項

テナント ルーテッド マルチキャスト（TRM）には、次の注意事項と制約事項があります。

- Cisco NX-OS リリース 10.1(2) 以降では、vPC BGW を使用した TRM マルチサイトがサポートされています。
- Cisco NX-OS リリース 10.2(1q)F 以降、VXLAN TRM は Cisco Nexus N9K-C9332D-GX2B プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN TRM は Cisco Nexus 9364D-GX2A および 9348D-GX2A プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN TRM は Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、VXLAN TRM は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN TRM は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- テナント ルーテッド マルチキャストが有効になっている場合、FEX はサポートされません。
- VXLAN TRM 機能が VTEP で有効になっている場合、VXLAN ファブリックへの IGMP メッセージの送信が停止します。
- VXLAN のガイドラインと制限事項は TRM にも適用されます。
- TRM が有効になっている場合、コアリンクとしての SVI はサポートされません。
- TRM が設定されている場合、ISSU は中断を伴います。

- TRM は IPv4 マルチキャストのみをサポートします。
- TRM には、スパース モードとも呼ばれる PIM Any Source Multicast（ASM）を使用した IPv4 マルチキャスト ベースのアンダーレイが必要です。
- TRM は、オーバーレイ PIM ASM および PIM SSM のみをサポートします。PIM BiDir はオーバーレイではサポートされていません。
- RP は、ファブリックの内部または外部のいずれかに設定する必要があります。
- 内部 RP は、ボーダー ノードを含むすべての TRM 対応 VTEP で設定する必要があります。
- 外部 RP は、ボーダー ノードの外部にある必要があります。
- RP は、外部 RP IP アドレス（スタティック RP）を指す VRF 内で設定する必要があります。これにより、特定の VRF の外部 RP に到達するためのユニキャストおよびマルチキャスト ルーティングが有効になります。
- トランジット ルーティング マルチキャスト（TRM）展開では、プロトコル独立マルチキャスト（PIM）対応インターフェイスでフラッピングが発生すると、RP-on-Stick モデルによってトラフィックがドロップされることがあります。RP につながるターンアラウンド ルータで **ip pim spt-switch-graceful** コマンドを使用します。このコマンドを使用すると、フラッピング中に最短パス ツリー（SPT）にグレースフルに切り替えることができ、トラフィックのドロップを最小限に抑えることができます。
- 最初のパケットの複製は、Cisco Nexus 9300（EX、FX、FX2 ファミリスイッチ）でのみサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、最初のパケットのレプリケーションは Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- マルチサイトでの TRM は、Cisco Nexus 9504-R プラットフォームではサポートされません。
- TRM は複数のボーダー ノードをサポートします。複数のボーダー リーフ スイッチを介した外部 RP/送信元への到達可能性は、ECMP でサポートされ、対称ユニキャスト ルーティングが必要です。
- VXLAN vPC セットアップで L3 VNI の VLAN で PIM と **ip igmp snooping vxlan** の両方を有効にする必要があります。
- 外部 RP を使用する内部ソースおよび外部 L3 レシーバを使用するトラフィック ストリームの場合、外部 L3 レシーバは PIM S、G 加入要求を内部ソースに送信することがあります。これを行うと、ファブリック FHR で S、G の再作成がトリガーされ、この S、G がクリアされるまでに最大 10 分かかることがあります。
- Cisco NX-OS リリース 10.3(1)F 以降、TRM のリアルタイム/フレックス統計は Cisco Nexus 9300-X クラウド スケール スイッチでサポートされています。

レイヤ3テナントルーテッドマルチキャストの注意事項と制約事項

レイヤ3テナントルーテッドマルチキャスト（TRM）には次の設定の注意事項と制限事項があります。

- Cisco NX-OS リリース 9.3(3) から Cisco NX-OS リリース 9.3(6) にアップグレードするとき、Cisco NX-OS リリース 9.3(3) から TRM 対応 VRF の設定を保持しない場合や、アップグレード後に新しい VRF を作成する場合、**feature ngmvpn** が有効な際に、**ip multicast multipath s-g-hash next-hop-based** CLI の自動生成は発生しません。TRM 対応 VRF ごとに CLI を手動で有効にする必要があります。
- レイヤ 3 TRM は、Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3/FXP および 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、レイヤ 3 TRM が Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、レイヤ 3 TRM は Cisco Nexus 9332D-H2R スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、レイヤ 3 TRM は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、レイヤ 3 TRM は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 9.3(7) 以降では、Cisco Nexus N9K-C9316D-GX、N9K-C9364C-GX、および N9K-X9716D-GX プラットフォーム スイッチは、レイヤ 3 TRM と EVPN マルチサイトの組み合わせをサポートしています。
- Cisco Nexus 9300-GX プラットフォーム スイッチは、Cisco NX-OS リリース 9.3(5) でのレイヤ 3 TRM と EVPN マルチサイトの組み合わせをサポートしていません。
- Cisco NX-OS リリース 10.2(3)F 以降、レイヤ 3 TRM と EVPN マルチサイトの組み合わせが Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、レイヤ 3 TRM と EVPN マルチサイトの組み合わせが Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、レイヤ 3 TRM と EVPN マルチサイトの組み合わせが Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、レイヤ 3 TRM と EVPN マルチサイトの組み合わせが Cisco Nexus 9364C-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 9.3(3) 以降、-R/RX ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチは、レイヤ 3 モードで TRM をサポートします。この

機能は、IPv4 オーバーレイでのみサポートされます。レイヤ 2 モードと L2/L3 混合モードはサポートされていません。

-R/RX ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチは、レイヤ 3 ユニキャスト トラフィックのボーダー リーフとして機能できます。エニーキャスト機能の場合、RP は内部、外部、またはあらゆる場所の RP にすることができます。

- TRM VXLAN BGP EVPN を設定する場合、次のプラットフォームがサポートされます。
  - Cisco Nexus 9200、9332C、9364C、9300-EX、および 9300-FX/FX2/FX3/FXP プラットフォーム スイッチ。
  - 9700-EX ライン カード、9700-FX ライン カード、または両方のライン カードを組み合わせた Cisco Nexus 9500 プラットフォーム スイッチ。
- レイヤ 3 TRM と VXLAN EVPN マルチサイトは、同じ物理スイッチでサポートされます。詳細については、「マルチサイトの設定」を参照してください。
- TRM マルチサイト機能は、-R/RX ライン カードを搭載した Cisco Nexus 9504 プラットフォーム スイッチではサポートされません。
- 一方または両方のVTEPが -R/RX ライン カードを備えた Cisco Nexus 9504 または 9508 プラットフォーム スイッチである場合、パケット TTL は 2 回デクリメントされます。1 回は送信元リーフの L3 VNI にルーティングするため、もう 1 回は宛先 L3 VNI から宛先リーフの宛先 VLAN に転送するためです。
- vPC ボーダーリーフを使用した TRM は、Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3/GX/GX2/H2R/H1 プラットフォーム スイッチと、-EX/FX または -R / RX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチでのみサポートされます。この機能をサポートするには、ボーダー リーフで **advertise-pip** コマンドと **advertise virtual-rmac** コマンドを有効にする必要があります。設定情報については、「VIP/PIP の設定」の項を参照してください。
- 既知のローカル スコープ マルチキャスト（224.0.0.0/24）は TRM から除外され、ブリッジされます。
- インターフェイス NVE がボーダー リーフでダウンした場合、VRF ごとの内部オーバーレイ RP をダウンする必要があります。
- Cisco NX-OS リリース 10.3(1)F 以降、新しい L3VNI モード CLI の TRM サポートが Cisco Nexus 9300-X クラウド スケール スイッチで提供されます。
- Cisco NXOS リリース 10.2(1)F 以降、TRM フローパスの可視化は、単一の VXLAN EVPN サイト内のフローでサポートされます。
- Cisco NXOS リリース 10.3(2)F 以降、TRM フロー パスの可視化のサポートは、Cisco Nexus 9000 シリーズ プラットフォーム スイッチの以下のトラフィック パターンに拡張されました。
  - TRM マルチサイト DCI マルチキャスト

- TRM マルチサイト DCI IR
- TRM データ MDT
- 仮想 MCT vPC 上の TRM
- 新しい L3VNI を使用した TRM
- BUM トラフィックの可視性はサポートされていません。

- Cisco NX-OS リリース 10.4(3)F 以降、Cisco Nexus X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチ上の TRM マルチサイト エニーキャスト BGW は、次の機能をサポートします。
  - TRMv4
  - コア全体の DCI ピア間での入力レプリケーション
  - ファブリック ピアのマルチキャスト アンダーレイ。
  - 新しい L3VNI モードのみがサポートされます。一方、従来の L3VNI モードはサポートされません。

  Cisco Nexus X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチ上の TRM マルチサイト エニーキャスト BGW は、次の機能をサポートしません。
  - TRMv6
  - Data MDT
  - コア全体の DCI ピア間のマルチキャスト アンダーレイはサポートされていません。

レイヤ 2/レイヤ 3 テナント ルーテッド マルチキャスト（混合モード）の注意事項と制約事項

レイヤ2 /レイヤ3テナントルーテッドマルチキャスト（TRM）には、次の設定の注意事項と制約事項があります。

- すべての TRM レイヤ 2/レイヤ 3 設定済みスイッチはアンカー DR である必要があります。これは、TRM レイヤ 2/レイヤ 3 では、同じトポロジ内に共存する TRM レイヤ 2 モードでスイッチを設定できるためです。このモードは、非 TRM およびレイヤ 2 TRM モードのエッジ デバイス（VTEP）が同じトポロジに存在する場合に必要です。
- アンカー DR はオーバーレイの RP である必要があります。
- アンカー DR には追加のループバックが必要です。
- 非 TRM およびレイヤ 2 TRM モード エッジ デバイス（VTEP）では、マルチキャスト対応 VLAN ごとに設定された IGMP スヌーピング クエリアが必要です。TRM マルチキャスト制御パケットは VXLAN 経由で転送されないため、すべての非 TRM およびレイヤ 2 TRM

モード エッジ デバイス（VTEP）には、この IGMP スヌーピング クエリア設定が必要です。

- IGMP スヌーピング クエリアの IP アドレスは、非 TRM およびレイヤ 2 TRM モードのエッジ デバイス（VTEP）で再利用できます。
- VPC ドメイン内の IGMP スヌーピング クエリアの IP アドレスは、VPC メンバーデバイスごとに異なる必要があります。
- インターフェイス NVE がボーダー リーフでダウンすると、VRF ごとの内部オーバーレイ RP がダウンします。
- **ip multicast overlay-distributed-dr** コマンドの設定中は、NVE インターフェイスをシャットダウンおよびシャットダウン解除する必要があります。
- Cisco NX-OS リリース 9.2(1) 以降では、vPC ボーダー リーフを使用した TRM がサポートされています。Advertise-PIP および Advertise Virtual-Rmac は、機能でサポートするためにボーダー リーフで有効にする必要があります。advertise-pip と advertise virtual-rmac の設定については、「VIP/PIP の設定」の項を参照してください。
- Anchor DR は次のハードウェア プラットフォームではサポートされません。
  - Cisco Nexus 9200、9300-EX および 9300-FX/FX2 プラットフォーム スイッチ
  - 9700-EX ライン カード、9700-FX ライン カード、または両方のライン カードの組み合わせを備えた Cisco Nexus 9500 プラットフォーム スイッチ
- Cisco NX-OS リリース 10.2(3)F 以降、アンカー DR は Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- レイヤ 2/レイヤ 3 テナント ルーテッド マルチキャスト（TRM）は、Cisco Nexus 9300-FX3/GX/GX2/H2R/H1 プラットフォーム スイッチではサポートされません。

テナント ルーテッド マルチキャストのランデブー ポイント

TRM を有効にすると、内部および外部 RP がサポートされます。次の表に、RP の位置付けがサポートされているか、サポートされていない最初のリリースを示します。

	RP 内部	**RP 外部**	**PIM ベースの RP Everywhere**
TRM L2 モード	N/A	N/A	N/A

	RP 内部	RP 外部	PIM ベースの RP Everywhere
TRM L3 モード	7.0(3)I7(1)、9.2(x)	7.0(3)I7(4)、9.2(3)	7.0(3)I7(5) 以降の 7.0(3)I7(x) リリースでサポート 9.2(x) ではサポートされない 次の Nexus 9000 スイッチの 9.3(1) 以降の NX-OS リリースでサポートされます。 • Cisco Nexus 9200 スイッチ シリーズ • Cisco Nexus 9364C プラットフォーム スイッチ • Cisco Nexus 9300-EX/FX/FX2 プラットフォーム スイッチ（Cisco Nexus 9300-FXP プラットフォーム スイッチを除く） Cisco NX-OSリリース 9.3(5) から始まるサポート対象 Cisco Nexus 9300-FX3 プラットフォーム スイッチ
TRM L2L3 モード	7.0(3)I7(1)、9.2(x)	N/A	N/A

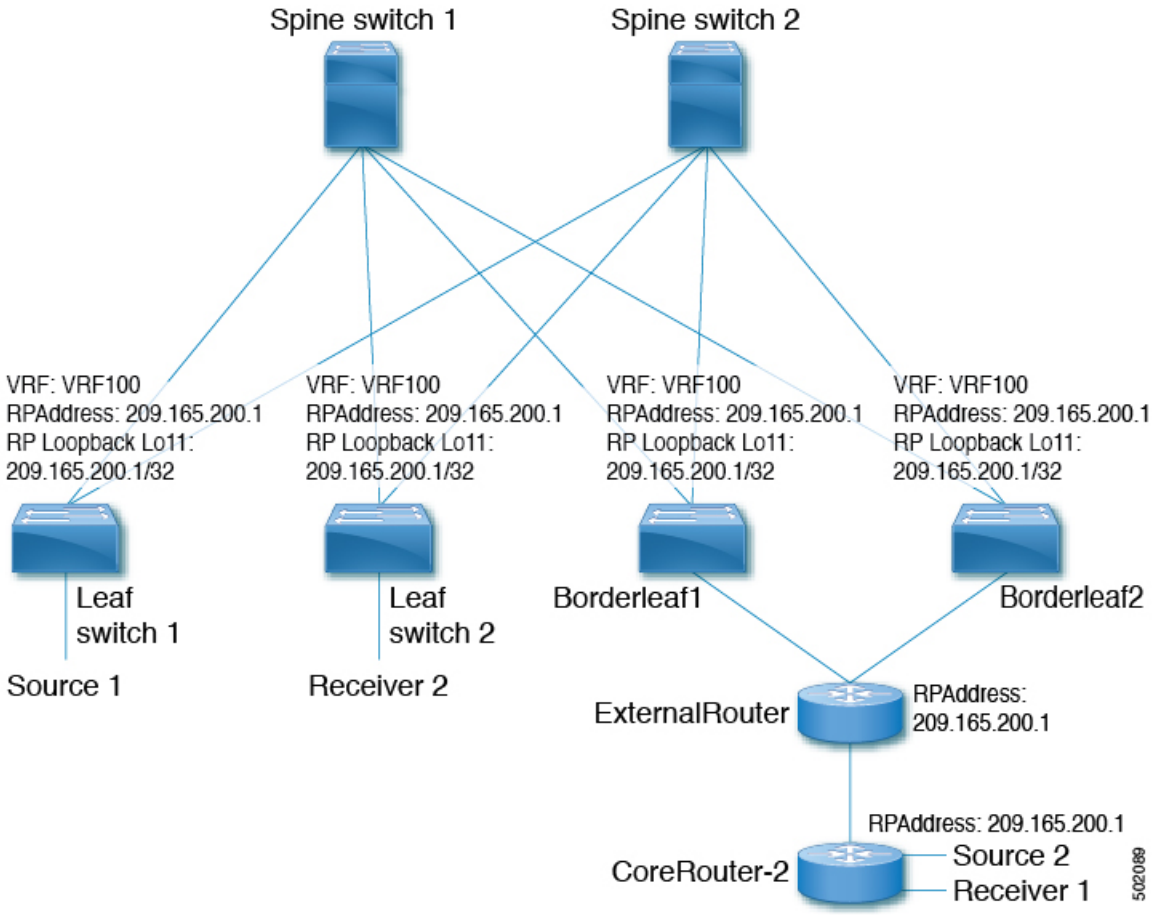
テナント ルーテッド マルチキャストのランデブー ポイントの設定

テナント ルーテッド マルチキャストでは、次のランデブー ポイント オプションがサポートされています。

- VXLAN ファブリック内のランデブー ポイントの設定 （384 ページ）

VXLAN ファブリック内のランデブー ポイントの設定

すべてのデバイス（VTEP）で次のコマンドを使用して、TRM VRF のループバックを設定します。EVPN 内で到達可能であることを確認します（アドバタイズ/再配布）。



手順の概要

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vxlan-number*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **vrf context** *vrf-name*

7. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

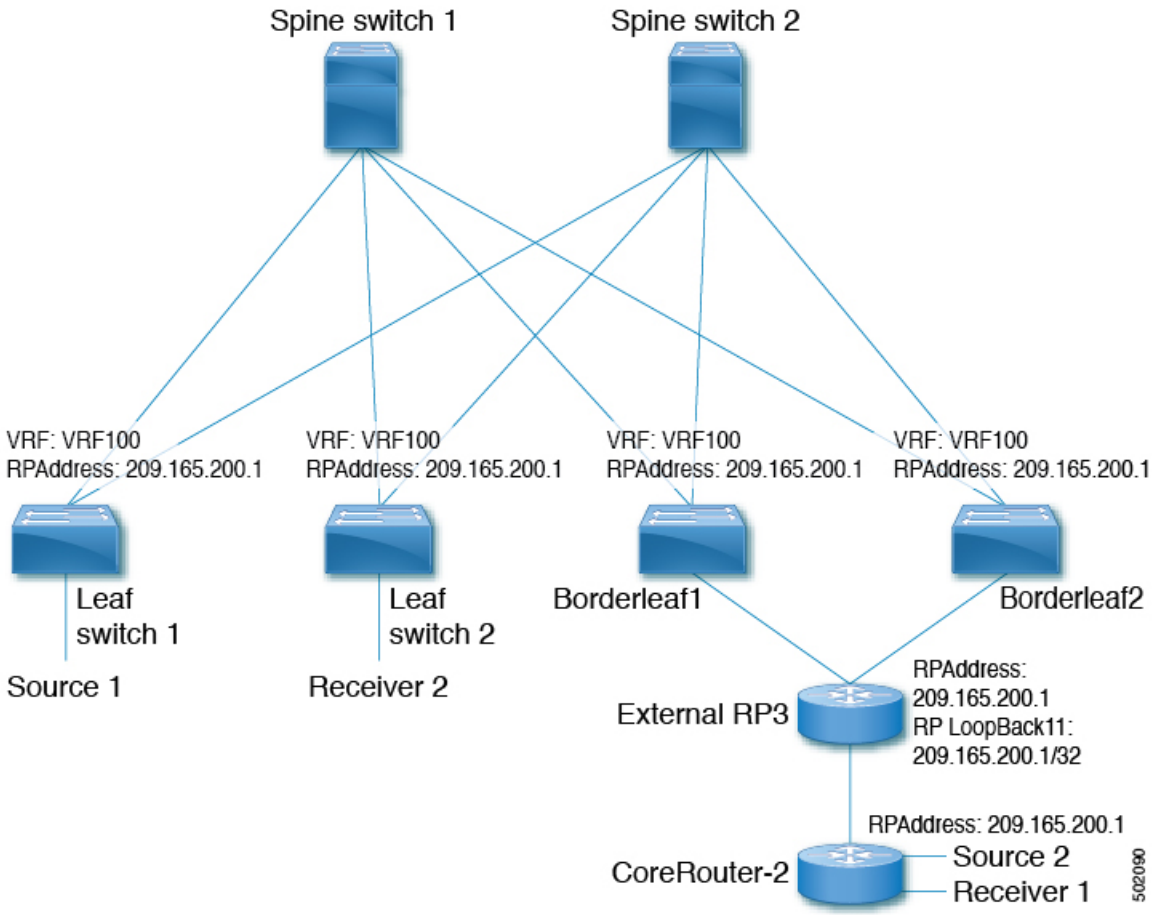
手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	すべての TRM 対応ノードでループバック インターフェイスを設定します。これにより、ファブリック内のランデブー ポイントが有効になります。
ステップ 3	**vrf member** *vxlan-number* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 4	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.1/32`	IP アドレスを指定します。
ステップ 5	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 6	**vrf context** *vrf-name* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 7	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジ デバイス（VTEP）に同じ IP アドレスが必要です。

外部ランデブー ポイントの設定

すべてのデバイス（VTEP）の TRM VRF 内の外部ランデブー ポイント（RP）IP アドレスを設定します。さらに、ボーダー ノードを介した VRF 内の外部 RP の到達可能性を確認します。



手順の概要

1. **configure terminal**
2. **vrf context vrf100**
3. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

手順の詳細

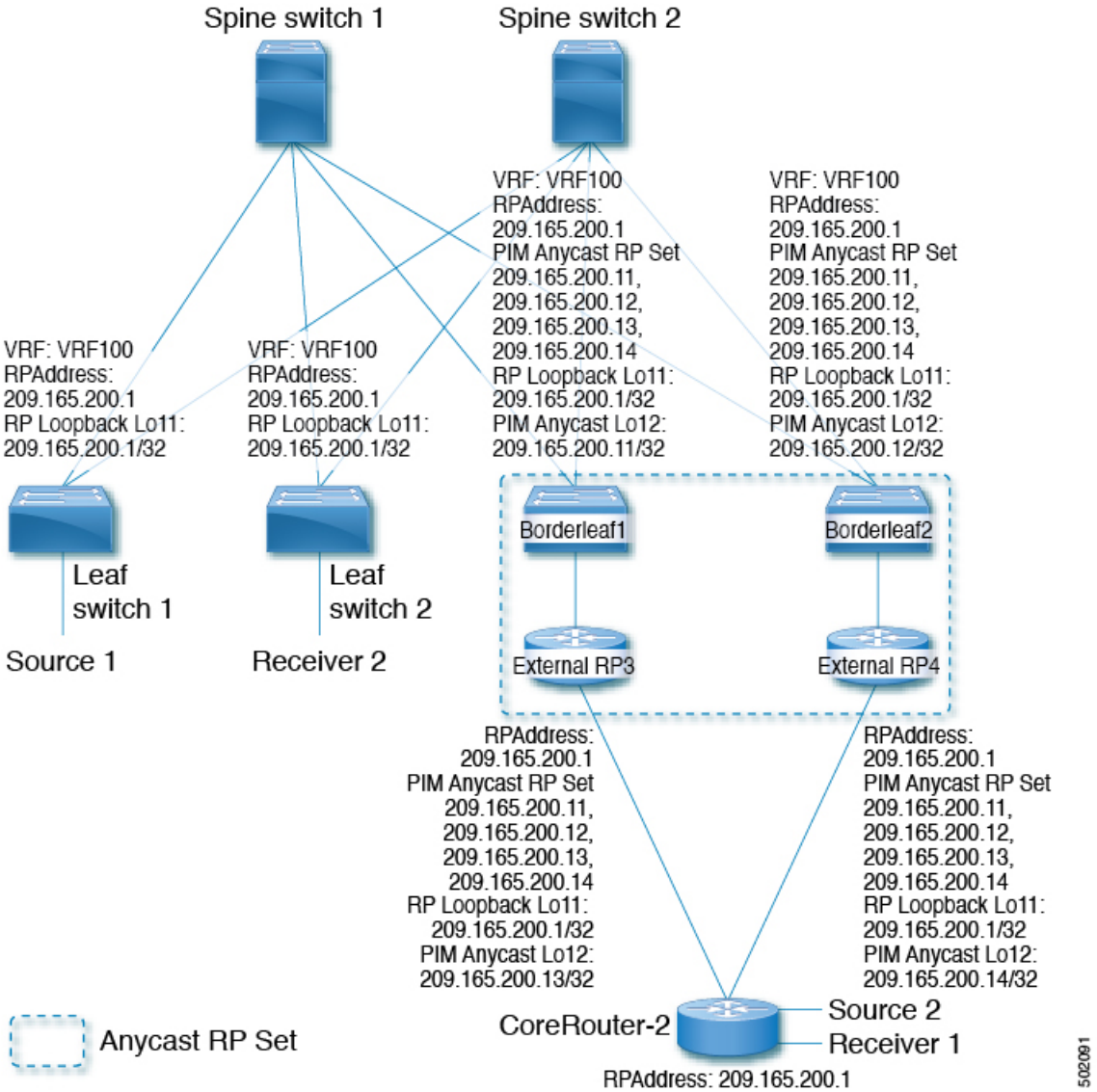
手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**vrf context vrf100** 例： `switch(config)# vrf context vrf100`	コンフィギュレーション モードを入力します。

	コマンドまたはアクション	目的
ステップ 3	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP のすべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。

PIM エニーキャストを使用した RP Everywhere の設定

PIM エニーキャスト ソリューションによる RP Everywhere の設定。



PIM エニーキャストを使用した RP Everywhere の設定については、次を参照してください。

- PIM エニーキャストを使用した RP Everywhere の TRM リーフ ノードの設定 （388 ページ）
- PIM エニーキャストを使用した RP Everywhere の TRM ボーダー リーフ ノードの設定 （389 ページ）
- PIM エニーキャストを使用した RP Everywhere の外部ルータの設定 （391 ページ）

PIM エニーキャストを使用した RP Everywhere の TRM リーフ ノードの設定

RP Everywhere のテナント ルーテッド マルチキャスト（TRM）リーフ ノードの設定。

手順の概要

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vrf-name*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **vrf context** *vxlan*
7. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	VXLAN VTEP でループバック インターフェイスを設定します。
ステップ 3	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 4	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.1/32`	IP アドレスを指定します。

	コマンドまたはアクション	目的
ステップ 5	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 6	**vrf context** *vxlan* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 7	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジ デバイス（VTEP）に同じ IP アドレスが必要です。

PIM エニーキャストを使用した RP Everywhere の TRM ボーダー リーフ ノードの設定

PIM エニーキャストを使用した RP Anywhere の TRM ボーダー リーフノードの設定。

手順の概要

1. **configure terminal**
2. **{ip | ipv6} pim evpn-border-leaf**
3. **interface loopback** *loopback_number*
4. **vrf member** *vrf-name*
5. **ip address** *ip-address*
6. **ipv6 pim sparse-mode**
7. **interface loopback** *loopback_number*
8. **vrf member** *vxlan-number*
9. **ipv6 address** *ipv6-address*
10. **ipv6 pim sparse-mode**
11. **vrf context** *vrf-name*
12. **ipv6 pim rp-address** *ipv6-address-of-router* **group-list** *group-range-prefix*
13. **ipv6 pim anycast-rp** *anycast-rp-address address-of-rp*
14. **ipv6 pim anycast-rp** *anycast-rp-address address-of-rp*
15. **ipv6 pim anycast-rp** *anycast-rp-address address-of-rp*
16. **ipv6 pim anycast-rp** *anycast-rp-address address-of-rp*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**{ip | ipv6} pim evpn-border-leaf** 例： `switch(config)# ipv6 pim evpn-border-leaf`	VXLAN VTEP を TRM ボーダー リーフノードとして設定します。
ステップ 3	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	VXLAN VTEP でループバック インターフェイスを設定します。
ステップ 4	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 5	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.1/32`	IP アドレスを指定します。
ステップ 6	**ipv6 pim sparse-mode** 例： `switch(config-if)# ipv6 pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 7	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 12`	PIM エニーキャスト set RP ループバック インターフェイスの設定
ステップ 8	**vrf member** *vxlan-number* 例： `switch(config-if)# vrf member vxlan-number`	VRF 名を設定します。
ステップ 9	**ipv6 address** *ipv6-address* 例： `switch(config-if)# ip address 209.165.200.11/32`	IP アドレスを指定します。
ステップ 10	**ipv6 pim sparse-mode** 例： `switch(config-if)# ipv6 pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。

	コマンドまたはアクション	目的
ステップ 11	**vrf context** *vrf-name* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 12	**ipv6 pim rp-address** *ipv6-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ipv6 pim rp-address 2090:165:200::1 group ff1e::/16`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。
ステップ 13	**ipv6 pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ipv6 pim anycast-rp 2090:165:2000::1 2090:165:2000::11`	PIM エニーキャスト RP セットを設定します。
ステップ 14	**ipv6 pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ipv6 pim anycast-rp 2090:165:2000::1 2090:165:2000::12`	PIM エニーキャスト RP セットを設定します。
ステップ 15	**ipv6 pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ipv6 pim anycast-rp 2090:165:2000::1 2090:165:2000::13`	PIM エニーキャスト RP セットを設定します。
ステップ 16	**ipv6 pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ipv6 pim anycast-rp 2090:165:2000::1 2090:165:2000::14`	PIM エニーキャスト RP セットを設定します。

PIM エニーキャストを使用した RP Everywhere の外部ルータの設定

RP Everywhere の外部ルータを設定するには、次の手順を使用します。

手順の概要

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vrf-name*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **interface loopback** *loopback_number*
7. **vrf member** *vxlan-number*
8. **ip address** *ip-address*

9. **ip pim sparse-mode**
10. **vrf context** *vxlan*
11. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
12. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
13. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
14. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
15. **ip pim anycast-rp** *anycast-rp-address address-of-rp*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	VXLAN VTEP でループバック インターフェイスを設定します。
ステップ 3	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vfr100`	VRF 名を設定します。
ステップ 4	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.1/32`	IP アドレスを指定します。
ステップ 5	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 6	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 12`	PIM エニーキャスト set RP ループバック インターフェイスの設定
ステップ 7	**vrf member** *vxlan-number* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 8	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.13/32`	IP アドレスを指定します。

	コマンドまたはアクション	目的
ステップ 9	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 10	**vrf context** *vxlan* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 11	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。
ステップ 12	**ip pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.11`	PIM エニーキャスト RP セットを設定します。
ステップ 13	**ip pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.12`	PIM エニーキャスト RP セットを設定します。
ステップ 14	**ip pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.13`	PIM エニーキャスト RP セットを設定します。
ステップ 15	**ip pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.14`	PIM エニーキャスト RP セットを設定します。

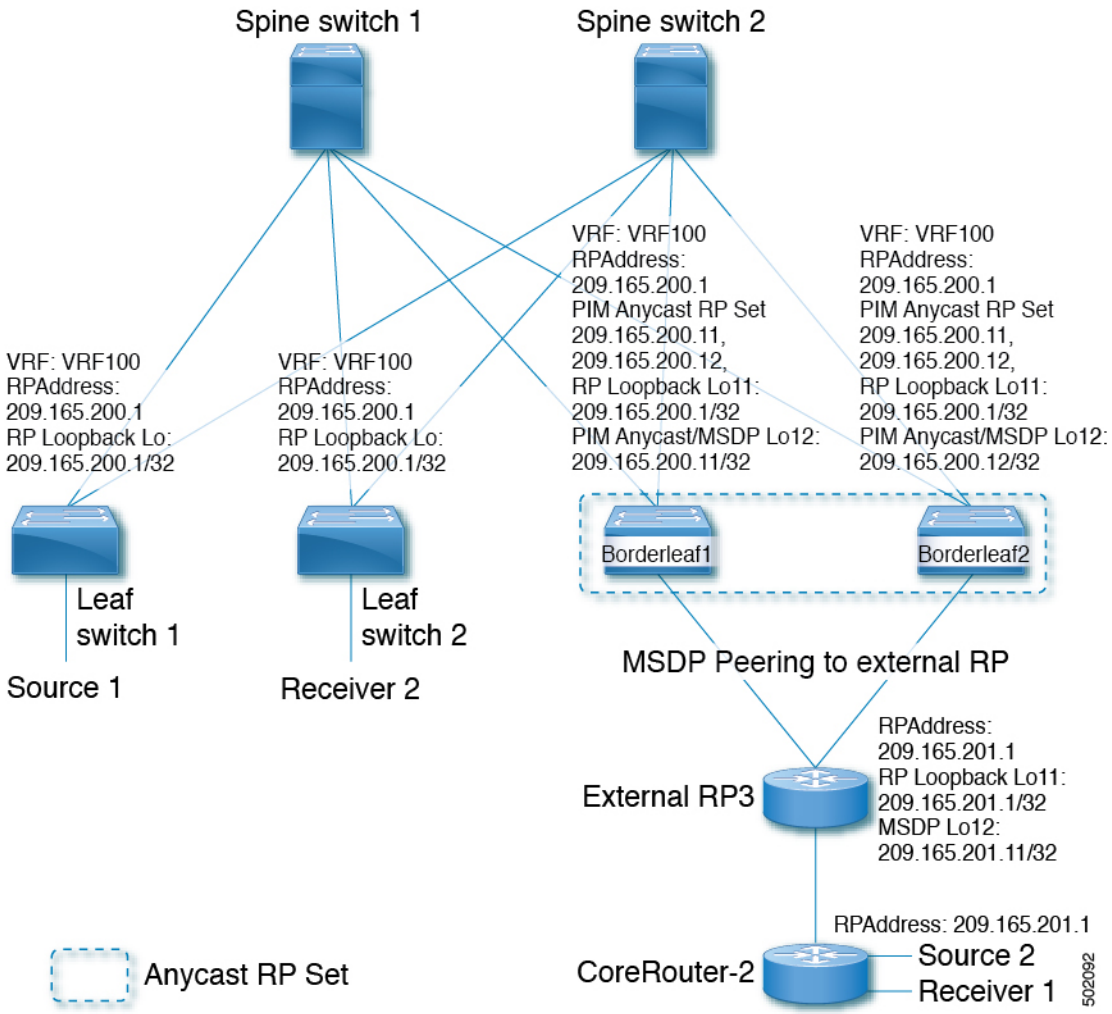
MSDP ピアリングを使用した RP Everywhere の設定

次の図では、MSDP RP ソリューションによる RP Everywhere の構成を示します。

MSDP ピアリングを使用した RP Everywhere の設定については、次を参照してください。

- MSDP ピアリングを使用した RP Everywhere の TRM リーフ ノードの設定 （394 ページ）
- MSDP ピアリングを使用した RP Everywhere の TRM ボーダー リーフ ノードの設定 （395 ページ）

• MSDP ピアリングを使用した RP Everywhere の外部ルータの設定 （398 ページ）



MSDP ピアリングを使用した RP Everywhere の TRM リーフ ノードの設定

MSDP ピアリングを使用した RP Everywhere の TRM リーフ ノードの設定。

手順の概要

1. **configure terminal**
2. **interface loopback** *loopback_number*
3. **vrf member** *vrf-name*
4. **ip address** *ip-address*
5. **ip pim sparse-mode**
6. **vrf context** *vrf-name*

7. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	VXLAN VTEP でループバック インターフェイスを設定します。
ステップ 3	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 4	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.1/32`	IP アドレスを指定します。
ステップ 5	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 6	**vrf context** *vrf-name* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 7	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジ デバイス（VTEP）に同じ IP アドレスが必要です。

MSDP ピアリングを使用した RP Everywhere の TRM ボーダー リーフ ノードの設定

PIM エニーキャストを使用した RP Everywhere の TRM ボーダー リーフを設定するには、次の手順を使用します。

手順の概要

1. **configure terminal**
2. **feature msdp**
3. **ip pim evpn-border-leaf**
4. **interface loopback** *loopback_number*
5. **vrf member** *vrf-name*
6. **ip address** *ip-address*
7. **ip pim sparse-mode**
8. **interface loopback** *loopback_number*
9. **vrf member** *vrf-name*
10. **ip address** *ip-address*
11. **ip pim sparse-mode**
12. **vrf context** *vrf-name*
13. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
14. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
15. **ip pim anycast-rp** *anycast-rp-address address-of-rp*
16. **ip msdp originator-id** *loopback*
17. **ip msdp peer** *ip-address* **connect-source** *loopback*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**feature msdp** 例： `switch(config)# feature msdp`	MSDP 機能を有効にします。
ステップ 3	**ip pim evpn-border-leaf** 例： `switch(config)# ip pim evpn-border-leaf`	VXLAN VTEP を TRM ボーダー リーフノードとして設定します。
ステップ 4	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	VXLAN VTEP でループバック インターフェイスを設定します。
ステップ 5	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。

	コマンドまたはアクション	目的
ステップ 6	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.1/32`	IP アドレスを指定します。
ステップ 7	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 8	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 12`	PIM エニーキャスト set RP ループバック インターフェイスの設定
ステップ 9	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 10	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.200.11/32`	IP アドレスを指定します。
ステップ 11	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 12	**vrf context** *vrf-name* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 13	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ip pim rp-address 209.165.200.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。
ステップ 14	**ip pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.11`	PIM エニーキャスト RP セットを設定します。
ステップ 15	**ip pim anycast-rp** *anycast-rp-address address-of-rp* 例： `switch(config-vrf)# ip pim anycast-rp 209.165.200.1 209.165.200.12`	PIM エニーキャスト RP セットを設定します。

	コマンドまたはアクション	目的
ステップ 16	**ip msdp originator-id** *loopback* 例： `switch(config-vrf)# ip msdp originator-id loopback12`	MSDP 発信者 ID を設定します。
ステップ 17	**ip msdp peer** *ip-address* **connect-source** *loopback* 例： `switch(config-vrf)# ip msdp peer 209.165.201.11 connect-source loopback12`	ボーダー ノードと外部 RP ルータ間の MSDP ピアリングを設定します。

MSDP ピアリングを使用した RP Everywhere の外部ルータの設定

手順の概要

1. **configure terminal**
2. **feature msdp**
3. **interface loopback** *loopback_number*
4. **vrf member** *vrf-name*
5. **ip address** *ip-address*
6. **ip pim sparse-mode**
7. **interface loopback** *loopback_number*
8. **vrf member** *vrf-name*
9. **ip address** *ip-address*
10. **ip pim sparse-mode**
11. **vrf context** *vrf-name*
12. **ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix*
13. **ip msdp originator-id loopback12**
14. **ip msdp peer** *ip-address* **connect-source loopback12**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**feature msdp** 例： `switch(config)# feature msdp`	MSDP 機能を有効にします。

	コマンドまたはアクション	目的
ステップ 3	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 11`	VXLAN VTEP でループバック インターフェイスを設定します。
ステップ 4	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 5	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.201.1/32`	IP アドレスを指定します。
ステップ 6	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 7	**interface loopback** *loopback_number* 例： `switch(config)# interface loopback 12`	PIM エニーキャスト set RP ループバック インターフェイスの設定
ステップ 8	**vrf member** *vrf-name* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 9	**ip address** *ip-address* 例： `switch(config-if)# ip address 209.165.201.11/32`	IP アドレスを指定します。
ステップ 10	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパースモード PIM を設定します。
ステップ 11	**vrf context** *vrf-name* 例： `switch(config-if)# vrf context vrf100`	VXLAN テナント VRF を作成します。
ステップ 12	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ip pim rp-address 209.165.201.1 group-list 224.0.0.0/4`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP の場合、すべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。

	コマンドまたはアクション	目的
ステップ 13	**ip msdp originator-id loopback12** 例： `switch(config-vrf)# ip msdp originator-id loopback12`	MSDP 発信者 ID を設定します。
ステップ 14	**ip msdp peer** *ip-address* **connect-source loopback12** 例： `switch(config-vrf)# ip msdp peer 209.165.200.11 connect-source loopback12`	外部 RP ルータとすべての TRM ボーダー ノード間の MSDP ピアリングを設定します。

レイヤ 3 テナント ルーテッド マルチキャストの設定

この手順では、テナントルーテッドマルチキャスト（TRM）機能を有効にします。TRMは、BGP MVPN シグナリングを使用して、主に IP マルチキャストのレイヤ 3 転送モードで動作します。レイヤ 3 モードの TRM は、TRM 対応 VXLAN BGP EVPN ファブリックの主要な機能であり、唯一の要件です。非 TRM 対応エッジ デバイス（VTEP）が存在する場合は、レイヤ 2/レイヤ 3 モードとレイヤ 2 モードを相互運用性について考慮する必要があります。

レイヤ 3 クラウドの送信者と受信者、および TRM vPC 境界リーフの VXLAN ファブリック間でマルチキャストを転送するには、VIP/PIP 設定を有効にする必要があります。詳細については、VIP/PIP の設定を参照してください。

（注） TRMは、always-routeアプローチに従って、転送されるIPマルチキャストトラフィックの存続可能時間（TTL）を減らします。

始める前に

VXLAN EVPN **feature nv overlay** および **nv overlay evpn** を設定する必要があります。

ランデブー ポイント（RP）を設定する必要があります。

TRM v4/v6 を有効化/無効化するには、PIM v4/v6 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 terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**feature ngmvpn** 例： `switch(config)# feature ngmvpn`	次世代マルチキャスト VPN（ngMVPN）コントロール プレーンを有効にします。BGP で新しいアドレス ファミリ コマンドが使用可能になります。 （注） **no feature ngmvpn** コマンドは、BGP の下の MVPN 構成を削除しません。 このコマンドを有効にすると、syslog メッセージが表示されます。このメッセージは、**ip multicast multipath s-g-hash next-hop-based** が推奨されるマルチパス ハッシュ アルゴリズムであり、TRM 対応 VRF に対して有効にする必要があることを通知します。 **ip multicast multipath s-g-hash next-hop-based** コマンドの自動生成は、**feature ngmvpn** コマンドをイネーブルにした後は行われません。VRF 設定の一部として **ip multicast multipath s-g-hash next-hop-based** を設定する必要があります。
ステップ 3	**ip igmp snooping vxlan** 例： `switch(config)# ip igmp snooping vxlan`	VXLAN VLAN の IGMP スヌーピングを設定します。
ステップ 4	**interface nve1** 例： `switch(config)# interface nve 1`	NVE インターフェイスを設定します。
ステップ 5	**member vni** *vni-range* **associate-vrf** 例： `switch(config-if-nve)# member vni 200100 associate-vrf`	レイヤ 3 仮想ネットワーク識別子を設定します。*vni-range* の範囲は 1 ～ 16,777,214 です。
ステップ 6	**mcast-group** *ip-prefix* 例： `switch(config-if-nve-vni)# mcast-group 225.3.3.3`	VRF VNI（レイヤ 3 VNI）のデフォルト マルチキャスト配信ツリーを構築します。 マルチキャスト グループは、関連付けられているレイヤ 3 VNI（VRF）内のすべてのマルチキャスト ルーティングのアンダーレイ（コア）で使用されます。 （注） レイヤ 2 VNI、デフォルト MDT、およびデータ MDT のアンダーレイ マルチキャスト グループは共有しないことを推奨します。重複しない個別のグループを使用します。

	コマンドまたはアクション	目的
ステップ 7	**exit** 例： `switch(config-if-nve-vni)# exit`	コマンド モードを終了します。
ステップ 8	**exit** 例： `switch(config-if)# exit`	コマンド モードを終了します。
ステップ 9	**router bgp <as-number>** 例： `switch(config)# router bgp 100`	自律システム番号の設定
ステップ 10	**vni** *number* 例： `switch(config-router)# vni 500001 l3`	テナント VRF の VNI を指定します。 Cisco NX-OS リリース 10.3(1)F 以降、新しい L3VNI 設定が有効になっていることを示すために **L3** キーワードが提供されています。 Cisco NX-OS リリース 10.4(3)F 以降、**L3** オプションを指定したこのコマンドは、Cisco Nexus X9836DMA および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチでサポートされます。
ステップ 11	**neighbor** *ip-addr* 例： `switch(config-router)# neighbor 1.1.1.1`	ネイバーの IP アドレスを設定します。
ステップ 12	**address-family ipv4 mvpn** 例： `switch(config-router-neighbor)# address-family ipv4 mvpn`	マルチキャスト VPN を設定します。
ステップ 13	**send-community extended** 例： `switch(config-router-neighbor-af)# send-community extended`	アドレス ファミリ シグナリングの ngMVPN をイネーブルにします。**send community extended** コマンドにより、拡張コミュニティがこのアドレス ファミリに確実に交換されます。
ステップ 14	**exit** 例： `switch(config-router-neighbor-af)# exit`	コマンド モードを終了します。
ステップ 15	**exit** 例： `switch(config-router)# exit`	コマンド モードを終了します。

	コマンドまたはアクション	目的
ステップ 16	**vrf context** *vrf_name* 例： `switch(config-router)# vrf context vrf100`	VRF 名を構成します。
ステップ 17	**mvpn vri id** <*id*> 例： `switch(config-router)#mvpn vri 100`	TRM の VRI を生成します。 router bgp <as-number> サブモードでこのコマンドを実行します。 vri id の範囲は 1 ～ 65535 です。 （注） このコマンドは vPC リーフ ノードで必須であり、値は vPC ペア全体で同じであり、TRM ドメインで一意である必要があります。また、値はサイト ID 値と衝突してはなりません。 （注） このコマンドは、site-id 値が 2 バイトを超え、値がすべての同じサイト BGW で同じで、TRM ドメインで一意である必要がある場合、BGW で必要です。また、値はサイト ID 値と衝突してはなりません。
ステップ 18	**[no] mdt** [*v4*|*v6*] vxlan 例： `switch(config-router)#mdt v4 vxlan`	指定された VRF で TRM v4/v6 を有効にします。TRM v4/v6 はデフォルトで有効になっています。 **no** オプションは、指定された VRF で TRM v4/v6 を無効にします。 新しい L3VNI 構成のサブモードでこのコマンドを実行します。 （注） このコマンドは、new-L3VNI で設定された VRF にのみ適用されます。
ステップ 19	**ip multicast multipath s-g-hash next-hop-based** 例： `switch(config-vrf)# ip multicast multipath s-g-hash next-hop-based`	RPF インターフェイスを選択するために、マルチキャスト マルチパスを設定し、（デフォルトの S/RP、G ベース ハッシュではなく）S、G、ネクストホップ ハッシュで開始させます。
ステップ 20	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ip pim rp-address 209.165.201.1 group-list 226.0.0.0/8`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP のすべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。 オーバーレイ RP の配置オプションについては、テナント ルーテッド マルチキャストのランデブー ポイントの設定（383 ページ）セクションを参照してください。

	コマンドまたはアクション	目的
ステップ 21	**address-family ipv4 unicast** 例： `switch(config-vrf)# address-family ipv4 unicast`	ユニ キャスト アドレス ファミリを設定します。
ステップ 22	**route-target both auto mvpn** 例： `switch(config-vrf-af-ipv4)# route-target both auto mvpn`	カスタマー マルチキャスト（C_Multicast）ルート（ngMVPN ルートタイプ 6 および 7）に拡張コミュニティ属性として追加される BGP ルート ターゲットを定義します。 自動ルート ターゲットは、2 バイトの自律システム番号（ASN）とレイヤ 3 VNI によって構築されます。
ステップ 23	**ip multicast overlay-spt-only** 例： `switch(config)# ip multicast overlay-spt-only`	送信元がローカルに接続されている場合の Gratuitally Originate（S、A）ルート。**ip multicast overlay-spt-only** コマンドは、すべての MVPN 対応 Cisco Nexus 9000 シリーズ スイッチ（通常はリーフ ノード）でデフォルトで有効になっています。
ステップ 24	**interface***vlan_id* 例： `switch(config)# interface vlan11`	ファーストホップ ゲートウェイ（レイヤ 2 VNI の分散エニーキャスト ゲートウェイ）を設定します。このインターフェイスでは、ルータ PIM ピアリングは発生しません。
ステップ 25	**no shutdown** 例： `switch(config-if)# no shutdown`	インターフェイスをディセーブルにします。
ステップ 26	**vrf member** *vrf-num* 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 27	**ipv6 address** *ipv6_address* 例： `switch(config-if)# ip address 11.1.1.1/24`	IP アドレスを設定します。
ステップ 28	**ipv6 pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	SVI で IGMP および PIM をイネーブルにします。これは、この VLAN にマルチキャスト送信元や受信者が存在する場合に必要です。
ステップ 29	**fabric forwarding mode anycast-gateway** 例： `switch(config-if)# fabric forwarding mode anycast-gateway`	エニーキャスト ゲートウェイ転送モードを設定します。

	コマンドまたはアクション	目的
ステップ 30	**ip pim neighbor-policy NONE*** 例： `switch(config-if)# ip pim neighbor-policy NONE*`	IP PIM ネイバー ポリシーを作成して、VLAN 内の PIM ルータとの PIM ネイバーシップを回避します。**none** キーワードは、すべての ipv4 アドレスを拒否するように構成されたルートマップで、エニーキャスト IP を使用した PIM ネイバーシップ ポリシーの確立を回避します。 （注） PIM ピアリングに分散型エニーキャスト ゲートウェイを使用しないでください。
ステップ 31	**exit** 例： `switch(config-if)# exit`	コマンド モードを終了します。
ステップ 32	**interface** *vlan_id* 例： `switch(config)# interface vlan100`	VRF およびレイヤ 3 VNI を設定します。
ステップ 33	**no shutdown** 例： `switch(config-if)# no shutdown`	インターフェイスを無効にします。
ステップ 34	**vrf member vrf100** 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 35	**ip forward** 例： `switch(config-if)# ip forward`	インターフェイスで IP 転送を有効にします。
ステップ 36	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパース モード PIM を設定します。レイヤ 3 VNI で発生する PIM ピアリングはありませんが、転送にはこのコマンドが必要です。

VXLAN EVPN スパインでの TRM の設定

この手順では、VXLAN EVPN スパイン スイッチでテナント ルーテッド マルチキャスト（TRM）を有効にします。

始める前に

VXLAN BGP EVPN スパインを設定する必要があります。スパインでの EVPN の iBGP の設定（151 ページ）を参照してください。

手順の概要

1. **configure terminal**
2. **route-map permitall permit 10**
3. **set ip next-hop unchanged**
4. **exit**
5. **router bgp [autonomous system]** *number*
6. **address-family ipv4 mvpn**
7. **retain route-target all**
8. **neighbor** *ip-address* [**remote-as** *number*]
9. **address-family ipv4 mvpn**
10. **disable-peer-as-check**
11. **rewrite-rt-asn**
12. **send-community extended**
13. **route-reflector-client**
14. **route-map permitall out**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**route-map permitall permit 10** 例： `switch(config)# route-map permitall permit 10`	ルート マップを設定します。 （注） ルート マップでは、EVPN ルート用にネクスト ホップを変更しないまま保持します。 • eBGP では必須です。 • iBGP ではオプションです。
ステップ 3	**set ip next-hop unchanged** 例：	ネクスト ホップ アドレスを設定します。

	コマンドまたはアクション	目的
	`switch(config-route-map)# set ip next-hop unchanged`	（注） ルート マップでは、EVPN ルート用にネクスト ホップを変更しないまま保持します。 • eBGP では必須です。 • iBGP ではオプションです。
ステップ 4	**exit** 例： `switch(config-route-map)# exit`	EXEC モードに戻ります。
ステップ 5	**router bgp [autonomous system]** *number* 例： `switch(config)# router bgp 65002`	BGP を指定します。
ステップ 6	**address-family ipv4 mvpn** 例： `switch(config-router)# address-family ipv4 mvpn`	BGP でアドレス ファミリ IPv4 MVPN を設定します。
ステップ 7	**retain route-target all** 例： `switch(config-router-af)# retain route-target all`	アドレス ファミリ IPv4 MVPN [global] で、すべてのルート ターゲットの保持を設定します。 （注） eBGP では必須です。インポート ルート ターゲットに一致するように設定されたローカル VNI が存在しない場合、スパインがすべての MVPN ルートを保持およびアドバタイズできるようにします。
ステップ 8	**neighbor** *ip-address* [**remote-as** *number*] 例： `switch(config-router-af)# neighbor 100.100.100.1`	ネイバーを定義します。
ステップ 9	**address-family ipv4 mvpn** 例： `switch(config-router-neighbor)# address-family ipv4 mvpn`	BGP ネイバーでアドレス ファミリ IPv4 MVPN を設定します。
ステップ 10	**disable-peer-as-check** 例： `switch(config-router-neighbor-af)# disable-peer-as-check`	ルート アドバタイズメント時のピア AS 番号のチェックをディセーブルにします。すべてのリーフが同じ AS を使用しているが、スパインがリーフと異なる AS を使用している場合、このパラメータを eBGP 用のスパインに設定します。

	コマンドまたはアクション	目的
		（注） eBGP では必須です。
ステップ 11	**rewrite-rt-asn** 例： `switch(config-router-neighbor-af)# rewrite-rt-asn`	発信ルート ターゲットの AS 番号をリモート AS 番号と一致するように正規化します。BGP で設定されたネイバーのリモート AS を使用します。**rewrite-rt-asn** コマンドは、Route Target Auto 機能を使用して EVPN ルート ターゲットを設定する場合に必要です。
ステップ 12	**send-community extended** 例： `switch(config-router-neighbor-af)# send-community extended`	BGP ネイバーのコミュニティを設定します。
ステップ 13	**route-reflector-client** 例： `switch(config-router-neighbor-af)# route-reflector-client`	ルート リフレクタを設定します。 （注） ルート リフレクタを使用する iBGP に必要です。
ステップ 14	**route-map permitall out** 例： `switch(config-router-neighbor-af)# route-map permitall out`	ルート マップを適用してネクストホップを変更しないまま保持します。 （注） eBGP では必須です。

レイヤ 2/レイヤ 3 混合モードでのテナント ルーテッド マルチキャストの設定

この手順では、テナント ルーテッド マルチキャスト（TRM）機能を有効にします。これにより、レイヤ 2 とレイヤ 3 の両方のマルチキャスト BGP シグナリングが有効になります。このモードは、TRM 以外のエッジ デバイス（VTEP）が Cisco Nexus 9000 シリーズ スイッチ（第 1 世代）に存在する場合にのみ必要です。Cisco Nexus 9000-EX および 9000-FX スイッチのみがレイヤ 2/レイヤ 3 モード（Anchor-DR）を実行できます。

レイヤ 3 クラウドの送信者と受信者、および TRM vPC 境界リーフの VXLAN ファブリック間でマルチキャストを転送するには、VIP/PIP 設定を有効にする必要があります。詳細については、VIP/PIP の設定を参照してください。

すべての Cisco Nexus 9300-EX および 9300-FX プラットフォーム スイッチはレイヤ 2/レイヤ 3 モードである必要があります。

始める前に

VXLAN EVPN を設定する必要があります。

ランデブー ポイント（RP）を設定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 terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**feature ngmvpn** 例： `switch(config)# feature ngmvpn`	次世代マルチキャスト VPN（ngMVPN）コントロール プレーンを有効にします。BGP で新しいアドレス ファミリ コマンドが使用可能になります。 （注） **no feature ngmvpn** コマンドは、BGP の下の MVPN 構成を削除しません。
ステップ 3	**advertise evpn multicast** 例： `switch(config)# advertise evpn multicast`	非 TRM 対応スイッチに向けて、IMET および SMET ルートを BGP EVPN にアドバタイズします。
ステップ 4	**ip igmp snooping vxlan** 例： `switch(config)# ip igmp snooping vxlan`	VXLAN VLAN の IGMP スヌーピングを設定します。
ステップ 5	**ip multicast overlay-spt-only** 例： `switch(config)# ip multicast overlay-spt-only`	送信元がローカルに接続されている場合に、(S,A) ルートを無償で発信します。この **ip multicast overlay-spt-only** コマンドは、すべての MVPN 対応 Cisco Nexus 9000 シリーズ スイッチ（通常はリーフ ノード）でデフォルトで有効になっています。
ステップ 6	**ip multicast overlay-distributed-dr** 例： `switch(config)# ip multicast overlay-distributed-dr`	この VTEP で分散アンカー DR 機能を有効にします。 （注） このコマンドを設定するときは、NVE インターフェイスをシャットおよびアンシャットする必要があります。
ステップ 7	**interface nve1** 例： `switch(config)# interface nve 1`	NVE インターフェイスを設定します。
ステップ 8	**[no] shutdown** 例： `switch(config-if-nve)# shutdown`	NVE インターフェイスをシャットダウンします。**no shutdown** コマンドは、インターフェイスを起動します。

	コマンドまたはアクション	目的
ステップ 9	**member vni** *vni-range* **associate-vrf** 例： `switch(config-if-nve)# member vni 200100 associate-vrf`	レイヤ 3 仮想ネットワーク識別子を設定します。*vni-range* の範囲は 1 ～ 16,777,214 です。
ステップ 10	**mcast-group** *ip-prefix* 例： `switch(config-if-nve-vni)# mcast-group 225.3.3.3`	分散アンカーDRのマルチキャストグループを設定します。
ステップ 11	**exit** 例： `switch(config-if-nve-vni)# exit`	コマンド モードを終了します。
ステップ 12	**interface loopback** *loopback_number* 例： `switch(config-if-nve)# interface loopback 10`	すべての分散アンカーDRデバイスでループバック インターフェイスを設定します。
ステップ 13	**ip address** *ip_address* 例： `switch(config-if)# ip address 100.100.1.1/32`	IP アドレスを設定します。この IP アドレスは、すべての分散アンカー DR で同じです。
ステップ 14	**ip router ospf** *process-tag* **area** *ospf-id* 例： `switch(config-if)# ip router ospf 100 area 0.0.0.0`	IP アドレス形式の OSPF エリア ID
ステップ 15	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパース モード PIM を設定します。
ステップ 16	**interface nve1** 例： `switch(config-if)# interface nve1`	NVE インターフェイスを設定します。
ステップ 17	**shutdown** 例： `switch(config-if-nve)# shutdown`	インターフェイスを無効にします。
ステップ 18	**mcast-routing override source-interface loopback** *int-num* 例： `switch(config-if-nve)# mcast-routing override source-interface loopback 10`	TRMがVTEPのデフォルトの送信元インターフェイスとは異なるループバックインターフェイスを使用していることをイネーブルにします。 *loopback10* 変数は、同じIPアドレスを持つアンダーレイ内のすべてのTRM対応VTEP（アンカーDR）

	コマンドまたはアクション	目的
		で設定する必要があります。このループバックとそれぞれの **override** コマンドは、TRM VTEP を非 TRM VTEP と共存させるために必要です。
ステップ 19	**exit** 例： `switch(config-if-nve)# exit`	コマンド モードを終了します。
ステップ 20	**router bgp 100** 例： `switch(config)# router bgp 100`	自律システム番号の設定
ステップ 21	**neighbor** *ip-addr* 例： `switch(config-router)# neighbor 1.1.1.1`	ネイバーの IP アドレスを設定します。
ステップ 22	**address-family ipv4 mvpn** 例： `switch(config-router-neighbor)# address-family ipv4 mvpn`	マルチキャスト VPN を設定します。
ステップ 23	**send-community extended** 例： `switch(config-router-neighbor-af)# send-community extended`	コミュニティ属性を送信します。
ステップ 24	**exit** 例： `switch(config-router-neighbor-af)# exit`	コマンド モードを終了します。
ステップ 25	**exit** 例： `switch(config-router)# exit`	コマンド モードを終了します。
ステップ 26	**vrf** *vrf_name* **vrf100** 例： `switch(config)# vrf context vrf100`	VRF 名を設定します。
ステップ 27	**ip pim rp-address** *ip-address-of-router* **group-list** *group-range-prefix* 例： `switch(config-vrf)# ip pim rp-address 209.165.201.1 group-list 226.0.0.0/8`	*ip-address-of-router* パラメータの値は RP の値です。完全に分散された RP のすべてのエッジデバイス（VTEP）に同じ IP アドレスが必要です。 オーバーレイ RP の配置オプションについては、テナント ルーテッド マルチキャストのランデブー ポ

	コマンドまたはアクション	目的
		イントの設定（383 ページ）-「内部RP」の項を参照してください。
ステップ 28	**address-family ipv4 unicast** 例： `switch(config-vrf)# address-family ipv4 unicast`	ユニ キャスト アドレス ファミリを設定します。
ステップ 29	**route-target both auto mvpn** 例： `switch(config-vrf-af-ipv4)# route-target both auto mvpn`	mvpn ルートのターゲットを指定します。
ステップ 30	**exit** 例： `switch(config-vrf-af-ipv4)# exit`	コマンド モードを終了します。
ステップ 31	**exit** 例： `switch(config-vrf)# exit`	コマンド モードを終了します。
ステップ 32	**interface** *vlan_id* 例： `switch(config)# interface vlan11`	レイヤ 2 VNIを設定します。
ステップ 33	**no shutdown** 例： `switch(config-if)# no shutdown`	インターフェイスを無効にします。
ステップ 34	**vrf member vrf100** 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 35	**ip address** *ip_address* 例： `switch(config-if)# ip address 11.1.1.1/24`	IP アドレスを設定します。
ステップ 36	**ip pim sparse-mode** 例： e `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパース モード PIM を設定します。
ステップ 37	**fabric forwarding mode anycast-gateway** 例：	エニーキャスト ゲートウェイ転送モードを設定します。

	コマンドまたはアクション	目的
	`switch(config-if)# fabric forwarding mode anycast-gateway`	
ステップ 38	**ip pim neighbor-policy NONE*** 例： `switch(config-if)# ip pim neighbor-policy NONE*`	**none** キーワードは、任意の IP を使用して PIM ネイバーシップ ポリシーの確立を回避するために IPv4 アドレスを拒否するように設定されたルート マップです。
ステップ 39	**exit** 例： `switch(config-if)# exit`	コマンド モードを終了します。
ステップ 40	**interface** *vlan_id* 例： `switch(config)# interface vlan100`	VRF およびレイヤ 3 VNI を設定します。
ステップ 41	**no shutdown** 例： `switch(config-if)# no shutdown`	インターフェイスを無効にします。
ステップ 42	**vrf member vrf100** 例： `switch(config-if)# vrf member vrf100`	VRF 名を設定します。
ステップ 43	**ip forward** 例： `switch(config-if)# ip forward`	インターフェイスで IP 転送を有効にします。
ステップ 44	**ip pim sparse-mode** 例： `switch(config-if)# ip pim sparse-mode`	インターフェイスでスパース モード PIM を設定します。

レイヤ 2 テナント ルーテッド マルチキャストの設定

この手順では、テナント ルーテッド マルチキャスト（TRM）機能を有効にします。これにより、レイヤ 2 マルチキャスト BGP シグナリングが有効になります。

IGMP スヌーピング クエリアは、すべてのレイヤ 2 TRM リーフ スイッチのマルチキャスト対応 VXLAN VLAN ごとに設定する必要があります。

始める前に

VXLAN EVPN を設定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 terminal** 例： `switch# configure terminal`	コンフィギュレーション モードを入力します。
ステップ 2	**feature ngmvpn** 例： `switch(config)# feature ngmvpn`	EVPN/MVPN 機能をイネーブルにします。 （注） **no feature ngmvpn** コマンドは、BGP の下の MVPN 構成を削除しません。
ステップ 3	**advertise evpn multicast** 例： `switch(config)# advertise evpn multicast`	L2 マルチキャスト機能をアドバタイズします。
ステップ 4	**ip igmp snooping vxlan** 例： `switch(config)# ip igmp snooping vxlan`	IGMP の設定スヌーピング VXLAN の場合。
ステップ 5	**vlan configuration** *vlan-id* 例： `switch(config)# vlan configuration 101`	VLAN 101 の設定モードを開始します。
ステップ 6	**ip igmp snooping querier** *querier-ip-address* 例： `switch(config-vlan-config)# ip igmp snooping querier 2.2.2.2`	マルチキャスト対応 VXLAN VLAN ごとに IGMP スヌーピング クエリアを設定します。

vPC サポートを使用した TRM の設定

このセクションでは、vPC サポートを使用して TRM を設定する手順について説明します。Cisco NX-OS リリース 10.1(2) 以降では、vPC BGW を使用した TRM マルチサイトがサポートされています。

手順の概要

1. **configure terminal**
2. **feature vpc**
3. **feature interface-vlan**
4. **feature lacp**
5. **feature pim**
6. **feature ospf**

7. **ip pim rp-address** *address* **group-list** *range*
8. **vpc domain** *domain-id*
9. **peer switch**
10. **peer gateway**
11. **peer-keepalive destination** *ipaddress*
12. **ip arp synchronize**
13. **ipv6 nd synchronize**
14. vPC ピアリンクを作成します。
15. **system nve infra-vlans** *range*
16. **vlan** *number*
17. SVI を作成します。
18. （任意） **delay restore interface-vlan** *seconds*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**feature vpc** 例： `switch(config)# feature vpc`	デバイス上で vPC をイネーブルにします。
ステップ 3	**feature interface-vlan** 例： `switch(config)# feature interface-vlan`	デバイスのインターフェイス VLAN 機能をイネーブルにします。
ステップ 4	**feature lacp** 例： `switch(config)# feature lacp`	デバイスの LACP 機能をイネーブルにします。
ステップ 5	**feature pim** 例： `switch(config)# feature pim`	デバイスの PIM 機能をイネーブルにします。
ステップ 6	**feature ospf** 例： `switch(config)# feature ospf`	デバイスの OSPF 機能をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 7	**ip pim rp-address** *address* **group-list** *range* 例： `switch(config)# ip pim rp-address 100.100.100.1 group-list 224.0.0/4`	アンダーレイ マルチキャスト グループ範囲に、PIM RP アドレスを設定します。
ステップ 8	**vpc domain** *domain-id* 例： `switch(config)# vpc domain 1`	デバイス上に vPC ドメインを作成し、設定目的で vpc-domain 設定モードを開始します。デフォルトはありません。範囲は 1 ～ 1000 です。
ステップ 9	**peer switch** 例： `switch(config-vpc-domain)# peer switch`	ピア スイッチを定義します。
ステップ 10	**peer gateway** 例： `switch(config-vpc-domain)# peer gateway`	仮想ポート チャネル（vPC）のゲートウェイ MAC アドレスを宛先とするパケットのレイヤ3転送をイネーブルにするには、**peer-gateway** コマンドを使用します。
ステップ 11	**peer-keepalive destination** *ipaddress* 例： `switch(config-vpc-domain)# peer-keepalive destination 172.28.230.85`	vPC ピアキープアライブ リンクのリモート エンドの IPv4 アドレスを設定します。 （注） vPC ピアキープアライブ リンクを設定するまで、vPC ピア リンクは構成されません。 管理ポートと VRF がデフォルトです。 （注） 独立した VRF を設定し、vPC ピアキープアライブ リンクのための VRF 内の各 vPC ピア デバイスからのレイヤ 3 ポートを使用することを推奨します。 VRF の作成および設定の詳細については、『Cisco Nexus 9000 Series NX-OS Series Unicast Routing Config Guide、9.3(x)』を参照してください。
ステップ 12	**ip arp synchronize** 例： `switch(config-vpc-domain)# ip arp synchronize`	vPC ドメインで IP ARP 同期を有効にして、デバイスのリロード後の ARP テーブルの生成を高速化します。
ステップ 13	**ipv6 nd synchronize** 例： `switch(config-vpc-domain)# ipv6 nd synchronize`	vPC ドメインで IPv6 nd 同期を有効にして、デバイスのリロード後の nd テーブルの高速化を促進します。

	コマンドまたはアクション	目的

ステップ 14	vPC ピアリンクを作成します。
	例：
	`switch(config)# interface port-channel 1`
	`switch(config)# switchport`
	`switch(config)# switchport mode trunk`
	`switch(config)# switchport trunk allowed vlan 1,10,100-200`
	`switch(config)# mtu 9216`
	`switch(config)# vpc peer-link`
	`switch(config)# no shut`
	
	`switch(config)# interface Ethernet 1/1, 1/21`
	`switch(config)# switchport`
	`switch(config)# mtu 9216`
	`switch(config)# channel-group 1 mode active`
	`switch(config)# no shutdown`
		vPC ピアリンク ポート チャネル インターフェイスを作成し、2つのメンバーインターフェイスを追加します。

ステップ 15	**system nve infra-vlans** *range* 例： `switch(config)# system nve infra-vlans 10`	バックアップ ルーテッド パスとして非 VXLAN 対応 VLAN を定義します。
ステップ 16	**vlan** *number* 例： `switch(config)# vlan 10`	インフラ VLAN として使用する VLAN を作成します。

ステップ 17	SVI を作成します。
	例：
	`switch(config)# interface vlan 10`
	`switch(config)# ip address 10.10.10.1/30`
	`switch(config)# ip router ospf process UNDERLAY area 0`
	`switch(config)# ip pim sparse-mode`
	`switch(config)# no ip redirects`
	`switch(config)# mtu 9216`
	`switch(config)# no shutdown`
		vPC ピアリンク上のバックアップルーテッドパスに使用される SVI を作成します。

ステップ 18	（任意） **delay restore interface-vlan** *seconds* 例： `switch(config-vpc-domain)# delay restore interface-vlan 45`	SVI の遅延復元タイマーをイネーブルにします。SVI/VNI スケールが大きい場合は、この値を調整することを推奨します。たとえば、SCIカウントが1000の場合、delay restoreを **interface-vlan** から 45 秒に設定することを推奨します。

vPC サポートを使用した TRM の設定（Cisco Nexus 9504-R および 9508-R）

手順の概要

1. **configure terminal**
2. **feature vpc**
3. **feature interface-vlan**
4. **feature lacp**
5. **feature pim**
6. **feature ospf**
7. **ip pim rp-address** *address* **group-list** *range*
8. **vpc domain** *domain-id*
9. **hardware access-list tcam region mac-ifacl**
10. **hardware access-list tcam region vxlan 10**
11. **reload**
12. **peer switch**
13. **peer gateway**
14. **peer-keepalive destination** *ipaddress*
15. **ip arp synchronize**
16. **ipv6 nd synchronize**
17. vPC ピアリンクを作成します。
18. **system nve infra-vlans** *range*
19. **vlan** *number*
20. SVI を作成します。
21. （任意） **delay restore interface-vlan** *seconds*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**feature vpc** 例： `switch(config)# feature vpc`	デバイス上で vPC をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 3	**feature interface-vlan** 例： `switch(config)# feature interface-vlan`	デバイスのインターフェイス VLAN 機能をイネーブルにします。
ステップ 4	**feature lacp** 例： `switch(config)# feature lacp`	デバイスの LACP 機能をイネーブルにします。
ステップ 5	**feature pim** 例： `switch(config)# feature pim`	デバイスの PIM 機能をイネーブルにします。
ステップ 6	**feature ospf** 例： `switch(config)# feature ospf`	デバイスの OSPF 機能をイネーブルにします。
ステップ 7	**ip pim rp-address** *address* **group-list** *range* 例： `switch(config)# ip pim rp-address 100.100.100.1 group-list 224.0.0/4`	アンダーレイ マルチキャスト グループ範囲に、PIM RP アドレスを設定します。
ステップ 8	**vpc domain** *domain-id* 例： `switch(config)# vpc domain 1`	デバイス上に vPC ドメインを作成し、設定目的で vpc-domain 設定モードを開始します。デフォルトはありません。範囲は 1 ～ 1000 です。
ステップ 9	**hardware access-list tcam region mac-ifacl** 例： `switch(config)# hardware access-list tcam region mac-ifacl 0`	ACL データベースの TCAM リージョンをカービングします。 （注） この TCAM カービング コマンドは、N9K-X9636C-RX ライン カードのみの TRM 転送を有効にするために必要です。**mac-ifacl** の TCAM リージョンが切り分けられていない場合、TCAM リソースは TRM に使用されます。
ステップ 10	**hardware access-list tcam region vxlan 10** 例： `switch(config)# hardware access-list tcam region vxlan 10`	VXLAN で使用する TCAM リージョンを割り当てます。 （注） この TCAM カービング コマンドは、N9K-X9636C-RX ライン カードのみの TRM 転送を有効にするために必要です。
ステップ 11	**reload** 例：	TCAM 割り当てのスイッチ設定をリロードして、アクティブにします。

	コマンドまたはアクション	目的
	`switch(config)# reload`	
ステップ 12	**peer switch** 例： `switch(config-vpc-domain)# peer switch`	ピア スイッチを定義します。
ステップ 13	**peer gateway** 例： `switch(config-vpc-domain)# peer gateway`	仮想ポート チャネル（vPC）のゲートウェイ MAC アドレスを宛先とするパケットのレイヤ3転送をイネーブルにするには、**peer-gateway** コマンドを使用します。
ステップ 14	**peer-keepalive destination** *ipaddress* 例： `switch(config-vpc-domain)# peer-keepalive destination 172.28.230.85`	vPC ピアキープアライブ リンクのリモート エンドの IPv4 アドレスを設定します。 （注） vPC ピアキープアライブ リンクを設定するまで、vPC ピア リンクは構成されません。 管理ポートと VRF がデフォルトです。 （注） 独立した VRF を設定し、vPC ピアキープアライブ リンクのための VRF 内の各 vPC ピア デバイスからのレイヤ 3 ポートを使用することを推奨します。 VRF の作成および設定の詳細については、『Cisco Nexus 9000 Series NX-OS Series Unicast Routing Config Guide、9.3(x)』を参照してください。
ステップ 15	**ip arp synchronize** 例： `switch(config-vpc-domain)# ip arp synchronize`	vPC ドメインで IP ARP 同期を有効にして、デバイスのリロード後の ARP テーブルの生成を高速化します。
ステップ 16	**ipv6 nd synchronize** 例： `switch(config-vpc-domain)# ipv6 nd synchronize`	vPC ドメインで IPv6 と同期を有効にして、デバイスのリロード後のテーブルの作成を高速化します。

ステップ 17	vPC ピアリンクを作成します。
	例：
	`switch(config)# interface port-channel 1`
	`switch(config)# switchport`
	`switch(config)# switchport mode trunk`
	`switch(config)# switchport trunk allowed vlan 1,10,100-200`
	`switch(config)# mtu 9216`
	`switch(config)# vpc peer-link`
		vPC ピアリンク ポート チャネル インターフェイスを作成し、2つのメンバー インターフェイスを追加します。

	コマンドまたはアクション	目的
	switch(config)# **no shut** switch(config)# **interface Ethernet 1/1, 1/21** switch(config)# **switchport** switch(config)# **mtu 9216** switch(config)# **channel-group 1 mode active** switch(config)# **no shutdown**	
ステップ 18	**system nve infra-vlans** *range* 例： switch(config)# **system nve infra-vlans 10**	バックアップ ルーテッド パスとして非 VXLAN 対応 VLAN を定義します。
ステップ 19	**vlan** *number* 例： switch(config)# **vlan 10**	インフラ VLAN として使用する VLAN を作成します。
ステップ 20	SVI を作成します。 例： switch(config)# **interface vlan 10** switch(config)# **ip address 10.10.10.1/30** switch(config)# **ip router ospf process UNDERLAY area 0** switch(config)# **ip pim sparse-mode** switch(config)# **no ip redirects** switch(config)# **mtu 9216** switch(config)# **no shutdown**	vPC ピアリンク上のバックアップルーテッドパスに使用される SVI を作成します。
ステップ 21	（任意） **delay restore interface-vlan** *seconds* 例： switch(config-vpc-domain)# **delay restore interface-vlan 45**	SVI の遅延復元タイマーをイネーブルにします。SVI/VNI スケールが大きい場合は、この値を調整することを推奨します。たとえば、SCIカウントが1000の場合、delay restoreを **interface-vlan** から 45 秒に設定することを推奨します。

TRM のフレックス統計

Cisco NX-OS リリース 10.3(1)F 以降、TRM のリアルタイム/フレックス統計が Cisco Nexus 9300-X Cloud スケールスイッチでオーバーレイ ルートに対してサポートされます。フレックス統計はアンダーレイ ルートではサポートされていません

（注） VXLAN NVE VNI 入力および出力、NVE ピアごとの入力、およびトンネル送信統計はサポートされません。

VXLAN TRM セットアップで、オーバーレイ mroute の mroute 統計が必要な場合は、デフォルト テンプレートで **hardware profile multicast flex-stats-enable** コマンドを構成する必要があり

ます。設定の詳細については、TRM のフレックス統計の構成 （422 ページ） を参照してください。

フレックス統計 CLI を有効にすると、次の CLI はサポートされなくなります。

- `sh nve vni <vni_id>/<all> counters`
- `sh nve peers <peer-ip> interface nve 1 counters`
- `sh int tunnel <Tunnel interface number> counters`

TRM のフレックス統計の構成

この手順では、VXLAN TRM セットアップでフレックス統計カウンタを有効/無効にします。

手順の概要

1. **configure terminal**
2. **[no] hardware profile multicast flex-stats-enable**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	構成モードに入ります。
ステップ 2	**[no] hardware profile multicast flex-stats-enable** 例： `switch(config)# hardware profile multicast flex-stats-enable`	TRM のフレックス統計を有効にします。 **no** オプションは、TRM のフレックス統計を無効にします。 （注） 構成中に行った変更を反映するには、スイッチがリロードされていることを確認してください。

TRM データ MDT の構成

TRM データ MDT について

テナント ルーテッド マルチキャスト（TRM）は、BGP ベースの EVPN コントロール プレーンを使用する VXLAN ファブリック内でのマルチキャスト転送を有効にします。TRM は、VTEP

のローカルまたはVTEP間で同じサブネット内または異なるサブネット内の送信元と受信側の間にマルチテナント対応のマルチキャスト転送を実装します。

既存の TRM ソリューションでは、デフォルトのマルチキャスト配布ツリー（デフォルトの MDT）を使用したマルチキャスト転送が可能です。デフォルトの MDT では、ノード（PE）は、オーバーレイに関心のある受信者が存在するかどうかに関係なく、常にアンダーレイでトラフィックを受信します。

このドキュメントで説明されているソリューションにより、S-PMSI（データ MDT）を使用して最適化されたマルチキャスト転送を実行できます。S-PMSIを使用すると、送信元トラフィックは選択的なマルチキャストトンネルにカプセル化されます。関心のある受信者を持つリーフのみが選択的マルチキャスト配信ツリーに参加します。

データ MDT へのスイッチオーバーは、即時にすることも、トラフィック帯域幅に基づいて行うこともできます（しきい値ベースの設定）。

TRM データ MDT の注意事項と制約事項

TRM データ MDT には、次の注意事項および制限事項があります。

- Cisco NX-OS リリース 10.3(2)F 以降、TRM データ MDT は、Cisco Nexus 9300 EX/FX/FX2/FX3/GX/GX2 スイッチ、および9700-EX/FX/GX ラインカードを備えた 9500 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、TRM データ MDT は Cisco Nexus 9332D-H2R スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、TRM データ MDT は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、TRM データ MDT は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- ファブリック内のデータ MDT は、特定の VRF の DCI IR でのみサポートされます。ファブリック内のデータ MDT は、サイト BGW の特定の VRF の DCI マルチキャストではサポートされません。
- データ MDT 構成は VRF 固有であり、L3 VRF で構成されます。
- 次の TRM データ MDT 機能がサポートされています。
  - データMDTでは、ASMおよびSSMグループ範囲がサポートされています。PIM-Biderアンダーレイは、データ MDT ではサポートされていません。
  - データMDTは、IPv4およびIPv6オーバーレイ マルチキャスト トラフィックをサポートします。
  - データ MDT は、vPC、VMCT リーフ、および vPC/エニーキャスト BGW によってサポートされます。また、L2、L3 オーファン/外部ネットワークは vPC ノードに接続できます。
  - L3 VRF ごとのデータ MDT 設定。

- データ MDT 発信（即時およびしきい値ベース）。
- 3 秒のデータ MDT カプセル化ルート プログラミング遅延。ユーザー定義の遅延は現在サポートされていません。

- L2、L2 -L3 混合モードはサポートされません。
- 新しい L3VNI モードがサポートされます。
- アンダーレイ グループ（L2 BUM、デフォルト MDT、およびデータ MDT グループ）の合計数が 512 であることを確認します。

TRM データ MDT の構成

次の手順に従って、TRM データ MDT を構成します：

始める前に

リアルタイム フロー レートに基づいてデータ MDT グループへの切り替えを有効にするには、次のコマンドが必要です。

hardware profile multicast flex-stats-enable

（注） このコマンドでは、スイッチのリロードが必要です。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **address-family {ipv4 | ipv6} unicast**
4. **[no] mdt data vxlan** <*group-range-1*> **[threshold]** [**route-map** <*value*> <*policy-name_1*>] [**seq** <*sequence-number*>]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**vrf context** *vrf-name* 例：	VRF を設定します。

	コマンドまたはアクション	目的
	`switch(config)# vrf context vrf1`	

ステップ 3	**address-family {ipv4 | ipv6} unicast**
	例：
	インターネット ユーザに商品やサービスを提供する IPv4
	`switch(config-vrf)# address-family ipv4 unicast`
	IPv6 の場合
	`switch(config-vrf)# address-family ipv6 unicast`
		IPv4 または IPv6 ユニキャスト アドレス ファミリを構成します。

ステップ 4	**[no] mdt data vxlan** <*group-range-1*> **[threshold]** [**route-map** <*value*> <*policy-name_1*>] [**seq** <*sequence-number*>] 例： `switch(config-vrf-af)# mdt data vxlan 224.7.8.0/24 route-map map1 10`	データ MDT は、アドレス ファミリごとに有効化/無効化できます。Cisco Nexus は、VRF 間およびアドレス ファミリ間の VRF 内でグループ範囲のオーバーラップをサポートします。 • しきい値とルートマップはオプションです。トラフィックのしきい値は、送信元のトラフィックであり、kbps で測定されます。しきい値を超えると、トラフィックがデータ MDT に切り替わるまでに 3 秒かかります。 • グループ範囲はコマンドキーの一部です。アドレス ファミリごとに複数のグループ範囲を設定できます。 • BUM およびデフォルトの MDT グループは、データ MDT グループと重複してはなりません。 • データ MDT は、重複する構成範囲を持つことができます。

TRM データ MDT の設定の検証

TRM データ MDT 構成情報を表示するには、次のいずれかのコマンドを入力します。

コマンド	目的
show nve vni { <*vni-id*> **| all } mdt [{ local | remote | peer-sync }] [{** <*cs*> <*cg*>**} | {** <*cs6*> <*cg6*>**}]**	顧客送信元（CS）、顧客グループ（ （DS）、データ グループ（DG）へ ます。
show nve vrf [x] mdt [local | remote | peer-sync] [y] [z]	VRF での CS、CG 割り当てを表示し
show bgp ipv4 mvpn route-type 3 detail	IPv4 オーバーレイ ルートの BGP S-P す。

コマンド	目的
show bgp ipv6 mvpn route-type 3 detail	IPv6 オーバーレイ ルートの BGP S-PMS す。
show fabric multicast [ipv4 | ipv6] spmsi-ad-route [*Source Address*] [*Group address*] vrf <*vrf_name*>	指定のテナント VRF のファブリック マ IPV4/IPv6 情報を表示します。
show ip mroute detail vrf <*vrf_name*>	デフォルト VRF の IP マルチキャスト ル
show l2route spmsi {all | topology <*vlan*>}	L2RIB（Encap ルート プログラミング） へのマッピング情報を表示します。
show forwarding distribution multicast vxlan mdt-db	MFDM/MFIB データ MDT db を表示しま
show nve resource multicast	データ MDT のリソース使用状況と失敗 ます。

IGMP スヌーピングの設定

VXLAN を介した IGMP スヌーピングの概要

デフォルトでは、VXLAN 上のマルチキャスト トラフィックは、ブロードキャストおよび不明なユニキャスト トラフィックと同様に、VNI/VLAN でフラッディングされます。IGMP スヌーピングを有効にすると、各 VTEP は IGMP レポートをスヌーピングし、マルチキャスト トラフィックのみを対象の受信者に転送できます。

IGMP スヌーピングの設定は、通常の VLAN ドメインでの IGMP スヌーピングの設定と VXLAN で同じです。IGMP スヌーピングの詳細は、『*Cisco Nexus 7000 Series NX-OS Multicast Routing Configuration Guide, Release 7.x*』の「Configuring IGMP Snooping」を参照してください。

VXLAN を介した IGMP スヌーピングに関する注意事項と制限事項

VXLAN を介した IGMP スヌーピングに関する注意事項と制限事項は次のとおりです。

- VXLAN を介した IGMP スヌーピングは FEX メンバー ポートを介した VLAN ではサポートされません。
- VXLAN を介した IGMP スヌーピングは IR とマルチキャスト アンダーレイの両方でサポートされます。
- VXLAN を介した IGMP スヌーピングは、BGP EVPN トポロジでサポートされます。フラッディングおよび学習トポロジではありません。

VXLAN を介した IGMP スヌーピングの設定

手順の概要

1. switch# **configure terminal**
2. switch(config)# **ip igmp snooping vxlan**
3. switch(config)# **ip igmp snooping disable-nve-static-router-port**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	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	switch(config)# **ip igmp snooping vxlan**	VXLAN VLAN の IGMP スヌーピングを有効にします。VXLAN VLANのスヌーピングを有効にするには、このコマンドを明示的に設定する必要があります。
ステップ 3	switch(config)# **ip igmp snooping disable-nve-static-router-port**	このグローバル CLI コマンドを使用して、VXLAN 経由の IGMP スヌーピングを設定し、静的 mrouter ポートとしてNVEを含めないようにします。VXLAN を介した IGMP スヌーピングには、デフォルトで mrouter ポートとして NVE インターフェイスがあります。

第 20 章

VXLAN OAM の設定

この章は、次の内容で構成されています。

VXLAN OAM の概要

イーサネット運用管理およびメンテナンス（OAM）は、イーサネット ネットワークの設置、モニタリング、およびトラブルシューティングのためのプロトコルで、VXLAN ベースのオーバーレイ ネットワークの管理機能が強化されます。

IPネットワークの問題を迅速に特定できる ping、traceroute、または pathtrace ユーティリティと同様に、VXLAN ネットワークの問題を診断するための同等のトラブルシューティング ツールが導入されています。VXLAN OAM ツール（ping、pathtrace、traceroute など）は、VXLAN ネットワーク内のホストおよび VTEP に到達可能性情報を提供します。OAM チャネルは、これらの OAM パケットに存在する VXLAN ペイロードのタイプを識別するために使用されます。

次の 2 種類のペイロードがサポートされています。

- 追跡対象の宛先への従来の ICMP パケット
- 有用な情報を伝送する特別な NVO3 ドラフト Tissa OAM ヘッダー

ICMP チャネルは、新しい OAM パケット形式をサポートしない従来のホストまたはスイッチに到達するのに役立ちます。NVO3 ドラフトの Tissa チャネルは、サポートされているホストまたはスイッチに到達し、重要な診断情報を伝送します。VXLAN NVO3 ドラフトの Tissa OAM

メッセージは、さまざまなプラットフォームでの実装に応じて、予約済みの OAM EtherType を介して、または OAM パケットの既知の予約済み送信元 MAC アドレスを使用して識別できます。これは、VXLAN OAM パケットを認識するためのシグニチャを構成します。VXLAN OAM ツールは、次の表に示すように分類されます。

表 *7: VXLAN OAM* ツール

Category	Tools
障害検査	loopback メッセージ
障害の隔離	パス トレース メッセージ
パフォーマンス	遅延測定、損失測定
AUX	アドレス バインディング検証、IP エンドステーション ロケータ、エラー通知、OAM コマンド メッセージ、ECMP カバレッジの診断ペイロード検出

ループバック（ping）メッセージ

ループバック メッセージ（ping とループバック メッセージは同じで、このガイドでは同じ意味で使用されます）は、障害の検証に使用されます。ループバック メッセージユーティリティは、さまざまなエラーやパス障害を検出するために使用されます。次の例では、Spine 1、Spine 2、Spine 3 というラベルの付いた 3 つのコア（スパイン）スイッチと 5 つのリーフ スイッチが Clos トポロジで接続されているトポロジを考えます。リーフ 5 のリーフ 1 から開始されたサンプル ループバック メッセージのパスは、スパイン 3 を経由するときに表示されます。リーフ 1 によって開始されたループバック メッセージはスパイン 3 に到達すると、外部ヘッダーに基づいて VXLAN カプセル化データ パケットとして転送します。パケットはスパイン 3 のソフトウェアに送信されません。リーフ 3 では、適切なループバック メッセージシグ シグニチャに基づいて、パケットがソフトウェア VXLAN OAM モジュールに送信され、ソフトウェア VXLAN OAM モジュールがループバック応答を生成して、発信元 Leaf 1 に送り返します。

ループバック（ping）メッセージは、VM またはリーフ スイッチ（VTEP）を宛先とすることができます。この ping メッセージは、異なる OAM チャネルを使用できます。ICMP チャネルが使用されている場合、VM の IP アドレスが指定されていれば、ループバック メッセージは VM に到達します。NVO3 ドラフトの Tissa チャネルが使用されている場合、このループバック メッセージは、VM に接続されているリーフ スイッチで終端されます。これは、VM が NVO3 ドラフトの Tissa ヘッダーをサポートしていないためです。この場合、リーフ スイッチはこのメッセージに応答して、VM の到達可能性を示します。ping メッセージは、次の到達可能性オプションをサポートします。

ping

ネットワークの到達可能性を確認します（**Ping** コマンド）。

- Leaf 1（VTEP 1）から Leaf 2（VTEP 2）（ICMP または NVO3 ドラフト Tissa チャネル）

- Leaf 1（VTEP 1）から VM 2（別の VTEP に接続されたホスト）へ（ICMP または NVO3 ドラフト Tissa チャネル）

図 *32 : loopback* メッセージ

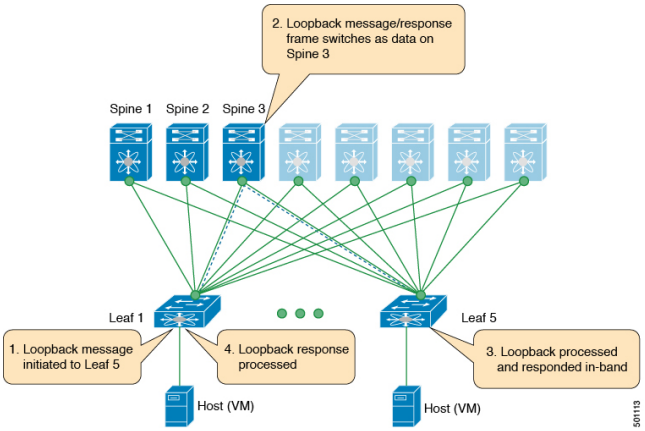
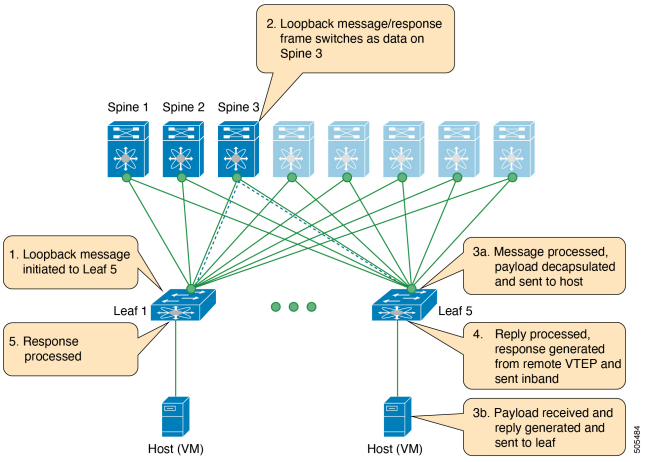


図 *33 :* リモート *VM* への *NVO3* ドラフト *Tissa ping*



Traceroute または Pathtrace メッセージ

traceroute または pathtrace メッセージは、障害分離に使用されます。VXLAN ネットワークでは、宛先に到達するためにフレームが通過するスイッチのリストを見つけることが望ましい場合があります。送信元スイッチから宛先スイッチへのループバックテストが失敗した場合、次の手順はパス内の問題のあるスイッチを見つけることです。パス トレース メッセージの動作は、送信元スイッチが TTL 値1の VXLAN OAM フレームを送信することから始まります。ネクスト ホップ スイッチはこのフレームを受信し、TTL をデクリメントし、TTL が 0 であることを検出すると、TTL 期限切れメッセージを送信元スイッチに送信します。送信元スイッチは、このメッセージを最初のホップスイッチからの成功を示すものとして記録します。次に、送信元スイッチは、次のパス トレース メッセージで TTL 値を 1 増やして、2 番目のホップを見つけます。新しい送信ごとに、メッセージ内のシーケンス番号が増加します。通常のVXLAN転送の場合と同様に、パス上の各中間スイッチは TTL 値を1減らします。

このプロセスは、宛先スイッチから応答を受信するか、パス トレース プロセスのタイムアウトが発生するか、ホップ カウントが設定された最大値に達するまで続きます。VXLAN OAM フレームのペイロードは、フロー エントロピーと呼ばれます。フロー エントロピーは、送信元スイッチと宛先スイッチ間の複数の ECMP パスから特定のパスを選択するように設定できます。TTL 期限切れメッセージは、実際のデータ フレームの中間スイッチによって生成されることもあります。元のパストレース要求と同じペイロードが、応答のペイロードに対して保持されます。

traceroute メッセージと pathtrace メッセージは似ていますが、traceroute は ICMP チャネルを使用しますが、pathtrace は NVO3 ドラフトの Tissa チャネルを使用します。Pathtrace は、NVO3 ドラフトの Tissa チャネルを使用して、追加の診断情報（たとえば、これらのメッセージによって取得されたホップのインターフェイス ロードおよび統計情報）を伝送します。中間デバイスが NVO3 ドラフトの Tissa チャネルをサポートしていない場合、パストレースは単純な traceroute として動作し、ホップ情報のみを提供します。

traceroute

Traceroute コマンドを使用して、VXLANオーバーレイでパケットが通過するパスをトレースします。

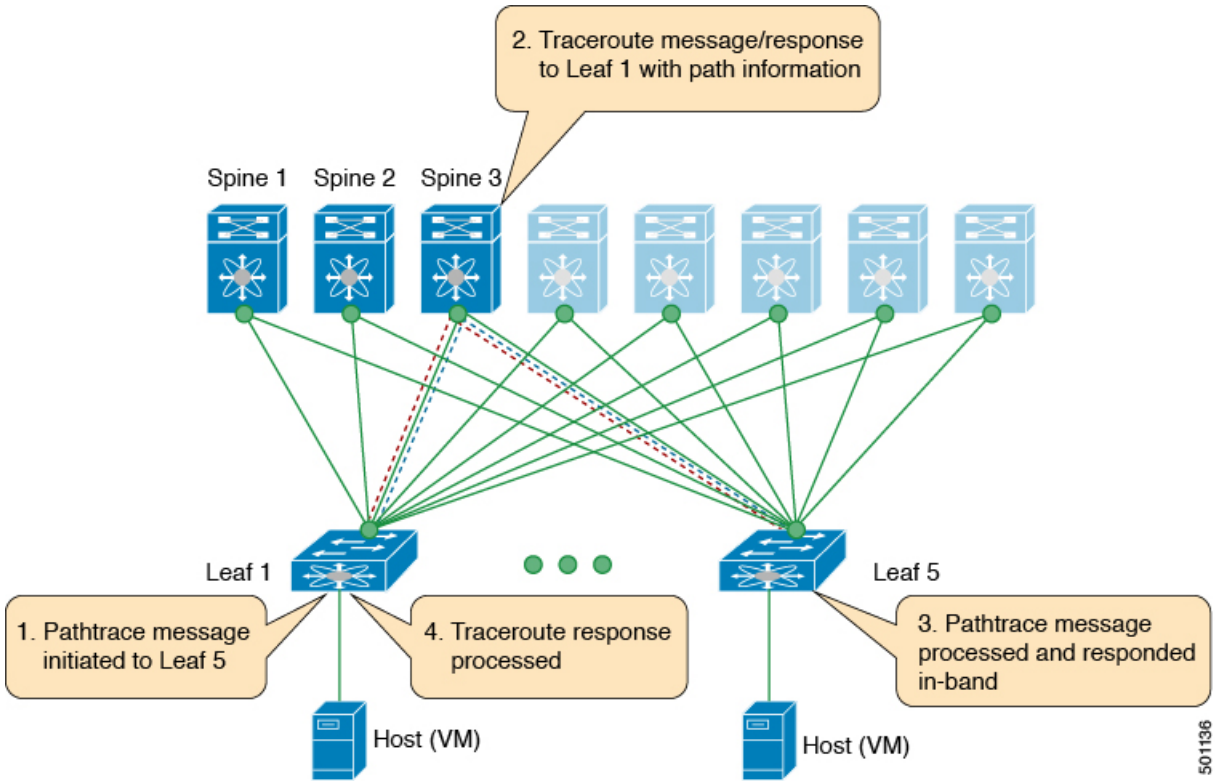
- traceroute は、VXLAN カプセル化でカプセル化された ICMP パケット（チャネル 1）を使用してホストに到達します。

パス トレース

Pathtrace コマンドを使用して、NVO3 ドラフト Tissa チャネルを使用して、VXLAN オーバーレイでパケットが通過するパスをトレースします。

- パストレースは、パスに関する追加情報（入力インターフェイスや出力インターフェイスなど）を提供するために、NVO3 ドラフトの Tissa や TISSA（チャネル2）などの特別な制御パケットを使用します。これらのパケットは VTEP で終端し、ホストに到達しません。したがって、VTEP のみが応答します。
- NX-OS リリース9.3(3) 以降、コマンドの `Received` フィールドは、要求がそのノード宛てかどうかに関係なく、**show ngoam pathtrace statistics summary** コマンドが実行されたノードによって受信されたすべてのパストレース要求を示します。

図 *34 : Traceroute* メッセージ



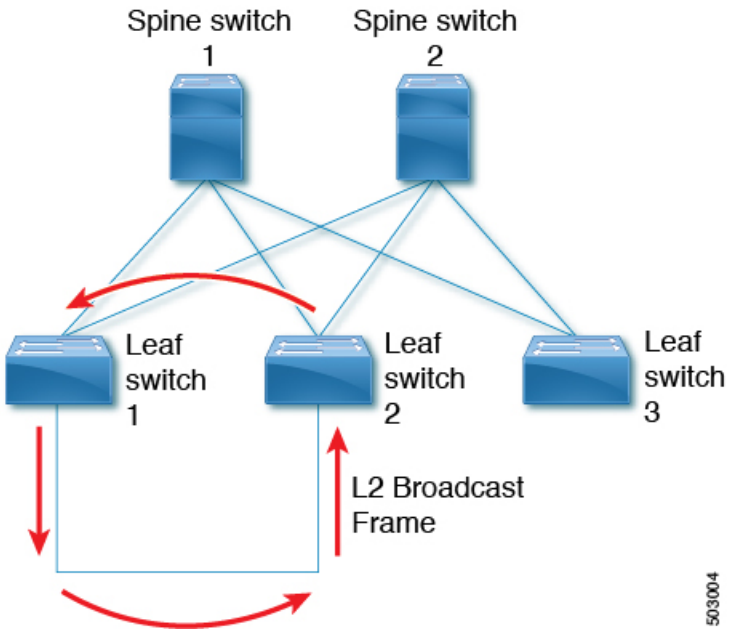
VXLAN EVPN ループの検出と緩和について

ループは通常、ファブリックの南側（アクセス側）の配線が正しくないために、VXLAN EVPN ファブリックで発生します。ブロードキャストパケットがループでネットワークに注入されると、フレームはループ内でブリッジされたままになります。より多くのブロードキャスト フレームがループに入ると、それらが蓄積され、サービスの重大な中断を引き起こす可能性があります。

Cisco NX-OS リリース 9.3(5) では、VXLAN EVPN ループの検出と緩和が導入されています。この機能は、単一の VXLAN EVPN ファブリックまたはマルチサイト環境でレイヤ 2 ループを検出します。ポート/VLAN レベルで動作し、ループが検出された各ポートで VLAN を無効にします。管理者は、（syslog を介して）条件についても通知されます。このように、この機能により、ネットワークが稼働したままになります。

次の図は、2 つのリーフデバイス（Leaf1 および Leaf2）が南側で直接接続されている EVPN ファブリックを示しています。このトポロジでは、Leaf3 は L2 ブロードキャスト フレームを Leaf1 に転送します。次に、ブロードキャスト フレームは Leaf1 と Leaf2 の間で、南側とファブリックを介して繰り返し転送されます。不正なケーブル接続が修正されるまで、転送が続行されます。

図 *35 :* 直接接続された *2* つのリーフ ノード



この機能は、次の 3 つのフェーズで動作します。

1. ループ検出：次の状況でループ検出プローブを送信します。定期的なプローブタスクの一部として、クライアントから要求されたとき、およびポートが起動するとすぐに送信します。

2. ループ緩和：ループが検出されると、ポート上の VLAN をブロックし、次のような syslog メッセージを表示します。

   ```
   2020 Jan 14 09:58:44 Leaf1 %NGOAM-4-SLD_LOOP_DETECTED: Loop detected - Blocking vlan
    1001 :: Eth1/3
   ```

   ループは不正なローカル MAC アドレスの学習につながる可能性があるため、このフェーズではローカルおよびリモート MAC アドレスもフラッシュされます。これにより、誤って学習された MAC アドレスが削除されます。

   前の図では、リモート リーフ（Leaf3）の背後にあるホストからのパケットがアクセス側から Leaf1 と Leaf2 の両方に到達できるため、MAC アドレスが誤って学習される可能性があります。その結果、ホストは Leaf1 および Leaf2 に対してローカルに誤って表示され、リーフは MAC アドレスを学習します。

3. ループ リカバリ：特定のポートまたは VLAN でループが検出され、リカバリ間隔が経過すると、リカバリ プローブが送信され、ループがまだ存在するかどうかが判断されます。ループから NGAM が回復すると、次のような syslog メッセージが表示されます。

   ```
   2020 Jan 14 09:59:38 Leaf1 %NGOAM-4-SLD_LOOP_GONE: Loop cleared - Enabling vlan 1001

   :: Eth1/3
   ```

（注） NGAM のデフォルトのロギングレベルでは、syslog メッセージは生成されません。「logging level ngoam 5」を使用して NGAM のロギングレベルを 5 に変更すると、ループが検出されたときに syslog メッセージが生成されます。

VXLAN NGOAM の注意事項と制約事項

VXLAN NGOAM には、次の注意事項と制限事項があります。

- Cisco NX-OS リリース9.2(3) 以降では、-R ライン カードを備えた Cisco Nexus 9504 および 9508 スイッチのサポートが追加されています。
- Cisco NX-OS リリース 9.3(3) 以降、Cisco Nexus 9300-GX プラットフォーム スイッチに対するサポートが追加されています。
- Cisco NX-OS リリース9.3(5) 以降では、Cisco Nexus 9300-FX3 プラットフォームスイッチのサポートが追加されています。
- Cisco NX-OS Release 10.2(3)F 移行、VXLAN NGOAM は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN NGOAM は Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、VXLAN NGOAM は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN NGOAM は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.2(3)F 以降、中間ノードで NGOAM 機能を使用する **feature nv overlay** コマンドを使用して VXLAN 機能を有効にする必要はありません。
- Cisco NX-OS リリース 10.4(3)F 以降、NGOAM ping、traceroute、および pathtrace は Cisco Nexus 9800 スイッチでサポートされますが、Xconnect およびサウスバウンド ループ検出（SLD）はサポートされません。
- Cisco NX-OS リリース 10.4(3)F 以降、レイヤ 3 イーサネットおよびレイヤ 3 ポートチャネル インターフェイスの NGOAM サウスバウンド ループ検出（SLD）は、Cisco Nexus プラットフォーム シリーズ スイッチでサポートされます。ただし、 サブインターフェイスはサポートされていません。

VXLAN EVPN ループの検出と緩和のガイドラインと制限事項

VXLAN EVPN ループの検出と緩和には、次のガイドラインと制限事項があります。

- VXLAN EVPN ループの検出と緩和は、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。
- 次のプラットフォームは、VXLAN EVPN ループの検出と緩和をサポートします。
  - Cisco Nexus 9332C および 9364C プラットフォーム スイッチ
  - Cisco Nexus 9300-EX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX/FX2/FXP プラットフォーム スイッチ
  - Cisco Nexus 9300-GX プラットフォーム スイッチ
  - -EX/FX ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ
- Cisco NX-OS リリース 10.1(1) 以降では、VXLAN EVPN ループの検出と緩和が Cisco Nexus 9300-FX3 および -GX プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.2(3)F 以降では、VXLAN EVPN ループの検出と緩和が Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降では、VXLAN EVPN ループの検出と緩和が Cisco Nexus 9332D-H2R プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降では、VXLAN EVPN ループの検出と緩和が Cisco Nexus 93400LD-H1 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降では、VXLAN EVPN ループの検出と緩和が Cisco Nexus 9364C-H1 プラットフォーム スイッチでサポートされます。
- VXLAN EVPN ループの検出と緩和は、STP および STP なしの両方の環境でサポートされます。
- VXLAN EVPN マルチサイト展開のサイト間でループを検出できるようにするには、この機能が展開されているサイト内のすべての境界ゲートウェイで **ngoam loop-detection** コマンドを設定する必要があります。
- VXLAN EVPN ループの検出と緩和は、次の機能ではサポートされません。
  - プライベート VLAN
  - VLAN 変換
  - ESI ベースのマルチホーミング
  - VXLAN クロス コネクト

VXLAN OAM の設定

- Q-in-VNI
- EVPN セグメント ルーティング（レイヤ2）

（注） これらの機能が設定されたポートまたはVLANは、VXLAN EVPNループの検出および緩和から除外する必要があります。これらを除外するには、**disable**{**vlan** *vlan-range*} [**port** *port-range*] コマンドを使用できます。

VXLAN OAM の設定

始める前に

前提条件として、VXLAN の設定が完了していることを確認します。

（注） Cisco NX-OS リリース 10.2(3) 以降、中間ノードで NGOAM 機能を設定するために VXLAN 機能を有効にする必要はありません。

手順の概要

1. switch# **configure terminal**
2. switch(config)# **feature ngoam**
3. switch(config)# **hardware access-list tcam region arp-ether 256 double-wide**
4. switch(config)# **ngoam install acl**
5. （任意） **bcm-shell module 1 "fp show group 62"**

手順の詳細

手順

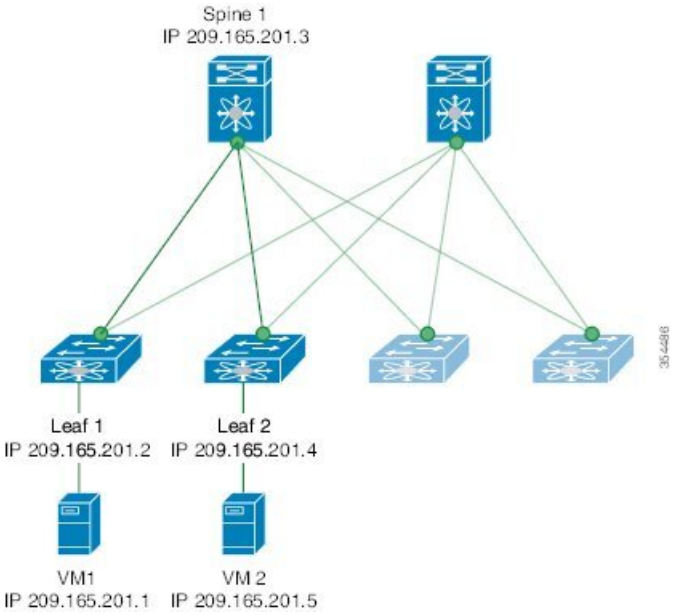
	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# **feature ngoam**	NGOAM 機能を開始します。
ステップ 3	switch(config)# **hardware access-list tcam region arp-ether 256 double-wide**	Network Forwarding Engine（NFE）を備えた Cisco Nexus 9300 プラットフォーム スイッチの場合、このコマンドを使用して ARP-ETHER の TCAM リージョンを設定します。この手順は、ACL ルールをハード

	コマンドまたはアクション	目的
		ウェアでプログラミングするために不可欠であり、ACL ルールをインストールする前の前提条件です。 （注） TCAM リージョンを設定するには、ノードをリブートする必要があります。
ステップ 4	switch(config)# **ngoam install acl**	NFAM アクセス コントロール リスト（ACL）をインストールします。 （注） このコマンドは、Cisco NX-OS リリース 9.3(5) 以降では廃止され、以前のリリースでのみ必要です。
ステップ 5	（任意） **bcm-shell module 1 "fp show group 62"**	ネットワーク転送エンジン（NFE）を搭載した Cisco Nexus 9300 シリーズ スイッチの場合は、次の確認手順を実行します。コマンドを入力した後、EtherType で data=0x8902 のエントリ/eid のルックアップを実行します。

例

次の設定トポロジの例を参照してください。

図 *36 : VXLAN* ネットワーク



VXLAN OAM は、スイッチ レベルでホストの可視性を提供し、 **ping nve** コマンドを使用してリーフがホストに ping を実行できるようにします。

次に、チャネル 1（一意のループバック）およびチャネル 2（NVO3 ドラフト Tissa）を使用して、スパイン 1 を介してリーフ 1 から VM2 に ping を実行する例を示します。

```
switch# ping nve ip 209.165.201.5 vrf vni-31000 source 1.1.1.1 verbose

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'D' - Destination Unreachable, 'X' - unknown return code,
'm' - malformed request(parameter problem),
'c' - Corrupted Data/Test, '#' - Duplicate response


Sender handle: 34
! sport 40673 size 39,Reply from 209.165.201.5,time = 3 ms
! sport 40673 size 39,Reply from 209.165.201.5,time = 1 ms
! sport 40673 size 39,Reply from 209.165.201.5,time = 1 ms
! sport 40673 size 39,Reply from 209.165.201.5,time = 1 ms
! sport 40673 size 39,Reply from 209.165.201.5,time = 1 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/18 ms
Total time elapsed 49 ms
                                                <<<<< add space here
switch# ping nve ip unknown vrf vni-31000 payload ip 209.165.201.5 209.165.201.4
payload-end verify-host
<snip>
Sender handle: 34
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/18 ms
Total time elapsed 49 ms
```

（注） 上記の例で使用されている送信元 IP アドレス 1.1.1.1 は、宛先 IP アドレスと同じ VRF のリーフ 1 に設定されているループバック インターフェイスです。たとえば、この例の VRF は vni-31000 です。

次に、スパイン 1 を介してリーフ 1 から VM 2 に traceroute を実行する例を示します。

```
switch# traceroute nve ip 209.165.201.5 vrf vni-31000 source 1.1.1.1 verbose


Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'D' - Destination Unreachable, 'X' - unknown return code,
'm' - malformed request(parameter problem),
'c' - Corrupted Data/Test, '#' - Duplicate response


Traceroute request to peer ip 209.165.201.4 source ip 209.165.201.2
Sender handle: 36
  1 !Reply from 209.165.201.3,time = 1 ms
  2 !Reply from 209.165.201.4,time = 2 ms
  3 !Reply from 209.165.201.5,time = 1 ms
```

次に、リーフ 2 からリーフ 1 にパス トレースする例を示します。

```
switch# pathtrace nve ip 209.165.201.4 vni 31000 verbose

Path trace Request to peer ip 209.165.201.4 source ip 209.165.201.2

Sender handle: 42
```

VXLAN OAM の設定

```
TTL   Code  Reply                 IngressI/f    EgressI/f    State
=============================================================================
1     !Reply from 209.165.201.3,  Eth5/5/1      Eth5/5/2     UP/UP
2     !Reply from 209.165.201.4,  Eth1/3        Unknown      UP/DOWN
```

次の例は、NVO3 ドラフト Tissa チャネルを使用して、リーフ 2 からリーフ 1 に MAC ping を実行する方法を示しています。

```
switch# ping nve mac 0050.569a.7418 2901 ethernet 1/51 profile 4 verbose

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'D' - Destination Unreachable, 'X' - unknown return code,
'm' - malformed request(parameter problem),
'c' - Corrupted Data/Test, '#' - Duplicate response

Sender handle: 408
!!!!Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/5 ms
Total time elapsed 104 ms

switch# show run ngoam
feature ngoam
ngoam profile 4
oam-channel 2
ngoam install acl
```

次に、リーフ 2 からリーフ 1 へのペイロードに基づいてパス トレースする例を示します。

```
switch# pathtrace nve ip unknown vrf vni-31000 payload mac-addr 0050.569a.d927
0050.569a.a4fa
ip 209.165.201.5 209.165.201.1 port 15334 12769 proto 17 payload-end

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'D' - Destination Unreachable, 'X' - unknown return code,
'm' - malformed request(parameter problem),
'c' - Corrupted Data/Test, '#' - Duplicate response

Path trace Request to peer ip 209.165.201.4 source ip 209.165.201.2
Sender handle: 46
TTL Code Reply IngressI/f EgressI/f State
=============================================================================
1 !Reply from 209.165.201.3, Eth5/5/1 Eth5/5/2 UP/UP
2 !Reply from 209.165.201.4, Eth1/3 Unknown UP/DOWN
```

（注） 最終宛先までの合計ホップ カウントが 5 を超える場合、パス トレースのデフォルト TTL 値は 5 です。**max-ttl** オプションを使用して、VXLAN OAM パス トレースを完全に終了します。

次に例を示します。 **pathtrace nve ip unknown vrf vrf-vni13001 payload ip 200.1.1.71 200.1.1.23 payload-end verbose max-ttl 10**

NGOAM プロファイルの設定

NGOAM プロファイルを設定する手順は、次のとおりです。

手順の概要

1. switch(config)# **[no] feature ngoam**
2. switch(config)# **[no] ngoam profile <profile-id>**
3. switch(config-ng-oam-profile)# **?**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **[no] feature ngoam**	NGOAM 機能をイネーブルまたはディセーブルにします。
ステップ 2	switch(config)# **[no] ngoam profile <profile-id>**	OAM プロファイルを設定します。profile-id の範囲は、1ー1023 です。このコマンドにはデフォルト値はありません。**config-ngoam-profile submode** を入力してNGAM固有のコマンドを設定します。 （注） すべてのプロファイルにはデフォルト値があり、**show run all**CLIコマンドによってデフォルト値が表示されます。デフォルト値は、CLIコマンドでは表示されません。**show run**

ステップ 3	switch(config-ng-oam-profile)# **?**
	例：
	`switch(config-ng-oam-profile)# ?`
	`  description  Configure description of the profile`
	`  dot1q  Encapsulation dot1q/bd`
	`  flow  Configure ngoam flow`
	`  hop  Configure ngoam hop count`
	`  interface  Configure ngoam egress interface`
	`  no  Negate a command or set its defaults`
	`  oam-channel  Oam-channel used`
	`  payload  Configure ngoam payload`
	`  sport  Configure ngoam Udp source port range`
		NGOAM プロファイルを設定するためのオプションを表示します。

例

次の例を参照して、NGOAM プロファイルと NGOAM フローを設定します。

```
switch(config)#
ngoam profile 1
oam-channel 1
flow forward
payload pad 0x2
sport 12345, 54321

switch(config-ngoam-profile)#flow {forward }
Enters config-ngoam-profile-flow submode to configure forward flow entropy specific
information
```

VXLAN EVPN ループの検出と緩和の設定

VXLAN ループの検出と緩和を設定するには、次の手順に従います。

始める前に

NGOAM 機能を有効にします。

TCAM ing-sup リージョン用のスペースを作成するには、次のコマンドを使用します。

```
hardware access-list tcam region ing-racl 0
hardware access-list tcam region ing-sup 768
```

（注） TCAM リージョンを設定するには、ノードをリブートする必要があります。

手順の概要

1. switch# **configure terminal**
2. switch(config)# [**no**] **ngoam loop-detection**
3. （任意） switch(config-ng-oam-loop-detection)# [**no**] **disable** {**vlan** *vlan-range*} [**port** *port-range*]
4. （任意） switch(config-ng-oam-loop-detection)# [**no**] **periodic-probe-interval** *value*
5. （任意） switch(config-ng-oam-loop-detection)# [**no**] **port-recovery-interval** *value*
6. （任意） switch# **show ngoam loop-detection summary**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	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	switch(config)# [**no**] **ngoam loop-detection**	すべての VLAN またはポートの VXLAN EVPN ループ検出と緩和を有効にします。この機能はデフォルトで無効に設定されています。
ステップ 3	（任意） switch(config-ng-oam-loop-detection)# [**no**] **disable** {**vlan** *vlan-range*} [**port** *port-range*]	特定の VLAN またはポートの VXLAN EVPN ループ検出および緩和を無効にし、ループ検出されたポートを起動します。このコマンドの **no** 形式は、これらの VLAN またはポートのアクティブ モニタリングを再開します。
ステップ 4	（任意） switch(config-ng-oam-loop-detection)# [**no**] **periodic-probe-interval** *value*	定期的なループ検出プローブの送信頻度を指定します。範囲は 60ー3600 秒（60 分）です。デフォルトは 300 秒（5 分）です。
ステップ 5	（任意） switch(config-ng-oam-loop-detection)# [**no**] **port-recovery-interval** *value*	ポートまたはVLANがシャットダウンされると、回復プローブが送信される頻度を指定します。範囲は 300ー3600 秒（60分）です。デフォルト値は 600 秒（10 分）です。
ステップ 6	（任意） switch# **show ngoam loop-detection summary**	ループ検出の設定と現在のループの概要を表示します。

次のタスク

スパインの QoS ポリシーを設定します。（設定例については、VXLAN EVPN ループの検出と緩和の設定例 （446 ページ） を参照してください）。

レイヤ 3 インターフェイスの NGOAM ループ検出の設定

NX-OS リリース 10.4(3)F 以降、Cisco Nexus スイッチは、レイヤ 3（L3）イーサネットおよび L3 ポート チャネル インターフェイスでの NGOAM サウスバウンド ループ検出（SLD）の有効化をサポートしています。

L3 インターフェイスで有効にすると、この機能は定期的なプローブを送信して、ダウンストリームテナントのレイヤ2 ドメインのループを検出します。ユーザーが条件を修正するためのアクションを実行するまで、ループをモニターし続けます。

イーサネットおよび L3 ポート チャネル インターフェイスで NGOAM ループ検出を有効にするには、次の手順を実行します。

始める前に

NGOAM 機能を有効にします。

TCAM ing-sup リージョン用のスペースを作成するには、次のコマンドを使用します。

```
hardware access-list tcam region ing-racl 0
hardware access-list tcam region ing-sup 768
```

（注） TCAM リージョンを設定するには、ノードをリブートする必要があります。

手順の概要

1. **configure terminal**
2. [**no**] **ngoam loop-detection**
3. [**no**] **l3 ethernet port** *port-range*
4. [**no**] **l3 port-channel port** *port-range*
5. （任意） **show ngoam loop-detection status l3**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# config terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	[**no**] **ngoam loop-detection** 例： `switch(config)# ngoam loop-detection`	すべての VLAN またはポートの VXLAN EVPN ループ検出と緩和を有効にします。この機能はデフォルトで無効に設定されています。
ステップ 3	[**no**] **l3 ethernet port** *port-range* 例： `switch(config-ng-oam-loop-detection)# l3 ethernet port Eth1/49`	イーサネット インターフェイスで L3 ループ検出を有効にします。 イーサネット インターフェイスで L3 ループ検出を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 4	[**no**] **l3 port-channel port** *port-range* 例：	ポート チャネル インターフェイスで L3 ループ検出を有効にします。

	コマンドまたはアクション	目的
	`switch(config-ng-oam-loop-detection)# l3 port-channel port port-channel1`	ポートチャネル インターフェイスで L3 ループ検出を無効にするには、このコマンドの **no** 形式を使用します。

ステップ 5	（任意） **show ngoam loop-detection status l3**
	例：
	`switch# show ngoam loop-detection status l3`
	`Port Status NumLoops DetectionTime ClearedTime`
	`Eth1/49 BLOCKED 1 Tue Jun 6 20:01:27 2023 Never`
		L3インターフェイスで検出されたループを表示します。

ループの検出とオンデマンドでのポートの呼び出し

ループを検出するか、ブロックされたポートをオンデマンドで起動するには、この項の手順に従います。

始める前に

VXLAN EVPN ループの検出と緩和を有効にします。

手順の概要

1. （任意） switch# **ngoam loop-detection probe** {**vlan** *vlan-range*} [**port** *port-range*]
2. （任意） switch# **ngoam loop-detection bringup** {**vlan** *vlan-range*} [**port** *port-range*]
3. （任意） switch# **show ngoam loop-detection status** [**history**] [**vlan** *vlan-range*] [**port** *port-range*]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	（任意） switch# **ngoam loop-detection probe** {**vlan** *vlan-range*} [**port** *port-range*]	指定されたVLANまたはポートでループ検出プローブを送信し、プローブが正常に送信されたかどうかを通知します。
ステップ 2	（任意） switch# **ngoam loop-detection bringup** {**vlan** *vlan-range*} [**port** *port-range*]	以前にブロックされたVLANまたはポートを起動します。また、このコマンドを実行すると、NGOAMにスタックしているエントリがクリアされます。

	コマンドまたはアクション	目的
		（注） ループが解消されてからポートが起動するまでに、最大で2つのポート回復インターバルが必要です。**ngoam loop-detection bringup vlan** {**vlan** *vlan-range*} [**port** *port-range*] コマンドを使用して手動でタイマーを上書きすることで、リカバリを高速化できます。
ステップ 3	（任意） switch# **show ngoam loop-detection status** [**history**] [**vlan** *vlan-range*] [**port** *port-range*]	VLAN またはポートのループ検出ステータスを表示します。ステータスは、次のいずれかになります。 • BLOCKED：ループが検出されたため、VLAN またはポートがシャットダウンされました。 • FORWARDING：ループが検出されず、VLAN またはポートが動作しています。 • RECOVERING：以前に検出されたループがまだ存在するかどうかを判断するために、回復プローブが送信されています。 **history** オプションは、ブロックされたポート、転送中のポート、および回復中のポートを表示します。**history** オプションを指定しない場合、コマンドはブロックされたポートと回復中のポートのみを表示します。

VXLAN EVPN ループの検出と緩和の設定例

次に、VXLAN EVPN ループの検出と緩和を設定する例を示します。

```
switch(config)# ngoam loop-detection
switch(config-ng-oam-loop-detection)# periodic-probe-interval 200
switch(config-ng-oam-loop-detection)# port-recovery-interval 300
```

次に、特定の VLAN または VLAN ポートで VXLAN EVPN ループの検出と緩和を無効にする例を示します。

```
switch(config-ng-oam-loop-detection)# disable vlan 1200 port ethernet 1/1
switch(config-ng-oam-loop-detection)# disable vlan 1300
```

次に、スパインに QoS ポリシーを設定し、ループ検出が有効なリーフが接続されているすべてのスパイン インターフェイスに適用する例を示します。

```
class-map type qos match-any Spine-DSCP56
match dscp 56
policy-map type qos Spine-DSCP56
class Spine-DSCP56
set qos-group 7
```

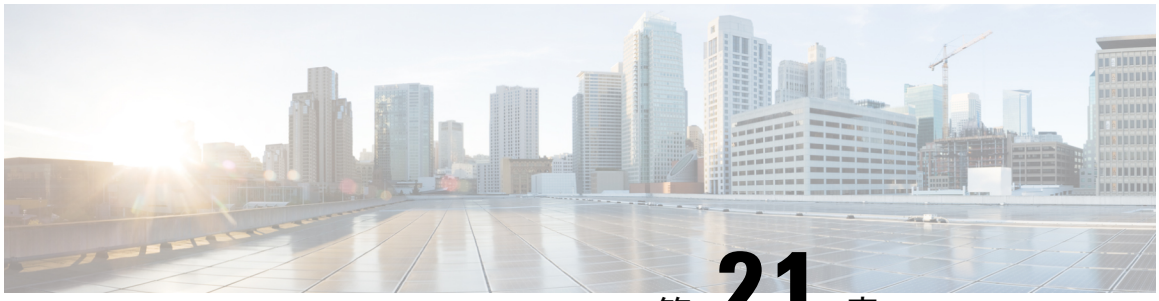
```
interface Ethernet1/31
mtu 9216
no link dfe adaptive-tuning
service-policy type qos input Spine-DSCP5663
no ip redirects
ip address 27.4.1.2/24
ip router ospf 200 area 0.0.0.0
ip pim sparse-mode
no shutdown
```

次の出力例は、ループ検出の設定と現在のループの概要を示しています。

```
switch# show ngoam loop-detection summary
Loop detection:enabled
Periodic probe interval: 200
Port recovery interval: 300
Number of vlans: 1
Number of ports: 1
Number of loops: 1
Number of ports blocked: 1
Number of vlans disabled: 0
Number of ports disabled: 0
Total number of probes sent: 214
Total number of probes received: 102
Next probe window start: Thu May 14 15:14:23 2020 (0 seconds)
Next recovery window start: Thu May 14 15:54:23 2020 (126 seconds)
```

次の出力例は、**history** オプションを使用した場合と使用しない場合の、指定されたVLANまたはポートのループ検出ステータスを示しています。

```
switch# show ngoam loop-detection status
VlanId Port   Status     NumLoops  Detection Time                 ClearedTime
====== ====== ========== ========= ============================== ===============
100    Eth1/3 BLOCKED     1        Tue Apr 14 20:07:50.313 2020  Never

switch# show ngoam loop-detection status history
VlanId Port   Status     NumLoops  Detection Time                 ClearedTime
====== ====== ========== ========= ============================== ===============
100    Eth1/3 BLOCKED     1        Tue Apr 14 20:07:50.313 2020  Never
200    Eth1/2 FORWARDING  1        Tue Apr 14 21:19:52.215 2020  May 11 21:30:54.830
2020
```



第 21 章

VXLAN QoS の設定

この章は、次の内容で構成されています。

VXLAN QoS に関する情報

VXLAN QoS を使用すると、VXLAN でトンネリングされるトラフィックに Quality of Service（QoS）機能を提供できます。

VXLAN オーバーレイのトラフィックは、さまざまな QoS プロパティに割り当てることができます。

- 異なるプロパティを割り当てるためのトラフィックの分類。
- 異なるプライオリティのトラフィック マーキングを含む。
- 保護されたトラフィックのプライオリティを有効にするためのトラフィックのキューイング。
- 不正なトラフィックのポリシング。
- インターフェイスごとの速度を制限するトラフィックのシェーピング。
- トラフィック ドロップの影響を受けやすいトラフィックのプロパティ。

（注） QoS では、ネットワーク トラフィックの分類、トラフィック フローのポリシングとプライオリティ設定、および輻輳回避が可能です。QoS の設定の詳細については、『Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide、Release 9.2(x)』を参照してください。

ここでは、次の内容について説明します。

VXLAN QoS の用語

ここでは、VXLAN QoS の用語をいくつか定義します。

表 *8 : VXLAN QoS* の用語

用語	定義
Frames	レイヤ 2 でトラフィックを伝送します。レイヤ 2 フレームは、レイヤ 3 パケットを伝送します。
パケット	レイヤ 3 でトラフィックを伝送します。
VxLAN パケット	VXLAN IP/UDP ヘッダーにカプセル化された元のフレームを伝送します。
元のフレーム	VXLANヘッダーにカプセル化する前にレイヤ 3 パケットを伝送するレイヤ 2 またはレイヤ 2 フレーム。
カプセル化解除されたフレーム	VXLANヘッダーのカプセル化解除後にレイヤ 3 パケットを伝送するレイヤ 2 またはレイヤ 2 フレーム。
入力 VTEP	トラフィックが VXLAN ヘッダーにカプセル化され、VXLAN トンネルに入るポイント。
出力 VTEP	トラフィックが VXLAN ヘッダーからカプセル化解除され、VXLAN トンネルを出るポイント。
サービス クラス（CoS）	スイッチドネットワークを通過するときにイーサネット フレームのプライオリティを示す 802.1Qヘッダーの 3 ビットのことです。802.1Q ヘッダーの CoS ビットは通常 802.1p ビットと呼ばれます。802.1X は、VXLAN トンネル内に CoS 値が存在しない VXLAN ヘッダー内のフレーム カプセル化の前に廃棄されます。パケットが VXLAN トンネルに入るときに QoS を維持するために、タイプオブサービス（ToS）と CoS 値が相互にマッピングされます。
IP precedence	IP ヘッダーの ToS バイトの最上位 3 ビットです。

用語	定義
Diffserv コード ポイント（DSCP）	IP ヘッダーの ToS バイトの最初の 6 ビット。DSCP は、IP パケットだけに存在します。
明示的輻輳通知（ECN）	IP ヘッダーの ToS バイトの最後の 2 ビット。ECN は、IP パケットだけに存在します。
QoSタグ	レイヤ 3 パケットおよびレイヤ 2 フレームで伝達されるプライオリティ値です。レイヤ 2 CoS ラベルは、0（ロー プライオリティ）～ 7（ハイ プライオリティ）の範囲です。レイヤ 3 IP precedence ラベルは、0（ロー プライオリティ）～ 7（ハイ プライオリティ）の範囲です。IP precedence 値は、1 バイトの ToS バイトの最上位 3 ビットで定義されます。レイヤ 3 DSCP ラベルは、0 ～ 63 の値を持つことができます。DSCP 値は 1 バイトの IP ToS フィールドのうち最上位 6 ビットで定義されます。
分類	QoS のトラフィックの選択に使用されるプロセス
マーキング	設定プロセス：フレームのレイヤ 2 COS 値、パケットのレイヤ 3 DSCP 値、およびパケットのレイヤ 3 ECN 値。マーキングはまた、CoS、DSCP、ECJ フィールドで異なった値を選択してパケットにマーキングし、輻輳時にパケットが必要なプライオリティを持つようにするプロセスでもあります。
ポリシング	トラフィック フローが使用する帯域幅を制限する処理です。ポリシングによって、トラフィックのマーキングまたは廃棄が可能になります。
MQC	Cisco モジュラ QoS コマンドライン インターフェイス（MQC）フレームワークです。QoS 展開において、モジュラ式で拡張性に優れています。

VXLAN QoS機能

次のトピックでは、VXLAN ネットワークでサポートされる VXLAN QoS 機能について説明します。

信頼境界

信頼境界は、ネットワークの境界を形成します。ネットワークはスイッチのマーキングを信頼します（オーバーライドしません）。既存の ToS 値は、VXLAN ファブリックで受信されると信頼されます。

分類

分類は、トラフィックをクラスに区分けするのに使用します。トラフィックは、ポート特性またはパケット ヘッダー フィールドに基づいて分類します。パケット ヘッダー フィールドには、IP precedence、DiffServ コード ポイント（DSCP）、レイヤ 3 からレイヤ 4 までのパラメータ、およびパケット長が含まれます。

トラフィックの分類に使用する値を、一致基準と呼びます。トラフィック クラスを定義する場合、一致基準を複数指定することも、特定の基準について照合しないように選択することも、一部または全部の基準を照合することによってトラフィック クラスを決定することもできます。

どのクラスにも一致しないトラフィックは、class-default と呼ばれるデフォルトのトラフィック クラスに割り当てられます。

マーキング

マーキングとは、パケットに関連する QoS 情報を設定することです。パケット マーキングを利用すれば、ネットワークを複数の優先プライオリティ レベルまたはサービス クラスに分割することができます。COS、IP precedence、および DSCP の標準 QoS フィールドの値を設定できます。その後のアクションで使用できる内部ラベル（QoS グループなど）のために、QoS フィールドも設定できます。QoS グループ マーキングは、トラフィックのキューイング、およびスケジューリングに対応したトラフィック タイプを識別するのに使用します。

ポリシング

ポリシングを行うと、設定レートを超えたトラフィックは廃棄されるか、またはより高いドロップ優先順位にマークダウンされます。

シングルレート ポリサーは、トラフィックの指定の認定情報レート（CIR）を監視します。デュアルレート ポリサーは、CIR と最大情報レート（PIR）の両方を監視します。

キューイングおよびスケジューリング

キューイングおよびスケジューリング プロセスでは、トラフィック クラスに割り当てられるキューの使用量と帯域幅を制御できるようにします。これにより、スループットと遅延の間の望ましいトレードオフを実現できます。

スタティックまたはダイナミックな制限を適用することで、トラフィックの特定のクラスについてキューのサイズを制限できます。

重み付けランダム早期検出（WRED）をトラフィックのクラスに適用できます。これにより、サービス クラス（QoS）グループに基づいてパケットをドロップできます。WRED のアルゴリズムにより、キューを予防的に管理してトラフィックの輻輳を防ぐことができます。

ECNは、パケットをドロップする代わりに輻輳状態をマーキングするために、特定のトラフィック クラスで WRED とともに使用できます。VXLAN トンネルでの ECN マーキングは外部ヘッダーで実行され、出力 VTEP でカプセル化解除されたフレームにコピーされます。

トラフィック シェーピング

トラフィックのクラスに対して最大データ レートを強制してトラフィックをシェーピングすることができます。これにより、超過パケットがキューに保持され、出力レートが平滑化（制限）されます。さらに、トラフィック クラスに最小帯域幅保証を提供するために、最小帯域幅のシェーピングを設定できます。

トラフィック シェーピングは、各ポートの出力キューに最大トラフィック レートを強制することで、パケット フローを制御および均一化します。しきい値を超えたパケットはキューに配置され、後で送信されます。トラフィック シェーピングはトラフィック ポリシングと似ていますが、パケットはドロップされません。パケットがバッファに入れられるため、トラフィック シェーピングでは、（キュー長に基づく）パケット損失が最小限に抑えられ、TCP トラフィックに対してより優れたトラフィック動作が実現します。

トラフィック シェーピングを使用すると、次を制御できます。

- 使用可能な帯域幅へのアクセスを制御する。
- トラフィックが、このトラフィック用に設定したポリシーと一致するようにする。
- 出力トラフィックがそのリモートのターゲット インターフェイスのアクセス速度を超過したときに発生する可能性のある輻輳を回避するためのトラフィックのフロー制御。

たとえば、ポリシーによって、そのインターフェイスのレートが（平均で）特定のレートを上回るべきではないとされている場合に、帯域幅へのアクセスを制御できます。アクセスレートが速度を超えている場合でも例外ではありません。

ネットワーク QoS

ネットワーク QoS ポリシーは各 CoS 値の特性を定義します。これらの特性は、スイッチを介してネットワーク全体に適用できます。ネットワーク QoS ポリシーを使用して、次のことを設定できます。

- 一時停止動作：CoS が輻輳時のパケット損失を防ぐプライオリティ フロー制御（PFC）メカニズムを使用して提供されるロスレス動作を必要とするかどうかを決定できます。drop（ドロップできるこの CoS 値を持つフレーム）および no drop（ドロップできないこの CoS 値を持つフレーム）を設定できます。また、drop および no drop 設定では、ポート単位で PFC をイネーブル化する必要もあります。PFC の詳細については、「プライオリティ フロー制御の設定」を参照してください。

一時停止動作は、特定のキューグループの VXLAN トンネルで実現できます。

VXLAN プライオリティ トンネリング

VXLAN トンネルでは、外部ヘッダーの DSCP 値を使用して、トンネルのエンドツーエンドで QoS 透過性が提供されます。外部ヘッダーの DSCP 値は、レイヤ 3 パケットの DSCP 値または

レイヤ 2 フレームの CoS 値から取得されます。VXLAN トンネル出力ポイントでは、カプセル化解除されたトラフィックのプライオリティがモードに基づいて選択されます。詳細については、カプセル化解除されたパケットの優先順位の選択 （458 ページ） を参照してください。

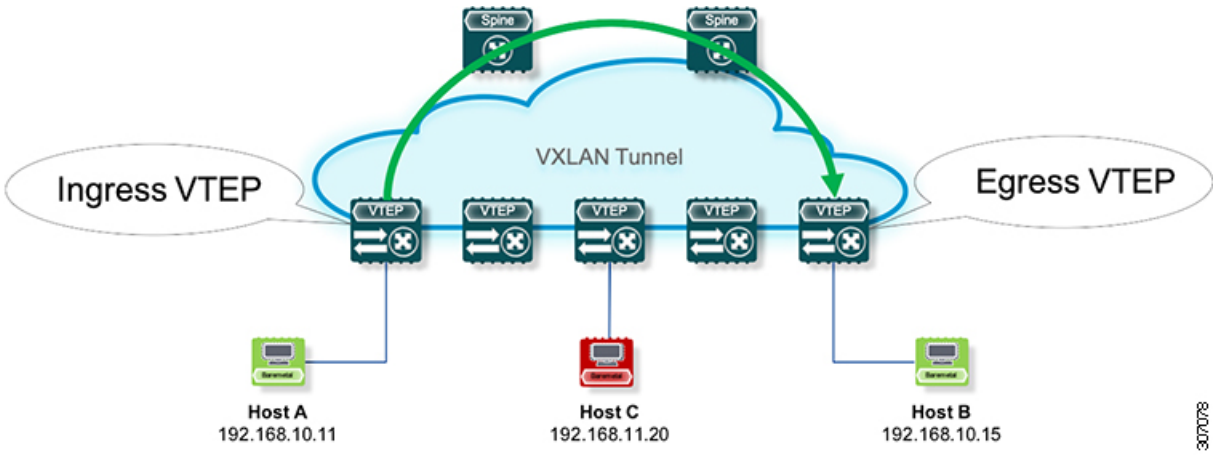
MQC CLI

VXLAN QoS で使用可能な QoS 機能はすべて、モジュラ QoS コマンドライン インターフェイス（CLI）から管理します。モジュラ QoS CLI（MQC）では、トラフィック クラス（クラス マップ）を定義し、トラフィック ポリシー（ポリシー マップ）を作成して設定し、インターフェイスへのポリシー マップ（サービス ポリシー）で定義されたアクションを実行することができます。

VXLAN QoS トポロジとロール

ここでは、VXLAN QoS を実装するときのネットワーク デバイスの役割について説明します。

図 *37 : VXLAN* ネットワーク



ネットワークは双方向ですが、前の図では、トラフィックは左から右に移動しています。

VXLAN ネットワークでは、元のトラフィックが VXLAN ヘッダーにカプセル化される入力 VTEP が対象となります。スパインは、入力 VTEP と出力 VTEP を接続する転送ホップです。出力 VTEP は、VXLAN カプセル化トラフィックがカプセル化解除され、VTEP を従来のイーサネット トラフィックとして出力するポイントです。

（注） 入力および出力 VTEP は、VXLAN トンネルと IP ネットワーク間の境界です。

ここでは、次の内容について説明します。

VXLAN トンネルでの入力 VTEP とカプセル化

入力 VTEP で、VTEP は次のようにパケットを処理します。

手順

ステップ 1 レイヤ 2 またはレイヤ 3 トラフィックは VXLAN ネットワークのエッジに入ります。

ステップ 2 スイッチは入力インターフェイスからトラフィックを受信し、802.1p ビットまたは DSCP 値を使用して、分類、マーキング、およびポリシングを実行します。また、VXLAN ヘッダーの外部 DSCP 値も取得します。着信 IP パケットの分類については、入力サービス ポリシーもアクセス コントロール リスト（ACL）を使用することができます。

ステップ 3 各着信パケットについて、スイッチは IP アドレスで検索を実行し、ネクスト ホップを決定します。

ステップ 4 パケットはVXLANヘッダーにカプセル化されます。カプセル化されたパケットのVXLANヘッダーには、QoS ルールに基づく DSCP 値が割り当てられます。

ステップ 5 スイッチは、カプセル化されたパケットを適切な処理用出力インターフェイスに転送します。

ステップ 6 DSCP 値でマークされたカプセル化されたパケットは、VXLAN トンネル出力インターフェイスに送信されます。

VXLAN トンネルを介したトランスポート

VXLAN トンネルを通過するトランスポートでは、スイッチは VXLAN パケットを次のように処理します。

手順

ステップ 1 VXLANカプセル化パケットは、トランスポートスイッチの入力インターフェイスで受信されます。スイッチは、外部ヘッダーを使用して分類、マーキング、およびポリシングを実行します。

ステップ 2 スイッチは、外部ヘッダーの IP アドレスのルックアップを実行して、ネクスト ホップを決定します。

ステップ 3 スイッチは、カプセル化されたパケットを適切な処理用出力インターフェイスに転送します。

ステップ 4 VXLANは、カプセル化されたパケットを出力インターフェイス経由で送信します。

出力 VTEP と VXLAN トンネルのカプセル化解除

VXLAN トンネルの出力 VTEP 境界で、VTEP は次のようにパケットを処理します。

手順

ステップ 1 VXLAN でカプセル化されたパケットは、出力 VTEP の NVE インターフェイスで受信され、スイッチは内部ヘッダーの DSCP 値を使用して分類、マーキング、およびポリシングを実行します。

ステップ 2 スイッチはパケットから VXLAN ヘッダーを削除し、カプセル化解除されたパケットのヘッダーに基づいてルックアップを実行します。

ステップ 3 スイッチは、カプセル化されたパケットを適切な処理用出力インターフェイスに転送します。

ステップ 4 パケットが送信される前に、カプセル化解除のプライオリティまたはレイヤ 2 フレームのマーキングに基づいて、DSCP 値がレイヤ 3 パケットに割り当てられます。

ステップ 5 カプセル化解除されたパケットは、発信インターフェイスを介して IP ネットワークに送信されます。

入力 VTEP、スパイン、および出力 VTEP での分類

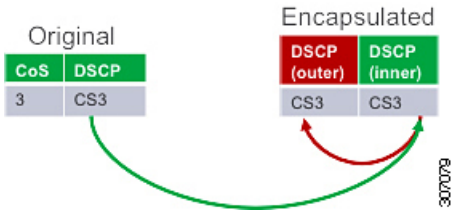
このセクションは、次のトピックで構成されています。

IP から VXLAN へ

VXLAN トンネルの入力ポイントである入力 VTEP では、トラフィックは VXLAN ヘッダーにカプセル化されます。入力 VTEP 上のトラフィックは、元のヘッダーの優先順位に基づいて分類されます。分類は、CoS、DSCP、および IP precedence 値を照合するか、元のフレーム データに基づいてトラフィックを ACL と照合することで実行できます。

トラフィックが VXLAN でカプセル化されると、レイヤ 3 パケットの DSCP 値が VXLAN カプセル化パケットの元のヘッダーから外部ヘッダーにコピーされます。この動作は、次の図に示します。

図 *38 :* レイヤ *3* パケットから *VXLAN* 外部ヘッダーへの優先順位のコピー



IP ヘッダーのないレイヤ 2 フレームの場合、外部ヘッダーの DSCP 値は、VXLAN QoS のデフォルト設定 （464 ページ）に示すハードウェアに存在する CoS/DSCP マッピングから取得されます。このようにして、元の QoS 属性が VXLAN トンネルに保持されます。この動作は、次の図に示します。

図 *39 :* レイヤ *2* フレームから *VXLAN* 外部ヘッダーへの優先順位のコピー

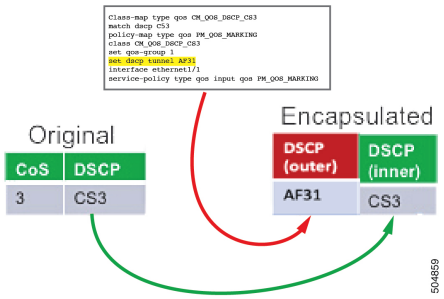
レイヤ 2 フレームでは、IP ヘッダーがフレームに存在しないため、DSCP 値は存在しません。レイヤ 2 フレームがカプセル化されると、元の CoS 値は VXLAN トンネルに保存されません。

外部 DSCP を使用した IP から VXLAN

Cisco NX-OS リリース 10.4(1)F 以降では、外部 DSCP アクションが設定されたポリシーは、入力方向のアクセス インターフェイスに適用できます。

レイヤ 3 パケット向けにトラフィックが VXLAN でカプセル化されると、元のパケットからの DSCP 値が内部ヘッダーにコピーされ、ユーザーが構成した DSCP 値は、VXLAN カプセル化パケットの外部ヘッダーで設定されます。この動作は、次の図に示します。

図 *40 :* セット構成から適用された *VXLAN* 外部 *DSCP* 値

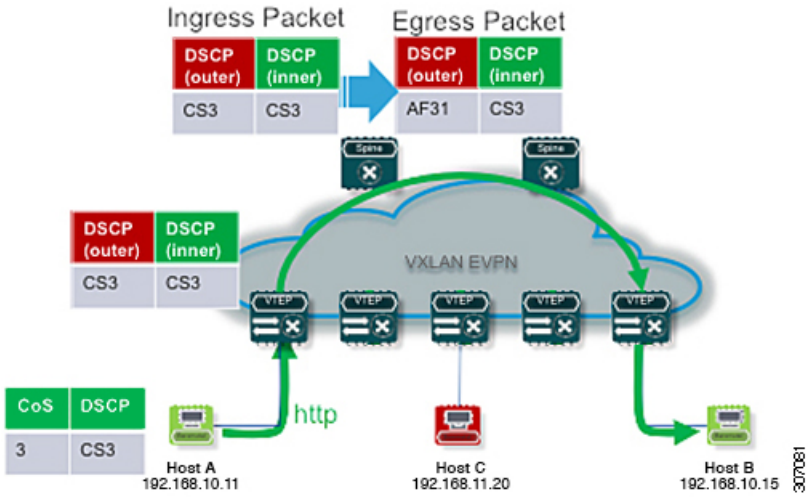


VXLAN トンネルの内部

VXLAN トンネル内では、トラフィックの分類は外部ヘッダーの DSCP 値に基づきます。分類は、DCSP 値と照合するか、または分類に ACL を使用して実行できます。

VXLAN カプセル化トラフィックが信頼境界を通過する場合、パケットのマーキングを変更して、トンネル内の QoS 動作に一致させることができます。マーキングは、新しい DSCP 値が外部ヘッダーにのみ適用される VXLAN トンネルの内部で実行できます。新しい DSCP 値は、VXLAN トンネル内のさまざまな QoS 動作に影響を与える可能性があります。元の DSCP 値は内部ヘッダーに保持されます。

図 *41 : VXLAN* トンネル内部のマーキング



VXLAN から IP

出力 VTEP での分類は、VXLAN トンネルを出るトラフィックに対して実行されます。出力 VTEP での分類では、内部ヘッダーおよび外部 DSCP 値が使用されます。内部または外部 DSCP 値は、優先順位ベースの分類に使用されます。分類は ACL を使用して実行できます。

分類は、すべての VXLAN トンネル トラフィックの NVE インターフェイスで実行されます。

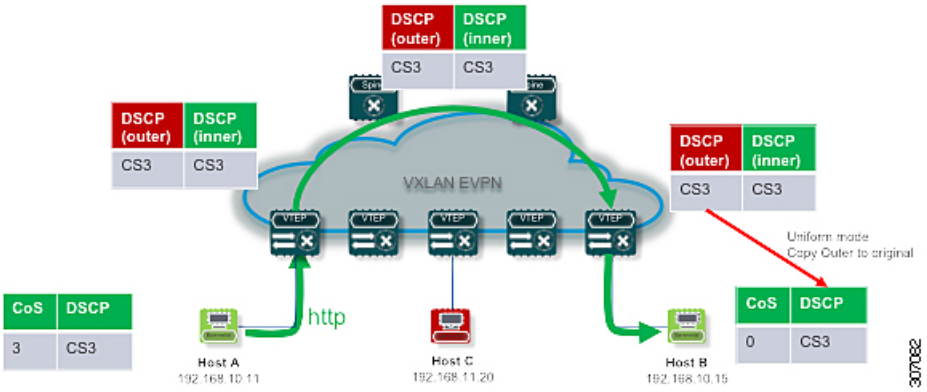
マーキングおよびポリシングは、トンネル トラフィックの NVE インターフェイスで実行できます。マーキングが設定されている場合は、カプセル化解除されたパケットに新しくマーキングされた値が存在します。元の CoS 値はカプセル化されたパケットに保持されないため、ネットワークの残りの部分で QoS の 802.1p フィールドを予期するデバイスのカプセル化解除されたパケットに対してマーキングを実行できます。

カプセル化解除されたパケットの優先順位の選択

出力 VTEP では、パケットから VXLAN ヘッダーが削除され、カプセル化解除されたパケットは DSCP 値を使用してスイッチから出力されます。スイッチは、2 つのモードに基づいてカプセル化解除されたパケットの DSCP 値を割り当てます。

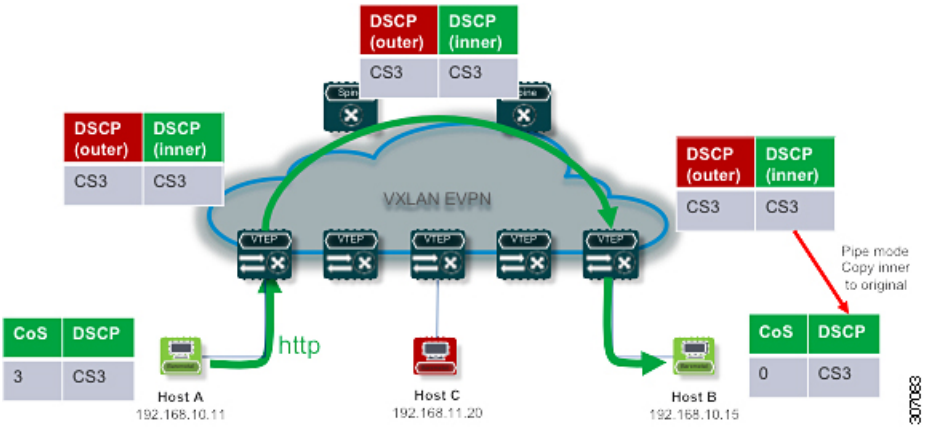
- 均一モード：VXLAN パケットの外部ヘッダーからの DSCP 値がカプセル化解除されたパケットにコピーされます。VXLAN トンネルでの DSCP 値の変更は保持され、カプセル化解除されたパケットに存在します。ユニフォーム モードは、カプセル化解除されたパケット優先選択のデフォルト モードです。

図 *42:* ユニフォーム モードの外部 ***DSCP*** 値がレイヤ *3* パケットのカプセル化解除されたパケット ***DSCP*** 値にコピーされる



- パイプ モード：元の DSCP 値は VXLAN トンネル エンドで保持されます。出力 VTEP で、システムはカプセル化解除されたパケットDSCP値に内部 DSCP 値をコピーします。このように、元の DSCP 値は VXLAN トンネルの終了時に保持されます。

図 *43:* パイプモードの内部 ***DSCP*** 値がレイヤ *3* パケットのカプセル化解除されたパケット ***DSCP*** 値にコピーされる



CoS の保持

Cisco NX-OS リリース 10.4(1)F 以降では、非 IP パケットの CoS 保存を提供するために、**default-vxlan-in-tnl-dscp-policy** QoS ポリシーマップ テンプレートが追加されています。

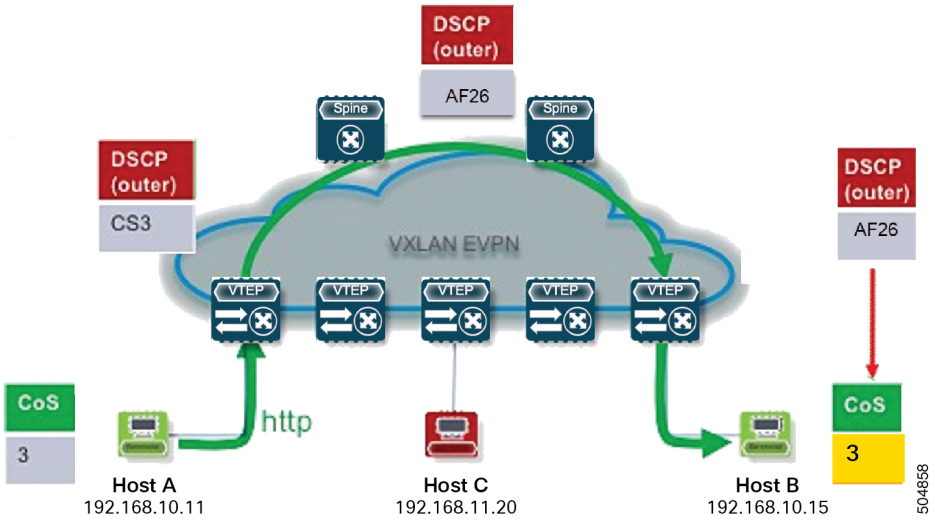
このテンプレートが NVE インターフェイスで有効になっている場合、スイッチは VXLAN パケットの外部 DSCP で照合を実行し、固定外部 DSCP から CoS へのマッピングに基づいて、出力 VTEP のカプセル化解除されたイーサネット パケットの CoS を書き換えます。

次の表に、レイヤ 2 フレームの出力 VTEP でのデフォルトの外部 DSCP-to-CoS マッピングを示します。

表 *9*: デフォルトの外部 *DSCP-to-CoS* マッピング

外部 VXLAN ヘッダーの DSCP	元のレイヤ 2 フレームの CoS
0	0
8	1
16	2
26	3
32	4
46	5
48	6
56	7

図 *44*: カプセル化解除されたパケットで復元された非 *IP CoS* 値



VXLAN QoS の注意事項および制約事項

（注） この機能を設計どおりに動作させるには、QoS ポリシーをエンドツーエンドで設定する必要があります。

VXLAN QoS 設定時の注意事項と制約事項は次のとおりです。

- Cisco Nexus 9364C、9300-EX、9300-FX/FX2/FX3 プラットフォーム スイッチと、-EX/FX および -R/RX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチがサポートされています。

- Cisco NX-OS リリース 9.3(3) 以降、Cisco Nexus 9300-GX プラットフォーム スイッチは、デフォルト モードで VXLAN QoS をサポートします。
- Cisco NX-OS リリース 10.2(3)F 以降、デフォルト モードの VXLAN QoS は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、デフォルト モードの VXLAN QoS は Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、デフォルト モードの VXLAN QoS は Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、デフォルト モードの VXLAN QoS は Cisco Nexus 9364C-H1 スイッチでサポートされます。
- 次の機能は、-R/RX ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチでサポートされます。
  - 物理インターフェイス レベルのキューイングは、通常の L2/L3 キューイング/QoS として機能する必要があります。
  - IPv4 ブリッジ ケースは、内部 ToS を外部 VXLAN ToS にコピーするという点で機能します。
- 次の機能は、-R および -RX ラインカードを備えた Cisco Nexus 9504 および 9508 プラットフォーム スイッチではサポートされません。
  - NVE インターフェイスのポリシー
  - 内部から VXLAN 外部コピーへの IPv6 タイプ オブ サービス（ToS）
  - QoS の IPv4 ルーテッド ケース。内部からの ToS が外部 VXLAN ヘッダーにコピーされない
- -RX ライン カードを使用した Cisco Nexus 9504 および 9508 プラットフォーム スイッチの場合、デフォルト モードは VXLAN カプセル化解除のパイプです（内部パケット DSCP は外部 IP ヘッダー DSCP 値に基づいて変更されません）。これは、他のライン カード タイプとの動作の違いです。-RX ライン カードと他のライン カードを同じネットワークで使用する場合、同じ動作をさせるために、非RX ライン カードが存在するスイッチでこの **qos-mode pipe** コマンドを使用できます。コンフィギュレーション コマンドの詳細については、出力 VTEP でのタイプ QoS の設定 （465 ページ）を参照してください。
- VXLAN QoS は EVPN ファブリックでサポートされます。
- 元の IEEE 802.1Q ヘッダーは VXLAN トンネルに保存されません。CoS 値は、VXLAN カプセル化パケットの内部ヘッダーに存在しません。
- NVE インターフェイスの統計情報（カウンタ）が存在します。
- 出力ポリシングは、encap（入力）VXLAN VTEP の発信インターフェイス（スパインに接続するアップリンク）ではサポートされません。

- vPC で、両方のピアでカプセル化解除されたパケットプライオリティ選択の変更を設定します。
- NVE インターフェイスのこのサービスは、入力方向でのみアタッチできます。
- NVE インターフェイスに DSCP マーキングが存在する場合、BUD ノードへのトラフィックは内部および外部ヘッダーのマーキングを保持します。NVE インターフェイスでマーキング アクションが設定されている場合、Cisco Nexus 9364C および 9300-EX プラットフォーム スイッチでは、BUM トラフィックが新しい DSCP 値でマーキングされます。
- NVEインターフェイスに適用される分類ポリシーは、VXLANカプセル化トラフィックにのみ適用されます。他のすべてのトラフィックでは、着信インターフェイスに分類ポリシーを適用する必要があります。
- カプセル化解除されたパケットに CoS 値をマーキングするには、マーキング ポリシーを NVE インターフェイスに付加して、VLAN ヘッダーが存在するパケットに CoS 値をマーキングする必要があります。
- DCI ハンドオフ ノードの VXLAN QoS 設定には、次のガイドラインと制限事項が適用されます。
  - Cisco NX-OS リリース 9.3(5) 以降、Cisco Nexus 9300-GX プラットフォーム スイッチは、DCI ハンドオフ ノードでの VXLAN QoS 設定をサポートします。
  - Cisco NX-OS リリース 10.2(3)F 以降、Cisco Nexus 9300-GX2 スイッチは、DCI ハンドオフ ノードでの VXLAN QoS 構成をサポートします。
  - Cisco NX-OS リリース 10.4(1)F 以降、Cisco Nexus 9332D-H2R スイッチは、DCI ハンドオフ ノードでの VXLAN QoS 構成をサポートします。
  - Cisco NX-OS リリース 10.4(2)F 以降、Cisco Nexus 93400LD-H1 スイッチは、DCI ハンドオフ ノードでの VXLAN QoS 構成をサポートします。
  - Cisco NX-OS リリース 10.4(3)F 以降、Cisco Nexus 9364C-H1 スイッチは、DCI ハンドオフ ノードでの VXLAN QoS 構成をサポートします。
  - DCI ハンドオフノードの VXLAN QoS 設定は、Cisco Nexus 9336C-FX2、93240YC-FX2、および 9300-GX プラットフォーム スイッチのエンドツーエンド プライオリティ フロー制御（PFC）をサポートしません。
  - VXLAN でカプセル化されたパケットでは、マイクロバースト、ダイナミック パケットプライオリティ（DPP）、およびおおよそのフェアドロップ（AFD）がサポートされます。
- 以下の注意事項および制約事項は、外部 DSCP ベース VXLAN QoS ポリシー機能に適用されます。
  - Cisco NX-OS リリース 10.4(1)F 以降、外部 DSCP ベースの VXLAN QoS ポリシー機能は、Cisco Nexus 9300-FX2/FX3/GX/GX2 プラットフォーム スイッチおよび N9K-X9716D-GX ライン カードを搭載した 9500 スイッチでサポートされます。

- Cisco NX-OS リリース 10.4(2)F 以降、外部 DSCP ベースの VXLAN QoS ポリシー機能は Cisco Nexus 9332D-H2R、および 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、外部 DSCP ベースの VXLAN QoS ポリシー機能は Cisco Nexus 9364C-H1 スイッチでサポートされます。
- VXLAN QoS ポリシーでは、この **match dscp tunnel** コマンドは NVE インターフェイスおよび入力方向にのみ適用できます。
- VXLAN QoS ポリシーでは、内部と外部の両方の DSCP 照合ルールはサポートされていません。ただし、NVE インターフェイスに適用される同じポリシー内の **ip access-lists** や **mac access-list** などの一致基準は、常に内部ヘッダーで照合されます。
- 非 IP パケットの場合、NVE インターフェイスの外部ヘッダー QoS ポリシーは、L2 書き換えおよびトラフィック クラス割り当てまたは発信キューのみをサポートします。ポリサーのようなアクションはサポートされません。
- VXLAN QoS ポリシーでは、NVE インターフェイスの **match dscp tunnel** コマンドは、現在の VTEP 宛ての VXLAN パケットに対して照合を実行します。ここで、トンネルの終了が発生し、パケットのカプセル化が解除されます。
- VXLAN QoS ポリシーでは、**match dscp tunnel** コマンドは非 IP パケットをサポートしません。このため、CoS の保持は IPv6 アンダーレイでは機能しません。
- VXLAN QoS ポリシーでは、**set dscp tunnel** コマンドは非 IP パケットをサポートしません。非 IP パケットの場合、外部 DSCP 値は、スイッチ上のデフォルトの CoS から DSCP へのマッピング情報に基づいて適用されます。
- VXLAN QoS ポリシーでは**set dscp tunnel** コマンドはカプセル化パケットに適用されるため、このコマンドは NVE インターフェイスに適用できません。
- **set dscp tunnel** コマンドが VXLAN マルチサイトの入力 VTEP に適用されると、ボーダー ゲートウェイでパイプ モードが構成されている場合、外部 DSCP 値が内部 DSCP に置き換えられる可能性があります。新しい外部 DSCP ヘッダーをリモート サイトに伝送するように、ボーダー ゲートウェイで均一モードを設定することを推奨します。
- 外部 DSCP ベースの VXLAN QoS ポリシー機能は、VXLAN マルチサイト展開ではサポートされていません。

- ボーダー ゲートウェイ（BGW）スパインを使用する場合、VXLAN QoS ポリシーには次の制限が適用されます。
  - マルチキャスト アンダーレイを使用する VNI のサイト内 BUM トラフィックに QoS ポリシーが必要であり、そのマルチキャスト アンダーレイ グループが BGW スパインで定義された VNI によっても所有されている場合は、QoS ポリシーを NVE インターフェイスに適用する必要があります。NVE インターフェイスは着信インターフェイスとして機能するため、ファブリック インターフェイスに適用される QoS ポリシーはこれらのフローを変更しません。
  - マルチキャスト アンダーレイを使用する VNI のサイト内 BUM トラフィックに QoS ポリシーが必要であり、そのマルチキャスト グループが BGW スパインで定義された

VNI によって所有されていない場合は、QoS ポリシーをファブリック インターフェイスに適用する必要があります。NVE インターフェイスに適用される QoS ポリシーは、NVE が着信インターフェイスと見なされないため、これらのフローを変更しません。

- BGW スパインの NVE インターフェイスが、ローカル ファブリック内の BUM トラフィックに使用されるマルチキャスト グループを所有している場合、そのマルチキャスト グループのサイト内フローとサイト間フローの処理を区別するために、ファブリック インターフェイスと NVE インターフェイスの両方に QoS ポリシーを適用することはできません。

- Cisco NX-OS リリース 10.4(3)F 以降、X9836DM-A および X98900CD-A ライン カードを搭載した Cisco Nexus 9808/9804 スイッチは、BGW スパインを使用する場合、VXLAN QoS ポリシーをサポートしますが、次の制限があります。
  - 物理入力 QoS ポリシーとシステムレベルの QoS ポリシーがサポートされます。
  - NVE 上の QoS ポリシーはサポートされません。
  - 明示的輻輳通知（ECN）または ECN 対応トランスポート（ECT）マーキングは保持されません。

VXLAN QoS のデフォルト設定

次の表に、レイヤ 2 フレームの入力 VTEP でのデフォルトの CoS/DSCP マッピングを示します。

表 *10*：デフォルトの *CoS-to-DSCP* マップ

元のレイヤ 2 フレームの CoS	外部 VXLAN ヘッダーの DSCP
0	0
1	8
2	16
3	26
4	32
5	46
6	48
7	56

VXLAN QoS の設定

VXLAN QoS の設定

VXLAN QoS の設定は、MQC モデルを使用して行われます。QoS 設定に使用されるのと同じ設定が VXLAN QoS に適用されます。QoS の設定の詳細については、『Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide、Release 9.2(x)』を参照してください。

VXLAN QoS では、NVE（ネットワーク仮想インターフェイス）という新しいサービスポリシー接続ポイントが導入されています。出力VTEPでは、トラフィックがカプセル化解除されるポイントはNVEインターフェイスです。すべてのVXLANトラフィックを考慮するには、サービス ポリシーを NVE インターフェイスにアタッチする必要があります。

次のセクションでは、出力VTEPでの分類の設定と、NVEインターフェイスへの**service-policy type qos** 接続について説明します。

出力 VTEP でのタイプ QoS の設定

VXLAN QoS の設定は、MQC モデルを使用して行われます。同じ設定が VXLAN QoS の QoS 設定に使用されます。QoS の設定の詳細については、『Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide、Release 9.2(x)』を参照してください。

VXLANQoSは、ネットワーク仮想インターフェイス（NVE）である新しいサービス ポリシー接続ポイントを導入します。出力 VTEP で、NVE インターフェイスはトラフィックがカプセル化解除される場所を指します。すべてのVXLANトラフィックを考慮するには、サービス ポリシーを NVE インターフェイスにアタッチする必要があります。

この手順では、出力 VTEP での分類の設定と、NVE インターフェイスへの **service-policy type qos** 接続について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**[no] class-map [type [qos]] | [match-all] | [match-any]** *class-map-name* 例： `switch(config)# class-map type qos class1`	*class--map-name* という名前のクラス マップを作成するか、またはそのクラスマップにアクセスして、**class-map** モードを開始します。*class--map-name* 引数は、英字、ハイフン、またはアンダースコア文字を含むことができ、最大 40 文字を含むことができます。（**no** オプションが選択され、複数の match ステートメントが入力される場合、デフォルトは match-any です）。

	コマンドまたはアクション	目的
ステップ 3	**[no] match [access-group | cos | dscp [tunnel] | precedence]** *{name | 0-7 | 0-63 | 0-7}* 例： `switch(config-cmap-qos)# match dscp tunnel 26`	アクセスリスト、**cos** 値、**dscp** 値、または IP **precedence** 値に基づいてパケットを照合することにより、トラフィック クラスを設定します。 Cisco NX-OS リリース 10.4(1)F 以降では、入力パケットの外部 VXLAN ヘッダーの DSCP 値と一致する **トンネル** オプションが提供されます。 （注） **match dscp tunnel** コマンドは、出力 VTEP の NVE インターフェイスに適用される入力サービス ポリシーで使用されます。
ステップ 4	**[no] policy-map type qos** *policy-map-name* 例： `switch(config-cmap-qos)# policy-map type qos policy`	*policy-map-name* という名前のポリシー マップを作成するか、そのポリシー マップにアクセスし、ポリシー マップ モードを開始します。ポリシー マップ名には、アルファベット、ハイフン、またはアンダースコア文字を含めることができます。ポリシー マップ名は大文字と小文字が区別され、最大 40 文字まで設定できます。
ステップ 5	**[no] class** *class-name* 例： `switch(config-pmap-qos)# class class1`	class-name への参照を作成し、ポリシー マップ クラス コンフィギュレーション モードを開始します。insert-before を使用して事前挿入するクラスを指定しない限り、ポリシー マップの末尾にクラスが追加されます。ポリシー マップ内のクラスと現在一致していないトラフィックをすべて選択するには、class-default キーワードを使用します。
ステップ 6	**[no] set qos-group** *qos-group-value* 例： `switch(config-pmap-c-qos)# set qos-group 1`	QoS グループの値を *qos-group-value* に設定します。値の範囲は 0 ～ 126 です。 **qos-group** は、一致基準として type queuing および type network-qos で参照されます。
ステップ 7	**exit** 例： `switch(config-pmap-c-qos)# exit`	クラスマップ モードを終了します。
ステップ 8	**[no] interface nve** *nve-interface-number* 例： `switch(config)# interface nve 1`	インターフェイス モードを開始して、NVE インターフェイスを設定します。
ステップ 9	**[no] service-policy type qos input** *policy-map-name* 例： `switch(config-if-nve)# service-policy type qos input policy`	入力方向のインターフェイスに service-policy *policy-map-name* を追加します。NVE インターフェイスには 1 つの入力ポリシーにのみ付加できます。

	コマンドまたはアクション	目的
ステップ 10	（任意） **[no] qos-mode [pipe]** 例： `switch(config-if-nve)# qos-mode pipe`	カプセル化解除されたパケットの優先順位の選択およびパイプ モードの使用。このコマンドの **no** 形式を入力すると、パイプ モードが無効になり、デフォルトは均一モードになります。

入力 VTEP での外部 DSCP の構成

VXLAN QoS ポリシーは、すべての VXLAN トラフィックに対して新しい外部 DSCP 設定アクションを導入します。サービス ポリシーは、入力 VTEP のアクセス（入力）インターフェイスに接続する必要があります。

手順の概要

1. **configure terminal**
2. **[no] class-map [type qos] [match-all]| [match-any]** *class-map-name*
3. **[no] policy-map type qos** *policy-map-name*
4. **[no] class** *class-name*
5. **[no] set dscp [tunnel]** *dscp-val*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**[no] class-map [type qos] [match-all]| [match-any]** *class-map-name* 例： `switch(config)# class-map type qos class1`	*class--map-name* という名前のクラス マップを作成するか、またはそのクラス マップにアクセスして、**class-map** モードを開始します。*class--map-name* 引数は、英字、ハイフン、またはアンダースコア文字を含むことができ、最大 40 文字を含むことができます。（**no** オプションが選択され、複数の match ステートメントが入力される場合、デフォルトは match-any です）。
ステップ 3	**[no] policy-map type qos** *policy-map-name* 例： `switch(config-cmap-qos)# policy-map type qos policy`	*policy-map-name* という名前のポリシー マップを作成するか、そのポリシー マップにアクセスし、ポリシー マップ モードを開始します。ポリシー マップ名には、アルファベット、ハイフン、またはアンダースコア文字を含めることができます。ポリシー

	コマンドまたはアクション	目的
		マップ名は大文字と小文字が区別され、最大 40 文字まで設定できます。
ステップ 4	**[no] class** *class-name* 例： `switch(config-pmap-qos)# class class1`	class-name への参照を作成し、ポリシー マップ クラス コンフィギュレーション モードを開始します。insert-before を使用して事前挿入するクラスを指定しない限り、ポリシーマップの末尾にクラスが追加されます。ポリシーマップ内のクラスと現在一致していないトラフィックをすべて選択するには、class-default キーワードを使用します。
ステップ 5	**[no] set dscp [tunnel]** *dscp-val* 例： `switch(config-pmap-c-qos)# set dscp tunnel 32`	入力パケットの外部 VXLAN ヘッダーに DSCP 値を設定します。

VXLAN QoS 設定の確認

表 *11 : VXLAN QoS* 検証コマンド

コマンド	目的
show class map	すべての設定されたクラス マップに関する情報を表示します。
show policy-map	すべての設定済みのポリシー マップに関する情報を表示します。
show running ipqos	スイッチに設定済の QoS を表示します。

VXLAN QoS 設定例

入力 VTEP の分類とマーキング

次に、ACL とトラフィックを分類するための **class-map type qos** コマンドを設定する例を示します。**policy-map type qos** コマンドを入力して、トラフィックを qos-group 1 に入れ、DSCP 値を設定します。入力方向で入力インターフェイスに接続する **service-policy type qos** コマンドを入力して、ACL に一致するトラフィックを分類します。

```
access-list ACL_QOS_DSCP_CS3 permit ip any any eq 80

class-map type qos CM_QOS_DSCP_CS3
 match access-group name ACL_QOS_DSCP_CS3

policy-map type qos PM_QOS_MARKING
```

```
 class CM_QOS_DSCP_CS3
 set qos-group 1
 set dscp 24

interface ethernet1/1
 service-policy type qos input PM_QOS_MARKING
```

トランジット スイッチ：スパイン分類

次に、入力 VTEP で設定された DSCP 24 に一致する分類の **class-map type qos** コマンドを設定する例を示します。コマンドを入力して、トラフィックをqos-group 1に入れます。**policy-map type qos** 入力方向で入力インターフェイスに付加する **service-policy type qos** コマンドを入力して、トラフィック一致基準を分類します。

```
class-map type qos CM_QOS_DSCP_CS3
 match dscp 24

policy-map type qos PM_QOS_CLASS
 class CM_QOS_DSCP_CS3
 set qos-group 1

interface Ethernet 1/1
 service-policy type qos input PM_QOS_CLASS
```

出力 VTEP の分類とマーキング

次に、DSCP値でトラフィックを分類するためのコマンドを設定する例を示します。**class-map type qos** qos-group 1 にトラフィックを配置し、出力フレームでCoS値をマークするには、**policy-map type qos** を入力します。**service-policy type qos** コマンドは入力方向の NVE インターフェイスに適用され、VXLAN トンネルから発信されるトラフィックを分類します。

```
class-map type qos CM_QOS_DSCP_CS3
 match dscp 24

policy-map type qos PM_QOS_MARKING
 class CM_QOS_DSCP_CS3
 set qos-group 1
 set cos 3

interface nve 1
 service-policy type qos input PM_QOS_MARKING
```

キューイング

次に、qos-group 1 のトラフィックに対して **policy-map type queueing** コマンドを設定する例を示します。qos-group 1 にマッピングされたq1に使用可能な帯域幅の 50% を割り当て、**system qos** コマンドを使用してすべてのポートに出力方向のポリシーを適用します。

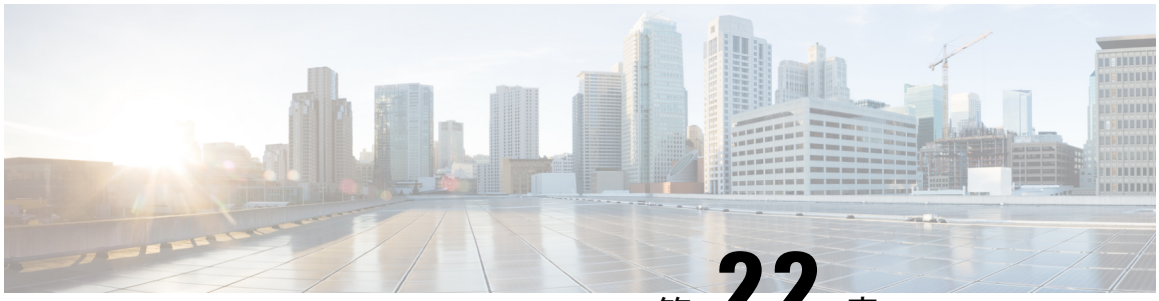
```
policy-map type queuing PM_QUEUING
class type queuing c-out-8q-q7
      priority level 1
    class type queuing c-out-8q-q6
      bandwidth remaining percent 0
    class type queuing c-out-8q-q5
      bandwidth remaining percent 0
    class type queuing c-out-8q-q4
```

```
      bandwidth remaining percent 0
    class type queuing c-out-8q-q3
      bandwidth remaining percent 0
    class type queuing c-out-8q-q2
      bandwidth remaining percent 0
    class type queuing c-out-8q-q1
      bandwidth remaining percent 50
    class type queuing c-out-8q-q-default
      bandwidth remaining percent 50

system qos
 service-policy type queueing output PM_QUEUING
```

CoS 保存の設定

次の例は、NVE インターフェイスで CoS 保存を設定する方法を示しています。

```
interface nve 1
   service-policy type qos input default-vxlan-in-tnl-dscp-policy
```



第 22 章

BGP EVPN フィルタリングの設定

この章は、次の内容で構成されています。

BGP EVPNフィルタリングについて

この機能では、アドレス ファミリ L2VPN EVPN の BGP NLRI の実装に起因する、ルート フィルタリングと属性処理の要件について説明します。

EVPN ルートは、NLRI 形式の通常の IPv4 および IPv6 ルートとは大きく異なります。これらには多くのフィールドが含まれ、EVPN に固有の属性を保持します。ルート マップを使用すると、これらの属性に基づいてルートをフィルタリングできます。EVPN アドレス ファミリに属するルートには、次のルート フィルタリング オプションを使用できます。

- EVPN ルート タイプに基づく照合：EVPN では 6 種類の NLRI を使用できます。照合は、route-map match ステートメントで指定されたタイプに基づきます。
- NLRI の MAC アドレスに基づく照合：このオプションは、NLRI に組み込まれた IP アドレスに基づく照合に似ています。EVPN タイプ 2 ルートには、IP アドレスとともに MAC アドレスが含まれています。このオプションは、このようなルートをフィルタリングするために使用できます。
- RMAC 拡張コミュニティに基づく照合：EVPN タイプ 2 およびタイプ 5 ルートは、MAC アドレスを伝送するルータ MAC（RMAC）拡張コミュニティを伝送します。RMAC は、他の拡張コミュニティ情報とともにネイバーへの更新メッセージの一部としてアドバタイズされます。ルートのリモート ネクスト ホップの MAC アドレスを指定します。このオプションを使用すると、この RMAC 拡張コミュニティとの照合が可能になります。
- RMAC 拡張コミュニティの設定：このオプションでは、EVPN NLRI の RMAC 拡張コミュニティ値を変更できます。

- EVPN ネクスト ホップ IP アドレスの設定：このオプションは、一致条件が満たされると、EVPN ルートのネクスト ホップ IP アドレスを設定します。EVPN ルートのネクスト ホップ IP アドレスを設定するには、転送の正確性を確保するために RMAC 拡張コミュニティを設定する必要があります。
- ルート タイプ 5 のゲートウェイ IP アドレスの設定：ゲートウェイ IP アドレスは、タイプ 5 EVPN ルートを形成する IP プレフィックスのオーバーレイ IP インデックスをエンコードします。更新メッセージで EVPN NLRI の一部としてアドバタイズされます。デフォルト値は 0.0.0.0 です。他の値に設定されている場合、VRF コンテキスト内のルートのネクストホップは、指定されたゲートウェイ IP アドレスに変更されます。
- テーブル マップの使用：テーブル マップを設定して、レイヤ 2 ルーティング情報ベース（L2RIB）にダウンロードされた MAC ルートをフィルタリングできます。

この章の残りの部分では、これらのオプションの設定と適用について説明します。

BGP フィルタリングの注意事項と制限事項

BGP EVPN フィルタリングの注意事項と制約事項は次のとおりです。

Cisco Nexus 9000 シリーズスイッチは、BGP EVPN フィルタリングをサポートしています。

ルートの EVPN アドレス ファミリのフィルタリングには、次の match および set オプションを使用できます。

- ルート タイプに基づく照合
- NLRI の MAC アドレスに基づく照合
- RMAC 拡張コミュニティに基づく照合
- RMAC 拡張コミュニティの設定
- EVPN ネクスト ホップ IP アドレスの設定：複数のネクスト ホップ IP アドレスが設定されている場合、最初のアドレスのみが使用され、EVPN に使用されます。IPv4 および IPv6 は、ネクストホップ アドレスとして使用できます。
- ルート タイプ 5 のゲートウェイ IP アドレスの設定：**route-map** コマンドを使用して IPv4 ゲートウェイ IP アドレスを設定できます。
- テーブル マップの使用：MAC ルートをフィルタリングするためのテーブル マップがレイヤ 2 ルーティング情報ベース（L2RIB）にダウンロードされます。

BGP EVPN フィルタリングの設定

EVPN アドレス ファミリ ルートのルート フィルタリングを実行するには、次のタスクを実行します。

テーブル マップ設定モードでは、次のタスクを実行できます。

match および set 句を使用したルート マップの設定

match および set 句とともに既存のルートマップ設定を使用して、必要なフィルタリングの種類を決定できます。

EVPN ルート タイプに基づく照合

手順の概要

1. **configure terminal**
2. **route-map** *route-map-name*
3. **match evpn route-type {1 | 2 | 2-mac-ip | 2-mac-only | 3 | 4 | 5 | 6}**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。

	コマンドまたはアクション	目的
ステップ 3	**match evpn route-type {1 | 2 | 2-mac-ip | 2-mac-only | 3 | 4 | 5 | 6}** 例： `switch(config-route-map)# match evpn route-type 6`	BGP EVPN ルートを照合します。

NLRI の MAC アドレスに基づく照合

手順の概要

1. **configure terminal**
2. **mac-list** *list-name* **[seq** *seq-number***] {deny | permit}** *mac-address* **[mac-mask]**
3. **route-map** *route-map-name*
4. **match mac-list** *mac-list-name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**mac-list** *list-name* **[seq** *seq-number***] {deny | permit}** *mac-address* **[mac-mask]** 例： `switch(config)# mac-list MAC_LIST_1 permit E:E:E`	MAC リストを構築します。
ステップ 3	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。
ステップ 4	**match mac-list** *mac-list-name* 例： `switch(config-route-map)# match mac-list MAC_LIST_1`	MAC リストのエントリを照合します。最大で 63 文字です。

RMAC 拡張コミュニティに基づく照合

手順の概要

1. **configure terminal**
2. **ip extcommunity-list standard** *list-name* **seq** *5* **{deny | permit} rmac** *mac-addr*
3. **route-map** *route-map-name*
4. **match extcommunity** *list-name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**ip extcommunity-list standard** *list-name* **seq** *5* **{deny | permit} rmac** *mac-addr* 例： `switch(config)# ip extcommunity-list standard EXTCOMM_LIST_RMAC seq 5 permit rmac a8b4.56e4.7edf`	extcommunity リスト エントリを追加します。*list-name* 引数は 63 文字を超えることはできません。
ステップ 3	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。
ステップ 4	**match extcommunity** *list-name* 例： `switch(config-route-map)# match extcommunity EXTCOMM_LIST_RMAC`	拡張コミュニティ リスト名と一致します。

RMAC 拡張コミュニティの設定

手順の概要

1. **configure terminal**
2. **route-map** *route-map-name*
3. **set extcommunity evpn rmac** *mac-address*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。
ステップ 3	**set extcommunity evpn rmac** *mac-address* 例： `switch(config-route-map)# set extcommunity evpn rmac EEEE.EEEE.EEEE`	BGP RMAC extcommunity 属性を設定します。

EVPN ネクストホップ IP アドレスの設定

手順の概要

1. **configure terminal**
2. **route-map** *route-map-name*
3. **set ip next-hop** *next-hop*
4. **set ipv6 next-hop** *next-hop*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。
ステップ 3	**set ip next-hop** *next-hop* 例：	EVPN IP ネクスト ホップの IP アドレスを設定します。

	コマンドまたはアクション	目的
	`switch(config-route-map)# set ip next-hop 209.165.200.226`	
ステップ 4	**set ipv6 next-hop** *next-hop* 例： `switch(config-route-map)# set ipv6 next-hop 2001:0DB8::1`	IPv6 ネクストホップ アドレスを設定します。

ルート タイプ 5 のゲートウェイ IP アドレスの設定

手順の概要

1. **configure terminal**
2. **route-map** *route-map-name*
3. **set evpn gateway-ip** *gw-ip-address*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。
ステップ 3	**set evpn gateway-ip** *gw-ip-address* 例： `switch(config-route-map)# set evpn gateway-ip 209.165.200.227`	ゲートウェイの IP アドレスを設定します。

着信または発信レベルでのルート マップの適用

要件に基づいて match および set 句を使用してルートマップを設定したら、この手順を使用してインバウンドまたはアウトバウンド レベルでルートマップを適用します。

手順の概要

1. **configure terminal**
2. **router bgp** *as-num*

3. **neighbor** *address*
4. **address-family l2vpn evpn**
5. **route-map** ルート マップ **{in | out}**

手順の詳細

手順

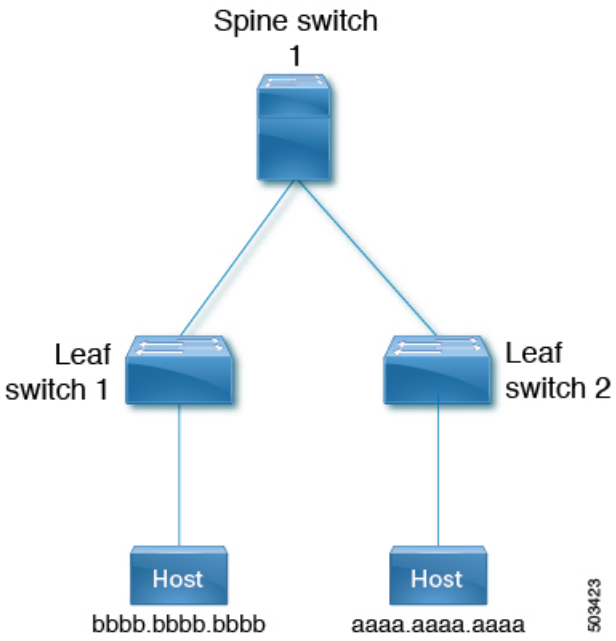
	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router bgp** *as-num* 例： `switch(config)# router bgp 100`	ルーティング プロセスをイネーブルにします。*as-num* の範囲は 1 ～ 65535 です。
ステップ 3	**neighbor** *address* 例： `switch(config-router)# neighbor 1.1.1.1`	BGP ネイバーを設定します。
ステップ 4	**address-family l2vpn evpn** 例： `switch(config-router-neighbor)# address-family l2vpn evpn`	L2VPN アドレス ファミリを設定します。
ステップ 5	**route-map** ルート マップ **{in	out}** 例： `switch(config-router-neighbor-af)# route-map ROUTE_MAP_1 in`	ルート マップをネイバーに適用します。

BGP EVPN フィルタリングの設定例

ここでは、EVPN ルートをフィルタリングするための設定例を示します。

例 1

次に、EVPN タイプ 2 ルートをフィルタリングし、RMAC 拡張コミュニティを 52fc.c310.2e80 として設定する例を示します。



1. 次の出力は、ルート マップが適用される前の EVPN テーブル内のルートとタイプ 2 EVPN MAC ルートを示しています。

```
leaf1(config)# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 12, Local Router ID is 1.1.1.1
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 - best2

Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 1.1.1.1:32868    (L2VNI 101)
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i

Route Distinguisher: 3.3.3.3:3
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
33.33.33.33                          100          0 i
*>i[5]:[0]:[0]:[24]:[101.0.0.0]/224
3.3.3.3                  0           100          0 ?

Route Distinguisher: 3.3.3.3:32868
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i

Route Distinguisher: 1.1.1.1:3    (L3VNI 100)
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
33.33.33.33                          100          0 i
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i
*>l[5]:[0]:[0]:[24]:[10.0.0.0]/224
1.1.1.1                  0           100      32768 ?
*>l[5]:[0]:[0]:[24]:[100.0.0.0]/224
1.1.1.1                  0           100      32768 ?
*>i[5]:[0]:[0]:[24]:[101.0.0.0]/224
3.3.3.3                  0           100          0 ?

leaf1(config)# show bgp l2vpn evpn aaaa.aaaa.aaaa
```

```
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1.1.1.1:32868    (L2VNI 101)
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 12
Paths: (1 available, best #1)
Flags: (0x000212) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop, in rib
Imported from 3.3.3.3:32868:[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:
[101.0.0.3]/272
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
Router MAC:52fc.d83a.1b08
Originator: 3.3.3.3 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 3.3.3.3:32868
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 8
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
Imported to 3 destination(s)
Imported paths list: vni100 default default
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
Router MAC:52fc.d83a.1b08
Originator: 3.3.3.3 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 1.1.1.1:3    (L3VNI 100)
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 11
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
Imported from 3.3.3.3:32868:[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:
[101.0.0.3]/272
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
Router MAC:52fc.d83a.1b08
Originator: 3.3.3.3 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer
```

2. 次に、ルート マップの設定例を示します。

```
leaf1(config)# show run rpm

!Command: show running-config rpm
!Running configuration last done at: Thu Sep  3 22:32:23 2020
!Time: Thu Sep  3 22:32:31 2020

version 9.3(5) Bios:version
route-map FILTER_EVPN_TYPE2 permit 10
    match evpn route-type 2
    set extcommunity evpn rmac 52fc.c310.2e80
route-map allow permit 10
```

3. 次に、ルート マップをインバウンド ルート マップとして EVPN ピアに適用する例を示します。

```
leaf1(config-router-neighbor-af)# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Mon Aug  3 18:08:24 2020
!Time: Mon Aug  3 18:08:28 2020

version 9.3(5) Bios:version
feature bgp

router bgp 100
  event-history detail size large
  neighbor 101.101.101.101
    remote-as 100
    update-source loopback0
    address-family l2vpn evpn
      send-community extended
      route-map FILTER_EVPN_TYPE2 in
  vrf vni100
    address-family ipv4 unicast
      advertise l2vpn evpn
      redistribute direct route-map allow
```

4. 次の出力は、ルート マップが適用された後の EVPN テーブルのルートとタイプ 2 EVPN MAC ルートを示しています。

```
leaf1(config)# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 19, Local Router ID is 1.1.1.1
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 - best2

Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 1.1.1.1:32868    (L2VNI 101)
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
                     33.33.33.33                       100          0 i

Route Distinguisher: 3.3.3.3:3
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
                     33.33.33.33                       100          0 i

Route Distinguisher: 3.3.3.3:32868
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
                     33.33.33.33                       100          0 i

Route Distinguisher: 1.1.1.1:3    (L3VNI 100)
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
```

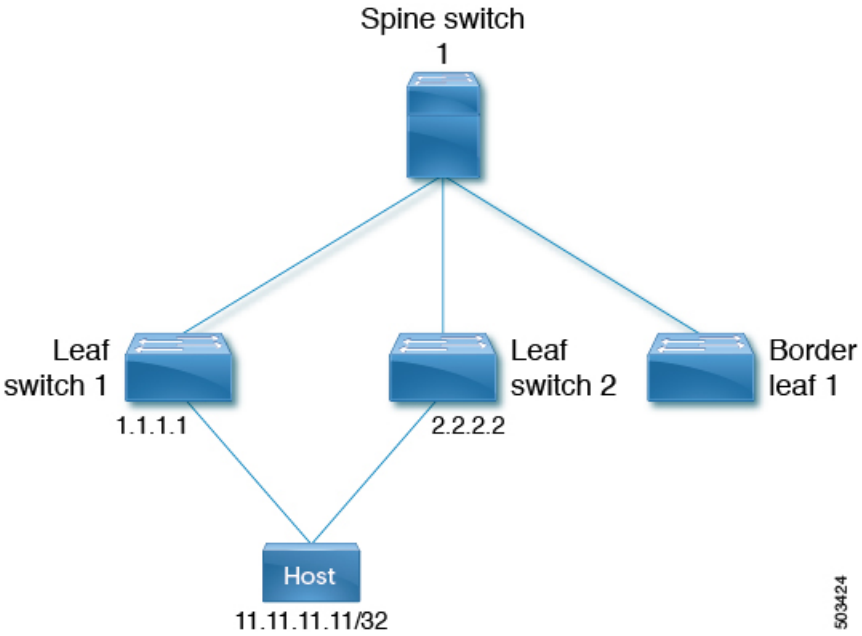
```
                      33.33.33.33                          100          0 i
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
                      33.33.33.33                          100          0 i
*>l[5]:[0]:[0]:[24]:[10.0.0.0]/224
                      1.1.1.1                  0           100      32768 ?
*>l[5]:[0]:[0]:[24]:[100.0.0.0]/224
                      1.1.1.1                  0           100      32768 ?

leaf1(config)# show bgp l2vpn evpn aaaa.aaaa.aaaa
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1.1.1.1:32868    (L2VNI 101)
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 19
Paths: (1 available, best #1)
Flags: (0x000212) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop, in rib
Imported from 3.3.3.3:32868:[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:
[101.0.0.3]/272
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
Router MAC:52fc.c310.2e80
Originator: 3.3.3.3 Cluster list: 101.101.101.101
Path-id 1 not advertised to any peer

Route Distinguisher: 3.3.3.3:32868
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 15
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
Imported to 3 destination(s)
Imported paths list: vni100 default default
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
Router MAC:52fc.c310.2e80
Originator: 3.3.3.3 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 1.1.1.1:3    (L3VNI 100)
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 18
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
Imported from 3.3.3.3:32868:[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:
[101.0.0.3]/272
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
```

```
Router MAC:52fc.c310.2e80
Originator: 3.3.3.3 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer
```

同様に、他の EVPN 固有の match 句と set 句を既存のルート マップ オプションとともに使用して、必要に応じて EVPN ルートをフィルタリングできます。

例 2

次に、EVPN ルート フィルタリングを使用して、EVPN ルートが学習された VTEP とは異なる VTEP にトラフィックをリダイレクトする例を示します。これには、ネクスト ホップ IP アドレスと、他の VTEP に対応するルートの RMAC の設定が含まれます。



この例では、次のことを示します。

- ホスト 1 は VRF evpn-tenant-0002 と VLAN 3002 に属し、リーフ 1 とリーフ 2 に接続されます。
- ホスト 1 への到達可能性は、リーフ 1 およびリーフ 2 によって BL1 にアドバタイズされます。

BL1では、11.11.11.11 / 32 への両方のルートが次のように受信されます。

- リーフ 1 である 1.1.1.1 から 1 つ
- リーフ2 である 2.2.2.2 から 1 つ

1. 最初に、11.11.11.11 に到達するためのベスト パスは 1.1.1.1 です。

```
bl1(config)# show bgp l2 e 11.11.11.11
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1.1.1.1:3
```

```
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 15
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
    Imported to 2 destination(s)
    Imported paths list: evpn-tenant-0002 default
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  1.1.1.1 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0074.caf5
    Originator: 1.1.1.1 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 2.2.2.2:4
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 79
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
    Imported to 2 destination(s)
    Imported paths list: evpn-tenant-0002 default
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  2.2.2.2 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0090.433e
    Originator: 2.2.2.2 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 3.3.3.3:3    (L3VNI 3003002)
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 80
Paths: (2 available, best #2)Flags: (0x000002) (high32 00000000) on xmit-list, is
not in l2rib/evpn, is not in HW

Path type: internal, path is valid, not best reason: Router Id, no labeled nexthop
    Imported from 2.2.2.2:4:[5]:[0]:[0]:[32]:[11.11.11.11]/224
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  2.2.2.2 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0090.433e
    Originator: 2.2.2.2 Cluster list: 101.101.101.101

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
    Imported from 1.1.1.1:3:[5]:[0]:[0]:[32]:[11.11.11.11]/224
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  1.1.1.1 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0074.caf5
    Originator: 1.1.1.1 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer
```

```
Route Distinguisher: 3.3.3.3:4    (L3VNI 3003003)
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 24
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn

Advertised path-id 1
Path type: local, path is valid, is best path, no labeled nexthop
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  3.3.3.3 (metric 0) from 0.0.0.0 (3.3.3.3)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003003
    Extcommunity: RT:1:3003003 ENCAP:8 Router MAC:5254.006a.435b
    Originator: 1.1.1.1 Cluster list: 101.101.101.101

Path-id 1 advertised to peers:
101.101.101.101

bl1(config)# show ip route 11.11.11.11
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

11.11.11.11/32, ubest/mbest: 1/0
*via 1.1.1.1, [200/0], 00:02:51, bgp-1, internal, tag 150 (evpn) segid: 3003
002 tunnelid: 0x1010101 encap: VXLAN
```

2. トラフィックを他の VTEP リーフ 2 にリダイレクトするには、ルートマップ設定を使用して 11.11.11.11/32 ルートのネクスト ホップと RMAC を設定します。

```
bl1(config-route-map)# show run rpm

Command: show running-config rpm
!Running configuration last done at: Wed Mar 27 00:12:14 2019
!Time: Wed Mar 27 00:12:17 2019

version 9.2(3) Bios:version
ip prefix-list PFX_LIST1_1 seq 5 permit 11.11.11.11/32
route-map TEST_SET_IP_NEXTHOP permit 10
    match ip address prefix-list PFX_LIST1_1
    set ip next-hop 2.2.2.2
    set extcommunity evpn rmac 5254.0090.433e
```

3. BL1 のインバウンド レベルでルート マップを適用すると、ルート 11.11.11.11/32 のルート出力は次のようになります。

```
bl1(config-router-neighbor-af)# show bgp l2 e 11.11.11.11
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1.1.1.1:3
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 81
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
    Imported to 2 destination(s)
    Imported paths list: evpn-tenant-0002 default
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
```

```
  2.2.2.2 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0090.433e
    Originator: 1.1.1.1 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 2.2.2.2:4
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 79
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
    Imported to 2 destination(s)
    Imported paths list: evpn-tenant-0002 default
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  2.2.2.2 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0090.433e
    Originator: 2.2.2.2 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 3.3.3.3:3    (L3VNI 3003002)
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 82
Paths: (2 available, best #2)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW

Path type: internal, path is valid, not best reason: Router Id, no labeled nexthop
    Imported from 2.2.2.2:4:[5]:[0]:[0]:[32]:[11.11.11.11]/224
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  2.2.2.2 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0090.433e
    Originator: 2.2.2.2 Cluster list: 101.101.101.101

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop
    Imported from 1.1.1.1:3:[5]:[0]:[0]:[32]:[11.11.11.11]/224
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
  2.2.2.2 (metric 81) from 101.101.101.101 (101.101.101.101)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003002
    Extcommunity: RT:1:3003002 ENCAP:8 Router MAC:5254.0090.433e
    Originator: 1.1.1.1 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer

Route Distinguisher: 3.3.3.3:4    (L3VNI 3003003)
BGP routing table entry for [5]:[0]:[0]:[32]:[11.11.11.11]/224, version 24
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn

Advertised path-id 1
Path type: local, path is valid, is best path, no labeled nexthop
Gateway IP: 0.0.0.0
AS-Path: 150 , path sourced external to AS
```

```
  3.3.3.3 (metric 0) from 0.0.0.0 (3.3.3.3)
    Origin incomplete, MED 0, localpref 100, weight 0
    Received label 3003003
    Extcommunity: RT:1:3003003 ENCAP:8 Router MAC:5254.006a.435b
    Originator: 1.1.1.1 Cluster list: 101.101.101.101

Path-id 1 advertised to peers:
101.101.101.101


bl1(config-router-neighbor-af)# show ip route 11.11.11.11
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

11.11.11.11/32, ubest/mbest: 1/0
*via 2.2.2.2, [200/0], 00:02:37, bgp-1, internal, tag 150 (evpn) segid: 3003
002 tunnelid: 0x2020202 encap: VXLAN
```

ルート マップを使用してネクスト ホップと RMAC 値が設定されると、以前に 1.1.1.1 を介して転送されたトラフィックは、2.2.2.2 を介して転送されます。

テーブル マップの設定

テーブル マップを設定および適用するには、次のタスクを実行します。

- MAC リストおよび MAC リストと一致するルート マップの設定 （487 ページ）
- テーブル マップの適用 （488 ページ）

MAC リストおよび MAC リストと一致するルート マップの設定

手順の概要

1. **configure terminal**
2. **mac-list** *list-name* **[seq** *seq-number***] {deny | permit}** *mac-address* **[mac-mask]**
3. **route-map** *route-map-name*
4. **match mac-list** *mac-list-name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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。

	コマンドまたはアクション	目的
ステップ 2	**mac-list** *list-name* **[seq** *seq-number***] {deny | permit}** *mac-address* **[mac-mask]** 例： `switch(config)# mac-list MAC_LIST_1 permit E:E:E`	MAC リストを構築します。
ステップ 3	**route-map** *route-map-name* 例： `switch(config)# route-map ROUTE_MAP_1`	ルート マップを作成します。
ステップ 4	**match mac-list** *mac-list-name* 例： `switch(config-route-map)# match mac-list MAC_LIST_1`	MAC リストのエントリを照合します。最大で 63 文字です。

テーブル マップの適用

手順の概要

1. **configure terminal**
2. **evpn**
3. **vni** *vni-id* **l2**
4. **table-map** *route-map-name* **[filter]**

手順の詳細

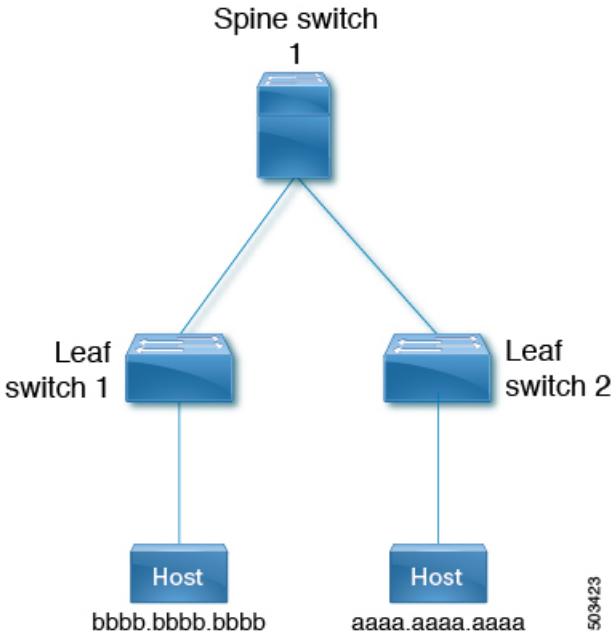
手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**evpn** 例： `switch(config)# evpn`	EVPN 設定モードを開始します。
ステップ 3	**vni** *vni-id* **l2** 例： `switch(config-evpn)# vni 101 l2`	イーサネット VPN ID を設定します。*vni-range* の範囲は 1 ～ 16,777,214 です。
ステップ 4	**table-map** *route-map-name* **[filter]** 例：	EVPN VNI 設定レベルでテーブル マップを適用します。**filter** オプションが指定されている場合、ルート

	コマンドまたはアクション	目的
	`switch(config-evpn-evi)# table-map ROUTE_MAP_1 filter`	マップ検証によって拒否されたルートはL2RIBにダウンロードされません。

テーブル マップの設定例

次のテーブル マップの設定例は、MACルートaaaa.aaaa.aaaa が L2RIB にダウンロードされないようにフィルタリングする方法を示しています。



1. 次の例は、ルートマップが適用される前の、EVPNテーブルのルートとL2RIBのMACルートの出力を示しています。

```
leaf1(config)# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 25, Local Router ID is 1.1.1.1
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 - best2

Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 1.1.1.1:32868    (L2VNI 101)
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i

Route Distinguisher: 3.3.3.3:3
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
33.33.33.33                          100          0 i

Route Distinguisher: 3.3.3.3:32868
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i

Route Distinguisher: 1.1.1.1:3    (L3VNI 100)
```

```
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
33.33.33.33                       100          0 i
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                       100          0 i
*>l[5]:[0]:[0]:[24]:[10.0.0.0]/224
1.1.1.1                  0        100      32768 ?
*>l[5]:[0]:[0]:[24]:[100.0.0.0]/224
1.1.1.1                  0        100      32768 ?

leaf1(config)# show l2route evpn mac all

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv (AD):Auto-Delete (D):Del Pending
(S):Stale (C):Clear, (Ps):Peer Sync (O):Re-Originated (Nho):NH-Override
(Pf):Permanently-Frozen, (Orp): Orphan

Topology  Mac Address     Prod  Flags  Seq No  Next-Hops
--------- ------------    ----  -----  ------- ----------
100       52fc.d83a.1b08  VXLAN  Rmac  0       33.33.33.33
101       aaaa.aaaa.aaaa  BGP    Spl   0       33.33.33.33 (Label: 101)

leaf1(config-evpn-evi)# show mac address-table vlan 101
Legend: * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False, C - ControlPlane MAC, ~ - vsanVLAN     MAC Address
Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------
C  101     aaaa.aaaa.aaaa   dynamic  0         F      F    nve1(33.33.33.33)
G  101     521d.7cef.1b08   static   -         F      F    sup-eth1(R)
```

2. 次に、MAC ルートをフィルタするようにルート マップを設定する例を示します。

```
leaf1(config)# show run rpm

!Command: show running-config rpm
!Running configuration last done at: Thu Sep  3 21:47:48 2020
!Time: Thu Sep  3 22:27:57 2020

version 9.4(1) Bios:version
mac-list FILTER_MAC_AAA seq 5 deny aaaa.aaaa.aaaa ffff.ffff.ffff
route-map TABLE_MAP_FILTER permit 10
  match mac-list FILTER_MAC_AAA
```

3. 次に、BGP EVPN レベルでルート マップを適用する例を示します。

```
leaf1(config-evpn-evi)# show run bgp | section evpn
evpn
    vni 101 l2
      table-map TABLE_MAP_FILTER filter
      rd auto
      route-target import auto
      route-target export auto
      route-target both auto evpn
```

4. 次の例は、テーブルマップが設定された後のEVPNテーブルのルートとL2RIBのMACルートの出力を示しています。

```
leaf1(config-evpn-evi)# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 26, Local Router ID is 1.1.1.1
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
```

```
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 - best2
Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 1.1.1.1:32868    (L2VNI 101)
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i

Route Distinguisher: 3.3.3.3:3
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
33.33.33.33                          100          0 i

Route Distinguisher: 3.3.3.3:32868
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i

Route Distinguisher: 1.1.1.1:3    (L3VNI 100)
*>i[2]:[0]:[0]:[48]:[52fc.d83a.1b08]:[0]:[0.0.0.0]/216
33.33.33.33                          100          0 i
*>i[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/272
33.33.33.33                          100          0 i
*>l[5]:[0]:[0]:[24]:[10.0.0.0]/224
1.1.1.1                  0           100      32768 ?
*>l[5]:[0]:[0]:[24]:[100.0.0.0]/224
1.1.1.1                  0           100      32768 ?

leaf1(config-evpn-evi)# show l2route evpn mac all

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv (AD):Auto-Delete (D):Del Pending
(S):Stale (C):Clear, (Ps):Peer Sync (O):Re-Originated (Nho):NH-Override
(Pf):Permanently-Frozen, (Orp): Orphan

Topology    Mac Address    Prod   Flags  Seq No  Next-Hops
----------- -------------- ------ ------ ------- ------------
100         52fc.d83a.1b08 VXLAN  Rmac   0       33.33.33.33

leaf1(config-evpn-evi)# show mac address-table vlan 101
Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False, C - ControlPlane MAC, ~ - vsan
VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------
G  101     521d.7cef.1b08   static   -         F      F    sup-eth1(R)
```

BGP EVPN フィルタリングの確認

BGP EVPN フィルタリング設定のステータスを表示するには、次のコマンドを入力します。

表 *12 : BGP EVPN* フィルタリングの表示

コマンド	目的
show mac-list	MAC リストを表示します。
show route-map *name*	ルート マップの情報を表示します。

コマンド	目的
show running-config bgp	BGP の設定を表示します。
show running-config rpm	すべてのルート ポリシー マネージャー（RPM）情報を表示します。
show bgp l2vpn evpn	BRIB のルートを表示します。

show mac-list コマンドの例：

```
switch(config)# show mac-list
mac-list list1: 5 entries
   seq 5 deny 0000.836d.f8b7 ffff.ffff.ffff
   seq 6 deny 0000.836d.f8b5 ffff.ffff.ffff
   seq 7 permit 0000.0422.6811 ffff.ffff.ffff
   seq 8 deny 0000.836d.f8b1 ffff.ffff.ffff
   seq 10 permit 0000.0000.0000 0000.0000.0000
mac-list list2: 3 entries
   seq 5 deny 0000.836e.f8b6 ffff.ffff.ffff
   seq 8 deny 0000.0421.6818 ffff.ffff.ffff
   seq 10 permit 0000.0000.0000 0000.0000.0000
mac-list list3: 2 entries
   seq 5 deny 0000.836d.f8b6 ffff.ffff.ffff
   seq 10 permit 0000.836d.f8b7 ffff.ffff.ffff
```

show route-map コマンドの例：

```
switch# show route-map pol10
route-map pol10, permit, sequence 10
  Match clauses:
    mac-list: list2
  Set clauses:
    ip next-hop 6.6.6.1 3.3.3.10
    ipv6 next-hop 303:304::1
```

show running-config bgp コマンドの例：

```
switch# show running-config bgp | beg "5000"
vni 5000 l2
table-map pol1 filter
rd auto
route-target import auto
route-target export auto
vni 5001 l2
rd auto
route-target import auto
route-target export auto
```

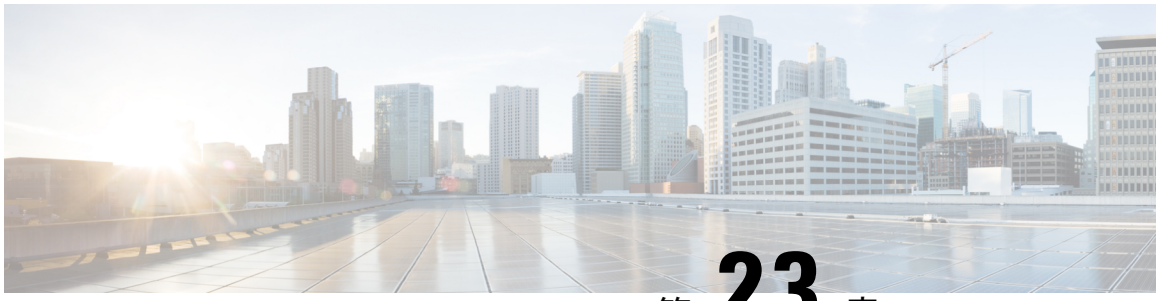
show running-config rpm コマンドの例：

```
switch# show running-config rpm
!Running configuration last done at: Thu May 23 13:58:31 2019
!Time: Thu May 23 13:58:47 2019

version 9.3(1) Bios:version 07.65
feature pbr

mac-list list1 seq 5 permit 0001.0001.0001 ffff.ffff.ffff
```

```
mac-list mclist seq 5 permit 0001.0001.0001 ffff.ffff.ffff
route-map test permit 10
match evpn route-type 5
set evpn gateway-ip 1.1.1.2
```

EVPN ルート aaaa.aaaa.aaaa に関する詳細情報を表示する **show bgp l2vpn evpn aaaa.aaaa.aaaa** コマンドの例

```
switch(config-evpn-evi)# show bgp l2 e aaaa.aaaa.aaaa

BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1.1.1.1:32868 (L2VNI 101)
BGP routing table entry for [2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:[101.0.0.3]/2
72, version 11
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, table-ma
p filtered, is not in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, remote nh not installed, no
labeled nexthop
Imported from 3.3.3.3:32868:[2]:[0]:[0]:[48]:[aaaa.aaaa.aaaa]:[32]:
[101.0.0.3]/272
AS-Path: NONE, path sourced internal to AS
33.33.33.33 (metric 81) from 101.101.101.101 (101.101.101.101)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101 100
Extcommunity: RT:100:100 RT:100:101 SOO:33.33.33.33:0 ENCAP:8
Router MAC:5254.009b.4275
Originator: 3.3.3.3 Cluster list: 101.101.101.101

Path-id 1 not advertised to any peer
```



第 23 章

VXLAN BGP-EVPN Null ルートの構成

この章は、次の内容で構成されています。

EVPN null ルートについて

EVPN ファブリック内のホストに対する分散型サービス拒否（DDoS）攻撃は、ネットワーク帯域幅技術情報を消費し、他のホストへの正当なトラフィックに影響を与えます。

DDoS 攻撃は、次の設定のいずれかから発生する可能性があります：

- ローカル サイト内のリーフ スイッチに接続されたホスト
- リモート サイトのリーフ スイッチに接続されたホスト
- WAN などの外部ネットワーク

DDoS 攻撃は、サブネット内（MAC ベース）またはサブネット間（ホスト ベース – IPv4/IPv6）の可能性があります。

null ルート フィルタ処理は、特にサービス プロバイダ ネットワークで DDoS 攻撃を軽減するために伝統的に使用されてきました。

null ルートは、どこへも到達しないネットワーク ルート（ルーティング テーブル エントリ）です。このルートに一致したパケットは、転送されるのではなくドロップ（無視またはリダイレクト）されるので、このルートは一種の制限付きファイアウォールとして機能します。null ルートを使用する行為は、多くの場合、null ルート フィルタリングと呼ばれます。

NX-OS には、IPv4/IPv6/MAC の null/drop ルートを構成するメカニズムがすでにあります。null ルートは、ファブリック内のすべての VTEP で構成する必要があります。

IPv4/IPv6 ベースの攻撃の場合、次のコマンドを使用して、null インターフェイスで IPv4/IPv6 スタティック ルートを構成します：

- **ip route x.x.x.x/y Null0**
- **ipv6 route X:X:X::X/Y Null0**

MAC ベースの攻撃の場合、次のコマンドを使用して、パケットをドロップするように drop 隣接関係を持つ MAC アドレスを構成します。

- **mac address-table static xxxx.yyyy.zzzz vlan <VLAN-ID> drop**

多数の VTEP があり、複数のサイトにまたがるファブリックでは、Nexus Dashboard Fabric Controller（NDFC）や他の Orchestrator がない場合、すべての VTEP に手動で drop ルートを構成および管理するのは困難な作業です。

EVPN null ルーティング機能は、NDFC やその他のオーケストレータなど、中央の場所から null ルートを構成して挿入する方法がない場合に使用されます。

EVPN null ルーティング機能により、ネットワーク内の VTEP は、特定のコミュニティでタグ付けされたタイプ 2 およびタイプ 5 ルートを送信できます。

シングルサイトおよびマルチサイトの他のVTEP（ボーダーおよびリーフ）は、MACまたはIP（IPv4/IPv6）テーブルでエントリをインストールすることが可能で、個別にMACまたはIP宛てのトラフィックはエッジまたはリーフスイッチでドロップされ、サイト内およびサイト全体の帯域幅の使用を防止します。

プログラムされた null ルート エントリは、ホスト IP（/32 または /128）、プレフィックス（VLSM）、または MAC です。

VXLAN BGP-EVPN null ルートの注意事項および制限事項

- null ルート（静的）MAC 構成には、一致する静的 ARP/ND 構成が必要です。つまり、MAC が null ルート MAC として構成されているダイナミック ARP/ND を使用してはなりません。
- L2 サービスのみを使用している場合（かつダイナミック ARP/ND 学習につながる構成がない場合）、「MAC ドロップ」構成のみが許可されます。他のすべての場合、「MAC ドロップ」構成とともに静的 ARP/ND 構成も必要になります。
- vPC の場合、null ルート（MAC、mac-ip、プレフィックス）を両方の vPC ボックス（VMCT と PMCT）で構成する必要があります。これが両方のボックスで構成されていない場合、動作は未定義です。同じことが、null ルートの構成解除中にも当てはまります。この機能の vPC 整合性チェッカーはサポートされていません。
- ルートマップは、リモート VTEP に適用する必要があります。この入力ルートマップは、タイプ 5 ルートにとって重要です。

- マルチキャスト トラフィックとの機能の相互作用はありません。
- VTEP でリモート スタティックが表示され、同じ MAC をローカル スタティック（有効なインターフェイスを持つ静的MAC、またはドロップ/null ルートMACに設定されたMAC）として構成する場合、修正する必要があるファブリックで構成の重複について警告する syslog が生成されます。ただし、構成は拒否されません。ローカルの静的構成は、その VTEP のリモートのスタティック構成よりも優先されます。
- 有効なインターフェイスを持つローカル静的 MAC が VTEP に構成されており、この静的 MAC を同じ VTEP 上の null ルートMAC に変換する場合、null ルート MAC が有効になります。
- リモート ダイナミック MAC ルートは、MAC-IP ルート スプリットから派生したリモート MAC ルートがエントリを上書きして MAC マネージャに伝播することを許可しますが、リモート静的 MAC ルートは、これらの派生 MAC によるエントリの上書きを尊重しません。その結果、リモート スタティック MAC が削除されるまで、MAC エントリは変更されません。
- null ルート MAC は、静的 MAC 構成のみの別の形式です。

スタティック MAC の構成

始める前に

スタティック ドロップ MAC アドレスを構成できます。これらのスタティック MAC アドレスは、インターフェイス上でダイナミックに学習された MAC アドレスを書き換えます。

手順の概要

1. **configure terminal**
2. **mac address-table static** *mac-address* **vlan** *vlan-id* {[**drop**| **interface**{*type slot/port*} | **port-channel** *number*]}
3. **exit**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル構成モードを開始します。

	コマンドまたはアクション	目的

ステップ 2	**mac address-table static** *mac-address* **vlan** *vlan-id* {[**drop**| **interface**{*type slot/port*} | **port-channel** *number*]}
	例：
	`switch(config)# mac address-table static 3001.3010.99aa vlan 3001 drop`
	`switch(config)#`
		レイヤ 2 MAC アドレス テーブルに追加するスタティック MAC アドレスを指定します。

ステップ 3	**exit**
	例：
	`switch# exit`
	`switch#`
		コンフィギュレーション モードを終了します。

ARP/ND の構成

対応する SVI の IPv4/IPv6 ルートで ARP/ND ホストを構成できます。

始める前に

MAC がドロップ エントリとして構成されているスイッチで静的 MAC-IP 構成を構成してください。これにより、MAC-IP モビリティが回避され、DROP MAC と MAC-IP の両方が同じ VTEP から発信されます。

手順の概要

1. **configure terminal**
2. **interface** *vlan-number*
3. **vrf member** *vrf-name*
4. **no ip redirects**
5. **ip address** アドレスを取得
6. **ipv6 address** アドレスを取得
7. **ipv6 neighbor address** *ipv6address mac_addr*
8. **no ipv6 redirects**
9. **ip arp address** *ipaddr mac_addr*
10. **fabric forwarding mode anycast-gateway**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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。

	コマンドまたはアクション	目的
	`switch# configure terminal` `switch(config)#`	

ステップ 2	**interface** *vlan-number*
	例：
	`switch(config)# interface Vlan 3001`
	`switch(config-if)#`
		VLAN インターフェイスを指定します。

ステップ 3	**vrf member** *vrf-name*
	例：
	`switch(config-if)# vrf member cgw_3001_3050`
	`switch(config-if)#`
		VLAN インターフェイスをテナント VRF に割り当てます。

ステップ 4	**no ip redirects**
	例：
	`switch(config-if)# no ip redirects`
	`switch(config-if)#`
		IPv4 リダイレクトを無効にします。

ステップ 5	**ip address** アドレスを取得
	例：
	`switch(config-if)# ip address 30.1.0.1/16`
	`switch(config-if)#`
		IP アドレスを指定します。

ステップ 6	**ipv6 address** アドレスを取得
	例：
	`switch(config-if)# ipv6 address 2001:3001::1/64`
	`switch(config-if)#`
		IPv6 アドレスを指定します。

ステップ 7	**ipv6 neighbor address** *ipv6address mac_addr*
	例：
	`switch(config-if)# ipv6 neighbor 2001:3001::99 3001.3010.99aa`
	`switch(config-if)#`
		静的 IPv6 ネイバーを構成します。

ステップ 8	**no ipv6 redirects**
	例：
	`switch(config-if)# no ipv6 redirects`
	`switch(config-if)#`
		IPv6 リダイレクトを無効にします。

ステップ 9	**ip arp address** *ipaddr mac_addr*
	例：
	`switch(config-if)# ip arp 30.1.0.99 3001.3010.99aa`
	`switch(config-if)#`
		IP アドレスを MAC アドレスにスタティック エントリとして関連付けます。

ステップ 10	**fabric forwarding mode anycast-gateway** 例：	VLAN 構成モードでエニーキャスト ゲートウェイと SVI を関連付けます。

	コマンドまたはアクション	目的
	`switch# fabric forwarding mode anycast-gateway` `switch#`	

ローカル VTEP でのプレフィックスヌル ルートの構成

Null ルートが構成されているローカル VTEP で、ルート マップを構成して、静的 ルートにブラックホール コミュニティを設定し、BGP に再配布します。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **ip route** {<*ip*>/*mask*} **Null0 tag** <*tag-number*> or **ip route** {<*ipv6*>/*mask*} **Null0 tag** <*tag-number*>
4. **route-map** *map-name* [**permit** | **deny**] [*seq*]
5. **match tag** <*tag-number*>
6. **set weight** *value*
7. **set community blackhole**
8. **router bgp** *as-number*
9. **vrf** *vrf-name*
10. **address-family ipv4/ipv6 unicast**
11. **redistribute static route-map** *route-map name*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**vrf context** *vrf-name*
	例：
	`switch(config)# vrf context tenant-0001`
	`switch(config-vrf)#`
		テナント VRF を構成します。

ステップ 3	**ip route** {<*ip*>/*mask*} **Null0 tag** <*tag-number*> or **ip route** {<*ipv6*>/*mask*} **Null0 tag** <*tag-number*> 例：	Null0 ネクストホップと一致するタグを使用して、接続先プレフィックスのスタティック ルートを設定します。

	コマンドまたはアクション	目的

	インターネット ユーザに商品やサービスを提供する IPv4
	`switch(config-vrf)# ip route 50.1.0.0/24 Null0 tag 6666`
	`switch(config-vrf)#`
	IPv6 の場合
	`switch(config-vrf)# ipv6 route 50::1:0/120 Null0 tag 6666`
	`switch(config-vrf)#`	

ステップ 4	**route-map** *map-name* [**permit** | **deny**] [*seq*]
	例：
	`switch(config)# route-map SET_BHC permit 10`
	`switch(config-route-map)#`
		ルート マップを作成するか、または既存のルート マップに対応するルート マップ設定モードを開始します。seq を使用して、ルート マップ エントリを順序付けます。

ステップ 5	**match tag** <*tag-number*>
	例：
	`switch(config-route-map)# match tag 6666`
	`switch(config-route-map)#`
		構成されたタグを持つルートを照合します。

ステップ 6	**set weight** *value*
	例：
	`switch (config-route-map)# set weight 65535`
	`switch(config-route-map)#`
		ブラックホール コミュニティのある着信ルートの重みを設定します。**セット ウェイト**値を最大値に設定して、nullルートに最高の優先順位を与えることをお勧めします。**設定ウェイト**の最大値は65535です。

ステップ 7	**set community blackhole**
	例：
	`switch(config-route-map)# set community blackhole`
	`switch(config-route-map)#`
		コミュニティを Blackhole (well-known community) として設定します。

ステップ 8	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 100`
	`switch(config-router)#`
		ルーティング プロセスをイネーブルにします。as-num の範囲は 1–65535 です。

ステップ 9	**vrf** *vrf-name*
	例：
	`switch(config-router)# vrf tenant-0001`
	`switch(config-router-vrf)#`
		テナント VRF を構成します。

ステップ 10	**address-family ipv4/ipv6 unicast**
	例：
	`switch(config-router-vrf)# address-family ipv4 unicast`
	`switch(config-router-vrf-af)#`
		IPv4/IPv6 アドレス ファミリを構成。この構成は、IPv4/IPv6 アンダーレイを使用した IPv4/IPv6 over VXLAN に必要です。

	コマンドまたはアクション	目的

ステップ 11	**redistribute static route-map** *route-map name*
	例：
	`switch(config-router-vrf-af)# redistribute static route-map SET_BHC`
	`switch(config-router-vrf-af)#`
		構成されたルート マップを使用して、prefix-null 静的 ルートを BGP に再配布します。

リモート VTEP での RPM ルート マップの構成

始める前に

リモート VTEP では、コミュニティ リストとルート マップを使用して null ルートに優先順位を付けます。

手順の概要

1. **configure terminal**
2. **ip community-list standard** <*community-list-name*> **seq** <*seq-number*> **permit blackhole**
3. **route-map** *map-name***[permit | deny]** <*seq-number*>
4. **match community** <*community-list*>
5. **set weight** *value*
6. **route-map** *map-name***permit** <*seq-number*>
7. **router bgp** *as-number*
8. **route-map** *route-map* {**in** | **out**}

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル構成モードを開始します。

ステップ 2	**ip community-list standard** <*community-list-name*> **seq** <*seq-number*> **permit blackhole**
	例：
	`switch (config)# ip community-list standard BH seq 10 permit blackhole`
	`switch(config)#`
		コミュニティ リストを設定し、よく知られた「ブラックホール」コミュニティ値を持つルートを許可します。
		Cisco NX-OS リリース 10.3（2）F 以降、ブラックホール（既知のコミュニティ）が既存の IP コミュニティ リストに追加されています。

	コマンドまたはアクション	目的

ステップ 3	**route-map** *map-name***[permit | deny]** <*seq-number*>
	例：
	`switch(config)# route-map PREFER_BHC permit 10`
	`switch(config-route-map)#`
		ルートマップ構成モードを開始します。

ステップ 4	**match community** <*community-list*>
	例：
	`switch(config-route-map)# match community BH`
	`switch(config-route-map)#`
		BGP ルートは、コミュニティ リストを使用して照合されます。

ステップ 5	**set weight** *value*
	例：
	`switch (config-route-map)# set weight 65535`
	`switch(config-route-map)#`
		ブラックホール コミュニティのある着信ルートの重みを設定します。**セット ウェイト**値を最大値に設定して、null ルートに最高の優先順位を与えることをお勧めします。**設定ウェイト**の最大値は 65535 です。

ステップ 6	**route-map** *map-name***permit** <*seq-number*>
	例：
	`switch(config-route-map)# route-map PREFER_BHC permit 20`
	`switch(config-route-map)#`
		他のルートを許可するフォールバック許可句を使用してルート マップを構成します。

ステップ 7	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 100`
	`switch(config-router)#`
		ルーティング プロセスをイネーブルにします。as-num の範囲は 1 ～ 65535 です。

ステップ 8	**route-map** *route-map* {**in** | **out**} 例： `switch(config-router-neighbor-af)# route-map PREFER_BHC in`	構成された方向のネイバーにルート マップを適用します。

Null ルートの構成例

次の例は、プレフィックスヌルおよびMAC/MAC-IP ドロップ ルートにローカル/リモート構成を構成する方法を示しています：

構成 – プレフィックス Null

Type5 null ルートがアドバタイズされるローカル VTEP（ボーダー リーフ スイッチ）で、次の手順を実行します。

1. Null0 隣接で静的 IPv4/IPv6 アドレスを構成する

```
vrf context tenant-0001
  vni 3100001
```

```
  ip route 50.1.0.0/24 Null0 tag 6666
  ipv6 route 50::1:0/120 Null0 tag 6666
```

2. スタティック ルートに null ルート コミュニティを設定し、BGP に再配布するようにルート マップを構成します

```
route-map SET_BHC permit 10
  match tag 6666
  set community blackhole
router bgp 100
  router-id 10.1.0.21
  vrf tenant-0001
    address-family ipv4 unicast
      redistribute static route-map SET_BHC
    address-family ipv6 unicast
      redistribute static route-map SET_BHC
```

他のすべてのリモート VTEP で、次の手順を実行します。

1. null ルート コミュニティに一致するようにルート マップを構成し、重みを最大値に設定して、null ルートが常に優先されるようにします。

```
ip community-list standard BH seq 10 permit blackhole
route-map PREFER_BHC permit 10
  match community BH
  set weight 65535
route-map PREFER_BHC permit 20
router bgp 100
 router-id 10.1.0.13
 address-family l2vpn evpn
 template peer LEAF_to_FABRIC_IBGP_OVERLAY
    remote-as 100
    address-family l2vpn evpn
      send-community
      send-community extended
      route-map PREFER_BHC in
```

構成 – MAC/MAC-IP ドロップ

Type2 null ルートがアドバタイズされるローカル VTEP で、次の手順を実行します。

1. ドロップ隣接を使用して静的 MAC アドレスを構成します

```
mac address-table static 0013.e001.0001 vlan 2 drop
```

2. 同じアドレスの静的 ARP/ND ネイバーを構成する

```
interface Vlan2
  no shutdown
  vrf member tenant-0001
  ip address 5.0.63.254/18
  ipv6 address 5::3f7f/114
  ipv6 neighbor 5::17fe 0013.e001.0001
  no ipv6 redirects
  ip arp 5.0.23.254 0013.e001.0001
  fabric forwarding mode anycast-gateway
```

他のすべてのリモート VTEP で、次の手順を実行します：

1. ブラックホール コミュニティに一致するようにルート マップを構成し、重みを最大値に設定して、null ルートが常に優先されるようにします。

```
ip community-list standard BH seq 10 permit blackhole
route-map PREFER_BHC permit 10
  match community BH
  set weight 65535
route-map PREFER_BHC permit 20
router bgp 100
 router-id 10.1.0.13
 address-family l2vpn evpn
 template peer LEAF_to_FABRIC_IBGP_OVERLAY
    remote-as 100
    address-family l2vpn evpn
      send-community
      send-community extended
      route-map PREFER_BHC in
  neighbor 10.1.0.31
    inherit peer LEAF_to_FABRIC_IBGP_OVERLAY
```

EVPN Null ルート構成の確認

EVPN Null ルート構成情報を表示するには、次のいずれかのコマンドを入力します。

コマンド	目的
show bgp l2vpn evpn	ルーティング テーブルの情報を表示します。
show ip arp static vlan <*vlan-id*> **vrf** <*vrf-name*>	ローかつ ARP 情報を表示します。
<vlan-id> <vrf-name>　show ip arp static remote vlanvrf	リモート ARP 情報を表示します。
<vlan-id> <vrf-name>　show ip adjacency vlandetail vrf	ローカル隣接関係情報を表示します。
show ipv6 icmp neighbour static remote [*vlan* <*id*>] [*vrf* <*name*>]	リモートスタティック隣接情報を表示します。
show mac address-table static vlan <*vlan-id*>	ローカル/リモート MAC 情報を表示します。
show ip community-list *name*	IP コミュニティ リストに関する情報を表示します。
show route-map *name*	ルート マップの情報を表示します。

次の例では、**show bgp l2vpn evpn** コマンドの Type-2 EVPN ルート サンプル出力を表示します。

```
switch# show bgp l2vpn evpn 1111.1111.1111
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 53.53.53.53:32769    (L2VNI 1000002)
BGP routing table entry for [2]:[0]:[0]:[48]:[1111.1111.1111]:[32]:[100.100.100.51]/272,
 version 23
Paths: (1 available, best #1)
Flags: (0x000102) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP
  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop, has esi_gw
  AS-Path: NONE, path locally originated
```

```
53.53.53.53 (metric 0) from 0.0.0.0 (53.53.53.53)
Origin IGP, MED not set, localpref 100, weight 32768
Received label 1000002 1000100
Community: Blackhole
Extcommunity: RT:23456:1000002 RT:23456:1000100 ENCAP:8
Router MAC:0476.b0f0.8157
Path-id 1 advertised to peers:
111.111.54.1
```

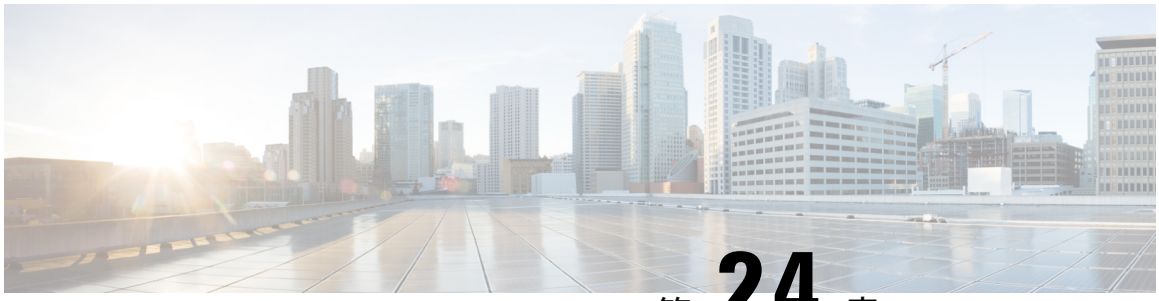
次の例では、**show bgp l2vpn evpn** コマンドの Type-5 EVPN ルート（送信）サンプル出力を表示します。

```
switch# sh bgp ipv4 uni 44.44.44.0 vrf 100
BGP routing table information for VRF 100, address family IPv4 Unicast
BGP routing table entry for 44.44.44.0/24, version 6
Paths: (1 available, best #1)
Flags: (0x80c0002) (high32 0x000020) on xmit-list, is not in urib, exported, has label
vpn: version 5, (0x00000000100002) on xmit-list
local label: 492287

Advertised path-id 1, VPN AF advertised path-id 1
Path type: redist, path is valid, is best path, no labeled nexthop, is extd
AS-Path: NONE, path locally originated
0.0.0.0 (metric 0) from 0.0.0.0 (44.44.44.44)
Origin incomplete, MED 0, localpref 100, weight 32768
Community: blackhole
Extcommunity: RT:23456:1000100

VRF advertise information:
Path-id 1 not advertised to any peer

VPN AF advertise information:
Path-id 1 not advertised to any peer

switch# sh bgp l2 e 44.44.44.0
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 53.53.53.53:4 (L3VNI 1000100)
BGP routing table entry for [5]:[0]:[0]:[24]:[44.44.44.0]/224, version 5
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP

Advertised path-id 1
Path type: local, path is valid, is best path, no labeled nexthop, has esi_gw
Gateway IP: 0.0.0.0
AS-Path: NONE, path locally originated
53.53.53.53 (metric 0) from 0.0.0.0 (53.53.53.53)
Origin incomplete, MED 0, localpref 100, weight 32768
Received label 1000100
Community: blackhole
Extcommunity: RT:23456:1000100 ENCAP:8 Router MAC:0476.b0f0.8157

Path-id 1 advertised to peers:
111.111.54.1
```

次の例では、**show bgp l2vpn evpn** コマンドの Type-5 EVPN ルート（受信）サンプル出力を表示します。

```
switch# sh bgp l2 e 44.44.44.0
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 53.53.53.53:4
BGP routing table entry for [5]:[0]:[0]:[24]:[44.44.44.0]/224, version 2
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
```

```
Multipath: eBGP iBGP

Advertised path-id 1
Path type: external, path is valid, is best path, no labeled nexthop, has esi_gw
Imported to 2 destination(s)
Imported paths list: 100 L3-1000100
Gateway IP: 0.0.0.0
AS-Path: 4241653625 , path sourced external to AS
53.53.53.53 (metric 2) from 111.111.53.1 (53.53.53.53)
Origin incomplete, MED 0, localpref 100, weight 0
Received label 1000100
Community: blackhole
Extcommunity: RT:11000:1000100 Route-Import:53.53.53.53:100
Source AS:4241653625:0 SOO:50529024:00000000 ENCAP:8
Router MAC:0476.b0f0.8157
Path-id 1 not advertised to any peer

switch# show bgp ipv4 uni 44.44.44.0 vrf 100
BGP routing table information for VRF 100, address family IPv4 Unicast
BGP routing table entry for 44.44.44.0/24, version 3
Paths: (1 available, best #1)
Flags: (0x8008001a) (high32 00000000) on xmit-list, is in urib, is best urib route, is
in HW
vpn: version 3, (0x00000000100002) on xmit-list

Advertised path-id 1, VPN AF advertised path-id 1
Path type: external, path is valid, is best path, no labeled nexthop, in rib, has esi_gw

Imported from 53.53.53.53:4:[5]:[0]:[0]:[24]:[44.44.44.0]/224
AS-Path: 4241653625 , path sourced external to AS
53.53.53.53 (metric 2) from 111.111.53.1 (53.53.53.53)
Origin incomplete, MED 0, localpref 100, weight 0
Received label 1000100
Community: blackhole
Extcommunity: RT:11000:1000100 Route-Import:53.53.53.53:100
Source AS:4241653625:0 SOO:50529024:00000000 ENCAP:8
Router MAC:0476.b0f0.8157

VRF advertise information:
Path-id 1 not advertised to any peer
```



第 24 章

ポート VLAN マッピングの設定

この章は、次の内容で構成されています。

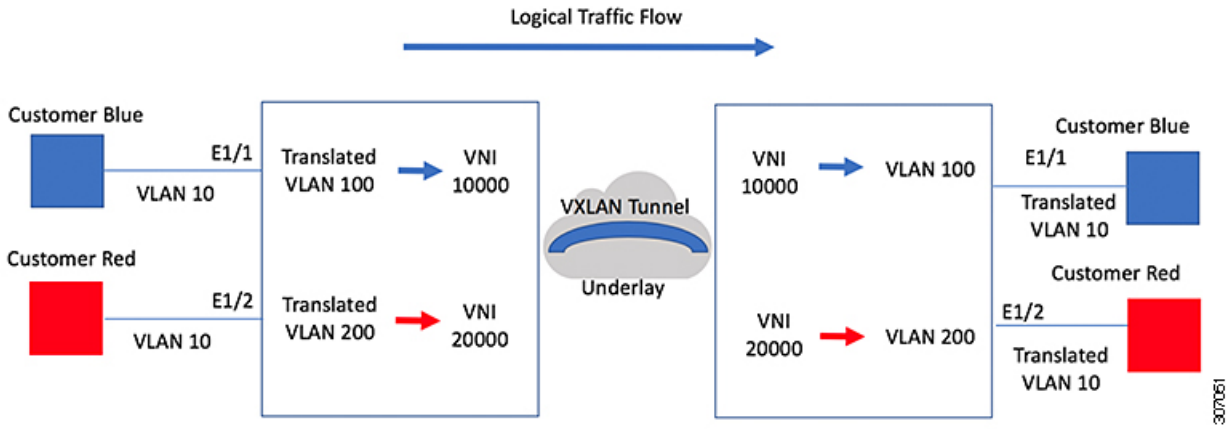
着信 VLAN の変換について

VLAN 変換が必要な場合や必要な場合があります。このような使用例の 1 つは、サービス プロバイダーが、同じ VLAN カプセル化を使用して同じ物理スイッチに接続している複数のカスタマーを持っているが、それらが同じ Layer 2 セグメント上に存在しない場合です。このような場合、着信 VLAN を一意の VLAN に変換してから VNI にマッピングするのが、セグメントを拡張する正しい方法です。次の図では、Blue と Red の両方がカプセル化として VLAN 10 を使用してリーフに接続しています。

お客様の青と赤は、同じ VNI 上に存在することはできません。この例では、Customer Blue の VLAN 10（インターフェイスE1/1）が VLAN 100 にマッピング/変換され、Customer Red の VLAN 10（インターフェイスE1/2）が VLAN 200 にマッピングされます。次に、VLAN 100 は VNI 10000 にマッピングされ、VLAN 200 は VNI 20000 にマッピングされます。

もう一方のリーフでは、このマッピングが逆に適用されます。VNI 10000 上の着信 VXLAN カプセル化トラフィックは VLAN 100 にマッピングされ、VLAN 100 はインターフェイス E1/1 の VLAN 10 にマッピングされます。VNI 20000 の VXLAN カプセル化トラフィックは VLAN 200 にマッピングされ、VLAN 200 はインターフェイス E1/2 の VLAN 10 にマッピングされます。

図 *45 :* 論理的トラフィック フロー



入力（着信）VLAN とポートにあるローカル（変換先）VLAN との間での VLAN 変換を設定できます。VLAN変換がイネーブルにされたインターフェイスに到着するトラフィックにおいて、着信 VLAN は VXLAN がイネーブルにされた変換先 VLAN にマッピングされます。

アンダーレイ上で、これは VNI にマッピングされ、内部 dot1q が削除されて、VXLAN ネットワークに切り替えられます。出力スイッチで、VNI は変換先 VLAN にマッピングされます。VLAN 変換が設定された発信インターフェイスで、トラフィックは元の VLAN に変換されてから出力されます。トラフィック カウンタについては、入力 VLAN ではなく、変換先 VLAN にある VLAN カウンタを参照してください。ポート VLAN（PV）マッピングは、アクセス側の機能であり、マルチキャストおよび入力複製の両方で VXLAN 用の BGP EVPN モードおよびフラッディングと学習がサポートされています。

ポート VLAN マッピングに関する注意事項と制限事項：

次に、ポート VLAN マッピングに関する注意事項と制限事項を示します。

- vPCファブリック ピアリングのサポートが追加されました。
- Cisco NX-OS リリース 10.3(3)F 以降、VLAN 変換は VXLAN と非 VXLAN の両方の VLAN でサポートされます。
- 入力（着信）VLAN は、スイッチで VLAN として設定する必要はありません。変換先 VLAN は設定が必要であり、vn-segment マッピングを与えておく必要があります。VNI マッピングを使用する NVE インターフェイスは、これに不可欠です。
- すべてのレイヤ 2 送信元アドレスの学習およびレイヤ 2 MAC 宛先のルックアップは、変換先 VLAN で行われます。入力（着信）VLAN ではなく、変換先 VLAN にある VLAN カウンタを参照してください。
- ポート VLAN マッピングは、Cisco Nexus 9300、9300-EX、および 9300-FX3 プラットフォーム スイッチでサポートされます。

- Cisco Nexus 9300 および 9500 スイッチは、オーバーラップ VLAN インターフェイスでのスイッチングとルーティングをサポートします。Cisco Nexus 9300-EX/FX/FX2/FX3 プラットフォームスイッチおよび -EX/FX ライン カードを備えた Cisco Nexus 9500 には、VLAN マッピングスイッチングのみが適用されます。
- ポート VLAN ルーティングは、次のプラットフォームでサポートされます。
  - Cisco NX-OS リリース 7.x 以降、この機能は Cisco Nexus 9300-EX/FX/FX2 プラットフォーム スイッチでサポートされています。
  - Cisco NX-OS リリース 9.2(x) 以降、この機能は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされています。
  - Cisco NX-OS リリース 9.3(x) 以降、この機能は Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされています。
  - Cisco NX-OS リリース 10.2(3)F 以降、この機能は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
  - Cisco NX-OS リリース 10.4(1)F 以降、この機能は Cisco Nexus 9332D-H2R スイッチでサポートされています。
  - Cisco NX-OS リリース 10.4(2)F 以降、この機能は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
  - Cisco NX-OS リリース 10.4(3)F 以降、この機能は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 9.3(3) 以降、PV 変換は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、PV 変換は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、PV 変換は Cisco Nexus 9332D-H2R スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、PV 変換は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、PV 変換は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- Cisco Nexus 9300 シリーズ スイッチでは、PV ルーティングは 40 G ポートではサポートされません。
- PV ルーティングは、変換先 VLAN での SVI 設定について、VXLAN 用の BGP EVPN モードおよびフラッディングと学習をサポートしています。
- VLAN 変換（マッピング）は、ネットワーク フォワーディング エンジン（NFE）を搭載した Cisco Nexus 9000 シリーズ スイッチでサポートされます。

- 変換先 VLAN のプロパティを変更する場合、当該 VLAN を変換先 VLAN として設定するマッピングのあるポートで、フラッピングをして正しい動作をしているか確認する必要があります。これは、次のプラットフォームにのみ適用されます。
  - N9K-C9504 モジュール
  - N9K-C9508 モジュール
  - N9K-C9516 モジュール
  - Nexus 9400 ライン カード
  - Nexus 9500 ライン カード
  - Nexus 9600 ライン カード
  - Nexus 9700-X クラウド スケール ライン カード
  - Nexus 9600-R および R2 ライン カード

```
Int eth 1/1
switchport vlan mapping 101 10
.
.
.

/***Deleting vn-segment from vlan 10.***/
/***Adding vn-segment back.***/
/***Flap Eth 1/1 to ensure correct behavior.***/
```

- 次に、ローカル VLAN 100 にマッピングされる着信 VLAN 10 の例を示します。ローカル VLAN 100 は、VXLAN VNI にマッピングされます。

```
interface ethernet1/1
switchport vlan mapping 10 100
```

- 次に、PV 変換用のオーバーラップ VLAN の例を示します。最初のステートメントでは、VLAN-102 は VNI マッピングを使用して変換された VLAN です。2 番目のステートメントでは、VLAN-102 は VNI マッピングを使用して VLAN-103 に変換されます。

```
interface ethernet1/1
switchport vlan mapping 101 102
switchport vlan mapping 102 103/
```

- force コマンドを使用して既存のポート チャネルにメンバーを追加する場合、「mapping enable」設定は一貫している必要があります。次に例を示します。

```
Int po 101
switchport vlan mapping enable
switchport vlan mapping 101 10
switchport trunk allowed vlan 10

int eth 1/8
/***No configuration***/
```

（注） **switchport vlan mapping enable** コマンドは、ポート モードが trunk の場合にのみサポートされます。

- ポート VLAN マッピングは、Cisco Nexus 9200 プラットフォーム スイッチではサポートされません。
- VLANマッピングは、ポートごとにVLANをスコーピングすることで、ポートへのVLANのローカリゼーションに役立ちます。一般的な使用例は、サービスプロバイダーのリーフスイッチに、重複する VLAN を持つ異なるカスタマーがあり、異なるポートに着信するサービス プロバイダー環境です。たとえば、顧客 A には Eth 1/1 に着信する VLAN 10 があり、顧客 B には Eth 2/2 に着信する VLAN 10があります。

  このシナリオでは、カスタマー VLAN をプロバイダー VLAN にマッピングし、それをレイヤ 2 VNI にマッピングできます。さまざまなカスタマーVLANを終端し、それらをファブリック管理 VLAN、L2 VNI にマッピングすると、運用上の利点があります。
- ポート VLAN 変換が機能するには、VNI マッピングを使用する NVE インターフェイスを設定する必要があります。
- **system dot1q-tunnel transit vlan** *<id>* コマンドのプロバイダー VLAN リストでスーパー ブリッジング VLAN を有効にしないでください。有効にすると、回復不能な機能および転送への影響が発生します。
- ポート VLAN マッピングは、FEX ポートではサポートされていません。
- Cisco NX-OS リリース 10.3(3)F 以降、IPv6 アンダーレイは Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 スイッチおよび、9700-EX/FX/GX ライン カードを搭載した Cisco Nexus 9500 スイッチで、VXLAN EVPN のポート VLAN マッピングでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、IPv6 アンダーレイは Cisco Nexus 9332D-H2R スイッチ上の VXLAN EVPN のポート VLAN マッピングでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、IPv6 アンダーレイは Cisco Nexus 93400LD-H1 スイッチ上の VXLAN EVPN のポート VLAN マッピングでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、IPv6 アンダーレイは Cisco Nexus 9364C-H1 スイッチ上の VXLAN EVPN のポート VLAN マッピングでサポートされています。

トランク ポート上のポート VLAN マッピングの設定

始める前に

- VLAN 変換を実装する物理またはポート チャネルがレイヤ 2 トランク ポートとして設定されていることを確認します。
- 変換先 VLAN がスイッチで作成されており、レイヤ 2 トランク ポートのトランク許可 VLAN の vlan-list にも追加されていることを確認します。

（注） ベスト プラクティスとして、入力 VLAN ID をインターフェイスのスイッチポート許可 vlan-list に追加しないでください。

- すべての変換先 VLAN で VXLAN がイネーブルであることを確認します。

手順の概要

1. **configure terminal**
2. **interface** *type/port*
3. **[no] switchport vlan mapping enable**
4. **[no] switchport vlan mapping** *vlan-id translated-vlan-id*
5. **[no] switchport vlan mapping all**
6. **copy running-config startup-config**
7. **show interface** [*if-identifier*] **vlan mapping**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**interface** *type/port* 例： `switch(config)# interface Ethernet1/1`	設定するインターフェイスを指定します。
ステップ 3	**[no] switchport vlan mapping enable** 例： `switch(config-if)# [no] switchport vlan mapping enable`	スイッチ ポートでの VLAN 変換をイネーブルにします。VLAN 変換はデフォルトでディセーブルです。 （注） VLAN変換を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 4	**[no] switchport vlan mapping** *vlan-id translated-vlan-id* 例： `switch(config-if)# switchport vlan mapping 10 100`	VLAN を他の VLAN に変換します。 • *vlan-id* 引数と *translated-vlan-id* 引数の範囲は 1 ～ 4094 です。 • 入力（着信）VLAN とポートにあるローカル（変換先）VLAN との間での VLAN 変換を設定できます。VLAN 変換がイネーブルにされたインターフェイスに到着するトラフィックにおい

	コマンドまたはアクション	目的
		て、着信 VLAN は VXLAN がイネーブルにされた変換先 VLAN にマッピングされます。 アンダーレイ上で、これは VNI にマッピングされ、内部 dot1q が削除されて、VXLAN ネットワークに切り替えられます。出力スイッチで、VNI はローカル変換された VLAN にマッピングされます。VLAN 変換が設定された発信インターフェイスで、トラフィックは元の VLAN に変換されてから出力されます。 （注） このコマンドの **no** 形式を使用すると、VLAN ペア間のマッピングがクリアされます。
ステップ 5	**[no] switchport vlan mapping all** 例： `switch(config-if)# switchport vlan mapping all`	インターフェイスに設定されたすべての VLAN のマッピングを削除します。
ステップ 6	**copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。 （注） VLAN 変換の設定は、スイッチ ポートが動作トランク ポートになるまで有効になりません。
ステップ 7	**show interface** [*if-identifier*] **vlan mapping** 例： `switch# show interface ethernet1/1 vlan mapping`	インターフェイスの範囲または特定のインターフェイスについて、VLAN マッピング情報を表示します。

例

次に、（入力）VLAN 10 と（ローカル）VLAN 100 間で VLAN 変換を設定する例を示します。show vlan counters コマンド出力は、カスタマー VLAN ではなく変換先 VLAN として統計情報カウンタを表示します。

```
switch#  configure terminal
switch(config)#  interface ethernet1/1
switch(config-if)#  switchport vlan mapping enable
switch(config-if)#  switchport vlan mapping 10 100
switch(config-if)#  switchport trunk allowed vlan 100
switch(config-if)#  show interface ethernet1/1 vlan mapping
Interface eth1/1:
Original VLAN           Translated VLAN
------------------      ---------------
10                           100

switch(config-if)#  show vlan counters
```

```
Vlan Id                             :100
Unicast Octets In                   :292442462
Unicast Packets In                  :1950525
Multicast Octets In                 :14619624
Multicast Packets In                :91088
Broadcast Octets In                 :14619624
Broadcast Packets In                :91088
Unicast Octets Out                  :304012656
Unicast Packets Out                 :2061976
L3 Unicast Octets In                :0
L3 Unicast Packets In               :0
```

トランク ポートでの内部 VLAN および外部 VLAN マッピングの設定

トランクポートでの内部 VLAN および外部 VLAN マッピングの設定は、Cisco Nexus 9300 プラットフォームにのみ適用され、Cisco Nexus 9200、9300-EX、9300-FX、9300-FX2、9300-FX3、9300-GX、9300-GX2、9332D-H2R、93400LD-H1、9364C-H1、9364C、9332C プラットフォームではサポートされません。

内部 VLAN および外部 VLAN からポートのローカル（変換先）VLAN への VLAN 変換を設定できます。VLAN 変換がイネーブルにされたインターフェイスに着信するダブル タグ VLAN トラフィックについては、内部 VLAN および外部 VLAN が、VXLAN がイネーブルにされた変換先 VLAN にマッピングされます。

内部 VLAN および外部 VLAN マッピングに関する注意点

- 内部および外部 VLAN は、これらが設定されているポートのトランク許可リストに含めることはできません。

  次に例を示します。

```
switchport vlan mapping 11 inner 12 111
switchport trunk allowed vlan 11-12,111 /***Not valid because 11 is outer VLAN and
12 is inner VLAN.***/
```

- 同じポート上で、2 つのマッピング（変換）設定に、同じ内容の外部（あるいはオリジナル）VLAN もしくは変換先 VLAN を含めることはできません。複数の内部 VLAN および外部 VLAN のマッピング設定については、同じ内部 VLAN を含めることができます。

  次に例を示します。

```
switchport vlan mapping 101 inner 102 1001
switchport vlan mapping 101 inner 103 1002  /***Not valid because 101 is already
used as an original VLAN.***/
switchport vlan mapping 111 inner 104 1001  /***Not valid because 1001 is already
used as a translated VLAN.***/
switchport vlan mapping 106 inner 102 1003  /***Valid because inner vlan can be the
 same.***/
```

- 内部オプションでイネーブルになっているポートでパケットが二重タグ付けされた場合、ブリッジングのみがサポートされます。
- VXLAN PV ルーティングは、二重タグ付きフレームではサポートされません。

手順の概要

1. **configure terminal**
2. **interface** *type port*
3. [**no**] **switchport mode trunk**
4. **switchport vlan mapping enable**
5. **switchport vlan mapping** *outer-vlan-id* **inner** *inner-vlan-id translated-vlan-id*
6. （任意） **copy running-config startup-config**
7. （任意） **show interface** [*if-identifier*] **vlan mapping**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**interface** *type port*	インターフェイス設定モードを開始します。
ステップ 3	[**no**] **switchport mode trunk**	トランク コンフィギュレーション モードを開始します。
ステップ 4	**switchport vlan mapping enable**	スイッチ ポートでの VLAN 変換をイネーブルにします。VLAN 変換はデフォルトでディセーブルです。 （注） VLAN変換を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 5	**switchport vlan mapping** *outer-vlan-id* **inner** *inner-vlan-id translated-vlan-id*	内部 VLAN および外部 VLAN を他の VLAN に変換します。
ステップ 6	（任意） **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。 （注） スイッチ ポートが動作するトランク ポートになるまで、VLAN 変換設定は有効になりません。

	コマンドまたはアクション	目的
ステップ 7	（任意） **show interface** [*if-identifier*] **vlan mapping**	インターフェイスの範囲または特定のインターフェイスについて、VLAN マッピング情報を表示します。

例

この例では、ダブル タグ VLAN トラフィック（内部 VLAN 12、外部 VLAN 11）から VLAN 111 への変換を設定する方法を示します。

```
switch# configure terminal
switch(config)# interface ethernet1/1
switch(config-if)# switchport mode trunk
switch(config-if)# switchport vlan mapping enable
switch(config-if)# switchport vlan mapping 11 inner 12 111
switch(config-if)# switchport trunk allowed vlan 101-170
switch(config-if)# no shutdown

switch(config-if)# show mac address-table dynamic vlan 111

Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link,
        (T) - True, (F) - False
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------
*  111     0000.0092.0001   dynamic  0         F      F    nve1(100.100.100.254)
*  111     0000.0940.0001   dynamic  0         F      F    Eth1/1
```

ポート マルチ VLAN マッピングについて

ポート マルチ VLAN マッピング機能を使用すると、複数の VLAN がトランク インターフェイスで単一のグローバル VLAN/VNI にマッピングされます。レイヤ 2（L2）サブインターフェイスをマッピング用に作成し、qTag を各 L2 サブインターフェイスに提供する必要があります。

異なるポート VLAN は、同じ物理インターフェイス上で異なるサービスを提供できます。

トランク ポートごとのポート マルチ VLAN マッピングの場合、L2 サブインターフェイスを使用するマッピングごとに ACL がインストールされます。一部の ACL はデフォルトで自動的にインストールされ、一部は静的 MAC アドレス設定でインストールされます。L2 サブインターフェイスには qtag、flood-domain、または provider-VLAN があります。プロバイダー VLAN はスイッチ上で設定され、トラフィック転送に使用されます。スイッチ上に存在できるプロバイダー VLAN は 1 つだけです。

この静的 MAC 設定は、L2 サブインターフェイスの親ポートで設定された **switchport mac-address static-only** コマンドを使用して行われます。このコマンドは、親ポートの MAC ラーニングを無効にし、L2 サブインターフェイスに設定された各スタティック MAC ごとに MAC-ACL を有効にします。

ポート マルチ VLAN マッピングに関する注意事項と制限事項：

ポートマルチ VLAN マッピングの注意事項と制約事項は次のとおりです。

- Cisco NX-OS リリース 10.2(3) 以降、ポート マルチ VLAN 機能は N9K-C9316D-GX、N9K-C93600CD-GX、N9K-C9364C-GX、および Cisco Nexus 9300-GX2 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、ポート マルチ VLAN 機能は Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、ポート マルチ VLAN 機能は Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、ポート マルチ VLAN 機能は Cisco Nexus 9364C-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.1(2) 以降、ポートマルチ VLAN マッピングは Cisco Nexus 9300-EX、FX、および FX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、ポート Multi-VLAN マッピングが Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- ポート VLAN（PV）マッピングは、アクセス側の機能であり、VXLAN フラッディングと学習モードのマルチキャストおよび入力複製の両方でサポートされています。この機能は、Cisco NX-OS リリース 10.1(2) の VXLAN MP-BGP EVPN モードではサポートされません。
- Cisco Nexus リリース 10.1（2）または Cisco Nexus リリース 10.2(1)F ND-ISSU で実行されているデバイスでは、L2 サブインターフェイスが設定されている場合はサポートされません。
- この機能は、vPC ファブリック ピアリング設定ではサポートされていません。
- ブロードキャストまたはマルチキャスト フラッドから保護するために、ARP および NS/ND を除くすべてのフラッディング トラフィックがドロップされます。
- レイヤ 2 セキュリティはサポートされていません。
- STP はサポートされていません。
- ToR では、リモート VTEP へのスタティック デフォルト ルートまたは特定のルートを設定することを推奨します。
- QinQ/QinVNI、ポート VLAN マッピング、PVLAN、Xconnect などの他のアクセス機能との相互作用はサポートされていません。

次に、親インターフェイスに関する注意事項と制限事項を示します。

- TCAMエントリは、親ポートが存在するスライスにのみインストールされます。TCAM使用率を確認するには、**show system internal access-list resource utilization** コマンドを使用します。
- ポート スライスを確認するには、**show interface hardware-mappings** コマンドを使用します。
- 静的 ARP を使用するホストの場合、インターフェイス nve 1 のリモート ホストの ToR 静的 MAC エントリを追加します。例：

```
 mac address-table static 0034.0100.0001 vni 10013001 interface nve 1 peer-ip
192.168.75.2
```

- Port-security/dot1x は親インターフェイスではサポートされません。
- vPC モードは、親インターフェイスまたは L2 サブインターフェイスではサポートされません。

次に、サブインターフェイスに関する注意事項と制限事項を示します。

- スイッチごとに最大 510 のサブインターフェイスがサポートされます。
- サブインターフェイスごとの ACL およびストーム制御は、スイッチポートマッピングでは設定できません。
- 最大 510 L2 サブインターフェイスをサポートするには、TCAM リージョンを再設定する必要があります。各 L2 サブインターフェイスには、9 つの TCAM ing-pacl-sb エントリが割り当てられます。
- 静的 MAC は、親インターフェイスで **switchport mac-address static-only** コマンドを使用して L2 サブインターフェイスで設定されます。
- L2 サブインターフェイスは、VXLAN 展開なしではサポートされません。プロバイダー VLAN は VXLAN VLAN である必要があります。
- 動的 MAC ラーニングは L2 サブインターフェイスでディセーブルです。
- ストーム制御インターフェイスの統計はサポートされていません。
- **hardware profile svi-and-si flex-stats-enable** コマンドは、入力 L2 サブインターフェイス カウンタのみをサポートします。profile statistics コマンドは、出力 L2 サブインターフェイス カウンタおよび VxLAN 統計情報をサポートしません。
- IGMP スヌーピングは、L2 サブインターフェイスが設定されているプロバイダー VLAN ではサポートされません。

ポート マルチ VLAN マッピングの設定

ポート マルチ VLAN マッピングの設定例を次に示します。

```
feature ospf
feature pim
```

```
feature bfd
feature interface-vlan
feature vn-segment-vlan-based
feature private-vlan
feature lacp
feature nv overlay

hardware access-list tcam region ing-pacl-sb 2560
hardware profile svi-and-si flex-stats-enable

ip pim rp-address 2.0.0.254 group-list 224.0.0.0/4

vlan 3001
  vn-segment 10013001

interface Ethernet1/22
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 3001
  mtu 9216
  storm-control broadcast level 0.01
  storm-control action trap
  switchport isolated
  switchport mac-address static-only
  no shutdown

interface Ethernet1/22.1
  encapsulation dot1q 301 provider-vlan 3001
  no shutdown

interface Ethernet1/22.2
  encapsulation dot1q 302 provider-vlan 3001
  no shutdown

interface Ethernet1/22.3
  encapsulation dot1q 303 provider-vlan 3001
  no shutdown

interface Ethernet1/22.4
  encapsulation dot1q 304 provider-vlan 3001
  no shutdown

interface Ethernet1/22.5
  encapsulation dot1q 305 provider-vlan 3001
  no shutdown

interface port-channel1
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 3001
  mtu 9216
  storm-control broadcast level 0.01
  storm-control multicast level 0.01
  storm-control unicast level 0.01
  storm-control action trap
  switchport isolated
  switchport mac-address static-only

interface port-channel1.1
  encapsulation dot1q 301 provider-vlan 3001
  no shutdown

interface port-channel1.2
  encapsulation dot1q 302 provider-vlan 3001
```

```
  no shutdown

interface port-channel1.3
  encapsulation dot1q 303 provider-vlan 3001
  no shutdown

interface port-channel1.4
  encapsulation dot1q 304 provider-vlan 3001
  no shutdown

interface port-channel1.5
  encapsulation dot1q 305 provider-vlan 3001
  no shutdown

interface Ethernet1/24
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 3001
  mtu 9216
  storm-control broadcast level 0.01
  storm-control multicast level 0.01
  storm-control unicast level 0.01
  storm-control action trap
  switchport isolated
  switchport mac-address static-only
  channel-group 1 mode active
  no shutdown

interface Ethernet1/25
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 3001
  mtu 9216
  storm-control broadcast level 0.01
  storm-control multicast level 0.01
  storm-control unicast level 0.01
  storm-control action trap
  switchport isolated
  switchport mac-address static-only
  channel-group 1 mode active
  no shutdown

mac address-table static 0035.0100.0001 vlan 3001 interface Ethernet1/22.1
mac address-table static 0035.0100.0002 vlan 3001 interface Ethernet1/22.2
mac address-table static 0035.0100.0003 vlan 3001 interface Ethernet1/22.3
mac address-table static 0035.0100.0004 vlan 3001 interface Ethernet1/22.4
mac address-table static 0035.0100.0005 vlan 3001 interface Ethernet1/22.5

mac address-table static 003b.0100.0001 vlan 3001 interface port-channel1.1
mac address-table static 003b.0100.0002 vlan 3001 interface port-channel1.2
mac address-table static 003b.0100.0003 vlan 3001 interface port-channel1.3
mac address-table static 003b.0100.0004 vlan 3001 interface port-channel1.4
mac address-table static 003b.0100.0005 vlan 3001 interface port-channel1.5


router ospf p1
  bfd
  router-id 192.168.210.1

interface loopback0
  ip address 192.168.210.1/32
  ip router ospf p1 area 0.0.0.0
  ip pim sparse-mode
```

```
interface loopback1
  description NVE_IP
  ip address 192.168.210.2/32
  ip router ospf p1 area 0.0.0.0
  ip pim sparse-mode

interface Ethernet1/49
  mtu 9216
  no ip redirects
  ip address 10.0.1.16/31
  ip router ospf p1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet1/54
  mtu 9216
  no ip redirects
  ip address 10.0.1.18/31
  ip router ospf p1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface nve1
  no shutdown
  source-interface loopback1
 member vni 10013001
    mcast-group 227.1.1.1
```

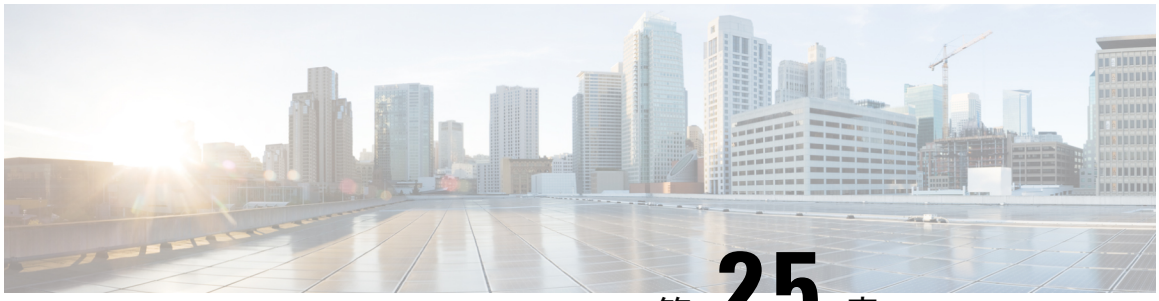
次に、ポート マルチ VLAN マッピングに関連する show コマンドの出力例を示します。

```
switch# show hardware access-list resource utilization | grep Super

Ingress PACL Super Bridge               2445    115     95.50
 Ingress PACL Super Bridge IPv4         0               0.00
 Ingress PACL Super Bridge IPv6         0               0.00
 Ingress PACL Super Bridge MAC          0               0.00
 Ingress PACL Super Bridge ALL          1956            76.40
 Ingress PACL Super Bridge OTHER        489             19.10

switch # show hardware access-list resource entries | in Super

Ingress PACL Super Bridge                : 2445 valid entries   115 free entries

switch# show interface ethernet 1/22.1-5 brief
--------------------------------------------------------------------------------
Ethernet        VLAN    Type Mode   Status  Reason                 Speed     Port
Interface                                                                    Ch #
--------------------------------------------------------------------------------
Eth1/22.1       301     eth  trunk  up      none                    10G(D) --
Eth1/22.2       302     eth  trunk  up      none                    10G(D) --
Eth1/22.3       303     eth  trunk  up      none                    10G(D) --
Eth1/22.4       304     eth  trunk  up      none                    10G(D) --
Eth1/22.5       305     eth  trunk  up      none                    10G(D) --


switch# show interface port-channel 1.1-5 brief
--------------------------------------------------------------------------------------------
Port-channel VLAN    Type Mode   Status  Reason     Speed   Protocol
Interface
--------------------------------------------------------------------------------------------
Po1.1        301     eth  trunk  up      none        a-10G(D)    --
Po1.2        302     eth  trunk  up      none        a-10G(D)    --
Po1.3        303     eth  trunk  up      none        a-10G(D)    --
Po1.4        304     eth  trunk  up      none        a-10G(D)    --
```

```
Po1.5        305     eth  trunk  up      none        a-10G(D)    --

switch# show interface ethernet 1/22.1 counters

--------------------------------------------------------------------------------
Port                                    InOctets                    InUcastPkts
--------------------------------------------------------------------------------
Eth1/22.1                          1145503766466                     125246421

--------------------------------------------------------------------------------
Port                                  InMcastPkts                   InBcastPkts
--------------------------------------------------------------------------------
Eth1/22.1                                     0                             0

--------------------------------------------------------------------------------
Port                                   OutOctets                   OutUcastPkts
--------------------------------------------------------------------------------
Eth1/22.1                                     0                             0

--------------------------------------------------------------------------------
Port                                 OutMcastPkts                   OutBcastPkts
--------------------------------------------------------------------------------
Eth1/22.1                                     0                             0

switch# show consistency-checker l2 sub-interface port-channel 1.1

Getting details for port-channel1.1 (0x16001000)
========================================
Running CC for port-channel1.1
========================================
CC for Permit Static: PASSED
CC for Deny ACL: PASSED
CC for Permit ARP ACL: PASSED
CC for Permit Multi-Dest ACL: PASSED
CC for info_src_idx: PASSED
CC for info_bd_xlate_idx: PASSED
CC for info_vlan_mbr_chk_bypasss: PASSED
CC for info_set_dont_learn: PASSED
CC for VlanXlate Table: PASSED
CC for BD State Table: PASSED
CC for QSMT BD State Table: PASSED
CC for Local Multipath Table: PASSED
CC for Rw VifTable: PASSED
CC for Rwx VlanXlate Table: PASSED


switch# show system internal access-list interface eth 1/22.1

slot 1
=======

Policies in ingress direction:
Policy type Policy Id Policy name
------------------------------------------------------------
PACL Super Bridge 341 l2fm-acl-mac-Eth1/22.1
PACL Super Bridge 342 l2fm-acl-ipv6-Eth1/22.1

No Netflow profiles in ingress direction

INSTANCE 0x0
---------------

Tcam 20 resource usage:
----------------------
```

```
LBL AB = 0x11
Bank 0
------
IPv6 Class
Policies: PACL Super Bridge(l2fm-acl-ipv6-Eth1/22.1)
Netflow profile: 0
Netflow deny profile: 0
2 tcam entries
MAC Class
Policies: PACL Super Bridge(l2fm-acl-mac-Eth1/22.1)
Netflow profile: 0
Netflow deny profile: 0
3 tcam entries

0 l4 protocol cam entries
0 mac etype/proto cam entries
0 lous
0 tcp flags table entries
0 adjacency entries

No egress policies
No Netflow profiles in egress direction
```

```
switch# show system internal access-list interface eth 1/22.1 input statistics

slot 1
=======
INSTANCE 0x0
---------------

Tcam 20 resource usage:
----------------------
LBL AB = 0xb
Bank 0
------
IPv6 Class
Policies: PACL Super Bridge(l2fm-acl-ipv6-Eth1/22.1)
Netflow profile: 0
Netflow deny profile: 0
Entries:
[Index] Entry [Stats]
---------------------
[0x0038:0x0038:0x0038] permit lbl(0x0) 0000.0000.0000 ffff.ffff.ffff 0000.0000.0000
ffff.ffff.ffff vlan 502 [9]
[0x003a:0x003a:0x003a] permit lbl(0x0) 0000.0000.0000 ffff.ffff.ffff 0000.0000.0000
ffff.ffff.ffff vlan 502 [0]
MAC Class
Policies: PACL Super Bridge(l2fm-acl-mac-Eth1/22.1)
Netflow profile: 0
Netflow deny profile: 0
Entries:
[Index] Entry [Stats]
---------------------
[0x003c:0x003c:0x003c] permit lbl(0x0) arp [7]
[0x003d:0x08de:0x08de] permit lbl(0x0) 0035.0100.0001 ffff.ffff.ffff 0000.0000.0000
ffff.ffff.ffff vlan 502 [6279856]
[0x08dd:0x08e0:0x08e0] deny lbl(0x0) 0000.0000.0000 ffff.ffff.ffff 0000.0000.0000
ffff.ffff.ffff vlan 502 [279]
```



第 25 章

グループ ポリシー オプションを使用した VXLAN ファブリックのマイクロセグメンテーション（GPO）

概要

ネットワーク管理者は、マイクロセグメンテーションを使用して、アプリケーション属性などの特定の基準に基づいてネットワーク リソースを論理的にグループ化できます。セキュリティ グループ（SG）およびセキュリティ グループ ACL（SGACL）でマイクロセグメンテーションを使用すると、ネットワーク トポロジに関係なく、セキュリティ グループ間でカスタマイズされたアプリケーション中心のセキュリティ ポリシーを作成および適用できます。

従来のデータセンター環境では、アプリケーションまたはワークロードのセキュリティは、外部のデータセンター ファブリックからのユーザーが入る境界または南北境界に実装されることがよくあります。これは、多くの場合、境界ファイアウォールやその他のセキュリティ検査デバイスを使用して実装されます。ただし、このアプローチは、最近の攻撃の高度な性質に対しては効果的ではありません。攻撃対象領域は、East-West および North-South フローを含むデータセンター全体に及ぶ。

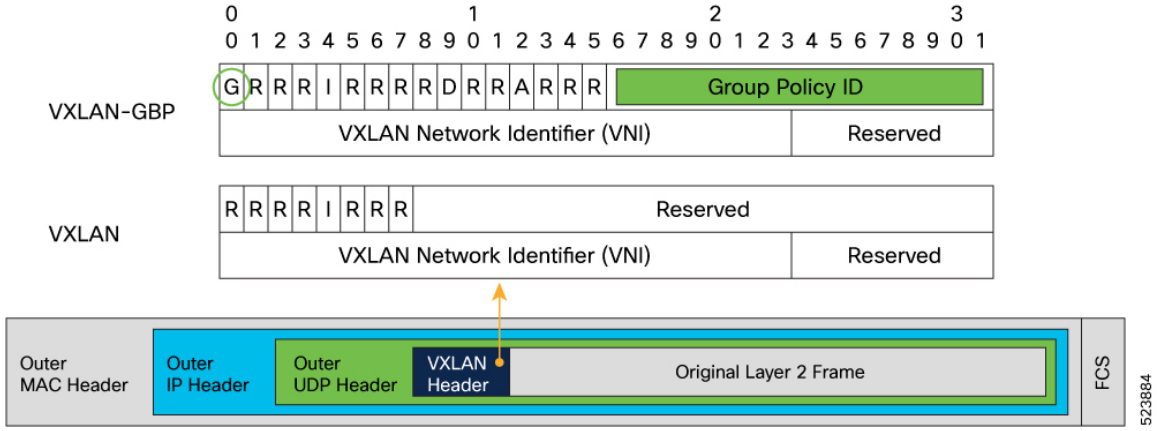
セキュリティ グループおよびセキュリティ グループ ACL でマイクロセグメンテーションを使用すると、この機能は NX-OS プラットフォームのユーザーに効果的なセキュリティ ソリューションを提供できます。マイクロセグメンテーションは従来の汎用アクセス制御リスト（ACL）よりも柔軟性が高く、複雑さが軽減されます。マイクロセグメンテーションを使用すると、組

織は、アプリケーションがネットワーク内のどこに常駐するかに関係なく、アプリケーションワークロードの通信方法を操作する固有のポリシーを提供できます。

GPO

グループポリシーオプション（GPO）はVXLANの下位互換性のある拡張機能であり、セキュリティ ポリシーを適用するために VXLAN ヘッダーにセキュリティ グループ タグ（SGT）を追加します。

図 *46 : VXLAN* ヘッダーのセキュリティ グループ タグ



GPO 対応の VXLAN ネットワークでは、VXLAN EVPN ファブリックにセキュリティ グループを作成してセグメンテーションを定義できます。小規模で分離されたアプリケーションセグメントを定義することで、アプリケーション階層間およびアプリケーション間のネットワーク トラフィックのフローをより適切に制御できるマイクロセグメンテーション ポリシーを展開できます。マイクロセグメンテーションにより、セキュリティ ポリシーが必要な場所にのみ適用され、アプリケーションとワークロードのセキュリティが向上し、セキュリティ体制が向上します。

複数の属性に基づいて、ネットワーク リソースをセキュリティ グループ タグに分類できます。セキュリティ グループ間のトラフィックは、セキュリティ グループ アクセス コントロール リスト（SGACL）（セキュリティ コントラクトとも呼ばれる）によって制御できます。これは、セキュリティ グループ タグを使用して送信元と接続先のセキュリティ グループを照合します。

用語

Security Group（SG）

セキュリティ グループは、属性/セレクタに基づいて分類される物理または仮想ネットワーク エンドポイントの収集を含む論理エンティティです。

送信元セキュリティ グループ タグ（S-SGT）

送信元属性から派生したタグは、送信元セキュリティ グループ タグと呼ばれます。

宛先セキュリティグループタグ（D-SGT）

宛先属性から派生したタグは、宛先セキュリティ グループ タグと呼ばれます。

セキュリティ グループ アクセス コントロール リスト（SGACL）

SGACL は、異なるセキュリティ グループ間で特定のセキュリティ ルール（L4 フィルタ）を適用するためにセキュリティ タグを使用します。タグは、IP、VLAN、VM 属性から取得されます。SGACL を使用すると、SG 間でセキュリティ ポリシーを適用できます。SGACL はコントラクトとも呼ばれます。このドキュメントの一部では、SGACL はコントラクトと呼ばれます。

VRF レベルの適用

セキュリティ グループ セレクタは、どのエンドポイントおよび外部 IP がセキュリティ グループに属するかを定義します。セキュリティ グループには、さまざまな VRF の一部であるエンドポイントを含めることができます。異なる VRF のエンドポイント部分が同じ SG に関連付けられている場合、それらの間の通信は、必要な VRF ルートリーク構成を適用した後にのみ可能になります。

デフォルトでは、新しく定義されたテナント VRF のポリシー適用は [適用なし（Unenforced）] に設定されています。つまり、セキュア グループ間の分類基準と SGACL がプロビジョニングされた場合でも、ポリシーを適用することはできません。 VRF で SGACL の適用を有効にするには、VRF を **[適用（ Enforced）]** モードで明示的に構成する必要があります。

適用モードで VRF を構成する場合、デフォルトの動作を次のいずれかに定義できます。

- **[拒否（Deny）]** : 許可リストで許可されていない限り、すべてのユニキャスト トラフィック フローがドロップされます。
- **[許可（Permit）]** : 拒否リストによって拒否されない限り、すべてのユニキャスト トラフィック フローが許可されます。

SG 内のホストは、明示的な SGACL なしで自由に通信できます。SGACL はセキュリティ ルールのみを作成します。

注意事項と制約事項

GPO には、次の注意事項と制限事項 があります。

- サポートされているすべての N9000 プラットフォームでは、この機能をサポートするために 24GB 以上のシステム メモリが必要です。
- Cisco NX-OS リリース 10.4(3)F 以降、GPO は次のプラットフォームでサポートされます。
  - 9300-FX3

- 9300-GX
- 9300-GX2

- SGACL は VXLAN EVPN 展開のコンテキストでのみサポートされSGACL は、VXLAN が有効になっていない VRF には展開できません。
- SGACL は BUM およびマルチキャスト トラフィックには適用されません。BUM およびマルチキャスト トラフィックには、システムによって生成されたデフォルトの許可ポリシーが存在します。
- **system reserved-vlan-range** 値の VLAN 部分を使用して VLAN ベースのセキュリティ グループ セレクタを構成することはできません。
- VLAN ベースのセキュリティ グループ セレクタがすでに構成されている場合、SG セレクタで使用される VLAN 値を含めるように system-reserved-vlan-range を変更することはできません。
- GPO は、ポリシー対応ノードとポリシー非対応ノードがあるサイトではサポートされません。GPO は、ポリシー対応サイト、またはポリシー対応サイトとポリシー非対応サイトの混在でサポートされます。

Configuring Micro-Segmentation using GPO

GPO の有効化

マイクロセグメンテーション機能を有効にするには、次の手順を実行します。この機能を初めて有効にする場合は、ルーティング テンプレートを **system routing template-security-groups** に構成する必要があります。

警告

- このルーティングテンプレートは、 **feature security-group**に必要です。テンプレート モードを適用した後、機能が有効になっていることを確認します。
- このルーティング テンプレートには、拡張 SSD の再パーティション化が必要です。これは、 **copy running-config startup-config** および **system flash sda resize extended** コマンドを実行することで実現できます。

（注） 続行する前に、ブートフラッシュ、ログフラッシュ、および実行コンフィギュレーション内の内容をバックアップすることを推奨します。詳細については、『Cisco Nexus 9000 シリーズ NX-OS 基本構成ガイド、リリース 10.4(x)』を参照してください。

feature security-group の連続した無効化および再有効化は、スイッチリロードする必要なく完了できます。

手順の概要

1. **configure terminal**
2. **system routing template-security-groups**
3. **copy running-config startup-config**
4. **system flash sda resize extended**
5. **[no] feature security-group**
6. **show nve peers detail**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**system routing template-security-groups** 例： `switch(config-if)# system routing template-security-groups`	スイッチのルーティング プロファイルを変更します。 （注） ルーティング テンプレートは、**system routing template-security-groups** に構成する必要があります。ルーティング テンプレートには、リロードを開始する **system flash sda resize** を介して実行される拡張SSDパーティショニングが必要です。
ステップ 3	**copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします

ステップ 4	**system flash sda resize extended**
	例：
	`switch(config-if)# system flash sda resize extended`
	`!!!! WARNING !!!!`
	
	`Attempts will be made to preserve drive contents during`
	`the resize operation, but risk of data loss does exist.`
		ストレージ容量を増やします。

	コマンドまたはアクション	目的
	Backing up of bootflash, logflash, and running configuration is recommended prior to proceeding. !!!! WARNING !!!! current scheme is sda 8:0 0 119.2G 0 disk |-sda1 8:1 0 512M 0 part |-sda2 8:2 0 32M 0 part /mnt/plog |-sda3 8:3 0 128M 0 part /mnt/pss |-sda4 8:4 0 110.5G 0 part /bootflash |-sda5 8:5 0 64M 0 part /mnt/cfg/0 |-sda6 8:6 0 64M 0 part /mnt/cfg/1 `-sda7 8:7 0 8G 0 part /logflash target scheme is sda 8:0 0 120GB|250GB 0 disk |-sda1 8:1 0 512M 0 part |-sda2 8:2 0 32M 0 part /mnt/plog |-sda3 8:3 0 128M 0 part /mnt/pss |-sda4 8:4 0 rem 0 part /bootflash |-sda5 8:5 0 1.0G 0 part /mnt/cfg/0 |-sda6 8:6 0 1.0G 0 part /mnt/cfg/1 |_sda7 8:7 0 39G 0 part /logflash Continue? (y/n) [n] y A module reload is required for the resize operation to proceed Please, do not power off the module during this process.	
ステップ 5	**[no] feature security-group** 例： `switch(config-if)# **feature security-group**`	グループ ポリシー オプション（GPO）機能を有効にします。機能を無効にするには、「no」プレフィックスを使用します。GPO機能は、実行時に無効または有効にできます。
ステップ 6	**show nve peers detail** 例： `switch(config-if)# **show nve peers detail**` Details of nve Peers: ---------- Peer-Ip: 1.1.1.1 NVE Interface : nve1 Peer State : Up Peer Uptime : 1d12h Router-Mac : 5292.ca60.1b08 Peer First VNI : 101 Time since Create : 1d12h Configured VNIs : 100-101,200-201 Provision State : peer-add-complete	ピア デバイスにグループ ポリシー オプションが有効になっていることを確認します。

	コマンドまたはアクション	目的
	Learnt CP VNIs : 100-101,200-201 vni assignment mode : SYMMETRIC Peer Location : FABRIC Group policy option : yes ----------	

セキュリティ グループの作成

セキュリティ グループを作成または更新し、メンバー選択基準を構成するには、次の手順を実行します。グループ メンバーを選択するには、次の属性を任意に組み合わせて指定できます。

- 接続されたエンドポイントと外部サブネットの IPv4 アドレスまたはサブネット。
- 接続されたエンドポイントと外部サブネットの IPv6 アドレスまたはサブネット。
- スイッチ レベルで VLAN を照合します。

手順の概要

1. **configure terminal**
2. **security-group***sg-id* **name***sg-name*
3. **[no] match [connected-endpoints | external-subnets] vrf***vrf-name***[ipv4|ipv6]** *ip-prefix*
4. **[no] match vlan** *vlan-id*
5. **show security-group id** *sg-id*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**security-group***sg-id* **name***sg-name*
	例：
	`switch(config)# security-group 100 name webservers`
	`switch(config-security-group)#`
		一意の ID が *sg-id* で名前が *sg-name*であるセキュリティ グループを作成（または既存を選択）します。

ステップ 3	**[no] match [connected-endpoints | external-subnets] vrf***vrf-name***[ipv4|ipv6]** *ip-prefix* 例： `switch(config-security-group)# match connected-endpoints vrf vrf_blue ipv4 61.1.1.141/32`	このコマンドは、ホスト（接続されたエンドポイント）または外部（外部サブネット）リソースの IPv4-VRF または IPv6-VRF セレクタです。 特定の分類を無効にするには、「**no**」プレフィックスを使用します。

	コマンドまたはアクション	目的
	switch(config-security-group)# **match external-subnets vrf vrf_blue ipv4 10.0.0.0/8** switch(config-security-group)# **match connected-endpoints vrf vrf_blue ipv6 61:1:1:2:1::141/128** switch(config-security-group)# **match external-subnets vrf vrf_blue ipv6 10:11:12:13::/64**	
ステップ 4	**[no] match vlan** *vlan-id* 例： switch(config-security-group)# **match vlan 10**	スイッチ レベルで VLAN セレクタを構成します。
ステップ 5	**show security-group id** *sg-id* 例： switch(config-if)# show security-group id all Security Group ID 100 , Name webservers, Type Layer4-7 Service Selector Type : External IPv4 Subnets VRF-Name IPv4-Address/mask-len vrf_blue 10.0.0.0/8 Selector Type : Connected IPv4 Endpoints VRF-Name IPv4-Address/mask-len vrf_blue 61.1.1.141/32 Selector Type : External IPv6 Subnets VRF-Name IPv6-Address/mask-len vrf_blue 10:11:12:13::/64 Selector Type : Connected IPv6 Endpoints VRF-Name IPv6-Address/mask-len vrf_blue 61:1:1:2:1::141/128 Selector Type : Vlan VLAN-Id Vlan10 Vlan11 Vlan12 Vlan13 Vlan14 Vlan15 Vlan16 Vlan17 Vlan18 Vlan19 Vlan20 Vlan30 Vlan35 Vlan40 Vlan41 Vlan42 Vlan43 Vlan44	グループ ポリシー セレクタを確認します。

	コマンドまたはアクション	目的
	Vlan45 Selector Type : Interface Interface Vlan10	

セキュリティ クラスマップの作成

クラスマップは、クラスマップ内で設定されたさまざまな一致基準に基づいてネットワーク トラフィックを分類します。特定のトラフィック フローを識別するフィルタを定義するセキュリティ クラスマップを作成するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **class-map type security match-any***web-class*
3. **match [default | ip | ipv4 | ipv6]**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**class-map type security match-any***web-class* 例： `switch(config)# class-map type security match-any web-class2`	特定のトラフィック フローを識別するセキュリティ クラス マップを作成します。

ステップ 3	**match [default | ip | ipv4 | ipv6]**
	例：
	`switch(config-cmap-sec)# match ipv4 udp sport 399 to 402 dport 400 to 403`
	`switch(config-cmap-sec)# match ipv6 udp sport 399 to 402 dport 400 to 403`
		トラフィック タイプに基づいて照合することで、セキュリティ クラスを構成します。

セキュリティ ポリシー マップの作成

ポリシー マップは、クラス マップおよびACL を使用して分類されたトラフィックの処理内容を示すポリシーを定義します。次の手順を実行して、セキュリティ ポリシー マップを作成し、

以前に作成したセキュリティ クラス マップによって識別されるアクション（許可、拒否、ログ トラフィック フローを定義します。

手順の概要

1. **configure terminal**
2. **policy-map type security***policy-map*
3. **class web-class**
4. **[no] [permit | deny | log]**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**policy-map type security***policy-map*	セキュリティ ポリシー マップを作成します。
ステップ 3	**class web-class**	ルールが適用されるトラフィックを定義するポリシーマップに関連付けるセキュリティ クラスマップを指定します。
ステップ 4	**[no] [permit | deny | log]**	一致するトラフィックに対して実行するアクションを定義します。 • **[拒否（Deny）]**：一致するトラフィックを拒否します。 • **[ログ（Log）]**：一致するトラフィックをログに記録します。 • **[許可（Permit）]**：一致するトラフィックを許可します。 何も指定されていない場合、**[許可（Permit）]**がデフォルトのアクションです。**[ログ（Log）]**アクションは、許可または拒否で設定できます。一致するトラフィックは、 **show logging ip access-list cache [detail]** に記録されます。

セキュリティ グループ間のセキュリティ コントラクトの構成

この手順では、セキュリティ グループ間のセキュリティ ポリシーを適用する SGACL（コントラクト）を作成します。

始める前に

- セキュリティ グループの作成
- セキュリティ クラスマップの作成
- セキュリティ ポリシー マップの作成

手順の概要

1. **configure terminal**
2. **vrf context***vrf-name*
3. **security enforce tag***sg-id* **default [permit | deny]**
4. **security contract source [***sg-id* | *any***] destination [***sg-id* | *any***] policy policy-map-name** [*bidir* | *unidir*]

手順の詳細

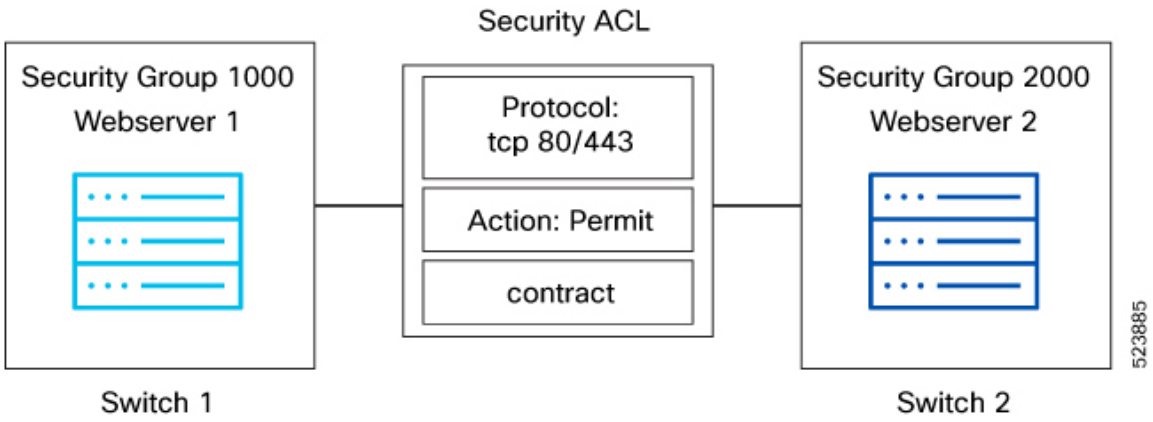
手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**vrf context***vrf-name* 例： `switch(config)# vrf context blue`	指定した VRF の構成モードを開始します。
ステップ 3	**security enforce tag***sg-id* **default [permit | deny]** 例： `switch(config-vrf)# security enforce tag 100 default deny`	VRF を強制モードに移行します。 • **sg-id**： デフォルト オプションが追加されるテナント VRF のセキュリティ グループ タグを定義します。 • **default deny**： 明示的なセキュリティ コントラクトなしでVRF内のすべてのトラフィックを拒否します。 • **default permit**： 明示的なセキュリティ コントラクトなしでVRF内のすべてのトラフィックを許可します。

	コマンドまたはアクション	目的
ステップ 4	**security contract source [***sg-id* | *any***] destination [***sg-id* | *any***] policy policy-map-name** [*bidir* | *unidir*] 例： `switch(config-vrf)# security contract source 100 destination 200 policy policyMap1 bidir`	以前に定義したセキュリティ ポリシー マップを、対応するアクションとともに、指定したセキュリティ グループ間で適用します。 方向が指定されていない場合、デフォルトのオプションは **bidir** です。デフォルト オプションの **bidir** は、SGACL を両方向のトラフィックに適用します（たとえば、100 から 200 および 200 から 100）。 たとえば、宛先ポート 80 を指定するフィルタを使用して SG 100 と SG 200 の間にセキュリティ ルールを作成する場合、 **bidir** を使用すると、送信元ポート 80 を使用する SG 200 と SG 100 間の通信にもルールが適用され、双方向通信を正常に確立できます。

GPO の構成例

図 *47:* セキュリティ グループの作成



ステップ 1：GPO を有効にします

```
Switch1# configure terminal
Switch1(config)# system routing template-security-groups
Switch1(config)#feature security-group

Switch2# configure terminal
Switch2(config)# system routing template-security-groups
Switch2(config)#feature security-group
```

ステップ 2：特定のトラフィック フローを識別するためのセキュリティ クラス マップを作成します。

```
Switch1(config)#class-map type security match-any web-class
match ipv4 tcp dport 443
match ipv4 tcp dport 80
```

```
Switch2(config)#class-map type security match-any web-class
match ipv4 tcp dport 443
match ipv4 tcp dport 80
```

ステップ 3：セキュリティ ポリシー マップを作成します

```
Switch1(config)#policy-map type security policyMap1
class web-class
permit
Switch2(config)#policy-map type security policyMap1
class web-class
permit
```

ステップ 4：セキュリティ グループを作成します。

```
switch1(config)security-group 1000 name webserver1
switch1(config-security-group)# match connected-endpoints vrf vrf_blue ipv4 61.1.1.141/32
switch1(config-security-group)# match external-subnets vrf vrf_blue ipv4 10.0.0.0/8
switch1(config-security-group)# match connected-endpoints vrf vrf_blue ipv6
61:1:1:2:1::141/128
switch1(config-security-group)# match external-subnets vrf vrf_blue ipv6 10:11:12:13::/64
switch1(config-security-group)# match connected-endpoints vrf vrf_red ipv4 100.5.150.125/32

switch1(config-security-group)# match connected-endpoints vrf vrf_red ipv6
100:1:1:495::125/128
switch1(config-security-group)# match external-subnets vrf vrf_red ipv4 11.0.0.0/8
switch1(config-security-group)# match vlan 10

switch2(config)security-group 2000 name webserver2
switch2(config-security-group)# match connected-endpoints vrf vrf_blue ipv4 61.1.1.142/32
switch2(config-security-group)# match external-subnets vrf vrf_blue ipv4 20.0.0.0/8
switch2(config-security-group)# match connected-endpoints vrf vrf_blue ipv6
61:1:1:2:1::142/128
switch2(config-security-group)# match external-subnets vrf vrf_blue ipv6 20:11:12:14::/64
switch2(config-security-group)# match connected-endpoints vrf vrf_red ipv4 100.5.150.126/32

switch2(config-security-group)# match connected-endpoints vrf vrf_red ipv6
100:1:1:495::126/128
switch2(config-security-group)# match external-subnets vrf vrf_red ipv4 21.0.0.0/8
switch2(config-security-group)# match vlan 10
```

ステップ 5：VRF を強制モードに移行します

```
switch1(config)# vrf context vrf_blue
switch1(config-vrf)# security enforce tag 100 default deny
switch2(config)# vrf context vrf_red
switch2(config-vrf)# security enforce tag 101 default deny
```

ステップ 6：契約を適用します

```
switch1(config-vrf-blue)# security contract source 1000 destination 2000 policy policyMap1

switch1(config-vrf-red)# security contract source 1000 destination 2000 policy policyMap1


switch2(config-vrf-blue)# security contract source 1000 destination 2000 policy policyMap1

switch2(config-vrf-red)# security contract source 1000 destination 2000 policy policyMap1
```

GPO の確認

次に、GPO 構成に関連するいくつかの show コマンドを示します。

show contracts

すべての vrfs のスイッチに適用されているすべてのコントラクトを表示します。

```
switch(config)# show contracts

VRF              SGT   DGT   Policy           Dir    Stats        Class            Action
       OperSt
-------------------------------------------------------------------------------------------------
vrf_blue             1000   2000 policyMap1        bidir  350370       web-class
 permit,log   enabled
vrf_red              1000   2000 policyMap1        bidir  373270       web-class
 permit,log   enabled
```

show run security-group

スイッチのセキュリティ グループ関連の構成をすべて表示します。

```
switch1(config)# show run security-group
!Command: show running-config security-group
!Running configuration last done at: Fri Dec  8 12:23:52 2023
!Time: Fri Dec  8 12:27:09 2023
version 10.4(2) Bios:version 05.50
feature security-group
security-group 1000 name webserver1
match connected-endpoints vrf vrf_blue ipv4 61.1.1.141/32
match external-subnets vrf vrf_blue ipv4 10.0.0.0/8
match connected-endpoints vrf vrf_blue ipv6 61:1:1:2:1::141/128
match external-subnets vrf vrf_blue ipv6 10:11:12:13::/64
match connected-endpoints vrf vrf_red ipv4 100.5.150.125/32
match connected-endpoints vrf vrf_red ipv6 100:1:1:495::125/128
match external-subnets vrf vrf_red ipv4 11.0.0.0/8
match vlan 10

class-map type security match-any web-class
  match ip udp
  match ip tcp

policy-map type security policyMap1
  class web-class

vrf context vrf_blue
  security contract source 1000 destination 2000 policy policyMap1
  security enforce tag 100 default deny

vrf context vrf_red
  security contract source 1000 destination 2000 policy policyMap1
  security enforce tag 101 default deny
```

show contracts detail

クラスマップとポリシーマップの詳細を含む、スイッチに適用されるすべてのコントラクトの詳細を表示します。

```
switch1(config)# show contracts detail

VRF: vrf_blue
  Contract source group any dest group 2000
```

```
    Policy: policyMap1 Direction: bidir
      Stats: 350370
      Class: web-class
        match ip udp
        match ip tcp
      Action: permit,log
      OperSt: enabled

VRF: vrf_red
  Contract source group any dest group 2000
    Policy: policyMap1 Direction: bidir
      Stats: 373270
      Class: web-class
        match ip udp
        match ip tcp
      Action: permit,log
      OperSt: enabled
```

show contracts policy policyMap1

ポリシー名に基づいてコントラクトを表示します。

```
Switch1(config)show contracts policy policyMap1
VRF             SGT   DGT   Policy          Dir    Stats        Class            Action
       OperSt

-------------------------------------------------------------------------------------------------


vrf_blue            1000  2000   policyMap1       bidir  0            web-class
permit       enabled
vrf_red             1000  2000   policyMap1       bidir  0            web-class
permit       enabled
```

show contracts vrf vrf_blue

vrf に基づいてコントラクトを表示します。

```
switch1(config)show contracts vrf vrf_blue
VRF             SGT   DGT   Policy          Dir    Stats        Class            Action
       OperSt
-------------------------------------------------------------------------------------------------
vrf_blue        1000  2000   policyMap1       bidir  0           web-class       permit
       enabled
```

show contracts sgt 1000

特定の SGT に基づいてコントラクトを表示します。

```
switch1(config)# show contracts sgt 1000

VRF             SGT   DGT   Policy          Dir    Stats        Class            Action
       OperSt
-------------------------------------------------------------------------------------------------
vrf_blue     1000  2000  policyMap1        bidir  0            web-class
permit,log   enabled
vrf_red      1000  2000  policyMap1        bidir  0            web-class
permit,log   enabled
```

show contracts dgt 2000

特定の DGT に基づいてコントラクトを表示します。

```
switch1(config)# show contracts dgt 2000

VRF             SGT   DGT   Policy          Dir    Stats        Class            Action
       OperSt
```

```
-------------------------------------------------------------------------------------------------------
vrf_blue     1000  2000  policyMap1       bidir  0           web-class        permit,log
   enabled
vrf_red      1000  2000  policyMap1       bidir  0           web-class        permit,log
   enabled
```

show contracts sgt 1000 dgt 2000

特定の SGT および DGT に基づいてコントラクトを表示します。

```
switch1(config)# show contracts sgt 1000 dgt 2000

VRF             SGT   DGT   Policy           Dir    Stats       Class            Action
       OperSt
-------------------------------------------------------------------------------------------------------
vrf_blue     1000  2000  policyMap1       bidir  0           web-class        permit,log
   enabled
vrf_red      1000  2000  policyMap1       bidir  0           web-class        permit,log
   enabled
```

VXLAN マルチサイトと GPO の相互運用性

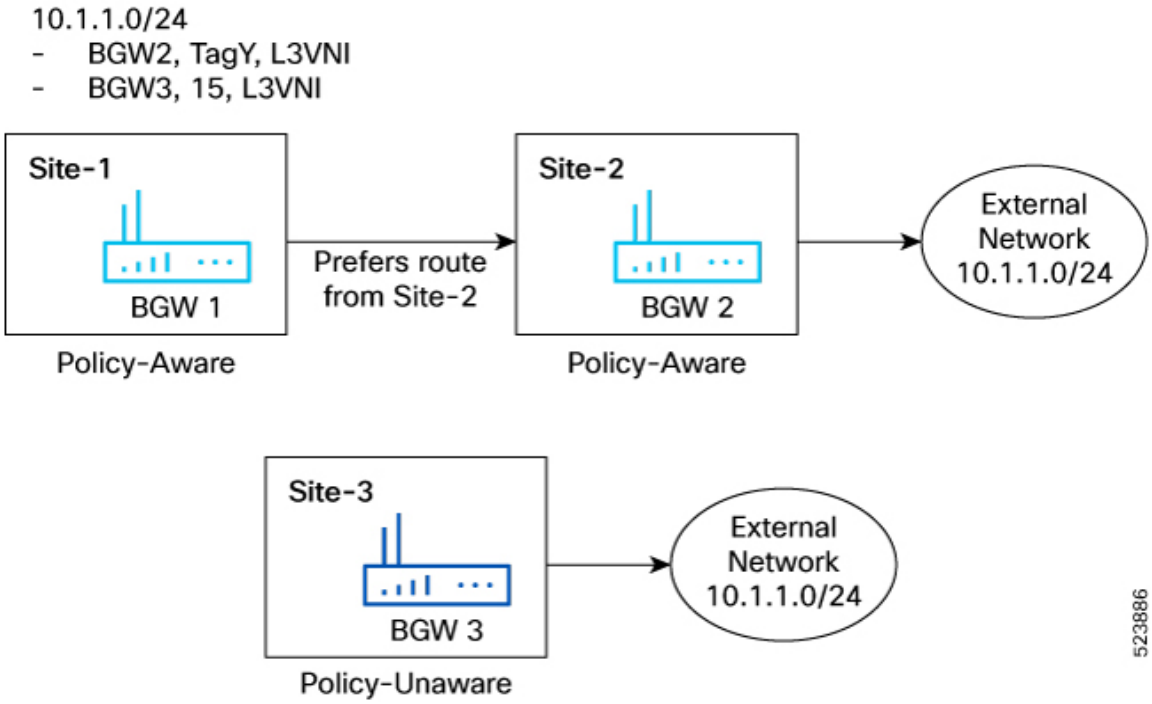
Cisco NX-OS リリース 10.4(3)F 以降、SG および SGACL は、同じマルチサイト ドメインの VXLAN EVPN ファブリック部分でサポートされます。ポリシー対応ファブリックとポリシー非対応ファブリックは、同じマルチサイト ドメインの一部として展開できます。

エニーキャスト ボーダーゲートウェイ（BGW）および vPC BGW は、SGACL 機能でサポートされます。ポリシー対応サイトのボーダー ゲートウェイ ノードは、ポリシー対応サイトおよびポリシー非対応サイトとのマルチサイト EVPN 接続を確立できます。SGACL は、別のファブリックでローカルに定義された SG 間に適用できます。さらに、SG を複数のファブリックに拡張することもできます。

ポリシー非対応サイトからのすべての EVPN ルートは、ポリシー対応サイトに配布され、予約済みのセキュリティ グループ タグ値 15 でインストールされます。ポリシー非対応サイトとポリシー対応サイト間のワークロード通信を許可するには、タグ 15 と目的の宛先タグを使用して明示的なコントラクトを作成する必要があります。

リモートプレフィックスがポリシー非対応ファブリックとポリシー対応ファブリックの混在に属する複数のネクスト ホップで学習される場合、ポリシー対応ファブリックのネクスト ホップが優先されます。複数のネクストホップがすべてポリシー非対応ファブリックに属している場合、受信したプレフィックスルートがインストールされ、ポリシー非対応タグを使用してポリシー対応ファブリック内に再発信されます。

図 *48* : ポリシー対応およびポリシー非対応タグ ネクストホップ プリファレンス



この図では、BGW 1 と BGW 2 はポリシー対応サイトの一部であり、BGW 3 はポリシー非対応サイトに属しています。外部ネットワーク 10.1.1.0/24 は、サイト 3 およびサイト 2 からサイト 1 にアドバタイズされます。サイト 2 はポリシー対応であるため、ルート 10.1.1.0/24 はサイト 2 で構成された「TagY」でアドバタイズされます。サイト 3 はポリシー非対応であるため、BGW3 からアドバタイズされたルートはポリシー情報を伝送せず、BGW1 が受信したときにデフォルトのポリシー非対応タグ「15」を持つようにローカルに割り当てられます。

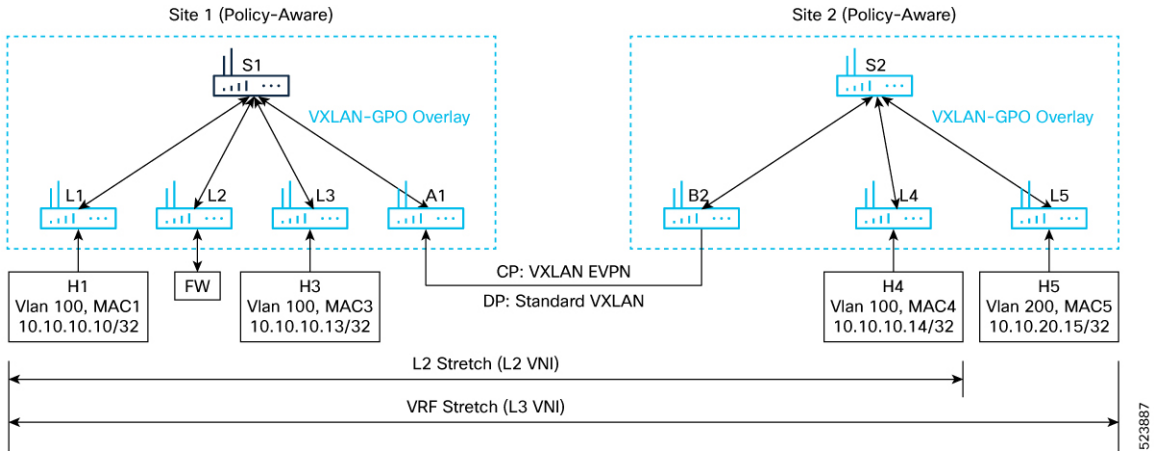
BGW1 では、ルートには BGW2 と BGW3 のオーバーレイ ECMP があり、それぞれ TagY と '15' のタグが付いています。ただし、TagY はポリシー対応サイトからの有効なタグであるため、10.1.1.0/24 は Tag 'TagY' でプログラムされます。同様に、BGW1 がサイト 1 のリーフへのルートを再発信すると、タグ「TagY」がルートに追加されます。

一般的なエニーキャスト BGW セットアップでは、L2VNI の SVI 構成はありません。ただし、L2 ブリッジド IP トラフィックの GPO をサポートするには、エニーキャスト BGW で L2VNI の SVI を構成する必要があります。SVI の構成には、IP アドレスまたは「ip forward」コマンドは必要ありません。唯一の要件は、テナント VRF で構成することです。これは、L2VNI セグメントに接続されたエンドポイントのホスト IP ルックアップ用のテナント VRF 情報を取得し、セキュリティ ポリシーを適用するために必要な SGT および DGT タグを提供するために必要です。

マルチサイト ドメインのポリシー対応ファブリック

ここでは、ポリシー対応マルチサイト ドメイン内の 2 つのホスト間でルート アドバタイズメントとパケット移動がどのように行われるかについて説明します。

図 *49 :* ポリシー対応マルチサイト



上記のトポロジでは、サイト 1 とサイト 2 がポリシー対応です。

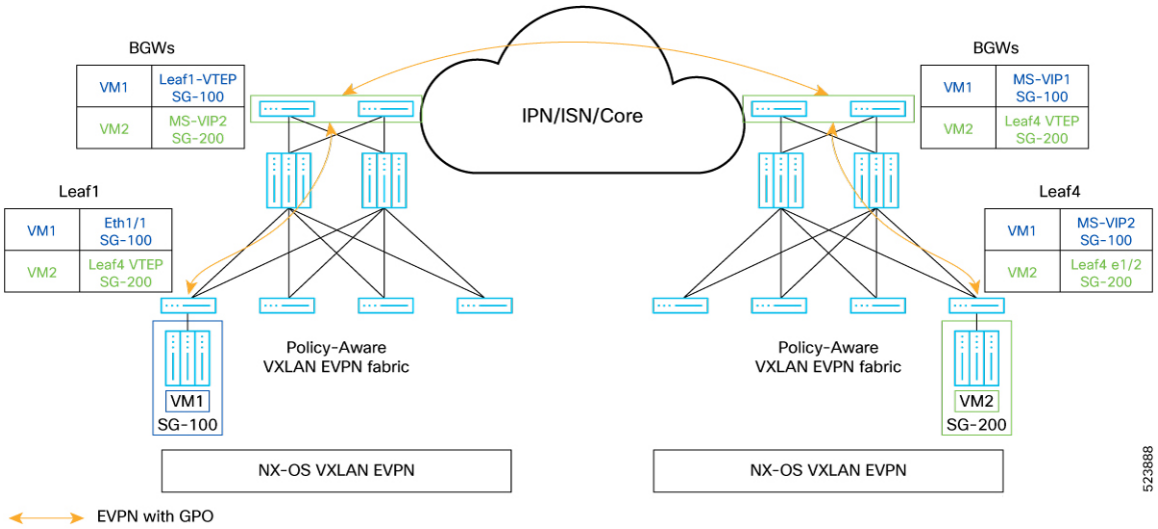
リーフ L1 からリーフ L5 へのホスト H1 のルートアドバタイズメント

ホスト ルート H1 が L1 サイト 1 から L5 サイト 2 にアドバタイズされる場合、ルート アドバタイズメント フローは次のようになります。

1. ネクストホップ ルータ L1 は、ルータ L1 のポリシーによって構成された L1 ネクストホップと送信元グループ タグ（TagX）を使用してルートをアドバタイズします。
2. サイト 1 の BGW であるルータ A1 は、ネクストホップとして BGW のマルチサイト VIP を使用してルートを再発信しますが、L1 から受信した SGT タグ（TagX）を保持します。
3. 同様に、ルータ B2（サイト 2 の BGW）は、ネクストホップとして B2 BGW のマルチサイト VIP と同じ SGT タグ（TagX）を使用して、ローカル ファブリック内のルート H1 を再発信します。
4. L5 はルータ B2 からルートを受信し、関連付けられたセキュリティ タグ「TagX」とともに転送テーブルにインストールします。このようにして、発信元リーフからのタグは、マルチサイト ドメイン全体で保持されます。

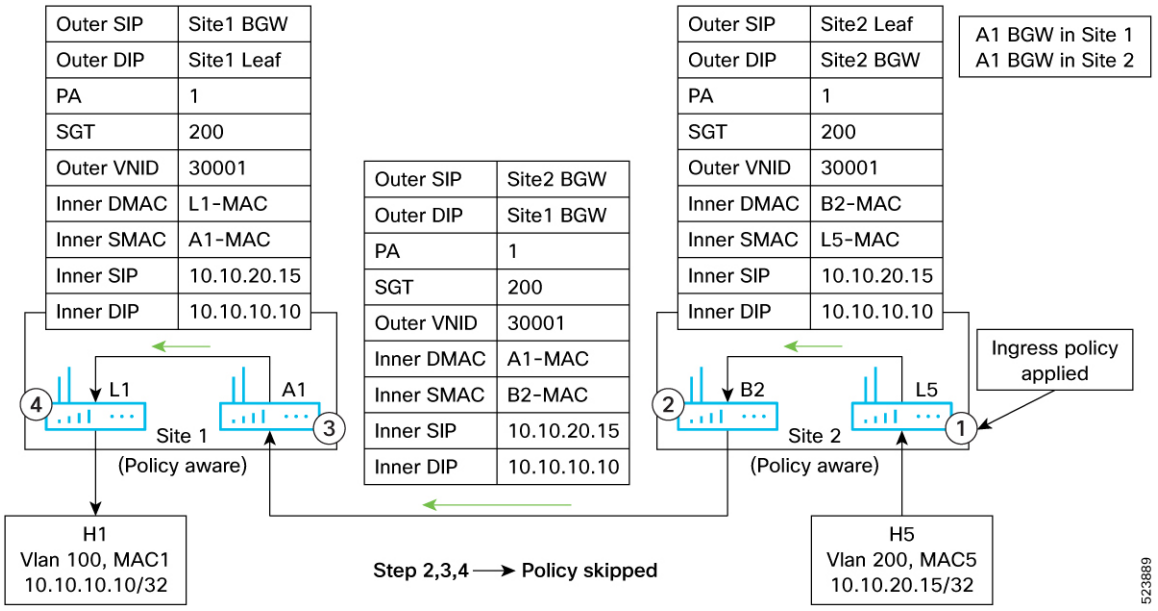
さまざまなノードで学習されたエンドポイント情報については、次の図を参照してください。

図 *50* : ポリシー対応マルチサイトでのルート アドバタイズメント



ホスト H5 からホスト H1 へのパケットフロー

図 *51* : ホスト *H5* からホスト *H1* へのパケットフロー



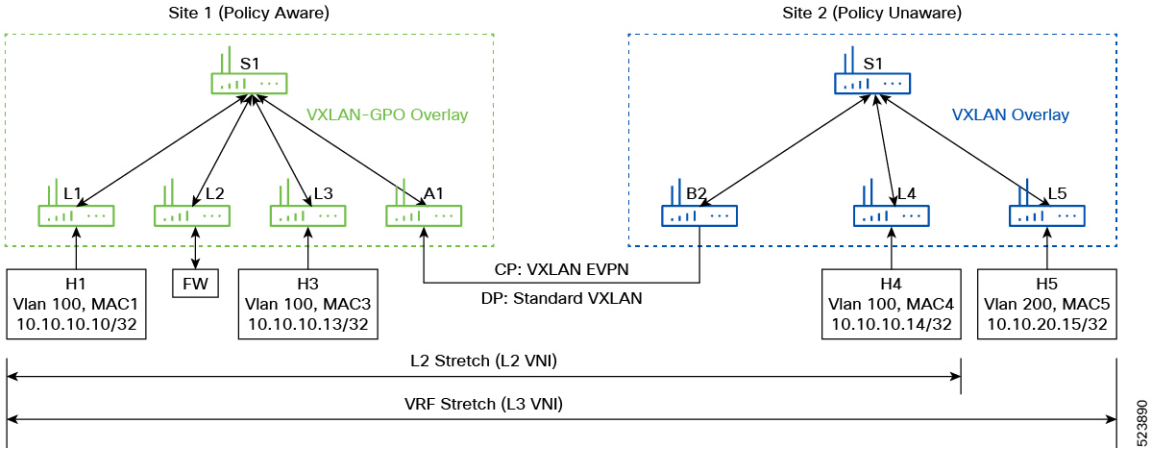
ホスト H5 と H1 間のパケット フローは次のようになります。

1. ルータ L5 は、ホスト H5 からトラフィックを受信します。SRC タグと DST タグの両方を使用できるため、L5はセキュリティポリシーをローカルに適用します。ポリシーアクションが許可されている場合、VXLAN-GPOヘッダーにポリシー適用（PA）ビットを設定し、宛先ホスト H1 に到達するためのネクストホップを表す BGW ルータ B2 にトラフィックを送信します。

2. これとともに、PA ビット設定は VXLAN GPO ヘッダー B2 に保持されます。VXLAN ヘッダーに設定された PA ビットを確認すると、ポリシーは再適用されず、BGW A1 に送信するトラフィックのカプセル化が解除され、再カプセル化され、サイト 1 は、宛先ホスト H1 に到達するためのネクスト ホップを表します。これとともに、PA ビット設定は VXLAN GPO ヘッダーに保持されます。
3. BGW A1 は BGW B2 と同様のアクションを実行し、トラフィックをルータ L1 に転送します。
4. ルータ L1 は、PA ビットが設定されているため、ポリシーも適用せず、トラフィックを宛先ホスト H1 に転送します。

マルチサイト ドメインのポリシー対応ファブリックとポリシー非対応ファブリック

図 *52* : ポリシー対応ポリシー非対応マルチサイト



上の図では、サイト 1 はポリシー対応サイトで、サイト 2 はポリシー非対応サイトです。

ルータ L1 からルータ L5 へのホスト H1 のルート アドバタイズメント

ルータ L1 からルータ L5 へのホスト H1 のルート アドバタイズメントは次のようになります。

1. ルータ L1 は、L1 の構成に基づいて、ネクストホップを L1 とし、SGT タグ（TagY）を使用してルート H1 をアドバタイズします。
2. ルータ A1 は、ネクストホップ L1（および関連付けられたタグ TagY）を持つルートをローカルにインストールし、A1 マルチサイト VIP および同じ SGT タグ（TagY）としてネクストホップを持つリモート BGW B2 へのルートを再発信します。
3. BGW B2 は、SGT タグ（TagY）を含むルートを受信すると、A1 をネクストホップとするルートをローカルにインストールし、SGT タグはポリシー非対応であるため無視します。次に、ネクストホップを B2 マルチサイト VIP としてルータ L5 へのルートを再発信します。
4. ルータ L5 はルート H1 をインストールします。

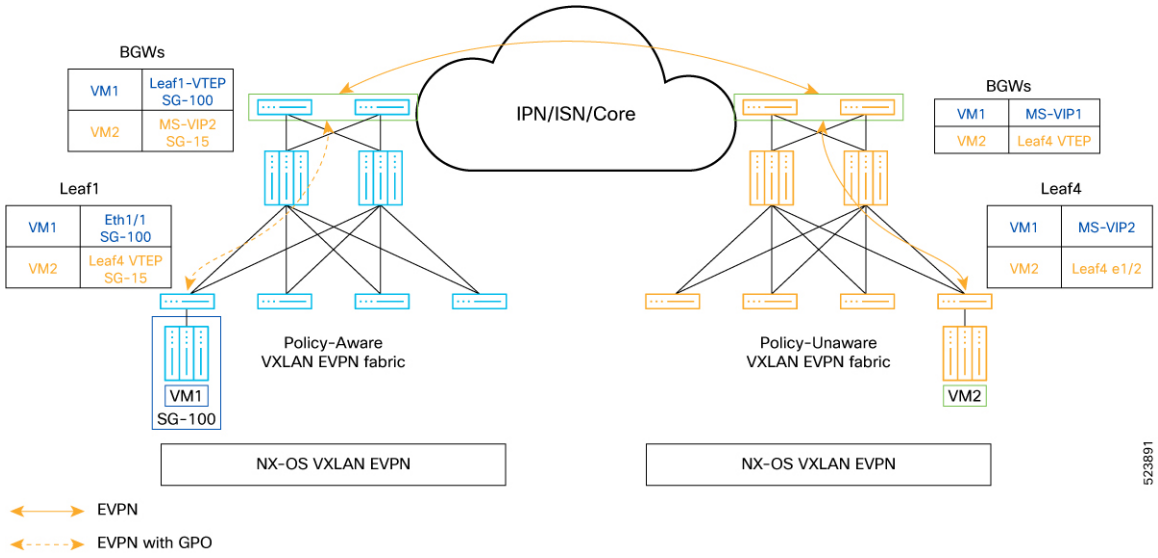
ルータ L5 からルータ L1 へのホスト H5 のルート アドバタイズメント

ルータ L5 からルータ L1 へのホスト H5 のルート アドバタイズメントは次のようになります。

1. ルータ L5 は、ネクストホップを L5 としてホスト H5 のルートをアドバタイズします。
2. BGW B2 は、ネクストホップとして L5 を持つルートをローカルにインストールし、ネクストホップを持つプレフィックスを B2 マルチサイト VIP として再発信します。
3. A1 はポリシー対応サイトにあり、ポリシー非対応ファブリックからルートを受信したため、A1 はポリシー非対応サイトのデフォルト タグ（タグ '15'）を持つルートをインストールします。
4. BGW ルータ A1 は、ネクストホップを A1 マルチサイト VIP およびタグ 15 として、L1 へのルートを再発信します。
5. ルータ L1 は、SGT TAG 15 およびネクストホップを A1 マルチサイト VIP として使用するルートをインストールします。

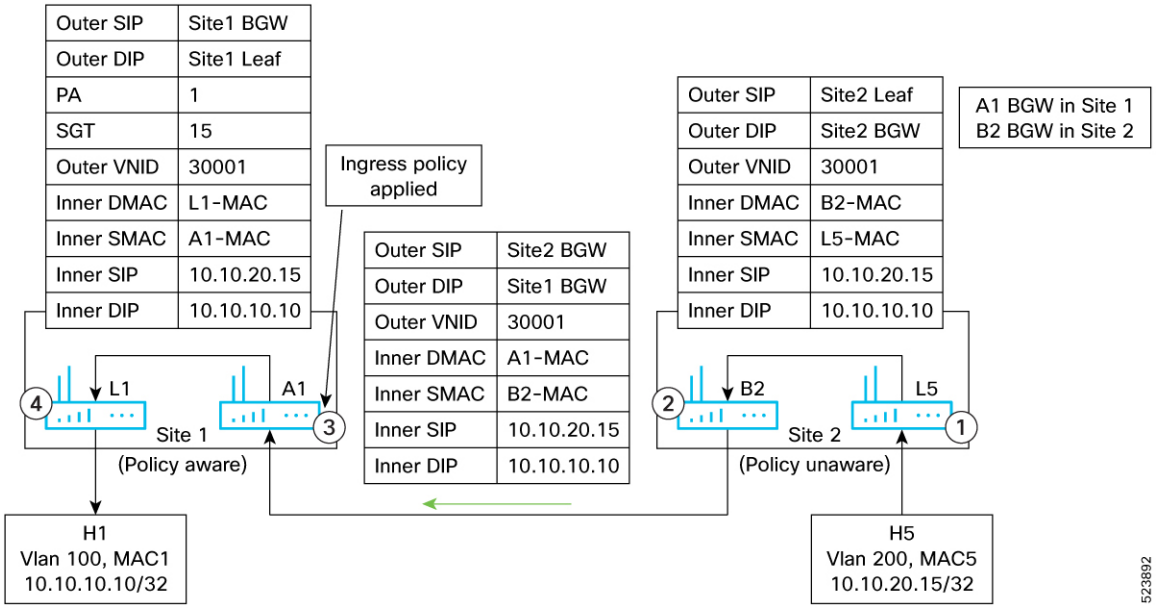
さまざまなノードで学習されたエンドポイント情報については、次の図を参照してください。

図 *53*：ポリシー非認識から認識サイトへのルートアドバタイズメント



ホスト H5 からホスト H1 へのパケット フロー

図 *54 :* ホスト *H5* からホスト *H1* へのパケット フロー

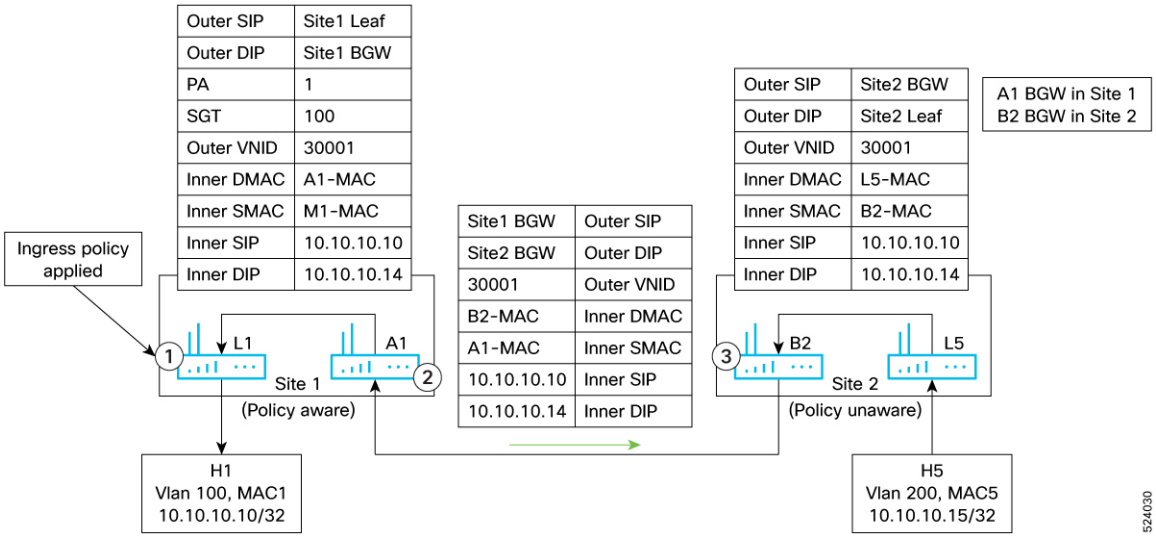


ホスト H5 と H1 間のパケット フローは次のようになります。

1. ルータ L5 はホスト H5 からトラフィックを受信し、標準 VXLAN ヘッダーを使用してトラフィックを B2 にルーティングします。
2. B2 は、標準 VXLAN ヘッダーでカプセル化を解除し、再カプセル化します。
3. A1 はトラフィックのカプセル化を解除し、送信元タグ「15」と宛先「タグ Y」に基づいてポリシーを適用します。ポリシーに基づいて、トラフィックは VXLAN GPO トンネルとポリシー適用ビットが設定された L1 に転送されます。
4. L1 はトラフィックのカプセル化を解除し、PA ビットが設定されているため、ポリシーは再度適用されず H1 に転送されます。

ホスト H1 からホスト H5 へのパケット フロー

図 *55* : ホスト *H1* からホスト *H5* へのパケット フロー



ホスト H1 からホスト H5 へのパケット フローは次のようになります。

1. ルータ L1 は H1 からトラフィックを受信し、送信元タグ「TagY」、宛先タグ「15」に基づいてポリシーを適用します。ポリシーの結果に基づいて、L1 はポリシー適用（PA）ビットが設定された VXLAN GPO トンネルを使用してトラフィックを A1 にルーティングします。
2. A1 はカプセル化を解除し、PA ビットが設定されているため、ポリシーを再適用せず、標準 VXLAN トンネルの B2 にトラフィックを転送します。
3. B2 から H5 へのトラフィック フローは、マルチサイト展開と同様です。

NX-OS ユーザーは、マイクロセグメンテーションと GPO を使用して、ネットワーク内に小規模で分離されたセグメントを作成し、セキュリティ ポリシーを適用できます。これにより、ユーザーはトラフィック フローをより適切に制御し、必要な場所にのみセキュリティ ポリシーを適用できます。

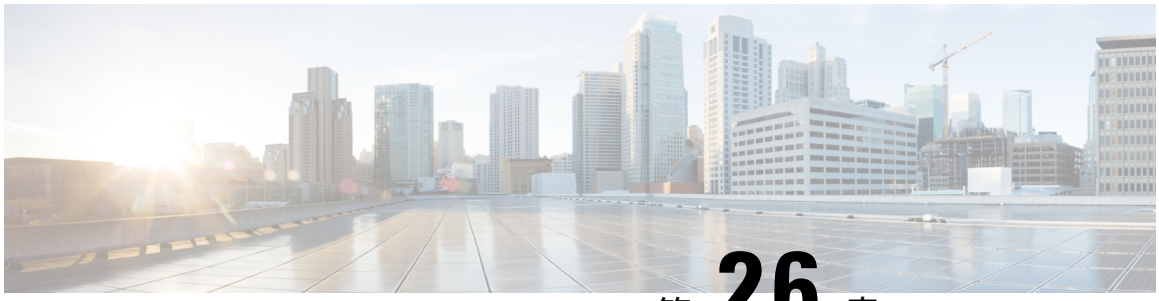
関連資料

https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/104x/configuration/scalability/cisco-nexus-9000-series-nx-os-verified-scalability-guide-1043.html

https://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/nx-os/licensing/guide/cisco-nexus-nx-os-smart-licensing-using-policy-user-guide-102x.html

https://www.cisco.com/c/dam/en/us/td/docs/Website/datacenter/platform/platform.html

https://developer.cisco.com/docs/cisco-nexus-3000-and-9000-series-nx-api-rest-sdk-user-guide-and-api-reference-release-10-4-x/



第 26 章

レイヤ 4 - レイヤ 7 サービスの構成

この章は、次の内容で構成されています。

VXLAN レイヤ 4 - レイヤ 7 サービスについて

この章では、VXLAN ファブリックへのレイヤ 4ーレイヤ 7 サービス（ファイアウォール、ロード バランサなど）の挿入について説明します。

L4-L7 サービスがデフォルト ゲートウェイ（集約/配信）をホストするスイッチに接続されている従来の 3 層ネットワーク トポロジとは異なり、VXLAN ファブリック内の L4-L7 サービスは通常、しばしばサービス リーフと呼ばれる、リーフ スイッチまたは境界スイッチに接続されます。

L4-L7 サービス デバイスは、さまざまな方法で VXLAN ファブリックに接続できます。この章では、L4-L7 サービス デバイスの接続方法、およびデバイスとネットワークの要件に応じて考慮すべき事項について説明します。

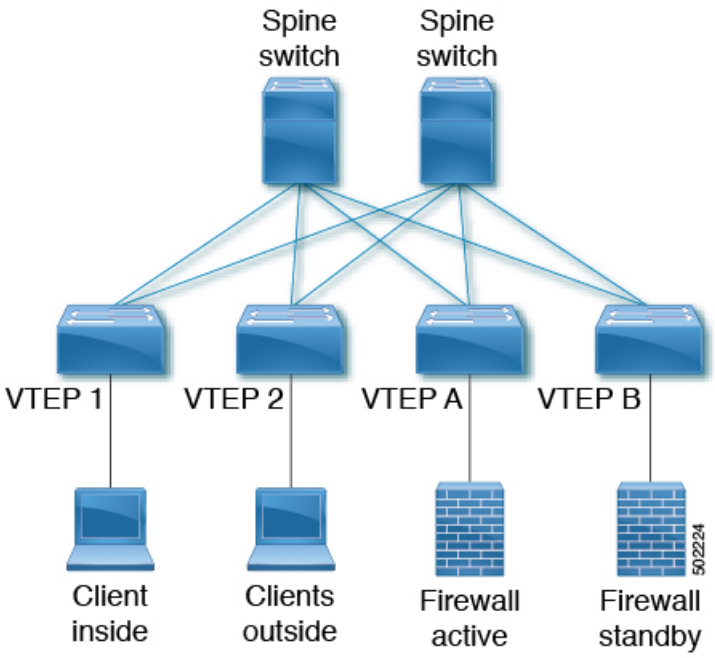
VXLAN ファブリックでのレイヤ 3 ファイアウォールの統合

ここでは、VXLAN EVPN ファブリック内にファイアウォールを統合する方法について詳しく説明します。レイヤ 3 ファイアウォールでは、異なるセキュリティ ゾーンを分離する必要があります。

VXLAN EVPN ファブリックにレイヤ 3 ファイアウォールを分散型エニーキャスト ゲートウェイと統合する場合、これらの各ゾーンはファブリック上の VRF/テナントに対応する必要があります。テナント内のトラフィックは、ファブリックによってルーティングされます。テナント間のトラフィックは、ファイアウォールによってルーティングされます。このシナリオは、多くの場合、テナント間またはテナント エッジ ファイアウォールに関連しています。

内部ゾーンと外部ゾーンの2つのゾーンを検討します。このシナリオでは、ファブリック上の VRF 定義が必要です。VRF を内部 VRF および外部 VRF と呼ぶことができます。同じ VRF 内のサブネット間のトラフィックは、分散ゲートウェイを使用して VXLAN ファブリックでルーティングされます。VRF 間のトラフィックは、ルールが適用されるファイアウォールによってルーティングされます。

図 *56 :* ファイアウォール接続を使用したトポロジの概要



静的ルーティングを使用するシングル接続ファイアウォール

ファイアウォールがルーティング プロトコルの実行をサポートしていない場合は、各 VTEP にネクスト ホップとしてファイアウォールを指す静的ルートが必要です。ファイアウォールには、ネクスト ホップとしてエニーキャスト ゲートウェイ IP を指す静的ルートもあります。静的ルートの課題は、アクティブ ファイアウォールを備えた VTEP が、ファブリックへのルートをアドバタイズする必要があることです。これを実現する1つの方法は、HMM を介してアクティブなファイアウォールの到達可能性を追跡し、この追跡を使用してルートをファブリックにアドバタイズすることです。アクティブなファイアウォールが VTEP A に接続されている場合、VTEP A には、ファイアウォール IP が HMM ルートとして学習された場合にルートがアドバタイズされる場所を追跡する静的ルートがあります。ファイアウォールに障害が発生し、スタンバイ ファイアウォールが引き継ぐと、VTEP A は BGP を使用してファイアウォール IP を学習し、VTEP B は HMM を使用してファイアウォール IP を学習します。VTEP A はルートを

取り消し、VTEP B はファブリックにルートをアドバタイズします。次の例を参照してください。

VTEP A および VTEP B:

```
Vlan 10
 Name inside
 Vn-segment 10010

Vlan 20
 Name outside
 Vn-segment 10020


Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
 IP address 10.1.1.254/24
 fabric forwarding mode anycast-gateway

Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 fabric forwarding mode anycast-gateway

interface nve1
 no shutdown
 host-reachability protocol bgp
 source-interface loopback1
 member vni 10010
  mcastgroup 239.1.1.1
 member vni 10020
  mcastgroup 239.1.1.1
 member vni 1001000 associate-vrf
 member vni 1002000 associate-vrf


track 10 ip route 10.1.1.1/32 reachability hmm
  vrf member INSIDE
!
VRF context INSIDE
 Vni 1001000
 IP route 20.1.1.0/24 10.1.1.1 track 10


track 20 ip route 20.1.1.1/32 reachability hmm
  vrf member OUTSIDE

!
VRF context OUTSIDE
 Vni 1001000
 IP route 10.1.1.0/24 20.1.1.1 track 20


VTEPA# show track 10 Track 10
IP Route 20.1.1.1/32 Reachability Reachability is UP

VTEPA# show ip route 20.1.1.0/24 vrf INSIDE
IP Route Table for VRF "INSIDE"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
```

```
20.1.1.0/24, ubest/mbest: 1/0
  *via 10.1.1.1 [1/0], 00:00:08, static

Firewall Failure on VTEP A caused the track to go down causing VTEP A to withdraw the
static route.

VTEPA# show track 20 Track 20
IP Route 20.1.1.1/32 Reachability Reachability is DOWN

VTEPA# show ip route 20.1.1.0/24 vrf INSIDE
IP Route Table for VRF "RED"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

Route not found
```

ファブリックの残りの部分に配布される再帰静的ルート

このアプローチでは、内部または外部 VRF が存在する場所に静的ルートが設定されます。ネクスト ホップはホストルート（EVPN Route-Type2）を介して到達可能であるため、アクティブファイアウォールのスタンバイへの変更、およびその逆の変更はローカルでのみ行われ、他のVXLANファブリックにチャーンは発生しません。このアプローチは、拡張性の向上とコンバージェンスの向上に役立ちます。

任意の VTEP：

```
VRF context OUTSIDE
 Vni 1002000
 IP route 10.1.1.0/24 20.1.1.1
! static route on VTEP pointing to Firewall next hop
! firewall VIP 20.1.1.1

VRF context INSIDE
 Vni 1001000
 IP route 20.1.1.0/24 10.1.1.1
! static route on VTEP pointing to Firewall next hop
! firewall VIP 10.1.1.1
```

スタティック ルートを BGP に再配布し、残りのファブリックにアドバタイズする

再配布によって、示されているアクティブなファイアウォールへのルートを、それが存在するVTEP に作成します。ルートはプレフィックスルート（EVPN Route-Type5）と見なされ、アクティブなファイアウォールがあるVTEPへのルートのみが表示されます。ファイアウォールのアクティブ/スタンバイ変更の場合、トラッキングは変更を検出し、この変更をすべてのリモートVTEPに通知する必要があります。この動作は、ルートが「削除」され、その後に「追加」されることに相当します。このアプローチでは、VRFを使用してすべてのVTEPに通知する必要があるため、より大きなチャーンが見られます。

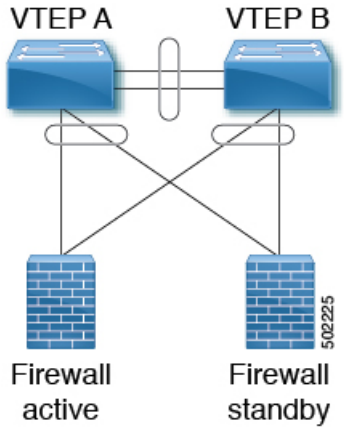
VTEP A および VTEP B:

```
router bgp 65000
 vrf OUTSIDE
  address-family ipv4 unicast
```

```
redistribute static route-map Static-to-BGP
```

静的ルーティングを使用するデュアル接続ファイアウォール

図 *57 :* 静的ルーティングを使用するデュアル接続ファイアウォール



VTEP A および VTEP B:

```
Vlan 10
 Name inside
 Vn-segment 10010

Vlan 20
 Name outside
 Vn-segment 10020

interface nve1
 no shutdown
 host-reachability protocol bgp
 source-interface loopback1
 member vni 10010
  mcastgroup 239.1.1.1
 member vni 10020
  mcastgroup 239.1.1.1
 member vni 1001000 associate-vrf
 member vni 1002000 associate-vrf


Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
 IP address 10.1.1.254/24
 fabric forwarding mode anycast-gateway

Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 fabric forwarding mode anycast-gateway

VRF context INSIDE
 Vni 1001000
 IP route 20.1.1.0/24 10.1.1.1
```

```
! static route on VTEP pointing to Firewall next hop
! firewall VIP 10.1.1.1
VRF context OUTSIDE
 Vni 1002000
 IP route 10.1.1.0/24 20.1.1.1
! static route on VTEP pointing to Firewall next hop
! firewall VIP 20.1.1.1


router bgp 65000
 vrf INSIDE
  address-family ipv4 unicast
   redistribute static route-map INSIDE-to-BGP
 vrf OUTSIDE
  address-family ipv4 unicast
   redistribute static route-map OUTSIDE-to-BGP
```

eBGP ルーティングを使用するシングル接続ドファイアウォール

ファイアウォールがBGPをサポートしている場合、1つのオプションは、ファイアウォールとサービス VTEP 間のプロトコルとして BGP を使用することです。エニーキャスト IP を使用したピアリングはサポートされていません。推奨される設計は、ループバックを使用して各VTEPおよびピアで専用ループバック IP を使用することです。ループバック インターフェイスがEVPN を介してアドバタイズされない限り、同じIPアドレスをすべての属するVTEPで使用できます。VTEP 単位で個々の IP アドレスを使用することを推奨します。

ファイアウォールからループバックへの到達可能性は、VTEP 上のエニーキャスト ゲートウェイ IP を指すファイアウォール上のスタティック ルートを使用して設定できます。

次の例では、AS 65000 にある VTEP と AS 65002 にあるファイアウォールから eBGP ピアリングが確立されます。iBGP との BGP ピアリングはサポートされていません。

（注） 異なる VTEP に接続されたアクティブ/スタンバイファイアウォールへの **export-gateway-ip** を有効にする必要があります。

BGP ピアリングにエニーキャスト ゲートウェイを使用しないでください。

VTEP A:

```
Vlan 10
 Name inside
 Vn-segment 10010

Vlan 20
 Name outside
 Vn-segment 10020


Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
IP address 10.1.1.254/24
fabric forwarding mode anycast-gateway
```

```
Interface loopback100
 Vrf member INSIDE
 Ip address 172.16.1.253/32

 Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 fabric forwarding mode anycast-gateway

 Interface loopback101
 Vrf member OUTSIDE
 Ip address 172.18.1.253/32


router bgp 65000
 vrf INSIDE
 ! peer with Firewall Inside
 neighbor 10.1.1.0/24 remote-as 65123
 update-source loopback100
 ebgp-multihop 5
 address-family ipv4 unicast
  local-as 65051 no-prepend replace-as

 vrf OUTSIDE
 ! peer with Firewall Outside
 neighbor 20.1.1.0/24 remote-as 65123
 update-source loopback101
 ebgp-multihop 5
 address-family ipv4 unicast
  local-as 65052 no-prepend replace-as
```

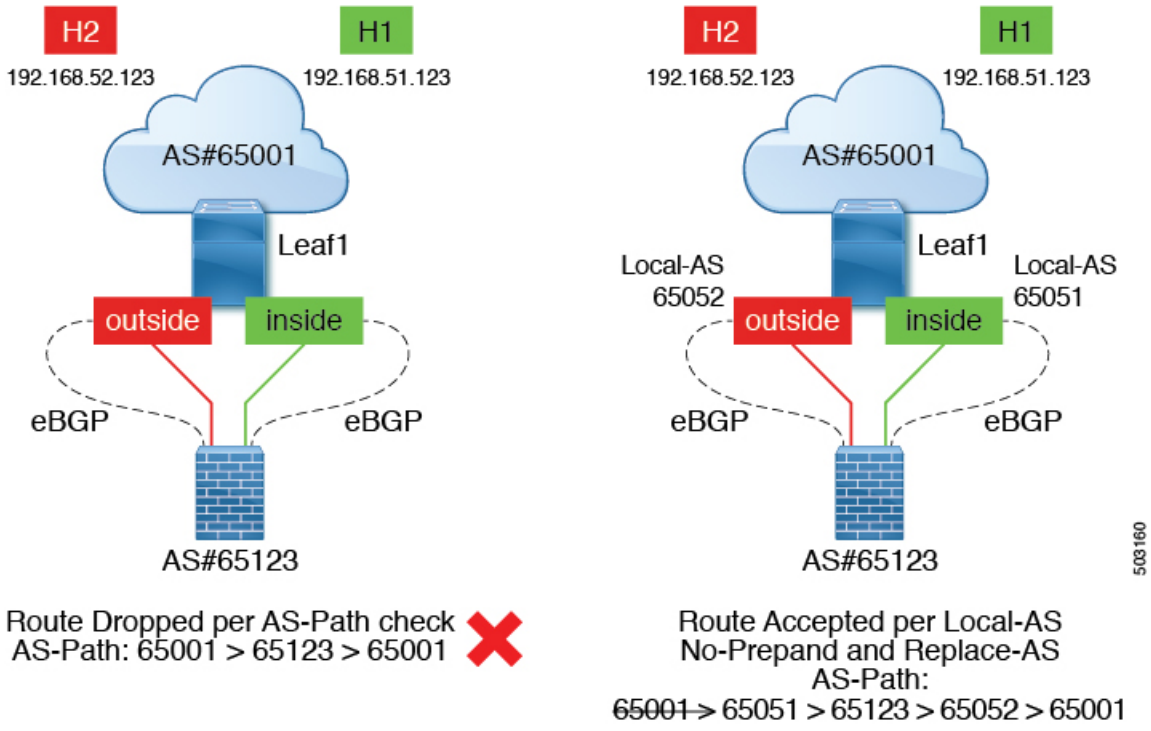
VTEP B :

```
Vlan 10
 Name inside
 Vn-segment 10010

Vlan 20
 Name outside
 Vn-segment 10020
Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
IP address 10.1.1.254/24
fabric forwarding mode anycast-gateway

Interface loopback100
 Vrf member INSIDE
 Ip address 172.16.1.254/32

 Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 fabric forwarding mode anycast-gateway

 Interface loopback101
 Vrf member OUTSIDE
 Ip address 172.18.1.254/32


router bgp 65000
 vrf INSIDE
```

```
! peer with Firewall Inside
neighbor 10.1.1.0/24 remote-as 65123
update-source loopback100
ebgp-multihop 5
address-family ipv4 unicast
 local-as 65051 no-prepend replace-as

vrf OUTSIDE
! peer with Firewall Outside
neighbor 20.1.1.0/24 remote-as 65123
update-source loopback101
ebgp-multihop 5
address-family ipv4 unicast
 local-as 65052 no-prepend replace-as
```

通常、VXLAN ファブリックは単一の BGP 自律システム（AS）内にあるため、内部 VRF と外部 VRF の AS は同じです。BGP は、自身の AS から受信したルートをインストールしません。したがって、このルールをオーバーライドするには、AS パスを調整する必要があります。BGP が自身の AS からルートをドロップするというルールを無効にするなど、さまざまなアプローチが存在します。これは、ネットワークにさらに影響を与えます。すべての BGP 保護メカニズムを維持するために、「local-as」アプローチでは、異なる AS から発信されたルートを模倣できます。VRF ごとに異なる「local-as」を持つ各ファイアウォール ピアリングに「local-as # ASN # no-prepend replace-as」を挿入することを推奨します。

図 *58 : eBGP AS-Path* チェック



eBGP ルーティングを使用するデュアル接続ファイアウォール

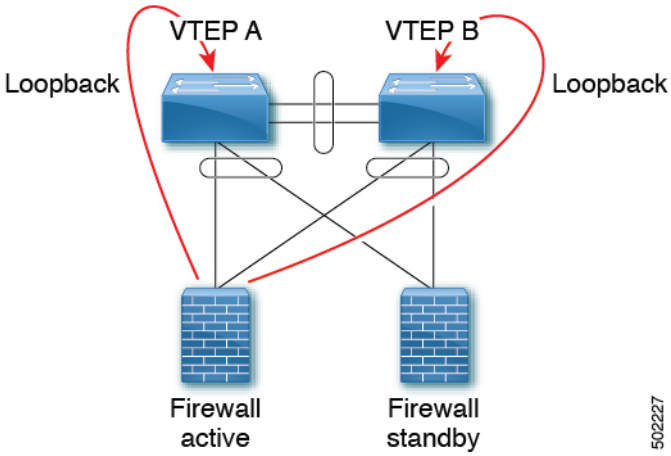
ファイアウォールがBGPをサポートしている場合、1つのオプションは、ファイアウォールとサービス VTEP 間のプロトコルとして BGP を使用することです。エニーキャスト IP を使用したピアリングはサポートされていません。推奨される設計は、ループバックを使用して各VTEPおよびピアで専用ループバック IP を使用することです。ループバック インターフェイスがEVPNを介してアドバタイズされない限り、同じIPアドレスをすべての属するVTEPで使用できます。VTEP 単位で個々のIPアドレスを使用することを推奨します。vPC 環境の場合は必須です。

ファイアウォールからループバックへの到達可能性は、VTEP 上のエニーキャスト ゲートウェイ IP を指すファイアウォール上のスタティック ルートを使用して設定できます。

vPC 導入では、vPC ピアリンクを介した VRF ごとのピアリングが必要です。VRF 単位のピアリングに加えて、**advertise-pip** コマンドを使用してプレフィックス ルートのアドバタイズメント（EVPN ルート タイプ 5）を有効にできます。ファブリック ピアリングを使用する vPC の場合、VRF ごとのピアリングは必要なく、プレフィックス ルートのアドバタイズメント（EVPN Route-Type5）が必要です。

次の例では、AS 65000 にある VTEP と AS 65002 にあるファイアウォールから eBGP ピアリングが確立されます。iBGP との BGP ピアリングはサポートされていません。

図 *59 : eBGP* を使用したデュアル接続ファイアウォール



（注） 異なる VTEP に接続されたアクティブ/スタンバイファイアウォールへの **export-gateway-ip** を有効にする必要があります。

BGP ピアリングにエニーキャスト ゲートウェイを使用しないでください。

VTEP A:

```
Vlan 10
 Name inside
 Vn-segment 10010
```

```
Vlan 20
 Name outside
 Vn-segment 10020


Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
IP address 10.1.1.254/24
fabric forwarding mode anycast-gateway

Interface loopback100
 Vrf member INSIDE
 Ip address 172.16.1.253/32

 Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 fabric forwarding mode anycast-gateway

 Interface loopback101
 Vrf member OUTSIDE
 Ip address 172.18.1.253/32


router bgp 65000
vrf INSIDE
 ! peer with Firewall Inside
 neighbor 10.1.1.0/24 remote-as 65123
 update-source loopback100
 ebgp-multihop 5
 address-family ipv4 unicast
  local-as 65051 no-prepend replace-as

 vrf OUTSIDE
 ! peer with Firewall Outside
 neighbor 20.1.1.0/24 remote-as 65123
 update-source loopback101
 ebgp-multihop 5
 address-family ipv4 unicast
  local-as 65052 no-prepend replace-as
```

VTEP B :

```
Vlan 10
 Name inside
 Vn-segment 10010

Vlan 20
 Name outside
 Vn-segment 10020


Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
IP address 10.1.1.254/24
fabric forwarding mode anycast-gateway

Interface loopback100
 Vrf member INSIDE
 Ip address 172.16.1.254/32
```

```
 Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 fabric forwarding mode anycast-gateway

 Interface loopback101
 Vrf member OUTSIDE
 Ip address 172.18.1.254/32


router bgp 65000
 vrf INSIDE
 ! peer with Firewall Inside
 neighbor 10.1.1.0/24 remote-as 65123
 update-source loopback100
 ebgp-multihop 5
 address-family ipv4 unicast
  local-as 65051 no-prepend replace-as

 vrf OUTSIDE
 ! peer with Firewall Outside
 neighbor 20.1.1.0/24 remote-as 65123
 update-source loopback101
 ebgp-multihop 5
 address-family ipv4 unicast
  local-as 65052 no-prepend replace-as
```

vPC ピアリングによる Per-VRF ピアリング

VTEP A および VTEP B:

```
vlan 3966
! vlan use for peering between the vPC VTEPS

vlan 3967
! vlan use for peering between the vPC VTEPS

system nve infra-vlans 3966,3967

interface vlan 3966
vrf memner INSIDE
 ip address 100.1.1.1/31

interface vlan 3967
 vrf memner OUTSIDE
 ip address 100.1.2.1/31

router bgp 65000
 vrf INSIDE
 neighbor 100.1.1.0 remote-as 65000
 update-source vlan 3966
 next-hop self
 address-family ipv4 unicast

 vrf OUTSIDE
 neighbor 100.1.2.0 remote-as 65000
 update-source vlan 3967
 next-hop self
 address-family ipv4 unicast
```

各 VRF で学習されたルートは、BGP EVPN 更新を介してファブリックの残りの部分にアドバタイズされます。

OSPF を使用したシングル接続ファイアウォール

次の例は、ファイアウォールで OSPF ピアリングを実行している VTEP A からの設定スニペットを示しています。

SVI は、内部および外部の両方の VRF の VTEP で定義されます。これらの各 VRF 上のファイアウォールを持つ VTEP ピアは、1 つの VRF から別の VRF に移動するためのルーティング情報を動的に学習します。

VTEP A および VTEP B:

```
vlan 10
 name inside
 vn-segment 10010

vlan 20
 name outside
 vn-segment 10020

interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
 IP address 10.1.1.254/24
 IP router ospf 1 area 0
 fabric forwarding mode anycast-gateway

Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 IP router ospf 1 area 0
 fabric forwarding mode anycast-gateway

interface nve1
 no shutdown
 host-reachability protocol bgp
 source-interface loopback1
 member vni 10010
  mcastgroup 239.1.1.1
 member vni 10020
  mcastgroup 239.1.1.1
 member vni 1001000 associate-vrf
 member vni 1002000 associate-vrf

router ospf 1
 router-id 192.168.1.1
  vrf INSIDE
  VRF OUTSIDE


VTEPA# show ip route ospf-1 vrf OUTSIDE
 IP Route Table for VRF "OUTSIDE"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
```

```
10.1.1.0/24, ubest/mbest: 1/0
  *via 20.1.1.1 Vlan20, [110/41], 1w5d, ospf-1, intra

VTEPA# show ip route ospf-1 vrf INSIDE
IP Route Table for VRF "INSIDE"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

20.1.1.0/24, ubest/mbest: 1/0
  *via 10.1.1.1 Vlan10, [110/41], 1w5d, ospf-1, intra
```

次に、このルートはBGPに再配布され、EVPNファブリックを介してアドバタイズされます。これにより、他のすべての VTEP が、ネクスト ホップとして VTEP A をポイントする各 VRF 内のすべてのルートを持つようになります。

OSPF ルートを BGP に再配布し、残りのファブリックにアドバタイズする

VTEP A および VTEP B:

```
router bgp 65000
 vrf OUTSIDE
  address-family ipv4 unicast
   redistribute ospf 1 route-map OUTSIDEOSPF-to-BGP
 vrf INSIDE
  address-family ipv4 unicast
   redistribute ospf 1 route-map INSIDEOSPF-to-BGP


VTEPA# show ip route 10.1.1.0/24 vrf OUTSIDE

10.1.1.0/24 ubest/mbest: 1/0
  *via 10.1.1.18%default, [200/41], 1w1d, bgp-65000, internal, tag 65000 (evpn) segid:
200100 tunnelid: 0xa010112 encap: VXLAN
```

トラフィックは、VTEP からサービス VTEP にカプセル化された VXLAN であり、カプセル化解除されてファイアウォールに送信されます。ファイアウォールはルールを適用し、トラフィックを内部 VRF のサービス VTEP に送信します。このトラフィックは VXLAN でカプセル化され、宛先 VTEP に送信されます。宛先 VTEP では、トラフィックがカプセル化解除されてエンド クライアントに送信されます。

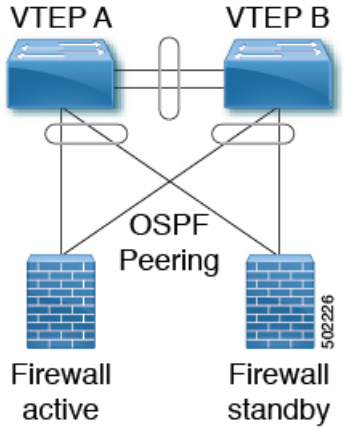
ファイアウォール フェールオーバー

アクティブ ファイアウォールに障害が発生し、スタンバイ ファイアウォールが引き継ぐと、ルートはサービス VTEP A から取り消され、サービス VTEP B によってファブリックにアドバタイズされます。

OSPF を使用したデュアル接続ファイアウォール

Cisco NX-OS は、レイヤ 3 を使用した vPC 経由のダイナミック OSPF ピアリングをサポートします。これにより、vPCを使用したファイアウォール接続が可能になり、このリンク上でOSPFピアリングが確立されます。Cisco Nexus 9000 スイッチとファイアウォール間のピアリングを確立するために使用される VLAN は、非 VXLAN 対応 VLAN である必要があります。

図 *60 : OSPF* を使用したデュアル接続ファイアウォール



（注） OSPF 隣接にはエニーキャスト ゲートウェイを使用しないでください。

VTEP A:

```
Vlan 10
 Name inside

Vlan 20
 Name outside

Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
 IP address 10.1.1.253/24
 Ip router ospf 1 area 0

Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.253/24
 Ip router ospf 1 area 0

vpc domain 100
 layer3 peer-router
 peer-gateway
 peer-switch
 peer-keepalive destination x.x.x.x source x.x.x.x peer-gateway
 ipv6 nd synchronize
 ip arp synchronize

router ospf 1
 vrf INSIDE VRF OUTSIDE
```

VTEP B :

```
Vlan 10
 Name inside

Vlan 20
 Name outside
```

```
Interface VLAN 10
 Description inside_vlan
 VRF member INSIDE
 IP address 10.1.1.254/24
 Ip router ospf 1 area 0

Interface VLAN 20
 Description outside_vlan
 VRF member OUTSIDE
 IP address 20.1.1.254/24
 Ip router ospf 1 area 0

vpc domain 100
 layer3 peer-router
 peer-gateway
 peer-switch
 peer-keepalive destination x.x.x.x source x.x.x.x peer-gateway
 ipv6 nd synchronize
 ip arp synchronize

router ospf 1
 vrf INSIDE VRF OUTSIDE


VTEPA# show ip route ospf-1 vrf OUTSIDE
IP Route Table for VRF "OUTSIDE"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.1.1.0/24, ubest/mbest: 1/0
  *via 20.1.1.1 Vlan20, [110/41], 1w5d, ospf-1, intra

VTEPA# show ip route ospf-1 vrf INSIDE
IP Route Table for VRF "INSIDE"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

20.1.1.0/24, ubest/mbest: 1/0
  *via 10.1.1.1 Vlan10, [110/41], 1w5d, ospf-1, intra
```

OSPF ルートを BGP に再配布し、残りのファブリックにアドバタイズする

VTEP A および VTEP B:

```
router bgp 65000
 vrf OUTSIDE
  address-family ipv4 unicast
   redistribute ospf 1 route-map OUTSIDEOSPF-to-BGP
 vrf INSIDE
  address-family ipv4 unicast
   redistribute ospf 1 route-map INSIDEOSPF-to-BGP
```

デフォルト ゲートウェイとしてのファイアウォール

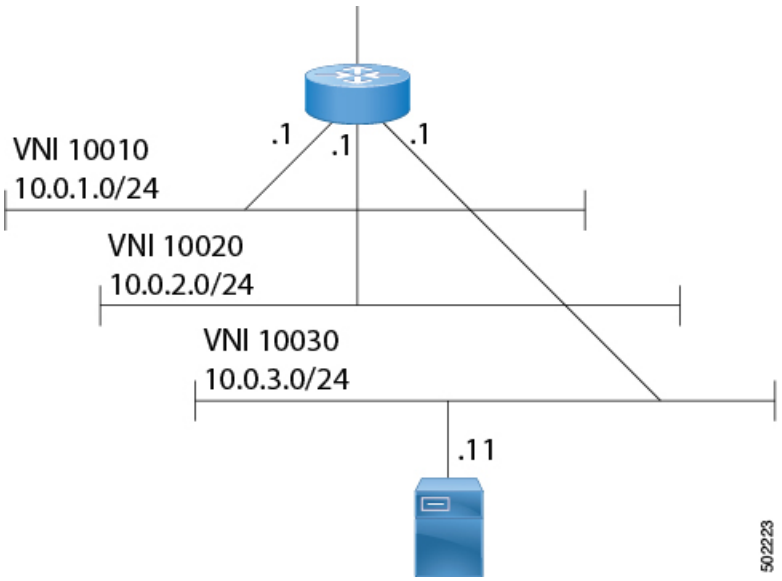
この導入モデルでは、VXLAN ファブリックはレイヤ 2 ファブリックであり、デフォルト ゲートウェイはファイアウォール上にあります。

次に例を示します。

```
vlan 10
  name WEB
  vn-segment 10010
vlan 20
  name APPLICATION
  vn-segment 10020
vlan 30
  name DATABASE
  vn-segment 10030

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 10010
   mcastgroup 239.1.1.1
  member vni 10020
   mcastgroup 239.1.1.1
  member vni 10030
   mcastgroup 239.1.1.1
```

ファイアウォールは、各 VNI に論理インターフェイスを持ち、すべてのエンドポイントのデフォルト ゲートウェイです。すべての VNI 間通信はファイアウォールを通過します。ファイアウォールがボトルネックにならないように、ファイアウォールのサイジングには特に注意してください。したがって、この設計は、低帯域幅要件の環境で使用してください。

図 *61* : レイヤ *2 VXLAN* ファブリックを使用したデフォルト ゲートウェイとしてのファイアウォール



トランスペアレント ファイアウォール挿入

トランスペアレント ファイアウォールまたはレイヤ 2 ファイアウォール（IPS/IDS を含む）は、通常、内部 VLAN と外部 VLAN をブリッジし、トラフィックが通過するときに検査します。VLAN スティッチングは、サービスのデフォルト ゲートウェイを内部 VLAN に配置することによって行われます。このゲートウェイへのレイヤ 2 の到達可能性は、外部 VLAN で行われます。

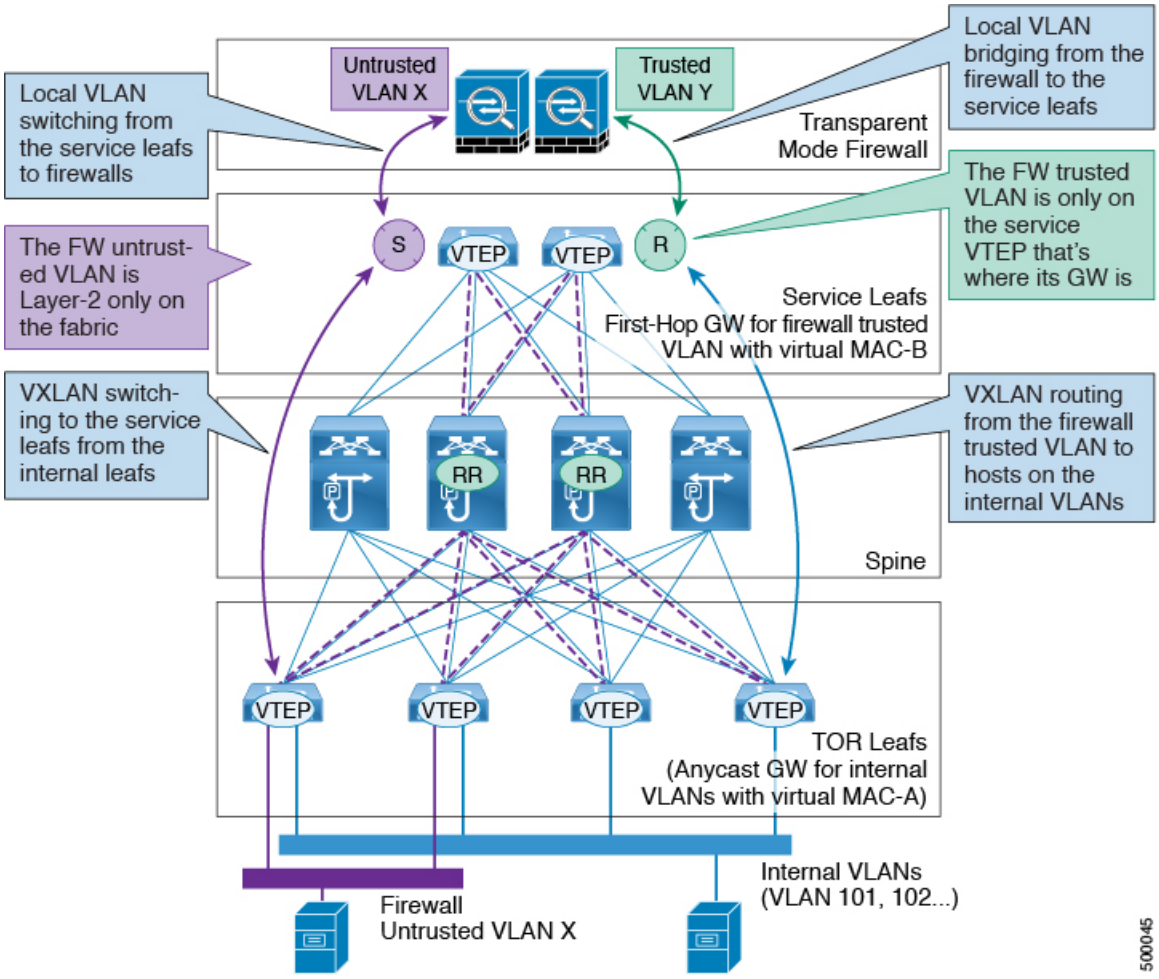
EVPN でのトランスペアレント ファイアウォール挿入の概要

トポロジには、次のタイプの VLAN が含まれます。

- 内部 VLAN（通常の VXLAN を ToR リーフにエニーキャスト ゲートウェイ付きで配置）
- ファイアウォール非信頼 VLAN X
- ファイアウォール信頼 VLAN Y

このトポロジにおいて、VLAN X から他の VLAN へのトラフィックは、サービス リーフに接続されているトランスペアレント レイヤ2ファイアウォールを経由する必要があります。このトポロジは、信頼できない VLAN X と信頼できる VLAN Y のアプローチを使用します。すべての ToR リーフにはレイヤ 2 VNI VLAN X があります。VLANX の SVI はありません。ファイアウォールに接続されているサービス リーフにはレイヤ 2 VNI VLAN X 、非 VXLAN VLAN Y、および HSRP ゲートウェイを使用する SVIY があります。

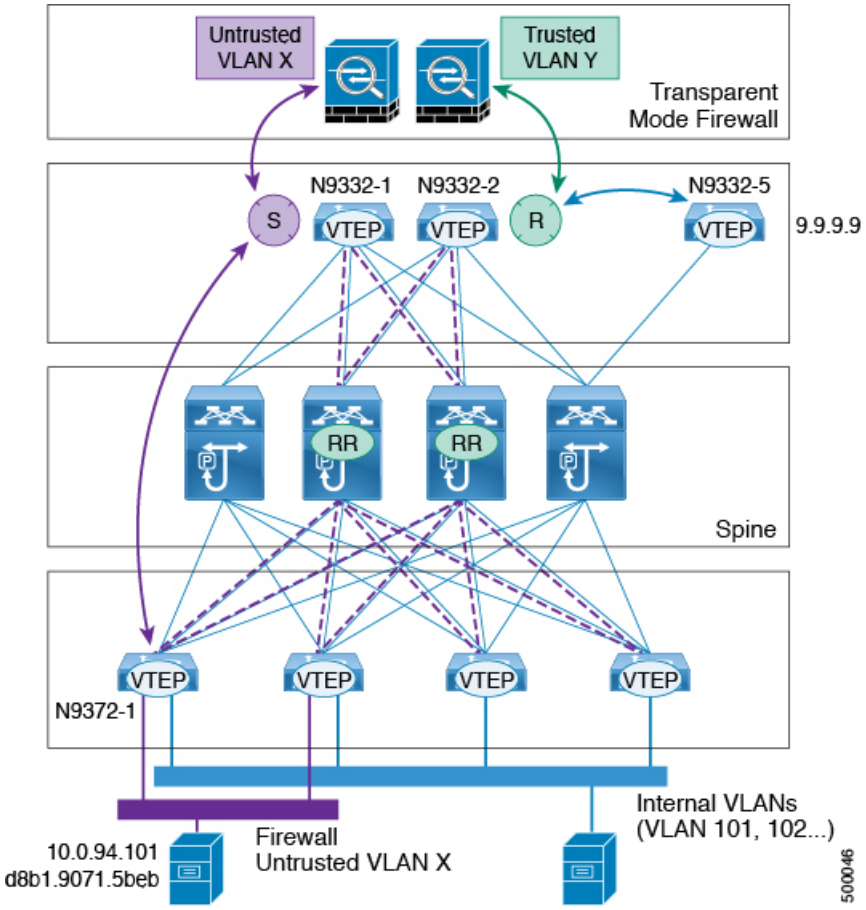
EVPN でのトランスペアレント ファイアウォール挿入の概要



（注） VXLAN EVPN の場合、トランスペアレント ファイアウォールを挿入した分散型エニーキャスト ゲートウェイを使用することを推奨します。これにより、すべての VLAN を VXLAN 対応にできます。HSRP/VRRP ベースのファーストホップ ゲートウェイを使用する場合、SVI の VLAN は VXLAN 対応にできず、冗長性のために vPC ペア上に存在する必要があります。

EVPN でのトランスペアレント ファイアウォール挿入の例

EVPN でのトランスペアレント ファイアウォール挿入の例



- VLAN X のホスト: 10.1.94.101
- ToR リーフ: N9372-1
- vPC 中のサービス リーフ: N9332-1 および N9332-2
- ボーダー リーフ : N9332-5

ToR リーフ設定

```
vlan 94
vn-segment 100094

interface nve1
  member vni 100094
   mcastgroup 239.1.1.1

router bgp 64500
  routerid 1.1.2.1
  neighbor 1.1.1.1 remote-as 64500
  address-family l2vpn evpn
```

```
    send-community extended
  neighbor 1.1.1.2 remote-as 64500
  address-family l2vpn evpn
    send-community extended
  vrf Ten1
   address-family ipv4 unicast
     advertise l2vpn evpn

evpn
vni 100094 l2
  rd auto
  route-target import auto
  route-target export auto
```

HSRP を使用したサービス リーフ 1 設定

```
vlan 94
description untrusted_vlan
  vn-segment 100094

vlan 95
  description trusted_vlan

vpc domain 10
  peer-switch
  peer-keepalive destination 10.1.59.160
  peer-gateway
  auto-recovery
  ip arp synchronize

interface Vlan2
description vpc_backup_svi_for_overlay
  no shutdown
  no ip redirects
  ip address 10.10.60.17/30
  no ipv6 redirects
  ip router ospf 100 area 0.0.0.0
  ip ospf bfd
  ip pim sparsemode

interface Vlan95
description SVI_for_trusted_vlan
  no shutdown
  mtu 9216
  vrf member Ten-1
  no ip redirects
  ip address 10.0.94.2/24
  hsrp 0
   preempt priority 255
   ip 10.0.94.1

interface nve1
  member vni 100094
   mcast-group 239.1.1.1

router bgp 64500
  routerid 1.1.2.1
  neighbor 1.1.1.1 remote-as 64500
  address-family l2vpn evpn
   send-community extended
  neighbor 1.1.1.2 remote-as 64500
   address-family l2vpn evpn
    send-community extended
  vrf Ten-1
```

```
   address-family ipv4 unicast
    network 10.0.94.0/24 /*advertise /24 for SVI 95 subnet; it is not VXLAN anymore*/
    advertise l2vpn evpn

evpn
vni 100094 l2
 rd auto
 route-target import auto
 route-target export auto
```

HSRP を使用したサービス リーフ 2 設定

```
vlan 94
  description untrusted_vlan
  vnsegment 100094

vlan 95
  description trusted_vlan

vpc domain 10
  peer-switch
  peer-keepalive destination 10.1.59.159
  peer-gateway
  auto-recovery
  ip arp synchronize

interface Vlan2
description vpc_backup_svi_for_overlay
  no shutdown
  no ip redirects
  ip address 10.10.60.18/30
  no ipv6 redirects
  ip router ospf 100 area 0.0.0.0
  ip pim sparsemode

interface Vlan95
  description SVI_for_trusted_vlan
  no shutdown
  mtu 9216
  vrf member Ten-1
  no ip redirects
  ip address 10.0.94.3/24
  hsrp 0
   preempt priority 255
   ip 10.0.94.1

interface nve1
  member vni 100094
   mcastgroup 239.1.1.1

router bgp 64500
  router-id 1.1.2.1
  neighbor 1.1.1.1 remote-as 64500
  address-family l2vpn evpn
    send-community extended
  neighbor 1.1.1.2 remote-as 64500
  address-family l2vpn evpn
   send-community extended
  vrf Ten-1
    address-family ipv4 unicast
     network 10.0.94.0/24 /*advertise /24 for SVI 95 subnet; it is not VXLAN anymore*/
     advertise l2vpn evpn

evpn
```

```
vni 100094 l2
  rd auto
  route-target import auto
  route-target export auto
```

show コマンドの例

入力リーフが学習したホストからのローカル MAC の情報を表示します。

```
switch# sh mac add vl 94 | i 5b|MAC
* primary entry, G - Gateway MAC, (R) Routed - MAC, O - Overlay MAC
VLAN MAC Address Type age Secure NTFY Ports
* 94 d8b1.9071.5beb dynamic 0 F F Eth1/1
```

サービス リーフが検出したホストの MAC の情報を表示します。

（注） VLAN 94 において、サービス リーフが学習するホスト MAC は、BGP によってリモート ピアから得られます。

```
switch# sh mac add vl 94 | i VLAN|eb

VLAN MAC Address Type age Secure NTFY Ports
* 94 d8b1.9071.5beb dynamic 0 F F nve1(1.1.2.1)

switch#  sh mac add vl 94 | i VLAN|eb

VLAN MAC Address Type age Secure NTFY Ports
* 94 d8b1.9071.5beb dynamic 0 F F nve1(1.1.2.1)

switch#  sh mac add vl 95 | i VLAN|eb

VLAN MAC Address Type age Secure NTFY Ports
+ 95 d8b1.9071.5beb dynamic 0 F F Po300

switch#  sh mac add vl 95 | i VLAN|eb

VLAN MAC Address Type age Secure NTFY Ports
+ 95 d8b1.9071.5beb dynamic 0 F F Po300
```

サービス リーフが学習した VLAN 95 にあるホストの ARP の情報を表示します。

```
switch# sh ip arp vrf ten-1
Address     Age      MAC Address     Interface
10.0.94.101 00:00:26 d8b1.9071.5beb  Vlan95
```

サービス リーフは EVPN から 9.9.9.9 を学習します。

```
switch# sh ip route vrf ten-1 9.9.9.9
IP Route Table for VRF "Ten-1"
'*' denotes best ucast nexthop
'**' denotes best mcast nexthop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

9.9.9.9/32, ubest/mbest: 1/0
```

```
  *via 1.1.2.7%default, [200/0], 02:57:27, bgp64500,internal, tag 65000 (evpn) segid:
10011
tunnelid: 0x1
010207 encap: VXLAN
```

ボーダー リーフが学習した BGP によるホスト ルートの情報を表示します。

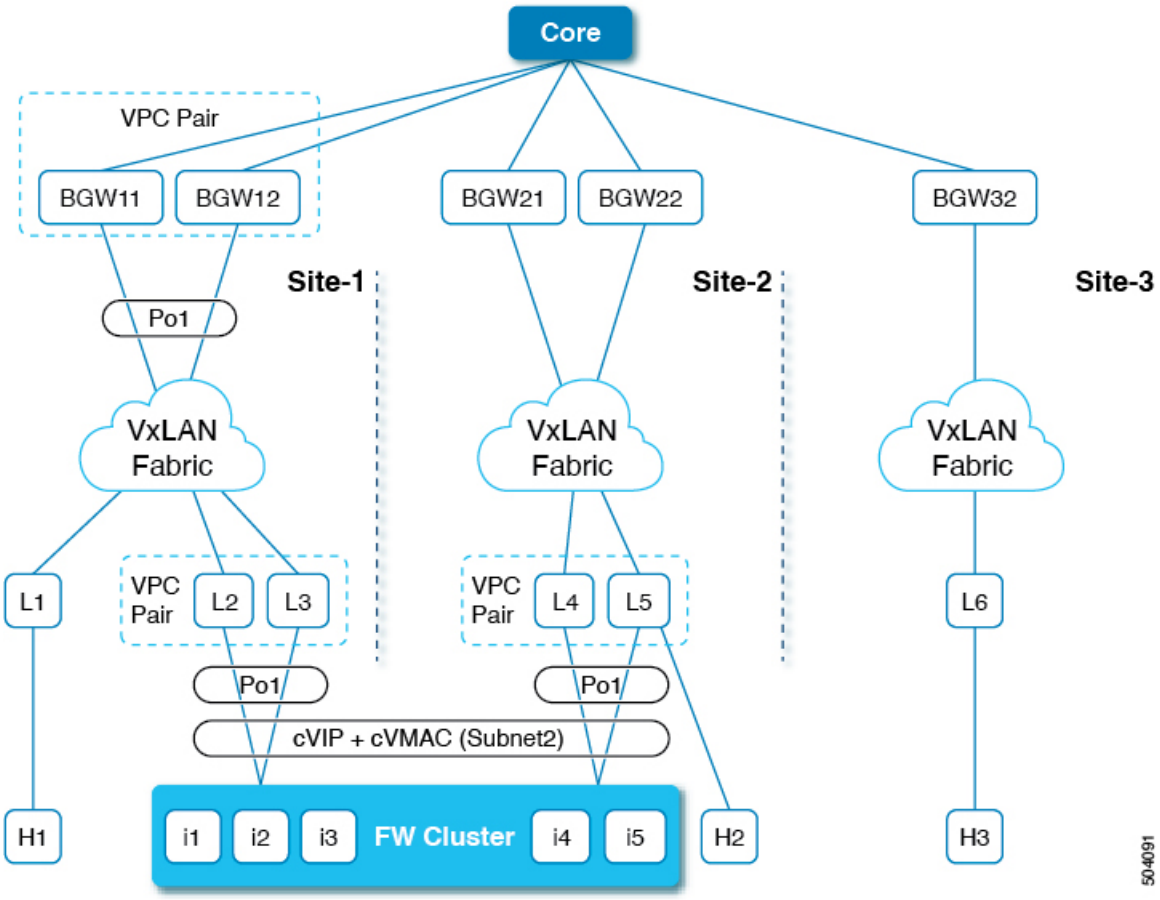
```
switch# sh ip route 10.0.94.101

IP Route Table for VRF "default"
'*' denotes best ucast nexthop
'**' denotes best mcast nexthop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.0.94.0/24, ubest/mbest: 1/0
  *via 10.100.5.0, [20/0], 03:14:27, bgp65000,external, tag 6450
```

VXLAN BGP EVPN を使用したファイアウォール クラスタリング

このセクションでは、BGP EVPN コントロール プレーンを使用して VXLAN ファブリックを実行している複数のサイトにまたがるファイアウォール クラスタを構成する方法について詳しく説明します。

次のトポロジは、VXLAN EVPN を使用したファイアウォール クラスタリングを示しています。

図 *62 : VXLAN EVPN* によるファイアウォール クラスタリング



このトポロジは、次のものをカバーします。

- ファイアウォール クラスタは、単一デバイスとして動作する複数のインスタンスで構成されています。
- ファイアウォールへのルーテッドアクセスは、異なるサブネットまたは同じサブネットを介して行うことができます。
- ファイアウォールは、すべてのインスタンスにまたがる L2 ポート チャネルを採用しています。
- 共通の ESI では、ファイアウォール クラスタに接続するすべての vPC ポートチャネルが示されます。
- すべてのインスタンスに単一の VIP/VMAC が存在します。
- サイトごとの BGP-EVPN VXLAN オーバーレイは、ボーダー ゲートウェイでステッチされます。

- 同じサイト内のアクティブからアクティブへのインスタンスのエニーキャスト転送と、トラフィック フローのためのサイト全体のファイアウォールへのアクティブからバックアップへのアクセスがサポートされています。
- 各サイトには、ポートチャネルインターフェイスが割り当てられたクラスタに接続された単一の vPC ペアがあります。
- クラスタ VIP およびクラスタ VMAC は、BGP EVPN ルート ターゲット -2s として VXLAN EVPN ファブリックにアドバタイズされます（ESI は各 vPC のポート チャネル インターフェイスで構成された値に設定されます）。ルート ターゲット 2 のネクスト ホップは、vPC ペアの VTEP VIP アドレスです。
- 各サイトには複数のクラスターが含まれる場合があります。クラスタは、固有の ESI を持つ個々のポートチャネルを使用して vPC ペアに接続されます。
- 各クラスタには、BGP EVPN ルート ターゲット -2s として VXLAN EVPN ファブリックにアドバタイズされる独自の cVIP と cVMAC があります（ESI はその vPC のポートチャネル インターフェイスで構成された値に設定されています）。
- クラスタには、vPC ペアに接続されたポート チャネル上に複数の VLAN がある場合があります。VLAN で学習された各 cVIP/cVMAC は、対応する L2VNI を使用してルート T-2 EVPN ルートとしてアドバタイズされます。
- VIP および VMAC（ファイアウォール ホスト）は、単一の spanned Ether-channel に接続されます。
- Spanned Ether-channel はサイト全体に拡張されます。
- VIP へのエニーキャスト転送は、既存の BGP パス属性と最適パスの選択を利用して決定されます。

ファイアウォール クラスタに接続されている VTEP リーフでは、BGP はルート マップを使用してコミュニティをファイアウォール クラスタ関連の EVPN EAD/ES（タイプ 1）および MAC/IP（タイプ 2）ルートに接続します。

```
router bgp 12000
 address-family l2vpn evpn
originate-map set_esi
 template peer SITE-BGW
   remote-as 12000
   update-source loopback1
   address-family l2vpn evpn
     send-community
     send-community extended
 template peer VTEP-PEERS
   remote-as 12000
   update-source loopback1
   address-family l2vpn evpn
     send-community
     send-community extended
```

ボーダー ゲートウェイでは、BGP はルート マップを使用して、EVPN EAD/ES（タイプ 1）および MAC/IP（タイプ 2）ルートに接続されたファイアウォール クラスタリング コミュニティを照合します。

```
router bgp 11000
  bestpath as-path multipath-relax
  neighbor 111.111.10.1 remote-as 12000
   peer-type fabric-external
   address-family l2vpn evpn
     send-community
     send-community extended
     route-map preserve_esi out
     rewrite-evpn-rt-asn
```

ファイアウォール クラスタに接続されている VTEP リーフで、コミュニティをファイアウォール クラスタ関連の EVPN EAD/ES（タイプ 1）および MAC/IP（タイプ 2）ルートに接続するようにルート マップを構成する必要があります。

```
route-map set_esi permit 10
  match tag 100000
  match evpn route-type 1 2
  set community 23456:12345
route-map set_esi permit 15
```

⚠

注意 ネイバー アドレス ファミリ モードの下の route-map <*name*> 外 BGP コマンドに関連付けられているルートマップの **match tag** コマンドは、 address-family l2vpn evpn の下で構成されている場合のみサポートされます。

ボーダー ゲートウェイでは、EVPN EAD/ES（タイプ 1）および MAC/IP（タイプ 2）ルートに接続されたファイアウォール クラスタリング コミュニティと一致するように、ファブリック内部ピアとファブリック外部ピアに個別のルート マップを構成する必要があります。

アウトバウンド L2VPN/EVPN ルート マップをファブリック内部ピアに一致させる：

```
route-map preserve_esi permit 10
  match community preserve_esi
  match evpn route-type 2
  set esi unchanged
route-map preserve_esi permit 15
route-map preserve_esi permit 30
```

アウトバウンド L2VPN/EVPN ルート マップをファブリック外部ピアに一致させる：

```
route-map preserve_esi_external permit 10
  match community preserve_esi
  match evpn route-type 2
  set esi unchanged
route-map preserve_esi_external permit 15
  match community preserve_esi
  match evpn route-type 1
route-map preserve_esi_external permit 20
  match evpn route-type 1
  match route-type local
route-map preserve_esi_external deny 25
  match evpn route-type 1
route-map preserve_esi_external permit 30
```

イーサネット セグメントは、vPC ポート チャネルの下でのみ構成できます。

```
interface port-channel 100
  ethernet-segment vpc
  esi <esi> [ tag <uint >]
interface port-channel 200
  ethernet-segment vpc
  esi system-mac <system-mac> <local-identifier> [tag <uint>]
```

共通の ESI では、ファイアウォール クラスタに接続するすべての vPC ポートチャネルが示されます。vPC ポート チャネルで ESI を構成できます。

```
evpn esi multihoming
port-channel 100
  ethernet-segment 1
     system-mac aa.bb.cc <anycast-host>
```

同じファイアウォール クラスタをホストするすべての vPC ポート チャネルに対して、同じシステム MAC を維持します。

ファイアウォールの詳細については、「VXLAN ファブリックでのレイヤ 3 ファイアウォールの統合」を参照してください。

VXLAN EVPN ファブリックのサービス リダイレクト

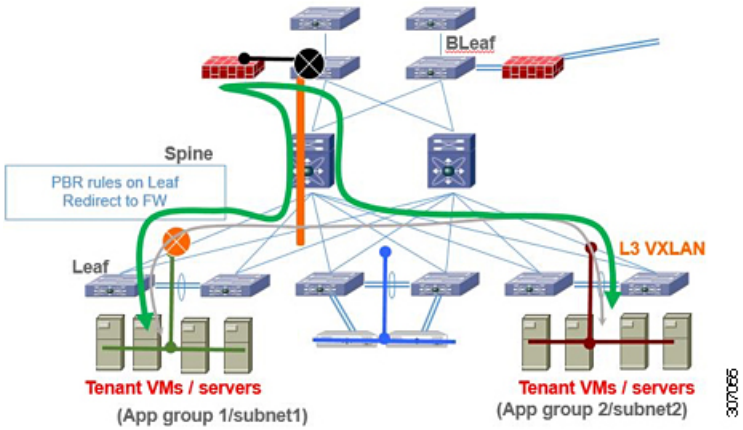
今日では、データセンター内のアプリケーションを保護および最適化するために、ファイアウォール、ロードバランサなどのサービス アプライアンス（サービス ノードまたはサービス エンドポイントとも呼ばれる）の挿入が必要です。このセクションでは、VXLAN EVPN ファブリックで提供されるレイヤ 4 ～ レイヤ 7 サービスの挿入およびリダイレクト機能について説明します。これらのサービスにトラフィックをオンボードして選択的にリダイレクトする高度なメカニズムを提供します。

サービス挿入のポリシーベース リダイレクトの使用

ポリシーベースのリダイレクト（PBR）は、ルーティング テーブル ルックアップをバイパスし、VXLAN 経由で到達可能なネクスト ホップ IP にトラフィックをリダイレクトするメカニズムを提供します。この機能により、ファイアウォールやロード バランサなどのレイヤ 4-レイヤ 7 デバイスへのサービス リダイレクションが可能になります。

PBR では、トラフィックの転送先を指定するルールを使用してルート マップを設定します。ルート マップは、テナント側の SVI に適用され、ホスト側のインターフェイスからファブリック経由で到達可能なネクスト ホップへのトラフィックに影響を与えます。

トラフィックがオーバーレイから VTEP に着信し、別のネクスト ホップにリダイレクトする必要があるシナリオでは、レイヤ 3 VNI インターフェイスに面するファブリックに PBR ポリシーを適用する必要があります。



前の図では、アプリケーション グループ 1 とアプリケーション グループ 2 間の通信は、デフォルトでテナント VRF のVLAN 間/VNI ルーティングを介して行われます。アプリケーション グループ 1 からアプリケーション グループ 2 へのトラフィックがファイアウォールを通過する必要があるという要件がある場合、PBR ポリシーを使用してトラフィックをリダイレクトできます。「ポリシーベース リダイレクトの構成例」のセクションの例では、トラフィック フローをリダイレクトするために必要な構成が示されています。

この VXLAN PBR 機能は非常に基本的なものであり、VXLAN ファブリックにサービスを適切に挿入するために必要な機能の多くが不足しています。したがって、「Enhanced-Policy Based Redirect（ePBR） （583 ページ）」セクションで説明されているすべての理由から、代わりに ePBR を確認することをお勧めします。

ポリシーベースのリダイレクトの注意事項と制約事項

PBR over VXLAN には、次の注意事項と制限事項が適用されます。

- 次のプラットフォームは、PBR over VXLAN をサポートしています。
  - Cisco Nexus 9332C および 9364C スイッチ
  - Cisco Nexus 9300-EX スイッチ
  - Cisco Nexus 9300-FX/FX2/FX3 スイッチ
  - Cisco Nexus 9300-GX スイッチ
  - Cisco Nexus 9300-GX2 スイッチ
  - Cisco Nexus 9332D-H2R スイッチ
  - Cisco Nexus 93400LD-H1 スイッチ
  - Cisco Nexus 9364C-H1 スイッチ
  - -EX/FX ライン カードを搭載した Cisco Nexus 9504 および 9508 スイッチ
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN PBR 機能は、すべての TOR スイッチの VXLANv6 でサポートされます。

- PBR over VXLAN は、**set {ip | ipv6} next-hop***ip-address* コマンドの VTEP ECMP、および **load-share** キーワードをサポートしていません。

ポリシーベース リダイレクト機能のイネーブル化

高度な（および推奨される）ePBR 機能が展開されていない場合に基本的な PBR を構成するには、次のセクションを参照してください。

- ポリシーベース リダイレクト機能のイネーブル化 （579 ページ）
- ルート ポリシーの設定 （580 ページ）
- ポリシーベース リダイレクトの設定の確認 （581 ページ）
- ポリシー ベース リダイレクトの設定例 （582 ページ）

始める前に

ルート ポリシーを設定するには、あらかじめポリシーベース リダイレクト機能をイネーブル化しておく必要があります。

手順の概要

1. **configure terminal**
2. **[no] feature pbr**
3. （任意） **show feature**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**[no] feature pbr** 例： `switch(config)# feature pbr`	ポリシーベース ルーティング機能をイネーブルにします。
ステップ 3	（任意） **show feature** 例： `switch(config)# show feature`	有効および無効にされた機能を表示します。

	コマンドまたはアクション	目的
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	この設定変更を保存します。

ルート ポリシーの設定

ポリシーベース ルーティングでルートマップを使用すると、着信インターフェイスにルーティング ポリシーを割り当てることができます。Cisco NX-OS はネクスト ホップおよびインターフェイスを検出するときに、パケットをルーティングします。

（注） スイッチには、IPv4 トラフィック用の RACL TCAM リージョンがデフォルトで用意されています。

始める前に

ポリシーベース ルーティング ポリシーを適用するには、あらかじめ RACL TCAM リージョンを（TCAM カービングを使用して）設定する必要があります。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide, Release 9.2(x)』の「Configuring ACL TCAM Region Sizes」の項を参照してください。

手順の概要

1. **configure terminal**
2. **interface** *type slot/port*
3. **{ip | ipv6} policy route-map** *map-name*
4. **route-map** *map-name* **[permit | deny]** [*seq*]
5. **match {ip | ipv6} address access-list-name** *name* [*name...*]
6. **set ip next-hop** *address1*
7. **set ipv6 next-hop** *address1*
8. （任意） **set interface null0**
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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。

	コマンドまたはアクション	目的
ステップ 2	**interface** *type slot/port* 例： `switch(config)# interface ethernet 1/2`	インターフェイス設定モードを開始します。
ステップ 3	**{ip | ipv6} policy route-map** *map-name* 例： `switch(config-inf)# ip policy route-map Testmap`	IPv4 または IPv6 ポリシーベース ルーティング用のルートマップをインターフェイスに割り当てます。
ステップ 4	**route-map** *map-name* **[permit | deny]** [*seq*] 例： `switch(config-inf)# route-map Testmap`	ルート マップを作成するか、または既存のルートマップに対応するルートマップ設定モードを開始します。ルートマップのエントリを順序付けるには、*seq* を使用します。
ステップ 5	**match {ip | ipv6} address access-list-name** *name* [*name...*] 例： `switch(config-route-map)# match ip address access-list-name ACL1`	1 つまたは複数の IPv4 または IPv6 アクセス コントロール リスト（ACL）に対して IPv4 または IPv6 アドレスを照合します。このコマンドはポリシーベース ルーティング用であり、ルート フィルタリングまたは再配布では無視されます。
ステップ 6	**set ip next-hop** *address1* 例： `switch(config-route-map)# set ip next-hop 192.0.2.1`	ポリシーベース ルーティング用の IPv4 ネクストホップ アドレスを設定します。
ステップ 7	**set ipv6 next-hop** *address1* 例： `switch(config-route-map)# set ipv6 next-hop 2001:0DB8::1`	ポリシーベース ルーティング用の IPv6 ネクストホップ アドレスを設定します。
ステップ 8	（任意） **set interface null0** 例： `switch(config-route-map)# set interface null0`	ルーティングに使用するインターフェイスを設定します。パケットをドロップするには **null0** インターフェイスを使用します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config-route-map)# copy running-config startup-config`	この設定変更を保存します。

ポリシーベース リダイレクトの設定の確認

ポリシーベース リダイレクト設定情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show [ip | ipv6] policy [*name*]	IPv4 または IPv6 ポリシーに関する情報を表示します。
show route-map [*name*] **pbr-statistics**	ポリシー統計情報を表示します。

route-map *map-name* **pbr-statistics** コマンドを使用してポリシーを有効にします。**clear route-map** *map-name* **pbr-statistics** コマンドを使用してこれらのポリシーをクリアします。

ポリシー ベース リダイレクトの設定例

サービス VTEP を除くすべてのテナント VTEP で次の設定を実行します。

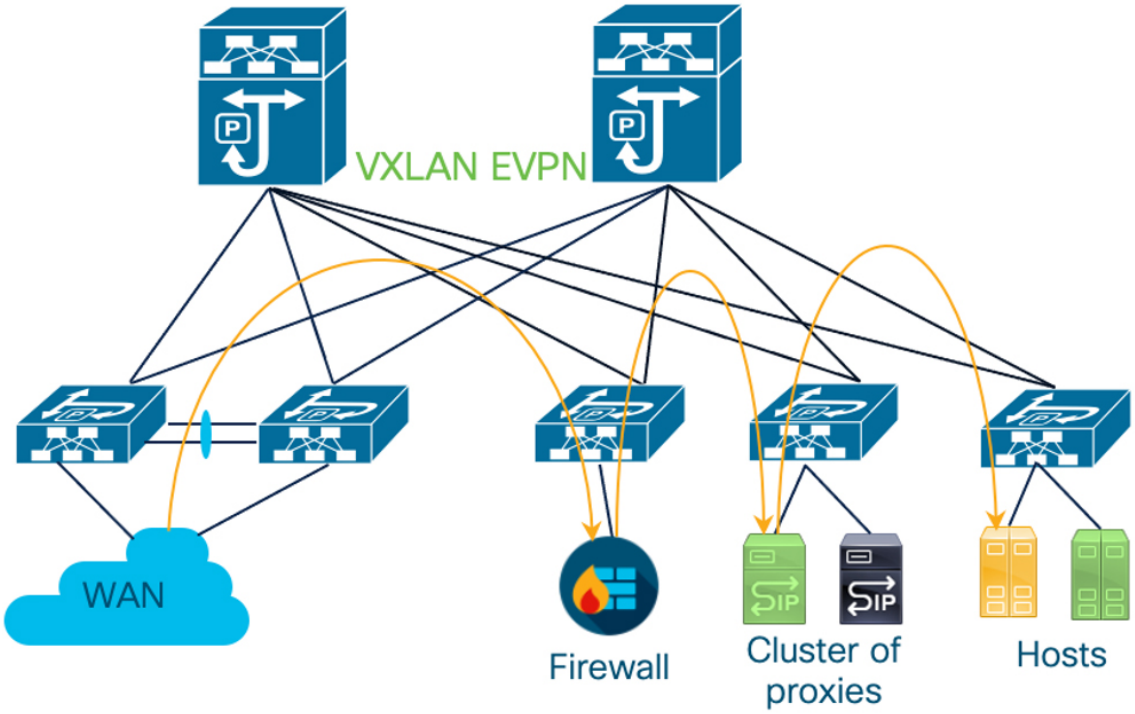
```
feature pbr

ipv6 access-list IPV6_App_group_1
10 permit ipv6 any 2001:10:1:1::0/64

ip access-list IPV4_App_group_1
10 permit ip any 10.1.1.0/24

ipv6 access-list IPV6_App_group_2
10 permit ipv6 any 2001:20:1:1::0/64

ip access-list IPV4_App_group_2
10 permit ip any 20.1.1.0/24

route-map IPV6_PBR_Appgroup1 permit 10
  match ipv6 address IPV6_App_group_2
  set ipv6 next-hop 2001:100:1:1::20  (next hop is that of the firewall)

route-map IPV4_ PBR_Appgroup1 permit 10
  match ip address IPV4_App_group_2
  set ip next-hop 10.100.1.20 (next hop is that of the firewall)

route-map IPV6_PBR_Appgroup2 permit 10
  match ipv6 address IPV6_App_group1
  set ipv6 next-hop 2001:100:1:1::20  (next hop is that of the firewall)

route-map IPV4_ PBR_Appgroup2 permit 10
  match ip address IPV4_App_group_1
  set ip next-hop 10.100.1.20 (next hop is that of the firewall)


interface Vlan10
! tenant SVI appgroup 1
vrf member appgroup
 ip address 10.1.1.1/24
 no ip redirect
 ipv6 address 2001:10:1:1::1/64
 no ipv6 redirects
 fabric forwarding mode anycast-gateway
ip policy route-map IPV4_ PBR_Appgroup1
ipv6 policy route-map IPV6_PBR_Appgroup1
interface Vlan20
! tenant SVI appgroup 2
vrf member appgroup
 ip address 20.1.1.1/24
 no ip redirect
```

```
 ipv6 address 2001:20:1:1::1/64
 no ipv6 redirects
 fabric forwarding mode anycast-gateway
ip policy route-map IPV4_ PBR_Appgroup2
ipv6 policy route-map IPV6_PBR_Appgroup2

On the service VTEP, the PBR policy is applied on the tenant VRF SVI. This ensures the
traffic post decapsulation will be redirected to firewall.
feature pbr

ipv6 access-list IPV6_App_group_1
10 permit ipv6 any 2001:10:1:1::0/64

ip access-list IPV4_App_group_1
10 permit ip any 10.1.1.0/24

ipv6 access-list IPV6_App_group_2
10 permit ipv6 any 2001:20:1:1::0/64

ip access-list IPV4_App_group_2
10 permit ip any 20.1.1.0/24

route-map IPV6_PBR_Appgroup1 permit 10
  match ipv6 address IPV6_App_group_2
  set ipv6 next-hop 2001:100:1:1::20  (next hop is that of the firewall)

route-map IPV6_PBR_Appgroup permit 20
  match ipv6 address IPV6_App_group1
  set ipv6 next-hop 2001:100:1:1::20  (next hop is that of the firewall)

route-map IPV4_ PBR_Appgroup permit 10
  match ip address IPV4_App_group_2
  set ip next-hop 10.100.1.20 (next hop is that of the firewall)

route-map IPV4_ PBR_Appgroup permit 20
  match ip address IPV4_App_group_1
  set ip next-hop 10.100.1.20 (next hop is that of the firewall)


interface vlan1000
!L3VNI SVI for Tenant VRF
vrf member appgroup
ip forward
ipv6 forward
ipv6 ipv6 address use-link-local-only
ip policy route-map IPV4_ PBR_Appgroup
ipv6 policy route-map IPV6_PBR_Appgroup
```

Enhanced-Policy Based Redirect（ePBR）

トラフィックを選択的にリダイレクトするソリューションとしてのVXLANPBRは、単純なトラフィックのリダイレクト要件にのみ対応できます。サービス チェーン、対称ロード バランシング、サービス アプライアンスの正常性の追跡など、より複雑なユースケースでは、PBR の使用が困難になります。PBR を使用したサービス チェーンの課題は、ユーザーがノードごとに一意のポリシーを作成し、チェーン内のすべてのノードでリダイレクション ルールを手動で管理する必要があることです。また、サービス ノードのステートフルな性質を考えると、PBR ルールはリバース トラフィックの対称性を保証する必要があり、これにより PBR ポリシーの構成と管理がさらに複雑になります。

Enhanced Policy-Based Redirect（ePBR）は、サービス ノードを挿入し、トラフィックを選択的にリダイレクトしてロードバランシングするための包括的なソリューションを提供します。ePBR は、トラフィック チェーンとロードバランシング ルールを作成するための簡素化されたワークフローを提供するとともに、サービス アプライアンスのヘルスをプローブ/モニタし、障害が発生した場合に修正措置を講じるためのオプションを提供します。 ePBR は、単一サイトとマルチサイトの両方の VXLAN EVPN 展開でサポートされます。



この図では、WANから発信される選択的なトラフィックがファイアウォールにチェーンされ、宛先ホストに転送される前に、トラフィックはプロキシのクラスタ全体で負荷分散されます。ePBR は、順方向と逆方向の両方のトラフィックが TCP プロキシのクラスタ内の同じサービス エンドポイントにリダイレクトされるようにすることで、特定のフローの対称性を維持します。

ePBR の詳細、注意事項、および構成例については、『Cisco Nexus 9000 Series NX-OS ePBR 構成ガイド』、『』、『』、『』および『拡張ポリシーベース リダイレクト ホワイト ペーパーを持つレイヤ 4 からレイヤ Layer 7 サービス リダイレクト』を参照してください。

第 27 章

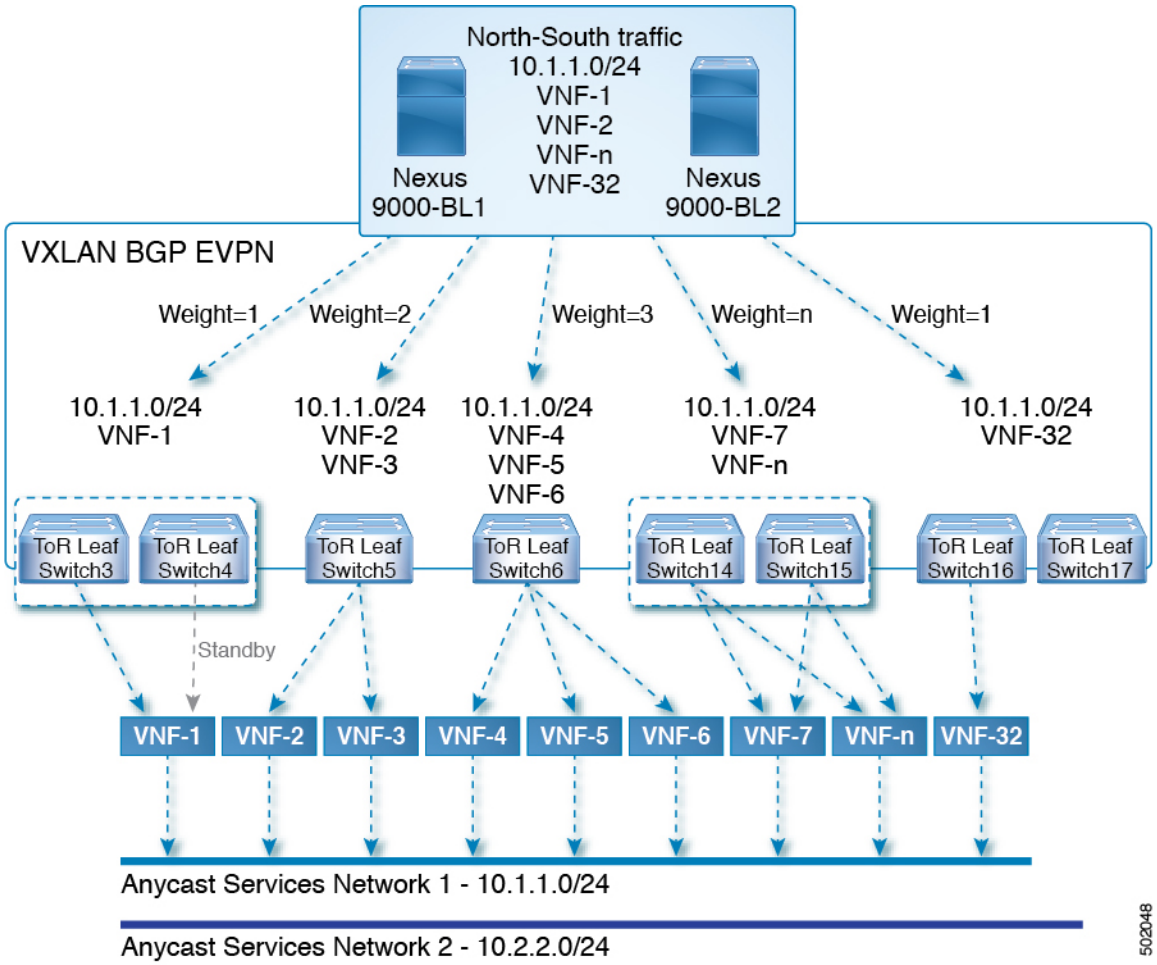
VNF の比例マルチパスの設定

この章は、次の項で構成されています。

VNF の比例マルチパスについて

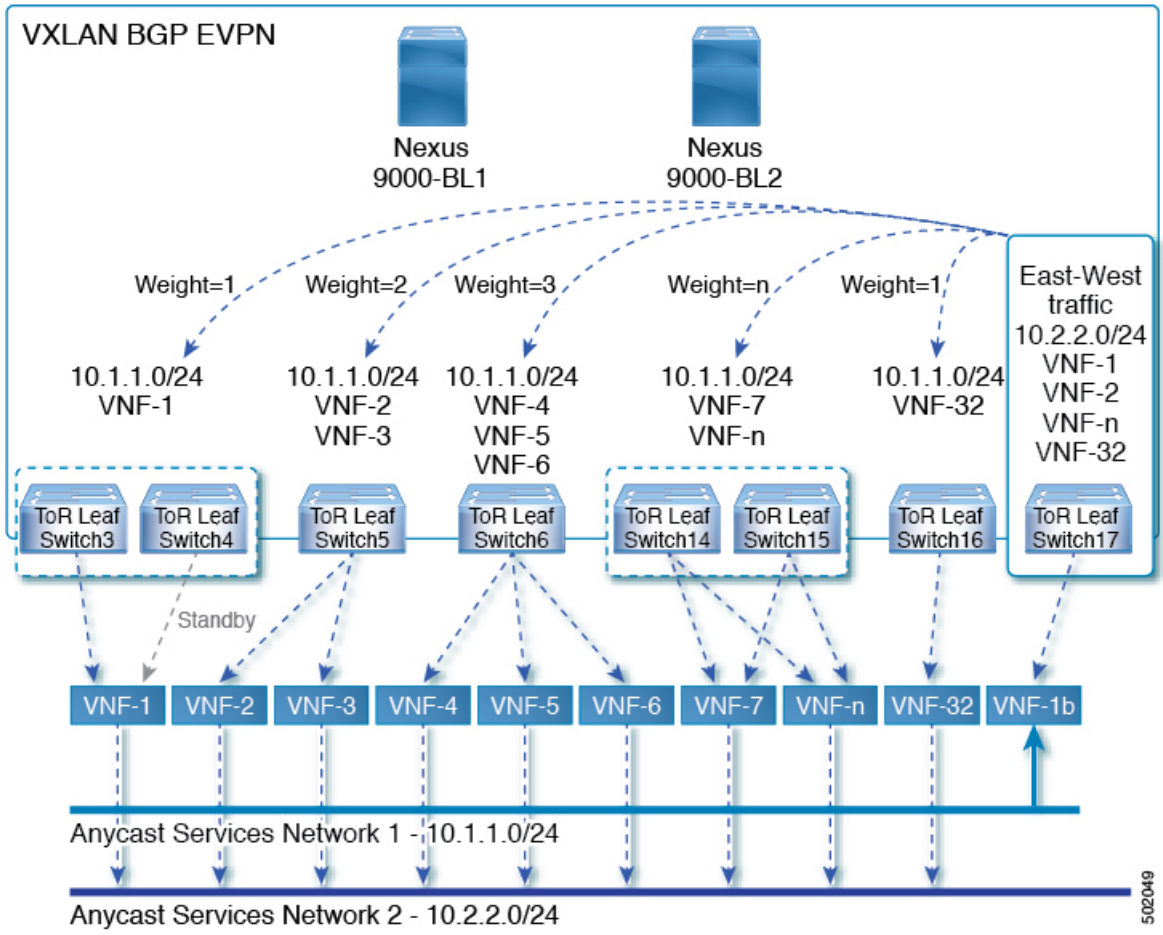
ネットワーク機能仮想化インフラストラクチャ（NFVi）では、エニーキャスト サービス ネットワークが複数の仮想ネットワーク機能（VNF）からアドバタイズされます。VNF の比例マルチパス の機能により、特定の宛先ネットワークへのすべての使用可能なネクスト ホップのアドバタイズが可能になります。この機能により、スイッチは特定のルートへのすべてのパスを等コストマルチパス（ECMP）と見なすことができ、複数の ToR にまたがる使用可能なすべてのリンクを使用してトラフィックを転送できます。

図 *63 :* サンプル トポロジ（***North-Sourth*** トラフィック）



上記の図では、ボーダー リーフで VXLAN ファブリックに入る North-South トラフィックは、すべての出力エンドポイントに送信されます。トラフィックは、出力 Top of Rack（ToR）から宛先ネットワークへのリンク数に比例して転送されます。

図 *64* : サンプル トポロジ（*East-West* トラフィック）



East-West トラフィックは、各 ToR スイッチによって宛先ネットワークにアドバタイズされるネクストホップの数に比例して、VXLAN トンネル エンドポイント（VTEP）間で転送されます。

スイッチは、レイヤ 2 VPN（L2VPN）/イーサネット VPN（EVPN）アドレス ファミリを使用して、ファブリック内の到達可能性をアドバタイズします。すべての ToR スイッチとボーダー リーフが同じ自律システム（AS）内にある場合、ルート リフレクタを使用するか、または各 BGP ルータを他のすべてのルータとピアリングすることによって、完全な内部 BGP（iBGP）メッシュが設定されます。

各 ToR とボーダー リーフは、VXLAN ファブリックの VTEP を構成します。VTEP 間のフル メッシュの BGP セッションを VTEP とルート リフレクタ間の単一の BGP セッションに削減するために、BGP ルート リフレクタを使用できます。仮想ネットワーク識別子（VNI）がオーバーレイ内でグローバルに一意になっています。各 Virtual Routing and Forwarding（VRF）インスタンスが一意の VNI にマッピングされています。VXLAN ヘッダーの内部宛先 MAC アドレスが、VXLAN ペイロードのルーティングを行う受信 VTEP に属しています。この MAC アドレスは、EVPN ルートとともに BGP 属性として配布されます。

顧客ネットワークのアドバタイズメント

カスタマー ネットワークは静的に設定されるか、またはプロバイダー エッジ（PE）-カスタマー エッジ（CE）リンクを介して内部ゲートウェイ プロトコル（IGP）または外部 BGP（eBGP）を使用してローカルに学習されます。これらのネットワークは BGP に再配布され、VXLAN ファブリックにアドバタイズされます。

接続された仮想マシン（VM）によって ToR にアドバタイズされたネットワークは、次を含む EVPN タイプ 5 ルートとして VXLAN ファブリックにアドバタイズされます。

- ルート識別子（RD）は、レイヤ 3 VNI の設定済みRDです。
- ゲートウェイ IP フィールドにネクスト ホップが入力されます。
- EVPN ルートのネクスト ホップは、引き続き VTEP IP となります。
- ルートのエクスポート ルート ターゲットは、関連付けられている レイヤ 3 VNI の設定済みエクスポート ルート ターゲットから取得されます。

複数の VRF ルートは、ゲートウェイ IP フィールドによってのみ区別される同じタイプ 5 ネットワーク層到達可能性情報（NLRI）を生成できます。ルートは L3VNI の RD でアドバタイズされ、ゲートウェイ IP はタイプ 5 NLRI のキーの一部ではありません。NLRI は、更新メッセージを使用して BGP ルータ間で交換されます。これらのルートは、ECMP を含むように BGP エクスポート メカニズムを拡張し、EVPN AF で addpath BGP 機能を使用して、EVPN AF にアドバタイズされます。

VNF の比例マルチパス 機能を使用して作成された EVPN AF 内の各タイプ 5 ルートには、受信したルート ターゲットの一致に基づいて対応する VRF にインポートされる複数のパスがあり、VRF 内および EVPN AF 内で ECMP が有効になっています。VRF 内では、ルートは複数のパスを持つ単一のプレフィックスです。各パスは、タイプ 5 EVPN パスまたは VRF 内でローカルに学習されたパスを表します。VNF の比例マルチパス 機能が有効になっている EVPN タイプ 5 ルートには、ゲートウェイ IP フィールドから派生した VRF のネクスト ホップがあります。BGP が EVPN タイプ 5 ルートでゲートウェイ IP をアドバタイズできるようにするには、**export-gateway-ip** コマンドを使用します。

maximum-paths mixed コマンドを使用して、BGP およびユニキャスト ルーティング情報ベース（URIB）を有効にし、次のパスを ECMP として見なします。

- iBGP パス
- eBGP パス
- BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス

パスは、デバイスに対してローカル（スタティック、iBGP、または eBGP）またはリモート（BGP-EVPN 経由で学習された eBGP または iBGP）のいずれかです。これは、ローカル ルートがリモート ルートよりも優先されるデフォルトのルート選択動作を上書きします。URIB は、ローカルに学習されたルートとユーザ設定のルートを含む、ルートのすべてのネクスト ホップを Unicast FIB Distribution Module（uFDM）/Forwarding Information Base（FIB）にダウンロードします。

Cisco NX-OSリリース9.3(5) 以降では、混合パスを使用する必要はありません。eBGP または iBGP のみで ECMP パスをフィルタリングするように選択できます。

Cisco NX-OS Release 9.3(5) 以降の **maximum-paths mixed** コマンドを入力すると、BGP はデフォルトで AS パス長をチェックします。AS パス長を無視する場合（たとえば、BGW や VTEP などのパケット転送に参加しているノード上）は、**bestpath as-path ignore** コマンドを入力する必要があります。 以前のリリースで **maximum-paths mixed** コマンドが有効になっている場合、BGP は AS パス長を無視し、URIB は ECMP を選択するときにアドミニストレーティブ ディスタンスを無視します。 影響がないことを確認するには、このコマンドを入力する前に Cisco NX-OSリリース9.3(5) にアップグレードすることを推奨します。

レガシー ピア サポート

ゲートウェイ IP が設定された EVPN タイプ 5 ルートをアドバタイズするには、**advertise-gw-ip** コマンドを使用します。次に、ToR はゲートウェイ IP をタイプ 5 NLRI でアドバタイズします。ただし、Cisco NX-OS リリース 9.2(1) よりも古い NX-OS バージョンで実行されているレガシー ピアは、予期しない動作を引き起こす可能性があるゲートウェイ IP を処理できません。このシナリオが発生しないようにするには、**no advertise-gw-ip** コマンドを使用してレガシーピアの VNF の比例マルチパス 機能を無効にします。BGP は、アドバタイズされるパスに有効なゲートウェイ IP がある場合でも、タイプ 5 NLRI のゲートウェイ IP フィールドをゼロに設定します。

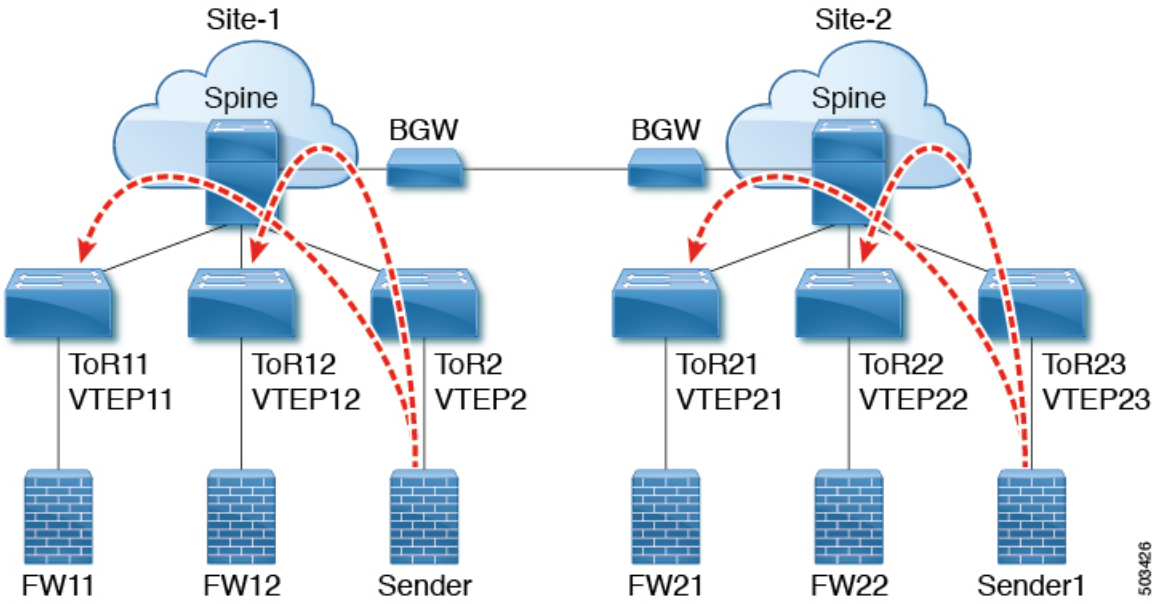
no advertise-gw-ip コマンドは、指定されたピア セッションを可能な限り適切にフラップします。ピアがこの機能をサポートしている場合、リモート ピアはグレースフル リスタートをトリガーします。セッションが再確立されると、ローカル ピアは、**advertise-gw-ip** コマンドが使用されたかどうかに応じて、ゲートウェイIPが設定されているか、ゲートウェイ IP がゼロである EVPN タイプ 5 ルートをアドバタイズします。デフォルトでは、このノブは有効になっており、ゲートウェイ IP フィールドに適切なネクスト ホップ値が入力されます。

マルチサイトでの VNF の比例マルチパス

Cisco NX-OS リリース 9.3(6) 以降のリリースでは、マルチサイトでの VNF の比例マルチパスがサポートされています。この機能により、ローカル VNF が使用できない場合に、サイト間でトラフィックを送信できます。

ToR はローカル VNF の使用を優先します。ただし、ローカル VNF が使用できない場合は、別のサイトで VNF を使用できます。次のトポロジでは、サイト 2 の ToR は VNF 21 および 22 を使用します。ただし、これらの VNF が使用できない場合、サイト 2 の送信者 1 はサイト 1 の VNF 11 および 12 にトラフィックを送信できます。

図 *65 :* マルチサイトトポロジの *VNF*



この機能を使用するには、VNF の比例マルチパスを設定し、マルチサイトを有効にします。構成例については、マルチサイトでの VNF の比例マルチパスの設定例（614 ページ）を参照してください。

VNF の比例マルチパスの前提条件

必要に応じて、Cisco NX-OS リリース 9.3(5) にアップグレードする前に、次のアクションを実行します。

- 再配布されたパスのルートマップを設定し、ローカルで再配布されたパスを使用してゲートウェイIPアドレスをエクスポートする場合は、**set ip next-hop redist-unchanged** コマンドを使用します。このコマンドは、ローカルに再配布されたパスのネクストホップを保持します。次に例を示します。

```
route-map redist-rtmap permit 10
match ip prefix-list vm-pfx-list
set ip next-hop redist-unchanged
```

- BGW や VTEP など、パケット転送に参加するノードで **bestpath as-path ignore** コマンドを入力します。このコマンドにより、BGP は AS パス長を無視します。

VNF の比例マルチパスのガイドラインと制限事項

Proportional Multipath for VNFには、次の注意事項と制約事項があります。

- VNF の比例マルチパス機能が有効になっている場合、BGP はすべてのパスを混合マルチパス モードでインストールするため、メンテナンス モードの分離は機能しません。または、ユーザ定義プロファイルを使用してスイッチがメンテナンス モードになったときに、アウトバウンド BPG 更新を拒否するためにルートマップが使用されます。
- この機能は、Cisco Nexus 9364C、9300-EX、および 9300-FX/FX2/FX3 プラットフォーム スイッチと、N9K-C9508-FM-E2 ファブリック モジュールおよび -EX または -FX ライン カードを備えたCisco Nexus 9500 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降では、VNF の比例マルチパス機能は、Cisco Nexus 9300-GX/GX2B プラットフォームスイッチでサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、VNF の比例マルチパス機能は、Cisco Nexus 9332D-H2R スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(2)F 以降、VNF の比例マルチパス機能は、Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、VNF の比例マルチパス機能は、Cisco Nexus 9364C-H1 スイッチでサポートされます。
- VNF の比例マルチパス機能が有効になっている場合は、スタティック ルートと直接ルートを BGP に再配布する必要があります。
- OSPF または EIGRP が IGP として使用されている場合、ルートは BGP に再配布できません。
- VNF のプロポーショナル マルチパスが有効で、ルートが BGP に再配布されない場合、URIB からのローカル ルートが BGP およびリモート TOR で EVPN パスとして表示されないため、トラフィックの非対称ロード バランシングが発生する可能性があります。
- 混合マルチパスが有効になっているデバイスは、同じロード バランシング アルゴリズムをサポートする必要があります。
- VNF インスタンスが複数の TOR にマルチホーム接続されている場合は、ネットワーク コマンドを使用してポリシーを設定するか、BGP ルートを作成する必要があります。その結果、VNF への各 TOR 接続が BGP ルーティング テーブルに表示されます。各 TOR は、VNF がマルチホームである他の TOR への VNF の直接ルートを確認できるようになりました。その結果、各 TOR は他の TOR を介してゲートウェイ IP へのパスをアドバタイズできるため、ネクスト ホップ解決ループが発生します。

  VNF が 2 つの TOR（TOR1 と TOR2）にマルチホーム接続されているシナリオを考えます。TOR への個々のリンクは、1.1.1.1 および 2.2.2.2 として扱われます。VNF が TOR を介して 192.168.1.0/24 サービスをアドバタイズする場合、TOR は EVPN ルートをそれぞれ 192.168.1.0/24 にゲートウェイ IP 1.1.1.1 および 2.2.2.2 でアドバタイズします。

  その結果、リモート TOR（TOR3 など）の再帰ネクスト ホップ（RNH）解決で問題が発生します。ゲートウェイ IP は、別のゲートウェイ IP を指す /24 ルートに解決されます。この 2 番目のゲートウェイ IP は、最初のゲートウェイ IP を指すルートによって解決されます。このシナリオでは、ゲートウェイ IP 1.1.1.1 は 2.2.2.2 を指す 1.1.1.0/24 によって解決されます。2.2.2.2 は、1.1.1.1 を指す 2.2.2.0/24 によって解決されます。

この状態は、VNF に接続された両方の TOR が VNF の接続されたルートをアドバタイズしているときに発生します。TOR1 は 1.1.1.0/24 および 2.2.2.0/24 をアドバタイズしています。ただし、1.1.1.0 は TOR1 に接続されたサブネットであるため、ゲートウェイ IP なしでアドバタイズされます。また、2.2.2.0 は、TOR1 に接続された VNF のアドレスである 1.1.1.1 を指す OSPF ルートです。

同様に、TOR2 は両方のサブネットをアドバタイズし、ゲートウェイ IP が直接 TOR2 に接続されているため、2.2.2.0/24 はゲートウェイ IP なしで送信されます。1.1.1.0 は OSPF 経由で学習され、TOR2 に接続された VNF のアドレスである 2.2.2.2 のゲートウェイ IP で送信されます。1.1.1.1/32 および 2.2.2.2/32 は、各 TOR の隣接マネージャ（AM）ルートであるため、アドバタイズされません。

この問題には、タイプ 5 ルートが関係する場合の解決策はありません。ただし、TOR がネットワーク コマンドを使用してゲートウェイ IP の /32 アドレスをアドバタイズする場合は、このシナリオを回避できます。ゲートウェイ IP がタイプ 2 EVPN MAC/IP ルートによって解決される場合、ゲートウェイ IP は /32 IP ルートによって解決されるため、このシナリオは回避できます。

- 次のガイドラインと制限事項は、マルチサイトでの VNF の比例マルチパスに適用されます。
  - この機能は、Cisco Nexus 9364C、9300-EX、および 9300-FX/FX2/FX3 プラットフォーム スイッチと、N9K-C9508-FM-E2 ファブリック モジュールおよび -EX または -FX ライン カードを備えたCisco Nexus 9500 プラットフォーム スイッチでサポートされます。
  - サイト間の VNF 移動はサポートされていません。
- 最大パス混合構成の比例マルチパスは、vPC リーフ スイッチに接続された VNF ではサポートされません。ただし、最大パス混合構成が使用されていない場合、vPC はサポートされます。
- マルチサイト ボーダー ゲートウェイをメンテナンス モードにすると、次の注意事項と制限事項が適用されます。
  - リモートファブリックからのBUMトラフィックは、メンテナンス モードのボーダー ゲートウェイに引き続き引き付けられます
  - メンテナンス モードのボーダー ゲートウェイは引き続き指定フォワーダ選択に参加します
  - デフォルトのメンテナンス モード プロファイルでは、コマンド「ip pim isolate」が適用されるため、ボーダー ゲートウェイは S,G ツリーからファブリック方向に分離されます。これにより、BUM トラフィック損失が発生するので、デフォルトよりも適切なメンテナンス モード プロファイルをボーダー ゲートウェイに使用する必要があります。

ルート リフレクタの設定

手順の概要

1. **configure terminal**
2. **router bgp** *number*
3. **address-family l2vpn evpn**
4. **additional-paths send**
5. **additional-paths receive**
6. **additional-paths selection route-map passall**
7. **route-map passall permit** *seq-num*
8. **set path-selection all advertise**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**router bgp** *number* 例： `switch(config)# router bgp 2`	BGP を設定します。
ステップ 3	**address-family l2vpn evpn** 例： `switch(config-router)# address-family l2vpn evpn`	**router bgp** コンテキストの下にあるアドレス ファミリのレイヤ 2 VPN EVPN を設定します。
ステップ 4	**additional-paths send** 例： `switch(config-router-af)# additional-paths send`	送信用の additional-paths 設定。
ステップ 5	**additional-paths receive** 例： `switch(config-router-af)# additional-paths receive`	受信用の additional-paths パス。
ステップ 6	**additional-paths selection route-map passall** 例：	additional-paths 設定により、ルート マップが適用されました。

	コマンドまたはアクション	目的
	`switch(config-router-af)# additional-paths selection route-map passall`	
ステップ 7	**route-map passall permit** *seq-num* 例： `switch(config)# route-map passall permit 10`	ルート マップを設定します。
ステップ 8	**set path-selection all advertise** 例： `switch(config-route-map)# set path-selection all advertise`	additional-paths 機能に関連するルートマップを設定します。

ToR の設定

この手順では、ToR の設定方法について説明します。

手順の概要

1. **configure terminal**
2. **router bgp** *number*
3. **address-family l2vpn evpn**
4. [**no**] **maximum-paths** [*eBGP max-paths* |**mixed** | **ibgp** |**local** | **eibgp**] *mpath-count*
5. **additional-paths send**
6. **additional-paths receive**
7. **additional-paths selection route-map passall**
8. **exit**
9. **vrf evpn-tenant-1001**
10. **address-family ipv4 unicast**
11. **export-gateway-ip**
12. [**no**] **maximum-paths** [*eBGP max-paths* |**mixed** | **ibgp** |**local** | **eibgp**] *mpath-count*
13. **redistribute static route-map redist-rtmap**
14. **maximum-paths local** *number*
15. **exit**
16. **address-family ipv6 unicast**
17. **export-gateway-ip**
18. [**no**] **maximum-paths** [*eBGP max-paths* |**mixed** | **ibgp** |**local** | **eibgp**] *mpath-count*
19. **redistribute static route-map redist-rtmap**
20. **maximum-paths local** *number*
21. **exit**
22. **route-map passall permit** *seq-num*
23. **set path-selection all advertise**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**router bgp** *number* 例： `switch(config)# router bgp 2`	BGP を設定します。
ステップ 3	**address-family l2vpn evpn** 例： `switch(config-router)# address-family l2vpn evpn`	**router bgp** コンテキストの下にあるアドレス ファミリのレイヤ 2 VPN EVPN を設定します。

ステップ 4	[**no**] **maximum-paths** [*eBGP max-paths* |**mixed**	**ibgp** |**local**	**eibgp**] *mpath-count*
			例：
			`switch(config-router-af)# maximum-paths ?`
			`  <1-64>  Number of parallel paths`
			`  *Default value is 1`
			`  eibgp  Configure multipath for both EBGP and IBGP paths`
			`  ibgp  Configure multipath for IBGP paths`
			`  local  Configure multipath for local paths`
			`  mixed  Configure multipath for local and remote paths`
			`switch(config-router-af)# maximum-paths mixed 32`
			例：
			`switch(config-router-af)# maximum-paths ibgp 32`
				次のオプションを使用できます。
				• *eBGP max-path*–eBGP 最大パスをいネーブル化します。範囲は 1 ～ 64 パラレル パスです。デフォルト値は 1 です。
				• **mixed**–BGP およびユニキャスト ルーティング情報ベース（URIB）をいネーブル化して、次のパスを等コスト マルチ パス（ECMP）と見なすことができます。
				  • eBGP パス
				  • eiBGP パス
				  • iBGP パス
				  • BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス
				• **ibgp**– iBGPを使用して ECMP パスをフィルタリングします。
				• **local**–ローカル パスのマルチパスを有効にします。
				• **mixed** または **ibgp** オプションを指定せずにコマンドを入力すると、eBGP が ECMP パスのフィルタリングに使用されます。

	コマンドまたはアクション	目的
		（注） 最大パス数ではなく単一のパスを使用する場合は、このコマンドの **no** 形式を使用します。
ステップ 5	**additional-paths send** 例： `switch(config-router-af)# additional-paths send`	送信用の additional-paths 設定。
ステップ 6	**additional-paths receive** 例： `switch(config-router-af)# additional-paths receive`	受信用の additional-paths パス。
ステップ 7	**additional-paths selection route-map passall** 例： `switch(config-router-af)# additional-paths selection route-map passall`	additional-paths 設定により、ルート マップが適用されました。
ステップ 8	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 9	**vrf evpn-tenant-1001** 例： `switch(config-router)# vrf evpn-tenant-1001`	VRF コンフィギュレーション モードに切り替えます。
ステップ 10	**address-family ipv4 unicast** 例： `switch(config-router)# address-family ipv4 unicast`	IPv4 のアドレス ファミリを設定します。
ステップ 11	**export-gateway-ip** 例： `switch(config-router-vrf-af)# export-gateway-ip`	BGP が EVPN タイプ 5 ルートでゲートウェイ IP をアドバタイズできるようにします。その VRF のすべてのプレフィックスのゲートウェイ IP をエクスポートします。

	コマンドまたはアクション	目的

		（注） ゲートウェイ IP をエクスポートする特定のプレフィックスを選択する場合は、**export-gateway-ip** コマンドの代わりに次の設定を使用します。
		`route-map name permit sequence`
		`  match ip address prefix-list name`
		`  set evpn gateway-ip use-next-hop`
		
		`vrf context vrf`
		`  address-family ipv4 unicast`
		`  export map name`

ステップ 12	[**no**] **maximum-paths** [*eBGP max-paths* |**mixed**	**ibgp** |**local**	**eibgp**] *mpath-count*
			例：
			`switch(config-router-vrf-af)# maximum-paths ?`
			`  <1-64>  Number of parallel paths`
			`  *Default value is 1`
			`  eibgp  Configure multipath for both EBGP and IBGP paths`
			`  ibgp  Configure multipath for IBGP paths`
			`  local  Configure multipath for local paths`
			`  mixed  Configure multipath for local and remote paths`
			`switch(config-router-vrf-af)# maximum-paths mixed 32`
			例：
			`switch(config-router-vrf-af)# maximum-paths ibgp 32`
				次のオプションを使用できます。
				• *eBGP max-path*–eBGP 最大パスをいネーブル化します。範囲は 1 ～ 64 パラレル パスです。デフォルト値は 1 です。
				• **mixed**–BGP およびユニキャスト ルーティング情報ベース（URIB）をいネーブル化して、次のパスを等コスト マルチ パス（ECMP）と見なすことができます。
				  • eBGP パス
				  • eiBGP パス
				  • iBGP パス
				  • BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス
				• **ibgp**– iBGPを使用して ECMP パスをフィルタリングします。
				• **local**–ローカル パスのマルチパスを有効にします。
				• **mixed** または **ibgp** オプションを指定せずにコマンドを入力すると、eBGP が ECMP パスのフィルタリングに使用されます。
				（注） 最大パス数ではなく単一のパスを使用する場合は、このコマンドの **no** 形式を使用します。

ステップ 13	**redistribute static route-map redist-rtmap** 例：	再配布されたパスのネクストホップを保持します。

	コマンドまたはアクション	目的
	`switch(config-router-vrf-af)# redistribute static route-map redist-rtmap`	
ステップ 14	**maximum-paths local** *number* 例： `switch(config-router-vrf-af)# maximum-paths local 32`	ルートのBGPベストパスとして再配布されるローカルパスの数を指定します。有効な範囲は 0 ～ 32 です。デフォルト値は 1 です。 （注） このコマンドは、**maximum-paths mixed** *mpath-count* コマンドではサポートされていません。一緒に設定しようとすると、エラー メッセージが表示されます。 （注） **set ip next-hop redist-unchanged** コマンドは、**maximum-paths local** コマンドが機能するために必要です。
ステップ 15	**exit** 例： `switch(config-router-vrf-af)# exit`	コマンド モードを終了します。
ステップ 16	**address-family ipv6 unicast** 例： `switch(config-router-vrf)# address-family ipv6 unicast`	IPv6 のアドレス ファミリを設定します。

ステップ 17	**export-gateway-ip**
	例：
	`switch(config-router-vrf-af)# export-gateway-ip`
		BGP が EVPN タイプ 5 ルートでゲートウェイ IP をアドバタイズできるようにします。その VRF のすべてのプレフィックスのゲートウェイ IP をエクスポートします。
		（注） ゲートウェイ IP をエクスポートする特定のプレフィックスを選択する場合は、**export-gateway-ip** コマンドの代わりに次の設定を使用します。
		`route-map name permit sequence`
		`  match ip address prefix-list name`
		`  set evpn gateway-ip use-next-hop`
		`vrf context vrf`
		`  address-family ipv4 unicast`
		`  export map name`

ステップ 18	[**no**] **maximum-paths** [*eBGP max-paths* |**mixed**	**ibgp** |**local**	**eibgp**] *mpath-count*
			例：
			`switch(config-router-vrf-af)# maximum-paths ?`
			`  <1-64>  Number of parallel paths`
				次のオプションを使用できます。
				• *eBGP max-path*–eBGP 最大パスをいネーブル化します。範囲は 1 ～ 64 パラレル パスです。デフォルト値は 1 です。

	コマンドまたはアクション	目的
	*Default value is 1 eibgp Configure multipath for both EBGP and IBGP paths ibgp Configure multipath for IBGP paths local Configure multipath for local paths mixed Configure multipath for local and remote paths switch(config-router-vrf-af)# **maximum-paths mixed 32** 例： switch(config-router-vrf-af)# **maximum-paths ibgp 32**	• **mixed**–BGP およびユニキャスト ルーティング情報ベース（URIB）をいネーブル化して、次のパスを等コスト マルチ パス（ECMP）と見なすことができます。 • eBGP パス • eiBGP パス • iBGP パス • BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス • **ibgp**– iBGPを使用して ECMP パスをフィルタリングします。 • **local**–ローカル パスのマルチパスを有効にします。 • **mixed** または **ibgp** オプションを指定せずにコマンドを入力すると、eBGP が ECMP パスのフィルタリングに使用されます。 （注） 最大パス数ではなく単一のパスを使用する場合は、このコマンドの **no** 形式を使用します。
ステップ 19	**redistribute static route-map redist-rtmap** 例： switch(config-router-vrf-af)# **redistribute static route-map redist-rtmap**	再配布されたパスのネクストホップを保持します。
ステップ 20	**maximum-paths local** *number* 例： switch(config-router-vrf-af)# **maximum-paths local 32**	ルートのBGPベストパスとして再配布されるローカルパスの数を指定します。有効な範囲は 0 ～ 32 です。デフォルト値は 1 です。 （注） このコマンドは、**maximum-paths mixed** *mpath-count* コマンドではサポートされていません。一緒に設定しようとすると、エラー メッセージが表示されます。
ステップ 21	**exit** 例： switch(config-router-vrf-af)# **exit**	コマンド モードを終了します。

	コマンドまたはアクション	目的
ステップ 22	**route-map passall permit** *seq-num* 例： `switch(config)# route-map passall permit 10`	ルート マップを設定します。
ステップ 23	**set path-selection all advertise** 例： `switch(config-route-map)# set path-selection all advertise`	additional-paths 機能に関連するルートマップを設定します。

ボーダー リーフの設定

この手順では、ボーダー リーフの設定方法について説明します。

手順の概要

1. **configure terminal**
2. **router bgp** *number*
3. **address-family l2vpn evpn**
4. [**no**] **maximum-paths** [*eBGP max-paths* |**mixed** | **ibgp** |**local** | **eibgp**] *mpath-count*
5. **additional-paths send**
6. **additional-paths receive**
7. **additional-paths selection route-map passall**
8. **exit**
9. **vrf evpn-tenant-1001**
10. **address-family ipv4 unicast**
11. **export-gateway-ip**
12. [**no**] **maximum-paths** [*eBGP max-paths* |**mixed** | **ibgp** |**local** | **eibgp**] *mpath-count*
13. **redistribute static route-map redist-rtmap**
14. **maximum-paths local** *number*
15. **address-family ipv6 unicast**
16. **export-gateway-ip**
17. [**no**] **maximum-paths** [*eBGP max-paths* |**mixed** | **ibgp** |**local** | **eibgp**] *mpath-count*
18. **redistribute static route-map redist-rtmap**
19. **maximum-paths local** *number*
20. **exit**
21. **route-map passall permit** *seq-num*
22. **set path-selection all advertise**
23. **ip load-sharing address source-destination rotate** *rotate* **universal-id** *seed*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**router bgp** *number* 例： `switch(config)# router bgp 2`	BGP を設定します。
ステップ 3	**address-family l2vpn evpn** 例： `switch(config-router)# address-family l2vpn evpn`	**router bgp** コンテキストの下にあるアドレス ファミリのレイヤ 2 VPN EVPN を設定します。

ステップ 4	[**no**] **maximum-paths** [*eBGP max-paths* |**mixed**	**ibgp** |**local**	**eibgp**] *mpath-count*
			例：
			`switch(config-router-af)# maximum-paths ?`
			`  <1-64>  Number of parallel paths`
			`  *Default value is 1`
			`  eibgp  Configure multipath for both EBGP and IBGP paths`
			`  ibgp  Configure multipath for IBGP paths`
			`  local  Configure multipath for local paths`
			`  mixed  Configure multipath for local and remote paths`
			`switch(config-router-af)# maximum-paths mixed 32`
			例：
			`switch(config-router-af)# maximum-paths ibgp 32`
				次のオプションを使用できます。
				• *eBGP max-path*–eBGP 最大パスをいネーブル化します。範囲は 1 ～ 64 パラレル パスです。デフォルト値は 1 です。
				• **mixed**–BGP およびユニキャスト ルーティング情報ベース（URIB）をいネーブル化して、次のパスを等コスト マルチ パス（ECMP）と見なすことができます。
				  • eBGP パス
				  • eiBGP パス
				  • iBGP パス
				  • BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス
				• **ibgp**– iBGPを使用して ECMP パスをフィルタリングします。
				• **local**–ローカル パスのマルチパスを有効にします。
				• **mixed** または **ibgp** オプションを指定せずにコマンドを入力すると、eBGP が ECMP パスのフィルタリングに使用されます。

	コマンドまたはアクション	目的
		（注） 最大パス数ではなく単一のパスを使用する場合は、このコマンドの **no** 形式を使用します。
ステップ 5	**additional-paths send** 例： `switch(config-router-af)# additional-paths send`	送信用の additional-paths 設定。
ステップ 6	**additional-paths receive** 例： `switch(config-router-af)# additional-paths receive`	受信用の additional-paths パス。
ステップ 7	**additional-paths selection route-map passall** 例： `switch(config-router-af)# additional-paths selection route-map passall`	additional-paths 設定は、追加パス機能を有効にします。
ステップ 8	**exit** 例： `switch(config-router-af)# exit`	コマンド モードを終了します。
ステップ 9	**vrf evpn-tenant-1001** 例： `switch(config-router)# vrf evpn-tenant-1001`	VRF コンフィギュレーション モードに切り替えます。
ステップ 10	**address-family ipv4 unicast** 例： `switch(config-router)# address-family ipv4 unicast`	IPv4 のアドレス ファミリを設定します。
ステップ 11	**export-gateway-ip** 例： `switch(config-router-vrf-af)# export-gateway-ip`	BGP が EVPN タイプ 5 ルートでゲートウェイ IP をアドバタイズできるようにします。その VRF のすべてのプレフィックスのゲートウェイ IP をエクスポートします。

	コマンドまたはアクション	目的

		（注） ゲートウェイ IP をエクスポートする特定のプレフィックスを選択する場合は、**export-gateway-ip** コマンドの代わりに次の設定を使用します。
		`route-map name permit sequence`
		`  match ip address prefix-list name`
		`  set evpn gateway-ip use-next-hop`
		
		`vrf context vrf`
		`  address-family ipv4 unicast`
		`  export map name`

ステップ 12	[**no**] **maximum-paths** [*eBGP max-paths* |**mixed**	**ibgp** |**local**	**eibgp**] *mpath-count*
			例：
			`switch(config-router-af)# maximum-paths ?`
			`  <1-64>  Number of parallel paths`
			`  *Default value is 1`
			`  eibgp  Configure multipath for both EBGP and IBGP paths`
			`  ibgp  Configure multipath for IBGP paths`
			`  local  Configure multipath for local paths`
			`  mixed  Configure multipath for local and remote paths`
			`switch(config-router-vrf-af)# maximum-paths mixed 32`
			例：
			`switch(config-router-vrf-af)# maximum-paths ibgp 32`
				次のオプションを使用できます。
				• *eBGP max-path*–eBGP 最大パスをいネーブル化します。範囲は 1 ～ 64 パラレル パスです。デフォルト値は 1 です。
				• **mixed**–BGP およびユニキャスト ルーティング情報ベース（URIB）をいネーブル化して、次のパスを等コスト マルチ パス（ECMP）と見なすことができます。
				  • eBGP パス
				  • eiBGP パス
				  • iBGP パス
				  • BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス
				• **ibgp**– iBGPを使用して ECMP パスをフィルタリングします。
				• **local**–ローカル パスのマルチパスを有効にします。
				• **mixed** または **ibgp** オプションを指定せずにコマンドを入力すると、eBGP が ECMP パスのフィルタリングに使用されます。
				（注） 最大パス数ではなく単一のパスを使用する場合は、このコマンドの **no** 形式を使用します。

ステップ 13	**redistribute static route-map redist-rtmap** 例：	再配布されたパスのネクストホップを保持します。

	コマンドまたはアクション	目的
	`switch(config-router-vrf-af)# redistribute static route-map redist-rtmap`	
ステップ 14	**maximum-paths local** *number* 例： `switch(config-router-vrf-af)# maximum-paths local 32`	ルートのBGPベストパスとして再配布されるローカルパスの数を指定します。有効な範囲は 0 ～ 32 です。デフォルト値は 1 です。 （注） このコマンドは、**maximum-paths mixed** *mpath-count* コマンドではサポートされていません。一緒に設定しようとすると、エラー メッセージが表示されます。
ステップ 15	**address-family ipv6 unicast** 例： `switch(config-router-vrf)# address-family ipv6 unicast`	IPv6 のアドレス ファミリを設定します。

ステップ 16	**export-gateway-ip**
	例：
	`switch(config-router-vrf-af)# export-gateway-ip`
		BGP が EVPN タイプ 5 ルートでゲートウェイ IP をアドバタイズできるようにします。その VRF のすべてのプレフィックスのゲートウェイ IP をエクスポートします。
		（注） ゲートウェイ IP をエクスポートする特定のプレフィックスを選択する場合は、**export-gateway-ip** コマンドの代わりに次の設定を使用します。
		`route-map name permit sequence`
		`  match ip address prefix-list name`
		`  set evpn gateway-ip use-next-hop`
		
		`vrf context vrf`
		`  address-family ipv4 unicast`
		`  export map name`

ステップ 17	[**no**] **maximum-paths** [*eBGP max-paths* |**mixed**	**ibgp** |**local**	**eibgp**] *mpath-count*
			例：
			`switch(config-router-vrf-af)# maximum-paths ?`
			`  <1-64>  Number of parallel paths`
			`  *Default value is 1`
			`  eibgp  Configure multipath for both EBGP and IBGP paths`
			`  ibgp  Configure multipath for IBGP paths`
			`  local  Configure multipath for local paths`
			`  mixed  Configure multipath for local and remote paths`
			`switch(config-router-vrf-af)# maximum-paths mixed 32`
				次のオプションを使用できます。
				• *eBGP max-path*–eBGP 最大パスをいネーブル化します。範囲は 1 ～ 64 パラレル パスです。デフォルト値は 1 です。
				• **mixed**–BGP およびユニキャスト ルーティング情報ベース（URIB）をいネーブル化して、次のパスを等コスト マルチ パス（ECMP）と見なすことができます。
				  • eBGP パス
				  • eiBGP パス

	コマンドまたはアクション	目的
	例： `switch(config-router-vrf-af)# maximum-paths ibgp 32`	• iBGP パス • BGP に再配布または挿入される他のプロトコル（スタティックなど）からのパス • **ibgp**– iBGPを使用して ECMP パスをフィルタリングします。 • **local**–ローカルパスのマルチパスを有効にします。 • **mixed** または **ibgp** オプションを指定せずにコマンドを入力すると、eBGP が ECMP パスのフィルタリングに使用されます。 （注） 最大パス数ではなく単一のパスを使用する場合は、このコマンドの **no** 形式を使用します。
ステップ 18	**redistribute static route-map redist-rtmap** 例： `switch(config-router-vrf-af)# redistribute static route-map redist-rtmap`	再配布されたパスのネクストホップを保持します。
ステップ 19	**maximum-paths local** *number* 例： `switch(config-router-vrf-af)# maximum-paths local 32`	ルートのBGPベストパスとして再配布されるローカルパスの数を指定します。有効な範囲は 0 ～ 32 です。デフォルト値は 1 です。 （注） このコマンドは、**maximum-paths mixed** *mpath-count* コマンドではサポートされていません。一緒に設定しようとすると、エラー メッセージが表示されます。
ステップ 20	**exit** 例： `switch(config-router-vrf-af)# exit`	コマンド モードを終了します。
ステップ 21	**route-map passall permit** *seq-num* 例： `switch(config)# route-map passall permit 10`	ルート マップを設定します。
ステップ 22	**set path-selection all advertise** 例： `switch(config-route-map)# set path-selection all advertise`	additional-paths 機能に関連するルートマップを設定します。

	コマンドまたはアクション	目的
ステップ 23	**ip load-sharing address source-destination rotate** *rotate* **universal-id** *seed* 例： `ip load-sharing address source-destination rotate 32 universal-id 1`	データ トラフィックに対するユニキャスト FIB のロード シェアリング アルゴリズムを設定します。 • **universal-id** オプションは、ハッシュ アルゴリズムのランダム シードを設定し、フローをあるリンクから別のリンクにシフトします。 汎用 ID を設定する必要はありません。ユーザが設定しなかった場合は、Cisco NX-OS が汎用 ID を選択します。*seed* 範囲は 1 ～ 4294967295 です。 • **rotate** オプションを使用すると、ハッシュ アルゴリズムはネットワーク内のすべてのノードで同じリンクを継続的に選択しないように、リンク ピッキング選択を循環させます。これは、ハッシュ アルゴリズムのビット パターンに影響を与えることによって機能します。このオプションは、あるリンクから別のリンクにフローをシフトし、最初の ECMP レベルからすでにロード バランシング（極性化）されているトラフィックのロード バランシングを複数のリンク間で行います。 **rotate** 値を指定すると、64 ビットのストリームが循環回転でそのビット位置から解釈されます。**rotate** 値の範囲は 1 ～ 63 で、デフォルトは 32 です。 （注） 多層レイヤ 3 トポロジでは、極性が発生する可能性があります。極性を回避するには、トポロジの各層で異なる循環ビットを使用します。 （注） ポート チャネルの rotation 値を設定するには、**port-channel load-balance src-dst ip-l4port rotate** *rotate* コマンドを使用します。このコマンドの詳細については、『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide、Release 9.x*』を参照してください。

BGP レガシー ピアの設定

9.2(1) より前の Cisco Nexus リリースを実行している場合は、次の手順に従って、そのピアへのゲートウェイ IP アドレスの送信を無効にします。

手順の概要

1. **configure terminal**
2. **router bgp** *number*
3. **neighbor** *address* **remote-as** *number*
4. **address-family l2vpn evpn**
5. **no advertise-gw-ip**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**router bgp** *number* 例： `switch(config)# router bgp 2000000`	BGP を設定します。
ステップ 3	**neighbor** *address* **remote-as** *number* 例： `switch(config-router)# neighbor 8.8.8.8 remote-as 2000000`	ネイバーを定義します。
ステップ 4	**address-family l2vpn evpn** 例： `switch(config-router-neighbor)# address-family l2vpn evpn`	アドレス ファミリのレイヤ 2 VPN EVPN を設定します。
ステップ 5	**no advertise-gw-ip** 例： `switch(config-router-neighbor-af)# no advertise-gw-ip`	レガシー ピアの BGP EVPN 混合パスおよび比例レイヤ 3 マルチパス機能をディセーブルにします。

メンテナンス モード用のユーザ定義プロファイルの設定

手順の概要

1. **configure terminal**
2. **configure maintenance profile maintenance-mode**
3. **route-map** *name* **deny** *sequence*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**configure maintenance profile maintenance-mode** 例： `switch(config)# configure maintenance profile maintenance-mode`	メンテナンス モード プロファイルの設定
ステップ 3	**route-map** *name* **deny** *sequence* 例： `switch(config-mm-profile)# route-map GIR deny 5`	ルート マップを設定します。*sequence* の値の範囲は 0 ～ 65535 です。デフォルト値は 10 です。

通常モードのユーザ定義プロファイルの設定

手順の概要

1. **configure terminal**
2. **configure maintenance profile normal-mode**
3. **route-map** *name* **permit** *sequence*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**configure maintenance profile normal-mode** 例： `switch(config)# configure maintenance profile normal-mode`	メンテナンス モードを設定します。
ステップ 3	**route-map** *name* **permit** *sequence* 例： `switch(config-mm-profile)# route-map GIR permit 5`	ルート マップを設定します。*sequence* の値の範囲は 0 ～ 65535 です。デフォルト値は 10 です。

デフォルト ルート マップの設定

手順の概要

1. **configure terminal**
2. **route-map** *name* **permit** *sequence*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**route-map** *name* **permit** *sequence* 例： `switch(config-mm-profile)# route-map GIR permit 5`	ルート マップを設定します。*sequence* の値の範囲は 0 ～ 65535 です。デフォルト値は 10 です。

ルート リフレクタへのルート マップの適用

手順の概要

1. **configure terminal**
2. **router bgp** *number*
3. **neighbor** *ip-address*
4. **address-family l2vpn evpn**
5. **route-map** *name* **out**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**router bgp** *number* 例： `switch(config)# router bgp 2`	BGP を設定します。
ステップ 3	**neighbor** *ip-address* 例： `switch(config-router)# neighbor 10.1.1.1`	ルート リフレクタである BGP ネイバーの IP アドレスを設定します。 *ip-address* には、IPv4 または IPv6 のアドレスまたはプレフィックスを指定できます。
ステップ 4	**address-family l2vpn evpn** 例： `switch(config-router-neighbor)# address-family l2vpn evpn`	レイヤ 2 VPN EVPN アドレス ファミリを設定します。
ステップ 5	**route-map** *name* **out** 例： `switch(config-router-neighbor-af)# route-map GIR out`	ルート マップをネイバー ルート リフレクタに適用します。

VNF の比例マルチパスの確認

コマンド	目的
show bgp ipv4 unicast	IPv4 ユニキャスト アドレス ファミリのボーダーゲートウェイ プロトコル（BGP）情報を表示します。
show bgp l2vpn evpn	レイヤ 2 バーチャル プライベート ネットワーク（L2VPN）イーサネット バーチャル プライベート ネットワーク（EVPN）アドレス ファミリの BGP 情報を表示します。
show ip route	ユニキャスト RIB から受け取ったルートを表示します。
show maintenance profile maintenance-mode	メンテナンスモードのGIRユーザ定義プロファイルを表示します。
show maintenance profile normal-mode	通常モードの GIR ユーザ定義プロファイルを表示します。

次に、L2VPN EVPN アドレス ファミリの BGP 情報を表示する例を示します。

```
switch# show bgp l2vpn evpn 11.1.1.0
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 13.13.13.13:3       //     Remote route
BGP routing table entry for [5]:[0]:[0]:[24]:[11.1.1.0]/224, version 1341
Paths: (3 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported to 2 destination(s)
  Gateway IP: 11.1.1.133
  AS-Path: 2000000 100000 , path sourced external to AS
    11.11.11.11 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 3
      Extcommunity: RT:23456:22001 Route-Import:11.11.11.11:2001 ENCAP:8
          Router MAC:003a.7d7d.1dbd

  Path type: external, path is valid, not best reason: Neighbor Address, multipath
             Imported to 2 destination(s)
  Gateway IP: 11.1.1.233
  AS-Path: 2000000 100 , path sourced external to AS
    33.33.33.33 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 2
      Extcommunity: RT:23456:22001 Route-Import:33.33.33.33:2001 ENCAP:8
```

```
          Router MAC:e00e.da4a.589d

  Path type: external, path is valid, not best reason: Neighbor Address, multipath
             Imported to 2 destination(s)
  Gateway IP: 11.1.1.100
  AS-Path: 2000000 500000 , path sourced external to AS
    22.22.22.22 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 1
      Extcommunity: RT:23456:22001 Route-Import:22.22.22.22:2001 ENCAP:8
          Router MAC:e00e.da4a.62a5

  Path-id 1 not advertised to any peer

Route Distinguisher: 4.4.4.4:3    (L3VNI 22001)       //   Local L3VNI
BGP routing table entry for [5]:[0]:[0]:[24]:[11.1.1.0]/224, version 3465
Paths: (3 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported from 13.13.13.13:3:[5]:[0]:[0]:[24]:[11.1.1.0]/224
  Gateway IP: 11.1.1.100
  AS-Path: 2000000 500000 , path sourced external to AS
    22.22.22.22 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 1
      Extcommunity: RT:23456:22001 Route-Import:22.22.22.22:2001 ENCAP:8
          Router MAC:e00e.da4a.62a5

  Path type: external, path is valid, not best reason: newer EBGP path, multipat
h
             Imported from 13.13.13.13:3:[5]:[0]:[0]:[24]:[11.1.1.0]/224
  Gateway IP: 11.1.1.233
  AS-Path: 2000000 100 , path sourced external to AS
    33.33.33.33 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 2
      Extcommunity: RT:23456:22001 Route-Import:33.33.33.33:2001 ENCAP:8
          Router MAC:e00e.da4a.589d

  Path type: external, path is valid, not best reason: newer EBGP path, multipat
h
             Imported from 13.13.13.13:3:[5]:[0]:[0]:[24]:[11.1.1.0]/224
  Gateway IP: 11.1.1.133
  AS-Path: 2000000 100000 , path sourced external to AS
    11.11.11.11 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 3
      Extcommunity: RT:23456:22001 Route-Import:11.11.11.11:2001 ENCAP:8
          Router MAC:003a.7d7d.1dbd

  Path-id 1 not advertised to any peer
```

次に、IPv4 ユニキャスト アドレス ファミリの BGP 情報を表示する例を示します。

```
switch# show bgp ipv4 unicast 11.1.1.0 vrf cust_1
BGP routing table information for VRF cust_1, address family IPv4 Unicast
BGP routing table entry for 11.1.1.0/24, version 4
```

```
Paths: (3 available, best #1)
Flags: (0x80080012) on xmit-list, is in urib, is backup urib route, is in HW
  vpn: version 1093, (0x100002) on xmit-list
Multipath: eBGP iBGP

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: external, path is valid, is best path, in rib
             Imported from 13.13.13.13:3:[5]:[0]:[0]:[24]:[11.1.1.0]/224
  AS-Path: 2000000 500000 , path sourced external to AS
    11.1.1.100 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 1
      Extcommunity: RT:23456:22001 Route-Import:22.22.22.22:2001 ENCAP:8
          Router MAC:e00e.da4a.62a5

  Path type: external, path is valid, not best reason: Neighbor Address, multipath, in
rib
             Imported from 13.13.13.13:3:[5]:[0]:[0]:[24]:[11.1.1.0]/224
  AS-Path: 2000000 100 , path sourced external to AS
    11.1.1.233 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 2
      Extcommunity: RT:23456:22001 Route-Import:33.33.33.33:2001 ENCAP:8
          Router MAC:e00e.da4a.589d

  Path type: external, path is valid, not best reason: Neighbor Address, multipath, in
rib
             Imported from 13.13.13.13:3:[5]:[0]:[0]:[24]:[11.1.1.0]/224
  AS-Path: 2000000 100000 , path sourced external to AS
    11.1.1.133 (metric 5) from 102.102.102.102 (102.102.102.102)
      Origin incomplete, MED not set, localpref 100, weight 0
      Received label 22001
      Received path-id 3
      Extcommunity: RT:23456:22001 Route-Import:11.11.11.11:2001 ENCAP:8
          Router MAC:003a.7d7d.1dbd

  VRF advertise information:
  Path-id 1 not advertised to any peer

  VPN AF advertise information:
  Path-id 1 not advertised to any peer
```

次に、VNF のプロポーショナルマルチパス機能を設定した後に、ユニキャスト RIB からのルートを表示する例を示します。

```
switch# show ip route 1.1.1.0 vrf cust_1
IP Route Table for VRF "cust_1"
…
1.1.1.0/24, ubest/mbest: 22/0, all-best (0x300003d)
    *via 3.0.0.1, [1/0], 08:13:17, static
         recursive next hop: 3.0.0.1/32
    *via 3.0.0.2, [1/0], 08:13:17, static
         recursive next hop: 3.0.0.2/32
    *via 3.0.0.3, [1/0], 08:13:16, static
         recursive next hop: 3.0.0.3/32
    *via 3.0.0.4, [1/0], 08:13:16, static
         recursive next hop: 3.0.0.4/32
   *via 2.0.0.1, [200/0], 06:09:19, bgp-2, internal, tag 2 (evpn) segid: 3003802 tunnelid:
 0x300003e encap: VXLAN
          BGP-EVPN: VNI=3003802 (EVPN)
         client-specific data: 3b
```

```
        recursive next hop: 2.0.0.1/32
        extended route information: BGP origin AS 2 BGP peer AS 2
   *via 2.0.0.2, [200/0], 06:09:19, bgp-2, internal, tag 2 (evpn) segid: 3003802 tunnelid:
0x300003e encap: VXLAN
         BGP-EVPN: VNI=3003802 (EVPN)
        client-specific data: 3b
        recursive next hop: 2.0.0.2/32
        extended route information: BGP origin AS 2 BGP peer AS 2
```

次に、メンテナンスモードの GIR ユーザ定義プロファイルを表示する例を示します。

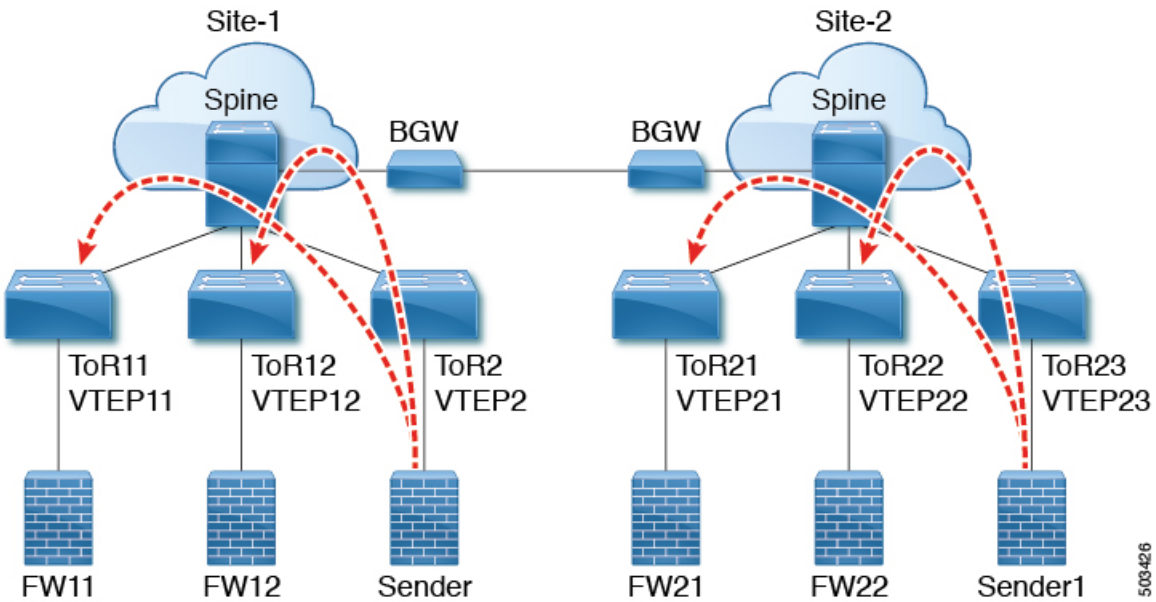
```
switch# show maintenance profile maintenance-mode
  [Maintenance Mode]
  ip pim isolate
  router bgp 2
   isolate
  router isis 1
   isolate
  route-map GIR deny 5
```

次に、通常モードの GIR ユーザ定義プロファイルを表示する例を示します。

```
switch# show maintenance profile normal-mode
  [Normal Mode]
  no ip pim isolate
  router bgp 2
   no isolate
  router isis 1
   no isolate
  route-map GIR permit 5
```

マルチサイトでの VNF の比例マルチパスの設定例

図 *66 :* マルチサイトトポロジの *VNF*



次の設定例では、ローカル VNF が使用できない場合に、サイト間でトラフィックを送信できます。

```
feature telnet
feature nxapi
feature bash-shell
feature scp-server
nv overlay evpn
feature ospf
feature bgp
feature pim
feature interface-vlan
feature vn-segment-vlan-based
feature bfd
feature nv overlay

no password strength-check
username admin password 5 password role network-admin
ip domain-lookup
copp profile strict
evpn multisite border-gateway 1
  delay-restore time 30
snmp-server user admin network-admin auth md5 0x66a8185ad28d9df13d9214f6e19aad37 priv
0x66a8185ad28d9df13d9214f6e19aad37 localizedkey

fabric forwarding anycast-gateway-mac 0000.2222.3333
ip pim ssm range 232.0.0.0/8
vlan 1,14,24,100-110,120-150,1000-1010,1100-1110,2000-2010,2100-2110,3000-3010
vlan 100
  name l2-vni-vlan-0-for-vrf100
  vn-segment 2000100
vlan 101
  name l2-vni-vlan-0-for-vrf101
  vn-segment 2000101
vlan 1100
  name l2-vni-vlan-1-for-vrf100
  vn-segment 2001100
vlan 1101
  name l2-vni-vlan-1-for-vrf101
  vn-segment 2001101
vlan 2100
  name l3-vni-vlan-for-vrf100
  vn-segment 3000100
vlan 2101
  name l3-vni-vlan-for-vrf101
  vn-segment 3000101

route-map passall permit 10
  set path-selection all advertise
route-map permit-all permit 10
  set path-selection all advertise
route-map permit-all-v6 permit 10

vrf context vrf100
  vni 3000100
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn
vrf context vrf101
 vni 3000101
```

```
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn

interface Vlan14
  no shutdown
  vrf member vrf100
  ip address 192.14.0.1/24
  ipv6 address 192:14::1/64

interface Vlan24
  no shutdown
  vrf member vrf101
  ip address 192.24.0.1/24
  ipv6 address 192:24::1/64

interface Vlan100
  description "L3VRF.VLANNUM.0.222"
  no shutdown
  vrf member vrf100
  ip address 100.0.0.222/24
  ipv6 address 100::222/64
  fabric forwarding mode anycast-gateway

interface Vlan101
  description "L3VRF.VLANNUM.0.222"
  no shutdown
  vrf member vrf101
  ip address 101.0.0.222/24
  ipv6 address 101::222/64
  fabric forwarding mode anycast-gateway

interface Vlan1100
  description "L3VRF.VLANNUM.0.222"
  no shutdown
  vrf member vrf100
  ip address 100.1.0.222/16
  ipv6 address 100:1::222/64
  fabric forwarding mode anycast-gateway

interface Vlan1101
  description "L3VRF.VLANNUM.0.222"
  no shutdown
  vrf member vrf101
  ip address 101.1.0.222/16
  ipv6 address 101:1::222/64
  fabric forwarding mode anycast-gateway

interface Vlan2100
  no shutdown
  vrf member vrf100
  ip forward
  ipv6 address use-link-local-only

interface Vlan2101
  no shutdown
  vrf member vrf101
  ip forward
  ipv6 address use-link-local-only
```

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  multisite border-gateway interface loopback2
  member vni 2000100-2000110
    suppress-arp
    mcast-group 227.1.1.1
  member vni 2000120-2000150
    suppress-arp
    mcast-group 227.1.1.1
  member vni 2001100-2001110
    suppress-arp
    mcast-group 227.1.1.1
  member vni 3000100-3000110 associate-vrf
  member vni 3100100-3100110 associate-vrf

interface Ethernet1/22
  description "BGW11 to BGW2"
  medium p2p
  ip unnumbered loopback0
  ip ospf cost 40
  ip ospf network point-to-point
  ip router ospf 12 area 0.0.0.0
  no shutdown
  evpn multisite dci-tracking

interface Ethernet1/25
  description "BGW11 to Spine11"
  medium p2p
  ip unnumbered loopback0
  ip ospf cost 40
  ip ospf network point-to-point
  ip router ospf 1 area 0.0.0.0
  no shutdown
  evpn multisite fabric-tracking

interface Ethernet1/27
  description "BGW11 to Spine12"
  medium p2p
  ip unnumbered loopback0
  ip ospf cost 40
  ip ospf network point-to-point
  ip router ospf 1 area 0.0.0.0
  no shutdown
  evpn multisite fabric-tracking

interface Ethernet1/34
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 14,24
  no shutdown

interface loopback0
  ip address 1.1.11.0/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode

interface loopback1
  ip address 1.1.11.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode

interface loopback2
```

```
  ip address 11.11.11.11/32
  ip router ospf 12 area 0.0.0.0
  ip pim sparse-mode

router ospf 1
  redistribute direct route-map permit-all
router ospf 12
  redistribute direct route-map permit-all
ip load-sharing address source-destination rotate 32 universal-id 1

router bgp 1
  log-neighbor-changes
  address-family l2vpn evpn
    maximum-paths 8
    maximum-paths ibgp 8
    additional-paths send
    additional-paths receive
    additional-paths selection route-map passall
  neighbor 1.2.11.1
    remote-as 1
    description "SPINE-11"
    update-source loopback1
    address-family l2vpn evpn
      send-community extended
  neighbor 1.2.12.1
    remote-as 1
    description "SPINE-12"
    update-source loopback1
    address-family l2vpn evpn
      send-community extended
  neighbor 2.1.2.1
    remote-as 2
    description "BGW-2"
    update-source loopback1
    ebgp-multihop 3
    peer-type fabric-external
    address-family ipv4 unicast
    address-family l2vpn evpn
      send-community extended
      rewrite-evpn-rt-asn
  vrf vrf100
    address-family ipv4 unicast
      redistribute direct route-map permit-all
      maximum-paths 8
      maximum-paths ibgp 8
      export-gateway-ip
    address-family ipv6 unicast
      redistribute direct route-map permit-all
      maximum-paths 8
      maximum-paths ibgp 8
      export-gateway-ip
  vrf vrf101
    address-family ipv4 unicast
      redistribute direct route-map permit-all
      maximum-paths 8
      maximum-paths ibgp 8
      export-gateway-ip
    address-family ipv6 unicast
      redistribute direct route-map permit-all
      maximum-paths 8
      maximum-paths ibgp 8
      export-gateway-ip
evpn
  vni 2000100 l2
```

```
    rd auto
    route-target import auto
    route-target export auto
  vni 2000101 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001100 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001101 l2
    rd auto
    route-target import auto
    route-target export auto
```

次の例は、サイト 1 の VTEP がローカル VNF（FW）を優先することを示しています。

```
leaf1# show bgp l2vpn evpn 200.100.1.1
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1.3.12.0:3
BGP routing table entry for [5]:[0]:[0]:[32]:[200.100.1.1]/224, version 77902
Paths: (4 available, best #2)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP iBGP Local

  Path type: internal, path is valid, not best reason: Neighbor Address, no labeled
nexthop
  Gateway IP: 100.0.0.12
  AS-Path: 99 100 , path sourced external to AS
    1.3.12.1 (metric 81) from 1.2.12.1 (1.2.12.0)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 3000100
      Received path-id 2
      Extcommunity: RT:1:3000100 ENCAP:8 Router MAC:00be.7547.13bf
      Originator: 1.3.12.0 Cluster list: 1.2.12.0

  Advertised path-id 2
  Path type: local, path is valid, not best reason: Locally originated, multipath, no
labeled nexthop
  Gateway IP: 100.0.0.11
  AS-Path: 99 100 , path sourced external to AS
    1.3.11.1 (metric 0) from 0.0.0.0 (1.3.11.0)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 3000100
      Received path-id 1
      Extcommunity: RT:1:3000100 ENCAP:8 Router MAC:d478.9bb3.c1a1
```

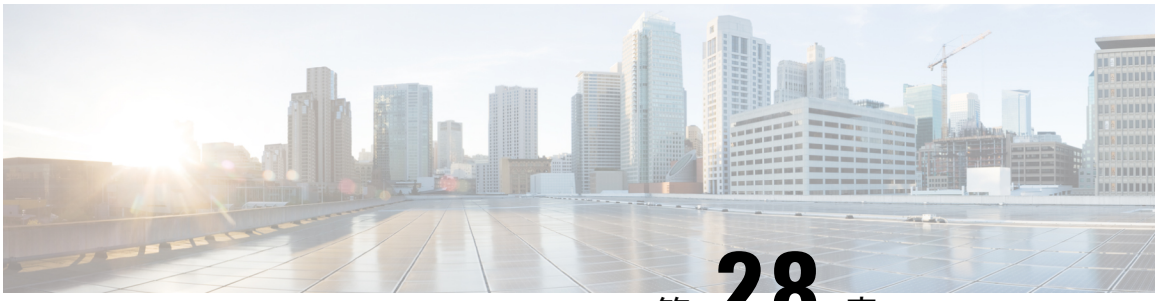
次の例は、サイト 2 からの VNF が使用されるようにローカル VNF を無効にする方法を示しています。BGP 隣接は、サイト 1 の VTEP11 と FW11 の間、および VTEP12 と FW12 の間でシャットダウンされます。

```
leaf1(config-router)# vrf vrf100
leaf1(config-router-vrf)# neighbor 100::11
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100::12
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100:1::11
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100:1::12
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100.0.0.11
```

```
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100.0.0.12
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100.1.0.11
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# neighbor 100.1.0.12
leaf1(config-router-vrf-neighbor)# shut
leaf1(config-router-vrf-neighbor)# end
```

次の例は、プレフィックスがサイト 2 からの VNF（FW）を使用することを示しています。

```
leaf1# show bgp l2vpn evpn 200.100.1.1
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 1:3000100
BGP routing table entry for [5]:[0]:[0]:[32]:[200.100.1.1]/224, version 97269
Paths: (3 available, best #3)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP iBGP Local

  Path type: internal, path is valid, not best reason: Neighbor Address, no labeled
nexthop
  Gateway IP: 100.1.0.21
  AS-Path: 2 99 100 , path sourced external to AS
    11.11.11.11 (metric 20) from 1.2.12.1 (1.2.12.0)
      Origin IGP, MED 2000, localpref 100, weight 0
      Received label 3000100
      Received path-id 2
      Extcommunity: RT:1:3000100 SOO:03030100:00000000 ENCAP:8
          Router MAC:0200.0b0b.0b0b
      Originator: 1.1.12.0 Cluster list: 1.2.12.0
```



第 28 章

EVPN 分散型 NAT

EVPN 分散型 NAT

Cisco NX-OS リリース 10.2(1)F 以降では、N9K-C9336C-FX2、N9K-C93240YC-FX2、N9K-C93360YC-FX2 TOR スイッチで EVPN 分散 NAT 機能がサポートされています。分散型 Elastic NAT機能は、VXLANトポロジのリーフとスパインでNATを有効にします。

EVPN分散NATのガイドラインと制限事項

EVPN分散型NATは次をサポートします。

- 最大 8192 の NAT 変換
- スタティック NAT
- IPv4 NAT
- VRF対応NATでの一致
- スタティック内部設定のアドルート

EVPN分散型NATは、次をサポートしません。

- IPv6 NAT
- ダイナミック NAT
- NATモビリティ
- サブネットベースのフィルタリング
- ルールごとの統計情報
- NATはvPCを認識しません。NAT設定はｖＰＣピアの両方に同一でなければなりません。

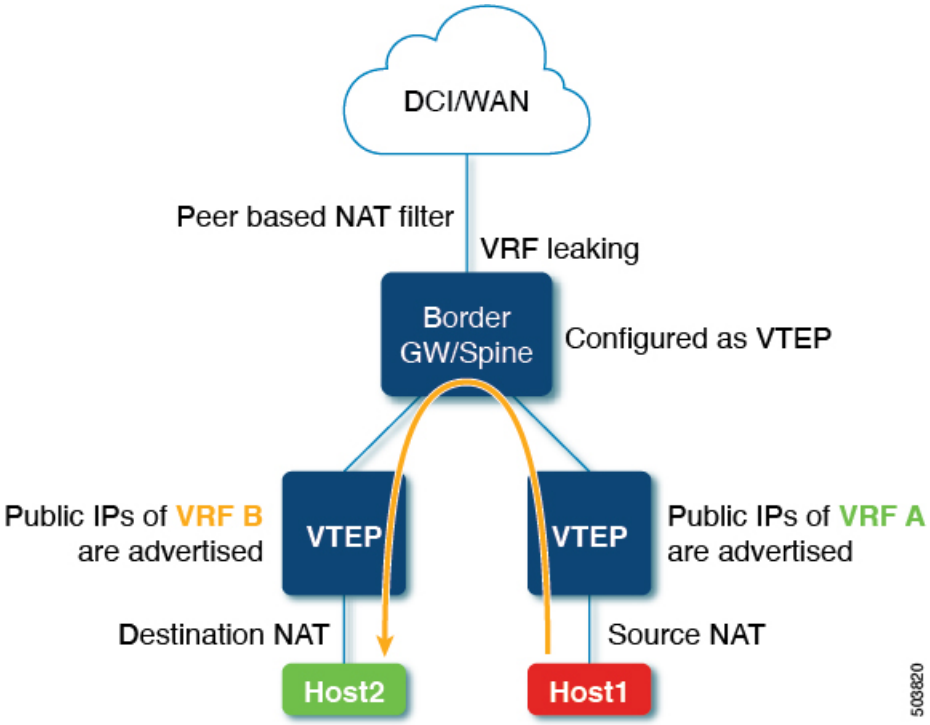
EVPN 分散型 NAT

- 送信元ホストと宛先ホストが同じVRFにある場合、ファブリック内では通常のNATを使用できます。EVPN分散NATは、同じVRF内ではサポートされません。異なるVRF間でサポートされます。

EVPN分散NATトポロジ

次のトポロジは、VTEPでのEVPN分散NAT設定を示しています。

図 *67 : EVPN*分散*NAT*設定トポロジ



上記のトポロジでは、次のようになります。

- EVPN分散NATは、VTEPでのみ設定されます。
- スパインには、EVPN分散NAT関連の設定は必要ありません。
- スパインはVTEPとして設定されます。
- VxLANアンダーレイルーティングプロトコルを使用した到達可能性のために、ルートだけがスパインにリークされます。
- 送信元と宛先NATは両方のリーフで設定されます。
- 送信元NATは、送信元に直接接続されたスイッチで実行されます。
- 宛先NATは、宛先に直接接続されたスイッチで実行されます。
- 送信元と宛先の両方が同じスイッチ上にある場合、最初に送信元NATが実行されます。パケットはスパインを介してループされ、宛先NATが実行されます。

- ホストは、要件に応じて、プライベートIPアドレスまたはパブリックIPアドレスを使用してトラフィックを送信できます。
- VXLANピアベースNATフィルタリングが設定されます。

ピアベースNATフィルタ

- ピアベースNATフィルタは、設定されたトンネルエンドポイント宛てのフローに対してのみNATを許可し、残りのフローは影響を受けません。
- ピアベースNATフィルタは、多数のプレフィックスをNAT変換する必要がある場合に役立ちます。
- NAT ACL領域は、ピアベースのNATフィルタが機能するように最初に切り分けられる必要があります。
- 境界ノードでピアベースのフィルタを設定できます。
- ピアベースNATフィルタは、集中型VRFリークが設定されているサービスリーフなどのVRF間ケースに役立ちます。
- を使用してピアベースNATフィルタを設定できます。<peer-ip>コマンド。**system nve nat peer-ip**

VRF 対応 NAT

- VRF対応NATにより、スイッチはVRF（仮想ルーティングおよび転送インスタンス）のアドレス空間を認識し、パケットを変換できます。これにより、NAT機能は2つのVRF間で使用される重複アドレス空間のトラフィックを変換できます。
- コマンドを使用してFPタイルベースのNATを有効にできます。**system routing vrf-aware-nat**
- VRF対応NATの詳細については、『Cisco Nexus 9000 NX-OS Interfaces Configuration Guide』を参照してください。https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/102x/configuration/interfaces/cisco-nexus-9000-nx-os-interfaces-configuration-guide-102x/b-cisco-nexus-9000-nx-os-interfaces-configuration-guide-93x_chapter_01011.html#concept_6EB0DB9C8EDC40FB8C21EAA918A56627

EVPN分散NATの設定

次に、リーフ1のEVPN分散NAT設定を示します。

```
feature bgp
feature interface-vlan
feature vn-segment-vlan-based
feature nat
feature nv overlay

hardware access-list tcam region nat 512  (Carves NAT TCAM)


system routing vrf-aware-nat
system nve nat peer-ip 100.100.100.3      (peer-ip is the Spine address which is leaking
```

EVPN 分散型 NAT

```
 the route)

ip nat inside source static 21.1.1.10 172.21.1.10 vrf vrf1 match-in-vrf add-route

ip nat inside source static 31.1.1.10 172.31.1.10 vrf vrf2 match-in-vrf add-route

vlan 202
  vn-segment 20202

vlan 301
  vn-segment 20301

vlan 3200
  vn-segment 33200

vlan 3300
 vn-segment 33300


interface Vlan202
  no shutdown
  vrf member vrf1
  ip address 22.1.1.1/24
  fabric forwarding mode anycast-gateway
  ip nat inside

interface Vlan3200
  no shutdown
  vrf member vrf1
  ip forward
  ip nat outside

interface Vlan301
  no shutdown
  vrf member vrf2
  ip address 31.1.1.1/24
  fabric forwarding mode anycast-gateway
  ip nat inside

interface Ethernet1/11
  switchport mode trunk

interface Ethernet1/35
  switchport mode trunk

vrf context vrf1
  vni 33200
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn

vrf context vrf2
  vni 33300
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn

router bgp 100
  vrf vrf1
    address-family ipv4 unicast
      network 172.21.1.10/32
      advertise l2vpn evpn
```

```
vrf vrf2
    address-family ipv4 unicast
      network 172.31.1.10/32
      advertise l2vpn evpn
```

次に、リーフ2のEVPN分散NAT設定を示します。

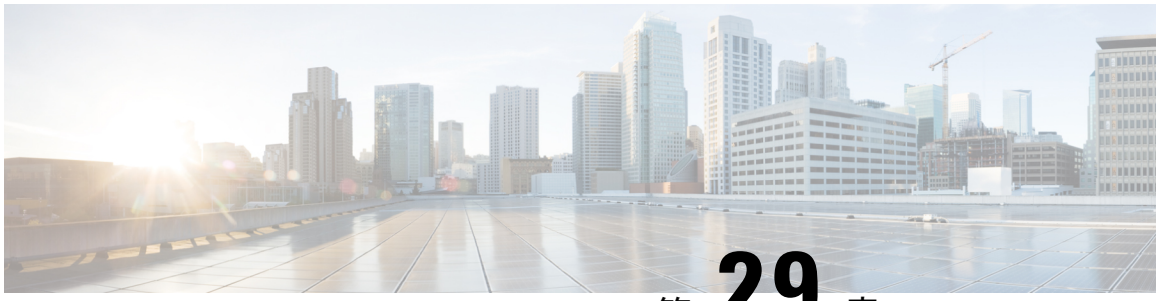
```
feature bgp
feature interface-vlan
feature vn-segment-vlan-based
feature nat
feature nv overlay

system routing vrf-aware-nat
system nve nat peer-ip 100.100.100.3   (peer-ip is the spine address which is leaking
the route)

ip nat inside source static 21.1.1.20 172.21.1.20 vrf vrf1 match-in-vrf add-route

ip nat inside source static 31.1.1.20 172.31.1.20 vrf vrf2 match-in-vrf add-route

vlan 202
  vn-segment 20202

vlan 301
  vn-segment 20301

vlan 3200
  vn-segment 33200

vlan 3300
 vn-segment 33300

interface Vlan202
  no shutdown
  vrf member vrf1
  ip address 22.1.1.1/24
  fabric forwarding mode anycast-gateway
  ip nat inside


interface Vlan3200
  no shutdown
  vrf member vrf1
  ip forward
  ip nat outside

interface Vlan301
  no shutdown
  vrf member vrf2
  ip address 31.1.1.1/24
  fabric forwarding mode anycast-gateway
  ip nat inside

interface Vlan3300
  no shutdown
  vrf member vrf2
  ip forward
  ip nat outside

interface Ethernet1/16
  switchport
  switchport mode trunk

interface Ethernet1/43
```

EVPN 分散型 NAT

```
  switchport
  switchport mode trunk

vrf context vrf1
  vni 33200
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
vrf context vrf2
  vni 33300
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn


router bgp 100
  vrf vrf1
    address-family ipv4 unicast
      network 172.21.1.20/32
      advertise l2vpn evpn
vrf vrf2
    address-family ipv4 unicast
      network 172.31.1.20/32
     advertise l2vpn evpn
```

次のshowコマンドは、EVPN分散型NATのスイッチで設定された絶縁ポリシーを表示します。

```
show ip nat translations
Pro Inside global Inside local Outside local Outside global
any 174.2.216.2 42.2.216.2 --- ---
any 174.3.217.2 42.3.217.2 --- ---
```



第 29 章

VXLAN BGP EVPN 中の DHCP リレーの概要

DHCP リレーは、ホストと DHCP サーバ間で DHCP パケットを転送するために使用されます。VXLAN VTEP は、マルチテナント VXLAN 環境で DHCP リレー サービスを提供することにより、リレー エージェントとして動作できます。

DHCP リレーを使用する場合、DHCP メッセージは同じスイッチ内を双方向に送信されることが必要です。DHCP リレーの GiAddr（ゲートウェイ IP アドレス）は一般に、スコープの選択と DHCP 応答メッセージに使用されます。分散 IP エニーキャスト ゲートウェイを備えた VXLAN ファブリックでは、DHCPメッセージは、それぞれのゲートウェイ IP アドレス（GiAddr）をホストする任意のスイッチに返すことができます。

ソリューションには、各スイッチのスコープ選択と一意の IP アドレスの異なる方法が必要です。スイッチごとの固有ループバック インターフェイスは、正しいスイッチに応答するための GiAddr になります。Option 82（dhcp option vpn）は、L2VNI に基づくスコープ選択に使用されます。

マルチテナント EVPN 環境で DHCP リレーは、オプション 82 の次のサブオプションを使用します。

- サブオプション 151(0x97)：仮想サブネットの選択（RFC#6607 で定義）

  MPLS-VPN および VXLAN EVPN マルチテナント環境中の DHCP サーバへの VRF 関連情報の伝達に使用されます。

- サブオプション 11(0xb)：サーバ IDに のオーバーライド（RFC#5107 で定義）

  サーバ識別子（サーバ ID）のオーバーライド サブオプションは、DHCP リレー エージェントによるサーバ ID オプションへの新しい値の指定を可能にし、これは DHCP サーバにより応答パケットに挿入されます。このサブオプションによって DHCP リレー エージェントは実際の DHCP サーバとして機能するようになり、たとえば renew 要求は DHCP サーバではなくリレー エージェントに直接届くようになります。サーバ ID オーバーライド サブオプションには着信インターフェイスの IP アドレスが含まれており、これはクライアントからアクセス可能なリレーエージェント上の IP アドレスです。この情報を使用して、DHCP クライアントは renew および release 要求パケットをすべてリレー エージェントへ送ります。リレー エージェントは適切なサブオプションをすべて付加した後、renew および release 要求パケットをオリジナルの DHCP サーバに転送します。この機能におけるシ

スコ独自の実装は、サブオプション 152（0x98）です。機能の制御には、**ip dhcp relay sub-option type cisco** コマンドを使用できます。

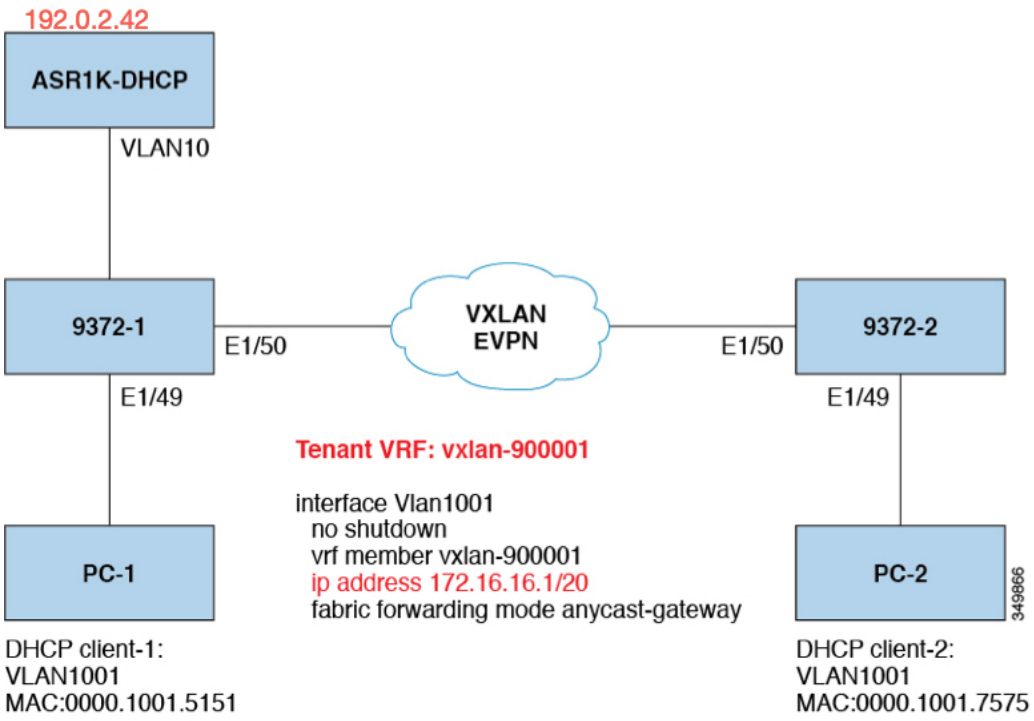
- サブオプション 5（0x5）：リンクの選択（RFC#3527 で定義）

  リンクの選択サブオプションが提供するのは、DHCP クライアントが存在するサブネット/リンクを、リレー エージェントとの通信に DHCP サーバが使用するゲートウェイ アドレス（giaddr）から分離するための機構です。リレー エージェントは正しいサブスクライバ サブネットにサブオプションを設定し、DHCP サーバはこの値を使用して giaddr 値ではなく IP アドレスを割り当てます。リレー エージェントは、giaddr を自身の IP アドレスに設定することで、DHCP メッセージがネットワーク上を転送できるようにします。この機能におけるシスコ独自の実装は、サブオプション 150（0x96）です。機能の制御には、**ip dhcp relay sub-option type cisco** コマンドを使用できます。

- VXLAN BGP EVPN 中の DHCP リレーの例 （629 ページ）
- VTEP の DHCP リレー （630 ページ）
- テナント VRF にあるクライアントと異なるレイヤ 3 デフォルト VRF にあるサーバ （630 ページ）
- テナント VRF（SVI X）にあるクライアントと同じテナント VRF（SVI Y）にあるサーバ （634 ページ）
- テナント VRF（VRF X）にあるクライアントと異なるテナント VRF（VRF Y）にあるサーバ （638 ページ）
- テナント VRF にあるクライアントと非デフォルトの非 VXLAN VRF にあるサーバ （640 ページ）
- vPC ピアの設定例 （643 ページ）
- vPC VTEP DHCP リレーの設定例 （645 ページ）

VXLAN BGP EVPN 中の DHCP リレーの例

図 *68* : トポロジの例



トポロジの特性：

- スイッチ 9372-1 と 9372-2 は、VXLAN ファブリックに接続された VTEP です。
- client1 と client2 は、vlan1001 中の DHCP クライアントです。これらはテナント VRF vxlan-900001 に属します。
- DHCP サーバは ASR1K であり、これは vlan10 に存在するルータです。
- DHCP サーバ設定

```
ip vrf vxlan900001
ip dhcp excluded-address vrf vxlan900001 172.16.16.1 172.16.16.9
ip dhcp pool one
 vrf vxlan900001
 network 172.16.16.0 255.240.0.0
 defaultrouter 172.16.16.1
```

VTEP の DHCP リレー

次に示したのは、一般的な展開シナリオです。

- テナント VRF にあるクライアントと異なるレイヤ 3 デフォルト VRF にあるサーバ。
- テナント VRF（SVI X）にあるクライアントと同じテナント VRF（SVI Y）にあるサーバ。
- テナント VRF（VRF X）にあるクライアントと異なるテナント VRF（VRF Y）にあるサーバ。
- テナント VRF にあるクライアントと非デフォルトの非 VXLAN VRF にあるサーバ。

次に示すのは、これとは異なるシナリオとして、vlan10 を別の VRF に移動させたものです。

テナント VRF にあるクライアントと異なるレイヤ 3 デフォルト VRF にあるサーバ

DHCP サーバ（192.0.2.42）をデフォルト VRF に設置して、9372-1 と 9372-2 の両方からデフォルト VRF を介してそこに到達可能であることを確認します。

```
9372-1# sh run int vl 10

!Command: show running-config interface Vlan10
!Time: Mon Aug 24 07:51:16 2018

version 7.0(3)I1(3)

interface Vlan10
  no shutdown
  ip address 192.0.2.25/24
  ip router ospf 1 area 0.0.0.0

9372-1# ping 192.0.2.42 cou 1

PING 192.0.2.42 (192.0.2.42): 56 data bytes
64 bytes from 192.0.2.42: icmp_seq=0 ttl=254 time=0.593 ms
- 192.0.2.42 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
roundtrip min/avg/max = 0.593/0.592/0.593 ms

9372-2# ping 192.0.2.42 cou 1
PING 192.0.2.42 (192.0.2.42): 56 data bytes
64 bytes from 192.0.2.42: icmp_seq=0 ttl=252 time=0.609 ms
- 192.0.2.42 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.609/0.608/0.609 ms
```

DHCP リレー設定

- 9372-1

```
9372—1# sh run dhcp

!Command: show running—config dhcp
!Time: Mon Aug 24 08:26:00 2018

version 7.0(3) I1(3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface Vlan1001
  ip dhcp relay address 192.0.2.42 use—vrf default
```

- 9372-2

```
9372-2# sh run dhcp

!Command: show running—config dhcp
!Time: Mon Aug 24 08:26:16 2018

version 7.0(3)11(3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interfaoe Vlan1001
  ip dhcp relay address 192.0.2.42 use—vrf default
```

debug コマンドの出力例

- 次に示すのは、DHCP のインタラクティブ シーケンスのパケット ダンプです。

```
9372-1# ethanalyzer local interface inband display-filter
"udp.srcport==67 or udp.dstport==67" limit-captured frames 0

Capturing on inband
20180824 08:35:25.066530 0.0.0.0 -> 255.255.255.0 DHCP DHCP Discover - Transaction
ID 0x636a38fd
20180824 08:35:25.068141 192.0.2.25 -> 192.0.2.42 DHCP DHCP Discover - Transaction
ID 0x636a38fd
20180824 08:35:27.069494 192.0.2.42 -> 192.0.2.25 DHCP DHCP Offer Transaction - ID
0x636a38fd
20180824 08:35:27.071029 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer Transaction -
ID 0x636a38fd
20180824 08:35:27.071488 0.0.0.0 -> 255.255.255.0 DHCP DHCP Request Transaction -
ID 0x636a38fd
20180824 08:35:27.072447 192.0.2.25 -> 192.0.2.42 DHCP DHCP Request Transaction -
ID 0x636a38fd
```

```
20180824 08:35:27.073008 192.0.2.42 -> 192.0.2.25 DHCP DHCP ACK Transaction - ID
0x636a38fd
20180824 08:35:27.073692 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK Transaction - ID
 0x636a38fd
```

（注） Ethanalyzer はすべての DHCP パケットをキャプチャできない可能性がありますが、これは、フィルタ使用時のインバンドの解釈に問題があるためです。これは SPAN を使用することで回避できます。

- DHCP Discover パケット 9372-1 は DHCP サーバに送信されています。

  giaddr は 192.0.2.25（vlan10 の IP アドレス）に設定され、それに応じてサブオプション 5/11/151 を設定します。

```
Bootp flags: 0x0000 (unicast)
client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 192.0.2.25 (192.0.2.25)
client MAC address Hughes_01:51:51 (00:00:10:01:51:51)
client hardware address padding: 00000000000000000000
Server host name not given
Boot file name not given
Magic cookie: DHCP
Option: (53) DHCP Message Type
  Length: 1
  DHCP: Discover (1)
Option: (55) Parameter Request List
  Length: 4
  Parameter Request List Item: (1) Subnet Mask
  Parameter Request List Item: (3) Router
  Parameter Request List Item: (58) Renewal Time Value
  Parameter Request List Item: (59) Rebinding Time Value
Option: (61) client identifier
  Length: 7
  Hardware type: Ethernet (0x01)
  Client MAC address: Hughes_01:51:51 (00:00:10:01:51:51)
Option: (82) Agent Information Option
  Length: 47
Option 82 Suboption: (1) Agent Circuit ID
  Length: 10
  Agent Circuit ID: 01080006001e88690030
Option 82 Suboption: (2) Agent Remote ID
  Length: 6
  Agent Remote ID: f8c2882333a5
Option 82 Suboption: (151) VRF name/VPN ID
Option 82 Suboption: (11) Server ID Override
  Length: 4
  Server ID Override: 172.16.16.1 (172.16.16.1)
Option 82 Suboption: (5) Link selection
  Length: 4
  Link selection: 172.16.16.0 (172.16.16.0)
```

```
ASR1K-DHCP# sh ip dhcp bin
Bindings from all pools not associated with VRF:
IP address ClientID/ Lease expiration Type State Interface
         Hardware address/
         User name

Bindings from VRF pool vxlan900001:
IP address ClientID/ Lease expiration Type State Interface
         Hardware address/
         User name
172.16.16.10 0100.0010.0175.75 Aug 25 2018 09:21 AM Automatic Active
GigabitEthernet2/1/0
172.16.16.11 0100.0010.0151.51 Aug 25 2018 08:54 AM Automatic Active
GigabitEthernet2/1/0

9372-1# sh ip route vrf vxlan900001
IP Route Table for VRF "vxlan900001"
'*' denotes best ucast nexthop
'**' denotes best mcast nexthop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.11.11.11/8, ubest/mbest: 2/0, attached
  *via 10.11.11.11, Lo1, [0/0], 18:31:57, local
  *via 10.11.11.11, Lo1, [0/0], 18:31:57, direct
10.22.22.22/8, ubest/mbest: 1/0
  *via 1.2.2.2%default, [200/0], 18:31:57, bgp65535,internal, tag 65535 (evpn)segid:
 900001 tunnelid: 0x2020202
encap: VXLAN

172.16.16.0/20, ubest/mbest: 1/0, attached
 *via 172.16.16.1, Vlan1001, [0/0], 18:31:57, direct
172.16.16.1/32, ubest/mbest: 1/0, attached
 *via 172.16.16.1, Vlan1001, [0/0], 18:31:57, local
172.16.16.10/32, ubest/mbest: 1/0
 *via 1.2.2.2%default, [200/0], 00:00:47, bgp65535,internal, tag 65535 (evpn)segid:
 900001 tunnelid: 0x2020202
encap: VXLAN

172.16.16.11/32, ubest/mbest: 1/0, attached
 *via 172.16.16.11, Vlan1001, [190/0], 00:28:10, hmm

9372-1# ping 172.16.16.11 vrf vxlan900001 count 1
PING 172.16.16.11 (172.16.16.11): 56 data bytes
64 bytes from 172.16.16.11: icmp_seq=0 ttl=63 time=0.846 ms
- 172.16.16.11 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.846/0.845/0.846 ms

9372-1# ping 172.16.16.10 vrf vxlan900001 count 1
PING 172.16.16.10 (172.16.16.10): 56 data bytes
64 bytes from 172.16.16.10: icmp_seq=0 ttl=62 time=0.874 ms
- 172.16.16.10 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.874/0.873/0.874 ms
```

テナント VRF（SVI X）にあるクライアントと同じテナント VRF（SVI Y）にあるサーバ

DHCP サーバ（192.0.2.42）を vxlan-900001 の VRF に設置して、9372-1 と 9372-2 の両方から vxlan-900001 の VRF を介してそこに到達可能であることを確認します。

```
9372-1# sh run int vl 10

!Command: show running-config interface Vlan10
!Time: Mon Aug 24 09:10:26 2018

version 7.0(3)I1(3)

interface Vlan10
  no shutdown
  vrf member vxlan-900001
  ip address 192.0.2.25/24
```

172.16.16.1 はすべての VTEP に設定された vlan1001 のエニーキャスト アドレスであるため、DHCP サーバからの応答をオリジナルの DHCP リレー エージェントへ確実に配送させるためには、DHCP リレー パケットの送信元アドレスとして一意のアドレスをピック アップする必要があります。このシナリオでは、loopback1 を使用しており、loopback1 には VRF vxlan-900001 のどこからでも到達可能であることを確認する必要があります。

```
9372-1# sh run int lo1

!Command: show running-config interface loopback1
!Time: Mon Aug 24 09:18:53 2018

version 7.0(3)I1(3)

interface loopback1
  vrf member vxlan-900001
  ip address 10.11.11.11/8

9372-1# ping 192.0.2.42 vrf vxlan900001 source 10.11.11.11 cou 1
PING 192.0.2.42 (192.0.2.42) from 10.11.11.11: 56 data bytes
64 bytes from 192.0.2.42: icmp_seq=0 ttl=254 time=0.575 ms
- 192.0.2.42 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.575/0.574/0.575 ms

9372-2# sh run int lo1

!Command: show running-config interface loopback1
!Time: Mon Aug 24 09:19:30 2018

version 7.0(3)I1(3)

interface loopback1
  vrf member vxlan900001
  ip address 10.22.22.22/8
```

```
9372-2# ping 192.0.2.42 vrf vxlan-900001 source 10.22.22.22 cou 1
PING 192.0.2.42 (192.0.2.42) from 10.22.22.22: 56 data bytes
64 bytes from 192.0.2.42: icmp_seq=0 ttl=253 time=0.662 ms
- 192.0.2.42 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.662/0.662/0.662 ms
```

DHCP リレー設定

- 9372-1

```
9372—1# sh run dhcp

!Command: show running-config dhcp
!Time: Mon Aug 24 08:26:00 2018

version 7.0(3)11(3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
I4ip dhcp relay information option vpn
ipv6 dhcp relay

interface Vlan1001
  ip dhcp relay address 192.0.2.42
  ip dhcp relay source—interface loopback1
```

- 9372-2

```
9372—2# sh run dhcp

!Command: show running-config dhcp
!Time: Mon Aug 24 08:26:16 2018

version 7.0(3) 11(3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface Vlan1001
  ip dhcp relay address 192.0.2.42
  ip dhcp relay source—interface loopback1
```

debug コマンドの出力例

- 次に示すのは、DHCP のインタラクティブ シーケンスのパケット ダンプです。

```
9372-1# ethanalyzer local interface inband display-filter
"udp.srcport==67 or udp.dstport==67" limit-captured frames 0
```

```
Capturing on inband
20180824 09:31:38.129393 0.0.0.0 -> 255.255.255.0 DHCP DHCP Discover - Transaction
ID 0x860cd13
20180824 09:31:38.129952 10.11.11.11 -> 192.0.2.42 DHCP DHCP Discover - Transaction
 ID 0x860cd13
20180824 09:31:40.130134 192.0.2.42 -> 10.11.11.11 DHCP DHCP Offer - Transaction ID
 0x860cd13
20180824 09:31:40.130552 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer - Transaction
ID 0x860cd13
20180824 09:31:40.130990 0.0.0.0 -> 255.255.255.0 DHCP DHCP Request - Transaction
ID 0x860cd13
20180824 09:31:40.131457 10.11.11.11 -> 192.0.2.42 DHCP DHCP Request - Transaction
ID 0x860cd13
20180824 09:31:40.132009 192.0.2.42 -> 10.11.11.11 DHCP DHCP ACK - Transaction ID
0x860cd13
20180824 09:31:40.132268 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK - TransactionID
0x860cd13
```

（注） Ethanalyzer はすべての DHCP パケットをキャプチャできない可能性がありますが、これは、フィルタ使用時のインバンドの解釈に問題があるためです。これは SPAN を使用することで回避できます。

- DHCP Discover パケット 9372-1 は DHCP サーバに送信されています。

  giaddr は 10.11.11.11（loopback1）に設定され、それに応じてサブオプション 5/11/151 を設定します。

```
Bootstrap Protocol
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 1
  Transaction ID: 0x0860cd13
  Seconds elapsed: O
  Bootp flags: 0x0000 (unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 0.0.0.0 (0.0.0.0)
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent iP address: 10.11.11.11 (10.11.11.11)
  Client MAC address: Hughes_01:51:51 (00:00:10:01:51:51)
  Client hardware address padding: 00000000000000000000
  Server host name not given
  Boot file name not given
  Magic cookie: DHCP
  Option: (53) DHCP Message Type
    Length: 1
    DHCP: Discover (1)
  Option: (55) Parameter Request List
  Option: (61) Client Identifier
  Option: (82) Agent Information Option
    Length: 47
  Option 82 suboption: (1) Agent Circuit ID
  Option 82 suboption: (151) Agent Remote ID
  Option 82 suboption: (11) Server ID Override
    Length: 4
```

```
    Server ID override: 172.16.16.1 (172.16.16.1)
  Option 82 suboption: (5) Link selection
    Length: 4
    Link selection: 172.16.16.0 (172.16.16.0)



ASR1K-DHCP# sh ip dhcp bin
Bindings from all pools not associated with VRF:
IP address ClientID/Lease expiration Type State Interface
         Hardware address/
         User name

Bindings from VRF pool vxlan-900001:
IP address ClientID/Lease expiration Type State Interface
         Hardware address/
         User name

172.16.16.10 0100.0010.0175.75 Aug 25 2018 10:02 AM Automatic Active
GigabitEthernet2/1/0
172.16.16.11 0100.0010.0151.51 Aug 25 2018 09:50 AM Automatic Active
GigabitEthernet2/1/0

9372-1# sh ip route vrf vxlan-900001
IP Route Table for VRF "vxlan-900001"
'*' denotes best ucast nexthop
'**' denotes best mcast nexthop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.11.11.11/8, ubest/mbest: 2/0, attached
  *via 10.11.11.11, Lo1, [0/0], 19:13:56, local
  *via 10.11.11.11, Lo1, [0/0], 19:13:56, direct
10.22.22.22/8, ubest/mbest: 1/0
  *via 2.2.2.2%default, [200/0], 19:13:56, bgp65535,internal, tag 65535 (evpn)segid:
 900001 tunnelid: 0x2020202
encap: VXLAN
172.16.16.0/20, ubest/mbest: 1/0, attached
  *via 172.16.16.1, Vlan1001, [0/0], 19:13:56, direct
172.16.16.1/32, ubest/mbest: 1/0, attached
  *via 172.16.16.1, Vlan1001, [0/0], 19:13:56, local
172.16.16.10/32, ubest/mbest: 1/0
  *via 2.2.2.2%default, [200/0], 00:01:27, bgp65535,
internal, tag 65535 (evpn)segid: 900001 tunnelid: 0x2020202
encap: VXLAN
172.16.16.11/32, ubest/mbest: 1/0, attached
  *via 172.16.16.11, Vlan1001, [190/0], 00:13:56, hmm
192.0.2.20/24, ubest/mbest: 1/0, attached
  *via 192.0.2.25, Vlan10, [0/0], 00:36:08, direct
192.0.2.25/24, ubest/mbest: 1/0, attached
  *via 192.0.2.25, Vlan10, [0/0], 00:36:08, local
9372-1# ping 172.16.16.10 vrf vxlan-900001 cou 1
PING 172.16.16.10 (172.16.16.10): 56 data bytes
64 bytes from 172.16.16.10: icmp_seq=0 ttl=62 time=0.808 ms
- 172.16.16.10 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.808/0.808/0.808 ms

9372-1# ping 172.16.16.11 vrf vxlan-900001 cou 1
PING 172.16.16.11 (172.16.16.11): 56 data bytes
64 bytes from 172.16.16.11: icmp_seq=0 ttl=63 time=0.872 ms
- 172.16.16.11 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
```

```
round-trip min/avg/max = 0.872/0.871/0.872 ms
```

テナント VRF（VRF X）にあるクライアントと異なるテナント VRF（VRF Y）にあるサーバ

DHCP サーバは他のテナント VRF vxlan-900002 の中に置かれて、DHCP 応答パケットがオリジナルのリレー エージェントにアクセスできるようにされます。ここでは loopback2 を使用して、DHCP リレー パケットの送信元アドレスとされているエニーキャスト IP アドレスをすべて回避します。

```
9372-1# sh run int vl 10
!Command: show runningconfig interface Vlan10
!Time: Tue Aug 25 08:48:22 2018

version 7.0(3)I1(3)
interface Vlan10
  no shutdown
  vrf member vxlan900002
  ip address 192.0.2.40/24

9372-1# sh run int lo2
!Command: show runningconfig interface loopback2
!Time: Tue Aug 25 08:48:57 2018
version 7.0(3)I1(3)
interface loopback2
  vrf member vxlan900002
  ip address 10.33.33.33/8

9372-2# sh run int lo2
!Command: show runningconfig interface loopback2
!Time: Tue Aug 25 08:48:44 2018
version 7.0(3)I1(3)
interface loopback2
  vrf member vxlan900002
  ip address 10.44.44.44/8

9372-1# ping 192.0.2.42 vrf vxlan-900002 source 10.33.33.33 cou 1
PING 192.0.2.42 (192.0.2.42) from 10.33.33.33: 56 data bytes
64 bytes from 192.0.2.42: icmp_seq=0 ttl=254 time=0.544 ms
- 192.0.2.42 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.544/0.544/0.544 ms

9372-2# ping 192.0.2.42 vrf vxlan-900002 source 10.44.44.44 count 1
PING 192.0.2.42 (192.0.2.42) from 10.44.44.44: 56 data bytes
64 bytes from 192.0.2.42: icmp_seq=0 ttl=253 time=0.678 ms
- 192.0.2.42 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.678/0.678/0.678 ms
```

DHCP リレー設定

- 9372-1

```
9372—1# sh run dhcp

!Command: show running—config dhcp
!Time: Mon Aug 24 08:26:00 2018

version 7.0(3) Ii (3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlanlOOl
  ip dhcp relay address 192.0.2.42 use—vrf vxlan—900002
  ip dhcp relay source—interface loopback2
```

- 9372-2

```
!Command: show running-config dhcp
!Time: Mon Aug 24 08:26:16 2018

version 7.0(3)11(3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlaniOOl
  ip dhcp relay address 192.0.2.42 use-vrf vxlan—900002
  ip dhcp relay source—interface loopback2
```

debug コマンドの出力例

- 次に示すのは、DHCP のインタラクティブ シーケンスのパケット ダンプです。

```
9372-1# ethanalyzer local interface inband display-filter "udp.srcport==67 or
udp.dstport==67" limit-captured-frames 0
Capturing on inband
20180825 08:59:35.758314 0.0.0.0 -> 255.255.255.0 DHCP DHCP Discover - Transaction
ID 0x3eebccae
20180825 08:59:35.758878 10.33.33.33 -> 192.0.2.42 DHCP DHCP Discover - Transaction
 ID 0x3eebccae
20180825 08:59:37.759560 192.0.2.42 -> 10.33.33.33 DHCP DHCP Offer - Transaction ID
 0x3eebccae
20180825 08:59:37.759905 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer - Transaction
ID 0x3eebccae
20180825 08:59:37.760313 0.0.0.0 -> 255.255.255.0 DHCP DHCP Request - Transaction
ID 0x3eebccae
20180825 08:59:37.760733 10.33.33.33 -> 192.0.2.42 DHCP DHCP Request - Transaction
ID 0x3eebccae
20180825 08:59:37.761297 192.0.2.42 -> 10.33.33.33 DHCP DHCP ACK - Transaction ID
```

```
0x3eebccae
20180825 08:59:37.761554 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK - Transaction ID
 0x3eebccae
```

- DHCP Discover パケット 9372-1 は DHCP サーバに送信されています。

  giaddr は 10.33.33.33（loopback2）に設定され、それに応じてサブオプション 5/11/151 を設定します。

```
Bootstrap Protocol
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 1
  Transaction ID: Ox3eebccae
  Seconds elapsed: O
  Bootp flags: 0x0000 (unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 0.0.0.0 (0.0.0.0)
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent IP address: 10.33.33.33 (10.33.33.33)
  Client MAC address: i-iughes_01:51:51 (00:00:10:01:51:51)
  Client hardware address padding: 00000000000000000000
  Server host name not given
  Boot file name not given
  Magic cookie: DHCP
  Option: (53) DHCP Message Type
    Length: 1
    DHCP: Discover (1)
  Option: (55) Parameter Request List
  Option: (61) client identifier
  Option: (82) Agent Information option
    Length: 47
  Option 82 Suboption: (1) Agent circuit W
  Option 82 suboption: (2) Agent Remote 10
  Option 82 suboption: (151) VRF name/VPN ID
  Option 82 Suboption: (11) Server ID Override
    Length: 4
    Server ID Override: 172.16.16.1 (172.16.16.1)
  Option 82 Suboption: (5) Link selection
    Length: 4
    Link selection: 172.16.16.0 (172.16.16.0)
```

テナント VRF にあるクライアントと非デフォルトの非 VXLAN VRF にあるサーバ

DHCP サーバは管理 VRF に配置され、M0 インターフェイスを介して到達可能です。それに応じて IP アドレスは 10.122.164.147 に変更されます。

```
9372-1# sh run int m0
!Command: show running-config interface mgmt0
!Time: Tue Aug 25 09:17:04 2018
```

```
version 7.0(3)I1(3)
interface mgmt0
  vrf member management
  ip address 10.122.165.134/8

9372-1# ping 10.122.164.147 vrf management cou 1
PING 10.122.164.147 (10.122.164.147): 56 data bytes
64 bytes from 10.122.164.147: icmp_seq=0 ttl=251 time=1.024 ms
- 10.122.164.147 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 1.024/1.024/1.024 ms

9372-2# sh run int m0
!Command: show running-config interface mgmt0
!Time: Tue Aug 25 09:17:47 2018
version 7.0(3)I1(3)
interface mgmt0
  vrf member management
  ip address 10.122.165.148/8

9372-2# ping 10.122.164.147 vrf management cou 1
PING 10.122.164.147 (10.122.164.147): 56 data bytes
64 bytes from 10.122.164.147: icmp_seq=0 ttl=251 time=1.03 ms
- 10.122.164.147 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 1.03/1.03/1.03 ms
```

DHCP リレー設定

- 9372-1

```
9372—1# sh run dhcp 9372—2# sh run dhcp

!Command: show running-config dhcp
!Time: Mon Aug 24 08:26:00 2018

version 7.0(3)11(3)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlanlOOl
  ip dhcp relay address 10.122.164.147 use—vrf management
```

- 9372-2

```
9372-2# sh run dhcp
!Command: show running-config dhcp
!Time: Tue Aug 25 09:17:47 2018

version 7.0(3)11(3)
feature dhcp

service dhcp
ip dhcp relay
```

```
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlanlOOl
ip dhcp relay address 10.122.164.147 use—vrf management
```

debug コマンドの出力例

- 次に示すのは、DHCP のインタラクティブ シーケンスのパケット ダンプです。

```
9372-1# ethanalyzer local interface inband display-filter "udp.srcport==67 or
udp.dstport==67" limit-captured-frames 0
Capturing on inband
20180825 09:30:54.214998 0.0.0.0 -> 255.255.255.0 DHCP DHCP Discover - Transaction
ID 0x28a8606d
20180825 09:30:56.216491 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer - Transaction
ID 0x28a8606d
20180825 09:30:56.216931 0.0.0.0 -> 255.255.255.0 DHCP DHCP Request - Transaction
ID 0x28a8606d
20180825 09:30:56.218426 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK - Transaction ID
 0x28a8606d

9372-1# ethanalyzer local interface mgmt display-filter "ip.src==10.122.164.147 or
ip.dst==10.122.164.147" limit-captured-frames 0
Capturing on mgmt0
20180825 09:30:54.215499 10.122.165.134 -> 10.122.164.147 DHCP DHCP Discover -
Transaction ID 0x28a8606d
20180825 09:30:56.216137 10.122.164.147 -> 10.122.165.134 DHCP DHCP Offer - Transaction
 ID 0x28a8606d
20180825 09:30:56.217444 10.122.165.134 -> 10.122.164.147 DHCP DHCP Request -
Transaction ID 0x28a8606d
20180825 09:30:56.218207 10.122.164.147 -> 10.122.165.134 DHCP DHCP ACK - Transaction
 ID 0x28a8606d
```

- DHCP Discover パケット 9372-1 は DHCP サーバに送信されています。

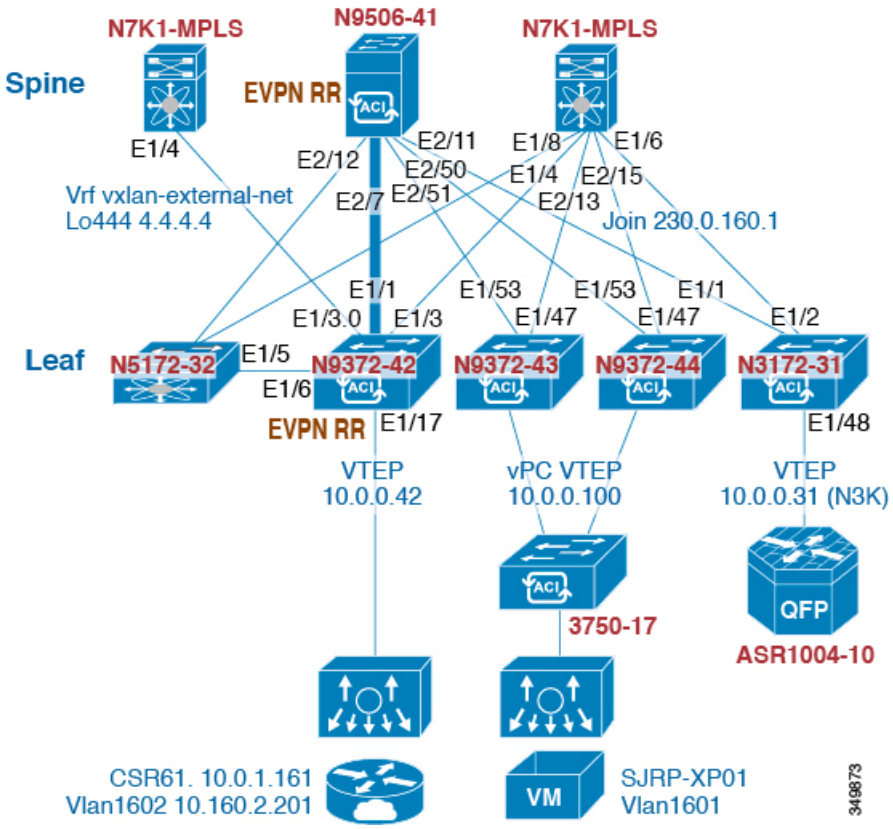
  giaddr は 10.122.165.134（mgmt0）に設定され、それに応じてサブオプション 5/11/151 を設定します。

```
Bootstrap Protocol
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 1
  Transaction ID: 0x28a8606d
  Seconds elapsed: O
  Bootp flags: 0x0000 (Unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 0.0.0.0 (0.0.0.0)
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent IP address: 10.122.165.134 (10.122.165.134)
  Client MAC address: Hughes_01:51:51 (00:00:10:01:51:51)
  Client hardware address padding: 00000000000000000000
  Server host name not given
  Boot file name not given
  Magic cookie: DHCP
  Option: (53) DHCP Message Type
    Length: 1
    DHCP: Discover (1)
```

```
Option: (55) Parameter Request List
Option: (61) Client identifier
Option: (82) Agent Information Option
  Length: 47
  Option 82 Suboption: (1) Agent Circuit ID
  Option 82 Suboption: (2) Agent Remote ID
  Option 82 Suboption: (151) VRF name/VPN ID
  Option 82 Suboption: (11) Server ID Override
    Length: 4
    Server ID Override: 172.16.16.1 (172.16.16.1)
  Option 82 Suboption: (5) Link selection
    Length: 4
    Link selection: 172.16.16.0 (172.16.16.0)
```

vPC ピアの設定例

次の例では、DHCP リレー設定用のオーバーレイ VLAN にある vPC ピア間のルーティングを設定します。



- DHCP サービスをイネーブルにします。

```
service dhcp
```

- DHCP リレーを設定します。

```
ip dhcp relay
ip dhcp relay information option
ip dhcp relay sub-option type cisco
ip dhcp relay information option vpn
```

- DHCP リレー サービスを必要とする VRF でループバックを作成します。

```
interface loopback601
  vrf member evpn-tenant-kk1
  ip address 192.0.2.36/24
  ip router ospf 1 area 0     /* Only required for vPC VTEP. */
```

- レイヤ 3 VRF BGP に LoX をアドバタイズします。

```
Router bgp 2
vrf X
   network 10.1.1.42/8
```

- VRF で SVI に DHCP リレーを設定します。

```
interface Vlan1601
  vrf member evpn-tenant-kk1
  ip address 10.160.1.254/8
  fabric forwarding mode anycast-gateway
  ip dhcp relay address 10.160.2.201
  ip dhcp relay source-interface loopback601
```

- レイヤ 3 VNI SVI を **ip forward** で設定します。

```
interface Vlan1600
  vrf member evpn-tenant-kk1
    ip forward
```

- vPC VRF のルーティング VLAN/SVI を作成します。

（注） vPC VTEP でのみ必要です。

```
Vlan 1605
interface Vlan1605
  vrf member evpn-tenant-kk1
  ip address 10.160.5.43/8
  ip router ospf 1 area 10.10.10.41
```

- VRF ルーティングを作成します。

（注） vPC VTEP でのみ必要です。

```
router ospf 1
vrf evpn-tenant-kk1
    router-id 10.160.5.43
```

vPC VTEP DHCP リレーの設定例

vPC VLAN など、MCT/ピア リンク全体で許可される VLAN を設定する必要性に応えるため、SVI は VLAN に関連付けることが可能であり、テナント VRF 内部で作成されます。これが OSPF など、アンダーレイ プロトコル付きのアンダーレイ ピアリングとなりますが、これはルーティング プロセスでインスタンス化されるテナント VRF を必要とします。

あるいは、ルーティング プロトコル中への SVI の配置およびルーティング プロセス下でのテナント VRF のインスタンス化の代わりに、MCT 全体の vPC ピア間でスタティック ルートを使用することが可能です。このアプローチにより、サーバからの応答が正しい場所に返され、各 VTEP が GiAddr について異なるループバック インターフェイスを使用することが保証されます。

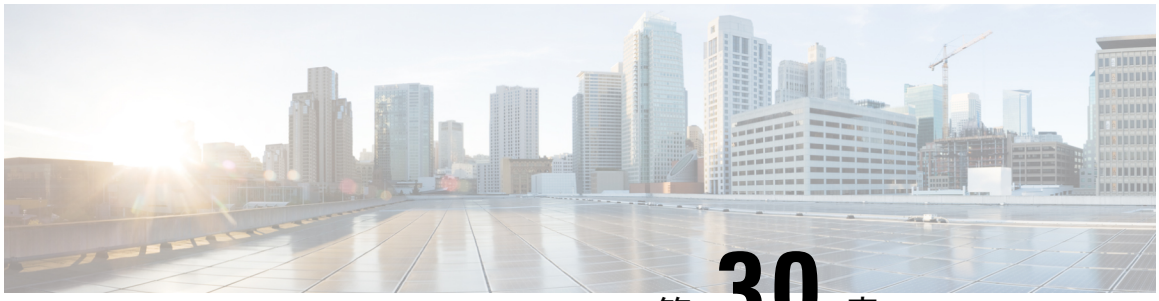
次に示すのは、これらの設定例です。

- アンダーレイ ルーティング内での SVI の設定：

```
/* vPC Peer-1 */

router ospf UNDERLAY
vrf tenant-vrf

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.1/16
  ip router ospf UNDERLAY area 0.0.0.0


/* vPC Peer-2 */

router ospf UNDERLAY
vrf tenant-vrf

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.2/16
  ip router ospf UNDERLAY area 0.0.0.0
```

- MCT 全体での vPC ピア間のスタティック ルートを使用した SVI 設定：

```
/* vPC Peer-1 */

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.1/16

vrf context tenant-vrf
ip route 192.168.1.2/16 192.168.1.1


/* vPC Peer-2 */

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.2/16

vrf context tenant-vrf
ip route 192.168.1.1/16 192.168.1.2
```



第 30 章

クロス コネクトの設定

この章は、次の内容で構成されています。

VXLAN クロス コネクトについて

この機能は、ある VTEP から別の VTEP へのデータおよび制御パケットのポイントツーポイント トンネリングを提供します。すべての接続回線は、一意のプロバイダー VNI の一部になります。BGP EVPN シグナリングは、プロバイダー VNI がファブリック内でどのように拡張されるかに基づいて、これらのエンドポイントを検出します。すべての内部 customer.1q タグはそのまま保持され、パケットはカプセル化VTEP でプロバイダーVNI にカプセル化されます。カプセル化解除エンドポイントでは、プロバイダー VNI はパケット内のすべての customer.1q タグを保持したまま、パケットを接続回線に転送します。

（注） Cross Connect と xconnect は同義語です。

VXLAN Cross Connect は vPC ファブリック ピアリングをサポートします。

VXLAN クロスコネクトは、次のスイッチで VXLAN ポイント ツーポイント機能を有効にします。

- Cisco Nexus 9332PQ
- Cisco Nexus 9336C-FX2
- Cisco Nexus 9372PX
- Cisco Nexus 9372PX-E
- Cisco Nexus 9372TX
- Cisco Nexus 9372TX-E
- Cisco Nexus 93120TX
- Cisco Nexus 93108TC-EX
- Cisco Nexus 93108TC-FX
- Cisco Nexus 93180LC-EX
- Cisco Nexus 93180YC-EX
- Cisco Nexus 93180YC-FX
- Cisco Nexus 93240YC-FX2
- Cisco Nexus N9K-C93180YC-FX3S
- Cisco Nexus 9316D-GX
- Cisco Nexus 9364C-GX
- Cisco Nexus 93600CD-GX

VXLAN Cross Connect は、VXLAN クラウド全体のすべての制御フレーム（CDP、LLDP、LACP、STP、BFD、および PAGP）のトンネリングを可能にします。

VXLAN クロス コネクトの注意事項と制限事項

VXLAN クロス コネクトには、次の注意事項と制限事項があります。

- Cisco NX-OS リリース 7.0(3)I7(4) から Cisco NX-OS リリース 9.2(x) コードに無停止でアップグレードを実行し、VLAN を作成して xconnect として設定する場合は、 **copy running-config startup-config** コマンドを入力してスイッチをリロードします。ボックスが Cisco NX-OS リリース 9.2(x) コードに破壊的にアップグレードされた場合、VLAN を xconnect として設定する際にリロードは必要ありません。
- MAC 学習は xconnect VNI では無効になり、トンネル アクセス ポートではホスト MAC は学習されません。
- BGP EVPN トポロジでのみサポートされます。

- 接続回線の LACP バンドリングはサポートされていません。
- 特定の VTEP でプロバイダー VNI に設定できる接続回線は1つだけです。
- VNI はポイントツーポイント方式でのみ拡張できます。ポイントツーマルチポイント トンネルはサポートされません。
- xconnect VLAN 上の SVI はサポートされていません。
- ARP 抑制は、xconnect VLAN VNI ではサポートされません。VLAN で ARP 抑制がイネーブルになっている場合、VLAN で xconnect をイネーブルにすると、xconnect 機能が優先されます。
- xconnect は次のスイッチではサポートされていません。
  - Cisco Nexus 9504
  - Cisco Nexus 9508
  - Cisco Nexus 9516
- xconnect VLAN の規模は、スイッチで使用可能なポートの数によって異なります。すべての xconnect VLAN は、すべての 4k カスタマー VLAN をトンネリングできます。
- vpc-vtep の xconnect または Crossconnect 機能には、vPC ピアリンクのネイティブ VLAN として backup-svi が必要です。
- リンク フラップを回避するために、ISSU/パッチのアクティブ化を試行する前に、すべての VTEP で NGAM xconnect hb-interval が 5000 ミリ秒に設定されていることを確認します。
- cfs プロセスのパッチをアクティブ化する前に、Ngoam xconnect hb-interval を最大値の 5000 ミリ秒に移動する必要があります。これにより、パッチのアクティブ化中のインターフェイス フラップが防止されます。
- VNI ごとの vPC 孤立トンネルポートは、vPC プライマリ スイッチまたはセカンダリ スイッチのいずれかに存在する必要があります。
- xconnect トンネル インターフェイスでの静的 MAC の設定はサポートされていません。
- xconnect は FEX ポートではサポートされません。
- vpc-vtep では、xconnect VLAN の両方の vPC ピアでスパニング ツリーを無効にする必要があります。
- Xconnect アクセス ポートは、すべての VTEP で NGAM を無効にした後にフラップする必要があります。
- VLAN を削除および追加した後、または VLAN から xconnect を削除した後は、物理ポートを NFAM でフラップする必要があります。
- Cisco NX-OS Release 9.3(3) 以降では、次のスイッチのサポートが追加されています。
  - Cisco Nexus C93600CD-GX

    Cisco Nexus C9364C-GX

Cisco Nexus C9316D-GX

- Cisco NX-OS リリース 10.2(3)F 以降、xconnect は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、xconnect は Cisco Nexus 9332D-H2R スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、xconnect は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、xconnect は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- VXLAN クロス コネクトは、マルチサイト ソリューションの一部としてサポートされていません。

VXLAN クロス コネクトの設定

この手順では、VXLAN クロスコネクト機能を設定する方法について説明します。

手順の概要

1. **configure terminal**
2. **vlan** *vlan-id*
3. **vn-segment** *vnid*
4. **xconnect**
5. **exit**
6. **interface** *type port*
7. **switchport mode dot1q-tunnel**
8. **switchport access vlan** *vlan-id*
9. **exit**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**vlan** *vlan-id* 例： `switch(config)# vlan 10`	VLAN を指定します。

	コマンドまたはアクション	目的
ステップ 3	**vn-segment** *vnid* 例： `switch(config-vlan)# vn-segment 10010`	VXLAN VNID（仮想ネットワーク ID）を指定します。
ステップ 4	**xconnect** 例： `switch(config-vlan)# xconnect`	VNI が接続されたプロバイダー VLAN を相互接続モードに定義します。
ステップ 5	**exit** 例： `switch(config-vlan)# exit`	コマンド モードを終了します。
ステップ 6	**interface** *type port* 例： `switch(config)# interface ethernet 1/1`	インターフェイス設定モードを開始します。
ステップ 7	**switchport mode dot1q-tunnel** 例： `switch(config-if)# switchport mode dot1q-tunnel`	ポートに 802.1q トンネルを作成します。インターフェイス モードを変更すると、ポートはダウンし、再初期化（ポート フラップ）されます。トンネル インターフェイスでは BPDU フィルタリングがイネーブルになり、CDP がディセーブルになります。
ステップ 8	**switchport access vlan** *vlan-id* 例： `switch(config-if)# switchport access vlan 10`	インターフェイスのアクセス VLAN を設定します。
ステップ 9	**exit** 例： `switch(config-vlan)# exit`	コマンド モードを終了します。

例

この例は、VXLAN クロスコネクトの設定方法を示します。

```
switch# configure terminal
switch(config)# vlan 10
switch(config)# vn-segment 10010
switch(config)# xconnect
switch(config)# vlan 20
switch(config)# vn-segment 10020
switch(config)# xconnect
switch(config)# vlan 30
switch(config)# vn-segment 10030
switch(config)# xconnect
```

次の例では、アクセス ポートを設定する方法を示します。

```
switch# configure terminal
switch(config)# interface ethernet1/1
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 10
switch(config-if)# exit
switch(config)# interface ethernet1/2
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 20
switch(config-if)# exit
switch(config)# interface ethernet1/3
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 30
```

VXLAN クロス コネクト設定の確認

VXLANクロスコネクト設定のステータスを表示するには、次のコマンドの1つを入力します。

表 *13 : VXLAN* クロスコネクト情報の表示

コマンド	目的
show running-config vlan *session-num*	VLAN 情報を表示します。
show nve vni	VXLAN VNI ステータスを表示します。
show nve vni *session-num*	VNI ごとの VXLAN VNI ステータスを表示します。

show run vlan 503 コマンドの例：

```
switch(config)# sh run vlan 503

!Command: show running-config vlan 503
!Running configuration last done at: Mon Jul  9 13:46:03 2018
!Time: Tue Jul 10 14:12:04 2018

version 9.2(1) Bios:version 07.64
vlan 503
vlan 503
  vn-segment 5503
  xconnect
```

show nve vni 5503 コマンドの例：

```
switch(config)# sh nve vni 5503
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP
       SU - Suppress Unknown Unicast
Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      5503     225.5.0.3         Up    CP   L2 [503]           SA       Xconn
```

show nve vni コマンドの例：

```
switch(config)# sh nve vni
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP
       SU - Suppress Unknown Unicast
Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      5501     225.5.0.1         Up    CP   L2 [501]           SA
nve1      5502     225.5.0.2         Up    CP   L2 [502]           SA
nve1      5503     225.5.0.3         Up    CP   L2 [503]           SA       Xconn
nve1      5504     UnicastBGP        Up    CP   L2 [504]           SA       Xconn
nve1      5505     225.5.0.5         Up    CP   L2 [505]           SA       Xconn
nve1      5506     UnicastBGP        Up    CP   L2 [506]           SA       Xconn
nve1      5507     225.5.0.7         Up    CP   L2 [507]           SA       Xconn
nve1      5510     225.5.0.10        Up    CP   L2 [510]           SA       Xconn
nve1      5511     225.5.0.11        Up    CP   L2 [511]           SA       Xconn
nve1      5512     225.5.0.12        Up    CP   L2 [512]           SA       Xconn
nve1      5513     UnicastBGP        Up    CP   L2 [513]           SA       Xconn
nve1      5514     225.5.0.14        Up    CP   L2 [514]           SA       Xconn
nve1      5515     UnicastBGP        Up    CP   L2 [515]           SA       Xconn
nve1      5516     UnicastBGP        Up    CP   L2 [516]           SA       Xconn
nve1      5517     UnicastBGP        Up    CP   L2 [517]           SA       Xconn
nve1      5518     UnicastBGP        Up    CP   L2 [518]           SA       Xconn
```

VXLAN クロス コネクト用の NGAM の設定

この手順では、VXLAN Cross Connect 用に NGOAM を設定する方法について説明します。

手順の概要

1. **configure terminal**
2. **feature ngoam**
3. **ngoam install acl**
4. （任意） **ngoam xconnect hb-interval** *interval*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**feature ngoam** 例： `switch(config)# feature ngoam`	NGOAM 機能を開始します。
ステップ 3	**ngoam install acl** 例： `switch(config)# ngoam install acl`	NGOAM アクセス コントロール リスト（ACL）をインストールします。

	コマンドまたはアクション	目的
ステップ 4	（任意） **ngoam xconnect hb-interval** *interval* 例： `switch(config)# ngoam xconnect hb-interval 5000`	ハート ビート間隔を設定します。*interval* の範囲は 150ー5000 です。デフォルト値は 190 です。

VXLAN クロス コネクトの NGAM の確認

VXLAN クロス コネクト設定の NGOAM ステータスを表示するには、次のコマンドの 1 つを入力します。

表 *14 : VXLAN* クロスコネクト情報の表示

コマンド	目的
show ngoam xconnect session all	xconnect セッションの要約を表示します。
show ngoam xconnect session *session-num*	セッションの詳細な xconnect 情報を表示します。

show ngoam xconnect session all コマンドの例：

```
switch(config)# sh ngoam xconnect session all

States: LD = Local interface down, RD = Remote interface Down
          HB = Heartbeat lost, DB = Database/Routes not present
          * - Showing Vpc-peer interface info
Vlan          Peer-ip/vni      XC-State       Local-if/State    Rmt-if/State
================================================================================
507       6.6.6.6 /  5507      Active        Eth1/7 / UP        Eth1/5 / UP
508       7.7.7.7 /  5508      Active        Eth1/8 / UP        Eth1/5 / UP
509       7.7.7.7 /  5509      Active        Eth1/9 / UP        Eth1/9 / UP
510       6.6.6.6 /  5510      Active         Po303 / UP         Po103 / UP
513       6.6.6.6 /  5513      Active        Eth1/6 / UP        Eth1/8 / UP
```

show ngoam xconnect session 507 コマンドの例：

```
switch(config)# sh ngoam xconnect session 507
Vlan ID: 507
Peer IP: 6.6.6.6  VNI : 5507
State: Active
Last state update: 07/09/2018 13:47:03.849
Local interface: Eth1/7  State:  UP
Local vpc interface Unknown  State:  DOWN
Remote interface: Eth1/5  State:  UP
Remote vpc interface: Unknown  State:  DOWN
switch(config)#
```

NGOAM 認証

NGOAMは、パストレース応答でインターフェイス統計情報を提供します。NGOAMは、HMAC MD5 認証メカニズムを使用してパストレース要求を認証し、統計情報を提供します。

NGOAM 認証は、インターフェイスの統計情報を提供する前にパストレース要求を検証します。NGOAM 認証は、**req-stats** オプションを使用したパストレース要求に対してのみ有効です。他のすべてのコマンドは、認証設定の影響を受けません。要求元ノードで NGOAM 認証キーが設定されている場合は、このキーを使用して MD5 アルゴリズムを実行し、16 ビットの MD5 ダイジェストを生成します。このダイジェストは、パストレース要求メッセージで type-length-value（TLV）としてエンコードされます。

パストレース要求を受信すると、NGOAM は **req-stats** オプションとローカルの NGOAM 認証キーをチェックします。ローカル NGOAM 認証キーが存在する場合、要求のローカル キーを使用して MD5 を実行し、MD5 ダイジェストを生成します。両方のダイジェストが一致すると、インターフェイス統計情報が含まれます。両方のダイジェストが一致しない場合は、インターフェイス名のみが送信されます。MD5 ダイジェストを含む NGOAM 要求にローカル認証キーが設定されていない場合、そのダイジェストは無視され、すべてのインターフェイス統計情報が送信されます。ネットワーク全体を保護するには、すべてのノードで認証キーを設定します。

NGOAM 認証キーを設定するには、 **ngoam authentication-key** <*key*> CLI コマンドを使用します。**show running-config ngoam** CLI コマンドを使用して、認証キーを表示します。

```
switch# show running-config ngoam
!Time: Tue Mar 28 18:21:50 2017
version 7.0(3)I6(1)
feature ngoam
ngoam profile 1
  oam-channel 2
ngoam profile 3
ngoam install acl
ngoam authentication-key 987601ABCDEF
```

次の例では、同じ認証キーが要求側スイッチと応答側スイッチで設定されます。

```
switch# pathtrace nve ip 12.0.22.1 profile 1 vni 31000 req-stats ver
Path trace Request to peer ip 12.0.22.1 source ip 11.0.22.1
Hop   Code   ReplyIP   IngressI/f  EgressI/f   State
=====================================================
  1 !Reply from 55.55.55.2, Eth5/7/1  Eth5/7/2  UP / UP
  Input Stats: PktRate:0 ByteRate:0 Load:0 Bytes:339573434 unicast:14657 mcast:307581
bcast:67 discards:0 errors:3 unknown:0 bandwidth:42949672970000000
Output Stats: PktRate:0 ByteRate:0 load:0 bytes:237399176 unicast:2929 mcast:535710
bcast:10408 discards:0 errors:0 bandwidth:42949672970000000
  2 !Reply from 12.0.22.1, Eth1/7  Unknown  UP / DOWN
  Input Stats: PktRate:0 ByteRate:0 Load:0 Bytes:4213416 unicast:275 mcast:4366 bcast:3
 discards:0 errors:0 unknown:0 bandwidth:42949672970000000
switch# conf t
switch(config)# no ngoam authentication-key 123456789
switch(config)# end
```

次の例では、認証キーが要求元スイッチで設定されていません。したがって、応答するスイッチはインターフェイス統計情報を送信しません。中間ノードには認証キーが設定されておらず、常にインターフェイス統計情報で応答します。

```
switch# pathtrace nve ip 12.0.22.1 profile 1 vni 31000 req-stats ver
Path trace Request to peer ip 12.0.22.1 source ip 11.0.22.1
Sender handle: 10
Hop   Code   ReplyIP   IngressI/f  EgressI/f   State
=================================================
  1 !Reply from 55.55.55.2, Eth5/7/1  Eth5/7/2  UP / UP
  Input Stats: PktRate:0 ByteRate:0 Load:0 Bytes:339580108 unicast:14658 mcast:307587
bcast:67 discards:0 errors:3 unknown:0 bandwidth:42949672970000000
Output Stats: PktRate:0 ByteRate:0 load:0 bytes:237405790 unicast:2929 mcast:535716
bcast:10408 discards:0 errors:0 bandwidth:42949672970000000
  2 !Reply from 12.0.22.1, Eth1/17  Unknown  UP / DOWN
```

Q-in-VNI の注意事項と制約事項

Q-in-VNI には、次の注意事項と制約事項があります。

- Q-in-VNI および選択的 Q-in-VNI は、VXLAN フラッドアンドラーニング（入力複製あり）および VXLAN EVPN（入力複製あり）でサポートされます。
- Q-in-VNI、選択的 Q-in-VNI、および QinQ-QinVNI は、Cisco Nexus 9000-EX プラットフォーム スイッチのマルチキャスト アンダーレイではサポートされません。
- vPC VTEP でこの機能を実行する場合は、**system dot1q-tunnel transit** [**vlan** *vlan-range*] コマンドが必要です。
- ポート VLAN マッピングと Q-in-VNI は同じポートに共存できません。
- **system dot1q-tunnel transit** コマンドが有効になっている場合、ポート VLAN マッピングと Q-in-VNI はスイッチ上で共存できません。Cisco NX-OS リリース9.3(5) 以降では、ポート VLANマッピングとQ-in-VNIは、同じスイッチ上で、**system dot1q-tunnel transit vlan** *vlan-range* コマンドを使用して設定された異なるポートおよび異なるプロバイダーVLAN 上で共存できます。
- Cisco NX-OS リリース 10.1(1) 以降、同じポートでの選択的 Q-in-VNI および VXLAN VLAN 機能は、Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- vPC VTEPでのL3アップリンク障害時の適切な動作のために、バックアップSVIを設定し、**system nve infra-vlans** *backup-svi-vlan* コマンドを入力します。Cisco Nexus 9000-EXプラットフォームスイッチでは、バックアップSVI VLANがピアリンクのネイティブVLANである必要があります。
- Q-in-VNI は VXLAN でのブリッジングをサポートします。VXLAN ルーティングはサポートされません。
- dot1q トンネル モードは Cisco Nexus 9300 シリーズおよび Cisco Nexus 9500 プラットフォーム スイッチの ALE ポートでサポートしません。
- Q-in-VNI は FEX をサポートしません。

- ネットワーク フォワーディング エンジン（NFE）またはリーフ スパイン エンジン（LSE）を使用して Cisco Nexus 9000 シリーズ スイッチのアクセス ポートとトランク ポートを設定する場合、同じスイッチ上の異なるインターフェイスにアクセス ポート、トランク ポート、および dot1q ポートを設定できます。
- 同じ VLAN に dot1q とトランク ポート/アクセス ポートの両方を設定することはできません。
- プロバイダー VNI で、カスタマー VLAN から発信された ARP トラフィックの ARP 抑制を無効にします。

```
switch(config)# interface nve 1
switch(config-if-nve)# member VNI 10000011
switch(config-if-nve-vni)# no suppress-arp
```

- Cisco Nexus 9300 プラットフォーム スイッチは単一タグをサポートします。これを有効にするには、NVEインターフェイスに対して **no overlay-encapsulation vxlan-with-tag** コマンドを入力します。

```
switch(config)# interface nve 1
switch(config-if-nve)# no overlay-encapsulation vxlan-with-tag
switch# show run int nve 1

!Command: show running-config interface nve1
!Time: Wed Jul 20 23:26:25 2016

version 7.0(3u)I4(2u)

interface nve1
  no shutdown
  source-interface loopback0
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2000980
    mcast-group 225.4.0.1
```

- Cisco Nexus 9500 プラットフォーム スイッチは単一タグをサポートしていません。ダブルタグのみをサポートします。
- Cisco Nexus 9300 プラットフォーム スイッチは単一タグをサポートしていません。単一のタグのみをサポートします。
- Cisco Nexus 9300-EX プラットフォーム スイッチは、Q-in-VNI 用に設定されたポートとトランク用に設定されたポート間のトラフィックをサポートしません。
- Q-in-VNI は、レイヤ 3 サブインターフェイスが設定されている VTEP と共存できません。Cisco NX-OS リリース9.3(5) 以降、この制限は Cisco Nexus 9332C、9364C、9300-FX/FX2、および 9300-GX プラットフォーム スイッチには適用されません。
- Cisco NX-OS リリース 10.2(3)F 以降、Cisco Nexus 9300-FX3/GX2 プラットフォーム スイッチは、レイヤ 3 サブインターフェイスが構成されている VTEP と共存する Q-in-VNI をサポートします。
- Cisco NX-OS リリース 10.4(1)F 以降、Cisco Nexus 9332D-H2R スイッチは、レイヤ 3 サブインターフェイスが構成されている VTEP と共存する Q-in-VNI をサポートします。

- Cisco NX-OS リリース 10.4(2)F 以降、Cisco Nexus 93400LD-H1 スイッチは、レイヤ 3 サブインターフェイスが構成されている VTEP と共存する Q-in-VNI をサポートします。
- Cisco NX-OS リリース 10.4(3)F 以降、Cisco Nexus 9364C-H1 スイッチは、レイヤ 3 サブインターフェイスが構成されている VTEP と共存する Q-in-VNI をサポートします。
- VLAN1 が複数のプロバイダー タグを使用して選択的 Q-in-VNI を使用してネイティブ VLAN として設定されている場合、ネイティブ VLAN 上のトラフィックはドロップされます。ポートが選択的 Q-in-VNI で設定されている場合は、VLAN1 をネイティブ VLAN として設定しないでください。VLAN1 がカスタマー VLAN として設定されている場合、VLAN1 のトラフィックはドロップされます。
- 基本ポート モードでは、dot1q トンネル ポートにアクセス VLAN が設定されている必要があります。
- ポートのアクセス VLAN には VNI マッピングが必要です。
- ある Cisco Nexus 9300-EX シリーズ スイッチ VTEP に Q-in-VNI があり、別の Cisco Nexus 9300-EX シリーズ スイッチ VTEP にトランクがある場合、双方向トラフィックは 2 つのポート間で送信されません。
- プロバイダー インターフェイスと VXLAN アップリンクが混在する VXLAN および Q-in-Q を実行する Cisco Nexus 9300-EX シリーズのスイッチは考慮されません。VXLAN アップリンクは、Q-in-Q プロバイダーまたはカスタマー インターフェイスから分離する必要があります。

  vPC の使用例では、VXLAN と Q-in-Q が同じスイッチで使用される場合、次の考慮事項を考慮する必要があります。

  - オーファン ポート間通信を確保するには、vPC ピアリンクをプロバイダーインターフェイスとして明確に設定する必要があります。このような場合、トラフィックは 2 つの IEEE 802.1q タグ（ダブル dot1q タギング）で送信されます。内側の dot1q はカスタマー VLAN ID で、外側の dot1q はプロバイダー VLAN ID（アクセス VLAN）です。
  - vPC ピアリンクは、アップリンクに障害が発生した場合に VXLAN カプセル化トラフィックのバックアップ パスとして使用されます。Q-in-Q では、vPC ピアリンクはプロバイダー インターフェイス（オーファン ポート間通信）としても機能します。この組み合わせでは、トラフィックのバックアップ VLAN としてネイティブ VLAN を使用して、アップリンク障害シナリオを処理します。また、バックアップ VLAN がシステム インフラ VLAN（system nve infra-vlans）として設定されていることを確認します。

- Cisco NX-OS リリース 9.3(5) 以降、Q-in-VNI は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、Q-in-VNI は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、Q-in-VNI は Cisco Nexus 9332D-H2R スイッチでサポートされています。

- Cisco NX-OS リリース 10.4(2)F 以降、Q-in-VNI は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、Q-in-VNI は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 9.3(5) 以降、Q-in-VNI は vPC ファブリック ピアリングをサポートします。
- STP BPDU のトンネリングはサポートされていないため、選択的 Q-in-VNI には BPDU フィルタが必要です。
- Cisco NX-OS リリース 10.3(3)F 以降、IPv6アンダーレイは、Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 スイッチの VXLAN EVPN のQ-in-VNI、選択的 Q-in-VNI、Q-in-Q-in-VNI でサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、IPv6 アンダーレイは、Cisco Nexus 9332D-H2R スイッチの VXLAN EVPN のQ-in-VNI、選択的 Q-in-VNI、Q-in-Q-in-VNI でサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、IPv6 アンダーレイは、Cisco Nexus 93400LD-H1 スイッチの VXLAN EVPN のQ-in-VNI、選択的 Q-in-VNI、Q-in-Q-in-VNI でサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、IPv6 アンダーレイは、Cisco Nexus 9364C-H1 スイッチの VXLAN EVPN のQ-in-VNI、選択的 Q-in-VNI、Q-in-Q-in-VNI でサポートされています。

Q-in-VNI の設定

Q-in-VNI を使用することで、マッピングによる特定ポートへのトラフィックの分離が行えます。マルチテナント環境では、テナントにポートを指定でき、VXLAN オーバーレイでのパケットの送受信ができます。

手順の概要

1. **configure terminal**
2. **interface** *type port*
3. **switchport mode dot1q-tunnel**
4. **switchport access vlan** *vlan-id*
5. **spanning-tree bpdufilter enable**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	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	**interface** *type port*	インターフェイス設定モードを開始します。
ステップ 3	**switchport mode dot1q-tunnel**	ポートに 802.1Q トンネルを作成します。
ステップ 4	**switchport access vlan** *vlan-id*	VLAN に割り当てられたポートを指定します。
ステップ 5	**spanning-tree bpdufilter enable**	指定したスパニングツリー エッジ インターフェイスの BPDU フィルタリングをイネーブルにします。デフォルトでは、BPDU フィルタリングはディセーブルです。

例

次に示すのは、Q-in-VNI の設定例です。

```
switch# config terminal
switch(config)# interface ethernet 1/4
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 10
switch(config-if)# spanning-tree bpdufilter enable
switch(config-if)#
```

選択的 Q-in-VNI の設定

選択的 Q-in-VNI は、ポート上のユーザ固有の範囲のカスタマー VLAN を 1 つの特定のプロバイダー VLAN に関連付けることができる VXLAN トンネリング機能です。ポートに設定されたカスタマー VLAN のいずれかに一致する VLAN タグが付いたパケットは、サービス プロバイダー VNI のプロパティを使用して VXLAN ファブリック全体でトンネリングされます。VXLAN カプセル化パケットは、内部パケットの L2 ヘッダーの一部としてカスタマー VLAN タグを伝送します。

選択的 Q-in-VNI 設定ポートの設定済みカスタマー VLAN の範囲内に存在しない VLAN タグが付いたパケットはドロップされます。これには、ポート上のネイティブ VLAN に一致する VLAN タグが付いたパケットが含まれます。タグなしまたはネイティブ VLAN タグ付きのパケットは、選択的 Q-in-VNI ポート（VXLAN なし）で設定されたネイティブ VLAN の SVI を使用して L3 ルーティングされます。

選択的 Q-in-VNI については、次のガイドラインを参照してください。

- 選択的 Q-in-VNI は、Cisco Nexus 9300-EX および 9300-FX/FXP/FX2/FX3 および 9300-GX プラットフォーム スイッチの vPC ポートと非 vPC ポートの両方でサポートされます。この機能は、Cisco Nexus 9200 および 9300 プラットフォーム スイッチではサポートされていません。
- Cisco NX-OS リリース 9.3(5) 以降、選択的 Q-in-VNI は vPC ファブリック ピアリングをサポートします。
- 1 つの VTEP での選択的 Q-in-VNI の設定と、VXLAN ピアでのプレーン Q-in-VNI の設定がサポートされています。同じスイッチ上で、1 つのポートを選択的 Q-in-VNI で、もう 1 つのポートをプレーン Q-in-VNI で設定できます。
- 選択的 Q-in-VNI は、入力 VLAN タグ ポリシング機能です。選択的 Q-in-VNI 設定範囲に関しては、入力 VLAN タグ ポリシングのみが実行されます。

  たとえば、選択的 Q-in-VNI カスタマー VLAN 範囲 100ー200 は VTEP 1 で設定され、カスタマー VLAN 範囲 200ー300 は VTEP 2 で設定されます。VLAN タグが 175 のトラフィックが VTEP 1 から VTEP 2 に送信されると、VLAN は設定された範囲内にあり、VTEP2 に転送されるため、トラフィックは VTEP1 で受け入れられます。VTEP2 では、VLAN タグ 175 が設定された範囲に含まれていなくても、パケットは選択的 Q-in-VNI ポートから出力されます。パケットが VTEP1 から VLAN タグ 300 で送信される場合、300 は VTEP1 の選択的 Q-in-VNI 設定範囲にないため、パケットはドロップされます。
- Cisco NX-OS リリース 10.1(1) 以降、VTEP での選択的 Q-in-VNI およびアドバタイズ PIP 機能は、Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 9.3(5) 以降では、VTEP の選択的 Q-in-VNI で **advertise-pip** コマンドがサポートされています。
- ポート VLAN マッピングと選択的 Q-in-VNI を同じポートに共存させることはできません。
- **system dot1q-tunnel transit** コマンドが有効になっている場合、ポート VLAN マッピングと選択的 Q-in-VNI はスイッチ上で共存できません。Cisco NX-OS リリース 9.3（5）以降では、ポート VLAN マッピングと Q-in-VNI は、同じスイッチ上で、vlan-range コマンドを使用して設定された異なるポートおよび異なるプロバイダー VLAN 上で共存できます。 **system dot1q-tunnel transit vlan**
- 選択的な Q-in-VNI 設定で vPC スイッチに **system dot1q-tunnel transit** [**vlan** *vlan-id*] コマンドを設定します。このコマンドは、vPC ピアの 1 つに孤立ポートがある場合に、パケットが vPC ピア リンクを通過するときに内部 Q タグを保持するために必要です。この CLI 設定では、**vlan dot1Q tag native** 機能は動作しません。Cisco NX-OS リリース 9.3(5) 以前では、スイッチで作成されたすべての VLAN はプロバイダー VLAN であり、他の目的には使用できません。

  Cisco NX-OS リリース9.3(5) 以降では、選択的 Q-in-VNI および VXLAN VLAN を同じポートでサポートできます。[**vlan** *vlan-range*] オプションを使用すると、プロバイダー VLAN を指定し、他の VLAN を通常の VXLAN トラフィックに使用できます。次の例では、VXLAN VLAN は 50、プロバイダー VLAN は 501、カスタマー VLAN は 31ー40、ネイティブ VLAN は 2400 です。

```
system dot1q-tunnel transit vlan 501
interface Ethernet1/1/2
  switchport
  switchport mode trunk
  switchport trunk native vlan 2400
  switchport vlan mapping 31-40 dot1q-tunnel 501
  switchport trunk allowed vlan 50,501,2400
  spanning-tree port type edge trunk
  mtu 9216
  no shutdown
```

- 選択的 Q-in-VNI ポートに設定されたネイティブ VLAN は、カスタマー VLAN 範囲の一部にはできません。ネイティブ VLAN がカスタマー VLAN 範囲の一部である場合、設定は拒否されます。

  プロバイダー VLAN は、カスタマー VLAN 範囲とオーバーラップできます。たとえば、**switchport vlan mapping 100-1000 dot1q-tunnel 200** のようになります。

- デフォルトでは、ネイティブ VLAN は VLAN 1 です。VLAN 1 が **switchport vlan mapping** <*range*>**dot1q-tunnel** <*sp-vlan*> CLI を使用してカスタマーVLAN範囲の一部として設定されている場合、VLAN 1 がポートのネイティブ VLAN であるときに、カスタマー VLAN 1 のトラフィックが伝送されません。顧客が VLAN 1 トラフィックを VXLAN クラウド上で伝送する場合は、顧客のVLAN範囲外の値を持つポートにダミーのネイティブVLANを設定する必要があります。

- 選択的Q-in-VNIポートで設定されたスイッチポートVLANマッピング範囲から一部のVLANまたはVLANの範囲を削除するには、**no** 形式 **switchport vlan mapping** <*range*>**dot1q-tunnel** <*sp-vlan*> のコマンド範囲を指定します。

  たとえば、VLAN 100～1000 がポートに設定されているとします。設定された範囲から VLAN 200ー300を削除するには、**no switchport vlan mapping** <*200-300*> **dot1q-tunnel** <*sp-vlan*> コマンドを使用します。

```
interface Ethernet1/32
  switchport
  switchport mode trunk
  switchport trunk native vlan 4049
  switchport vlan mapping 100-1000 dot1q-tunnel 21
  switchport trunk allowed vlan 21,4049
  spanning-tree bpdufilter enable
  no shutdown

switch(config-if)# no sw vlan mapp 200-300 dot1q-tunnel 21
switch(config-if)# sh run int e 1/32

version 7.0(3)I5(2)

interface Ethernet1/32
  switchport
  switchport mode trunk
  switchport trunk native vlan 4049
  switchport vlan mapping 100-199,301-1000 dot1q-tunnel 21
  switchport trunk allowed vlan 21,4049
  spanning-tree bpdufilter enable
  no shutdown
```

次の設定例を参照してください。

- プロバイダー VLAN の設定については、次の例を参照してください。

```
vlan 50
 vn-segment 10050
```

- VXLAN フラッドと学習と入力レプリケーションの設定については、次の例を参照してください。

```
member vni 10050
    ingress-replication protocol static
      peer-ip 100.1.1.3
      peer-ip 100.1.1.5
      peer-ip 100.1.1.10
```

- インターフェイス nve の設定については、次の例を参照してください。

```
interface nve1
 no shutdown
 source-interface loopback0  member vni 10050
mcast-group 230.1.1.1
```

- ネイティブ VLAN で SVI をルーティング トラフィックに設定するには、次の例を参照してください。

```
vlan 150
interface vlan150
 no shutdown
 ip address 150.1.150.6/24
 ip pim sparse-mode
```

- ポートでの選択的Q-in-VNIの設定については、次の例を参照してください。この例では、ネイティブ VLAN 150 がタグなしパケットのルーティングに使用されます。カスタマー VLAN 200ー700 は dot1q トンネルを介して伝送されます。ネイティブ VLAN 150 とプロバイダー VLAN 50 のみが許可されます。

```
switch# config terminal
switch(config)#interface Ethernet 1/31
switch(config-if)#switchport
switch(config-if)#switchport mode trunk
switch(config-if)#switchport trunk native vlan 150
switch(config-if)#switchport vlan mapping 200-700 dot1q-tunnel 50
switch(config-if)#switchport trunk allowed vlan 50,150
switch(config-if)#no shutdown
```

- プロバイダー VNI で、カスタマー VLAN から発信された ARP トラフィックの ARP 抑制を無効にします。

```
switch(config)# interface nve 1
switch(config-if-nve)# member VNI 10000011
 switch(config-if-nve-vni)# no suppress-arp
```

レイヤ 2 プロトコル トンネリングを使用した Q-in-VNI 構成

L2PT を使用した Q-in-VNI の概要

レイヤ 2 プロトコル トンネリング（L2PT）を使用した Q-in-VNI は、マルチタグ トラフィックの VXLAN EVPN ファブリック全体で制御パケットとデータ パケットを転送するために使用されます。

VLAN レベルで L2PT を使用した Q-in-VNI を有効にするには、L2 プロトコル パケットを含むすべてのパケットをトンネリングするために VLAN をマークする **l2protocol tunnel vxlan vlan** <*vlan-range*> コマンドを使用します。この **switchport trunk allow-multi-tag** コマンドは、VXLAN ファブリックが複数のタグを持つパケットをトンネリングするためにも必要です。

L2PT を使用した Q-in-VNI 構成の詳細については、L2PT を使用した Q-in-VNI の構成 （665 ページ） を参照してください。

L2PT を搭載した Q-in-VNI の注意事項と制約事項

L2PT を搭載した Q-in-VNI には、次の注意事項と制約事項があります。

- Cisco NX-OS リリース 10.3(2)F 以降で、L2PT を搭載した Q-in-VNI は Cisco Nexus 9300-FX/FX2/FX3/GX/GX2 ToR スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降、L2PT を持つ Q-in-VNI は Cisco Nexus 9332D-H2R ToR スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、L2PT を持つ Q-in-VNI は Cisco Nexus 93400LD-H1 スイッチでサポートされています。 c
- Cisco NX-OS リリース 10.4(3)F 以降、L2PT を持つ Q-in-VNI は Cisco Nexus 9364C-H1 スイッチでサポートされています。 c
- コマンドがインターフェイスで実行されると、コマンド内のすべての VLAN がトンネリング VLAN になり、他のポートで他の目的に使用することはできません。**l2protocol tunnel vxlan**
- トンネル VLAN のメンバーになれるのは、ネットワーク内の 2 つのインターフェイスだけです。vPC の場合、vPC スイッチと MCT の両方の vPC ポートもトンネル VLAN の一部になります。
- 同じ VLAN を複数のインターフェースでトンネリングしてはなりません。
- **l2protocol tunnel vxlan** コマンドは、トランク ポートでのみ許可されます。また、vxlan ファブリック全体で複数のタグを保持するには、「マルチタグ」構成も必要です。

- クロス コネクト機能と **l2protocol tunnel vxlan** コマンドは、スイッチ上で同時に使用できません。
- 「STP」などの既存の L2PT コマンド オプションは、**l2protocol tunnel vxlan**コマンドと一緒に 使用できません。
- Cisco NX-OS リリース 10.3(3)F 以降では、L2PT を搭載した Q-in-VNI の Ethertype サポートは、Cisco Nexus 9300-FX2/FX3/GX/GX2 ToR スイッチで提供されます。
- Cisco NX-OS リリース 10.4(1)F 以降、L2PT を持つ Q-in-VNI の Ethertype サポートは Cisco Nexus 9332D-H2R ToR スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、L2PT を持つ Q-in-VNI の Ethertype サポートは Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、L2PT を持つ Q-in-VNI の Ethertype サポートは Cisco Nexus 9364C-H1 スイッチでサポートされています。

L2PT を使用した Q-in-VNI の構成

次の手順に従って、VXLAN VLAN で L2PT を使用した Q-in-VNI 構成します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. **switchport**
4. **switchport mode trunk**
5. **switchport dot1q ethertype** *ethertype-value*
6. **switchport trunk allow-multi-tag**
7. **switchport trunk allowed vlan** *vlan-list*
8. **l2protocol tunnel vxlan vlan** <*vlan-range*>

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**interface ethernet** *slot/port* 例： `switch(config)# interface ethernet1/1`	設定するインターフェイスを指定します。

	コマンドまたはアクション	目的
ステップ 3	**switchport** 例： `switch(config-inf)# switchport`	ポートをレイヤ 2 ポートとして設定します。
ステップ 4	**switchport mode trunk** 例： `switch(config-inf)# switchport mode trunk`	インターフェイスをレイヤ2 トランク ポートとして設定します。
ステップ 5	**switchport dot1q ethertype** *ethertype-value* 例： `switch(config-inf)# switchport dot1q ethertype 0x88a8`	ポートの Ethertype を設定します。
ステップ 6	**switchport trunk allow-multi-tag** 例： `switch(config-inf)# switchport trunk allow-multi-tag`	許可された VLAN をネイティブ VLAN を除くプロバイダー VLAN として設定します。次に挙げる構成例では、VLAN 1201 と 1202 はプロバイダー VLAN であり、複数の内部 Q タグを伝送できます。
ステップ 7	**switchport trunk allowed vlan** *vlan-list* 例： `switch(config-inf)# switchport trunk allowed vlan 1201-1202`	トランク インターフェイスの許可 VLAN を設定します。
ステップ 8	**l2protocol tunnel vxlan vlan** <*vlan-range*> 例： `switch(config-inf)# l2protocol tunnel vxlan vlan 1201-1202`	コマンドのすべての VLAN をトンネリング VLAN として設定します。これらの VLAN は、他のポートで他の目的に使用することはできません。

L2PT を使用した Q-in-VNI の構成の確認

L2PT を使用した Q-in-VNI 構成のステータスを表示するには、次のコマンドを入力します。

コマンド	目的
show run interface ethernet *slot/port*	L2PT VXLAN VLAN インターフェイス情報を表示します。
show run l2pt	L2PT VXLAN VLAN 構成情報を表示します。
show l2protocol tunnel interface ethernet *slot/port*	L2PT インターフェイス情報を表示します。
show vpc consistency-parameters interface *slot/port*	L2PT VXLAN VLAN を含むすべての vPC インターフェイス全体で一貫している必要があるパラメータのステータスを表示します。

次の例は、**show run interface ethernet** *slot/port* コマンドのサンプル出力を示しています。

```
switch(config-if)# sh run int e1/1
 interface Ethernet1/1
  switchport
  switchport mode trunk
  switchport trunk allow-multi-tag
  switchport trunk allowed vlan 1201-1202
  l2protocol tunnel vxlan vlan 1201-1202
  no shutdown
```

次の例は、**show run l2pt** コマンドのサンプル出力を示しています。

```
switch# sh run l2pt
interface Ethernet1/1
  switchport mode trunk
  l2protocol tunnel vxlan vlan 1201-1202
  no shutdown
```

次の例は、**show l2protocol tunnel interface ethernet** *slot/port* コマンドのサンプル出力を示しています。

```
switch# show l2protocol tunnel interface e1/1
COS for Encapsulated Packets: 5
Interface: Eth1/1 Vxlan Vlan 1201-1202
```

次の例は、**show vpc consistency-parameters interface** *slot/port* コマンドのサンプル出力を示しています。

```
switch# sh run int po101

interface port-channel101
  switchport
  switchport mode trunk
  switchport trunk native vlan 80
  switchport trunk allow-multi-tag
  switchport trunk allowed vlan 80,1201-1203,1301
  spanning-tree port type edge trunk
  vpc 101
  l2protocol tunnel vxlan vlan 1201-1203,1301

switch# sh vpc consistency-parameters interface po101

    Legend:
        Type 1 : vPC will be suspended in case of mismatch

Name                        Type  Local Value            Peer Value
-------------               ----  ---------------------- -----------------------
delayed-lacp                1     disabled               disabled
lacp suspend disable        1     enabled                enabled
mode                        1     active                 active
Switchport Isolated         1     0                      0
Interface type              1     port-channel           port-channel
LACP Mode                   1     on                     on
Virtual-ethernet-bridge     1     Disabled               Disabled
Speed                       1     25 Gb/s                25 Gb/s
Duplex                      1     full                   full
MTU                         1     1500                   1500
Port Mode                   1     trunk                  trunk
Native Vlan                 1     80                     80
Admin port mode             1     trunk                  trunk
Port-type External          1     Disabled               Disabled
STP Port Guard              1     Default                Default
STP Port Type               1     Edge Trunk Port        Edge Trunk Port
```

```
STP MST Simulate PVST       1     Default                Default
lag-id                      1     [(7f9b,                [(7f9b,
                                  0-23-4-ee-be-4, 8065,  0-23-4-ee-be-4, 8065,
                                  0, 0), (8000,          0, 0), (8000,
                                  a8-9d-21-f8-4b-31, 64, a8-9d-21-f8-4b-31, 64,
                                   0, 0)]                 0, 0)]
Allow-Multi-Tag             1     Enabled                Enabled
Vlan xlt mapping            1     Disabled               Disabled
L2PT Vxlan Vlans            2     1201-1203,1301         1201-1203,1301
vPC card type               1     N9K TOR                N9K TOR
Allowed VLANs               -     80,1201-1203,1301      80,1201-1203,1301
Local suspended VLANs       -     -                      -
```

Q-in-VNI での LACP トンネリングの設定

Q-in-VNI は、LACP パケットのトンネルを設定できます。

手順の概要

1. **configure terminal**
2. **interface** *type port*
3. **switchport mode dot1q-tunnel**
4. **switchport access vlan** *vlan-id*
5. **interface nve** *x*

手順の詳細

手順

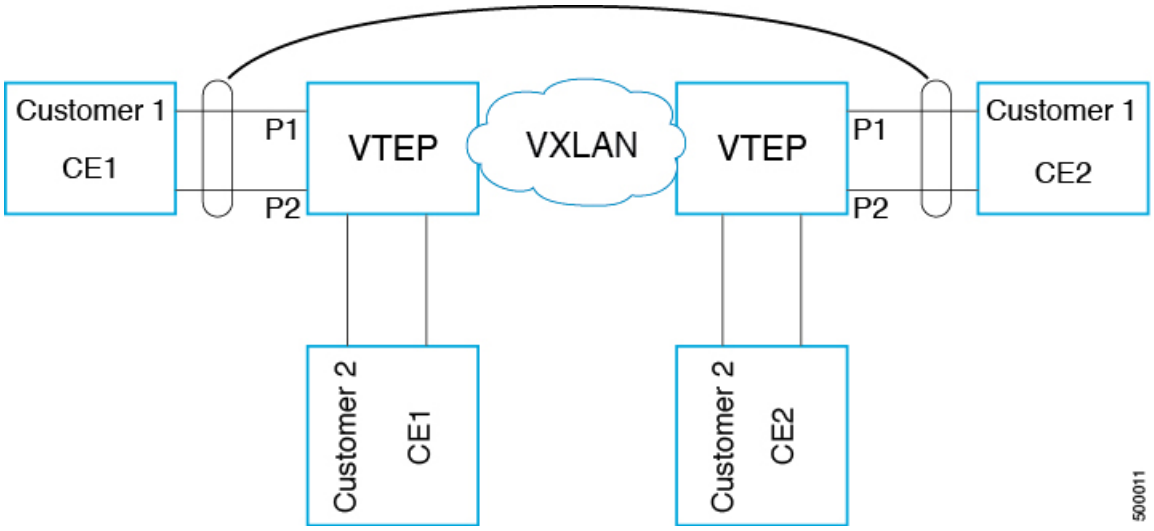
	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type port*	インターフェイス設定モードを開始します。
ステップ 3	**switchport mode dot1q-tunnel**	dot1q-tunnel モードをイネーブルにします。
ステップ 4	**switchport access vlan** *vlan-id*	VLAN に割り当てられたポートを指定します。
ステップ 5	**interface nve** *x*	VXLAN トンネルの終端となる VXLAN オーバーレイ インターフェイスを作成します。

例

- 次に示すのは、ポートチャネルペアの各ポートを一意のVMにピン止めするトポロジの例です。ポートチャネルがCEの視点から広げられています。VTEPにポー

トチャネルはありません。CE1 の P1 にあるトラフィックは Q-in-VNI を使用して CE2 の P1 に中継されます。

図 *69 : VXLAN P2P* トンネルを通じた *LACP* トンネリング



（注）

- Q-in-VNI は、LACP パケットのトンネルを設定できます（データセンターにまたがるポート チャネル接続を提供できます）。
  - データセンターにまたがる L1 接続とコロケーションの感覚を得られます。
  - 存在するのは 2 つのサイトです。CE1 の P1 からのトラフィックは、CE2 の P1 から送出されます。CE1 の P1 がダウンした場合は、LACP がこれをカバーして（経時的）、トラフィックを P2 にリダイレクトします。
- フラッディングおよび学習を行う VXLAN による静的入力複製を使用します。ポート チャネル上の各ポートに Q VNI が設定されます。ポート チャネルの各メンバーには複数の VNI があり、各ポートが特定の VNI にピン止めされます。
  - MAC の飽和状態を回避するには、VLAN の学習をオフ/ディセーブルにしてください。
- Q-in-VNI による LACP パケットのトンネル設定は、VXLAN EVPN ではサポートされません。
- サポートされるポート チャネルのメンバー数は、VTEP でサポートされるポートの数です。

複数プロバイダー VLAN を使用した選択的 Q-in-VNI

複数プロバイダー VLAN を使用した選択的 Q-in-VNI について

複数のプロバイダー VLAN を持つ選択的 Q-in-VNI は、VXLAN トンネリング機能です。この機能により、ポート上のユーザ固有の範囲のカスタマー VLAN を 1 つの特定のプロバイダー VLAN に関連付けることができます。また、ポート上で複数のカスタマー VLAN からプロバイダー VLAN へのマッピングを行うことができます。ポートに設定されたカスタマー VLAN のいずれかと一致する VLAN タグが付いたパケットは、サービス プロバイダー VNI のプロパティを使用して VXLAN ファブリック上でトンネリングされます。VXLAN カプセル化パケットは、内部パケットのレイヤ 2 ヘッダーの一部としてカスタマー VLAN タグを伝送します。

複数プロバイダー VLAN を使用した選択的 Q-in-VNI の注意事項と制約事項

複数プロバイダー VLAN を使用した選択的 Q-in-VNI には、次の注意事項と制約事項があります。

- 選択的 Q-in-VNI に関する既存の注意事項と制限事項がすべて適用されます。
- この機能は、VXLAN BGP EVPN IR モードでのみサポートされます。
- vPC ポート チャネルで複数のプロバイダー VLAN をイネーブルにする場合は、vPC ピア間で設定が一貫していることを確認してください。
- ポート VLAN マッピングと選択的 Q-in-VNI を同じポートに共存させることはできません。
- **system dot1q-tunnel transit** コマンドが有効になっている場合、ポート VLAN マッピングと選択的 Q-in-VNI はスイッチ上で共存できません。Cisco NX-OS リリース 9.3(5) 以降、ポート VLAN マッピングと選択的 Q-in-VNI は、同じスイッチ上に存在しますが、異なるポートと異なるプロバイダー VLAN 上に存在し、**system dot1q-tunnel transit vlan** *vlan-range* コマンド を使用して設定できます。
- **system dot1q-tunnel transit** [**vlan** *vlan-range*] コマンドは、vPC VTEP でこの機能を使用する場合に必要です。
- vPC VTEP でのレイヤ 3 アップリンク障害シナリオ中の適切な動作のために、バックアップ SVI を設定し、**system nve infra-vlans** *backup-svi-vlan* コマンドを入力します。Cisco Nexus 9000-EX プラットフォーム スイッチでは、バックアップ SVI VLAN がピアリンクのネイティブ VLAN である必要があります。
- ベストプラクティスとして、通常のトランクではプロバイダー VLAN を許可しないでください。
- カスタマー VLAN からプロバイダー VLAN へのマッピングが設定されているスイッチでは、カスタマー VLAN を作成または許可しないことを推奨します。

- **switchport vlan mapping all dot1q-tunnel** コマンド入力時の特定のネイティブ VLAN 設定はサポートされていません。
- Cisco NX-OS リリース 9.3(5) 以降では、複数のプロバイダー タグを使用した選択的 Q-in-VNI は vPC ファブリック ピアリングをサポートします。
- プロバイダー VNI で、カスタマー VLAN から発信された ARP トラフィックの ARP 抑制を無効にします。

```
switch(config)# interface nve 1
switch(config-if-nve)# member VNI 10000011
switch(config-if-nve-vni)# no suppress-arp
```

- インターフェイスが **switchport vlan mapping all dot1q-tunnel** コマンドで設定されている場合、すべての着信トラフィックにタグを付ける必要があります。

複数のプロバイダー VLAN を使用した選択的 Q-in-VNI の設定

複数のプロバイダー VLAN で選択的 Q-in-VNI を設定できます。

始める前に

プロバイダー VLAN を設定し、VLAN を vn-segment に関連付ける必要があります。

手順の概要

1. 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。
2. レイヤ 2 VLAN を設定し、それらを vn-segment に関連付けます。
3. トラフィックが dot1Q VLAN タグ付きで着信するインターフェイス設定モードを開始します。

手順の詳細

手順

ステップ 1 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。

```
switch# configure terminal
```

ステップ 2 レイヤ 2 VLAN を設定し、それらを vn-segment に関連付けます。

```
switch(config)# vlan 10
vn-segment 10000010
switch(config)# vlan 20
 vn-segment 10000020
```

ステップ 3 トラフィックが dot1Q VLAN タグ付きで着信するインターフェイス設定モードを開始します。

```
switch(config)# interf port-channel 10
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
```

```
switch(config-if)# switchport trunk native vlan 3962
switch(config-if)# switchport vlan mapping 2-400 dot1q-tunnel 10
switch(config-if)# switchport vlan mapping 401-800 dot1q-tunnel 20
switch(config-if)# switchport vlan mapping 801-1200 dot1q-tunnel 30
switch(config-if)# switchport vlan mapping 1201-1600 dot1q-tunnel 40
switch(config-if)# switchport vlan mapping 1601-2000 dot1q-tunnel 50
switch(config-if)# switchport vlan mapping 2001-2400 dot1q-tunnel 60
switch(config-if)# switchport vlan mapping 2401-2800 dot1q-tunnel 70
switch(config-if)# switchport vlan mapping 2801-3200 dot1q-tunnel 80
switch(config-if)# switchport vlan mapping 3201-3600 dot1q-tunnel 90
switch(config-if)# switchport vlan mapping 3601-3960 dot1q-tunnel 100
switch(config-if)# switchport trunk allowed vlan 10,20,30,40,50,60,70,80,90,100,3961-3967
```

例

次に、複数のプロバイダー VLAN で選択的 Qinvni を設定する例を示します。

```
switch# show run vlan 121
vlan 121
vlan 121
  vn-segment 10000021

switch#
switch# sh run interf port-channel 5

interface port-channel5
  description VPC PO
  switchport
  switchport mode trunk
  switchport trunk native vlan 504
  switchport vlan mapping 11 dot1q-tunnel 111
  switchport vlan mapping 12 dot1q-tunnel 112
  switchport vlan mapping 13 dot1q-tunnel 113
  switchport vlan mapping 14 dot1q-tunnel 114
  switchport vlan mapping 15 dot1q-tunnel 115
  switchport vlan mapping 16 dot1q-tunnel 116
  switchport vlan mapping 17 dot1q-tunnel 117
  switchport vlan mapping 18 dot1q-tunnel 118
  switchport vlan mapping 19 dot1q-tunnel 119
  switchport vlan mapping 20 dot1q-tunnel 120
  switchport trunk allowed vlan 111-120,500-505
  vpc 5

switch#

switch# sh spanning-tree vlan 111

VLAN0111
  Spanning tree enabled protocol rstp
  Root ID    Priority    32879
             Address     7079.b3cf.956d
             This bridge is the root
             Hello Time  2  sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32879  (priority 32768 sys-id-ext 111)
             Address     7079.b3cf.956d
             Hello Time  2  sec  Max Age 20 sec  Forward Delay 15 sec

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
```

```
Po1             Desg FWD 1         128.4096 (vPC peer-link) Network P2p
Po5             Desg FWD 1         128.4100 (vPC) P2p
Eth1/7/2        Desg FWD 10        128.26   P2p

switch#


switch# sh vlan internal info mapping | b Po5
  ifindex Po5(0x16000004)
  vlan mapping enabled: TRUE
  vlan translation mapping information (count=10):
    Original Vlan       Translated Vlan
    -------------       ---------------
    11                   111
    12                   112
    13                   113
    14                   114
    15                   115
    16                   116
    17                   117
    18                   118
    19                   119
    20                   120
switch#


switch# sh consistency-checker vxlan selective-qinvni interface port-channel 5
Performing port specific checks for intf port-channel5
Port specific selective QinVNI checks for interface port-channel5 : PASS
Performing port specific checks for intf port-channel5
Port specific selective QinVNI checks for interface port-channel5 : PASS

switch#
```

QinQ-QinVNI の設定

QinQ-QinVNI の概要

- QinQ-QinVNI は VXLAN トンネリング機能で、トランク ポートをマルチタグ ポートとして設定して、ネットワーク上で伝送されるカスタマー VLAN を維持できます。
- マルチタグとして設定されているポートでは、パケットは複数のタグまたは少なくとも 1 つのタグが含まれていると想定されます。マルチタグパケットがこのポートに入力されると、最も外側のタグまたは最初のタグが provider-tag または provider-vlan として扱われます。残りのタグは、customer-tag またはcustomer-vlan として扱われます。
- この機能は、vPC ポートと非 vPC ポートの両方でサポートされます。
- **switchport trunk allow-multi-tag** コマンドが両方の vPC ピアで設定されていることを確認します。これはタイプ 1 の整合性チェックです。
- この機能は、VXLAN Flood と Learn および VXLAN EVPN でサポートされます。

QinQ-QinVNI の注意事項と制約事項

QinQ-QinVNI には、次の注意事項と制約事項があります。

- この機能は、Cisco Nexus 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、QinQ-QinVNI は Cisco Nexus 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、QinQ-QinVNI は Cisco Nexus 9332D-H2R スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(2)F 以降、QinQ-QinVNI は Cisco Nexus 93400LD-H1 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(3)F 以降、QinQ-QinVNI は Cisco Nexus 9364C-H1 スイッチでサポートされています。
- この機能は、vPC ファブリック ピアリングをサポートします。
- マルチタグポートでは、プロバイダー VLAN はポートの一部である必要があります。これらは、そのパケットの VNI を取得するために使用されます。
- タグなしパケットは、ネイティブ VLAN に関連付けられます。ネイティブ VLAN が設定されていない場合、パケットはデフォルト VLAN（VLAN 1）に関連付けられます。
- マルチタグ ポートで許可された VLAN の範囲内に存在しない、最も外側の VLAN タグ（provider-vlan）を持つパケットはドロップされます。
- ネイティブ VLAN に一致する最も外側のVLANタグ（provider-vlan）タグが付いたパケットは、ネイティブ VLAN のドメインでルーティングまたはブリッジングされます。
- この機能は VXLAN ブリッジングをサポートしますが、VXLAN ルーティングはサポートしません。
- VXLAN VLAN でスヌーピングが有効になっている場合、3 つ以上の Q タグを持つマルチキャスト データ トラフィックはサポートされません。
- 両方の vPC ピアでプロバイダー VLAN をアップ状態にするために、少なくとも 1 つのマルチタグ トランク ポートが必要です。そうしないと、これらのプロバイダー VLAN のピアリンクを経由するトラフィックは、すべての内部 C タグを伝送しません。
- vPC VTEPでこの機能を実行する場合は、**system dot1q-tunnel transit vlan** *vlan-range* コマンドが必要です。

QinQ-QinVNI の設定

（注） 同じマルチタグ トランクポートでネイティブ VLAN（タグなしトラフィック）を伝送することもできます。

マルチタグ ポート上のネイティブ VLAN は、別のマルチタグ ポート上のプロバイダー VLAN または同じスイッチ上の dot1q 対応ポートとして設定できません。

allow-multi-tag コマンドは、トランク ポートでのみ使用できます。アクセス ポートまたは dot1q ポートでは使用できません。

allow-multi-tag コマンドは、ピア リンク ポートでは使用できません。マルチタグが有効になっているポート チャネルは、vPC ピアリンクとして設定しないでください。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. **switchport**
4. **switchport mode trunk**
5. **switchport trunk native vlan** *vlan-id*
6. **switchport trunk allowed vlan** *vlan-list*
7. **switchport trunk allow-multi-tag**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**interface ethernet** *slot/port* 例： `switch(config)# interface ethernet1/7`	設定するインターフェイスを指定します。
ステップ 3	**switchport** 例： `switch(config-inf)# switchport`	ポートをレイヤ 2 ポートとして設定します。
ステップ 4	**switchport mode trunk** 例：	インターフェイスをレイヤ2トランク ポートとして設定します。

	コマンドまたはアクション	目的
	`switch(config-inf)# switchport mode trunk`	
ステップ 5	**switchport trunk native vlan** *vlan-id* 例： `switch(config-inf)# switchport trunk native vlan 30`	802.1Q トランクのネイティブ VLAN を設定します。有効な値は 1 ～ 4094 です。デフォルト値は VLAN 1 です。
ステップ 6	**switchport trunk allowed vlan** *vlan-list* 例： `switch(config-inf)# switchport trunk allowed vlan 10,20,30`	トランク インターフェイスの許可 VLAN を設定します。デフォルトでは、トランク インターフェイス上のすべての VLAN（1 ～ 3967 および 4048 ～ 4094）が許可されます。VLAN 3968 ～ 4047 は、内部で使用するデフォルトで予約されている VLAN です。
ステップ 7	**switchport trunk allow-multi-tag** 例： `switch(config-inf)# switchport trunk allow-multi-tag`	許可された VLAN をネイティブ VLAN を除くプロバイダー VLAN として設定します。次の例では、VLAN 10 および 20 はプロバイダー VLAN であり、複数の内部 Q タグを伝送できます。ネイティブ VLAN 30 は内部 Q タグを伝送しません。

例

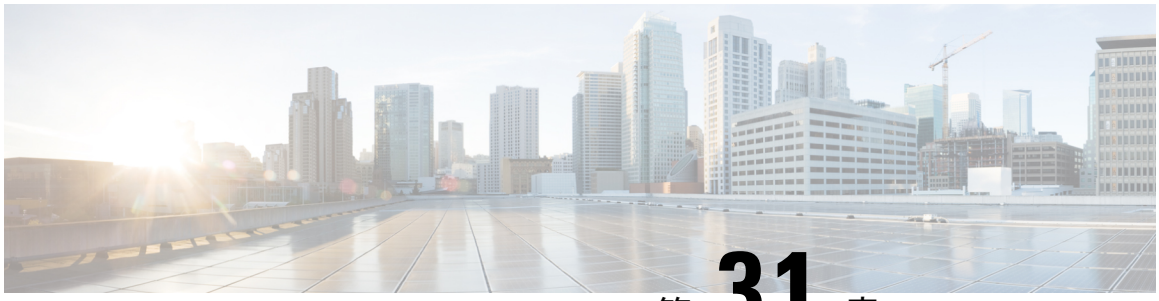
```
interface Ethernet1/7
switchport
switchport mode trunk
switchport trunk native vlan 30
switchport trunk allow-multi-tag
switchport trunk allowed vlan 10,20,30
no shutdown
```

VNI の削除

VNI を削除するには、次の手順を実行します。

手順

ステップ 1 NVE で VNI を削除します。
ステップ 2 BGP から VRF を削除します（レイヤ 3 VNI のデコミッション時に適用）。
ステップ 3 SVI を削除します。
ステップ 4 VLAN と VNI を削除します。



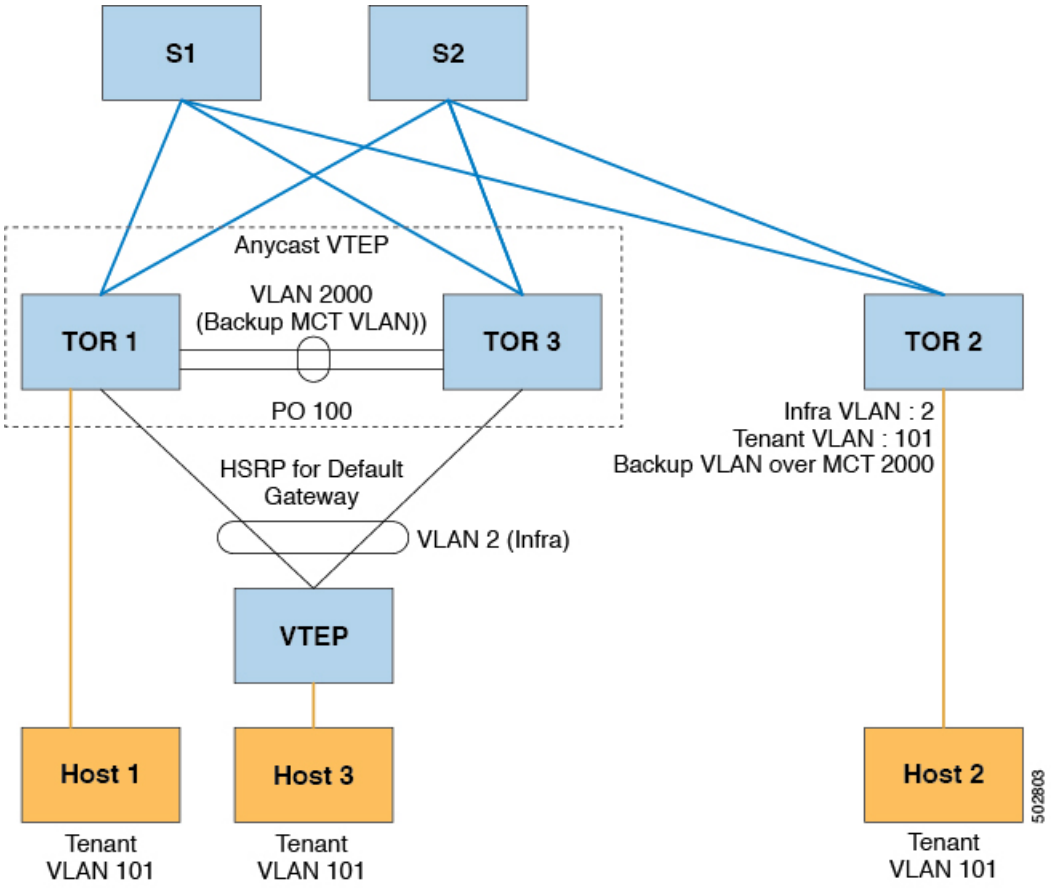
第 31 章

バドノードの設定

この章は、次の内容で構成されています。

vPC での VXLAN バドノードの概要

図 *70 : PIM-SM* および *OSPF* ベースのアンダーレイ ネットワーク



（注） バドノードトポロジでは、vPC の背後にある VTEP の送信元 IP は、インフラ VLAN と同じサブネットに属している必要があります。この SVI では、プロキシ ARP を有効にする必要があります。次に例を示します。

```
Interface Vlan2
ip proxy-arp
```

（注） **system nve infra-vlans** コマンドは、すべての SVI インターフェイス、バドノード トポロジに関するアップリンク インターフェイス、およびVXLANのvPCピアリンクに使用されるVLANをインフラ VLAN として指定します。インフラ VLAN の特定の組み合わせを設定しないでください。たとえば、2 と 514、10 と 522 は 512 離れています。

Cisco Nexus 9200、9300-EX、および 9300-FX/FX2/FX3 および 9300-GX プラットフォーム スイッチの場合は、**system nve infra-vlans** コマンドを使用して、インフラ VLAN として使用される VLAN を設定します。

vPC トポロジでの VXLAN バドノードの例

- 必要な機能のイネーブル化

```
feature ospf
feature pim
feature interface-vlan
feature vn-segment-vlan-based
feature hsrp
feature lacp
feature vpc
feature nv overlay
```

- PIM anycast RP の設定

この例では、1.1.1.1 がエニーキャスト RP アドレスです。

```
ip pim rp-address 1.1.1.1 group-list 225.0.0.0/8
```

- VLAN コンフィギュレーション

この例では、テナント VLAN 101 ～ 103 が vn-segment にマッピングされます。

```
vlan 1-4,101-103,2000
vlan 101
  vn-segment 10001
vlan 102
  vn-segment 10002
vlan 103
  vn-segment 10003
```

- vPC の設定

```
vpc domain 1
  peer-switch
  peer-keepalive destination 172.31.144.213
  delay restore 180
```

```
  peer-gateway
  ipv6 nd synchronize
  ip arp synchronize
```

- インフラ VLAN SVI の構成

```
interface Vlan2
  no shutdown
  no ip redirects
  ip proxy-arp
  ip address 10.200.1.252/24
  no ipv6 redirects
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  ip igmp static-oif route-map match-mcast-groups
  hsrp version 2
  hsrp 1
    ip 10.200.1.254
```

- マルチキャスト グループの照合用ルート マップ

  個々の VXLAN マルチキャスト グループは、バックアップ SVI MCT にスタティック OIF を必要とします。

```
route-map match-mcast-groups permit 1
  match ip multicast group 225.1.1.1/32
```

- バックアップ SVI の MCT での設定
  - 設定オプション 1：

```
interface Vlan2000
no shutdown
ip address 20.20.20.1/24
ip router ospf 1 area 0.0.0.0
ip pim sparse-mode
```

  - 設定オプション 2：

```
interface Vlan2000
no shutdown
ip address 20.20.20.1/24
ip router ospf 1 area 0.0.0.0
ip pim sparse-mode
```

- インフラ VLAN 伝送用の vPC インターフェイスの設定

```
interface port-channel1
  switchport mode trunk
```

```
  switchport trunk allowed vlan 2
  vpc 1
```

- MCT の設定

```
interface port-channel100
  switchport mode trunk
  spanning-tree port type network
  vpc peer-link
```

（注） NVE インターフェイスを作成するには、次の 2 つのコマンド プロシージャのいずれかを選択できます。VNIの数が少ない場合は、最初のものを使用します。多数の VNI を設定するには、2 番目の手順を使用します。

NVE の設定

オプション 1

```
interface nve1
  no shutdown
  source-interface loopback0
  member vni 10001 mcast-group 225.1.1.1
  member vni 10002 mcast-group 225.1.1.1
  member vni 10003 mcast-group 225.1.1.1
```

オプション 2

```
interface nve1
  no shutdown
  source-interface loopback0
  global mcast-group 225.1.1.1
  member vni 10001
  member vni 10002
  member vni 10003
```

- ループバック インターフェイスの設定

```
interface loopback0
  ip address 101.101.101.101/32
  ip address 99.99.99.99/32 secondary
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

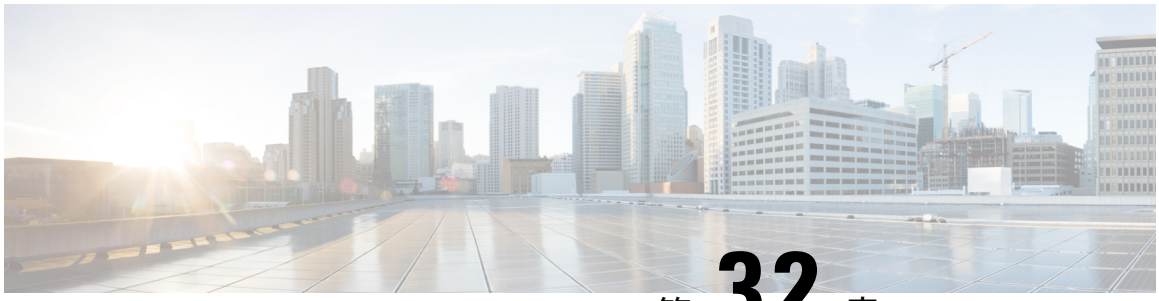
- show コマンド

```
tor1# sh nve vni
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP

Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      10001    225.1.1.1         Up    DP   L2 [101]
nve1      10002    225.1.1.1         Up    DP   L2 [102]
nve1      10003    225.1.1.1         Up    DP   L2 [103]

tor1# sh nve peers
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- ---------------  ----- --------- -------- -----------------
nve1      10.200.1.1       Up    DP        00:07:23 n/a
nve1      10.200.1.2       Up    DP        00:07:18 n/a
nve1      102.102.102.102  Up    DP        00:07:23 n/a


tor1# sh ip mroute 225.1.1.1
IP Multicast Routing Table for VRF "default"

(*, 225.1.1.1/32), uptime: 00:07:41, ip pim nve static igmp
  Incoming interface: Ethernet2/1, RPF nbr: 10.1.5.2
  Outgoing interface list: (count: 3)
    Vlan2, uptime: 00:07:23, igmp
    Vlan2000, uptime: 00:07:31, static
    nve1, uptime: 00:07:41, nve

(10.200.1.1/32, 225.1.1.1/32), uptime: 00:07:40, ip mrib pim nve
  Incoming interface: Vlan2, RPF nbr: 10.200.1.1
  Outgoing interface list: (count: 3)
    Vlan2, uptime: 00:07:23, mrib, (RPF)
    Vlan2000, uptime: 00:07:31, mrib
    nve1, uptime: 00:07:40, nve

(10.200.1.2/32, 225.1.1.1/32), uptime: 00:07:41, ip mrib pim nve
  Incoming interface: Vlan2, RPF nbr: 10.200.1.2
  Outgoing interface list: (count: 3)
    Vlan2, uptime: 00:07:23, mrib, (RPF)
    Vlan2000, uptime: 00:07:31, mrib
    nve1, uptime: 00:07:41, nve

(99.99.99.99/32, 225.1.1.1/32), uptime: 00:07:41, ip mrib pim nve
  Incoming interface: loopback0, RPF nbr: 99.99.99.99
  Outgoing interface list: (count: 3)
    Vlan2, uptime: 00:07:23, mrib
    Vlan2000, uptime: 00:07:31, mrib
    Ethernet2/5, uptime: 00:07:39, pim
(102.102.102.102/32, 225.1.1.1/32), uptime: 00:07:40, ip mrib pim nve
  Incoming interface: Ethernet2/1, RPF nbr: 10.1.5.2
  Outgoing interface list: (count: 1)
    nve1, uptime: 00:07:40, nve


tor1# sh vpc
Legend:
                - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 1
Peer status                       : peer adjacency formed ok
vPC keep-alive status             : peer is alive
Configuration consistency status  : success
```

```
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : secondary, operational primary
Number of vPCs configured         : 4
Peer Gateway                      : Enabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Disabled
Delay-restore status              : Timer is off.(timeout = 180s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)

vPC Peer-link status
---------------------------------------------------------------------
id   Port   Status Active vlans
--   ----   ------ --------------------------------------------------
1    Po100  up     1-4,101-103,2000

vPC status
----------------------------------------------------------------------------
id   Port   Status Consistency Reason                     Active vlans
--   ----   ------ ----------- ------                     ------------
1    Po1    up     success     success                    2

2    Po2    up     success     success                    2

tor1# sh vpc consistency-parameters global

    Legend:
        Type 1 : vPC will be suspended in case of mismatch

Name                        Type  Local Value            Peer Value
-------------               ----  ---------------------- -----------------------
Vlan to Vn-segment Map      1     3 Relevant Map(s)      3 Relevant Map(s)
STP Mode                    1     Rapid-PVST             Rapid-PVST
STP Disabled                1     None                   None
STP MST Region Name         1     ""                     ""
STP MST Region Revision     1     0                      0
STP MST Region Instance to  1
 VLAN Mapping
STP Loopguard               1     Disabled               Disabled
STP Bridge Assurance        1     Enabled                Enabled
STP Port Type, Edge         1     Normal, Disabled,      Normal, Disabled,
BPDUFilter, Edge BPDUGuard        Disabled               Disabled
STP MST Simulate PVST       1     Enabled                Enabled
Nve Oper State, Secondary   1     Up, 99.99.99.99, DP    Up, 99.99.99.99, DP
IP, Host Reach Mode
Nve Vni Configuration       1     10001-10003            10001-10003
Interface-vlan admin up     2     2,2000                 2,2000
Interface-vlan routing      2     1-4,2000               1-4,2000
capability
Allowed VLANs               -     1-4,101-103,2000       1-4,101-103,2000
Local suspended VLANs       -     -
```

第 部

VXLAN セキュリティの構成



第 32 章

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの設定

この章は、次の項で構成されています。

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトについて

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトは、VXLAN ベースのマルチサイトファブリックのデータ セキュリティとデータ整合性を保証します。この機能は、UDP パケットの IEEE MACsec の暗号化メカニズムを使用して、許可された VXLAN EVPN エンドポイント間にセキュアなトンネルを提供します。

CloudSec セッションは、2 つの異なるサイトのボーダー ゲートウェイ（BGW）間の DCI を介したポイントツーポイントです。サイト間のすべての通信は、VIP の代わりにマルチサイト PIP を使用します。移行情報の詳細については、VIP を使用するマルチサイトから PIP を使用するマルチサイトへの移行 （710 ページ） を参照してください。

CloudSec を使用したセキュア VXLAN EVPN マルチサイトが、ピアごとに有効になっていることを確認します。CloudSec をサポートしないピアは、CloudSec をサポートするピアと動作できますが、トラフィックは暗号化されません。CloudSec 非対応サイトから CloudSec 対応サイトへの移行中にのみ、暗号化されていないトラフィックを許可することをお勧めします。

CloudSec キー交換では BGP が使用され、MACsec では MACsec Key Agreement（MKA）が使用されます。CloudSec コントロール プレーンは、BGP IPv4 アドレス ファミリをキー情報の交換に使用します。CloudSecキーは、アンダーレイBGPセッションを使用するBGP IPv4ルートのトンネルカプセル化（トンネルタイプ18）属性の一部として伝送されます。

キー ライフタイムおよびヒットレス キー ロールオーバー

CloudSec キー チェーンには、キー ID とオプションのライフタイムが設定された複数の事前共有キー（PSK）を含めることができます。事前共有キーは、トラフィックの暗号化と整合性検証のためにさらにキーを取得するために使用されるシードキーです。事前共有キーのリストは、異なるライフタイムを持つキーチェーンで設定できます。

キーのライフタイムには、キーが期限切れになる時刻が指定されます。ライフタイムが設定されている場合、ライフタイムの期限が切れた後に、MKA はキー チェーン内の次に設定された事前共有キーにロール オーバーします。キーのタイム ゾーンは、ローカルまたは UTC を指定できます。デフォルトの時間帯は UTC です。ライフタイム設定が存在しない場合は、無期限のデフォルト ライフタイムが使用されます。

CloudSec キー チェーンを設定するには、CloudSec キーチェーンとキーの設定 （694 ページ）を参照してください。

最初のキーのライフタイムが期限切れになると、リスト内の次のキーに自動的にロールオーバーします。同一のキーがリンクの両側で同時に設定されている場合、キーのロールオーバーはヒットレスになります。つまり、キーはトラフィックを中断せずにロールオーバーされます。つまり、トラフィックが中断されることなくキーがロールオーバーされます。キーのライフタイムは、ヒットレス キー ロールオーバーを実現するためにオーバーラップする必要があります。

証明書の有効期限と交換

証明書は、マスター セッション キーの交換に使用されます。証明書の有効期限が切れると、それ以降の MSK キーの再生成は行われません。現在のセキュリティで保護されたセッションは引き続き稼働し、SAK キーの再生成は構成どおりに実行されます。証明書はトラストポイントの下から削除する必要があり、さらに MSK キー再生成を実行するには、新しい証明書をインポートする必要があります。

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの注意事項と制約事項

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトには、次の注意事項と制約事項があります。

- Cisco NX-OS リリース 10.2(2)F 以降、vPC ボーダー ゲートウェイは Cisco Nexus 9300-FX2、-FX3 スイッチでサポートされます。
- CloudSec を使用しているセキュアな VXLAN EVPN マルチサイトは、Cisco NX-OS リリース 9.3(5) 以降 Cisco Nexus 9300-FX2 プラットフォーム スイッチでサポートされます。
- CloudSec を使用しているセキュアな VXLAN EVPN マルチサイトは、Cisco NX-OS リリース 10.1(1) 以降から Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- L3 インターフェイスおよび L3 ポートチャネルは DCI リンクとしてサポートされます。
- スイッチ宛ての CloudSec トラフィックは、DCI アップリンクを介してスイッチに入る必要があります。
- CloudSec を使用したセキュアな VXLAN EVPN マルチサイトは、ルートサーバ経由で接続されているサイト、またはフル メッシュ（ルート サーバなし）を使用して接続されているサイトでサポートされます。ルート サーバを介して接続されているサイトの場合は、サーバを Cisco NX-OS リリース 9.3(5) 以降のリリースにアップグレードし、CloudSec VXLAN EVPN トンネル暗号化の有効化 （691 ページ） の手順に従います。
- Cisco NX-OS リリース 10.1(1) 以降、VXLAN トンネル暗号化機能は Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- VXLAN トンネル暗号化機能は、Cisco Nexus 9348GC-FX3、9348GC-FX3PH、および 9332D-H2R、93400LD-H1、9364C-H1 スイッチでサポートされません。
- ICV は、Cisco NX-OS リリース 9.3(7) ではデフォルトで無効になっています。以前のリリース（Cisco NX-OS リリース 9.3(6)）のノードと cloudsec トンネル セッションを形成する場合は、ノードで ICV を無効にする必要があります。
- Cisco NX-OS リリース 10.3.3 以降、VXLAN トンネル暗号化機能は、事前共有キー（PSK）または公開キー インフラストラクチャ（PKI）を使用した証明書を使用して構成できます。
- CloudSec を使用して、同じサイト上のすべての BGW をセキュア VXLAN EVPN マルチサイト用に設定する必要があります。
- DCI リンクで CloudSec を使用するセキュア VXLAN EVPN マルチサイトと、内部ファブリックで MACsec を共存させることができます。ただし、同じポートまたはポート グループ（MAC ID）で同時に有効にすることはできません。
- CloudSec ピアを使用するセキュアな VXLAN EVPN マルチサイトは、それらの間のセキュアなトラフィックを復号化するために同じキー チェーン設定を持つ必要があります。

- Cisco Nexus 9300-FX2 ファミリ スイッチのセキュリティ キー配布の BGP IPv4 アップデートでは、最大 60 のピアがサポートされます。
- Cisco NX-OS リリース 10.2(3) 以降、セキュリティ キー配布の BGP IPv4 アップデートは Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。
- アクティブ タイマーが設定されたすべてのキーが期限切れになったときにセッションを維持するには、キーチェーンごとにライフタイムなしで 1 つのキーだけを設定します。ベスト プラクティスとして、キーごとにライフタイムを設定することを推奨します。
- CloudSec キーは、アンダーレイ BGP セッションを使用する BGP IPv4 ルートでトンネルカプセル化属性を使用して BGW間で交換されます。

  この属性が中間ノードによって伝播されない場合は、CloudSec エンドポイント ノード、つまり BGW間で直接 BGP IPv4 ユニキャスト セッションを設定する必要があります。
- 次の場合、各サイトの BGW 間で直接 eBGP ピアリングを確立する必要があります。
  - BGP は IPv4 ユニキャスト ルーティング プロトコルとして使用されますが、トンネル暗号化属性は DCI を介して伝播されません。
  - BGP 以外のルーティング プロトコルは、DCI の IPv4 ユニキャスト ルーティングに使用されます（OSPF など）。
- eBGP ピアリングは、次のインターフェイスとは異なるループバック インターフェイスを介して確立されます。
  - tunnel-encryption source-interface
  - nve source-interface
- eBGP ピアリングは、隣接関係の送信元として使用されるループバック IP をフィルタリングする必要があります。たとえば、Loopback10 を使用して CloudSec の eBGP ピアリングを確立する場合、Lo10 の IP はこの隣接関係でアドバタイズされません。
- CloudSec を使用したセキュアな VXLAN EVPN マルチサイトは、次をサポートします。
  - ボーダー ゲートウェイ上の直接接続された L2 ホスト
  - DCI インターフェイスの IP アンナンバード設定
  - マルチキャスト アンダーレイ
  - OAM パストレース
  - TRM
  - ボーダー ゲートウェイの VIP 専用モデル
  - ダウンストリーム VNI を使用した VXLAN EVPN
- Cisco NX-OS リリース 10.3(1) 以降、DSVNI を使用する vPC cloudsec は Cisco Nexus 9000 シリーズ スイッチでサポートされません。

- CloudSec が有効になっている場合、非中断の ISSU はサポートされません。
- Cloudsec PKI の展開では、異なる証明書タイプ（SUDI、サードパーティ RSA、サードパーティ ECC）を混在させることはできません。すべてのノードに同じタイプの証明書が必要です
- 異なる RSA キーサイズを持つノードは、暗号化/復号化に互換性があります。
- PSK セッションと PKI セッションは、展開内で共存できません。
- 証明書のサイズは 1.5 KB（2048 ビット キー サイズ）を超えることはできません。
- MCT レス VPC BGW はサポートされていません。
- 異なる証明書タイプ間の移行は、should-secure に移行し、すべての参加ノードからトラストポイント構成を削除してから、すべてのノードで新しいトラストポイントを構成することで実行できます。
- **feature tunnel-encryption** コマンドを使用して Cloudsec を最初に有効にすると、vPC ピア リンク ポートチャネルとその物理メンバー インターフェイスがフラップします。

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの設定

CloudSec を使用してセキュアな VXLAN EVPN マルチサイトを設定するには、次の手順に従います。

CloudSec VXLAN EVPN トンネル暗号化の有効化

CloudSec VXLAN EVPN トンネル暗号化を有効にするには、次の手順を実行します。

始める前に

IPv4 ユニキャスト アドレス ファミリで BGP ピアを設定します。IPv4 プレフィックスが CloudSec キーを伝送するトンネル コミュニティ属性とともに伝播されていることを確認します。

VXLAN EVPN マルチサイトを設定し、次のコマンドを使用して、CloudSec VXLAN EVPN トンネル暗号化のピア IP アドレスをアドバタイズします。

```
evpn multisite border-gateway ms-id
dci-advertise-pip
```

注意 **dci-advertise-pip** なしで VXLAN EVPN マルチサイトを設定すると、ボーダー ゲートウェイを VIP 専用モードに戻します。これは CloudSec VXLAN EVPN トンネル暗号化ではサポートされません。

ルート サーバを介して接続されているサイトには、次の 2 つのオプションがあります。

- デュアル RD を有効にする：このデフォルトの動作により、メモリが限られたリーフデバイスを処理するために、以前のリリースと同じメモリスケールが維持されます。すべての同一サイト BGW は、リモート BGW に EVPN ルートをアドバタイズする間、再発信ルートに同じ RD 値を使用します。
- デュアル RD の無効化：リーフデバイスのメモリ制限がない場合は、BGW で **no dual rd** コマンドを設定できます。EVPN ルートをリモート BGW にアドバタイズする間、同じ BGW で再発信されたルートに異なる RD 値が使用されます。

BGW でデュアル RD が有効になっているかどうかに応じて、次のいずれかの操作を実行します。

- デュアル RD が BGW で設定されている場合は、次の手順を実行します。

  1. BGW に BGP 追加パスを適用します。

```
router bgp as-num
    address-family l2vpn evpn
      maximum-paths number
    additional-paths send
    additional-paths receive
```

  2. BGW で各 L3VNI VRF のマルチパスを設定します。

```
vrf evpn-tenant-00001
  address-family ipv4 unicast
    maximum-paths 64
  address-family ipv6 unicast
    maximum-paths 64
```

  3. ルート サーバに BGP 追加パスを適用します。

```
router bgp as-num
  address-family l2vpn evpn
    retain route-target all
    additional-paths send
    additional-paths receive
    additional-paths selection route-map name

route-map name permit 10
  set path-selection all advertise
```

- **no dual rd** が BGW で設定されている場合、またはフル メッシュが設定されている場合は、次の手順を実行します。

  1. BGW でアドレス ファミリと最大パスを設定します。

```
router bgp as-num
  address-family l2vpn evpn
    maximum-paths number
```

  2. BGW で各 L3VNI VRF のマルチパスを設定します。

```
vrf evpn-tenant-00001
    address-family ipv4 unicast
      maximum-paths 64
    address-family ipv6 unicast
      maximum-paths 64
```

（注） BGP 追加パスは、ルート サーバでは必要ありません。

手順の概要

1. **configure terminal**
2. [**no**] **feature tunnel-encryption**
3. [**no**] **tunnel-encryption source-interface loopback** *number*
4. **tunnel-encryption icv**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	[**no**] **feature tunnel-encryption** 例： `switch(config)# feature tunnel-encryption`	CloudSec VXLAN EVPN トンネル暗号化を有効にします。
ステップ 3	[**no**] **tunnel-encryption source-interface loopback** *number* 例： `switch(config)# tunnel-encryption source-interface loopback 2`	トンネルの送信元をループバック インターフェイスとしてBGPループバックを指定します。設定された送信元インターフェイスのIPアドレスは、CloudSec VXLAN EVPN トンネル暗号化キー ルートを通知するためのプレフィックスとして使用されます。 （注） NVE送信元インターフェイスではなく、BGPループバック インターフェイスを入力します。 （注） MTU の変更は、インターフェイスのトンネル暗号化設定の前に行う必要があります。これにより、CRC ドロップ エラーが回避されます。

	コマンドまたはアクション	目的
ステップ 4	**tunnel-encryption icv** 例： `switch(config)# tunnel-encryption icv`	Integrity Check Value（ICV）を有効にします。ICV は、ポートに到着するフレームの整合性チェックを行います。生成された ICV がフレーム内の ICV と同じであれば、そのフレームは受け入れられ、同じでなければ破棄されます。これは、Cisco NX-OS リリース 9.3(7) 以降、サポートされます。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

次のタスク

CloudSec VXLAN EVPN トンネル暗号化を有効にした後、次の認証手順のいずれかを実行できます。

CloudSec キーチェーンとキーの設定。

または

PKI を使用した CloudSec 証明書ベースの認証構成 （696 ページ）

CloudSec キーチェーンとキーの設定

デバイスに CloudSec キーチェーンとキーを作成できます。

始める前に

CloudSec を使用したセキュア VXLAN EVPN マルチサイトが有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **key chain** *name* **tunnel-encryption**
3. [**no**] **key** *key-id*
4. [**no**] **key-octet-string** *octet-string* **cryptographic-algorithm** {**AES_128_CMAC** | **AES_256_CMAC**}
5. [**no**] **send-lifetime** *start-time* **duration** *duration*
6. （任意） **show key chain** *name*
7. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	[**no**] **key chain** *name* **tunnel-encryption**
	例：
	`switch(config)# key chain kc1 tunnel-encryption`
	`switch(config-tunnelencryptkeychain)#`
		CloudSec キーチェーンを作成して CloudSec キーのセットを保持し、トンネル暗号化キーチェーン設定モードを開始します。

ステップ 3	[**no**] **key** *key-id*
	例：
	`switch(config-tunnelencryptkeychain)# key 2000`
	`switch(config-tunnelencryptkeychain-tunnelencryptkey)#`
		CloudSec キーを作成し、トンネル暗号キー設定モードを開始します。範囲は 1ー32 オクテットで、最大サイズは 64 です。
		（注） キーの文字数は偶数でなければなりません。

ステップ 4	[**no**] **key-octet-string** *octet-string* **cryptographic-algorithm** {**AES_128_CMAC** | **AES_256_CMAC**} 例： `switch(config-tunnelencryptkeychain-tunnelencryptkey)# key-octet-string abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789 cryptographic-algorithm AES_256_CMAC`	そのキーの octet ストリングを設定します。*octet-string* 引数には、最大 64 文字の 16 進数文字を含めることができます。octet キーは内部でエンコードされるため、クリア テキストのキーは **show running-config tunnel-encryption** コマンドの出力に表示されません。
ステップ 5	[**no**] **send-lifetime** *start-time* **duration** *duration* 例： `switch(config-tunnelencryptkeychain-tunnelencryptkey)# send-lifetime 00:00:00 May 06 2020 duration 100000`	キーの送信ライフタイムを設定します。デフォルトでは、デバイスは開始時間を UTC として扱います。 *start-time* 引数は、キーがアクティブになる日時です。*duration* 引数はライフタイムの長さ（秒）です。範囲は 1800ー2147483646 秒（約68年）です。
ステップ 6	（任意） **show key chain** *name* 例： `switch(config-tunnelencryptkeychain-tunnelencryptkey)# show key chain kc1`	キーチェーンの設定を表示します。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-tunnelencryptkeychain-tunnelencryptkey)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

CloudSec ポリシーの設定。

PKI を使用した CloudSec 証明書ベースの認証構成

この章は、次の項で構成されています。

CloudSec への証明書のアタッチ

Cisco NX-OS デバイスとトラストポイント CA を関連付ける必要があります。Cisco NX-OS は、RSA アルゴリズムおよび ECC（224 および 521 ビット）アルゴリズム証明書をサポートします。トラストポイントまたは Secure Unique Device Identifier（SUDI）を cloudsec に関連付けるには、次の手順に従います。ユーザーは、次のいずれかのコマンドを実行する必要があります。

始める前に

トラストポイントを構成し、有効な証明書をインストールまたはインポートする方法については、「PKI の構成」を参照してください。

手順の概要

1. **tunnel-encryption pki trustpoint** *name*
2. **tunnel-encryption pki sudi** *name*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**tunnel-encryption pki trustpoint** *name*
	例：
	`switch# tunnel-encryption pki trustpoint myCA_2K`
	`switch(config)#`
		トラストポイントをクラウドセキュリティに関連付けます。または、ステップ 2 のコマンドを実行します。データトラフィックが中断されるため、トラストポイントラベルの動的な変更は実行できません。

ステップ 2	**tunnel-encryption pki sudi** *name*
	例：
	`switch(config)# tunnel-encryption pki sudi`
	`switch(config-trustpoint)#`
		SUDI をクラウドセキュリティに関連付けます。
		（注） Cisco デバイスには、Secure Unique Device Identifier（SUDI）証明書と呼ばれる一意の識別子があります。このハードウェア証明書は、ステップ 1 の代わりに利用できます。

個別のループバック

PKI ループバックを構成するには、次のいずれかの手順を実行します。

手順の概要

1. **tunnel-encryption pki source-interface** *loopback*
2. **tunnel-encryption pki source-interface cloudsec-loopback**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**tunnel-encryption pki source-interface** *loopback*
	例：
	`switch# tunnel-encryption pki source-interface loopback0`
	`switch(config)#`
		個別のループバックを構成します。または、ステップ 2 のコマンドを実行します。

ステップ 2	**tunnel-encryption pki source-interface cloudsec-loopback** 例： `switch(config)# tunnel-encryption pki source-interface cloudsec-loopback`	cloudsec 送信元インターフェイス ループバックと同じループバックを使用します。

CloudSec ポリシーの設定

異なるパラメータを使用して複数の CloudSec ポリシーを作成できます。しかし、1 つのインターフェイスでアクティブにできるポリシーは 1 つのみです。

始める前に

CloudSec を使用したセキュア VXLAN EVPN マルチサイトが有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. （任意） [**no**] **tunnel-encryption must-secure-policy**
3. [**no**] **tunnel-encryption policy** *name*
4. （任意） [**no**] **cipher-suite** *name*
5. （任意） [**no**] **window-size** *number*
6. （任意） [**no**] **sak-rekey-time** *time*
7. （任意） **show tunnel-encryption policy**
8. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	（任意）[**no**] **tunnel-encryption must-secure-policy** 例： `switch(config)# tunnel-encryption must-secure-policy`	暗号化されていないパケットがセッションの回線を介して送信されないようにします。CloudSec ヘッダーを伝送しないパケットはドロップされます。 このコマンドの **no** 形式は、暗号化されていないトラフィックを許可します。CloudSec 非対応サイトから CloudSec 対応サイトへの移行中にのみ、暗号化されていないトラフィックを許可することをお勧めします。デフォルトでは、CloudSec を使用するセキュアな VXLAN EVPN マルチサイトは「セキュア」モードで動作することが必要です。

ステップ 3	[**no**] **tunnel-encryption policy** *name*
	例：
	`switch(config)# tunnel-encryption policy p1`
	`switch(config-tunenc-policy)#`
		CloudSec ポリシーを作成します。

ステップ 4	（任意）[**no**] **cipher-suite** *name* 例： `switch(config-tunenc-policy)# cipher-suite GCM-AES-XPN-256`	GCM-AES-XPN-128 または GCM-AES-XPN-256 のいずれかを設定します。デフォルト値は GCM-AES-XPN-256 です。
ステップ 5	（任意）[**no**] **window-size** *number* 例： `switch(config-tunenc-policy)# window-size 134217728`	インターフェイスが設定されたウィンドウサイズ未満のパケットを受け入れないように、再生保護ウィンドウを設定します。範囲は 134217728 ー 1073741823 IP パケットです。デフォルト値は 268435456 です。
ステップ 6	（任意）[**no**] **sak-rekey-time** *time* 例： `switch(config-tunenc-policy)# sak-rekey-time 1800`	SAK キー再生成を強制する時間を秒単位で設定します。このコマンドを使用して、セッションキーを予測可能な時間間隔に変更できます。有効な範囲は 1800 ～ 2592000 秒です。デフォルト値はありません。すべてのピアに同じキー再作成値を使用することを推奨します。
ステップ 7	（任意） **show tunnel-encryption policy** 例：	CloudSec ポリシー設定を表示します。

	コマンドまたはアクション	目的
	`switch(config-tunenc-policy)# show tunnel-encryption policy`	
ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config-tunenc-policy)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

CloudSec ピアの設定

CloudSec ピアの設定

この章は、次の内容で構成されています。

CloudSec ピアの設定

CloudSec ピアを設定できます。

始める前に

CloudSec を使用したセキュアな VXLAN EVPN マルチサイト

手順の概要

1. **configure terminal**
2. [**no**] **tunnel-encryption peer-ip** *peer-ip-address*
3. [**no**] **keychain** *name* **policy** *name*
4. **pki policy** *policy name*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	[**no**] **tunnel-encryption peer-ip** *peer-ip-address* 例： `switch(config)# tunnel-encryption peer-ip 33.1.33.33`	ピアの NVE 送信元インターフェイスの IP アドレスを指定します。

	コマンドまたはアクション	目的
ステップ 3	[**no**] **keychain** *name* **policy** *name* 例： `switch(config)# keychain kc1 policy p1`	CloudSec ピアにポリシーをアタッチします。ステップ 4 は、このステップの代わりに使用できます。
ステップ 4	**pki policy** *policy name* 例： `switch(config)# pki policy p1`	PKI を使用してピアに cloudsec ポリシーをアタッチしています。

次のタスク

DCI アップリンクで CloudSec を使用したセキュアな VXLAN EVPN マルチサイトを有効にする

DCI アップリンクで CloudSec を使用したセキュアな VXLAN EVPN マルチサイトを有効にする

すべての DCI アップリンクで CloudSec を使用してセキュアな VXLAN EVPN マルチサイトを有効にするには、次の手順に従います。

（注） この設定は、レイヤ 2 ポートには適用できません。

（注） CloudSec が動作中の DCI アップリンクに適用または削除されると、リンクがフラップします。リンクが数秒間ダウンしたままになる可能性があるため、瞬間的なフラップであるとは限りません。

始める前に

CloudSec を使用したセキュア VXLAN EVPN マルチサイトが有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **interface ethernet** *port/slot*
3. [**no**] **tunnel-encryption**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	[**no**] **interface ethernet** *port/slot*
	例：
	`switch(config)# interface ethernet 1/1`
	`switch(config-if)#`
		インターフェイス設定モードを開始します。

ステップ 3	[**no**] **tunnel-encryption** 例： `switch(config-if)# tunnel-encryption`	指定したインターフェイスで CloudSec を使用してセキュアな VXLAN EVPN マルチサイトを有効にします。

CloudSec を使用したセキュアな VXLAN EVPN マルチサイト

CloudSec 設定情報を使用してセキュアな VXLAN EVPN マルチサイトを表示するには、以下のタスクのいずれかを実行します。

コマンド	目的
show tunnel-encryption info global	CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの設定情報を表示します。
show tunnel-encryption policy [*policy-name*]	特定の CloudSec ポリシーまたはすべての CloudSec ポリシーの設定を表示します。
show tunnel-encryption session [**peer-ip** *peer-ip-address*] [**detail**]	エンドポイント間のセッションがセキュアかどうかなど、CloudSec セッションに関する情報を表示します。
show running-config tunnel-encryption	CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの実行中の設定報を表示します。
show bgp ipv4 unicast *ip-address*	BGP ルートのトンネル暗号化情報を表示します。

コマンド	目的
show bgp l2vpn evpn	レイヤ 2 VPN EVPN アドレス ファミリとルーティング テーブル情報を表示します。
show ip route *ip-address* **vrf** *vrf*	VRF ルートを表示します。
show l2route evpn mac evi *evi*	レイヤ 2 ルート情報を表示します。
show nve interface *interface* **detail**	NVE インターフェイスの詳細を表示します。
show running-config rpm	実行中の設定でキー テキストを表示します。 （注） **key-chain tunnelencrypt-psk no-show** コマンドを実行する前にコマンドを入力すると、キー テキストは実行中の設定で非表示になります（アスタリスク付き）。**reload ascii** コマンドを入力すると、キー テキストは実行中の設定から省略されます。
show running-config cert-enroll	トラストポイントとキーペアの構成を表示します。
show crypto ca certificates <*trustpoint_label*>	トラストポイントの証明書の内容を表示します。

次の例では、CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの設定情報を表示します。

```
switch# show tunnel-encryption info global
Global Policy Mode: Must-Secure
 SCI list: 0000.0000.0001.0002 0000.0000.0001.0004
 No. of Active Peers        : 1
```

次に、設定されているすべての CloudSec ポリシーを表示する例を示します。出力には、各ポリシーの暗号、ウィンドウ サイズ、および SAK 再試行時間が表示されます。

```
switch# show tunnel-encryption policy
Tunnel-Encryption Policy      Cipher            Window     SAK Rekey time
----------------------------  ----------------  ---------  --------------
cloudsec                      GCM-AES-XPN-256   134217728  1800
p1                            GCM-AES-XPN-256   1073741823
system-default-tunenc-policy  GCM-AES-XPN-256   268435456
```

次の例では、CloudSec セッションに関する情報を表示します。出力には、ピアの IP アドレスとポリシー、使用可能なキーチェーン、およびセッションがセキュアかどうかが示されます。

```
switch# show tunnel-encryption session
Tunnel-Encryption  Peer Policy  Keychain  RxStatus        TxStatus
-----------------  -----------  --------  -----------     --------------
33.1.33.33         p1           kc1       Secure (AN: 0)  Secure (AN: 2)
33.2.33.33         p1           kc1       Secure (AN: 0)  Secure (AN: 2)
```

```
33.3.33.33          p1            kc1       Secure (AN: 0)  Secure (AN: 2)
44.1.44.44          p1            kc1       Secure (AN: 0)  Secure (AN: 0)
44.2.44.44          p1            kc1       Secure (AN: 0)  Secure (AN: 0)
```

次の例は、 PKI 証明書トラストポイントに基づく Cloudsec セッションに関する情報を表示しています。

```
switch# sh tunnel-encryption session
Tunnel-Encryption Peer   Policy                                    Keychain
RxStatus         TxStatus
------------------------ ----------------------------------------
---------------------------------------- ----------------- -----------------
20.20.20.2               p1                                       PKI: myCA (RSA)
Secure (AN: 0)    Secure (AN: 0)
32.11.11.4               p1                                       PKI: myCA (RSA)
Secure (AN: 0)    Secure (AN: 0)
```

次に、BGP ルートのトンネル暗号化情報の例を示します。

```
switch# show bgp ipv4 unicast 199.199.199.199 □ Source-loopback configured on peer BGW
for CloudSec
BGP routing table information for VRF default, address family IPv4 Unicast
BGP routing table entry for 199.199.199.199/32, version 109
Paths: (1 available, best #1)
Flags: (0x8008001a) (high32 0x000200) on xmit-list, is in urib, is best urib route, is
in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop, in rib
  AS-Path: 1000 200 , path sourced external to AS
    89.89.89.89 (metric 0) from 89.89.89.89 (89.89.89.89)
      Origin IGP, MED not set, localpref 100, weight 0
      Tunnel Encapsulation attribute: Length 120

  Path-id 1 advertised to peers:
    2.2.2.2
```

次の例は、MAC が仮想 ESI に接続されているかどうかを示しています。

```
switch(config)# show bgp l2vpn evpn 0012.0100.000a
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 110.110.110.110:32876
BGP routing table entry for [2]:[0]:[0]:[48]:[0012.0100.000a]:[0]:[0.0.0.0]/216, version
 13198
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
             Imported paths list: l2-10109
  AS-Path: 1000 200 , path sourced external to AS
    10.10.10.10 (metric 0) from 89.89.89.89 (89.89.89.89)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10109
      Extcommunity: RT:100:10109 ENCAP:8
      ESI: 0300.0000.0000.0200.0309

  Path-id 1 not advertised to any peer

Route Distinguisher: 199.199.199.199:32876
```

```
BGP routing table entry for [2]:[0]:[0]:[48]:[0012.0100.000a]:[0]:[0.0.0.0]/216, version
 24823
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
             Imported paths list: l2-10109
  AS-Path: 1000 200 , path sourced external to AS
    9.9.9.9 (metric 0) from 89.89.89.89 (89.89.89.89)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10109
      Extcommunity: RT:100:10109 ENCAP:8
      ESI: 0300.0000.0000.0200.0309

  Path-id 1 not advertised to any peer
```

次に、リモート サイトから受信した EVPN タイプ 5 ルート用に作成された ECMP の例を示します。

```
switch(config)# show ip route 205.205.205.9 vrf vrf903
IP Route Table for VRF "vrf903"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

205.205.205.9/32, ubest/mbest: 2/0
    *via 9.9.9.9%default, [20/0], 11:06:32, bgp-100, external, tag 1000, segid: 900003
tunnelid: 0x9090909 encap: VXLAN

    *via 10.10.10.10%default, [20/0], 3d05h, bgp-100, external, tag 1000, segid: 900003
 tunnelid: 0xa0a0a0a encap: VXLAN
```

次の例は、リモート サイトから受信した MAC に ESI ベースの MAC マルチパスが設定されているかどうかを示しています。

```
switch(config)# show l2route evpn mac evi 109 mac 0012.0100.000a detail

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv (AD):Auto-Delete (D):Del Pending
(S):Stale (C):Clear, (Ps):Peer Sync (O):Re-Originated (Nho):NH-Override
(Pf):Permanently-Frozen, (Orp): Orphan

Topology Mac Address    Prod   Flags  Seq No Next-Hops
-------- -------------- ------ ------ ------ ----------------------
109      0012.0100.000a BGP    SplRcv 0          9.9.9.9 (Label: 10109)
                                             10.10.10.10 (Label: 10109)
         Route Resolution Type: ESI
         Forwarding State: Resolved (PL)
         Resultant PL: 9.9.9.9, 10.10.10.10
         Sent To: L2FM
         ESI : 0300.0000.0000.0200.0309
         Encap: 1
```

次の例は、PIP を使用した VXLAN EVPN マルチサイトが設定されていることを示しています。

```
switch(config)# show nve interface nve1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
```

```
VPC Capability: VPC-VIP-Only [not-notified]
Local Router MAC: 700f.6a15.c791
Host Learning Mode: Control-Plane
Source-Interface: loopback0 (primary: 14.14.14.14, secondary: 0.0.0.0)
Source Interface State: Up
Virtual RMAC Advertisement: No
NVE Flags:
Interface Handle: 0x49000001
Source Interface hold-down-time: 180
Source Interface hold-up-time: 30
Remaining hold-down time: 0 seconds
Virtual Router MAC: N/A
Virtual Router MAC Re-origination: 0200.2e2e.2e2e
Interface state: nve-intf-add-complete
Multisite delay-restore time: 180 seconds
Multisite delay-restore time left: 0 seconds
Multisite dci-advertise-pip configured: True
Multisite bgw-if: loopback1 (ip: 46.46.46.46, admin: Up, oper: Up)
Multisite bgw-if oper down reason:
```

次の例は、実行中の設定のキーテキストを示しています。**key-chain tunnelencrypt-psk no-show** コマンドを入力すると、キーテキストは非表示になります。

```
switch# show running-config rpm
!Command: show running-config rpm
!Running configuration last done at: Mon Jun 15 14:41:40 2020
!Time: Mon Jun 15 15:10:27 2020

version 9.3(5) Bios:version 05.40
key chain inter tunnel-encryption
  key 3301
    key-octet-string 7
075f79696a58405441412e2a577f0f077d6461003652302552040a0b76015a504e370c
7972700604755f0e22230c03254323277d2f5359741a6b5d3a5744315f2f cryptographic-algorithm
AES_256_CMAC
key chain kc1 tunnel-encryption
  key 3537
    key-octet-string 7
072c746f172c3d274e33592e22727e7409106d00372532575803780077755621 3d4e0c7c00770576772
d08515e0804553124577f5a522e046d6a5f485c35425f59 cryptographic-algorithm AES_256_CMAC
    send-lifetime local 09:09:40 Apr 15 2020 duration 1800
  key 2001
    key-octet-string 7
075f79696a58405441412e2a577f0f077d6461003652302552040a0b76015a504e370c7972700604755
f0e22230c03254323277d2f5359741a6b5d3a5744315f2f cryptographic-algorithm AES_256_CMAC
  key 2065
    key-octet-string 7
0729791f6f5e3d213347292d517308730c156c7737223554270f787c07722a513e450a0a0703070c062
e0256210d0e204120510d29222a051f1e594c2135375359 cryptographic-algorithm AES_256_CMAC
  key 2129
    key-octet-string 7
075c796f6f2a4c2642302f5c56790e767063657a4b564f2156777c0a020228564a32780e0472007005530
c5e560f04204056577f2a222d056d1f5c4c533241525d cryptographic-algorithm AES_256_CMAC
  key 2193
    key-octet-string 7
07577014195b402336345a5f260f797d7d6264044b50415755047a7976755a574d350b7e720a0202715d7
a50530d715346205d0c2d525c001f6b5b385046365a29 cryptographic-algorithm AES_256_CMAC

switch# configure terminal
switch(config)# key-chain tunnelencrypt-psk no-show
switch(config)# show running-config rpm

!Command: show running-config rpm
```

```
!Running configuration last done at: Mon Jun 15 15:10:44 2020
!Time: Mon Jun 15 15:10:47 2020

version 9.3(5) Bios:version 05.40
key-chain tunnelencrypt-psk no-show
key chain inter tunnel-encryption
  key 3301
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
key chain kc1 tunnel-encryption
  key 3537
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
    send-lifetime local 09:09:40 Apr 15 2020 duration 1800
  key 2001
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
  key 2065
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
  key 2129
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
  key 2193
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
```

次の例は、トラストポイントとキーペアの設定を示しています。

```
switch# show running-config cert-enroll
!Command: show running-config cert-enroll
!Running configuration last done at: Fri Apr 21 10:53:30 2023
!Time: Fri Apr 21 12:07:31 2023

version 10.3(3) Bios:version 05.47
crypto key generate rsa label myRSA exportable modulus 1024
crypto key generate rsa label myKey exportable modulus 1024
crypto key generate rsa label tmpCA exportable modulus 2048
crypto key generate ecc label src15_ECC_key exportable modulus 224
crypto ca trustpoint src15_ECC_CA
     ecckeypair switch_ECC_key and so on
     revocation-check  crl
crypto ca trustpoint myRSA
     rsakeypair myRSA
     revocation-check  crl
crypto ca trustpoint tmpCA
     rsakeypair tmpCA
     revocation-check  crl
crypto ca trustpoint myCA
     rsakeypair myKey
     revocation-check  crl
```

次の例は、トラストポイント下での証明書コンテンツを示しています。

```
switch(config)# show crypto ca certificates myCA
Trustpoint: myCA
certificate:
subject=CN = switch, serialNumber = FBO22411ABC
issuer=C = US, ST = CA, L = San Jose, O = Org, OU = EN, CN = PKI, emailAddress =
abc@xyz.com
serial=2F24FCE6823FCBE5A8AC72C82D0E8E24EB327B0C
notBefore=Apr 19 19:43:48 2023 GMT
notAfter=Aug 31 19:43:48 2024 GMT
SHA1 Fingerprint=D0:F8:1E:32:6E:6D:44:21:6B:AE:92:69:69:AD:88:73:69:76:B9:18
purposes: sslserver sslclient

CA certificate 0:
subject=C = US, ST = CA, L = San Jose, O = Org, OU = EN, CN = PKI, emailAddress =
abc@xyz.com
issuer=C = US, ST = CA, L = San Jose, O = Cisco, OU = EN, CN = PKI, emailAddress =
```

```
ca@ca.com
serial=1142A22DDDE63A047DE0829413359362042CCC31
notBefore=Jul 12 13:25:59 2022 GMT
notAfter=Jul 12 13:25:59 2023 GMT
SHA1 Fingerprint=33:37:C6:D5:F1:B3:E1:79:D9:5A:71:30:FD:50:E4:28:7D:E1:2D:A3
purposes: sslserver sslclient
```

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの統計情報の表示

次のコマンドを使用して、CloudSec 統計情報を使用してセキュア VXLAN EVPN マルチサイトを表示またはクリアできます。

コマンド	目的
show tunnel-encryption statistics [**peer-ip** *peer-ip-address*]	CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの統計情報を表示します。
clear tunnel-encryption statistics [**peer-ip** *peer-ip-address*]	CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの統計情報をクリアします。

次の例は CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの統計情報の例を示します。

```
switch# show tunnel-encryption statistics
Peer 16.16.16.16 SecY Statistics:

SAK Rx Statistics for AN [0]:
Unchecked Pkts: 0
Delayed Pkts: 0
Late Pkts: 0
OK Pkts: 8170598
Invalid Pkts: 0
Not Valid Pkts: 0
Not-Using-SA Pkts: 0
Unused-SA Pkts: 0
Decrypted In-Pkts: 8170598
Decrypted In-Octets: 4137958460 bytes
Validated In-Octets: 0 bytes

SAK Rx Statistics for AN [3]:
Unchecked Pkts: 0
Delayed Pkts: 0
Late Pkts: 0
OK Pkts: 0
Invalid Pkts: 0
Not Valid Pkts: 0
Not-Using-SA Pkts: 0
Unused-SA Pkts: 0
Decrypted In-Pkts: 0
Decrypted In-Octets: 0 bytes
Validated In-Octets: 0 bytes

SAK Tx Statistics for AN [0]:
Encrypted Protected Pkts: 30868929
Too Long Pkts: 0
```

```
Untagged Pkts: 0
Encrypted Protected Out-Octets: 15758962530 bytes
```

（注） トンネル暗号化の統計情報で、遅延パケットの増加と同時にトラフィックのドロップが見られる場合は、次のいずれかの理由が考えられます。

- パケットはリプレイ ウィンドウの外で受信されたため、廃棄されています。
- トンネル暗号化ピアが同期していません。
- 実際にセキュリティ リスクがあります。

このような状況では、対応するリモート ピアでトンネル暗号化ピアを削除してから再設定し、ピア セッションをリセットして再度同期する必要があります。

CloudSec を使用したセキュアな VXLAN EVPN マルチサイトの設定例

次に、keychain を使用してセキュア VXLAN EVPN マルチサイトを構成する例を示します。

```
key chain kc1 tunnel-encryption
key 2006
key-octet-string 7 075f79696a58405441412e2a577f0f077d6461003652302552040
a0b76015a504e370c7972700604755f0e22230c03254323277d2f5359741a6b5d3a5744315f2f
cryptographic-algorithm AES_256_CMAC

feature tunnel-encryption
tunnel-encryption source-interface loopback4
tunnel-encryption must-secure-policy

tunnel-encryption policy p1
  window-size 1073741823

tunnel-encryption peer-ip 11.1.11.11
  keychain kc1 policy p1
tunnel-encryption peer-ip 11.2.11.11
  keychain kc1 policy p1
tunnel-encryption peer-ip 44.1.44.44
  keychain kc1 policy p1
tunnel-encryption peer-ip 44.2.44.44
  keychain kc1 policy p1

interface Ethernet1/1
  tunnel-encryption

interface Ethernet1/7
  tunnel-encryption

interface Ethernet1/55
  tunnel-encryption

interface Ethernet1/59
  tunnel-encryption
```

```
evpn multisite border-gateway 111
dci-advertise-pip

router bgp 1000
router-id 12.12.12.12
no rd dual
address-family ipv4 unicast
  maximum-paths 10
address-family l2vpn evpn
  maximum-paths 10
vrf vxlan-900101
address-family ipv4 unicast
  maximum-paths 10
address-family ipv6 unicast
  maximum-paths 10

show tunnel-encryption session
Tunnel-Encryption Peer  Policy  Keychain  RxStatus        TxStatus
----------------------  ------  --------  --------------  --------------
11.1.11.11               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
11.2.11.11               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
44.1.44.44               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
44.2.44.44               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
```

次に、CloudSec 証明書ベースの認証を使用してセキュア VXLAN EVPN マルチサイトを構成する例を示します。

```
feature tunnel-encryption

tunnel-encryption must-secure-policy
tunnel-encryption pki trustpoint myCA
tunnel-encryption pki source-interface loopback3
tunnel-encryption source-interface loopback2
tunnel-encryption policy with-rekey
  sak-rekey-time 1800
tunnel-encryption peer-ip 7.7.7.7
  pki policy system-default-tunenc-policy


interface Ethernet1/20
  tunnel-encryption

interface Ethernet1/21
  tunnel-encryption

interface Ethernet1/25/1
  tunnel-encryption
```

次の例は、アウトバウンドルートマップを設定して、BGW のパスを最適なパスにする方法を示しています。この設定は、vPC BGW が BGP でピア vPC BGW の PIP アドレスを学習するときに行われます。

```
ip prefix-list pip_ip seq 5 permit 44.44.44.44/32 <<PIP2 address>>
route-map pip_ip permit 5
    match ip address prefix-list pip_ip
    set as-path prepend last-as 1
neighbor 45.10.45.10 <<R1 neighbor - Same route-map required for every DCI side underlay
 BGP peer>>
    inherit peer EBGP-PEERS
    remote-as 12000
    address-family ipv4 unicast
      route-map pip_ip out
```

VIP を使用するマルチサイトから PIP を使用するマルチサイトへの移行

VIP を使用するマルチサイトから PIP を使用するマルチサイトにスムーズに移行するには、次の手順を実行します。移行は一度に 1 つのサイトで実行する必要があります。移行中のトラフィック損失は最小限に抑えることができます。

1. すべてのサイトのすべての BGW を Cisco NX-OS リリース9.3(5) 以降のリリースにアップグレードします。
2. すべての BGW で BGP 最大パスを設定します。これは、ESI ベースの MAC マルチパスおよび BGP が EVPN タイプ 2 およびタイプ 5 ルートのすべてのネクスト ホップをダウンロードするために必要です。
3. 移行するサイトを 1 つずつ選択します。
4. 1 つの BGW を除き、同じサイトの BGW をシャットダウンします。NVE **shutdown** コマンドを使用して、BGW をシャットダウンできます。
5. トラフィックの損失を回避するには、アクティブな BGW で PIP を備えたマルチサイトを有効にする前に数分間待機します。これにより、同じサイトのシャットダウンBGWがEVPNルートを取り消すことができるため、リモート BGW はアクティブ BGW だけにトラフィックを送信します。
6. **dci-advertise-pip** コマンドを設定して、アクティブな BGW で PIP を使用したマルチサイトを有効にします。

   PIP 対応 BGW を備えたマルチサイトは、仮想 ESI の EVPN EAD-per-ES ルートをアドバタイズします。

   PIP 対応 BGW を備えたマルチサイトは、仮想 ESI、ネクスト ホップを PIP アドレス、PIP インターフェイス MAC を RMAC（該当する場合）として DCI にアドバタイズします。ファブリックへの EVPN タイプ 2 およびタイプ 5 ルートのアドバタイズに関する変更はありません。

   MAC ルートが ESI で受信されると、リモート BGW は ESI ベースの MAC マルチパスを実行します。
7. **dci-advertise-pip** コマンドを入力して、同じサイトの BGW を一度に 1 つずつ解除し、PIP でマルチサイトを有効にします。

   ESI はすべての同じサイト BGW と同じであるため、リモート BGW は MAC ルートの ESI ベースの MAC マルチパスを実行します。

   リモート BGW では、BGP はパスをマルチパスとして選択し、EVPN タイプ 5 ルートのすべてのネクスト ホップをダウンロードします。

既存の vPC BGW の移行

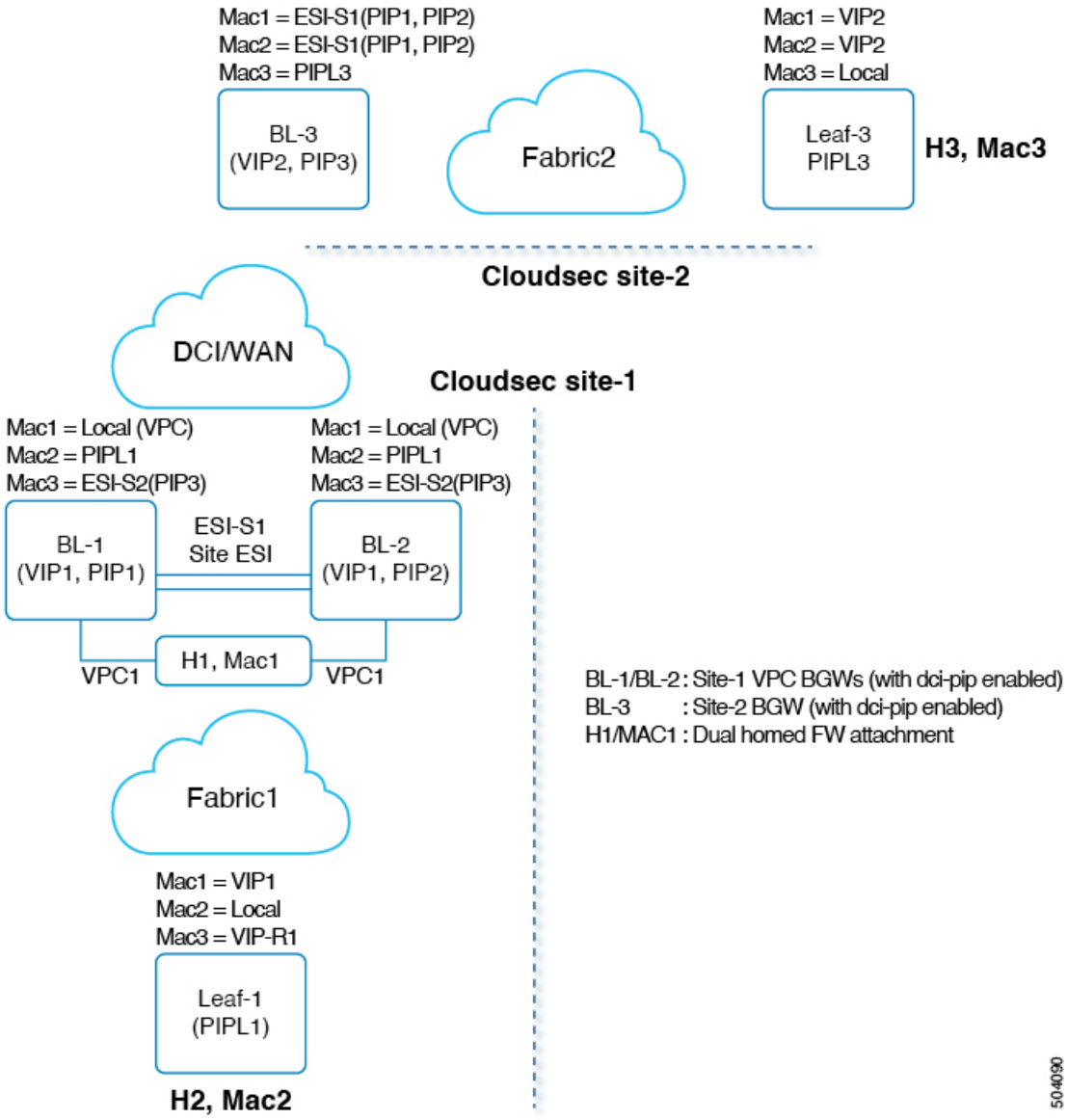
Cloudsec を使用できるように、既存の vPC BGW をスムーズに移行するには、次の手順に従います。移行は一度に 1 つのサイトで実行する必要があります。移行中のトラフィック損失は最小限に抑えることができます。

1. 両方の vPC BGW を、vPC Cloudsec が更新された最新のイメージにアップグレードします。
2. vPC セカンダリのインターフェイス nve1 をシャットダウンします。
3. vPC プライマリで **dci-advertise-pip** を有効にします。
4. インターフェイス nve1 がまだ vPC セカンダリでシャットモードになっている状態で、vPC セカンダリで **dci-advertise-pip** を構成します。
5. vPC セカンダリのインターフェイス nve1 のシャットダウンを解除します。

Cloudsec の vPC ボーダー ゲートウェイのサポート

次のトポロジは、Cloudsec の vPC ボーダー ゲートウェイ（BGW）のサポートを示しています。

図 *71 : Cloudsec の vPC BGW* サポート



vPC は、BGW へのデュアルホーム アタッチ/接続です。BGW は冗長性のための単一の VXLAN エンドポイントとして仮想的に機能し、両方のスイッチは共通のエミュレート/仮想 IP アドレス（VIP）を共有することによってアクティブ モードで機能します。DCI 上の VXLAN カプセル化は、BGW VTEP のプライマリ IP アドレスに基づいています。

上記のトポロジでは、ホスト H1/MAC1 は、Cloudsec 対応の vPC BGW BL-1/BL-2 にデュアルホーム接続されています。H1 は、ファブリックへの vPC BGW（VIP1）のセカンダリ ループバック IP アドレスで引き続きアドバタイズされます。ただし DCI に対しては、BL-1/BL-2 の両方が PIP としてネクストホップを使用して H1 をアドバタイズし、サイト ESI もタイプ 2 NLRI に追加されます。

エニーキャストおよび vPC BGW の Cloudsec 機能の場合、dci-advertise-pip はタイプ 2/タイプ 5 ルートが DCI にアドバタイズされる方法に関して、BGP 手順を変更するように構成されています。サイト内部ネットワークから受信したすべてのタイプ 2/タイプ 5 ルートは、vPC BGW の PIP としてネクスト ホップを使用して DCI にアドバタイズされます。

両方の vPC BGW は、それぞれのプライマリ IP アドレスを使用してルートをアドバタイズします。Site-ESI 属性が Type-2 NLRI に追加されます。vPC BGW 上のすべてのデュアル接続ホストは、PIP としてネクストホップでアドバタイズされ、サイト ESI 属性は DCI を介して接続されます。すべての孤立ホストは、DCI への PIP としてネクストホップでアドバタイズされ、サイト ESI 属性は付加されません。

vPC BGW がピア vPC BGW の PIP アドレスを学習し、DCI 側でアドバタイズする場合、両方の vPC BGW からの BGP パス属性は同じになります。したがって、DCI 中間ノードは PIP アドレスを所有していない vPC BGW からのパスを選択することになる可能性があります。このシナリオでは、リモート サイトからの暗号化されたトラフィックに MCT リンクが使用されます。vPC BGW BGP は、次の場合にピア vPC BGW の PIP アドレスを学習します。

- iBGP は vPC BGW 間で構成されます。
- BGP は、ファブリック側のアンダーレイ ルーティング プロトコルとして使用されます。
- アンダーレイ ルーティング プロトコルとして使用される IGP、および IGP ルートが BGP に再配布されます。

vPC BGW が BGP でピア vPC BGW の PIP アドレスを学習する場合、アウトバウンド ルートマップを構成して、BGW のパスを最適なパスにする必要があります。

リモート サイト BGW では、直接接続された L3 ホストは両方の vPC BGW から学習されます。通常直接接続された BGW からのパスは、AS パスが低いため優先されます。L3 ホストまたは L3 ネットワークが vPC ペア BGW に二重接続されている場合、ローカル パスは両方の vPC ペアで選択されます。

vPC BGW CloudSec 展開の拡張コンバージェンス

従来、単一のループバック インターフェイスは NVE 送信元 インターフェイスとして設定され、vPC コンプレックスの PIP と VIP の両方が構成されています。Cisco NX-OS リリース 10.3(2)F 以降では、CloudSec 対応の vPC BGW に個別のループバックを構成できます。vPC 展開でのコンバージェンスを向上させるために、NVE の下で送信元とエニーキャスト IP アドレスに個別のループバック インターフェイスを使用することをお勧めします。送信元インターフェイスに構成されている IP アドレスは vPC ノードの PIP であり、エニーキャスト インターフェイスに構成されている IP アドレスはその vPC コンプレックスの VIP です。NVE エニーキャスト インターフェイスも構成されている場合、NVE ソース インターフェイスで設定されたセカンダリ IP は効果がないことに注意してください。

個別のループバックを使用すると、DCI 側を宛先とするデュアル接続 EVPN タイプ 2 およびタイプ 5 トラフィックのコンバージェンスが改善されます。

エニーキャスト インターフェイスへの移行

ユーザーがエニーキャスト インターフェイスを指定したい場合、ユーザーは既存の送信元 インターフェイスを構成解除し、送信元 インターフェイスとエニーキャスト インターフェイスの両方で再構成する必要があります。これにより、一時的なトラフィック損失が発生します。すべてのグリーン フィールド展開では、指定されたコンバージェンスの問題を回避するために、送信元 インターフェイスとエニーキャスト インターフェイスの両方を設定することをお勧めします。

vPC BGW CloudSec 展開用の拡張コンバージェンスを使用した NVE インターフェイスの構成

ユーザーは、vPC BGW の NVE 送信元インターフェイスとともにエニーキャストインターフェイスを指定する必要があります。現在の VxLANv6 展開では、送信元 インターフェイスとエニーキャストインターフェイスの両方を指定するプロビジョニングがすでに存在しています。VxLANv4 の vPC コンバージェンスを改善するには、エニーキャスト オプションが必須です。

設定例：

```
interface nve <number>
        source-interface <interface> [anycast <anycast-intf>]
```

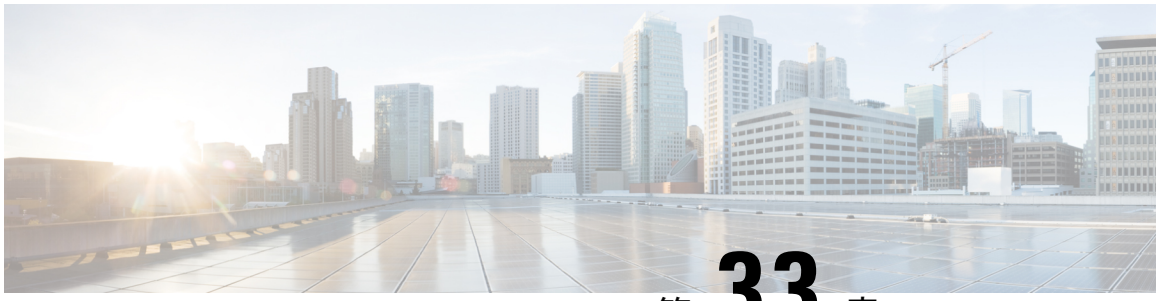
iBGP セッションの要件

アンダーレイ IPv4/IPv6 ユニキャスト iBGP セッションは、vPC BGW ピア ノード間で構成する必要があります。これは、vPC BGW での DCI 分離中のキー伝播に対応するためです。

PSK CloudSec 構成から証明書ベース認証 CloudSec 構成への移行

自動キーイングへの移行中は、サイトが新しい構成または機能リストに移行している間、VTEP 間セッションでクリア トラフィックを送受信することが期待されます。この間、暗号化されていないトラフィックがセッションでドロップされないように、ポリシーを **should-secure** として構成する必要があります。

1. すべてのノードで tunnel-encryption 設定を **should-secure** に変更します。
2. 一度に 1 ノードずつ移行を実行します。
3. ピアからキーチェーンと cloudsec ポリシーを削除します。
4. SSL 証明書を使用する場合は、有効な CA を使用してトラスト ポイントと証明書を構成するか、または SUDI 証明書を構成します。
5. トラスト ポイントを Cloudsec に接続します。
6. cloudsec ポリシーをピアに適用します。
7. すべてのノードが自動キーイングに変更されたら、必要に応じて構成を **must-secure** に変更します。



第 33 章

VXLAN ACL の構成

この章は、次の内容で構成されています。

アクセス コントロール リストについて

表 *15 : Cisco Nexus 92300YC*、*92160YC-X*、*93120TX*、*9332PQ*、および *9348GC-FXP* スイッチで *VXLAN* トラフィックに使用できる *ACL* オプション

シナリオ	ACL の方向	ACL タイプ	VTEP タイプ	ポート タイプ	フローの方向	トラフィック タイプ	サポート対象
1	入力	PACL	入力 VTEP	L2 ポート	ネットワークにアクセス [GROUP : encap direction]	ネイティブ L2 トラフィック [GROUP : inner]	YES
2		VACL	入力 VTEP	VLAN	ネットワークにアクセス [GROUP : encap direction]	ネイティブ L2 トラフィック [GROUP : inner]	YES

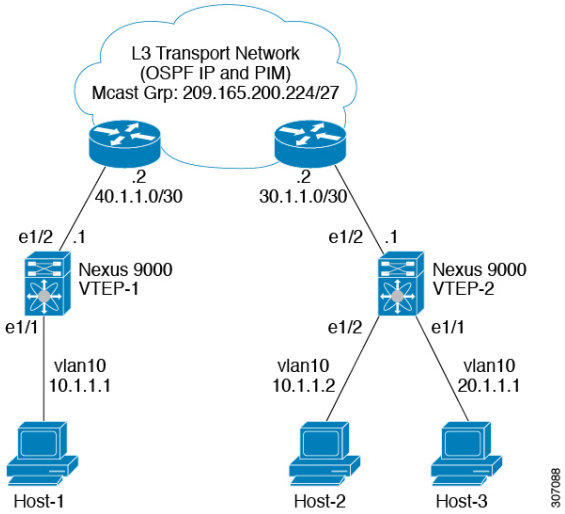
シナリオ	ACL の方向	ACL タイプ	VTEP タイプ	ポート タイプ	フローの方向	トラフィック タイプ	サポート対象
3	入力	RACL	入力 VTEP	テナント L3 SVI	ネットワークにアクセス [GROUP：encap direction]	ネイティブ L3 トラフィック [GROUP：inner]	YES
4	出力	RACL	入力 VTEP	アップリンク L3/L3-PO/SVI	ネットワークにアクセス [GROUP：encap direction]	VXLAN encap [GROUP：outer]	NO
5	入力	RACL	出力 VTEP	アップリンク L3/L3-PO/SVI	ネットワークにアクセス [GROUP：decap direction]	VXLAN encap [GROUP：outer]	NO
6	出力	PACL	出力 VTEP	L2 ポート	ネットワークにアクセス [GROUP：decap direction]	ネイティブ L2 トラフィック [GROUP：inner]	NO
7a		VACL	出力 VTEP	VLAN	ネットワークにアクセス [GROUP：decap direction]	ネイティブ L2 トラフィック [GROUP：inner]	YES
7b		VACL	出力 VTEP	宛先 VLAN	ネットワークにアクセス [GROUP：decap direction]	ネイティブ L3 トラフィック [GROUP：inner]	YES

シナリオ	ACL の方向	ACL タイプ	VTEP タイプ	ポート タイプ	フローの方向	トラフィック タイプ	サポート対象
8	出力	RACL	出力 VTEP	テナント L3 SVI	ネットワークにアクセス [GROUP：decap direction]	Post-decap L3 トラフィック [GROUP：inner]	YES

VXLAN の ACL 実装は、通常の IP トラフィックと同じです。ホスト トラフィックは、カプセル化スイッチで入力方向にカプセル化されません。ACL の分類は内部ペイロードに基づいているため、VXLAN カプセル化解除トラフィックでのカプセル化トラフィックの実装は少し異なります。VXLAN でサポートされている ACL のシナリオについては、次のトピックで説明します。また、カプセル化とカプセル化解除の両方のスイッチでサポートされていないケースについても説明します。

前の表に記載されているすべてのシナリオは、次のホストの詳細で説明されています。

図 *72 : VXLAN Encap* スイッチのポート *ACL*



- Host-1: 10.1.1.1/24 VLAN-10
- Host-2: 10.1.1.2/24 VLAN-10
- Host-3: 20.1.1.1/24 VLAN-20
- ケース1：VLAN-10 の Host-1 と Host-2 の間を流れるレイヤ 2 トラフィック/L2 VNI。
- ケース2：VLAN-10 および VLAN-20 上の Host-1 と Host-3 の間を流れるレイヤ 3 トラフィック/L3 VNI。

VXLAN ACL の注意事項と制約事項

VXLAN には、次の注意事項と制限事項があります。

- 着信 VLAN-10 およびアップリンク ポート（eth1/2）の SVI 上のルータ ACL（RACL）は、出力方向の外部または内部ヘッダーを持つカプセル化された VXLAN トラフィックのフィルタリングをサポートしません。この制限は、レイヤ 3 ポート チャネルアップリンク インターフェイスにも適用されます。
- SVI およびレイヤ 3 アップリンク ポートのルータ ACL（RACL）は、入力方向の外部または内部ヘッダーを持つカプセル化された VXLAN トラフィックをフィルタリングするためにサポートされていません。この制限は、レイヤ3ポートチャネルアップリンクインターフェイスにも適用されます。
- ポート ACL（PACL）は、ホストが接続されているレイヤ 2 ポートには適用できません。Cisco NX-OS は、出力方向の PACL をサポートしていません。

VXLANトンネル カプセル化 スイッチ

入力のアクセス ポートのポート ACL

カプセル化スイッチでホストが接続されているレイヤ2トランクまたはアクセスポートにポート ACL（PACL）を適用できます。ネットワークへのアクセスからの着信トラフィックは通常の IP トラフィックであるため。レイヤ 2 ポートに適用されている ACL は、非 VXLAN 環境の IP トラフィックと同様にフィルタリングできます。

ing-ifacl TCAM リージョンは、次のように分割する必要があります。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region ing-ifacl 256**
3. **ip access-list** *name*
4. *sequence-number* **permit ip** *source-address destination-address*
5. **exit**
6. **interface ethernet** *slot/port*
7. **ip port access-group** *pacl-name***in**
8. **switchport**
9. **switchport mode trunk**
10. **switchport trunk allowed vlan** *vlan-list*
11. **no shutdown**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**hardware access-list tcam region ing-ifacl 256** 例： `switch(config)# hardware access-list tcam region ing-ifacl 256`	**ing-ifacl** TCAM リージョンに UDF を接続します。これは IPv4 または IPv6 ポート ACL に適用されます。
ステップ 3	**ip access-list** *name* 例： `switch(config)# ip access list PACL_On_Host_Port`	IPv4 ACL を作成し、IP ACL コンフィギュレーション モードを開始します。name 引数は 64 文字以内で指定します。
ステップ 4	*sequence-number* **permit ip** *source-address destination-address* 例： `switch(config-acl)# 10 permit ip 10.1.1.1/32 10.1.1.2/32`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 5	**exit** 例： `switch(config-acl)# exit`	IP ACL 設定モードを終了します。
ステップ 6	**interface ethernet** *slot/port* 例： `switch(config)# interface ethernet1/1`	インターフェイス設定モードを開始します。
ステップ 7	**ip port access-group** *pacl-name***in** 例： `switch(config-if)# ip port access-group PACL_On_Host_Port in`	インターフェイスにレイヤ 2 PACL を適用します。ポート ACL では、インバウンド フィルタリングだけがサポートされています。1 つのインターフェイスに 1 つのポート ACL を適用できます。
ステップ 8	**switchport** 例： `switch(config-if)# switchport`	そのインターフェイスを、レイヤ 2 インターフェイスとして設定します。

	コマンドまたはアクション	目的
ステップ 9	**switchport mode trunk** 例： `switch(config-if)# switchport mode trunk`	インターフェイスをレイヤ 2 トランク ポートとして設定します。
ステップ 10	**switchport trunk allowed vlan** *vlan-list* 例： `switch(config-if)# switchport trunk allowed vlan 10,20`	トランク インターフェイスの許可 VLAN を設定します。デフォルトでは、トランク インターフェイス上のすべての VLAN（1 ～ 3967 および 4048 ～ 4094）が許可されます。VLAN 3968 ～ 4047 は、内部で使用するデフォルトで予約されている VLAN です。
ステップ 11	**no shutdown** 例： `switch(config-if)# no shutdown`	**shutdown** コマンドを無効にします。

サーバ VLAN の VLAN ACL

VLAN ACL（VACL）は、ホストが接続されている着信 VLAN-10 に適用できます。ネットワークへのアクセスからの着信トラフィックは通常の IP トラフィックであるため、VLAN-10 に適用されている ACL は、非 VXLAN 環境の IP トラフィックと同様にフィルタリングできます。VACL の詳細については、アクセス コントロール リストについて （715 ページ） を参照してください。

手順の概要

1. **configure terminal**
2. **ip access-list** *name*
3. *sequence-number* **permit ip** *source-address destination-address*
4. **vlan access-map** *map-name* **[***sequence-number***]**
5. **match ip address** *ip-access-list*
6. **action forward**
7. **vlan access-map** *name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル構成モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**ip access-list** *name* 例： `switch(config)# ip access list Vacl_On_Source_VLAN`	IPv4 ACL を作成し、IP ACL コンフィギュレーション モードを開始します。name 引数は 64 文字以内で指定します。
ステップ 3	*sequence-number* **permit ip** *source-address destination-address* 例： `switch(config-acl)# 10 permit ip 10.1.1.1 10.1.1.2`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 4	**vlan access-map** *map-name* **[***sequence-number***]** 例： `switch(config-acl)# vlan access-map Vacl_on_Source_Vlan 10`	指定した VLAN アクセス マップの VLAN アクセス マップ コンフィギュレーション モードを開始します。VLAN アクセス マップが存在しない場合は、デバイスによって作成されます。 シーケンス番号を指定しなかった場合、デバイスによって新しいエントリが作成され、このシーケンス番号はアクセス マップの最後のシーケンス番号よりも 10 大きい番号となります。
ステップ 5	**match ip address** *ip-access-list* 例： `switch(config-acl)# match ip address Vacl_on_Source_Vlan`	アクセス マップ エントリに ACL を指定します。
ステップ 6	**action forward** 例： `switch(config-acl)# action forward`	ACL に一致したトラフィックにデバイスが適用する処理を指定します。
ステップ 7	**vlan access-map** *name* 例： `switch(config-acl)# vlan access map Vacl_on_Source_Vlan`	指定した VLAN アクセス マップの VLAN アクセス マップ コンフィギュレーション モードを開始します。

入力の SVI のルーテッド ACL

入力方向のルータ ACL（RACL）は、カプセル化スイッチに接続するホストの着信 VLAN-10 の SVI に適用できます。ネットワークへのアクセスからの着信トラフィックは通常の IP トラフィックであるため、SVI 10 に適用されている ACL は、非 VXLAN 環境の IP トラフィックと同様にフィルタリングできます。

ing-racl TCAM リージョンは、次のように分割する必要があります。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region ing-ifacl 256**
3. **ip access-list** *name*
4. *sequence-number* **permit ip** *source-address destination-address*
5. **exit**
6. **interface ethernet** *slot/port*
7. **no shutdown**
8. **ip access-group** *pacl-name***in**
9. **vrf member** *vxlan-number*
10. **no ip redirects**
11. **ip address** *ip-address*
12. **no ipv6 redirects**
13. **fabric forwarding mode anycast-gateway**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**hardware access-list tcam region ing-ifacl 256** 例： `switch(config)# hardware access-list tcam region ing-ifacl 256`	**ing-racl** TCAM リージョンに UDF を接続します。これは IPv4 または IPv6 ポート ACL に適用されます。
ステップ 3	**ip access-list** *name* 例： `switch(config)# ip access list PACL_On_Host_Port`	IPv4 ACL を作成し、IP ACL コンフィギュレーション モードを開始します。name 引数は 64 文字以内で指定します。
ステップ 4	*sequence-number* **permit ip** *source-address destination-address* 例： `switch(config-acl)# 10 permit ip 10.1.1.1/32 10.1.1.2/32`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 5	**exit** 例：	IP ACL 設定モードを終了します。

	コマンドまたはアクション	目的
	`switch(config-acl)# exit`	
ステップ 6	**interface ethernet** *slot/port* 例： `switch(config)# interface ethernet1/1`	インターフェイス設定モードを開始します。
ステップ 7	**no shutdown** 例： `switch(config-if)# no shutdown`	**shutdown** コマンドを無効にします。
ステップ 8	**ip access-group** *pacl-name***in** 例： `switch(config-if)# ip port access-group Racl_On_Source_Vlan_SVI in`	インターフェイスにレイヤ 2 PACL を適用します。ポート ACL では、インバウンド フィルタリングだけがサポートされています。1 つのインターフェイスに 1 つのポート ACL を適用できます。
ステップ 9	**vrf member** *vxlan-number* 例： `switch(config-if)# vrf member Cust-A`	ホストの SVI を設定します。
ステップ 10	**no ip redirects** 例： `switch(config-if)# no ip redirects`	デバイスがリダイレクトを送信しないようにします。
ステップ 11	**ip address** *ip-address* 例： `switch(config-if)# ip address 10.1.1.10`	このインターフェイスの IP アドレスを設定します。
ステップ 12	**no ipv6 redirects** 例： `switch(config-if)# no ipv6 redirects`	ICMP のリダイレクト メッセージが BFD 対応インターフェイスでディセーブルであることを確認します。
ステップ 13	**fabric forwarding mode anycast-gateway** 例： `switch(config-if)# fabric forwarding mode anycast-gateway`	エニーキャスト ゲートウェイ転送モードを構成します。

出力のアップリンクのルーテッド ACL

着信 VLAN-10 の SVI およびアップリンク ポート（eth1/2）の RACL は、出力方向の外部または内部ヘッダーを持つカプセル化された VXLAN トラフィックをフィルタリングするためにサポートされていません。この制限は、レイヤ 3 ポート チャネル アップリンク インターフェイスにも適用されます。

VXLANトンネル カプセル化解除スイッチ

入力のアップリンクのルーテッド ACL

SVI およびレイヤ 3 アップリンク ポートの RACL は、入力方向の外部または内部ヘッダーを持つカプセル化されたVXLANトラフィックをフィルタリングするためにサポートされていません。この制限は、レイヤ 3 ポート チャネル アップリンク インターフェイスにも適用されます。

出力のアクセス ポートのポート ACL

ホストが接続されているレイヤ 2 ポートに PACL を適用しないでください。Cisco Nexus 9000 シリーズ スイッチは、出力方向の PACL をサポートしていません。

レイヤ 2 VNI トラフィックの VLAN ACL

レイヤ 2 VNI トラフィックが Host-1 から Host-2 に流れている場合、VLAN ACL（VACL）を VLAN-10 に適用して内部ヘッダーでフィルタリングできます。VACL の詳細については、アクセス コントロール リストについて （715 ページ） を参照してください。

VACL TCAM リージョンは、次のように分割する必要があります。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region vacl 256**
3. **ip access-list** *name*
4. **statistics per-entry**
5. *sequence-number* **permit ip** *source-address destination-address*
6. *sequence-number* **permit** *protocol source-address destination-address*
7. **exit**
8. **vlan access-map** *map-name* **[***sequence-number***]**
9. **match ip address** *list-name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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。

	コマンドまたはアクション	目的
ステップ 2	**hardware access-list tcam region vacl 256** 例： `switch(config)# hardware access-list tcam region vacl 256`	ACL TCAM リージョン サイズを変更します。
ステップ 3	**ip access-list** *name* 例： `switch(config)# ip access list VXLAN-L2-VNI`	IPv4 ACL を作成し、IP ACL コンフィギュレーション モードを開始します。name 引数は 64 文字以内で指定します。
ステップ 4	**statistics per-entry** 例： `switch(config-acl)# statistics per-entry`	その VACL のルールと一致するパケットのグローバル統計をデバイスが維持するように設定します。
ステップ 5	*sequence-number* **permit ip** *source-address destination-address* 例： `switch(config-acl)# 10 permit ip 10.1.1.1/32 10.1.1.2/32`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 6	*sequence-number* **permit** *protocol source-address destination-address* 例： `switch(config-acl)# 20 permit tcp 10.1.1.2/32 10.1.1.1/32`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 7	**exit** 例： `switch(config-acl)# exit`	ACL 設定モードを終了します。
ステップ 8	**vlan access-map** *map-name* **[***sequence-number***]** 例： `switch(config)# vlan access-map VXLAN-L2-VNI 10`	指定した VLAN アクセス マップの VLAN アクセス マップ コンフィギュレーション モードを開始します。VLAN アクセス マップが存在しない場合は、デバイスによって作成されます。 シーケンス番号を指定しなかった場合、デバイスによって新しいエントリが作成され、このシーケンス番号はアクセス マップの最後のシーケンス番号よりも 10 大きい番号となります。
ステップ 9	**match ip address** *list-name* 例： `switch(config-access-map)# match ip VXLAN-L2-VNI`	IP リスト名を設定します。

レイヤ3 VNIトラフィックのVLAN ACL

VLAN ACL（VACL）は、レイヤ3 VNIトラフィックがホスト1からホスト3に流れている場合に、内部ヘッダーでフィルタリングするために宛先VLAN 20に適用できます。これは、レイヤ3トラフィックのVACLがシステムの出力で考慮されるため、前のケースとは若干異なります。キーワード **output** は、レイヤ 3 VNI トラフィックの VACL エントリをダンプするときに使用する必要があります。VACL の詳細については、アクセス コントロール リストについて （715 ページ） を参照してください。

VACL TCAM リージョンは、次のようにカービングする必要があります。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region vacl 256**
3. **ip access-list** *name*
4. **statistics per-entry**
5. *sequence-number* **permit ip** *source-address destination-address*
6. *sequence-number* **permit** *protocol source-address destination-address*
7. **vlan access-map** *map-name* **[***sequence-number***]**
8. **action forward**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**hardware access-list tcam region vacl 256** 例： `switch(config)# hardware access-list tcam region vacl 256`	ACL TCAM リージョン サイズを変更します。
ステップ 3	**ip access-list** *name* 例： `switch(config)# ip access list VXLAN-L3-VNI`	IPv4 ACL を作成し、IP ACL コンフィギュレーション モードを開始します。name 引数は 64 文字以内で指定します。
ステップ 4	**statistics per-entry** 例： `switch(config)# statistics per-entry`	その VACL のルールと一致するパケットのグローバル統計をデバイスが維持するように設定します。

	コマンドまたはアクション	目的
ステップ 5	*sequence-number* **permit ip** *source-address destination-address* 例： `switch(config-acl)# 10 permit ip 10.1.1.1/32 20.1.1.1/32`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 6	*sequence-number* **permit** *protocol source-address destination-address* 例： `switch(config-acl)# 20 permit tcp 20.1.1.1/32 10.1.1.1/32`	特定の HTTP メソッドをサーバにリダイレクトするように ACL を設定します。
ステップ 7	**vlan access-map** *map-name* **[***sequence-number***]** 例： `switch(config-acl)# vlan access-map VXLAN-L3-VNI 10`	指定した VLAN アクセス マップの VLAN アクセス マップ コンフィギュレーション モードを開始します。VLAN アクセス マップが存在しない場合は、デバイスによって作成されます。 シーケンス番号を指定しなかった場合、デバイスによって新しいエントリが作成され、このシーケンス番号はアクセス マップの最後のシーケンス番号よりも 10 大きい番号となります。
ステップ 8	**action forward** 例： `switch(config-acl)# action forward`	ACL に一致したトラフィックにデバイスが適用する処理を指定します。

出力の SVI のルーテッド ACL

出力方向のルータ ACL（RACL）は、Host-3 がデキャップ スイッチで接続されている宛先 VLAN-20 の SVI に適用して、ネットワークからアクセスへのトラフィックフローの内部ヘッダーでフィルタリングできます。これは通常のカプセル化解除された IP トラフィック ポストです。SVI 20 に適用されている ACL は、非 VXLAN 環境内の IP トラフィックの場合と同様にフィルタリングできます。ACL の詳細については、アクセス コントロール リストについて（715 ページ） を参照してください。

egr-racl TCAM リージョンは、次のように切り分ける必要があります。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region egr-racl 256**
3. **ip access-list** *name*
4. *sequence-number* **permit ip** *source-address destination-address*

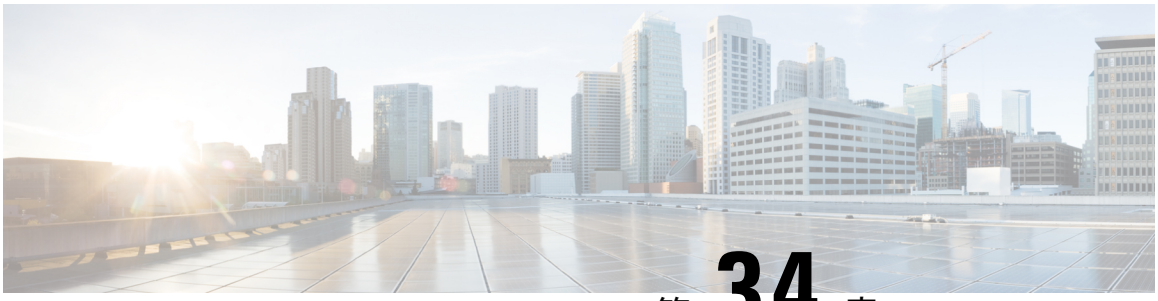
5. **interface vlan** *vlan-id*
6. **no shutdown**
7. **ip access-group** *access-list* **out**
8. **vrf member** *vxlan-number*
9. **no ip redirects**
10. **ip address** *ip-address/length*
11. **no ipv6 redirects**
12. **fabric forwarding mode anycast-gateway**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	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	**hardware access-list tcam region egr-racl 256** 例： `switch(config)# hardware access-list tcam region egr-racl 256`	ACL TCAM リージョン サイズを変更します。
ステップ 3	**ip access-list** *name* 例： `switch(config)# ip access-list Racl_on_Source_Vlan_SVI`	IPv4 ACL を作成し、IP ACL コンフィギュレーション モードを開始します。name 引数は 64 文字以内で指定します。
ステップ 4	*sequence-number* **permit ip** *source-address destination-address* 例： `switch(config-acl)# 10 permit ip 10.1.1.1/32 20.1.1.1/32`	条件に一致する IPv4 トラフィックを許可または拒否する、ACL のルールを作成します。 *source-address destination-address* 引数には、IP アドレスとネットワーク ワイルドカード、IP アドレスと可変長サブネット マスク、ホスト アドレス、または任意のアドレスを指定する **any** などがあります。
ステップ 5	**interface vlan** *vlan-id* 例： `switch(config-acl)# interface vlan vlan20`	インターフェイス コンフィギュレーション モードを開始します。*vlan-id* は、DHCP サーバ IP アドレスを設定する VLAN の ID です。
ステップ 6	**no shutdown** 例： `switch(config-if)# no shutdown`	shutdown コマンドを使用してください。

	コマンドまたはアクション	目的
ステップ 7	**ip access-group** *access-list* **out** 例： `switch(config-if)# ip access-group Racl_On_Detination_Vlan_SVI out`	IPv4 ACL または IPv6 ACL を、指定方向のトラフィックのレイヤ 3 インターフェイスに適用します。各方向にルータ ACL を 1 つ適用できます。
ステップ 8	**vrf member** *vxlan-number* 例： `switch(config-if)# vrf member Cust-A`	ホストの SVI を設定します。
ステップ 9	**no ip redirects** 例： `switch(config-if)# no ip redirects`	デバイスがリダイレクトを送信しないようにします。
ステップ 10	**ip address** *ip-address/length* 例： `switch(config-if)# ip address 20.1.1.10/24`	このインターフェイスの IP アドレスを設定します。
ステップ 11	**no ipv6 redirects** 例： `switch(config-if)# no ipv6 redirects`	ICMP のリダイレクト メッセージが BFD 対応インターフェイスでディセーブルであることを確認します。
ステップ 12	**fabric forwarding mode anycast-gateway** 例： `switch(config-if)# fabric forwarding mode anycast-gateway`	エニーキャスト ゲートウェイ転送モードを構成します。



第 34 章

PVLAN の設定

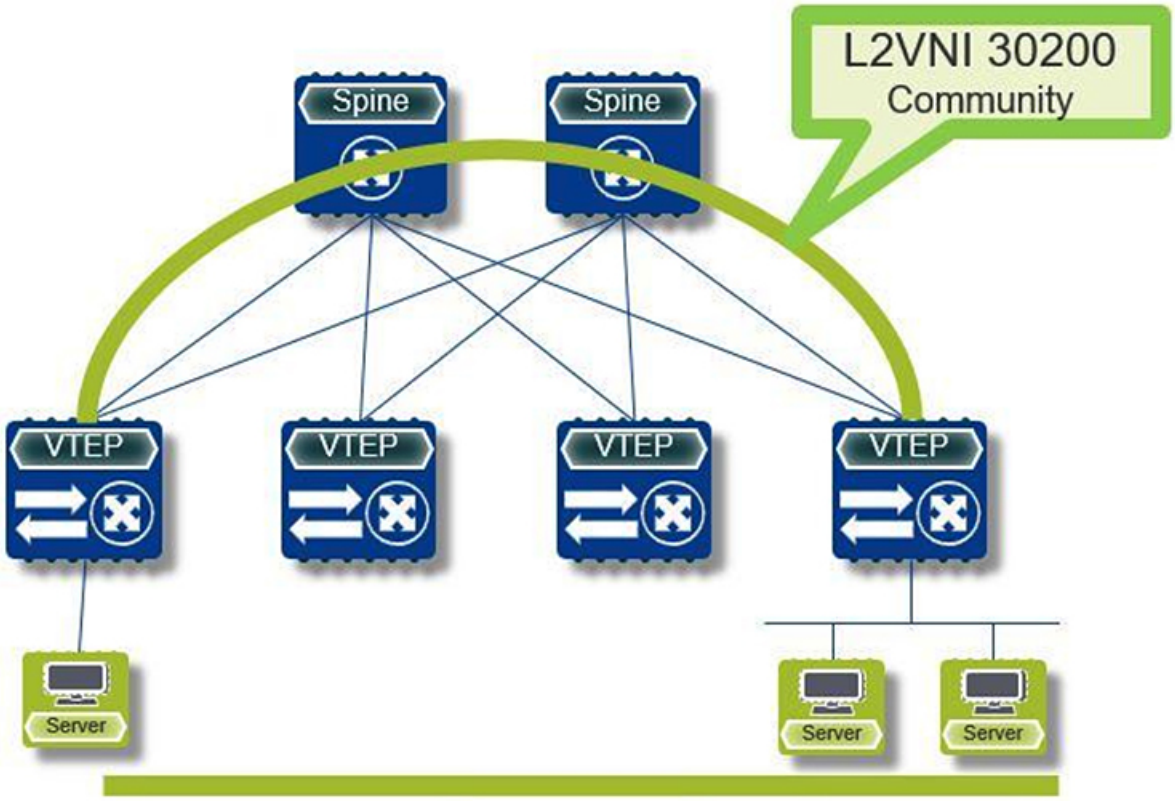
この章は、次の内容で構成されています。

VXLAN 上のプライベート VLAN について

プライベート VLAN の機能は、VLAN のレイヤ 2 ブロードキャスト ドメインをサブドメインに分割できます。サブドメインは、プライマリ VLAN とセカンダリ VLAN で構成されるプライベート VLAN のペアで表されます。プライベート VLAN ドメインには複数のプライベート VLAN のペアを設定でき、それぞれのペアを各サブドメインに割り当てることができます。プライベート VLAN ドメイン内のすべての VLAN ペアは、同じプライマリ VLAN を共有します。セカンダリ VLAN ID は、各サブドメインの区別に使用されます。

プライベート VLAN over VXLAN は、プライベート VLAN を VXLAN 全体に拡張します。セカンダリ VLAN は、VXLAN 上の複数の VTEP に存在できます。MAC アドレスの学習は、プライマリ VLAN 上で行われ、BGP EVPN を介してアドバタイズされます。トラフィックがカプセル化される場合、使用される VNI はセカンダリ VLAN の VNI です。この機能は、エニーキャスト ゲートウェイもサポートします。エニーキャスト ゲートウェイは、プライマリ VLAN を使用して定義する必要があります。

図 *73 : L2VNI 30200* コミュニティ



VXLAN にわたるプライベート VLAN に関する注意事項および制約事項

VXLAN にわたるプライベート VLAN に関する注意事項と制約事項は次のとおりです。

- 次のプラットフォームは、VXLAN 経由のプライベート VLAN をサポートします。
  - Cisco Nexus 9300-EX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX/FX2 プラットフォーム スイッチ
  - Cisco Nexus 9300-GX プラットフォーム スイッチ
- Cisco NX-OS リリース 9.3(9) 以降、vPC ピアリンク インターフェイスでは PVLAN 構成は許可されません。
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN 経由のプライベート VLAN は Cisco Nexus 9300-FX3/GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN 経由のプライベート VLAN は Cisco Nexus 9332D-H2R スイッチでサポートされます。

- Cisco NX-OS リリース 10.4(2)F 以降、VXLAN 経由のプライベート VLAN は Cisco Nexus 93400LD-H1 スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(3)F 以降、VXLAN 経由のプライベート VLAN はCisco Nexus 9364C-H1 スイッチでサポートされます。
- アンダーレイのフラッディングと学習はサポートされていません。
- ファブリック エクステンダ（FEX）VLAN は、プライベート VLAN にマッピングできません。
- vPC ファブリック ピアリングはプライベート VLAN をサポートします。
- Cisco NX-OS リリース 10.4(1)F 以降、プライベート VLAN は Cisco Nexus C9348GCFX3 および Cisco C9348GC-FX3PH でサポートされます。

プライベート VLAN の設定例

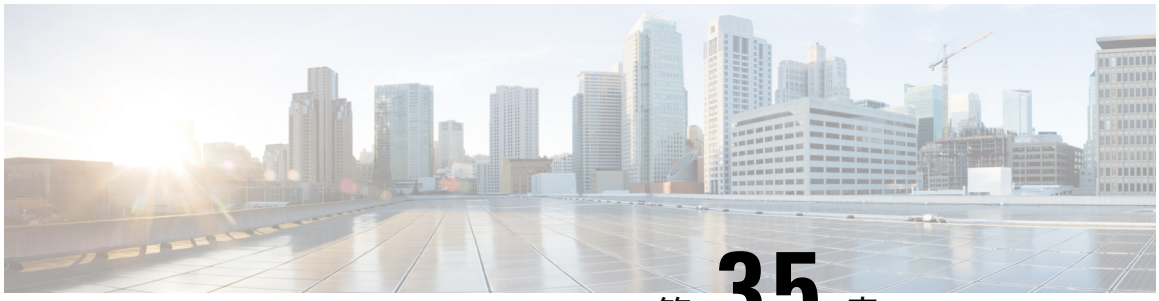
次に、プライベート VLAN の設定例を示します。

```
vlan 500
  private-vlan primary
  private-vlan association 501-503
  vn-segment 5000
vlan 501
  private-vlan isolated
  vn-segment 5001
vlan 502
  private-vlan community
  vn-segment 5002
vlan 503
  private-vlan community
  vn-segment 5003

vlan 1001
  !L3 VNI for tenant VRF
  vn-segment 900001

interface Vlan500
  no shutdown
  private-vlan mapping 501-503
  vrf member vxlan-900001
  no ip redirects
  ip address 50.1.1.1/8
  ipv6 address 50::1:1:1/64
  no ipv6 redirects
  fabric forwarding mode anycast-gateway

interface Vlan1001
  no shutdown
  vrf member vxlan-900001
  no ip redirects
  ip forward
  ipv6 forward
  ipv6 address use-link-local-only
  no ipv6 redirects

interface nve 1
```

```
  no shutdown
  host-reachability protocol bgp
  source-interface loopback0
  member vni 5000
    mcast-group 225.5.0.1
  member vni 5001
    mcast-group 225.5.0.2
  member vni 5002
    ingress-replication protocol bgp
  member vni 5003
    mcast-group 225.5.0.4
member vni 900001 associate-vrf
```

（注） 外部ゲートウェイを使用する場合は、外部ルータへのインターフェイスを PVLAN 無差別ポートとして設定する必要があります。

```
interface ethernet 2/1
switchport
switchport mode private-vlan trunk promiscuous
switchport private-vlan mapping trunk 500 199,200,201
exit
```



第 35 章

初期ホップ セキュリティの構成

この章は、次の内容で構成されています。

VXLAN BGP EVPN 中の DHCP スヌーピングの概要

初期ホップセキュリティ（FHS）は、アクセス（ホストがネットワーク内の最初のスイッチに接続する場所）でネットワークにセキュリティを提供するアクセス セキュリティ機能です。Dot1x、ポート セキュリティ、DHCP スヌーピングは、アクセス セキュリティ機能の例です。これらのセキュリティ機能が連携してホストを許可および認証し、正当なホストだけがネットワークを使用できるようにすることで、ネットワークを保護します。

現在、ダイナミック ARP 検査（DAI）および IP ソース ガード（IPSG）などの DHCP スヌーピングおよび関連する機能は、シングルスイッチに制限されています。Cisco NX-OS リリース 10.4(1)F 以降、これらの 3 機能のサポートは、Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 プラットフォーム スイッチや、9700-EX/FX/GX ライン カードを搭載した Cisco Nexus 9500 スイッチで VXLAN ファブリック全体に拡張されます。

Cisco NX-OS リリース 10.4(2)F 以降、初期ホップ セキュリティ機能は Cisco Nexus 9332D-H2R、および 93400LD-H1 スイッチでサポートされます。

Cisco NX-OS リリース 10.4(2)F 以降、初期ホップ セキュリティ機能は Cisco Nexus 9364C-H1 スイッチでサポートされます。

VXLAN トポロジでの DHCP スヌーピング

VXLAN ファブリックでは、ホストを 1 つの VTEP のインターフェイスに接続し、DHCP サーバーを別の VTEP のインターフェイスに接続できます。

図に示すように、ホスト H1 は VTEP1 に接続され、DHCP サーバは VTEP3 に接続されます。

ホストと DHCP サーバーは、このホスト IP 割り当て手順の一部として一連のメッセージを交換します。これらは、一般に Discover-Offer-Request-Ack（DORA）交換メッセージとして知られています。

特定のホスト（H1）の DORA 交換は、リモート DHCP サーバー（VTEP3）に到達するために VXLAN ファブリックを介して送信する必要があります。

VTEP3 は、「Offer」および「Ack」メッセージ（DORA シーケンスの一部）と、それらが DHCP サーバーから来ていること、そして VTEP3 の信頼できるインターフェイスで受信されたことを確認します。

DORA 交換が完了すると、VTEP1 は「DHCP スヌーピング DB」エントリを作成します。この DB には、ホストの MAC アドレス、DHCP サーバーによってホストに割り当てられた IP アドレス、VLAN、および「リース時間」などのその他の詳細が含まれています。この機能の主な仕組みは、「ローカル スヌーピング DB エントリ」としてホスト（H1）の VTEP1 で作成されたスヌーピング DB エントリが、BGP-EVPN を使用してリモート VTEP にも伝播され、ホスト（H1）からの「リモート スヌーピング DB エントリ」と見なされることです。したがって、この DHCP スヌーピング DB は VTEP 全体で「分散 DB」と見なされ、スヌーピング エントリはすべての VTEP と同期されます。

ホストへの IP アドレス割り当てが事前に定義されているユース ケースでは、**ip source binding** *ip address* **vlan** *vlan-id* **interface** *interface* コマンドを使用してスヌーピング DB エントリを構成できます。このコマンドを使用して追加されたスヌーピング エントリは、スタティック エントリと呼ばれ、これらもすべての VTEP に分散されます。

分散 DHCP スヌーピング DB は次のように使用されます。

- DAI を使用してホストから送信された ARP/GARP を検証します。これにより、異なるホスト クレデンシャルを使用した ARP/GARP のスプーフィング、そしてその後のネットワーク内での悪意のある ARP ストームが防止されます。

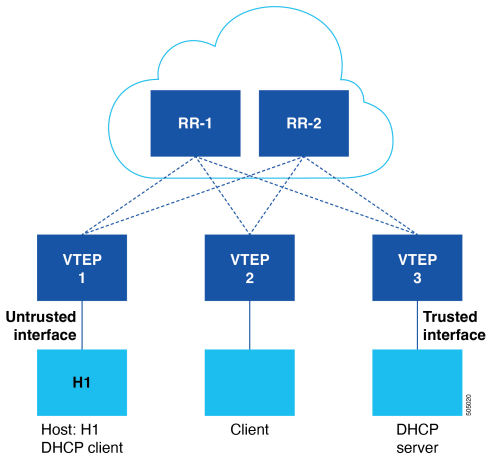
  VXLAN 環境では、host-move を考慮する必要があります。DHCP スヌーピング DB はファブリック全体に複製されるため、DAI は host-move の後もファブリック全体で動作できるようになりました。したがって、コントロールプレーンは VXLAN 環境で保護されます。

（注） DB に一致するエントリがない場合、ARP/GARP はドロップされます。

- IPSG を使用してホストからのデータプレーン トラフィックを検証します。これにより、データ トラフィックが検証され、悪意のあるホストがネットワークにデータ トラフィックを送信するのを防ぐことができます。

  DHCP スヌーピング エントリは、ファブリック全体に複製されます。その VTEP のローカル DHCP クライアントのみが IPSG でプログラムされます。ローカル DHCP クライアントは、DHCP スヌーピング テーブルでアンカー フラグが true に設定されて識別されます。ホストが別の VTEP に移動して安定した場合、IPSG は新しい VTEP の背後にあるクライアントを再プログラムして、データトラフィックを検証する必要があります。古い VTEP では、IPSG はこの DHCP クライアントを削除する必要があります。アンカー フラグはそれに応じて変更されます。ホストの移動は、ホストが移動した新しい VTEP で受信されたホストからの ARP 要求の受信によってトリガーされます。

図 *74 : VXLAN* での *DHCP* スヌーピング



VXLAN 上の DHCP スヌーピングの注意事項および制約事項

VXLAN 機能での DHCP スヌーピングには、次の構成の注意事項および制約事項があります。

- Cisco NX-OS リリース 10.4(1)F 以降では、DHCP スヌーピングと、ダイナミック ARP 検査（DAI）や IP ソース ガード（IPSG）のサポートなどの関連機能が、Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 プラットフォーム スイッチおよび 9700-EX/FX/GX ライン カードを使用する Cisco Nexus 9500 スイッチの VXLAN ファブリックに拡張されています。

  Cisco NX-OS リリース 10.4(2)F 以降、初期ホップ セキュリティ機能は Cisco Nexus 9332D-H2R、および 93400LD-H1 スイッチでサポートされます。

  Cisco NX-OS リリース 10.4(2)F 以降、初期ホップ セキュリティ機能は Cisco Nexus 9364C-H1 スイッチでサポートされます。

- DHCP スヌーピング、DAI、および IPSG がすべての VTEP で同時に有効になっていることを確認します。

（注） DAI と IPSG は DHCP スヌーピングに依存します。DHCP スヌーピングはスヌーピング DB を作成し、この DB は DAI と IPSG によって使用されます。

- IPv4 マルチキャスト アンダーレイのみがサポートされています。ただし、IPv4 入力レプリケーション アンダーレイ、IPv6 入力レプリケーション アンダーレイ、および IPv6 マルチキャスト アンダーレイはサポートされていません。
- IPv4 DHCP ホストのみがサポートされます。
- ホスト移動は、ARP/GARP/RARP 受信によって示されます。RARP（MAC 情報のみを含む）の場合、VTEP は MAC に対して学習した IP の ARP 更新を開始します。したがって、基本的に ARP-GARP はホスト移動のトリガであり、他のデータ パケットではありません。
- vPC VTEP の場合、物理 MCT のみがサポートされます。
- この機能は、FabricPath から VXLAN への移行機能およびカウンタ ACL（CNT ACL）機能と共存できません。
- 入力 SUP リージョンでは、 **hardware access-list tcam region ing-sup** コマンドを使用して入力 ACL を設定するには、TCAM をデフォルトの 512 エントリではなく 768 エントリにカービングする必要があります。TCAM カービングの変更を反映するには、スイッチのリロードが必要です。
- マルチサイトで vPC BGW を使用する場合、vPC BGW で DHCP スヌーピングが有効になっている場合は、DHCP クライアントと DHCP サーバが同じサイトにあることを確認します。

（注）
- DHCP スヌーピングは、DHCP サービスを使用する必要がある DHCP ホストに属する VLAN に対して（VTEP で）有効にする必要があります。
- ファブリック内の DHCP サーバがサービスを提供するすべての VLAN は、ファブリックのすべての VTEP で DHCP スヌーピングを有効にする必要があります。

DHCP スヌーピングの前提条件

DHCP の前提条件は、次のとおりです。

- DHCP スヌーピングまたは DHCP リレー エージェントを設定するためには、DHCP についての知識が必要です。
- DHCP スヌーピング、DAI、および IPSG 機能がリーフ VTEP で同時に有効になっていることを確認します。

VXLAN での DHCP スヌーピングの有効化

シングルボックス機能で DHCP スヌーピングを有効または無効にすることも、ファブリック全体の VLAN に対してこの機能を有効にすることもできます。デフォルトでは、DHCP スヌーピングはすべての VLAN でディセーブルになります。

始める前に

- DHCP 機能がイネーブルにされていることを確認します。
- **nv overlay evpn** コマンドが構成されていることを確認します。
- DHCP スヌーピング、DAI、および IPSG 機能が有効になっていることを確認します。詳細については、DHCP スヌーピングの前提条件 （738 ページ） セクションを参照してください。
- DHCP スヌーピングと DAI がすべての VXLAN ノードで有効になっていることを確認します。構成の詳細については『Cisco Nexus 9000 シリーズ NX-OS セキュリティ構成ガイド』の「**DHCP スヌーピングの構成**」 を参照してください。
- DHCP サーバー ノードに接続されているインターフェイスで、DHCP スヌーピングの信頼と ARP インスペクションの信頼が有効になっていることを確認します。構成の詳細については『Cisco Nexus 9000 シリーズ NX-OS セキュリティ構成ガイド』の「**DHCP スヌーピングの構成**」 を参照してください。
- DHCP クライアント ノードに接続されているインターフェイスで IP ソース ガードが有効になっていることを確認します。構成の詳細については『Cisco Nexus 9000 シリーズ NX-OS セキュリティ構成ガイド』の「**DHCP スヌーピングの構成**」 を参照してください。

手順の概要

1. **configure terminal**
2. [**no**] **ip dhcp snooping vlan** *vlan-list* **evpn**
3. （任意） **show running-config dhcp**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	[**no**] **ip dhcp snooping vlan** *vlan-list* **evpn** 例： `switch(config)# ip dhcp snooping vlan 100,200,250-252 evpn`	*vlan-list* で指定する VLAN の DHCP スヌーピングをイネーブルにします。 Cisco NX-OS リリース 10.4(1)F 以降では、同じ VTEP または他の VTEP 上の他のインターフェイスへのホストの移動をサポートするための **evpn** オプションが提供されています。 （注） • **evpn** オプションを使用してこの機能を有効にすると、**nve** は信頼できるインターフェイスとして暗黙的に追加されます。 • **evpn** キーワードを指定した vlan-list-1 と、evpn キーワードを指定しない vlan-list-2 を使用できます。 このコマンドの **no** 形式を使用すると、指定した VLAN の DHCP スヌーピングがディセーブルになります。
ステップ 3	（任意） **show running-config dhcp** 例： `switch(config)# show running-config dhcp`	DHCP 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

永続的な凍結後の重複ホストのクリア

FHS 対応 VTEP の DHCP クライアントのモビリティおよび重複検出ロジックは、BGP EVPN モビリティおよび重複検出ロジックと同じです。ただし、非 FHS 展開のいずれかの VTEP で

重複検出が発生する可能性があります。FHS 展開では、DHCP バインディング エントリがリモートである VTEP でホストの重複が常に検出されます。

モビリティと重複検出の詳細については、「IP アドレスと MAC アドレスの重複データ検出（156 ページ）」セクションを参照してください。

MAC または MAC-IP が永続的に凍結された場合に、モビリティまたは重複チェックシーケンスを再開する自動回復メカニズムはありません。MAC および MAC-IP の永続的な凍結状態をクリアするには、次のコマンドを使用します。

- MAC の場合：

  clear l2route evpn mac [*mac-address*] [**topo**] **permanently-frozen-list**

- MAC-IP の場合：

  clear fabric forwarding dup-host [{ **ip**|**ipv6** *address* }] [**vrf** {*vrf-name* | *vrf-known-name* | **all**}]

DHCP スヌーピング バインディングの確認

DHCP スヌーピング バインディング情報を表示するには、次のコマンドを入力します。

コマンド	目的
show ip dhcp snooping binding evpn	DHCP スヌーピング バインディング データベースからすべてのエントリを表示します。
show l2route fhs [**topology** *topology id* | **all**]	L2RIB データベースのすべてのエントリを表示します。

次の例は、**show ip dhcp snooping binding evpn** コマンドのサンプル出力を示しています。

```
switch(config)# show ip dhcp snooping binding evpn
MacAddress          IpAddress        Lease(Sec)  Type      BD    Interface       anchor
     Freeze
----------------- -------------- ----------- ---------- ---- ----------------- ----------
 ---------
00:10:00:10:00:10  10.10.10.10      infinite  static      2001  Ethernet1/48         YES
       NONE
00:15:06:00:00:01  100.1.150.156    86282     dhcp-snoop  2001  Ethernet1/31         YES
       NONE
00:17:06:00:00:01  100.1.150.155    86265     dhcp-snoop  2001  nve1(peer-id: 1)     NO
        NONE
```

次の例は、**show l2route fhs** コマンドのサンプル出力を示しています。

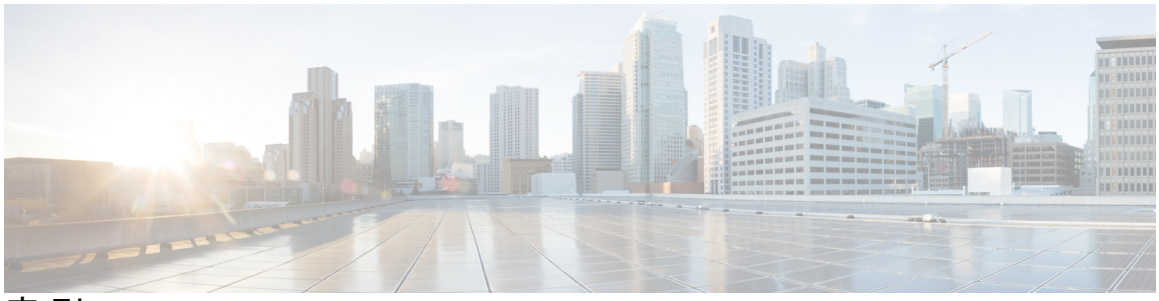
```
switch(config)# show l2route fhs all
Flags - (Stt):Static (Dyn):Dynamic (R):Remote
Topo ID  Mac Address     Host IP            Prod          Flags      Seq No     Next-Hops
---- -- ---- ----------- ------------------ ---------- ----------- ------------
-------------
2001     0015.0600.0001  100.1.150.156      DHCP_DYNAMIC  Dyn,        0          Eth1/31
2001     0017.0600.0001  100.1.150.155      BGP           Dyn,R,      0          1.13.13.13
 (Label: 0)
switch(config)#
```

次の例は、DHCP クライアントを使用した VTEP の DHCP 構成を示しています。

```
feature dhcp
service dhcp
ip dhcp snooping
ip dhcp snooping vlan 2001-2002 evpn
ip arp inspection vlan 2001-2002

interface Ethernet1/31
ip verify source dhcp-snooping-vlan
```

次の例は、DHCP サーバーを使用した VTEP の DHCP 構成を示しています。

```
feature dhcp
service dhcp
ip dhcp snooping
ip dhcp snooping vlan 2001-2002 evpn
ip arp inspection vlan 2001-2002

interface Ethernet1/47
ip dhcp snooping trust
ip arp inspection trust
```



索 引

記号

A

C

E

F

H

I

K

M

N

P

Q

R

S

T

U

V

W

い

き

く

ね

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。