



## VXLAN の実装

このモジュールでは、一般的な VXLAN の概念情報と、Cisco ASR 9000 シリーズ ルータでのレイヤ 2 VXLAN の設定情報を示します。レイヤ 3 VXLAN の設定情報については、『Cisco ASR 9000 Series Aggregation Services Router MPLS Layer 3 VPN Configuration Guide』の「Implementing L3 VXLAN」の章を参照してください。VXLAN は、VLAN の場合と同じイーサネットレイヤ 2 ネットワークサービスを提供しますが、より優れた拡張性と柔軟性を備えています。

表 1: VXLAN の機能の履歴

リリース	変更内容
リリース 5.2.0	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 5.3.1	VXLAN エニーキャストゲートウェイ機能が導入されました。
リリース 6.1.2	次の機能が追加されました。 <ul style="list-style-type: none"><li>• EVPN VXLAN レイヤ 2 Data Center Interconnect ゲートウェイ</li><li>• EVPN ESI マルチパス</li></ul>

- [VXLAN の実装の前提条件 \(2 ページ\)](#)
- [VXLAN の実装に関する情報 \(2 ページ\)](#)
- [レイヤ 2 VXLAN ゲートウェイの設定 \(5 ページ\)](#)
- [レイヤ 2 VXLAN ゲートウェイの実装の設定例 \(11 ページ\)](#)
- [EVPN VXLAN レイヤ 2 Data Center Interconnect ゲートウェイ \(13 ページ\)](#)
- [EVPN VXLAN レイヤ 2 Data Center Interconnect ゲートウェイの設定 \(15 ページ\)](#)
- 例：エニーキャスト VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの設定 (30 ページ)
- 例：一意の VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの設定 (31 ページ)

## VXLANの実装の前提条件

この前提条件は、VXLANの実装に適用されます。

適切なタスク ID を含むタスク グループに関連付けられているユーザグループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれません。

ユーザグループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

## VXLANの実装に関する情報

VXLAN を実装するには、次の概念を理解している必要があります。

### VXLAN

VXLAN は、VLAN の場合と同じイーサネットレイヤ 2 ネットワークサービスを提供しますが、より優れた拡張性と柔軟性を備えています。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、コア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。コアネットワークでの転送プロトコルは IP と UDP です。VLAN と比較して、VXLAN には次の利点があります。

- データセンター全体にマルチテナントセグメントを柔軟に配置します。テナントのワークロードが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ソリューションを提供します。
- より多くのレイヤ 2 セグメントをアドレス指定するための拡張性が高くなります。VLAN は 12 ビットの VLAN ID を使用してレイヤ 2 セグメントをアドレス指定します。このため、拡張性は制限され VLAN の数は最大 4094 個になります。VXLAN は、VXLAN ネットワーク識別子 (VNID) と呼ばれる 24 ビットのセグメント ID を使用します。これにより、最大 1600 万の VXLAN セグメントを同じ管理ドメインに共存させることができます。
- 基盤となるインフラストラクチャで使用可能なネットワークパスの使用率が向上します。VLAN はループ防止のためにスパニングツリープロトコルを使用します。このため、冗長パスをブロックすることによってネットワーク内の半数のネットワークリンクを使用しません。一方、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経由で転送されます。VXLAN では、レイヤ 3 ルーティング、Equal Cost Multipath (ECMP; 等コストマルチパス) ルーティング、およびリンク アグリゲーションプロトコルを活用して、すべての利用可能なパスを使用できます。

## VXLAN エニーキャストゲートウェイ

VXLAN エニーキャストゲートウェイ機能は、エニーキャスト機能を VXLAN に拡張します。これにより、アンダーレイ マルチキャスト ロードバランシングおよび冗長性のためにネットワーク上でエニーキャストルーティングを使用できるようになります。

VXLAN エニーキャスト ソリューションは次のとおりです。

- 完全なアクティブ-アクティブファーストホップゲートウェイを許可します（フロー単位でアクティブ-アクティブ）。
- 新しいコントロールプレーンプロトコルや管理プレーンプロトコル、またはどのような形式の外部 SDN コントローラや NMS もゲートウェイの調整や同期を行いません。

エニーキャストゲートウェイ機能は、次の基本的な概念に従います。

- 複数の VXLAN ゲートウェイ間で仮想レイヤ 3 ゲートウェイと仮想 VTEP を作成する。これらのゲートウェイは、オーバーレイ IP アドレス、オーバーレイ MAC アドレス、およびアンダーレイ VTEP IP アドレスと同じ設定を使用します。
- 特定のタイプのオーバーレイ制御パケットのデータプレーンミラーとして使用する、ゲートウェイ間のプライベート マルチキャスト グループを作成する。



(注) VXLAN エニーキャストゲートウェイ機能は、Cisco ASR 9000 高密度 100GE イーサネットラインカードのみでサポートされます。

### 推奨事項

VXLAN エニーキャストゲートウェイ機能を設定する前に、次の推奨事項を考慮する必要があります。

- BGP は、データセンター内の VXLAN エニーキャスト機能とは連動しません。
- IGP は、データセンター内のアンダーレイネットワークで動作します。
- BGP および IGP は、WAN 側で使用する必要があります。
- データセンターのトップオブラック (TOR) スイッチは、ルータのカスタマー IP とエニーキャストゲートウェイ間のスタティックルートを使用します。

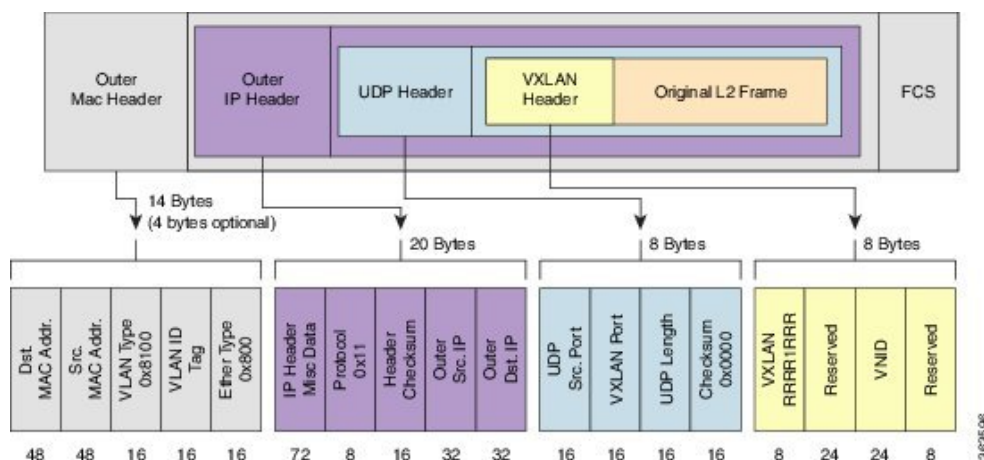
### VxLAN エニーキャストゲートウェイを展開するための要件

マルチキャストグループは制御フレームのミラーリングに使用されるため、IPv6 ネイバーアドバタイズメントの場合は、2つのルータ（またはインターフェイス）間で同じアドレスが検出されることにより、重複アドレス検出 (DAD) プロトコルがサービスをダウンさせます。したがって、BVI インターフェイスで IPv6 DAD を無効にし、不要ノード検出 (ND) 応答を有効にする必要があります。

## VXLAN のパケット形式

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 のパケット形式を次の図に示します。

図 1: VXLAN のパケット形式



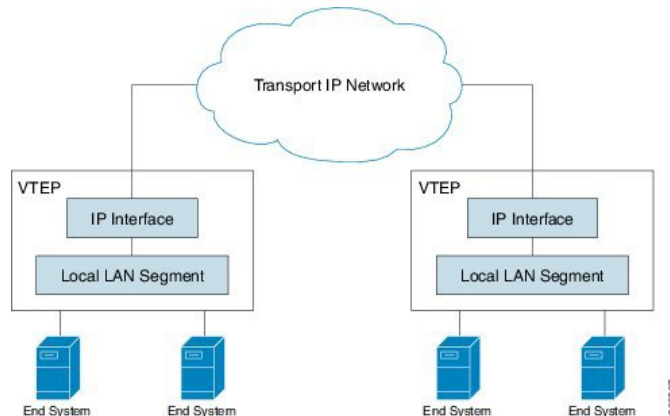
上図に示すように、VXLAN は 24 ビット VNID といくつかの予約ビットで構成される 8 バイト VXLAN ヘッダーを導入します。VXLAN ヘッダーおよび元のイーサネットフレームは、UDP ペイロードに入ります。24 ビット VNID は、レイヤ 2 セグメントを識別し、セグメント間でレイヤ 2 の分離を維持するために使用されます。VNID のすべての 24 ビットを使用して、VXLAN は約 1600 万個の LAN セグメントをサポートできます。

## VXLAN トンネル エンドポイント

VXLAN は VXLAN トンネルエンドポイント (VTEP) デバイスを使用してテナントのエンドデバイスを VXLAN セグメントへマッピングし、VXLAN のカプセル化およびカプセル解除を実行します。各 VTEP 機能には 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てローカルエンドポイントの通信をサポートするためのローカル LAN セグメント上のスイッチインターフェイスで、もう 1 つは、転送 IP ネットワークのための IP インターフェイスです。

IP インターフェイスには一意の IP アドレスがあります。これは、インフラストラクチャ VLAN として知られる、転送 IP ネットワーク上の VTEP を識別します。VTEP デバイスはこの IP アドレスを使用してイーサネットフレームをカプセル化し、カプセル化されたパケットを、IP インターフェイスを介して転送ネットワークへ送信します。また、VTEP デバイスはリモート VTEP で VXLAN セグメントを検出し、IP インターフェイスを介してリモートの MAC Address-to-VTEP マッピングについて学習します。次の図に、VTEP の機能コンポーネントとトランスポート IP ネットワークを介したレイヤ 2 接続用に作成された論理トポロジを示します。

図 2: VTEP



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

## レイヤ 2 VXLAN ゲートウェイの設定

レイヤ 2 VXLAN ゲートウェイは、同じレイヤ 2 ネットワーク内の VXLAN セグメントと非 VXLAN セグメント (VLAN や VPLS など) の間のトラフィックをブリッジします。VXLAN レイヤ 2 ゲートウェイの動作は、データプレーン MAC アドレスラーニングと、IP マルチキャストによるマルチデスティネーショントラフィックのフラッドイング (未知のユニキャスト、マルチキャスト、ブロードキャストフレームなど) に基づいています。次のセクションでは、ASR 9000 シリーズルータを、同じ L2 ドメイン内の VLAN および VXLAN セグメント間のレイヤ 2 VXLAN ゲートウェイとして設定する方法を示します。

### 前提条件

VXLAN レイヤ 2 ゲートウェイとして Cisco ASR 9000 シリーズルータを設定するための前提条件を、次に示します。

- ループバックインターフェイスを設定します。これは、ローカル VTEP の送信元インターフェイスとして機能します。
- リモート VTEP へのユニキャストの到達可能性を設定します。
- Bidirectional Protocol Independent Multicast (Bidir PIM) または PIM スパースモードを設定します。詳細については、*Multicast Configuration Guide for Cisco ASR 9000 Series Routers* を参照してください。

## 機能制限

VXLAN の設定時には、次の制限事項を考慮してください。

- VXLAN は、オーバーレイトランスポート仮想化 (OTV) および VXLAN UDP ポートでのみ設定します。
- ループバック インターフェイスのみを送信元インターフェイスにできます。
- 複数の NVE インターフェイス間で VNI、マルチキャストグループ、またはソースインターフェイスを共有することはできません。
- VNI 範囲とマルチキャスト範囲は、どちらも連続した範囲のみを指定できます。カンマ区切り値を使用した連続していない範囲はサポートされていません。
- マルチキャストグループへの VNI のマッピングには、1:1 または N:1 のいずれかを使用できます。次に例を示します。
  - 「member vni 5000 mcast-group 239.1.1.1」 コマンドは、有効な 1:1 のマッピングを設定します。
  - 「member vni 5000-5005 mcast-group 239.1.1.1」 コマンドは、有効な N:1 のマッピングを設定します。
- VNI が VNI 範囲の一部として設定されている場合は、同じ範囲の一部としてのみ変更または削除できます。たとえば、「member vni 5000-5002 mcast-group 239.1.1.1」 コマンドが設定されている場合は、「no member vni 5001」 コマンドを使用して NVE インターフェイスから VNI 5001 の関連付けのみを解除することはできません。
- スタティック MAC 設定はサポートされていません。
- システムごとに最大 128k のレイヤ 2 およびレイヤ 3 サブインターフェイスを設定できます。この設定には、レイヤ 2 サブインターフェイスとレイヤ 3 サブインターフェイス両方の組み合わせを使用できます。または、すべてレイヤ 2 サブインターフェイスにすることや、すべてレイヤ 3 サブインターフェイスにすることもできます。

システムでは、システムごとに 128k を超えるサブインターフェイスを設定できますが、サービスにこの設定を使用することはできません。128k のサブインターフェイスのしきい値に達すると、システムにより警告メッセージが表示されますが、設定は引き続き適用されます。ただし、サービスにこの設定を使用することはできません。

## ネットワーク仮想化エンドポイント (NVE) インターフェイスの作成と設定

NVE インターフェイスを作成し、VXLAN の VXLAN トンネルエンドポイント (VTEP) として設定するには、次の作業を実行します。

## 手順の概要

1. **interface nve** *nve-identifier*
2. (オプション) **overlay-encapsulation vxlan**
3. **source-interface loopback** *loopback-interface-identifier*
4. **member vni vni\_number [ -end\_vni\_range ] mcast-group ip\_address [ end\_ip\_address\_range ]**
5. (オプション) **anycast source-interface loopback loopback-interface-identifier sync-group ip\_address**
6. **commit** コマンドまたは **end** コマンドを使用します。

## 手順の詳細

ステップ 1 **interface nve** *nve-identifier*

例 :

```
RP/0/RSP0/cpu 0: router(config)# interface nve 1
```

NVE インターフェイスを作成し、NVE インターフェイス設定サブモードを開始します。

ステップ 2 (オプション) **overlay-encapsulation vxlan**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# overlay-encapsulation vxlan
```

NVE インターフェイスの VXLAN カプセル化を設定します。VXLAN は、NVE インターフェイスのデフォルトのカプセル化です。この手順は、カプセル化を変更していない場合はオプションです。

ステップ 3 **source-interface loopback** *loopback-interface-identifier*

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# 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 **member vni vni\_number [ -end\_vni\_range ] mcast-group ip\_address [ end\_ip\_address\_range ]**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# member vni 1-10 mcast-group 224.2.2.2 224.2.2.10
```

VXLAN ネットワーク識別子 (VNI) を使用して単一の VXLAN または連続する範囲の VXLAN を NVE インターフェイスに関連付け、この VNI に関連付けられるマルチキャストアドレスまたは連続するマルチキャストアドレス範囲を指定します。

- (注)
- VNI とマルチキャストグループとのマッピングは、1 対 1 または多対 1 です。
  - 不連続な VXLAN または VXLAN 範囲を NVE インターフェイスに関連付けるには、VXLAN または VXLAN 範囲ごとに次の手順を実行します。たとえば、

```
RP/0/RSP0/cpu 0: router(config-if)# member vni 10 mcast-group 224.2.2.10
RP/0/RSP0/cpu 0: router(config-if)# member vni 23 mcast-group 224.2.2.23
RP/0/RSP0/cpu 0: router(config-if)# member vni 50-59 mcast-group 224.2.2.50 224.2.2.59
RP/0/RSP0/cpu 0: router(config-if)# member vni 100-120 mcast-group 224.2.2.100 224.2.2.120
```

**ステップ5** (オプション) **anycast source-interface loopback loopback-interface-identifier sync-group ip\_address**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# anycast source-interface loopback 1 sync-group 192.23.2.20
```

この VTEP のエニーキャスト モード パラメータを設定します。

**ステップ6** **commit** コマンドまたは **end** コマンドを使用します。

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

**次のタスク**

設定された NVE インターフェイス情報を表示するには、**show nve interface** コマンドを使用します。

## レイヤ2サブインターフェイスの作成と設定

VLAN セグメントに関連付けられたレイヤ2サブインターフェイスを作成するには、次の作業を実行します。

**手順の概要**

1. **interface gigabitEthernet interface-identifier l2transport**
2. **dot1q vlan vlan-identifier**
3. **commit** コマンドまたは **end** コマンドを使用します。

**手順の詳細****ステップ1** **interface gigabitEthernet interface-identifier l2transport**

例 :

```
RP/0/RSP0/cpu 0: router(config)# interface gigabitEthernet 0/0/0/0.100 l2transport
```

レイヤ2サブインターフェイスを作成し、サブインターフェイス設定モードを開始します。

**ステップ2** **dot1q vlan vlan-identifier**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# dot1q vlan 100
```

インターフェイスの VLAN を設定します。

**ステップ3** **commit** コマンドまたは **end** コマンドを使用します。



**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

---

## VLAN および VXLAN のブリッジドメインへの関連付け

VLAN および VXLAN をブリッジドメインに関連付けるには、次の作業を実行します。

### 手順の概要

1. **l2vpn**
2. **bridge group** *bridge-group-name*
3. **bridge-domain** *bridge-domain-name*
4. **member vni** *vlan-identifier*
5. **interface gigabitEthernet** *sub-interface-identifier*
6. **commit** コマンドまたは **end** コマンドを使用します。

### 手順の詳細

---

#### ステップ 1 **l2vpn**

例 :

```
RP/0/RSP0/cpu 0: router(config)# l2vpn
```

l2vpn 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 **bridge group** *bridge-group-name*

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn)# bridge group bridgegroup1
```

ブリッジグループ設定モードを開始します。

#### ステップ 3 **bridge-domain** *bridge-domain-name*

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg)# bridge-domain bdomain1
```

ブリッジドメイン設定モードを開始します。

#### ステップ 4 **member vni** *vlan-identifier*

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg-bd)# member vni 100
```

VXLAN をブリッジドメインに関連付けます。

### ステップ5 interface gigabitEthernet sub-interface-identifier

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg-bd)# interface gigabitEthernet 0/0/0/0.200
```

VLAN サブインターフェイスを使用して、VLAN をブリッジドメインに関連付けます。

### ステップ6 commit コマンドまたは end コマンドを使用します。

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

---

## VXLAN 送信元 UDP ポートの設定

これはオプションのタスクです。デフォルトでは、カプセル化 VXLAN セグメントの送信元 UDP ポートは、内部ペイロードのレイヤ2アドレスフィールドのハッシュ関数によって計算されます。内部ペイロードのレイヤ2またはレイヤ3アドレスフィールドのいずれかで実行されるハッシュ関数を設定するには、次の作業を実行します。

### 手順の概要

1. l2vpn
2. load-balancing flow [ src-dst-mac | src-dst-ip ]

### 手順の詳細

---

#### ステップ1 l2vpn

例 :

```
RP/0/RSP0/cpu 0: router(config)# l2vpn
```

l2vpn 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 load-balancing flow [ src-dst-mac | src-dst-ip ]

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn)# load-balancing flow src-dst-mac
```

ハッシュ関数用に内部ペイロードのレイヤ2またはレイヤ3アドレスフィールドのいずれかを選択します。

---

## VXLAN 宛先 UDP ポートの設定

UDP ポート番号 4789 と 8472 はそれぞれ VXLAN と OTV に割り当てられます。カプセル化 VXLAN セグメントの宛先 UDP ポート番号を設定するには、次の作業を実行します。デフォルトでは、カプセル化 VXLAN データグラムの宛先 UDP ポート番号が 4789 に設定されているため、これはオプションのタスクです。宛先 VTEP が OTV ポートを使用して VXLAN をサポートしている場合は、宛先 UDP ポート番号を 8472 に設定する必要があります。

### 手順の概要

#### 1. `vxlan udp port port-number`

### 手順の詳細

---

`vxlan udp port port-number`

例 :

```
RP/0/RSP0/cpu 0: router(config)# vxlan udp port 4789
```

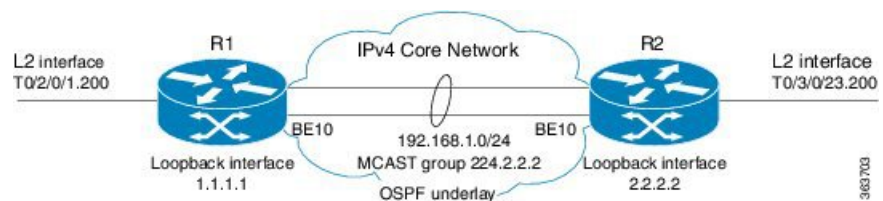
カプセル化 VXLAN セグメントの宛先 UDP ポート番号を設定します。

---

## レイヤ 2 VXLAN ゲートウェイの実装の設定例

次の例は、PE ルータ間のバンドルリンク接続としてコアネットワークが簡素化されたサンプルネットワークトポロジの、2台のプロバイダーエッジ (PE) ルータ (R1 および R2) でのレイヤ 2 VXLAN ゲートウェイ設定を示しています。

図 3: レイヤ 2 VXLAN ゲートウェイを使用するネットワーク



R1 での設定 :

```
interface Bundle-Ether10
  ipv4 address 192.168.1.1/24
!
interface Loopback0
  ipv4 address 1.1.1.1/32
!
interface T0/2/0/1
  no shut
!
interface T0/2/0/1.200 l2transport
  encapsulation dot1q 200
```

```

!
router ospf underlay
  router-id 1.1.1.1
  area 0
    interface Bundle-Ether10
    interface Loopback0
  !
interface nve 1
  member vni 1 mcast-group 224.2.2.2 0.0.0.0
  overlay-encapsulation vxlan
  source-interface Loopback0
!
l2vpn
  bridge group vxlan
  bridge-domain vxlan
    interface T0/2/0/1.200
    member vni 1
  !
multicast-routing
  address-family ipv4
    interface Loopback0
      enable
    interface Bundle-Ether10
      enable
  !
router pim
  address-family ipv4
    rp-address 1.1.1.1 bidir

```

**R2 での設定 :**

```

interface Bundle-Ether10
  ipv4 address 192.168.1.2/24
!
interface Loopback0
  ipv4 address 2.2.2.2/32
!
interface T0/3/0/23
  no shut
!
interface T0/3/0/23.200 l2transport
  encapsulation dot1q 200
!
router ospf underlay
  router-id 2.2.2.2
  area 0
    interface Bundle-Ether10
    interface Loopback0
  !
Interface nve 1
  member vni 1 mcast-group 224.2.2.2 0.0.0.0
  overlay-encapsulation vxlan
  source-interface Loopback0
!
l2vpn
  bridge group vxlan
  bridge-domain vxlan
    interface T0/3/0/23.200
    member vni 1
  !
multicast-routing
  address-family ipv4
    interface Loopback0
      enable
    interface Bundle-Ether10

```

```
        enable
!
router pim
  address-family ipv4
    rp-address 1.1.1.1 bidir
```

## EVPN VXLAN レイヤ 2 Data Center Interconnect ゲートウェイ

Cisco ASR 9000 シリーズ ルータはデータセンター相互接続 (DCI) レイヤ 2 ゲートウェイとして機能し、MPLS ベースの L2VPN ネットワークを介して EVPN VXLAN ベースのデータセンター間にレイヤ 2 接続を提供します。データセンターは、中間サービス プロバイダー ネットワークを通じて接続されます。EVPN VXLAN 対応データセンターは、EVPN コントロールプレーンを使用して、1つのデータセンターから別のデータセンターへのレイヤ 2 転送情報を配信します。この機能によって冗長性、復元力、プロビジョニング簡便性が得られます。

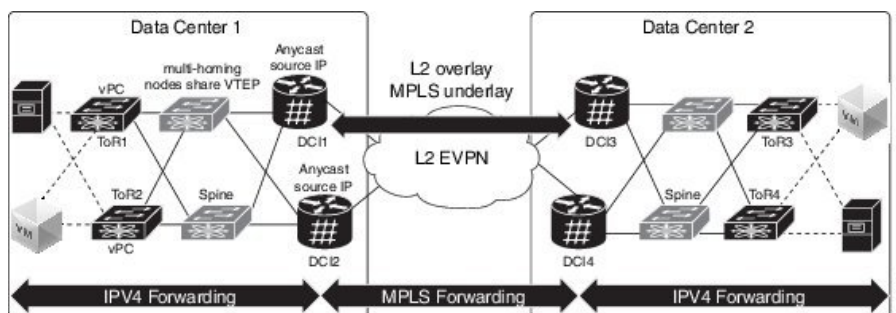
EVPN VXLAN レイヤ 2 DCI ゲートウェイ機能は次の機能をサポートしています。

- シングル ホーミングでの VXLAN アクセス
- エニーキャスト VXLAN 端末エンドポイント (VTEP) IP アドレスを使用したオールアクティブ マルチホーミングでの VXLAN アクセス
- 一意の VTEP IP アドレスを使用したオールアクティブ マルチホーミングでの VXLAN アクセス
- VXLAN カプセル化を使用した EVPN ESI マルチパス

### エニーキャスト VTEP IP アドレスを使用したオールアクティブ マルチホーミング

DCI は エニーキャスト VTEP IP アドレスを使用したオールアクティブ マルチホーミングに同じエニーキャスト VTE IP アドレスを使用します。Top of Rack (ToR) が複数のパスを使用して DCI に接続されており、トラフィックは ToR から DCI に複数の物理パスを通じて渡され、ロードバランシングにエニーキャスト IP アドレスが使用されているトポロジを考えてみます。DCI1 と DCI2 は、ネクストホップと同じエニーキャスト IP アドレスを使用して MAC ルートを ToR にアドバタイズします。つまり、ToR は DCI の同じエニーキャスト IP アドレスにトラフィックを送信し、ロードバランシングに IGP ECMP を使用します。仮想 PortChannel (vPC) では、ToR1 と ToR2 で同じ IP 設定を使用できます。ToR1 と ToR2 は、ネクストホップと同じ IP アドレスを使用して MAC ルートを DCI にアドバタイズします。そのため、DCI は ToR の同じ IP アドレスにトラフィックを送信し、ロードバランシングに IGP ECMP を使用します。DCI は、MPLS 転送を通じてリモート データセンターにトラフィックを送信します。

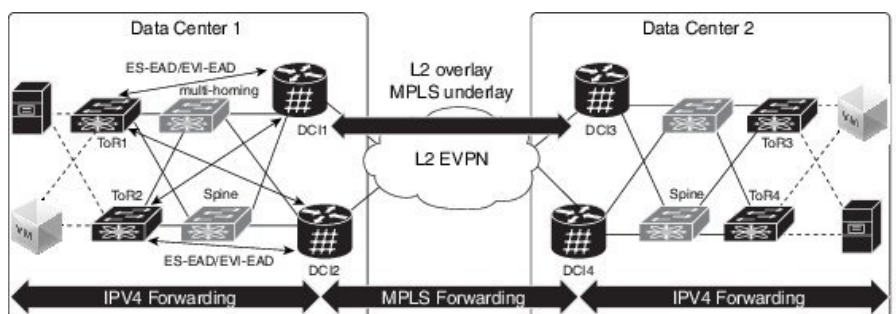
図 4: エニーキャスト VTEP IP アドレスを使用したオールアクティブ マルチホーミング



## 一意の VTEP IP アドレスを使用したオールアクティブ マルチホーミング

DCIではオールアクティブ マルチホーミングのエニーキャスト VTEP IP アドレスを一意の VTEP IP アドレスと共有しません。各 DCI は一意の VTEP IP アドレスを使用します。ToR が DCI から MAC ルートを受け取る次のトポロジを考えてみます。各 MAC ルートには一意のネクストホップがあります。DCI1 と DCI2 は両方とも異なるネクストホップを持つ同じ MAC のルートをアドバタイズするため、ToR には同じ MAC に 2 つの等コストネクストホップがあります。ToR は MAC にトラフィックを送信すると、ToR は両方のネクストホップ上でトラフィックのロード バランシングを実行します。

図 5: 一意の VTEP IP アドレスを使用したオールアクティブ マルチホーミング

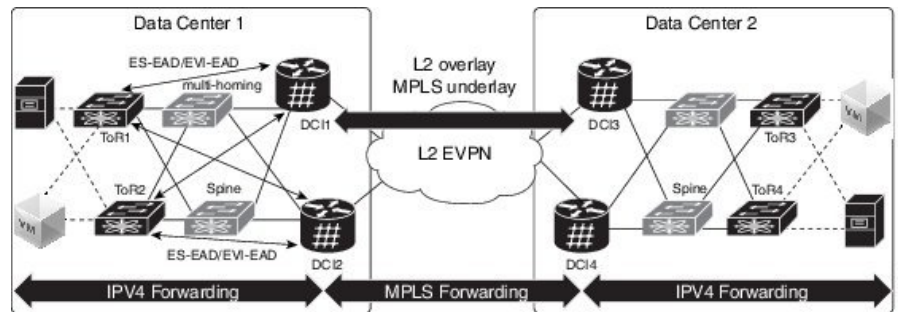


## VXLAN の EVPN ESI マルチパス : EVI ベースのロード バランシング

EVPN イーサネットセグメント識別子 (ESI) マルチパス機能は、アクティブ-アクティブのデュアルホーム接続 ToR と DCI へのマルチパストラフィックをサポートし、データセンター内に冗長接続を実現します。ESI マルチパスは、EVPN シグナリングを通じて ASR9k DCI ルータによって検出されます。パスは、イーサネットセグメント識別子 (ESI) と EVPN インスタンス (EVI) に基づいて選択されます。受信した MAC ルートのパスを解決するには、RFC 7432 に指定されているとおり、ES ごとにイーサネット A-D ルート (ES-EAD) を、EVI ごとにイーサネット A-D (EVI-EAD) を使用します。

DCI が ToR から MAC ルートを受信し、各 MAC ルートに各 ToR のネクストホップがある次のトポロジを考えてみます。同様に、DCI は ToR へのさまざまなネクストホップを使用して MAC ルートをアドバタイズします。ToR のペアの背後にある VM へ DCI がトラフィックを送信する場合は、すべての MAC に 2 つのパスが存在します。DCI は、2 つのパス上でトラフィックをロードバランスします。パスの選択は、EVI に基づいています。たとえば、DCI1 と DCI2 は EVI1 で学習した MAC アドレス宛のすべてのトラフィックに ToR1 を選択します。また、DCI1 と DCI2 は EVI2 で学習した MAC アドレス宛のすべてのトラフィックに ToR2 を選択します。

図 6: EVPN ESI マルチパス



## EVPN VXLAN レイヤ 2 Data Center Interconnect ゲートウェイの設定

EVPN VXLAN レイヤ 2 Data Center Interconnect ゲートウェイを設定するには、次のタスクを実行します。

EVPN ESI マルチパス機能を設定する場合は、ユニキャスト IP アドレスは設定しないでください。残りの設定タスクは同じです。

### BGP ルーティング プロセスでの L2 EVPN アドレス ファミリの設定

BGP ルーティング プロセスで EVPN アドレス ファミリを有効にするには、次のタスクを実行します。

#### 手順の概要

1. `configure`
2. `router bgp asn_id`
3. `nsr`
4. `bgp graceful-restart`
5. `bgp router-id ip-address`
6. `address-family l2vpn evpn`
7. `commit` コマンドまたは `end` コマンドを使用します。

## 手順の詳細

ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

ステップ 2 **router bgp *asn\_id***

例 :

```
RP/0/RSP0/cpu 0: router(config)# router bgp 100
```

BGP AS 番号を指定し、BGP コンフィギュレーション モードを開始します。このモードでは、BGP ルーティング プロセスを設定できます。

ステップ 3 **nsr**

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp)# nsr
```

ノンストップルーティングを有効にします。

ステップ 4 **bgp graceful-restart**

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp)# bgp graceful-restart
```

ルータのグレースフル リスタートをイネーブルにします。

ステップ 5 **bgp router-id *ip-address***

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp)# bgp router-id 209.165.200.227
```

指定したルータ ID で、ルータを設定します。

ステップ 6 **address-family *l2vpn evpn***

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp)# address-family l2vpn evpn
```

BGP ルーティングプロセスでグローバルに EVPN アドレスファミリを有効にし、EVPN アドレスファミリ設定サブモードを開始します。

ステップ 7 **commit** コマンドまたは **end** コマンドを使用します。

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。



- [Cancel] : 設定変更をコミットせずに、コンフィギュレーション モードに留まります。

## DCI と ToR 間のルーティング セッションの設定

DCI と ToR 間のルーティングセッションを設定するには、次の作業を実行します。

### 手順の概要

1. **configure**
2. **router bgp *asn\_id***
3. **neighbor *ip-address***
4. **remote-as *autonomous-system-number***
5. **ebgp-multihop *maximum hop count***
6. **update-source *loopback***
7. **address-family *l2vpn evpn***
8. **import *stitching-rt reoriginate***
9. **route-policy *route-policy-name in***
10. **encapsulation-type *type***
11. **route-policy *route-policy-name out***
12. **advertise *l2vpn evpn re-originated stitching-rt***
13. **commit** コマンドまたは **end** コマンドを使用します。

### 手順の詳細

#### ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

#### ステップ 2 **router bgp *asn\_id***

例 :

```
RP/0/RSP0/cpu 0: router(config)# router bgp 100
```

BGP AS 番号を指定し、BGP コンフィギュレーション モードを開始します。このモードでは、BGP ルーティング プロセスを設定できます。

#### ステップ 3 **neighbor *ip-address***

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp)# neighbor 209.165.200.225
```

ルータを BGP ルーティングのネイバー設定モードにして、ネイバーの IP アドレス 209.165.200.225 を BGP ピアとして設定します。

**ステップ 4** `remote-as autonomous-system-number`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# remote-as 2000
```

ネイバーを作成し、そのネイバーをリモート自律システム番号に割り当てます。

**ステップ 5** `ebgp-multihop maximum hop count`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# ebgp-multihop 255
```

外部 BGP ネイバーとのマルチホップ ピアリングをイネーブルにします。

**ステップ 6** `update-source loopback`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# update-source loopback1
```

BGP セッションが、特定のインターフェイスのプライマリ IP アドレスをローカルアドレスとして使用できるようにします。

**ステップ 7** `address-family l2vpn evpn`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# address-family l2vpn evpn
```

EVPN アドレスファミリを設定します。

**ステップ 8** `import stitching-rt reoriginate`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# import stitching-rt reoriginate
```

スティッチングルート ターゲット識別子と一致するルートターゲット識別子を持つ BGP EVPN NLRI からのルーティング情報のインポートを有効にし、この再発信後のルーティング情報を L2VPN BGP ネイバーにエクスポートします。

**ステップ 9** `route-policy route-policy-name in`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# route-policy pass-all in
```

着信ユニキャストルートにルートポリシーを適用します。

**ステップ 10** `encapsulation-type type`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# encapsulation-type vxlan
```

カプセル化タイプとして VXLAN を設定します。

**ステップ 11** `route-policy route-policy-name out`

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# route-policy pass-all out
```

発信ユニキャストルートにルートポリシーを適用します。

#### ステップ 12 **advertise l2vpn evpn re-originated stitching-rt**

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# advertise l2vpn evpn re-originated stitching-rt
```

L2VPN BGP ネイバーから受信する L2VPN EVPN ルートのアドバタイズメントを設定します。

#### ステップ 13 **commit** コマンドまたは **end** コマンドを使用します。

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずに設定セッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

## リモート DCI 接続の BGP セッションの設定

リモート DCI 接続に BGP セッションを設定するには、次のタスクを実行します。

### 手順の概要

1. **configure**
2. **router bgp *asn\_id***
3. **neighbor *ip-address***
4. **remote-as *autonomous-system-number***
5. **update-source *loopback***
6. **address-family l2vpn evpn**
7. **import re-originate stitching-rt**
8. **advertise l2vpn evpn re-originated**
9. **commit** コマンドまたは **end** コマンドを使用します。

### 手順の詳細

#### ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーションモードを開始します。

#### ステップ 2 **router bgp *asn\_id***

例 :

```
RP/0/RSP0/cpu 0: router(config)# router bgp 200
```

BGP AS 番号を指定し、BGP コンフィギュレーション モードを開始します。このモードでは、BGP ルーティング プロセスを設定できます。

### ステップ 3 neighborip-address

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp)# neighbor 209.165.201.1
```

ルータを BGP ルーティングのネイバー設定モードにして、ネイバーの IP アドレス 209.165.201.1 を BGP ピアとして設定します。

### ステップ 4 remote-as autonomous-system-number

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# remote-as 100
```

ネイバーを作成し、そのネイバーをリモート自律システム番号に割り当てます。

### ステップ 5 update-source loopback

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# update-source loopback2
```

BGP セッションが、特定のインターフェイスのプライマリ IP アドレスをローカルアドレスとして使用できるようにします。

### ステップ 6 address-family l2vpn evpn

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr)# address-family l2vpn evpn
```

EVPN アドレスファミリーを設定します。

### ステップ 7 import re-originate stitching-rt

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# import re-originate stitching-rt
```

スティッチングルートターゲット識別子と一致するルートターゲット識別子を持つ BGP EVPN NLRI からのルーティング情報のインポートを有効にし、この再発信後のルーティング情報を L2VPN BGP ネイバーにエクスポートします。

### ステップ 8 advertise l2vpn evpn re-originated

例 :

```
RP/0/RSP0/cpu 0: router(config-bgp-nbr-af)# advertise l2vpn evpn re-originated
```

L2VPN BGP ネイバーから受信する L2VPN EVPN ルートのアドバタイズメントを設定します。

### ステップ 9 commit コマンドまたは end コマンドを使用します。

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずに設定セッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

## ネットワーク仮想化エンドポイント (NVE) インターフェイスの設定

VNE インターフェイスを作成し、VxLAN の VXLAN トンネル エンドポイント (VTEP) として設定します。

### 手順の概要

1. **configure**
2. **interface nve** *nve-identifier*
3. **source-interface loopback** *loopback-interface-identifier*
4. **anycast source-interface loopback** *loopback-interface-identifier*
5. **redundancy**
6. **backbone vxlan**
7. **iccp group** *group number*
8. **exit**
9. **backbone mpls**
10. **iccp group** *group number*
11. **exit**
12. **exit**
13. **member vni** *vni number*
14. **load-balance per-evi**
15. **suppress-unknown-unicast-flooding**
16. **mcast-group** *ip\_address*
17. **host-reachability protocol** *protocol*
18. **commit** または **end** コマンドを使用します

### 手順の詳細

#### ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

#### ステップ 2 **interface nve** *nve-identifier*

例 :

```
RP/0/RSP0/cpu 0: router(config)# interface nve 1
```

NVE インターフェイスを作成し、NVE インターフェイス設定サブモードを開始します。

**ステップ 3** **source-interface loopback loopback-interface-identifier**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# 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** **anycast source-interface loopback loopback-interface-identifier**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# anycast source-interface loopback 1
```

エニーキャストモードのパラメータと、エニーキャストモードの送信元インターフェイスを設定します。

エニーキャスト IP アドレスは、ファブリック側の BGP ネクストホップに使用されます。ESI マルチパス機能を設定する場合は、エニーキャスト IP アドレスは設定しないでください。

**ステップ 5** **redundancy**

例 :

```
RP/0/RSP0/cpu 0: router(config-if)# redundancy
```

冗長パスを設定します。

**ステップ 6** **backbone vxlan**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red)# backbone vxlan
```

シャーシ間通信プロトコル (ICCP) VXLAN バックボーンの設定

**ステップ 7** **iccp group group number**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red-backbone-vxlan)# iccp group 11
```

ICCP グループ番号を設定します。

**ステップ 8** **exit**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red-backbone-vxlan)# exit
```

バックボーン vxlan サブモードを終了し、冗長サブモードに戻ります。

**ステップ 9** **backbone mpls**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red)# backbone mpls
```

ICCP MPLS バックボーンを設定します。

**ステップ 10** **iccp group group number**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red-backbone-mpls)# iccp group 12
```

MPLS バックボーン の ICCP グループ番号を設定します。

#### ステップ 11 **exit**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red-backbone-mpls)# exit
```

バックボーン mpls サブモードを終了し、冗長サブモードに戻ります。

#### ステップ 12 **exit**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-red)# exit
```

冗長サブモードを終了し、インターフェイスサブモードに戻ります。

#### ステップ 13 **member vni vni\_number**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve)# member vni 1
```

VxLAN ネットワーク識別子 (VNI) を使用して単一の VxLAN を NVE インターフェイスに関連付け、この VNI に関連付けられるマルチキャストアドレスを指定します。

#### ステップ 14 **load-balance per-evi**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-vni)# load-balance per-evi
```

EVI 単位のロードバランスモードを設定します (デフォルトはフロー単位)。

#### ステップ 15 **suppress-unknown-unicast-flooding**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-vni)# suppress-unknown-unicast-flooding
```

不明なユニキャストフラッドの抑制を設定します。

#### ステップ 16 **mcast-group ip\_address**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-vni)# mcast-group 209.165.202.129
```

VNI に関連付けられるマルチキャストアドレスを指定します。

#### ステップ 17 **host-reachability protocol protocol**

例 :

```
RP/0/RSP0/cpu 0: router(config-nve-vni)# host-reachability protocol bgp
```

VxLAN トンネルエンドポイント到達可能性の BGP 制御プロトコルを設定します。

#### ステップ 18 **commit** または **end** コマンドを使用します

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

---

## ブリッジドメインの設定

次のステップを実行して DCI ゲートウェイ上にブリッジドメインを設定します。

### 手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group***bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **evi ethernet vpn id**
6. **exit**
7. **member vni vxlan-id**
8. **commit** コマンドまたは **end** コマンドを使用します。

### 手順の詳細

---

#### ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

#### ステップ 2 **l2vpn**

例 :

```
RP/0/RSP0/cpu 0: router(config)# l2vpn
```

l2vpn 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 **bridge group***bridge-group-name*

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn)# bridge group bg1
```

ブリッジグループ設定モードを開始します。

#### ステップ 4 **bridge-domain** *bridge-domain-name*

例 :



```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg) # bridge-domain bd1
```

ブリッジドメイン設定モードを開始します。

#### ステップ 5 `evi ethernet vpn id`

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg-bd) # evi 1
```

イーサネット VPN ID を作成します。

#### ステップ 6 `exit`

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg-bd-evi) # exit
```

EVI 設定モードを終了して、ブリッジドメイン設定モードに戻ります。

#### ステップ 7 `member vni vxlan-id`

例 :

```
RP/0/RSP0/cpu 0: router(config-l2vpn-bg-bd) # member vni 1
```

ブリッジドメインにメンバー VNI を関連付けます。

#### ステップ 8 `commit` コマンドまたは `end` コマンドを使用します。

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

## BGP ルート ターゲットのインポート/エクスポート ルールの設定

デフォルトでは、次のパラメータが DCI の設定から自動生成されます。

- グローバルイーサネットセグメントテーブルのルート識別 (RD)

デフォルト : ループバック IP アドレスに基づく自動生成 RD

- EVI の BGP ルート識別子 (RD)

デフォルト : ループバック IP アドレスに基づく自動生成 RD

- EVI の BGP ルートターゲット。デフォルト : EVI ID に基づく自動生成 RT

次のタスクを実行して自動生成 BGP RD/RT 値を上書きし、転送情報のインポートとエクスポートに使用するルート ターゲットを定義します。

## 手順の概要

1. **configure**
2. **evpn**
3. **bgp**
4. **rd** { *2-byte as\_number* | *4-byte as\_number* | *IP\_address* | **none** } : { *nn* }
5. **exit**
6. **evi** *evi\_id*
7. **bgp**
8. **route-target import** { *2-byte as\_number* | *4-byte as\_number* | *IP\_address* | **none** } : { *nn* }  
[**stitching**]
9. **route-target export** { *2-byte as\_number* | *4-byte as\_number* | *IP\_address* | **none** } : { *nn* }  
[**stitching**]
10. **commit** コマンドまたは **end** コマンドを使用します。

## 手順の詳細

ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

ステップ 2 **evpn**

例 :

```
RP/0/RSP0/cpu 0: router(config)# evpn
```

EVPN 設定モードを開始します。

ステップ 3 **bgp**

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn)# bgp
```

EVPN BGP 設定モードを開始し、イーサネットセグメント ES:GLOBAL EVI (ES ルートの処理に使用) のスタティック BGP 設定を行います。

ステップ 4 **rd** { *2-byte as\_number* | *4-byte as\_number* | *IP\_address* | **none** } : { *nn* }

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-bgp)# rd 200:50
```

ルート識別子を設定します。

ステップ 5 **exit**

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-bgp)# exit
```

現在の設定モードを終了し、EVPN サブモードに戻ります

**ステップ 6** `evi evi_id`

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn)# evi 1
```

イーサネット VPN ID を設定します。

EVI ID の範囲は 1 ~ 65534 です。

**ステップ 7** `bgp`

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-evi)# bgp
```

特定の EVI の BGP 設定モードを開始します。

**ステップ 8** `route-target import { 2-byte as_number | 4-byte as_number | IP_address | none } : { nn } [stitching]`

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-evi-bgp)# route-target import 101:1 stitching
```

一致するルートターゲット値を持つ L2 EVPN BGP NLRI からのルートのインポートを設定します。

**ステップ 9** `route-target export { 2-byte as_number | 4-byte as_number | IP_address | none } : { nn } [stitching]`

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-evi-bgp)# route-target export 101:1 stitching
```

L2 EVPN BGP NLRI へのルートのエクスポートを設定し、指定されたルートターゲット識別子を BGP EVPN NLRI に割り当てます。

**ステップ 10** `commit` コマンドまたは `end` コマンドを使用します。**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

## イーサネット セグメント識別子の設定

イーサネット セグメント識別子 (ESI) を設定するには、次のタスクを実行します。

### 手順の概要

1. `configure`
2. `evpn`
3. `interface nve nve-identifier`

4. **ethernet-segment**
5. **identifier type** *esi-type esi-identifier*
6. **bgp route-target** *route target value*
7. **commit** または **end** コマンドを使用します

## 手順の詳細

---

### ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

### ステップ 2 **evpn**

例 :

```
RP/0/RSP0/cpu 0: router# evpn
```

EVPN 設定モードを開始します。

### ステップ 3 **interface nve nve-identifier**

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn)# interface nve 1
```

NVE インターフェイスを作成し、NVE インターフェイス設定サブモードを開始します。

### ステップ 4 **ethernet-segment**

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-ac)# ethernet-segment
```

EVPN イーサネットセグメント設定モードを開始します。

### ステップ 5 **identifier type esi-type esi-identifier**

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-ac-es)# identifier type 0 88.00.00.00.00.00.00.01
```

イーサネットセグメント識別子を設定します。

### ステップ 6 **bgp route-target route target value**

例 :

```
RP/0/RSP0/cpu 0: router(config-evpn-ac-es)# bgp route-target 8888.0000.0001
```

イーサネットセグメントの BGP インポートルートターゲットを設定します。

### ステップ 7 **commit** または **end** コマンドを使用します

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

## ICCP グループの設定

シャーン間通信プロトコル (ICCP) パラメータを設定するには、次のタスクを実行します。

コアインターフェイストラッキングにICCPグループを設定します。すべてのインターフェイスがダウンしている場合、DCIはコア/ファブリックネットワークから分離されます。関連付けられているNVEインターフェイスがダウンし、BGP NLRIが撤回されます。

### 手順の概要

1. **configure**
2. **redundancy**
3. **iccp group group number**
4. **mode singleton**
5. **backbone**
6. **interface GigabitEthernet GigabitEthernet Interface Instance**
7. **commit** または **end** コマンドを使用します

### 手順の詳細

#### ステップ 1 **configure**

例 :

```
RP/0/RSP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

#### ステップ 2 **redundancy**

例 :

```
RP/0/RSP0/cpu 0: router(config)# redundancy
```

冗長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 **iccp group group number**

例 :

```
RP/0/RSP0/cpu 0: router(config-redundancy)# iccp group 11
```

ICCP グループ番号を設定します。

#### ステップ 4 **mode singleton**

例：

```
RP/0/RSP0/cpu 0: router(config-redundancy-iccp-group)# mode singleton
```

グループをシングルトンモードで実行できるようにします。

## ステップ5 backbone

例：

```
RP/0/RSP0/cpu 0: router(config-redundancy-iccp-group)# backbone
```

ICCP バックボーン インターフェイスを設定します。

## ステップ6 interface GigabitEthernet *GigabitEthernet Interface Instance*

例：

```
RP/0/RSP0/cpu 0: router(config-redundancy-group-iccp-backbone)# interface GigabitEthernet 0/2/0/12
```

GigabitEthernet インターフェイスを設定します。

## ステップ7 commit または end コマンドを使用します

**commit** : 設定の変更を保存し、コンフィギュレーションセッションに留まります。

**end** : 次のいずれかのアクションを実行することをユーザに要求します。

- [Yes] : 設定変更を保存し、コンフィギュレーションセッションを終了します。
- [No] : 設定変更をコミットせずにコンフィギュレーションセッションを終了します。
- [Cancel] : 設定変更をコミットせずに、コンフィギュレーションモードに留まります。

# 例：エニーキャスト VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの設定

次に、エニーキャスト VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの例を示します。

```
interface nve1
source-interface loopback1
anycast source-interface loopback2
member vni 5100
mcast-address 239.1.1.1
host-reachabilty protocol bgp
!

evpn
evi 10
bgp
route-target import 100:10
route-target import 200:5100 stitching
route-target export 200:5100 stitching
```

```

!
!
l2vpn
  bridge group DCI
    bridge-domain V1
      evi 10
        member vni 5100
!
router bgp 100
  bgp router-id 209.165.200.226
  address-family l2vpn evpn

!
neighbor 209.165.201.2
  remote-as 100
  description core-facing
  update-source Loopback1
  address-family l2vpn evpn
    import re-originate stitching-rt
    advertise l2vpn evpn re-originated
!
neighbor 209.165.202.130
  remote-as 200
  ebgp-multihop 255
  update-source Loopback1
  address-family l2vpn evpn
    import stitching-rt re-originate
    route-policy passall in
    encapsulation-type vxlan
    route-policy passall out
    advertise l2vpn evpn re-originated stitching-rt
!

```

## 例：一意の VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの設定

次に、一意の VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの例を示します。

```

interface nve1
  source-interface loopback1
  member vni 5100
    mcast-address 239.1.1.1
    host-reachabilty protocol bgp
!
evpn
  evi 10
    bgp
      route-target import 100:10
      route-target import 200:5100 stitching
      route-target export 200:5100 stitching
!
!
l2vpn
  bridge group DCI
    bridge-domain V1
      evi 10
        member vni 5100

```

例：一意の VTEP IP アドレス設定を使用したオールアクティブ マルチホーミングの設定

```
!
router bgp 100
  bgp router-id 209.165.200.226
  address-family l2vpn evpn

!
neighbor 209.165.201.2
  remote-as 100
  description core-facing
  update-source Loopback1
  address-family l2vpn evpn
    import re-originate stitching-rt
    multipath
    advertise l2vpn evpn re-originated
!
neighbor 209.165.202.130
  remote-as 200
  ebgp-multihop 255
  update-source Loopback1
  address-family l2vpn evpn
    import stitching-rt re-originate
    multipath
    route-policy passall in
    encapsulation-type vxlan
    route-policy passall out
    advertise l2vpn evpn re-originated stitching-rt
!
```