



EVPN の概要

イーサネット VPN (EVPN) は、MPLS ネットワークを介してイーサネット マルチポイント サービスを提供する次世代のソリューションです。EVPN は、コアでコントロールプレーン ベースの MAC ラーニングを可能にする既存の仮想プライベート LAN サービス (VPLS) とは 対照的に動作します。EVPN では、EVPN インスタンスに参加している PE が MP-BGP プロト コルを使用してコントロールプレーン内でカスタマー MAC ルートを学習します。コントロ ールプレーン MAC ラーニングは、フローごとのロードバランシングによるマルチホーミングの サポートなど、VPLS の欠点に EVPN で対処できるようにする数多くの利点をもたらします。

EVPN は、ネットワークでの次の新たなニーズに対応するソリューションをネットワーク オペ レータに提供します。

- データセンター相互接続操作 (DCI)
- クラウドおよびサービスの仮想化
- プロトコルの排除とネットワークの簡素化
- 同じ VPN を介した L2 サービスと L3 サービスの統合
- サービスとワークロードの柔軟な配置
- L2 および L3 VPN によるマルチテナント
- 最適な転送とワークロード モビリティ
- 高速コンバージェンス
- 効率的な帯域幅利用

EVPN の利点

EVPN には次の利点があります。

- 統合サービス : L2 および L3 VPN サービスの統合、拡張性と制御における L3VPN のよ うな原則と運用経験、ECMP を使用したオールアクティブマルチホーミングと PE ロードバ ランシング、複数の PE に対してマルチホームである CE との間で発着信するトラフィッ クのロードバランシングが可能。

- ネットワーク効率：フラッドと学習メカニズムの排除、デュアルホーム接続サーバへのリンクでの障害発生時の高速再ルーティング、復元力、および高速な再コンバージェンス、ブロードキャスト、不明ユニキャスト、マルチキャスト（BUM）トラフィック配信の最適化。
- サービスの柔軟性：MPLS データプレーンのカプセル化、既存および新しいサービスタイプのサポート（E-LAN、E-Line）、ピア PE 自動検出、および冗長グループ自動感知。

EVPN のモード

次の EVPN モードがサポートされています。

- シングルホーミング：これにより、カスタマー エッジ（CE）デバイスをプロバイダー エッジ（PE）デバイス 1 台に接続できます。
- マルチホーミング：これにより、カスタマーエッジ（CE）デバイスを複数のプロバイダーエッジ（PE）デバイスに接続できます。マルチホーミングにより、冗長接続が確保されます。冗長 PE デバイスは、ネットワーク障害が発生している場合にトラフィックが中断されないようにします。次にマルチホーミングのタイプを示します。
 - オールアクティブ：オールアクティブモードでは、特定のイーサネットセグメントに接続されているすべての PE が、そのイーサネットセグメントとの間で発着信するトラフィックを転送できます。

- [EVPN の概念（2 ページ）](#)
- [EVPN 動作（3 ページ）](#)
- [EVPN ルートタイプ（4 ページ）](#)
- [EVPN L2 ブリッジング サービスの設定（6 ページ）](#)
- [EVPN ソフトウェア MAC ラーニング（6 ページ）](#)
- [EVPN アウトオブサービス（17 ページ）](#)
- [EVPN ルーティングポリシー（20 ページ）](#)

EVPN の概念

EVPN 機能を実装するには、次の概念を理解する必要があります。

- イーサネットセグメント（ES）：イーサネットセグメントは、マルチホームデバイスに接続する一連のイーサネットリンクです。マルチホームデバイスまたはネットワークが 2 つ以上の PE に一連のイーサネットリンクを通じて接続されている場合に、その一連のリンクをイーサネットセグメントと呼びます。イーサネットセグメントルートはルートタイプ 4 とも呼びます。このルートは、BUM トラフィックの指定フォワーダ（DF）の選択に使用されます。
- イーサネットセグメント識別子（ESI）：イーサネットセグメントには一意の非ゼロの識別子が割り当てられます。これをイーサネットセグメント識別子（ESI）と呼びます。ESI は、ネットワーク全体にわたってイーサネットセグメントを一意に表します。

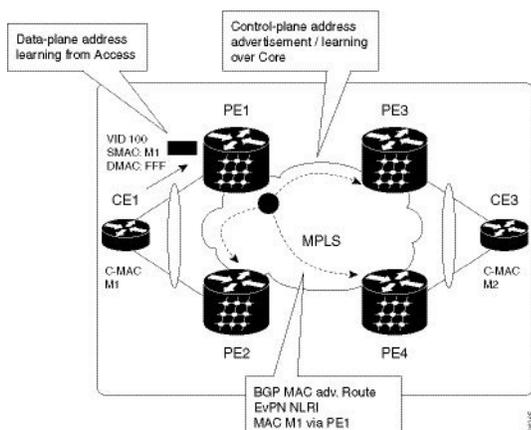
- **EVI** : EVPN インスタンス (EVI) は仮想ネットワーク識別子 (VNI) で表されます。EVI は、PE ルータ上の VPN を表します。EVI は IP VPN ルーティングおよび転送 (VRF) と同じ役割を果たし、インポート/エクスポートルートターゲット (RT) が割り当てられます。ユーザ ネットワーク インターフェイス (UNI) でのサービス多重化動作に応じて、ポート上のすべてのトラフィック (すべて対1のバンドリング)、VLAN 上のトラフィック (1対1のマッピング)、または VLAN のリスト/範囲のトラフィック (選択的バンドリング) をブリッジドメイン (BD) にマップできます。このBDはEVIに関連付けられ、MPLS コアに転送されます。
- **EAD/ES** : ES ごとのイーサネット自動検出ルートはルートタイプ1とも呼ばれます。このルートは、アクセス失敗のシナリオ時にトラフィックを早急に収束するために使用されません。このルートにはイーサネット タグ 0xFFFFFFFF が使用されます。
- **EAD/EVI** : EVI ごとのイーサネット自動検出ルートはルートタイプ1とも呼ばれます。このルートは、トラフィックはスイッチの1つにのみハッシュされる時のエイリアシングとロードバランシングに使用されます。EAD/ES ルートと区別するため、このルートにはイーサネット タグ値 0xFFFFF を使用できません。
- **エイリアシング** : ルートタイプ1のEAD/EVI ルートを使用する所定のイーサネット セグメントで接続されているすべてのスイッチへのトラフィックのロードバランシングに使用されます。これはホストを実際に学習するスイッチとは関係なく実行されます。
- **大量撤回** : ルートタイプ1のEAD/ES ルートを使用し、アクセス障害シナリオ時に早急に収束するために使用されます。
- **DF の選択** : ループの転送を防ぐために使用されます。カプセル化を解除し、所定のイーサネット セグメントにトラフィックを転送するため、単一のルータのみを使用します。

EVPN 動作

以下をアドバタイズするため、PE は起動時に EVPN ルートを交換します。

- **VPN メンバーシップ** : PE は所定のリモート PE のすべてのメンバーを検出します。マルチキャスト入力レプリケーションモデルの場合、EVI に関連付けられている PE フラッドリストの構築にこの情報が使用されます。MAC アドレスを学習した時点で BUM ラベルとユニキャスト ラベルが交換されます。
- **イーサネット セグメント到達可能性** : マルチホーミングのシナリオでは、PE がリモート PE と対応するそれらの冗長モード (オールアクティブまたはシングルアクティブ) を自動的に検出します。セグメント障害が発生した場合、PE はこの段階で使用していたルートを撤回し、リモート PE 上の MAC 大量撤回をシグナリングすることで高速コンバージェンスをトリガーします。
- **冗長グループメンバーシップ** : 同じイーサネットセグメントに接続している (マルチホーミング) PE は互いに自動的に検出され、所定の EVI に対するブロードキャスト、不明ユニキャストおよびマルチキャスト (BUM) トラフィックの転送を担う指定フォワーダ (FD) を選択します。

図 1: EVPN 動作



EVPN はシングルホーミング モードまたはデュアルホーミング モードで動作できます。PE 上で EVPN が有効になっており、各 PE が所定の EVPN インスタンスの他のすべてのメンバー PE を検出したときにルートタイプ 3 がアドバタイズされるシングルホーミングのシナリオを考えてみます。不明ユニキャスト（または BUM）MAC を PE で受信すると、EVPN ルートタイプ 2 として他の PE にアドバタイズされます。MAC ルートは EVPN ルートタイプ 2 を使用して他の PE にアドバタイズされます。マルチホーミングのシナリオでは、ルートタイプ 1、3、および 4 がアドバタイズされ、他の PE とそれらの冗長モード（シングルアクティブまたはオールアクティブ）を検出します。ルートタイプ 1 を使用するのには、同じ CE をホストする他の PE を自動検出するためです。この他にも、このルートタイプは CE と PE 間の破損リンクから離れている高速ルートユニキャストトラフィックにも使用されます。ルートタイプ 4 は、指定フォワーダの選択に使用されます。たとえば、カスタマートラフィックが PE に着信し、ローカルイーサネットセグメント上で学習した各カスタマー MAC アドレスの到達可能性情報を EVPN MAC アドバタイズメントルートでコアを介して配布するトポロジを考えてみます。各 EVPN MAC ルートは、カスタマー MAC アドレスと、MAC を学習したポートに関連付けられたイーサネットセグメントおよびその関連付けられた MPLS ラベルをアナウンスします。この EVPN MPLS ラベルは、アドバタイズされた MAC アドレス宛にトラフィックを送信するときにリモート PE によって後で使用されます。

EVPN ルート タイプ

EVPN ネットワーク層到達可能性情報（NLRI）は、さまざまなルートタイプを提供します。

表 1: EVPN ルートタイプ

ルートタイプ	名前	使用法
1	イーサネット自動検出（AD）ルート	ES ごとの少数ルートの送信、ES に属する EVI のリストの伝送

ルートタイプ	名前	使用法
2	MAC/IP アドバタイズメントルート	MACのアドバタイズ、アドレス到達可能性、IP/MACバインディングのアドバタイズ
3	包括的なマルチキャストイーサネットタグルート	マルチキャストトンネルエンドポイントの検出
4	イーサネットセグメントルート	冗長グループの検出、DFの選択
5	IPプレフィックスルート	IPプレフィックスのアドバタイズ

ルートタイプ1：イーサネット自動検出（AD）ルート

イーサネット自動検出（AD）ルートは、EVIごととESIごとにアドバタイズされます。これらのルートは、ESごとに送信されます。これらはESに属しているEVIのリストを伝送します。ESIフィールドは、CEがシングルホームの場合はゼロに設定されます。このルートタイプは、ロードバランシングのためのMACアドレスの大量撤回とエイリアシングに使用されません。

ルートタイプ2：MAC/IP アドバタイズメントルート

これらのルートはVLANごとのルートであるため、VNIに含まれているPEのみにこれらのルートが必要です。ホストのIPアドレスとMACアドレスがNRLI内のピアにアドバタイズされます。MACアドレスのコントロールプレーン学習は不明ユニキャストのフラッディングを削減します。

ルートタイプ3：包括的なマルチキャストイーサネットタグルート

このルートは、送信元PEからリモートPEへのブロードキャスト、不明ユニキャスト、およびマルチキャスト（BUM）トラフィック用の接続を確立します。このルートは、VLANごととESIごとにアドバタイズされます。

ルートタイプ4：イーサネットセグメントルート

イーサネットセグメントルートではCEデバイスを2台のデバイスまたはPEデバイスを接続できます。ESルートでは同じイーサネットセグメントに接続されているPEデバイスを検出できます。

ルートタイプ5：IPプレフィックスルート

IPプレフィックスがMACアドバタイズメントルートとは関係なくアドバタイズされます。EVPN IRBでは、ホストルート/32はRT-2を使用してアドバタイズされ、サブネット/24はRT-5を使用してアドバタイズされます。



(注) EVPN IRB では、ホストルート /32 は RT-2 を使用してアドバタイズされ、サブネット /24 は RT-5 を使用してアドバタイズされます。

EVPN L2 ブリッジング サービスの設定

EVPN L2 ブリッジング サービスを設定するには、次のステップを実行します。

```
Router# configure
Router(config)# l2vpn
Router(config-l2vpn)# bridge group 1
Router(config-l2vpn-bg)# bridge-domain 1-1
Router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/0/0/1.1
Router(config-l2vpn-bg-bd-ac)# evi 1
Router(config-l2vpn-bg-bd-ac-evi)# commit
Router(config-l2vpnbg-bd-ac-evi)# exit
```

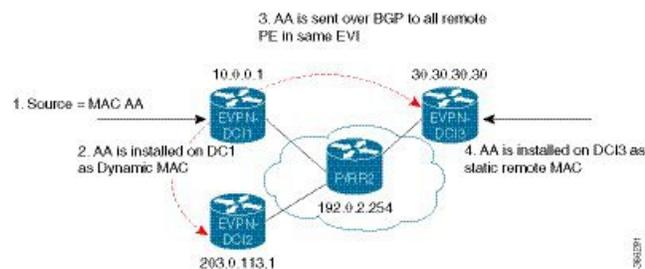
実行コンフィギュレーション

```
configure
l2vpn
  bridge group 1
    bridge-domain 1-1
      interface GigabitEthernet 0/0/0/1.1
        evi 1
!
```

EVPN ソフトウェア MAC ラーニング

あるデバイス上で学習した MAC アドレスは、VLAN 内の別のデバイス上で学習されるか、配布されるようにする必要があります。EVPN ソフトウェア MAC ラーニング機能では、あるデバイス上で学習された MAC アドレスをネットワークに接続された別のデバイスに配布できます。MAC アドレスは、BGP を使用してリモートデバイスから学習されます。

図 2: EVPN ソフトウェア MAC ラーニング



上の図は、ソフトウェア MAC ラーニングのプロセスを示しています。次に、このプロセスに関わるステップを示します。

1. トラフィックは、ブリッジドメイン内の1つのポートに着信します。
2. 送信元 MAC アドレス (AA) は PE 上で学習され、ダイナミック MAC エントリとして格納されます。
3. MAC アドレス (AA) がタイプ 2 BGP ルーに交換され、BGP を介して同じ EVI 内のすべてのリモート PE に送信されます。
4. MAC アドレス (AA) は、リモート MAC アドレスとして PE で更新されます。

EVPN ソフトウェア MAC ラーニングの設定



(注) EVPN ブリッジングでは、**control-word-disable** コマンドはサポートされていません。

次の項では、EVPN ソフトウェア MAC ラーニングの設定方法について説明します。

```

/* Configure bridge domain. */

RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group EVPN_SH
RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain EVPN_2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface TenGigE0/4/0/10.2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface BundleEther 20.2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# storm-control broadcast pps 10000
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-nbr)# evi 2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg)# exit
RP/0/RSP0/CPU0:router(config-l2vpn)# exit

/* Configure advertisement of MAC routes, suppress unknown unicast, disable the control
word, configure the flow label, configure BGP route-exchange using RT. */

RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# evi 2001

/*Use the advertise-mac command to control the advertisement of MAC routes through BGP
to other neighbors. */
RP/0/RSP0/CPU0:router(config-evpn-evi)# advertise-mac

/* Use the unknown-unicast-suppress command to prevent unknown unicast traffic from going
to the MPLS core and then to all remote PE bridge-ports. */
RP/0/RSP0/CPU0:router(config-evpn-evi)# unknown-unicast-suppress

/* Use the control-word-disable command to prevent the control word from being sent in
the packet that is sent to MPLS core. The control word functionality is enabled by
default. */
RP/0/RSP0/CPU0:router(config-evpn-evi)# control-word-disable

/* Use the load-balance flow label static command to add additional flow label header
to the packet that is sent to MPLS core. The loadbalance flow functionality is disabled
by default. */
RP/0/RSP0/CPU0:router(config-evpn-evi)# load-balance flow label static

/* Perform the following steps to configure BGP route-exchange using RT */
RP/0/RSP0/CPU0:router(config-evpn-evi)# bgp

```

```

RP/0/RSP0/CPU0:router(config-evpn-evi)# route-target import 200:101
RP/0/RSP0/CPU0:router(config-evpn-evi)# route-target export 200:101

/* Configure address family session in BGP. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# router bgp 200
RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 209.165.200.227
RP/0/RSP0/CPU0:router(config-bgp)# address-family l2vpn evpn

RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 200
RP/0/RSP0/CPU0:router(config-bgp-nbr)# description MPLSFACINGPEER
RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0
RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family l2vpn evpn

```

EVPN ソフトウェア MAC ラーニングでサポートされているモード

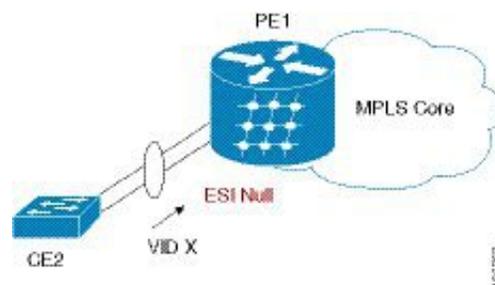
EVPN ソフトウェア MAC ラーニングでサポートされているモードは次のとおりです。

- シングル ホーム デバイス (SHD) またはシングル ホーム ネットワーク (SHN)
- デュアル ホーム デバイス (DHD) : オール アクティブ ロード バランシング

シングル ホーム デバイスまたはシングル ホーム ネットワーク モード

次の項では、EVPN ソフトウェア MAC ラーニング機能をシングル ホーム デバイスまたはシングル ホーム ネットワーク (SHD/SHN) モードで設定する方法について説明します。

図 3: シングル ホーム デバイスまたはシングル ホーム ネットワーク モード



上の図では、PE (PE1) はバンドルインターフェイスまたは物理インターフェイスを使用してイーサネットセグメントに接続されています。SHD/SHN にはヌルイーサネットセグメント識別子 (ESI) を使用します。

シングル ホーム デバイスまたはシングル ホーム ネットワーク モードでの EVPN の設定

この項では、シングル ホーム デバイスまたはシングル ホーム ネットワーク モードで EVPN ソフトウェア MAC ラーニング機能を設定する方法について説明します。

```

/* Configure bridge domain. */

RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group EVPN_ALL_ACTIVE
RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain EVPN_2001

```

```

RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface BundleEther1.2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# evi 2001

/* Configure advertisement of MAC routes. */

RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# evi 2001
RP/0/RSP0/CPU0:router(config-evpn-evi)# advertise-mac

/* Configure address family session in BGP. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router#(config)# router bgp 200
RP/0/RSP0/CPU0:router#(config-bgp)# bgp router-id 09.165.200.227
RP/0/RSP0/CPU0:router#(config-bgp)# address-family l2vpn evpn
RP/0/RSP0/CPU0:router#(config-bgp)# neighbor 10.10.10.10
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# remote-as 200
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# description MPLSFACING-PEER
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# update-source Loopback 0
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# address-family l2vpn evpn

```

実行コンフィギュレーション

```

l2vpn
bridge group EVPN_ALL_ACTIVE
  bridge-domain EVPN_2001
    interface BundleEther1.2001
      evi 2001
!
evpn
  evi 2001
  advertise-mac
!
router bgp 200 bgp
  router-id 40.40.40.40
  address-family l2vpn evpn
  neighbor 10.10.10.10
    remote-as 200 description MPLS-FACING-PEER
  updatesource Loopback0
  addressfamily l2vpn evpn

```

確認

シングル ホーム デバイスの EVPN を確認します。

```

RP/0/RSP0/CPU0:router# show evpn ethernet-segment interface Te0/4/0/10 detail

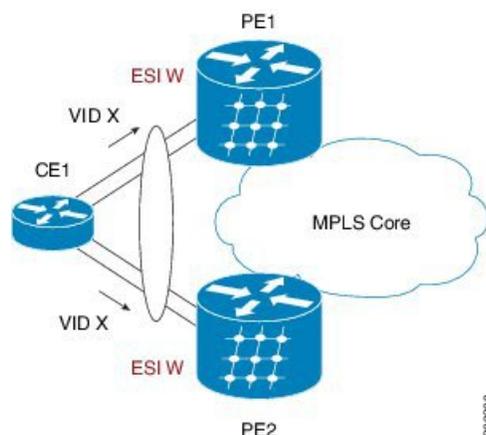
Ethernet Segment Id      Interface      Nexthops
-----
N/A                      Te0/4/0/10   20.20.20.20
.....
Topology :
Operational : SH
Configured : Single-active (AAPS) (default)

```

デュアルホームデバイス：オールアクティブロードバランシングモード

次の項では、デュアルホームデバイス（DHD）にオールアクティブロードバランシングモードでEVPNソフトウェアMACラーニング機能を設定する方法について説明します。

図4:デュアルホームデバイス：オールアクティブロードバランシングモード



オールアクティブロードバランシングはフローごとのアクティブ/アクティブ（AApF）と呼ばれています。上の図では、両方のEVPN PEに同一のイーサネットセグメント識別子を使用しています。PEは、バンドルインターフェイスを使用してイーサネットセグメントに接続されています。CEでは、単一のバンドルが2つのEVPN PEに向けて設定されます。このモードでは、学習したMACアドレスがPE1とPE2の両方に格納されます。PE1とPE2は両方とも同じEVI内でトラフィックを転送できます。

デュアルホームデバイスでのEVPNソフトウェアMACラーニングの設定：オールアクティブモード

この項では、オールアクティブモードのデュアルホームデバイスでEVPNソフトウェアMACラーニング機能を設定する方法について説明します。

```

/* Configure bridge domain. */

RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group EVPN_ALL_ACTIVE
RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain EVPN_2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface BundleEther1.2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# evi 2001

/* Configure advertisement of MAC routes. */

RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# evi 2001
RP/0/RSP0/CPU0:router(config-evpn-evi)# advertise-mac
RP/0/RSP0/CPU0:router(config-evpn-evi)# exit
RP/0/RSP0/CPU0:router(config-evpn)# interface bundle-ether1
RP/0/RSP0/CPU0:router(config-evpn-ac)# ethernet-segment

```

```

RP/0/RSP0/CPU0:router(config-evpn-ac-es)# identifier type 0 01.11.00.00.00.00.00.01

/* Configure address family session in BGP. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# router bgp 200
RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 209.165.200.227
RP/0/RSP0/CPU0:router(config-bgp)# address-family l2vpn evpn
RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.10.10.10
RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 200
RP/0/RSP0/CPU0:router(config-bgp-nbr)# description MPLSFACING-PEER
RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0
RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family l2vpn evpn

/* Configure Multichassis Link Aggregation Control Protocol (MLACP) bundle. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether1 300
RP/0/RSP0/CPU0:router(config-if)# lACP switchover suppress-flaps 300
RP/0/RSP0/CPU0:router(config-if)# mlACP iccp-group 11
RP/0/RSP0/CPU0:router(config-if)# mlACP port-priority 1
RP/0/RSP0/CPU0:router(config-if)# exit

/* Configure VLAN Header Rewrite.*/

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bundle-Ether1.2001 l2transport
RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q 10
RP/0/RSP0/CPU0:router(config-if)# rewrite ingress tag pop 1 symmetric

/* Configure Interchassis Communication Protocol. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# redundancy iccp group 100
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# mlACP node 1
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# mlACP system mac 0111.0222.0111
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# mlACP system priority 1
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# mode singleton
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# backbone
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group-backbone)# interface Bundle-Ether10
RP/0/RSP0/CPU0:router(config-redundancy-iccp-group-backbone)# interface HundredGigE
0/2/0/5

```

実行コンフィギュレーション

```

l2vpn
bridge group EVPN_ALL_ACTIVE
  bridge-domain EVPN_2001
  interface Bundle-Ether1.2001
  evi 2001
evpn
  evi 2001
  advertise-mac
  interface bundle-ether1
  ethernet-segment
  identifier type 0 01.11.00.00.00.00.00.01

router bgp 200
bgp router-id 209.165.200.227

```

```

address-family l2vpn evpn
!
neighbor 10.10.10.10
  remote-as 200
  description MPLS-FACING-PEER
  update-source Loopback0
  address-family l2vpn evpn
interface Bundle-Ether1
l2vpn switchover suppress-flaps 300
mlacp iccp-group 11
mlacp port-priority 1
load-interval 30
!
interface bundle-Ether1.2001 l2transport
  encapsulation dot1aq 2001
  rewrite ingress tag pop 1 symmetric
!
redundancy
iccp
  group 11
  mlacp node 1
  mlacp system mac 0111.0222.0111
  mlacp system priority 1
  mode singleton
  backbone
  interface Bundle-Ether10
  interface HundredGigE0/2/0/5

```

確認

オールアクティブモードのデュアルホームデバイスのEVPNを確認します。

```
RP/0/RSP0/CPU0:router# show evpn ethernet-segment interface bundle-Ether 1 carvin$
```

```

Ethernet Segment Id      Interface  Nexthops
-----
0100.211b.fce5.df00.0b00 BE11      10.10.10.10
209.165.201.1
Topology :
Operational : MHN
Configured : All-active (AApF) (default)
Primary Services : Auto-selection
Secondary Services: Auto-selection
Service Carving Results:
Forwarders : 4003
Elected : 2002
EVI E : 2000, 2002, 36002, 36004, 36006, 36008
.....
Not Elected : 2001
EVI NE : 2001, 36001, 36003, 36005, 36007, 36009

MAC Flushing mode : Invalid

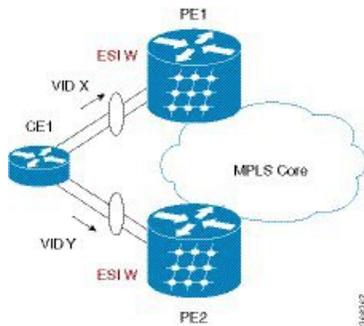
Peering timer : 3 sec [not running]
Recovery timer : 30 sec [not running]
Local SHG label : 34251
Remote SHG labels : 1
38216 : nexthop 209.165.201.1

```

デュアルホーム デバイス : シングルアクティブロードバランシングモード

次の項では、デュアルホーム デバイス (DHD) にシングルアクティブロードバランシングモードで EVPN ソフトウェア MAC ラーニング機能を設定する方法について説明します。

図 5: デュアルホーム デバイス : シングルアクティブロードバランシング



また、シングルアクティブロードバランシングは、サービスごとのオールアクティブ (AApS) とも呼ばれています。

両方の EVPN PE に同一の ESI が設定されます。CE では、2 つの EVPN PE への個別のバンドルまたは独立した物理インターフェイスが設定されます。このモードでは、学習した MAC アドレスが PE1 と PE2 の両方に格納されます。所定の時間に 1 つの PE のみが EVI 内にトラフィックを転送できます。

デュアルホーム デバイスでの EVPN ソフトウェア MAC ラーニングの設定 : シングルアクティブモード

この項では、シングルアクティブモードのデュアルホームで EVPN ソフトウェア MAC ラーニングを設定する方法について説明します。

```
/* Configure bridge domain. */

RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group EVPN_ALL_ACTIVE
RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain EVPN_2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface BundleEther1.2001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# evi 2001

/* Configure VLAN Header Rewrite (Single-tagged sub-interface).*/

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bundle-Ether1.21 l2transport
RP/0/RSP0/CPU0:router(config-if)# lacp switchover suppress-flaps 300
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether1.2001 l2transport
RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q 10
RP/0/RSP0/CPU0:router(config-if)# rewrite ingress tag pop 1 symmetric

/* Configure advertisement of MAC routes. */
```

```

RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# evi 2001
RP/0/RSP0/CPU0:router(config-evpn-evi)# advertise-mac

/* Configure load balancing. */

RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# evi 2001
RP/0/RSP0/CPU0:router(config-evpn-evi)# advertise-mac
RP/0/RSP0/CPU0:router(config-evpn-evi)# exit
RP/0/RSP0/CPU0:router(config-evpn)# interface bundle-ether1
RP/0/RSP0/CPU0:router(config-evpn)# ethernet-segment
RP/0/RSP0/CPU0:router(config-evpn-es)# load-balancing-mode single-active
RP/0/RSP0/CPU0:router(config-evpn-es)# identifier type 0 12.12.00.00.00.00.00.02
RP/0/RSP0/CPU0:router(config-evpn-es)# bgp route-target 1212.0000.0002

/* Configure address family session in BGP. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router#(config)# router bgp 200
RP/0/RSP0/CPU0:router#(config-bgp)# bgp router-id 209.165.200.227
RP/0/RSP0/CPU0:router#(config-bgp)# address-family l2vpn evpn
RP/0/RSP0/CPU0:router#(config-bgp)# neighbor 10.10.10.10
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# remote-as 200
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# description MPLSFACING-PEER
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# update-source Loopback 0
RP/0/RSP0/CPU0:router#(config-bgp-nbr)# address-family l2vpn evpn

```

確認

シングルアクティブモードのデュアルホームデバイスのEVPNを確認します。

```

RP/0/RSP0/CPU0:router# show evpn ethernet-segment int bundleEther 21 carving detail

...
Ethernet Segment Id      Interface      Nexthops
-----
0012.1200.0000.0000.0002  BE21          10.10.10.10  209.165.201.1

ESI type : 0
Value : 12.1200.0000.0000.0002
ES Import RT : 1212.0000.0000 (from ESI)

Source MAC : 0000.0000.0000 (N/A)
Topology :
Operational : MHN
Configured : Single-active (AaPS)
Primary Services : Auto-selection
Secondary Services: Auto-selection

Service Carving Results:
Forwarders : 2
Elected : 1
EVI E : 500
Not Elected : 1
EVI NE : 501

```

EVPN ソフトウェア MAC ラーニングの確認

パケット ドロップ統計情報を確認します。

```
RP/0/RSP0/CPU0:router# show l2vpn bridge-domain bd-name EVPN_2001 details
```

```
Bridge group: EVPN_ALL_ACTIVE, bridge-domain: EVPN_2001, id: 1110,
state: up, ShgId: 0, MSTi: 0
List of EVPNs:
EVPN, state: up
evi: 2001
XC ID 0x80000458
Statistics:
packets: received 28907734874 (unicast 9697466652), sent
76882059953
bytes: received 5550285095808 (unicast 1861913597184), sent
14799781851396
MAC move: 0
List of ACs:
AC: TenGigE0/4/0/10.2001, state is up
Type VLAN; Num Ranges: 1
...
Statistics:
packets: received 0 (multicast 0, broadcast 0, unknown
unicast 0, unicast 0), sent 45573594908
bytes: received 0 (multicast 0, broadcast 0, unknown unicast
0, unicast 0), sent 8750130222336
MAC move: 0
.....
```

VPN-ID と MAC アドレス フィルタを使用して EVPN ID を確認します。

```
RP/0/RSP0/CPU0:router# show evpn evi vpn-id 2001 neighbor
```

```
Neighbor IP      vpn-id
-----
209.165.200.225  2001
209.165.201.30   2001
```

BGP L2VPN EVPN の概要を確認します。

```
RP/0/RSP0/CPU0:router# show bgp l2vpn evpn summary
```

```
...
Neighbor      Spk  AS      MsgRcvd  MsgSent  TblVer  InQ  OutQ  Up/Down  St/PfxRcd
209.165.200.225  0    200     216739  229871   200781341  0    0      3d00h   348032
209.165.201.30  0    200     6462962 4208831  200781341  10   0      2d22h   35750
```

ラインカードの L2FIB テーブルへの MAC の更新を確認します。

```
RP/0/RSP0/CPU0:router# show l2vpn mac mac all location 0/6/cPU0
```

```
Topo ID Producer Next Hop(s)      Mac Address      IP Address
-----
1112      0/6/CPU0 Te0/6/0/1.36001 00a3.0001.0001
```

ルート スイッチ プロセッサ (RSP) の L2FIB テーブルへの MAC の更新を確認します。

```
RP/0/RSP0/CPU0:router# show l2vpn mac mac all location 0/6/CPU0
```

```

Topo ID  Producer Next Hop(s)      Mac Address      IP Address
-----  -
1112     0/6/CPU0 Te0/6/0/1.36001 00a3.0001.0001

```

MAC アドレスの概要情報を確認します。

```
RP/0/RSP0/CPU0:router# show l2vpn forwarding bridge-domain EVPN_ALL_ACTIVE:EVPN_2001
mac-address location 0/6/CPU0
```

```

.....
Mac Address      Type      Learned from/Filtered on  LC learned  Resync Age/Last Change
Mapped to
0000.2001.5555  dynamic  Te0/0/0/2/0.2001        N/A         11 Jan 14:37:22
N/A <-- local dynamic
00bb.2001.0001  dynamic  Te0/0/0/2/0.2001        N/A         11 Jan 14:37:22
N/A
0000.2001.1111  EVPN     BD id: 1110              N/A         N/A
N/A <-- remote static
00a9.2002.0001  EVPN     BD id: 1110              N/A         N/A
N/A

```

VPN-ID と MAC アドレス フィルタを使用して EVPN ID を確認します。

```
RP/0/RSP0/CPU0:router# show evpn evi vpn-id 2001 mac
```

```

EVI      MAC address      IP address      Nexthop      Label
----      -
2001     00a9.2002.0001  ::              10.10.10.10  34226        <-- Remote MAC
2001     00a9.2002.0001  ::              209.165.201.30 34202
2001     0000.2001.5555  20.1.5.55      TenGigE0/0/0/2/0.2001 34203        <-- local MAC

```

```
RP/0/RSP0/CPU0:router# RP/0/RSP0/CPU0:router# show evpn evi vpn-id 2001 mac 00a9.2002.0001
detail
```

```

EVI      MAC address      IP address      Nexthop      Label
----      -
2001     00a9.2002.0001  ::              10.10.10.10  34226
2001     00a9.2002.0001  ::              209.165.201.30 34202

```

```
Ethernet Tag : 0
```

```
Multi-paths Resolved : True <-- aliasing to two remote PE with All-Active load balancing
```

```
Static : No
```

```
Local Ethernet Segment : N/A
```

```
Remote Ethernet Segment : 0100.211b.fce5.df00.0b00
```

```
Local Sequence Number : N/A
```

```
Remote Sequence Number : 0
```

```
Local Encapsulation : N/A
```

```
Remote Encapsulation : MPLS
```

EVPNに関連付けられているBGPルートをブリッジドメインフィルタを使用して確認します。

```

RP/0/RSP0/CPU0:router# show bgp l2vpn evpn bridge-domain EVPN_2001 route-type 2
* > [2] [0] [48] [00bb.2001.0001] [0]/104
      0.0.0.0                0 i <----- locally learnt MAC
* > i [2] [0] [48] [00a9.2002.00be] [0]/104
      10.10.10.10 100        0 i <----- remotely learnt MAC
* i 209.165.201.30 100 0 i

```

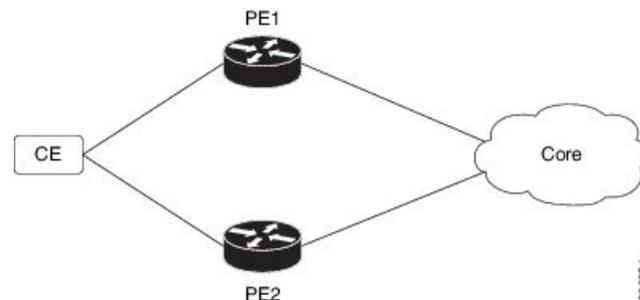
EVPN アウトオブサービス

EVPN アウトオブサービス機能では、Link Aggregation Control Protocol (LACP) を設定したイーサネットセグメントに含まれているバンドルインターフェイスの状態を制御することができます。この機能を使用すると、ノードをアウトオブサービス (OOS) に移行させることができます。プロバイダーエッジ (PE) のすべてのバンドルを手動でシャットダウンする必要はありません。

`cost-out` コマンドを使用してノード上のイーサネット VPN (EVPN) のイーサネットセグメントに属するすべてのバンドルインターフェイスをダウンさせます。イーサネット A-D のイーサネットセグメント (ES EAD) ルートは、バンドルをシャットダウンする前に撤回されます。PE は接続されているカスタマーエッジ (CE) デバイスにシグナリングし、対応するバンドルメンバーをダウンさせます。こうすることで、トラフィックを中断させることなく、トラフィックをこの PE ノードからそらします。CE からイーサネットセグメントへのトラフィックは、マルチホーミング環境内のピア PE へと方向付けられます。

次に、CE が PE1 と PE2 に接続されているトポロジを示します。PE1 に `cost-out` コマンドを設定すると、イーサネットセグメント上のすべてのバンドルインターフェイスがダウンします。また、CE 上の対応するバンドルメンバーもダウンします。したがって、このイーサネットセグメントのトラフィックは CE から PE2 へと送信されるようになります。

図 6: EVPN アウトオブサービス



ノードをサービス状態に戻すには、`no cost-out` コマンドを使用します。これにより、PE 上の EVPN イーサネットセグメントに属するすべてのバンドルインターフェイスと CE 上の対応するバンドルメンバーが起動します。

ノードがコストアウト状態にある場合に新しいバンドルイーサネットセグメントを追加するとそのバンドルがダウンします。同様に、バンドルイーサネットセグメントを削除するとそのバンドルは起動します。

リロード時に指定した時間が経過した後にノードをサービス状態に戻すには、`startup-cost-in` コマンドを使用します。EVPN が初期化された時点でノードがコストアウトになり、設定時間までコストアウト状態が維持されます。タイマー実行中に `evpn no startup-cost-in` コマンドを実行すると、タイマーが停止し、ノードがコストイン状態になります。

「cost-out」設定は「startup-cost-in」タイマーよりも常に優先されます。そのため、両方の設定でリロードすると、コストアウト状態は「cost-out」設定で制御されます。タイマーは関係ありません。同様に、起動タイマーでリロードし、タイマーが実行している間に「cost-out」を設定するとタイマーが停止し、OOS 状態は「cost-out」設定のみで制御されます。

`startup-cost-in timer` が実行している間に何らかのプロシージャを実行すると、ノードはコストアウト状態を維持し、タイマーが再起動します。

EVPN アウトオブサービスの設定

この項では、EVPN アウトオブサービスを設定する方法について説明します。

```
/* Configuring node cost-out on a PE */

Router# configure
Router(config)# evpn
Router(config-evpn)# cost-out
Router(config-evpn) commit

/* Bringing up the node into service */

Router# configure
Router(config)# evpn
Router(config-evpn)# no cost-out
Router(config-evpn) commit

/* Configuring the timer to bring up the node into service after the specified time on
reload */

Router# configure
Router(config)# evpn
Router(config-evpn)# startup-cost-in 6000
Router(config-evpn) commit
```

実行コンフィギュレーション

```
configure
evpn
  cost-out
!

configure
evpn
  startup-cost-in 6000
!
```

確認

EVPN アウト オブ サービスの設定を確認します。

```
/* Verify the node cost-out configuration */

Router# show evpn summary
Fri Apr 7 07:45:22.311 IST
Global Information
-----
Number of EVIs : 2
Number of Local EAD Entries : 0
Number of Remote EAD Entries : 0
Number of Local MAC Routes : 0
Number of Local MAC Routes : 5
      MAC : 5
      MAC-IPv4 : 0
      MAC-IPv6 : 0
Number of Local ES:Global MAC : 12
Number of Remote MAC Routes : 7
      MAC : 7
      MAC-IPv4 : 0
      MAC-IPv6 : 0
Number of Local IMCAST Routes : 56
Number of Remote IMCAST Routes: 56
Number of Internal Labels : 5
Number of ES Entries : 9
Number of Neighbor Entries : 1
EVPN Router ID : 192.168.0.1
BGP Router ID : ::
BGP ASN : 100
PBB BSA MAC address : 0207.1fee.be00
Global peering timer : 3 seconds
Global recovery timer : 30 seconds
EVPN cost-out : TRUE
      startup-cost-in timer : Not configured
```

```
/* Verify the no cost-out configuration */
```

```
Router# show evpn summary
Fri Apr 7 07:45:22.311 IST
Global Information
-----
Number of EVIs : 2
Number of Local EAD Entries : 0
Number of Remote EAD Entries : 0
Number of Local MAC Routes : 0
Number of Local MAC Routes : 5
      MAC : 5
      MAC-IPv4 : 0
      MAC-IPv6 : 0
Number of Local ES:Global MAC : 12
Number of Remote MAC Routes : 7
      MAC : 7
      MAC-IPv4 : 0
      MAC-IPv6 : 0
Number of Local IMCAST Routes : 56
Number of Remote IMCAST Routes: 56
Number of Internal Labels : 5
Number of ES Entries : 9
Number of Neighbor Entries : 1
EVPN Router ID : 192.168.0.1
```

```

BGP Router ID      : ::
BGP ASN            : 100
PBB BSA MAC address : 0207.1fee.be00
Global peering timer : 3 seconds
Global recovery timer : 30 seconds
EVPN cost-out      : FALSE
startup-cost-in timer : Not configured

/* Verify the startup-cost-in timer configuration */

Router# show evpn summary
Fri Apr 7 07:45:22.311 IST
Global Information
-----
Number of EVIs : 2
Number of Local EAD Entries : 0
Number of Remote EAD Entries : 0
Number of Local MAC Routes : 0
Number of Local MAC Routes : 5
MAC : 5
MAC-IPv4 : 0
MAC-IPv6 : 0
Number of Local ES:Global MAC : 12
Number of Remote MAC Routes : 7
MAC : 7
MAC-IPv4 : 0
MAC-IPv6 : 0
Number of Local IMCAST Routes : 56
Number of Remote IMCAST Routes: 56
Number of Internal Labels : 5
Number of ES Entries : 9
Number of Neighbor Entries : 1
EVPN Router ID : 192.168.0.1
BGP Router ID : ::
BGP ASN : 100
PBB BSA MAC address : 0207.1fee.be00
Global peering timer : 3 seconds
Global recovery timer : 30 seconds
EVPN node cost-out : TRUE
startup-cost-in timer : 6000

```

EVPN ルーティング ポリシー

EVPN ルーティング ポリシー機能では、アドレスファミリ L2VPN EVPN のルートポリシー サポートを提供します。この機能は、EVPN ルートフィルタリング機能をルーティングポリシー 言語 (RPL) に追加します。フィルタリングはさまざまな EVPN 属性に基づきます。

ピアから受け入れるか、ピアにアダプタイズされる、または1個のルーティングプロトコルから別のプロトコルへ再配布されるときに、ルートを検査し、フィルタリングして、属性を変更するように、ルーティング ポリシーがルータに指示します。

この機能により、より粒度が高いルートポリシーの定義を提供するルートポリシー一致基準の EVPN ルートタイプ 1 ~ 5 の EVPN ネットワーク層到達可能性情報 (NLRI) 属性を使用して ルートポリシーを設定できます。たとえば、ルートポリシーを特定の EVPN ルートタイプのみ に適用したり、任意の組み合わせの EVPN NLRI 属性に適用できます。この機能は、ルートポ

ポリシーを有効にして EVPN NLRI 属性でフィルタリングすることで、ソリューションの設定および展開に柔軟性をもたらしめます。

この機能を実装するには、次の概念を理解する必要があります。

- ルーティング ポリシー言語
- ルーティング ポリシー言語の構造
- ルーティング ポリシー言語コンポーネント
- ルーティング ポリシー言語使用方法
- ポリシー定義
- パラメータ化
- ポリシー適用のセマンティック
- ポリシー ステートメント
- 接続点

これらの概念については「[ルーティング ポリシーの実装](#)」を参照してください。

現在、この機能は接続ポイント「イン」または「アウト」の BGP ネイバーでのみサポートされています。ルート ポリシーは BGP ネイバーのインバウンドまたはアウトバウンドのみに適用できます。

EVPN ルートタイプ

EVPN NLRI には次のさまざまなルートタイプがあります。

ルートタイプ 1: イーサネット自動検出 (AD) ルート

イーサネット (AD) ルートは、EVI ごととイーサネットセグメント識別子 (ESI) ごとにアドバタイズされます。これらのルートは、イーサネットセグメント (ES) ごとに送信されます。これらは ES に属している EVI のリストを伝送します。ESI フィールドは、CE がシングルホームの場合はゼロに設定されます。

イーサネット A-D ルートタイプ固有の EVPN NLRI は次のフィールドで構成されます。

```

+-----+
|Route Type (1 octet)          |*
+-----+
|Length (1 octet)             |
+-----+
|Route Distinguisher (RD) (8 octets) |*
+-----+
|Ethernet Segment Identifier (10 octets) |*
+-----+
|Ethernet Tag ID (4 octets)      |*
+-----+
|MPLS Label (3 octets)         |
+-----+

```

NLRI の形式 : ルートタイプ 1 :

[Type] [Len] [RD] [ESI] [ETag] [MPLS Label]

ネット属性 : [Type] [RD] [ESI] [ETag]

パス属性 : [MPLS Label]

例

```

route-policy evpn-policy
  if rd in (1.1.1.1:0) [and/or evpn-route-type is 1] [and/or esi in
(0a1.a2a3.a4a5.a6a7.a8a9)] [and/or etag is 4294967295] then
    set ..
  endif
end-policy
!
route-policy evpn-policy
  if rd in (1.1.1.2:0) [and/or evpn-route-type is 1] [and/or esi in
(00a1.a2a3.a4a5.a6a7.a8a9)] [and/or etag is 4294967295] then
    set ..
  endif
end-policy

```

ルートタイプ 2 : MAC/IP アドバタイズメント ルート

ホストの IP アドレスと MAC アドレスが NLRI 内のピアにアドバタイズされます。MAC アドレスのコントロールプレーン学習は不明ユニキャストのフラッドを削減します。

MAC/IP アドバタイズメント ルート タイプ固有の EVPN NLRI は次のフィールドで構成されます。

Route Type (1 octet)	*
Length (1 octet)	
RD (8 octets)	*
Ethernet Segment Identifier (10 octets)	
Ethernet Tag ID (4 octets)	*
MAC Address Length (1 octet)	*
MAC Address (6 octets)	*
IP Address Length (1 octet)	*
IP Address (0, 4, or 16 octets)	*
MPLS Label1 (3 octets)	
MPLS Label2 (0 or 3 octets)	

000306

NLRI の形式 : ルートタイプ 2 :

[Type][Len][RD][ESI][ETag][MAC Addr Len][MAC Addr][IP Addr Len][IP Addr][MPLS Label1][MPLS Label2]

ネット属性 : [Type][RD][ETag][MAC Addr Len][MAC Addr][IP Addr Len][IP Addr]

パス属性 : [ESI], [MPLS Label1], [MPLS Label2]

例

```
route-policy evpn-policy
  if rd in (1.1.1.2:0) [and/or evpn-route-type is 2] [and/or esi in
(0000.0000.0000.0000.0000)] [and/or etag is 0] [and/or macaddress in (0013.aabb.ccdd)]
[and/or destination in (1.2.3.4/32)] then
    set ..
  endif
end-policy
```

ルートタイプ 3 : 包括的なマルチキャストイーサネットタグルート

このルートは、送信元 PE からリモート PE へのブロードキャスト、不明ユニキャスト、およびマルチキャスト (BUM) トラフィック用の接続を確立します。このルートは、VLAN ごとと ESI ごとにアドバタイズされます。

包括的マルチキャストイーサネットタグルートタイプ固有のEVPN NLRIは次のフィールドで構成されます。

Route Type (1 octet)	*
Length (1 octet)	
RD (8 octets)	*
Ethernet Tag ID (4 octets)	*
IP Address Length (1 octet)	*
Originating Router's IP Address (4 or 16 octets)	*

NLRI の形式 : ルートタイプ 3 :

[Type][Len][RD][ETag][IP Addr Len][Originating Router's IP Addr]

ネット属性 : [Type][RD][ETag][IP Addr Len][Originating Router's IP Addr]

例

```
route-policy evpn-policy
  if rd in (1.1.1.1:300) [and/or evpn-route-type is 3] [and/or etag is 0] [and/or
evpn-originator in (1.1.1.1)] then
    set ..
  endif
end-policy
```

ルートタイプ 4 : イーサネットセグメントルート

イーサネットセグメントルートではCEデバイスを2台のデバイスまたはPEデバイスを接続できます。ESルートでは同じイーサネットセグメントに接続されているPEデバイスを検出できます。

イーサネットセグメントルートタイプ固有のEVPNNLRIは次のフィールドで構成されます。

```

+-----+
|Route Type (1 octet)          |*
+-----+
|Length (1 octet)             |
+-----+
|RD (8 octets)                 |*
+-----+
|Ethernet Segment Identifier (10 octets)|*
+-----+
|IP Address Length (1 octet)   |*
+-----+
|Originating Router's IP Address |*
|(4 or 16 octets)             |
+-----+

```

3-603-08

NLRI の形式 : ルートタイプ 4 :

[Type][Len][RD][ESI][IP Addr Len][Originating Router's IP Addr]

ネット属性 : [Type][RD][ESI][IP Addr Len][Originating Router's IP Addr]

例

```

route-policy evpn-policy
  if rd in (1.1.1.1:0) [and/or evpn-route-type is 4] [and/or esi in
(00a1.a2a3.a4a5.a6a7.a8a9)] [and/or evpn-originator in (1.1.1.1)] then
    set ..
  endif
end-policy

```

ルートタイプ 5 : IP プレフィックスルート

IP プレフィックスルート タイプ固有の EVPN NLRI は次のフィールドで構成されます。

Route Type (1 octet)	*
Length (1 octet)	
RD (8 octets)	*
Ethernet Segment Identifier (10 octets)	
Ethernet Tag ID (4 octets)	*
IP Address Length (1 octet)	*
IP Address (4 or 16 octets)	*
GW IP Address (4 or 16 octets)	
MPLS Label (3 octets)	

NLRI の形式 : ルートタイプ 5 :

[Type][Len][RD][ESI][ETag][IP Addr Len][IP Addr][GW IP Addr][Label]

ネット属性 : [Type][RD][ETag][IP Addr Len][IP Addr]

パス属性 : [ESI], [GW IP Addr], [Label]

例

```
route-policy evpn-policy
  if rd in (30.30.30.30:1) [and/or evpn-route-type is 5] [and/or esi in
(0000.0000.0000.0000.0000)] [and/or etag is 0] [and/or destination in (12.2.0.0/16)]
[and/or evpn-gateway in (0.0.0.0)] then
    set ..
  endif
end-policy
```

EVPN RPL 属性

ルート識別子

ルート識別子 (rd) 属性は、8 オクテットで構成されます。rd は EVPN ルートのタイプそれぞれに指定できます。この属性は、ルートポリシーでは必須ではありません。

例

```
rd in (1.2.3.4:0)
```

EVPN ルート タイプ

EVPN ルートタイプ属性は、1 オクテットで構成されます。これによって EVPN ルートタイプが指定されます。EVPN ルートタイプ属性は、特定の EVPN NLRI プレフィックス形式を識別するために使用されます。これは、すべての EVPN ルートタイプのネット属性の 1 つです。

例

```
evpn-route-type is 3
```

The following are the various EVPN route types that can be used:

- 1 - ethernet-ad
- 2 - mac-advertisement
- 3 - inclusive-multicast
- 4 - ethernet-segment
- 5 - ip-advertisement

IP プレフィックス

IP プレフィックス属性は、それぞれ 4 つの部分（アドレス、マスク長、最小一致長、最大一致長）がある IPv4 または IPv6 プレフィックス一致指定を保持しています。アドレスは必須ですが、他の 3 つの部分は任意です。EVPN ルートタイプ 2 での IP プレフィックスの指定により、IPv4 または IPv6 のいずれかのホスト IP アドレスを表します（/32 または /128）。EVPN ルートタイプ 5 の IP プレフィックスでの指定により、IPv4 または IPv6 のサブネットを表します。これは、EVPN ルート 2 と 5 のネット属性の 1 つです。

例

```
destination in (128.47.10.2/32)
destination in (128.47.0.0/16)
destination in (128:47::1/128)
destination in (128:47::0/112)
```

esi

イーサネットセグメント識別子（ESI）属性は、10 オクテットで構成されます。これは EVPN ルートタイプ 1 と 4 のネット属性であり、EVPN ルートタイプ 2 と 5 のパス属性です。

例

```
esi in (ffff.ffff.ffff.ffff.fff0)
```

etag

イーサネットタグ属性は 4 オクテットで構成されます。イーサネットタグは、特定のブロードキャストドメイン（VLAN など）を識別します。EVPN インスタンスは 1 つまたは複数のブ

ロードキャスト ドメインで構成されます。これは EVPN ルート タイプ 1、2、3、および 5 の ネット属性です。

例

```
etag in (10000)
```

mac

MAC 属性は 6 オクテットで構成されます。これは、EVPN ルート 2 の ネット属性です。

例

```
mac in (0206.acb1.e806)
```

evpn-originator

evpn-originator 属性は、発信元ルータの IP アドレス（4 または 16 オクテット）を指定します。これは、EVPN ルート 3 と 4 の ネット属性です。

例

```
evpn-originator in (1.2.3.4)
```

evpn-gateway

evpn-gateway 属性は、ゲートウェイの IP アドレスを指定します。ゲートウェイ IP アドレスは 32 ビットまたは 128 ビットのフィールド（IPv4 または IPv6）であり、IP プレフィックスに応じてオーバーレイ ネクストホップをエンコードします。ゲートウェイ IP アドレス フィールドは、オーバーレイ ネクストホップとして使用しない場合はゼロに設定できます。これは、EVPN ルート 5 の パス属性です。

例

```
evpn-gateway in (1.2.3.4)
```

EVPN RPL 属性セット

このコンテキストでは、セットという用語を、順序付けのない固有のエレメントの集合を意味する数学的な概念で使用されます。ポリシー言語は、セットをマッチング用の値のグループに対するコンテナとして提供します。セットは、条件式で使用されます。セットの要素はカンマで区切ります。ヌル（空）のセットは許可されます。

prefix-set

prefix-set は、それぞれ 4 つの部分（アドレス、マスク長、最小一致長、最大一致長）がある IPv4 または IPv6 プレフィックス一致指定を保持しています。アドレスは必須ですが、他の 3 つの部分は任意です。prefix-set は 1 つまたは複数の IP プレフィックスを指定します。

例

```
prefix-set ip_prefix_set
14.2.0.0/16,
54.0.0.0/16,
12.12.12.0/24,
50:50::1:0/112
end-set
```

mac-set

mac-set は 1 つまたは複数の MAC プレフィックスを指定します。

例

```
mac-set mac_address_set
1234.2345.6789,
2345.3456.7890
end-set
```

esi-set

esi-set は、1 つまたは複数の ESI を指定します。

例

```
esi-set evpn_esi_set
1234.2345.3456.4567.5678,
1234.2345.3456.4567.5670
end-set
```

etag-set

etag-set は、1 つまたは複数のイーサネット タグを指定します。

例

```
etag-set evpn_etag_set
10000,
20000
end-set
```

EVPN RPL 機能の設定

次の項では、mac-set、esi-set、evpn-gateway、および evpn-originator を設定する方法について説明します。

```
/* Configuring a mac-set and referring it in a route-policy (Attach point - neighbor-in)
*/
Router# configure
Router(config)# mac-set demo_mac_set
Router(config-mac)# 1234.ffff.aaa3,
Router(config-mac)# 2323.4444.ffff
Router(config-mac)# end-set
Router(config)# !
Router(config)# route-policy policy_use_pass_mac_set
Router(config-rpl)# if mac in demo_mac_set then
Router(config-rpl-if)# set med 200
Router(config-rpl-if)# else
Router(config-rpl-else)# set med 1000
Router(config-rpl-else)# endif
Router(config-rpl)# end-policy
Router(config)# commit
Router(config)# router bgp 100
Router(config-bgp)# address-family ipv4 unicast
Router(config-bgp-af)# !
Router(config-bgp-af)# neighbor 10.0.0.10
Router(config-bgp-nbr)# remote-as 8
Router(config-bgp-nbr)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# route-policy policy_use_pass_mac_set in
Router(config-bgp-nbr-af)# commit

/* Configuring a esi-set and referring it in a route-policy (Attach point - neighbor-in)
*/
Router# configure
Router(config)# esi-set demo_esi
Router(config-esi)# ad34.1233.1222.ffff.44ff,
Router(config-esi)# ad34.1233.1222.ffff.6666
Router(config-esi)# end-set
Router(config)# !
Router(config)# route-policy use_esi
Router(config-rpl)# if esi in demo_esi then
Router(config-rpl-if)# set local-preference 100
Router(config-rpl-if)# else
Router(config-rpl-else)# set local-preference 300
Router(config-rpl-else)# endif
Router(config-rpl)# end-policy
Router(config)# commit

/* Configuring evpn-gateway/evpn-originator in a route-policy (Attach point - neighbor-in
and out) */
Router# configure
Router(config)# route-policy gateway_demo
Router(config-rpl)# if evpn-gateway in (10.0.0.0/32) then
Router(config-rpl-if)# pass
Router(config-rpl-if)# endif
Router(config-rpl)# end-policy
Router(config)# commit
Router(config)# route-policy originator_demo
Router(config-rpl)# if evpn-originator in (10.0.0.1/32) then
Router(config-rpl-if)# set local-preference 100
Router(config-rpl-if)# else
```

```
Router(config-rpl-else)# set med 200
Router(config-rpl-else)# endif
Router(config-rpl)# end-policy
Router(config)# commit
Router(config)# router bgp 100
Router(config-bgp)# address-family ipv4 unicast
Router(config-bgp-af)# !
Router(config-bgp-af)# neighbor 10.0.0.10
Router(config-bgp-nbr)# remote-as 8
Router(config-bgp-nbr)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# route-policy gateway_demo in
Router(config-bgp-nbr-af)# route-policy originator_demo out
Router(config-bgp-nbr-af)# commit
```

実行コンフィギュレーション

```
/* Configuring a mac-set and refering it in a route-policy (Attach point - neighbor-in)
*/
mac-set demo_mac_set
  1234.ffff.aaa3,
  2323.4444.ffff
end-set
!
route-policy policy_use_pass_mac_set
  if mac in demo_mac_set then
    set med 200
  else
    set med 1000
  endif
end-policy
!
router bgp 100
  address-family ipv4 unicast
  !
  neighbor 10.0.0.10
    remote-as 8
    address-family ipv4 unicast
    route-policy policy_use_pass_mac_set in
  !
  !
end

/* Configuring a esi-set and refering it in a route-policy (Attach point - neighbor-in)
*/
Wed Oct 26 11:52:23.720 IST
esi-set demo_es
  ad34.1233.1222.ffff.44ff,
  ad34.1233.1222.ffff.6666
end-set
!
route-policy use_es
  if esi in demo_es then
    set local-preference 100
  else
    set local-preference 300
  endif
end-policy
```

EVPN ルート ポリシーの例

```
route-policy ex_2
  if rd in (2.2.18.2:1004) and evpn-route-type is 1 then
    drop
  elseif rd in (2.2.18.2:1009) and evpn-route-type is 1 then
    drop
  else
    pass
  endif
end-policy
!
route-policy ex_3
  if evpn-route-type is 5 then
    set extcommunity bandwidth (100:9999)
  else
    pass
  endif
end-policy
!
route-policy samp
end-policy
!
route-policy samp1
  if rd in (30.0.101.2:0) then
    pass
  endif
end-policy
!
route-policy samp2
  if rd in (30.0.101.2:0, 1:1) then
    pass
  endif
end-policy
!
route-policy samp3
  if rd in (*:*) then
    pass
  endif
end-policy
!
route-policy samp4
  if rd in (30.0.101.2:*) then
    pass
  endif
end-policy
!
route-policy samp5
  if evpn-route-type is 1 then
    pass
  endif
end-policy
!
route-policy samp6
  if evpn-route-type is 2 or evpn-route-type is 5 then
    pass
  endif
end-policy
!
route-policy samp7
  if evpn-route-type is 4 or evpn-route-type is 3 then
    pass
```

```
endif
end-policy
!
route-policy samp8
  if evpn-route-type is 1 or evpn-route-type is 2 or evpn-route-type is 3 then
    pass
  endif
end-policy
!
route-policy samp9
  if evpn-route-type is 1 or evpn-route-type is 2 or evpn-route-type is 3 or
  evpn-route-type is 4 then
    pass
  endif
end-policy
!
route-policy test1
  if evpn-route-type is 2 then
    set next-hop 10.2.3.4
  else
    pass
  endif
end-policy
!
route-policy test2
  if evpn-route-type is 2 then
    set next-hop 10.10.10.10
  else
    drop
  endif
end-policy
!
route-policy test3
  if evpn-route-type is 1 then
    set tag 9988
  else
    pass
  endif
end-policy
!
route-policy samp21
  if mac in (6000.6000.6000) then
    pass
  endif
end-policy
!
route-policy samp22
  if extcommunity rt matches-any (100:1001) then
    pass
  else
    drop
  endif
end-policy
!
route-policy samp23
  if evpn-route-type is 1 and esi in (aaaa.bbbb.cccc.dddd.eeee) then
    pass
  else
    drop
  endif
end-policy
!
route-policy samp24
  if evpn-route-type is 5 and extcommunity rt matches-any (100:1001) then
```

```
        pass
      else
        drop
      endif
    end-policy
  !
  route-policy samp25
    if evpn-route-type is 2 and esi in (1234.1234.1234.1234.1236) then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp26
    if etag in (20000) then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp27
    if destination in (99.99.99.1) and etag in (20000) then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp31
    if evpn-route-type is 1 or evpn-route-type is 2 or evpn-route-type is 3 or
    evpn-route-type is 4 or evpn-route-type is 5 then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp33
    if esi in evpn_esi_set1 then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp34
    if destination in (90:1:1::9/128) then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp35
    if destination in evpn_prefix_set1 then
      pass
    else
      drop
    endif
  end-policy
  !
  route-policy samp36
```

```
    if evpn-route-type is 3 and evpn-originator in (80:1:1::3) then
        pass
    else
        drop
    endif
end-policy
!
route-policy samp37
    if evpn-gateway in (10:10::10) then
        pass
    else
        drop
    endif
end-policy
!
route-policy samp38
    if mac in evpn_mac_set1 then
        pass
    else
        drop
    endif
end-policy
!
route-policy samp39
    if mac in (6000.6000.6002) then
        pass
    else
        drop
    endif
end-policy
!
route-policy samp41
    if evpn-gateway in (10.10.10.10, 10:10::10) then
        pass
    else
        drop
    endif
end-policy
!
route-policy samp42
    if evpn-originator in (24.162.160.1/32, 70:1:1::1/128) then
        pass
    else
        drop
    endif
end-policy
!
route-policy example
    if rd in (62300:1903) and evpn-route-type is 1 then
        drop
    elseif rd in (62300:19032) and evpn-route-type is 1 then
        drop
    else
        pass
    endif
end-policy
!
route-policy samp100
    if evpn-route-type is 4 or evpn-route-type is 5 then
        drop
    else
        pass
    endif
end-policy
```

```
!  
route-policy samp101  
  if evpn-route-type is 4 then  
    drop  
  else  
    pass  
  endif  
end-policy  
!  
route-policy samp102  
  if evpn-route-type is 4 then  
    drop  
  elseif evpn-route-type is 5 then  
    drop  
  else  
    pass  
  endif  
end-policy  
!  
route-policy samp103  
  if evpn-route-type is 2 and destination in evpn_prefix_set1 then  
    drop  
  else  
    pass  
  endif  
end-policy  
!  
route-policy samp104  
  if evpn-route-type is 1 and etag in evpn_etag_set1 then  
    drop  
  elseif evpn-route-type is 2 and mac in evpn_mac_set1 then  
    drop  
  elseif evpn-route-type is 5 and esi in evpn_esi_set1 then  
    drop  
  else  
    pass  
  endif  
end-policy  
!
```