仮想プライベート LAN サービス（VPLS）および VPLS BGP ベースの自動検出の設定

- VPLS の設定 （1 ページ）
- VPLS BGP ベースの自動検出の設定 （12 ページ）

VPLS の設定

以下のセクションでは、VPLS の設定方法について説明します。

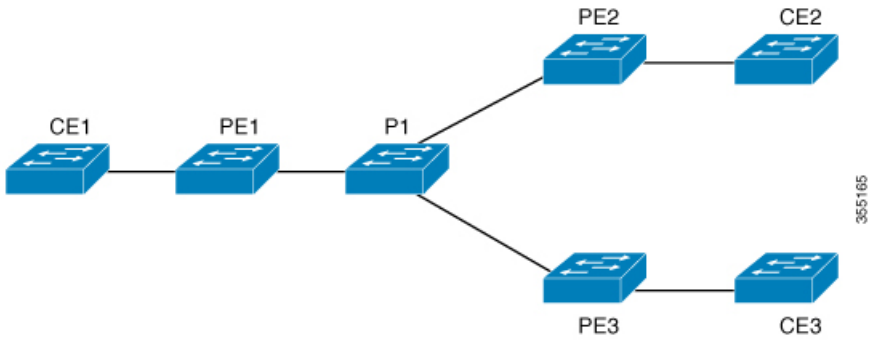
VPLS について

VPLS の概要

VPLS（仮想プライベート LAN サービス）により、企業では、サービス プロバイダーから提供されたインフラストラクチャを解して、複数のサイトからのイーサネット ベースの LAN をまとめてリンクすることが可能になります。企業の側からは、サービス プロバイダーのパブリック ネットワークは、1 つの大きなイーサネット LAN のように見えます。サービス プロバイダーからすると、VPLS は、大規模な設備投資なしで、既存のネットワーク上に収益を生み出す新たなサービスを導入するチャンスになります。オペレータは、ネットワークでの機器の運用年数を延長できます。

Virtual Private LAN Service（VPLS）は、プロバイダーコアを使用して複数の接続回線を 1 つにまとめ、複数の接続回線をまとめて接続する仮想ブリッジをシミュレートします。VPLS のトポロジは、カスタマーからは認識されません。すべての CE デバイスは、プロバイダー コアによってエミュレートされた論理ブリッジに接続されているように見えます。

図 *1*: *VPLS* トポロジ



フルメッシュの設定

フルメッシュの設定では、VPLS に参加するすべての PE 間でトンネル ラベル スイッチド パス（LSP）のフルメッシュが必要です。フルメッシュでは、シグナリングのオーバーヘッドと、PE 上でプロビジョニング対象の各 VC に対するパケット複製の要件が多くなる場合があります。

VPLS のセットアップは、まず参加する各 PE ルータで Virtual Forwarding Instance（VFI）を作成して行います。VFI によって VPLS ドメインの VPN ID、そのドメインの他の PE デバイスのアドレス、トンネルのシグナリングのタイプ、各ピア PE ルータのカプセル化のメカニズムが指定されます。

エミュレーテッド VC の相互接続で形成される VFI のセットは、VPLS インスタンスと呼ばれます。これは、パケット スイッチド ネットワークを介して論理ブリッジを構成する VPLS インスタンスです。VPLS インスタンスには、一意の VPN ID が割り当てられます。

PE デバイスは、VFI を使用して、エミュレートされた VC から VPLS インスタンスの他のすべての PE デバイスまでのフルメッシュ LPS を確立します。PE デバイスは、Cisco IOS CLI を使用して、スタティック設定を通じた VPLS インスタンスのメンバーシップを取得します。

フル メッシュ設定を行うと、PE ルータは、単一のブロードキャスト ドメインを維持できます。したがって、接続回線でブロードキャスト、マルチキャスト、または未知のユニキャスト パケットを受信すると、PE ルータは、他のすべての接続回線およびその VPLS インスタンスに属する他のすべての CE デバイスへのエミュレート回線にパケットを送信します。CE デバイスでは、VPLS インスタンスを、エミュレート LAN として認識します。

プロバイダー コアでのパケット ループの問題を回避するために、PE デバイスは、エミュレート VC に「スプリット ホライズン」の原則を適用します。つまり、エミュレート VC でパケットを受信した場合、パケットは、他のいずれのエミュレート VC にも転送されません。

VFI を定義したら、CE デバイスへの接続回線にバインドする必要があります。

パケット転送の判断は、特定の VPLS ドメインのレイヤ 2 仮想転送インスタンス（VFI）を検索することによって行われます。

特定の PE ルータの VPLS インスタンスは、特定の物理または論理ポートに着信するイーサネット フレームを受信し、イーサネット スイッチによる動作同様に、MAC テーブルに入力しま

す。PE ルータでは、この MAC アドレスを使用して、リモート サイトにある別の PE ルータに配布するために、このようなフレームを適切な LSP に切り替えることができます。

MAC アドレスが MAC アドレス テーブルにない場合、PE ルータは、イーサネット フレームを複製し、直前に送信された入力ポートを除くその VPLS インスタンスに関連付けられたすべての論理ポートにフラッディングします。PE ルータは、個々のポートでパケットを受信したときに MAC テーブルを更新し、一定期間使用されていないアドレスを削除します。

VPLS の制約事項

- レイヤ 2 プロトコルトンネリングの設定はサポートされていません。
- Integrated Routing and Bridging（IRB）の設定はサポートされていません。
- 明示的 null の仮想回線接続検証（VCCV）ping はサポートされていません。
- スイッチは、ハブとしてではなく、階層型仮想プライベート LAN サービス（VPLS）でスポークとして設定されている場合にのみサポートされます。
- レイヤ 2 VPN インターワーキング機能はサポートされていません。
- **ip unnumbered** コマンドは、マルチプロトコル ラベル スイッチング（MPLS）構成ではサポートされていません。
- フラッドトラフィックの場合、仮想回線（VC）統計情報は、**show mpls l2 vc vcid detail** コマンドの出力に表示されません。
- 接続回線では、Dot1q トンネル構成はサポートされていません。

CE デバイスへのレイヤ 2 PE デバイスインターフェイスの設定

CE デバイスへのレイヤ 2 PE デバイスインターフェイスを設定する必要があります。CE デバイスからのタグ付きトラフィック用に PE デバイスで 802.1Q トランクを設定するか、CE デバイスからのタグなしトラフィック用に PE デバイスで 802.1Q アクセスポートを設定できます。その両方の設定について、以下のセクションで説明します。

CE デバイスからのタグ付きトラフィックを受け取る PE デバイスの 802.1Q トランクの設定

PE デバイスで 802.1Q トランクを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# 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 3	**interface** *interface-id* 例： `Device(config)# interface TenGigabitEthernet1/0/24`	トランクとして設定するインターフェイスを定義し、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。
ステップ 4	**no ip address** *ip_address mask* [**secondary**] 例： `Device(config-if)# no ip address`	IP 処理をディセーブルにして、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。
ステップ 5	**switchport** 例： `Device(config-if)# switchport`	レイヤ 2 スイッチドインターフェイスのスイッチング特性を変更します。
ステップ 6	**switchport trunk encapsulation dot1q** 例： `Device(config-if)# switchport trunk encapsulation dot1q`	スイッチ ポートのカプセル化形式を 802.1Q に設定します。
ステップ 7	**switchport trunk allow vlan** *vlan_ID* 例： `Device(config-if)# switchport trunk allow vlan 2129`	許可 VLAN のリストを設定します。
ステップ 8	**switchport mode trunk** 例： `Device(config-if)# switchport mode trunk`	トランキング VLAN レイヤ 2 インターフェイスへのインターフェイスを設定します。
ステップ 9	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

CE デバイスからのタグなしトラフィックを受け取る PE デバイスの 802.1Q アクセスポートの設定

PE デバイスで 802.1Q アクセスポートを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**interface** *interface-id* 例： `Device(config)# interface TenGigabitEthernet1/0/24`	トランクとして設定するインターフェイスを定義し、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。
ステップ 4	**no ip address** *ip_address mask* [**secondary**] 例： `Device(config-if)# no ip address`	IP 処理をディセーブルにします。
ステップ 5	**switchport** 例： `Device(config-if)# switchport`	レイヤ 2 スイッチドインターフェイスのスイッチング特性を変更します。
ステップ 6	**switchport mode access** 例： `Device(config-if)# switchport mode access`	インターフェイスタイプを、非トランキング、タグなし、シングル VLAN レイヤ 2 インターフェイスとして設定します。
ステップ 7	**switchport access vlan** *vlan_ID* 例： `Device(config-if)# switchport access vlan 2129`	インターフェイスがアクセス モードのときに VLAN を設定します。
ステップ 8	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	`Device(config-if)# end`	

PE デバイスでのレイヤ 2 VLAN インスタンスの設定

PE デバイスにレイヤ 2 VLAN インターフェイスを設定すると、VLAN データベースへの PE デバイス上のレイヤ 2 VLAN インスタンスで、VPLS と VLAN 間のマッピングを設定できます。

PE デバイスでレイヤ 2 VLAN インスタンスを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**vlan** *vlan-id* 例： `Device(config)# vlan 2129`	特定の VLAN を設定します。
ステップ 4	**interface vlan** *vlan-id* 例： `Device(config-vlan)# interface vlan 2129`	この VLAN にインターフェイスを設定します。
ステップ 5	**end** 例： `Device(config-vlan)# end`	特権 EXEC モードに戻ります。

PE デバイス上での MPLS の設定

PE デバイスで MPLS を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**mpls ip** 例： `Device(config)# mpls ip`	MPLS ホップバイホップ転送を設定します。
ステップ 4	**mpls label protocol ldp** 例： `Device(config)# mpls label protocol ldp`	プラットフォームの Label Distribution Protocol（LDP; ラベル配布プロトコル）を指定します。
ステップ 5	**mpls ldp logging neighbor-changes** 例： `Device(config)# mpls ldp logging neighbor-changes`	（任意）ネイバーの変更の記録を指定します。
ステップ 6	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

PE デバイスでの VFI の設定

VFI によって VPLS ドメインの VPN ID、そのドメインの他の PE デバイスのアドレス、トンネルのシグナリングのタイプ、各ピアデバイスのカプセル化のメカニズムが指定されます。

PE デバイスで VFI および関連する VC を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# 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 3	**l2 vfi** *vfi-name* **manual** 例： `Device(config)# l2 vfi 2129 manual`	レイヤ 2 VFI 手動コンフィギュレーション モードをイネーブルにします。
ステップ 4	**vpn id** *vpn-id* 例： `Device(config-vfi)# vpn id 2129`	VPLS ドメインの VPN ID を設定します。このレイヤ 2 Virtual Routing Forwarding（VRF）にバインドされたエミュレート VC でシグナリングにこの VPN ID が使用されます。 （注） *vpn-id* は *vlan-id* と同じです。
ステップ 5	**neighbor** *router-id* {**encapsulation mpls**} 例： `Device(config-vfi)# neighbor remote-router-id encapsulation mpls`	リモートピアリングルータ ID と、エミュレート VC をセットアップするために使用されるトンネルカプセル化タイプまたは疑似回線（PW）プロパティを指定します。
ステップ 6	**end** 例： `Device(config-vfi)# end`	特権 EXEC モードに戻ります。

PE デバイスでの VFI への接続回線の関連付け

VFI を定義したら、1 つ以上の接続回線に関連付ける必要があります。

接続回線を VFI に関連付けるには、次の手順を実行します。

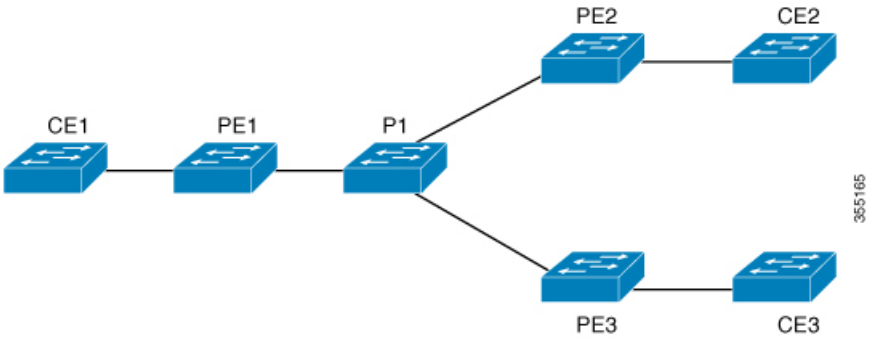
手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# 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 3	**interface vlan** *vlan-id* 例： `Device(config)# interface vlan 2129`	動的なスイッチ仮想インターフェイス（SVI）を作成するか、使用します。 （注）　*vlan-id* は *vpn-id* と同じです。
ステップ 4	**no ip address** 例： `Device(config-if)# no ip address`	IP 処理をディセーブルにします。（IP アドレスを設定する場合は、VLAN のレイヤ 3 インターフェイスを設定できます）。
ステップ 5	**xconnect vfi** *vfi-name* 例： `Device(config-if)# xconnect vfi 2129`	VLAP ポートにバインドするレイヤ 2 VFI を指定します。
ステップ 6	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

VPLS の設定例

図 *2 : VPLS* トポロジ



PE1 の設定	PE2 の設定
pseudowire-class vpls2129 encapsulation mpls ! l2 vfi 2129 manual vpn id 2129 neighbor 44.254.44.44 pw-class vpls2129 neighbor 188.98.89.98 pw-class vpls2129 ! interface TenGigabitEthernet1/0/24 switchport trunk allowed vlan 2129 switchport mode trunk ! interface Vlan2129 no ip address xconnect vfi 2129 !	pseudowire-class vpls2129 encapsulation mpls no control-word ! l2 vfi 2129 manual vpn id 2129 neighbor 1.1.1.72 pw-class vpls2129 neighbor 188.98.89.98 pw-class vpls2129 ! interface TenGigabitEthernet1/0/47 switchport trunk allowed vlan 2129 switchport mode trunk end ! interface Vlan2129 no ip address xconnect vfi 2129 !

show mpls 12transport vc detailコマンドは、仮想回線に関する情報を提示します。

```
Local interface: VFI 2129 vfi up
  Interworking type is Ethernet
  Destination address: 44.254.44.44, VC ID: 2129, VC status: up
    Output interface: Gi1/0/9, imposed label stack {18 17}
    Preferred path: not configured
    Default path: active
    Next hop: 177.77.177.2
  Create time: 19:09:33, last status change time: 09:24:14
    Last label FSM state change time: 09:24:14
  Signaling protocol: LDP, peer 44.254.44.44:0 up
    Targeted Hello: 1.1.1.72(LDP Id) -> 44.254.44.44, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
MPLS VC labels: local 512, remote 17
    Group ID: local n/a, remote 0
    MTU: local 1500, remote 1500
    Remote interface description:
  Sequencing: receive disabled, send disabled
```

```
  Control Word: Off
  SSO Descriptor: 44.254.44.44/2129, local label: 512
  Dataplane:
    SSM segment/switch IDs: 20498/20492 (used), PWID: 2
  VC statistics:
    transit packet totals: receive 0, send 0
    transit byte totals:   receive 0, send 0
    transit packet drops:  receive 0, seq error 0, send 0
```

show l2vpn atom vcは、ATM over MPLS が VC に設定されていることを示します。

```
pseudowire100005 is up, VC status is up PW type: Ethernet
  Create time: 19:25:56, last status change time: 09:40:37
    Last label FSM state change time: 09:40:37
  Destination address: 44.254.44.44 VC ID: 2129
    Output interface: Gi1/0/9, imposed label stack {18 17}
    Preferred path: not configured
    Default path: active
    Next hop: 177.77.177.2
  Member of vfi service 2129
    Bridge-Domain id: 2129
    Service id: 0x32000003
  Signaling protocol: LDP, peer 44.254.44.44:0 up
    Targeted Hello: 1.1.1.72(LDP Id) -> 44.254.44.44, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
     PWid FEC (128), VC ID: 2129
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------
------------------------------
    Label        512                            17
    Group ID     n/a                            0
    Interface

    MTU          1500                           1500
    Control word off                            off
    PW type      Ethernet                       Ethernet
```

```
  VCCV CV type 0x02                               0x02
               LSPV [2]                             LSPV [2]

  VCCV CC type 0x06                               0x06
               RA [2], TTL [3]                      RA [2], TTL [3]
  Status TLV   enabled                            supported
SSO Descriptor: 44.254.44.44/2129, local label: 512
Dataplane:
  SSM segment/switch IDs: 20498/20492 (used), PWID: 2
Rx Counters
  0 input transit packets, 0 bytes
  0 drops, 0 seq err
Tx Counters
  0 output transit packets, 0 bytes
  0 drops
```

VPLS BGP ベースの自動検出の設定

次の項では、VPLS BGP ベースの自動検出の設定方法について説明します。

VPLS BGP ベースの自動検出について

VPLS BGP ベースの自動検出

VPLS 自動検出を使用すると、各仮想プライベート LAN サービス（VPLS）プロバイダー エッジ（PE）デバイスで、同じ VPLS ドメインの一部である他の PE デバイスを検出できます。VPLS 自動検出は、いつ PE デバイスが、いつ VPLS ドメインで追加および削除されたかも追跡します。そのため、VPLS 自動検出を有効にすると、VPLS ドメインを手動で設定したり、PE デバイスが追加または削除されたときに設定をメンテナンスしたりする必要がなくなります。VPLS 自動検出は、ボーダー ゲートウェイ プロトコル（BGP）を使用して、VPLS メンバを検出し、VPLS ドメイン内の擬似回線をセットアップおよび解除します。

BGP では、エンドポイント プロビジョニング情報を保存する際にレイヤ 2 VPN（L2VPN）ルーティング情報ベース（RIB）が使用されます。これは、レイヤ 2 仮想転送インスタンス（VFI）が設定される度に更新されます。プレフィックスおよびパス情報は L2VPN データベースに保存され、ベスト パスが BGP により決定されるようになります。BGP により、アップデート メッセージですべての BGP ネイバーにエンドポイント プロビジョニング情報が配布されるとき、L2VPN ベースのサービスをサポートするために、このエンドポイント情報を使用して擬似回線メッシュが設定されます。

BGP 自動検出のメカニズムにより、VPLS 機能に必要不可欠な L2VPN サービスの設定が簡易化されます。VPLS は、高速イーサネット使用した堅牢でスケーラブルな IP マルチプロトコル ラベル スイッチング（MPLS）ネットワークによる大規模な LAN として、地理的に分散した拠点間を接続することで柔軟なサービスの展開を実現します。

VPLS BGP ベースの自動検出のイネーブル化

VPLS BGP ベースの自動検出を有効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**l2 vfi** *vfi-name* **autodiscovery** 例： `Device(config)# l2 vfi 2128 autodiscovery`	PE デバイス上で VPLS 自動検出を有効にして、L2 VFI 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	**vpn id** *vpn-id* 例： `Device(config-vfi)# vpn id 2128`	VPLS ドメインの VPN ID を設定します。
ステップ 5	**end** 例： `Device(config-vfi)# end`	特権 EXEC モードに戻ります。

VPLS 自動検出を有効にする BGP の設定

VPLS 自動検出を有効にするように BGP を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# 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 3	**router bgp** *autonomous-system-number* 例： `Device(config)# router bgp 1000`	指定したルーティング プロセスのルータ 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	**no bgp default ipv4-unicast** 例： `Device(config-router)# no bgp default ipv4-unicast`	BGP ルーティング プロセスで使用される IPv4 ユニキャスト アドレス ファミリを無効にします。 （注） IPv4 ユニキャスト アドレス ファミリのルーティング情報は、**neighbor remote-as router** コマンドを使用して設定された各 BGP ルーティング セッションに対して、デフォルトでアドバタイズされます。ただし、**neighbor remote-as** コマンドを設定する前に、**no bgp default ipv4-unicast** コマンドを設定した場合は除きます。既存のネイバー コンフィギュレーションは影響されません。
ステップ 5	**bgp log-neighbor-changes** 例： `Device(config-router)# bgp log-neighbor-changes`	BGP ネイバー リセットのロギングを有効にします。
ステップ 6	**neighbor remote-as** { *ip-address* | *peer-group-name* } **remote-as** *autonomous-system-number* 例： `Device(config-router)# neighbor 44.254.44.44 remote-as 1000`	指定された自律システム内のネイバーの IP アドレスまたはピア グループ名を、ローカル デバイスの IPv4 マルチプロトコル BGP ネイバー テーブルに追加します。 • *autonomous-system-number* 引数が、**router bgp** コマンドで指定された自律システム番号と一致する場合、ネイバーは内部ネイバーになります。 • *autonomous-system-number* 引数が、**router bgp** コマンドで指定された自律システム番号と一致

	コマンドまたはアクション	目的
		しない場合、ネイバーは外部ネイバーになります。
ステップ 7	**neighbor** { *ip-address* | *peer-group-name* } **update-source** *interface-type interface-number* 例： `Device(config-router)# neighbor 44.254.44.44 update-source Loopback300`	（任意）ルーティング テーブル アップデートを受信するための特定のソースまたはインターフェイスを選択するようにデバイスを設定します。
ステップ 8	他の BGP ネイバーを設定する場合は、ステップ 6 と 7 を繰り返します。	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 9	**address-family l2vpn** [**vpls**] 例： `Device(config-router)# address-family l2vpn vpls`	レイヤ 2 VPN アドレスファミリを指定し、アドレス ファミリ コンフィギュレーション モードを開始します。 オプションの **vpls** キーワードは、VPLS エンドポイント プロビジョニング情報が BGP ピアに配布されるように指定します。
ステップ 10	**neighbor** { *ip-address* | *peer-group-name* } **activate** 例： `Device(config-router-af)# neighbor 44.254.44.44 activate`	BGP ネイバーとの情報交換を有効にします。
ステップ 11	**neighbor** { *ip-address* | *peer-group-name* } **send-community** { **both** | **standard** | **extended** } 例： `Device(config-router-af)# neighbor 44.254.44.44 send-community both`	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 12	ステップ 10 と 11 を繰り返して、L2VPN アドレスファミリ内の他の BGP ネイバーをアクティブにします。	
ステップ 13	**exit-address-family** 例： `Device(config-router-af)# exit-address-family`	アドレス ファミリ コンフィギュレーション モードを終了し、ルータ コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 14	**end** 例： `Device(config-router)# end`	ルータコンフィギュレーションモードを終了して、特権 EXEC モードに戻ります。

VPLS BGP-AD の設定例

PE の設定

```
router bgp 1000
 bgp log-neighbor-changes
 bgp graceful-restart
 neighbor 44.254.44.44 remote-as 1000
 neighbor 44.254.44.44 update-source Loopback300
!
 address-family l2vpn vpls
  neighbor 44.254.44.44 activate
  neighbor 44.254.44.44 send-community both
 exit-address-family
!
l2 vfi 2128 autodiscovery
 vpn id 2128
interface Vlan2128
 no ip address
 xconnect vfi 2128
!
```

次に、**show platform software fed sw 1 matm macTable vlan 2000** コマンドの出力例を示します。

```
VLAN  MAC               Type      Seq#   macHandle          siHandle
        diHandle           *a_time *e_time   ports
2000  2852.6134.05c8   0X8002    0      0xffbba312c8       0xffbb9ef938
       0x5154             0         0        Vlan2000
2000  0000.0078.9012   0X1       32627  0xffbb665ec8       0xffbb60b198
       0xffbb653f98       300       278448   Port-channel11
2000  2852.6134.0000   0X1       32651  0xffba15e1a8       0xff454c2328
       0xffbb653f98       300       63       Port-channel11
2000  0000.0012.3456   0X2000001 32655  0xffba15c508       0xff44f9ec98
       0x0                300       1        2000:33.33.33.33
Total Mac number of addresses:: 4
*a_time=aging_time(secs)  *e_time=total_elapsed_time(secs)
Type:
MAT_DYNAMIC_ADDR     0x1       MAT_STATIC_ADDR       0x2
MAT_CPU_ADDR         0x4       MAT_DISCARD_ADDR      0x8
MAT_ALL_VLANS        0x10      MAT_NO_FORWARD        0x20
MAT_IPMULT_ADDR      0x40      MAT_RESYNC            0x80
MAT_DO_NOT_AGE       0x100     MAT_SECURE_ADDR       0x200
MAT_NO_PORT          0x400     MAT_DROP_ADDR         0x800
```

```
MAT_DUP_ADDR          0x1000    MAT_NULL_DESTINATION  0x2000
MAT_DOT1X_ADDR        0x4000    MAT_ROUTER_ADDR       0x8000
MAT_WIRELESS_ADDR     0x10000   MAT_SECURE_CFG_ADDR   0x20000
MAT_OPQ_DATA_PRESENT  0x40000   MAT_WIRED_TUNNEL_ADDR 0x80000
MAT_DLR_ADDR          0x100000  MAT_MRP_ADDR          0x200000
MAT_MSRP_ADDR         0x400000  MAT_LISP_LOCAL_ADDR   0x800000
MAT_LISP_REMOTE_ADDR 0x1000000 MAT_VPLS_ADDR         0x2000000
```

次に、**show bgp l2vpn vpls all** コマンドの出力例を示します。

```
BGP table version is 6, local router ID is 222.5.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
  r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
  x best-external, a additional-path, c RIB-compressed,
  t secondary path,
 Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 1000:2128
*>   1000:2128:1.1.1.72/96
                 0.0.0.0                              32768 ?
*>i  1000:2128:44.254.44.44/96
                 44.254.44.44             0    100      0 ?
```

VPLS および VPLS BGP ベースの自動検出の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	VPLS および VPLS BGP ベースの自動検出の設定	VPLSにより、企業は、サービスプロバイダーから提供されるインフラストラクチャを介して、複数サイトからのイーサネットベースのLANをまとめてリンクできます。 VPLS自動検出を使用すると、各 PE デバイスで、同じ VPLS ドメインの一部である他の PE デバイスを検出できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator にアクセスするには、https://cfnng.cisco.com/にアクセスします。

http://www.cisco.com/go/cfn。

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。