Cisco IOS XE Everest 16.6.x（Catalyst 9300 スイッチ）マルチプロトコル ラベル スイッチング（MPLS）コンフィギュレーション ガイド

初版：2017 年 07 月 31 日

最終更新：2017 年 10 月 30 日

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

【注意】シスコ製品をご使用になる前に、安全上の注意（**www.cisco.com/jp/go/safety_warning/**）をご確認ください。本書は、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

目 次

第 1 章

マルチプロトコル ラベル スイッチング（MPLS）の設定

マルチプロトコル ラベル スイッチング

このモジュールでは、マルチプロトコル ラベル スイッチングと Cisco スイッチでの設定方法について説明します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

マルチプロトコル ラベル スイッチングに関する情報

マルチプロトコル ラベル スイッチング（MPLS）は、レイヤ 3（ネットワーク層）ルーティングの実績のある拡張性とレイヤ 2（データ リンク層）スイッチングのパフォーマンスおよび機能を組み合わせたものです。MPLS により、既存のネットワーク インフラストラクチャを犠牲にすることなく、サービスを差別化する機会を提供しながら、ネットワーク使用率の急激な増加の課題に対処できるようになります。MPLS アーキテクチャは柔軟性があり、レイヤ 2 テクノロジーを任意に組み合わせて使用することができます。MPLS のサポートは、すべてのレイヤ 3 プロトコルに対して提供され、今日のネットワークで一般的に提供されているものよりもはるかに優れたスケーリングが可能です。

マルチプロトコル ラベル スイッチングの機能の説明

ラベル スイッチングは、高性能のパケット転送テクノロジーであり、データリンク層（レイヤ 2）スイッチングのパフォーマンスおよびトラフィック管理機能と、ネットワーク層（レイヤ 3）ルーティングの拡張性、柔軟性、およびパフォーマンスが統合されています。

ラベル スイッチング機能

従来のレイヤ 3 転送メカニズムでは、パケットがネットワークを通過するとき、各スイッチがパケットの転送に関連するすべての情報をレイヤ 3 ヘッダーから抽出します。この情報をルーティング テーブル検索のインデックスとして使用して、パケットのネクスト ホップを決定します。

最も一般的なケースでは、ヘッダーで唯一該当するフィールドは宛先アドレス フィールドですが、場合によっては、他のヘッダー フィールドが該当する場合もあります。その結果、ヘッダーの分析はパケットが通過する各スイッチで個別に実行する必要があります。また、各スイッチで複雑なテーブル検索も行う必要があります。

ラベル スイッチングでは、レイヤ 3 ヘッダーの分析が一度だけ実行されます。その後、レイヤ 3 ヘッダーは、ラベルという固定長の非構造化値にマップされます。

複数の異なるヘッダーで常に同じネクスト ホップが選択される場合は、これらのヘッダーを同じラベルにマッピングできます。実際、ラベルは転送等価クラス（つまり、パケットがそれぞれ別のものである可能性はあるが、転送機能によって識別不能な一連のパケット）を表します。

最初のラベル選択は、レイヤ 3 パケット ヘッダーの内容だけに基づいている必要はありません。たとえば、後続ホップでの転送判断はルーティング ポリシーに基づくこともあります。

ラベルを割り当てると、短いラベル ヘッダーがレイヤ 3 パケットの前に追加されます。このヘッダーは、パケットの一部としてネットワークを介して伝送されます。ネットワーク内の各 MPLS スイッチを介する後続ホップでは、ラベルはスワップされ、パケット ヘッダーで伝送されるラベルの MPLS 転送テーブル検索を使用して転送が判断されます。そのため、ネットワークを介したパケットの送信中にパケット ヘッダーを再評価する必要はありません。ラベルは構造化されていない固定長の値であるため、MPLS 転送テーブル検索プロセスは簡単かつ高速です。

ラベル バインディングの配布

ネットワーク内の各ラベル スイッチング ルータ（LSR）は、転送同等クラスを表すためにどのラベル値を使用するかについて独立したローカルな決定を行います。このアソシエーションは、ラベル バインディングと呼ばれます。各 LSR は、自身が行ったラベル バインディングをネイバーに通知します。このようにネイバースイッチにラベルバインディングを認識させる処理は、次のプロトコルによって促進されます。

- ラベル配布プロトコル（LDP）：MPLS ネットワーク内のピア LSR は、MPLS ネットワークでのホップバイホップ転送をサポートするためのラベル バインディング情報を交換できます
- Border Gateway Protocol（BGP）：MPLS バーチャル プライベート ネットワーク（VPN）をサポートするために使用

ラベル付きパケットが LSR A からネイバー LSR B に送信されている場合、単一の IP パケットによって伝送されるラベル値は、パケットの転送等価クラスを表すために LSR B によって割り当てられたラベル値です。このため、IP パケットがネットワークを通過するにつれて、ラベル値は変更されます。

LDP 設定の詳細については、次にある「MPLS: LDP Configuration Guide」を参照してください。
http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mpls/config_library/xe-3s/mp-xe-3s-library.html

（注） ラベル エントリの規模は制限されているため（特に ECMP では）、LDP ラベル フィルタリングを有効にすることが推奨されます。LDP ラベルは、ルータのループバック インターフェイスなどのウェルノウン プレフィックスおよびグローバル ルーティング テーブルで到達可能にする必要があるプレフィックスにのみ割り当てるものとします。

MPLS レイヤ 3 VPN

マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）は、MPLS プロバイダー コア ネットワークによって相互接続された一連のサイトで構成されます。各カスタマー サイトでは、1 つ以上のカスタマー エッジ（CE）ルータが、1 つ以上のプロバイダー エッジ（PE）ルータに接続されます。

MPLS レイヤ 3 VPN を設定する前に、MPLS、ラベル配布プロトコル（LDP）、およびシスコ エクスプレス フォワーディング（CEF）が、ネットワークにインストールされている必要があります。PE ルータを含む、コア内のすべてのルータは、CEF および MPLS 転送をサポートできる必要があります。

MPLS QoS EXP の分類とマーキング

QoS EXP Matching 機能を使用すれば、IP パケットのマルチプロトコル ラベル スイッチング（MPLS）Experimental ビット（EXP ビット）フィールドを変更して、ネットワーク トラフィックを分類してマーキングすることができます。

QoS EXP Matching 機能を使用すれば、MPLS パケットの MPLS EXP フィールドに値を設定することによってネットワーク トラフィックを整理できます。MPLS EXP フィールドで異なった値を選択することにより、輻輳時にパケットが必要なプライオリティを持つようパケットをマーキングすることができます。MPLS EXP 値の設定によって次のことが可能になります。

- **トラフィックの分類**：分類プロセスでマーキングするトラフィックが選択されます。分類は、トラフィックを複数の優先順位レベル、つまり、サービス クラスに分割することによりこのプロセスを実施します。トラフィック分類は、クラス ベースの QoS プロビジョニングのプライマリ コンポーネントです。
- **トラフィックのポリシングとマーキング**：ポリシングでは、設定されたレートを上回るトラフィックが廃棄されるか、別のドロップ レベルにマーキングされます。トラフィックのマーキングは、パケット フローを特定してそれらを区別する方法です。パケット マーキングを利用すれば、ネットワークを複数の優先プライオリティ レベルまたはサービス クラスに分割することができます。

制限事項

以下に、MPLS QoS EXP の分類とマーキングに関する制約事項の一覧を示します。

- 均一モードとパイプ モードのみがサポートされます。ショートパイプ モードはサポートされません。
- サポートされる QoS グループ値の範囲は 0 ～ 30 です。（合計 31 の QoS グループ）。
- QoS ポリシーを使用した EXP マーキングは外部ラベルでのみサポートされます。内部の EXP マーキングはサポートされません。

マルチプロトコル ラベル スイッチングの設定方法

このセクションでは、MPLS スイッチングと転送用にスイッチを準備するために必要な基本設定を行う方法について説明します。

MPLS スイッチング用のスイッチの設定

シスコ スイッチ上の MPLS スイッチングでは、Cisco Express Forwarding がイネーブルである必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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	**ip cef distributed** 例： `Device(config)# ip cef distributed`	スイッチでシスコ エクスプレス フォワーディングをイネーブルにします。
ステップ 4	**mpls label range** *minimum-value maximum-value* 例： `Device(config)# mpls label range 16 4096`	パケット インターフェイス上で MPLS アプリケーションで使用可能なローカル ラベルの範囲を設定します。
ステップ 5	**mpls label protocol ldp** 例： `Device(config)# mpls label protocol ldp`	プラットフォームの Label Distribution Protocol を指定します。

MPLS 転送用のスイッチの設定

シスコ スイッチ上の MPLS 転送では、IPv4 パケットの転送がイネーブルになっている必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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** *type slot/subslot /port* 例： `Device(config)# interface gigabitethernet 1/0/0`	ギガビット イーサネット インターフェイスを指定し、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。スイッチ仮想インターフェイス（SVI）の場合の例を次に示します。 `Device(config)# interface vlan 1000`
ステップ 4	**mpls ip** 例： `Device(config-if)# mpls ip`	ルーテッド物理インターフェイス（ギガビット イーサネット）、スイッチ仮想インターフェイス（SVI）、またはポート チャネルに沿った IPv4 パケットの MPLS 転送を有効にします。
ステップ 5	**mpls label protocol ldp** 例： `Device(config-if)# mpls label protocol ldp`	インターフェイスの Label Distribution Protocol を指定します。 （注） MPLS LDP は、Virtual Routing and Forwarding（VRF）インターフェイスで有効にすることはできません。
ステップ 6	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

マルチプロトコル ラベル スイッチングの設定の確認

このセクションでは、MPLS のスイッチングと転送の設定に問題がないことを確認する方法について説明します。

MPLS スイッチングの構成の確認

Cisco Express Forwarding が正しく設定されていることを確認するには、**show ip cef summary** コマンドを発行します。次に示すような出力が生成されます。

手順

show ip cef summary

例：

```
Switch# show ip cef summary

IPv4 CEF is enabled for distributed and running
VRF Default
 150 prefixes (149/1 fwd/non-fwd)
 Table id 0x0
 Database epoch:        4 (150 entries at this epoch)
Switch#
```

MPLS 転送の構成の確認

MPLS 転送が正しく設定されていることを確認するには、**show mpls interfaces detail** コマンドを発行します。次に示すような出力が生成されます。

手順

ステップ 1 **show mpls interfaces detail**

例：

```
For physical (Gigabit Ethernet) interface:
Switch# show mpls interfaces detail interface GigabitEthernet 1/0/0

        Type Unknown
        IP labeling enabled
        LSP Tunnel labeling not enabled
        IP FRR labeling not enabled
        BGP labeling not enabled
        MPLS not operational
        MTU = 1500

For Switch Virtual Interface (SVI):
Switch# show mpls interfaces detail interface Vlan1000

        Type Unknown
        IP labeling enabled (ldp) :
          Interface config
        LSP Tunnel labeling not enabled
        IP FRR labeling not enabled
        BGP labeling not enabled
        MPLS operational
        MTU = 1500
```

ステップ 2 **show running-config interface**

例：

```
For physical (Gigabit Ethernet) interface:
Switch# show running-config interface interface GigabitEthernet 1/0/0

Building configuration...


Current configuration : 307 bytes
!
interface TenGigabitEthernet1/0/0
no switchport
ip address xx.xx.x.x xxx.xxx.xxx.x
mpls ip
mpls label protocol ldp
end

For Switch Virtual Interface (SVI):
Switch# show running-config interface interface Vlan1000

Building configuration...


Current configuration : 187 bytes
!
interface Vlan1000
ip address xx.xx.x.x xxx.xxx.xxx.x
mpls ip
mpls label protocol ldp
end
```

ステップ 3 **show mpls forwarding**

例：

```
For physical (Gigabit Ethernet) interface:
Switch#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
500        No Label   l2ckt(3)         0             Gi3/0/22   point2point
501        No Label   l2ckt(1)         12310411816789 none       point2point
502        No Label   l2ckt(2)         0             none       point2point
503        566        15.15.15.15/32   0             Po5        192.1.1.2
504        530        7.7.7.7/32       538728528     Po5        192.1.1.2
505        573        6.6.6.10/32      0             Po5        192.1.1.2
506        606        6.6.6.6/32       0             Po5        192.1.1.2
507        explicit-n 1.1.1.1/32       0             Po5        192.1.1.2
556        543        19.10.1.0/24     0             Po5        192.1.1.2
567        568        20.1.1.0/24      0             Po5        192.1.1.2
568        574        21.1.1.0/24      0             Po5        192.1.1.2
574        No Label   213.1.1.0/24[V]  0             aggregate/vpn113
575        No Label   213.1.2.0/24[V]  0             aggregate/vpn114
576        No Label   213.1.3.0/24[V]  0             aggregate/vpn115
577        No Label   213:1:1::/64     0             aggregate
594        502        103.1.1.0/24     0             Po5        192.1.1.2
595        509        31.1.1.0/24      0             Po5        192.1.1.2
596        539        15.15.1.0/24     0             Po5        192.1.1.2
597        550        14.14.1.0/24     0             Po5        192.1.1.2
633        614        2.2.2.0/24       0             Po5        192.1.1.2
634        577        90.90.90.90/32   873684        Po5        192.1.1.2
635        608        154.1.1.0/24     0             Po5        192.1.1.2
636        609        153.1.1.0/24     0             Po5        192.1.1.2
Switch#
end
```

マルチプロトコルラベルスイッチングに関するその他の参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
この章で使用するコマンドの完全な構文および使用方法の詳細。	*Command Reference (Catalyst 9300 Series Switches)* の「マルチプロトコル ラベル スイッチング（MPLS）コマンド」の項を参照してください。

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

マルチプロトコル ラベル スイッチングの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *1*：マルチプロトコル ラベル スイッチングの機能情報

リリース	変更箇所
Cisco IOS XE Everest 16.5.1a	この機能が導入されました。

第 2 章

eiBGP マルチパスの設定

MPLS-VPN における eBGP および iBGP に対する BGP マルチパス ロード シェアリング

eBGP および iBGP に対する BGP マルチパス ロード シェアリング機能によって、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）を使用するように設定されたボーダー ゲートウェイ プロトコル（BGP）ネットワークで、外部 BGP（eBGP）パスおよび内部 BGP（iBGP）パスの両方を使用してマルチパス ロード バランシングを設定できます。この機能によって、ロード バランシングの配備能力およびサービス提供能力が向上します。また、この機能は、マルチホーム ネットワークおよびスタブ ネットワークから eBGP パスおよび iBGP パスの両方をインポートするマルチホーム自律システムおよびプロバイダー エッジ（PE）ルータのために役立ちます。

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロード シェアリングの前提条件

Cisco Express Forwarding（CEF）または分散型 CEF（dCEF）が、参加するすべてのでデバイスでイネーブルになっている必要があります。

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロード シェアリングの制約事項

アドレス ファミリのサポート

この機能は、VPN ルーティング/転送（VRF）インスタンス単位で設定されます。この機能は IPv4 および IPv6 の VRF アドレス ファミリの両方で設定できます。

メモリ消費の制約事項

各 BGP マルチパス ルーティング テーブル エントリでは、追加のメモリを使用します。使用できるメモリが少ないデバイスや、特にフル インターネット ルーティング テーブルを送受信するデバイスでは、この機能の使用はお勧めしません。

パス数の制限

サポートされるパスの数は、2 つの BGP マルチパスに限定されます。iBGP マルチパス 2 つか、または iBGP マルチパス 1 つと eBGP マルチパス 1 つのいずれかです。

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロード シェアリングについて

eBGP と iBGP 間のマルチパス ロード シェアリング

BGP ルーティング プロセスではデフォルトで、1 つのパスを最良パスとしてルーティング情報ベース（RIB）にインストールします。maximum-paths コマンドを使用すると、マルチパス ロード シェアリングのために複数のパスを RIB にインストールするように BGP を設定できます。BGP は最良パス アルゴリズムを使用して 1 つのマルチパスを最良パスとして選択し、その最良パスを BGP ピアにアドバタイズします。

（注） 設定できるマルチパスのパス数は、maximum-paths コマンド リファレンスのページに記載されています。

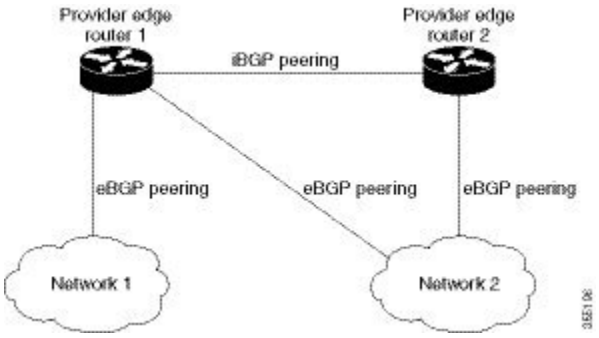
マルチパス全体でのロード バランシングは CEF によって実行されます。CEF ロード バランシングは、パケット単位のラウンド ロビンまたはセッション単位（送信元と宛先のペア）を基準として設定されます。CEF については、『Cisco IOS IP Switching Configuration Guide』（http://ciscosystems.com/en/US/docs/ios/ipswitch/configuration/guide/12_2sx/isw_12_2sx_book.html）のドキュメントを参照してください。MPLS VPN 機能における eBGP と iBGP の両方に対する BGP マルチパス ロード シェアリングは、IPv4 VRF アドレス ファミリおよび IPv6 VRF アドレス ファミリのコンフィギュレーション モードでイネーブルになります。この機能がイネーブルにされると、VRF にインポートされた eBGP パスまたは iBGP パスあるいはその両方でロード バランシングを実行できます。マルチパスの数は VRF 単位で設定されます。別々の VRF マルチパス設定は、固有ルート識別子によって分離されます。

（注） MPLS VPN における eBGP および iBGP に対する BGP マルチパス ロード シェアリング機能は、設定されたアウトバウンド ルーティング ポリシーのパラメータの範囲内で動作します。

BGP MPLS ネットワークにおける eBGP および iBGP のマルチパス ロード シェアリング

次の図に、2 つのリモート ネットワークを PE ルータ 1 および PE ルータ 2 に接続したサービス プロバイダー BGP MPLS ネットワークを示します。PE ルータ 1 および PE ルータ 2 には、いずれも VPNv4 ユニキャスト iBGP ピアリングが設定されています。ネットワーク 2 は、PE ルータ 1 および PE ルータ 2 に接続されているマルチホーム ネットワークです。またネットワーク 2 は、ネットワーク 1 とのエクストラネット VPN サービスが設定されています。ネットワーク 1 とネットワーク 2 は両方とも、PE ルータを使用した eBGP ピアリングが設定されています。

図 *1*：サービス プロバイダー *BGP MPLS* ネットワーク



PE ルータ 1 には、MPLS VPN における eBGP および iBGP の両方に BGP マルチパス ロード シェアリング機能が設定でき、これによって、iBGP パスと eBGP パスの両方をマルチパスとして選択し、VRF にインポートできます。マルチパスは CEF によって使用され、ロード バランシングが実行されます。ネットワーク 1 からネットワーク 2 に送信される IP トラフィックでは、PE ルータ 1 が eBGP パスを使用してロード シェアリングします。これは、IP トラフィックと iBGP パスが MPLS トラフィックとして送信されるためです。

（注）

- ローカル CE とローカル PE 間の eBGP セッションはサポートされていません。
- • ローカル PE からリモート CE への eBGP セッションはサポートされています。

eBGP および iBGP の両方に対するマルチパス ロード シェアリングの利点

MPLS VPN における eBGP および iBGP に対する BGP マルチパス ロード シェアリング機能を使用すると、マルチホーム自律システムおよび PE ルータで、eBGP パスおよび iBGP パスの両方を経由してトラフィックを配信するように設定できます。

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロード シェアリングの設定方法

ここでは、次の手順について説明します。

eBGP および iBGP の両方に対するマルチパス ロード シェアリングの設定

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configure**{**terminal**|**memory**|**network**} 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 as-number** 例： `Device(config)# router bgp 40000`	ルータ コンフィギュレーション モードを開始して、BGP ルーティング プロセスを作成または設定します。
ステップ 4	**address-family ipv4 vrf***vrf-name* 例： `Device(config-router)# address-family ipv4 vrf RED`	ルータをアドレス ファミリ コンフィギュレーション モードにします。 • 別々の VRF マルチパス設定は、固有ルート識別子によって分離されます。
ステップ 5	**address-family ipv6 vrf***vrf-name* 例： `Device(config-router)# address-family ipv6 vrf RED`	ルータをアドレス ファミリ コンフィギュレーション モードにします。 • 別々の VRF マルチパス設定は、固有ルート識別子によって分離されます。
ステップ 6	**maximum-paths eibgp number**[*import number*] 例： `Device(config-router-af)# maximum-paths eibgp 2`	ルーティング テーブルにインストールできるパラレルの iBGP ルートおよび eBGP ルートの数を設定します。 （注） maximum-paths eibgp コマンドは IPv4 VRF アドレス ファミリ コンフィギュレーション モードだけで設定でき、他のすべてのアドレス ファミリ コンフィギュレーション モードでは設定できません。

eBGP および iBGP の両方に対するマルチパス ロード シェアリングの設定の確認

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**show ip bgp neighbors** 例： `Device# show ip bgp neighbors`	ネイバーへの TCP 接続および BGP 接続についての情報を表示します。
ステップ 3	**show ip bgp vpnv4 vrf***vrf name* 例： `Device# show ip bgp vpnv4 vrf RED`	VPN アドレス情報を BGP テーブルから表示します。このコマンドは、VRF が BGP によって受信されたことを確認するために使用します。
ステップ 4	**show ip route vrf***vrf-name* 例： `Device# show ip route vrf RED`	VRF インスタンスに関連する IP ルーティング テーブルを表示します。show ip route vrf コマンドは、該当する VRF がルーティング テーブルにあることを確認するために使用します。

次の作業

•

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロード シェアリング機能の設定例

次に、この機能の設定方法および確認方法の例を示します。

eBGP および iBGP のマルチパス ロード シェアリングの設定例

次の設定例では、ルータを IPv4 アドレスファミリ モードで設定して、2 つの BGP ルート（eBGP または iBGP）をマルチパスとして選択します。

```
Device router bgp 40000
 Deviceaddress-family ipv4 vrf RED
Devicemaximum-paths eibgp 2
Deviceend
```

次の設定例では、ルータを IPv6 アドレスファミリ モードで設定して、2 つの BGP ルート（eBGP または iBGP）をマルチパスとして選択します。

```
Device router bgp 40000
 Deviceaddress-family ipv6 vrf RED
Devicemaximum-paths eibgp 2
Deviceend
```

その他の参考資料

関連資料

表 *2*：関連資料

関連項目	参照先
BGP コマンド：コマンド構文、コマンド モード、コマンド履歴、デフォルト、使用に関する注意事項、および例	•『Cisco IOS IP Command Reference, Volume 2 of 4: Routing Protocols, Release 12.3T』
BGP 設定作業	•『Cisco IOS IP Configuration Guide, Release 12.3』
総合的な BGP リンク帯域幅の設定例および作業	•『BGP Link Bandwidth』
CEF 設定作業	•『Cisco IOS Switching Services Configuration Guide』

表 *3*：標準

標準	タイトル
この機能によってサポートされる新しい規格または変更された規格はありません。またこの機能による既存規格のサポートに変更はありません。	--

表 *4* : *RFC*

RFC	タイトル
RFC 1771	『A Border Gateway Protocol 4（BGP4）』
RFC 2547	『BGP/MPLS VPNs』
RFC 2858	『Multiprotocol Extensions for BGP-4』

表 *5* : テクニカル サポート

説明	リンク
TAC のホームページには、3 万ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、ツール等へのリンクもあります。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	*http://www.cisco.com/public/support/tac/home.shtml*

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロード シェアリングの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *6*：*MPLS-VPN* における *eBGP* および *iBGP* の両方に対する *BGP* マルチパス ロード シェアリングの機能情報

機能名（Feature Name）	リリース	機能情報
MPLS-VPN における eBGP および iBGP に対する BGP マルチパス ロード シェアリング	Cisco IOS 16.6.1	eBGP および iBGP に対する BGP マルチパス ロード シェアリング機能によって、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）を使用するように設定されたボーダー ゲートウェイ プロトコル（BGP）ネットワークで、外部 BGP（eBGP）パスおよび内部 BGP（iBGP）パスの両方を使用してマルチパス ロード バランシングを設定できます。この機能によって、ロード バランシングの配備能力およびサービス提供能力が向上します。また、この機能は、マルチホーム ネットワークおよびスタブ ネットワークから eBGP パスおよび iBGP パスの両方をインポートするマルチホーム自律システムおよびプロバイダー エッジ（PE）ルータのために役立ちます。

第 3 章

EIGRP MPLS VPN PE-CE Site of Origin の設定

EIGRP MPLS VPN PE-CE Site of Origin

EIGRP MPLS VPN PE-CE Site of Origin 機能によって、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）トラフィックを、Enhanced Interior Gateway Routing Protocol（EIGRP）ネットワークに対してサイト単位でフィルタリングする機能が追加されます。Site of Origin（SoO）フィルタリングは、インターフェイス レベルで設定され、これを使用して MPLS VPN トラフィックを管理し、複雑で複合的なネットワーク トポロジにおいて過渡的なルーティング ループが発生しないようにします。この機能は、プロバイダー エッジ（PE）とカスタマー エッジ（CE）間の EIGRP に対する MPLS VPN Support 機能をサポートするために設計されています。EIGRP MPLS VPN をサポートしている PE ルータ上にインストールされている場合、この機能によってバックドア リンクに対するサポートが提供されます。

EIGRP MPLS VPN PE-CE Site of Origin の前提条件

このドキュメントでは、ネットワーク コア（またはサービス プロバイダー バックボーン）にボーダー ゲートウェイ プロトコル（BGP）が設定されていることを前提にしています。この機能を設定する前に、次のタスクも完了している必要があります。

- • この機能は、プロバイダー エッジとカスタマー エッジ間の EIGRP に対する MPLS VPN Support 機能をサポートするために導入されており、この機能は、EIGRP MPLS VPN の作成後に設定する必要があります。

- • EIGRP MPLS VPN をサポートするために設定されるすべての PE ルータは、SoO の拡張コミュニティにサポートを提供している Cisco IOS XE リリース 2.1 以降を実行している必要があります。

EIGRP MPLS VPN PE-CE Site of Origin の制約事項

- VPN サイトがパーティション化されていて、バックドア ルータ インターフェイスで SoO 拡張コミュニティ属性が設定されている場合は、このバックドア リンクを、同じサイトの他のパーティションを起点とするプレフィックスへの代替パスとして使用することはできません。
- • VPN サイトごとに、一意の SoO 値を設定する必要があります。同じ VPN サイトをサポートしているすべてのプロバイダー エッジ、およびカスタマー エッジ インターフェイスには（SoO が CE ルータ上に設定されている場合）、同じ値を設定する必要があります。

EIGRP MPLS VPN PE-CE Site of Origin について

EIGRP MPLS VPN PE-CE Site of Origin サポートの概要

EIGRP MPLS VPN PE-CE Site of Origin 機能によって、EIGRP から BGP へ、および BGP から EIGRP への再配布に対するサポートが追加されます。SoO 拡張コミュニティは BGP 拡張コミュニティ属性の 1 つで、これを使用して、あるサイトから生じたルートを特定し、そのプレフィックスが送信元サイトへ再アドバタイズメントされないようにします。SoO 拡張コミュニティは、PE ルータがルートを学習したサイトを一意に識別します。SoO サポートには、EIGRP サイト単位で MPLS VPN トラフィックをフィルタリングする機能があります。SoO のフィルタリングはインターフェイス レベルで設定されており、これを使用して MPLS VPN トラフィックを管理し、（VPN とバックドア リンクの両方が含まれている EIGRP VPN サイトなどの）複雑で複合的なネットワーク トポロジにおいてルーティング ループが発生しないようにします。

SoO 拡張コミュニティの設定によって、サイト単位で MPLS VPN トラフィックをフィルタリングできます。SoO 拡張コミュニティは、PE ルータ上の着信 BGP ルートマップで設定され、インターフェイスに適用されます。SoO 拡張コミュニティは、より細かくフィルタリングするために、カスタマー サイトのすべての exit ポイントに適用することができますが、VPN サービスを提供する PE ルータから CE ルータへのすべてのインターフェイスに設定する必要があります。

バックドア リンクに対する Site of Origin のサポート

EIGRP MPLS VPN PE-CE Site of Origin（SoO）機能によって、バックドア リンクに対するサポートが追加されます。バックドア リンクまたはルートは、リモート サイトとメイン サイトの間の VPN の外部に設定される接続で、たとえば、リモートサイトを企業ネットワークへ接続する WAN 専用線などがあります。バックドア リンクは通常、VPN リンクが停止した、または使用できなく

なった場合に EIGRP のサイト間でバックアップ ルートとして使用されます。VPN リンクの障害がない場合はバックドア ルータを介したルートが選択されないように、メトリックはバックドア リンク上に設定されます。

SoO 拡張コミュニティは、バックドア ルータのインターフェイス上に定義されます。これはローカル サイト ID を特定するもので、同じサイトをサポートしている PE ルータで使用される値と一致している必要があります。バックドア ルータが、バックドア リンクを介してネイバーから EIGRP アップデート（またはリプライ）を受信すると、ルータは、SoO 値のアップデートを調べます。EIGRP アップデート内の SoO 値がローカルなバックドア インターフェイスの SoO 値と一致している場合、そのルートは拒否され、EIGRP トポロジ テーブルには追加されません。このシナリオは通常、受信した EIGRP アップデート内で値が設定されたローカル SoO を備えたルートが他の VPN サイトで学習され、他の VPN サイト内のバックドア ルータによって、バックドア リンクを介してアドバタイズされたときに発生します。バックドア リンクにおける SoO フィルタリングでは、ローカル サイト ID を伝送するルートが含まれている EIGRP アップデートをフィルタリングすることによって、過渡的なルーティング ループが発生しないようにします。

PE ルータ、およびカスタマー サイトのバックドア ルータでこの機能が有効になっており、PE ルータとバックドア ルータの両方で SoO 値が定義されている場合は、PE ルータおよびバックドア ルータは VPN サイト間の統合をサポートします。カスタマー サイトの他のルータでは、ルートがネイバーへ転送されるため、ルートによって伝送される SoO 値を伝搬するだけですみます。これらのルータは、通常の拡散更新アルゴリズム（DUAL）計算以上は統合に影響を与えず、サポートもしません。

Site of Origin 拡張コミュニティとルータとの相互運用

SoO 拡張コミュニティを設定すると、EIGRP MPLS VPN PE-CE Site of Origin 機能をサポートしているルータが、各ルートの起点となるサイトを識別できます。この機能が有効になっていると、PE または CE ルータ上の EIGRP ルーティング プロセスは、受信したそれぞれのルートを SoO 拡張コミュニティに対してチェックし、次の条件に基づいてフィルタリングします。

- BGP または CE ルータから受信したルートには、受信側インターフェイス上の SoO 値と一致する SoO 値が含まれている場合：受信側インターフェイス上に設定されている SoO 値と一致する関連 SoO 値とともにルートを受信した場合、そのルートは別の PE ルータまたはバックドア リンクから学習したルートであるため、フィルタリングされます。この動作は、ルーティング ループを回避するために設計されています。

- • CE ルータから受信したルートが一致しない SoO 値で設定されている場合：あるルートが、関連付けられている SoO 値とともに受信され、その値が、受信インターフェイス上で設定されている SoO 値と一致しない場合、そのルートは、BGP へ再配布されるように EIGRP トポロジ テーブルに追加されます。ルートが すでに EIGRP トポロジ テーブルにインストールされているが、別の SoO 値と関連付けられている場合は、そのルートが BFP へ再配布されるときに、トポロジ テーブルの SoO 値が使用されます。

- • CE ルータから受信したルートに SoO 値が含まれていない場合：受信したルートに SoO 値がない場合、そのルートは EIGRP トポロジ テーブルに受け入れられます。ルートが BGP へ

再配布される前に、ネクスト ホップ CE ルータに到達するために使用されるインターフェイスの SoO 値がそのルートに付加されます。

SoO 拡張コミュニティをサポートする BGP および EIGRP ピアがこれらのルートを受信する場合には、関連付けられている SoO 値も受信します。次に、これらの値を、SoO 拡張コミュニティをサポートしている他の BGP および EIGRP ピアへ渡します。このフィルタリングは、過渡的なルートが発信元サイトから再学習されないように、つまり過渡的なルーティング ループが発生しないようにする目的で設計されています。

Site of Origin を EIGRP に伝送する BGP VPN ルートの再配布

PE ルータ上の EIGRP ルーティング プロセスが、BGP VPN ルートを EIGRP トポロジ テーブルへ再配布する場合、EIGRP は、付加された BGP 拡張コミュニティ属性から（SoO 値があれば）SoO 値を抽出し、EIGRP トポロジテーブルへ追加する前に、その SoO 値をルートへ付加します。アップデートを CE ルータへ送信する前に、EIGRP は各ルートについて SoO 値をテストします。インターフェイス上で設定されている SoO 値と一致する SoO 値に関連付けられているルートは、CE ルータに渡される前にフィルタリングされます。EIGRP ルーティング プロセスが、異なる SoO 値に関連付けられているルートを受信すると、その SoO 値は CE ルータに渡され、CE サイトを介して伝送されます。

EIGRP MPLS VPN PE-CE Site of Origin サポート機能の利点

EIGRP MPLS VPN PE-CE Site of Origin サポート機能の設定によって、サイト単位の VPN フィルタリングが導入されます。これにより、バックドア リンクを備えた MPLS VPN、複数の PE ルータに対してデュアルホーム接続になっている CE ルータ、同じ virtual routing and forwarding（VRF）インスタンス内のさまざまなサイトから CE ルータをサポートしている PE ルータなどの複雑なトポロジに対するサポートが改善されます。

EIGRP MPLS VPN PE-CE Site of Origin サポートの設定方法

Site of Origin 拡張コミュニティの設定

SoO 拡張コミュニティの設定によって、サイト単位で MPLS VPN トラフィックをフィルタリングできます。SoO 拡張コミュニティは、PE ルータ上の着信 BGP ルートマップで設定され、インターフェイスに適用されます。SoO 拡張コミュニティは、より細かくフィルタリングするために、カスタマー サイトのすべての exit ポイントに適用することができますが、VPN サービスを提供する PE ルータから CE ルータへのすべてのインターフェイスに設定する必要があります。

はじめる前に

- • ネットワーク コア（またはサービス プロバイダー バックボーン）にボーダー ゲートウェイ プロトコル（BGP）が設定されていることを確認します。

- • この機能を設定する前に、EIGRP MPLS VPN を設定します。
- • EIGRP MPLS VPN をサポートするよう設定されているすべての PE ルータは、SoO 拡張コミュニティをサポートする必要があります。
- • 各 VPN サイトに対して一意の SoO 値を設定する必要があります。各 VPN サイトでは、CE ルータに接続する PE ルータのインターフェイス上で同じ値を使用する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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	**route-map** *map-name* {**permit**|**deny**}[*sequence-number*] 例： `Device(config)# route-map Site-of-Origin permit 10`	ルート マップ コンフィギュレーション モードを開始して、ルート マップを作成します。 • この手順でルート マップが作成され、SoO 拡張コミュニティが適用されるようになります。
ステップ 4	**set extcommunity soo***extended-community-value* 例： `Device(config-route-map)# set extcommunity soo 100:1`	BGP 拡張コミュニティ属性を設定します。 • soo キーワードには、Site of Origin 拡張コミュニティ属性を指定します。 • extended-community-value 引数には、設定する値を指定します。この値では、次のいずれかの形式を使用できます。 ◦ • autonomous-system-number：ネットワーク番号 ◦ • ip-address：ネットワーク番号 自律システム番号とネットワーク番号、または IP アドレスとネットワーク番号の区切りにはコロンを使用します。

	コマンドまたはアクション	目的
ステップ 5	**exit** 例： `Device(config-route-map)# exit`	ルート マップ コンフィギュレーション モードを終了し、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 6	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 1/0/1`	特定のインターフェイスを設定するため、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 7	**no switchport** 例： `Device(config-if)# no switchport`	インターフェイスをレイヤ2ポートとして動作することを停止し、シスコルーテッド（レイヤ3）ポートにします。
ステップ 8	**vrf forwarding** *vrf-name* 例： `Device(config-if)# ip vrf forwarding VRF1`	VRF をインターフェイスまたはサブインターフェイスに関連付けます。 • • この手順で設定された VRF 名は、プロバイダーエッジとカスタマーエッジ間の EIGRP に対する MPLS VPN Support 機能を備えた EIGRP MPLS VPN に対して作成された VRF 名と一致している必要があります。
ステップ 9	**ip vrf sitemap** *route-map-name* 例： `Device(config-if)# ip vrf sitemap Site-of-Origin`	VRF をインターフェイスまたはサブインターフェイスに関連付けます。 • • この手順で設定されたルートマップ名は、手順 3 で、SoO 拡張コミュニティを適用するために作成されたルート マップ名と一致している必要があります。
ステップ 10	**ip address** *ip-address subnet-mask* 例： `Device(config-if)# ip address 10.0.0.1 255.255.255.255`	インターフェイスの IP アドレスを設定します。 • • IP アドレスは、VRF フォワーディングをイネーブルにした後で再設定する必要があります。
ステップ 11	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードを開始します。

次の作業

- • バックドア ルートが含まれている、複合的な EIGRP MPLS VPN ネットワーク トポロジの場合は、次に、バックドア ルートに対して「準最適パス」コスト コミュニティを設定します。

SoO 拡張コミュニティの設定の確認

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**show ip bgp vpnv4** {**all**|**rd***route-distinguisher*|**vrf***vrf-name*}[*ip-prefix/length*] 例： `Device# ip bgp vpnv4 vrf SOO-1 20.2.1.1/32`	VPN アドレス情報を BGP テーブルから表示します。 • • show ip bgp vpnv4 コマンドと all キーワードを使用して、指定したルートが、SoO 拡張コミュニティ属性で設定されていることを検証します。

EIGRP MPLS VPN PE-CE SoO の設定例

Site of Origin 拡張コミュニティの設定例

次に、グローバル コンフィギュレーション モードで開始し、インターフェイス上で SoO 拡張コミュニティを設定する例を示します。

```
 route-map Site-of-Origin permit 10
 set extcommunity soo 100:1
exit
GigabitEthernet1/0/1
ip vrf forwarding RED
ip vrf sitemap Site-of-Origin
ip address 10.0.0.1 255.255.255.255
end
```

Site of Origin 拡張コミュニティの確認の例

次の例では、BGP テーブルの VPN アドレス情報を表示し、SoO 拡張コミュニティの設定を確認します。

```
switch# show ip bgp vpnv4 all 10.0.0.1
 BGP routing table entry for 100:1:10.0.0.1/32, version 6
Paths: (1 available, best #1, no table)
Advertised to update-groups:
1
100 300
192.168.0.2 from 192.168.0.2 (172.16.13.13)
Origin incomplete, localpref 100, valid, external, best
Extended Community: SOO:100:1
```

カスタマー エッジ デバイス show コマンド

```
CE1#show ip eigrp topo 20.2.1.1/32
EIGRP-IPv4 Topology Entry for AS(30)/ID(30.0.0.1) for 20.2.1.1/32
  State is Passive, Query origin flag is 1, 2 Successor(s), FD is 131072
  Descriptor Blocks:
  31.1.1.2 (GigabitEthernet1/0/13), from 31.1.1.2, Send flag is 0x0
      Composite metric is (131072/130816), route is External
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5020 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
        Originating router is 30.0.0.2
      Extended Community: SoO:100:1
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0
        Administrator tag is 0 (0x00000000)
```

プロバイダー エッジ デバイス show コマンド

```
PE2#show ip eigrp vrf SOO-1 topology 31.1.1.0/24
EIGRP-IPv4 VR(L3VPN) Topology Entry for AS(30)/ID(2.2.2.22)
           Topology(base) TID(0) VRF(SOO-1)
EIGRP-IPv4(30): Topology base(0) entry for 31.1.1.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 1310720
  Descriptor Blocks:
  1.1.1.1, from VPNv4 Sourced, Send flag is 0x0
      Composite metric is (1310720/0), route is Internal (VPNv4 Sourced)
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 10000000 picoseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 0
        Originating router is 1.1.1.11
      Extended Community: SoO:100:1
```

その他の参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
BGP コストコミュニティ機能と「pre-bestpath」挿入ポイント	
CEF コマンド	『Cisco IOS IP Switching Command Reference』
CEF 設定作業	『Cisco Express Forwarding Overview module of the Cisco IOS IP Switching Configuration Guide』
EIGRP コマンド	『Cisco IOS IP Routing: EIGRP Command Reference』
EIGRP の設定タスク	『Configuring EIGRP』
MPLS VPN	『MPLS Layer 3 VPNs module of the Cisco IOS Multiprotocol Label Switching Configuration Guide』

表 *7*：標準

標準	タイトル
なし	--

表 *8*：*MIB*

MIB	MIB リンク
なし	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャ セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://tools.cisco.com/ITDIT/MIBS/servlet/index

表 *9* : *RFC*

RFC	タイトル
なし	--

表 *10* : テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/c/en/us/support/index.html

EIGRP MPLS VPN PE-CE Site of Origin の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *11* : *EIGRP MPLS VPN PE-CE Site of Origin* の機能情報

機能名（**Feature Name**）	リリース	機能情報
EIGRP MPLS VPN PE-CE Site of Origin	Cisco IOS 16.6.1	EIGRP MPLS VPN PE-CE Site of Origin機能によって、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）トラフィックを、Enhanced Interior Gateway Routing Protocol（EIGRP）ネットワークに対してサイト単位でフィルタリングする機能が追加されます。Site of Origin（SoO）フィルタリングは、インターフェイス レベルで設定され、これを使用して MPLS VPN トラフィックを管理し、複雑で複合的なネットワーク トポロジにおいて過渡的なルーティング ループが発生しないようにします。

第 4 章

Ethernet-over-MPLS（EoMPLS）および疑似回線冗長性（PWR）の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

EoMPLS の設定

EoMPLS について

EoMPLS は AToM トランスポート タイプの 1 つです。EoMPLS は MPLS パケットに Ethernet PDU をカプセル化し、MPLS ネットワーク上で転送することにより機能します。各 PDU は単一パケットとして転送されます。

Cisco IOS XE Everest 16.6.1 は、次のモードのみをサポートしています。

- ポート モード：ポートのすべてのトラフィックが MPLS ネットワーク上の単一の VC を共有できるようにします。ポート モードは VC タイプ 5 を使用します。

スケール番号

表 *12*：*EoMPLS VC* スケール

サポートされる VC の数	3650	3850	9300	9500
256	256	256	1k	

EoMPLS の前提条件

EoMPLS を設定する前に、ネットワークが次のように設定されていることを確認してください。

- PE ルータが IP を通して相互に到達できるように、コアに IP ルーティングを設定します。
- PE ルータ間にラベル スイッチド パス（LSP）が存在するように、コアに MPLS を設定します。
- 接続回線に Xconnect を設定する前に、**no switchport**, **no keepalive** と **no ip address** を設定します。
- ロードバランシングを行うには、**port-channel load-balance** コマンドを設定する必要があります。

EoMPLS の制約事項

- VLAN モードはサポートされていません。イーサネット フロー ポイントはサポートされていません。
- ポートチャネルは接続回線としてサポートされていません。
- QoS：カスタマー DSCP 再マーキングは VPWS と EoMPLS ではサポートされていません。
- 明示的 null の VCCV ping はサポートされていません。
- L2 VPN インターワーキングはサポートされていません。
- L2 プロトコル トンネリング CLI はサポートされていません。
- タグなし、タグ付き、802.1Q in 802.1Q が着信トラフィックとしてサポートされています。
- Flow Aware Transport 疑似回線冗長性（FAT PW）は、プロトコル CLI モードでのみサポートされています。サポートされているロード バランシング パラメータは、送信元 IP、送信元 MAC アドレス、宛先 IP、および宛先 MAC アドレスです。

- 制御ワードのイネーブル化またはディセーブル化がサポートされています。
- MPLS QoS は、パイプおよび均一モードでサポートされています。デフォルト モードはパイプ モードです。
- 両方：レガシー Xconnect モードとプロトコル CLI（インターフェイス疑似回線設定）モードがサポートされています。

デフォルトでは、EoMPLS PW は CDP や STP のようなすべてのプロトコルをトンネリングします。EoMPLS PW は L2 プロトコル トンネリング CLI の一環として選択的なプロトコル トンネリングを実行できません。

ポートモード EoMPLS の設定

ポートモード EoMPLS は、2 つのモードで設定できます。

- Xconnect モード
- プロトコル CLI 方式

Xconnect モード

Xconnect モードでポートモード EoMPLS を設定するには、次の作業を実行します。

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 モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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 TenGigabitEthernet1/0/36`	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。

	コマンドまたはアクション	目的
ステップ 4	**no switchport** 例： `Device(config-if)# no switchport`	物理ポートに限り、レイヤ3モードを開始します。
ステップ 5	**no ip address** 例： `Device(config-if)# no ip address`	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	**no keepalive** 例： `Device(config-if)# no keepalive`	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 7	**xconnect** *peer-device-id vc-id* **encapsulation mpls** 例： `Device(config-if)# xconnect 1.1.1.1 962 encapsulation mpls`	接続回線を疑似接続 VC にバインドします。このコマンドの構文は、その他のレイヤ2トランスポートの場合と同じです。
ステップ 8	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

プロトコル CLI 方式

プロトコル CLI モードでポートモード EoMPLS を設定するには、次の作業を実行します。

手順

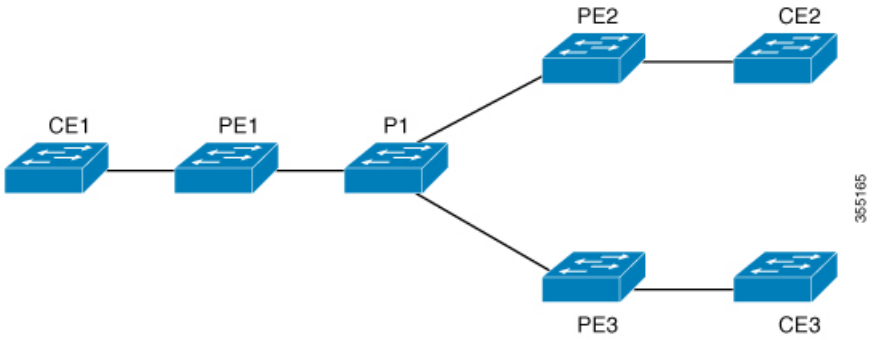
	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**port-channel load-balance dst-ip** 例： `Device(config)# port-channel load-balance 192.168.2.25`	負荷分散方式を宛先 IP アドレスに設定します。 • *dst-ip*：宛先 IP アドレス
ステップ 4	**interface** *interface-id* 例： `Device(config)# interface TenGigabitEthernet1/0/21`	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 5	**no switchport** 例： `Device(config-if)# no switchport`	物理ポートに限り、レイヤ 3 モードを開始します。
ステップ 6	**no ip address** 例： `Device(config-if)# no ip address`	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 7	**no keepalive** 例： `Device(config-if)# no keepalive`	デバイスがキープアライブ メッセージを送信しないことを確認します。

	コマンドまたはアクション	目的
ステップ 8	**exit** 例： `Device(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了します。
ステップ 9	**interface pseudowire** *number* 例： `Device(config-if)# interface pseudowire 17`	指定した値でインターフェイス擬似回線を確立して、擬似回線コンフィギュレーション モードを開始します。 • *number*：設定する疑似回線の番号を指定します。
ステップ 10	**encapsulation mpls** 例： `Device(config-if)# encapsulation mpls`	トンネリング カプセル化を指定します。
ステップ 11	**neighbor** *peer-device-id vc-id* 例： `Device(config-if)# neighbor 4.4.4.4 17`	レイヤ 2 VPN（L2VPN）疑似回線のピア IP アドレスと仮想回線（VC）ID を指定します。
ステップ 12	**load-balance** *dst-ip* 例： `Device(config-if)# load-balance 192.168.2.25`	等コスト マルチパス（ECMP）を使用して、複数のコア側インターフェイスにわたるトラフィックのエッジ ロード バランシングをイネーブルにします。 • *dst-ip*：宛先 IP アドレス
ステップ 13	**load-balance flow-label both** 例： `Device(config-if)# load-balance flow-label both`	フローラベルに基づいてコア ロードバランシングをイネーブルにします。

	コマンドまたはアクション	目的
ステップ 14	**l2vpn xconnect context** *context-name* 例： `Device(config-if)# l2vpn xconnect context vpws17`	レイヤ 2 VPN（L2VPN）クロス コネクト コンテキストを作成して、Xconnect コンテキスト コンフィギュレーション モードを開始します。
ステップ 15	**member** *interface-id* 例： `Device(config-if)# member TenGigabitEthernet1/0/21`	レイヤ 2 VPN（L2VPN）クロス コネクトを形成するインターフェイスを指定します。
ステップ 16	**member pseudowire** *number* 例： `Device(config-if)# member pseudowire 17`	レイヤ 2 VPN（L2VPN）クロス コネクトを形成する疑似回線インターフェイスを指定します。
ステップ 17	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

EoMPLS の設定例

図 *2* : ***EoMPLS*** トポロジ



PE の設定	CE の設定
mpls ip mpls label protocol ldp mpls ldp graceful-restart mpls ldp router-id loopback 1 force interface Loopback1 ip address 1.1.1.1 255.255.255.255 ip ospf 100 area 0 router ospf 100 router-id 1.1.1.1 nsf system mtu 9198 port-channel load-balance dst-ip ! interface GigabitEthernet2/0/39 no switchport no ip address no keepalive ! interface pseudowire101 encapsulation mpls neighbor 4.4.4.4 101 load-balance flow ip dst-ip load-balance flow-label both l2vpn xconnect context pw101 member pseudowire101 member GigabitEthernet2/0/39 ! interface TenGigabitEthernet3/0/10 switchport trunk allowed vlan 142 switchport mode trunk channel-group 42 mode active ! interface Port-channel42 switchport trunk allowed vlan 142 switchport mode trunk ! interface Vlan142 ip address 142.1.1.1 255.255.255.0 ip ospf 100 area 0 mpls ip mpls label protocol ldp !	interface GigabitEthernet1/0/33 switchport trunk allowed vlan 912 switchport mode trunk spanning-tree portfast trunk ! interface Vlan912 ip address 10.91.2.3 255.255.255.0 !

次に、**show mpls l2 vc vcid vc-id detail** コマンドの出力例を示します。

```
Local interface: Gi1/0/1 up, line protocol up, Ethernet up
 Destination address: 1.1.1.1, VC ID: 101, VC status: up
Output interface: Vl182, imposed label stack {17 16}
Preferred path: not configured
Default path: active
Next hop: 182.1.1.1
Load Balance: ECMP
flow classification: ip dst-ip
Create time: 06:22:11, last status change time: 05:58:42
Last label FSM state change time: 05:58:42  Signaling protocol:
LDP, peer 1.1.1.1:0 up
Targeted Hello: 4.4.4.4(LDP Id) -> 1.1.1.1, LDP is UP
Graceful restart: not configured and not 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 16
Group ID: local n/a, remote 0
MTU: local 9198, remote 9198
Remote interface description:   Sequencing: receive disabled, send disabled

Control Word: On (configured: autosense)
SSO Descriptor: 1.1.1.1/101, local label: 512
Dataplane:
SSM segment/switch IDs: 4096/4096 (used), PWID: 1
VC statistics:    transit packet totals: receive 172116845, send 172105364

transit byte totals:   receive 176837217071, send 172103349728
 transit packet drops:  receive 0, seq error 0, send 0
```

次に、**show l2vpn atom vc vcid vc-id detail** コマンドの出力例を示します。

```
pseudowire101 is up, VC status is up PW type: Ethernet
Create time: 06:30:41, last status change time: 06:07:12
Last label FSM state change time: 06:07:12
Destination address: 1.1.1.1 VC ID: 101
Output interface: Vl182, imposed label stack {17 16}
 Preferred path: not configured
 Default path: active    Next hop: 182.1.1.1
Load Balance: ECMP  Flow classification: ip dst-ip
Member of xconnect service pw101
Associated member Gi1/0/1 is up, status is up
Interworking type is Like2Like    Service id: 0xe5000001
Signaling protocol: LDP, peer 1.1.1.1:0 up
Targeted Hello: 4.4.4.4(LDP Id) -> 1.1.1.1, LDP is UP
Graceful restart: not configured and not enabled
Non stop routing: not configured and not enabled
PWid FEC (128), VC ID: 101    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                            16
Group ID     n/a                            0
Interface

MTU          9198                           9198
Control word on (configured: autosense)     on
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
Flow Label   T=1, R=1                      T=1, R=1
SSO Descriptor: 1.1.1.1/101, local label: 512
Dataplane:
SSM segment/switch IDs: 4096/4096 (used), PWID: 1
Rx Counters    176196691 input transit packets, 181028952597 bytes
 0 drops, 0 seq err
Tx Counters    176184928 output transit packets, 176182865992 bytes
0 drops

The following is a sample output of show mpls forwarding-table network
mask command.

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop

Label      Label      or Tunnel Id     Switched      interface

57         No Label   1.1.1.1/32       0             Po45       145.1.1.1

           No Label   1.1.1.1/32       0             Te1/0/2    147.1.1.1

           No Label   1.1.1.1/32       0             Te1/0/11   149.1.1.1

           No Label   1.1.1.1/32       0             Te1/0/40   155.1.1.1
```

疑似回線冗長性の設定

疑似回線冗長性について

L2VPN擬似回線冗長性機能を使用すると、ネットワーク内の障害を検出して、サービスの提供を続行可能な別のエンドポイントにレイヤ2（L2）サービスを再ルーティングするようにネットワークを設定できます。この機能を使用すると、リモート プロバイダー エッジ（PE）ルータまたはPE とカスタマー エッジ（CE）ルータの間のリンクの障害から復旧できます。

疑似回線冗長性（PWR）は、Xconnect とプロトコル CLI 方式の両方を使用して設定できます。

疑似回線冗長性の前提条件

- Xconnect を設定して接続回線に接続する前に、**no switchport**, **no keepalive** と **no ip address** を設定します。
- ロードバランシングを行うには、**port-channel load-balance** コマンドを設定する必要があります。

疑似回線冗長性の制約事項

- VLAN モード、EFP(イーサネット フロー ポイント)、および IGMP スヌーピングはサポートされていません。
- PWR は、ポート モードの EoMPLS のみでサポートされています。
- タグなし、タグ付き、802.1Q in 802.1Q が着信トラフィックとしてサポートされています。
- カスタマーの送信元 IP、宛先 IP、送信元 MAC アドレス、および宛先 MAC に基づいたコアネットワークでの ECMP ロード バランシングのフロー ラベル。
- 制御ワードのイネーブル化またはディセーブル化がサポートされています。
- MPLS QoS は、パイプおよび均一モードでサポートされています。デフォルト モードはパイプ モードです。
- ポートチャネルは接続回線としてサポートされていません。
- QoS：カスタマー DSCP 再マーキングは VPWS と EoMPLS ではサポートされていません。
- 明示的 null の VCCV ping はサポートされていません。
- L2 VPN インターワーキングはサポートされていません。
- 複数のバックアップ擬似回線はサポートされていません。
- PW 冗長グループのスイッチオーバーはサポートされていません。

疑似回線冗長性の設定

疑似回線冗長性は、2 つのモードで設定できます。

- Xconnect モード
- プロトコル CLI 方式

Xconnect モード

Xconnect モードで疑似回線冗長性を設定するには、次の作業を実行します。

（注） ロード バランスをイネーブルにするには、「ポートモード EoMPLS の設定」のXconnect モード, （35 ページ）の項の対応する load-balance コマンドを使用します。

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 モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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 GigabitEthernet1/0/44`	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 switchport** 例： `Device(config-if)# no switchport`	物理ポートに限り、レイヤ 3 モードを開始します。

	コマンドまたはアクション	目的
ステップ 5	**no ip address** 例： `Device(config-if)# no ip address`	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	**no keepalive** 例： `Device(config-if)# no keepalive`	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 7	**xconnect** *peer-device-id vc-id* **encapsulation mpls** 例： `Device(config-if)# xconnect 1.1.1.1 117 encapsulation mpls`	接続回線を疑似接続 VC にバインドします。このコマンドの構文は、その他のレイヤ 2 トランスポートの場合と同じです。
ステップ 8	**backup peer** *peer-router-ip-addr* **vcid** *vc-id* [**priority** *value*] 例： `Device(config-if)# backup peer 6.6.6.6 118 priority 9`	疑似回線仮想回線（VC）用に冗長ピアを指定します。
ステップ 9	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

プロトコル CLI 方式

プロトコル CLI モードで疑似回線冗長性を設定するには、次の作業を実行します。

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モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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 GigabitEthernet2/0/39`	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 switchport** 例： `Device(config-if)# no switchport`	物理ポートに限り、レイヤ3モードを開始します。
ステップ 5	**no ip address** 例： `Device(config-if)# no ip address`	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	**no keepalive** 例： `Device(config-if)# no keepalive`	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 7	**exit** 例： `Device(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 8	**interface pseudowire** *number* 例： `Device(config)# interface pseudowire 101`	指定した値でインターフェイス擬似回線を確立して、擬似回線コンフィギュレーション モードを開始します。
ステップ 9	**encapsulation mpls** 例： `Device(config-if)# encapsulation mpls`	トンネリング カプセル化を指定します。
ステップ 10	**neighbor** *peer-device-id vc-id* 例： `Device(config-if)# neighbor 4.4.4.4 101`	レイヤ 2 VPN（L2VPN）疑似回線のピア IP アドレスと仮想回線（VC）ID を指定します。
ステップ 11	**exit** 例： `Device(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了します。
ステップ 12	**interface pseudowire** *number* 例： `Device(config)# interface pseudowire 102`	指定した値でインターフェイス擬似回線を確立して、擬似回線コンフィギュレーション モードを開始します。
ステップ 13	**encapsulation mpls** 例：	

疑似回線冗長性の設定例

PE の設定	CE の設定
mpls ip mpls label protocol ldp mpls ldp graceful-restart mpls ldp router-id loopback 1 force ! interface Loopback1 ip address 1.1.1.1 255.255.255.255 ip ospf 100 area 0 router ospf 100 router-id 1.1.1.1 nsf ! interface GigabitEthernet2/0/39 no switchport no ip address no keepalive ! interface pseudowire101 encapsulation mpls neighbor 4.4.4.4 101 ! interface pseudowire102 encapsulation mpls neighbor 3.3.3.3 101 l2vpn xconnect context pw101 member pseudowire101 group pwgrp1 priority 1 member pseudowire102 group pwgrp1 priority 15 member GigabitEthernet2/0/39 ! interface TenGigabitEthernet3/0/10 switchport trunk allowed vlan 142 switchport mode trunk channel-group 42 mode active ! interface Port-channel42 switchport trunk allowed vlan 142 switchport mode trunk ! interface Vlan142 ip address 142.1.1.1 255.255.255.0 ip ospf 100 area 0 mpls ip mpls label protocol ldp !	interface GigabitEthernet1/0/33 switchport trunk allowed vlan 912 switchport mode trunk spanning-tree portfast trunk ! interface Vlan912 ip address 10.91.2.3 255.255.255.0 !

次に、**show mpls l2transport vc vc-id** コマンドの出力例を示します。

```
Local intf   Local circuit              Dest address VC ID    Status
------------ -------------------------- ---------------      ----------
 ----------
Gi2/0/39   Ethernet                    4.4.4.4            101          UP


show mpls l2transport vc 102
Local intf   Local circuit              Dest address VC ID    Status
------------ -------------------------- ---------------      ----------
```

```
 ----------
Gi2/0/39   Ethernet                    3.3.3.3            102
STANDBY
```

第 5 章

MPLS を介した IPv6 プロバイダー エッジ（6PE）の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

6PE の設定

6PE について

6PE は、IPv4 MPLS を介してグローバル IPv6 到達可能性を提供する技術です。これにより、他のすべてのデバイスに対して 1 つの共有ルーティング テーブルを使用できるようになります。6PE を使用することで、IPv6 ドメインは IPv4 を介して相互に通信できるようになります。IPv6 ドメインごとに 1 つの IPv4 アドレスのみが必要であり、明示的にトンネルを設定する必要はありません。

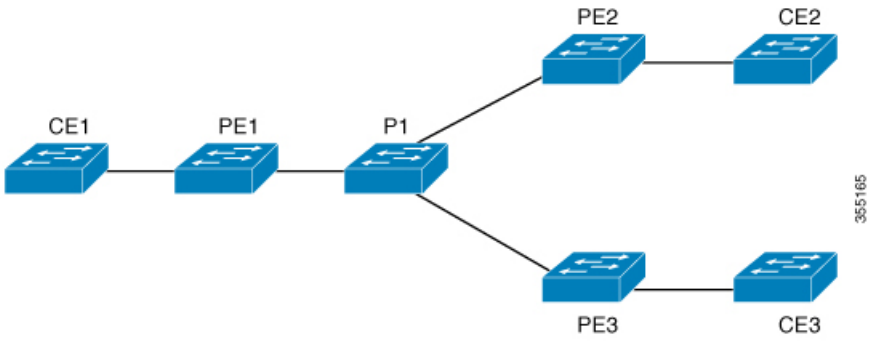
6PE 実装時は、プロバイダー エッジ ルータが 6PE をサポートするようにアップグレードされますが、残りのコア ネットワークに影響することはありません（IPv6 非対応）。転送が IP ヘッダー

自体ではなくラベルに基づいて行われるため、この実装にはコア ルータの再設定は必要ありません。これにより、IPv6 の導入を費用効率性の高い戦略で実現できます。マルチプロトコル ボーダー ゲートウェイ プロトコル（mp-iBGP）の拡張機能を使用して PE ルータによって IPv6 到達可能性情報が交換されます。

6PE は PE ルータの IPv4 ネットワーク設定の mp-iBGP に基づき、アドバタイズする各 IPv6 アドレス プレフィックスの MPLS の他にIPv6 到達可能性情報を交換します。PE ルータは、IPv4 と IPv6 の両方を実行するデュアル スタックとして設定され、IPv4 マッピング IPv6 アドレスを使用して IPv6 プレフィックスの到達可能性情報を交換します。6PE および 6VPE プレフィックスついて PE ルータがアドバタイズするネクスト ホップは、この場合も IPv4 L3 VPN ルートに使用される IPv4 アドレスです。値 ::FFFF: が IPv4 ネクスト ホップの先頭に追加されます。これは、IPv4 マッピングの IPv6 アドレスです。

次の図に 6PE トポロジを示します。

図 *3*：*6PE* トポロジ



スケール番号

プラットフォーム	3650	3850	9300	9500
MPLS L3VPN VRF	127	127	256	256
MPLS L3VPN ルート VRF （すべてのプロバイダー エッジが VRF 単位のラベル割り当てモードであること）	7k	7k	7k	32k
MPLS L3VPN ルート プレフィックス	3k	3k	3k	4k

** 表に示されているプレフィックス番号は IPv4 の場合に使用します。IPv6 の番号は、IPv4 の半分になります。

6PE の前提条件

PE-CE IGP IPv6 ルートをコア BGP に再配布し、また、コア BGP を PE-CE IGP IPv6 ルートに再配布します。

6PE の制約事項

eBGP は CE-PE としてサポートされていません。スタティック ルート、OSPFv3、ISIS、RIPv2 は CE-PE としてサポートされています。

6PE の設定

6PE を設定する PE ルータが IPv4 クラウドおよび IPv6 クラウドの両方に参加していることを確認します。

PE ルータ上で実行する BGP は、他の PE で実行する BGP と（IPv4）ネイバー関係を確立する必要があります。その後、IPv6 テーブルから学習した IPv6 プレフィックスをそれらのネイバーにアドバタイズする必要があります。BGP がアドバタイズした IPv6 プレフィックスには、アドバタイズメントのネクストホップアドレスとして IPv4 エンコードの IPv6 アドレスが自動的に設定されます。

6PE 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 モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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	**ipv6 unicast-routing** 例： `Device(config)# ipv6`	IPv6 ユニキャスト データグラムの転送をイネーブルにします。

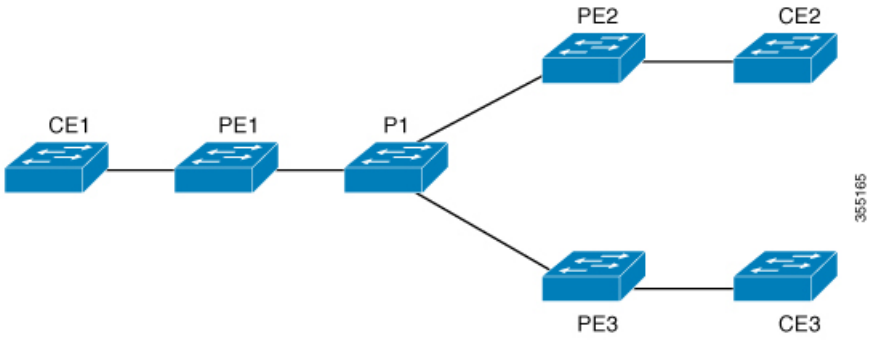
	コマンドまたはアクション	目的
	unicast-routing	
ステップ 4	**router bgp** *as-number* 例： `Device(config)# router bgp 65001`	ルータが存在する自律システム（AS）を識別する番号を入力します。 *as-number*：自律システム番号。2 バイトの番号の範囲は 1 ～ 65535 です。4 バイトの番号の範囲は 1.0 ～ 65535.65535 です。
ステップ 5	**bgp router-id interface** *interface-id* 例： `Device(config-router)# bgp router-id interface Loopback1`	ローカル ボーダー ゲートウェイ プロトコル（BGP）ルーティング プロセスの固定ルータ ID を設定します。
ステップ 6	**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 7	**bgp graceful-restart** 例： `Device(config-router)# bgp graceful-restart`	すべての Border Gateway Protocol（BGP）ネイバーで BGP グレースフル リスタート機能をグローバルで有効にします。
ステップ 8	**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **remote-as** *as-number* 例： `Device(config-router)# neighbor 33.33.33.33 remote-as 65001`	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。 • *ip-address*：ルーティング情報を交換するピア ルータの IP アドレス。 • *ipv6-address*：ルーティング情報を交換するピア ルータの IPv6 アドレス。 • *peer-group-name*：BGP ピア グループの名前。 • *remote-as*：リモート自律システムを指定します。 • *as-number*：ネイバーが属する自律システムの 1 ～ 65535 の範囲内の番号。

	コマンドまたはアクション	目的
ステップ 9	**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **update-source** *interface-type interface-number* 例： `Device(config-router)# neighbor 33.33.33.33 update-source Loopback1`	BGP セッションが TCP 接続の動作インターフェイスを使用できるように設定します。
ステップ 10	**address-family ipv6** 例： `Device(config-router)# address-family ipv6`	標準 IPv6 アドレス プレフィックスを使用する BGP などのルーティング セッションを設定するために、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。
ステップ 11	**redistribute** *protocol as-number***match** { **internal** | **external 1** | **external 2** 例： `Device(config-router-af)# redistribute ospf 11 match internal external 1`	ルートを 1 つのルーティング ドメインから他のルーティング ドメインに再配布します。
ステップ 12	**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **activate** 例： `Device(config-router-af)# neighbor 33.33.33.33 activate`	BGP ネイバーとの情報交換を有効にします。
ステップ 13	**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **send-label** 例： `Device(config-router-af)# neighbor 33.33.33.33 send-label`	隣接 BGP ルータに BGP ルートを含む MPLS ラベルを送信します。
ステップ 14	**exit-address-family** 例： `Device(config-router-af)# exit-address-family`	BGP アドレス ファミリ サブモードを終了します。

	コマンドまたはアクション	目的
ステップ 15	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

6PE の設定例

図 *4*：*6PE* トポロジ



PE の設定	CE の設定

`address-family ipv6 unicast`
`redistribute bgp 65001`
`exit-address-family`
`!`
`router bgp 65001`
`bgp router-id interface Loopback1`
`bgp log-neighbor-changes`
`bgp graceful-restart`
`neighbor 33.33.33.33 remote-as 65001`
`neighbor 33.33.33.33 update-source Loopback1`
`!`
`address-family ipv4`
`neighbor 33.33.33.33 activate`
`!`
`address-family ipv6`
`redistribute ospf 11 match internal external 1 external 2`
`include-connected`
`neighbor 33.33.33.33 activate`
`neighbor 33.33.33.33 send-label`
`!`
	`ipv6 unicast-routing`
	`!`
	`interface vlan4`
	`no ip address`
	`ipv6 address 10:1:1:2::2/64`
	`ipv6 enable`
	`ospfv3 11 ipv6 area 0`
	`!`
	`router ospfv3 11`
	`address-family ipv6 unicast`
	`exit-address-family`
	`!`

次に、**show bgp ipv6 unicast summary** の出力例を示します。

```
BGP router identifier 1.1.1.1, local AS number 100
BGP table version is 34, main routing table version 34
4 network entries using 1088 bytes of memory
4 path entries using 608 bytes of memory
4/4 BGP path/bestpath attribute entries using 1120 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 2816 total bytes of memory
BGP activity 6/2 prefixes, 16/12 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
  State/PfxRcd
2.2.2.2         4          100      21      21       34    0    0 00:04:57
        2

sh ipv route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, I1 - ISIS L1, I2 - ISIS L2
       IA - ISIS interarea, IS - ISIS summary, D - EIGRP, EX - EIGRP
external
      ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect

       RL - RPL, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid lA
 - LISP away
C   10:1:1:2::/64 [0/0]
     via Vlan4, directly connected
L   10:1:1:2::1/128 [0/0]
     via Vlan4, receive
LC  11:11:11:11::11/128 [0/0]
     via Loopback1, receive
B   30:1:1:2::/64 [200/0]
     via 33.33.33.33%default, indirectly connected
B   40:1:1:2::/64 [200/0]
     via 44.44.44.44%default, indirectly connected
```

次に、**show bgp ipv6 unicast** コマンドの出力例を示します。

```
BGP table version is 112, local router ID is 11.11.11.11
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
 *>   10:1:1:2::/64    ::                       0         32768 ?
 *>i  30:1:1:2::/64    ::FFFF:33.33.33.33
                                                0    100      0 ?
 *>i  40:1:1:2::/64    ::FFFF:44.44.44.44
                                                0    100      0 ?
 *>i  173:1:1:2::/64   ::FFFF:33.33.33.33
                                                2    100      0 ?
```

次に、**show ipv6 cef 40:1:1:2::0/64 detail** コマンドの出力例を示します。

```
40:1:1:2::/64, epoch 6, flags [rib defined all labels]
  recursive via 44.44.44.44 label 67
    nexthop 1.20.4.2 Port-channel103 label 99-(local:147)
```

第 6 章

MPLS を介した IPv6 VPN プロバイダー エッジ（6VPE）の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

6VPE の設定

6VPE について

6VPE は IPv4 バックボーンを使用して VPN IPv6 サービスを提供するメカニズムです。使用可能な IPv4 MPLS バックボーンを利用することで、MPLS コア内でのデュアルスタッキングが不要になります。つまり、運用コストを節減し、6PE アプローチのセキュリティ上の制限に対処します。6VPE は、通常の IPv4 MPLS-VPN プロバイダー エッジとほぼ同じですが、VRF 内に IPv6 サポートが追加されています。これは、VPN メンバーデバイス用に、論理的に分割されたルーティング テーブル エントリを提供します。

MPLS ベースの 6VPE ネットワークのコンポーネント

- VPN ルート ターゲット コミュニティ：VPN コミュニティのその他すべてのメンバのリスト。
- VPN コミュニティ PE ルータのマルチプロトコル BGP（MP-BGP）ピアリング：VPN コミュニティのすべてのメンバに VRF 到達可能性情報を伝播します。
- MPLS 転送：VPN サービスプロバイダー ネットワークのすべての VPN コミュニティ メンバ間にすべてのトラフィックを転送します。

MPLS VPN モデルでは共通のルーティング テーブルを共有するサイトの集合として VPN が定義されます。カスタマー サイトは1つ以上のインターフェイスでサービス プロバイダー ネットワークに接続され、サービス プロバイダーは、VRF テーブルと呼ばれる VPN ルーティング テーブルと各インターフェイスを関連付けます。

スケール番号

プラットフォーム	3650	3850	9300	9500
MPLS L3VPN VRF	127	127	256	256
MPLS L3VPN ルート VRF （すべてのプロバイダー エッジが VRF 単位のラベル割り当てモードであること）	7k	7k	7k	32k
MPLS L3VPN ルート プレフィックス	3k	3k	3k	4k

** 表に示されているプレフィックス番号は IPv4 の場合に使用します。IPv6 の番号は、IPv4 の半分になります。

6VPE の制約事項

- Inter-AS および Carrier Supporting Carrier（CSC）はサポートされていません。
- VRF ルートリーキングはサポートされていません。
- EIGRP と eBGP は CE-PE としてサポートされていません。
- OSPFv3、RIP、ISIS、スタティック ルートは、CE-PE としてサポートされています。

- サポートされている MPLS ラベル割り当てモードは VRF 単位とプレフィックス単位です。プレフィックス単位がデフォルトのモードです。

6VPE について

6VPE は IPv4 バックボーンを使用して VPN IPv6 サービスを提供するメカニズムです。使用可能な IPv4 MPLS バックボーンを利用することで、MPLS コア内でのデュアルスタッキングが不要になります。つまり、運用コストを節減し、6PE アプローチのセキュリティ上の制限に対処します。6VPE は、通常の IPv4 MPLS-VPN プロバイダー エッジとほぼ同じですが、VRF 内に IPv6 サポートが追加されています。これは、VPN メンバーデバイス用に、論理的に分割されたルーティング テーブル エントリを提供します。

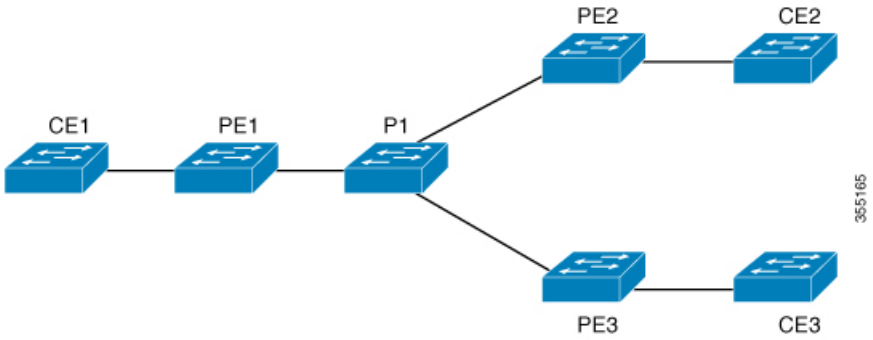
MPLS ベースの 6VPE ネットワークのコンポーネント

- VPN ルート ターゲット コミュニティ：VPN コミュニティのその他すべてのメンバのリスト。
- VPN コミュニティ PE ルータのマルチプロトコル BGP（MP-BGP）ピアリング：VPN コミュニティのすべてのメンバに VRF 到達可能性情報を伝播します。
- MPLS 転送：VPN サービスプロバイダー ネットワークのすべての VPN コミュニティ メンバ間にすべてのトラフィックを転送します。

MPLS VPN モデルでは共通のルーティング テーブルを共有するサイトの集合として VPN が定義されます。カスタマー サイトは 1 つ以上のインターフェイスでサービス プロバイダー ネットワークに接続され、サービス プロバイダーは、VRF テーブルと呼ばれる VPN ルーティング テーブルと各インターフェイスを関連付けます。

6VPE の設定例

図 *5* : *6VPE* トポロジ



PE の設定	CE の設定
	interface TenGigabitEthernet1/0/38 no switchport ip address 10.3.1.2 255.255.255.0 ip ospf 2 area 0 ipv6 address 10:111:111:111::2/64 ipv6 enable ipv6 ospf 1 area 0 ! router ospfv3 1 nsr graceful-restart address-family ipv6 unicast !

PE の設定	CE の設定
vrf definition 6VPE-1 rd 65001:11 route-target export 1:1 route-target import 1:1 ! address-family ipv4 exit-address-family ! address-family ipv6 exit-address-family ! interface TenGigabitEthernet1/0/38 no switchport vrf forwarding 6VPE-1 ip address 10.3.1.1 255.255.255.0 ip ospf 2 area 0 ipv6 address 10:111:111:111::1/64 ipv6 enable ospfv3 1 ipv6 area 0 ! router ospf 2 vrf 6VPE-1 router-id 1.1.11.11 redistribute bgp 65001 subnets ! router ospfv3 1 nsr graceful-restart ! address-family ipv6 unicast vrf 6VPE-1 redistribute bgp 65001 exit-address-family ! router bgp 65001 bgp router-id interface Loopback1 bgp log-neighbor-changes bgp graceful-restart neighbor 33.33.33.33 remote-as 65001 neighbor 33.33.33.33 update-source Loopback1 ! address-family ipv4 vrf 6VPE-1 redistribute ospf 2 match internal external 1 external 2 exit-address-family address-family ipv6 vrf 6VPE-1 redistribute ospf 1 match internal external 1 external 2 include-connected exit-address-family ! address-family vpnv4 neighbor 33.33.33.33 activate neighbor 33.33.33.33 send-community both neighbor 44.44.44.44 activate neighbor 44.44.44.44 send-community both neighbor 55.55.55.55 activate neighbor 55.55.55.55 send-community both exit-address-family ! address-family vpnv6 neighbor 33.33.33.33 activate neighbor 33.33.33.33 send-community both neighbor 44.44.44.44 activate neighbor 44.44.44.44 send-community both neighbor 55.55.55.55 activate neighbor 55.55.55.55 send-community both exit-address-family !	

次に、**show mpls forwarding-table vrf** の出力例を示します。

```
Local Outgoing Prefix Bytes Label Outgoing Next Hop
Label Label or Tunnel Id Switched interface
29 No Label A:A:A:565::/64[V] \ 0 aggregate/VRF601
32 No Label A:B5:1:5::/64[V] 2474160 Vl601 FE80::200:7BFF:FE62:2636
33 No Label A:B5:1:4::/64[V] 2477978 Vl601 FE80::200:7BFF:FE62:2636
35 No Label A:B5:1:3::/64[V] 2477442 Vl601 FE80::200:7BFF:FE62:2636
36 No Label A:B5:1:2::/64[V] 2476906 Vl601 FE80::200:7BFF:FE62:2636
37 No Label A:B5:1:1::/64[V] 2476370 Vl601 FE80::200:7BFF:FE62:2636
```

次に、**show vrf counter** コマンドの出力例を示します。

```
Maximum number of VRFs supported: 256
Maximum number of IPv4 VRFs supported: 256
Maximum number of IPv6 VRFs supported: 256
Maximum number of platform iVRFs supported: 10
Current number of VRFs: 127
Current number of IPv4 VRFs: 6
Current number of IPv6 VRFs: 127
Current number of VRFs in delete state: 0
Current number of platform iVRFs: 1
```

次に、**show ipv6 route vrf** コマンドの出力例を示します。

```
IPv6 Routing Table - VRF1 - 8 entries Codes: C - Connected, L - Local, S
 - Static, U - Per-user Static route B - BGP, R - RIP, I1 - ISIS L1, I2
- ISIS L2 IA - ISIS interarea, IS - ISIS summary, D - EIGRP, EX - EIGRP
external ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr -
Redirect RL - RPL, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1 OE2
 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2 la - LISP
alt, lr - LISP site-registrations, ld - LISP dyn-eid lA - LISP away

B 1:1:1:1::1/128 [200/1] via 1.1.1.11%default, indirectly connected
O 2:2:2:2::2/128 [110/1] via FE80::A2E0:AFFF:FE30:3E40,
TenGigabitEthernet1/0/7
B 3:3:3:3::3/128 [200/1] via 3.3.3.33%default, indirectly connected
B 10:1:1:1::/64 [200/0] via 1.1.1.11%default, indirectly connected
C 10:2:2:2::/64 [0/0] via TenGigabitEthernet1/0/7, directly connected
L 10:2:2:2::1/128 [0/0] via TenGigabitEthernet1/0/7, receive
B 10:3:3:3::/64 [200/0] via 3.3.3.33%default, indirectly connected
L FF00::/8 [0/0] via Null0, receive
```

第 7 章

MPLS レイヤ 3 VPN の設定

MPLS バーチャル プライベート ネットワーク（VPN）は、マルチプロトコル ラベル スイッチング（MPLS）プロバイダー コア ネットワークによって相互接続された一連のサイトで構成されます。各カスタマー サイトでは、1 つ以上のカスタマー エッジ（CE）デバイスが、1 つ以上のプロバイダー エッジ（PE）デバイスに接続されます。このモジュールでは、MPLS レイヤ 3 VPN の作成方法について説明します。

- MPLS レイヤ 3 VPNs，65 ページ

MPLS レイヤ 3 VPNs

MPLS バーチャル プライベート ネットワーク（VPN）は、マルチプロトコル ラベル スイッチング（MPLS）プロバイダー コア ネットワークによって相互接続された一連のサイトで構成されます。各カスタマー サイトでは、1 つ以上のカスタマー エッジ（CE）デバイスが、1 つ以上のプロバイダーエッジ（PE）デバイスに接続されます。この章では、MPLS VPN の作成方法について説明します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

MPLS バーチャル プライベート ネットワークの前提条件

- マルチプロトコル ラベル スイッチング（MPLS）、ラベル配布プロトコル（LDP）、および Cisco Express Forwarding がネットワークにインストールされていることを確認します。
- プロバイダー エッジ（PE）デバイスを含む、コア内のすべてのデバイスは、シスコ エクスプレス フォワーディングおよびMPLS転送をサポートできる必要があります。「MPLS バーチャル プライベート ネットワーク カスタマーのニーズの評価」を参照してください。
- Cisco Express Forwarding は、PE デバイスを含め、コア内のすべてのデバイスでイネーブルにする必要があります。Cisco Express Forwarding がイネーブルになっているかどうかを確認する方法については、『*Cisco Express Forwarding Configuration Guide*』の「Configuring Basic Cisco Express Forwarding」の章を参照してください。

MPLS バーチャル プライベート ネットワークの制約事項

マルチプロトコル ラベル スイッチング（MPLS）またはMPLS バーチャル プライベート ネットワーク（VPN）環境でスタティック ルートを設定する場合は、**ip route** コマンドおよび **ip route vrf** コマンドの一部のバリエーションがサポートされません。スタティック ルートを設定するときは、次の注意事項に従ってください。

MPLS 環境でサポートされるスタティック ルート

MPLS環境でスタティック ルートを設定する場合は、次の **ip route** コマンドがサポートされます。

- **ip route** *destination-prefix mask interface next-hop-address*

MPLS 環境でスタティック ルートを設定し、スタティックな非再帰ルートと特定の発信インターフェイスを使用するロード シェアリングを設定する場合は、次の **ip route** コマンドがサポートされます。

- **ip route** *destination-prefix mask***interface1next-hop1**
- **ip route** *destination-prefix mask***interface2next-hop2**

TFIB を使用する MPLS 環境でサポートされないスタティック ルート

MPLS 環境でスタティック ルートを設定する場合は、次の **ip route** コマンドがサポートされません。

- **ip route** *destination-prefix mask next-hop-address*

MPLS環境でスタティック ルートを設定し、2つのパスでネクストホップに到達できる場所でロード シェアリングをイネーブルにする場合は、次の **ip route** コマンドがサポートされません。

- **ip route** *destination-prefix mask next-hop-address*

MPLS環境でスタティック ルートを設定し、2つのネクストホップで宛先に到達できる場所でロード シェアリングをイネーブルにする場合は、次の **ip route** コマンドがサポートされません。

- **ip route** *destination-prefix mask***next-hop1**
- **ip route** *destination-prefix mask***next-hop2**

スタティック ルートを指定する場合は、*interface* an *next-hop* 引数を使用します。

MPLS VPN 環境でサポートされるスタティック ルート

次の **ip route vrf** コマンドは、MPLS VPN 環境でスタティック ルートを設定し、ネクスト ホップとインターフェイスが同じ VRF に存在する場合はサポートされません。

- **ip route vrf** *vrf-name destination-prefix mask next-hop-address*
- **ip route vrf** *vrf-name destination-prefix mask interface next-hop-address*
- **ip route vrf** *vrf-name destination-prefix mask***interface1next-hop1**
- **ip route vrf** *vrf-name destination-prefix mask***interface2next-hop2**

MPLS VPN環境でスタティック ルートを設定し、ネクスト ホップがグローバル ルーティング テーブルの MPLS クラウドのグローバル テーブルに存在する場合は、次の **ip route vrf** コマンドがサポートされます。たとえば、ネクスト ホップがインターネット ゲートウェイを指している場合は、次のコマンドがサポートされます。

- **ip route vrf** *vrf-name destination-prefix mask next-hop-address***global**
- **ip route vrf** *vrf-name destination-prefix mask interface next-hop-address*（このコマンドは、ネクスト ホップおよびインターフェイスがコアにある場合にサポートされます。）

MPLS VPN 環境でスタティック ルートを設定し、スタティックな非再帰ルートと特定の発信インターフェイスを使用するロード シェアリングをイネーブルにする場合は、次の **ip route** コマンドがサポートされます。

- **ip route** *destination-prefix mask***interface1next-hop1**
- **ip route** *destination-prefix mask***interface2next-hop2**

TFIB を使用する MPLS VPN 環境でサポートされないスタティック ルート

MPLS VPN 環境でスタティック ルートを設定し、ネクスト ホップがコア内の MPLS クラウドのグローバル テーブルに存在し、2 つのパスでネクスト ホップに到達できる場所でロード シェアリングをイネーブルにする場合は、次の**ip route** コマンドがサポートされません。

- **ip route vrf** *destination-prefix mask next-hop-address***global**

MPLS VPN 環境でスタティック ルートを設定し、ネクスト ホップがコア内の MPLS クラウドのグローバル テーブルに存在し、2 つのネクストホップで宛先に到達できる場所でロード シェアリングを有効にする場合は、次の**ip route** コマンドがサポートされません。

- **ip route vrf** *destination-prefix mask***next-hop1global**

- **ip route vrf** *destination-prefix mask***next-hop2global**

MPLS VPN 環境でスタティック ルートを設定し、ネクスト ホップおよびインターフェイスが同じ VRF にある場合は、次の **iproutevrf** コマンドがサポートされません。

- **ip route vrf** *vrf-name destination-prefix mask***next-hop1** *vrf-name destination-prefix mask***next-hop1**
- **ip route vrf** *vrf-name destination-prefix mask***next-hop2**

ネクスト ホップが CE デバイス上のグローバル テーブルに存在する MPLS VPN 環境でサポートされるスタティック ルート

MPLS VPN 環境でスタティック ルートを設定し、ネクスト ホップがカスタマー エッジ（CE）側のグローバル テーブルにある場合は、次の **ip route vrf** コマンドがサポートされます。たとえば、外部ボーダー ゲートウェイ プロトコル（EBGP）マルチホップの場合と同様に、宛先プレフィックスが CE デバイスのループバック アドレスである場合は、次のコマンドがサポートされます。

- **ip route vrf** *vrf-name destination-prefix mask interface next-hop-address*

MPLS VPN 環境でスタティック ルートを設定し、ネクスト ホップが CE 側のグローバル テーブルに存在し、スタティックな非再帰ルートと特定の発信インターフェイスを使用するロード シェアリングをイネーブルにする場合は、次の **ip route** コマンドがサポートされます。

- **ip route** *destination-prefix mask***interface1nexthop1**
- **ip route** *destination-prefix mask***interface2nexthop2**

MPLS バーチャル プライベート ネットワークに関する情報

MPLS バーチャル プライベート ネットワークの定義

マルチプロトコル ラベル スイッチング バーチャル プライベート ネットワーク（MPLS VPN）を定義する前に、一般的な VPN を定義する必要があります。VPN の説明を次に示します。

- パブリック インフラストラクチャを介してプライベート ネットワーク サービスを提供する、IP ベースのネットワーク
- インターネットまたはその他のパブリック ネットワークやプライベート ネットワークを介してプライベートに相互通信できる一連のサイト

通常の VPN は、完全メッシュのトンネル、または相手先固定接続（PVC）を VPN 内のすべてのサイトに設定することで作成されます。このタイプの VPN は、新しいサイトを追加した場合に VPN 内の各エッジデバイスを変更する必要があるため、維持または拡張が簡単ではありません。

MPLS ベースの VPN は、レイヤ 3 に作成され、ピア モデルに基づきます。ピア モデルによって、サービス プロバイダーおよびカスタマーは、レイヤ 3 のルーティング情報を交換できます。サービス プロバイダーは、カスタマー サイト間でデータをリレーします。このとき、カスタマー側では何をする必要もありません。

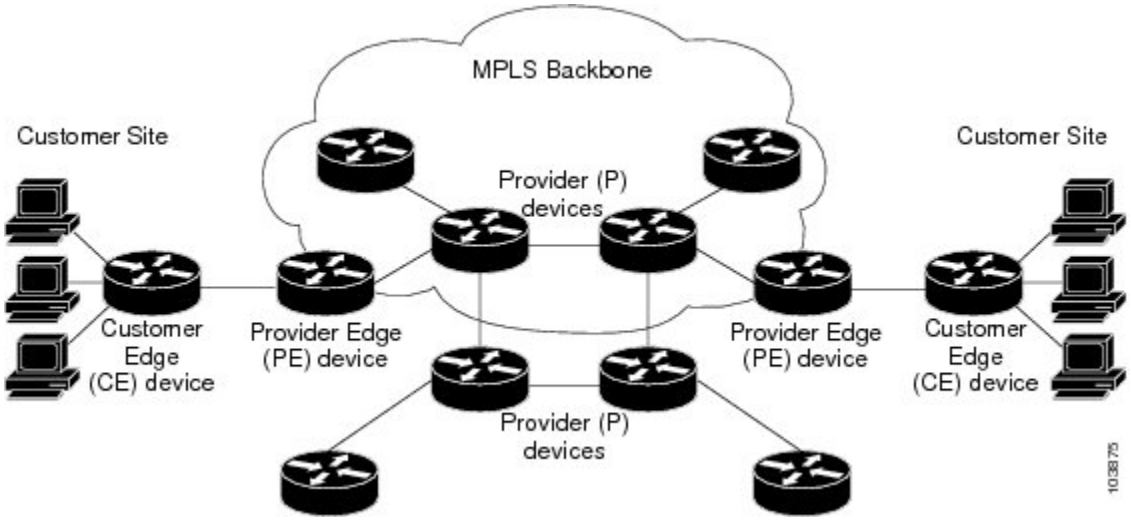
MPLS VPN の管理や拡張は、従来の VPN よりも簡単です。新しいサイトが MPLS VPN に追加された場合、更新する必要があるのは、カスタマーサイトにサービスを提供するサービスプロバイダーのエッジ デバイスだけです。

MPLS VPN のさまざまな部分について、次に説明します。

- プロバイダー（P）デバイス：プロバイダー ネットワークのコア内のデバイス。P デバイスは MPLS スイッチングを実行し、ルーティングされるパケットに VPN ラベルを付加しません。各ルートの MPLS ラベルは、プロバイダー エッジ（PE）デバイスによって割り当てられます。VPN ラベルは、データ パケットを正しい出力デバイスに誘導するために使用されます。
- PE デバイス：着信パケットが受信されるインターフェイスまたはサブインターフェイスに基づいて、着信パケットに VPN ラベルを付加するデバイス。PE デバイスは、カスタマー エッジ（CE）デバイスに直接接続されます。
- カスタマー（C）デバイス：ISP または企業ネットワークのデバイス。
- CE デバイス：ネットワーク上の PE デバイスに接続する、ISP のネットワーク上のエッジ デバイス。CE デバイスは、PE デバイスとインターフェイスする必要があります。

次の図に、基本的な MPLS VPN を示します。

図 *6*：基本的 *MPLS VPN* 用語



MPLS バーチャル プライベート ネットワークの仕組み

マルチプロトコル ラベル スイッチング バーチャル プライベート ネットワーク（MPLS VPN）機能は、MPLS ネットワークのエッジでイネーブルになっています。プロバイダー エッジ（PE）デバイスは、次の機能を実行します。

- カスタマー エッジ（CE）デバイスとルーティング アップデートを交換する。

- CE ルーティング情報を VPNv4 ルートに変換する。
- マルチプロトコル ボーダー ゲートウェイ プロトコル（MP-BGP）を介して、他の PE デバイスと VPNv4 ルートを交換する。

ここでは、MPLS VPN の機能について説明します。

MPLS バーチャル プライベート ネットワークの主要コンポーネント

マルチプロトコル ラベル スイッチング（MPLS）ベースのバーチャル プライベート ネットワーク（VPN）には、次の 3 つの主要コンポーネントがあります。

- VPN ルート ターゲット コミュニティ：VPN ルート ターゲット コミュニティは、VPN コミュニティのすべてのメンバのリストです。VPN ルート ターゲットは、各 VPN コミュニティ メンバに設定する必要があります。
- VPN コミュニティ プロバイダー エッジ（PE）デバイスのマルチプロトコル BGP（MP-BGP）ピアリング：MP-BGP は、VPN コミュニティのすべてのメンバに仮想ルーティングおよび転送（VRF）到達可能性情報を伝播します。MP-BGP ピアリングは、VPN コミュニティのすべての PE デバイスで設定されている必要があります。
- MPLS 転送：MPLS は、VPN サービス プロバイダー ネットワーク上のすべての VPN コミュニティ メンバ間のすべてのトラフィックを転送します。

1 対 1 の関係は、カスタマー サイトと VPNs 間に必ずしも存在する必要はありません。1 つの指定されたサイトを複数の VPN のメンバにできます。ただし、サイトは、1 つの VRF とだけ関連付けることができます。カスタマー サイトの VRF には、そのサイトがメンバとなっている VPN からサイトへの、利用できるすべてのルートが含まれています。

MPLS バーチャル プライベート ネットワークの利点

マルチプロトコル ラベル スイッチング バーチャル プライベート ネットワーク（MPLS VPN）を使用すると、サービス プロバイダーは、スケーラブルな VPN を展開し、次のような付加価値サービスを提供する基盤を構築できます。

コネクションレス型サービス

MPLS VPN の重要な技術的メリットとして、コネクションレスであることを挙げることができます。インターネットの成功には、TCP/IP という基礎的な技術が貢献しています。TCP/IP は、パケットを基礎とする、コネクションレス ネットワーク パラダイムに基づいて構築されています。これは、ホスト間の通信を確立するための事前のアクションが不要となり、2 者間の通信が簡単になることを意味します。現在の VPN ソリューションでは、コネクションレス型の IP 環境でプライバシーを確立するために、ネットワーク上でコネクション型ポイントツーポイントのオーバーレイを行っています。VPN がコネクションレス型ネットワーク上で動作しても、VPN では接続の容易さや、コネクションレス型ネットワークで利用できる多様なサービスを活用できません。コネクションレス VPN を作成すると、ネットワーク プライバシーのためのトンネルおよび暗号化が不要となり、その結果、複雑さが大幅に軽減されます。

集中型サービス

レイヤ 3 に VPN を構築すると、VPN に代表されるユーザ グループに目的のサービスを配布できます。VPN がサービス プロバイダーに提供する内容は、ユーザがイントラネット サービスにプライベートに接続するためのメカニズムだけではありません。VPN では、付加価値サービスを対象のカスタマーに柔軟に提供する方法も提供する必要があります。カスタマーがそのイントラネットやエクストラネットでサービスをプライベートに使用できるようにするために、スケーラビリティが重要となります。MPLS VPN は、プライベート イントラネットと見なされ、次のような新しい IP サービスを使用できます。

- マルチキャスト（Multicast）
- Quality of Service（QoS）
- VPN でのテレフォニー サポート
- コンテンツや VPN への Web ホスティングを含む、集中型サービス

カスタマーごとに特化したサービスを、複数組み合わせてカスタマイズできます。たとえば、IP マルチキャストを低遅延のサービス クラスに組み合わせると、ビデオ会議をイントラネット内で実施できます。

拡張性

コネクション型ポイントツーポイントのオーバーレイ、フレーム リレー、または ATM 仮想接続（VC）を使用する VPN を作成する場合、その VPN では、主にスケーラビリティが問題となります。特に、カスタマー サイト間での完全メッシュ接続のないコネクション型 VPN は、最適ではありません。MPLS ベースの VPN では、スケーラビリティの高い VPN ソリューションを活用するために、代わりに、ピア モデルとレイヤ 3 コネクションレス型アーキテクチャを使用します。このピア モデルでは、カスタマー サイトがピアリングする必要があるのは、VPN のメンバであるその他のすべてのカスタマーエッジ（CE）デバイスではなく、1 つのプロバイダーエッジ（PE）デバイスだけとなります。コネクションレス型アーキテクチャによって、レイヤ 3 に VPN を作成することができ、トンネルまたは VC を行う必要がなくなります。

MPLS VPN のその他のスケーラビリティ機能は、PE デバイス間での VPN ルートのパーティショニング、およびコア ネットワークでの PE デバイスとプロバイダー（P）デバイス間での VPN と Interior Gateway Protocol（IGP）ルートのパーティショニングから得られます。

- PE デバイスは、メンバである VPN に対して VPN ルートを維持する必要があります。
- P デバイスでは、VPN ルートを一切維持する必要がありません。

これにより、プロバイダーのコアのスケーラビリティが高まり、いずれのデバイスもスケーラビリティのボトルネックとなりません。

セキュリティ

MPLS VPN はコネクション型 VPN と同じレベルのセキュリティを提供します。1 つの VPN からのパケットが、間違って別の VPN に送信されることはありません。

セキュリティは、次の領域で提供されます。

- プロバイダー ネットワークのエッジでは、カスタマーから受信したパケットが、正しいVPNに配置されることが保証されます。
- バックボーンでは、VPNトラフィックが常に分離されます。悪意のあるスプーフィング（PEデバイスへのアクセスを取得するための試行）は、ほぼ不可能です。これは、カスタマーから受信するパケットがIPパケットであるためです。これらのIPパケットは、VPNラベルと一意に識別される特定のインターフェイスまたはサブインターフェイスで受信される必要があります。

作成の容易さ

VPNを最大限に活用するには、カスタマーは、新しいVPNとユーザ コミュニティを簡単に作成できる必要があります。MPLS VPNはコネクションレスであるため、特定のポイントツーポイント接続マップまたはトポロジは必要ありません。イントラネットやエクストラネットにサイトを追加して、非公開ユーザ グループを形成できます。この方法でVPNを管理すると、指定された任意のサイトを複数のVPNのメンバにできるため、イントラネットやエクストラネットを構築する場合の柔軟性が最大限に高められます。

柔軟なアドレッシング

VPNサービスへのアクセスをより簡単にするために、サービス プロバイダーのカスタマーは、サービス プロバイダーのその他のカスタマーのアドレッシング計画とは関係なく、独自のアドレッシング計画を設計できます。多くのカスタマーは、RFC 1918で定義されているようにプライベート アドレス空間を使用しており、イントラネットの接続性を得るために時間と費用をかけてパブリックIPアドレスに変換することは望んでいません。MPLS VPNを使用すると、カスタマーは、アドレスのパブリック ビューとプライベート ビューを提供することで、ネットワーク アドレス変換（NAT）を使用することなく現在のアドレス空間を引き続き使用できます。NATは、重複するアドレス空間を持つ2つのVPNが通信する必要がある場合にだけ必要となります。これにより、カスタマーは、パブリックIPネットワーク上で、独自の未登録プライベートアドレスを使用して自由に通信できます。

統合 QoS サポート

QoSは、多くのIP VPNカスタマーにとって重要な要件です。統合QoSを使用すると、次の2つの基本的なVPN要件に対処できます。

- 予測可能なパフォーマンスおよびポリシーの実装
- MPLS VPNにおける複数レベルのサービスのサポート

ネットワーク トラフィックは、ネットワークのエッジで分類およびラベル付けされたあとに、加入者によって定義されたポリシーに従って集約され、プロバイダーで実行され、プロバイダー コア経由で転送されます。その後、破棄確率または遅延ごとに、ネットワークのエッジおよびコアでのトラフィックを異なるクラスに分けることができます。

直接的な移行

サービスプロバイダーは、VPNサービスを迅速に展開するために、直接的な移行パスを使用します。MPLS VPNの独自な特長として、IP、ATM、フレーム リレー、およびハイブリッド ネット

ワークを含む、複数のネットワーク アーキテクチャ上に構築できることを挙げることができます。

CEデバイス上でMPLSをサポートする必要がなく、カスタマーのイントラネットに変更を加える必要がないため、エンド ユーザの移行作業は簡単になります。

MPLS バーチャル プライベート ネットワークの設定方法

コア ネットワークの設定

MPLS バーチャル プライベート ネットワーク カスタマーのニーズの評価

マルチプロトコル ラベル スイッチング仮想プライベート ネットワーク（MPLS VPN）を設定する前に、コア ネットワーク トポロジを識別して、MPLS VPN カスタマーに最適なサービスが提供されるようにする必要があります。コア ネットワーク トポロジを識別するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	ネットワークのサイズを識別します。	必要となるデバイスとポートの数を決定するために、次の内容を識別します。 • サポートする必要があるカスタマーの数 • カスタマーごとに必要となる VPN の数 • 各 VPN に存在する、仮想ルーティングおよび転送インスタンスの数
ステップ 2	コアにおけるルーティング プロトコルを識別します。	コア ネットワークで必要なルーティング プロトコルを決定します。
ステップ 3	MPLS VPN ハイ アベイラビリティのサポートが必要であるかどうかを判断します。	MPLS VPN ノンストップ フォワーディングおよびグレースフル リスタートは、選択デバイスおよび Cisco IOS ソフトウェア リリースでサポートされています。Cisco サポートに問い合わせて、正確な要件およびハードウェア サポートを確認してください。
ステップ 4	MPLS VPN コアで Border Gateway Protocol（BGP）ロード シェアリングおよび冗長パスが必要であるかどうかを決定します。	設定手順については、『*MPLS Layer 3 VPNs Inter-AS and CSC Configuration Guide*』の「Load Sharing MPLS VPN Traffic」モジュールを参照してください。

コアにおける MPLS の設定

コアのすべてのデバイスでマルチプロトコル ラベル スイッチング（MPLS）をイネーブルにするには、ラベル配布プロトコルとして次のいずれかを設定する必要があります。

- MPLS ラベル配布プロトコル（LDP）。設定については、『*MPLS Label Distribution Protocol Configuration Guide*』の「MPLS Label Distribution Protocol (LDP)」モジュールを参照してください。
- MPLS トラフィック エンジニアリング リソース予約プロトコル（RSVP）。設定については、『*MPLS Traffic Engineering Path Calculation and Setup Configuration Guide*』の「MPLS Traffic Engineering and Enhancements」モジュールを参照してください。

MPLS バーチャル プライベート ネットワーク カスタマーの接続

カスタマーの接続を可能にするための、PE デバイスでの VRF の定義

次の手順を使用して、IPv4 の仮想ルーティングおよび転送（VRF）設定を定義します。IPv4 と IPv6 の VRF を定義するには、『*MPLS Layer 3 VPNs Configuration Guide*』の「IPv6 VPN over MPLS」モジュールの「Configuring a Virtual Routing and Forwarding Instance for IPv6」セクションを参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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	**ip vrf** *vrf-name* 例： `Device(config)# ip vrf vpn1`	バーチャル プライベート ネットワーク（VRF）名を割り当て、VRF コンフィギュレーション モードを開始することにより、Virtual Routing and Forwarding（VPN）ルーティング インスタンスを定義します。 • *vrf-name* 引数は、VRF に割り当てる名前です。

	コマンドまたはアクション	目的
ステップ 4	**rd** *route-distinguisher* 例： `Device(config-vrf)# rd 100:1`	ルーティング テーブルと転送テーブルを作成します。 • *route-distinguisher* 引数によって、8 バイトの値が IPv4 プレフィックスに追加され、VPN IPv4 プレフィックスが作成されます。ルート識別子（RD）は、次のいずれかの形式で入力できます。 ◦ 16 ビットの AS 番号：32 ビットの番号。101:3 など。 ◦ 32 ビットの IP アドレス：16 ビットの番号。10.0.0.1:1 など。
ステップ 5	**route-target** {**import** | **export** | **both**} *route-target-ext-community* 例： `Device(config-vrf)# route-target both 100:1`	VRF 用にルート ターゲット拡張コミュニティを作成します。 • **import** キーワードを使用すると、ターゲット VPN 拡張コミュニティからルーティング情報がインポートされます。 • **export** キーワードを使用すると、ルーティング情報がターゲット VPN 拡張コミュニティにエクスポートされます。 • **both** キーワードを使用すると、ターゲット VPN 拡張コミュニティとの間でルーティング情報がインポートおよびエクスポートされます。 • *route-target-ext-community* 引数により、route-target 拡張コミュニティ属性が、インポートやエクスポートの route-target 拡張コミュニティの VRF リストに追加されます。
ステップ 6	**exit** 例： `Device(config-vrf)# exit`	（任意）終了して、グローバル コンフィギュレーション モードに戻ります。

各 VPN カスタマー用の PE デバイスでの VRF インターフェイスの設定

プロバイダー エッジ（PE）デバイス上のインターフェイスまたはサブインターフェイスに仮想ルーティングおよび転送（VRF）インスタンスを関連付けるには、次の作業を実行します。

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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** *type number* 例： `Device(config)# interface`	設定するインターフェイスを指定し、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。 • *type* 引数で、設定するインターフェイスのタイプを指定します。 • *number* 引数には、ポート、コネクタ、またはインターフェイス カード番号を指定します。
ステップ 4	**ip vrf** *vrf-name* 例： `Device(config-if)# ip vrf vpn1`	指定したインターフェイスまたはサブインターフェイスに VRF を関連付けます。 • *vrf-name* 引数は、VRF に割り当てる名前です。
ステップ 5	**end** 例： `Device(config-if)# end`	（任意）終了して、特権 EXEC モードに戻ります。

PE デバイスと CE デバイス間でのルーティング プロトコルの設定

カスタマー エッジ（CE）デバイスで使用されているのと同じルーティング プロトコルを使用して、プロバイダー エッジ（PE）デバイスを設定します。Border Gateway Protocol（BGP）、Routing Information Protocol バージョン 2（RIPv2）、または PE デバイスと CE デバイス間のスタティック ルートを設定できます。

バーチャル プライベート ネットワークの設定の確認

ルート識別子を Virtual Routing and Forwarding（VRF）インスタンス用に設定する必要があります。また、VRF を伝送するインターフェイス上でマルチプロトコル ラベル スイッチング（MPLS）を設定する必要があります。**show ip vrf** コマンドを使用して、VRF 用に設定されているルート識別子（RD）とインターフェイスを確認します。

手順

show ip vrf
一連の定義済み VRF インスタンスおよび関連付けられているインターフェイスを表示します。また、この出力では、VRF インスタンスが設定済みルート識別子にマップされます。

MPLS バーチャル プライベート ネットワーク サイト間の接続の確認

ローカルおよびリモートのカスタマー エッジ（CE）デバイスがマルチプロトコル ラベル スイッチング（MPLS）コアを介して通信できることを確認するには、次の作業を実行します。

MPLS コアを介した CE デバイスから CE デバイスへの IP 接続の確認

手順

ステップ 1 **enable**
特権 EXEC モードをイネーブルにします。

ステップ 2 **ping** [*protocol*] {*host-name* | *system-address*}
AppleTalk、コネクションレス型モード ネットワーク サービス（CLNS）、IP、Novell、Apollo、Virtual Integrated Network Service（VINES）、DECnet、または Xerox Network Service（XNS）ネットワークでの基本的なネットワーク接続を診断します。**ping** コマンドを使用して、ある CE デバイスから別の CE デバイスへの接続を確認します。

ステップ 3 **trace** [*protocol*] [*destination*]
パケットがその宛先に送信されるときに取るルートを検出します。**trace** コマンドは、2 つのデバイスが通信できない場合に問題のある箇所を分離するのに役立ちます。

ステップ 4 **show ip route** [*ip-address* [*mask*] [**longer-prefixes**]] | *protocol* [*process-id*]] | [**list** [*access-list-name* | *access-list-number*]
ルーティング テーブルの現在の状態を表示します。*ip-address* 引数を使用して、CE1 に CE2 へのルートが含まれていることを確認します。CE1 から学習したルートを確認します。CE2 へのルートがリストされていることを確認します。

ローカル CE デバイスとリモート CE デバイスが PE ルーティング テーブルに存在することの確認

手順

ステップ 1 **enable**
特権 EXEC モードをイネーブルにします。

ステップ 2 **show ip route vrf** *vrf-name* [*prefix*]
Virtual Routing and Forwarding（VRF）インスタンスに関連する IP ルーティング テーブルを表示します。ローカル カスタマー エッジ（CE）デバイスとリモート カスタマー エッジ（CE）デバイスのループバック アドレスが、プロバイダー エッジ（PE）でデバイスのルーティング テーブルに存在することを確認します。

ステップ 3 **show ip cef vrf** *vrf-name* [*ip-prefix*]
VRF に関連付けられているシスコ エクスプレス フォワーディング テーブルを表示します。次のように、リモート CE デバイスのプレフィックスが、シスコ エクスプレス フォワーディング テーブルに存在することを確認します。

MPLS バーチャル プライベート ネットワーク（VPN）の設定例

例：RIP を使用した MPLS バーチャル プライベート ネットワークの設定

PE の設定	CE の設定
ip vrf vpn1 rd 100:1 route-target export 100:1 route-target import 100:1 ! ip cef mpls ldp router-id Loopback0 force mpls label protocol ldp ! interface Loopback0 ip address 10.0.0.1 255.255.255.255 ! interface ip vrf forwarding vpn1 ip address 192.0.2.3 255.255.255.0 no cdp enable interface ip address 192.0.2.2 255.255.255.0 mpls label protocol ldp mpls ip ! router rip version 2 timers basic 30 60 60 120 ! address-family ipv4 vrf vpn1 version 2 redistribute bgp 100 metric transparent network 192.0.2.0 distribute-list 20 in no auto-summary exit-address-family ! router bgp 100 no synchronization bgp log-neighbor changes neighbor 10.0.0.3 remote-as 100 neighbor 10.0.0.3 update-source Loopback0 no auto-summary ! address-family vpnv4 neighbor 10.0.0.3 activate neighbor 10.0.0.3 send-community extended bgp scan-time import 5 exit-address-family ! address-family ipv4 vrf vpn1 redistribute connected redistribute rip no auto-summary no synchronization exit-address-family	ip cef mpls ldp router-id Loopback0 force mpls label protocol ldp ! interface Loopback0 ip address 10.0.0.9 255.255.255.255 ! interface ip address 192.0.2.1 255.255.255.0 no cdp enable router rip version 2 timers basic 30 60 60 120 redistribute connected network 10.0.0.0 network 192.0.2.0 no auto-summary

例：スタティック ルートを使用した MPLS バーチャル プライベート ネットワークの設定

PE の設定	CE の設定
ip vrf vpn1 rd 100:1 route-target export 100:1 route-target import 100:1 ! ip cef mpls ldp router-id Loopback0 force mpls label protocol ldp ! interface Loopback0 ip address 10.0.0.1 255.255.255.255 ! interface ip vrf forwarding vpn1 ip address 192.0.2.3 255.255.255.0 no cdp enable ! interface ip address 192.168.0.1 255.255.0.0 mpls label protocol ldp mpls ip ! router ospf 100 network 10.0.0. 0.0.0.0 area 100 network 192.168.0.0 255.255.0.0 area 100 ! router bgp 100 no synchronization bgp log-neighbor changes neighbor 10.0.0.3 remote-as 100 neighbor 10.0.0.3 update-source Loopback0 no auto-summary ! address-family vpnv4 neighbor 10.0.0.3 activate neighbor 10.0.0.3 send-community extended bgp scan-time import 5 exit-address-family ! address-family ipv4 vrf vpn1 redistribute connected redistribute static no auto-summary no synchronization exit-address-family ! ip route vrf vpn1 10.0.0.9 255.255.255.255 192.0.2.2 ip route vrf vpn1 192.0.2.0 255.255.0.0 192.0.2.2	ip cef ! interface Loopback0 ip address 10.0.0.9 255.255.255.255 ! interface ip address 192.0.2.2 255.255.0.0 no cdp enable ! ip route 10.0.0.9 255.255.255.255 192.0.2.3 3 ip route 198.51.100.0 255.255.255.0 192.0.2.3 3

その他の参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
この章で使用するコマンドの完全な構文および使用方法の詳細。	*Command Reference (Catalyst 9300 Series Switches)* の「MPLS コマンド」の項を参照してください。
Cisco Express Forwarding の設定	『*Cisco Express Forwarding Configuration Guide*』の「Configuring Basic Cisco Express Forwarding」モジュール
LDP の設定	『*MPLS Label Distribution Protocol Configuration Guide*』の「MPLS Label Distribution Protocol (LDP)」モジュール

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

MPLS バーチャル プライベート ネットワークの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *13* : *MPLS* バーチャル プライベート ネットワークの機能情報

リリース	変更箇所
Cisco IOS XE Everest 16.5.1a	この機能が導入されました。

第 8 章

MPLS QoS：EXP の分類およびマーキング

MPLS EXP の分類とマーキング

QoS EXP Matching 機能を使用すれば、マルチプロトコル ラベル スイッチング（MPLS）Experimental ビット（EXP ビット）フィールドを変更することで、ネットワーク トラフィックを分類し、マーキングすることができます。このモジュールでは、MPLS EXP フィールドを使用してネットワーク トラフィックを分類してマーキングするための概念情報と設定作業について説明します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

MPLS EXP の分類とマーキングの前提条件

- スイッチは MPLS プロバイダー エッジ（PE）またはプロバイダー（P）ルータとして設定する必要があります。この設定には、有効なラベル プロトコルと基礎となる IP ルーティング プロトコルの設定を含めることができます。

MPLS EXP の分類とマーキングの制約事項

- MPLS の分類とマーキングは、運用可能な MPLS ネットワーク内でのみ実行できます。
- MPLS EXP 分類とマーキングは、MPLS がイネーブルになっているインターフェイスか、またはその他のインターフェイス上の MPLS トラフィックでのみサポートされます。
- パケットが入力で IP タイプ オブ サービス（ToS）またはサービス クラス（CoS）によって分類された場合は、出力で MPLS EXP によって再分類できません（インポジション ケース）。ただし、パケットが入力で MPLS によって分類された場合は、出力で IP ToS、CoS、または Quality of Service（QoS）グループによって再分類できます（ディスポジション ケース）。
- プロトコルの境界を越えてトラフィックに QoS を適用するには、QoS グループを使用します。入力トラフィックを分類し、QoS グループに割り当てることができます。その後に、出力で QoS グループを分類し、QoS を適用することができます。
- パケットが MPLS でカプセル化されている場合は、IP などの他のプロトコルの MPLS ペイロードをチェックして分類またはマーキングすることはできません。MPLS EXP マーキングのみが MPLS によってカプセル化されたパケットに影響します。

MPLS EXP の分類とマーキングに関する情報

MPLS EXP の分類とマーキングの概要

QoS EXP Matching 機能を使用すれば、MPLS パケットの MPLS EXP フィールドに値を設定することによってネットワーク トラフィックを整理できます。MPLS EXP フィールドで異なった値を選択することにより、輻輳時にパケットが必要なプライオリティを持つようパケットをマーキングすることができます。MPLS EXP 値の設定によって次のことが可能になります。

- トラフィックの分類
  分類プロセスでマーキングするトラフィックが選択されます。分類は、トラフィックを複数の優先順位レベル、つまり、サービス クラスに分割することによりこのプロセスを実施します。トラフィック分類は、クラス ベースの QoS プロビジョニングのプライマリ コンポーネントです。詳細については、『Classifying Network Traffic』モジュールを参照してください。
- トラフィックのポリシングとマーキング
  ポリシングでは、設定されたレートを上回るトラフィックが廃棄されるか、別のドロップ レベルにマーキングされます。トラフィックのマーキングは、パケット フローを特定してそれらを区別する方法です。パケット マーキングを利用すれば、ネットワークを複数の優先プライオリティ レベルまたはサービス クラスに分割することができます。詳細については、『Marking Network Traffic』モジュールを参照してください。

MPLS 実験フィールド

MPLS Experimental ビット（EXP）フィールドは、ノードからパケットに付加される QoS 処理（Per-Hop Behavior）を定義するために使用可能な MPLS ヘッダー内の 3 ビット フィールドです。IP ネットワークでは、DiffServ コード ポイント（DSCP）（6 ビット フィールド）でクラスとドロップ優先順位が定義されます。EXP ビットは、IP DSCP でエンコードされた情報の一部を伝達するためにも、ドロップ優先順位をエンコードするためにも使用できます。

デフォルトで、Cisco IOS ソフトウェアは、IP パケットの DSCP または IP precedence の上位 3 ビットを MPLS ヘッダー内の EXP フィールドにコピーします。このアクションは、MPLS ヘッダーが初めて IP パケットに付加されたときに実行されます。ただし、DSCP または IP precedence と EXP ビットとの間のマッピングを定義することによって、EXP フィールドを設定することもできます。このマッピングは、**set mpls experimental** コマンドまたは **police** コマンドを使用して設定します。詳細については、「MPLS EXP の分類とマーキングの方法」を参照してください。

MPLS EXP マーキング操作を実行するには、テーブルマップを使用します。入力ポリシー内の別のトラフィック クラスに QoS グループを割り当て、テーブルマップを使用して QoS グループを出力ポリシー内の DSCP および EXP マーキングに変換することをお勧めします。

MPLS EXP の分類とマーキングのメリット

ネットワーク経由で伝送されるパケットの IP precedence フィールド値をサービス プロバイダーが変更したくない場合は、MPLS EXP フィールド値を使用して IP パケットを分類してマーキングできます。

MPLS EXP フィールド用の複数の値を選択することにより、ネットワーク輻輳が発生した場合に重大なパケットが優先されるようにそのようなパケットをマーキングすることができます。

MPLS EXP の分類とマーキングの方法

MPLS カプセル化パケットの分類

match mpls experimental topmost コマンドを使用すれば、MPLS ドメイン内のパケット EXP 値に基づくトラフィック クラスを定義できます。これらのクラスは、**police** コマンドを使用して EXP トラフィックをマーキングするサービス ポリシーを定義するために使用できます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**class-map** [**match-all**| **match-any**] *class-map-name* 例： `Switch(config)# class-map exp3`	トラフィックを指定したクラスにマッチングするために使用するクラスマップを作成し、クラス マップ コンフィギュレーション モードを開始します。 • クラス マップ名を入力します。
ステップ 4	**match mpls experimental topmost** *mpls-exp-value* 例： `Switch(config-cmap)# match mpls experimental topmost 3`	一致基準を指定します。 （注） **match mpls experimental topmost** コマンドは、最上位ラベル ヘッダー内の EXP 値に基づいてトラフィックを分類します。
ステップ 5	**end** 例： `Switch(config-cmap)# end`	（任意）特権 EXEC モードに戻ります。

最も外側のラベルでの MPLS EXP のマーキング

インポーズされたラベル エントリの MPLS EXP フィールドの値を設定するには、次の作業を実行します。

はじめる前に

通常の設定では、インポジションでの MPLS パケットのマーキングが IP ToS または CoS フィールドに基づく入力分類で使用されます。

（注） IP インポジション マーキングでは、デフォルトで、IP precedence 値が MPLS EXP 値にコピーされます。

（注） **set mpls experimental imposition** コマンドは、新しいまたは追加の MPLS ラベルが追加されたパケットに対してのみ機能します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map** *policy-map-name* 例： `Switch(config)# policy-map mark-up-exp-2`	作成されるポリシーマップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 • ポリシー マップ名を入力します。
ステップ 4	**class** *class-map-name* 例： `Switch(config-pmap)# class prec012`	トラフィックを指定したクラスにマッチングするために使用するクラスマップを作成し、クラス マップ コンフィギュレーション モードを開始します。 • クラス マップ名を入力します。
ステップ 5	**set mpls experimental imposition** *mpls-exp-value* 例： `Switch(config-pmap-c)# set mpls experimental imposition 2`	インポーズされたすべてのラベル エントリの MPLS EXP フィールドの値を設定します。
ステップ 6	**end** 例： `Switch(config-pmap-c)# end`	（任意）特権 EXEC モードに戻ります。

ラベル スイッチド パケットでの MPLS EXP のマーキング

ラベル スイッチドパケットでの MPLS EXP フィールドを設定するには、次の作業を実行します。

はじめる前に

（注） **set mpls experimental topmost** コマンドは、MPLS トラフィックの最も外側のラベルに EXP をマークします。入力ポリシーでのこのマーキングにより、出力ポリシーに MPLS EXP 値に基づく分類を含める必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map** *policy-map-name* 例： `Switch(config)# policy-map mark-up-exp-2`	作成されるポリシーマップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 • ポリシー マップ名を入力します。
ステップ 4	**class** *class-map-name* 例： `Switch(config-pmap)# class-map exp012`	トラフィックを指定したクラスにマッチングするために使用するクラスマップを作成し、クラス マップ コンフィギュレーション モードを開始します。 • クラス マップ名を入力します。
ステップ 5	**set mpls experimental topmost** *mpls-exp-value* 例： `Switch(config-pmap-c)# set mpls experimental topmost 2`	出力インターフェイスの最上位ラベルの MPLS EXP フィールド値を設定します。

	コマンドまたはアクション	目的
ステップ 6	**end** 例： `Switch(config-pmap-c)# end`	（任意）特権 EXEC モードに戻ります。

条件付きマーキングの設定

すべてのインポーズされたラベルに MPLS EXP フィールドの値を条件付きで設定するには、次の作業を実行します。

はじめる前に

（注） **set-mpls-exp-topmost-transmit** アクションは、MPLS カプセル化パケットにのみ影響します。**set-mpls-exp-imposition-transmit** アクションは、パケットに追加されたすべての新しいラベルに影響します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map** *policy-map-name* 例： `Switch(config)# policy-map ip2tag`	作成されるポリシーマップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 • ポリシー マップ名を入力します。

	コマンドまたはアクション	目的
ステップ 4	**class** *class-map-name* 例： `Switch(config-pmap)# class iptcp`	トラフィックと指定されたクラスを照合するために使用するクラスマップを作成し、ポリシーマップ クラス コンフィギュレーション モードを開始します。 • クラス マップ名を入力します。
ステップ 5	**police cir** *bps* **bc pir** *bps* **be** 例： `Switch(config-pmap-c)# police cir 1000000 pir 2000000`	分類するトラフィック用のポリサーを定義し、ポリシーマップ クラス ポリシング コンフィギュレーション モードを開始します。
ステップ 6	**conform-action transmit** 例： `Switch(config-pmap-c-police)# conform-action transmit 3`	ポリサーで指定された値に適合するパケットに対して実行するアクションを定義します。 • この例では、パケットが認定情報レート（cir）に適合する場合または適合バースト（bc）サイズ以内の場合に、MPLS EXP フィールドが 3 に設定されます。
ステップ 7	**exceed-action set-mpls-exp-topmost-transmit dscp table** *dscp-table-value* 例： `Switch(config-pmap-c-police)# exceed-action set-mpls-exp-topmost-transmit dscp table dscp2exp`	ポリサーで指定された値を上回るパケットに対して実行するアクションを定義します。
ステップ 8	**violate-actiondrop** 例： `Switch(config-pmap-c-police)# violate-action drop`	レートが最大情報レート（pir）を超えており、bc と be の範囲外のパケットに対して実行するアクションを定義します。 • 違反アクションを指定する前に、超過アクションを指定する必要があります。 • この例では、パケット レートが pir レートを超えており、bc と be の範囲外の場合に、パケットがドロップされます。

	コマンドまたはアクション	目的
ステップ 9	**end** 例： `Switch(config-pmap-c-police)# end`	（任意）特権 EXEC モードに戻ります。

MPLS EXP の分類とマーキングの設定例

例：MPLS カプセル化パケットの分類

MPLS EXP クラス マップの定義

次に、MPLS 実験値 3 を含むパケットと一致する exp3 という名前のクラス マップを定義する例を示します。

```
Switch(config)# class-map exp3
Switch(config-cmap)# match mpls experimental topmost 3
Switch(config-cmap)# exit
```

ポリシー マップの定義とポリシー マップの入力インターフェイスへの適用

次の例では、上の例でポリシーマップを定義するために作成したクラスマップを使用します。また、この例では、入力トラフィックの物理インターフェイスにポリシー マップを適用します。

```
Switch(config)# policy-map change-exp-3-to-2
Switch(config-pmap)# class exp3
Switch(config-pmap-c)# set mpls experimental topmost 2
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# service-policy input change-exp-3-to-2
Switch(config-if)# exit
```

ポリシー マップの定義とポリシー マップの出力インターフェイスへの適用

次の例では、上の例でポリシーマップを定義するために作成したクラスマップを使用します。また、この例では、出力トラフィックの物理インターフェイスにポリシー マップを適用します。

```
Switch(config)# policy-map WAN-out
Switch(config-pmap)# class exp3
Switch(config-pmap-c)# shape average 10000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# service-policy output WAN-out
Switch(config-if)# exit
```

最も外側のラベルでの MPLS EXP のマーキング

インポーズされたラベル エントリの MPLS EXP フィールドの値を設定するには、次の作業を実行します。

はじめる前に

通常の設定では、インポジションでの MPLS パケットのマーキングが IP ToS または CoS フィールドに基づく入力分類で使用されます。

（注） IP インポジション マーキングでは、デフォルトで、IP precedence 値が MPLS EXP 値にコピーされます。

（注） **set mpls experimental imposition** コマンドは、新しいまたは追加の MPLS ラベルが追加されたパケットに対してのみ機能します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map** *policy-map-name* 例： `Switch(config)# policy-map mark-up-exp-2`	作成されるポリシーマップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 • ポリシー マップ名を入力します。
ステップ 4	**class** *class-map-name* 例： `Switch(config-pmap)# class prec012`	トラフィックを指定したクラスにマッチングするために使用するクラスマップを作成し、クラス マップ コンフィギュレーション モードを開始します。 • クラス マップ名を入力します。

	コマンドまたはアクション	目的
ステップ 5	**set mpls experimental imposition** *mpls-exp-value* 例： `Switch(config-pmap-c)# set mpls experimental imposition 2`	インポーズされたすべてのラベル エントリの MPLS EXP フィールドの値を設定します。
ステップ 6	**end** 例： `Switch(config-pmap-c)# end`	（任意）特権 EXEC モードに戻ります。

例：ラベル スイッチド パケットの MPLS EXP のマーキング

MPLS EXP ラベル スイッチド パケット ポリシー マップの定義

次の例では、転送されたパケットの MPLS EXP 値に基づいて MPLS EXP 最上位値を 2 に設定するポリシー マップを定義します。

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# class-map exp012
Switch(config-cmap)# match mpls experimental topmost 0 1 2
Switch(config-cmap)# exit
Switch(config-cmap)# policy-map mark-up-exp-2
Switch(config-pmap)# class exp012
Switch(config-pmap-c)# set mpls experimental topmost 2
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

メイン インターフェイスへの MPLS EXP ラベル スイッチド パケット ポリシー マップの適用

次に、ポリシー マップのメイン インターフェイスへの適用例を示します。

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# service-policy input mark-up-exp-2
Switch(config-if)# exit
```

例：条件付きマーキングの設定

この例では、**ip2tag** ポリシーマップに含まれる **iptcp** クラス用のポリサーを作成し、そのポリシーマップをギガビット イーサネット インターフェイスに適用します。

```
Switch(config)# policy-map ip2tag
Switch(config-pmap)# class iptcp
Switch(config-pmap-c)# police cir 1000000 pir 2000000
Switch(config-pmap-c-police)# conform-action transmit
```

```
Switch(config-pmap-c-police)# exceed-action set-mpls-exp-imposition-transmit 2
Switch(config-pmap-c-police)# violate-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet 0/0/1
Switch(config-if)# service-policy input ip2tag
```

その他の参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
QoS コマンド	『*Cisco IOS Quality of Service Solutions Command Reference*』

標準および RFC

標準/RFC	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

QoS MPLS EXP の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *14*：*QoS MPLS EXP* の機能情報

リリース	変更箇所
Cisco IOS XE Everest 16.5.1a	この機能が導入されました。

第 9 章

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

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

VPLS の設定

VPLS について

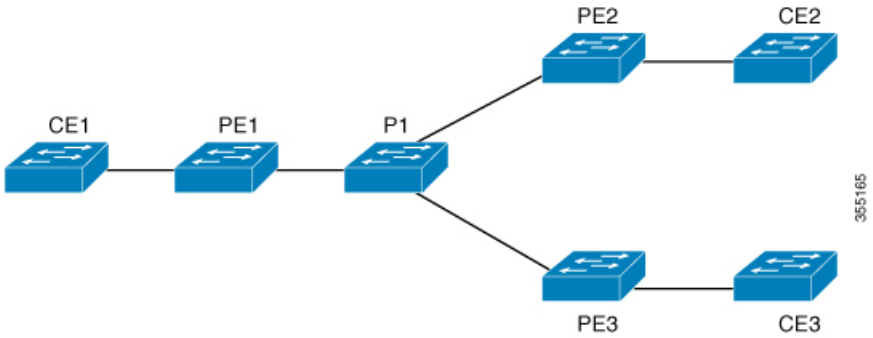
VPLS の概要

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 Services（VPLS）は、プロバイダー コアを使用して複数のアタッチメント回路を 1 つにまとめることで、複数のアタッチメント回路を 1 つに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 *7*：*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 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、最適パスが 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。

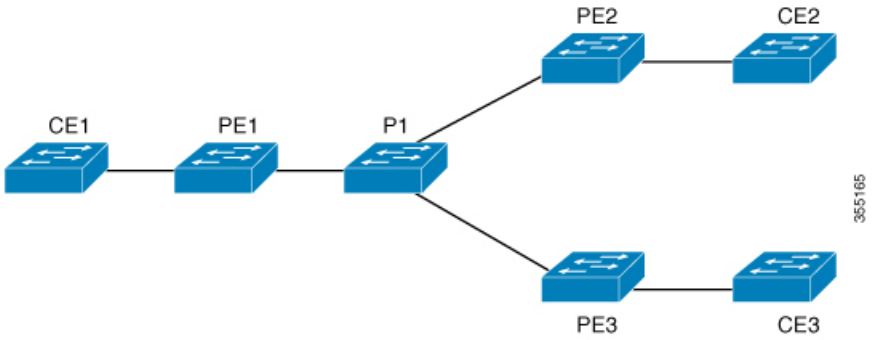
スケール番号

表 *15* : *VPLS* スケール

プラットフォーム	SDM に従ったスケール番号
3650	32 VFI、32 VLAN、VFI ごとに 8 ネイバー、256 VC/PW
3850	32 VFI、32 VLAN、VFI ごとに 8 ネイバー、256 VC/PW
9300	128 VFI、32 VLAN、VFI ごとに 32 ネイバー、1024 VC/PW
9500	128 VFI、32 VLAN、VFI ごとに 32 ネイバー、4096 VC/PW

VPLS の設定例

図 *8*：*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 2129manual 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 コマンドは、仮想回線に関する情報を提供します。

```
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 atm 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 の制約事項

- プロトコルベースの CLI 方式（インターフェイス疑似回線設定）はサポートされていません。VFI および Xconnect モードのみがサポートされています。
- Flow-Aware Transport 疑似回線（FAT PW）はサポートされていません。
- IGMP スヌーピングはサポートされていません。IGMP スヌーピングがディセーブルの場合にマルチキャスト トラフィックがフラッディングします。
- L2 プロトコル トンネリングはサポートされていません。
- Integrated Routing and Bridging（IRB）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。
- VPLS では疑似回線冗長性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、H-VPLS のスポークとしてのみサポートされています。
- MAC アドレスの取り消しはサポートされていません。
- L2 VPN インターワーキング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、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。
- Q-in-Q トラフィックはサポートされていません。
- 接続回線では、dot1q トンネルはサポートされていません。

CE への PE レイヤ 2 インターフェイスの設定

CE からタグ付きトラフィックを受け取る 802.1Q トランクの設定

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 モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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。

	コマンドまたはアクション	目的
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)# end`	特権 EXEC モードに戻ります。

CE からタグなしトラフィックを受け取る 802.1Q アクセス ポートの設定

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 モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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。

	コマンドまたはアクション	目的
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 mode access** 例： `Device(config-if)# switchport mode access`	インターフェイスを、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 タイプとして設定します。
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： `Device(config)# end`	特権 EXEC モードに戻ります。

PE でのレイヤ 2 VLAN インスタンスの設定

PE 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 ルータ上のレイヤ 2 VLAN インスタンスで、VPLS と VLAN 間のマッピングを設定できるようになります。

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モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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`	特定の仮想LAN（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)# end`	特権 EXEC モードに戻ります。

PE における MPLS の設定

PE に MPLS を設定するには、必須 MPLS パラメータを指定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権EXECモードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configureterminal** 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-vlan)# mpls label protocol ldp`	プラットフォームのデフォルト ラベル配布プロトコルを指定します。
ステップ 5	**mpls label protocol ldp** 例： `Device(config-vlan)# interface vlan 2129`	プラットフォームのデフォルト ラベル配布プロトコルを指定します。
ステップ 6	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**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。

PE における VFI の設定

仮想スイッチ インスタンス（VFI）は、VPLS ドメインの VPN ID、このドメインにある他の PE デバイスのアドレス、および各ピアのトンネル シグナリングのタイプとカプセル化のメカニズムを指定します（これは、VFI と関連付けられた VC を作成するピアです）。次のように VFI を設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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 VRF にバインドされたエミュレート VC では、シグナリングにこの VPN ID を使用します。 （注） *vpn-id* は *vlan-id* と同じです。
ステップ 5	**neighbor remote-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配線プロパティを指定します。
ステップ 6	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

PE での VFI への接続回線の関連付け

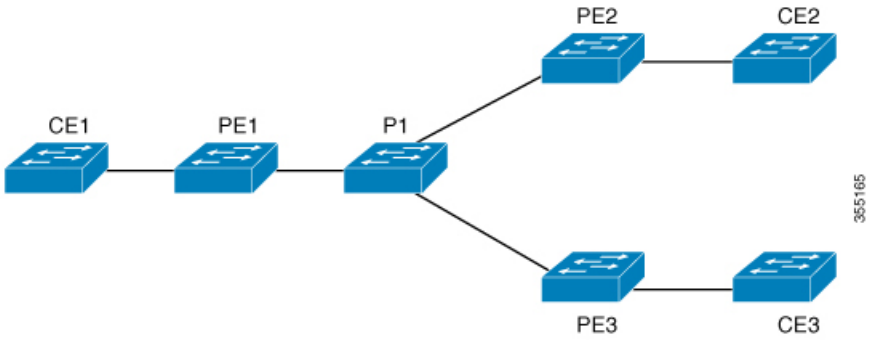
VFI を定義したら、1 つ以上の接続回線にバインドする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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-vlan)# 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します）。
ステップ 5	**xconnect vfi** *vfi-name* 例： `Device(config-vlan)# 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)# end`	特権 EXEC モードに戻ります。

VPLS の設定例

図 *9*：*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 2129manual 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 コマンドは、仮想回線に関する情報を提供します。

```
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 atm 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

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、最適パスが 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。

スケール番号

表 *16*：*BGP - AD* スケール

プラットフォーム	SDM に従ったスケール番号
3650	32 VFI、32 VLAN、VFI ごとに 8 ネイバー、256 VC/PW
3850	32 VFI、32 VLAN、VFI ごとに 8 ネイバー、256 VC/PW
9300	128 VFI、32 VLAN、VFI ごとに 32 ネイバー、1024 VC/PW
9500	128 VFI、32 VLAN、VFI ごとに 32 ネイバー、4096 VC/PW

VPLS BGP ベースの自動検出のイネーブル化

仮想プライベート LAN サービス（VPLS）PE デバイスで同じ VPLS ドメインに属している他の PE デバイスを検出できるようにするには、次のタスクを実行します。

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 モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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。

	コマンドまたはアクション	目的
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)# 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

Border Gateway Protocol（BGP）Layer 2 VPN（L2VPN）アドレス ファミリは、仮想プライベート LAN サービス（VPLS）自動検出用のエンドポイント プロビジョニング情報が含まれている個別の L2VPN ルーティング情報ベース（RIB）をサポートします。BGP は、レイヤ 2 仮想転送インスタンス（VFI）が設定されたときに毎回アップデートされる L2VPN データベースからのエンドポイント プロビジョニング情報を学習します。BGP がすべての BGP ネイバーにアップデート メッセージでエンドポイント プロビジョニング情報を配布すると、そのエンドポイント情報を使用して L2VPN ベースのサービスをサポートするように擬似回線メッシュが設定されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configureterminal** 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。

	コマンドまたはアクション	目的
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 ファミリをディセーブルにします。 （注） IPv4 ユニキャスト アドレス ファミリに関するルーティング情報は、neighbor remote-as ルータ コンフィギュレーション 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 ルータ コンフィギュレーション コマンドを設定した場合は例外です。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イネーブルにします。
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しない場合、ネイバーは外部ネイバーになります。

	コマンドまたはアクション	目的
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** *number* 例： `Device(config-router)# address-family l2vpn vpls`	L2VPN 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イネーブルにします。
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。	

	コマンドまたはアクション	目的
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。
ステップ 14	**end** 例： `Device(config-router-af)# 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 ?
```

第 10 章

MPLS VPN ルート ターゲット書き換えの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

MPLS VPN ルート ターゲット書き換えの前提条件

- マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）の設定方法を知っている必要があります。
- 自律システム（AS）向けに RT 置換ポリシーおよびターゲット デバイスを識別する必要があります。

MPLS VPN ルート ターゲット書き換えの制約事項

ルート ターゲットの書き換えは、単一 AS トポロジにのみ実装できます。

MPLS VPN ルート ターゲット書き換えに関する情報

ルート ターゲット置換ポリシー

ピアのルーティング ポリシーには、インバウンドまたはアウトバウンドのルーティング テーブル アップデートに影響する可能性のある設定がすべて含まれています。インバウンドおよびアウトバウンドの Border Gateway Protocol（BGP）アップデートに対してルート ターゲットの置換を有効にすると、MPLS VPN ルート ターゲット書き換え機能がルーティング テーブル アップデートに影響する可能性があります。BGP バーチャル プライベート ネットワーク IP バージョン 4（VPNv4）のアップデートでは、ルート ターゲットが拡張コミュニティ属性として送信されます。ルート ターゲット拡張コミュニティ属性を使用して、一連のサイト、および設定されたルート ターゲットを使用するルートを受信できる VPN ルーティングおよび転送（VRF）インスタンスが識別されます。

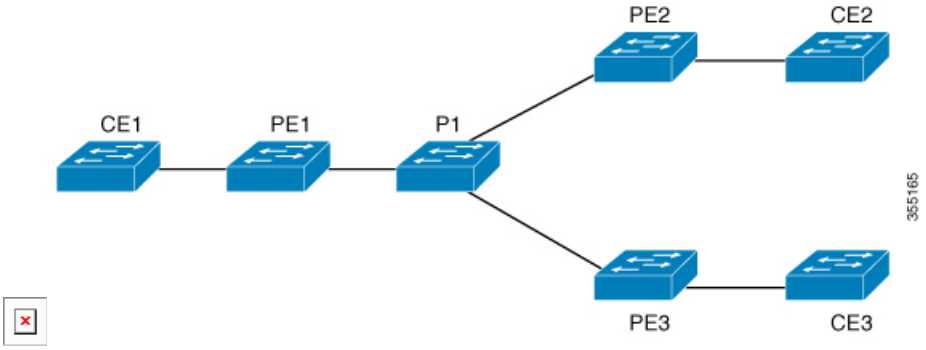
MPLS VPN ルート ターゲットの書き換え機能は、プロバイダー エッジ（PE）デバイスで設定できます。

次の図に、マルチプロトコル ラベル スイッチング（MPLS）VPN の単一自律システム トポロジ内の PE デバイスでルート ターゲットを置換する例を示します。この例には、次の設定が含まれています。

- PE1 は、VRF カスタマー A の RT 65000:1 をインポートおよびエクスポートして、RT 65000:1 のすべてのインバウンド VPNv4 プレフィックスを RT 65000:2 に書き換えるように設定されています。

- PE2 は、VRF カスタマー B の RT 65000:2 をインポートおよびエクスポートして、RT 65000:2 のすべてのインバウンド VPNv4 プレフィックスを RT 65000:1 に書き換えるように設定されています。

図 *10*：単一の *MPLS VPN* 自律システム トポロジのプロバイダー エッジ（*PE*）デバイスでのルート ターゲットの置換



ルート マップおよびルート ターゲットの置換

MPLS VPN ルート ターゲット書き換え機能によって Border Gateway Protocol（BGP）インバウンド/アウトバウンド ルート マップ機能が拡張され、ルート ターゲットの置換がイネーブルになります。ルート マップ コンフィギュレーション モードで入力した **set extcomm-list delete** コマンドを使用すると、拡張コミュニティ リストに基づいてルート ターゲット拡張コミュニティ属性を削除できます。

MPLS VPN ルート ターゲット書き換えの設定方法

ルート ターゲット置換ポリシーの設定

インターネットワークにルート ターゲット（RT）置換ポリシーを設定するには、次の作業を実行します。

RT *x* を RT *y* に書き換えるようにプロバイダー エッジ（PE）を設定したとき、その PE に RT *x* をインポートする仮想ルーティングおよび転送（VRF）インスタンスが設定されている場合は、RT *x* に加えて RT *y* をインポートする VRF も設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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	**ip extcommunity-list** {*standard-list-number* | *expanded-list-number*} {**permit** | **deny**} [*regular-expression*] [**rt** | **soo** *extended-community-value*] 例： `Device(config)# ip extcommunity-list 1 permit rt 65000:2`	拡張コミュニティ アクセス リストを作成し、リストへのアクセスを制御します。 • *standard-list-number* 引数は 1 ～ 99 の整数で、拡張コミュニティの 1 つまたは複数の許可グループまたは拒否グループを指定します。 • *expanded-list-number* 引数は 100 ～ 500 の整数で、拡張コミュニティの 1 つまたは複数の許可グループまたは拒否グループを指定します。拡張リストには正規表現を設定できますが、標準リストには設定できません。 • **permit** キーワードを指定すると、一致する条件へのアクセスを許可します。 • **deny** キーワードを指定すると、一致する条件へのアクセスを拒否します。 • *regular-expression* 引数には、マッチングを行う入力ストリング パターンを指定します。拡張された拡張コミュニティ リストを使用してルート ターゲットのマッチングを行う場合は、正規表現にパターン RT: を追加します。 • **rt** キーワードには、ルート ターゲット拡張コミュニティ属性を指定します。**rt** キーワードは標準拡張コミュニティ リストだけに設定できます。拡張された拡張コミュニティ リストには設定できません。 • **soo** キーワードには、Site of Origin（SOO）拡張コミュニティ属性を指定します。**soo** キーワードは標準拡張コミュニティ リストだけに設定でき

	コマンドまたはアクション	目的
		ます。拡張された拡張コミュニティ リストには設定できません。 • *extended-community-value* 引数には、ルート ターゲットまたは Site of Origin を指定します。この値には次の組み合わせのいずれかを指定できます。 • autonomous-system-number:network-number • ip-address:network-number 自律システム番号とネットワーク番号、または IP アドレスとネットワーク番号の区切りにはコロンを使用します。
ステップ 4	**route-map** *map-name* [**permit** | **deny**] [*sequence-number*] 例： `Device(config)# route-map rtrewrite permit 10`	ルーティング プロトコル間でルートを再配布する条件を定義するか、ポリシー ルーティングをイネーブルにしてルート マップ コンフィギュレーション モードを開始します。 • *map-name* 引数では、ルート マップに意味のある名前を定義します。**redistribute** ルータ コンフィギュレーション コマンドはこの名前を使用して、このルート マップを参照します。複数のルート マップで同じマップ名を共有できます。 • このルート マップの一致基準が満たされた場合、**permit** キーワードが指定されていると、設定アクションに従ってルートが再配布されます。ポリシー ルーティングの場合、パケットはポリシーに従ってルーティングされます。 一致基準が満たされなかった場合、**permit** キーワードが指定されていると、同じマップ タグを持つ次のルート マップがテストされます。あるルートが、同じ名前を共有するルート マップ セットの一致基準のいずれをも満たさない場合、そのセットによる再配布は行われません。 **permit** キーワードがデフォルトです。 • ルート マップの一致基準が満たされた場合でも、**deny** キーワードが指定されているとルートは再配布されません。ポリシー ルーティングの場合、パケットはポリシーに従ってルーティングされません。また、同じマップ タグ名を共有

	コマンドまたはアクション	目的
		するルート マップは、これ以上検証されません。パケットがポリシー ルーティングの対象にならない場合、通常の転送アルゴリズムが使用されます。 • *sequence-number* 引数は、同じ名前で設定済みのルートマップのリストに新しいルートマップが入る位置を示す番号です。このコマンドの **no** 形式を指定すると、このルート マップの位置が削除されます。

ステップ 5	**match extcommunity** {*standard-list-number* | *expanded-list-number*}
	例：
	`Device(config-route-map)# match extcommunity 1`
	例：
	`Device(config-route-map)# match extcommunity 101`
		Border Gateway Protocol（BGP）拡張コミュニティ リスト属性とマッチングします。
		• *standard-list-number* 引数は 1 ～ 99 の番号で、拡張コミュニティ属性の 1 つまたは複数の許可グループまたは拒否グループを指定します。
		• *expanded-list-number* 引数は 100 ～ 500 の番号で、拡張コミュニティ属性の 1 つまたは複数の許可グループまたは拒否グループを指定します。

ステップ 6	**set extcomm-list** *extended-community-list-number***delete** 例： `Device(config-route-map)# set extcomm-list 1 delete`	インバウンドまたはアウトバウンド BGP バーチャル プライベート ネットワーク バージョン 4（VPNv4）アップデートの拡張コミュニティ属性からルート ターゲットを削除します。 • *extended-community-list-number* 引数には、拡張コミュニティ リスト番号を指定します。
ステップ 7	**set extcommunity** {**rt** *extended-community-value* [**additive**] | **soo** *extended-community-value*} 例： `Device(config-route-map)# set extcommunity rt 65000:1 additive`	BGP 拡張コミュニティ属性を設定します。 • **rt** キーワードには、ルート ターゲット拡張コミュニティ属性を指定します。 • **soo** キーワードには、Site of Origin 拡張コミュニティ属性を指定します。 • *extended-community-value* 引数には、設定値を指定します。この値には次の組み合わせのいずれかを指定できます。 • autonomous-system-number:network-number • ip-address:network-number

	コマンドまたはアクション	目的
		自律システム番号とネットワーク番号、または IP アドレスとネットワーク番号の区切りにはコロンを使用します。 • **additive**キーワードを指定すると、既存のルートターゲットを置換することなく、既存のルートターゲットリストにルートターゲットが追加されます。
ステップ 8	**end** 例： `Device(config-route-map)# end`	（任意）特権 EXEC モードに戻ります。
ステップ 9	**show route-map** *map-name* 例： `Device# show route-map extmap`	（任意）マッチングと設定されたエントリが正しいことを確認します。 • *map-name* 引数には、特定のルート マップの名前を指定します。

ルート ターゲット置換ポリシーの適用

ネットワークにルート ターゲット置換ポリシーを適用するには、次の作業を実行します。

特定の BGP ネイバーへのルート マップの割り当て

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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。

	コマンドまたはアクション	目的
ステップ 3	**router bgp** *as-number* 例： `Device(config)# router bgp 100`	Border Gateway Protocol（BGP）ルーティング プロセスを設定し、デバイスでルータ コンフィギュレーション モードを開始します。 • *as-number* 引数は、デバイスを他の BGP デバイスに対して識別し、転送するルーティング情報にタグを設定する自律システムの番号を示します。 指定できる範囲は 0 ～ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ～ 65535 です。
ステップ 4	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *as-number* 例： `Device(config-router)# neighbor 172.10.0.2 remote-as 200`	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。 • *ip-address* 引数には、ネイバーの IP アドレスを指定します。 • *peer-group-name* 引数には、BGP ピア グループの名前を指定します。 • *as-number* 引数には、ネイバーが属している自律システムを指定します。
ステップ 5	**address-family vpnv4** [**unicast**] 例： `Device(config-router)# address-family vpnv4`	アドレス ファミリ コンフィギュレーション モードを開始して、標準バーチャル プライベート ネットワーク バージョン 4（VPNv4）アドレス プレフィックスを使用する、BGP などのルーティング セッションを設定します。 • **unicast** キーワード（任意）では、VPNv4 ユニキャスト アドレス プレフィックスを指定します。
ステップ 6	**neighbor** {*ip-address* | *peer-group-name*} **activate** 例： `Device(config-router-af)# neighbor 172.16.0.2 activate`	ネイバー BGP デバイスとの情報交換を有効にします。 • *ip-address* 引数には、ネイバーの IP アドレスを指定します。 • *peer-group-name* 引数には、BGP ピア グループの名前を指定します。
ステップ 7	**neighbor** {*ip-address* | *peer-group-name*} **send-community** [**both** | **extended** | **standard**]	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。 • *ip-address* 引数には、BGP 対応ネイバーの IP アドレスを指定します。

	コマンドまたはアクション	目的
	例： `Device(config-router-af)# neighbor 172.16.0.2 send-community extended`	• *peer-group-name* 引数には、BGP ピア グループの名前を指定します。 • **both** キーワードを指定すると、標準および拡張コミュニティ属性が送信されます。 • **extended** キーワードを指定すると、拡張コミュニティ属性が送信されます。 • **standard** キーワードを指定すると、標準コミュニティ属性が送信されます。
ステップ 8	**neighbor** {*ip-address* | *peer-group-name*} **route-map** *map-name* {**in** | **out**} 例： `Device(config-router-af)# neighbor 172.16.0.2 route-map extmap in`	着信ルートまたは発信ルートにルート マップを適用します。 • *ip-address* 引数には、ネイバーの IP アドレスを指定します。 • *peer-group-name* 引数には、BGP ピア グループまたはマルチプロトコル ピア グループの名前を指定します。 • *map-name* 引数には、ルート マップの名前を指定します。 • **in** キーワードを指定すると、受信ルートにルート マップが適用されます。 • **out** キーワードを指定すると、発信ルートにルート マップが適用されます。
ステップ 9	**end** 例： `Device(config-router-af)# end`	（任意）特権 EXEC モードに戻ります。

ルート ターゲット置換ポリシーの確認

手順

ステップ 1 **enable**
特権EXECモードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。

例：

```
Device> enable
Device#
```

ステップ 2 **show ip bgp vpnv4 vrf** *vrf-name*

指定したルート ターゲット（RT）拡張コミュニティ属性を持つバーチャル プライベート ネットワーク バージョン4（VPNv4）が適切なRT拡張コミュニティ属性で置換されることを確認して、プロバイダー エッジ（PE）デバイスが書き換えられた RT 拡張コミュニティ属性を受け取ることを確認します。

PE1 でルート ターゲットの置換を確認するには、次のコマンドを入力します。

例：

```
Device# show ip bgp vpnv4 vrf Customer_A 192.168.1.1/32 internal
BGP routing table entry for 65000:1:192.168.1.1/32, version 6901
Paths: (1 available, best #1, table Customer_A)
  Advertised to update-groups:
     5
  Refresh Epoch 1
  650002
    3.3.3.3 (metric 3) (via default) from 3.3.3.3 (55.5.4.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:65000:1
      mpls labels in/out nolabel/3025
      rx pathid: 0, tx pathid: 0x0
      net: 0xFFB0A72E38, path: 0xFFB0E6A370, pathext: 0xFFB0E5D970
      flags: net: 0x0, path: 0x7, pathext: 0x181
```

ステップ 3 **exit**

ユーザ EXEC モードに戻ります。

例：

```
Device# exit
Device>
```

MPLS VPN ルート ターゲット書き換えの設定例

例：ルート ターゲット置換ポリシーの設定

次に、バーチャル プライベート ネットワーク バージョン4（VPNv4）プレフィックスを別のプロバイダーエッジ（PE）デバイスと交換するプロバイダーエッジ（PE）デバイスのルート ターゲット（RT）置換の設定例を示します。インバウンド アップデートで RT を置換するようにルート

マップ extmap が設定されています。RT 65000:2 を持つ着信アップデートはすべて RT 65000:1 に置換されます。

```
!
ip extcommunity-list 1 permit rt 65000:2
!
route-map rtrewrite permit 10
match extcommunity 1
set extcomm-list 1 delete
set extcommunity rt 65000:1 additive
!
```

次に、アップデートに複数の置換ルールを適用する必要がある場合に、ルート マップ コンフィギュレーションの **continue** コマンドを使用する例を示します。次の例では、着信アップデートの 7777:222222222 が RT 65000:2 に置換されます。**continue 20** コマンドを指定しない場合、シーケンス 10 でマッチングが行われるとルート マップの評価は停止します。 **continue 20** コマンドを指定した場合、シーケンス 10 で一致した場合でもルート マップの評価はシーケンス 20 まで継続します。

```
!
ip extcommunity-list 2 permit rt 7777:222222222
ip extcommunity-list 3 permit rt 2:2
ip extcommunity-list 4 permit rt 20000:111
!
route-map extmap1 permit 10
match extcommunity 2
continue 20
set extcomm-list 2 delete
set extcommunity rt 65000:2 additive
!
route-map extmap1 permit 20
match extcommunity 3
continue 30
set extcomm-list 3 delete
!
route-map extmap1 permit 30
match extcommunity 4
set extcomm-list 4 delete
!
```

（注） アウトバウンド ルート マップは、ルート マップ コンフィギュレーションの **continue** コマンドをサポートしていません。

例：ルート ターゲット置換ポリシーの適用

例：特定の BGP ネイバーへのルート マップの割り当て

次に、Border Gateway Protocol（BGP）ネイバーにルート マップ extmap を関連付ける例を示します。BGP インバウンド ルート マップは、着信アップデートのルート ターゲット（RT）を置換するように設定されています。

```
router bgp 1
address-family vpnv4
neighbor 2.2.2.2 route-map rtrewrite in
```

次に、アウトバウンドBGPネイバーに同じルートマップを関連付ける例を示します。このルートマップは、発信アップデートの RT を置換するように設定されています。

```
router bgp 1
address-family vpnv4
neighbor 2.2.2.2 route-map rtrewrite out
```

第 11 章

マルチキャスト バーチャル プライベート ネットワークの設定

マルチキャスト VPN の設定

マルチキャスト VPN（MVPN）機能は、レイヤ 3 VPN 上でマルチキャストをサポートできるようにします。企業がマルチキャストアプリケーションの範囲を拡大するにつれて、サービスプロバイダーは、マルチプロトコル ラベル スイッチング（MPLS）コア ネットワークを通じてそれらに対応できます。IP マルチキャストは、ビデオ、音声、およびデータを MPLS VPN ネットワーク コア経由でストリーミングするために使用します。

従来、ポイントツーポイント トンネルはサービス プロバイダー ネットワークに接続する唯一の方法でした。このようなトンネル ネットワークは、スケーラビリティの問題が発生する傾向がありますが、IP マルチキャスト トラフィックを VPN に通過させる唯一の方法でした。

レイヤ 3 VPN はユニキャスト トラフィック接続のみをサポートするため、レイヤ 3 VPN を併用して MPLS を導入することによって、サービス プロバイダーは、レイヤ 3 VPN のカスタマーにユニキャスト接続とマルチキャスト接続の両方を提供できます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

マルチキャスト VPN の設定に関する前提条件

「Configuring Basic IP Multicast」モジュールに記載されているタスクを使用して、IP マルチキャストを有効にして PIM インターフェイスを設定します。

マルチキャスト VPN の設定の制限

- ボーダー ゲートウェイ プロトコル（BGP）ピアリングのアップデート ソース インターフェイスは、デフォルト マルチキャスト配信ツリー（MDT）を適切に設定するために、デバイス上に設定されたすべての BGP ピアリングで同じにする必要があります。BGP ピアリングにループバック アドレスを使用する場合は、ループバック アドレスで PIM スパース モードをイネーブルにする必要があります。
- MVPN では、複数の BGP ピアリング更新送信元をサポートしていません。
- 複数の BGP 更新送信元はサポートされていません。これらを設定すると、リバース パス フォワーディング（RPF）のチェックが中断される可能性があります。MVPN トンネルの送信元 IP アドレスは、BGP ピアリング更新送信元に使用される最高の IP アドレスによって決まります。この IP アドレスが、リモートのプロバイダー エッジ（PE）デバイスを含む BGP ピアリング アドレスとして使用される IP アドレスでない場合、MVPN は適切に機能しません。

マルチキャスト VPN の設定について

マルチキャスト VPN の操作

MVPN IP を使用すると、サービス プロバイダーは MPLS VPN 環境でマルチキャスト トラフィックを設定およびサポートできます。この機能は、個々の VRF インスタンスでのマルチキャスト パケットのルーティングおよび転送をサポートし、サービス プロバイダーのバックボーンに VPN マルチキャスト パケットを転送するメカニズムも提供します。

VPN は、ISP などの共有インフラストラクチャを介するネットワークの接続性です。その役割は、プライベート ネットワークとして、同じポリシーとパフォーマンスを低い所有コストで提供することによって、業務とインフラストラクチャを通して、多くのコスト削減の機会を作り出すことです。

MVPN により、企業はサービス プロバイダーのネットワーク バックボーンでプライベート ネットワークをトランスペアレントに相互接続することができます。このように MVPN を使用して企業ネットワークを相互接続しても、企業ネットワークの管理方法や、企業の全体的な接続性は変更されません。

マルチキャスト VPN の利点

- 複数の場所に情報を動的に送信するスケーラブルなメソッドを提供します。

- 高速な情報伝送を提供します。
- 共有インフラストラクチャを介して接続性を提供します。

マルチキャスト VPN ルーティングおよび転送とマルチキャスト ドメイン

MVPNは、VPNルーティングおよび転送テーブルにマルチキャストルーティング情報を導入します。プロバイダー エッジ（PE）デバイスがマルチキャスト データまたは制御パケットをカスタマー エッジ（CE）ルータから受信すると、マルチキャスト VPN ルーティングおよび転送インスタンス（MVRF）の情報に従って転送が実行されます。MVPN は、ラベル スイッチングを使用しません。

マルチキャスト トラフィックを相互に送信できる MVRF のセットは、マルチキャスト ドメインの構成要素です。たとえば、特定タイプのマルチキャスト トラフィックをすべてのグローバルな従業員に送信するカスタマーのマルチキャスト ドメインは、そのエンタープライズと関連するすべての CE ルータから構成されます。

マルチキャスト配信ツリー

MVPN は、各マルチキャスト ドメインにスタティック デフォルト マルチキャスト配信ツリー（MDT）を確立します。デフォルト MDT は、PE ルータが使用するパスを定義し、マルチキャスト ドメインにある他のすべての PE ルータに、マルチキャスト データとコントロール メッセージを送信します。

Source Specific Multicast（SSM; 送信元特定マルチキャスト）がコア マルチキャスト ルーティング プロトコルとして使用される場合、デフォルト MDT およびデータ MDT に使用されるマルチキャスト IP アドレスは、すべての PE ルータの SSM 範囲内に設定する必要があります。

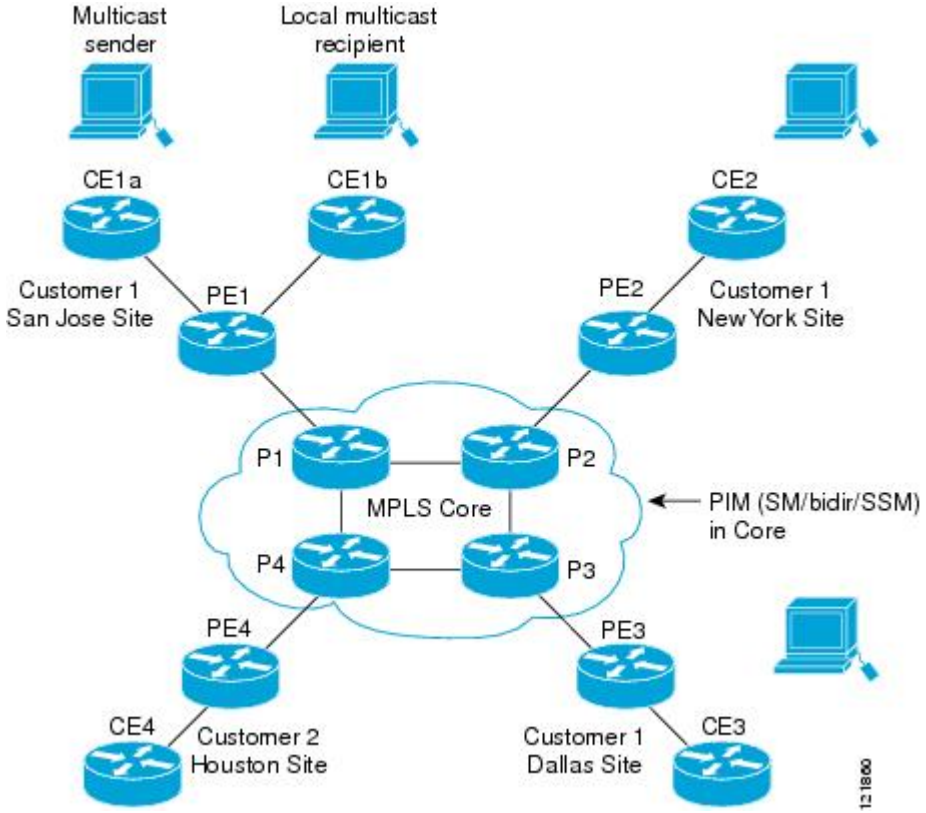
また、MVPN は、高帯域幅伝送用の MDT のダイナミックな作成もサポートします。データ MDT は、Cisco IOS ソフトウェアに一意な機能です。データ MDT は、VPN 内のフルモーション ビデオなどの高帯域幅の送信元向けであり、MPLS VPN コアの最適なトラフィック転送を確保することを目的としています。データ MDT が作成されるしきい値は、ルータ単位または VRF 単位で設定できます。マルチキャスト伝送が定義されたしきい値を超えると、送信側の PE ルータがデータ MDT を作成し、データ MDT に関する情報を含む UDP メッセージをデフォルト MDT のすべてのルータに送信します。マルチキャスト ストリームがデータ MDT のしきい値を超えたかどうかを判断する統計情報は、1 秒に 1 回確認されます。PE ルータは UDP メッセージを送信した後、切り替わるまでに 3 秒以上待機します。最も長くかかる場合は 13 秒、最良の場合は 3 秒です。

データ MDT は、VRF マルチキャスト ルーティング テーブル内で、（S,G）マルチキャスト ルート エントリ専用に作成されます。個々のソースデータ レートの値に関係なく、(*, G) エントリ用には作成されません。

次の例のサービス プロバイダーには、San Jose、New York、Dallas にオフィスがあるマルチキャスト カスタマーがいます。San Jose では、一方向のマルチキャスト プレゼンテーションが行われています。サービス プロバイダー ネットワークでは、このカスタマーと関連する 3 つすべてのサイト、および別のエンタープライズ カスタマーの Houston サイトがサポートされます。

エンタープライズ カスタマーのデフォルト MDT は、プロバイダーのルータ P1、P2、P3、およびその関連 PE ルータから構成されています。PE4 は別のカスタマーに関連付けられているため、デフォルト MDT の一部ではありません。次の図からは、San Jose 以外はマルチキャストに加入していないため、データがデフォルト MDT に沿って転送されていないことがわかります。

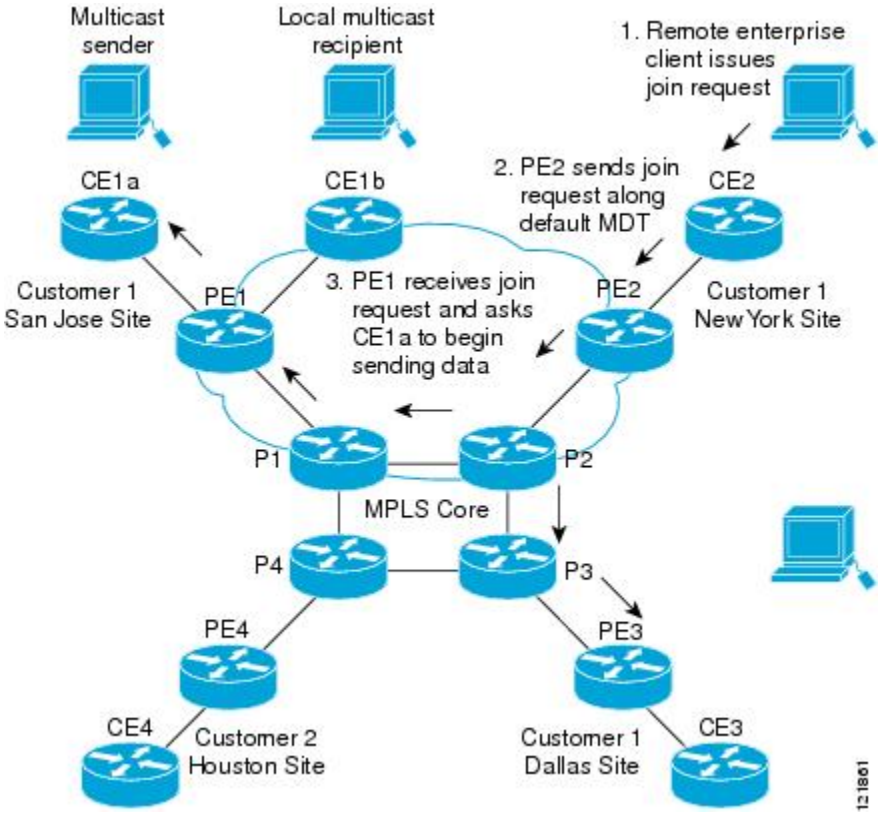
図 *11*：デフォルト マルチキャスト配信ツリーの概要



New York の従業員がマルチキャスト セッションに加入します。New York のサイトに関連付けられている PE ルータは、カスタマーのマルチキャスト ドメインのデフォルト MDT を介して転送される加入要求を送信します。PE1 は、マルチキャスト セッションの送信元に関連付けられている

PEルータであり、この要求を受信します。次の図は、PEルータが、マルチキャスト送信元（CE1a）と関連する CE ルータに要求を転送する方法を示しています。

図 *12*：データ *MDT* の初期化



CE ルータ（CE1a）が関連する PE ルータ（PE1）へマルチキャスト データの送信を開始すると、PE ルータ（PE1）は、デフォルト MDT に沿ってマルチキャスト データを送信します。PE1 は、マルチキャスト データを送信すると、マルチキャスト データがデータ MDT を作成する対象の帯域幅のしきい値を超えていることを認識します。したがって、PE1 はデータ MDT を作成し、データ MDT に関する情報を含むデフォルト MDT を使用して、すべてのルータにメッセージを送信し、3 秒後、データ MDT を使用して、その特定のストリームのマルチキャスト データを送信し始めます。このソースに関係する受信先は PE2 だけにあるので、PE2 だけがデータ MDT に加入し、データ MDT でトラフィックを受信します。

PE ルータは、デフォルト MDT を介して他の PE ルータと PIM 関係を維持するとともに、直接接続された PE ルータとの PIM 関係をも維持します。

マルチキャスト トンネル インターフェイス

マルチキャスト ドメインごとに作成される MVRF では、デバイスは、すべての MVRF トラフィックが発信されるトンネルインターフェイスを作成する必要があります。マルチキャスト トンネル インターフェイスは、MVRF がマルチキャスト ドメインにアクセスするために使用するインター

フェイスです。これはMVRFとグローバルMVRFをつなぐコンジットと見なすことができます。MVRF ごとに 1 つのトンネル インターフェイスが作成されます。

マルチキャスト VPN での BGP の MDT アドレス ファミリ

MDT アドレス ファミリ セッションを設定するために、**mdt** キーワードが **address-family ipv4** コマンドに追加されました。MDTアドレス ファミリ セッションは、Border Gateway Protocol（BGP）MDT Subaddress Family Identifier（SAFI）のアップデートを使用して PIM に送信元 PE アドレスと MDT グループ アドレスを渡すために使用されます。

マルチキャスト VPN サポートの BGP アドバタイズメント方式

1 つの自律システムで、MVPN のデフォルト MDT がランデブー ポイント（RP）のあるスパース モード（PIM-SM）を使用している場合、ソース PE とレシーバ PE は RP を通して互いを検出するため、PIM は、マルチキャスト トンネル インターフェイス（MTI）に隣接を確立できます。このシナリオでは、ローカル PE（送信元 PE）が RP に登録メッセージを送信し、次に RP が送信元 PE に向けて最短パスツリーを構築します。次にリモート PE（MDT マルチキャスト グループの受信者として機能します）が RP に向けて (*, G) 加入メッセージを送信し、そのグループの配信ツリーに参加します。

しかし、デフォルト MDT グループが PIM-SM 環境ではなく PIM Source Specific Multicast（PIM-SSM）環境で設定されている場合、受信側 PE は送信元 PE とデフォルト MDT グループに関する情報を必要とします。この情報は、送信元 PE に向けて (S, G) 加入メッセージを送信し、送信元 PE からの配信ツリーを構築するために使用されます。（RP は必要ありません）。送信元 PE アドレスとデフォルト MDT グループ アドレスは、BGP を使用して送信されます。

BGP 拡張コミュニティ

BGP 拡張コミュニティを使用すると、PE ループバック（発信元アドレス）情報は VPNv4 プレフィックスとしてルート識別子（RD）タイプ 2 を使用して送信されます（ユニキャスト VPNv4 プレフィックスと区別するため）。MDT グループ アドレスは、BGP 拡張コミュニティに伝えられます。VPNv4 アドレスに組み込まれた送信元と拡張コミュニティ内のグループの組み合わせを使用すると、同じ MVRF インスタンス内の PE ルータは相互に SSM ツリーを確立できます。

（注） MDT SAFI サポートが導入される前、BGP 拡張コミュニティの属性は、IETF によって標準化される前のソース PE およびデフォルト MDT グループの IP アドレスをアドバタイズするための暫定的ソリューションとして使用されていました。しかし、MVPN 環境の BGP 拡張コミュニティ属性には一定の制限があります。AS 間シナリオでは使用できず（属性が非推移的であるため）、RD タイプ 2 が使用されます（これはサポートされる標準ではありません）。

マルチキャスト VPN の設定方法

データ マルチキャスト グループの設定

データ MDT グループには、VPN 、VRF、PE デバイスごとに最大 256 のマルチキャスト グループを含むことができます。データ MDT グループの作成に使用されるマルチキャスト グループは、設定済み IP アドレスのプールからダイナミックに選択されます。デバイス でデータ マルチキャスト グループを設定するには、次の手順を使用します。

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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	**vrf definition** *vrf-name* 例： `Device(config)# vrf definition vrf1`	VRF コンフィギュレーション モードを開始し、VRF 名を割り当てることにより VPN ルーティング インスタンスを定義します。
ステップ 4	**rd** *route-distinguisher* 例： `Device(config-vrf)# rd 1:1`	VRF のルーティング テーブルと転送テーブルを作成します。 • *route-distinguisher* 引数では、8 バイトの値を IPv4 プレフィックスに追加して VPN IPv4 プレフィックスを作成することを指定します。*route-distinguisher* は、次のいずれかの形式で入力できます。 • 16 ビット ASN：32 ビット数値。たとえば、101:3 と指定します。 • 32 ビット IP アドレス：16 ビット数値。たとえば、192.168.122.15:1 と指定します。

	コマンドまたはアクション	目的
ステップ 5	**route-target both** *ASN:nn or IP-address:nn* 例： `Device(config-vrf)# route-target both 1:1`	VRF 用にルート ターゲット拡張コミュニティを作成します。**both** キーワードを使用すると、ルーティング情報のターゲット VPN 拡張コミュニティからのインポート、およびターゲット VPN 拡張コミュニティへのエクスポートの両方が行われます。
ステップ 6	**address family ipv4 unicast** *value* 例： `Device(config-vrf)# address family ipv4 unicast`	VRF アドレス ファミリ コンフィギュレーション モードを開始して、VRF のアドレス ファミリを指定します。 • **ipv4** キーワードは、VRF の IPv4 アドレス ファミリを指定します。
ステップ 7	**mdt default** *group-address* 例： `Device(config-vrf-af)# mdt default 226.10.10.10`	VRF に、データ MDT グループのマルチキャスト グループ アドレスの範囲を設定します。 • このコマンドによって、トンネルインターフェイスが作成されます。 • デフォルト MDT グループ アドレス設定は、同じ VRF 内のすべての PE で同一にする必要があります。
ステップ 8	**mdt data** グループ番号 例： `Device(config-vrf-af)# mdt data 232.0.1.0 0.0.0.31`	データ MDT プールで使用されるアドレスの範囲を指定します。
ステップ 9	**mdt datathreshold** *kbps* 例： `Device(config-vrf-af)# mdt data threshold 50`	しきい値を *kbps* 単位で指定します。範囲は 1 ～ 4294967 です。
ステップ 10	**mdt log-reuse** 例： `Device(config-vrf-af)# mdt log-reuse`	（任意）データ MDT 再使用の記録をイネーブルにし、データ MDT が再使用された場合に、syslog メッセージを生成します。

	コマンドまたはアクション	目的
ステップ 11	**end** 例： `Device(config-vrf-af)# end`	特権 EXEC モードに戻ります。

VRF のデフォルト MDT グループの設定

VRF にデフォルト MDT グループを設定するには、次の作業を実行します。

デフォルト MDT グループは、同じ VPN に属するすべてのデバイスに設定された同じグループである必要があります。送信元IPアドレスは、BGPセッションの送信元を特定するために使用するアドレスです。

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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	**ip multicast-routing** 例： `Device(config)# ip multicast-routing`	マルチキャスト ルーティングをイネーブルにします。
ステップ 4	**ip multicast-routing vrf** *vrf-name* 例： `Device(config)# ip multicast-routing vrf vrf1`	MVPN VRF インスタンスをサポートします。

	コマンドまたはアクション	目的
ステップ 5	**vrf definition** *vrf-name* 例： `Device(config)# vrf definition vrf1`	VRF コンフィギュレーション モードを開始し、VRF 名を割り当てることにより VPN ルーティング インスタンスを定義します。
ステップ 6	**rd** *route-distinguisher* 例： `Device(config-vrf)# rd 1:1`	VRF のルーティング テーブルと転送テーブルを作成します。 • *route-distinguisher* 引数では、8 バイトの値を IPv4 プレフィックスに追加して VPN IPv4 プレフィックスを作成することを指定します。*route-distinguisher* は、次のいずれかの形式で入力できます。 • 16 ビット ASN：32 ビット数値。たとえば、101:3 と指定します。 • 32 ビット IP アドレス：16 ビット数値。たとえば、192.168.122.15:1 と指定します。
ステップ 7	**route-target both** *ASN:nn or IP-address:nn* 例： `Device(config-vrf)# route-target both 1:1`	VRF 用にルート ターゲット拡張コミュニティを作成します。**both** キーワードを使用すると、ルーティング情報のターゲット VPN 拡張コミュニティからのインポート、およびターゲット VPN 拡張コミュニティへのエクスポートの両方が行われます。
ステップ 8	**address family ipv4 unicast** *value* 例： `Device(config-vrf)# address family ipv4 unicast`	VRF アドレス ファミリ コンフィギュレーション モードを開始して、VRF のアドレス ファミリを指定します。 • **ipv4** キーワードは、VRF の IPv4 アドレス ファミリを指定します。
ステップ 9	**mdt default** *group-address* 例： `Device(config-vrf-af)# mdt default 226.10.10.10`	VRF に、データ MDT グループのマルチキャスト グループ アドレスの範囲を設定します。 • このコマンドによって、トンネル インターフェイスが作成されます。 • デフォルト MDT グループ アドレス設定は、同じ VRF 内のすべての PE で同一にする必要があります。

	コマンドまたはアクション	目的
ステップ 10	**end** 例： `Device(config-vrf-af)# end`	特権 EXEC モードに戻ります。
ステップ 11	**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 12	**ip pim vrf** *vrf-name***rp-address** *value* 例： `Device(config-vrf-af)# ip pim vrf vrf1 rp-address 1.1.1.1`	RP コンフィギュレーション モードを開始します。

マルチキャスト VPN での BGP の MDT アドレス ファミリの設定

PE デバイスに MDT アドレス ファミリ セッションを設定し、MVPN の MDT ピアリング セッションを確立するには、次の作業を実行します。

はじめる前に

MDT アドレス ファミリを通して MVPN ピアリングを確立する前に、CE デバイスに VPN サービスを提供する PE デバイス上の BGP ネットワークおよびマルチプロトコル BGP に、MPLS およびシスコ エクスプレス フォワーディング（CEF）を設定する必要があります。

（注） 次のポリシー設定パラメータは、サポートされていません。

- ルートオリジネータ属性
- ネットワーク層到着可能性情報（NLRI）プレフィックス フィルタリング（プレフィックス リスト、配信リスト）
- 拡張コミュニティ属性（ルート ターゲットおよび発信元サイト）

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 モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
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** *as-number* 例： `Device(config)# router bgp 65535`	ルータ コンフィギュレーション モードを開始して、BGP ルーティング プロセスを作成します。
ステップ 4	**address-family ipv4 mdt** 例： `Device(config-router)# address-family ipv4 mdt`	アドレス ファミリ コンフィギュレーションを開始し、IP MDT アドレス ファミリ セッションを作成します。
ステップ 5	**neighbor** *neighbor-address***activate** 例： `Device(config-router-af)# neighbor 192.168.1.1 activate`	このネイバーの MDT アドレス ファミリをイネーブルにします。
ステップ 6	**neighbor** *neighbor-address***send-community** [**both** | **extended** | **standard**] 例： `Device(config-router-af)# neighbor 192.168.1.1 send-community extended`	指定されたネイバーとのコミュニティおよび（または）拡張コミュニティの交換をイネーブルにします。
ステップ 7	**exit** 例： `Device(config-router-af)# exit`	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 8	**address-family vpnv4** 例： `Device(config-router)# address-family vpnv4`	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、VPNv4 アドレス ファミリ セッションを作成します。
ステップ 9	**neighbor** *neighbor-address***activate** 例： `Device(config-router-af)# neighbor 192.168.1.1 activate`	このネイバーの VPNv4 アドレス ファミリをイネーブルにします。
ステップ 10	**neighbor** *neighbor-address***send-community** [**both** | **extended** | **standard**] 例： `Device(config-router-af)# neighbor 192.168.1.1 send-community extended`	指定されたネイバーとのコミュニティおよび（または）拡張コミュニティの交換をイネーブルにします。
ステップ 11	**end** 例： `Device(config-router-af)# end`	アドレス ファミリ コンフィギュレーション モードを終了して、特権 EXEC モードを開始します。

MDT デフォルト グループの情報の確認

手順

ステップ 1 **enable**

例：
`Device>` **enable**

特権 EXEC モードをイネーブルにします。

- プロンプトが表示されたら、パスワードを入力します。

ステップ 2 **showippim**[**vrf** *vrf-name*] **mdtbgp**

例：
```
Device# show ip pim mdt bgp

MDT-default group 232.2.1.4
rid:1.1.1.1 next_hop:1.1.1.1
```

MDT デフォルト グループの RD の BGP アドバタイズメントに関する情報を表示します。

ステップ 3 **showippim**[**vrf** *vrf-name*] **mdtsend**

例：
```
Device# show ip pim mdt send

MDT-data send list for VRF:vpn8
  (source, group)                      MDT-data group      ref_count
  (10.100.8.10, 225.1.8.1)             232.2.8.0           1
  (10.100.8.10, 225.1.8.2)             232.2.8.1           1
  (10.100.8.10, 225.1.8.3)             232.2.8.2           1
  (10.100.8.10, 225.1.8.4)             232.2.8.3           1
  (10.100.8.10, 225.1.8.5)             232.2.8.4           1
  (10.100.8.10, 225.1.8.6)             232.2.8.5           1
  (10.100.8.10, 225.1.8.7)             232.2.8.6           1
  (10.100.8.10, 225.1.8.8)             232.2.8.7           1
  (10.100.8.10, 225.1.8.9)             232.2.8.8           1
  (10.100.8.10, 225.1.8.10)            232.2.8.9           1
```

指定されたデバイスが行った MDT アドバタイズメントを含む MDT データ グループに関する詳細情報を表示します。

ステップ 4 **showippimvrf** *vrf-name* **mdthistoryinterval** *minutes*

例：
```
Device# show ip pim vrf vrf1 mdt history interval 20

   MDT-data send history for VRF - vrf1 for the past 20 minutes
MDT-data group        Number of reuse
     10.9.9.8           3
     10.9.9.9           2
```

過去に設定されたインターバル中に再利用されたデータ MDT を表示します。

マルチキャスト VPN の設定例

例：MVPN および SSM の設定

次の例では、PIM-SSM がバックボーンに設定されています。そのため、デフォルト グループとデータ MDT グループは、IP アドレスの SSM 範囲内に設定されています。VPN の内部では、PIM-SM が設定され、Auto-RP アナウンスのみが受け入れられます。

```
ip vrf vrf1
 rd 1:1
 route-target export 1:1
 route-target import 1:1
 mdt default 232.0.0.1
```

```
 mdt data 232.0.1.0 0.0.0.255 threshold 500 list 101
!
ip pim ssm default
ip pim vrf vrf1 accept-rp auto-rp
```

例：マルチキャスト ルーティングの VPN のイネーブル化

次の例では、マルチキャスト ルーティングは、vrf1 という VPN ルーティング インスタンスを使用してイネーブル化されます。

```
ip multicast-routing vrf1
```

例：データ MDT グループ用のマルチキャスト グループ アドレス範囲の設定

次の例では、VPN ルーティング インスタンスは、blue という VRF が割り当てられます。VPN VRF の MDT デフォルト グループは 239.1.1.1、MDT グループのマルチキャスト グループ アドレスの範囲は 239.1.2.0（ワイルドカード ビットが 0.0.0.3）です。

```
ip vrf blue
 rd 55:1111
 route-target both 55:1111
 mdt default 239.1.1.1
 mdt data 239.1.2.0 0.0.0.3
 end
```

例：マルチキャスト ルートの数の制限

次の例では、マルチキャスト ルーティング テーブルに追加できるマルチキャスト ルートの数が 200,000 に設定され、警告メッセージが発生する原因となる mroute の数のしきい値が 20,000 に設定されています。

```
!
ip multicast-routing
ip multicast-routing vrf cisco
ip multicast cache-headers
ip multicast route-limit 200000 20000
ip multicast vrf cisco route-limit 200000 20000
no mpls traffic-eng auto-bw timers frequency 0
!
```

マルチキャスト VPN の設定に関するその他の参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
この章で使用するコマンドの完全な構文および使用方法の詳細。	*Command Reference (Catalyst 9300 Series Switches)* の「Multicast VPN Commands」の項を参照してください

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

マルチキャスト VPN の設定の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよび Cisco ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *17*：マルチキャスト *VPN* の機能情報

リリース	変更箇所
Cisco IOS XE Everest 16.5.1a	この機能が導入されました。

通告

Trademarks

索 引

ま