Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ L2VPN およびイーサネット サービス コンフィギュレーション ガイド

Cisco IOS XR ソフトウェア リリース 4.3.x

Text Part Number: OL-28379-01-J

【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。

Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ L2VPN およびイーサネット サービス コンフィギュレーション ガイド

CONTENTS

はじめに

『*Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ L2VPN およびイーサネット サービス コンフィギュレーション ガイド*』で説明する内容は、次のとおりです。

マニュアルの変更履歴

表 1 に、初版後、このマニュアルに加えられた技術的な変更の履歴を示します。

表 1　　マニュアルの変更履歴

リビジョン	日付	変更点
OL-28379-01-J	2012 年 12 月	このマニュアルの最初のリリース。

マニュアルの入手方法およびテクニカル サポート

マニュアルの入手方法、テクニカル サポート、その他の有用な情報については、次の URL で、毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

『*What's New in Cisco Product Documentation*』は RSS フィードとして購読できます。また、リーダー アプリケーションを使用してコンテンツがデスクトップに直接配信されるように設定することもできます。RSS フィードは無料のサービスです。シスコは現在、RSS バージョン 2.0 をサポートしています。

リリース 4.3.x の新機能と変更点

次の表に、『Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ L2VPN およびイーサネット サービス コンフィギュレーション ガイド』における新機能および変更点の情報を要約し、その参照先を示します。

表 2　新機能および変更された機能

機能	説明	導入 / 変更が適用されたリリース	参照先
Any Transport over MPLS（AToM）iMSG	この機能が導入されました。	リリース 4.3.0	*ポイントツーポイント レイヤ 2 サービス モジュールの実装* • AToM iMSG • AToM IP インターワーキングの設定 AToM iMSG 機能の設定および確認に使用するコマンドについては、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference, Release 4.3.x*』の「*Point to Point Layer 2 Services Commands*」を参照してください。
L2VPN over GRE	この機能が導入されました。	リリース 4.3.0	*マルチポイント レイヤ 2 サービス モジュールの実装* • L2VPN over GRE • L2VPN over GRE の設定
疑似回線ヘッドエンド	この機能が導入されました。	リリース 4.3.0	*マルチポイント レイヤ 2 サービス モジュールの実装* • 疑似回線ヘッドエンド • 疑似回線ヘッドエンドの設定 疑似回線ヘッドエンド機能の設定および確認に使用するコマンドについては、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference, Release 4.3.x*』の「*Point to Point Layer 2 Services Commands*」を参照してください。
L2VPN ノンストップ ルーティング	この機能が導入されました。	リリース 4.3.0	*ポイントツーポイント レイヤ 2 サービス モジュールの実装* • L2VPN ノンストップ ルーティング • L2VPN ノンストップ ルーティングの設定 L2VPN ノンストップ ルーティング機能の設定および確認に使用するコマンドについては、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference, Release 4.3.x*』の「*Point to Point Layer 2 Services Commands*」を参照してください。

機能	説明	導入 / 変更が適用されたリリース	参照先
プロバイダー バックボーン ブリッジ VPLS	この機能が導入されました。	リリース 4.3.0	*IEEE 802.1ah プロバイダー バックボーン ブリッジ モジュールの実装* • PBB VPLS の設定 • PBB-VPLS の設定：例 プロバイダー バックボーン ブリッジ VPLS 機能の設定および確認に使用するコマンドについては、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference, Release 4.3.x*』の「*Point to Point Layer 2 Services Commands*」、「*Multipoint Layer 2 Services Commands*」、および「*Provider Backbone Bridge Commands*」を参照してください。
Multiple I-SID Registration Protocol Lite	この機能が導入されました。	リリース 4.3.0	*IEEE 802.1ah プロバイダー バックボーン ブリッジ モジュールの実装* • Multiple I-SID Registration Protocol Lite • MIRP Lite の設定：例 プロバイダー バックボーン ブリッジ VPLS 機能の設定および確認に使用するコマンドについては、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference, Release 4.3.x*』の「*Point to Point Layer 2 Services Commands*」、「*Multipoint Layer 2 Services Commands*」、および「*Provider Backbone Bridge Commands*」を参照してください。
バンドル インターフェイス上の Per-VLAN スパニングツリー アクセス ゲートウェイ	この機能が導入されました。	リリース 4.3.0	*マルチ スパニングツリー プロトコル モジュールの実装* • バンドル インターフェイスの PVSTAG
疑似回線のグループ化	この機能が導入されました。	リリース 4.3.0	*ポイントツーポイント レイヤ 2 サービス モジュールの実装* • 疑似回線のグループ化 • 疑似回線のグループ化のイネーブル化

Cisco ASR 9000 シリーズ ルータ キャリア イーサネット モデル

このモジュールでは、レイヤ 2（L2）の機能および規格について紹介します。このモジュールでは、Cisco IOS XR ソフトウェアをサポートする Cisco ASR 9000 シリーズ アグリゲーション サービス ルータの L2VPN 機能を設定する方法についても説明します。

分散ギガビット イーサネットおよび 10 ギガビット イーサネットのアーキテクチャと機能により、サービス プロバイダーは、ルータと POP 内の他のシステム（コア ルータ、エッジ ルータ、L2 スイッチ、レイヤ 3（L3）スイッチなど）を相互接続するために設計された、高密度、高帯域幅のネット ワーキング ソリューションを提供でき、その一方でネットワークのスケーラビリティおよびパフォーマンスも提供されます。

（注） ここでは、[Management Ethernet] インターフェイスの設定情報は説明しません。管理イーサネット インターフェイスを設定し、Telnet サーバをイネーブルにする方法については、『*Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ Getting Started Guide*』を参照してください。ルーティングのために管理イーサネット インターフェイスを設定するには、または管理イーサネット インターフェイスの設定を変更するには、「*Advanced Configuration and Modification of the Management Ethernet Interface on the Cisco ASR 9000 Series Router*」モジュールを参照してください。

Cisco ASR 9000 シリーズ ルータ のイーサネット インターフェイス設定の機能履歴

リリース	変更内容
リリース 3.7.2	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 4.1.1	バンドル インターフェイスの EFP のスケーラビリティが導入されました。

内容

レイヤ 2 イーサネット インターフェイスを設定するための前提条件

イーサネット インターフェイスを設定する前に、次のタスクと条件が満たされていることを確認してください。

- このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

  ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- 次のラインカードの少なくとも 1 つが Cisco ASR 9000 シリーズ ルータに取り付けられていることを確認してください。
  - 4 ポート 10 ギガビット イーサネット（4 x 10 GE）ラインカード
  - 8 ポート 10 ギガビット イーサネット（4 x 10 GE）ラインカード
  - 40 ポート 1 ギガビット イーサネット ラインカード
- インターフェイスの IP アドレスがわかっていること。
- 汎用インターフェイス名に汎用表記法の *rack/slot/module/port* を適用する方法を理解しています。

Cisco ASR 9000 シリーズ ルータ レイヤ 2 理論と規格準拠

イーサネット インターフェイスを設定するには、次の概念を理解している必要があります。

- 「イーサネット テクノロジーの概要」（P.19）
- 「キャリア イーサネット サービス」（P.19）
- 「イーサネット インターフェイスでのレイヤ 2 VPN」（P.23）
- 「ギガビット イーサネット プロトコル規格の概要」（P.24）
- 「MAC アドレス」（P.25）
- 「イーサネット MTU」（P.25）
- 「イーサネット インターフェイスでのフロー制御」（P.26）
- 「VRRP」（P.26）
- 「HSRP」（P.26）
- 「イーサネット インターフェイスのリンクのオートネゴシエーション」（P.27）
- 「イーサネット フロー ポイントとは」（P.27）
- 「EFP 出力フィルタリング」（P.29）
- 「802.1Q VLAN」（P.33）

イーサネット テクノロジーの概要

イーサネットは IEEE 802.3 国際規格によって定義されています。イーサネットによって、同軸ケーブル、ツイストペアケーブル、または光ファイバケーブルで、最大 1024 ノードの接続が可能になります。

Cisco ASR 9000 シリーズ ルータは、ギガビット イーサネット（1000 Mbps）インターフェイスおよび 10 ギガビット イーサネット（10 Gbps）インターフェイスをサポートしています。

キャリア イーサネット サービス

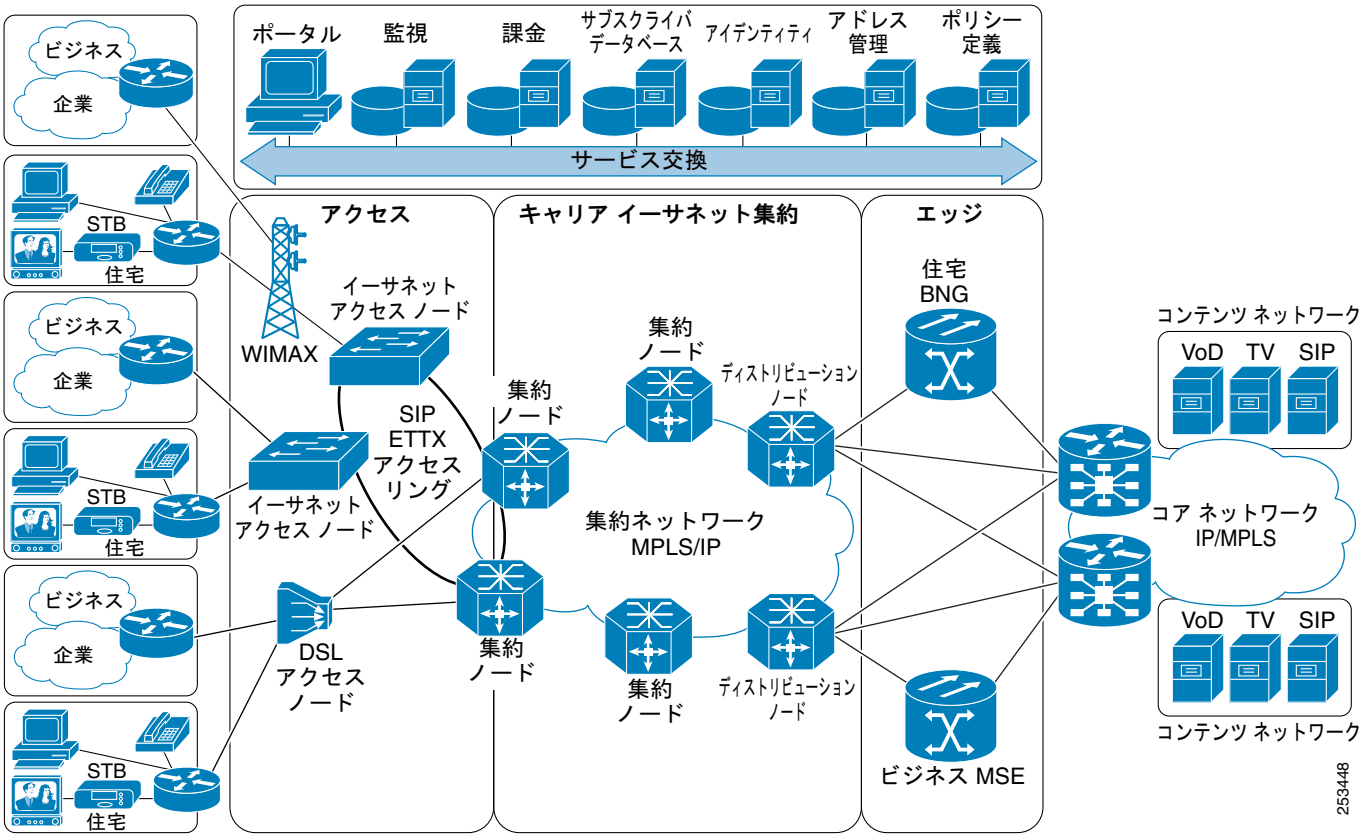
シスコおよびメトロ イーサネット フォーラム（MEF）は、次の主な L2 イーサネット サービス タイプを承認しています。サービス名は異なりますが、機能は同じです。次のサービスです。

- イーサネット ワイヤ サービス（EWS）
- イーサネット リレー サービス（ERS）
- イーサネット マルチポイント サービス（EMS）
- イーサネット フロー ポイント（EFP）
- Ethernet Virtual Connection（EVC）

イーサネット WAN（EWAN）について説明する際に、次の用語を使用します。

- CE（カスタマー エッジ）：サービス プロバイダーに接続するカスタマー デバイス
- PE（プロバイダー エッジ）：カスタマーに接続するサービス プロバイダー デバイス
- UNI：CE と PE 間の接続
- AC：CE を PE に接続する物理または仮想回線
- 多重化 UNI：複数の VLAN フローをサポートする UNI
- 疑似回線：サービス プロバイダー ネットワーク内のエンドツーエンド パスを示すために使用する用語

図 1　　EWAN の用語



イーサネット ワイヤ サービス

イーサネット ワイヤ サービスは、ポイントツーポイントのイーサネット セグメントをエミュレートするサービスです。これは、プロバイダー エッジが L2 で動作し、通常 L2+ ネットワークで実行される以外、イーサネット専用回線（EPL）、レイヤ 1 ポイントツーポイント サービスに似ています。EWS は特定の UNI で受信されたすべてのフレームをカプセル化し、フレームに含まれる内容を参照せずに、これらのフレームを単一出力 UNI に転送します。このサービスの動作は EWS を VLAN タグ付きフレームで使用できることを示します。VLAN タグは、一部の例外を除いて EWS（ブリッジ プロトコル データ ユニット（BPDU））に対して透過的です。これらの例外には、IEEE 802.1x、IEEE 802.2ad、および IEEE 802.3x が含まれます。これは、これらのフレームがローカルで意味を持ち、カスタマーと SP の両方がそれらのフレームをローカルで終端処理できる利点があるためです。

サービス プロバイダーはインターフェイスでフレームを単純に受け取り、実際のフレームを参照せずにこれらを送信するため（ただし、形式と長さが特定のインターフェイスに適合していることは確認します）、EWS はカスタマーのイーサネット フレーム内にある VLAN タグに関与しません。

EWS は all-to-one バンドリングの概念に対応しています。つまり、EWS はポイントツーポイント回線の一方のエンドのポートと他方のエンドのポートをマッピングします。EWS はポート間サービスです。したがって、カスタマーが 1 つのスイッチまたはルータを n 個のスイッチまたはルータに接続する必要がある場合は、n 個のポートおよび n 個の疑似回線または論理回線が必要になります。

考慮すべき 1 つの重要なポイントは、EWS はイーサネット レイヤ 1 接続を広範にエミュレートするにもかかわらず、サービスは共有インフラストラクチャで提供され、したがって、すべてのインターフェイス帯域幅を常に使用できる可能性は低く、またそのようにする必要もないということです。EWS は、通常、多くのユーザが伝送パスのどこかで回線を共有する、サブライン レート サービスです。その結

果、コストが EPL のコストよりも、ほとんどの場合、小さくなります。SP は、レイヤ 1 EPL とは異なり、特定契約の特定目的を達成するために、QoS およびトラフィック エンジニアリングを実装する必要があります。ただし、カスタマー アプリケーションに本当の意味でのワイヤ レート透過サービスが必要な場合、DWDM（高密度波長分割多重）、CDWM（低密度波長分割多重）、SONET/SDH などの光送信デバイスを使用して提供される EPL サービスを検討する必要があります。

イーサネット リレー サービス

イーサネット リレー サービスは、ポイントツーポイント接続を提供する点で EWS に似ています。EWS と ERS の主な違いは、ERS は、VLAN タグを使用して、宛先の異なる複数の疑似回線を 1 つのポートとの間で多重化する点です。つまり、EPL および EWS とは異なり、ERS は、1 対多の多重化サービスです。サービス多重化は、複数の疑似回線が 1 つのアクセス インターフェイスまたは UNI を使用することを意味します。これらの回線は L2VPN 内、たとえばインターネット ゲートウェイにおいて終端可能です。サービス ユーザの観点からは、このサービス多重化機能により、インターフェイス使用の効率化、ケーブル設備の単純化、および追加インターフェイスに関連するメンテナンス コストの削減が実現します。

1 つのルータが他の n 個のルータに接続する上の同じ例を使用した場合、送信元ルータには、EWS の場合と同様に、サービス用ポートは n 個ではなく 1 個のみ必要です。サービスは、ポート間で提供する必要はなく、論理的疑似回線間でも提供できます。ERS の場合、各回線は、別のリモート ロケーションで終端可能です（図 4）。一方、EWS を使用した場合、すべてのフレームが 1 つの回線にマップされます。したがって、1 つの出力ポイントにマップされることとなります。

図 2　　ERS サービス多重化の例：1 ポート（左）をすべての宛先（右）に対し使用可能

ERS では、フレーム リレーと同様に、カスタマー デバイスはサービス プロバイダー ネットワークに接続されている単一の物理ポートを介して複数の接続にアクセスできます。ERS で提供されるサービスは、VLAN 番号が、フレーム リレーのデータ リンク接続識別子（DLCI）と同様の方法で、仮想回線識別子として使用される点において、フレーム リレーと概念が類似していると考えることができます。EWS とは異なり、ERS は BPDU を転送しません。これは、IEEE 802.1Q（VLAN タギング）がデフォルト VLAN で BPDU だけを送信するためです。ハブアンドスポーク ネットワークでは、最大で 1 つのスポークしか BPDU を受信しないため、ネットワークの残りの部分ではスパニングツリーは中断されます。したがって、ERS は、BPDU を一切送信せず、イーサネット スパニングツリーの代わりにルーティング プロトコルを実行します。こうしたルーティング プロトコルは、カスタマーおよびプロバイダーに対し、より優れた柔軟性、トラフィック決定特性、および付加価値サービスを提供します。

イーサネット マルチポイント サービス

イーサネット マルチポイント サービス（EMS）は、マルチポイント接続モデルを提供する点において EWS および ERS と異なります。EMS サービスの定義は、IETF バーチャル プライベート LAN サービス（VPLS）ワーク グループ内でまだ検討中ですので注意してください。EMS はマルチポイント モデルを使用しますが、1 つの宛先へユニキャスト パケットを転送できます。つまり、ポイントツーポイント接続をサポートします。エンド ユーザには、ネットワークは、エンドツーエンド疑似回線リンクではなく、各カスタマーが独自の VLAN またはブロードキャスト ドメインを使用する巨大イーサネット スイッチのように見えます。

EMS の例

EMS は特定のポイントツーポイント疑似回線にインターフェイスまたは VLAN をマッピングしません。代わりに、仮想イーサネット スイッチの動作を模倣します。つまり、EMS はカスタマーの MAC アドレスを使用して、サービス プロバイダー ネットワーク内の適切な出力 UNI にフレームを転送します。EMS は、イーサネット スイッチのサービス属性のエミュレートとインターフェース アソシエーションのための送信元 MAC の学習、不明ブロードキャストおよびマルチキャスト フレームのフラッディング、およびサービス ユーザのスパニングツリー プロトコルのモニタ（オプション）を実行します。注意する 1 つの重要なポイントは、サービス プロバイダーは転送ネットワーク内でスパニングツリーを使用する場合があるにもかかわらず、サービス ユーザのスパニングツリーとの相互動作がないことです。

このサービスは、L3 ではなく L2 で動作することを除き、MPLS VPN に動作が似ています。VPLS EMS は実行可能なソリューションですが、これのスケーラビリティと QoS 制御は、MPLS VPN のスケーラビリティと QoS 制御に比べると低品質です。さらに、サービス プロバイダーが付加価値レイヤ 3 サービスを提供することは、はるかに困難であり、不可能な場合もあります（これは本マニュアルで後述しています）。

イーサネット フロー ポイント

イーサネット フロー ポイント（EFP）はメイン インターフェイスのサブストリーム パーティションです。Cisco ASR 9000 シリーズ ルータでは、EFP はカプセル化ステートメントにより L2 サブインターフェイスとして実装されます。

イーサネット仮想回線

イーサネット仮想回線（EVC）はポイントツーポイント トンネルです。Cisco ASR 9000 シリーズ ルータでは、EVC は疑似回線（PW）として実装されます。

イーサネット OAM プロトコル

メトロ エリア ネットワーク（MAN）またはワイドエリア ネットワーク（WAN）テクノロジーとしてのイーサネットでは、運用管理および保守（OAM）機能の実装によって大きな恩恵が得られます。OAM 機能により、サービス プロバイダーは MAN や WAN で接続の品質をモニタできます。サービス プロバイダーは、特定のイベントをモニタし、イベントに対してアクションを実行し、必要に応じて、トラブルシューティングのために特定のインターフェイスをループバック モードにできます。リンクの片側または両側をモニタするようにイーサネット OAM 機能を設定できます。

イーサネット OAM プロトコルの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』の「*Configuring Ethernet Interfaces on the Cisco ASR 9000 Series Router*」モジュールを参照してください。

イーサネット インターフェイスでのレイヤ 2 VPN

L2VPN 接続は、IP または MPLS 対応 IP ネットワーク間の LAN の動作をエミュレートすることで、イーサネット デバイス間が共通の LAN セグメントに接続した場合と同様に通信できるようになります。

L2VPN の機能によって、サービス プロバイダー（SP）は地理的に離れたカスタマー サイトにも L2 サービスを提供できるようになります。通常、SP はアクセス ネットワークを使用して、カスタマーをコア ネットワークに接続します。このアクセス ネットワークでは、イーサネット、フレーム リレーなどの L2 テクノロジーが併用される場合があります。カスタマー サイトと近接した SP エッジ ルータ間の接続は、接続回線（AC）と呼ばれます。カスタマーからのトラフィックは、このリンク上で SP コア ネットワークのエッジへ伝送されます。次に、SP コア ネットワーク上の疑似接続のトンネルを介して、別のエッジ ルータへ伝送されます。このトラフィックはエッジ ルータによって別の AC へと伝送され、そこからカスタマーのリモート サイトへ伝送されます。

L2VPN の機能によって、異なる種類の L2 接続回線と疑似回線間の接続が可能になります。その結果、ユーザはさまざまなエンドツーエンド サービスを実装できるようになります。

Cisco IOS XR ソフトウェアは、ポイントツーポイント エンドツーエンド サービスをサポートしています。つまり、2 つのイーサネット回路が相互に接続されます。L2VPN イーサネット ポートは、次の 2 モードのいずれかで動作します。

- ポート モード：このモードでは、ポートに到達するすべてのパケットは、パケットに指定されている VLAN タグに関係なく、疑似回線上で送信されます。VLAN モードでは、l2transport コンフィギュレーション モードで設定が実行されます。
- VLAN モード：CE（カスタマー エッジ）の各 VLAN または PE（プロバイダー エッジ）リンクへのアクセス ネットワークは個別の L2VPN 接続として設定できます（VC タイプ 4 または VC タイプ 5 を使用する）。VLAN 上で L2VPN を設定する方法については、このマニュアルの「Cisco ASR 9000 シリーズ ルータ キャリア イーサネット モデル」モジュールを参照してください。VLAN モードでは、個別のサブインターフェイスで設定を実行します。

切り替えは次の 3 つの方法で実行できます。

- AC-to-PW：PE に到達したトラフィックは PW（疑似回線）を介してトンネリングされます（反対に、PW を介して到達したトラフィックは AC を介して送信されます）。これが最も一般的なシナリオです。
- ローカルの切り替え：1 つの AC 上で到達するトラフィックは、疑似接続を介さずに別の AC へ送出されます。
- PW 切り替え：PW に到達するトラフィックは AC へ送信されませんが、別の PW 上でコアに返信されます。

イーサネット インターフェイスで L2VPN を設定する場合、次の点に気を付けてください。

- L2VPN リンクは QoS（Quality of Service）および MTU（最大伝送ユニット）の設定をサポートしています。
- ネットワークでパケットを透過的に伝送することを必須にしている場合、必要に応じて、サービス プロバイダー（SP）ネットワークのエッジでパケットの宛先 MAC（メディア アクセス コントロール）アドレスを変更します。こうすることで、SP ネットワークのデバイスによるパケットの消費が回避されます。

AC と疑似回線情報を表示するには、**show interfaces** コマンドを使用します。

ギガビット イーサネット プロトコル規格の概要

ギガビット イーサネット インターフェイスは、次のプロトコル規格をサポートしています。

- IEEE 802.3 物理イーサネット インフラストラクチャ
- IEEE 802.3ab 1000BASE-T ギガビット イーサネット
- IEEE 802.3z 1000 Mbps ギガビット イーサネット
- IEEE 802.3ae 10 Gbps イーサネット

各規格の詳細については、このマニュアルで後述します。

IEEE 802.3 物理イーサネット インフラストラクチャ

IEEE 802.3 プロトコル規格では、接続するイーサネットの物理層とデータリンク層の MAC 下位層が定義されています。IEEE 802.3 では、多様な物理メディアで、また多様な速度でキャリア検知多重アクセス/衝突検出（CSMA/CD）アクセスを使用します。IEEE 802.3 規格は 10 Mbps イーサネットに対応します。IEEE 802.3 規格の拡張では、ギガビット イーサネット、10 ギガビット イーサネット、およびファスト イーサネットの実装を規定しています。

IEEE 802.3ab 1000BASE-T ギガビット イーサネット

IEEE 802.3ab プロトコル規格、つまり銅線上のギガビット イーサネット（別名 1000BaseT）は、既存のファスト イーサネット規格の拡張です。この拡張は、すでに設置されているカテゴリ 5e/6 ケーブル配線システム上のギガビット イーサネットの動作を規定しており、費用有効性の高いソリューションを実現できます。結果として、ファスト イーサネットを実行する銅線ベースの環境では既存のインフラストラクチャ上でギガビット イーサネットも実行できるため、要求の厳しいアプリケーションでもネットワークのパフォーマンスが大幅に向上します。

IEEE 802.3z 1000 Mbps ギガビット イーサネット

ギガビット イーサネットはイーサネット プロトコルの上で構築されますが、速度はファスト イーサネットの 10 倍で、1000 Mbps（1 Gbps）に上がります。ギガビット イーサネットを使用すると、デスクトップで 10 Mbps または 100 Mbps、データセンターで最高 1000 Mbps までイーサネットを拡張できます。ギガビット イーサネットは IEEE 802.3z プロトコル規格に準拠します。

ネットワーク管理者は、現在のイーサネット規格と、すでに設置されているイーサネットおよびファスト イーサネットのスイッチおよびルータのベースを利用することで、ギガビット イーサネットをサポートするために新しいテクノロジーのトレーニングや学習をし直す必要はなくなります。

IEEE 802.3ae 10 Gbps イーサネット

国際標準化組織の開放型システム間相互接続（OSI）モデルでは、イーサネットは基本的に L2 プロトコルです。10 ギガビット イーサネットでは、IEEE 802.3 イーサネット MAC プロトコル、IEEE 802.3 イーサネット フレーム形式、および IEEE 802.3 の最小および最大フレーム サイズを使用します。10 Gbps イーサネットは IEEE 802.3ae プロトコル規格に準拠します。

イーサネット モデルに忠実だった 1000BASE-X と 1000BASE-T（ギガビット イーサネット）と同様に、10 ギガビット イーサネットも速度と距離の点でイーサネットが自然に発展した結果です。10 ギガビット イーサネットは全二重方式でファイバのみのテクノロジーなので、低速で半二重方式のイーサネット テクノロジーを定義する CSMA/CD プロトコルを使用した、通信事業者に影響される多重アクセスは必要ありません。他のどの点でも、10 ギガビット イーサネットは元のイーサネット モデルに忠実です。

一般的なイーサネット規格

- イーサネット II フレーム構成（別名 DIX）。
- IEEE 802.3 フレーム構成には、LLC および LLC/SNAP プロトコル フレーム形式も含まれます。
- IEEE 802.1d MAC ブリッジおよびスパニングツリー：この規格は、ブリッジング環境での MAC ラーニングと MAC エージングを指定します。また、元のスパニングツリー プロトコルを定義します。MSTP も IEEE 802.1s および IEEE 802.1q で定義されています。
- IEEE 802.1q VLAN タギング：この規格は、VLAN タギングを定義し、またスイッチ間の従来の VLAN トランキングも定義します。技術的には、QinQ タギングおよび MSTP も定義します。Cisco ASR 9000 シリーズ ルータは ISL をサポートしません。
- IEEE 802.1ad プロバイダー ブリッジ：この規格は 802.1q のサブセットであり、多くの場合 802.1ad と呼ばれます。Cisco ASR 9000 シリーズ ルータは、規格全体には準拠していませんが、規格の機能の大部分がサポートされます。

MAC アドレス

MAC アドレスは、L2 でインターフェイスを識別する一意の 6 バイト アドレスです。

イーサネット MTU

イーサネットの最大伝送単位（MTU）は、最大フレームのサイズから 4 バイトのフレーム チェック シーケンス（FCS）を引いた値です。この MTU がイーサネット ネットワークで伝送できるサイズです。パケットの宛先に到達するまでに経由する各物理ネットワークは、MTU が異なる可能性があります。

Cisco IOS XR ソフトウェアは、2 種類のフレーム転送プロセスをサポートしています。

- IPV4 パケットのフラグメンテーション：このプロセスでは、ネクスト ホップの物理ネットワークの MTU 内に収まるように、必要に応じて IPv4 パケットが分割されます。

  （注） IPv6 はフラグメンテーションをサポートしません。

- MTU の検出プロセスによる最大パケット サイズの決定。このプロセスは、すべての IPv6 デバイスと発信側の IPv4 デバイスに使用できます。このプロセスでは、分割せずに送信できる IPv6 または IPv4 パケットの最大サイズを、発信側の IP デバイスが決定します。最大パケットは、IP 発信元デバイスおよび IP 宛先デバイス間にあるすべてのネットワークの中で、最小 MTU と等値です。このパス内にあるすべてのネットワークの最小 MTU よりもパケットが大きい場合、そのパケットは必要に応じて分割されます。このプロセスによって、発信側のデバイスから大きすぎる IP パケットが送信されなくなります。

標準フレーム サイズを超えるフレームの場合、ジャンボ フレームのサポートが自動的にイネーブルになります。デフォルト値は標準フレームの場合は 1514、802.1Q タグ付きフレームの場合は 1518 です。この数値に 4 バイトの FCS は含まれません。

イーサネット インターフェイスでのフロー制御

10 ギガビット イーサネット インターフェイスでのフロー制御は、フロー制御ポーズ フレームを定期的に送信する処理で構成されます。この処理は、標準の管理インターフェイスで使用される通常の全二重および半二重のフロー制御とは根本的に異なります。Cisco ASR 9000 シリーズ ルータでは、入力および出力の両方でフロー制御はデフォルトではオフになっています。

VRRP

仮想ルータ冗長プロトコル（VRRP）によって、静的なデフォルトのルーティング環境に固有の単一障害点が除外されます。VRRP は、仮想ルータの役割を LAN 上の VPN コンセントレータの 1 つに動的に割り当てるという、選択プロトコルを規定します。仮想ルータに割り当てる IP アドレスを制御する VRRP VPN コンセントレータはマスターと呼ばれ、送信されたパケットをその IP アドレスに転送します。マスターが使用不可になると、バックアップ VPN コンセントレータがマスターの役割を引き継ぎます。

VRRP の詳細については、『*Cisco ASR 9000 Series Routers IP Addresses and Services Configuration Guide*』の「*Implementing* VRRP」モジュールを参照してください。

HSRP

Hot Standby Routing Protocol（HSRP）はシスコの独自プロトコルです。HSRP は障害の発生時にルータのバックアップを用意するルーティング プロトコルです。複数のルータが同じセグメントのイーサネット、FDDI、またはトークンリング ネットワークに接続し、LAN 上にある単一の仮想ルータにとして連携します。これらのルータは同じ IP アドレスおよび MAC アドレスを共有するため、ルータのいずれかに障害が発生した場合でも、LAN 上のホストはそのまま同じ IP アドレスおよび MAC アドレスにパケットを転送できます。ルーティングの担当デバイスの切り替えは、ユーザには検知されません。

HSRP は、特定の状況で IP トラフィックを中断しないフェールオーバーをサポートし、ホストからは単一のルータを使用しているように見え、使用している第 1 ホップのルータに障害が発生した場合でも接続を維持できるように設計されています。つまり、HSRP は、発信元のホストが第 1 ホップのルータの IP アドレスを動的に取得できない場合でも、第 1 ホップのルータの障害に対処できます。複数のルータが HSRP に参加し、連携して単一の仮想ルータであるように見せます。HSRP によって、確実に単一のルータが仮想ルータの代わりにパケットを転送します。エンド ホストがそのパケットを仮想ルータに転送します。

パケットを転送するルータは、*アクティブ ルータ*と呼ばれます。アクティブ ルータに障害が発生した場合、代わりになるスタンバイ ルータが選択されます。HSRP には、参加するルータの IP アドレスを使用して、アクティブ ルータとスタンバイ ルータを決定するメカニズムがあります。アクティブ ルータに障害が発生した場合、スタンバイ ルータが引き継ぐことができます。ホストの接続が長く切断することはありません。

HSRP はユーザ データグラム プロトコル（UDP）上で実行され、ポート番号 1985 を使用します。ルータは、プロトコル パケットの発信元アドレスとして仮想アドレスではなく実際の IP アドレスを使用するため、HSRP ルータは相互を識別できます。

HSRP の詳細については、『*Cisco ASR 9000 Series Routers IP Addresses and Services Configuration Guide*』の「*Implementing HSRP*」モジュールを参照してください。

イーサネット インターフェイスのリンクのオートネゴシエーション

リンクのオートネゴシエーションによって、リンク セグメントを共有するデバイスは、最高のパフォーマンス モードの相互運用で自動的に設定されます。イーサネット インターフェイスでリンクのオートネゴシエーションをイネーブルにするには、インターフェイス コンフィギュレーション モードで **negotiation auto** コマンドを使用します。ラインカードのイーサネット インターフェイスで、リンクのオートネゴシエーションはデフォルトでディセーブルです。

（注） **negotiation auto** コマンドは、ギガビット イーサネット インターフェイスだけで使用できます。

イーサネット フロー ポイントとは

イーサネット フロー ポイント（EFP）とは、物理またはバンドル インターフェイスにおいて、トラフィックの分類に使用されるレイヤ 2 の論理サブインターフェイスです。

物理インターフェイスは､ギガビット イーサネット 0/0/0/1 または 10 ギガビット イーサネット 0/0/0/0 インターフェイスの場合があり、ラインカードのポートがあります。バンドル インターフェイスは、物理インターフェイスをグループ化することにより作成される仮想インターフェイスです。

たとえば、ギガビット イーサネット 0/0/0/1、10 ギガビット イーサネット 0/0/0/0 などの物理インターフェイスは、バンドル インターフェイスのメンバーとして設定できます。

物理インターフェイスをグループ化すると、以下が可能になります。

- ルーティング エントリの削減
- バンドル インターフェイスの帯域幅の増加
- バンドル メンバー間でのトラフィックのバランシング

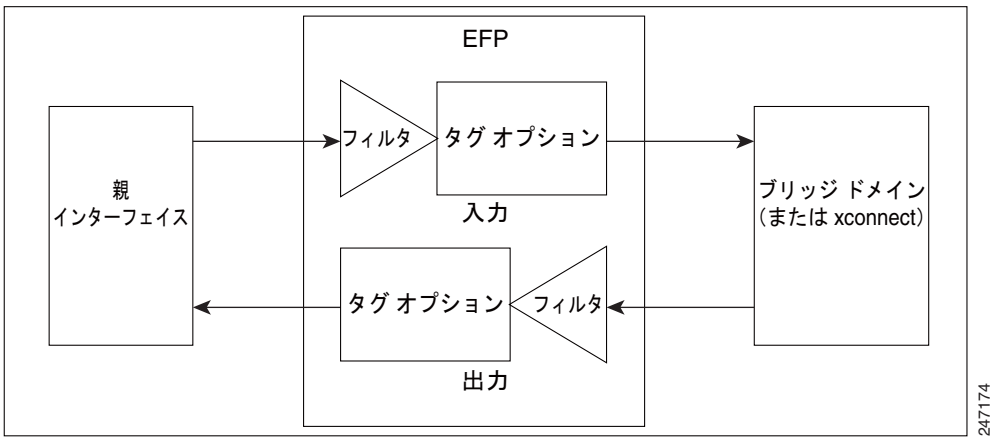
EFP の特徴は、次のとおりです。

- EFP は、インターフェイスで Ethernet Virtual Connection（EVC）の論理的な境界ポイントを表します。2 つ以上の UNI を関連付ける EVC では、EVC が通過するすべてのデバイスの各インターフェイスにフロー ポイントがあります。
- EFP は、特定のサービスのインスタンス化と見なすことができます。EFP は、一連のフィルタによって定義されます。これらのフィルタは、特定の EFP に属するフレームを分類するために、すべての入力トラフィックに適用されます。EFP *フィルタは*一連のエントリであり、各エントリはパケットの先頭部分に類似しています（送信元/宛先 MAC アドレスは無視します）。各エントリには、通常、0、1、または 2 つの VLAN タグが含まれます。パケットが、フィルタのエントリと同じタグで始まる場合、そのパケットはフィルタに一致することになります。パケットの先頭部分がフィルタのエントリに対応しない場合、パケットはフィルタに一致しません。
- EFP は次の 4 つの役割を果たします。
  - 特定のインターフェイスで特定のフローに属するすべてのフレームを識別します。
  - 入力および出力イーサネット ヘッダー処理を実行します。
  - 識別されたフレームに機能を追加します。
  - オプションで、データ パスでのフレームの転送方法を定義します。

ルータの各種インターフェイスに EFP が設定されている場合、トラフィック フローに対しさまざまな操作を実行できます。また、ルータの 1 つ以上の入力 EFP から 1 つ以上の出力 EFP に対し多数の方法でトラフィックをブリッジングまたはトンネリングできます。このトラフィックでは、VLAN ID、シングルまたはダブル（QinQ）カプセル化、および Ethertype が併用されます。

図 3 に、EFP のモデルを示します。

図 3　　EFP モデル



入力のどのトラフィックをその EFP に向けるか指定するために、EFP のサブインターフェイスを設定します。これは、入力で照合する VLAN、VLAN の範囲、または QinQ タギングを指定することで行います。入力のすべてのトラフィックは、各 EFP の一致条件と比較され、一致した場合には、その EFP によって処理されます。EFP によって実行される処理では、VLAN ID を変更すること、VLAN タグを追加または削除することや、Ethertype を変更することができます。

バンドル インターフェイスでの EFP のスケーラビリティの改善

次の 2 通りの方法でバンドル インターフェイスの EFP のスケーラビリティを改善できます。

- シャーシあたりの EFP の数を 32000 から 64000 に増やします。
- 単一ノード ポイントで、ラインカードあたりの EFP の数を、物理インターフェイス スケーリングと同じスケールに増やします。

次に、ラインカードあたりの EFP のスケーラビリティを改善する例を示します。

バンドル インターフェイス スケーリングが 4000、[1] 物理インターフェイス スケーリングが 16000 の B モジュール ラインカード タイプがあるとします。B モジュールの EFP のスケーラビリティは、バンドルあたり 4000 EFP のバンドルを 3 つ追加することで改善されます。

（注） バンドル インターフェイスに追加できる EFP の最大数は 4000 です。

ラインカードあたりの EFP の数は、16000 またはそれぞれ 4000 EFP の 4 つのバンドルに現在拡張されています。

EFP CLI の概要

Cisco IOS XR は、EFP および EVC 設定のための構造化 CLI を実装しています。EFP を設定するために、通常、次のコマンドが使用されます。

- **l2transport** コマンド：このコマンドは、サブインターフェイス（または物理ポート、バンドル ポートの親インターフェイス）を EFP として指定します。

1. ASR 9000 シリーズ ルータがサポートするラインカード タイプの 1 つ。

- **encapsulation** コマンド：このコマンドは、一致基準を指定するために使用されます。
- **rewrite** コマンド：このコマンドは、VLAN タグの書き換え条件を指定するために使用されます。

EFP 出力フィルタリング

EFP 出力フィルタリング機能は、EFP 出力トラフィックをフィルタリングする方法を提供し、指定するすべての EFP の出力トラフィックが入力一致基準に準拠するようにします。

入力 EFP は出力 EFP に似ています。ルータは、EFP の入力一致条件に一致するトラフィックを、その EFP のトラフィックとして送信するように設定されます。これが実行されないようにルータを設定することができます。不一致の出力 EFP トラフィックがルータを出ることを防ぐための予防手段はありません。

Cisco ASR 9000 シリーズ ルータ では、同じブリッジ ドメイン内の異なるポートで異なる VLAN を使用できます。これにより、ブリッジは、パケットの VLAN タグが設定されていないポートからパケットを転送できます。EFP 出力フィルタリングは、これを確認し、出力ポートで無効なパケットを廃棄します。

EFP のフレームの識別

EFP は、イーサネット カプセル化に関係なく、指定ポートで特定フローに属するフレームを識別します。EFP は、フレーム ヘッダー内のフィールドに基づいてフローまたは EFP に柔軟にフレームをマッピングできます。

以下を使用して、フレームと EFP を照合できます。

- VLAN タグ
- MAC アドレス（送信元アドレス、宛先アドレス、または両方）
- 802.1p CoS ビット
- 上の複数の項目の論理的な組み合わせ：VLAN、MAC および CoS
- デフォルトの一致（つまり、特定の EFP に一致しない他のトラフィック）
- プロトコル Ethertype

次の項目を使用して、フレームと EFP を照合することはできません。

- 以下のような、最も外側のイーサネット フレーム ヘッダーおよび関連するタグの外部の情報
  - IPv4、IPv6、または MPLS のタグ ヘッダーのデータ
  - C-DMAC、C-SMAC、または C-VLAN
- 上の有効なフレーム一致の論理和：VLAN、MAC、および CoS

特定の一致条件について、以降の各項で詳しく説明します。

VLAN タグの一致

表 1 で、さまざまなカプセル化タイプとそれぞれに対応する EFP 識別子について説明します。

表 1　VLAN タグの一致

カプセル化タイプ	EFP 識別子
タグなし	**encapsulation** コマンドで **untagged** キーワードを使用する、入力物理インターフェイスまたはサブインターフェイスの静的設定。タグなしサブインターフェイスは 1 つのみ使用できます。タグなしサブインターフェイスが作成されると、トラフィックは、メイン インターフェイスではなく、このインターフェイスに送られます。
プライオリティ タグ付きイーサネット フレーム	プライオリティ タグ付きフレームは、VLAN ID がゼロの、単一 802.1Q VLAN ヘッダーを持つフレームとして定義されます。
ネイティブ VLAN	Cisco ASR 9000 シリーズ ルータはネイティブ VLAN をサポートしていません。 次のコマンドを使用してください。 **encapsulation dot1q** <*vlan-id*>, **untagged**
単一タグ付きフレーム	802.1Q カスタマー タグ付きイーサネット フレーム
二重タグ付きフレーム	802.1Q（ethertype 0x8100）二重タグ付きフレーム 802.1ad 二重タグ付きフレーム レガシー 0x9100 および 0x9200 二重タグ付きフレーム
デフォルトのタギング	最大一致のワイルドカードが設定された EFP。目的は、同じ物理インターフェイスで他の EFP に一致しないトラフィックを受信することです。

特定の EFP にマッピングするフレームを定義するときに、ワイルドカードおよび VLAN の範囲を使用できます。EFP は、単一の VLAN タグ、VLAN タグの範囲、VLAN タグのスタック、または両方の組み合わせ（VLAN スタックとワイルドカード）に基づいてフローを区別できます。EFP は、EFP モデル、カプセル化非依存にする柔軟性を提供しています。また、新しいタギングまたはトンネリング方式を追加することで、EFP を拡張できるようになっています。

MAC アドレスの一致

送信元 MAC アドレス、宛先 MAC アドレス、または両方を照合できます。いずれの場合も、MAC アドレスは完全に一致する必要があります。ワイルドカード一致または部分一致では不十分な場合があります。

802.1p CoS ビットの一致

1 つ以上の精確な CoS 一致が指定されます。CoS は 3 ビットのみであるため、8 種類の選択に制限されます。

論理結合

上記の一致基準はすべて、個別の条件すべてを満たすフレームを選択的に組み合わせることができます。

デフォルトの一致

特定の EFP に一致していない他のすべてのトラフィックと一致する単一 EFP を定義できます。

照合順序と設定の検証

照合に使用する EFP の順序を決定できる、重複 EFP を設定できます。ただし、他の EFP または親トランク インターフェイスのサブインターフェイスと競合する EFP は、設定の検証でブロックする必要があります。

優先順位は、ハードウェアでの EFP 照合の適用方法に対し使用されます。このモデルは、あいまいな一致の前に、より精度の高い一致を処理するためのモデルです。

出力の動作

EFP 一致基準は出力でも使用でき、プラットフォーム サポートに基づいて、EFP から出力できるフレームをポリシングできます。条件（送信元/宛先 MAC 一致基準は入れ替わります）に一致しないフレームはドロップされます。

機能の適用

フレームが特定の EFP に一致した後、適切な機能を適用できます。このコンテキストでは、「機能」は、QoS、ACL などの機能、および設定により指定されたフレーム操作を意味します。イーサネット インフラストラクチャは、機能オーナーが EFP に機能を適用できるように適切なインターフェイスを提供しています。そのため、EFP を表すために IM インターフェイス ハンドルが使用され、これにより機能オーナーは、通常のインターフェイスまたはサブインターフェイスで機能が管理されるのと同じ方法で、EFP で機能を管理できます。

イーサネット インフラストラクチャの一部である EFP で適用できる唯一の L2 機能は、L2 ヘッダーのカプセル化の変更です。この L2 機能については、次の項で説明します。

カプセル化の変更

EFP は、入力と出力の両方で、次の L2 ヘッダーのカプセル化の変更をサポートしています。

- 1 つまたは 2 つの VLAN タグのプッシュ処理
- 1 つまたは 2 つの VLAN タグのポップ処理

（注） この変更では、EFP に部分一致するタグのポップ処理のみ実行できます。

- 1 つまたは 2 つの VLAN タグの書き換え
  - 外部タグの書き換え
  - 2 つの外部タグの書き換え
  - 外部タグの書き換え、および追加タグのプッシュ処理
  - 外部タグの削除、および内部タグの書き換え

各 VLAN ID 操作に対して、以下を指定できます。

- VLAN タグ タイプ、つまり、C-VLAN、S-VLAN、または I-TAG。802.1Q C-VLAN タグの Ethertype は、dot1q tunneling type コマンドで定義されます。
- VLAN ID。0 は、プライオリティ タグ付きフレームを生成するために、外部 VLAN タグに対し指定できます。

（注） タグの書き換えでは、以前のタグの CoS ビットを、802.1ad カプセル化フレームの DEI ビットと同じ方法で維持する必要があります。

データ転送動作の定義

データ パスで転送される特定のイーサネット フローに属するフレームを指定するために、EFP を使用できます。次の転送ケースが、Cisco IOS XR ソフトウェアでの EFP に対しサポートされます。

- L2 スイッチド サービス（ブリッジング）: EFP はブリッジ ドメインにマッピングされ、そこでフレームは宛先 MAC アドレスに基づいてスイッチングされます。これには、マルチポイント サービスが含まれます。
  - イーサネットとイーサネットのブリッジング
  - Virtual Private LAN Service（VPLS）
- L2 スイッチド サービス（AC と AC の xconnect）: これは、静的に確立されるポイントツーポイント L2 アソシエーションに対応し、MAC アドレス ルックアップを必要としません。
  - イーサネットとイーサネットのローカル スイッチング : EFP は同じポートまたは別のポートの S-VLAN にマッピングされます。S-VLAN は同一にすること、または別にすることができます。
- トンネル型サービス（xconnect）: EFP はレイヤ 3 トンネルにマッピングされます。これは、ポイントツーポイント サービスのみに対応します。
  - Ethernet over MPLS（EoMPLS）
  - L2TPv3
- L2 終端サービス（レイヤ 3 サービスへのイーサネット アクセス）: EFP は、グローバル アドレスを持つ IP インターフェイス、または VRF に属する IP インターフェイスにマッピングされます（IP および MPLS レイヤ 3 VPN の両方が含まれます）。

802.1Q VLAN

VLAN とは、実際は異なる LAN セグメント上のデバイスでも、同じセグメントで接続している場合と同様に通信できるように設定された、1 つまたは複数の LAN 上にあるデバイスのグループです。VLAN は、物理接続ではなく論理接続に基づいているため、ユーザ管理、ホスト管理、帯域割り当て、およびリソースの最適化がとても柔軟です。

IEEE の 802.1Q プロトコル規格では、ブロードキャストおよびマルチキャストのトラフィックが必要以上の帯域を消費しないように、大規模なネットワークを小規模なパーツに分割することで問題に対処しています。また、内部ネットワークのセグメント間に、より高レベルのセキュリティを実現できます。

802.1Q 仕様は、イーサネット フレームに VLAN メンバーシップ情報を挿入する標準方式を確立します。

Cisco IOS XR ソフトウェア は、ギガビット イーサネットおよび 10 ギガビット イーサネット インターフェイスでの VLAN サブ インターフェイスの設定をサポートしています。

802.1Q タグ付きフレーム

IEEE 802.1Q タグ ベースの VLAN は、MAC ヘッダーの特別なタグを使用し、ブリッジでのフレームの VLAN メンバーシップを識別できます。このタグは、VLAN および Quality of Service（QoS）のプライオリティの識別に使用されます。VLAN は、手動での入力によってスタティックに作成することも、Generic Attribute Registration Protocol（GARP）VLAN Registration プロトコル（GVRP）を介してダイナミックに作成することもできます。VLAN ID は、フレームを特定の VLAN に関連付けて、スイッチがネットワークでフレームを処理する必要があるという情報を提供します。タグ付きフレームは、タグなしフレームよりも 4 バイト長く、イーサネット フレームの Type および Length フィールドにある 2 バイトの Tag Protocol Identifier（TPID）フィールドと、イーサネット フレームの Source Address フィールドの後ろから始まる 2 バイトの Tag Control Information（TCI）が含まれます。

サブインターフェイス

サブインターフェイスは、ハードウェア インターフェイス上に作成される論理インターフェイスです。これらのソフトウェア定義のインターフェイスにより、単一のハードウェア インターフェイス上でトラフィックを論理チャネルに分割することができ、また、物理インターフェイス上で帯域幅を効率的に利用することができます。

サブインターフェイスは、インターフェイス名の末尾に拡張を追加することで、他のインターフェイスと区別されます。たとえば、物理インターフェイス TenGigE 0/1/0/0 上のイーサネット サブインターフェイス 23 は、TenGigE 0/1/0/0.23 となります。

サブインターフェイスがトラフィックを渡すことができるようにするには、有効なタグ付きプロトコルのカプセル化と VLAN 識別子の割り当てが必要です。すべてのイーサネット サブインターフェイスは常に、デフォルトで 802.1Q VLAN でカプセル化されます。ただし、VLAN 識別子は明示的に定義する必要があります。

サブインターフェイス MTU

サブインターフェイスの最大伝送単位（MTU）は、物理インターフェイスから継承されます。これには、802.1Q VLAN タグに許可されている追加の 4 バイトも含まれます。

イーサネット バンドルでの VLAN サブインターフェイス

イーサネット バンドルは、1 つ以上のイーサネット ポートのグループを集約し、1 つのリンクとして扱うようにしたものです。単一のイーサネット バンドルに複数の VLAN サブインターフェイスを追加することができます。

イーサネット バンドルの設定方法については、このマニュアルの「リンク バンドルの設定」モジュールを参照してください。イーサネット バンドルに VLAN サブインターフェイスを作成する手順は、物理イーサネット インターフェイスに VLAN サブインターフェイスを作成する手順とまったく同じです。

イーサネット バンドルに VLAN サブインターフェイスを作成するには、このモジュールで後述する「802.1Q VLAN インターフェイスの設定」（P.47）を参照してください。

VLAN インターフェイスでのレイヤ 2 VPN

レイヤ 2 バーチャル プライベート ネットワーク（L2VPN）機能によって、サービス プロバイダー（SP）は地理的に離れたカスタマー サイトにも L2 サービスを提供できるようになります。詳細は、このマニュアルで前述した「イーサネット インターフェイスの設定」（P.37）モジュールにある「イーサネット インターフェイスでのレイヤ 2 VPN」（P.23）の項を参照してください。

VLAN 接続回線（AC）を設定するための設定モデルは、基本の VLAN の設定に使用するモデルに類似しています。ユーザはまず VLAN サブインターフェイスを作成し、次にサブインターフェイス コンフィギュレーション モードで VLAN を設定します。接続回路を作成するには、**interface** コマンド文字列に **l2transport** キーワードを含めて、そのインターフェイスが L2 インターフェイスであることを指定する必要があります。

VLAN AC は、L2VPN 操作の 3 つのモードをサポートします。

- 基本の Dot1Q 接続回線：この接続回線は、特定の VLAN タグで送受信されるすべてのフレームに対応します。
- QinQ 接続回線：この接続回線は、特定の外部 VLAN タグおよび特定の内部 VLAN タグで送受信されるすべてのフレームに対応します。QinQ は、2 つのタグのスタックを使用する Dot1Q の拡張です。
- Q-in-Any 接続回線：この接続回線は、内部 VLAN タグが L3 終端でない限り、特定の外部 VLAN タグおよび任意の内部 VLAN タグで送受信されるすべてのフレームに対応します。Q-in-Any は、ワイルドカード化を使用して任意の 2 番目のタグに一致させる QinQ の拡張です。

（注） Q-in-Any モードは、基本の Dot1Q モードを変化させたものです。Q-in-Any モードではフレームは基本の QinQ カプセル化が行われていますが、Q-in-Any モードでは内部タグは無関係です。ただし、いくつかの特定の内部 VLAN タグが特定のサービス用に使用される場合を除きます。たとえば、一般的なインターネット アクセスに L3 サービスを提供するために、あるタグが使用されることがあります。

CE-to-PE リンクの各 VLAN は、（VC タイプ 4 または VC タイプ 5 を使用する）独立した L2VPN 接続として設定できます。VLAN に L2VPN を設定するには、「802.1Q VLAN サブインターフェイスの削除」（P.52）を参照してください。

VLAN に L2VPN を設定する場合は、次の事項に注意する必要があります。

- Cisco IOS XR ソフトウェアは、ラインカードごとに最大 4000 の接続回線をサポートしています。
- ポイントツーポイント接続では、2 つの接続回線を同じタイプにするべきではありません。たとえば、ポート モードのイーサネット接続回線は Dot1Q イーサネット接続回線に接続できます。
- 疑似回線は、VLAN モードまたはポート モードで実行できます。VLAN モードで実行される疑似回線に単一の Dot1Q タグを設定することができますが、ポート モードで実行される疑似回線にタグを設定することはできません。これらの異なるタイプの回路を接続するには、インターワーキングが必要です。この場合のインターワーキングは、タグのポップ、プッシュ、書き換えの形を取ります。L2VPN を使用するメリットは、まったく異なるタイプのメディアを接続するのに必要なインターワーキングを簡素化できることにあります。
- MPLS 疑似回線の両側にある接続回線は異なるタイプでもかまいません。この場合、接続回線の一方または両方のエンドで、疑似回線を行うための適切な変換が行われます。

接続回線と疑似回線の情報を表示するには、**show interfaces** コマンドを使用します。

（注） **show interfaces** コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Command Reference*』を参照してください。

イーサネット インターフェイスでのレイヤ 2 機能の設定方法

この項では、次の作業について説明します。

- 「ギガビット イーサネットおよび 10 ギガビット イーサネットのデフォルト設定値」（P.35）
- 「イーサネット インターフェイスの設定」（P.37）
- 「ギガビット イーサネット インターフェイスの設定」（P.39）
- 「イーサネット ポートでの接続回路の設定」（P.42）
- 「EFP 出力フィルタリングの設定」（P.45）
- 「802.1Q VLAN インターフェイスの設定」（P.47）

（注） インターフェイスの設定に関する詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』を参照してください。

ギガビット イーサネットおよび 10 ギガビット イーサネットのデフォルト設定値

表 2 は、ギガビット イーサネットまたは 10 ギガビット イーサネットのモジュラ サービス カードおよび PC の脅威対策 PLIM でインターフェイスをイネーブルにしたときに表示される、デフォルトのインターフェイス設定パラメータを示します。

（注） インターフェイスを管理上のダウン状態にするには、**shutdown** コマンドを使用する必要があります。インターフェイスのデフォルトは **no shutdown** です。ルータにモジュラ サービス カードを初めて挿入したときに、プリコンフィギュレーションが行われていない場合、設定マネージャによって shutdown 項目が設定に追加されます。この shutdown を削除できるのは、**no shutdown** コマンドを入力している場合のみです。

表 2　ギガビット イーサネットおよび 10 ギガビット イーサネット モジュラ サービス カードのデフォルト設定値

パラメータ	設定ファイルのエントリ	デフォルト値	制約事項[1]
フロー制御	**flow-control**	出力オン 入力オフ	なし
MTU	**mtu**	1514 バイト（通常のフレーム） 1518 バイト（802.1Q タグ付きフレーム） 1522 バイト（QinQ フレーム）	なし
MAC アドレス	**mac address**	ハードウェア バーンドイン アドレス（BIA[2]）	L3 のみ
L2 ポート	**l2transport**	off/L3	L2 サブインターフェイスには L3 メイン親インターフェイスが必要です。
出力フィルタリング	**Ethernet egress-filter**	off	なし
リンク ネゴシエーション	**negotiation**	off	物理メイン インターフェイスのみ
Tunneling Ethertype	**tunneling ethertype**	0X8100	メイン インターフェイスのみで設定されます。サブインターフェイスのみに適用されます。
VLAN タグの一致	**encapsulation**	メイン インターフェイスではすべてのフレーム。サブインターフェイスでは指定されたフレームのみ	encapsulation コマンドはサブインターフェイスのみ

1. 制約事項は L2 メイン インターフェイス、L2 サブインターフェイス、L3 メイン インターフェイス、インターフレックス L2 インターフェイスなどに適用されます。
2. 組み込みのアドレス

1. 組み込みのアドレス

イーサネット インターフェイスの設定

この項では、次の作業について説明します。

- 10 ギガビット イーサネット インターフェイスの設定
- ギガビット イーサネット インターフェイスの設定

イーサネット インターフェイスの設定については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』を参照してください。

10 ギガビット イーサネット インターフェイスの設定

イーサネット インターフェイスを設定するには、次の作業を行います。

手順の概要

1. **configure interface TenGigE [***instance***]**
2. **l2transport**
3. **mtu** *bytes*
4. **no shutdown**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure interface TenGigE [*instance*]** **例：** `RP/0/RSP0/CPU0:router# configure` `RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/0/0/1`	10 ギガビット イーサネット インターフェイスのインターフェイス コンフィギュレーション モードを開始します。
ステップ 2	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-if)#l2transport`	ポートでレイヤ 2 トランスポート モードをイネーブルにし、レイヤ 2 トランスポート コンフィギュレーション モードを開始します。
ステップ 3	**mtu** *bytes* **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# mtu 1448`	ブリッジ ドメインの最大パケット サイズまたは最大伝送単位（MTU）サイズを調整します。 • バイト単位で MTU サイズを指定するには、bytes 引数を使用します。範囲は 64 ～ 65535 です。
ステップ 4	**no shutdown** **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# no shutdown`	shutdown 設定を削除します。こうすることでインターフェイスが強制的に管理上のダウン状態になります。

	コマンドまたはアクション	目的
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# end` または `RP/0/RSP0/CPU0:router(config-if-l2)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ギガビット イーサネット インターフェイスの設定

基本的なギガビット イーサネットまたは 10 ギガビット イーサネット インターフェイスを設定するには、次の作業を行います。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **ipv4 address** *ip-address mask*
4. **flow-control** {**bidirectional | egress | ingress**}
5. **mtu** *bytes*
6. **mac-address** *value1.value2.value3*
7. **negotiation auto**（ギガビット イーサネット インターフェイスのみ）
8. **no shutdown**
9. **end**
   または
   commit
10. **show interfaces** [**GigabitEthernet** | **TenGigE**] *instance*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/0`	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、イーサネット インターフェイス名と *rack/slot/module/port* 表記を指定します。

	コマンドまたはアクション	目的
ステップ 3	**ipv4 address** *ip-address mask* **例：** `RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224`	IP アドレスとサブネット マスクをインターフェイスに割り当てます。 • *ip-address* をインターフェイスのプライマリ IPv4 アドレスに置き換えます。 • *mask* を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクは、次のいずれかの方法で指定できます。 – 4 分割ドット付き 10 進表記のアドレスでネットワーク マスクを指定します。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワーク アドレスに属することを示します。 – スラッシュ（/）と数字による表記。たとえば、/8 は、マスクの最初の 8 ビットが 1 で、対応するアドレスのビットがネットワーク アドレスであることを示します。
ステップ 4	**flow-control** {**bidirectional**｜ **egress** ｜ **ingress**} **例：** `RP/0/RSP0/CPU0:router(config-if)# flow control ingress`	（任意）フロー制御のポーズ フレームの送信および処理をイネーブルにします。 • **egress**：出力でフロー制御のポーズ フレームの送信をイネーブルにします。 • **ingress**：入力で受信したポーズ フレームの処理をイネーブルにします。 • **bidirectional**：出力でフロー制御のポーズ フレームの送信をイネーブルにし、入力で受信したポーズ フレームの処理をイネーブルにします。
ステップ 5	**mtu** *bytes* **例：** `RP/0/RSP0/CPU0:router(config-if)# mtu 1448`	（任意）インターフェイスの MTU 値を設定します。 • 通常フレームのデフォルトは 1514 バイト、802.1Q タグ付きフレームのデフォルトは 1518 バイトです。 • ギガビット イーサネットおよび 10 ギガビット イーサネットの mtu 値の範囲は 64 ～ 65535 バイトです。
ステップ 6	**mac-address** *value1.value2.value3* **例：** `RP/0/RSP0/CPU0:router(config-if)# mac address 0001.2468.ABCD`	（任意）[Management Ethernet] インターフェイスの MAC 層アドレスを設定します。 • 値は、それぞれ MAC アドレスの上位、中間、および下位の 2 バイト（16 進）です。各 2 バイト値の範囲は 0 ～ ffff です。
ステップ 7	**negotiation auto** **例：** `RP/0/RSP0/CPU0:router(config-if)# negotiation auto`	（任意）ギガビット イーサネット インターフェイスのオートネゴシエーションをイネーブルにします。 • オートネゴシエーションは接続の両エンドで明示的にイネーブルにするか、接続の両エンドで速度とデュプレックス設定を手動設定する必要があります。 • オートネゴシエーションがイネーブルの場合、手動で設定した速度またはデュプレックス モードの設定の方が優先されます。 **（注）** **negotiation auto** コマンドは、ギガビット イーサネット インターフェイスだけで使用できます。

	コマンドまたはアクション	目的
ステップ 8	**no shutdown** **例：** `RP/0/RSP0/CPU0:router(config-if)# no shutdown`	shutdown 設定を削除します。こうすることでインターフェイスが強制的に管理上のダウン状態になります。
ステップ 9	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 10	**show interfaces** [**GigabitEthernet** | **TenGigE**] *instance* **例：** `RP/0/RSP0/CPU0:router# show interfaces TenGigE 0/3/0/0`	（任意）ルータ上のインターフェイスに関する統計情報を表示します。

次の作業

- イーサネット インターフェイスで 802.1Q VLAN サブインターフェイスを設定する方法については、このマニュアルで後述する「Cisco ASR 9000 シリーズ ルータ キャリア イーサネット モデル」モジュールを参照してください。
- L2VPN 実装のイーサネット ポートで AC を設定する方法については、このモジュールで後述する「イーサネット ポートでの接続回路の設定」を参照してください。

イーサネット ポートでの接続回路の設定

ギガビット イーサネットまたは 10 ギガビット イーサネット ポートで接続回路を設定するには、次の手順を実行します。接続回線の設定の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』を参照してください。

（注） この手順の各操作では、EFP モードで操作する L2VPN イーサネット ポートを設定します。

手順の概要

1. **configure**
2. **interface** [**GigabitEthernet** | **TenGigE**] *instance.subinterface* **l2transport**
3. **encapsulation dot1q** *vlan-id*
4. **interface** [**GigabitEthernet** | **TenGigE**] *instance.subinterface* **l2transport**
5. **encapsulation dot1q** *vlan-id*
6. **l2vpn**
7. **bridge group** *group-name*
8. **bridge-domain** *domain-name*
9. **interface** [**GigabitEthernet** | **TenGigE**] *instance.subinterface*
10. **interface** [**GigabitEthernet** | **TenGigE**] *instance.subinterface*
11. **end**
    または
    commit
12. **show run interface** [**GigabitEthernet** | **TenGigE**] *instance.subinterface*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface [GigabitEthernet | TenGigE]** *instance.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/5/0/0.20 l2transport`	サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。 • instance 引数を次のインスタンスのいずれかに置換します。 – 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は rack/slot/module/port の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ～ 65535 です。 • subinterface 引数をサブインターフェイスの値に置き換えます。範囲は 0 ～ 4095 です。 • 名前の表記は instance.subinterface の形式で、表記の一部として引数をピリオドで区切る必要があります。
ステップ 3	**encapsulation dot1q** *vlan-id* **例：** `RP/0/RSP0/CPU0:router(config-subif)#encapsulation dot1q 50`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。
ステップ 4	**interface [GigabitEthernet | TenGigE]** *instance.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/5/0/0.20 l2transport`	サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。 • instance 引数を次のインスタンスのいずれかに置換します。 – 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は rack/slot/module/port の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ～ 65535 です。 • subinterface 引数をサブインターフェイスの値に置き換えます。範囲は 0 ～ 4095 です。 • 名前の表記は instance.subinterface の形式で、表記の一部として引数をピリオドで区切る必要があります。

	コマンドまたはアクション	目的
ステップ 5	**encapsulation dot1q** *vlan-id* **例：** `RP/0/RSP0/CPU0:router(config-subif)#encapsulation dot1q 50`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。
ステップ 6	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config-subif)#l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 7	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group ce-doc-examples`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 8	**bridge-domain domain-name** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain ac-example`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 9	**interface [GigabitEthernet | TenGigE] instance.subinterface** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	ブリッジ ドメインにインターフェイスを追加し、パケットの転送と、同じブリッジ ドメイン内の他のインターフェイスからのパケットの受信を可能にします。これで、インターフェイス EFP は、このブリッジ ドメイン上の接続回線になります。
ステップ 10	**interface [GigabitEthernet | TenGigE] instance.subinterface** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#interface GigabitEthernet0/5/0/1.15`	ブリッジ ドメインにインターフェイスを追加し、パケットの転送と、同じブリッジ ドメイン内の他のインターフェイスからのパケットの受信を可能にします。これで、インターフェイス EFP は、このブリッジ ドメイン上の接続回線になります。

	コマンドまたはアクション	目的
ステップ 11	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 12	**show run interface** [**GigabitEthernet** | **TenGigE**] *instance.subinterface* **例：** `RP/0/RSP0/CPU0:router#show run interface GigabitEthernet0/5/0/1.15`	（任意）ルータのサブインターフェイスの統計情報を表示します。

EFP 出力フィルタリングの設定

ここでは、Cisco ASR 9000 シリーズ ルータで EFP 出力フィルタリング機能を設定する手順について説明します。

EFP 出力フィルタリングは L2 サブインターフェイス固有の機能で、出力方向でサブインターフェイスカプセル化フィルタリングがどのように実行されるかを厳密に制御します。EFP の動作とモデルに従い、サブインターフェイスから送信されるすべてのパケットは、同じパケットがサブインターフェイスで受信される場合には、サブインターフェイスのカプセル化または書き換えの条件に一致する必要があります（送信元 MAC アドレスと宛先 MAC アドレスは交換されます）。

EFP 出力フィルタリングには 2 つの段階があります。第 1 段階では rewrite コマンドは使用されず、第 2 段階では rewrite コマンドが使用されます。

第 1 段階のフィルタリングでは、パケットはカプセル化と照合され、一致するかどうか確認されます。これは、パケットをその EFP に転送するかどうか判別するために入力でパケットをチェックするのと同じ方法です。

第 2 段階のフィルタリングでは、出力の書き換え前の状態のパケットが正しいことを確認するために、出力の書き換えが実行される前にパケットがチェックされます。これは、出力パケットの VLAN カプセル化が、入力書き換え後の仮想の入力パケットと同一である必要があることを意味します。

書き換えと EFP 出力フィルタリングの両方がインターフェイスに設定されており、EFP 出力フィルタリングが原因で、出力トラフィックが予期せずにドロップされる場合、ユーザはドロップがどの段階で発生するか最初に確認する必要があります。

（注） 出力ドロップ カウンタにより、そのインターフェイスの「show interface」表示で、出力 EFP フィルタリングが原因で発生したドロップが表示されます。出力ドロップ カウンタは、複数の原因によるドロップの合計であり、EFP 出力フィルタリングが必ずしも原因ではありません。

ethernet egress-filter コマンドを使用することで、グローバルまたは L2 サブインターフェイス モードで EFP 出力フィルタリングを設定できます。

- **ethernet egress-filter strict** は、グローバル コンフィギュレーション モードで EFP 出力フィルタリングを設定します。
- **ethernet egress-filter {strict | disabled}** は、L2 サブインターフェイス モードで EFP 出力フィルタリングを設定します。

手順の概要

1. **configure**
2. **ethernet egress-filter strict**
3. **interface {GigabitEthernet | TenGigE | FastEthernet | Bundle-Ether}** instance.subinterface
4. **ethernet egress-filter {strict | disabled}**
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** RP/0/RSP0/CPU0:PE44_ASR-9010# config Thu Jun 4 07:50:02.660 PST RP/0/RSP0/CPU0:PE44_ASR-9010(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ethernet egress-filter strict** **例：** RP/0/RSP0/CPU0:PE44_ASR-9010(config)# ethernet egress-filter strict	デバイス上のすべてのサブインターフェイスに対して厳密な出力フィルタリングをデフォルトでイネーブルにします。
ステップ 3	**interface {GigabitEthernet | TenGigE | FastEthernet | Bundle-Ether}** instance.subinterface **例：** RP/0/RSP0/CPU0:PE44_ASR-9010(config)# interface GigabitEthernet 0/1/0/1.1 RP/0/RSP0/CPU0:PE44_ASR-9010(config-subif)#	L2 サブインターフェイスを作成します。

	コマンドまたはアクション	目的
ステップ 4	**ethernet egress-filter {strict | disabled}** **例：** RP/0/RSP0/CPU0:PE44_ASR-9010(config-subif)# ethernet egress-filter strict	L2 サブインターフェイスに対し出力フィルタリングを明示的にイネーブルまたはディセーブルにすることができます。また、グローバル設定を上書きするために使用できます。
ステップ 5	**exit** **例：** RP/0/RSP0/CPU0:PE44_ASR-9010(config-subif)# exit RP/0/RSP0/CPU0:PE44_ASR-9010(config)# exit	コンフィギュレーション モードを終了します。

802.1Q VLAN インターフェイスの設定

この項では、次の手順について説明します。

- 「802.1Q VLAN サブインターフェイスの設定」（P.47）
- 「ネイティブ VLAN の設定」（P.49）
- 「802.1Q VLAN サブインターフェイスの削除」（P.52）
- 「802.1Q VLAN サブインターフェイスの削除」（P.52）

802.1Q VLAN サブインターフェイスの設定

ここでは、802.1Q VLAN サブインターフェイスの設定手順について説明します。これらのサブインターフェイスを削除するには、このモジュールの「802.1Q VLAN サブインターフェイスの削除」を参照してください。

手順の概要

1. **configure**
2. **interface** {**GigabitEthernet** | **TenGigE** | **Bundle-Ether**} *instance.subinterface*
3. **l2transport**
4. **encapsulation dot1q** *vlan-id*
5. **ethernet egress-filter strict**
6. **end**
   または
   commit
7. **show ethernet trunk bundle-ether** *instance*（任意）

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** {**GigabitEthernet** | **TenGigE** | **Bundle-Ether**} *instance.subinterface* **例：** `RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.10`	サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。 • *instance* 引数を次のインスタンスのいずれかに置換します。 – 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は *rack*/*slot*/*module*/*port* の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ～ 65535 です。 • *subinterface* 引数を、サブインターフェイス値に置き換えます。範囲は 0 ～ 4095 です。 • 名前の表記は *instance.subinterface* の形式で、表記の一部として引数をピリオドで区切る必要があります。
ステップ 3	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-subif)#`**`l2transport`**	ポートでレイヤ 2 トランスポート モードをイネーブルにし、レイヤ 2 トランスポート コンフィギュレーション モードを開始します。
ステップ 4	**encapsulation dot1q** *vlan-id* **例：** `RP/0/RSP0/CPU0:router(config-subif-l2)# encapsulation dot1q 100`	VLAN 接続回線をサブインターフェイスに割り当てます。 • vlan-id 引数にはサブインターフェイス ID を指定します。範囲は 1 ～ 4094 です（0 と 4095 は予約されています）。基本の Dot1Q 接続回線を設定するには、次の構文を使用します。 **encapsulation dot1q** *vlan-id* • QinQ 接続回線を設定するには、次の構文を使用します。 **encapsulation dot1q** *vlan-id* **second-dot1q** *vlan-id* **（注）** 以下は、各種の **encapsulation** コマンドです。 – encapsulation dot1q 100 – encapsulation dot1q 100 second-dot1q 101 – encapsulation dot1ad 200 dot1q 201

	コマンドまたはアクション	目的
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config)# end` または `RP/0/RSP0/CPU0:router(config)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 6	**show ethernet trunk bundle-ether** *instance* **例：** `RP/0/RSP0/CPU0:router# show ethernet trunk bundle-ether 5`	（任意）インターフェイス コンフィギュレーションを表示します。 イーサネット バンドル インスタンスの範囲は 1 ～ 65535 です。

ネイティブ VLAN の設定

ここでは、インターフェイスにネイティブ VLAN を設定する方法について説明します。

手順の概要

1. **configure**
2. **interface [GigabitEthernet | TenGigE | Bundle-Ether]** *instance.subinterface* **l2transport**
3. **encapsulation dot1q** <*vlan-id*>, **untagged**
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** [**GigabitEthernet** | **TenGigE** | **Bundle-Ether**] *instance.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/2/0/4.2 l2transport`	サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。 • instance 引数を次のインスタンスのいずれかに置換します。 – 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は rack/slot/module/port の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ～ 65535 です。 • subinterface 引数をサブインターフェイスの値に置き換えます。範囲は 0 ～ 4095 です。 • 名前の表記は instance.subinterface の形式で、表記の一部として引数をピリオドで区切る必要があります。 **（注）** コマンド文字列に l2transport キーワードを含める必要があります。そうしないと、接続回線ではなく、レイヤ 3 サブインターフェイスが作成されます。

	コマンドまたはアクション	目的
ステップ 3	**encapsulation** [**dot1q** *vlan-id*, **untagged**] **例：** `RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 400`	802.1Q トランク インターフェイスに関連付けられた、ネイティブの VLAN を定義します。 • *vlan-id* 引数は、サブインターフェイスの ID です。 • 範囲は 1 ～ 4094 です（0 と 4095 は予約されています）。 **untagged** キーワードを指定した **encapsulation** コマンドを発行することで、dot1q 400 とタグなしフレームの両方を受信できます。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# end` または `RP/0/RSP0/CPU0:router(config-subif)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

802.1Q VLAN サブインターフェイスの削除

ここでは、このモジュールの「802.1Q VLAN サブインターフェイスの設定」で設定した 802.1Q VLAN サブインターフェイスを削除する手順について説明します。

手順の概要

1. **configure**
2. **no interface** {**GigabitEthernet** | **TenGigE** | **Bundle-Ether**} *instance.subinterface*
3. ステップ 2 を繰り返し、その他の VLAN サブインターフェイスを削除します。
4. **end**
   または
   commit
5. **show ethernet trunk bundle-ether** *instance*（任意）

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**no interface** [**GigabitEthernet** | **TenGigE** | **Bundle-Ether**] *instance.subinterface* **例：** `RP/0/RSP0/CPU0:router(config)# no interface TenGigE 0/2/0/4.10`	サブインターフェイスを削除すると、そのサブインターフェイスに適用されているすべての設定も自動的に削除されます。 • *instance* 引数を次のインスタンスのいずれかに置換します。 – 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は *rack*/*slot*/*module*/*port* の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ～ 65535 です。 • *subinterface* 引数を、サブインターフェイス値に置き換えます。範囲は 0 ～ 4095 です。 名前の表記は *instance.subinterface* の形式で、表記の一部として引数をピリオドで区切る必要があります。

	コマンドまたはアクション	目的
ステップ 3	ステップ 2 を繰り返し、その他の VLAN サブインターフェイスを削除します。	—
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config)# end` または `RP/0/RSP0/CPU0:router(config)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 5	**show ethernet trunk bundle-ether** *instance* **例：** `RP/0/RSP0/CPU0:router# show ethernet trunk bundle-ether 5`	（任意）インターフェイス コンフィギュレーションを表示します。 イーサネット バンドル インスタンスの範囲は 1 ～ 65535 です。

設定例

ここでは、次の設定例を示します。

- イーサネット インターフェイスの設定：例
- L2VPN AC の設定：例
- VPWS へのリンク バンドルの設定：例
- イーサネット バンドルへの L2 および L3 サービスの設定：例
- VLAN サブインターフェイスの設定：例

イーサネット インターフェイスの設定：例

次に、10 ギガビット イーサネットのモジュラ サービス カードのインターフェイスを設定する例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/0/0/1
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if)# mtu 1448
RP/0/RSP0/CPU0:router(config-if)# no shutdown
RP/0/RSP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/RSP0/CPU0:router# show interfaces TenGigE 0/0/0/1

TenGigE0/0/0/1 is down, line protocol is down
  Hardware is TenGigE, address is 0001.2468.abcd (bia 0001.81a1.6b23)
  Internet address is 172.18.189.38/27
  MTU 1448 bytes, BW 10000000 Kbit
     reliability 0/255, txload Unknown, rxload Unknown
  Encapsulation ARPA,
  Full-duplex, 10000Mb/s, LR
  output flow control is on, input flow control is on
  loopback not set
  ARP type ARPA, ARP timeout 01:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 total input drops
     0 drops for unrecognized upper-level protocol
     Received 0 broadcast packets, 0 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 total output drops
     Output 0 broadcast packets, 0 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

L2VPN AC の設定 : 例

次に、イーサネット インターフェイスで L2VPN AC を設定する例を示します。

```
RP/0/RSP0/CPU0:router#configure
RP/0/RSP0/CPU0:router(config)#interface gigabitethernet 0/5/0/0.2 l2transport
RP/0/RSP0/CPU0:router(config-subif)#encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)#ethernet egress-filter strict
RP/0/RSP0/CPU0:router(config-subif)#l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)#clear

RP/0/RSP0/CPU0:router#configure
RP/0/RSP0/CPU0:router(config)#interface gigabitethernet 0/5/0/0.2 l2transport
RP/0/RSP0/CPU0:router(config-subif)#encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)#ethernet egress-filter strict
RP/0/RSP0/CPU0:router(config-subif)#interface gigabitethernet 0/5/0/1.100 l2transport
RP/0/RSP0/CPU0:router(config-subif)#encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)#ethernet egress-filter strict
RP/0/RSP0/CPU0:router(config-subif)#l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group example
RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain mybridge
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface gigabitethernet 0/5/0/0.2
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#interface gigabitethernet 0/5/0/1.100
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg)#exit
RP/0/RSP0/CPU0:router(config-l2vpn)#exit
RP/0/RSP0/CPU0:router(config)#show

Building configuration...
!! IOS XR Configuration 0.0.0
interface GigabitEthernet0/5/0/0.2 l2transport
 encapsulation dot1q 100
 ethernet egress-filter strict
!
interface GigabitEthernet0/5/0/1.100 l2transport
 encapsulation dot1q 100
 ethernet egress-filter strict
!
l2vpn
 bridge group example
  bridge-domain mybridge
   interface GigabitEthernet0/5/0/0.2
   !
   interface GigabitEthernet0/5/0/1.100
   !
  !
 !
end
```

VPWS へのリンク バンドルの設定：例

物理インターフェイス（ポート モード）

```
interface Bundle-Ether12
 l2transport
!
interface GigabitEthernet0/1/0/10
 negotiation auto
 l2transport
!
interface GigabitEthernet0/1/0/20
 bundle id 12 mode on
 negotiation auto
!
interface GigabitEthernet0/1/0/21
 bundle id 12 mode on
 negotiation auto
!
 !
l2vpn
 xconnect group test
  p2p test
   interface Bundle-Ether12
   !
   interface GigabitEthernet0/1/0/10
   !
  !
 !
!
```

サブインターフェイス（EFP モード）

```
interface Bundle-Ether12
!
interface Bundle-Ether12.1 l2transport
  encapsulation dot1q 12
 !
!
interface GigabitEthernet0/1/0/10
 negotiation auto
!
interface GigabitEthernet0/1/0/10.1  l2transport
  encapsulation dot1q 12
 !

!
interface GigabitEthernet0/1/0/20
 bundle id 12 mode on
 negotiation auto
!
interface GigabitEthernet0/1/0/21
 bundle id 12 mode on
 negotiation auto
!
 !
l2vpn
```

```
 xconnect group test
  p2p test
   interface Bundle-Ether12.1
   !
   interface GigabitEthernet0/1/0/10.1
   !
  !
 !
!
```

イーサネット バンドルへの L2 および L3 サービスの設定：例

次に、イーサネット バンドル インターフェイスに L3 サービスを設定する例を示します。

```
configure
interface Bundle-Ether 100
 ipv4 address 12.12.12.2 255.255.255.0

!
```

次に、イーサネット バンドル サブインターフェイスに L3 サービスを設定する例を示します。

```
configure
interface Bundle-Ether 100.1
 ipv4 address 13.13.13.2 255.255.255.0

!
```

次に、イーサネット バンドル インターフェイスに L2 サービスを設定する例を示します。

```
configure
 interface Bundle-Ether 101
 l2transport

!
```

次に、イーサネット バンドル インターフェイスに L2 サービスを設定する例を示します。

```
configure
 interface Bundle-Ether1.1 l2transport
!
```

VLAN サブインターフェイスの設定：例

次に、VLAN サブ インターフェイスを作成する例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 20
RP/0/RSP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.2 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 30
RP/0/RSP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.3 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 40
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

次に、イーサネット バンドルに 2 つの VLAN サブインターフェイスを一度に作成する例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether 1 l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# exit
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether  1.1 l2transport
RP/0/RSP0/CPU0:router(config-subif-l2)#  encapsulation dot1q 10
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether 1.2 l2transport
RP/0/RSP0/CPU0:router(config-subif-l2)# encapsulation dot1q 20
RP/0/RSP0/CPU0:router(config-subif)# exit
```

次に、基本の Dot1Q 接続回線を作成する例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 20
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

次に、QinQ 接続回線を作成する例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.2 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 20 second-dot1q 10
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

次に、Q-in-Any 接続回線を作成する例を示します。

```
RP/0/RSP/CPU0:router# configure
RP/0/RSP/CPU0:router(config)# interface TenGigE 0/2/0/4.3 l2transport
RP/0/RSP/CPU0:router(config-subif)# encapsulation dot1q 30 second-dot1q any
RP/0/RSP/CPU0:router(config-subif)# commit
RP/0/RSP/CPU0:router(config-subif)# exit
RP/0/RSP/CPU0:router(config)# exit
```

次の作業

イーサネット インターフェイスの設定が完了したら、イーサネット インターフェイスで各 VLAN サブインターフェイスを設定できます。VLAN サブインターフェイスの設定方法については、このマニュアルで後述する「Cisco ASR 9000 シリーズ ルータ キャリア イーサネット モデル」モジュールを参照してください。

IPv6 については、『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Debug Command Reference*』の「*Implementing Access Lists and Prefix Lists*」モジュールを参照してください。

その他の関連資料

ここでは、ギガビットおよび 10 ギガビット イーサネット インターフェイスの実装に関する参考資料を紹介します。

関連資料

関連項目	参照先
Cisco IOS XR マスター コマンド リファレンス	*『Cisco IOS XR Master Commands List』*

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して、選択されたプラットフォームに対応する MIB を検索およびダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

イーサネット機能

このモジュールでは、Cisco IOS XR ソフトウェアをサポートする Cisco ASR 9000 シリーズ アグリゲーション サービス ルータでのレイヤ 2（L2）イーサネット機能の設定方法について説明します。

イーサネット インターフェイスの設定の詳細については、このコンフィギュレーション ガイドの「Cisco ASR 9000 シリーズ ルータ キャリア イーサネット モデル」モジュールを参照してください。

Cisco ASR 9000 シリーズ ルータ のイーサネット インターフェイス設定の機能履歴

リリース	変更内容
リリース 3.9.1	ポリシー ベースの転送およびレイヤ 2 プロトコル トンネリング機能のサポートが追加されました。

内容

イーサネット機能を実装するための前提条件

このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

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

イーサネットの機能の実装に関する情報

10 ギガビット イーサネット インターフェイスを設定するには、次の概念を理解しておく必要があります。

- 「ポリシー ベースの転送」（P.62）
- 「レイヤ 2 プロトコル トンネリング」（P.62）

ポリシー ベースの転送

Cisco ASR 9000 シリーズ ルータでは、単一の MAC アドレスを、ポートの設定済みの VLAN とは異なる VLAN にマップできます。2 つの異なる EFP に入るトラフィックを分離するためには、送信元 VLAN タグおよび送信元 MAC アドレスを使用して EFP を定義する必要があります。

レイヤ 2 プロトコル トンネリング

レイヤ 2 プロトコル トンネリング（L2PT）は、レイヤ 2（L2）スイッチング ドメイン間でイーサネット プロトコル フレームをトンネリングするための、シスコ独自のプロトコルです。

L2 プロトコル フレームが L2 スイッチング デバイスのインターフェイスに着信すると、スイッチまたはルータはフレームで次のいずれかのアクションを実行します。

- 転送：フレームは例外的な処理なしでスイッチングまたはルーティングされます。
- ドロップ：フレームはルータで廃棄されます。
- 終端：ルータは、フレームが L2 プロトコル フレームであると認識し、プロトコル処理のためにこれをルータのコントロール プレーンに送信します。
- トンネリング：ルータは、フレームをカプセル化して、プロトコル フレームとしてのアイデンティティを非表示にします。これにより、フレームが別のルータで終端することを防ぎます。トンネルの反対側ではカプセル化を解除して、フレームを元の状態に戻します。

L2PT の機能

Cisco ASR 9000 シリーズ ルータ は、次の機能を提供します。

- 次のプロトコルをトンネリングします。
  - Cisco Discovery Protocol（CDP）
  - スパニングツリー プロトコル（STP およびそのバリエーション）
  - 仮想トランキング プロトコル（VTP）
- 次のトンネリング モードをサポートします。
  - 進む
  - 反転
- L2PT は VLAN ヘッダーを持つプロトコル フレームをカプセル化し、カプセル化を解除します。
- 巨大フレーム レートの処理機能をサポートします。Cisco ASR 9000 シリーズ ルータは、インターフェイス ライン レートで L2PT カプセル化とカプセル化解除を実行します。

（注） 専用の L2PT カウンタはありません。QoS またはその他のパラメータの L2PT 特定の調整はありません。

転送モードの L2PT

図 1 に、転送モードで設定された L2PT を示します。

図 1 転送モードの L2PT

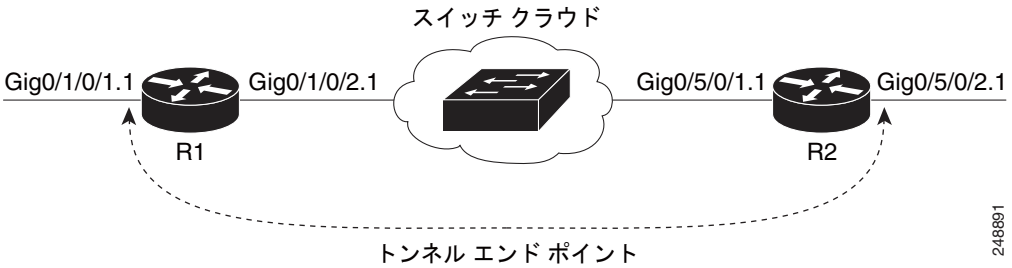


図 1 では、サービス プロバイダー ネットワーク（S ネットワーク）について説明します。カスタマー ネットワーク（C ネットワーク）は、GigabitEthernet サブインターフェイス 0/1/0/1.1 でルータ R1 に接続し、GigabitEthernet サブインターフェイス 0/5/0/2.1 でルータ R2 に接続します。C ネットワークは図に示されていません。ただし、C ネットワークは、S ネットワーク経由で L2 トラフィックを送信し、S ネットワークはエンド ツー エンドでトラフィックを切り替えます。カスタマー トラフィックは、L2 プロトコル フレームを伝送します。L2PT の目的は、これらのプロトコル フレームが S ネットワークを通過できるようにすることです。転送モードでは、L2PT は、S ネットワークのカスタマー側インターフェイスである R1 GigabitEthernet 0/1/0/1.1 と R2 GigabitEthernet 0/5/0/2.1 に適用されます。

図 1 は、転送モードの L2PT の設定を示します。

R1：

```
!
interface GigabitEthernet0/1/0/1
 negotiation auto
!
interface GigabitEthernet0/1/0/1.1 l2transport
 encapsulation default
 l2protocol cpsv tunnel
!
interface GigabitEthernet0/1/0/2
 negotiation auto
!
interface GigabitEthernet0/1/0/2.1 l2transport
 encapsulation default
!
l2vpn
 xconnect group examples
  p2p r1-connect
   interface GigabitEthernet0/1/0/1.1
   interface GigabitEthernet0/1/0/2.1
  !
 !
!
```

R2：

```
!
interface GigabitEthernet0/5/0/1
 negotiation auto
!
interface GigabitEthernet0/5/0/1.1 l2transport
 encapsulation default
!
interface GigabitEthernet0/5/0/2
 negotiation auto
!
interface GigabitEthernet0/5/0/2.1 l2transport
 encapsulation default
 l2protocol cpsv tunnel
!
l2vpn
 xconnect group examples
  p2p r2-connect
   interface GigabitEthernet0/5/0/1.1
   interface GigabitEthernet0/5/0/2.1
  !
 !
!
```

プロトコル トラフィックは、GigabitEthernet サブインターフェイス 0/1/0/1.1 でルータ R1 に入ります。ルータ R1 はプロトコル フレームとしてフレームを検出して、カスタマー側インターフェイスで L2PT カプセル化を実行します。R1 内では、ローカル接続 *r1-connect* は、R1 のカスタマー側インターフェイスとサービス プロバイダー側インターフェイスを接続します。トラフィックは、他の複数のサービス プロバイダー ネットワークのルータまたはスイッチ（スイッチ クラウド）を介して GigabitEthernet サブインターフェイス 0/1/0/2.1 のルータ R1 から GigabitEthernet サブインターフェイス 0/5/0/1.1 のルータ R2 に通過します。ルータ R2 は、ローカル接続 *r2-connect* を介してカスタマー側インターフェイスとサービス プロバイダー側インターフェイスを接続します。したがって、トラフィックは、カスタマー側インターフェイスの GigabitEthernet 0/5/0/2.1 に送信されます。このインターフェイスで、L2PT のカプセル化が解除され、プロトコル トラフィックはルータ R2 からカスタマー ネットワークに流れます。

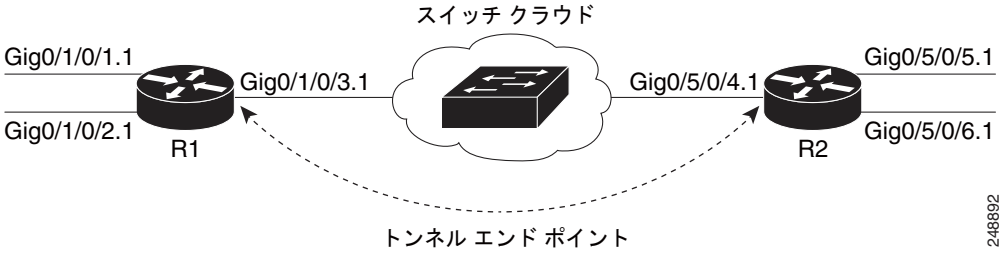
L2PT が設定されていない場合、R1 に送信されるカスタマー プロトコル フレームは終端します。カスタマー トラフィックは、さまざまなトラフィックで構成できます。プロトコル フレームは、全体的なトラフィック ストリームのうちわずかな割合で構成されます。

プロトコル フレーム タギングを使用した反転モードの L2PT

Cisco ASR 9000 シリーズ ルータは、VLAN ヘッダーを持つサポートされている L2 プロトコル フレームで L2PT カプセル化およびカプセル化解除を実行できます。L2 プロトコル フレームに VLAN ヘッダーは含まれません。ただし、カスタマー キャンパス間でカスタマー プロトコル トラフィックを転送するサービス プロバイダー（SP）ネットワークでは、この機能を配置して、SP ネットワーク内で使用できます。

図 2 に、反転モードで設定された L2PT を示します。R1 に入るカスタマー トラフィックはトランキングされており、すべてのトラフィックがタグ付きであると想定します。唯一のタグなしトラフィックは、カスタマー ネットワークから発信されるプロトコル トラフィックです。

図 2　反転モードの L2PT



反転モードで L2PT が設定されている場合、L2PT カプセル化は、フレームがインターフェイスを出ると行われます。同様に、反転モードのカプセル化解除は、フレームがインターフェイスに入ったときに実行されます。したがって、L2PT トンネルは、カスタマー側インターフェイスではなく、サービス プロバイダー側インターフェイス間で形成されます。

この例では、プロトコル トラフィックがルータ R1 に入ると、VLAN タグが追加されます。トラフィックがサービス プロバイダー ネットワークを通じて送信される前に、2 番目の VLAN タグが追加されます（100）。Cisco ASR 9000 シリーズ ルータは、二重タグ付きプロトコル フレームで L2PT カプセル化を実行します。

図 2 に、4 つのカスタマー側インターフェイス（R1 : GigabitEthernet サブインターフェイス 0/1/0.1.1、GigabitEthernet サブインターフェイス 0/1/0/2.1 および R2 : GigabitEthernet サブインターフェイス 0/5/0/5.1、GigabitEthernet サブインターフェイス 0/5/0/6.1）、および 2 つのサービス プロバイダー側インターフェイス（R1 : GigabitEthernet サブインターフェイス 0/1/0/3.1 と R2 : GigabitEthernet サブインターフェイス 0/5/0/4.1）を示します。

図 2 は、反転モードの L2PT の設定を示します。

R1 :

```
!
interface GigabitEthernet0/1/0/1
 negotiation auto
!
interface GigabitEthernet0/1/0/1.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 100 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/1/0/2
 negotiation auto
!
interface GigabitEthernet0/1/0/2.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 200 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/1/0/3
 negotiation auto
!
interface GigabitEthernet0/1/0/3.1 l2transport
 encapsulation dot1q 500
 rewrite ingress tag pop 1 symmetric
 l2protocol cpsv reverse-tunnel
 ethernet egress-filter strict
!
l2vpn
 bridge group examples
  bridge-domain r1-bridge
   interface GigabitEthernet0/1/0/1.1
```

```
   !
   interface GigabitEthernet0/1/0/2.1
   !
   interface GigabitEthernet0/1/0/3.1
   !
  !
 !
!
```

R2：

```
!
interface GigabitEthernet0/5/0/4
 negotiation auto
!
interface GigabitEthernet0/5/0/4.1 l2transport
 encapsulation dot1q 500
 rewrite ingress tag pop 1 symmetric
 l2protocol cpsv reverse-tunnel
 ethernet egress-filter strict
!
interface GigabitEthernet0/5/0/5
 negotiation auto
!
interface GigabitEthernet0/5/0/5.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 100 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/5/0/6
 negotiation auto
!
interface GigabitEthernet0/5/0/6.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 200 symmetric
 ethernet egress-filter strict
!
l2vpn
 bridge group examples
  bridge-domain r2-bridge
   interface GigabitEthernet0/5/0/4.1
   !
   interface GigabitEthernet0/5/0/5.1
   !
   interface GigabitEthernet0/5/0/6.1
   !
  !
 !
!
```

次のことが前提となっています。

- ルータ R1 に入るカスタマー トラフィックはトランキングされます。つまり、すべてのトラフィックがタグ付けされています。唯一のタグなしトラフィックは、カスタマー ネットワークから到着するプロトコル トラフィックです。
- ルータ R1 の GigabitEthernet 0/1/0/1 とルータ R2 の GigabitEthernet 0/5/0/5 のカスタマー側インターフェイスは、同じカスタマーに属しています。ルータ R1 の GigabitEthernet 0/1/0/2 とルータ R2 の GigabitEthernet 0/5/0/6 のカスタマー側インターフェイスは、別のカスタマーに属しています。
- 異なるカスタマーからのトラフィックは分離されたままになります。
- L2 プロトコル トラフィックだけがカスタマー側インターフェイスを経由して送信されます。

- カスタマー側インターフェイスに入る L2 プロトコル トラフィックはタグなしです。
- トラフィックは、スイッチ クラウドを正常にパススルーするには、L2PT カプセル化されている必要があります。

このトポロジの目的は、ルータ R1 と R2 が複数のカスタマー インターフェイスからカスタマー プロトコル トラフィックを受信する必要があり、単一のサービス プロバイダー インターフェイスとリンク間でトラフィックを多重化する必要があることです。カプセル化解除の最後に、反転が実行されます。GigabitEthernet サブインターフェイス 0/1/0/2.1 のルータ R1 に入るトラフィックは、GigabitEthernet サブインターフェイス 0/5/0/6.1 だけからルータ R2 を出るのに対して、GigabitEthernet サブインターフェイス 0/1/0/1.1 のルータ R1 に入るトラフィックは、GigabitEthernet サブインターフェイス 0/5/0/5.1 だけからルータ R2 を出ます。

GigabitEthernet インターフェイス 0/1/0/1 のルータ R1 に入るプロトコル フレームは、この方法でネットワークを通過します。

- プロトコル フレームは、フレームがタグなしであるため、GigabitEthernet サブインターフェイス 0/1/0/1.1 に送信されます。
- GigabitEthernet サブインターフェイス 0/1/0/1.1 で rewrite ステートメントを使用すると、ID 100 のタグがフレームに追加されます。
- フレームは、ルータ R1 のブリッジ ドメイン r1-bridge に入ります。
- ブリッジ（r1-bridge）は、発信元 AC（スプリット ホライズン AC）を除き、ブリッジ ドメイン上のすべての接続回線（AC）にフレームをフラッディングします。
- GigabitEthernet サブインターフェイス 0/1/0/2.1 でのイーサネット出力フィルタリングは、タグ ID のミスマッチを検出し、フレームをドロップします。このように、ブリッジ ドメインのフラッディングされたトラフィックは、他のカスタマー インターフェイスを出ることができません。
- フレームのフラッディングされたコピーは GigabitEthernet サブインターフェイス 0/1/0/3.1 に送信されます。
- GigabitEthernet サブインターフェイス 0/1/0/3.1 は 2 番目のタグを追加します。
- フレームは、GigabitEthernet インターフェイス 0/1/0/3 を介してルータ R1 を出る前に GigabitEthernet サブインターフェイス 0/1/0/3.1 によって L2PT カプセル化を受信します。

（注） 現在フレームには二重のタグが付いており（内部が 100、外部が 500）になっており、L2PT MAC DA があります。

- フレームは、L2PT カプセル化が原因で、ルータ R2 GigabitEthernet インターフェイス 0/5/0/4 に渡されます。
- フレームは、GigabitEthernet インターフェイス 0/5/0/4 のルータ R2 に入った後、GigabitEthernet サブインターフェイス 0/5/0/4.1 に送信されます。
- GigabitEthernet サブインターフェイス 0/5/0/4.1 に入るときに、L2PT カプセル解除動作がフレームで実行されます。
- 外部タグ ID 500 は、GigabitEthernet サブインターフェイス 0/5/0/4.1 によって削除されます。
- ルータ R2 のブリッジ（r2-bridge）は、すべての AC にフレームをフラッディングします。
- イーサネット出力フィルタリングは、フレームが出る AC を除くすべての AC でフレームをドロップします。
- フレームが GigabitEthernet サブインターフェイス 0/5/0/5.1 のルータ R2 を出るため、ID 100 のタグが削除されます。

- GigabitEthernet インターフェイス 0/5/0/5 のルータ R2 から出るフレームは、GigabitEthernet インターフェイス 0/1/0/1 を介してルータ R1 に入った元のフレームと同じです。

L2PT 設定メモ

L2PT を設定する際は、次の点に注意してください。

- **l2protocol** コマンドは、メインまたは L2 のいずれかのサブインターフェイスで設定できます。
- **l2protocol** コマンドは、物理またはバンドル インターフェイスで設定できます。
- **l2protocol** および **ethernet filtering** コマンドが同じインターフェイスで設定されている場合、L2PT カプセル化はイーサネット フィルタリングの前に発生します。これは、L2PT によって、CDP、STP、および VTP プロトコル フレームがイーサネット フィルタリングによってドロップされないようにすることを意味します。
- L2PT が他のインターフェイス機能で設定されている場合、L2PT カプセル化は、他のインターフェイス機能の処理の前に発生します。
- L2PT カプセル化およびカプセル化解除は、タグなしプロトコル フレーム、一重タグ フレーム、および二重タグ付きフレームでサポートされます。タグ Ethertype 0x8100、0x88A8、および 0x9100 はサポートされていますが、0x9200 はサポートされていません。

イーサネット機能の実装方法

この項では、次の作業について説明します。

- 「ポリシーベースの転送の設定」（P.69）
- 「レイヤ 2 プロトコル トンネリングの設定：例」（P.75）

（注） イーサネット インターフェイスの設定については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』を参照してください。

ポリシーベースの転送の設定

この項では、次の手順について説明します。

- 「ポリシーベースの転送のイネーブル化」（P.69）
- 「送信元バイパス フィルタの設定」（P.72）

ポリシーベースの転送のイネーブル化

ポリシーベースの転送をイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id.subinterface* **l2transport**
3. **encapsulation dot1q** *vlan-id* **ingress source-mac** *mac-address*
   または
   encapsulation dot1ad *vlan-id* **ingress source-mac** *mac-address*
   または
   encapsulation untagged ingress source-mac *mac-address*
   または
   encapsulation dot1ad *vlan-id* **dot1q** *vlan-id* **ingress source-mac** *mac-address*
   または
   encapsulation dot1q *vlan-id* **second-dot1q** *vlan-id* **ingress source-mac** *mac-address*
4. **rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric**
   または
   rewrite ingress tag push dot1q *vlan-id* **symmetric**
5. **ethernet egress-filter strict**
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/2/0/4.10 l2transport`	サブインターフェイス コンフィギュレーション モードを開始し、ポートでレイヤ 2 トランスポート モードをイネーブルにし、レイヤ 2 トランスポート コンフィギュレーション モードを開始します。
ステップ 3	**encapsulation dot1q** *vlan-id* **ingress source-mac** *mac-address* or **encapsulation dot1ad** *vlan-id* **ingress source-mac** *mac-address* or **encapsulation untagged ingress source-mac** *mac-address* or **encapsulation dot1ad** *vlan-id* **dot1q** *vlan-id* **ingress source-mac** *mac-address* or **encapsulation dot1q** *vlan-id* **second-dot1q** *vlan-id* **ingress source-mac** *mac-address* **例：** `RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 10 ingress source-mac 0.1.2` or `RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1ad 10 ingress source-mac 0.1.4` or `RP/0/RSP0/CPU0:router(config-subif)# encapsulation untagged ingress source-mac 0.1.3` or `RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1ad 10 dot1q 10 ingress source-mac 0.1.2` or `RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 10 second-dot1q 20 ingress source-mac 0.1.2`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。

	コマンドまたはアクション	目的
ステップ 4	**rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric** or **rewrite ingress tag push dot1q** *vlan-id* **symmetric** **例：** `RP/0/RSP0/CPU0:router(config-subif)# rewrite ingress tag translate 1-to-1 dot1q 100 symmetric` or `rewrite ingress tag push dot1q 101 symmetric`	サービス インスタンスへのフレーム入力で実行されるカプセル化調整を指定します。
ステップ 5	**ethernet egress-filter strict** **例：** `RP/0/RSP0/CPU0:router(config-subif)# ethernet egress-filter strict`	すべてのサブインターフェイスで厳密な出力フィルタリングをイネーブルにします。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# end` または `RP/0/RSP0/CPU0:router(config-subif)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

送信元バイパス フィルタの設定

送信元バイパス フィルタを追加するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id.subinterface* **l2transport**
3. **encapsulation dot1q** *vlan-id*
   または
   encapsulation dot1ad *vlan-id*
   または
   encapsulation untagged
   または
   encapsulation dot1ad *vlan-id* **dot1q** *vlan-id*
   または
   encapsulation dot1q *vlan-id* **second-dot1q** *vlan-id*
4. **rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric**
5. **ethernet egress-filter disable**
6. **ethernet source bypass egress-filter**
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/2/0/4.1 l2transport`	サブインターフェイス コンフィギュレーション モードを開始し、ポートでレイヤ 2 トランスポート モードをイネーブルにし、レイヤ 2 トランスポート 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	**encapsulation dot1q** *vlan-id* or **encapsulation dot1ad** *vlan-id* or **encapsulation untagged** or **encapsulation dot1ad** *vlan-id* **dot1q** *vlan-id* or **encapsulation dot1q** *vlan-id* **second-dot1q** *vlan-id* **例：** RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 10 or RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1ad 10 or RP/0/RSP0/CPU0:router(config-subif)# encapsulation untagged or RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1ad 10 dot1q 10 or RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 10 second-dot1q 20	一致する VLAN ID および EtherType をインターフェイスに割り当てます。
ステップ 4	**rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric** **例：** RP/0/RSP0/CPU0:router(config-subif)# rewrite ingress tag translate 1-to-1 dot1q 100 symmetric	サービス インスタンスへのフレーム入力で実行されるカプセル化調整を指定します。
ステップ 5	**ethernet egress-filter disable** **例：** RP/0/RSP0/CPU0:router(config-subif)# ethernet egress-filter strict	すべてのサブインターフェイスで出力フィルタリングをディセーブルにします。

	コマンドまたはアクション	目的
ステップ 6	**ethernet source bypass egress-filter** **例：** `RP/0/RSP0/CPU0:router(config-subif)# ethernet source bypass egress-filter`	サブインターフェイスで送信元バイパス出力フィルタリングをイネーブルにします。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# end` または `RP/0/RSP0/CPU0:router(config-subif)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

設定例

ここでは、次の設定例を示します。

- ポリシーベースの転送の設定：例
- レイヤ 2 プロトコル トンネリングの設定：例

ポリシーベースの転送の設定：例

次に、ポリシーベースの転送を設定する例を示します。

```
config
 interface GigabitEthernet0/0/0/2.3 l2transport
 encapsulation dot1q 10 ingress source-mac 0000.1111.2222
 rewrite ingress tag translate 1-to-1 dot1q 100 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/0/0/2.4 l2transport
 encapsulation untagged ingress source-mac 0000.1111.3333
 rewrite ingress tag push dot1q 101 symmetric
 ethernet egress-filter strict
!

interface GigabitEthernet0/0/0/0/3.1 l2transport
 encapsulation dot1q 1
 rewrite ingress tag translate 1-to-1 dot1q 4094 symmetric
 ethernet egress-filter disabled
 ethernet source-bypass-egress-filter
!
```

レイヤ 2 プロトコル トンネリングの設定：例

ここでは、転送モードと反転モードでの L2PT の設定例を示します。

転送モードでの L2PT の設定

次に、転送モードで L2PT を設定する例を示します。

カスタマー側ルータ（カプセル化側）：

```
!
interface GigabitEthernet0/1/0/1
 negotiation auto
!
interface GigabitEthernet0/1/0/1.1 l2transport
 encapsulation default
 l2protocol cpsv tunnel
!
interface GigabitEthernet0/1/0/2
 negotiation auto
!
interface GigabitEthernet0/1/0/2.1 l2transport
 encapsulation default
!
l2vpn
 xconnect group examples
```

```
  p2p r1-connect
   interface GigabitEthernet0/1/0/1.1
   interface GigabitEthernet0/1/0/2.1
  !
 !
!
```

カスタマー側ルータ（カプセル化解除側）：

```
!
interface GigabitEthernet0/5/0/1
 negotiation auto
!
interface GigabitEthernet0/5/0/1.1 l2transport
 encapsulation default
!
interface GigabitEthernet0/5/0/2
 negotiation auto
!
interface GigabitEthernet0/5/0/2.1 l2transport
 encapsulation default
 l2protocol cpsv tunnel
!
l2vpn
 xconnect group examples
  p2p r2-connect
   interface GigabitEthernet0/5/0/1.1
   interface GigabitEthernet0/5/0/2.1
  !
 !
!
```

反転モードでの L2PT の設定

次に、反転モードで L2PT を設定する例を示します。

カスタマー側ルータ（カプセル化側）：

```
!
interface GigabitEthernet0/1/0/1
 negotiation auto
!
interface GigabitEthernet0/1/0/1.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 100 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/1/0/2
 negotiation auto
!
interface GigabitEthernet0/1/0/2.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 200 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/1/0/3
 negotiation auto
!
```

```
interface GigabitEthernet0/1/0/3.1 l2transport
 encapsulation dot1q 500
 rewrite ingress tag pop 1 symmetric
 l2protocol cpsv reverse-tunnel
 ethernet egress-filter strict
!
l2vpn
 bridge group examples
  bridge-domain r1-bridge
   interface GigabitEthernet0/1/0/1.1
   !
   interface GigabitEthernet0/1/0/2.1
   !
   interface GigabitEthernet0/1/0/3.1
   !
  !
 !
!
```

カスタマー側ルータ（カプセル化解除側）：

```
!
interface GigabitEthernet0/5/0/4
 negotiation auto
!
interface GigabitEthernet0/5/0/4.1 l2transport
 encapsulation dot1q 500
 rewrite ingress tag pop 1 symmetric
 l2protocol cpsv reverse-tunnel
 ethernet egress-filter strict
!
interface GigabitEthernet0/5/0/5
 negotiation auto
!
interface GigabitEthernet0/5/0/5.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 100 symmetric
 ethernet egress-filter strict
!
interface GigabitEthernet0/5/0/6
 negotiation auto
!
interface GigabitEthernet0/5/0/6.1 l2transport
 encapsulation untagged
 rewrite ingress tag push dot1q 200 symmetric
 ethernet egress-filter strict
!
l2vpn
 bridge group examples
  bridge-domain r2-bridge
   interface GigabitEthernet0/5/0/4.1
   !
   interface GigabitEthernet0/5/0/5.1
   !
   interface GigabitEthernet0/5/0/6.1
   !
  !
 !
!
```

その他の関連資料

ここでは、ギガビットおよび 10 ギガビット イーサネット インターフェイスの実装に関する参考資料を紹介します。

関連資料

関連項目	参照先
Cisco IOS XR マスター コマンド リファレンス	『*Cisco IOS XR Master Commands List*』

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して、選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

リンク バンドルの設定

Cisco ASR 9000 シリーズ アグリゲーション サービス ルータでのバンドルは、1 つに集約され、単一のリンクとして扱われる 1 つ以上のポート グループです。1 つのバンドル内の各リンクの速度は異なっていてもよく、最も高速なリンクの速度は、最も低速なリンクの最大 4 倍とすることができます。各バンドルには、1 つの MAC、1 つの IP アドレス、1 つの設定セット（ACL または Quality of Service など）があります。

Cisco ASR 9000 シリーズ ルータでは、次のタイプのインターフェイスでバンドルがサポートされます。

- イーサネット インターフェイス
- VLAN サブインターフェイス

（注） バンドルには、モジュラ サービス カードとの 1 対 1 の関連付けはありません。

Cisco IOS XR ソフトウェアでのリンク バンドル設定機能の履歴

リリース	変更内容
リリース 3.7.2	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。

内容

この章で説明する内容は、次のとおりです。

リンク バンドルを設定するための前提条件

リンク バンドルを設定する前に、次のタスクと条件を満たしていることを確認してください。

- このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

  ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- インターフェイスの IP アドレスがわかっていること。
- 設定するバンドルに含めるリンクがわかっていること。
- イーサネット リンク バンドルを設定する場合、ルータに少なくとも次のイーサネット ラインカードのいずれかが搭載されていること。
  - 2 ポート 10 ギガビット イーサネット ラインカード
  - 4 ポート 10 ギガビット イーサネット ラインカード
  - 8 ポート 10 ギガビット イーサネット ラインカード
  - 16 ポート 10 ギガビット イーサネット ラインカード
  - 20 ポート ギガビット イーサネット ラインカード
  - 40 ポート ギガビット イーサネット ラインカード

（注） 物理インターフェイス、PLIM、およびモジュラ サービス カードの詳細については、『*Cisco ASR 9000 Series Routers Hardware Installation Guide*』を参照してください。

リンク バンドルの設定に関する情報

リンク バンドル機能を設定するには、次の概念を理解している必要があります。

リンク バンドルの概要

リンク バンドルは、1 つに束ねられたポートのグループであり、1 つのリンクとして振る舞います。リンク バンドルの利点は、次のとおりです。

- 複数のリンクが複数のラインカードにまたがり、1 つのインターフェイスを構成します。そのため、単一のリンクで障害が発生しても接続性は失われません。
- バンドルされたインターフェイスでは、バンドルの使用可能なすべてのメンバにわたってトラフィックが転送されるため、帯域幅の可用性が向上します。したがって、バンドル内のリンクの 1 つで障害が発生した場合、トラフィックは使用可能なリンクを通過できます。帯域幅はパケットフローを中断することなく追加できます。

1 つのバンドル内の個々のリンクにはさまざまな速度を設定できますが、バンドル内のすべてのリンクが同じタイプである必要があります。

Cisco IOS XR ソフトウェアでは、次の方法でイーサネット インターフェイスのバンドルを構成できます。

- IEEE 802.3ad：バンドル内のすべてのメンバー リンクの互換性を確保するため、Link Aggregation Control Protocol（LACP）を採用した標準テクノロジー。互換性がないリンクや障害になったリンクは、バンドルから自動的に削除されます。
- EtherChannel：ユーザがリンクを設定してバンドルに参加させることができるシスコの専用テクノロジー。バンドル内のリンクに互換性があるかどうかを確認するための仕組みはありません。

Cisco ASR 9000 シリーズ ルータ リンク バンドルの特性

このリストでは、Cisco ASR 9000 シリーズ ルータでのリンク バンドルの特性と制限事項を説明します。

- LACP（Link Aggregation Control Protocol）を使用するかにかかわらず、すべてのタイプのイーサネット インターフェイスをバンドルできます。
- バンドル メンバーシップは、1 つのルータにインストールされている複数のラインカードにまたがることができます。
- 1 つのバンドルは最大 64 個の物理リンクをサポートします。
- 1 つのバンドル内でリンク速度が異なっていてもよく、バンドルのメンバー間で許容される速度の差は、最大 4 倍です。
- 物理層とリンク層の設定は、バンドルの個々のメンバー リンクに対して実行します。
- ネットワーク層プロトコルおよび上位層のアプリケーションの設定は、バンドル自体に対して実行します。
- バンドルは、管理上イネーブルまたはディセーブルにできます。
- バンドル内のそれぞれのリンクは、管理上イネーブルまたはディセーブルにできます。
- イーサネット リンク バンドルは、イーサネット チャネルと同様の方法で作成され、両方のエンドシステムで同じコンフィギュレーションを入力します。
- バンドルに対して設定された MAC アドレスは、そのバンドル内の各リンクの MAC アドレスになります。
- LACP が設定されている場合、バンドル内の各リンクは、異なるメンバーに対して異なるキープアライブ周期を許可するよう設定できます。
- ロード バランシング（メンバー リンク間のデータの分散）は、パケットではなくフロー単位で実行されます。データはバンドル対するそのリンクの帯域幅に比例して、リンクに配信されます。

- QoS がサポートされており、各バンドル メンバーに均等に適用されます。
- CDP キープアライブや HDLC キープアライブなどのリンク層プロトコルは、バンドル内の各リンク上で独立して動作します。
- ルーティング アップデートや hello などの上位層プロトコルは、インターフェイス バンドルのどのメンバー リンク上でも送信されます。
- バンドルされたインターフェイスはポイント ツー ポイントです。
- リンクがバンドル内で distributing 状態になるには、その前にアップ状態なる必要があります。
- 1 つのバンドル内のすべてのリンクは、802.3ad（LACP）または EtherChannel（非 LACP）のいずれかを実行するように設定する必要があります。1 つのバンドル内の混合リンクはサポートされません。
- バンドル インターフェイスには、物理リンクと VLAN サブインターフェイスのみを含めることができます。
- リンク バンドルでのアクセス コントロール リスト（ACL）の設定は、通常のインターフェイスでの ACL の設定と同じです。
- マルチキャスト トラフィックは、バンドルのメンバー上でロード バランスされます。特定のフローに対し、内部処理によってメンバ リンクが選択され、そのフローのすべてのトラフィックがそのメンバ上で送信されます。

LACP を通じたリンク集約

異なるモジュラ サービス カード上のインターフェイスを集約することで、冗長性が提供され、インターフェイスまたはモジュラ サービス カードで障害が発生したときに、トラフィックをすばやく他のメンバー リンクにリダイレクトできます。

オプションの Link Aggregation Control Protocol（LACP）は IEEE 802 規格で定義されています。LACP では、2 台の直接接続されたシステム（ピア）間で通信し、バンドル メンバーの互換性が確認されます。Cisco ASR 9000 シリーズ ルータの場合、ピアは、別のルータまたはスイッチのいずれかにすることができます。LACP は、リンク バンドルの動作状態を監視し、次のことを確認します。

- すべてのリンクが同じ 2 台のシステム上で終端していること。
- 両方のシステムがリンクを同じバンドルの一部と見なしていること。
- すべてのリンクがピア上で適切に設定されていること

LACP は、ローカル ポート状態と、パートナー システムの状態のローカルなビューが格納されたフレームを送信します。これらのフレームが解析され、両方のシステムが同調していることが確認されます。

IEEE 802.3ad 規格

IEEE 802.3ad 規格では、一般にイーサネット リンク バンドルを構成する方法が定義されています。

バンドル メンバーとして設定された各リンクでは、この情報は、リンク バンドルの両端をホストするシステム間で交換されます。

- グローバルに一意のローカル システム ID
- リンクがメンバーになっているバンドルの ID（動作キー）
- リンクの ID（ポート ID）
- リンクの現在の集約ステータス

この情報は、リンク集約グループ ID（LAG ID）を構成するために使用されます。共通の LAG ID を共有するリンクは集約できます。個々のリンクには固有の LAG ID があります。

システム ID はルータを区別し、その一意性はシステムの MAC アドレスを使用することで保証されます。バンドル ID とリンク ID は、それを割り当てるルータでだけ意味を持ち、2 つのリンクが同じ ID を持たないことと、2 つのバンドルが同じ ID を持たないことが保証される必要があります。

ピア システムからの情報はローカル システムの情報と組み合わされ、バンドルのメンバーとして設定されたリンクの互換性が判断されます。

Cisco ASR 9000 シリーズ ルータのバンドルの MAC アドレスは、バックプレーンの予約済み MAC アドレス セットから取得されます。この MAC アドレスは、バンドル インターフェイスが存在する限りバンドルにあります。バンドルは、ユーザが別の MAC アドレスを設定するまで、この MAC アドレスを使用します。バンドルの MAC アドレスは、バンドル トラフィックを通過させる際にすべてのメンバー リンクによって使用されます。バンドルに対して設定されたすべてのユニキャスト アドレスまたはマルチキャスト アドレスも、すべてのメンバー リンクで設定されます。

（注） MAC アドレスを変更するとパケット転送に影響を与えるおそれがあるため、MAC アドレスは変更しないことを推奨します。

QoS およびリンク バンドル

入力方向では、バンドルのローカル インスタンスに QoS が適用されます。各バンドルはキューのセットに関連付けられます。QoS は、バンドル上で設定されているさまざまなネットワーク層プロトコルに適用されます。

出方向では、メンバー リンクへの参照を持つバンドルに QoS が適用されます。QoS は、メンバーの帯域幅の合計に基づいて適用されます。

QoS が入力または出力方向のいずれかのバンドルに適用される場合、QoS は各メンバー インターフェイスに適用されます。

リンク バンドル機能は、『*Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Configuration Guide*』に記載されているすべての QoS 機能をサポートします。

リンク バンドル機能は、次の QoS 機能をサポートします。

- 高優先順位 / 低優先順位：最大帯域幅は、バンドル インターフェイスの帯域幅のパーセンテージとして計算されます。このパーセンテージは出力上のすべてのメンバー リンクに適用されるか、入力上のローカル バンドル インスタンスに適用されます。
- 保証される帯域幅：パーセンテージで提供され、すべてのメンバー リンクに適用されます。
- トラフィック シェーピング：パーセンテージで提供され、すべてのメンバー リンクに適用されます。
- WRED：最小および最大パラメータは、メンバー リンクまたはバンドル インスタンスごとの正しい比率に変換され、バンドルに適用されます。
- マーキング：ポリシーに従ったパケットの QoS レベルの変更プロセス。
- テール ドロップ：キューが一杯のときにパケットはドロップされます。

イーサネット リンク バンドル上の VLAN

802.1Q VLAN サブインターフェイスを 802.3ad イーサネット リンク バンドル上で設定できます。イーサネット リンク バンドル上に VLAN を追加するときには、次の点に注意してください。

- バンドルごとに許可される VLAN の最大数は 4000 です。
- ルータごとに許可されるバンドル VLAN の最大数は 16000 です。

（注） バンドル VLAN のメモリ要件は、標準の物理インターフェイスよりも若干多くなります。

バンドル上で VLAN サブインターフェイスを作成するには、次のように、**interface Bundle-Ether** コマンドを使用して VLAN サブインターフェイス インスタンスを追加します。

interface Bundle-Ether *instance.subinterface*

イーサネット リンク バンドル上で VLAN を作成した後、すべての物理 VLAN サブインターフェイス コンフィギュレーションがそのリンク バンドル上でサポートされます。

リンク バンドルの設定の概要

リンク バンドルの設定プロセスの一般的な概要を次の手順に示します。リンクをバンドルに追加する前に、リンクから以前のネットワーク層コンフィギュレーションをすべてクリアする必要があることに注意してください。

1. グローバル コンフィギュレーション モードで、リンク バンドルを作成します。イーサネット リンク バンドルを作成するには、**interface Bundle-Ether** コマンドを入力します。
2. **ipv4 address** コマンドを使用して、IP アドレスとサブネット マスクを仮想インターフェイスに割り当てます。
3. インターフェイス コンフィギュレーション サブモードで **bundle id** コマンドを使用し、ステップ 1 で作成したバンドルにインターフェイスを追加します。1 つのバンドルに最大 32 個のリンクを追加できます。

（注） リンクは、そのリンクのインターフェイス コンフィギュレーション サブモードからバンドルのメンバーに設定できます。

カードのフェールオーバー時のノンストップ フォワーディング

Cisco IOS XR ソフトウェアは、アクティブおよびスタンバイ RSP カード間でのフェールオーバー時のノンストップ フォワーディングをサポートしています。ノンストップ フォワーディングを使用すると、フェールオーバーが発生したときにリンク バンドルの状態が変化しません。

たとえば、アクティブな RSP が障害になった場合、スタンバイ RSP が動作可能になります。障害になった RSP のコンフィギュレーション、ノードの状態、チェックポイント データは、スタンバイ RSP に複製されます。スタンバイ RSP がアクティブ RSP になったとき、バンドルされたインターフェイスはすべて存在します。

（注） フェールオーバー先は常にスタンバイ RSP です。

（注） スタンバイ インターフェイス コンフィギュレーションが維持されることを保証するために何かを設定する必要はありません。

リンクのフェールオーバー

バンドルのメンバ リンクの 1 つに障害が発生すると、トラフィックは動作可能な残りのメンバ リンクにリダイレクトされ、トラフィック フローは中断されません。

バンドル インターフェイス：冗長性、ロード シェアリング、集約

Cisco ASR 9000 シリーズ アグリゲーション サービス ルータでのバンドルは、1 つに集約され、単一のリンクとして扱われる 1 つ以上のポート グループです。1 つのバンドル内の各リンクの速度は異なっていてもよく、最も高速なリンクの速度は、最も低速なリンクの最大 4 倍とすることができます。各バンドルには、1 つの MAC、1 つの IP アドレス、1 つの設定セット（ACL または Quality of Service など）があります。

Cisco ASR 9000 シリーズ ルータでは、次のタイプのインターフェイスでバンドルがサポートされます。

- イーサネット インターフェイス
- VLAN サブインターフェイス

リンク バンドルの設定方法

この項では、次の手順について説明します。

イーサネット リンク バンドルの設定

ここでは、イーサネット リンク バンドルの設定方法について説明します。

（注） イーサネット リンク バンドルでは MAC アカウンティングはサポートされていません。

（注） イーサネット バンドルをアクティブにするためには、バンドルの両方の接続ポイントで同じ設定を行う必要があります。

手順の概要

イーサネット リンク バンドルを作成するには、次の手順のように、バンドルを作成し、そのバンドルにメンバー インターフェイスを追加します。

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **ipv4 address** *ipv4-address mask*
4. **bundle minimum-active bandwidth** *kbps*（任意）
5. **bundle minimum-active links** *links*（任意）
6. **bundle maximum-active links** *links*（任意）
7. **bundle maximum-active links** *links* **hot-standby**（任意）
8. **exit**
9. **interface** {**GigabitEthernet** | **TenGigE**} *instance*
10. **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}
11. **no shutdown**
12. **exit**
13. ステップ 2 で作成したバンドルにさらにリンクを追加するには、ステップ 8 から 11 を繰り返します。
14. **end**
    または
    commit
15. **exit**
16. **exit**
17. 接続のリモート エンドでステップ 1 から 15 を実行します。
18. **show bundle Bundle-Ether** *bundle-id* [**reasons**]

19. **show lacp Bundle-Ether** *bundle-id*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface Bundle-Ether** *bundle-id* **例：** `RP/0/RSP0/CPU0:router#(config)# interface Bundle-Ether 3`	新しいイーサネット リンク バンドルを作成し名前を付与します。 この **interface Bundle-Ether** コマンドを実行すると、インターフェイス コンフィギュレーション サブモードが開始されます。このモードでは、インターフェイス固有のコンフィギュレーション コマンドを入力できます。インターフェイス コンフィギュレーション サブモードを終了して通常のグローバル コンフィギュレーション モードに戻るには、**exit** コマンドを使用します。
ステップ 3	**ipv4 address** *ipv4-address mask* **例：** `RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0`	**ipv4 address** コンフィギュレーション サブコマンドを使用して、IP アドレスとサブネット マスクを仮想インターフェイスに割り当てます。
ステップ 4	**bundle minimum-active bandwidth** *kbps* **例：** `RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000`	（任意）ユーザがバンドルをアップ状態にする前に必要な最小帯域幅を設定します。
ステップ 5	**bundle minimum-active links** *links* **例：** `RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active links 2`	（任意）特定のバンドルをアップ状態にする前に必要なアクティブ リンク数を設定します。

	コマンドまたはアクション	目的
ステップ 6	**bundle maximum-active links** *links* **例：** `RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 1`	（任意）1 個のアクティブ リンクと、アクティブ リンクに障害が発生した場合に、バンドルに迅速に引き継ぐことができるスタンバイ モードの 1 個のリンクを指定します（1：1 保護）。 1 つのバンドルで許可されるデフォルトのアクティブ リンク数は 8 です。 **（注）** **bundle maximum-active** コマンドを実行すると、バンドル内で最もプライオリティが高いリンクだけがアクティブになります。プライオリティは、**bundle port-priority** コマンドの値に基づいて決定されます（値が小さいほど、プライオリティが高くなります）。したがって、アクティブにするリンクに高いプライオリティを設定することを推奨します。
ステップ 7	**bundle maximum-active links links hot-standby** **例：** `RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby`	**hot-standby** キーワードは、バンドルが一時的に最小リンクまたは帯域幅しきい値未満になる間にスイッチオーバーまたはスイッチバック イベントでバンドル フラップを回避するために役立ちます。 これは、このために、wait-while タイマーと suppress-flaps タイマーのデフォルト値を設定します。
ステップ 8	**exit** **例：** `RP/0/RSP0/CPU0:router(config-if)# exit`	イーサネット リンク バンドルのインターフェイス コンフィギュレーション サブモードを終了します。
ステップ 9	**interface** {**GigabitEthernet** | **TenGigE**} *instance* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 1/0/0/0`	指定したインターフェイスのインターフェイス コンフィギュレーション モードを開始します。 **GigabitEthernet** キーワードまたは **TenGigE** キーワードを入力して、インターフェイスの種類を指定します。*instance* 引数には、*rack*/*slot*/*module* 形式のノード ID を指定します。 混合帯域幅のバンドル メンバの設定は、1:1 冗長性が設定されている場合にだけサポートされます（これは、10 GigabitEthernet インターフェイスのバックアップとして 1 GigabitEthernet メンバしか設定できないことを意味します）。 **（注）** 混合リンク バンドル モードは、アクティブ/スタンバイ動作が設定されている場合にだけサポートされます（通常はスタンバイ モードで低速リンクです）。

	コマンドまたはアクション	目的
ステップ 10	**bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}] **例：** `RP/0/RSP0/CPU0:router(config-if)# bundle-id 3`	指定したバンドルにリンクを追加します。 バンドル上でアクティブ LACP またはパッシブ LACP をイネーブルにするには、オプションの **mode active** キーワードまたは **mode passive** キーワードをコマンド文字列に追加します。 LACP をサポートせずにバンドルにリンクを追加するには、オプションの **mode on** キーワードをコマンド文字列に追加します。 **（注）** **mode** キーワードを指定しない場合、デフォルトのモードは **on** になります（LACP はポート上で動作しません）。
ステップ 11	**no shutdown** **例：** `RP/0/RSP0/CPU0:router(config-if)# no shutdown`	（任意）リンクがダウン状態の場合はアップ状態にします。**no shutdown** コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。
ステップ 12	**exit** **例：** `RP/0/RSP0/CPU0:router(config-if)# exit`	イーサネット インターフェイスのインターフェイス コンフィギュレーション サブモードを終了します。
ステップ 13	（任意）バンドルにさらにリンクを追加するには、ステップ 8 から 11 を繰り返します。	—
ステップ 14	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 15	**exit** **例：** `RP/0/RSP0/CPU0:router(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 16	**exit** **例：** `RP/0/RSP0/CPU0:router(config)# exit`	グローバル コンフィギュレーション モードを終了します。
ステップ 17	接続のリモート エンドでステップ 1 から 15 を実行します。	リンク バンドルの他端をアップ状態にします。
ステップ 18	**show bundle Bundle-Ether** *bundle-id* [**reasons**] **例：** `RP/0/RSP0/CPU0:router# show bundle Bundle-Ether 3 reasons`	（任意）指定したイーサネット リンク バンドルに関する情報を表示します。
ステップ 19	**show lacp Bundle-Ether** *bundle-id* **例：** `RP/0/RSP0/CPU0:router# show lacp Bundle-Ether 3`	（任意）LACP ポートとそのピアに関する詳細情報を表示します。

VLAN バンドルの設定

ここでは、VLAN バンドルの設定方法について説明します。VLAN バンドルの作成では、主に次の 3 つの作業を行います。

1. イーサネット バンドルを作成します。
2. VLAN サブインターフェイスを作成し、イーサネット バンドルに割り当てます。
3. イーサネット リンクをイーサネット バンドルに割り当てます。

これらの作業について、以降の手順で詳しく説明します。

（注） VLAN バンドルをアクティブにするには、バンドル接続の両端で同じ設定を行う必要があります。

手順の概要

VLAN リンク バンドルの作成について、次の手順で説明します。

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **ipv4 address** *ipv4-address mask*
4. **bundle minimum-active bandwidth** *kbps*（任意）
5. **bundle minimum-active links** *links*（任意）
6. **bundle maximum-active links** *links*（任意）
7. **exit**
8. **interface Bundle-Ether** *bundle-id.vlan-id*

9. **encapsulation dot1q** *vlan-id*
10. **ipv4 address** *ipv4-address mask*
11. **no shutdown**
12. **exit**
13. ステップ 2 で作成したバンドルにさらに VLAN を追加するには、ステップ 7 から 12 を繰り返します。
14. **end**
    または
    commit
15. **exit**
16. **exit**
17. **show ethernet trunk bundle-Ether** *instance*
18. **configure**
19. **interface** {**GigabitEthernet** | **TenGigE**} *instance*
20. **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}]
21. **no shutdown**
22. ステップ 2 で作成したバンドルにさらにイーサネット インターフェイスを追加するには、ステップ 19 から 21 を繰り返します。
23. **end**
    または
    commit
24. 接続のリモート エンドでステップ 1 から 23 を実行します。
25. **show bundle Bundle-Ether** *bundle-id* [**reasons**]
26. **show ethernet trunk bundle-Ether** *instance*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例**： `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**interface Bundle-Ether** *bundle-id* **例：** RP/0/RSP0/CPU0:router#(config)# interface Bundle-Ether 3	新しいイーサネット リンク バンドルを作成し名前を付与します。 この **interface Bundle-Ether** コマンドを実行すると、インターフェイス コンフィギュレーション サブモードが開始されます。このモードでは、インターフェイス固有のコンフィギュレーション コマンドを入力できます。インターフェイス コンフィギュレーション サブモードを終了して通常のグローバル コンフィギュレーション モードに戻るには、**exit** コマンドを使用します。
ステップ 3	**ipv4 address** *ipv4-address mask* **例：** RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0	**ipv4 address** コンフィギュレーション サブコマンドを使用して、IP アドレスとサブネット マスクを仮想インターフェイスに割り当てます。
ステップ 4	**bundle minimum-active bandwidth** *kbps* **例：** RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000	（任意）ユーザがバンドルをアップ状態にする前に必要な最小帯域幅を設定します。
ステップ 5	**bundle minimum-active links** *links* **例：** RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active links 2	（任意）特定のバンドルをアップ状態にする前に必要なアクティブ リンク数を設定します。
ステップ 6	**bundle maximum-active links** *links* **例：** RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 1	（任意）1 個のアクティブ リンクと、アクティブ リンクに障害が発生した場合に、バンドルに迅速に引き継ぐことができるスタンバイ モードの 1 個のリンクを指定します（1：1 保護）。 **（注）** 1 つのバンドルで許可されるデフォルトのアクティブ リンク数は 8 です。 **（注）** **bundle maximum-active** コマンドを実行すると、バンドル内で最もプライオリティが高いリンクだけがアクティブになります。プライオリティは、**bundle port-priority** コマンドの値に基づいて決定されます（値が小さいほど、プライオリティが高くなります）。したがって、アクティブにするリンクに高いプライオリティを設定することを推奨します。
ステップ 7	**exit** **例：** RP/0/RSP0/CPU0:router(config-if)# exit	インターフェイス コンフィギュレーション 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 8	**interface Bundle-Ether** *bundle-id.vlan-id* **例：** `RP/0/RSP0/CPU0:router#(config)# interface Bundle-Ether 3.1`	新しい VLAN を作成し、その VLAN をステップ 2 で作成したイーサネット バンドルに割り当てます。 *bundle-id* 引数には、ステップ 2 で作成した *bundle-id* を指定します。 *vlan-id* にはサブインターフェイス ID を指定します。範囲は 1 ～ 4094 です（0 と 4095 は予約されています）。 **（注）** *.vlan-id* 引数を **interface Bundle-Ether** *bundle-id* コマンドに追加すると、サブインターフェイス コンフィギュレーション モードが開始されます。
ステップ 9	**encapsulation dot1q** *vlan-id* **例：** `RP/0/RSP0/CPU0:router#(config-subif)# encapsulation dot1q 10`	VLAN をサブインターフェイスに割り当てます。 *vlan-id* 引数にはサブインターフェイス ID を指定します。範囲は 1 ～ 4094 です（0 と 4095 は予約されています）。
ステップ 10	**ipv4 address** *ipv4-address mask* **例：** `RP/0/RSP0/CPU0:router#(config-subif)# ipv4 address 10.1.2.3/24`	IP アドレスおよびサブネット マスクをサブインターフェイスに割り当てます。
ステップ 11	**no shutdown** **例：** `RP/0/RSP0/CPU0:router#(config-subif)# no shutdown`	（任意）リンクがダウン状態の場合はアップ状態にします。**no shutdown** コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。
ステップ 12	**exit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# exit`	VLAN サブインターフェイスのサブインターフェイス コンフィギュレーション モードを終了します。
ステップ 13	ステップ 2 で作成したバンドルにさらに VLAN を追加するには、ステップ 7 から 12 を繰り返します。	（任意）バンドルにさらにサブインターフェイスを追加します。

	コマンドまたはアクション	目的
ステップ 14	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# end` または `RP/0/RSP0/CPU0:router(config-subif)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 15	**exit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# 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 16	**exit** **例：** `RP/0/RSP0/CPU0:router(config)# exit`	グローバル コンフィギュレーション モードを終了します。
ステップ 17	**show ethernet trunk bundle-ether** *instance* **例：** `RP/0/RP0/CPU0:router# show ethernet trunk bundle-ether 5`	（任意）インターフェイス コンフィギュレーションを表示します。 イーサネット バンドル インスタンスの範囲は 1 ～ 65535 です。
ステップ 18	**configure** **例：** `RP/0/RSP0/CPU0:router # configure`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 19	**interface** {**GigabitEthernet** | **TenGigE**} *instance* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 1/0/0/0`	バンドルに追加するイーサネット インターフェイスのインターフェイス コンフィギュレーション モードを開始します。 **GigabitEthernet** キーワードまたは **TenGigE** キーワードを入力して、インターフェイスの種類を指定します。*instance* 引数には、*rack*/slot/module 形式のノード ID を指定します。 **（注）** リンク バンドルの両端にイーサネット インターフェイスを追加するまでは、VLAN バンドルはアクティブになりません。
ステップ 20	**bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}] **例：** `RP/0/RSP0/CPU0:router(config-if)# bundle-id 3`	ステップ 2 から 13 で設定したバンドルにイーサネット インターフェイスを追加します。 バンドル上でアクティブ LACP またはパッシブ LACP をイネーブルにするには、オプションの **mode active** キーワードまたは **mode passive** キーワードをコマンド文字列に追加します。 LACP をサポートせずにバンドルにインターフェイスを追加するには、オプションの **mode on** キーワードをコマンド文字列に追加します。 **（注）** **mode** キーワードを指定しない場合、デフォルトのモードは **on** になります（LACP はポート上で動作しません）。
ステップ 21	`no shutdown` **例：** `RP/0/RSP0/CPU0:router(config-if)# no shutdown`	（任意）リンクがダウン状態の場合はアップ状態にします。**no shutdown** コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。
ステップ 22	VLAN バンドルにさらにイーサネットインターフェイスを追加するには、ステップ 19 から 21 を繰り返します。	—

	コマンドまたはアクション	目的
ステップ 23	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# end` または `RP/0/RSP0/CPU0:router(config-subif)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 24	VLAN バンドル接続のリモート エンドでステップ 1 から 23 を実行します。	リンク バンドルの他端をアップ状態にします。
ステップ 25	`show bundle Bundle-Ether` *bundle-id* [**reasons**] **例：** `RP/0/RSP0/CPU0:router# show bundle Bundle-Ether 3 reasons`	（任意）指定したイーサネット リンク バンドルに関する情報を表示します。 **show bundle Bundle-Ether** コマンドを実行すると、指定したバンドルに関する情報が表示されます。バンドルが正しく設定されており、トラフィックを伝送している場合は、**show bundle Bundle-Ether** コマンドの出力の State フィールドに数値 4 が表示されます。これは、指定された VLAN バンドル ポートが「分散している」ことを意味します。
ステップ 26	**show ethernet trunk bundle-ether** *instance* **例：** `RP/0/RP0/CPU0:router# show ethernet trunk bundle-ether 5`	（任意）インターフェイス コンフィギュレーションを表示します。 イーサネット バンドル インスタンスの範囲は 1 ～ 65535 です。

リンク バンドルの設定例

ここでは、次の設定例を示します。

- LACP が動作する EtherChannel バンドル：例
- イーサネット バンドル上での VLAN の作成：例
- Cisco 7600 EtherChannel に接続された ASR 9000 リンク バンドル：例

LACP が動作する EtherChannel バンドル：例

次に、2 つのポートを結合して、LACP が動作する EtherChannel バンドルを構成する例を示します。

```
RP/0/RSP0/CPU0:Router# config
RP/0/RSP0/CPU0:Router(config)# interface Bundle-Ether 3
RP/0/RSP0/CPU0:Router(config-if)# ipv4 address 1.2.3.4/24
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active links 1
RP/0/RSP0/CPU0:Router(config-if)# exit
RP/0/RSP0/CPU0:Router(config)# interface TenGigE 0/3/0/0
RP/0/RSP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RSP0/CPU0:Router(config-if)# no shutdown
RP/0/RSP0/CPU0:Router(config)# exit
RP/0/RSP0/CPU0:Router(config)# interface TenGigE 0/3/0/1
RP/0/RSP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RSP0/CPU0:Router(config-if)# no shutdown
RP/0/RSP0/CPU0:Router(config-if)# exit
```

イーサネット バンドル上での VLAN の作成：例

次に、イーサネット バンドル上で 2 つの VLAN を作成し起動状態にする例を示します。

```
RP/0/RSP0/CPU0:Router# config
RP/0/RSP0/CPU0:Router(config)# interface Bundle-Ether 1
RP/0/RSP0/CPU0:Router(config-if)# ipv4 address 1.2.3.4/24
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active links 1
RP/0/RSP0/CPU0:Router(config-if)# exit
RP/0/RSP0/CPU0:Router(config)# interface Bundle-Ether 1.1
RP/0/RSP0/CPU0:Router(config-subif)# dot1q vlan 10
RP/0/RSP0/CPU0:Router(config-subif)# ip addr 10.2.3.4/24
RP/0/RSP0/CPU0:Router(config-subif)# no shutdown
RP/0/RSP0/CPU0:Router(config-subif)# exit
RP/0/RSP0/CPU0:Router(config)# interface Bundle-Ether 1.2
RP/0/RSP0/CPU0:Router(config-subif)# dot1q vlan 20
RP/0/RSP0/CPU0:Router(config-subif)# ip addr 20.2.3.4/24
RP/0/RSP0/CPU0:Router(config-subif)# no shutdown
RP/0/RSP0/CPU0:Router(config-subif)# exit
RP/0/RSP0/CPU0:Router(config)#interface gig 0/1/5/7
RP/0/RSP0/CPU0:Router(config-if)# bundle-id 1 mode act
RP/0/RSP0/CPU0:Router(config-if)# commit
RP/0/RSP0/CPU0:Router(config-if)# exit
RP/0/RSP0/CPU0:Router(config)# exit
RP/0/RSP0/CPU0:Router # show ethernet trunk bundle-ether 1
```

Cisco 7600 EtherChannel に接続された ASR 9000 リンク バンドル：例

次に、ASR 9000 シリーズ ルータ（ASR-9010）と、L2 および L3 サービスの両方をサポートするメトロ イーサネット ネットワーク内の Cisco 7600 シリーズ ルータ（P19_C7609-S）間のバンドルのエンド ツー エンドの例を示します。

Cisco ASR 9000 シリーズ ルータでは、バンドルは、LACP、1:1 リンク保護、2 つの L2 サブインターフェイス、2 つのレイヤ 3 サブインターフェイスで設定されます。

IOS XR 側：

```
hostname PE44_ASR-9010

interface Bundle-Ether16
 description Connect to P19_C7609-S Port-Ch 16
 mtu 9216
 no ipv4 address
 bundle maximum-active links 1
!
interface Bundle-Ether16.160 l2transport
 description Connect to P19_C7609-S Port-Ch 16 EFP 160
 encapsulation dot1q 160
!
interface Bundle-Ether16.161 l2transport
 description Connect to P19_C7609-S Port-Ch 16 EFP 161
 encapsulation dot1q 161
!
interface Bundle-Ether16.162
 description Connect to P19_C7609-S Port-Ch 16.162
 ipv4 address 10.194.8.44 255.255.255.0
 encapsulation dot1q 162
!
interface Bundle-Ether16.163
 description Connect to P19_C7609-S Port-Ch 16.163
 ipv4 address 10.194.12.44 255.255.255.0
 encapsulation dot1q 163
!


interface GigabitEthernet0/1/0/16
 description Connected to P19_C7609-S GE 8/0/16
 bundle id 16 mode active
 bundle port-priority 1
!
interface GigabitEthernet0/1/0/17
 description Connected to P19_C7609-S GE 8/0/17
 bundle id 16 mode active
 bundle port-priority 2
!
```

IOS XR 側：CE デバイスへの接続：

```
hostname PE44_ASR-9010

interface GigabitEthernet0/1/0/3.160 l2transport
 description VLAN 160 over BE 16.160
 encapsulation dot1q 100 second-dot1q 160
 rewrite ingress tag pop 1 symmetric
!
interface GigabitEthernet0/1/0/3.161 l2transport
 description VLAN 161 over BE 16.161
 encapsulation dot1q 161
!
l2vpn
 !
 xconnect group 160
  p2p 160
   interface Bundle-Ether16.160
   interface GigabitEthernet0/1/0/3.160
   description VLAN_160_over_BE_16.160
  !
 !
 xconnect group 161
  p2p 161
   interface Bundle-Ether16.161
   interface GigabitEthernet0/1/0/3.161
   description VLAN_161_over_BE_16.161
  !
 !
```

IOS XR 側：CE デバイス：

```
hostname PE64_C3750-ME
!
vlan 161
!
interface GigabitEthernet1/0/1
 description Connected to PE65_ME-C3400 GE 0/1
 switchport access vlan 100
 switchport mode dot1q-tunnel
!
interface GigabitEthernet1/0/2
 description Connected to PE44_ASR-9010 GE 0/1/0/3
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 100,161
 switchport mode trunk
!
interface Vlan161
 description VLAN 161 over BE 16.161 on PE44
 ip address 161.0.0.64 255.255.255.0
!



hostname PE65_ME-C3400
!
vlan 160
!
interface GigabitEthernet0/1
 description Connected to PE64_C3750-ME GE 1/0/1
 port-type nni
```

```
 switchport trunk allowed vlan 160
 switchport mode trunk
!
interface Vlan160
 description VLAN 160 over BE 16.160 on PE44
 ip address 160.0.0.65 255.255.255.0
!
```

IOS 側 :

```
hostname P19_C7609-S

port-channel load-balance src-dst-port
!
interface Port-channel16
 description Connected to PE44_ASR-9010 BE 16
 mtu 9202
 no ip address
 logging event link-status
 logging event status
 speed nonegotiate
 mls qos trust dscp
 lacp fast-switchover
 lacp max-bundle 1
 service instance 160 ethernet
  description Connected to PE44_ASR-9010 BE 16.160
  encapsulation dot1q 160
 !
 service instance 161 ethernet
  description Connected to PE44_ASR-9010 BE 16.161
  encapsulation dot1q 161
 !
!
interface Port-channel16.162
 description Connected to PE44_ASR-9010 BE 16.162
 encapsulation dot1Q 162
 ip address 10.194.8.19 255.255.255.0
!
interface Port-channel16.163
 description Connected to PE44_ASR-9010 BE 16.163
 encapsulation dot1Q 163
 ip address 10.194.12.19 255.255.255.0
!

interface GigabitEthernet8/0/16
 no shut
 description Connected to PE44_ASR-9010 GE 0/1/0/16
 mtu 9202
 no ip address
 logging event link-status
 logging event status
 speed nonegotiate
 no mls qos trust dscp
 lacp port-priority 1
 channel-protocol lacp
 channel-group 16 mode active
!
interface GigabitEthernet8/0/17
 no shut
 description Connected to PE44_ASR-9010 GE 0/1/0/17
 mtu 9202
 no ip address
```

```
 logging event link-status
 logging event status
 speed nonegotiate
 no mls qos trust dscp
 lacp port-priority 2
 channel-protocol lacp
 channel-group 16 mode active
!
```

IOS 側：CE デバイスへの接続：

```
hostname P19_C7609-S

interface GigabitEthernet8/0/7
 description Connected to PE62_C3750-ME GE 1/0/2
 mtu 9000
 no ip address
 speed nonegotiate
 mls qos trust dscp
 service instance 160 ethernet
  description VLAN 160 over Port-Ch 16
  encapsulation dot1q 100 second-dot1q 160
  rewrite ingress tag pop 1 symmetric
 !
 service instance 161 ethernet
  description VLAN 161 over Port-Ch 16
  encapsulation dot1q 161
 !
!
connect eline-161 Port-channel16 161 GigabitEthernet8/0/7 161
 !
!
connect eline-160 Port-channel16 160 GigabitEthernet8/0/7 160
 !
!
```

IOS 側：CE デバイス：

```
hostname PE62_C3750-ME
!
vlan 161
!
interface GigabitEthernet1/0/1
 description Connected to PE63_ME-C3400 GE 0/1
 switchport access vlan 100
 switchport mode dot1q-tunnel
!
interface GigabitEthernet1/0/2
 description Connected to P19_C7609-S GE 8/0/7
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 100,161
 switchport mode trunk
!
interface Vlan161
 description VLAN 161 over Port-Chan 16 on P19
 ip address 161.0.0.62 255.255.255.0
!
```

```
hostname PE63_ME-C3400
!
vlan 160
!
interface GigabitEthernet0/1
 description Connected to PE62_C3750-ME GE 1/0/1
 port-type nni
 switchport trunk allowed vlan 160
 switchport mode trunk
!
interface Vlan160
 description VLAN 160 over Port-Chan 16 on P19
 ip address 160.0.0.63 255.255.255.0
!
```

その他の関連資料

ここでは、リンク バンドルの設定に関連する参考資料を示します。

関連資料

関連項目	参照先
Cisco ASR 9000 シリーズ ルータ マスター コマンド リファレンス	*『Cisco ASR 9000 Series Routers Master Commands List』*
Cisco ASR 9000 シリーズ ルータ インターフェイス コンフィギュレーション コマンド	*『Cisco ASR 9000 Series Routers Interface and Hardware Component Command Reference』*
Cisco IOS XR ソフトウェアを使用する Cisco ASR 9000 シリーズ ルータに関する初期システム ブートおよび設定情報	*『Cisco ASR 9000 Series Routers Getting Started Guide』*
ユーザ グループとタスク ID に関する情報	*『Cisco ASR 9000 Series Routers Interface and Hardware Component Command Reference』*
リモートの Craft Works Interface（CWI）クライアント管理アプリケーションからの、Cisco ASR 9000 シリーズ ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	*『Cisco ASR 9000 Series Routers Craft Works Interface Configuration Guide』*

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して、選択されたプラットフォームに対応する MIB を検索およびダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

ポイントツーポイント レイヤ 2 サービスの実装

このモジュールでは、Cisco ASR 9000 シリーズ アグリゲーション サービス ルータのポイントツーポイント レイヤ 2（L2）接続の概念および設定情報を提供します。

次のポイントツーポイント サービスがサポートされます。

- ローカル スイッチング：単一の Cisco ASR 9000 シリーズ ルータへのポイントツーポイント内部回線（別名ローカル接続）。
- 疑似回線：Cisco ASR 9000 シリーズ ルータからの仮想ポイントツーポイント回線。疑似回線は、MPLS 上で実装されます。

（注） Cisco ASR 9000 シリーズ ルータの MPLS レイヤ 2 VPN の詳細について、およびこのモジュールに記載されているコマンドの説明については、「関連資料」を参照してください。設定作業の実行時に使用する可能性があるその他のコマンドに関するドキュメントを見つけるには、オンラインの Cisco IOS XR ソフトウェア マスター コマンド索引を検索してください。

Cisco ASR 9000 シリーズ ルータへの MPLS レイヤ 2 VPN の実装に関する機能履歴

リリース	変更内容
リリース 3.7.2	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 3.9.0	スケール拡張機能が導入されました。スケール拡張機能の詳細については、表 4（P.449）を参照してください。
リリース 4.0.0	Any Transport over MPLS（AToM）機能のサポートが追加されました。
リリース 4.0.1	次の機能のサポートが追加されました。 • 疑似回線のロード バランシング • Any Transport over MPLS（AToM）機能 – HDLC over MPLS（HDLCoMPLS） – PPP over MPLS（PPPoMPLS）
リリース 4.1.0	Flexible ルータ ID 機能のサポートが追加されました。
リリース 4.2.0	次の機能のサポートが追加されました。 • MPLS トランスポート プロファイル • Circuit EMulation（CEM）over Packet
リリース 4.3.0	L2VPN ノンストップ ルーティング機能のサポートが追加されました。

内容

ポイントツーポイント レイヤ 2 サービス実装の前提条件

このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

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

ポイントツーポイント レイヤ 2 サービスの実装に関する情報

ポイントツーポイント レイヤ 2 サービスを実装するには、次の概念を理解する必要があります。

- レイヤ 2 バーチャル プライベート ネットワークの概要（P.LSC-106）
- L2VPN での ATMoMPLS の概要（P.LSC-107）
- L2VPN での仮想回線接続検証（P.LSC-108）
- Ethernet over MPLS（P.LSC-108）
- Quality of Service（P.LSC-111）
- ハイアベイラビリティ（P.LSC-112）
- 優先トンネル パス（P.LSC-112）
- マルチセグメント疑似回線（P.LSC-113）
- 疑似回線の冗長性（P.LSC-113）
- Any Transport over MPLS（P.LSC-118）
- MPLS トランスポート プロファイル（P.LSC-119）
- Circuit Emulation Over Packet Switched Network（P.LSC-121）
- L2VPN ノンストップ ルーティング（P.LSC-122）

レイヤ 2 バーチャル プライベート ネットワークの概要

レイヤ 2 バーチャル プライベート ネットワーク（L2VPN）は、IP または MPLS 対応の L2 スイッチド IP ネットワークで LAN の動作をエミュレートすることで、イーサネット デバイス同士が共通の LAN セグメントに接続した場合と同様に通信できるようにします。ポイントツーポイント L2 接続は、L2VPN を作成する場合に重要です。

インターネット サービス プロバイダー（ISP）が、フレーム リレーまたは非同期転送モード（ATM）インフラストラクチャを IP インフラストラクチャに置き換える場合、IP または MPLS 対応の L2 スイッチド IP インフラストラクチャを使用する標準的な方法を提供する必要があります。これらの方法は、カスタマーに実用的な L2 インターフェイスを提供し、具体的には、カスタマー サイトのペア間の仮想回線を提供します。

L2VPN システムを構築するには、ISP とカスタマーの間での調整が必要です。ISP は L2 接続を提供し、カスタマーは ISP から取得したデータ リンク リソースを使用してネットワークを構築します。L2VPN サービスでは、ISP は、カスタマーのネットワーク トポロジ、ポリシー、ルーティング情報、ポイントツーポイント リンクに関する情報や、他の ISP からのネットワーク ポイントツーポイント リンク関する情報を必要としません。

ISP には、次の機能を備えたプロバイダー エッジ（PE）ルータが必要です。

- レイヤ 3（L3）パケット内への L2 プロトコル データ ユニット（PDU）のカプセル化。
- any-to-any L2 転送のインターコネクト。
- パケット スイッチ ネットワーク上での L2 Quality-of-Service（QoS）のエミュレーション。
- L2 サービスの設定の簡素化。
- 各種のトンネリング メカニズム（MPLS、IPSec、GRE など）のサポート。
- L2VPN プロセス データベースには、回線および接続に関するすべての情報が含まれます。

レイヤ 2 ローカル スイッチングの概要

ローカル スイッチングにより、同じルータ上の同じタイプの 2 つのインターフェイス間で L2 データを切り替えることができます（たとえば、イーサネットからイーサネット）。インターフェイスは、同じラインカード上にあっても、2 つの異なるラインカード上にあってもかまいません。これらのタイプのスイッチング中、レイヤ 2 アドレスが、レイヤ 3 アドレスの代わりに使用されます。ローカル スイッチング接続は、一方の接続回線（AC）から他方の接続回線に L2 トラフィックを切り替えます。ローカル スイッチング接続で設定される 2 つのポートは、そのローカル接続に関連する AC です。ローカル スイッチング接続の動作は、2 つのブリッジ ポートしかないブリッジ ドメインの動作と類似しており、トラフィックはローカル接続の一方のポートに入り、他方のポートから出ます。ただし、ローカル接続に関するブリッジングがないため、MAC 学習やフラッディングはありません。また、インターフェイスの状態が DOWN の場合、ローカル接続の AC は UP 状態ではありません （この動作は、ブリッジ ドメインの動作に準拠したときにも異なります）。

ローカル スイッチング AC は、L2 トランク（メイン）インターフェイス、バンドル インターフェイス、EFP など、多種多様な L2 インターフェイスを使用します。

また、同一ポートのローカル スイッチング機能を使用すると、同じインターフェイス上の 2 つの回線の間でレイヤ 2 データをスイッチングできます。

L2VPN での ATMoMPLS の概要

ATMoMPLS は、MPLS コア上でのレイヤ 2 ポイントツーポイント接続の一種です。

ATMoMPLS 機能を実装する場合、Cisco ASR 9000 シリーズ ルータは、カスタマー エッジ（CE）デバイスが Cisco ASR 9000 シリーズ ルータに接続されているプロバイダー ネットワークのエッジでプロバイダー エッジ（PE）ルータの役割を担います。

L2VPN での仮想回線接続検証

仮想回線接続性検証（VCCV）は、L2VPN の運用、管理、およびメンテナンス（OAM）機能であり、ネットワーク オペレータが、指定した疑似回線上で IP ベースのプロバイダー エッジ間（PE-to-PE）キープアライブ プロトコルを実行できるようにし、疑似回線データ パス転送で障害が発生しないようにします。ディスポジション PE は、指定した疑似回線に関連付けられる制御チャネルで VCCV パケットを受信します。疑似回線が各方向の PE 間で確立されると、VCCV に使用される制御チャネル タイプと接続検証タイプがネゴシエートされます。

次の 2 種類のパケットが、ディスポジション出力に到達できます。

- タイプ 1：通常の Ethernet-over-MPLS（EoMPLS）データ パケットを指定します。
- タイプ 2：VCCV パケットを指定します。

Cisco ASR 9000 シリーズ ルータは、シグナリング中にイネーブルにされた場合にインバンド制御ワードを使用する、ラベル スイッチド パス（LSP）VCCV タイプ 1 をサポートしています。IPv4 では、VCCV エコー応答は、応答モードである IPv4 として送信されます。応答は IP、MPLS、またはその両方の組み合わせとして転送されます。

出力側の MPLS 転送では、VCCV pings カウンタがカウントされます。ただし、入力側では、これらはルート プロセッサから発信され、MPLS 転送カウンタとしてカウントされません。

Ethernet over MPLS

Ethernet-over-MPLS（EoMPLS）は、MPLS 対応 L3 コアを通じてイーサネット トラフィックのトンネリング メカニズムを提供し、（ラベル スタックを使用して）イーサネット プロトコル データ ユニット（PDU）を MPLS パケット内部にカプセル化して、それらを MPLS ネットワーク経由で転送します。

EoMPLS 機能は、次のサブセクションで説明します。

- イーサネット ポート モード（P.LSC-108）
- VLAN モード（P.LSC-109）
- Inter-AS モード（P.LSC-110）
- QinQ モード（P.LSC-110）
- QinAny モード（P.LSC-111）

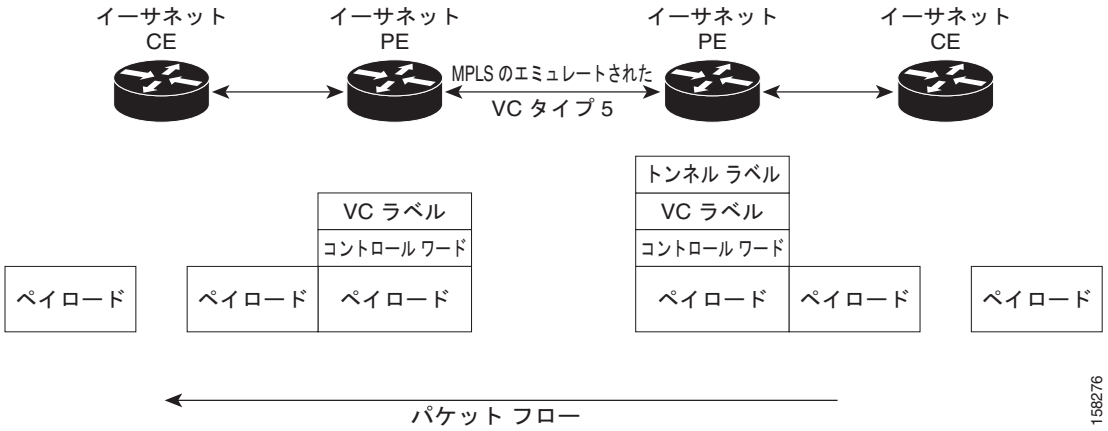
イーサネット ポート モード

イーサネット ポート モードでは、疑似回線の両端がイーサネット ポートに接続されます。このモードでは、ポートは疑似回線でトンネリングされるか、同じ PE ノードに接続されている一方の接続回線（AC）から他方の AC にパケットまたはフレームをスイッチするローカル スイッチング（別名、*接続回線間相互接続*）を使用してトンネリングされます。

図 1 に、イーサネット ポート モードの例を示します。

図 1　イーサネット ポート モードのパケット フロー

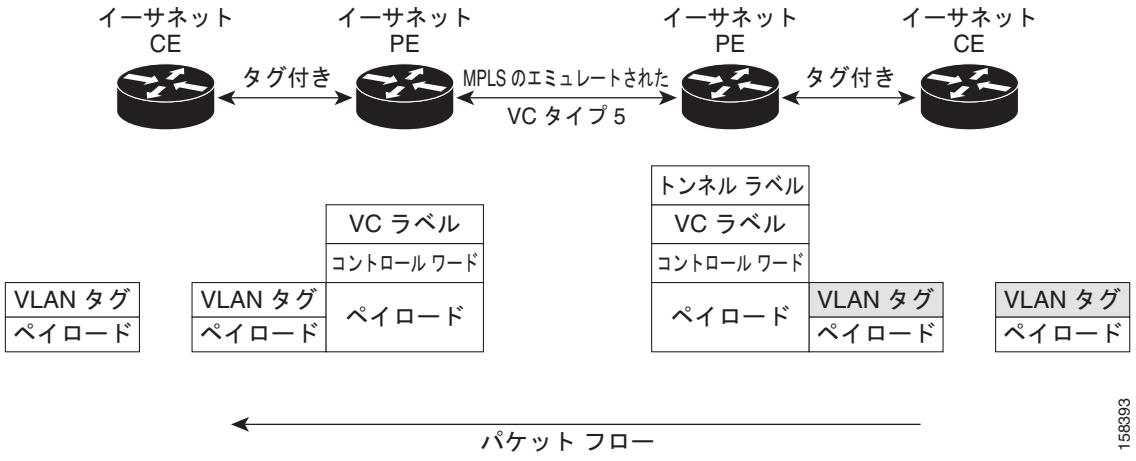
VLAN モード



VLAN モードでは、カスタマー側とプロバイダー側のリンクで、各 VLAN は、仮想接続（VC）タイプ 4 または VC タイプ 5 を使用して個別 L2VPN 接続として設定できます。VC タイプ 5 がデフォルト モードです。

図 2 に示されているように、イーサネット PE は、入力ポートから疑似回線にトラフィックを内部的に切り替えるために、イーサネット ポートに内部 VLAN タグを関連付けます。ただし、疑似回線にトラフィックを移動する前に、内部 VLAN タグを削除します。

図 2　VLAN モードのパケット フロー



出力 VLAN PE では、PE は、疑似回線から到着するフレームに VLAN タグを関連付け、トラフィックを内部的に切り替えた後、イーサネット トランク ポートにトラフィックを送信します。

（注）　ポートがトランク モードであるため、VLAN PE は VLAN タグを削除せず、追加されたタグを持つポート経由でフレームを転送します。

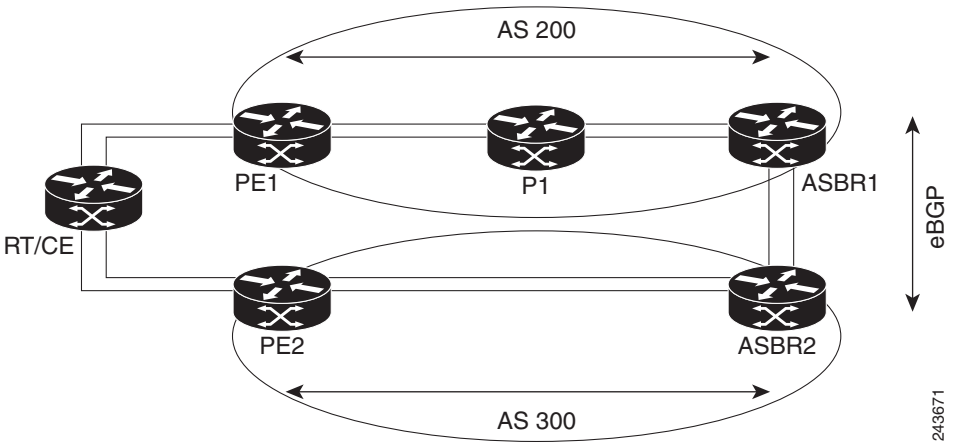
Inter-AS モード

Inter-AS は、複数のプロバイダーまたはマルチドメイン ネットワークを通じて VPN を拡張できるピアツーピア タイプ モデルです。これにより、サービス プロバイダーは相互にピアアップでき、地理的に離れた位置でエンドツーエンドの VPN 接続が実現します。

EoMPLS サポートでは、単一 AS トポロジを想定でき、このトポロジでは、ポイントツーポイント EoMPLS 相互接続の 2 つの終端にある PE ルータを接続する疑似回線が、同一自律システムに存在します。または、複数の AS トポロジを想定でき、このトポロジでは、PE ルータが iBGP および eBGP ピアリングを使用して 2 つの異なる AS に存在できます。

図 3 に、各 AS で iBGP/LDP が使用される、基本的なダブル AS トポロジの MPLS over Inter-AS を示します。

図 3　　EoMPLS over Inter-AS：基本的なダブル AS トポロジ



QinQ モード

QinQ は、複数の 802.1Q タグ（IEEE 802.1QinQ VLAN タグ スタッキング）を指定するための 802.1Q の拡張です。レイヤ 3 VPN サービス終了および L2VPN サービス転送は、QinQ サブインターフェイスではイネーブルです。

Cisco ASR 9000 シリーズ ルータは、プロバイダー エッジ ルータでのサブインターフェイスの設定に基づき、レイヤ 2 トンネリングまたはレイヤ 3 転送を実装します。この機能は、SPA および固定 PLIM で最大 2 つの QinQ タグのみサポートします。

- L2VPN 接続回線のレイヤ 2 QinQ VLAN : QinQ L2VPN 接続回線は、仮想回線タイプ 4 とタイプ 5 の両方の疑似回線を使用して、ポイントツーポイント EoMPLS ベース相互接続用のレイヤ 2 転送サブインターフェイスで設定されます。また、802.1q VLAN およびポート モードでの QinQ の完全なインターワーキングのサポートなど、ポイントツーポイント ローカル スイッチングベース相互接続用のレイヤ 2 転送サブインターフェイスで設定されます。
- レイヤ 3 QinQ VLAN : レイヤ 3 の終端ポイントとして使用されます。VLAN はいずれも入力プロバイダー エッジで削除され、フレームが転送されるときリモート プロバイダー エッジで追加され戻されます。

QinQ 上のレイヤ 3 サービスは次のとおりです。

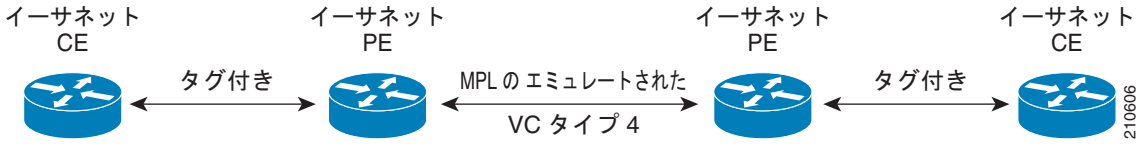
- IPv4 ユニキャストおよびマルチキャスト
- IPv6 ユニキャストおよびマルチキャスト

- MPLS
- Intermediate System-to-Intermediate System（IS-IS）で使用されるコネクションレス型ネットワーク サービス（CLNS）

QinQ モードでは、各 CE VLAN は SP VLAN 内に伝送されます。QinQ モードでは VC タイプ 5 を使用する必要がありますが、VC タイプ 4 もサポートされます。各イーサネット PE では、内部（CE VLAN）と外部（SP VLAN）の両方を設定する必要があります。

図 4 は、VC タイプ 4 を使用する Q-in-Q を表しています。

図 4　EoMPLS over QinQ モード



QinAny モード

QinAny モードでは、サービス プロバイダー VLAN タグは、プロバイダー エッジ VLAN の入力ノードと出力ノードの両方で設定されます。カスタマー エッジ VLAN タグが不明なため、カスタマー エッジ VLAN タグが疑似回線上のパケットで送信されることを除き、QinAny モードはタイプ 5 VC を使用する Q-in-Q モードに似ています。

Quality of Service

L2VPN テクノロジーを使用して、ポートおよび VLAN の動作モードの両方に Quality of Service（QoS）レベルを割り当てることができます。

L2VPN テクノロジーでは、PE ルータの QoS 機能が、エッジ方向のインターフェイス（別名、*接続回線*）で L2 ペイロードベースである必要があります。図 5 は、一般的な L2VPN ネットワークでの L2 および L3 QoS サービス ポリシーを表しています。

図 5　L2VPN QoS 機能の適用

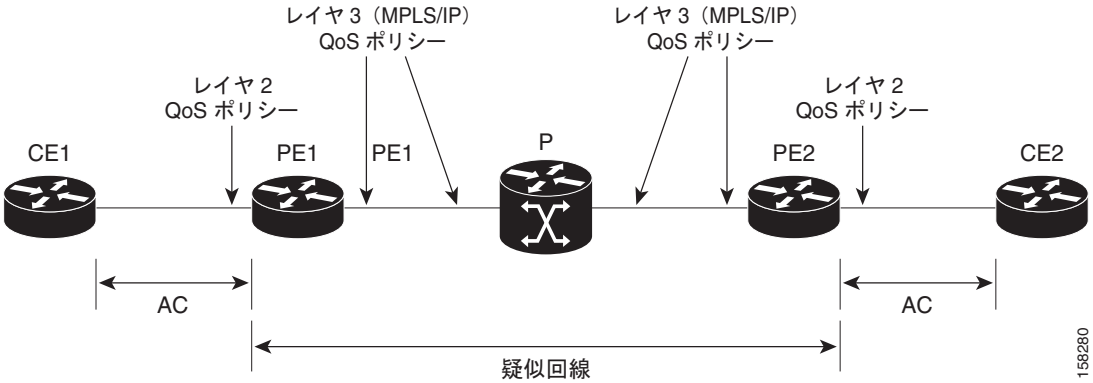
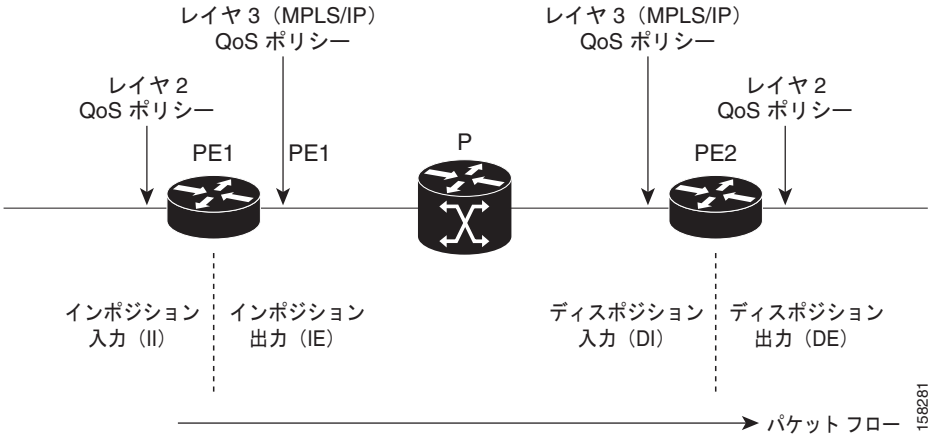


図 6 は、QoS サービス ポリシーを設定できるプロバイダー エッジ デバイス内の 4 つのパケット処理パスを表しています。L2VPN ネットワークでは、パケットはエッジ方向のインターフェイスで L2 パケットとして送受信され、コア方向のインターフェイスで MPLS（EoMPLS）パケットとして転送されます。

図 6　　L2VPN QoS リファレンス モデル



ハイアベイラビリティ

L2VPN は、ルート プロセッサとラインカードの両方でコントロール プレーンを使用し、ラインカードでフォワーディング プレーン要素を使用します。

L2VPN の可用性は次の要件を満たします。

- ルート プロセッサまたはラインカードでのコントロール プレーンの障害は、回線の転送パスには影響しません。
- ルータ プロセッサのコントロール プレーンは、ラインカードの制御およびフォワーディング プレーンに影響を与えずに、フェールオーバーをサポートします。
- L2VPN は既存のラベル配布プロトコル（LDP）のグレースフル リスタート メカニズムと統合されます。

優先トンネル パス

優先トンネル パスの機能により、特定のトラフィック エンジニアリング トンネルに疑似回線をマッピングできます。接続回線は、リモート PE ルータの IP アドレス（IGP または LDP を使用して到達可能）ではなく、特定の MPLS トラフィック エンジニアリング トンネル インターフェイスに相互接続されます。優先トンネル パスを使用する場合、L2 トラフィックを転送するトラフィック エンジニアリング トンネルが 2 台の PE ルータ間で動作することが常に想定されます（つまり、始端はインポジション PE ルータで、終端はディスポジション PE ルータです）。

（注）

- 現在、優先トンネル パス設定は MPLS カプセル化だけに適用されます。

マルチセグメント疑似回線

疑似回線は Public Switched Network（PSN）上でレイヤ 2 プロトコル データ ユニット（PDU）を転送します。マルチセグメント疑似回線は、静的または動的に設定された、複数の隣接する疑似回線セグメントのセットです。これらのセグメントは単一の疑似回線として機能し、以下を実行できます。

- 管理ドメインまたはプロビジョニング ドメインを隔離することで、エンドツーエンド サービスを管理する。
- 相互自律システム（Inter-AS）の境界を越えて、プロバイダー エッジ（PE）ノードの IP アドレスをプライベートにする。自律システム境界ルータ（ASBR）の IP アドレスを使用し、それらのルータを疑似回線の集約ルータとして扱う。ASBR は、2 つのドメインの疑似回線を結合します。

マルチセグメント疑似回線は、Inter-AS 境界または 2 つのマルチプロトコル ラベル スイッチング（MPLS）ネットワークにまたがることができます。

疑似回線は、2 台の PE ノード間のトンネルです。2 種類の PE ノードがあります。

- スイッチング PE（S-PE）ノード
  - マルチセグメント疑似回線の先行する疑似回線セグメントと後続の疑似回線セグメントの PSN トンネルを終端させます。
  - マルチセグメント疑似回線の先行する疑似回線セグメントと後続の疑似回線セグメントのコントロール プレーンとデータ プレーンを切り替えます。
- 終端 PE（T-PE）ノード
  - マルチセグメント疑似回線の最初と最後の両方のセグメントに配置されます。
  - このノードで、カスタマー方向の接続回線（AC）が疑似回線フォワーダにバインドされます。

疑似回線の冗長性

疑似回線の冗長性を使用すると、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 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。この機能により、リモート PE ルータで発生した障害、または PE ルータと CE ルータ間のリンクで発生した障害から回復できます。

L2VPN は、ルーティング プロトコルを通じて疑似回線冗長化機能を提供します。エンドツーエンド PE ルータ間の接続が障害になった場合、指示された LDP セッションとユーザ データの代替パスに引き継ぐことができます。ただし、ネットワークの一部は、この再ルーティング メカニズムでサービスの中断から保護されません。

疑似回線の冗長性を使用すると、バックアップ疑似回線を設定できます。ネットワークに冗長疑似回線と冗長ネットワーク エレメントを設定することもできます。

プライマリ疑似回線の障害前に、バックアップ疑似回線にトラフィックをスイッチングする機能が使用され、ルータのメンテナンスなどの計画された疑似回線の停止が処理されます。

（注） 疑似回線の冗長性は、ポイントツーポイント Virtual Private Wire Service（VPWS）疑似回線に対してのみ提供されます。

疑似回線のロード バランシング

冗長性を維持しつつ、ネットワークを最大限利用するには、通常、複数のリンクでのトラフィックのロード バランシングが必要です。精度の高い、より均等な分散を実現するには、プロビジョニングされたパイプの一部であるトラフィック フローのロード バランシングが理想的です。ロード バランシングは、IP アドレス、Mac アドレス、またはそれらの組み合わせに従い、フローベースにすることができます。またロード バランシングは、送信元または宛先の IP アドレス、あるいは送信元または宛先の MAC アドレスに従い、フローベースにすることができます。IP ヘッダーの処理に進むことができない場合、または IPv6 がフローベースの場合、トラフィックはデフォルトのフローベース MAC アドレスにフォールバックします。

この機能は、L2VPN 下の疑似回線に適用されます。これには、VPWS と VPLS の両方が含まれます。

（注） 疑似回線クラスに対し仮想回線（VC）ラベル ベースのロード バランシングをイネーブルにすると、L2VPN 下のグローバル フロー ベースのロード バランシングが上書きされます。

疑似回線のグループ化

疑似回線（PW）が確立されると、各 PW に、すべての PW に共通するグループ ID が割り当てられます。このグループ ID は、同一の物理ポートから作成されます。このため、物理ポートが機能しなくなるか削除されると、L2VPN は単一メッセージを送信し、グループに属するすべての PW のステータス変更をアドバタイズします。単一の L2VPN 信号であることにより、応答での煩雑な処理や切断を防ぐことができます。

（注） 疑似回線のグループ化はデフォルトでディセーブルです。

イーサネット ワイヤ サービス

イーサネット ワイヤ サービスは、ポイントツーポイントのイーサネット セグメントをエミュレートするサービスです。これは、プロバイダー エッジがレイヤ 2 で動作し、通常レイヤ 2 ネットワークで実行される以外、イーサネット専用回線（EPL）、レイヤ 1 ポイントツーポイント サービスに似ています。EWS は特定の UNI で受信されたすべてのフレームをカプセル化し、フレームに含まれる内容を参照せずに、これらのフレームを単一出力 UNI に転送します。このサービスの動作は EWS を VLAN タグ付きフレームで使用できることを示します。VLAN タグは、一部の例外を除いて EWS（ブリッジ プロトコル データ ユニット（BPDU））に対して透過的です。これらの例外には、IEEE 802.1x、IEEE 802.2ad、および IEEE 802.3x が含まれます。これは、これらのフレームがローカルで意味を持ち、カスタマーとサービス プロバイダーの両方がそれらのフレームをローカルで終了できるよう支援されるためです。

カスタマー側には 3 つのタイプがあります。

- タグなし
- 単一タグ付き
- 二重タグ付き
- 802.1q
- 802.1ad

E-Line サービス

E-Line サービスは 2 つの UNI 間のポイントツーポイント EVC を提供します。2 種類の E-Line サービスがあります。

- イーサネット専用回線（EPL）
  - サービス多重化は許可されていません
  - 透過
  - VLAN ID マップでカスタマーと SP 間は調整されません
- イーサネット仮想専用回線（EVPL）
  - サービス多重化が許可されています
  - サービス フレームの完全な透過性は必要ありません

イーサネット LAN（E-LAN）サービス

E-LAN サービスはマルチポイント接続を提供します（2 つ以上の UNI を接続できます）。すべてのサイトでイーサネットが相互接続されます（クラウド内にマルチポイントツーマルチポイント EVC があります）。

E-LAN サービスのタイプ

透過型 LAN サービス（TLS）

- バンドルされたサービス

Ethernet Virtual Connection Service（EVCS）

- Per-VLAN サービス多重化サービス

Cisco イーサネット リレー サービスの概念は MEF イーサネット仮想専用回線（EVPL）の概念に対応します。Cisco イーサネット ワイヤ サービスの概念は MEF イーサネット専用回線の概念に対応します。Cisco マルチポイント サービスの概念は MEF 透過型 LAN サービスの概念に対応します。Cisco マルチポイント リレー サービスの概念は MEF Ethernet Virtual Connection Service の概念に対応します。UNI は、CE とプロバイダー エッジ（PE）間の境界です。

イーサネット サービスは、サービス プロバイダーが UNI 間で提供するサービスです。

- イーサネット ライン サービス（E-Line）ポイントツーポイント
- イーサネット LAN サービス（E-LAN）マルチポイント
- イーサネット ツリー サービス（E-Tree）ポイントツーマルチポイント

これは、キャリア イーサネットです。これにより、高速化（GigE および 10GigE）などの利点のあるクラウド内のフレーム リレー /ATM を置き換えることができます。VPLS（バーチャル プライベート LAN サービス）は、MPLS ネットワークでマルチポイント イーサネット サービスを提供できるエンドツーエンド アーキテクチャです。このサービスの複数のインスタンスが同じ物理インフラストラクチャを共有するため、「仮想」です。サービスの各インスタンスが互いに独立して分離されるため、「プライベート」です。サブスクライバ間でレイヤ 2 のマルチポイント接続をエミュレートするため、「LAN サービス」です。

IGMP スヌーピング

IGMP スヌーピングは、レイヤ 2 でマルチキャスト トラフィックを抑制する方法を提供します。IGMP スヌーピング アプリケーションは、ブリッジ ドメインのホストによって送信された IGMP メンバーシップ レポートをスヌーピングすることで、レイヤ 2 マルチキャスト転送テーブルを設定して、少なくとも 1 つの関係メンバーを持つポートだけにトラフィックを送信できます。これにより、マルチキャスト トラフィックの量が大幅に削減されます。

IGMP は、レイヤ 3 で設定され、IPv4 マルチキャスト ネットワークのホストが、どのマルチキャスト トラフィックを対象とするか示すための手段を提供し、また、ルータがネットワーク（レイヤ 3）内のマルチキャスト トラフィックのフローを制御および制限するための手段を提供します。

IGMP スヌーピングは、レイヤ 2 の IP マルチキャスト トラフィックを制限するための、IGMP メンバーシップ レポート メッセージの情報を使用して、転送テーブルに対応する情報を構築します。転送テーブルのエントリは <ルート, OIF リスト> という形式です。

- ルートは <*,G> ルートまたは <S,G> ルートです。
- OIF リストは、指定されたルートと、ブリッジ ドメイン内のすべてのマルチキャスト ルータ（mrouter）ポートに関する IGMP メンバーシップ レポートを送信したすべてのブリッジ ポートで構成されます。

IGMP スヌーピング機能により、マルチキャスト ネットワークで次の利点が得られます。

- 基本的な IGMP スヌーピングは、VPLS ブリッジ ドメイン全体をフラッディングするマルチキャスト トラフィックを削減することで、帯域幅の使用量を減らします。
- オプションの設定オプションを使用すると、IGMP スヌーピングは、1 つのブリッジ ポートでホストから受信された IGMP レポートをフィルタリングし、他のブリッジ ポートでホストへの漏出を防止することで、ブリッジ ドメイン間のセキュリティを確保できます。
- オプションの設定オプションを使用すると、IGMP スヌーピングは、IGMP メンバーシップ レポート（IGMPv2）を抑制することで、またはアップストリーム IP マルチキャスト ルータへの IGMP プロキシ レポーター（IGMPv3）として動作することで、アップストリーム IP マルチキャスト ルータへのトラフィックの影響を低減できます。

IGMP スヌーピングの設定方法については、『*Cisco ASR 9000 Series Aggregation Services Router Multicast Configuration Guide*』の「*Implementing Layer 2 Multicast with IGMP Snooping*」モジュールを参照してください。

適用できる IGMP スヌーピング コマンドは『*Cisco ASR 9000 Series Aggregation Services Router Multicast Command Reference*』で説明します。

IP インターワーキング

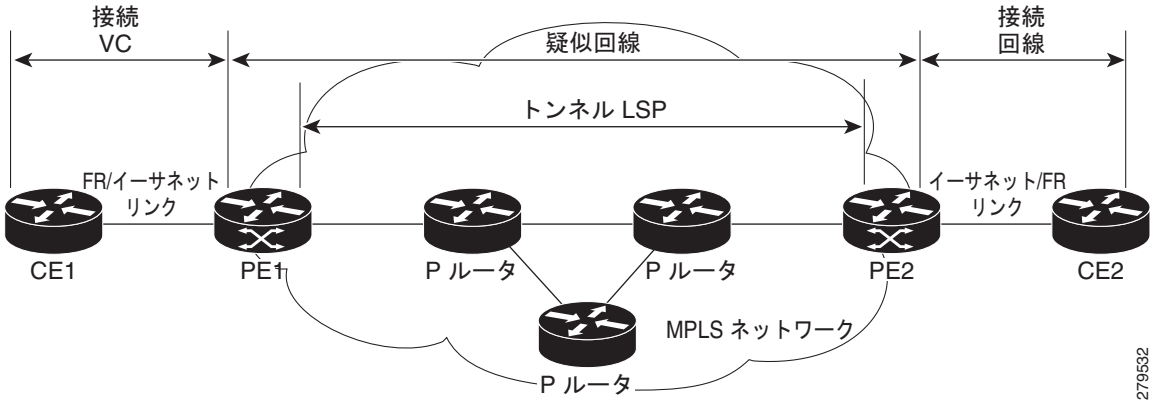
カスタマー環境では、ソリューションによりネットワーク終端で異種転送を使用する AToM をサポートする必要があります。このソリューションには、1 つのカスタマー エッジ（CE）デバイスの転送を別の転送に変換する機能（たとえば、フレーム リレーからイーサネットなど）が必要です。Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 および Cisco ASR 9000 シリーズ イーサネット ラインカードにより、Cisco ASR 9000 シリーズ ルータで複数のレガシー サービスをサポートできます。

IP インターワーキングは、IP/MPLS バックボーン上でレイヤ 2 トラフィックを転送するためのソリューションです。IP インターワーキングは、AToM トンネルを使用するイーサネット、フレーム リレーなど、多くのタイプのレイヤ 2 フレームに対応します。IP インターワーキングは、プロバイダー エッジ（PE）ルータでパケットをカプセル化し、それらをバックボーンを介してクラウドの反対側の PE ルータに転送し、カプセル化を削除し、それらを宛先に転送します。トランスポート層では、一方の側でイーサネットを使用し、もう一方の側でフレーム リレーを使用できます。IP インターワーキングは、AToM トンネルの異種エンドポイント間で実行されます。

（注） MPLS とローカル接続のシナリオでは、イーサネットとフレーム リレー ベースのネットワーク間でルーテッド インターワーキングのみサポートされます。

図 7 は、イーサネット接続 VC とフレーム リレー接続 VC 間の相互運用性を表しています。

図 7　　IP インターワーキング over MPLS コア



接続回線は（AC）は、CE デバイスを PE デバイスに接続する物理的または論理的なポートまたは回線です。疑似回線（PW）は、2 つの AC を接続する双方向仮想接続（VC）です。MPLS ネットワークでは、PW は LSP トンネル内で伝送されます。PE1 および PE2 のコア方向のラインカードとして、Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 または Cisco ASR 9000 シリーズ イーサネット ラインカードが使用可能です。

IP インターワーキング モードでは、入力 PE で受信されたパケットからレイヤ 2（L2）ヘッダーが削除され、IP ペイロードだけが出力 PE に送信されます。出力 PE では、パケットが出力ポートから送信される前に、L2 ヘッダーが付加されます。

図 7 では、CE1 および CE2 を、フレーム リレー（FR）インターフェイスまたはギガビット イーサネット（GigE）インターフェイスにすることができます。CE1 が FR で、CE2 が GigE または dot1q、あるいは QinQ であるとします。イーサネット CE（CE2）から着信するパケットの場合、CE 方向の PE（PE2）の入力 LC は、L2 フレーミングを削除し、そのパケットを、疑似回線上で IPoMPLS カプセル化を使用して出力 PE（PE1）に転送します。出力 PE のコア方向のラインカードは、MPLS ラベルを削除しますが、制御ワードを保持し、それを FR CE（CE1）方向の出力ラインカードに伝送します。FR PE では、ラベル ディスポジション後、レイヤ 3（L3）パケットは FR 上でカプセル化されます。

同様に、FR CE から着信した IP パケットは疑似回線上で IPoMPLS カプセル化に変換されます。コアから着信するパケットは IP ペイロードのみを伝送するため、イーサネット PE 側では、ラベル ディスポジション後、PE は、パケットを CE に伝送する前に、そのパケットに L2 イーサネット パケット ヘッダーを追加して戻します。

これらのモードは、AToM で IP インターワーキングをサポートします。

- イーサネットとフレーム リレー

  イーサネット CE デバイスから着信するパケットには、MAC（ポート モード、タグなし、シングル タグ、ダブル タグ）、IPv4 ヘッダー、およびデータが含まれます。イーサネット ラインカードは L2 フレーミングを削除し、その後、出力ラインカードに L3 パケットを転送します。出力ラインカードは、出力ポートからパケットを送信する前に、FR L2 ヘッダーを追加します。

- Ethernet to Ethernet

  CE デバイスは両方ともイーサネットです。各イーサネット インターフェイスは、ポート モード、タグなし、シングル タグ、またはダブル タグにすることができます。ただし、これは IP インターワーキングの一般的なシナリオではありません。

AToM iMSG

この機能により、アクセス ネットワーク内のインターワーキング レイヤですべての非イーサネット機能を終了し、これらの接続を、レイヤ 3 エッジ ルータで終端可能なイーサネット セントリック サービスに変換することができます。現在は、時分割多重（TDM）ベースのサービスはレイヤ 3 エッジ ルータ上で直接終端しています。L3 ネットワークの簡素でより低コストなモデルは、TDM の複雑さをアクセス レイヤに移動することによってイネーブルになります。

レイヤ 2 カプセル化は、入力ラインカード側の入力 PE の接続回線によって IP パケットから削除されます。MPLS カプセル化された IP パケットのペイロードは、ファブリックで出力ラインカード側のコアに送信されます。出力ラインカードは MPLS コアを介してパケットを送信します。リモート PE では、MPLS ラベルが削除され、出力 AC のレイヤ 2 ヘッダーが追加されて、パケットは最終的に接続された CE に送信されます。L2VPN VPWS は、次をサポートするように拡張されました。

- ポイントツーポイント プロトコル（PPP）
- High-Level Data Link Control（HDLC; ハイレベル データリンク コントロール）
- マルチリンク ポイントツーポイント プロトコル（MLPPP）
- すべてのカプセル化タイプの QoS サポート

  QoS の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Configuration Guide*』を参照してください。

TDM AC は、次の SPA で設定できます。

- SPA-8XCHT1/E1
- SPA-4XCT3/DS0
- SPA-1XCHSTM1/OC3
- SPA-2XCHOC12/DS0
- SPA-1XCHOC48/DS3
- SPA-4XT3/E3
- SPA-4XOC3-POS-V2
- SPA-8XOC3-POS
- SPA-8XOC12-POS
- SPA-1XOC48POS/RPR
- SPA-2XOC48POS/RPR

Any Transport over MPLS

Any Transport over MPLS（AToM）は、マルチプロトコル ラベル スイッチング（MPLS）バックボーン上でレイヤ 2 パケットを転送します。これにより、サービス プロバイダーは、単一の統合されたパケット ベース ネットワーク インフラストラクチャを使用することで、既存のレイヤ 2 ネットワークとカスタマー サイトを接続できます。この機能を使用すると、サービス プロバイダーは、別々のネットワークを使用する代わりに、MPLS バックボーン上でレイヤ 2 接続を提供できます。

AToM は、入力 PE ルータでレイヤ 2 フレームをカプセル化し、2 つの PE ルータ間を接続する疑似回線の反対側に位置する対応した PE ルータにそれらを送信します。出力 PE はカプセル化を削除し、レイヤ 2 フレームを送信します。

PE ルータ間でレイヤ 2 フレームを正常に転送するには、PE ルータを設定する必要があります。ルータ間で、疑似回線と呼ばれる接続を設定します。各 PE ルータで次の情報を指定します。

- イーサネット、フレーム リレーなどの疑似回線で転送されるレイヤ 2 データのタイプ。
- PE ルータが通信できる、ピア PE ルータのループバック インターフェイスの IP アドレス
- 疑似回線を識別するピア PE の IP アドレスと VC ID の一意の組み合せ

High-Level Data Link Control over MPLS

接続回線（AC）は、HDLC カプセル化が設定されたメイン インターフェイスです。AC との間のパケットは、MPLS コア ネットワーク上の他のプロバイダー エッジ（PE）との間の、VC タイプ 0x6 の疑似回線（PW）を使用して転送されます。

HDLC over MPLS では、HDLC パケット全体が転送されます。入力 PE ルータは、HDLC フラグおよび FCS ビットだけを削除します。

PPP over MPLS

接続回線（AC）は、PPP カプセル化が設定されたメイン インターフェイスです。AC との間で送受信されるパケットは、MPLS コア ネットワーク上の他のプロバイダー エッジ（PE）との間で、VC タイプ 0x7 の AToM PW を介して転送されます。

PPP over MPLS の場合、入力 PE ルータはフラグ、アドレス、制御フィールド、および FCS ビットを削除します。

Frame Relay over MPLS

Frame Relay over MPLS（FRoMPLS）は、2 つのフレーム リレー アイランド間の専用回線タイプの接続を提供します。フレーム リレー トラフィックは MPLS ネットワーク上で転送されます。

（注） データリンク接続識別子（DLCI）の DCLI-DLCI モードがサポートされます。追加の制御情報を伝えるために、制御ワード（DLCI-DLCI モードに必要）が使用されます。

プロバイダー エッジ（PE）ルータは、加入者サイトからフレーム リレー プロトコル パケットを受信すると、フレーム リレー ヘッダーおよびフレーム チェック シーケンス（FCS）を削除し、関連する仮想回線（VC）ラベルを付けます。削除された逆方向明示的輻輳通知（BECN）、順方向明示的輻輳通知（FECN）、廃棄適性（DE）、およびコマンド/応答（C/R）ビットが制御ワードを使用して個別に送信されます（DLCI-DLCI モードの場合）。

MPLS トランスポート プロファイル

MPLS トランスポート プロファイル（MPLS-TP）トンネルは、IP および MPLS トラフィックが通過する転送ネットワーク サービス レイヤを提供します。MPLS-TP 環境内では、疑似回線（PW）は MPLS-TP トンネルを転送メカニズムとして使用します。MPLS-TP トンネルは、SONET/SDH TDM テクノロジーからパケット スイッチングへの移行に役立つとともに、サービスの高帯域幅での使用と低コスト化をサポートします。転送ネットワークは、接続指向型で静的にプロビジョニングされ、寿命の長い接続を持ちます。通常、転送ネットワークは、ラベルなどの ID を変更する制御プロトコルを回避します。MPLS-TP トンネルは、静的にプロビジョニングされた双方向ラベル スイッチド パス（LSP）を介してこの機能を提供します。

MPLS トランスポート プロファイルの設定方法の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router MPLS Configuration Guide*』を参照してください。

MPLS-TP は、次のスタティックおよびダイナミックなマルチセグメント疑似回線の組み合わせをサポートします。

- スタティック - スタティック
- スタティック - ダイナミック
- ダイナミック - スタティック
- ダイナミック - ダイナミック

MPLS-TP は、次のスタティックおよびダイナミック疑似回線の組み合わせで 1 対 1 L2VPN 疑似回線冗長性をサポートします。

- スタティック疑似回線とスタティック バックアップ疑似回線
- スタティック疑似回線とダイナミック バックアップ疑似回線
- ダイナミック疑似回線とスタティック バックアップ疑似回線
- ダイナミック疑似回線とダイナミック バックアップ疑似回線

既存の TE 優先パス機能は、PW を MPLS-TP 転送トンネルにピン ダウンするために使用します。優先トンネル パスの設定の詳細については、優先トンネル パスの設定（P.LSC-151）を参照してください。ダイナミック疑似回線では、PW ステータスは LDP によって交換されますが、スタティック PW では、ステータスは PW OAM メッセージに転送されます。PW ステータス OAM の設定の詳細については、PW ステータス OAM の設定（P.LSC-153）を参照してください。デフォルトでは、PW を伝送する MPLS TP トンネルのステートの変化によって PW のステートが変化する場合、アラームは生成されません。

Circuit Emulation Over Packet Switched Network

Circuit Emulation over Packet（CEoP）は、パケット スイッチド ネットワークで TDM 回線を伝送する方法です。CEoP は物理接続に似ています。CEoP の目的は、専用回線およびレガシー TDM ネットワークを置き換えることです（図 8）。

CEoP は主に次の 2 つのモードで動作します。

- SAToP（Structure Agnostic TDM over Packet）と呼ばれる非構造化モード

  SAToP は、非フレーム化 E1、T1、E3 および T3 などの構造にとらわれない転送だけをアドレス指定します。これにより、すべての TDM サービスはビット ストリームに分割され、PW トンネルでの送信用にカプセル化されます。このプロトコルは、TDM トラフィック データおよび同期タイミング情報を透過的に送信できます。SAToP は完全に構造を無視するため、プロバイダー エッジ（PE）ルータは、TDM データを解釈したり TDM シグナリングに参加したりする必要がありません。このプロトコルは PDH ビットストリームを透過的に送信するための簡単な方法です。

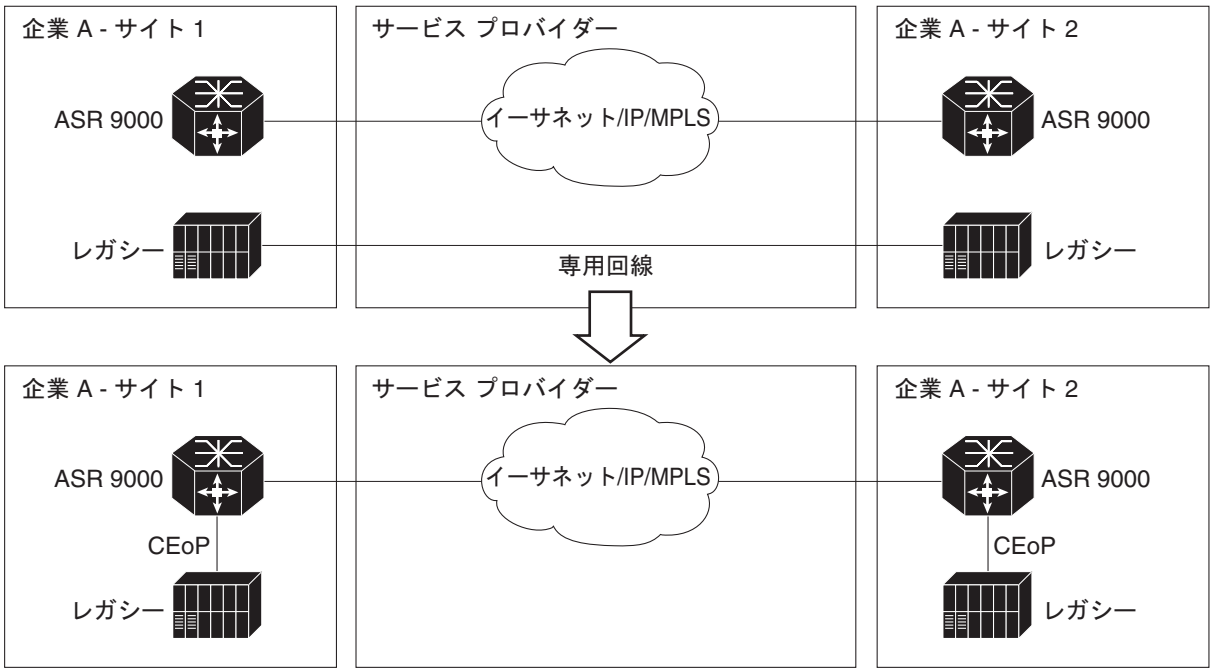
- CESoPSN（Circuit Emulation Service over Packet Switched Network）という名前の構造化モード

  SAToP と違い、CESoPSN は、エミュレートされた構造化 TDM 信号を送信します。つまり、TDM フレームのフレーム構造を識別して処理し、シグナリングを送信できます。これはアイドル タイムスロット チャネルを送信しない場合がありますが、E1 トラフィック ストリームから CE デバイスの有用なタイムスロットのみを抽出し、伝送用に PW パケットにカプセル化します。CEoP SPA は、ハーフハイト（HH）の共有ポート アダプタ（SPA）です。CEoP SPA ファミリは、非構造化 / 構造化（NxDS0）クォータ レート、ハーフ ハイト SPA である 24xT1/E1、2xT3/E3、および 1xOC3/STM1 で構成されます。

CEM 機能は、CEoP SPA を持つ Engine 5 ラインカードでのみサポートされています。CEM は、次でサポートされています。

- 1 ポート チャネライズド OC3 STM1 ATM CEoP SPA（SPA-1CHOC3-CE-ATM）

図 8　Circuit Emulation over Packet を使用した企業データのコンバージェンス



CESoPSN および SAToP は、基礎となる転送メカニズムとして MPLS、UDP/IP、および L2TPv3 を使用できます。このリリースでは、MPLS 転送メカニズムだけをサポートしています。

CEoP SPA は次の動作モードをサポートしています。

- 回線エミュレーション モード（CEM）
- ATM モード
- IMA モード

（注） サポートされるのは CEM モードだけです。

Circuit Emulation over Packet Switched Network の利点

CEM はサービス プロバイダーとエンド ユーザに次の利点を提供します。

- 機器の設置のコスト削減。
- ネットワーク運用のコストを削減します。高価な専用回線で、コストを節約するモードだけにアクセスを制限する必要がなくなります。
- メンテナンスが必要なのはコア ネットワークだけのため、メンテナンス コストを抑制できます。
- 投資をアクセス ネットワーク全体にとどめたまま、パケット スイッチド ネットワークでコア ネットワークのリソースをより効率的に利用できます。
- エンド ユーザにより安価なサービスを提供できます。

L2VPN ノンストップ ルーティング

L2VPN ノンストップ ルーティング（NSR）機能により、プロセス障害（クラッシュ）やルート プロセッサ フェールオーバー（RP FO）などの、イベントのフラッピングによるラベル配布パス（LDP）セッションを回避できます。NSR プロセス障害スイッチオーバーを使用して NSR をイネーブルにした場合、RP FO を実行することによって、プロセス障害（クラッシュ）での NSR がサポートされます。

NSR は、障害が発生したルータについて、グレースフル リスタート（GR）なしでコントロール プレーン ステートを維持できます。NSR は、定義上、プロトコル拡張の必要がないため、通常はステートフル スイッチ オーバー（SSO）を使用してコントロール プレーン ステートを維持します。

ポイントツーポイント レイヤ 2 サービスを実装する方法

ここでは、L2VPN を実装するために必要なタスクについて説明します。

L2VPN のインターフェイスまたは接続の設定

L2VPN のインターフェイスまたは接続を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **l2transport**
4. **exit**
5. **interface** *type interface-path-id*
6. **end**
   または
   commit
7. **show interface** *type interface-id*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0`	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、インターフェイスを設定します。
ステップ 3	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	選択したインターフェイスで L2 転送をイネーブルにします。
ステップ 4	**exit** **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# exit`	現在のコンフィギュレーション モードを終了します。
ステップ 5	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0`	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、インターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 7	**show interface** *type interface-id* **例：** `RP/0/RSP0/CPU0:router# show interface gigabitethernet 0/0/0/0`	（任意）コミットしたインターフェイスの設定を表示します。

ローカル スイッチングの設定

ローカル スイッチングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **interface** *type interface-path-id*
6. **interface** *type interface-path-id*
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface TenGigE 0/7/0/6.5`	インターフェイス タイプ ID を指定します。選択できる基準は、次のとおりです。 • GigabitEthernet：ギガビット イーサネット/IEEE 802.3 インターフェイス • TenGigE：TenGigabit イーサネット/IEEE 802.3 インターフェイス • CEM：回線エミュレーション インターフェイス

	コマンドまたはアクション	目的
ステップ 6	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface GigabitEthernet0/4/0/30`	インターフェイス タイプ ID を指定します。選択できる基準は、次のとおりです。 • GigabitEthernet：ギガビット イーサネット/IEEE 802.3 インターフェイス • TenGigE：TenGigabit イーサネット/IEEE 802.3 インターフェイス
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ローカル接続の冗長性の設定

ローカル接続の冗長性を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **backup interface** *type interface-path-id*
6. **interface** *type interface-path-id*
7. **interface** *type interface-path-id*
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**backup interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# backup interface Bundle-Ether 0/7/0/6.5`	ローカル接続の冗長性を設定します。
ステップ 6	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface Bundle-Ether 0/7/0/6.2`	インターフェイス タイプ ID を指定します。選択できる基準は、次のとおりです。 • GigabitEthernet：ギガビット イーサネット/IEEE 802.3 インターフェイス。 • TenGigE：TDR イーサネット/IEEE 802.3 インターフェイス。 • CEM：回線エミュレーション インターフェイス

	コマンドまたはアクション	目的
ステップ 7	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface Bundle-Ether 0/7/0/6.1`	インターフェイス タイプ ID を指定します。選択できる基準は、次のとおりです。 • GigabitEthernet：ギガビット イーサネット/IEEE 802.3 インターフェイス。 • TenGigE：TDR イーサネット/IEEE 802.3 インターフェイス。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

スタティック ポイントツーポイント相互接続の設定

スタティック ポイントツーポイント相互接続を設定するには、次の作業を実行します。

スタティック ポイントツーポイント相互接続を設定する場合、相互接続に関する次の情報を考慮します。

- 相互接続はペアにより一意に識別されます。相互接続名は、グループ内で一意である必要があります。
- セグメント（接続回線または疑似回線）は一意で、1 つの相互接続だけに属することができます。
- スタティック VC のローカル ラベルはグローバルに一意で、1 つの疑似回線だけで使用できます。
- 1 台のルータにつき 16,000 以下の相互接続を設定できます。

（注） スタティック疑似回線接続はシグナリングに LDP を使用しません。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **interface** *type interface-path-id*
6. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
7. **mpls static label local** {*value*} **remote** {*value*}
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。

	コマンドまたはアクション	目的
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface gigabitethernet 0/1/0/9`	インターフェイス タイプとインスタンスを指定します。
ステップ 6	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.2.2.2 pw-id 2000`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。
ステップ 7	**mpls static label local** {*value*} **remote** {*value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# mpls static label local 699 remote 890`	ローカルおよびリモート ラベル ID 値を設定します。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ダイナミック ポイントツーポイント相互接続の設定

ダイナミック ポイントツーポイント相互接続を設定するには、次の作業を実行します。

（注） ダイナミック相互接続では、LDP が稼働中である必要があります。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **interface** *type interface-path-id*
6. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント クロスコネクトの名前を入力します。

	コマンドまたはアクション	目的
ステップ 5	**interface** *type interface-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface GigabitEthernet0/0/0/0.1`	インターフェイス タイプ ID を指定します。選択できる基準は、次のとおりです。 • GigabitEthernet：GigabitEthernet/IEEE 802.3 インターフェイス。 • TenGigE：TenGigabitEthernet/IEEE 802.3 インターフェイス。 • CEM：回線エミュレーション インターフェイス
ステップ 6	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.2.2.2 pw-id 2000`	クロスコネクトの疑似回線セグメントを設定します。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

Inter-AS の設定

Inter-AS の設定手順は、L2VPN 相互接続の設定作業と同じです（「スタティック ポイントツーポイント相互接続の設定」（P.MPC-130）および「ダイナミック ポイントツーポイント相互接続の設定」（P.MPC-132）を参照）。ただし、相互接続設定で使用されるリモート PE の IP アドレスが iBGP ピアリングを通じて到達可能であることを除きます。

（注） この設定を完了するには、IBGP、EBGP、および ASBR の用語および設定に関する知識が必要です。

L2VPN Quality of Service の設定

ここでは、ポート モードおよび VLAN モードで L2VPN Quality of Service（QoS）を設定する方法について説明します。

制約事項

l2transport コマンドは任意の IP アドレス、L3、または CDP の設定では使用できません。

ポート モードでの L2VPN Quality of Service ポリシーの設定

この手順では、ポート モードでの L2VPN QoS ポリシーの設定方法について説明します。

（注） ポート モードでは、インターフェイス名の形式に、サブインターフェイス番号が含まれません（たとえば、GigabitEthernet0/1/0/1）。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **l2transport**
4. **service-policy** *[***input** | **output***] [policy-map-name]*
5. **end**
   または
   commit
6. **show qos interface** *type interface-path-id* **service-policy** [**input** | **output**] [*policy-map-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0`	インターフェイス接続回線を指定します。
ステップ 3	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	L2 スイッチングのインターフェイスまたは接続を設定します。

	コマンドまたはアクション	目的
ステップ 4	**service-policy** [**input** | **output**] [*policy-map-name*] **例：** RP/0/RSP0/CPU0:router(config-if)# service-policy input servpol1	入力インターフェイスまたは出力インターフェイスに、そのインターフェイスのサービス ポリシーとして使用する QoS ポリシーを付加します。
ステップ 5	**end** または **commit** **例：** RP/0/RSP0/CPU0:router(config-if)# end または RP/0/RSP0/CPU0:router(config-if)# commit	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 6	**show qos interface** *type interface-id* service-policy [input | output] [*policy-map-name*] **例：** RP/0/RSP0/CPU0:router# show qos interface gigabitethernet 0/0/0/0 input serpol1	（任意）定義した QoS サービス ポリシーを表示します。

VLAN モードでの L2VPN Quality of Service ポリシーの設定

この手順では、VLAN モードでの L2VPN QoS ポリシーの設定方法について説明します。

（注） VLAN モードでは、インターフェイス名にサブインターフェイスを含める必要があります。次に例を示します。
`GigabitEthernet 0`/1/0/1.1
l2transport コマンドは、同じ CLI 行のインターフェイス タイプに従う必要があります。次に例を示します。
interface `GigabitEthernet 0`/0/0/0.1 l2transport

手順の概要

1. **configure**
2. **interface** *type interface-path-id.subinterface* **l2transport**
3. **service-policy** [**input** | **output**] [*policy-map-name*]
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id.subinterface* **l2transport** **例：** `RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0.1 l2transport`	L2 スイッチングのインターフェイスまたは接続を設定します。 **（注）** VLAN モードでは、interface と同じ行に **l2transport** キーワードを入力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 3	**service-policy** *[***input** | **output***] [policy-map-name]* **例：** `RP/0/RP0/CPU0:router(config-if)# service-policy input servpol1`	入力インターフェイスまたは出力インターフェイスに、そのインターフェイスのサービス ポリシーとして使用する QoS ポリシーを付加します。
ステップ 4	**end** または **commit** **例：** `RP/0/RP0/CPU0:router(config-if)# end` または `RP/0/RP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

マルチセグメント疑似回線の設定

ここで説明する作業は、次のとおりです。

マルチセグメント疑似回線設定のプロビジョニング

ポイントツーポイント（p2p）相互接続としてマルチセグメント疑似回線を設定します。P2P 相互接続の詳細については、「スタティック ポイントツーポイント相互接続の設定」（P.MPC-130）を参照してください。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **neighbor** *A.B.C.D* **pw-id** *value*
6. **pw-class** *class-name*
7. **exit**
8. **neighbor** *A.B.C.D* **pw-id** *value*
9. **pw-class** *class-name*
10. **commit**

手順の詳細

	コマンド	目的
ステップ 1	`configure` **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	`l2vpn` **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	レイヤ 2 VPN コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 3	**xconnect group** *group-name* **例：** RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group MS-PW1	自由形式の 32 文字ストリングを使用して、相互接続グループ名を設定します。
ステップ 4	**p2p** *xconnect-name* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p ms-pw1	P2P コンフィギュレーション サブモードを開始します。
ステップ 5	**neighbor** *A.B.C.D* **pw-id** *value* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.165.200.25 pw-id 100	相互接続の疑似回線を設定します。 IP アドレスは、該当する PE ノードの IP アドレスです。 **pw-id** は PE ノードの **pw-id** と一致する必要があります。
ステップ 6	**pw-class** *class-name* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 7	**exit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# exit	疑似回線クラス サブモードを終了し、ルータを親コンフィギュレーション モードに戻します。
ステップ 8	**neighbor** *A.B.C.D* **pw-id** *value* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.165.202.158 pw-id 300	相互接続の疑似回線を設定します。 IP アドレスは、該当する PE ノードの IP アドレスです。 **pw-id** は PE ノードの **pw-id** と一致する必要があります。
ステップ 9	**pw-class** *class-name* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 10	**commit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit	実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを続行します。

グローバル マルチセグメント疑似回線のディスクリプションのプロビジョニング

S-PE ノードには、疑似回線切り替えポイントの Type-Length-Value（TLV）でディスクリプションが必要です。TLV は疑似回線が通過するすべてのスイッチング ポイントを記録し、トラブルシューティングのために便利な履歴を作成します。

各マルチセグメント疑似回線に独自のディスクリプションを設定できます。手順については、「相互接続のディスクリプションのプロビジョニング」（P.MPC-141）を参照してください。独自のディスクリプションがない場合、このグローバルなディスクリプションが使用されます。

手順の概要

1. **configure**
2. **l2vpn**
3. **description** *value*
4. **commit**

手順の詳細

	コマンド	目的
ステップ 1	`configure` **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	`l2vpn` **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	レイヤ 2 VPN 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	`description` *value* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# description S-PE1`	疑似回線切り替えポイント TLV を設定します。この TLV は、疑似回線が通過するすべてのスイッチング ポイントを記録します。 各マルチセグメント疑似回線に独自のディスクリプションを設定できます。独自のディスクリプションがない場合、このグローバルなディスクリプションが使用されます。
ステップ 4	`commit` **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# commit`	実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを続行します。

相互接続のディスクリプションのプロビジョニング

S-PE ノードには、疑似回線切り替えポイントの TLV でディスクリプションが必要です。TLV は疑似回線が通過するすべてのスイッチング ポイントを記録し、トラブルシューティングのために便利な履歴を作成します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **description** *value*
6. **commit**

手順の詳細

	コマンド	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	レイヤ 2 VPN 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	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group MS-PW1`	自由形式の 32 文字ストリングを使用して、相互接続グループ名を設定します。
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p ms-pw1`	P2P コンフィギュレーション サブモードを開始します。

	コマンド	目的
ステップ 5	**description** *value* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# description MS-PW from T-PE1 to T-PE2	疑似回線切り替えポイント TLV を設定します。この TLV は、疑似回線が通過するすべてのスイッチング ポイントを記録します。 各マルチセグメント疑似回線に独自のディスクリプションを設定できます。独自のディスクリプションがない場合、グローバルなディスクリプションが使用されます。詳細については、「マルチセグメント疑似回線設定のプロビジョニング」（P.MPC-138）を参照してください。
ステップ 6	**commit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# commit	実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを続行します。

スイッチング ポイント TLV セキュリティのプロビジョニング

セキュリティ上の理由から、TLV を非表示にでき、それにより、疑似回線が通過するすべてのスイッチングポイントを誰かが表示することを防ぐことができます。

仮想回線接続性検証（VCCV）は、**switching-tlv** パラメータが「hide」に設定されたマルチセグメント疑似回線では機能しない場合があります。VCCV の詳細については、「L2VPN での仮想回線接続検証」（P.MPC-108）を参照してください。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-class** *class-name*
4. **encapsulation mpls**
5. **protocol ldp**
6. **switching-tlv hide**
7. **commit**

手順の詳細

	コマンド	目的
ステップ 1	`configure` **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	`l2vpn` **例：** `RP/0/RSP0/CPU0:router (config)# l2vpn`	レイヤ 2 VPN 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	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# pw-class dynamic_mpls`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 4	`encapsulation mpls` **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。
ステップ 5	`protocol ldp` **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-pwc-encap-mpls)# protocol ldp`	LDP に疑似回線シグナリング プロトコルを設定します。

	コマンド	目的
ステップ 6	**switching-tlv hide** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-pwc-encap-mpls)# switching-tlv hide`	疑似回線 TLV を非表示に設定します。
ステップ 7	**commit** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-pwc-encap-mpls)# commit`	実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを続行します。

マルチセグメント疑似回線のイネーブル化

pw-status コマンドをイネーブルにした後、**pw-status** コマンドを使用します。**pw-status** コマンドはデフォルトではディセーブルです。**pw-status** コマンドを変更すると、L2VPN で設定されたすべての疑似回線が再プロビジョニングされます。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-status**
4. **commit**

手順の詳細

	コマンド	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router (config)# l2vpn`	レイヤ 2 VPN コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 3	`pw-status` **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# pw-status`	このレイヤ 2 VPN で設定されるすべての疑似回線をイネーブルにします。 **（注）** 疑似回線ステータスをディセーブルにするには、**pw-status disable** コマンドを使用します。
ステップ 4	`commit` **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# commit`	実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを続行します。

疑似回線の冗長性の設定

疑似回線の冗長性により、プライマリ疑似回線で障害が発生した場合のバックアップ疑似回線を設定できます。プライマリ疑似回線が障害になった場合、PE ルータをバックアップ疑似回線に切り替えることができます。復旧後にプライマリ疑似回線の運用が再開するように選択できます。

次のトピックでは、疑似回線の冗長性を設定する方法について説明します。

- バックアップ疑似回線の設定（P.LSC-145）
- ポイントツーポイント疑似回線の冗長性の設定（P.LSC-148）
- バックアップ疑似回線への強制的な手動切り替え（P.LSC-150）

バックアップ疑似回線の設定

ポイントツーポイント ネイバーのバックアップ疑似回線を設定するには、次の作業を実行します。

（注） プライマリ疑似回線を再プロビジョニングすると、2 秒でトラフィック再開されます。ただし、バックアップ疑似回線を再プロビジョニングすると、45 ～ 60 秒の遅延後にトラフィックが再開されます。これは想定されている動作です。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** {*xconnect-name*}
5. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
6. **backup** {**neighbor** *A.B.C.D*} {**pw-id** *value*}
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードに入ります。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/O/RSP0/CPU0:router(config-l2vpn)# xconnect group A` `RP/0/RSP0/CPU0:router(config-l2vpn-xc)#`	クロスコネクト グループの名前を入力します。
ステップ 4	**p2p** {*xconnect-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xc1` `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)#`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.1.1.2 pw-id 2`	クロスコネクトの疑似回線セグメントを設定します。

	コマンドまたはアクション	目的
ステップ 6	**backup** {neighbor *A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# backup neighbor 10.2.2.2 pw-id 5` `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)#`	相互接続のバックアップ疑似回線を設定します。 • **neighbor** キーワードを使用して、相互接続するピアを指定します。IP アドレス引数（*A.B.C.D*）は、ピアの IPv4 アドレスです。 • **pw-id** キーワードを使用して、疑似回線 ID を設定します。範囲は 1 ～ 4294967295 です。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ポイントツーポイント疑似回線の冗長性の設定

バックアップ遅延のためにポイントツーポイント疑似回線の冗長性を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-class** {***class-name***}
4. **backup disable** {**delay** *value* | **never**}
5. **exit**
6. **xconnect group** *group-name*
7. **p2p** {*xconnect-name*}
8. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
9. **pw-class** {*class-name*}
10. **backup** {**neighbor** *A.B.C.D*} {**pw-id** *value*}
11. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードに入ります。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**pw-class** {class-*name*} **例：** `RP/O/RSP0/CPU0:router(config-l2vpn)# pw-class path1` `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)#`	疑似回線クラス名を設定します。

	コマンドまたはアクション	目的
ステップ 4	**backup disable** {**delay** *value* | **never**} **例：** RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# backup disable delay 20	このコマンドは、プライマリ疑似回線がアクティブになった後、バックアップ疑似回線から引き継ぐまでの待ち時間を指定します。 • **delay** キーワードを使用して、プライマリ疑似回線がアップ状態になってから、セカンダリ疑似回線が非アクティブになるまでの経過秒数を指定します。範囲は 0 ～ 180 です。 • プライマリ疑似回線が再び使用できるようになった場合に、セカンダリ疑似回線で障害が発生しない限り、セカンダリ疑似回線からプライマリ疑似回線にフォール バックしないように指定するには、**never** キーワードを使用します。
ステップ 5	**exit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# exit RP/O/RSP0/CPU0:router(config-l2vpn)#	現在のコンフィギュレーション モードを終了します。
ステップ 6	**xconnect group** *group-name* **例：** RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group A RP/0/RSP0/CPU0:router(config-l2vpn-xc)#	クロスコネクト グループの名前を入力します。
ステップ 7	**p2p** {*xconnect-name*} **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xc1 RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)#	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 8	**neighbor** {*A.B.C.D*} {**pw-id** ***value***} **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.1.1.2 pw-id 2 RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)#	クロスコネクトの疑似回線セグメントを設定します。
ステップ 9	**pw-class** {class-***name***} **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class path1	疑似回線クラス名を設定します。

	コマンドまたはアクション	目的
ステップ 10	**backup** {**neighbor** *A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# backup neighbor 10.2.2.2 pw-id 5` `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)#`	相互接続のバックアップ疑似回線を設定します。 • **neighbor** キーワードを使用して、相互接続するピアを指定します。A.B.C.D 引数はピアの IPv4 アドレスです。 • **pw-id** キーワードを使用して、疑似回線 ID を設定します。範囲は 1 ～ 4294967295 です。
ステップ 11	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

バックアップ疑似回線への強制的な手動切り替え

ルータを強制的にバックアップに切り替える、またはプライマリ疑似回線に戻すには、EXEC モードで **l2vpn switchover** コマンドを使用します。

手動切り替えは、コマンドが入力されたとき、コマンドで指定されたピアが実際に使用可能であり、相互接続が完全なアクティブ状態に移行する場合に限り実行されます。

優先トンネル パスの設定

この手順では、優先トンネル パスを設定する方法について説明します。

（注） 優先パスの設定に使用されるトンネルは、MPLS トラフィック エンジニアリング（MPLS-TE）トンネルです。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-class** {*name*}
4. **encapsulation mpls**
5. **preferred-path** {**interface**} {**tunnel-ip** *value* | **tunnel-te** *value* | **tunnel-tp** *value*} [**fallback disable**]
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**pw-class** {*name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class path1`	疑似回線クラス名を設定します。
ステップ 4	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。

	コマンドまたはアクション	目的
ステップ 5	**preferred-path** {**interface**} {**tunnel-ip** *value* | **tunnel-te** *value* | **tunnel-tp** *value*} [**fallback disable**] **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel-te 11 fallback disable`	優先パス トンネルを設定します。フォールバックのディセーブル化の設定が使用されており、優先パスとして設定されている TE/TP トンネルがダウン状態になると、対応する疑似回線もダウン状態になることがあります。 **（注）** フォールバックがサポートされていることを確認します。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PW ステータス OAM の設定

疑似回線ステータス OAM を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-oam refresh transmit** *seconds*
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN 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	**pw-oam refresh transmit** *seconds* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-oam refresh transmit 100`	疑似回線 OAM 機能をイネーブルにします。 **（注）** リフレッシュの送信間隔範囲は 1 ～ 40 秒です。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

フローベースのロード バランシングのイネーブル化

フローベースのロード バランシングをイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **load-balancing flow** {**src-dst-mac** | **src-dst-ip**}
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**load-balancing flow** {**src-dst-mac** | **src-dst-ip**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# load-balancing flow src-dst-ip`	L2VPN 下のすべての疑似回線およびバンドル EFP に対しフロー ベースのロード バランシングをイネーブルにします。ただし、疑似回線クラスを通じて疑似回線に対して、および EFP-hash を通じてバンドルに対して明示的に指定されている場合は除きます。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線クラスのフローベースのロード バランシングのイネーブル化

疑似回線クラスに対しフローベースのロード バランシングをイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**

3. **pw-class** {*name*}
4. **encapsulation mpls**
5. **load-balancing pw-label**
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**pw-class** {*name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class path1`	疑似回線クラス名を設定します。
ステップ 4	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。

	コマンドまたはアクション	目的
ステップ 5	**load-balancing pw-label** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# load-balancing pw-label`	仮想回線ベースのロード バランシングを使用するために、定義されたクラスを使用してすべての疑似回線をイネーブルにします。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線のグループ化のイネーブル化

疑似回線のグループ化をイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-grouping**
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードに入ります。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN 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	**pw-grouping** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-grouping`	疑似回線のグループ化をイネーブルにします。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

マルチキャスト接続の設定

『*Cisco ASR 9000 Series Aggregation Services Router Multicast Configuration Guide*』の「*Implementing Multicast Routing on Cisco ASR 9000 Series Aggregation Services Routers*」モジュールおよび『*Multicast Routing and Forwarding Commands on Cisco ASR 9000 Series Aggregation Services Routers*』の「*Multicast Routing and Forwarding Commands on Cisco ASR 9000 Series Aggregation Services Routers*」モジュールを参照してください。

手順の概要

1. **configure**
2. **multicast-routing**
3. **address-family ipv4**
4. **nsf**
5. **interface all enable**
6. **accounting per-prefix**
7. **router pim**
8. **vrf default address-family ipv4**
9. **rp-address**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**multicast-routing** [**address-family ipv4**] **例：** `RP/0/RSP0/CPU0:router(config)# multicast-routing`	マルチキャスト ルーティング コンフィギュレーション モードを開始します。 • 次のマルチキャスト プロセスが起動します：MRIB、MFWD、PIM、および IGMP。 • IPv4 では、IGMP バージョン 3 はデフォルトでイネーブルです。 • IPv4 の場合、**address-family ipv4** キーワードを使用します。
ステップ 3	**interface all enable** **例：** `RP/0/RSP0/CPU0:router(config-mcast-ipv4)# interface all enable`	新規および既存のすべてのインターフェイスでマルチキャスト ルーティングおよび転送をイネーブルにします。
ステップ 4	**exit** **例：** `RP/0/RSP0/CPU0:router(config-mcast-ipv4)# exit`	マルチキャスト ルーティング コンフィギュレーション モードを終了し、ルータを親コンフィギュレーション モードに戻します。

	コマンドまたはアクション	目的
ステップ 5	**router igmp** **例：** `RP/0/RSP0/CPU0:router(config)# router igmp`	（任意）ルータ IGMP 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	**version** {**1** | **2** | **3**} **例：** `RP/0/RSP0/CPU0:router(config-igmp)# version 3`	（任意）ルータ インターフェイスで使用する IGMP バージョンを選択します。 • IGMP のデフォルトはバージョン 3 です。 • ホスト レシーバは、PIM-SSM 動作の IGMPv3 をサポートする必要があります。 • このコマンドがルータ IGMP コンフィギュレーション モードで設定されている場合、パラメータはすべての新規および既存インターフェイスによって継承されます。これらのパラメータは、インターフェイス コンフィギュレーション モードでインターフェイスごとに上書きできます。
ステップ 7	**end** or **commit** **例：** `RP/0/RSP0/CPU0:router(config-igmp)# end` または `RP/0/RSP0/CPU0:router(config-igmp)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 8	**show pim** [**ipv4**] **group-map** [*ip-address-name*] [**info-source**] **例：** `RP/0//CPU0:router# show pim ipv4 group-map`	（任意）グループと PIM モードのマッピングを表示します。
ステップ 9	**show pim** [**vrf** *vrf-name*] [**ipv4**] **topology** [*source-ip-address* [*group-ip-address*] | **entry-flag** *flag* | **interface-flag** | **summary**] [**route-count**] **例：** `RP/0/RSP0/CPU0:router# show pim topology`	（任意）特定のグループまたはすべてのグループの PIM トポロジ テーブル情報を表示します。

AToM IP インターワーキングの設定

AToM IP インターワーキングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **interworking ipv4**
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。

	コマンドまたはアクション	目的
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 6	**end** または **commit** **例：** `RP/0/RP0/CPU0:router(config-if)# end` または `RP/0/RP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PPP IP インターワーキングの設定

PPP IP インターワーキングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **encapsulation ppp**
4. **l2transport**
5. **end**
6. **l2vpn**
7. **xconnect group** *group-name*

8. **p2p** *xconnect-name*
9. **interface** *type interface-path-id*
10. **interface** *type interface-path-id*
11. **interworking ipv4**
12. **interface** *type interface-path-id*
13. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
14. **pw-class** *class-name*
15. **exit**
16. **interworking ipv4**
17. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface Serial0/2/1/0/1/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 3	**encapsulation ppp** **例：** `RP/0/RSP0/CPU0:router(config-if)# encapsulation ppp`	PPP にカプセル化タイプを設定します。
ステップ 4	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	選択したインターフェイスでレイヤ 2 トランスポートをイネーブルにします。
ステップ 5	**end** **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# end`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN 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 7	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。
ステップ 8	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p 1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 9	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface Serial0/2/1/0/1/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 10	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/1.1`	インターフェイス タイプとインスタンスを指定します。
ステップ 11	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 12	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface Serial0/0/0/0/2/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 13	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 120.120.120.120 pw-id 3`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。
ステップ 14	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw)# pw-class class_c1`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。

	コマンドまたはアクション	目的
ステップ 15	**exit** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw)# exit`	現在のコンフィギュレーション モードを終了します。
ステップ 16	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 17	**end** または **commit** **例：** `RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# end` または `RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PPP とイーサネット間の IP インターワーキングの設定

cHDLC IP インターワーキングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **l2transport**
4. **end**
5. **l2vpn**
6. **xconnect group** *group-name*
7. **p2p** *xconnect-name*

8. **interface** *type interface-path-id*
9. **interface** *type interface-path-id*
10. **interworking ipv4**
11. **interface** *type interface-path-id*
12. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
13. **pw-class** *class-name*
14. **exit**
15. **interworking ipv4**
16. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface Serial0/2/1/0/1/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 3	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-if)# l2transport`	選択したインターフェイスでレイヤ 2 トランスポートをイネーブルにします。
ステップ 4	**end** **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# end`	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 6	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。

	コマンドまたはアクション	目的
ステップ 7	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p 1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 8	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface Serial0/2/1/0/1/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 9	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/1.1`	インターフェイス タイプとインスタンスを指定します。
ステップ 10	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 11	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface Serial0/0/0/0/2/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 12	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 120.120.120.120 pw-id 3`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。
ステップ 13	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw)# pw-class class_cem`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 14	**exit** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw)# exit`	現在の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 15	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 16	**end** または **commit** **例：** `RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# end` または `RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

MLPPP IP インターワーキングの設定

cHDLC IP インターワーキングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **multilink** [**fragment** | **interleave** | **ncp**]
4. **l2transport**
5. **end**
6. **l2vpn**
7. **xconnect group** *group-name*
8. **p2p** *xconnect-name*
9. **interface** *type interface-path-id*
10. **interface** *type interface-path-id*
11. **interworking ipv4**

12. **interface** *type interface-path-id*
13. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
14. **pw-class** *class-name*
15. **exit**
16. **interworking ipv4**
17. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface Multilink0/2/1/0/1`	インターフェイス タイプとインスタンスを指定します。
ステップ 3	**multilink** [**fragment**|**interleave**|**ncp**] **例：** `RP/0/RSP0/CPU0:router(config-if)# multilink`	マルチリンク パラメータを変更します。
ステップ 4	**l2transport** **例：** `RP/0/RSP0/CPU0:router(config-if-multilink)# l2transport`	選択したインターフェイスでレイヤ 2 トランスポートをイネーブルにします。
ステップ 5	**end** **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# end`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 7	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。

	コマンドまたはアクション	目的
ステップ 8	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p 1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 9	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface Serial0/2/1/0/1/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 10	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/1.1`	インターフェイス タイプとインスタンスを指定します。
ステップ 11	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 12	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface Serial0/0/0/0/2/1/1:0`	インターフェイス タイプとインスタンスを指定します。
ステップ 13	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 120.120.120.120 pw-id 3`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。
ステップ 14	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw)# pw-class class_cem`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 15	**exit** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw)# exit`	現在の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 16	**interworking ipv4** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interworking ipv4`	P2P で IPv4 インターワーキングを設定します。
ステップ 17	**end** または **commit** **例：** `RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# end` または `RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

Circuit Emulation over Packet Switched Network の設定

CEoP を設定するには、次の作業を実行します。

- CEM 接続回線の疑似回線への追加（P.LSC-172）
- 疑似回線クラスの関連付け（P.LSC-174）
- 疑似回線ステータスのイネーブル化（P.LSC-177）
- バックアップ疑似回線の設定（P.LSC-178）

CEM 接続回線の疑似回線への追加

CEM 接続回線を疑似回線に追加するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*

5. **interface** *type interface-path-id*
6. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	クロスコネクト グループの名前を入力します。
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface CEM0/1/0/9:10`	インターフェイス タイプとインスタンスを指定します。

	コマンドまたはアクション	目的
ステップ 6	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.2.2.2 pw-id 11`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線クラスの関連付け

接続回線を疑似回線クラスと関連付けるには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-class** *class-name*
4. **encapsulation mpls**
5. **protocol ldp**
6. **end**
7. **xconnect group** *group-name*
8. **p2p** *xconnect-name*

9. **interface** *type interface-path-id*
10. **neighbor** ***A.B.C.D*** **pw-id** *pseudowire-id*
11. **pw-class** *class-name*
12. **end**
    または
    commit

手順の詳細

	コマンド	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router (config)# l2vpn`	レイヤ 2 VPN 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	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# pw-class class_cem`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 4	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。
ステップ 5	**protocol ldp** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-pwc-encap-mpls)# protocol ldp`	LDP に疑似回線シグナリング プロトコルを設定します。

	コマンド	目的
ステップ 6	**end** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# end`	システムから変更をコミットするように求められます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。
ステップ 7	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1`	相互接続グループを設定します。
ステップ 8	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント相互接続を設定します。
ステップ 9	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface CEM0/1/0/9:20`	インターフェイス タイプとインスタンスを指定します。
ステップ 10	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.2.2.2 pw-id 11`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 オプションで、コントロール ワードをディセーブルにするか、イーサネットまたは VLAN に transport-type を設定できます。

	コマンド	目的
ステップ 11	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p)# pw-class class_cem`	指定した疑似回線クラスを P2P 接続回線と関連付けます。
ステップ 12	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線ステータスのイネーブル化

疑似回線ステータスをイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-status**
4. **commit**

手順の詳細

	コマンド	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router (config)# l2vpn`	レイヤ 2 VPN 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	**pw-status** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# pw-status`	このレイヤ 2 VPN で設定されるすべての疑似回線をイネーブルにします。 **（注）** 疑似回線ステータスをディセーブルにするには、**pw-status disable** コマンドを使用します。
ステップ 4	**commit** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# commit`	実行コンフィギュレーション ファイルに設定変更を保存し、コンフィギュレーション セッションを続行します。

バックアップ疑似回線の設定

ポイントツーポイント ネイバーのバックアップ疑似回線を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** {*xconnect-name*}
5. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
6. **backup** {**neighbor** *A.B.C.D*} {**pw-id** *value*}
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードに入ります。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**xconnect group** *group-name* **例：** `RP/O/RSP0/CPU0:router(config-l2vpn)# xconnect group A` `RP/0/RSP0/CPU0:router(config-l2vpn-xc)#`	クロスコネクト グループの名前を入力します。
ステップ 4	**p2p** {*xconnect-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xc1` `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)#`	ポイントツーポイント クロスコネクトの名前を入力します。
ステップ 5	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface CEM0/1/0/9:20`	インターフェイス タイプとインスタンスを指定します。
ステップ 6	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.1.1.2 pw-id 11`	クロスコネクトの疑似回線セグメントを設定します。
ステップ 7	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw-backup)# pw-class class_cem`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 8	**backup** {neighbor *A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# backup neighbor 10.2.2.2 pw-id 5` `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)#`	相互接続のバックアップ疑似回線を設定します。 • **neighbor** キーワードを使用して、相互接続するピアを指定します。IP アドレス引数（*A.B.C.D*）は、ピアの IPv4 アドレスです。 • **pw-id** キーワードを使用して、疑似回線 ID を設定します。範囲は 1 ～ 4294967295 です。

	コマンドまたはアクション	目的
ステップ 9	**pw-class** *class-name* **例：** RP/0/RSP0/CPU0:router (config-l2vpn-xc-p2p-pw-backup)# pw-class class_cem	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 10	**end** または **commit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)# end または RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw-backup)# commit	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

L2VPN ノンストップ ルーティングの設定

L2VPN ノンストップ ルーティングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **nsr**
4. **logging nsr**
5. **end**
   または
   commit

手順の詳細

	コマンド	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router (config)# l2vpn`	レイヤ 2 VPN 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	**nsr** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# nsr`	L2VPN ノンストップ ルーティングをイネーブルにします。

	コマンド	目的
ステップ 4	**logging nsr** **例：** `RP/0/RSP0/CPU0:router (config-l2vpn)# logging nsr`	NSR イベントのロギングをイネーブルにします。
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ポイントツーポイント レイヤ 2 サービスの設定例

ここで示す設定例は、次のとおりです。

L2VPN インターフェイスの設定：例

次に、L2VPN インターフェイスを設定する例を示します。

```
configure
 interface GigabitEthernet0/0/0/0.1 l2transport
  encapsulation dot1q 1
  rewrite ingress pop 1 symmetric
 end
```

ローカル スイッチングの設定：例

次に、レイヤ 2 ローカル スイッチングを設定する例を示します。

```
configure
 l2vpn
  xconnect group examples
  p2p example1
  interface TenGigE0/7/0/6.5
  interface GigabitEthernet0/4/0/30
 commit
 end

show l2vpn xconnect group examples
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready

XConnect                    Segment 1                   Segment 2
Group      Name       ST    Description            ST    Description            ST
------------------------    --------------------------  --------------------------
examples   example1   UP    Te0/7/0/6.5            UP    Gi0/4/0/30             UP
------------------------------------------------------------------------------
```

ポイントツーポイント相互接続の設定：例

ここでは、スタティックおよびダイナミック p2p 相互接続の設定例を示します。

スタティック設定

次に、スタティック ポイントツーポイント相互接続の設定例を示します。

```
configure
  l2vpn
  xconnect group vlan_grp_1
  p2p vlan1
  interface GigabitEthernet0/0/0/0.1
  neighbor 10.2.1.1 pw-id 1
  mpls static label local 699 remote 890
commit
```

ダイナミック設定

次に、ダイナミック ポイントツーポイント相互接続の設定例を示します。

```
configure
  l2vpn
  xconnect group vlan_grp_1
  p2p vlan1
  interface GigabitEthernet0/0/0/0.1
  neighbor 10.2.1.1 pw-id 1
commit
```

Inter-AS：例

次に、AC1 から AC2 への AC 間相互接続の設定例を示します。

```
router-id Loopback0

interface Loopback0
 ipv4 address 10.0.0.5 255.255.255.255
!
interface GigabitEthernet0/1/0/0.1 l2transport
encapsulation dot1q 1
!
!
interface GigabitEthernet0/0/0/3
 ipv4 address 10.45.0.5 255.255.255.0
 keepalive disable
!
interface GigabitEthernet0/0/0/4
 ipv4 address 10.5.0.5 255.255.255.0
 keepalive disable
!
router ospf 100
 log adjacency changes detail
 area 0
  interface Loopback0
  !
  interface GigabitEthernet0/0/0/3
```

```
  !
  interface GigabitEthernet0/0/0/4
  !
 !
!
router bgp 100
 address-family ipv4 unicast
  allocate-label all
 !
 neighbor 10.2.0.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family ipv4 labeled-unicast
  !
 !
!
l2vpn
 xconnect group cisco
  p2p cisco1
   interface GigabitEthernet0/1/0/0.1
   neighbor 10.0.1.5 pw-id 101
  !
  p2p cisco2
   interface GigabitEthernet0/1/0/0.2
   neighbor 10.0.1.5 pw-id 102
  !
  p2p cisco3
   interface GigabitEthernet0/1/0/0.3
   neighbor 10.0.1.5 pw-id 103
  !
  p2p cisco4
   interface GigabitEthernet0/1/0/0.4
   neighbor 10.0.1.5 pw-id 104
  !
  p2p cisco5
   interface GigabitEthernet0/1/0/0.5
   neighbor 10.0.1.5 pw-id 105
  !
  p2p cisco6
   interface GigabitEthernet0/1/0/0.6
   neighbor 10.0.1.5 pw-id 106
  !
  p2p cisco7
   interface GigabitEthernet0/1/0/0.7
   neighbor 10.0.1.5 pw-id 107
  !
  p2p cisco8
   interface GigabitEthernet0/1/0/0.8
   neighbor 10.0.1.5 pw-id 108
  !
  p2p cisco9
   interface GigabitEthernet0/1/0/0.9
   neighbor 10.0.1.5 pw-id 109
  !
  p2p cisco10
   interface GigabitEthernet0/1/0/0.10
   neighbor 10.0.1.5 pw-id 110
  !
 !
!
mpls ldp
 router-id Loopback0
```

```
 log
  neighbor
 !
 interface GigabitEthernet0/0/0/3
 !
 interface GigabitEthernet0/0/0/4
 !
!
end
```

L2VPN Quality of Service：例

次に、ポート モードの L2 インターフェイスにサービス ポリシーをアタッチする例を示します。

```
configure
  interface GigabitEthernet 0/0/0/0
  l2transport
  service-policy input pmap_1
commit
```

疑似回線：例

例には、次のデバイスおよび接続が含まれます。

- T-PE1 ノードには次の項目があります。
  - AC インターフェイスとの相互接続（CE1 方向）
  - S-PE1 ノードへの疑似回線
  - IP アドレス：209.165.200.225
- T-PE2 ノード
  - AC インターフェイスとの相互接続（CE2 方向）
  - S-PE1 ノードへの疑似回線
  - IP アドレス：209.165.200.254
- S-PE1 ノード
  - T-PE1 ノードへの疑似回線セグメントによるマルチセグメント疑似回線相互接続
  - T-PE2 ノードへの疑似回線セグメント
  - IP アドレス：209.165.202.158

T-PE1 ノードのダイナミック疑似回線の設定：例

```
RP/0/RSP0/CPU0:T-PE1# configure
RP/0/RSP0/CPU0:T-PE1(config)# l2vpn
RP/0/RSP0/CPU0:T-PE1 (config-l2vpn)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:T-PE1(config-l2vpn)# xconnect group XCON1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc)# p2p xc1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# description T-PE1 MS-PW to 10.165.202.158
via 10.165.200.254
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# interface gigabitethernet 0/1/0/0.1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.200.254 pw-id 100
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p-pw)# commit
```

S-PE1 ノードのダイナミック疑似回線の設定：例

```
RP/0/RSP0/CPU0:S-PE1# configure
RP/0/RSP0/CPU0:S-PE1(config)# l2vpn
RP/0/RSP0/CPU0:S-PE1(config-l2vpn)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn)# xconnect group MS-PW1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc)# p2p ms-pw1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# description S-PE1 MS-PW between
10.165.200.225 and 10.165.202.158
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.200.225 pw-id 100
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.202.158 pw-id 300
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# commit
```

T-PE2 ノードのダイナミック疑似回線の設定：例

```
RP/0/RSP0/CPU0:T-PE2# configure
RP/0/RSP0/CPU0:T-PE2(config)# l2vpn
RP/0/RSP0/CPU0:T-PE2 (config-l2vpn)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE2 (config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:T-PE2 (config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:T-PE2 (config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:T-PE2 (config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:T-PE2 (config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:T-PE2(config-l2vpn)# xconnect group XCON1
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc)# p2p xc1
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# description T-PE2 MS-PW to 10.165.200.225 via
10.165.200.254
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# interface gigabitethernet 0/2/0/0.4
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# neighbor 10.165.200.254 pw-id 300
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p-pw)# commit
```

T-PE1 ノードのダイナミック疑似回線と優先パスの設定：例

```
RP/0/RSP0/CPU0:T-PE1# configure
RP/0/RSP0/CPU0:T-PE1(config)# l2vpn
RP/0/RSP0/CPU0:T-PE1(config-l2vpn)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel-te
1000
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:T-PE1(config-l2vpn)# xconnect group XCON1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc)# p2p xc1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# description T-PE1 MS-PW to 10.165.202.158
via 10.165.200.254
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# interface gigabitethernet 0/1/0/0.1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.200.254 pw-id 100
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p-pw)# commit
```

S-PE1 ノードのダイナミック疑似回線と優先パスの設定：例

```
RP/0/RSP0/CPU0:S-PE1# configure
RP/0/RSP0/CPU0:S-PE1(config)# l2vpn
RP/0/RSP0/CPU0:S-PE1(config-l2vpn)# pw-class dynamic_mpls1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel-te
1000
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn)# pw-class dynamic_mpls2
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel-te
2000
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn)# xconnect group MS-PW1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc)# p2p ms-pw1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# description S-PE1 MS-PW between
10.165.200.225 and 10.165.202.158
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.200.225 pw-id 100
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.202.158 pw-id 300
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls2
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# commit
```

T-PE2 ノードのダイナミック疑似回線と優先パスの設定：例

```
RP/0/RSP0/CPU0:T-PE2# configure
RP/0/RSP0/CPU0:T-PE2(config)# l2vpn
RP/0/RSP0/CPU0:T-PE2(config-l2vpn)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-pwc)# encapsulation mpls
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-pwc-encap-mpls)# protocol ldp
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-pwc-encap-mpls)# control-word disable
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel-te
2000
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-pwc-encap-mpls)# exit
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-pwc)# exit
RP/0/RSP0/CPU0:T-PE2(config-l2vpn)# xconnect group XCON1
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc)# p2p xc1
```

```
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# description T-PE2 MS-PW to 10.165.200.225 via
10.165.200.254
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# interface gigabitethernet 0/2/0/0.4
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# neighbor 10.165.200.254 pw-id 300
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p-pw)# commit
```

T-PE1 ノードのスタティック疑似回線の設定：例

```
RP/0/RSP0/CPU0:T-PE1# configure
RP/0/RSP0/CPU0:T-PE1(config)# l2vpn
RP/0/RSP0/CPU0:T-PE1(config-l2vpn)# xconnect group XCON1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc)# p2p xc1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# interface gigabitethernet 0/1/0/0.1
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.200.254 pw-id 100
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p-pw)# mpls static label local 50 remote 400
RP/0/RSP0/CPU0:T-PE1(config-l2vpn-xc-p2p-pw)# commit
```

S-PE1 ノードのスタティック疑似回線の設定：例

```
RP/0/RSP0/CPU0:S-PE1# configure
RP/0/RSP0/CPU0:S-PE1(config)# l2vpn
RP/0/RSP0/CPU0:S-PE1(config-l2vpn)# xconnect group MS-PW1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc)# p2p ms-pw1
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.200.225 pw-id 100
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# mpls static label local 400 remote 50
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# exit
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p)# neighbor 10.165.202.158 pw-id 300
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# mpls static label local 40 remote 500
RP/0/RSP0/CPU0:S-PE1(config-l2vpn-xc-p2p-pw)# commit
```

T-PE2 ノードのスタティック疑似回線の設定：例

```
RP/0/RSP0/CPU0:T-PE2# configure
RP/0/RSP0/CPU0:T-PE2(config)# l2vpn
RP/0/RSP0/CPU0:T-PE2(config-l2vpn)# xconnect group XCON1
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc)# p2p xc1
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# interface gigabitethernet 0/2/0/0.4
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p)# neighbor 10.165.200.254 pw-id 300
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p-pw)# mpls static label local 500 remote 40
RP/0/RSP0/CPU0:T-PE2(config-l2vpn-xc-p2p-pw)# commit
```

優先パス：例

次に、優先トンネル パスを設定する例を示します。

```
configure
 l2vpn
 pw-class path1
  encapsulation mpls
   preferred-path interface tunnel tp 50 fallback disable
```

MPLS トランスポート プロファイル：例

ここでは、次の例を示します。

- 優先トンネル パスの設定：例
- PW ステータス OAM の設定：例

優先トンネル パスの設定：例

この設定例では、優先トンネル パスを設定する方法を示します。

```
l2vpn
 pw-class foo
  encapsulation mpls
  preferred-path interface tunnel-tp 100 fallback disable
 commit
```

PW ステータス OAM の設定：例

この設定例では、PW ステータス OAM 機能を設定する方法を示します。

```
l2vpn
 pw-oam refresh transmit 100
 commit
```

疑似回線ステータスの表示：例

show l2vpn xconnect

```
RP/0/RSP0/CPU0:router# show l2vpn xconnect

Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        LU = Local Up, RU = Remote Up, CO = Connected
XConnect                    Segment 1                    Segment 2
Group      Name       ST    Description            ST    Description            ST
------------------------    -------------------------    -------------------------
MS-PW1     ms-pw1     UP    10.165.200.225     100    UP    10.165.202.158      300    UP
------------------------------------------------------------------------------
```

show l2vpn xconnect detail

```
RP/0/RSP0/CPU0:router# show l2vpn xconnect detail
Group MS-PW1, XC ms-pw1, state is up; Interworking none
  PW: neighbor 10.165.200.225, PW ID 100, state is up ( established )
    PW class not set
    Encapsulation MPLS, protocol LDP
    PW type Ethernet VLAN, control word enabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set
    PW Status TLV in use
        MPLS         Local                          Remote
      ------------ ------------------------------ ------------------------------
      Label        16004                          16006
      Group ID     0x2000400                      0x2000700
      Interface    GigabitEthernet0/1/0/2.2       GigabitEthernet0/1/0/0.3
      MTU          1500                           1500
      Control word enabled                        enabled
      PW type      Ethernet VLAN                  Ethernet VLAN
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x5                            0x7
                   (control word)                 (control word)
                                                  (router alert label)
                   (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------------------
    Incoming PW Switching TLVs (Label Mapping message):
      None
    Incoming Status (PW Status TLV and accompanying PW Switching TLV):
```

```
    Status code: 0x0 (no fault) in Notification message
  Outgoing PW Switching TLVs (Label Mapping message):
    Local IP Address: 10.165.200.254 , Remote IP address: 10.165.202.158 , PW ID: 300
    Description: S-PE1 MS-PW between 10.165.200.225 and 10.165.202.158
  Outgoing Status (PW Status TLV and accompanying PW Switching TLV):
    Status code: 0x0 (no fault) in Notification message
    Local IP Address: 10.165.200.254
  Create time: 04/04/2008 23:18:24 (00:01:24 ago)
  Last time status changed: 04/04/2008 23:19:30 (00:00:18 ago)
  Statistics:
    packet totals: receive 0
    byte totals: receive 0
PW: neighbor 10.165.202.158 , PW ID 300, state is up ( established )
  PW class not set
  Encapsulation MPLS, protocol LDP
  PW type Ethernet VLAN, control word enabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set
  PW Status TLV in use
      MPLS         Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        16004                          16006
    Group ID     0x2000800                      0x2000200
    Interface    GigabitEthernet0/1/0/0.3       GigabitEthernet0/1/0/2.2
    MTU          1500                           1500
    Control word enabled                        enabled
    PW type      Ethernet VLAN                  Ethernet VLAN
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x5                            0x7
           (control word)                 (control word)
                                                 (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------------------
  Incoming PW Switching TLVs (Label Mapping message):
    None
  Incoming Status (PW Status TLV and accompanying PW Switching TLV):
    Status code: 0x0 (no fault) in Notification message
  Outgoing PW Switching TLVs (Label Mapping message):
    Local IP Address: 10.165.200.254 , Remote IP address: 10.165.200.225, PW ID: 100
    Description: S-PE1 MS-PW between 10.165.200.225 and 10.165.202.158
  Outgoing Status (PW Status TLV and accompanying PW Switching TLV):
    Status code: 0x0 (no fault) in Notification message
```

```
      Local IP Address: 10.165.200.254
    Create time: 04/04/2008 23:18:24 (00:01:24 ago)
    Last time status changed: 04/04/2008 23:19:30 (00:00:18 ago)
    Statistics:
      packet totals: receive 0
      byte totals: receive 0
RP/0/RSP0/CPU0:router#
""Show l2vpn xconnect summary": added PW-PW count.
"Show l2vpn forwarding location <> (no change: does not display MS-PWs)
"Show l2vpn forwarding summary location <> (no change: does not display MS-PWs)
```

Any Transport over MPLS（AToM）の設定：例

次に、Any Transport over MPLS（AToM）を設定する例を示します。

```
config
l2vpn
 xconnect group test
 p2p test
 interface POS 0/1/0/0.1
 neighbor 10.1.1.1 pw-id 100
```

AToM IP インターワーキングの設定：例

次に、IP インターワーキングを設定する例を示します。

```
config
l2vpn
 xconnect group test
 p2p test
 interworking ipv4
```

PPP IP インターワーキングの設定：例

次に、PPP IP インターワーキングを設定する例を示します。

```
interface Serial0/2/1/0/1/1/1:0
 encapsulation ppp
l2transport
 !
!
interface Serial0/0/0/0/2/1/1:0
encapsulation ppp
 l2transport
 !
!

!! Local Switching Configuration
l2vpn
xconnect group ppp_ip_ls
  p2p 1
   interface Serial0/2/1/0/1/1/1:0
```

```
   interface GigabitEthernet0/0/0/1.1
   interworking ipv4
  !

!! PW Configuration
l2vpn
 xconnect group ppp_ip_iw
  p2p 1
   interface Serial0/0/0/0/2/1/1:0
   neighbor 120.120.120.120 pw-id 3
    pw-class class1
   !
   interworking ipv4
```

cHDLC IP インターワーキングの設定：例

次に、cHDLC IP インターワーキングを設定する例を示します。

```
interface Serial0/2/1/0/1/1/2:0
l2transport

interface Serial0/0/0/0/2/1/2:0
l2transport


!! Local Switching Configuration
l2vpn
xconnect group ppp_ip_ls
  p2p 1
   interface Serial0/2/1/0/1/1/2:0
   interface GigabitEthernet0/0/0/2.1
   interworking ipv4
  !

!! PW Configuration
l2vpn
 xconnect group ppp_ip_iw
  p2p 1
   interface Serial0/0/0/0/2/1/2:0
   neighbor 120.120.120.120 pw-id 3
    pw-class class1
   !
   interworking ipv4
```

MLPPP IP インターワーキングの設定：例

次に、MLPPP IP インターワーキングを設定する例を示します。

```
interface Multilink0/2/1/0/1
 multilink
l2transport
!

interface Multilink0/2/1/0/51
Multilink
l2transport
```

```
!! Local Switching Configuration
l2vpn
 xconnect group mlppp_ip_ls
  p2p 1
   interface Multilink0/2/1/0/1
   interface GigabitEthernet0/0/0/1.151
   interworking ipv4
!
!! PW Configuration
l2vpn
xconnect group mlppp_ip_iw
  p2p 151
   interface Multilink0/2/1/0/51
   neighbor 140.140.140.140 pw-id 151
    pw-class test
   !
   interworking ipv4
  !
```

Circuit Emulation over Packet Switched Network の設定：例

次に、Circuit Emulation Over Packet Switched Network を設定する例を示します。

CEM 接続回線の PW への追加

```
l2vpn
 xconnect group gr1
  p2p p1
   interface CEM 0/0/0/0:10
   neighbor 3.3.3.3 pw-id 11
   !
  !
```

疑似回線クラスの関連付け

```
l2vpn
 pw-class class-cem
  encapsulation mpls
   protocol ldp
  !
 !
xconnect group gr1
  p2p p1
   interface CEM0/0/0/0:20
   neighbor 1.2.3.4 pw-id 11
    pw-class class-cem
   !
```

疑似回線ステータスのイネーブル化

```
l2vpn
  pw-status
  commit
```

疑似回線ステータスのディセーブル化

```
l2vpn
  pw-status disable
```

```
 commit
```

バックアップ疑似回線の設定

```
l2vpn
 pw-status
 pw-class class-cem
  encapsulation mpls
   protocol ldp
  !
 !
 xconnect group gr1
  p2p p1
   interface CEM0/0/0/0:20
   neighbor 1.2.3.4 pw-id 11
    pw-class class-cem
    backup neighbor 9.9.9.9 pw-id 1221
     pw-class class-cem
    !
   !
```

L2VPN ノンストップ ルーティングの設定：例

次に、L2VPN ノンストップ ルーティングを設定する例を示します。

```
config
l2vpn
  nsr
  logging nsr
```

疑似回線のグループ化のイネーブル化：例

次に、疑似回線のグループ化をイネーブルにする例を示します。

```
config
l2vpn
  pw-grouping
```

その他の関連資料

MPLS レイヤ 2 VPN の実装に関する追加情報については、以下を参照してください。

関連資料

関連項目	参照先
Cisco IOS XR L2VPN コマンド	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』
レイヤ 2 VPN	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide*』
MPLS VPN over IP トンネル	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide*』
スタートアップ資料	『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』

標準

標準[1]	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

1. サポートされている規格がすべて記載されているわけではありません。

MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
RFC 4447	『*Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*』2006 年 4 月
RFC 4448	『*Encapsulation Methods for Transport of Ethernet over MPLS Networks*』2006 年 4 月

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

その他の関連資料

マルチポイント レイヤ 2 サービスの実装

このモジュールでは、マルチポイント レイヤ 2 ブリッジング サービス（Cisco ASR 9000 シリーズ アグリゲーション サービス ルータではバーチャル プライベート LAN サービス（VPLS）とも呼ばれます）の概念および設定情報を示します。VPLS は、レイヤ 2 VPN テクノロジーをサポートし、カスタマーにトランスペアレントなマルチポイント レイヤ 2 接続を提供します。

（注） このアプローチでは、サービス プロバイダーは、ブロードキャスト TV やレイヤ 2 VPN などの多数の新しいサービスをホストできます。Cisco ASR 9000 シリーズ ルータでの MPLS レイヤ 2 VPN の詳細、およびこのモジュールに記載されているコマンドの説明については、「関連資料」の項を参照してください。設定作業の実行時に使用する可能性があるその他のコマンドに関するドキュメントを見つけるには、オンラインの Cisco IOS XR ソフトウェア マスター コマンド索引を検索してください。

Cisco ASR 9000 シリーズ ルータでのマルチポイント レイヤ 2 サービスの実装機能の履歴

リリース	変更内容
リリース 3.7.2	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 3.9.0	次の機能が追加されました。 • 不明なユニキャスト フラッディングのブロック。 • MAC フラッシュのディセーブル化。 • マルチ スパニングツリー アクセス ゲートウェイ • スケール拡張機能が導入されました。スケール拡張機能の詳細については、表 4（P.449）を参照してください。
リリース 3.9.1	BGP オートディスカバリおよび LDP シグナリングによる VPLS のサポートが追加されました。
リリース 4.0.1	次の機能に対するサポートが追加されました。 • ダイナミック ARP インスペクション • IP SourceGuard • MAC アドレスのセキュリティ

リリース 4.1.0	ASR 9000 SIP-700 ラインカードでのこれらの VPLS 機能のサポートが追加されました。 • MAC 学習およびフォワーディング • MAC アドレス エージング サポート • MAC 制限 • スプリット ホライズン グループ • MAC アドレス取り消し • 未知のユニキャスト、ブロードキャスト、およびマルチキャスト パケットのフラッディング • アクセス疑似回線 • H-VPLS PW アクセス • PW の冗長性 G.8032 イーサネット リング保護機能のサポートが追加されました。
リリース 4.2.1	Flow Aware Transport（FAT）疑似回線機能のサポートが追加されました。
リリース 4.3.0	次の機能のサポートが追加されました。 • 疑似回線ヘッドエンド（PWHE） • ASR 9000 Enhanced Ethernet ラインカードのスケール拡張機能： – VPWS および VPLS 内の 128000 疑似回線のサポート – VPLS と VPWS インスタンスでの 128000 疑似回線のサポート – ブリッジの 512 疑似回線までのサポート – 128000 バンドル接続回線のサポート – 128000 VLAN のサポート • L2VPN over GRE

内容

マルチポイント レイヤ 2 サービス実装の前提条件

VPLS を設定する前に、次の作業が終了し、条件が満たされていることを確認してください。

- このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

  ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- プロバイダー エッジ（PE）ルータが IP によっ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。
- レイヤ 2 トラフィックを開始して終了するようにループバック インターフェイスを設定します。PE ルータが他のルータのループバック インターフェイスにアクセスできるようにします。

  （注） ループバック インターフェイスは、すべてのケースで必要というわけではありません。たとえば、VPLS が TE トンネルに直接マッピングされている場合、トンネル選択ではループバック インターフェイスは必要ありません。
- ラベル スイッチド パス（LSP）が PE ルータ間に存在するよう、コアで MPLS とラベル配布プロトコル（LDP）を設定します。
- コア側インターフェイスは、イーサネットベースでなければなりません。VPLS が設定されている場合は、POS、フレーム リレー、PPP/MLPPP インターフェイスはコア側インターフェイスとしてサポートされません。

マルチポイント レイヤ 2 サービスの実装に関する情報

バーチャル プライベート LAN サービス（VPLS）を実装するには、次の概念を理解する必要があります。

- バーチャル プライベート LAN サービスの概要（P.LSC-204）
- MPLS ベースのプロバイダー コアの VPLS（P.LSC-207）
- VPLS ディスカバリおよびシグナリング（P.LSC-209）
- MAC アドレス関連パラメータ（P.LSC-212）
- LSP Ping over VPWS および VPLS（P.LSC-215）
- スプリット ホライズン グループ（P.LSC-215）
- レイヤ 2 セキュリティ（P.LSC-216）

- G.8032 イーサネット リング保護（P.LSC-217）
- Flow Aware Transport 疑似回線（FAT PW）（P.LSC-223）
- 疑似回線ヘッドエンド（P.LSC-224）
- L2VPN over GRE（P.LSC-224）

バーチャル プライベート LAN サービスの概要

バーチャル プライベート LAN サービス（VPLS）を使用すると、地理的に離れたローカル エリア ネットワーク（LAN）セグメントを MPLS ネットワーク経由で単一ブリッジ ドメインとして相互接続できます。MAC アドレス ラーニング、エージング、およびスイッチングなどの従来の LAN の機能はすべて、単一のブリッジ ドメインに属する、リモート接続されたすべての LAN セグメント全体でエミュレートされます。

VPLS ネットワークの一部のコンポーネントについては、次の項で説明します。

ブリッジ ドメイン

ネイティブ ブリッジ ドメインは、一連の物理または仮想ポート（VFI を含む）から構成されるレイヤ 2 のブロードキャスト ドメインです。データ フレームは、宛先 MAC アドレスに基づいてブリッジ ドメイン内でスイッチングされます。マルチキャスト、ブロードキャスト、不明な宛先ユニキャスト フレームは、ブリッジ ドメイン内でフラッディングされます。また、送信元 MAC アドレス ラーニングは、ブリッジ ドメインのすべての着信フレームで行われます。学習されたアドレスは期限切れになります。着信フレームは、入力ポート、または入力ポートと MAC ヘッダー フィールドの両方の組み合わせのいずれかに基づいてブリッジ ドメインにマッピングされます。

デフォルトでは、スプリット ホライズンは同じ VFI 下の疑似回線でイネーブルです。ただし、デフォルト設定では、スプリット ホライズンは接続回線（インターフェイスまたは疑似回線）でイネーブルではありません。

フラッディング最適化

Cisco ASR 9000 シリーズ ルータは、ブリッジ ドメインでトラフィックをブリッジしながら、不必要にフラッディングするトラフィック量を最小限に抑えます。フラッディング最適化機能によって、この機能を実現します。ただし、特定の障害回復シナリオでは、実際には、トラフィックの損失を防止するには追加のフラッディングが推奨されます。トラフィック損失は、ブリッジ ポート リンクの 1 つが非アクティブになり、スタンバイ リンクによって置き換えられる一時的な間隔中に発生します。

一部の設定では、トラフィック フラッディングを最小化する最適化は、ブリッジのリンクの 1 つで障害が発生し、スタンバイ リンクによって置き換えられる短期間にトラフィック損失という犠牲を払って行われます。そのため、設定に適した特定のフラッディング動作を指定するには、さまざまなモードでフラッディング最適化で設定できます。

次のフラッディング最適化モードを設定できます。

- 帯域幅最適化モード
- コンバージェンス モード
- TE FRR 最適化モード

帯域幅最適化モード

フラッディング トラフィックは、ブリッジ ドメインに接続されたブリッジ ポートまたは疑似回線のラインカードだけに送信されます。これは、デフォルトのモードです。

コンバージェンス モード

フラッディング トラフィックはシステムのすべてのラインカードに送信されます。トラフィックは、ブリッジ ドメインに接続されたブリッジ ポートまたは疑似回線の有無に関係なくフラッディングされます。そのブリッジ ドメインに接続された等コスト MPLS パス（ECMP）が複数ある場合は、トラフィックはすべての ECMP にフラッディングされます。

コンバージェンス モードの目的は、障害によりブリッジ リンクが変更される短いインターバル中に失われる絶対トラフィック量を最小限にすることです。

TE FRR 最適化モード

トラフィック エンジニアリング高速再ルーティング（TE FRR）の最適化モードは、ブリッジ ドメインに接続された TE FRR 疑似回線に関するフラッディング動作を除き、帯域幅最適化モードに似ています。TE FRR 最適化モードでは、トラフィックは、プライマリおよびバックアップ FRR インターフェイスの両方にフラッディングされます。ブリッジ トラフィックが FRR の回復時間の制約に準拠するように、このモードは、FRR フェールオーバー中のトラフィック損失を最小限にするために使用されます。

ダイナミック ARP インスペクション

ダイナミック ARP インスペクション（DAI）は、アドレス解決プロトコル（ARP）スプーフィング攻撃から保護する方法です。不正な IP/MAC アドレス バインディングを持つ ARP パケットを代行受信し、ログに記録して、廃棄します。この機能により、ネットワークをある種の中間者攻撃から保護することができます。DAI 機能は、デフォルトではディセーブルです。

ARP では、IP アドレスを MAC アドレスにマッピングすることで、レイヤ 2 ブロードキャスト ドメイン内の IP 通信を可能にします。スプーフィング攻撃は、ARP 要求が実際に受信されなかった場合でも、ホストからの ARP 応答を許可するために発生します。次に攻撃が発生した後、攻撃下にあるデバイスからのすべてのトラフィックは、最初に攻撃者のシステムを通過し、次にルータ、スイッチ、またはホストを通過します。ARP スプーフィング攻撃は、サブネットに接続されているデバイスの ARP キャッシュに偽りの情報を送信することにより、レイヤ 2 ネットワークに接続されているデバイスに影響を及ぼす可能性があります。ARP キャッシュに偽りの情報を送信することを ARP キャッシュ ポイズニングといいます。

ダイナミック ARP インスペクション機能を使用することで、有効な ARP 要求と応答だけが中継されることを保証できます。ARP インスペクションは 2 種類あります。

- 必要なインスペクション：送信側の MAC アドレス、IPv4 アドレス、受信側のブリッジ ポート XID およびブリッジがチェックされます。
- オプション インスペクション：次の項目が検証されます。
  - 送信元 MAC：送信者および送信元の MAC がチェックされます。チェックは、すべての ARP または RARP パケットで行われます。
  - 宛先 MAC：ターゲットおよび宛先の MAC がチェックされます。チェックは、すべての応答または応答反転パケットで実行されます。
  - IPv4 アドレス：ARP 要求では、送信者の IPv4 アドレスが 0.0.0.0、マルチキャスト アドレス、またはブロードキャスト アドレスかどうかを調べるためにチェックが実行されます。ARP 応答および ARP 応答反転では、ターゲットの IPv4 アドレスが 0.0.0.0、マルチキャスト アドレス、またはブロードキャスト アドレスかどうかを調べるためにチェックが実行されます。このチェックは、要求、応答、および応答反転パケットに応じて実行されます。

（注） DAI 機能は、接続回線および EFP でサポートされます。現在、DAI 機能は疑似回線ではサポートされません。

IP ソース ガード

IP ソース ガード（IPSG）は、非ルーテッド レイヤ 2 インターフェイスで IP トラフィックを制限するために、DHCP スヌーピング バインディング データベースおよび手動で設定された IP ソース バインディングに基づいてトラフィックをフィルタリングするセキュリティ機能です。

IPSG 機能は、悪意のあるホストが正当のホストの IP アドレスを推測することによって正当のホストを操作しないように、レイヤ 2 ポートで送信元 IP アドレスをフィルタリングします。この機能は、動的な DHCP スヌーピングおよび静的な IP ソース バインディングを使用して、IP アドレスをホストと照合します。

まず、DHCP パケットを除き、IPSG 用に設定された EFP のすべての IP トラフィックがブロックされます。クライアントが DHCP サーバから IP アドレスを受信したあと、またスタティック IP ソース バインディングが管理者によって設定されたあと、その IP 送信元アドレスのある全トラフィックがそのクライアントから許可されます。他のホストからのトラフィックは拒否されます。このフィルタリングは、隣接ホストの IP アドレスを要求することによって、ホストのネットワーク攻撃を制限します。

（注） IPSG 機能は、接続回線および EFP でサポートされます。現在、IPSG 機能は疑似回線ではサポートされません。

疑似回線

疑似回線は、PE ルータのペア間のポイントツーポイント接続です。その主な機能は、共通 MPLS 形式にカプセル化することによって、基礎となるコア MPLS ネットワーク経由でイーサネットなどのサービスをエミュレートすることです。共通 MPLS 形式へのサービスのカプセル化によって、疑似回線では、通信事業者は MPLS ネットワークにサービスを統合できます。

次のスケール拡張機能は、ASR 9000 Enhanced Ethernet ラインカードに適用できます。

- VPWS および VPLS 内の 128000 疑似回線のサポート
- VPLS と VPWS インスタンスでの 128000 疑似回線のサポート
- ブリッジの 512 疑似回線までのサポート

（注） このスケール拡張機能は、RSP3 および ASR 9000 Enhanced Ethernet ラインカードが使用されるハードウェア設定内でサポートされます。ただし、これらの拡張機能は、RSP2、ASR 9000 イーサネット ラインカードおよび Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 ラインカードには適用されません。

疑似回線を介した DHCP スヌーピング

Cisco ASR 9000 シリーズ ルータでは、DHCP サーバが疑似回線に到達可能な DHCP スヌーピングを実行できます。疑似回線は信頼できるインターフェイスと見なされます。

dhcp ipv4 snoop profile {*dhcp-snooping-profile1*} コマンドは、ブリッジ上で DHCP スヌーピングをイネーブルにし、ブリッジに DHCP スヌーピング プロファイルを対応付けるために、ブリッジ ドメインで提供されます。

仮想転送インスタンス

VPLS は、仮想転送インスタンス（VFI）の特性に基づいています。VFI は、宛先 MAC アドレス、送信元 MAC アドレス ラーニングとエージングなどに基づいて、転送などのネイティブ ブリッジング機能を実行できる仮想ブリッジ ポートです。

VFI は、VPLS インスタンスごとに PE ルータ上に作成されます。PE ルータでは、特定の VPLS インスタンスの VFI を検索して、パケットの転送先が決定されます。VFI は、特定の VPLS インスタンスの仮想ブリッジのように動作します。VFI には、特定の VPLS に属する複数の接続回線を接続できます。PE ルータは、その VPLS インスタンス内にあるすべての他の PE ルータに対するエミュレート VC を構築し、これらのエミュレート VC を VFI に接続します。パケットの転送先の決定は、VFI に維持されているデータ構造に基づいて行われます。

MPLS ベースのプロバイダー コアの VPLS

VPLS はマルチポイント レイヤ 2 VPN テクノロジーであり、ブリッジング技法によって複数のカスタマー デバイスを接続します。Multipoint Bridging のビルディング ブロックのブリッジ ドメインは、各 PE ルータに存在します。PE ルータのブリッジ ドメインへのアクセス接続は、接続回線と呼ばれます。接続回線は、一連の物理ポート、仮想ポート、またはネットワーク内の各 PE デバイスのブリッジに接続されている両方ポートです。

接続回線をプロビジョニングした後、この特定のインスタンスの MPLS ネットワークを介したネイバー関係が、エンド PE を識別する一連の手動コマンドによって確立されます。ネイバー アソシエーションが完了すると、MPLS コアとカスタマー ドメイン間のゲートウェイである疑似回線のフル メッシュがネットワーク側プロバイダー エッジ デバイス間で確立されています。

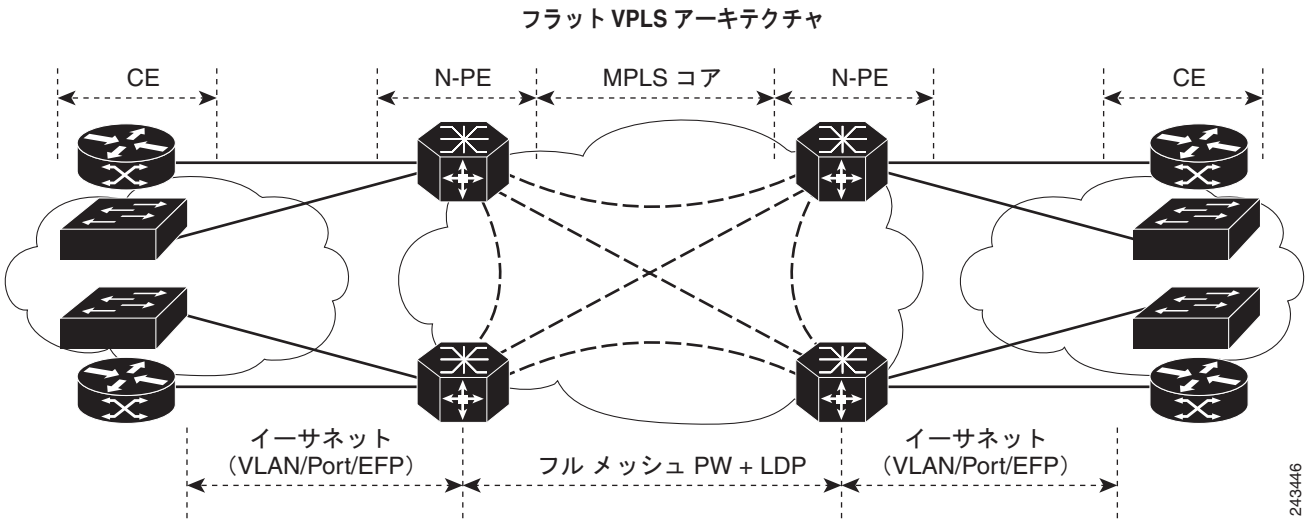
MPLS/IP プロバイダー コアは、1 つのブロードキャスト ドメインを構成するために、各 PE デバイス上の複数の接続回線を接続する仮想ブリッジをシミュレートします。また、これらの間でエミュレート仮想回線（VC）を構成するために、VPLS インスタンスに参加しているすべての PE ルータも必要です。

次に、サービス プロバイダー ネットワークは、宛先 MAC アドレスを調べてカスタマーに固有のブリッジ ドメイン内でパケットの交換を開始します。不明、ブロードキャスト、マルチキャストの宛先 MAC アドレスを持つすべてのトラフィックは、サービス プロバイダー ネットワークに接続するすべての接続済み CE カスタマー エッジ デバイスにフラッディングされます。ネットワーク側プロバイダー エッジ デバイスは、パケットがフラッディングされると送信元 MAC アドレスを学習します。トラフィックは、学習されたすべての MAC アドレスのカスタマー エッジ デバイスにユニキャストされます。

VPLS アーキテクチャ

基本的な VPLS アーキテクチャまたはフラット VPLS アーキテクチャでは、プロバイダー エッジ（PE）ルータ間のエンドツーエンド接続がマルチポイント イーサネット サービスを提供できます。図 9 に、IP/MPLS ネットワークでネットワーク プロバイダー エッジ（N-PE）ノード間の相互接続を示すフラット VPLS アーキテクチャを示します。

図 9　基本的な VPLS アーキテクチャ



VPLS ネットワークでは、各 PE ルータのブリッジ ドメイン（レイヤ 2 ブロードキャスト ドメイン）の作成が必要です。VPLS プロバイダー エッジ デバイスは、MAC テーブルおよびブリッジ ドメイン情報を転送するすべての VPLS を保持します。さらに、すべてのフラッディング ブロードキャスト フレームおよびマルチキャスト複製を処理します。

VPLS アーキテクチャの PE は、疑似回線（PW）のフル メッシュに接続します。仮想転送インスタンス（VFI）は、疑似回線のメッシュの相互接続に使用されます。ブリッジ ドメインは、PW メッシュを介してイーサネット マルチポイント ブリッジングを提供する仮想スイッチング インスタンス（VSI）を作成するために VFI に接続されます。VPLS ネットワークは、エミュレートされたイーサネット スイッチを作成するために、MPLS 疑似回線を使用して VSI をリンクします。

VPLS では、1 つの VPLS インスタンスに関与するすべてのカスタマー装置（CE）デバイスが同じ LAN 上に表示されるため、CE デバイスでポイントツーポイント回線のフル メッシュを必要とせずに、マルチポイント トポロジで相互に直接通信できます。サービス プロバイダーは、カスタマーごとに別のブリッジ ドメインを定義することで、MPLS ネットワーク上で複数のカスタマーに VPLS サービスを提供できます。あるブリッジ ドメインからのパケットが別のブリッジ ドメインには伝送または配信されることはないため、LAN サービスのプライバシーが確保されます。

VPLS は、同じレイヤ 2 ブロードキャスト ドメインに属する複数サイト間で、イーサネット IEEE 802.3、VLAN IEEE 802.1q、および VLAN-in-VLAN（Q-in-Q）トラフィックを転送します。VPLS は、フラッディング ブロードキャスト、マルチキャスト、およびブリッジで受信した不明なユニキャスト フレームを含む単純な VLAN サービスを提供します。VPLS ソリューションでは、PE ルータ間で確立された疑似回線のフル メッシュが必要です。VPLS 実装は、ラベル配布プロトコル（LDP）ベースの疑似回線シグナリングに基づきます。

レイヤ 2 スイッチングの VPLS

VPLS テクノロジーには、レイヤ 2 ブリッジングを実行するように Cisco ASR 9000 シリーズ ルータを設定する機能が含まれます。このモードではCisco ASR 9000 シリーズ ルータは、他のシスコ スイッチのように動作するように設定できます。

次の機能がサポートされています。

- ブリッジング IOS XR トランク インターフェイス
- EFP でのブリッジング

これらのブリッジング機能の例については、マルチポイント レイヤ 2 サービスの設定例の項を参照してください。

VPLS ディスカバリおよびシグナリング

VPLS はレイヤ 2 マルチポイント サービスであり、WAN サービスで LAN サービスをエミュレートします。VPLS は、サービス プロバイダーが、パケットスイッチド ネットワークで複数の LAN セグメントを相互接続し、単一の LAN として動作できるようにします。サービス プロバイダーは、VPLS を使用するお客様へのネイティブ イーサネット アクセス接続を提供できます。

VPLS のコントロール プレーンは、2 つの重要なコンポーネントであるオートディスカバリおよびシグナリングで構成されます。

- VPLS オートディスカバリは、手動で VPLS ネイバーをプロビジョニングする必要性をなくします。VPLS オートディスカバリは、各 VPLS PE ルータが同じ VPLS ドメインに属する他のプロバイダー エッジ（PE）ルータを検出できるようにします。
- PE が検出されると、VPLS ドメインの PE ルータで PW のフル メッシュを形成している PE ルータの各ペア間で、疑似回線（PW）がシグナリングおよび確立されます

図 10　　VPLS オートディスカバリとシグナリング

BGP ベースの VPLS オートディスカバリ

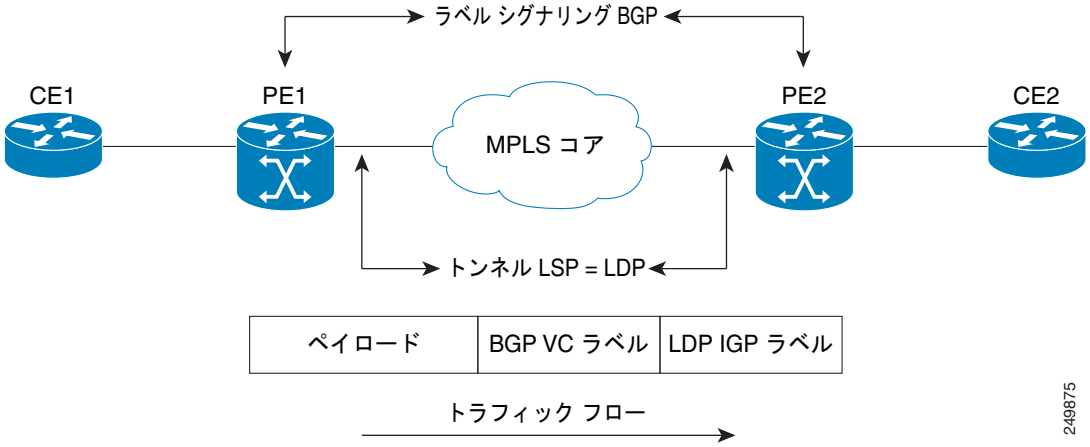
VPLS を含め VPN テクノロジーの重要な点は、ネットワーク デバイスが特定の VPN とのアソシエーションについて他のデバイスに自動的に信号を送信する機能です。オートディスカバリでは、この情報を VPN のすべてのメンバーに配布する必要があります。VPLS は、BGP が最適であるマルチポイント メカニズムです。

BGP ベースの VPLS オートディスカバリにより、VPLS ネイバーを手動でプロビジョニングする必要がなくなります。VPLS オートディスカバリは、各 VPLS PE ルータが同じ VPLS ドメインに属する他のプロバイダー エッジ（PE）ルータを検出できるようにします。VPLS オートディスカバリは、いつ PE ルータが追加されたか VPLS ドメインから削除されたかもトラックします。ディスカバリ プロセスが完了すると、各 PE ルータは、VPLS 疑似回線（PW）の設定に必要な情報を取得します。

BGP シグナリングによる BGP オートディスカバリ

ネットワークでの VPLS の実装では、プロバイダー エッジ（PE）ルータ間で PW のフル メッシュを確立する必要があります。PW には、BGP シグナリングを使用して信号を送信できます。

図 11　　ディスカバリおよびシグナリングの属性



BGP のシグナリングおよびオートディスカバリ方式には、次のコンポーネントがあります。

- PE が、特定の VPLS のメンバーであるリモート PE を学習するための方法。このプロセスをオートディスカバリといいます。
- PE が、特定の VPLS の特定のリモート PE で予期される疑似回線ラベルを学習する方法。このプロセスをシグナリングといいます。

BGP ネットワーク層到達可能性情報（NLRI）は、上記の 2 つのコンポーネントを同時に処理します。特定の PE によって生成される NLRI には、他の PE で必要な情報が含まれています。これらのコンポーネントは、各 PE の疑似回線を手動で設定することなく、各 VPLS の疑似回線のフル メッシュを自動的に設定できるようにします。

BGP AD とシグナリングによる VPLS の NLRI フォーマット

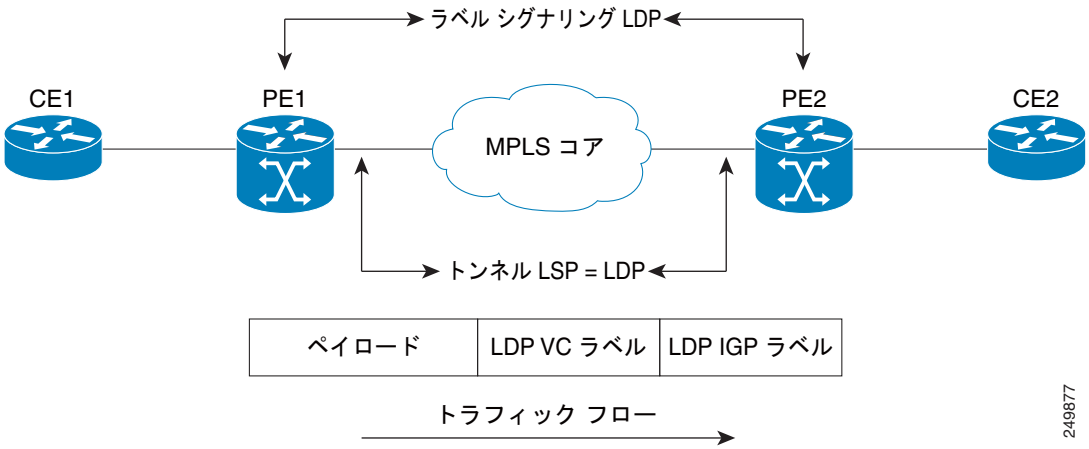
図 12 に、BGP AD とシグナリングによる VPLS の NLRI フォーマットを示します。

図 12　　NLRI フォーマット

LDP シグナリングによる BGP オートディスカバリ

疑似回線のシグナリングでは、2 つのエンドポイント間で情報を交換する必要があります。ラベル配布プロトコル（LDP）は、ポイントツーポイント シグナリングに適しています。プロバイダー エッジ デバイス間の疑似回線のシグナリングは、ターゲット LDP セッションを使用して、ラベルの値と属性を交換し、疑似回線を設定します。

図 13　　ディスカバリおよびシグナリングの属性



PE ルータは、各 VPLS の BGP で ID をアドバタイズします。この ID は、VPLS インスタンス内で一意であり、VPLS ID と同様に機能します。ID は、BGP アドバタイズメントを受信している PE ルータが、アドバタイズメントに関連付けられた VPLS を識別し、正しい VPLS インスタンスにインポートできるようにします。このようにして、VPLS ごとに、PE ルータは、VPLS のメンバーである他の PE ルータを学習します。

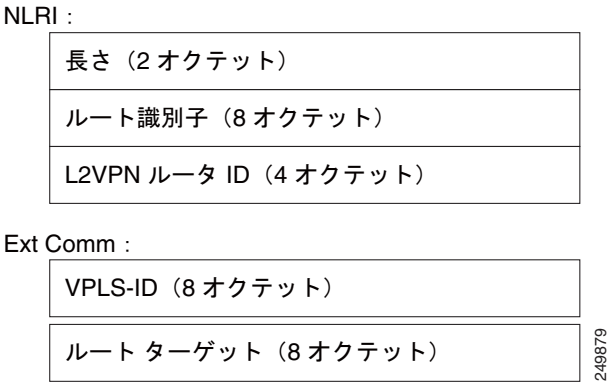
LDP プロトコルは、他のすべての PE ルータに疑似回線を設定するために使用されます。FEC 129 はシグナリングに使用されます。FEC 129 で伝送される情報には、VPLS ID、Target Attachment Individual Identifier（TAII）、および Source Attachment Individual Identifier（SAII）が含まれます。

LDP アドバタイズメントには、疑似回線上の着信トラフィックの予想される内部ラベルまたは VPLS ラベルも含まれます。これは、LDP ピアが、疑似回線を関連付ける VPLS インスタンスおよびその疑似回線でのトラフィックの送信時に使用することが予想されるラベル値を特定できるようにします。

NLRI と拡張コミュニティ

図 14 では、ネットワーク層到達可能性情報（NLRI）および拡張コミュニティ（Ext Comm）について説明します。

図 14　　NLRI と拡張コミュニティ



VPLS LDP シグナリングにおける Cisco IOS XR と Cisco IOS 間の相互運用性

Cisco IOS ソフトウェアは、BGP アップデート メッセージ内で、最初のバイト内の NLRI の長さをビット型式でエンコードします。ただし、Cisco IOS XR ソフトウェアは、NLRI の長さを 2 バイトで解釈します。したがって、VPLS-VPWS アドレス ファミリを使用する BGP ネイバーが IOS と IOS XR 間に設定されている場合、NLRI の不一致が発生し、ネイバー間のフラッピングの原因になります。この競合を避けるために、IOS は **prefix-length-size 2** コマンドをサポートしています。IOS が IOS XR とともに動作するようにするには、このコマンドをイネーブルにする必要があります。IOS で **prefix-length-size 2** コマンドが設定されている場合、NLRI の長さはバイト単位でエンコードされます。この設定は、IOS を IOS XR とともに動作させるために必要です。

次に、**prefix-length-size 2** コマンドを使用した IOS の設定の例を示します。

```
router bgp 1
 address-family l2vpn vpls
  neighbor 5.5.5.2 activate
  neighbor 5.5.5.2 prefix-length-size 2 --------> NLRI length = 2 bytes
 exit-address-family
```

MAC アドレス関連パラメータ

MAC アドレス テーブルには、既知の MAC アドレスおよび転送情報のリストが含まれます。現在の VPLS 設計では、MAC アドレス テーブルと管理が配信されます。つまり、MAC アドレス テーブルのコピーは、ルート プロセッサ（RP）カードおよびラインカードで維持されます。

次のトピックでは、MAC アドレス関連パラメータについて説明します。

- MAC アドレス フラッディング（P.LSC-213）
- MAC アドレスベース転送（P.LSC-213）
- MAC アドレスの送信元ベースの学習（P.LSC-213）
- MAC アドレス エージング（P.LSC-213）

- MAC アドレス制限（P.LSC-214）
- MAC アドレス取り消し（P.LSC-214）
- MAC アドレスのセキュリティ（P.LSC-215）

（注） ブリッジ ドメイン レベルで MAC の制限またはアクションを修正した後で、アクションを有効にするために、ブリッジ ドメインを非アクティブにしてアクティブにしてください。（トラフィックが通過する）接続回線での MAC の制限またはアクションをまたは変更した後で、アクションを有効にするために、接続回線を非アクティブにしてアクティブにする必要があります。

MAC アドレス フラッディング

イーサネット サービスでは、ブロードキャスト アドレスおよび不明な宛先アドレスに送信されるフレームをすべてのポートにフラッディングする必要があります。VPLS ブロードキャスト モデル内のフラッディングを取得するために、すべての不明ユニキャスト、ブロードキャスト、およびマルチキャスト フレームが、対応する疑似回線およびすべての接続回線にフラッディングされます。したがって、PE は、接続回線および疑似回線の両方にパケットを複製する必要があります。

MAC アドレスベース転送

フレームを転送するには、PE は、宛先 MAC アドレスを疑似回線または接続回線に関連付ける必要があります。このタイプのアソシエーションは、各 PE で静的設定によって行われるか、すべてのブリッジ ポートにフラッディングされるダイナミック学習によって行われます。

（注） たとえば接続回線または疑似回線で着信するフレームが同じ疑似回線で送信されるような場合、スプリット ホライズンの転送が適用されます。1 つの疑似回線で受信される疑似回線フレームは、同じ仮想転送インスタンス（VFI）の他の疑似回線には複製されません。

MAC アドレスの送信元ベースの学習

フレームがブリッジ ポート（たとえば、疑似回線または接続回路）に到達し、受信側 PE ルータが送信元 MAC アドレスを認識していない場合、送信元 MAC アドレスは、疑似回線または接続回線に関連付けられます。MAC アドレスへの送信フレームは、適切な疑似回線または接続回線に転送されます。

MAC アドレスの送信元ベースの学習は、ハードウェア転送パスで学習される MAC アドレス情報を使用します。更新された MAC テーブルはすべてのラインカード（LC）に送信され、ルータのハードウェアをプログラミングします。

学習される MAC アドレスの数は、設定可能なポート単位およびブリッジ ドメイン単位の MAC アドレス制限によって制限されます。

MAC アドレス エージング

MAC テーブルの MAC アドレスは、MAC アドレス エージング タイムの間だけ有効と見なされます。期限切れになると、関連する MAC エントリが再度読み込まれます。MAC エージング タイムをブリッジ ドメインだけで設定すると、ブリッジ ドメインのすべての疑似回線と接続回線が、設定されたその MAC エージング タイムを使用されます。

ブリッジは、ブリッジ テーブルに基づいてパケットの転送、フラッディング、ドロップを行います。ブリッジ テーブルは、スタティック エントリとダイナミック エントリの両方を保持します。スタティック エントリは、ネットワーク マネージャまたはブリッジ自体によって入力されます。ダイナミック エントリはブリッジ学習プロセスによって入力されます。ダイナミック エントリは、エントリが作成された時点か最後に更新された時点から、*エージング タイム*と呼ばれる指定された期間が経過すると、自動的に削除されます。

ブリッジ型ネットワークのホストが移動する可能性が高い場合、ブリッジが変更に迅速に適応できるようにエージング タイムを小さくします。ホストが連続して送信しない場合は、より長い時間ダイナミック エントリを記録するようにエージング タイムを長くして、ホストが再度送信する場合よりフラッディングの可能性を低減できます。

MAC アドレス制限

MAC アドレス制限は、学習される MAC アドレスの数を制限するために使用されます。制限は、ブリッジ ドメイン レベルとポート レベルで設定されます。ブリッジ ドメイン レベルの制限は常に設定され、ディセーブルにできません。ブリッジ ドメイン レベルの制限のデフォルト値は 4000 で、5 ～ 512000 の範囲で変更できます。

（注） Cisco ASR 9000 シリーズ ルータでは、ブリッジ ドメインのすべてのポートで設定されている場合に限りブリッジ ポートの MAC 制限がサポートされます。この場合、ブリッジ ドメインの制限は、ブリッジ ドメインのすべてのポートの制限の合計よりも高い値に設定する必要があります。

MAC アドレス制限に違反した場合は、表 1 にリストされている処理のうちの 1 つを実行するようにシステムが設定されます。

表 1　MAC アドレス制限処理

アクション	説明
Limit flood	新しい MAC アドレスを廃棄します。
Limit no-flood	新しい MAC アドレスを廃棄します。未知のユニキャスト パケットのフラッディングはディセーブルです。
Limit shutdown	MAC アドレスの転送をディセーブルにします。

制限を超えると、これらの通知を行うようシステムが設定されています。

- Syslog（デフォルト）
- 簡易ネットワーク管理プロトコル（SNMP）トラップ
- Syslog および SNMP トラップ
- なし（通知なし）

MAC 制限状態をクリアするには、MAC の数が、設定されている制限の 75 % を下回る必要があります。

MAC アドレス取り消し

高速な VPLS コンバージェンスでは、ダイナミックに学習された MAC アドレスを削除または学習解除できます。ラベル配布プロトコル（LDP）アドレス取り消しメッセージが MAC アドレスのリストと送信されます。これらのアドレスは、対応する VPLS サービスに参加する他のすべての PE で取り消す必要があります。

Cisco IOS XR VPLS 実装では、ダイナミックに学習された MAC アドレスの部分は、デフォルトで MAC アドレス エージング メカニズムを使用してクリアされます。MAC アドレス取り消し機能は、LDP アドレス取り消しメッセージによって追加されます。MAC アドレス取り消し機能をイネーブルにするには、l2vpn ブリッジ グループ ブリッジ ドメイン MAC コンフィギュレーション モードで **withdrawal** コマンドを使用します。MAC アドレス取り消しがイネーブルであることを確認するには、**detail** キーワードとともに **show l2vpn bridge-domain** コマンドを使用します。

（注） デフォルトでは、Cisco IOS XR では LDP MAC 取り消し機能はイネーブルです。

LDP MAC 取り消し機能は、次のイベントが原因で生成されます。

- 接続回線がダウンします。CLI から接続回線を削除または追加できます。
- MAC 取り消しメッセージは、VFI 疑似回線で受信され、アクセス疑似回線経由で伝播されません。RFC 4762 は、（空のタイプ、長さ、値（TLV）によって）ワイルドカードおよび特定の MAC アドレス取り消しの両方を指定します。Cisco IOS XR ソフトウェアはワイルドカードの MAC アドレス取り消しだけをサポートします。

MAC アドレスのセキュリティ

インターフェイス レベルとブリッジ アクセス ポート（サブインターフェイス）レベルで MAC アドレス セキュリティを設定できます。ただし、インターフェイスで設定された MAC セキュリティは、ブリッジ ドメイン レベルで設定された MAC セキュリティよりも優先されます。MAC セキュリティで設定された EFP で MAC アドレスを最初に学習して、次に同じ MAC アドレスを別の EFP で学習すると、次のイベントが発生します。

- パケットはドロップされます。
- 2 番めの EFP はシャットダウンされます。
- パケットが学習され、元の EFP からの MAC はフラッシュされます。

LSP Ping over VPWS および VPLS

ソフトウェアでは、Cisco IOS XR（LDP FEC128 を使用してシグナリングされる）ポイントツーポイント疑似回線のラベル スイッチド パス（LSP）ping と traceroute 検証メカニズムの既存のサポートは、VFI（VPLS）に関連付けられている疑似回線をカバーするために拡張されています。現在、LSP ping と traceroute のサポートは、（LDP FEC128 を使用してシグナリングされ）手動で設定された VPLS 疑似回線に制限されています。仮想回線接続検証（VCCV）サポートおよび **ping mpls pseudowire** コマンドの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router MPLS Command Reference*』を参照してください。

スプリット ホライズン グループ

IOS XR ブリッジ ドメインは、スプリット ホライズン グループと呼ばれる 3 つのグループの 1 つに接続回線（AC）と疑似回線（PW）を集約します。ブリッジ ドメインに適用した場合、スプリット ホライズンは、スプリット ホライズン グループのメンバー間のフラッディングと転送動作を示します。一般に、スプリット ホライズン グループの 1 つのメンバーで受信したフレームは、同じグループの他のメンバーにフラッディングされません。

ブリッジ ドメイン トラフィックは、ユニキャストまたはマルチキャストのいずれかです。

フラッディング トラフィックは、不明なユニキャスト宛先 MAC アドレス フレームで構成されます。これは、イーサネット マルチキャスト アドレス（スパニングツリー BPDU など）に送信されるフレームや、イーサネット ブロードキャスト フレーム（MAC アドレス FF-FF-FF-FF-FF-FF）です。

既知のユニキャスト トラフィックは、MAC 学習を使用するポートから学習されたブリッジ ポートに送信されるフレームで構成されます。

トラフィック フラッディングは、ブロードキャスト、マルチキャスト、不明なユニキャスト宛先アドレスに対して実行されます。ユニキャスト トラフィックは、MAC 学習を使用して学習されたブリッジ ポートに送信されるフレームで構成されます。

表 2　Cisco IOS XR でサポートされているスプリット ホライズン グループ

スプリット ホライズン グループ	このグループに属しているメンバー	グループ内のマルチキャスト	グループ内のユニキャスト
0	デフォルト：グループ 1 または 2 でカバーされないメンバー。	Yes	Yes
1	VFI で設定されるすべての PW。	No	No
2	**split-horizon** キーワードで設定された AC または PW。	No	No

スプリット ホライズン グループに関する重要事項：

- ブリッジ ドメインのメンバーであるすべてのブリッジ ポートまたは PW が、3 つのグループのうちの 1 つに属している必要があります。
- デフォルトでは、すべてのブリッジ ポートまたは PW がグループ 0 のメンバーです。
- ブリッジ ドメイン設定の VFI コンフィギュレーション サブモードは、このドメインのメンバーがグループ 1 に含まれていることを示しています。
- グループ 0 で設定された PW はアクセス疑似回線と呼ばれます。
- **split-horizon group** コマンドは、グループ 2 のメンバーとしてブリッジ ポートまたは PW を指定するために使用されます。
- ASR9000 は 1 個の VFI グループだけをサポートします。

レイヤ 2 セキュリティ

次のトピックでは、レイヤ 2 セキュリティをサポートするレイヤ 2 VPN の拡張について説明します。

- ポート セキュリティ（P.LSC-216）
- DHCP スヌーピング（P.LSC-217）

ポート セキュリティ

ポートへのトラフィック送信を許可する MAC アドレスを制限することによって、ダイナミックに学習される MAC アドレス、およびスタティック MAC アドレスを使用したポート セキュリティを使用して、ポートの入力トラフィックを制限します。セキュア ポートにセキュア MAC アドレスを割り当てると、ポートは、定義されたアドレスのグループ外に送信元アドレスがある入力トラフィックを転送しません。セキュア MAC アドレスの数を 1 つに制限し、単一のセキュア MAC アドレスを割り当てると、そのポートに接続されているデバイスはそのポートの全帯域を使用できます。

次のポート セキュリティ機能がサポートされます。

- ブリッジまたはポートの MAC テーブルのサイズを制限します。
- MAC アドレスの処理と通知を容易にします。
- ブリッジまたはポートの MAC エージング タイムとモードをイネーブルにします。
- ブリッジまたはポートのスタティック MAC アドレスをフィルタリングします。
- セキュアまたは非セキュアとしてポートをマークします。
- ブリッジまたはポートでフラッディングをイネーブルまたはディセーブルにします。

ポートにセキュア MAC アドレスの最大数を設定した後で、次のいずれかの方法でアドレス テーブルにセキュア アドレスを組み込むようポート セキュリティを設定できます。

- **static-address** コマンドを使用して、すべてのセキュア MAC アドレスをスタティックに設定します。
- 接続されているデバイスの MAC アドレスで、ポートがセキュア MAC アドレスをダイナミックに設定できるようにします。
- アドレス数をいくつかスタティックに設定し、残りのアドレスがダイナミックに設定されるようします。

DHCP スヌーピング

Dynamic Host Configuration Protocol（DHCP）スヌーピングは、信頼できないホストと信頼済み DHCP サーバとの間のファイアウォールのように機能するセキュリティ機能です。DHCP スヌーピング機能は次のアクティビティを実行します。

- 信頼できない送信元からの DHCP メッセージを検証し、無効なメッセージをフィルタ処理して除外します。
- 信頼できるソースおよび信頼できないソースからの DHCP トラフィックのレートを制限する。
- DHCP スヌーピングのバインディング データベースを構築し、管理する。このデータベースには、リース IP アドレスがある信頼できないホストに関する情報が保存されています。
- 信頼できないホストからの以降の要求を検証するために DHCP スヌーピングのバインディング データベースを使用する。

DHCP に関する追加情報については、『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Configuration Guide*』を参照してください。

G.8032 イーサネット リング保護

ITU-T G.8032 で定義されているイーサネット リング保護（ERP）プロトコルは、リング トポロジでイーサネット トラフィックを保護し、イーサネット レイヤのリング内でループが発生しないようにします。ループは、事前設定されたリンクまたは障害リンクのいずれかをブロックすることで防止されます。

概要

各イーサネット リング ノードは、2 個の独立したリンクを使用してイーサネット リングに参加する隣接イーサネット リング ノードに接続されます。リング リンクは、ネットワークに影響を及ぼすループの編成を許可しません。イーサネット リングは、イーサネット リングを保護するために特定のリンクを使用します。この特定のリンクは、リング予備リンク（RPL）と呼ばれます。リング リンクは、リング リンク（別名リング ポート）の 2 個の隣接するイーサネット リング ノードとポートで区切られます。

（注） イーサネット リングでのイーサネット リング ノードの最小数は 2 です。

リング保護スイッチングの基礎は次のとおりです。

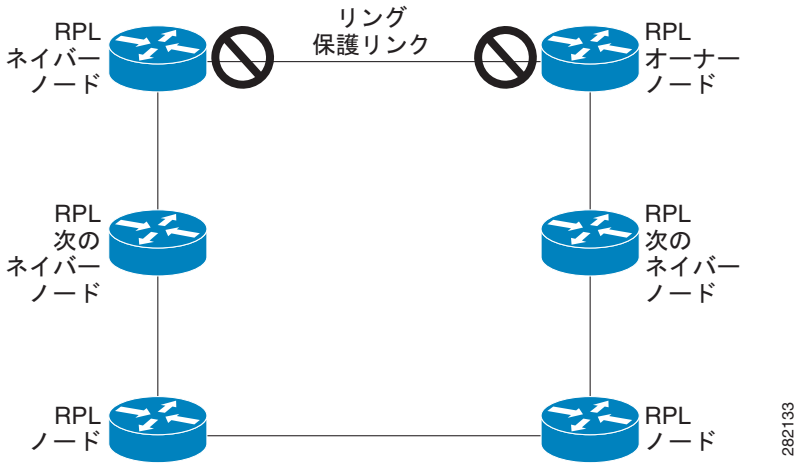
- ループ回避の原則
- 学習、転送、およびフィルタリング データベース（FDB）メカニズムの使用

イーサネット リングでのループ回避は、RPL である 1 つのリング リンクを除くすべてで常にトラフィック フローを確保することで行います。複数のノードが、リングの形成に使用されます。

- RPL オーナー：ループがイーサネット トラフィックで形成されないように、RPL を介してトラフィックをブロックします。リングには RPL オーナーは 1 つだけ存在します。
- RPL ネイバー ノード：RPL ネイバー ノードは、RPL に隣接するイーサネット リング ノードです。通常の状態で RPL の終了をブロックします。このノード タイプはオプションであり、保護されている場合 RPL の使用を防止します。
- RPL の次のネイバー ノード：RPL の次のネイバー ノードは、RPL オーナー ノードまたは RPL ネイバー ノードに隣接するイーサネット リング ノードです。これは、主にリングでの FDB フラッシュ最適化に使用されます。このノードはオプションです。

図 15 で、G.8032 イーサネット リングについて説明します。

図 15　　G.8032 イーサネット リング



リングのノードは、RAPS と呼ばれる制御メッセージを使用して、RPL リンクのオンとオフを切り替えるアクティビティを調整します。リンクの障害によって、障害が発生したリンクに面するポートをノードがブロックした後で、障害が発生したリンクに隣接するノードから両方の方向に RAPS 信号障害（RAPS SF）メッセージがトリガーされます。このメッセージの取得時に、RPL オーナーは、RPL ポートのブロックを解除します。

（注） リングの単一のリンク障害によって、ループフリー トポロジが確保されます。

リング リンクおよびノードの障害を検出するために、回線ステータスおよび接続障害管理プロトコルが使用されます。回復フェーズ中に、障害が発生したリンクが復元されると、復元されたリンクに隣接するノードは、RAPS no request（RAPS NR）メッセージを送信します。このメッセージの取得時に、RPL オーナーは RPL ポートをブロックし、RAPS no request, root blocked（RAPS NR, RB）メッセー

ジを送信します。これにより、リング内の RPL オーナー以外のその他すべてのノードが、すべてのブロックされたポートのブロックを解除します。ERP プロトコルは、リング トポロジの単方向障害と複数のリンク障害シナリオの両方で機能するために十分に強力です。

G.8032 リングは、次の基本的なオペレータ管理コマンドをサポートします。

- Force switch (FS)：オペレータは、特定のリング ポートを強制的にブロックできます。
  - 既存の SF 状態がある場合でも有効です。
  - サポートされるリング用の複数の FS コマンド。
  - 即時のメンテナンス操作を可能にするために使用できます。
- Manual switch (MS)：オペレータは、特定のリング ポートを手動でブロックできます。
  - 既存の FS または SF 状態では無効です。
  - 新しい FS または SF 状態によって上書きされます。
  - 複数の MS コマンドは、すべての MS コマンドを取り消します。
- Clear：リング ポートで既存の FS または MS コマンドを取り消します。
  - 非リバーティブ モードをクリアするために（RPL オーナーで）使用されます。

G.8032 リングは、複数のインスタンスをサポートできます。インスタンスは、物理的なリングに実行される論理リングです。そのようなインスタンスは、リング上のロード バランシング VLAN などのさまざまな理由で使用されます。たとえば、奇数の VLAN はリングの 1 方向に送信され、偶数の VLAN は他の方向に送信されることがあります。特定の VLAN は 1 つのインスタンスだけで設定できます。これらは複数のインスタンスと重複できません。重複すると、データ トラフィックまたは RAPS パケットは論理リングを通過する可能性があるため、望ましくありません。

G.8032 ERP は、リンク障害の検出に回線ステータスと接続障害管理（CFM）に依存する新しいテクノロジーを提供します。100ms の間隔で CFM Continuity Check Message（CCM）メッセージを実行することにより、SONET のようなスイッチング時間パフォーマンスとループ フリー トラフィックを実現できます。

イーサネット接続障害管理（CFM）と Ethernet Fault Detection（EFD）の設定の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』の「*Configuring Ethernet OAM on the Cisco ASR 9000 Series Router*」を参照してください。

タイマー

G.8032 は、競合状態および不要なスイッチング操作を回避するために異なる ERP タイマーを使用することを指定します。

- 遅延タイマー：RPL をブロックする前にネットワークが安定していることを確認するために RPL オーナーによって使用されます。
  - SF 状態の後で、SF が断続的に中断していないことを確認するために、Wait-to-Restore（WTR）タイマーが使用されます。WTR タイマーはオペレータが設定できます。デフォルトの時間間隔は 5 分です。時間間隔の範囲は 1 ～ 12 分です。
  - FS/MS コマンドの後で、バックグラウンド状態でないことを確認するために、Wait-to-Block タイマーが使用されます。

（注） Wait-to-Block タイマーは、Wait-to-Restore タイマーよりも短くなることがあります。

- ガード タイマー：状態の変更時にすべてのノードで使用されます。これは、潜在的な古いメッセージが不要な状態変更を引き起こさないようにします。ガード タイマーは設定可能であり、デフォルトの時間間隔は 500 ミリ秒です。時間間隔の範囲 10 ～ 2000 ミリ秒です。

- hold-off タイマー：断続的なリンク障害をフィルタリングするために、基盤となるイーサネット レイヤによって使用されます。hold-off タイマーは設定可能であり、デフォルトの時間間隔は 0 秒です。時間間隔の範囲は 0 ～ 10 秒です。
  - 障害は、このタイマーの有効期限が切れた場合だけリング保護メカニズムに報告されます。

単一のリンク障害

図 16 は、単一のリンク障害が発生した場合の保護スイッチングを表します。

図 16　　G.8032 の単一のリンク障害

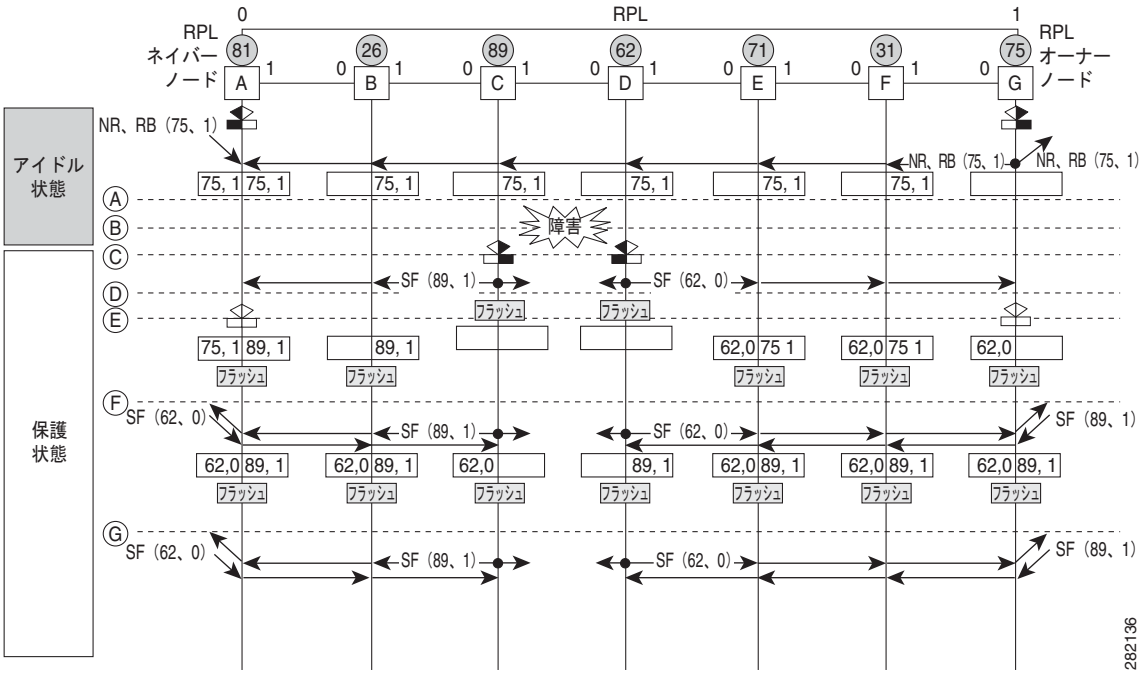


図 16 は、7 つのイーサネット リング ノードで構成されたイーサネット リングを表します。RPL は、イーサネット リング ノード A と G 間のリング リンクです。これらのシナリオでは、RPL の両端がブロックされます。イーサネット リング ノード G は RPL オーナー ノードで、イーサネット リング ノード A は RPL ネイバー ノードです。

次の記号が使用されます。

● メッセージの送信元
▶ R-APS チャネルのブロック
□ クライアント チャネルのブロック
ⓝ ノード ID　282135

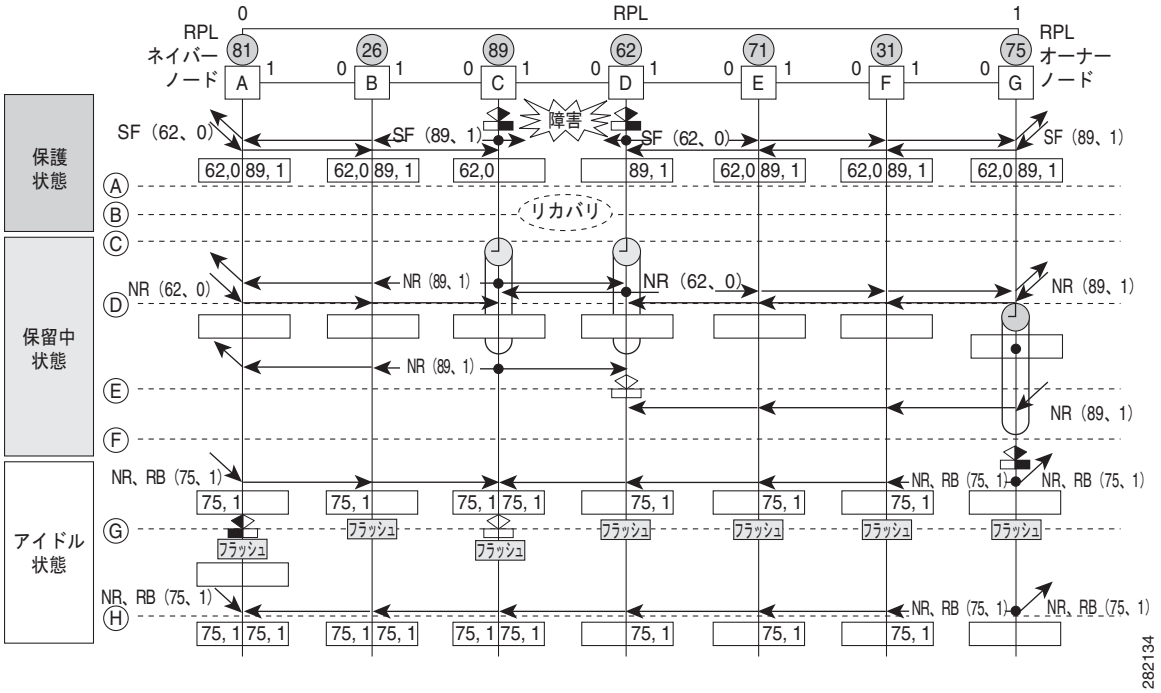
次の手順では、図 16 で表されている単一のリンク障害でのステップを説明します。

1. リンクは正常な状態で動作しています。
2. 障害が発生します。
3. イーサネット リング ノード C と D は、ローカルの信号障害を検出し、ホールドオフ時間間隔後に障害が発生したリング ポートをブロックし、FDB フラッシュを実行します。
4. イーサネット リング ノード C と D は、SF 状態が続いている間、両方のリング ポートの（ノード ID、BPR）ペアとともに RAPS（SF）メッセージの定期的な送信を開始します。
5. RAPS（SF）メッセージを受信するすべてのイーサネット リング ノードが FDB フラッシュを実行します。RPL オーナー ノード G および RPL のネイバー ノード A が RAPS（SF）メッセージを受信すると、イーサネット リング ノードは、RPL の終了のブロックを解除し、FDB フラッシュを実行します。

6. 2 番目の RAPS（SF）メッセージを受信するすべてのイーサネット リング ノードは、FDB フラッシュを再度実行します。これは、ノード ID と BPR ベース メカニズムが原因です。
7. 安定した SF 状態：イーサネット リングの RAPS メッセージ（SF）。これ以上の RAPS（SF）メッセージは、さらなるアクションをトリガーしません。

図 17 は、単一のリンク障害時の復帰を表します。

図 17　　単一のリンク障害回復（リバーティブ操作）



次の手順では、図 17 で表されている単一のリンク障害回復でのステップを説明します。

1. リンクが安定した SF 状態で動作しています。
2. リンク障害回復が行われます。
3. イーサネット リング ノード C と D は、信号障害（SF）状態のクリアを検出し、ガード タイマーを開始し、両方のリング ポートの RAPS（NR）メッセージの定期的な送信を開始します。（ガード タイマーは、RAPS メッセージの受信を防止します）。
4. イーサネット リング ノードが RAPS（NR）メッセージを受信すると、受信側リング ポートのノード ID および BPR のペアが削除され、RPL オーナー ノードは WTR タイマーを開始します。
5. イーサネット リング ノード C と D でガード タイマーの有効期限が切れると、受信する新しい RAPS メッセージを受け入れることがあります。イーサネット リング ノード D は、イーサネット リング ノード C から上位のノード ID を持つ RAPS（NR）メッセージを受信し、障害が発生していないリング ポートのブロックを解除します。
6. WTR タイマーの有効期限が切れると、RPL オーナー ノードは、RPL の終端をブロックし、（ノード ID、BPR）ペアを持つ RAPS（NR、RB）メッセージを送信し、FDB フラッシュを実行します。

7. イーサネット リング ノード C が RAPS（NR、RB）メッセージを受信すると、ブロックされたリング ポートのブロックを解除し、RAPS（NR）メッセージの送信を停止します。一方、RPL ネイバー ノード A が RAPS（NR、RB）メッセージを受信すると、RPL の終了をブロックします。さらに、イーサネット リング ノード A ～ F は、ノード ID と BPR ベース メカニズムが存在することが原因で、RAPS（NR、RB）メッセージを受信すると FDB フラッシュを実行します。

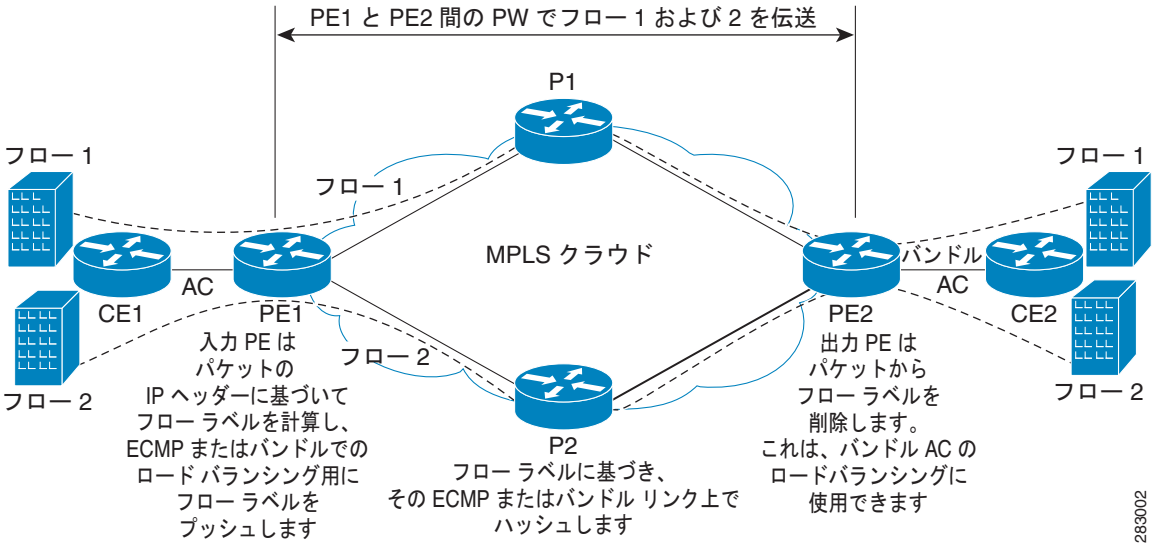
Flow Aware Transport 疑似回線（FAT PW）

ルータは通常、ラベル スタックの最低ラベル（特定の疑似回線のすべてのフローに対して同じラベル）に基づいてトラフィックをロードバランスします。このとき、非対称ロードバランシングが発生することがあります。このコンテキストでは、フローは同じ送信元/宛先ペアを持つパケットのシーケンスを示します。パケットは、送信元プロバイダー エッジ（PE）から宛先 PE に転送されます。

Flow-Aware Transport 疑似回線（FAT PW）は、疑似回線内の個々のフローを識別する機能を提供します。また、ルータに対してこれらのフローを使用してトラフィックをロードバランスする機能を提供します。等価コスト マルチパス（ECMP）が使用されている場合は、FAT PW はコア内のトラフィックのロードバランスに使用されます。疑似回線に伝送される個々のパケット フローに基づいてフロー ラベルが作成され、最低ラベルとしてパケットに挿入されます。ルータは、フロー ラベルをロードバランシングに使用できます。これにより、コア内の ECMP パスまたはリンクがバンドルされたパスでより適切なトラフィックの分配が実現します。

図 18 に、FAT PW と、ECMP およびバンドルされたリンクへ分配される 2 つのフローの例を示します。

図 18　FAT PW と ECMP およびバンドルされたリンクへ分配される 2 つのフロー



追加ラベルは、仮想回線（VC）のフロー情報を含むスタック（フロー ラベルと呼ばれる）に追加されます。フロー ラベルは、PW 内のフローを区別する一意の ID で、送信元/宛先 MAC アドレスと送信元/宛先 IP アドレスから取得されます。フロー ラベルにはラベル スタック（EOS）ビット セットの末尾が含まれ、VC ラベルの後ろや、コントロール ワード（存在する場合）の前に挿入されます。入力 PE は、フロー ラベルを計算し、転送します。FAT PW コンフィギュレーションは、フロー ラベルをイネーブルにします。出力 PE は、決定が行われないように、フロー ラベルを廃棄します。

すべてのコア ルータが、FAT PW でフローラベルに基づいてロード バランシングを実行します。これにより、ECMP とリンク バンドルへのフローの分配が可能になります。

疑似回線ヘッドエンド

疑似回線（PW）は、IP/MPLS パケットスイッチド ネットワーク（PSN）でのペイロードの透過的な伝送を可能にします。PW は、コア ネットワークに戻るカスタマー トラフィックのための、簡単で管理可能な軽いトンネルと見なされます。サービス プロバイダーは、PW 接続をネットワークのアクセスおよび集約の領域に拡張しています。

疑似回線ヘッドエンド（PWHE）は、レイヤ 3（VRF またはグローバル）ドメインまたはレイヤ 2 ドメインへのアクセス疑似回線（PW）の終端を可能にするテクノロジーです。PW は、共通の IP/MPLS ネットワーク インフラストラクチャへのカスタマー トラフィックのトンネリングのために、簡単でスケーラブルなメカニズムを提供します。PWHE により、カスタマーは、サービス プロバイダー エッジ（PE）ルータ上で、QoS アクセス リスト（ACL）、L3VPN などの機能を PWHE インターフェイス単位でプロビジョニングできます。

PWHE の利点

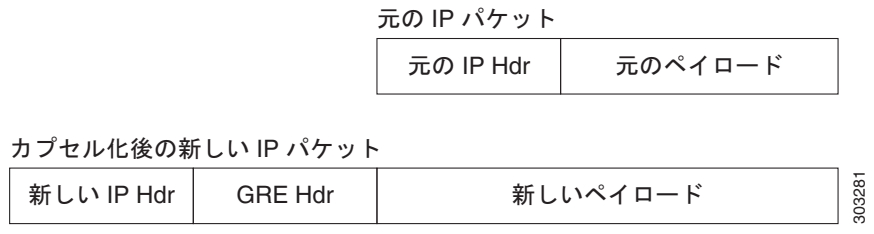
PWHE の実装には次のような利点があります。

- アクセスまたは集約ネットワークの基礎となる物理転送メディアからの、サービス PE のカスタマー側インターフェイス（CFI）の分離
- アクセスまたは集約ネットワークおよびサービス PE の CapEx の削減
- カスタマー側レイヤ 2 UNI インターフェイス セットの分配および拡大
- OAM 機能の統一方法の実装
- プロバイダーによるレイヤ 3 サービスのフットプリントの延長または拡張が可能
- 次世代ネットワーク（NGN）にカスタマー トラフィックの終端方法を提供

L2VPN over GRE

システムは、総称ルーティング カプセル化（GRE）トンネル上で IP パケットを転送するために、最初に GRE ヘッダーで元の IP パケットをカプセル化します。カプセル化された GRE パケットは、パケットを宛先に転送するために使用する外部の転送ヘッダーによって再びカプセル化されます。図 19 に、IP 転送ネットワークでの GRE のカプセル化の例を示します。

図 19　　GRE のカプセル化



（注）　新しい IP パケットでは、新しいペイロードは元の IP パケットに似ています。また、新しい IP ヘッダー（新しい IP Hdr）は、トンネル IP ヘッダーに似ており、転送ヘッダーにも似ています。

GRE トンネル エンドポイントで GRE パケットのカプセル化が解除されると、ペイロード タイプに基づいてそのパケットが転送されます。たとえば、ペイロードがラベル付きパケットの場合は、仮想回線（VC）ラベルまたは VPN ラベルに基づいて、L2VPN および L3VPN にそれぞれ転送されます。

GRE 配置シナリオ

L2VPN ネットワークでは、次のシナリオで GRE を配置できます。

- プロバイダー エッジ（PE）と PE ルータ間に GRE トンネルを設定
- P ルータと P ルータ間に GRE トンネルを設定
- P ルータと PE ルータ間に GRE トンネルを設定

次の図は、さまざまなシナリオを示します。

図 20　PE ルータと PE ルータ間に設定された GRE トンネル

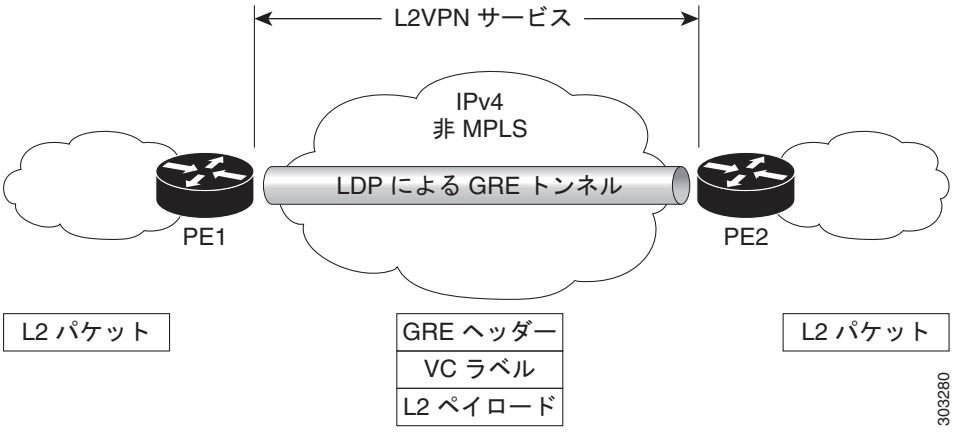


図 21　P ルータと P ルータ間に設定された GRE トンネル

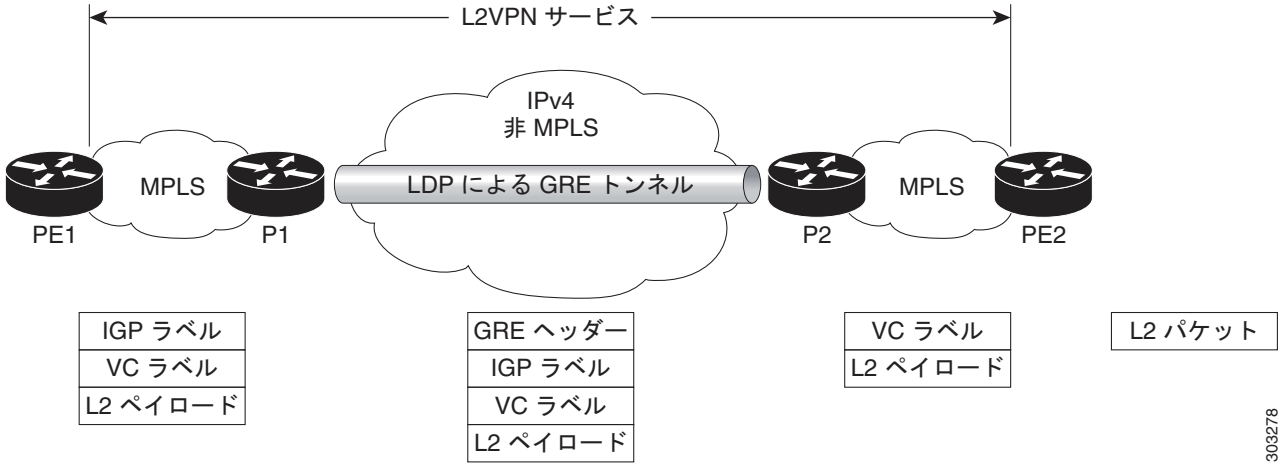
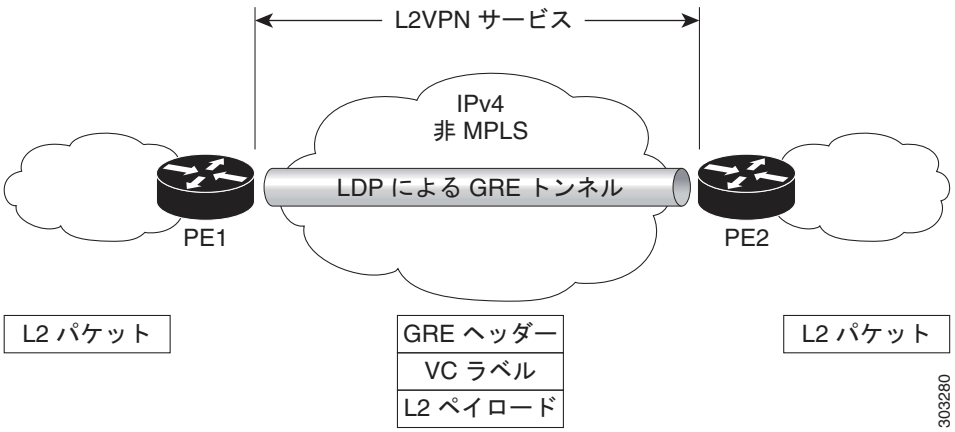


図 22　P ルータと PE ルータ間に設定された GRE トンネル



（注）　これらの配置シナリオは、VPWS および VPLS に適用されます。

優先パスとしての GRE トンネル

優先トンネル パス機能により、疑似回線を特定の GRE トンネルにマッピングできます。接続回線は、リモート PE ルータの IP アドレス（IGP または LDP を使用して到達可能）ではなく、GRE トンネル インターフェイスに相互接続されます。優先トンネル パスを使用する場合、L2 トラフィックを転送する GRE トンネルが 2 台の PE ルータ間で動作することが常に想定されます（つまり、始端はインポジション PE ルータで、終端はディスポジション PE ルータです）。

マルチポイント レイヤ 2 サービスの実装方法

ここでは、VPLS の実装に必要なタスクについて説明します。

ブリッジ ドメインの設定

次のトピックでは、ブリッジ ドメインの設定方法について説明します。

ブリッジ ドメインの作成

ブリッジ ドメインを作成するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを含めることができるブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線の設定

ブリッジ ドメインで疑似回線を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **exit**
7. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
8. **dhcp ipv4 snoop profile** {*dhcp_snoop_profile_name*}
9. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン VFI コンフィギュレーション モードを開始します。 • 指定した仮想転送インターフェイス名を設定するには、*vfi-name* 引数を使用します。
ステップ 6	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# exit` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	現在のコンフィギュレーション モードを終了します。
ステップ 7	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 **（注）** *A.B.C.D* は再帰的または非再帰的プレフィクスです。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。

	コマンドまたはアクション	目的
ステップ 8	**dhcp ipv4 snoop profile** {*dhcp_snoop_profile_name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# dhcp ipv4 snoop profile profile1`	ブリッジ上で DHCP スヌーピングをイネーブルにして、DHCP スヌーピング プロファイルを対応付けます。
ステップ 9	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

メンバのブリッジ ドメインへの関連付け

ブリッジ ドメインの作成後、ブリッジ ドメインにインターフェイスを割り当てるには、この作業を実行します。次のタイプのブリッジ ポートは、ブリッジ ドメインに関連付けられています。

- イーサネットおよび VLAN
- VFI

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **interface** *type interface-path-id*
6. **static-mac-address** {*MAC-address*}
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/4/0/0` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#`	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、同じブリッジ ドメインに属する他のインターフェイスからパケットを転送および受信できるブリッジ ドメインにインターフェイスを追加します。
ステップ 6	**static-mac-address** {*MAC-address*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# static-mac-address 1.1.1`	スタティック MAC アドレスを設定してリモート MAC アドレスを疑似回線またはその他のブリッジ インターフェイスに関連付けます。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ブリッジ ドメイン パラメータの設定

ブリッジ ドメイン パラメータを設定するには、ブリッジ ドメインに次のパラメータを関連付けます。

- Maximum transmission unit (MTU) : ブリッジ ドメインのすべてのメンバーに同じ MTU があることを指定します。MTU サイズが異なるブリッジ ドメイン メンバーは、まだブリッジ ドメインに関連付けられている場合でもブリッジ ドメインによって使用されません。
- Flooding : ブリッジ ドメインのフラッディングをイネーブルまたはディセーブルにします。デフォルトでは、フラッディングはイネーブルです。
- Dynamic ARP Inspection (DAI) : 有効な ARP 要求と応答だけが中継されるようにします。
- IP SourceGuard (IPSG) : レイヤ 2 ポートで送信元 IP アドレス フィルタリングをイネーブルにします。

（注） DAI および IPSG 機能が正常に動作していることを確認するには、DAI および IPSG 違反についてパケット ドロップ統計情報を調べます。パケット ドロップ統計情報は、**show l2vpn bridge-domain** *bd-name* <> **detail** コマンドの出力で確認できます。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **flooding disable**
6. **mtu** *bytes*
7. **dynamic-arp-inspection** {**address-validation** | **disable** | **logging**}
8. **ip-source-guard logging**
9. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN 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	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**flooding disable** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# flooding disable`	ブリッジ ドメイン レベルまたはブリッジ ポート レベルでトラフィックのフラッディングを設定します。
ステップ 6	**mtu** *bytes* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mtu 1000`	ブリッジ ドメインの最大パケット サイズまたは最大伝送単位（MTU）サイズを調整します。 • バイト単位で MTU サイズを指定するには、*bytes* 引数を使用します。範囲は 64 ～ 65535 です。
ステップ 7	**dynamic-arp-inspection** {**address-validation** | **disable** | **logging**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# dynamic-arp-inspection`	ダイナミック ARP インスペクション コンフィギュレーション サブモードを開始します。有効な ARP 要求および応答だけがリレーされるようになります。 **（注）** ブリッジ ドメインまたはブリッジ ポートのダイナミック ARP インスペクションを設定できます。

	コマンドまたはアクション	目的
ステップ 8	**ip-source-guard logging** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# ip-source-guard logging`	IP ソース ガード コンフィギュレーション サブモードを開始し、レイヤ 2 ポート上で送信元 IP アドレス フィルタリングをイネーブルにします。 ブリッジ ドメインまたはブリッジ ポートで IP ソース ガードをイネーブルにできます。デフォルトでは、ブリッジの下のブリッジ ポートは親ブリッジから IP ソース ガード設定を継承します。 デフォルトでは、すべてのブリッジに対して IP ソース ガードがディセーブルです。
ステップ 9	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ブリッジ ドメインのディセーブル化

ブリッジ ドメインをディセーブルにするには、次の作業を実行します。ブリッジ ドメインをディセーブルにすると、ブリッジ ドメインに関連付けられているすべての VFI がディセーブルになります。引き続き、ブリッジ ドメインに関連付けられたブリッジ ドメインと VFI にメンバーを接続するか、または取り外すことができます。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **shutdown**
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、l2vpn ブリッジ グループ ブリッジ ドメイン 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 5	**shutdown** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインをシャットダウンし、ブリッジと、ブリッジ下のすべての接続回線と疑似回線を管理ダウン状態に戻します。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

不明なユニキャスト フラッディングのブロック

ブリッジ ドメイン レベルで不明なユニキャスト トラフィックのフラッディングをディセーブルにするには、次の作業を実行します。

ブリッジ ドメイン、ブリッジ ポート、またはアクセス疑似回線レベルで不明なユニキャスト トラフィックのフラッディングをディセーブルにできます。デフォルトでは、不明なユニキャスト トラフィックは、ブリッジ ドメインのすべてのポートにフラッディングされます。

（注） ブリッジ ドメインで不明なユニキャスト トラフィックのフラッディングをディセーブルにすると、ブリッジ ドメイン内のすべてのポートがこの設定を継承します。ブリッジ ドメイン設定を上書きするように、ブリッジ ポートを設定できます。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group name*
4. **bridge-domain** *bridge-domain name*
5. **flooding unknown-unicast disable**
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。

	コマンドまたはアクション	目的
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、l2vpn ブリッジ グループ ブリッジ ドメイン 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	**flooding unknown-unicast disable** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# flooding unknown-unicast disable`	ブリッジ ドメイン レベルで不明なユニキャスト トラフィックのフラッディングをディセーブルにします。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

フラッディング最適化モードの変更

ブリッジ ドメインでフラッディング最適化モードを変更するには、次の作業を行います。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group name*
4. **bridge-domain** *bridge-domain name*
5. **flood mode convergence-optimized**

6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、l2vpn ブリッジ グループ ブリッジ ドメイン 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 5	**flood mode convergence-optimized** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# flood mode convergence-optimized`	デフォルトのフラッディング最適化モードを帯域幅最適化モードからコンバージェンス モードに変更します。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

レイヤ 2 セキュリティの設定

次のトピックでは、レイヤ 2 セキュリティの設定方法について説明します。

レイヤ 2 セキュリティのイネーブル化

ブリッジのレイヤ 2 ポート セキュリティをイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **security**
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	各ネットワーク インターフェイスをブリッジ グループに割り当てて、L2VPN ブリッジ グループ 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	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**security** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# security`	ブリッジのレイヤ 2 ポート セキュリティをイネーブルにします。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • end コマンドを実行すると、変更をコミットするように要求されます。 `uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

Dynamic Host Configuration Protocol（DHCP）プロファイルの対応付け

ブリッジ上で DHCP スヌーピングをイネーブルにし、ブリッジに DHCP スヌーピング プロファイルを対応付けるには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **dhcp ipv4 snoop** {**profile** *profile-name*}
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	各ネットワーク インターフェイスをブリッジ グループに割り当てて、L2VPN ブリッジ グループ 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	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**dhcp ipv4 snoop** {**profile** *profile-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# dhcp ipv4 snoop profile attach`	ブリッジ上で DHCP スヌーピングをイネーブルにし、ブリッジに DHCP スヌーピング プロファイルを対応付けます。 • DHCP プロファイルを対応付けるには、profile キーワードを使用します。profile-name 引数は、DHCPv4 スヌーピングのプロファイル名です。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit`	設定変更を保存します。 • end コマンドを実行すると、変更をコミットするように要求されます。 `uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

レイヤ 2 仮想転送インスタンスの設定

次のトピックでは、レイヤ 2 仮想転送インスタンス（VFI）の設定方法について説明します。

ブリッジ ドメインの仮想転送インスタンスの追加

ブリッジ ドメインのすべてのプロバイダー エッジ（PE）デバイスでレイヤ 2 仮想転送インスタンス（VFI）を作成するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。

	コマンドまたはアクション	目的
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン VFI コンフィギュレーション モードを開始します。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線の仮想転送インスタンスへの関連付け

VFI を作成したら、1 つ以上の疑似回線を VFI に関連付けるには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 6	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ブリッジ ドメインへの仮想転送インスタンスの関連付け

VFI をブリッジ ドメインのメンバーになるように関連付けるには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
7. **static-mac-address** {*MAC-address*}
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 6	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 7	**static-mac-address** {*MAC-address*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# static-mac-address 1.1.1`	スタティック MAC アドレスを設定してリモート MAC アドレスを疑似回線またはその他のブリッジ インターフェイスに関連付けます。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線への疑似回線クラスの接続

疑似回線に疑似回線クラスを接続するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
7. **pw-class** {*class-name*}
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 6	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 7	**pw-class** {*class-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# pw-class canada`	疑似回線に使用する疑似回線クラス テンプレート名を設定します。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

スタティック ラベルを使用した Any Transport over Multiprotocol 疑似回線の設定

スタティック ラベルを使用して Any Transport over Multiprotocol（AToM）疑似回線を設定するには、次の作業を実行します。疑似回線は、ローカルとリモートに MPLS スタティック ラベルを設定することでスタティック AToM 疑似回線になります。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
7. **mpls static label** {**local** *value*} {**remote** *value*}
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン VFI コンフィギュレーション モードを開始します。
ステップ 6	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#` `neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 7	**mpls static label** {**local** *value*} {**remote** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#` `mpls static label local 800 remote 500`	MPLS スタティック ラベルおよびアクセス疑似回線コンフィギュレーションのスタティック ラベルを設定します。ローカルおよびリモートの疑似回線ラベルを設定できます。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#` `end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#` `commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

仮想転送インスタンスのディセーブル化

VFI をディセーブルにするには、次の作業を実行します。VFI がディセーブルの場合、VFI に関連付けられた、以前に確立された疑似回線はすべて切断されます。LDP アドバタイズメントは、VFI に関連付けられた MAC アドレスを回収するために送信されます。ただし、シャットダウン後にも引き続き接続回線を VFI に接続したり切断したりできます。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** {*vfi-name*}
6. **shutdown**
7. **end**
   または
   commit
8. **show l2vpn bridge-domain** [**detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi v1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン VFI コンフィギュレーション モードを開始します。
ステップ 6	**shutdown** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# shutdown`	仮想転送インターフェイス（VFI）をディセーブルにします。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 8	**show l2vpn bridge-domain** [**detail**] **例：** `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain detail`	VFI の状態を表示します。たとえば、VFI をシャットダウンすると、VFI はブリッジ ドメインでシャットダウンされていると示されています。

MAC アドレス関連パラメータの設定

次のトピックでは、MAC アドレス関連パラメータの設定方法について説明します。

- MAC アドレスの送信元ベースの学習の設定（P.LSC-259）
- MAC アドレス回収のイネーブル化（P.LSC-262）
- MAC アドレス制限の設定（P.LSC-264）
- MAC アドレス エージングの設定（P.LSC-267）
- ブリッジ ポート レベルでの MAC フラッシュのディセーブル化（P.LSC-270）
- MAC アドレスのセキュリティの設定（P.LSC-272）

MAC テーブル属性は、ブリッジ ドメインについて設定されます。

MAC アドレスの送信元ベースの学習の設定

MAC アドレスの送信元ベースの学習を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **mac**
6. **learning disable**
7. **end**
   または
   commit
8. **show l2vpn bridge-domain** [**detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN 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	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**mac** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mac` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#`	L2VPN ブリッジ グループ ブリッジ ドメイン MAC 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	**learning disable** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# learning disable`	ブリッジ ドメイン レベルで MAC 学習をディセーブルにします。

	コマンドまたはアクション	目的
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 8	**show l2vpn bridge-domain** [**detail**] **例：** `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain detail`	MAC アドレスの送信元ベースの学習がブリッジでディセーブルになったことの詳細が表示されます。

MAC アドレス回収のイネーブル化

指定されたブリッジ ドメインの MAC アドレス回収をイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **mac**
6. **withdrawal**
7. **end**
   または
   commit
8. **show l2vpn bridge-domain** [**detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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	**mac** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mac` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#`	L2VPN ブリッジ グループ ブリッジ ドメイン MAC 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 6	**withdrawal** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# withdrawal`	特定のブリッジ ドメインについて MAC アドレス回収をイネーブルにします。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 8	**show l2vpn bridge-domain** [**detail**] **例：** `P/0/RSP0/CPU0:router# show l2vpn bridge-domain detail`	MAC アドレス回収をイネーブルにすることを指定する詳細な出力例が表示されます。また、出力例には、疑似回線から送信または受信した MAC 回収メッセージの数が表示されます。

MAC アドレス制限の設定

MAC アドレス制限のパラメータを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **mac**
6. **limit**
7. **maximum** {*value*}
8. **action** {**flood** | **no-flood** | **shutdown**}
9. **notification** {**both** | **none** | **trap**}
10. **end**
    または
    commit
11. **show l2vpn bridge-domain** [**detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**mac** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mac` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#`	L2VPN ブリッジ グループ ブリッジ ドメイン MAC 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	**limit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# limit` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-limit)#`	アクション、最大、通知の MAC アドレス制限を設定し、L2VPN ブリッジ グループ ブリッジ ドメイン MAC 制限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	**maximum** {*value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-limit)# maximum 5000`	ブリッジで学習される MAC アドレスの数が制限に到達したときの特定のアクションを設定します。
ステップ 8	**action** {**flood** | **no-flood** | **shutdown**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-limit)# action flood`	学習される MAC アドレスの数が設定された MAC 制限を超えたときのブリッジの動作を設定します。
ステップ 9	**notification** {**both** | **none** | **trap**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-limit)# notification both`	学習される MAC アドレスの数が設定された制限を超えたときに送信される通知のタイプを指定します。

	コマンドまたはアクション	目的
ステップ 10	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-limit)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-limit)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 11	**show l2vpn bridge-domain** [**detail**] **例：** `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain detail`	MAC アドレス制限の詳細が表示されます。

MAC アドレス エージングの設定

MAC アドレス エージングのパラメータを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **mac**
6. **aging**
7. **time** {*seconds*}
8. **type** {**absolute** | **inactivity**}
9. **end**
   または
   commit
10. **show l2vpn bridge-domain** [**detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**mac** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mac` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#`	L2VPN ブリッジ グループ ブリッジ ドメイン MAC 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	**aging** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# aging` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-aging)#`	MAC エージング コンフィギュレーション サブモードを開始し、時間やタイプなどのエージング パラメータを設定します。
ステップ 7	**time** {*seconds*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-aging)# time 300`	最大エージング タイムを設定します。 • MAC アドレス テーブル エントリの最大経過時間を指定するには、*seconds* 引数を使用します。範囲は 120 ～ 1000000 です。エージング タイムは最後にスイッチが MAC アドレスを検出した時点からカウントされます。デフォルト値は 300 秒です。
ステップ 8	**type** {**absolute** | **inactivity**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-aging)# type absolute`	MAC アドレス エージングを設定します。 • 絶対エージング タイプを設定するには、**absolute** キーワードを使用します。 • 非活動エージング タイプを設定するには、**inactivity** キーワードを使用します。

	コマンドまたはアクション	目的
ステップ 9	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-aging)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac-aging)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 10	**show l2vpn bridge-domain** [**detail**] **例：** `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain detail`	エージング フィールドに関する詳細を表示します。

ブリッジ ポート レベルでの MAC フラッシュのディセーブル化

ブリッジ ドメイン レベルで MAC フラッシュをディセーブルにするには、次の作業を実行します。

ブリッジ ドメイン、ブリッジ ポートまたはアクセス疑似回線レベルで MAC フラッシュをディセーブルにできます。デフォルトでは、そのポートが機能しなくなると、特定のポートで学習される MAC はただちにフラッシュされます。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group name*
4. **bridge-domain** *bridge-domain name*
5. **mac**
6. **port-down flush disable**
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、l2vpn ブリッジ グループ ブリッジ ドメイン 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 5	**mac** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mac` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#`	l2vpn ブリッジ グループ ブリッジ ドメイン MAC 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	**port-down flush disable** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#` `port-down flush disable`	ブリッジ ポートが機能しなくなったら、MAC フラッシュをディセーブルにします。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)#` `commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

MAC アドレスのセキュリティの設定

MAC アドレスのセキュリティを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group name*
4. **bridge-domain** *bridge-domain name*
5. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
6. **mac**
7. **secure**
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn` `RP/0/RSP0/CPU0:router(config-l2vpn)#`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	ブリッジ ドメインを確立し、l2vpn ブリッジ グループ ブリッジ ドメイン 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 5	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線をブリッジ仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 6	**mac** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# mac` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw-mac)#`	l2vpn ブリッジ グループ ブリッジ ドメイン MAC 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	**secure** [**action** | **disable** | **logging**] **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw-mac)# secure` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw-mac-secure)#`	MAC セキュア コンフィギュレーション モードを開始します。 デフォルトでは、ブリッジの下のブリッジ ポート（インターフェイスおよびアクセス疑似回線）は親ブリッジからセキュリティ設定を継承します。 **（注）** ブリッジ ポートがダウンした後は、ブリッジポートをアップにするには **clear** コマンドを発行する必要があります。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw-mac-secure)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw-mac-secure)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

AC スプリット ホライズン グループへの接続回線の設定

次の手順では、ブリッジ ドメインの接続回線（AC）のスプリット ホライズン グループにインターフェイスを追加する方法を示します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **interface** *type instance*
6. **split-horizon group**
7. **commit**
8. **end**
9. **show l2vpn bridge-domain detail**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group metroA`	名前付きブリッジ グループの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	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain east`	名前付きブリッジ ドメインの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	**interface** *type instance* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet0/1/0/6`	指定されたインターフェイスの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 6	**split-horizon group** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# **split-horizon group**	AC のスプリット ホライズン グループにこのインターフェイスを追加します。ブリッジ ドメインの AC のスプリット ホライズン グループは 1 つだけサポートされます。
ステップ 7	**commit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# **commit**	設定変更を保存します。
ステップ 8	**end** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# **end**	EXEC モードに戻ります。
ステップ 9	**show l2vpn bridge-domain detail** **例：** RP/0/RSP0/CPU0:router# **show l2vpn bridge-domain detail**	各 AC が AC スプリット ホライズン グループに属しているかどうかを含め、ブリッジに関する情報を表示します。

AC スプリット ホライズン グループへのアクセス疑似回線の追加

次の手順では、ブリッジ ドメインの接続回線（AC）のスプリット ホライズン グループのメンバーとしてアクセス疑似回線を追加する方法を示します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
6. **split-horizon group**
7. **commit**
8. **end**
9. **show l2vpn bridge-domain detail**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group metroA`	名前付きブリッジ グループの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	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain east`	名前付きブリッジ ドメインの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	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# neighbor 10.2.2.2 pw-id 2000`	疑似回線セグメントを設定します。

	コマンドまたはアクション	目的
ステップ 6	**split-horizon group** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# **split-horizon group**	AC のスプリット ホライズン グループにこのアクセス疑似回線を追加します。 **（注）** ブリッジ ドメインごとに AC とアクセス疑似回線のスプリット ホライズン グループは 1 つだけサポートされます。
ステップ 7	**commit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# **commit**	設定変更を保存します。
ステップ 8	**end** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# **end**	EXEC モードに戻ります。
ステップ 9	**show l2vpn bridge-domain detail** **例：** RP/0/RSP0/CPU0:router# **show l2vpn bridge-domain detail**	各アクセス疑似回線が AC スプリット ホライズン グループに属しているかどうかを含め、ブリッジに関する情報を表示します。

BGP オートディスカバリおよびシグナリングでの VPLS の設定

BGP ベースのオートディスカバリとシグナリングを設定するには、次の作業を実行します。

この設定で使用されるコマンドのマニュアルを検索するには、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』の「*Multipoint Layer 2 Services Commands*」を参照してください。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **vfi** *{vfi-name}*
6. **vpn-id** *vpn-id*
7. **autodiscovery bgp**
8. **rd** {*as-number:nn* | *ip-address:nn* | **auto**}
9. **route-target** {*as-number:nn* | *ip-address:nn* | **export** | **import**}
10. **route-target import** {*as-number:nn* | *ip-address:nn*}
11. **route-target export** {*as-number:nn* | *ip-address:nn*}
12. **signaling-protocol bgp**
13. **ve-id** {*number*}

14. **ve-range** {*number*}
15. **commit**
    または
    end

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group metroA`	名前付きブリッジ グループの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	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain east`	名前付きブリッジ ドメインの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	**vfi** *{vfi-name}* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi vfi-east`	仮想転送インスタンス（VFI）コンフィギュレーション モードを開始します。
ステップ 6	**vpn-id** *vpn-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# vpn-id 100`	VPLS サービスの ID を指定します。VPN ID は、PE ルータ内でグローバルに一意でなければなりません。つまり、同じ VPN ID が同じ PE ルータの複数の VFI に存在することはできません。また、VFI に指定できる VPN ID は 1 つだけです。
ステップ 7	**autodiscovery bgp** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# autodiscovery bgp`	すべての BGP オートディスカバリ パラメータが設定される BGP オートディスカバリ コンフィギュレーション モードを開始します。 このコマンドは、少なくとも VPN ID とシグナリング プロトコルが設定されるまで、BGP にプロビジョニングされません。

	コマンドまたはアクション	目的
ステップ 8	**rd** {*as-number:nn*|*ip-address:nn*|**auto**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# rd auto`	VFI でルート識別子（RD）を指定します。 RD は、VFI を識別するために BGP NLRI で使用されます。VFI ごとに RD を 1 つだけ設定できます。**rd auto** を除き、RD は同じ PE の複数の VFI で設定できません。 **rd auto** が設定されている場合、RD 値は、{BGP Router ID}:{16 bits auto-generated unique index} のようになります。
ステップ 9	**route-target** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# route-target 500:99`	VFI のルート ターゲット（RT）を指定します。 PE 間の BGP オートディスカバリを設定するには、少なくとも 1 つのインポートと 1 つのエクスポート ルート ターゲット（または両方のロールを持つ 1 つのルート ターゲットだけ）を各 PE で設定する必要があります。 export または import キーワードが指定されていない場合、RT はインポートおよびエクスポートの両方であることを意味します。VFI には、複数のエクスポートまたはインポート RT を設定できます。ただし、同じ PE の複数の VFI で、同じ RT を使用することはできません。
ステップ 10	**route-target import** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# route-target import 200:20`	VFI のインポート ルート ターゲットを指定します。 インポート ルート ターゲットは、PE が受信した NLRI の RT と比較する項目です。RT が同じ VPLS サービスに属することを判断するには、受信した NLRI の RT がインポート RT と一致する必要があります。
ステップ 11	**route-target export** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# route-target export 100:10`	VFI のエクスポート ルート ターゲットを指定します。 エクスポート ルートターゲットは、他の PE にアドバタイズされる NLRI 内に含められる RT です。
ステップ 12	**signaling-protocol bgp** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# signaling-protocol bgp`	BGP シグナリングをイネーブルにして、BGP シグナリング パラメータが設定される BGP シグナリング コンフィギュレーション サブモードを開始します。 このコマンドは、VE ID と VE ID の範囲が設定されるまで BGP にプロビジョニングされません。
ステップ 13	**ve-id** {*number*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# ve-id 10`	VPLS を設定するために VFI のローカル PE ID を指定します。 VE ID は、VPLS サービス内の VFI を識別します。これは、同じ VPLS サービスの VFI が同じ VE ID を共有できないことを意味します。VE ID のスコープは、ブリッジ ドメイン内だけに存在します。したがって、PE 内の異なるブリッジ ドメインの VFI は、同じ VE ID を使用できます。

	コマンドまたはアクション	目的
ステップ 14	**ve-range** {*number*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# ve-range 40`	VPLS エッジ（VE）ブロックの最小サイズを上書きします。 デフォルトの最小サイズは 10 です。設定する VE の範囲は、10 よりも高い必要があります。
ステップ 15	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

BGP オートディスカバリおよび LDP シグナリングでの VPLS の設定

BGP ベースのオートディスカバリとシグナリングを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **route-id**
4. **bridge group** *bridge-group-name*
5. **bridge-domain** *bridge-domain-name*
6. **vfi** *{vfi-name}*
7. **autodiscovery bgp**
8. **vpn-id** *vpn-id*
9. **rd** {*as-number:nn* | *ip-address:nn* | **auto**}
10. **route-target** {*as-number:nn* | *ip-address:nn* | **export** | **import**}
11. **route-target import** {*as-number:nn* | *ip-address:nn*}

12. **route-target export** {*as-number:nn* | *ip-address:nn*}
13. **signaling-protocol ldp**
14. **vpls-id** {*as-number:nn* | *ip-address:nn*}
15. **commit**
    または
    end

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**router-id** *ip-address* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# router-id 1.1.1.1`	プロバイダー エッジ（PE）ルータの一意のレイヤ 2（L2）ルータ ID を指定します。 ルータ ID は、LDP シグナリング用に設定する必要があり、BGP NLRI、SAII（ローカル L2 ルータ ID）、および TAII（リモート L2 ルータ ID）で L2 ルータ ID として使用されます。IPv4 アドレス形式の任意の値を使用できます。 **（注）** 各 PE には一意の L2 ルータ ID が必要です。PE が LDP ルータ ID を使用して自動的に L2 ルータ ID を生成するため、この CLI はオプションです。
ステップ 4	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group metroA`	名前付きブリッジ グループの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	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain east`	名前付きブリッジ ドメインの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	**vfi** *{vfi-name}* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi vfi-east`	仮想転送インスタンス（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 7	**vpn-id** *vpn-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# vpn-id 100`	VPLS サービスの ID を指定します。VPN ID は、PE ルータ内でグローバルに一意でなければなりません。つまり、同じ VPN ID が同じ PE ルータの複数の VFI に存在することはできません。また、VFI に指定できる VPN ID は 1 つだけです。
ステップ 8	**autodiscovery bgp** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# autodiscovery bgp`	すべての BGP オートディスカバリ パラメータが設定される BGP オートディスカバリ コンフィギュレーション モードを開始します。 このコマンドは、少なくとも VPN ID とシグナリング プロトコルが設定されるまで、BGP にプロビジョニングされません。
ステップ 9	**rd** {*as-number:nn*|*ip-address:nn*|**auto**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# rd auto`	VFI でルート識別子（RD）を指定します。 RD は、VFI を識別するために BGP NLRI で使用されます。VFI ごとに RD を 1 つだけ設定できます。**rd auto** を除き、RD は同じ PE の複数の VFI で設定できません。 **rd auto** が設定されている場合、RD 値は、{BGP Router ID}:{16 bits auto-generated unique index} のようになります。
ステップ 10	**route-target** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# route-target 500:99`	VFI のルート ターゲット（RT）を指定します。 PE 間の BGP オートディスカバリを設定するには、少なくとも 1 つのインポートと 1 つのエクスポート ルート ターゲット（または両方のロールを持つ 1 つのルート ターゲットだけ）を各 PE で設定する必要があります。 export または import キーワードが指定されていない場合、RT はインポートおよびエクスポートの両方であることを意味します。VFI には、複数のエクスポートまたはインポート RT を設定できます。ただし、同じ PE の複数の VFI で、同じ RT を使用することはできません。
ステップ 11	**route-target import** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# route-target import 200:20`	VFI のインポート ルート ターゲットを指定します。 インポート ルート ターゲットは、PE が受信した NLRI の RT と比較する項目です。RT が同じ VPLS サービスに属することを判断するには、受信した NLRI の RT がインポート RT と一致する必要があります。
ステップ 12	**route-target export** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# route-target export 100:10`	VFI のエクスポート ルート ターゲットを指定します。 エクスポート ルートターゲットは、他の PE にアドバタイズされる NLRI 内に含められる RT です。
ステップ 13	**signaling-protocol ldp** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# signaling-protocol ldp`	LDP シグナリ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	**vpls-id** {*as-number:nn*|*ip-address:nn*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# vpls-id 10:20`	シグナリング中に VPLS ドメインを識別する VPLS ID を指定します。 デフォルトの VPLS ID は BGP の ASN および設定済みの VPN ID を使用して自動的に生成されるため、同じ自律システム内にある（同じ ASN を共有する）すべての PE ではこのコマンドはオプションです（つまり、デフォルトの VPLS ID は ASN:VPN-ID です）。4 バイトの ASN を使用する場合は、VPLS ID を作成するために、ASN の下位 2 バイトが使用されます。InterAS の場合、VPLS ID を明示的に設定する必要があります。VFI ごとに 1 つの VPLS ID だけを設定でき、同じ VPLS ID を複数の VFI には使用できません。
ステップ 15	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

G.8032 イーサネット リング保護の設定

G.8032 動作を設定するには、次のものを別個に設定します。

- 次のものを示す ERP インスタンス：
  - APS チャネルとして使用する（サブ）インターフェイス
  - CFM によって監視する（サブ）インターフェイス）
  - インターフェイスが RPL リンクであるかどうか、RPL リンクである場合は RPL ノード タイプ
- リング リンクを監視する EFD による CFM

（注） 各モニタ リンクの MEP は、別のメンテナンス アソシエーションで設定する必要があります。

- レイヤ 2 トポロジを作成するブリッジ ドメイン。RAPS チャネルは、データ ブリッジ ドメインから分離した専用の管理ブリッジ ドメインで設定されます。
- デフォルト値と異なる場合は、ERP インスタンスに適用される動作の特性。これは任意です。

この項では、次の内容について説明します。

ERP プロファイルの設定

イーサネット リング保護（ERP）プロファイルを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **ethernet ring g8032 profile** *profile-name*
3. **timer** {**wtr** | **guard** | **holdoff**} *seconds*
4. **non-revertive**
5. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**Ethernet ring g8032 profile** *profile-name* **例：** `RP/0/RSP0/CPU0:router(config)# Ethernet ring g8032 profile p1`	G.8032 リング モードをイネーブルにし、G.8032 コンフィギュレーション サブモードを開始します。
ステップ 3	**timer** {**wtr** | **guard** | **hold-off**} *seconds* **例：** `RP/0/RSP0/CPU0:router(config-g8032-ring-profile)# timer hold-off 5`	ガード、hold-off、および wait-to-restore タイマーの間隔（秒単位）を指定します。

	コマンドまたはアクション	目的
ステップ 4	**non-revertive** **例：** `RP/0/RSP0/CPU0:router(config-g8032-ring-profile)# non-revertive`	非リバーティブ リング インスタンスを指定します。
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-g8032-ring-profile)# end` または `RP/0/RSP0/CPU0:router(config-g8032-ring-profile)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

CFM MEP の設定

イーサネット接続障害管理（CFM）の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide*』の「*Configuring Ethernet OAM on the Cisco ASR 9000 Series Router*」を参照してください。

ERP インスタンスの設定

ERP インスタンスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *aps-bridge-domain-name*
5. **interface** *type port0-interface-path-id.subinterface*
6. **interface** *type port1-interface-path-id.subinterface*
7. **bridge-domain** *data-bridge-domain-name*

8. **interface** *type interface-path-id.subinterface*
9. **ethernet ring g8032** *ring-name*
10. **instance** *number*
11. **description** *string*
12. **profile** *profile-name*
13. **rpl** {**port0** | **port1**} {**owner** | **neighbor** | **next-neighbor**}
14. **inclusion-list vlan-ids** *vlan-id*
15. **aps-channel**
16. **level** *number*
17. **port0 interface** *type interface-path-id*
18. **port1** {**interface** *type interface-path-id* | **bridge-domain** *bridge-domain-name* | **xconnect** *xconnect-name* | **none**}
19. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco` `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#`	ブリッジ ドメインを含めることができるブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 4	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain bd1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	R-APS チャネルのブリッジ ドメインを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン 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 5	**interface** *type port0-interface-path-id.subinterface* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/0/0/0.1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#`	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、同じブリッジ ドメインに属する他のインターフェイスからパケットを転送および受信できるブリッジ ドメインにインターフェイスを追加します。
ステップ 6	**interface** *type port1-interface-path-id.subinterface* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/0/0/1.1` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#`	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、同じブリッジ ドメインに属する他のインターフェイスからパケットを転送および受信できるブリッジ ドメインにインターフェイスを追加します。
ステップ 7	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain bd2` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#`	データ トラフィックのブリッジ ドメインを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン コンフィギュレーション モードを開始します。
ステップ 8	**interface** *type interface-path-id.subinterface* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/0/0/0.10` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#`	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、同じブリッジ ドメインに属する他のインターフェイスからパケットを転送および受信できるブリッジ ドメインにインターフェイスを追加します。
ステップ 9	**ethernet ring g8032** *ring-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# ethernet ring g8032 r1`	G.8032 リング モードをイネーブルにし、G.8032 コンフィギュレーション サブモードを開始します。
ステップ 10	**instance** *number* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# instance 1`	イーサネット リング G.8032 インスタンス コンフィギュレーション サブモードを開始します。
ステップ 11	**description** *string* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance)# description test`	このインスタンスの説明として機能するストリングを指定します。
ステップ 12	**profile** *profile-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance)#profile p1`	関連するイーサネット リング G.8032 プロファイルを指定します。

	コマンドまたはアクション	目的
ステップ 13	**rpl** {**port0** | **port1**} {**owner** | **neighbor** | **next-neighbor**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance)#rpl port0 neighbor`	RPL オーナー、ネイバー、または次のネイバーとしてローカル ノードのリング ポートを 1 つ指定します。
ステップ 14	**inclusion-list vlan-ids** *vlan-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance)# inclusion-list vlan-ids e-g`	現在のインスタンスと一連の VLAN ID を関連付けます。
ステップ 15	**aps-channel** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance)# aps-channel`	イーサネット リング G.8032 インスタンス aps-channel コンフィギュレーション サブモードを開始します。
ステップ 16	**level** *number* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance-aps)# level 5`	APS メッセージ レベルを指定します。範囲は 0 ～ 7 です。
ステップ 17	**port0 interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(configl2vpn-erp-instance-aps)# port0 interface GigabitEthernet 0/0/0/0.1`	G.8032 APS チャネル インターフェイスを port0 に関連付けます。

	コマンドまたはアクション	目的
ステップ 18	**port1** {**interface** *type interface-path-id* | **bridge-domain** *bridge-domain-name* | **xconnect** *xconnect-name* | **none**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance-aps)# port1 interface GigabitEthernet 0/0/0/1.1`	G.8032 APS チャネル インターフェイスを port1 に関連付けます。
ステップ 19	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance-aps)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-erp-instance-aps)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

ERP パラメータの設定

ERP パラメータを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **ethernet ring g8032** *ring-name*
4. **port0 interface** *type interface-path-id*
5. **monitor port0 interface** *type interface-path-id*
6. **exit**
7. **port1** {**interface** *type interface-path-id* | **virtual** | **none**}
8. **monitor port1 interface** *type interface-path-id*
9. **exit**
10. **exclusion-list vlan-ids** *vlan-id*

11. **open-ring**
12. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**ethernet ring g8032** *ring-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# ethernet ring g8032 r1`	G.8032 リング モードをイネーブルにし、G.8032 コンフィギュレーション サブモードを開始します。
ステップ 4	**port0 interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# port0 interface GigabitEthernet 0/1/0/6`	指定したポート（リング ポート）の G.8032 ERP をイネーブルにします。
ステップ 5	**monitor port0 interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-port0)# monitor port0 interface 0/1/0/2`	リング ポートごとにリング リンク障害を検出するために監視するポートを指定します。モニタ対象インターフェイスは、メイン インターフェイスのサブインターフェイスでなければなりません。
ステップ 6	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-port0)# exit`	port0 コンフィギュレーション サブモードを終了します。
ステップ 7	**port1** {**interface** *type interface-path-id* | **virtual** | **none**} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# port1 interface GigabitEthernet 0/1/0/8`	指定したポート（リング ポート）の G.8032 ERP をイネーブルにします。
ステップ 8	**monitor port1 interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-port1)# monitor port1 interface 0/1/0/3`	リング ポートごとにリング リンク障害を検出するために監視するポートを指定します。モニタ対象インターフェイスは、メイン インターフェイスのサブインターフェイスでなければなりません。

	コマンドまたはアクション	目的
ステップ 9	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp-port1)# exit`	port1 コンフィギュレーション サブモードを終了します。
ステップ 10	**exclusion-list vlan-ids** *vlan-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# exclusion-list vlan-ids a-d`	イーサネット リング保護メカニズムによって保護されていない一連の VLAN ID を指定します。
ステップ 11	**open-ring** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# open-ring`	開いたリングとしてイーサネット リング G.8032 を指定します。
ステップ 12	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-erp)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

TCN 伝播の設定

トポロジ変更通知（TCN）の伝播を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **tcn-propagation**

4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**tcn-propagation** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# tcn-propagation`	マイナー リングからメイン リング、および MSTP から G.8032 への TCN 伝播を許可します。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

Flow Aware Transport 疑似回線の設定

この項では、次の内容について説明します。

- VPWS の ECMP および FAT PW によるロード バランシングのイネーブル化
- VPLS の ECMP および FAT PW によるロード バランシングのイネーブル化

VPWS の ECMP および FAT PW によるロード バランシングのイネーブル化

VPWS の ECMP および FAT PW によるロード バランシングをイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **load-balancing flow** {**src-dst-mac** | **src-dst-ip**}
4. **pw-class** {*name*}
5. **encapsulation mpls**
6. **load-balancing flow-label** {**both** | **receive** | **transmit**} [**static**]
7. **exit**
8. **xconnect group** *group-name*
9. **p2p** *xconnect-name*
10. **interface type** *interface-path-id*
11. **neighbor** *A.B.C.D* **pw-id** *pseudowire-id*
12. **pw-class** {*name*}
13. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**load-balancing flow {src-dst-mac | src-dst-ip}** **例：** `RP/0/RSP0/CPU0:router(config)# load-balancing flow src-dst-ip`	フローに基づくロード バランシングをイネーブルにします。 • **src-dst-mac**：ハッシュ用の送信元/宛先 MAC アドレスを使用します。 • **src-dst-ip**：ハッシュ用の送信元/宛先 IP アドレスを使用します。 **（注）** **load-balancing flow** コマンドは、**src-dst-ip** キーワードとともに使用することを推奨します。

	コマンドまたはアクション	目的
ステップ 4	**pw-class** {*name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class path1`	疑似回線に使用する疑似回線クラス テンプレート名を設定します。
ステップ 5	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。
ステップ 6	**load-balancing flow-label** {**both** | **receive** | **transmit**} [**static**] **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# load-balancing flow-label both`	ECMP のロード バランシングをイネーブルにします。また、疑似回線のフロー ラベルのインポジションおよびディスポジションをイネーブルにします。 **（注）** **static** キーワードを指定しない場合は、FAT PW のエンドツーエンド ネゴシエーションがイネーブルになります。
ステップ 7	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)#exit`	疑似回線カプセル化サブモードを終了し、ルータを親コンフィギュレーション モードに戻します。
ステップ 8	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp1`	相互接続グループの名前を指定します。
ステップ 9	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p vlan1`	ポイントツーポイント相互接続の名前を指定します。
ステップ 10	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface GigabitEthernet0/0/0/0.1`	インターフェイス タイプとインスタンスを指定します。
ステップ 11	**neighbor** *A.B.C.D* **pw-id** *pseudowire-id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.2.2.2 pw-id 2000`	クロスコネクトの疑似回線セグメントを設定します。 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。

	コマンドまたはアクション	目的
ステップ 12	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class path1`	この疑似回線を疑似回線クラスと関連付けます。
ステップ 13	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

VPLS の ECMP および FAT PW によるロード バランシングのイネーブル化

VPLS の ECMP および FAT PW によるロード バランシングをイネーブルにするには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **load-balancing flow** {**src-dst-mac** | **src-dst-ip**}
4. **pw-class** {*class-name*}
5. **encapsulation mpls**
6. **load-balancing flow-label** {**both** | **receive** | **transmit**} [**static**]
7. **exit**
8. **bridge group** *bridge-group-name*
9. **bridge-domain** *bridge-domain-name*
10. **vfi** {*vfi-name*}
11. **autodiscovery bgp**

12. **signaling-protocol bgp**
13. **load-balancing flow-label** {**both** | **receive** | **transmit**} [**static**]
14. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**load-balancing flow {src-dst-mac | src-dst-ip}** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# load-balancing flow src-dst-ip`	フローに基づくロード バランシングをイネーブルにします。 • **src-dst-mac**：ハッシュ用の送信元/宛先 MAC アドレスを使用します。 • **src-dst-ip**：ハッシュ用の送信元/宛先 IP アドレスを使用します。 **（注）** **load-balancing flow** コマンドは、**src-dst-ip** キーワードとともに使用することを推奨します。
ステップ 4	**pw-class** {*class-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class class1`	この疑似回線を疑似回線クラスと関連付けます。
ステップ 5	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。
ステップ 6	**load-balancing flow-label** {**both** | **receive** | **transmit**} [**static**] **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# load-balancing flow-label both`	ECMP のロード バランシングをイネーブルにします。また、疑似回線のフロー ラベルのインポジションおよびディスポジションをイネーブルにします。 **（注）** **static** キーワードを指定しない場合は、FAT PW のエンドツーエンド ネゴシエーションがイネーブルになります。
ステップ 7	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# exit`	疑似回線カプセル化サブモードを終了し、ルータを親コンフィギュレーション モードに戻します。

	コマンドまたはアクション	目的
ステップ 8	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group group1`	ブリッジ ドメインを包含できるようにブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 9	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain domain1`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン コンフィギュレーション モードを開始します。
ステップ 10	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#vfi my_vfi`	仮想転送インスタンス（VFI）コンフィギュレーション モードを開始します。
ステップ 11	**autodiscovery bgp** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# autodiscovery bgp`	すべての BGP オートディスカバリ パラメータが設定される BGP オートディスカバリ コンフィギュレーション モードを開始します。
ステップ 12	**signaling-protocol bgp** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad)# signaling-protocol bgp`	BGP シグナリングをイネーブルにして、BGP シグナリング パラメータが設定される BGP シグナリング コンフィギュレーション サブ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 13	**load-balancing flow-label** {**both**|**receive**|**transmit**} [**static**] **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# load-balancing flow-label both static`	ECMP のロード バランシングをイネーブルにします。また、疑似回線のフロー ラベルのインポジションおよびディスポジションをイネーブルにします。
ステップ 14	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-ad-sig)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線ヘッドエンドの設定

PWHE は、**pw-ether** インターフェイスを設定することによって作成されます。PWHE を機能させるには、相互接続を完全に設定する必要があります。PWHE を機能させるための、その他のレイヤ 3（L3）パラメータ（VRF および IP アドレスなど）の設定は、任意で行います。ただし、レイヤ 3 サービスを動作可能にするには L3 機能が必要です（PW L3 の終端用）。

ここでは、次の内容について説明します。

PWHE 設定の制限事項

PWHE 設定に関する制限事項は、次のとおりです。

1. ピアごとに 8 つのインターフェイス リストだけがサポートされます。
2. インターフェイス リストごとに 8 つのレイヤ 3 リンクがサポートされます。
3. VLAN ID（**tag-impose**）は、**pw-ether** インターフェイスのみで相互接続に設定できます。
4. VLAN ID（**tag-impose**）は、**pw-ether** インターフェイスのみで VC タイプ 4 の下に設定できます。
5. 疑似回線の冗長性、優先パス、ローカル スイッチングまたは L2TP は、PWHE で設定された相互接続に対してはサポートされません
6. TE および LDP などのアプリケーションはインターフェイス タイプのチェックを行うため、PWHE を設定することはできません。
7. PWHE インターフェイス上では、アドレス ファミリ、CDP、および MPLS は設定できません。
8. eBGP およびスタティック ルートのみがサポートされています。
9. IPv6 設定は許可されていません。

PWHE インターフェイスの設定

PWHE インターフェイスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface pw-ether** *id*
3. **attach generic-interface-list** *interface_list_name*
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure` `RP/0/RSP0/CPU0:router(config)#`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface pw-ether** *id* **例：** `RP/0/RSP0/CPU0:router(config)# interface pw-ether <id>`	指定されたインターフェイスを設定し、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。

	コマンドまたはアクション	目的
ステップ 3	**attach generic-interface-list** *interface_list_name* **例：** `RP/0/RSP0/CPU0:router(config-if)# attach generic-interface-list interfacelist1`	指定されたインターフェイス リストにインターフェイスを接続します。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PWHE 相互接続の設定

PWHE 相互接続を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **xconnect group** *group-name*
4. **p2p** *xconnect-name*
5. **interface pw-ether** *id*
6. **neighbor** *A.B.C.D* **pw-id** *value*
7. **pw-class** *class-name*
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure` `RP/0/RSP0/CPU0:router(config)#`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	レイヤ 2 VPN 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	**xconnect group** *group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group MS-PW1`	自由形式の 32 文字ストリングを使用して、相互接続グループ名を設定します。
ステップ 4	**p2p** *xconnect-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p ms-pw1`	P2P コンフィギュレーション サブモードを開始します。
ステップ 5	**interface pw-ether** *id* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface pw-ether 100`	PWHE インターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 6	**neighbor** *A.B.C.D* **pw-id** *value* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.165.200.25 pw-id 100`	相互接続の疑似回線を設定します。 IP アドレスは、該当する PE ノードの IP アドレスです。 **pw-id** は PE ノードの **pw-id** と一致する必要があります。
ステップ 7	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class dynamic_mpls`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。 **（注）** 疑似回線クラスは、VC4 および VC5 の L2VPN の下で次のように定義する必要があります。 `pw-class vc_type_4` `encapsulation mpls` `transport-mode vlan` `!` `!` `pw-class vc_type_5` `encapsulation mpls` `transport-mode ethernet` `!` `!`
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

汎用インターフェイス リストの設定

汎用インターフェイス リストを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **generic-interface-list** *list-name*
3. **interface** *type interface-path-id*
4. **interface** *type interface-path-id*
5. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure` `RP/0/RSP0/CPU0:router(config)#`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**generic-interface-list** *list-name* **例：** `RP/0/RSP0/CPU0:router(config)#` `generic-interface-list list1`	汎用インターフェイス リストを設定します。
ステップ 3	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-if-list)#` `interface Bundle-Ether 100`	指定されたインターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 4	**interface** *type interface-path-id* **例：** `RP/0/RSP0/CPU0:router(config-if-list)# interface Bundle-Ether 200`	指定されたインターフェイスを設定します。
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if-list)# end` または `RP/0/RSP0/CPU0:router(config-if-list)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

送信元アドレスの設定

ローカル送信元アドレスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-class** *class-name*
4. **encapsulation mpls**
5. **ipv4 source** *source-address*
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure` `RP/0/RSP0/CPU0:router(config)#`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	レイヤ 2 VPN 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	**pw-class** *class-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class class1`	疑似回線クラス サブモードになり、疑似回線クラス テンプレートを定義できます。
ステップ 4	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。

	コマンドまたはアクション	目的
ステップ 5	**ipv4 source** *source-address* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# ipv4 source 10.1.1.1`	ローカル送信元 IPv4 アドレスを設定します。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PWHE インターフェイスのパラメータの設定

PWHE インターフェイスのパラメータを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface pw-ether** *id*
3. **attach generic-interface-list** *interface_list_name*
4. **l2overhead** *bytes*
5. **load-interval** *seconds*
6. **dampening** *decay-life*
7. **logging events link-status**
8. **mac-address** *MAC address*
9. **mtu** *interface_MTU*
10. **bandwidth** *kbps*
11. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** RP/0/RSP0/CPU0:router# **configure** RP/0/RSP0/CPU0:router(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface pw-ether** *id* **例：** RP/0/RSP0/CPU0:router(config)# interface pw-ether <id>	指定されたインターフェイスを設定し、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。
ステップ 3	**attach generic-interface-list** *interface_list_name* **例：** RP/0/RSP0/CPU0:router(config-if)# attach generic-interface-list interfacelist1	指定されたインターフェイス リストにインターフェイスを接続します。
ステップ 4	**l2overhead** *bytes* **例：** RP/0/RSP0/CPU0:router(config-if)#l2overhead 20	レイヤ 2 オーバーヘッドのサイズを設定します。
ステップ 5	**load-interval** *seconds* **例：** RP/0/RSP0/CPU0:router(config-if)#load-interval 90	インターフェイスの負荷計算の間隔（秒単位）を指定します。 秒数： • 0 に設定できます（0 は負荷計算をディセーブルにします）。 • 0 以外の場合は、30 ～ 600 の範囲の 30 の倍数で指定する必要があります。
ステップ 6	**dampening** *decay-life* **例：** RP/0/RSP0/CPU0:router(config-if)#dampening 10	特定のインターフェイスでのステート ダンプニングを設定します（分単位）。
ステップ 7	**logging events link-status** **例：** RP/0/RSP0/CPU0:router(config-if)#logging events link-status	インターフェイス ロギングごとに設定します。
ステップ 8	**mac-address** *MAC address* **例：** RP/0/RSP0/CPU0:router(config-if)#mac-address aaaa.bbbb.cccc	インターフェイスの MAC アドレス（xxxx.xxxx.xxxx）を設定します。

	コマンドまたはアクション	目的
ステップ 9	**mtu** *interface_MTU* **例：** `RP/0/RSP0/CPU0:router(config-if)#mtu 128`	インターフェイスの MTU を設定します。
ステップ 10	**bandwidth** *kbps* **例：** `RP/0/RSP0/CPU0:router(config-if)#bandwidth 200`	帯域幅を設定します。範囲は 0 ～ 4294967295 kbps です。
ステップ 11	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

L2VPN over GRE の設定

L2VPN over GRE を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface type** *interface-path-id*
3. **l2transport**
4. **exit**
5. **interface loopback** *instance*
6. **ipv4 address** *ip-address*
7. **exit**
8. **interface loopback** *instance*
9. **ipv4 address** *ip-address*

10. **router ospf** *process-name*
11. **area** *area-id*
12. **interface type** *interface-path-id*
13. **interface tunnel-ip** *number*
14. **exit**
15. **interface tunnel-ip** *number*
16. **ipv4 address** *ipv4-address mask*
17. **tunnel source type** *path-id*
18. **tunnel destination** *ip-address*
19. **end**
20. **l2vpn**
21. **bridge group** *bridge-group-name*
22. **bridge-domain** *bridge-domain-name*
23. **interface type** *interface-path-id*
24. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
25. **mpls ldp**
26. **router-id** {*router-id*}
27. **interface tunnel-ip** *number*
28. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router# interface TenGigE0/1/0/12`	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、インターフェイスを設定します。
ステップ 3	**l2transport** **例：** `RP/0/RSP0/CPU0:router# l2transport`	選択したインターフェイスでレイヤ 2 トランスポート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 4	**exit** **例：** `RP/0/RSP0/CPU0:router# exit`	現在のコンフィギュレーション モードを終了します。
ステップ 5	**interface loopback instance** **例：** `RP/0/RSP0/CPU0:router# interface Loopback0`	インターフェイス コンフィギュレーション モードを開始して、新しいループバック インターフェイスの名前を指定します。
ステップ 6	**ipv4 address ip-address** **例：** `RP/0/RSP0/CPU0:router# ipv4 address 100.100.100.100 255.255.255.255`	仮想ループバック インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 7	**exit** **例：** `RP/0/RSP0/CPU0:router# exit`	現在のコンフィギュレーション モードを終了します。
ステップ 8	**interface loopback** *instance* **例：** `RP/0/RSP0/CPU0:router# interface Loopback1`	インターフェイス コンフィギュレーション モードを開始して、新しいループバック インターフェイスの名前を指定します。
ステップ 9	**ipv4 address** *ip-address* **例：** `RP/0/RSP0/CPU0:router# ipv4 address 10.0.1.1 255.255.255.255`	仮想ループバック インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 10	**router ospf** *process-name* **例：** `RP/0/RSP0/CPU0:router# router ospf 1`	指定したルーティング プロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。
ステップ 11	**area** *area-id* **例：** `RP/0/RSP0/CPU0:router# area 0`	エリア コンフィギュレーション モードを開始し、OSPF プロセスのエリアを設定します。
ステップ 12	**interface loopback** *instance* **例：** `RP/0/RSP0/CPU0:router# interface Loopback0`	インターフェイス コンフィギュレーション モードを開始して、新しいループバック インターフェイスの名前を指定します。
ステップ 13	**interface tunnel-ip** *number* **例：** `RP/0/RSP0/CPU0:router# interface tunnel-ip1`	トンネル 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 14	**exit** **例：** `RP/0/RSP0/CPU0:router# exit`	現在のコンフィギュレーション モードを終了します。
ステップ 15	**interface tunnel-ip** *number* **例：** `RP/0/RSP0/CPU0:router(config)# interface tunnel-ip1`	トンネル 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。 • 番号はトンネル インターフェイスに関連付けられた番号です。
ステップ 16	**ipv4 address** *ipv4-address subnet-mask* **例：** `RP/0/RSP0/CPU0:router(config-if)# ipv4 address 12.0.0.1 255.255.255.0`	インターフェイスの IPv4 アドレスおよびサブネット マスクを指定します。 • ipv4-address は、インターフェイスの IP アドレスを指定します。 • subnet-mask は、インターフェイスのサブネット マスクを指定します。
ステップ 17	**tunnel source** *type path-id* **例：** `RP/0/RSP0/CPU0:router(config-if)# tunnel source Loopback1`	トンネル インターフェイスの送信元を指定します。
ステップ 18	**tunnel destination** *ip-address* **例：** `RP/0/RSP0/CPU0:router(config-if)# tunnel destination 100.100.100.20`	トンネルの宛先を指定します。
ステップ 19	**end** **例：** `RP/0/RSP0/CPU0:router(config-if)# end`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。

	コマンドまたはアクション	目的
ステップ 20	**l2vpn** **例：** `RP/0/RSP0/CPU0:router# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 21	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router# bridge group access-pw`	ブリッジ ドメインを含めることができるブリッジ グループを作成し、ブリッジ ドメインにネットワーク インターフェイスを割り当てます。
ステップ 22	**bridge-domain** *bridge-domain-name* **例：** `RP/0/RSP0/CPU0:router# bridge-domain test`	ブリッジ ドメインを確立し、L2VPN ブリッジ グループ ブリッジ ドメイン コンフィギュレーション モードを開始します。
ステップ 23	**interface type** *interface-path-id* **例：** `RP/0/RSP0/CPU0:router# interface TenGigE0/1/0/12`	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、同じブリッジ ドメインに属する他のインターフェイスからパケットを転送および受信できるブリッジ ドメインにインターフェイスを追加します。
ステップ 24	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router# neighbor 125.125.125.125 pw-id 100`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、A.B.C.D 引数を使用します。 **（注）** A.B.C.D は再帰的または非再帰的プレフィクスです。 • 疑似回線 ID および ID 値を設定するには、pw-id キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 25	**mpls ldp** **例：** `RP/0/RSP0/CPU0:router# mpls ldp`	MPLS LDP 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 26	**router-id** {*router-id*} **例：** `RP/0/RSP0/CPU0:router# router-id 100.100.100.100`	OSPF プロセスのルータ ID を設定します。 **（注）** 固定 IP アドレスをルータ ID として使用することを推奨します。

	コマンドまたはアクション	目的
ステップ 27	**interface tunnel-ip** *number* **例：** `RP/0/RSP0/CPU0:router# interface tunnel-ip1`	トンネル 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。 **（注）** number 引数は、トンネル インターフェイスに関連付けられた番号を示します。
ステップ 28	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

疑似回線の優先パスとしての GRE トンネルの設定

疑似回線の優先パスとして GRE トンネルを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pw-class** {*name*}
4. **encapsulation mpls**
5. **preferred-path** {**interface**} {**tunnel-ip** *value* | **tunnel-te** *value* | **tunnel-tp** *value*} [**fallback disable**]
6. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**pw-class** {*name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class gre`	疑似回線クラス名を設定します。
ステップ 4	**encapsulation mpls** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls`	MPLS に疑似回線カプセル化を設定します。

	コマンドまたはアクション	目的
ステップ 5	**preferred-path** {**interface**} {**tunnel-ip** *value* | **tunnel-te** *value* | **tunnel-tp** *value*} [**fallback disable**] **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel-ip 1 fallback disable`	優先パス トンネルを設定します。フォールバックのディセーブル化の設定が使用されており、優先パスとして設定されている TE/TP トンネルがダウン状態になると、対応する疑似回線もダウン状態になることがあります。 **（注）** フォールバックがサポートされていることを確認します。
ステップ 6	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-pwc-encap-mpls-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

マルチポイント レイヤ 2 サービスの設定例

ここで示す設定例は、次のとおりです。

プロバイダー エッジ間のバーチャル プライベート LAN サービスの設定：例

これらの設定は、参加 VPLS プロバイダー エッジ（PE）ノードのフル メッシュでレイヤ 2 VFI を作成する例を示しています。

この設定は、PE 1 を設定する例を示しています。

```
configure
 l2vpn
  bridge group 1
   bridge-domain PE1-VPLS-A
    GigabitEthernet0/0/0/1
    vfi 1
     neighbor 10.2.2.2 pw-id 1
     neighbor 10.3.3.3 pw-id 1
     !
   !
 interface loopback 0
  ipv4 address 10.1.1.1 255.255.255.25
```

この設定は、PE 2 を設定する例を示しています。

```
configure
 l2vpn
```

```
 bridge group 1
  bridge-domain PE2-VPLS-A
   interface GigabitEthernet0/0/0/1

   vfi 1
    neighbor 10.1.1.1 pw-id 1
    neighbor 10.3.3.3 pw-id 1
    !
  !
interface loopback 0
 ipv4 address 10.2.2.2 255.255.255.25
```

この設定は、PE 3 を設定する例を示しています。

```
configure
 l2vpn
  bridge group 1
   bridge-domain PE3-VPLS-A
    interface GigabitEthernet0/0/0/1
    vfi 1
     neighbor 10.1.1.1 pw-id 1
     neighbor 10.2.2.2 pw-id 1
     !
   !
interface loopback 0
 ipv4 address 10.3.3.3 255.255.255.25
```

プロバイダー エッジとカスタマー エッジ間のバーチャル プライベート LAN サービスの設定：例

この設定は、PE-to-CE ノードの VPLS の設定方法を示しています。

```
configure
 interface GigabitEthernet0/0/0/1
  l2transport---AC interface

  no ipv4 address
  no ipv4 directed-broadcast
  negotiation auto
  no cdp enable


configure
 interface GigabitEthernet0/0
  l2transport

  no ipv4 address
  no ipv4 directed-broadcast
  negotiation auto
  no cdp enable


configure
 interface GigabitEthernet0/0
  l2transport

  no ipv4 address
  no ipv4 directed-broadcast
  negotiation auto
  no cdp enable
```

MAC アドレス回収フィールドの表示：例

この出力は、MAC アドレス回収フィールドの例を示しています。

```
RP/0/RSP0/CPU0:router# show l2vpn bridge-domain detail


Bridge group: siva_group, bridge-domain: siva_bd, id: 0, state: up, ShgId: 0, MSTi: 0
  MAC Learning: enabled
  MAC withdraw: enabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown Unicast: enabled
  MAC address aging time: 300 s Type: inactivity
  MAC address limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  Security: disabled
  DHCPv4 Snooping: disabled
  MTU: 1500
  MAC Filter:  Static MAC addresses:
  ACs: 1 (1 up), VFIs: 1, PWs: 2 (1 up)
  List of ACs:
    AC: GigabitEthernet0/4/0/1, state is up
      Type Ethernet
      MTU 1500; XC ID 0x5000001; interworking none; MSTi 0 (unprotected)
      MAC Learning: enabled
      MAC withdraw: disabled
      Flooding:
        Broadcast & Multicast: enabled
        Unknown Unicast: enabled
      MAC address aging time: 300 s Type: inactivity
      MAC address limit: 4000, Action: none, Notification: syslog
      MAC limit reached: no
      Security: disabled
      DHCPv4 Snooping: disabled
      Static MAC addresses:
      Statistics:
        packet totals: receive 6,send 0
        byte totals: receive 360,send 4
  List of Access PWs:
  List of VFIs:
    VFI siva_vfi
      PW: neighbor 10.1.1.1, PW ID 1, state is down ( local ready )
        PW class not set, XC ID 0xff000001
        Encapsulation MPLS, protocol LDP
        PW type Ethernet, control word enabled, interworking none
       PW backup disable delay 0 sec
        Sequencing not set
                MPLS         Local                          Remote
          ------------ ------------------------------ --------------------------
               Label         30005                        unknown
               Group ID      0x0                          0x0
               Interface     siva/vfi                     unknown
               MTU           1500                         unknown
               Control word   enabled                     unknown
               PW type        Ethernet                    unknown
          ------------ ------------------------------ --------------------------
        Create time: 19/11/2007 15:20:14 (00:25:25 ago)
        Last time status changed: 19/11/2007 15:44:00 (00:01:39 ago)
        MAC withdraw message: send 0 receive 0
```

スプリット ホライズン グループ：例

次の例では、レイヤ 2 トランスポートのインターフェイスを設定し、ブリッジ ドメインに追加し、スプリット ホライズン グループに割り当てます。

```
RP/0/RSP0/CPU0:router(config)#l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group examples
RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain all_three
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet 0/0/0/0.99
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet 0/0/0/0.101
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#split-horizon group
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#neighbor 192.168.99.1 pw-id 1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#neighbor 192.168.99.9 pw-id 1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#split-horizon group
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#vfi abc
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#neighbor 192.168.99.17 pw-id 1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#show
Mon Oct 18 13:51:05.831 EDT
l2vpn
 bridge group examples
  bridge-domain all_three
   interface GigabitEthernet0/0/0/0.99
   !
   interface GigabitEthernet0/0/0/0.101
    split-horizon group
   !
   neighbor 192.168.99.1 pw-id 1
   !
   neighbor 192.168.99.9 pw-id 1
    split-horizon group
   !
   vfi abc
    neighbor 192.168.99.17 pw-id 1
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#
```

この例に従って、ブリッジ ドメイン **all_three** のスプリット ホライズン グループの割り当ては、次のようになります。

ブリッジ ポート / 疑似回線	スプリット ホライズン グループ
ブリッジ ポート：gig0/0/0/0.99	0
ブリッジ ポート：gig0/0/0/0.101	2
PW：192.168.99.1 pw-id 1	0
PW: 192.168.99.9 pw-id 1	2
PW: 192.168.99.17 pw-id 1	1

不明なユニキャスト フラッディングのブロック：例

不明なユニキャスト フラッディングは、次のレベルでブロックできます。

- ブリッジ ドメイン
- ブリッジ ポート（接続回線（AC））
- アクセス疑似回線（PW）

次に、ブリッジ ドメイン レベルで不明なユニキャスト フラッディングをブロックする例を示します。

```
configure
  l2vpn
    bridge-group group1
    bridge-domain domain1
    flooding unknown-unicast disable
  end
```

次に、ブリッジ ポート レベルで不明なユニキャスト フラッディングをブロックする例を示します。

```
configure
  l2vpn
    bridge-group group1
    bridge-domain domain1
    interface GigabitEthernet 0/1/0/1
    flooding unknown-unicast disable
  end
```

次に、アクセス疑似回線レベルで不明なユニキャスト フラッディングをブロックする例を示します。

```
configure
  l2vpn
    bridge-group group1
    bridge-domain domain1
    neighbor 10.1.1.1 pw-id 1000
    flooding unknown-unicast disable
  end
```

MAC フラッシュのディセーブル化：例

次のレベルで MAC フラッシュをディセーブルにできます。

- ブリッジ ドメイン
- ブリッジ ポート（接続回線（AC））
- アクセス疑似回線（PW）

次に、ブリッジ ドメイン レベルで MAC フラッシュをディセーブルにする例を示します。

```
configure
  l2vpn
    bridge-group group1
    bridge-domain domain1
    mac
    port-down flush disable
  end
```

次に、ブリッジ ポート レベルで MAC フラッシュをディセーブルにする例を示します。

```
configure
  l2vpn
    bridge-group group1
    bridge-domain domain1
    interface GigabitEthernet 0/1/0/1
    mac
    port-down flush disable
  end
```

次に、アクセス疑似回線レベルで MAC フラッシュをディセーブルにする例を示します。

```
configure
  l2vpn
    bridge-group group1
    bridge-domain domain1
    neighbor 10.1.1.1 pw-id 1000
    mac
    port-down flush disable
   end
```

IOS XR トランク インターフェイスでのブリッジング：例

次に、Cisco ASR 9000 シリーズ ルータを単純な L2 スイッチとして設定する例を示します。

特記事項：
4 本の接続回線（AC）があるブリッジ ドメインを作成します。各 AC は、IOS XR トランク インターフェイスです（つまり、サブインターフェイス/EFP ではありません）です。

- 次の例では、実行コンフィギュレーションが空であり、すべてのコンポーネントが作成されていると想定します。
- この例では、インターフェイス間のスイッチングを実行するように Cisco ASR 9000 シリーズ ルータを設定するために必要なすべての手順を示します。ただし、no shut、negotiation auto などのインターフェイスを準備するためのコマンドは除外されています。
- ブリッジ ドメインは、作成直後に **no shut** 状態になります。
- この例ではトランク（つまりメイン）インターフェイスだけが使用されます。
- トランク インターフェイスは、タグ付き（IEEE 802.1Q）またはタグなし（つまり VLAN ヘッダーなし）フレームを処理できます。
- ブリッジ ドメインは、MAC アドレスに基づいて学習、フラッディング、および転送を行います。この機能は、タグの設定に関係なくフレームで動作します。
- ブリッジ ドメイン エンティティは、システムのすべてのラインカードにわたります。単一の LC にすべてのブリッジ ドメイン AC を配置する必要はありません。これは、ブリッジ ドメインの設定に適用されます。
- ルータが予期したとおりに設定されていること、およびコマンドによあって新しい設定ステータスが表示されることを確認するには、**show bundle** および **show l2vpn bridge-domain** コマンドを使用します。
- 次の例の AC では、管理ダウン状態になっているインターフェイスを使用します。

設定例

```
RP/0/RSP0/CPU0:router#config
RP/0/RSP0/CPU0:router(config)#interface Bundle-ether10
RP/0/RSP0/CPU0:router(config-if)#l2transport
RP/0/RSP0/CPU0:router(config-if-l2)#interface GigabitEthernet0/2/0/5
RP/0/RSP0/CPU0:router(config-if)#bundle id 10 mode active
RP/0/RSP0/CPU0:router(config-if)#interface GigabitEthernet0/2/0/6
RP/0/RSP0/CPU0:router(config-if)#bundle id 10 mode active
RP/0/RSP0/CPU0:router(config-if)#interface GigabitEthernet0/2/0/0
RP/0/RSP0/CPU0:router(config-if)#l2transport
RP/0/RSP0/CPU0:router(config-if-l2)#interface GigabitEthernet0/2/0/1
RP/0/RSP0/CPU0:router(config-if)#l2transport
RP/0/RSP0/CPU0:router(config-if-l2)#interface TenGigE0/1/0/2
RP/0/RSP0/CPU0:router(config-if)#l2transport
RP/0/RSP0/CPU0:router(config-if-l2)#l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group examples
RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain test-switch
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface Bundle-ether10
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/2/0/0
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/2/0/1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface TenGigE0/1/0/2
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#commit
RP/0/RSP0/CPU0:Jul 26 10:48:21.320 EDT: config[65751]: %MGBL-CONFIG-6-DB_COMMIT :
Configuration committed by user 'lab'. Use 'show configuration commit changes 1000000973'
to view the changes.
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#end
RP/0/RSP0/CPU0:Jul 26 10:48:21.342 EDT: config[65751]: %MGBL-SYS-5-CONFIG_I : Configured
from console by lab
RP/0/RSP0/CPU0:router#show bundle Bundle-ether10

Bundle-Ether10
  Status:                                  Down
  Local links <active/standby/configured>:   0 / 0 / 2
  Local bandwidth <effective/available>:     0 (0) kbps
  MAC address (source):                    0024.f71e.22eb (Chassis pool)
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    64
  Wait while timer:                        2000 ms
  LACP:                                    Operational
    Flap suppression timer:                Off
  mLACP:                                   Not configured
  IPv4 BFD:                                Not configured

  Port                  Device           State        Port ID         B/W, kbps
  --------------------  ---------------  -----------  --------------  ----------
  Gi0/2/0/5             Local            Configured   0x8000, 0x0001     1000000
      Link is down
  Gi0/2/0/6             Local            Configured   0x8000, 0x0002     1000000
      Link is down

RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router#show l2vpn bridge-domain group examples
Bridge group: examples, bridge-domain: test-switch, id: 2000, state: up, ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 4 (1 up), VFIs: 0, PWs: 0 (0 up), PBBs: 0 (0 up)
  List of ACs:
    BE10, state: down, Static MAC addresses: 0
    Gi0/2/0/0, state: up, Static MAC addresses: 0
    Gi0/2/0/1, state: down, Static MAC addresses: 0
```

```
    Te0/5/0/1, state: down, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
RP/0/RSP0/CPU0:router#
```

次の表に、設定手順（アクション）およびこの例の対応する目的を示します。

	コマンドまたはアクション	目的
ステップ 1	**configure**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface Bundle-ether10**	新しいバンドル トランク インターフェイスを作成します。
ステップ 3	**l2transport**	Bundle-ether10 を L3 インターフェイスから L2 インターフェイスに変更します。
ステップ 4	**interface GigabitEthernet0/2/0/5**	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。GigabitEthernet0/2/0/5 で機能するようコンフィギュレーション モードを変更します。
ステップ 5	**bundle id 10 mode active**	GigabitEthernet0/2/0/5 を Bundle-ether10 のメンバーとして設定します。**mode active** キーワードは、LACP プロトコルを指定します。
ステップ 6	**interface GigabitEthernet0/2/0/6**	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。GigabitEthernet0/2/0/6 で機能するようコンフィギュレーション モードを変更します。
ステップ 7	**bundle id 10 mode active**	GigabitEthernet0/2/0/6 を Bundle-ether10 のメンバーとして設定します。**mode active** キーワードは、LACP プロトコルを指定します。
ステップ 8	**interface GigabitEthernet0/2/0/0**	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。GigabitEthernet0/2/0/0 で機能するようコンフィギュレーション モードを変更します。
ステップ 9	**l2transport**	GigabitEthernet0/2/0/0 を L3 インターフェイスから L2 インターフェイスに変更します。
ステップ 10	**interface GigabitEthernet0/2/0/1**	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。GigabitEthernet0/2/0/1 で機能するようコンフィギュレーション モードを変更します。
ステップ 11	**l2transport**	GigabitEthernet0/2/0/1 を L3 インターフェイスから L2 インターフェイスに変更します。
ステップ 12	**interface TenGigE0/1/0/2**	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。TenGigE0/1/0/2 で機能するようコンフィギュレーション モードを変更します。
ステップ 13	**l2transport**	TenGigE0/1/0/2 を L3 インターフェイスから L2 インターフェイスに変更します。
ステップ 14	**l2vpn**	L2VPN コンフィギュレーション モードを開始します。
ステップ 15	**bridge group examples**	ブリッジ グループ **examples** を作成します。
ステップ 16	**bridge-domain test-switch**	ブリッジ ドメイン **test-switch** を作成します。これは、ブリッジ グループ **examples** のメンバーです。
ステップ 17	**interface Bundle-ether10**	Bundle-ether10 をブリッジ ドメイン test-switch の AC として設定します。

	コマンドまたはアクション	目的
ステップ 18	**exit**	ブリッジ ドメイン AC コンフィギュレーション サブモードを終了し、次の AC を設定できるようにします。
ステップ 19	**interface GigabitEthernet0/2/0/0**	GigabitEthernet0/2/0/0 をブリッジ ドメイン **test-switch** の AC として設定します。
ステップ 20	**exit**	ブリッジ ドメイン AC コンフィギュレーション サブモードを終了し、次の AC を設定できるようにします。
ステップ 21	**interface GigabitEthernet0/2/0/1**	GigabitEthernet0/2/0/1 をブリッジ ドメイン **test-switch** の AC として設定します。
ステップ 22	**exit**	ブリッジ ドメイン AC コンフィギュレーション サブモードを終了し、次の AC を設定できるようにします。
ステップ 23	**interface TenGigE0/1/0/2**	インターフェイス TenGigE0/1/0/2 をブリッジ ドメイン test-switch の AC として設定します。
ステップ 24	**end** または **commit**	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

イーサネット フロー ポイントでのブリッジング：例

次に、イーサネット フロー ポイント（EFP）を通過するトラフィックでレイヤ 2 スイッチングを実行するように Cisco ASR 9000 シリーズ ルータを設定する例を示します。EFP トラフィックには通常、1 つ以上の VLAN ヘッダーがあります。IOS XR トランクと IOS-XR EFP の両方をブリッジ ドメインで接続回線として結合できますが、この例では EFP だけを使用します。

特記事項：

- EFP は、レイヤ 2 サブインターフェイスです。これは常に、トランク インターフェイスの下で作成されます。トランク インターフェイスは、EFP を作成する前に存在している必要があります。
- 空の設定では、バンドル インターフェイス トランクは存在しませんが、ラインカードを挿入すると物理トランク インターフェイスは自動的に設定されます。したがって、バンドル トランクだけが作成されます。
- この例では、サブインターフェイス番号および VLAN ID は同じですが、これは便利ではなく、必要性はありません。同じ値である必要はありません。
- ブリッジ ドメイン test-efp には、3 本の接続回線（AC）があります。AC はすべて EFP です。
- VLAN ID が 999 のフレームだけが EFP に入ります。これによって、このブリッジ ドメインのすべてのトラフィックで同じ VLAN カプセル化を確保できます。
- 次の例の AC は、管理ダウン状態になっているインターフェイス、またはラインカードが挿入されていないインターフェイス（**未解決**状態）を使用します。AC として存在しないインターフェイスを使用するブリッジ ドメイン、およびこのような設定のコミットは失敗します。この場合、ブリッジ ドメインのステータスは、欠落しているインターフェイスを設定するまで **unresolved** と表示されます。

設定例

```
RP/0/RSP1/CPU0:router#configure
RP/0/RSP1/CPU0:router(config)#interface Bundle-ether10
RP/0/RSP1/CPU0:router(config-if)#interface Bundle-ether10.999 l2transport
RP/0/RSP1/CPU0:router(config-subif)#encapsulation dot1q 999
RP/0/RSP1/CPU0:router(config-subif)#interface GigabitEthernet0/6/0/5
RP/0/RSP1/CPU0:router(config-if)#bundle id 10 mode active
RP/0/RSP1/CPU0:router(config-if)#interface GigabitEthernet0/6/0/6
RP/0/RSP1/CPU0:router(config-if)#bundle id 10 mode active
RP/0/RSP1/CPU0:router(config-if)#interface GigabitEthernet0/6/0/7.999 l2transport
RP/0/RSP1/CPU0:router(config-subif)#encapsulation dot1q 999
RP/0/RSP1/CPU0:router(config-subif)#interface TenGigE0/1/0/2.999 l2transport
RP/0/RSP1/CPU0:router(config-subif)#encapsulation dot1q 999
RP/0/RSP1/CPU0:router(config-subif)#l2vpn
RP/0/RSP1/CPU0:router(config-l2vpn)#bridge group examples
RP/0/RSP1/CPU0:router(config-l2vpn-bg)#bridge-domain test-efp
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd)#interface Bundle-ether10.999
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/6/0/7.999
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd-ac)#exit
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd)#interface TenGigE0/1/0/2.999
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd-ac)#commit
RP/0/RSP1/CPU0:router(config-l2vpn-bg-bd-ac)#end
RP/0/RSP1/CPU0:router#
RP/0/RSP1/CPU0:router#show l2vpn bridge group examples
Fri Jul 23 21:56:34.473 UTC Bridge group: examples, bridge-domain: test-efp, id: 0, state:
up, ShgId: 0, MSTi: 0
Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 3 (0 up), VFIs: 0, PWs: 0 (0 up), PBBs: 0 (0 up)
  List of ACs:
```

```
    BE10.999, state: down, Static MAC addresses: 0
    Gi0/6/0/7.999, state: unresolved, Static MAC addresses: 0
    Te0/1/0/2.999, state: down, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
RP/0/RSP1/CPU0:router#
```

次の表に、設定手順（アクション）およびこの例の対応する目的を示します。

	コマンドまたはアクション	目的
ステップ 1	**configure**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface Bundle-ether10**	新しいバンドル トランク インターフェイスを作成します。
ステップ 3	**interface Bundle-ether10.999 l2transport**	新しいバンドル トランクに EFP を作成します。
ステップ 4	**encapsulation dot1q 999**	この EFP に VLAN ID 999 を割り当てます。
ステップ 5	**interface GigabitEthernet0/6/0/5**	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。GigabitEthernet0/6/0/5 で機能するようコンフィギュレーション モードを変更します。
ステップ 6	**bundle id 10 mode active**	GigabitEthernet0/6/0/5 を Bundle-ether10 のメンバーとして設定します。**mode active** キーワードは、LACP プロトコルを指定します。
ステップ 7	**interface GigabitEthernet0/6/0/6**	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。GigabitEthernet0/6/0/6 で機能するようコンフィギュレーション モードを変更します。
ステップ 8	**bundle id 10 mode active**	GigabitEthernet0/6/0/6 を Bundle-ether10 のメンバーとして設定します。**mode active** キーワードは、LACP プロトコルを指定します。
ステップ 9	**interface GigabitEthernet0/6/0/7.999 l2transport**	GigabitEthernet0/6/0/7 に EFP を作成します。
ステップ 10	**encapsulation dot1q 999**	この EFP に VLAN ID 999 を割り当てます。
ステップ 11	**interface TenGigE0/1/0/2.999 l2transport**	TenGigE0/1/0/2 に EFP を作成します。
ステップ 12	**encapsulation dot1q 999**	この EFP に VLAN ID 999 を割り当てます。
ステップ 13	**l2vpn**	L2VPN コンフィギュレーション モードを開始します。
ステップ 14	**bridge group examples**	**examples** という名前のブリッジ グループを作成します。
ステップ 15	**bridge-domain test-efp**	**test-switch** という名前のブリッジ ドメインを作成します。これは、ブリッジ グループ **examples** のメンバーです。
ステップ 16	**interface Bundle-ether10.999**	Bundle-ether10.999 を **test-efp** という名前のブリッジ ドメインの AC として設定します。
ステップ 17	**exit**	ブリッジ ドメイン AC コンフィギュレーション サブモードを終了し、次の AC を設定できるようにします。
ステップ 18	**interface GigabitEthernet0/6/0/7.999**	GigabitEthernet0/6/0/7.999 を **test-efp** という名前のブリッジ ドメインの AC として設定します。

	コマンドまたはアクション	目的
ステップ 19	**exit**	ブリッジ ドメイン AC コンフィギュレーション サブモードを終了し、次の AC を設定できるようにします。
ステップ 20	**interface TenGigE0/1/0/2.999**	インターフェイス TenGigE0/1/0/2.999 を **test-efp** という名前のブリッジ ドメインの AC として設定します。
ステップ 21	**end** または **commit**	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

フラッディング最適化モードの変更：例

次に、ブリッジ ドメインでデフォルトのフラッディング最適化モードを変更する例を示します。

```
config
l2vpn
 bridge group MyGroup
 bridge-domain MyDomain
 flood mode convergence-optimized
```

BGP オートディスカバリおよびシグナリングでの VPLS の設定：例

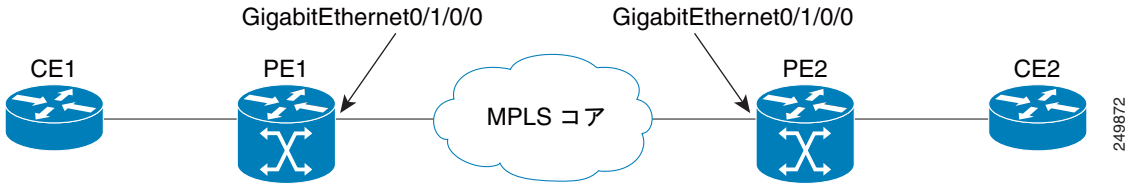
ここでは、BGP オートディスカバリとシグナリング機能を設定するための例を示します。

- LDP および BGP の設定
- BGP シグナリングによる BGP オートディスカバリの最小の L2VPN 設定
- BGP オートディスカバリおよび BGP シグナリングでの VPLS
- LDP シグナリングによる BGP オートディスカバリの最小設定
- BGP オートディスカバリおよび LDP シグナリングでの VPLS

LDP および BGP の設定

図 23 で、LDP および BGP の設定例について説明します。

図 23　　LDP および BGP の設定



PE1 での設定：

```
interface Loopback0
    ipv4 address 1.1.1.100 255.255.255.255
!
interface Loopback1
    ipv4 address 1.1.1.10 255.255.255.255
!
mpls ldp
    router-id 1.1.1.1
    interface GigabitEthernt0/1/0/0
!
router bgp 120
    address-family l2vpn vpls-vpws
!
    neighbor 2.2.2.20
    remote-as 120
    update-source Loopback1
    address-family l2vpn vpls-vpws
    signaling bgp disable
```

PE2 での設定：

```
interface Loopback0
    ipv4 address 2.2.2.200 255.255.255.255
!
interface Loopback1
    ipv4 address 2.2.2.20 255.255.255.255
!
mpls ldp
    router-id 2.2.2.2
    interface GigabitEthernt0/1/0/0
!
router bgp 120
    address-family l2vpn vpls-vpws
```

```
!
    neighbor 1.1.1.10
    remote-as 120
    update-source Loopback1
    address-family l2vpn vpls-vpws
```

BGP シグナリングによる BGP オートディスカバリの最小の L2VPN 設定

次に、デフォルト値を持つパラメータが設定されていない BGP シグナリングを使用する BGP オートディスカバリに必要な最小の L2VPN 設定例を示します。

```
(config)# l2vpn
(config-l2vpn)# bridge group {bridge group name}
(config-l2vpn-bg)# bridge-domain {bridge domain name}
(config-l2vpn-bg-bd)# vfi {vfi name}
(config-l2vpn-bg-bd-vfi)# autodiscovery bgp
(config-l2vpn-bg-bd-vfi-ad)# vpn-id 10
(config-l2vpn-bg-bd-vfi-ad)# rd auto
(config-l2vpn-bg-bd-vfi-ad)# route-target 1.1.1.1:100
(config-l2vpn-bg-bd-vfi-ad-sig)# signaling-protocol bgp
(config-l2vpn-bg-bd-vfi-ad-sig)# ve-id 1
(config-l2vpn-bg-bd-vfi-ad-sig)# commit
```

BGP オートディスカバリおよび BGP シグナリングでの VPLS

図 24 に、BGP オートディスカバリ（AD）および BGP シグナリングで VPLS を設定する例を示します。

図 24　BGP オートディスカバリおよび BGP シグナリングでの VPLS

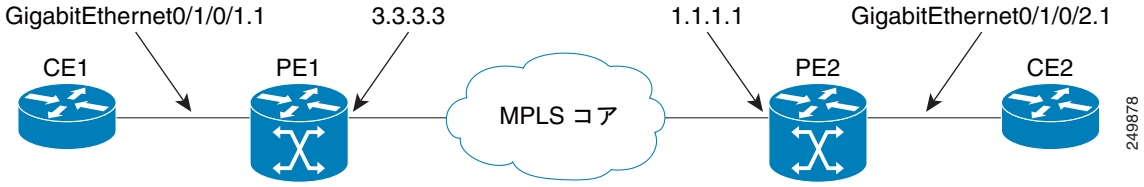
PE1 での設定：

```
l2vpn
  bridge group gr1
   bridge-domain bd1
    interface GigabitEthernet0/1/0/1.1
    vfi vf1
    ! AD independent VFI attributes
    vpn-id 100
    ! Auto-discovery attributes
    autodiscovery bgp
    rd auto
    route-target 2.2.2.2:100
    ! Signaling attributes
    signaling-protocol bgp
    ve-id 3
```

PE2 での設定：

```
l2vpn
  bridge group gr1
   bridge-domain bd1
    interface GigabitEthernet0/1/0/2.1
    vfi vf1
    ! AD independent VFI attributes
    vpn-id 100
    ! Auto-discovery attributes
    autodiscovery bgp
    rd auto
    route-target 2.2.2.2:100
    ! Signaling attributes
    signaling-protocol bgp
    ve-id 5
```

次に、BGP AD およびシグナリングを使用する VPLS の NLRI の例を示します。



ディスカバリ属性

PE1 で送信される NLRI：

```
Length = 19
Router Distinguisher = 3.3.3.3:32770
VE ID = 3
VE Block Offset = 1
VE Block Size = 10
Label Base = 16015
```

PE2 で送信される NLRI：

```
Length = 19
Router Distinguisher = 1.1.1.1:32775
VE ID = 5
VE Block Offset = 1
VE Block Size = 10
Label Base = 16120
```

LDP シグナリングによる BGP オートディスカバリの最小設定

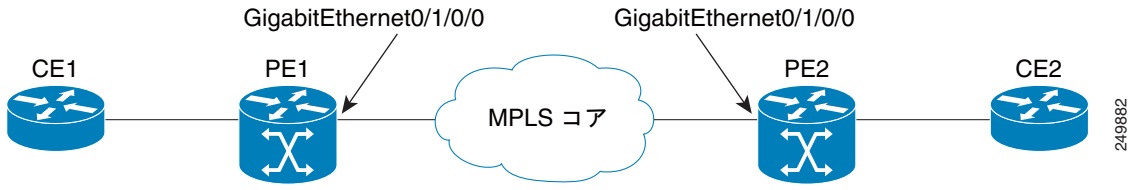
次に、デフォルト値を持つパラメータが設定されていない LDP シグナリングを使用する BGP オートディスカバリに必要な最小の L2VPN 設定例を示します。

```
(config)# l2vpn
(config-l2vpn)# bridge group {bridge group name}
(config-l2vpn-bg)# bridge-domain {bridge domain name}
(config-l2vpn-bg-bd)# vfi {vfi name}
(config-l2vpn-bg-bd-vfi)# autodiscovery bgp
(config-l2vpn-bg-bd-vfi-ad)# vpn-id 10
(config-l2vpn-bg-bd-vfi-ad)# rd auto
(config-l2vpn-bg-bd-vfi-ad)# route-target 1.1.1.1:100
(config-l2vpn-bg-bd-vfi-ad)# commit
```

BGP オートディスカバリおよび LDP シグナリングでの VPLS

図 25 に、BGP オートディスカバリ（AD）および LDP シグナリングで VPLS を設定する例を示します。

図 25　BGP オートディスカバリおよび LDP シグナリングでの VPLS



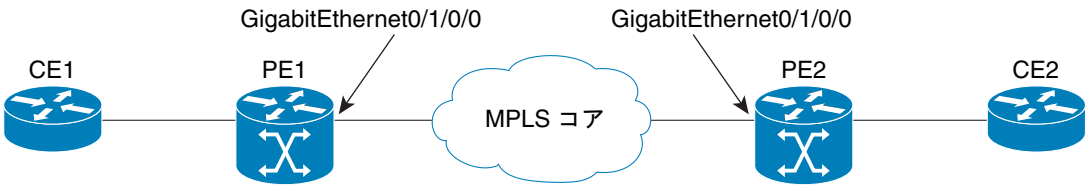
PE1 での設定：

```
l2vpn
  router-id 10.10.10.10
  bridge group bg1
   bridge-domain bd1
    vfi vf1
    vpn-id 100
    autodiscovery bgp
    rd 1:100
    router-target 12:12
```

PE2 での設定：

```
l2vpn
  router-id 20.20.20.20
  bridge group bg1
   bridge-domain bd1
    vfi vf1
     vpn-id 100
     autodiscovery bgp
      rd 2:200
      router-target 12:12
      signaling-protocol ldp
       vpls-id 120:100
```

ディスカバリおよびシグナリングの属性



PE1 での設定：

```
LDP Router ID - 1.1.1.1
BGP Router ID - 1.1.1.100
Peer Address - 1.1.1.10
L2VPN Router ID - 10.10.10.10
Route Distinguisher - 1:100
```

PE1 と PE2 間の共通の設定：

```
ASN - 120
VPN ID - 100
VPLS ID - 120:100
Route Target - 12:12
```

PE2 での設定：

```
LDP Router ID - 2.2.2.2
BGP Router ID - 2.2.2.200
Peer Address - 2.2.2.20
L2VPN Router ID - 20.20.20.20
Route Distinguisher - 2:200
```

ディスカバリ属性

PE1 で送信される NLRI：

```
Source Address - 1.1.1.10
Destination Address - 2.2.2.20
Length - 14
Route Distinguisher - 1:100
L2VPN Router ID - 10.10.10.10
VPLS ID - 120:100
Route Target - 12:12
```

PE2 で送信される NLRI：

```
Source Address - 2.2.2.20
Destination Address - 1.1.1.10
Length - 14
Route Distinguisher - 2:200
L2VPN Router ID - 20.20.20.20
VPLS ID - 120:100
Route Target - 12:12
```

ダイナミック ARP インスペクションの設定：例

次に、ブリッジ ドメインで基本的なダイナミック ARP インスペクションを設定する例を示します。

```
config
l2vpn
 bridge group MyGroup
 bridge-domain MyDomain
 dynamic-arp-inspection logging
```

次に、ブリッジ ポートで基本的なダイナミック ARP インスペクションを設定する例を示します。

```
config
l2vpn
 bridge group MyGroup
 bridge-domain MyDomain
 interface gigabitEthernet 0/1/0/0.1
 dynamic-arp-inspection logging
```

次に、ブリッジ ドメインでオプションのダイナミック ARP インスペクションを設定する例を示します。

```
l2vpn
  bridge group SECURE
    bridge-domain SECURE-DAI
      dynamic-arp-inspection
        logging
        address-validation
          src-mac
          dst-mac
           ipv4
```

次に、ブリッジ ポートでオプションのダイナミック ARP インスペクションを設定する例を示します。

```
l2vpn
  bridge group SECURE
    bridge-domain SECURE-DAI
      interface GigabitEthernet0/0/0/1.10
        dynamic-arp-inspection
          logging
          address-validation
           src-mac
           dst-mac
           ipv4
```

次に、**show l2vpn bridge-domain** *bd-name SECURE-DAI* **detail** コマンドの出力例を示します。

```
#show l2vpn bridge-domain bd-name SECURE-DAI detail
Bridge group: SECURE, bridge-domain: SECURE-DAI, id: 2, state: up,
…
Dynamic ARP Inspection: enabled, Logging: enabled
Dynamic ARP Inspection Address Validation:
   IPv4 verification: enabled
   Source MAC verification: enabled
    Destination MAC verification: enabled
…
List of ACs:
 AC: GigabitEthernet0/0/0/1.10, state is up
…
    Dynamic ARP Inspection: enabled, Logging: enabled
    Dynamic ARP Inspection Address Validation:
        IPv4 verification: enabled
        Source MAC verification: enabled
        Destination MAC verification: enabled
        IP Source Guard: enabled, Logging: enabled

…
   Dynamic ARP inspection drop counters:
      packets: 1000, bytes: 64000
```

次に、**show l2vpn forwarding interface** *interface-name* **detail location** *location-name* コマンドの出力例を示します。

```
#show l2vpn forwarding interface g0/0/0/1.10 det location 0/0/CPU0
Local interface: GigabitEthernet0/0/0/1.10, Xconnect id: 0x40001, Status: up

…
    Dynamic ARP Inspection: enabled, Logging: enabled
    Dynamic ARP Inspection Address Validation:
        IPv4 verification: enabled
        Source MAC verification: enabled
        Destination MAC verification: enabled
        IP Source Guard: enabled, Logging: enabled
```

…
次に、ロギング表示の例を示します。

```
LC/0/0/CPU0:Jun 16 13:28:28.697 : l2fib[188]: %L2-L2FIB-5-SECURITY_DAI_VIOLATION_AC :
Dynamic ARP inspection in AC GigabitEthernet0_0_0_7.1000 detected violated packet - source
MAC: 0000.0000.0065, destination MAC: 0000.0040.0000, sender MAC: 0000.0000.0064, target
MAC: 0000.0000.0000, sender IP: 5.6.6.6, target IP: 130.10.3.2

LC/0/5/CPU0:Jun 16 13:28:38.716 : l2fib[188]: %L2-L2FIB-5-SECURITY_DAI_VIOLATION_AC :
Dynamic ARP inspection in AC Bundle-Ether100.103 detected violated packet - source MAC:
0000.0000.0067, destination MAC: 0000.2300.0000, sender MAC: 0000.7800.0034, target MAC:
0000.0000.0000, sender IP: 130.2.5.1, target IP: 50.5.1.25
```

IP ソース ガードの設定：例

次に、ブリッジ ドメインで基本的な IP ソース ガードを設定する例を示します。

```
config
l2vpn
 bridge group MyGroup
 bridge-domain MyDomain
 ip-source-guard logging
```

次に、ブリッジ ポートで基本的な IP ソース ガードを設定する例を示します。

```
config
l2vpn
 bridge group MyGroup
 bridge-domain MyDomain
 interface gigabitEthernet 0/1/0/0.1
 ip-source-guard logging
```

次に、ブリッジ ドメインでオプションの IP ソース ガードを設定する例を示します。

```
l2vpn
  bridge group SECURE
    bridge-domain SECURE-IPSG
       ip-source-guard
         logging
```

次に、ブリッジ ポートでオプションの IP ソース ガードを設定する例を示します。

```
l2vpn
  bridge group SECURE
    bridge-domain SECURE-IPSG
      interface GigabitEthernet0/0/0/1.10
        ip-source-guard
          logging
```

次に、**show l2vpn bridge-domain** *bd-name ipsg-name* **detail** コマンドの出力例を示します。

```
# show l2vpn bridge-domain bd-name SECURE-IPSG detail
Bridge group: SECURE, bridge-domain: SECURE-IPSG, id: 2, state: up,
…
  IP Source Guard: enabled, Logging: enabled
…
List of ACs:
 AC: GigabitEthernet0/0/0/1.10, state is up
…

   IP Source Guard: enabled, Logging: enabled
…
```

```
   IP source guard drop counters:
      packets: 1000, bytes: 64000
```

次に、**show l2vpn forwarding interface** *interface-name* **detail location** *location-name* コマンドの出力例を示します。

```
# show l2vpn forwarding interface g0/0/0/1.10 detail location 0/0/CPU0
Local interface: GigabitEthernet0/0/0/1.10, Xconnect id: 0x40001, Status: up

…
    IP Source Guard: enabled, Logging: enabled
```

次に、ロギング表示の例を示します。

```
LC/0/0/CPU0:Jun 16 13:32:25.334 : l2fib[188]: %L2-L2FIB-5-SECURITY_IPSG_VIOLATION_AC : IP
source guard in AC  GigabitEthernet0_0_0_7.1001 detected violated packet - source MAC:
0000.0000.0200, destination MAC: 0000.0003.0000, source IP: 130.0.0.1, destination IP:
125.34.2.5


LC/0/5/CPU0:Jun 16 13:33:25.530 : l2fib[188]: %L2-L2FIB-5-SECURITY_IPSG_VIOLATION_AC : IP
source guard in AC Bundle-Ether100.100  detected violated packet - source MAC:
0000.0000.0064, destination MAC: 0000.0040.0000, source IP: 14.5.1.3, destination IP:
45.1.1.10
```

G.8032 イーサネット リング保護の設定：例

この設定例では、完全な G.8032 設定に含まれている要素について説明します。

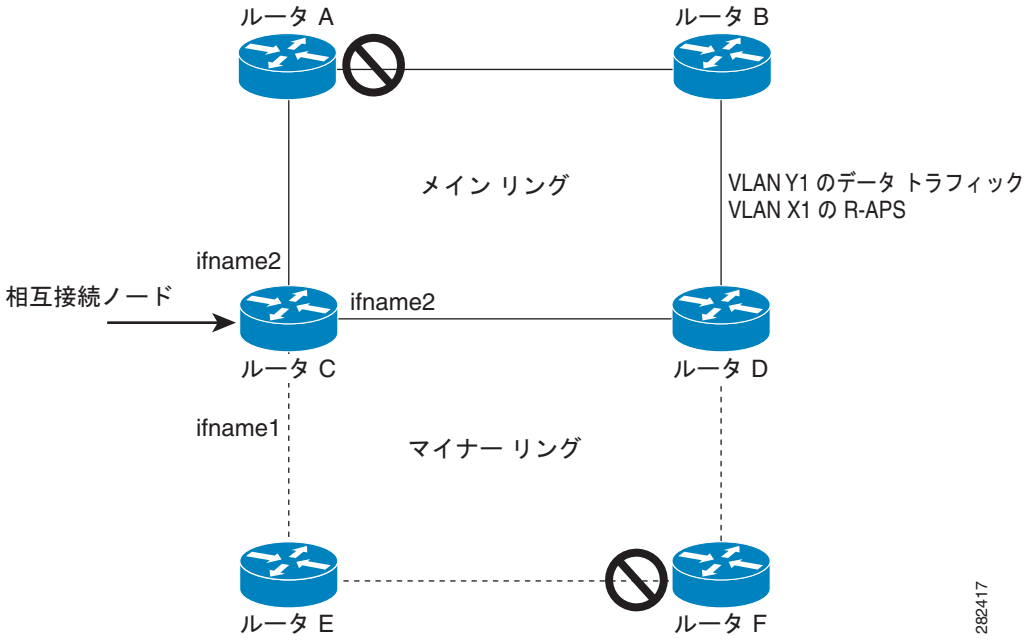
```
# Configure the ERP profile characteristics if ERP instance behaviors are non-default.
ethernet ring g8032 profile ERP-profile
  timer wtr 60
  timer guard 100
  timer hold-off 1
  non-revertive

# Configure CFM MEPs and configure to monitor the ring links.
ethernet cfm
  domain domain1
    service link1 down-meps
      continuity-check interval 100ms
      efd
      mep crosscheck
        mep-id 2
  domain domain2
    service link2 down-meps
      continuity-check interval 100ms
      efd protection-switching
      mep crosscheck
        mep id 2

Interface Gig 0/0/0/0
  ethernet cfm mep domain domain1 service link1 mep-id 1
Interface Gig 1/1/0/0
  ethernet cfm mep domain domain2 service link2 mep-id 1

# Configure the ERP instance under L2VPN
l2vpn
  ethernet ring g8032 RingA
    port0 interface g0/0/0/0
```

```
    port1 interface g0/1/0/0
    instance 1
      description BD2-ring
      profile ERP-profile
      rpl port0 owner
      vlan-ids 10-100
      aps channel
        level 3
        port0 interface g0/0/0/0.1
        port1 interface g1/1/0/0.1

# Set up the bridge domains
bridge group ABC
    bridge-domain BD2
      interface Gig 0/0/0/0.2
      interface Gig 0/1/0/0.2
      interface Gig 0/2/0/0.2

    bridge-domain BD2-APS
      interface Gig 0/0/0/0.1
      interface Gig 1/1/0/0.1

# EFPs configuration
interface Gig 0/0/0/0.1 l2transport
  encapsulation dot1q 5

interface Gig 1/1/0/0.1 l2transport
  encapsulation dot1q 5

interface g 0/0/0/0.2 l2transport
  encapsulation dot1q 10-100

interface g 0/1/0/0.2 l2transport
  encapsulation dot1q 10-100

interface g 0/2/0/0.2 l2transport
  encapsulation dot1q 10-100
```

相互接続ノードの設定：例

次に、相互接続ノードを設定する例を示します。図 26 で、開いたリング シナリオについて説明します。

図 26　リング シナリオ：相互接続ノード



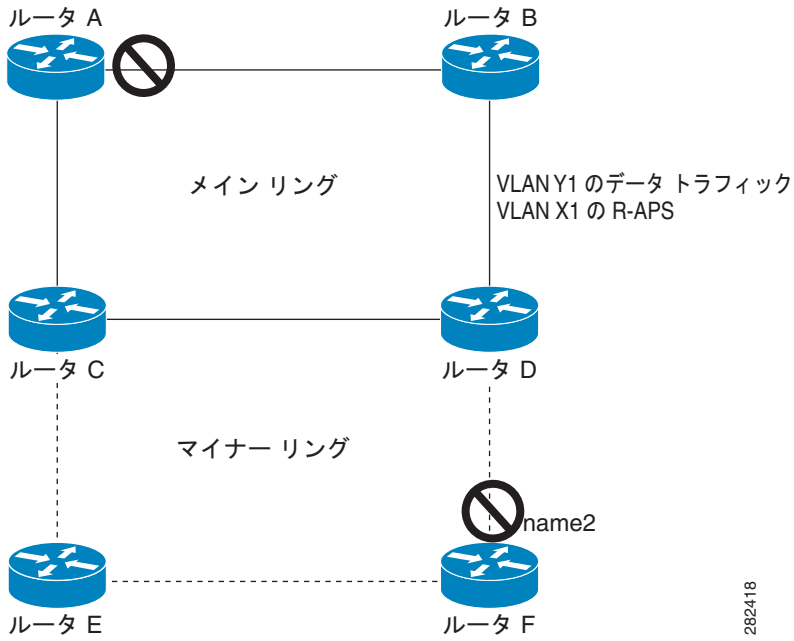
ルータ C（開いたリング：ルータ C）で G.8032 を設定するために必要な最小設定：

```
interface <ifname1.1> l2transport
    encapsulation dot1q X1
interface <ifname1.10> l2transport
    encapsulation dot1q Y1
interface <ifname2.10> l2transport
    encapsulation dot1q Y1
interface <ifname3.10> l2transport
    encapsulation dot1q Y1
l2vpn
ethernet ring g8032 <ring-name>
      port0 interface <main port ifname1>
      port1 interface none #? This router is connected to an interconnection node
      open-ring #? Mandatory when a router is part of an open-ring
      instance <1-2>
         inclusion-list vlan-ids  X1-Y1
         aps-channel
           Port0 interface <ifname1.1>
           Port1 none #? This router is connected to an interconnection node
    bridge group bg1
       bridge-domain bd-aps#? APS-channel has its own bridge domain
          <ifname1.1> #? There is only one APS-channel at the interconnection node
       bridge-domain bd-traffic #? Data traffic has its own bridge domain
          <ifname1.10>
          <ifname2.10>
          <ifname3.10>
```

開いたリングのノードの設定：例

次に、開いたリングのノード部分を設定する例を示します。図 27 で、開いたリング シナリオについて説明します。

図 27　開いたリング シナリオ



開いたリングのノード（ルータ F で開いたリングのノード部分）で G.8032 を設定するに必要な最小設定：

```
interface <ifname1.1> l2transport
    encapsulation dot1q X1
interface <ifname2.1> l2transport
    encapsulation dot1q X1
interface <ifname1.10> l2transport
    encapsulation dot1q Y1
interface <ifname2.10> l2transport
    encapsulation dot1q Y1
l2vpn
   ethernet ring g8032 <ring-name>
      port0 interface <main port ifname1>
      port1 interface <main port ifname2>
      open-ring #? Mandatory when a router is part of an open-ring
      instance <1-2>
         inclusion-list vlan-ids  X1-Y1
         rpl port1 owner #? This node is RPL owner and <main port ifname2> is blocked
         aps-channel
           port0 interface <ifname1.1>
           port1 interface <ifname2.1>
    bridge group bg1
       bridge-domain bd-aps#? APS-channel has its own bridge domain
          <ifname1.1>
          <ifname2.1>
       bridge-domain bd-traffic #? Data traffic has its own bridge domain
          <ifname1.10>
          <ifname2.10>
```

Flow Aware Transport 疑似回線の設定：例

この設定例では、VPWS の FAT PW によるロード バランシングをイネーブルにする方法を示します。

```
l2vpn
pw-class class1
    encapsulation mpls
        load-balancing flow-label transmit
    !
 !
pw-class class2
    encapsulation mpls
        load-balancing flow-label both
   !

xconnect group group1
    p2p p1
        interface GigabitEthernet 0/0/0/0.1
        neighbor 1.1.1.1 pw-id 1
            pw-class class1
        !
    !
!
```

この設定例では、VPLS の FAT PW によるロード バランシングをイネーブルにする方法を示します。

（注） VPLS の場合、ブリッジドメイン レベルでの設定は、すべての PW（アクセスおよび VFI PW）に適用されます。疑似回線クラスは、手動 PW の設定を上書きするために定義されます。

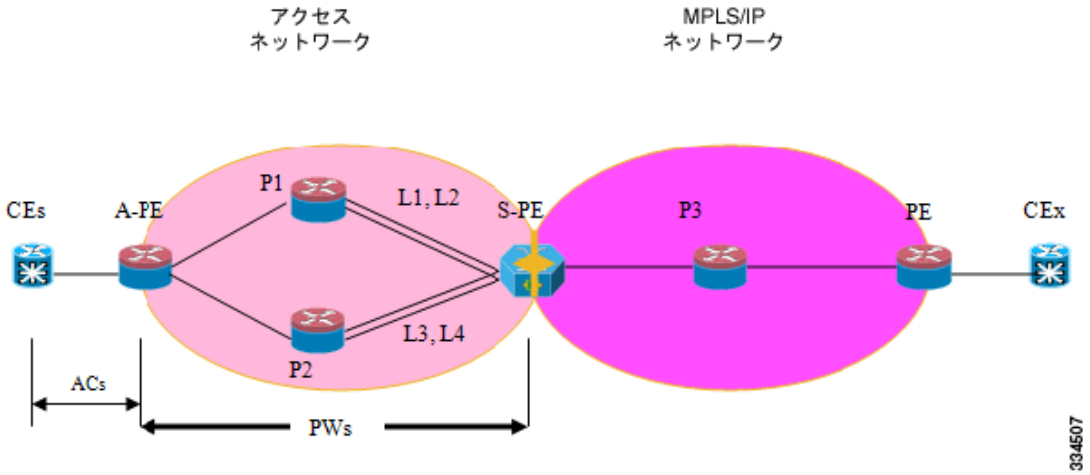
```
l2vpn
pw-class class1
 encapsulation mpls
 load-balancing flow-label both

bridge group group1
    bridge-domain domain1
       vfi vfi2-auto-bgp
                autodiscovery bgp
                   signaling-protocol bgp
                       load-balancing flow-label both static
                !
            !
        !
    !
    bridge-domain domain2
        vfi vfi2-auto-ldp
                autodiscovery bgp
                   signaling-protocol ldp
                       load-balancing flow-label both static
                !
            !
        !
    !
!
```

疑似回線ヘッドエンドの設定：例

次に、疑似回線ヘッドエンドを設定する例を示します。

図 28 のトポロジを検討します。

図 28　　疑似回線ヘッドエンドの例



A-PE に接続された複数の CE があります（各 CE は 1 つのリンクによって接続されます）。アクセス ネットワークの A-PE と S-PE 間に 2 つの P ルータがあります。S-PE は、2 つのリンクで P1 に接続されています。これらは L1 および L2（P1 および S-PE の 2 つの異なるラインカード上）をリンクします、たとえば、それぞれ Gig0/1/0/0 および Gig0/2/0/0 になります。

S-PE は、P2 に 2 つのリンクで接続され、L3 および L4（P2 および S-PE の 2 つの異なるラインカード上）をリンクします、たとえば、それぞれ Gig0/1/0/1 および Gig0/2/0/1 になります。各 CE-APE リンクについて、相互接続（AC-PW）が A-PE 上に設定されます。A-PE は、ルーティングと PW シグナリングに router-id 100.100.100.100 を使用します。PW シグナリングには、S-PE 上の 2 つの router-id（111.111.111.111 および 112.112.112.112（rx pin-down 用）など）が使用されます。ルーティングには router-id 110.110.110.110 が使用されます。

CE の設定

Ge0/3/0/0（CE1 と A-PE）および Ge0/3/0/1（CE2 と A-PE）を介して接続された 2 つの CE を考慮します。

CE1

```
interface Gig0/3/0/0
 ipv4 address 10.1.1.1/24
router static
 address-family ipv4 unicast
  110.110.110.110 Gig0/3/0/0
  A.B.C.D/N 110.110.110.110
```

CE2

```
interface Gig0/3/0/1
 ipv4 address 10.1.2.1/24
router static
 address-family ipv4 unicast
```

```
  110.110.110.110 Gig0/3/0/1
  A.B.C.D/N 110.110.110.110
```

A-PE の設定

A-PE の場合、各 CE 接続に 1 つの相互接続があります。ここで上記の 2 つの CE 接続を設定します。接続は両方とも相互接続である L2 リンクです。各相互接続には S-PE 宛ての PW がありますが、ここでは PW をピンダウンする場所（[L1, L4] または [L2, L3]）に応じて別のネイバー アドレスを使用します。

```
interface Gig0/3/0/0
 l2transport
interface Gig0/3/0/1
 l2transport

l2vpn
 xconnect group pwhe
  p2p pwhe_spe_1
   interface Gig0/3/0/0
   neighbor 111.111.111.111 pw-id 1
  p2p pwhe_spe_2
   interface Gig0/3/0/1
   neighbor 112.112.112.112 pw-id 2
```

P ルータの設定

S-PE の rx ピンダウン用の P ルータには、スタティック ルートが必要です。つまり、PW に、特定のリンクを介した特定のアドレスへの転送を強制します。

P1

```
router static
 address-family ipv4 unicast
  111.111.111.111 Gig0/1/0/0
  112.112.112.112 Gig0/2/0/0
```

P2

```
router static
 address-family ipv4 unicast
  111.111.111.111 Gig0/2/0/1
  112.112.112.112 Gig0/1/0/1
```

S-PE の設定

S-PE の場合、2 つの PW-HE インターフェイス（各 PW に 1 つ）があり、tx ピンダウンにそれぞれ異なるインターフェイス リストを使用します（tx ピンダウンは rx ピンダウン用の P ルータでスタティックな設定が一致する必要があります）。各 PW-HE には A-PE に向かう PW があります（pw-id が A-PE のものと一致する必要があります）。

```
generic-interface-list il1
 interface gig0/1/0/0
 interface gig0/2/0/0
generic-interface-list il2
 interface gig0/1/0/1
 interface gig0/2/0/1

interface pw-ether1
 ipv4 address 10.1.1.2/24
 attach generic-interface-list il1
interface pw-ether2
```

```
 ipv4 address 10.1.2.2/24
 attach generic-interface-list il2

l2vpn
 xconnect group pwhe
  p2p pwhe1
   interface pw-ether1
   neighbor 100.100.100.100 pw-id 1
  p2p pwhe2
   interface pw-ether2
   neighbor 100.100.100.100 pw-id 2
```

L2VPN over GRE の設定：例

IGP の下の PW コア インターフェイスを設定し、ループバックを到達可能にします。トンネル送信元を設定し、トンネルが現在のループバックになるように、およびピア PE ループバックの宛先になるようにします。ここでは、IGP（OSPF または ISIS）内、および **mpls ldp** の下に GRE トンネルを設定し、LDP ネイバーが PW の PE 間で確立されるようにします。これにより、トンネルで PW がアップするようになります。

PE1 の設定：

```
router ospf 1
router-id 1.1.1.1
 area 0
  interface Loopback0
  interface TenGigE0/0/0/1
router ospf 2
 router-id 200.200.200.200
 area 0
  interface Loopback1000
   interface tunnel-ip1
mpls ldp
 router-id 200.200.200.200
 interface tunnel-ip1
```

PE2 の設定：

```
router ospf 1
 router-id 3.3.3.3
 area 0
  interface Loopback0
interface TenGigE0/2/0/3
router ospf 2
 router-id 201.201.201.201
 area 0
  interface Loopback1000
   interface tunnel-ip1
!
mpls ldp
 router-id 201.201.201.201
 interface tunnel-ip1
```

疑似回線の優先パスとしての GRE トンネルの設定：例

次に、疑似回線の優先パスとして GRE トンネルを設定する例を示します。

```
l2vpn
 pw-class gre
```

```
encapsulation mpls
preferred-path interface tunnel-ip 1 fallback disable
```

その他の関連資料

VPLS の実装に関する詳細情報については、次を参照してください。

関連資料

関連項目	参照先
Cisco IOS XRL2VPN コマンド	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』の「*Point to Point Layer 2 Services Commands*」
MPLS VPLS-related コマンド	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』の「*Multipoint Layer 2 Services Commands*」
スタートアップ資料	『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』
VPLS ブリッジにおけるトラフィック ストーム制御	『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』の「*Traffic Storm Control under VPLS Bridges on Cisco ASR 9000 Series Routers*」
VPLS ブリッジのレイヤ 2 マルチキャスト	『*Cisco ASR 9000 Series Aggregation Services Router Multicast Configuration Guide*』の「*Layer 2 Multicast Using IGMP Snooping*」

標準

標準[1]	タイトル
draft-ietf-l2vpn-vpls-ldp-09	『*Virtual Private LAN Services Using LDP*』

1. サポートされている規格がすべて記載されているわけではありません。

MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
RFC 4447	『*Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*』2006 年 4 月
RFC 4448	『*Encapsulation Methods for Transport of Ethernet over MPLS Networks*』2006 年 4 月
RFC 4762	『*Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

その他の関連資料

IEEE 802.1ah プロバイダー バックボーン ブリッジの実装

このモジュールでは、Cisco ASR 9000 シリーズ ルータでの IEEE 802.1ah プロバイダー バックボーン ブリッジの概念および設定情報を提供します。IEEE 802.1ah 規格（Ref（4））は、大規模エンドツーエンド レイヤ 2 プロバイダー ブリッジ型ネットワークを構築するために、複数のプロバイダー ブリッジ型ネットワークを相互接続する手段を提供します。

Cisco ASR 9000 シリーズ アグリゲーション サービス ルータは現在、プロバイダー バックボーン ブリッジが VPLS ネットワークである場合のシナリオをサポートします。また、PBB エッジ ブリッジ ドメインおよびコア ブリッジ ドメインの疑似回線を設定できます。いずれのブリッジ ドメインでも、疑似回線の機能はネイティブ ブリッジ ドメインの場合と同様です。

IEEE 802.1ah プロバイダー バックボーン ブリッジを実装するための機能の履歴

リリース	変更内容
リリース 3.9.1	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 4.3.0	次の機能のサポートが追加されました。 • プロバイダー バックボーン ブリッジ VPLS • Multiple I-SID Registration Protocol Lite（MIRP Lite）

内容

802.1ah プロバイダー バックボーン ブリッジを実装するための前提条件

この前提条件は、802.1ah プロバイダー バックボーン ブリッジの実装に適用されます。

- このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

  ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- マルチポイント ブリッジングの概念に関する知識が必要です。マルチポイント レイヤ 2 サービスの実装モジュールを参照してください。

802.1ah サービス プロバイダー バックボーンブリッジの実装に関する情報

802.1ah を実装するには、次の概念を理解している必要があります。

- 「IEEE 802.1ah 規格の利点」（P.348）
- 「IEEE 802.1ah 規格プロバイダー バックボーン ブリッジ概要」（P.349）
- 「バックボーン エッジ ブリッジ」（P.350）
- 「IB-BEB」（P.351）
- 「Multiple I-SID Registration Protocol Lite」（P.352）

IEEE 802.1ah 規格の利点

IEEE 802.1ah プロバイダー バックボーン ブリッジの利点を以下に示します。

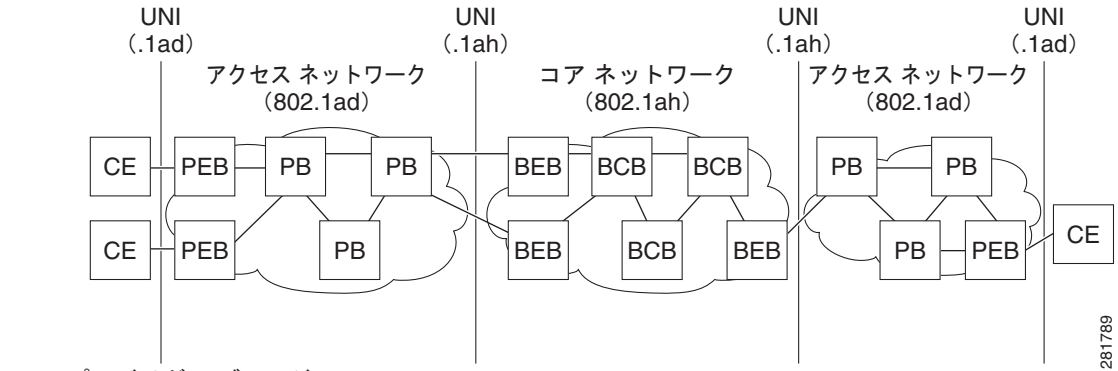
- サービス インスタンスのスケーラビリティの向上：サービス プロバイダーのプロバイダー ブリッジ型ネットワーク（PBN）でのサービス（サービス VLAN またはサービス インスタンス）の数を拡張できます。
- MAC アドレスのスケーラビリティ：MAC アドレスなどのカスタマー パケットを、新しい MAC アドレス（バックボーン ブリッジ MAC アドレス）を持つ新しいイーサネット フレームにカプセル化します。これは、バックボーン コア ブリッジが顧客ごとにすべての MAC アドレスを学習する必要性をなくし、バックボーン エッジ ブリッジの負荷を軽減します。
- VPLS 疑似回線の低減およびメッシュ スケーラビリティ：IP/MPLS コアの疑似回線の数を大幅に削減できます。これは、単一の VPLS サービスが複数のカスタマー サービス インスタンスを転送できるため、IP/MPLS コア内で、より少ない疑似回線で多くのカスタマー サービスを転送できるためです。
- レイヤ 2 バックボーン トラフィック エンジニアリング：サービス識別機能を分類することにより、レイヤ 2 トラフィック エンジニアリング機能の明示的な制御を可能にし、これを I-TAG に移動します。これによりレイヤ 2 トラフィック エンジニアリング機能に対してバックボーン VLAN が使用可能な状態が維持されます。
- ポイントツーポイント サービスのスケーラビリティおよび最適化：サービス多重化の複数のオプションとエンドポイント検出を含むポイントツーポイント サービスの実装をイネーブルにします。

- バックボーンのフラッディング トラフィックの削減 : ネットワークのコアの MAC アドレス数が少ないことにより、トポロジ変更で MAC テーブルがフラッシュされると、再学習される MAC アドレスの数が少ないためコア ネットワークのフラッディング トラフィックの量が削減されます。

IEEE 802.1ah 規格プロバイダー バックボーン ブリッジ概要

IEEE 802.1ah プロバイダー バックボーン ブリッジ機能は、プロバイダー バックボーン ブリッジ型ネットワーク（PBBN）のエッジで、バックボーン エッジ ブリッジ（BEB）のエンドユーザ トラフィックをカプセル化またはカプセル化解除します。バックボーン コア ブリッジ（BCB）ベースのネットワークは、PBBN 内での IEEE 802.1ah カプセル化フレームの内部転送を提供します。図 29 は、一般的な 802.1ah PBB のネットワークを表しています。

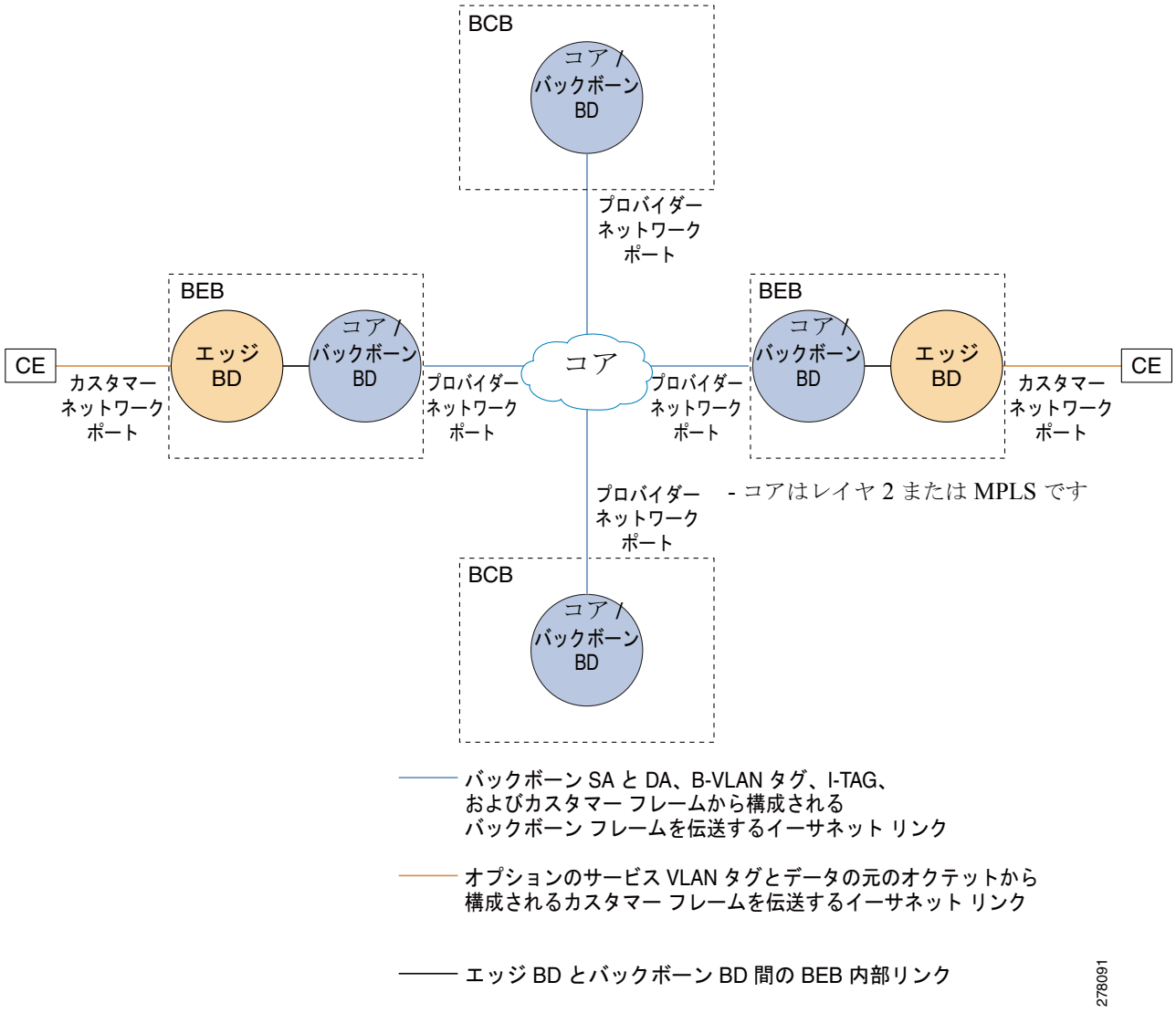
図 29　　IEEE 802.1ah プロバイダー バックボーン ブリッジ



PB : プロバイダー ブリッジ

図 30 は、一般的なプロバイダーのバックボーン ネットワーク トポロジを表しています。

図 30　　プロバイダー バックボーン ネットワークのトポロジ



バックボーン エッジ ブリッジ

バックボーン エッジ ブリッジ（BEB）には、I-Component または B-Component を含めることができます。I-Component は、サービス VLAN ID（S-VID）をサービス インスタンス ID（I-SID）にマッピングし、バックボーン VLAN タグ（B-Tag）なしのプロバイダー バックボーン ブリッジ（PBB）ヘッダーを追加します。B-Component は、I-SID をバックボーン VID（B-VID）にマッピングし、B-Tag を持つ PBB ヘッダーを追加します。

IEEE 802.1ah 規格では、次の 3 つのタイプの BEB が指定されています。

- B-BEB には、MAC-in-MAC ブリッジの B-Component が含まれます。これは、I-SID を検証し、フレームを Backbone VLAN（B-VLAN）にマッピングします。また、コア ブリッジ内の B-VLANS に基づいてトラフィックを切り替えます。

- I-BEB には、MAC-in-MAC ブリッジの I-Component が含まれます。これは、B-MAC カプセル化を実行し、プロバイダー VLAN タグ（S-tag）、カスタマー VLAN タグ（C-Tag）、または S-tag/C-tag のペアに基づいて I-SID を挿入します。
- IB-BEB には、LAN セグメントによって相互接続された 1 つ以上の I-Component と 1 つの B-Component が含まれます。

（注） Cisco ASR 9000 シリーズ ルータでは、IB-BEB だけがサポートされます。Cisco IOS XR は、エッジ ノードで IB-BEB ブリッジ タイプをサポートします。

IB-BEB

IB-BEB には、I-Component と B-Component の両方が含まれます。このブリッジは、B-MAC を選択し、プロバイダー VLAN タグ（S-tag）、カスタマー VLAN タグ（C-Tag）、または S-tag と C-Tag の両方に基づいて I-SID を挿入します。これは、I-SID を検証し、B-VLAN 上でフレームを送受信します。

IEEE 802.1ah プロバイダー バックボーン ブリッジ機能は、IEEE 802.1ah 規格で要求されるすべてのサービスをサポートし、さらにサービスを拡張して次の追加機能を提供します。

- S-Tagged サービス：
  - 多重化環境では、各 S-tag が I-SID にマッピングされ、各 S-tag は保持または削除できます。
  - バンドル環境では、複数の S-tag が同じ I-SID にマッピングされ、S-tag は保持する必要があります。
- C-Tagged サービス
  - 多重化環境では、各 C-tag が I-SID にマッピングされ、各 C-tag は保持または削除できます。
  - バンドル環境では、複数の C-tag が同じ I-SID にマッピングされ、C-tag は保持する必要があります。
- S/C-Tagged サービス：
  - 多重化環境では、各 S-tag/C-tag ペアが I-SID にマッピングされます。S-tag または S-tag/C-tag ペアは、保持または削除できます。
  - バンドル環境では、複数の S-tag/C-tag ペアが同じ I-SID にマッピングされ、S-tag/C-tag ペアは保持する必要があります。
- ポートベースのサービス
  - ポートベースのサービス インターフェイスは、カスタマー ネットワーク ポート（CNP）で提供されます。ポートベースのサービス インターフェイスは、C-VLAN ブリッジ、802.1d ブリッジ、ルータ、またはエンド ステーションに接続できます。このインターフェイスが提供するサービスは、単一のバックボーン サービス インスタンスのバックボーン上で、S-Tag なしですべてのフレームを転送します。ポートベース インターフェイスは、ヌル以外の VLAN ID を持つ S タグを含むすべてのフレームをドロップします。

    次に、ポートベースのサービスを設定する例を示します。

    ```
    interface GigabitEthernet0/0/0/10.100 l2transport
    encapsulation untagged
    ```
    --> タグなしフレームの EFP を作成します。

    ```
    interface GigabitEthernet0/0/0/10.101 l2transport
    encapsulation dot1ad priority-tagged
    ```
    --> ヌルの S-tag 付きフレームの EFP を作成します。

    ```
    interface GigabitEthernet0/0/0/10.102 l2transport
    ```

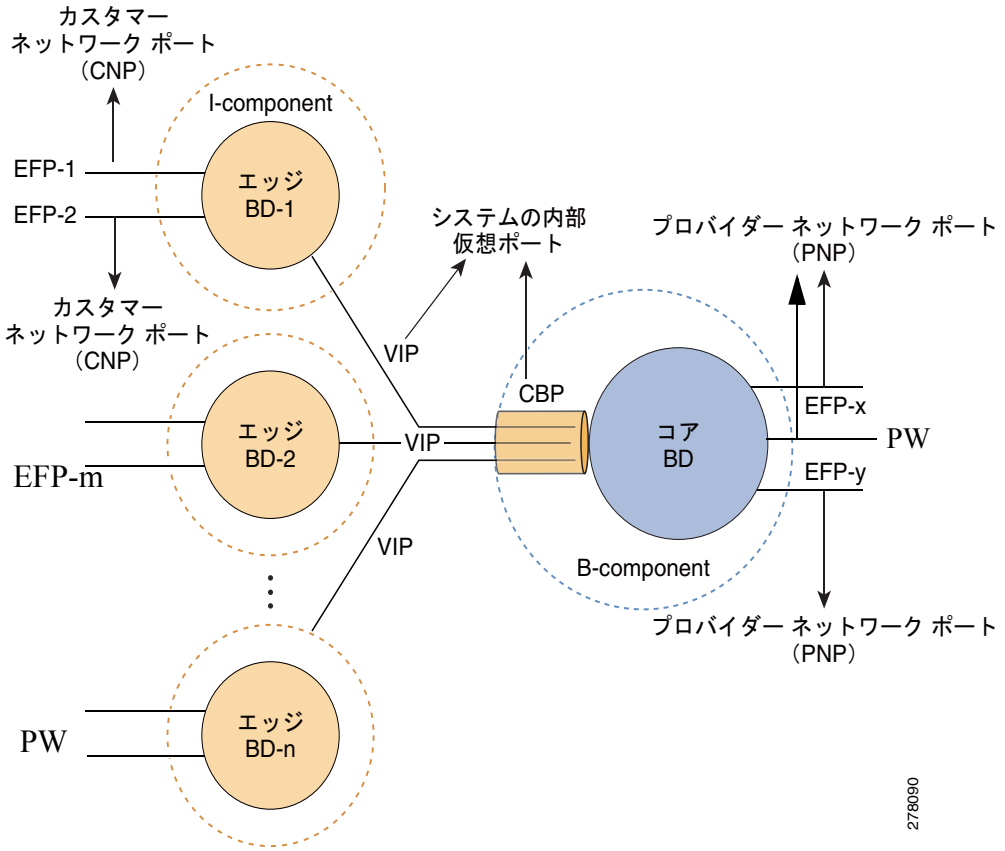
```
encapsulation dot1q priority-tagged
```
--> ヌルの C-tag 付きフレームの EFP を作成します。

```
interface GigabitEthernet0/0/0/10.103 l2transport
encapsulation dot1q any
```
--> C-tag 付きフレームの EFP を作成します。

（注） ポートベースのサービスを設定するには、上記のすべての EFP を、同じエッジ ブリッジ ドメインに追加する必要があります。

図 31 は、Cisco ASR 9000 シリーズ ルータでの PBB ブリッジ コンポーネント トポロジを表しています。

図 31　Cisco ASR 9000 シリーズ ルータでの PBB ブリッジ コンポーネント トポロジ



Multiple I-SID Registration Protocol Lite

802.1Qbe マルチ I-SID 登録プロトコル（MIRP）規格は、I-SID ごとに I-Component のフィルタリング データベースに保持される学習された MAC アドレスの登録エントリをフラッシュする機能を提供します。バックボーン サービス インスタンス ID（I-SID）は、フレームのバックボーン サービス インスタンスを一意に識別するバックボーン サービス インスタンス タグのフィールドです。MIRP は I-SID をフラッシュするメカニズムを定義します。また、プロバイダー バックボーン ブリッジ型ネットワークに接続されたネットワークで発生しているトポロジ変更を処理する必須機能を持っています。

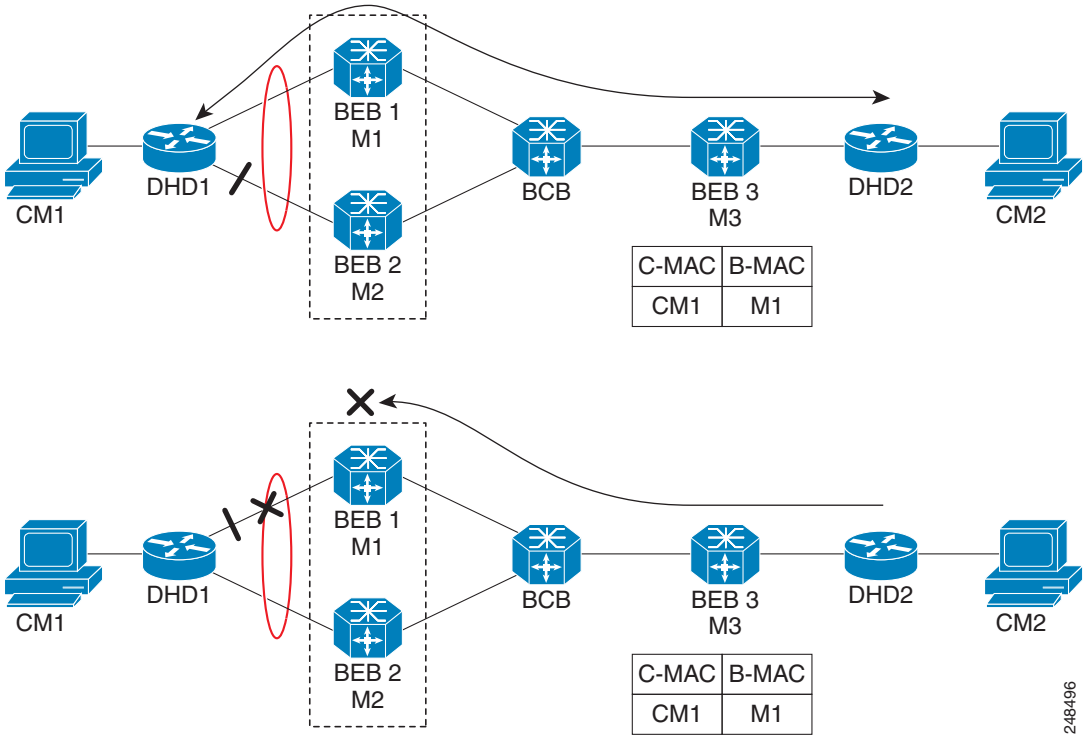
バックボーン エッジ ブリッジ（BEB）は、影響を受ける可能性のある（カスタマー MAC アドレスとバックボーン MAC アドレスについて、学習した特定の関連付けを変更する必要のある）他の BEB に信号を送信します。MIRP がない場合、プロバイダー バックボーン ネットワーク上のカスタマー接続では、アクセス ネットワークでのトポロジの変更後の接続の復元に数分かかることがあります。

以前のリリースでは、PBB エッジ ブリッジ ドメインでポートが使用不可能になるかスパニングツリートポロジが変更されることによりブリッジ フォワーディング トポロジの変更が発生すると、PBB トラフィックが MAC エージング サイクルにドロップされました。このため、PBB ブリッジの使用は厳しく制限されていました。

Cisco ASR 9000 シリーズ アグリゲーション サービス ルータは、Multiple I-SID Registration Protocol Lite（MIRP Lite）と呼ばれる MIRP プロトコルの簡略化された実装をサポートしています。MIRP Lite 機能は、サイトでのトポロジ変更の検出をイネーブルにします。サイトがトポロジの変更を検出すると、特別に定義されたパケットは、PBB ネットワークのすべてのリモート エッジ サイトにフラッディングされます。送信者のサイトでは、MAC フラッシュを必要とする I-SID を指定するために、I-Component の I-SID がフレーム ヘッダーの I-TAG に配置されます。受信者のサイトでは、各 PBB エッジ スイッチが I-SID のチェックを実行します。I-SID が I-Component の 1 つと一致すると、I-Component の MAC がフラッシュされます。

802.1ah ネットワーク内での MIRP の使用を図 32 に示します。

図 32　802.1ah ネットワーク内での MIRP



デバイス DHD1 は、2 つの 802.1ah バックボーン エッジ ブリッジ（BEB1 と BEB2）にデュアルホーム接続しています。当初のプライマリ パスは BEB1 経由であると想定しています。この構成では、BEB3 は、DHD1 の背後にあるホスト（MAC アドレスは CM1）は、宛先 B-MAC M1 を介して到達できることを学習しています。DHD1 と BEB1 間のリンクに障害が発生し、DHD1 の背後にあるホストが非アクティブのままになっていると、BEB3 の MAC キャッシュ テーブルは、新規のパス ビューが B-MAC アドレスが M2 の BEB2 経由であっても、BEB1 の MAC アドレスを引き続き参照します。DHD2 の背後にあるホストから DHD1 の背後にあるホストに転送されたブリッジ トラフィックは、誤って B-MAC M1 でカプセル化され、MAC トンネルを経由して BEB1 に送信されて、トラフィックがドロップされています。

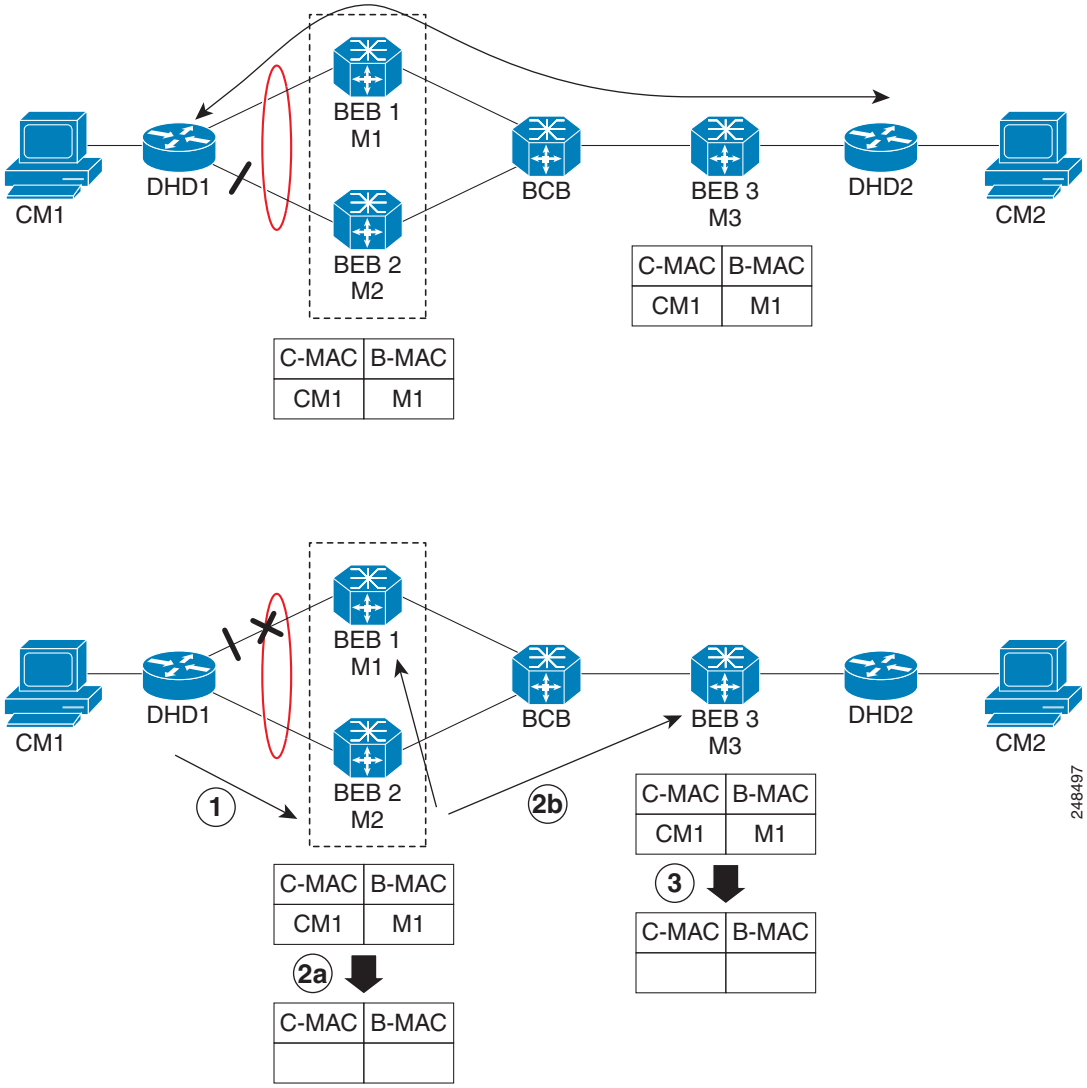
DHD1 と BEB1 間のリンクに障害が発生した場合にトラフィックがドロップされないように、BEB2 は次の 2 つのタスクを実行します。

- サービスに対して固有の MAC アドレス テーブルをフラッシュします。
- MIRP パケットを受信するリモート PE に、固有の MAC テーブルのクリアを要求します。MIRP メッセージは、バックボーン コア ブリッジ（BCB）に対して透過的です。MIRP メッセージは BEB 上で処理されます。BCB だけが B-MAC アドレスに基づいた取得と転送を行っており、C-MAC アドレスでは認識されないためです。

（注） MIRP は、ネイティブ 802.1ah と VPLS 経由の PBB の両方に C-MAC アドレス フラッシュをトリガーします。

図 33 に MIRP の動作を示します。

図 33　MIRP 動作


CM1
DHD1
BEB 1
M1
BEB 2
M2
BCB
BEB 3
M3
DHD2
CM2
C-MAC
B-MAC
CM1
M1
1
2a
2b
3
248497

802.1ah プロバイダー バックボーン ブリッジを実装する方法

この項では、次の手順について説明します。

802.1ah プロバイダー バックボーン ブリッジの実装に関する制約事項

次の機能はサポートされていません。

- MAC-in-MAC 上での相互接続ベースのポイントツーポイント サービス
- 1 つのエッジ ブリッジと複数のコア ブリッジのマッピング
- I タイプのバックボーン エッジ ブリッジ（I-BEB）と B タイプのバックボーン エッジ ブリッジ（B-BEB）
- IEEE 802.1ah over VPLS
- シャーシごとの複数の送信元 B-MAC アドレス
- ネイティブの MPLS LSP カプセル化を通じた 802.1ah フォーマット パケットのダイレクト カプセル化

CNP および PNP ポートでのイーサネット フロー ポイントの設定

カスタマー ネットワーク ポート（CNP）またはプロバイダー ネットワーク ポート（PNP）にイーサネット フロー ポイント（EFP）を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **interface** *type interface-path-id.subinterface* **l2transport**
3. **encapsulation dot1q** *vlan-id*
   または
   encapsulation dot1ad *vlan-id*
   または
   encapsulation dot1ad *vlan-id* **dot1q** *vlan-id*

4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type interface-path-id.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/10.100 l2transport`	L2 スイッチングのインターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 3	**encapsulation dot1q** *vlan-id* or **encapsulation dot1ad** *vlan-id* or **encapsulation dot1ad** *vlan-id* **dot1q** *vlan-id* **例：** `RP/0/RSP0/CPU0:router(config-subif)#` `encapsulation dot1q 100` `or` `encapsulation dot1ad 100` `or` `encapsulation dot1ad 100 dot1q 101`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-subif)# end` または `RP/0/RSP0/CPU0:router(config-subif)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)?` `[cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PBB エッジ ブリッジ ドメインおよびサービス インスタンス ID の設定

PBB エッジ ドメインおよびサービス ID を設定するには、次の作業を行います。

（注） PBB 機能を設定するには、admin ユーザ権限でログインし、**hw-module profile feature l2** コマンドを発行して、PBB 機能をサポートする ASR 9000 イーサネット ラインカードの ucode バージョンを選択します。この設定を行わない限り、PBB 機能は、ASR 9000 イーサネット ラインカードでサポートされません。機能プロファイル設定の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*』を参照してください。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **interface** *type interface-path-id.subinterface*
6. **pbb edge i-sid** service-id **core-bridge** *core-bridge-name*
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-edge`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。

	コマンドまたはアクション	目的
ステップ 5	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。この EFP はエッジ ブリッジの CNP と見なされます。
ステップ 6	**pbb edge i-sid** *service-id* **core-bridge** *core-bridge-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# pbb edge i-sid 1000 core-bridge pbb-core`	サービス ID および割り当てられたコア ブリッジ ドメインを指定して、ブリッジ ドメインを PBB エッジとして設定し、PBB エッジ コンフィギュレーション サブモードを開始します。 このコマンドは、指定したコア ブリッジ ドメインに PBB エッジ ブリッジ ドメインを関連付ける仮想インスタンス ポート（VIP）も作成します。 このブリッジ ドメインのすべてのインターフェイス（ブリッジ ポート）は、カスタマー ネットワーク ポート（CNP）として扱われます。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PBB コア ブリッジ ドメインの設定

PBB コア ブリッジ ドメインを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **interface** *type interface-path-id.subinterface*
6. **pbb core**
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-core`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 5	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。

	コマンドまたはアクション	目的
ステップ 6	**pbb core** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# pbb core`	ブリッジ ドメインを PBB コアとして設定し、PBB コア コンフィギュレーション サブモードを開始します。 このコマンドは、カスタマー ブリッジ ポート（CBP）と呼ばれる内部ポートを作成します。 このブリッジ ドメインのすべてのインターフェイス（ブリッジ ポート）は、プロバイダー ネットワーク ポート（PNP）として扱われます。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-core)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-core)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PBB コア ブリッジ ドメイン下でのバックボーン VLAN タグの設定

PBB コア ブリッジ ドメイン下でバックボーン VLAN タグを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **interface** *type interface-path-id.subinterface*
6. **interface** *type interface-path-id.subinterface*
7. **pbb core**
8. **rewrite ingress tag push dot1ad** *vlan-id* **symmetric**

9. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-core`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 5	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。
ステップ 6	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#interface GigabitEthernet0/5/0/1.15`	ブリッジ ドメインにインターフェイスを追加し、パケットの転送と、同じブリッジ ドメイン内の他のインターフェイスからのパケットの受信を可能にします。これで、このインターフェイスは、このブリッジ ドメイン上の接続回線になります。
ステップ 7	**pbb core** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# pbb core`	ブリッジ ドメインを PBB コアとして設定し、PBB コア コンフィギュレーション サブモードを開始します。 このコマンドは、カスタマー ブリッジ ポート（CBP）と呼ばれる内部ポートを作成します。 このブリッジ ドメインのすべてのインターフェイス（ブリッジ ポート）は、プロバイダー ネットワーク ポート（PNP）として扱われます。

	コマンドまたはアクション	目的
ステップ 8	**rewrite ingress tag push dot1ad** *vlan-id* **symmetric** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-core)# end`	Mac-in-MAC フレームのバックボーン VLAN タグを設定し、また、タグの書き換えポリシーを設定します。 **（注）** コア ブリッジ ドメインのすべての PNP で同じバックボーン VLAN を使用します。
ステップ 9	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-core)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-core)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

バックボーン送信元 MAC アドレスの設定

バックボーン送信元 MAC アドレス（B-SA）は、バックボーン ネットワークの一意のアドレスです。各 Cisco ASR 9000 シリーズ ルータは 1 つのバックボーン送信元 MAC アドレスを持ちます。B-SA が設定されていない場合、EEPROM の最も大きい MAC が PBB B-SA として使用されます。

（注） バックボーン送信元 MAC アドレスの設定は任意です。バックボーン送信元 MAC アドレスを設定しない場合、Cisco ASR 9000 シリーズ ルータは、シャーシ バックプレーン MAC プールからデフォルト バックボーン送信元 MAC アドレスを割り当てます。

バックボーン送信元 MAC アドレスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **pbb**
4. **backbone-source-mac** *mac-address*
5. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**pbb** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)# pbb`	PBB 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	**backbone-source-address** *mac-address* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pbb)# backbone-source-address 0045.1200.04`	バックボーン送信元 MAC アドレスを設定します。
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-pbb)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-pbb)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PBB エッジ ブリッジ ドメイン下での不明ユニキャスト バックボーン MAC の設定

PBB エッジ ブリッジ ドメイン下で不明ユニキャスト バックボーン MAC を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **interface** *type interface-path-id.subinterface*
6. **pbb edge i-sid** service-id **core-bridge** *core-bridge-name*
7. **unknown-unicast-bmac** *mac-address*
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-edge`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 5	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。

	コマンドまたはアクション	目的
ステップ 6	**pbb edge i-sid** *service-id* **core-bridge** *core-bridge-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# pbb edge i-sid 1000 core-bridge pbb-core`	サービス ID および割り当てられたコア ブリッジ ドメインを指定して、ブリッジ ドメインを PBB エッジとして設定し、PBB エッジ コンフィギュレーション サブモードを開始します。 このコマンドは、指定したコア ブリッジ ドメインに PBB エッジ ブリッジ ドメインを関連付ける仮想インスタンス ポート（VIP）も作成します。 このブリッジ ドメインのすべてのインターフェイス（ブリッジ ポート）は、カスタマー ネットワーク ポート（CNP）として扱われます。
ステップ 7	**unknown-unicast-bmac** *mac-address* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# unknown-unicast-bmac 1.1.1`	不明ユニキャスト バックボーンの MAC アドレスを設定します。 **（注）** Trident ラインカードで、不明ユニキャスト BMAC を設定すると、マルチキャスト、ブロードキャスト、および不明ユニキャスト宛先 MAC アドレスを持つカスタマー トラフィックを転送するために、BMAC が使用されます。
ステップ 8	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PBB エッジ ブリッジ ドメイン下でのスタティック MAC アドレスの設定

PBB エッジ ブリッジ ドメイン下でスタティック MAC アドレスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **interface** *type interface-path-id.subinterface*
6. **interface** *type interface-path-id.subinterface*
7. **pbb edge i-sid** service-id **core-bridge** *core-bridge-name*
8. **static-mac-address** *cda-mac-address* **bmac** *bda-mac-address*
9. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-edge`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 5	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。

	コマンドまたはアクション	目的
ステップ 6	**interface** type interface-path-id.subinterface **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#interface GigabitEthernet0/5/0/1.15	ブリッジ ドメインにインターフェイスを追加し、パケットの転送と、同じブリッジ ドメイン内の他のインターフェイスからのパケットの受信を可能にします。これで、このインターフェイスは、このブリッジ ドメイン上の接続回線になります。
ステップ 7	**pbb edge i-sid** *service-id* **core-bridge** *core-bridge-name* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# pbb edge i-sid 1000 core-bridge pbb-core	サービス ID および割り当てられたコア ブリッジ ドメインを指定して、ブリッジ ドメインを PBB エッジとして設定し、PBB エッジ コンフィギュレーション サブモードを開始します。 このコマンドは、指定したコア ブリッジ ドメインに PBB エッジ ブリッジ ドメインを関連付ける仮想インスタンス ポート（VIP）も作成します。 このブリッジ ドメインのすべてのインターフェイス（ブリッジ ポート）は、カスタマー ネットワーク ポート（CNP）として扱われます。
ステップ 8	**static-mac-address** cda-*mac-address* **bmac** *bda-mac-address* **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)#static-mac-address 0033.3333.3333 bmac 0044.4444.4444	PBB エッジ サブモードで CMAC と BMAC のスタティック マッピングを設定します。
ステップ 9	**end** または **commit** **例：** RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# end または RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# commit	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

PBB VPLS の設定

PBB VPLS を設定するには、次の作業を実行します。

I-Component のアクセス疑似回線の設定

PBB エッジ ブリッジ ドメイン下でスタティック MAC アドレスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **mac withdraw state-down**
6. **exit**
7. **interface** *type interface-path-id.subinterface*
8. **interface** *type interface-path-id.subinterface*
9. **neighbor** {*A.B.C.D*} **pw-id** *value*
10. **exit**
11. **pbb edge i-sid** service-id **core-bridge** *core-bridge-name*
12. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	ブリッジ グループ コンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。

	コマンドまたはアクション	目的
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-edge`	ブリッジ ドメイン コンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 5	**mac withdraw state-down** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# mac withdraw state-down`	（任意）MAC 取り消しをイネーブルにします。
ステップ 6	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# exit`	現在のコンフィギュレーション モードを終了します。
ステップ 7	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#interface GigabitEthernet0/5/0/0.20`	一致する VLAN ID および EtherType をインターフェイスに割り当てます。
ステップ 8	**interface** type interface-path-id.subinterface **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)#interface GigabitEthernet0/5/0/1.15`	ブリッジ ドメインにインターフェイスを追加し、パケットの転送と、同じブリッジ ドメイン内の他のインターフェイスからのパケットの受信を可能にします。これで、このインターフェイスは、このブリッジ ドメイン上の接続回線になります。
ステップ 9	**neighbor** {*A.B.C.D*} **pw-id** *value* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 **（注）** *A.B.C.D* は再帰的または非再帰的プレフィクスです。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 10	**exit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# exit`	現在の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 11	**pbb edge i-sid** *service-id* **core-bridge** *core-bridge-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# pbb edge i-sid 1000 core-bridge pbb-core`	サービス ID および割り当てられたコア ブリッジ ドメインを指定して、ブリッジ ドメインを PBB エッジとして設定し、PBB エッジ コンフィギュレーション サブモードを開始します。 このブリッジ ドメインのすべてのインターフェイス（ブリッジ ポート）は、カスタマー ネットワーク ポート（CNP）として扱われます。
ステップ 12	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pbb-edge)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

B-Component のコア疑似回線の設定

PBB エッジ ブリッジ ドメイン下でスタティック MAC アドレスを設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *group-name*
4. **bridge-domain** *domain-name*
5. **vfi** {***vfi-name***}
6. **neighbor** {*A.B.C.D*} {**pw-id** *value*}
7. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**l2vpn** **例：** `RP/0/RSP0/CPU0:router(config)# l2vpn`	L2VPN コンフィギュレーション モードを開始します。
ステップ 3	**bridge group** *bridge-group-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn)#bridge group pbb`	名前付きブリッジ グループのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ グループを作成するか、既存のブリッジ グループを変更します（ブリッジ グループが存在する場合）。ブリッジ グループは、ブリッジ ドメインを整理します。
ステップ 4	**bridge-domain** *domain-name* **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg)#bridge-domain pbb-core`	名前付きブリッジ ドメインのコンフィギュレーション モードを開始します。このコマンドは、新しいブリッジ ドメインを作成するか、既存のブリッジ ドメインを変更します（ブリッジ ドメインが存在する場合）。
ステップ 5	**vfi** {*vfi-name*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# vfi PBB-core-vfi`	仮想転送インターフェイス（VFI）パラメータを設定し、L2VPN ブリッジ グループ ブリッジ ドメイン VFI コンフィギュレーション モードを開始します。 • 指定した仮想転送インターフェイス名を設定するには、*vfi-name* 引数を使用します。

	コマンドまたはアクション	目的
ステップ 6	**neighbor** {*A.B.C.D*} {**pw-id** *value*} **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# neighbor 10.1.1.2 pw-id 1000` `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)#`	アクセス疑似回線ポートをブリッジ ドメインに追加するか、または疑似回線を仮想転送インターフェイス（VFI）に追加します。 • 相互接続ピアの IP アドレスを指定するには、*A.B.C.D* 引数を使用します。 **（注）** *A.B.C.D* は再帰的または非再帰的プレフィクスです。 • 疑似回線 ID および ID 値を設定するには、**pw-id** キーワードを使用します。範囲は 1 ～ 4294967295 です。
ステップ 7	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# end` または `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-pw)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)?` `[cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

802.1ah プロバイダー バックボーン ブリッジを実装するための設定例

ここでは、次の設定例を示します。

イーサネット フロー ポイントの設定：例

次に、イーサネット フロー ポイントを設定する例を示します。

```
config
interface GigabitEthernet0/0/0/10.100 l2transport
 encapsulation dot1q 100
or
 encapsulation dot1ad 100
or
 encapsulation dot1ad 100 dot1q 101
```

PBB エッジ ブリッジ ドメインおよびサービス インスタンス ID の設定：例

次に、PBB エッジ ブリッジ ドメインを設定する例を示します。

```
config
l2vpn
  bridge group PBB
    bridge-domain PBB-EDGE
    interface GigabitEthernet0/0/0/38.100
    !
    interface GigabitEthernet0/2/0/30.150
    !
    pbb edge i-sid 1000 core-bridge PBB-CORE
    !
 !
!
```

PBB コア ブリッジ ドメインの設定：例

次に、PBB コア ブリッジ ドメインを設定する例を示します。

```
config
l2vpn
 bridge group PBB
  bridge-domain PBB-CORE
  interface G0/5/0/10.100
  !
  interface G0/2/0/20.200
  !
   pbb core
  !
 !
!
```

バックボーン VLAN タグの設定：例

次に、バックボーン VLAN タグを設定する例を示します。

```
config
l2vpn
 bridge group PBB
  bridge-domain PBB-CORE
  interface G0/5/0/10.100
  !
  interface G0/2/0/20.200
  !
   pbb core
     rewrite ingress tag push dot1ad 100 symmetric
  !
 !
!
```

バックボーン送信元 MAC アドレスの設定：例

次に、バックボーン送信元 MAC アドレスを設定する例を示します。

```
config
l2vpn
 pbb
  backbone-source-mac 0045.1200.04
 !
!
```

PBB エッジ ブリッジ ドメイン下でのスタティック マッピングおよび不明ユニキャスト MAC アドレスの設定

次に、PBB エッジ ブリッジ ドメイン下でスタティック マッピングおよび不明ユニキャスト MAC アドレスを設定する例を示します。

```
config
l2vpn
 bridge group PBB
  bridge-domain PBB-EDGE
  interface GigabitEthernet0/0/0/38.100
  !
  interface GigabitEthernet0/2/0/30.150
  !
  pbb edge i-sid 1000 core-bridge PBB-CORE
    static-mac-address 0033.3333.3333 bmac 0044.4444.4444
    unknown-unicast-bmac 0123.8888.8888
  !
 !
!
```

PBB-VPLS の設定：例

次に、PBB VPLS を設定する例を示します。

I-Component のアクセス疑似回線の設定

```
l2vpn
  bridge group PBB
    bridge-domain PBB-EDGE
      mac withdraw state-down ------ can be used with MIRP, optional
      interface GigabitEthernet0/0/0/38.100
      interface GigabitEthernet0/2/0/30.150
      neighbor 10.10.10.1 pw-id 1010 ------- configures access PW
      !
      pbb edge i-sid 1200 core-bridge PBB-CORE
   !
 !
!
```

B-Component のコア疑似回線の設定

```
l2vpn
  bridge group PBB
    bridge-domain PBB-CORE
      interface G0/5/0/10.100
      !
    vfi PBB-CORE-vfi
     neighbor 1.1.1.1 pw-id 1004 ------- configures core PW
     !
!
 !
```

MIRP Lite の設定：例

MIRP 機能はデフォルトでイネーブルです。ただし、MIRP パケットは、接続回線が機能しない場合、および次のように **mac withdraw state-down** を設定した場合に送信されます。

```
l2vpn
  bridge group PBB
    bridge-domain PBB-EDGE
      mac withdraw state-down
```

ただし、mac withdraw state-down を設定しないと、MIRP パケットは接続回線が機能しているときに送信されます。

その他の関連資料

ここでは、Cisco ASR 9000 シリーズ ルータでの 802.1ah の実装に関する参考資料を紹介します。

関連資料

関連項目	参照先
802.1ah コマンド：すべてのコマンド構文、コマンド モード、コマンド履歴、デフォルト、使用上の注意、および例	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』の「*Provider Backbone Bridge Commands*」モジュール

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

その他の関連資料

マルチ スパニングツリー プロトコルの実装

このモジュールでは、Cisco ASR 9000 シリーズ ルータでのマルチ スパニングツリー プロトコル（MST）プロトコルの概念および設定情報について説明します。マルチ スパニングツリー プロトコル（MSTP）は、ブリッジ設定のループを防ぐために使用されるスパニングツリー プロトコルです。他のタイプの STP とは異なり、MSTP は VLAN ごとにポートを選択的にブロックできます。

マルチ スパニングツリー プロトコルの実装機能の履歴

リリース	変更内容
リリース 3.7.3	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 3.9.1	バンドル機能での MSTP のサポートが追加されました。
リリース 4.0.1	PVST+ および PVSTAG 機能のサポートが追加されました。
リリース 4.1.0	MSTAG エッジ モード機能のサポートが追加されました。
リリース 4.3.0	バンドル インターフェイスに PVSTAG のサポートが追加されました。

内容

マルチ スパニングツリー プロトコルを実装するための前提条件

この前提条件は、MSTP の実装に適用されます。

このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

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

マルチ スパニングツリー プロトコルの実装に関する情報

イーサネット サービス アクセス リストを実装するには、次の概念を理解している必要があります。

- スパニングツリー プロトコルの概要
- マルチ スパニングツリー プロトコルの概要
- MSTP サポート機能
- MSTP の設定に関する制約事項
- アクセス ゲートウェイ
- マルチ VLAN 登録プロトコル

スパニングツリー プロトコルの概要

イーサネットは、ネットワークの手段とホストの相互接続に使用される、単なるリンク層テクノロジーではありません。単純な*プラグ アンド プレイ*プロビジョニングの考え方と統合されている低コストで幅広い帯域幅機能によって、特にサービス プロバイダー ネットワークのアクセスおよび集約の領域で、イーサネットはネットワークを構築するための正規の技法に変換されています。

レイヤ 2（L2）ヘッダーの TTL フィールドがなく、マルチキャスト トラフィック ネットワーク全体が推奨されるか必要とされるイーサネット ネットワークは、ループが発生する場合にブロードキャスト ストームの影響を受けやすくなります。ただし、ループは、冗長パスを提供するため、望ましい特性です。スパニングツリー プロトコル（STP）は、イーサネット ネットワーク内のループ フリー トポロジを提供するために使用され、リンク障害に対処するようにネットワーク内の冗長性を確保できます。

STP には、多くのバリアントがあります。ただし、同じ基本原則で動作します。ループを含む可能性があるネットワーク内では、ループ フリーのスパニングツリーを確保できるように（つまり、ネットワーク内の任意の 2 台のデバイス間に 1 つだけパスが存在するように）、十分な数のインターフェイスが STP によってディセーブルになります。アクティブ リンクの 1 つに影響を与えるネットワークに障害がある場合、プロトコルは、すべてのデバイスが引き続き到達可能になるように、スパニングツリーを再計算します。STP は、単一の LAN セグメントに接続されているか、複数のセグメントが含まれてループがないことを確認するために STP を使用するスイッチド LAN に接続されているかを検出できないエンド ステーションに対してトランスペアレントです。

STP プロトコルの動作

STP のすべてのバリアントは同じ方法で動作します。STP フレーム（ブリッジ プロトコル データ ユニット（BPDU）とも呼ばれます）は、STP に参加しているネットワーク デバイス間でレイヤ 2 LAN セグメントを介して定期的に交換されます。このようなネットワーク デバイスはこれらのフレームを転送しませんが、ループ フリー スパニングツリーを構築するために情報を使用します。

スパニングツリーは、最初にスパニングツリーの*ルート*（*ルート ブリッジ*と呼ばれます）であるデバイスを選択してから、そのルート ブリッジからネットワーク内のその他すべてのデバイスへのループ フリー パスを判別することで構成されます。冗長パスは、適切なポートをブロック状態に設定することで無効にされます。ブロック状態では、STP フレームを引き続き交換できますが、データ トラフィックは転送されません。ネットワーク セグメントで障害が発生し、冗長パスが存在する場合、STP プロトコルがスパニングツリー トポロジを再計算し、適切なポートのブロックを解除することによって、冗長パスをアクティブにします。

STP ネットワーク内のルート ブリッジの選択は、各デバイスの設定されたプライオリティおよび組み込みブリッジ ID によって決まります。プライオリティが最低であるか、または等しく最低のプライオリティであるが最小ブリッジ ID を持つデバイスが、ルート ブリッジとして選択されます。

一連の冗長パス内でのアクティブ パスの選択は、主にポート パス コストによって決定されます。ポート パス コストは、そのポートとルート ブリッジ間の転送コストを表します。ポートがルート ブリッジから遠いほど、コストは高くなります。コストは、（デフォルトで）メディア速度に依存する量だけ、パスのリンクごとに増加します。指定された LAN セグメントからの 2 つのパスのコストが等しい場合、選択は、接続先装置のプライオリティとブリッジ ID で決まります。また、2 つの接続が同じデバイスに対するものである場合は、接続されたポートに設定されたポート プライオリティとポート ID によって決まります。

アクティブなパスを選択すると、アクティブ トポロジの一部にならないポートはすべてブロッキング状態に移行します。

トポロジの変更

スイッチド LAN のネットワーク デバイスは、MAC 学習を実行します。つまり、受信したデータ トラフィックを使用して、その MAC アドレス宛のフレームの送信先となるインターフェイスとユニキャスト MAC アドレスを関連付けます。STP を使用すると、スパニングツリーの再計算（たとえば、ネットワーク障害後）によって、この学習した情報を無効にできます。したがってプロトコルには、古い情報を削除（フラッシュ）して、新しいトポロジに基づいた新しい情報を学習できるように、ネットワーク全体でのトポロジ変更を通知するメカニズムが含まれます。

*トポロジ変更*通知は、STP がポートをブロッキング状態から転送状態に移行するたびに送信されます。これを受信すると、受信デバイスは、通知を受け取ったポート以外のブロックされないすべてのポートで MAC 学習エントリをフラッシュして、さらにこれらのポートから独自のトポロジ変更通知を送信します。このように、古い情報がネットワーク内のすべてのデバイスから削除されるようにします。

STP のバリアント

スパニングツリー プロトコルには、多くのバリアントがあります。

- レガシー STP（STP）：元の STP プロトコルは、IEEE 802.1D-1998 で定義されていました。これはすべての VLAN で使用する単一のスパニングツリーを作成し、コンバージェンスのほとんどはタイマーベースです。
- Rapid STP（RSTP）：これは、イベントベースであるためにより高速なコンバージェンスを提供するために IEEE 802.1D-2004 で定義された機能拡張です。ただし、引き続きすべての VLAN で単一のスパニングツリーを作成します。

- マルチ STP（MSTP）：さらなる拡張機能が IEEE 802.1Q-2005 で定義されました。マルチ スパニングツリーは、同じ物理トポロジで作成できます。異なるスパニングツリーに異なる VLAN を割り当てることによって、データ トラフィックは異なる物理リンクでロードバランスできます。作成できる異なるスパニングツリーの数は、可能な VLAN の数よりもさらに小さい値に制限されます。ただし、複数の VLAN を同じスパニングツリーに割り当てることができます。MSTP 情報の交換に使用される BPDU は常にタグなしで送信されます。VLAN およびスパニングツリー インスタンス データは BPDU 内で符号化されます。
- Per-VLAN STP（PVST）：これは、マルチ スパニングツリーを作成するための代替メカニズムです。MSTP の標準化の前にシスコが開発しました。PVST を使用して、別個のスパニングツリーが VLAN ごとに作成されます。PVST+（レガシー STP に基づく）および PVRST（RSTP に基づく）の 2 つのバリアントがあります。パケット レベルのスパニングツリーの分離は、適切な VLAN タグでタグ付けされた標準の STP または RSTP BPDU を送信して行われます。
- REP（シスコ独自のリング冗長プロトコル）：これは、リングで復元力を提供するためのシスコ独自のプロトコルです。これは、MSTP ピアとの相互運用を行うために使用する MSTP 互換モードが提供されるため、完全を期すために組み込まれています。

マルチ スパニングツリー プロトコルの概要

マルチ スパニングツリー プロトコル（MSTP）は、複数および独立したスパニングツリーを同じ物理ネットワークに作成できるようにする STP バリアントです。各スパニングツリーのパラメータは、ループフリー トポロジを形成するために、ルート ブリッジとして別のネットワーク デバイスを選択するか、別のパスを選択するように、別個に設定できます。その結果、特定の物理インターフェイスを一部のスパニングツリーではブロックして、その他のツリーではブロック解除できます。

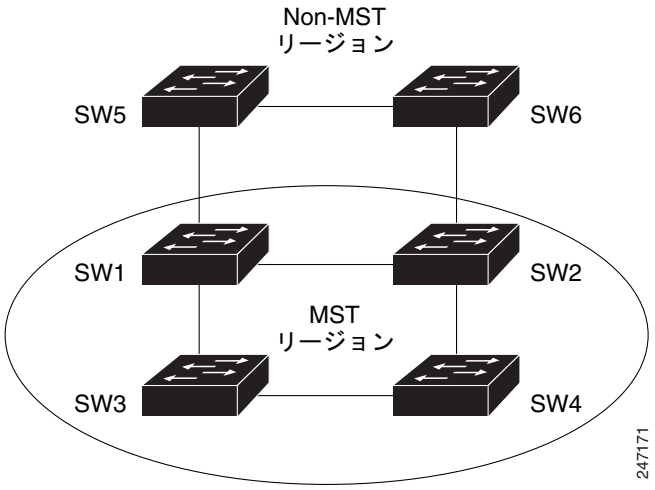
マルチ スパニングツリーを設定すると、使用中の VLAN セットをツリー間で分割できます。たとえば、VLAN 1 ～ 100 をスパニングツリー 1 に割り当てて、VLAN 101 ～ 200 をスパニングツリー 2 に割り当てて、VLAN 201 ～ 300 を VLAN 3 に割り当てることができます。各スパニングツリーには、異なるアクティブ リンクとの別のアクティブ トポロジがあるため、VLAN に基づいて、利用可能な冗長リンク間でデータ トラフィックを分割できます（ロード バランシングの実行）。

MSTP リージョン

マルチ スパニングツリーのサポートとともに、MSTP では、リージョンの概念が採用されています。リージョンは、同じ管理制御下にあるデバイス グループであり、類似した設定があります。特に、リージョン名の設定、リビジョン、スパニングツリー インスタンスへの VLAN のマッピングは、リージョン内のすべてのネットワーク デバイスで同じでなければなりません。同じリージョン内にあるかどうかを他のデバイスが確認できるように、この情報のダイジェストが、各デバイスによって送信される BPDU に含まれています。

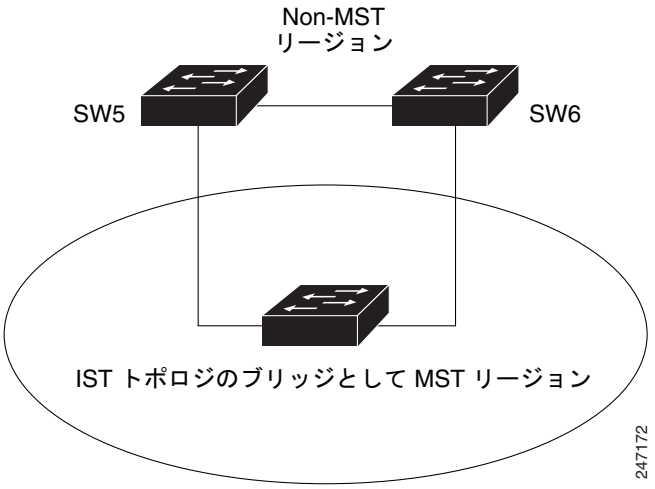
図 34 に、MSTP を実行するブリッジがレガシー STP または RSTP を実行するブリッジに接続されている場合の MST リージョンの動作を示します。この例では、スイッチ SW1、SW2、SW3、SW4 では MSTP がサポートされるのに対して、スイッチ SW5 および SW6 ではサポートされません。

図 34　非 MST リージョンとの MST の対話



この状況に対処するために、Internal Spanning Tree（IST）が使用されます。これは、常にスパニング ツリー インスタンス 0（ゼロ）です。MSTP 非認識デバイスと通信する場合、全体の MSTP リージョンは単一のスイッチとして表されます。図 35 に、この場合の論理 IST トポロジを示します。

図 35　非 MST ブリッジと対話する MST リージョンの論理トポロジ



同じメカニズムが、別のリージョンにある MSTP デバイスとの通信時に使用されます。たとえば、図 35 の SW5 は、すべてが SW1、SW2、SW3、SW4 とは別のリージョンにある多数の MSTP デバイスを表している可能性があります。

MSTP Port Fast

MSTP には、スイッチド イーサネット ネットワークのエッジでポートを処理するための *PortFast* 機能が組み込まれています。スイッチド ネットワーク（通常はホスト デバイス）へのリンクが 1 つだけあるデバイスでは、使用可能なパスが 1 しかないため、MSTP を実行する必要はありません。さらに、代替パスがないため、単一のリンクで障害が発生するか復元された場合に、トポロジの変更（およびその結果の MAC フラッシュ）が起動されることは望ましくありません。

デフォルトでは、MSTP は、BPDU を受け取らないポートを監視して、タイムアウト後に、MSTP に参加しないようにするエッジ *モード*にします。ただし、エッジ ポートを PortFast として明示的に設定することで、このプロセスを高速化（およびそれによってネットワーク全体のコンバージェンスを改善）できます。

（注） レガシー STP のシスコ実装では、PortFast はシスコ独自の拡張として実装されます。ただし、エッジ ポートと呼ばれる RSTP と MSTP 用の標準に含まれています。

MSTP ルート ガード

共有管理制御のネットワークでは、ネットワーク管理者が、ネットワーク トポロジの側面および特にルート ブリッジの場所を強化することを推奨します。デフォルトでは、より低いプライオリティまたはブリッジ ID がある場合、すべてのデバイスがスパニングツリーのルート ブリッジになることができます。ただし、ネットワークの中心の特定の場所にルート ブリッジを配置することで、より最適な転送トポロジを実現できます。

（注） 管理者は、ルート ブリッジの場所を保護するために、ルート ブリッジのプライオリティを 0 に設定できます。ただし、これによって、プライオリティが 0 で、低いブリッジ ID を持つ別のブリッジは保証されません。

ルート ガード機能は、管理者はルート ブリッジを強制的に配置できるメカニズムを提供します。ルート ガードがインターフェイスで設定されている場合、そのインターフェイスがルート ポート（つまり、ルートに到達できるポート）になるのを防ぎます。通常はルート ポートになるインターフェイス上で BPDU を介して優位情報を受信すると、代わりにバックアップ ポートまたは代替ポートになります。この場合、ブロッキング状態になり、データ トラフィックは転送されません。

ルート ブリッジ自体にはルート ポートがありません。このため、管理者は、デバイスのすべてのインターフェイス上でルート ガードを設定することでデバイスを強制的にルートにします。競合する情報を受信するインターフェイスはブロックされます。

（注） ルート ガードはレガシー STP および RSTP のシスコ実装でシスコ独自の拡張として実装されます。ただし、制限付きロールと呼ばれる MSTP 用の標準に含まれています。

MSTP のトポロジ変更の監視

特定の状況では、特定のポートで発信されたか受信したトポロジ変更を、ネットワークのその他の部分に伝播することが望ましい場合があります。これは、たとえば、ネットワークが単一の管理制御下になく、ネットワーク コアの外部にあるデバイスによるコアでの MAC アドレスのフラッシュを防ぐことが望ましいような場合です。この動作は、ポートのトポロジ変更を設定することでイネーブルにできます。

（注） トポロジ変更ガードは、MSTP 標準の *制限 TCN* と呼ばれます。

MSTP サポート機能

Cisco ASR 9000 シリーズ ルータでは、MSTP は、IEEE 802.1Q-2005 で定義されているように物理イーサネット インターフェイスおよびイーサネット バンドル インターフェイスでサポートされます。これには、レガシー STP、RSTP、および PVST の Cisco 実装にある PortFast、BackboneFast、UplinkFast、およびルート ガード機能が含まれることに注意してください。これらの機能は、標準の MSTP プロトコルに含まれるためです。Cisco ASR 9000 シリーズ ルータは、標準 802.1Q モードまたはプロバイダー エッジ（802.1ad）モードのいずれかで動作できます。プロバイダー エッジ モードでは、BPDU には別の MAC アドレスが使用され、802.1Q MAC アドレスで受信されたすべての BPDU がトランスペアレントに転送されます。

また、次の追加のシスコの機能がサポートされます。

- BPDU ガード：このシスコの機能は、エッジ ポートの設定ミスから保護します。
- Flush Containment：このシスコの機能は、トポロジを変更すると発生する不要な MAC フラッシュを防止するために役立ちます。
- 起動遅延：このシスコの機能は、トラフィックを転送する準備が完了する前に、インターフェイスがアクティブ トポロジに追加されないようにします。

（注） 802.1Q 規格で規定されているように、RSTP との相互運用がサポートされます。ただし、レガシー STP との相互運用性はサポートされません。

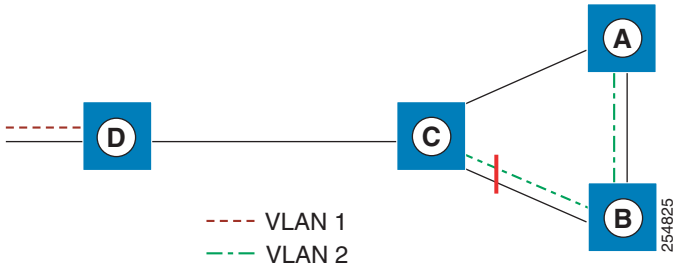
BPDU ガード

BPDU ガードは、エッジ ポートの設定ミスから保護するシスコの機能です。これは、MSTP の PortFast 機能の拡張です。PortFast がインターフェイスで設定されている場合、MSTP は、スパニングツリーの計算時に、そのインターフェイスをエッジ ポートであると見なし、考慮の対象から外します。BPDU ガードが設定されている場合、MSTP はさらに、MSTP BPDU を受信すると、errdisable を使用してインターフェイスをシャットダウンします。

Flush Containment

Flush Containment は、ネットワーク内の他の領域での非関連トポロジの変更が原因で発生する不要な MAC フラッシュを防止するために役立つシスコの機能です。これは、例で詳しく説明します。図 36 に、4 台のデバイスを含むネットワークを示します。2 つの VLAN が使用中です。VLAN 1 はデバイス D だけで使用され、VLAN 2 はデバイス A、B、および C またがっています。2 つの VLAN は同じスパニングツリー インスタンスにありますが、リンクを共有しません。

図 36　　Flush Containment



リンク AB がダウンすると、通常の動作では、C がブロックされたポートを起動し、D を含むその他すべてのインターフェイスでトポロジ変更通知を送信します。これにより、行われたトポロジ変更は VLAN 2 だけに影響を与えるにもかかわらず、VLAN 1 で MAC フラッシュが行われます。

Flush containment は、対象の MSTI で VLAN が設定されていないインターフェイスでトポロジ変更通知が送信されないようにすることで、この問題に対処します。ネットワーク例では、これは、トポロジ変更通知が C から D に送信されないこと、および行われる MAC フラッシュがネットワークの右側に制限されることを意味します。

（注） Flush containment はデフォルトでイネーブルにされますが、設定でディセーブルにできるため、IEEE 802.1Q 規格で規定されている動作が復元されます。

起動遅延

起動遅延は、インターフェイスがまだトラフィックを転送する準備が完了していない場合に、スパニングツリーの計算時に MSTP がインターフェイスを考慮しないようにするシスコの機能です。これは、データ プレーンがトラフィックを転送する準備が十分に完了する前に、そのカードのインターフェイスが *アップ*していることをシステムが宣言するため、ラインカードの最初の起動時に役立ちます。標準によると、MSTP は、*アップ*していることを宣言するとインターフェイスを考慮します。これによって、新しいインターフェイスが代わりに選択される場合に、他のインターフェイスがブロッキング状態に移行されることがあります。

起動遅延は、MSTP で設定されたインターフェイスが最初に現れたときに発生する設定可能な遅延期間を追加することで、この問題を解決します。この遅延時間が終了するまで、インターフェイスはブロッキング状態のままになり、スパニングツリーの計算時に考慮されません。

起動遅延は、MSTP ですでに設定されているインターフェイスの作成時（たとえば、カードのリロード時）だけ発生します。すでに存在しているインターフェイスが MSTP で設定されている場合は、遅延は発生しません。

MSTP の設定に関する制約事項

次の制限が、MSTP の使用時に適用されます。

- MSTP は、インターフェイス自体（L2 モードになっている場合）またはすべてのサブインターフェイスに単純なカプセル化が設定されているインターフェイスだけでイネーブルにする必要があります。これらのカプセル化の一致基準は単純であると見なされます。
  - 一重タグ付き 802.1Q フレーム
  - 二重タグ付き Q-in-Q フレーム（最も外側のタグだけが検査されます）
  - 802.1ad フレーム（MSTP がプロバイダー ブリッジ モードで動作している場合）
  - タグの範囲またはリスト（上記のいずれか）

（注） **デフォルト**および**タグなし**のカプセル化を使用するサブインターフェイスはサポートされません。

- L2 インターフェイスまたはサブインターフェイスが、複数の VLAN と一致するカプセル化を使用して設定されている場合、それらの VLAN はすべて同じスパニングツリー インスタンスにマップする必要があります。そのため、各 L2 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られたスパニングツリー インスタンスは 1 つだけ存在します。

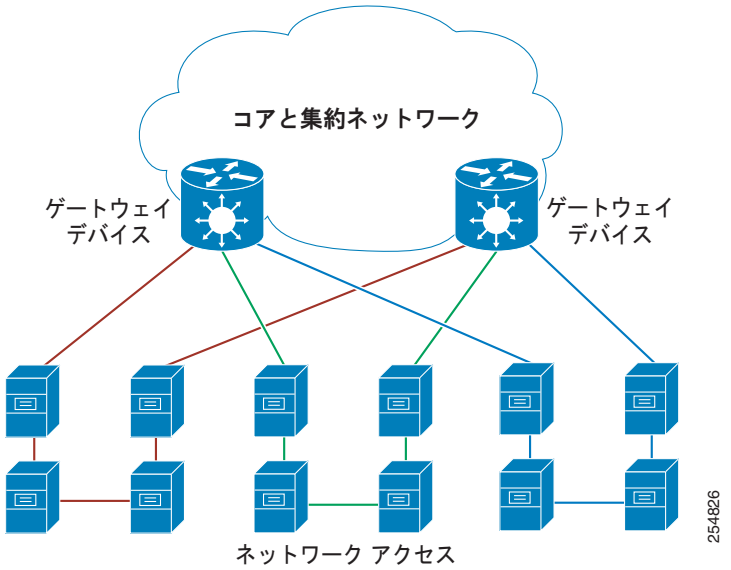
- 特定のブリッジ ドメインのすべてのインターフェイスまたはサブインターフェイスは、同じスパニングツリー インスタンスに関連付ける必要があります。
- 同じインターフェイス上の複数のサブインターフェイスは、これらのサブインターフェイスが同じスプリット ホライズン グループ内にある場合を除き、同じスパニングツリー インスタンスに関連付けることはできません。つまり、ヘアピニングはできません。
- ネットワーク全体で、L2 インターフェイスまたはサブインターフェイスを、各スパニングツリー インスタンスにマップされたすべての VLAN のすべての冗長パスで設定する必要があります。これは、ポートの STP ブロッキングが原因で接続が誤って切断されることを避けるためです。

注意 デフォルトまたはタグなしカプセル化を使用するサブインターフェイスは、MSTP ステート マシンの障害の原因となります。

アクセス ゲートウェイ

Cisco ASR 9000 シリーズ ルータに共通する 1 つの導入シナリオには、uPE アクセス デバイスのネットワークと集約ネットワークのコアの間に配置された nPE ゲートウェイ デバイスがあります。各ゲートウェイ デバイスは、図 37 に示すように、多数のアクセス ネットワークの接続を提供できます。アクセス ネットワーク（一般的にリング）には、コアまたは集約ネットワークへの冗長リンクがあるため、ネットワークがループフリーのままになるようにするには、STP のいくつかのバリアントまたは類似したプロトコルを使用する必要があります。

図 37　　コアまたは集約ネットワーク



ゲートウェイ デバイスは STP プロトコルにも参加できます。ただし、各ゲートウェイ デバイスは多くのアクセス ネットワークに接続されているため、これによって、2 つのソリューションのうちの 1 つになります。

- アクセス ネットワークをすべてカバーする単一のトポロジが維持されます。これは、1 つのアクセス ネットワークのトポロジ変更が、他のすべてのアクセス ネットワークに影響を与えることを意味するため、望ましくありません。

- ゲートウェイ デバイスは、STP プロトコルの複数のインスタンスを、アクセス ネットワークごとに 1 つずつ実行します。これは、アクセス ネットワークごとに別個のプロトコル データベースと別個のプロトコル ステート マシンが維持されることを意味します。これは、ゲートウェイ デバイスで必要なメモリと CPU リソースが原因で望ましくありません。

これらの両方のオプションには重要な欠点があることがわかります。

別の方法として、各アクセス ネットワークの *レグ*間でプロトコル BPDU をトンネリングするが、プロトコル自体には参加しないゲートウェイ デバイスがあります。これによって正確なループフリー トポロジになりますが、重要な欠点もあります。

- アクセス リングの *レグ*間に直接接続されていないため、*レグ*リンクの 1 つの障害が、他の *レグ*に接続されたアクセス デバイスですぐに検出されません。したがって、6 秒以上のトラフィック損失が発生する障害からの回復はプロトコル タイムアウトを待つ必要があります。
- ゲートウェイ デバイスはプロトコルに参加しないため、アクセス ネットワークの任意のトポロジ変更を認識できません。そのため集約ネットワークは、トポロジ変更に従って、誤った *レグ*によるアクセス ネットワーク宛のトラフィックを送信する場合があります。これにより、MAC 学習タイムアウト（デフォルトでは 5 分）の順序でトラフィック損失が発生する可能性があります。

アクセス ゲートウェイは、上記のソリューションの欠点を招くことなく、この導入シナリオに対処することを意図したシスコの機能です。

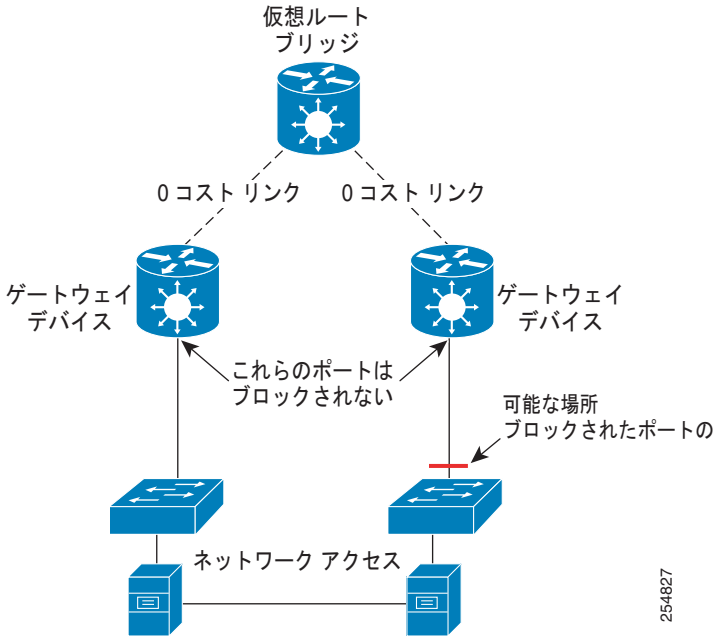
アクセス ゲートウェイの概要

アクセス ゲートウェイは次の 2 つの前提に基づいています。

- 両方のゲートウェイ デバイスが、常にコアまたは集約ネットワークへの接続を提供します。通常、これにあてはまることを確認するには、コアまたは集約ネットワーク内で使用される復元力メカニズムで十分です。ほとんどの導入では、この復元力を提供するために、コアまたは集約ネットワークで VPLS が使用されます。
- 各アクセス ネットワークのすべてのスパニングツリーで必要なルートは、ゲートウェイ デバイスの 1 つです。これは、（一般に）トラフィックの大部分がアクセス デバイスとコアまたは集約ネットワーク間に存在し、アクセス デバイス間にトラフィックがほとんど存在しない場合にあてはまります。

これらの前提では、STP トポロジには、すべてのスパニングツリーでゲートウェイ デバイスの背後に（つまり、コア側に）仮想ルート ブリッジがあり、両方のゲートウェイ デバイスに仮想ルート ブリッジへのゼロのコスト パスがあると考えることができます。この場合、ゲートウェイ デバイスをアクセス ネットワークに接続するポートは、スパニングツリー プロトコルによってブロックされませんが、常に転送状態にあります。図 38 で、これについて説明します。

図 38　ネットワーク アクセス



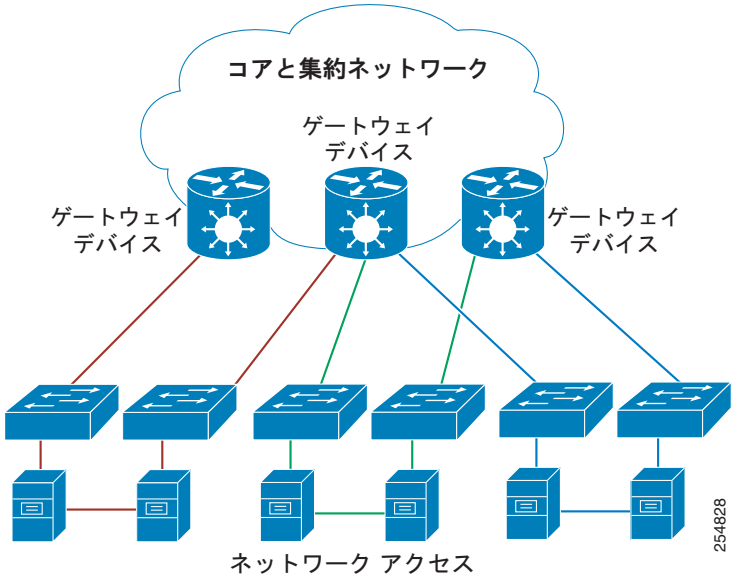
このトポロジでは、ゲートウェイ デバイスによって送信された BPDU が一定であることを確認することができます。これは、（集約またはコア ネットワークは常に接続を提供することを想定しているため）ルート ブリッジが変更されることはなく、ポートは常に転送しているという理由から、BPDU で送信される情報は変更されません。

アクセス ゲートウェイは、ゲートウェイ デバイスで完全な STP プロトコルおよび関連するステート マシンを実行する必要性をなくすことでこれを活用し、代わりに、スタティックに設定された BPDU を単にアクセス ネットワークに送信します。BPDU は、完全なプロトコルが実行されている場合に送信される同じ情報を含むように、上記の動作をシミュレート用に設定します。アクセス デバイスには、ゲートウェイ デバイスがプロトコルに完全に参加しているように表示されます。ただし、実際はゲートウェイ デバイスは、スタティック BPDU を送信しているだけであるため、ゲートウェイ デバイスではほとんどメモリまたは CPU リソースは必要なく、多くのネットワーク アクセスを同時にサポートできます。

たいていゲートウェイ デバイスは、アクセス ネットワークから受信した BPDU を無視できます。ただし、1 つの例外は、アクセス ネットワークがトポロジ変更を信号通知する場合です。ゲートウェイ デバイスは、たとえばコアまたは集約ネットワークが VPLS を使用した場合に LDP MAC 取り消しをトリガーすることで、これを適切に実行できます。

多くの場合、ゲートウェイ デバイス間の直接接続は必要ありません。ゲートウェイ デバイスは、アクセス リンク上で設定された BPDU をスタティックに送信するため、（それぞれの設定が一致している限り）それぞれ個別に設定できます。またこれは、図 39 に示すように、さまざまなアクセス ネットワークがゲートウェイ デバイスの異なるペアを使用できることを示します。

図 39　　ネットワーク アクセス



（注）　図 39 はアクセス リングを示していますが、一般にアクセス ネットワーク トポロジ、またはゲートウェイ デバイスへのリンクの数または場所に制限はありません。

アクセス ゲートウェイによって、次の障害の場合にループ フリー接続が確保されます

- アクセス ネットワークでのリンクの障害。
- アクセス ネットワークとゲートウェイ デバイス間のリンクの障害。
- アクセス デバイスの障害。
- ゲートウェイ デバイスの障害。

トポロジ変更の伝播

アクセス ネットワーク トポロジの変更を処理するために、2 台のゲートウェイ デバイスが互いに BPDU を交換する必要があるような場合があります。アクセス ネットワークの障害の結果、前にブロックされたポートが転送に移行されるトポロジ変更が発生する場合、アクセス デバイスは、残りのネットワークに変更について通知して、必要な MAC 学習フラッシュをトリガーするように、そのポートにトポロジ変更通知を送信します。通常、トポロジ変更通知は、アクセス ゲートウェイの場合はルート ブリッジ方向に送信されます。これは、いずれかのゲートウェイ デバイスに送信されることを意味します。

上記のように、これによって、ゲートウェイ デバイス自体が必要な処理を実行します。ただし、障害によりアクセス ネットワークが分割された場合は、残りのアクセス ネットワーク（つまり、他のゲートウェイ デバイスに接続されている部分）にトポロジ変更通知を伝播する必要が生じる場合もあります。これを行うには、ゲートウェイ デバイス間の接続を確認して、各ゲートウェイ デバイスが、受信するトポロジ変更通知をアクセス ネットワークから他のデバイスに伝播できるようにします。ゲートウェイ デバイスはトポロジ変更を示す BPDU を他のゲートウェイ デバイスから受信すると、スタティック BPDU でこれを信号通知します（つまり、アクセス ネットワークに向けて送信します）。

トポロジ変更の伝播は、次の 2 つの条件が満たされた場合だけ必要です。

- アクセス ネットワークに 3 台以上のアクセス デバイスが含まれる場合。デバイスが 3 台未満の場合、すべてのデバイスが、発生する可能性があるすべての障害を検出する必要があります。

- アクセス デバイスは、コアまたは集約ネットワーク間だけなく、相互にトラフィックを送信します。すべてのトラフィックがコアまたは集約ネットワーク間のトラフィックである場合、すべてのアクセス デバイスが、すでに正しい方向でトラフィックを送信しているか、トラフィックの発信元アクセス デバイスからのトポロジ変更を学習する必要があります。

プリエンプション遅延

アクセス ゲートウェイを支える前提の 1 つは、ゲートウェイ デバイスはコアまたは集約ネットワークへの接続を提供するために常時使用可能なことです。ただし、この前提が成り立たない可能性のある状況が 1 つあり、これは起動時に発生します。起動時に、トラフィックをコアまたは集約ネットワークに正常に転送できることを意味する、必要なすべてのシグナリングとコンバージェンスが完了する前に、アクセス側インターフェイスが使用可能になるような場合です。インターフェイスが起動するとすぐにアクセス ゲートウェイが BPDU の送信を開始するため、これによって、ゲートウェイ デバイスで受信する準備が完了する前に、アクセス デバイスがゲートウェイ デバイスにトラフィックを送信する可能性があります。この問題を回避するには、プリエンプション遅延機能が使用されます。

プリエンプション遅延機能によって、インターフェイスが起動した後、通常の値に戻るまでの期間にアクセス ゲートウェイは不良 BPDU を送信します。他のゲートウェイ デバイスもダウンしている場合を除き、アクセス ネットワークがすべてのトラフィックを他のゲートウェイ デバイスに送信するようにこれらの不良 BPDU を設定できます。他のゲートウェイ デバイスが使用できない場合、部分的にだけ使用可能でも、トラフィックを完全にドロップするのではなく、このデバイスに送信することを推奨します。したがって、BPDU をまったく送信しないのではなく、不良 BPDU はプリエンプション遅延時間中に送信されます。

サポートされるアクセス ゲートウェイ プロトコル

アクセス ゲートウェイは、次のプロトコルがアクセス ネットワークで使用されるときに Cisco ASR 9000 シリーズ ルータでサポートされます。

表 3　　プロトコル

ネットワーク プロトコルへのアクセス	アクセス ゲートウェイ バリアント
MSTP	MST アクセス ゲートウェイ（MSTAG）
REP	REP アクセス ゲートウェイ（REPAG）[1]
PVST+	PVST+ アクセス ゲートウェイ（PVSTAG）[2]
PVRST	PVRST アクセス ゲートウェイ（PVRSTAG）[3]

1. REP アクセス ゲートウェイは、ゲートウェイ デバイスに接続されているアクセス デバイス インターフェイスが REP MSTP 互換モードで設定されている場合にサポートされます。
2. トポロジ変更の伝播は PVSTAG ではサポートされません。
3. トポロジ変更の伝播は PVRSTAG ではサポートされません。

MSTAG エッジ モード

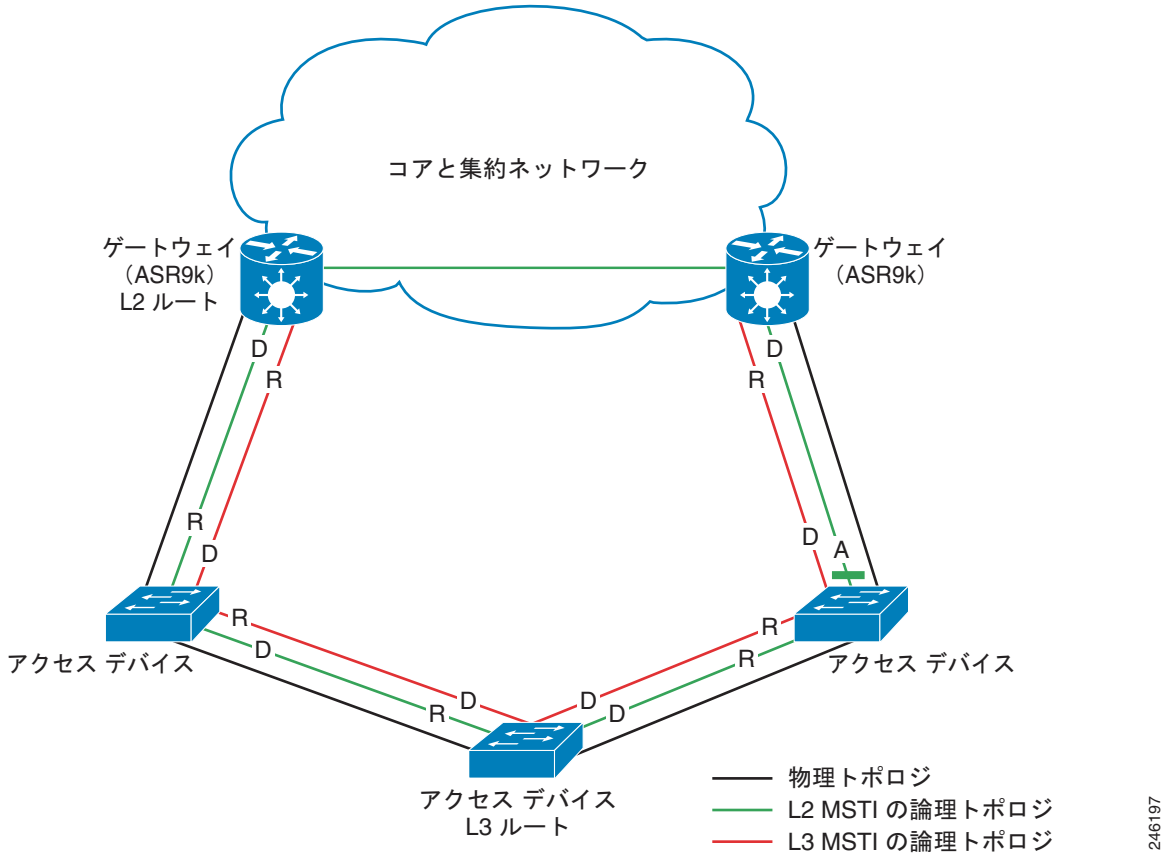
マルチ スパニングツリー インスタンス（MSTI）ごとに、各アクセス デバイスにコアまたは集約ネットワークへのパスが 1 つあることを確認するために、レイヤ 2（L2）環境ではアクセス ゲートウェイが使用されます。コアまたは集約ネットワークは、2 台のゲートウェイ デバイス間の L2（イーサネット）接続を提供します。そのため、障害がない場合、各 MSTI のアクセス ネットワークにブロックされたポートが少なくとも 1 つ必要です。アクセス リングの場合は、アクセス リングにブロック ポートが 1 つ必要です。各 MSTI では、これは通常、ゲートウェイ デバイスの 1 つに接続されているアップリンク ポートの 1 つです。これは、ゲートウェイ デバイスが最善のマルチスパニングツリー プロトコ

ル（MSTP）ルート ノードへの最適なパスを持つように MSTAG を設定することによって行われます。したがって、アクセス デバイスは、ルートに到達するために常にゲートウェイ デバイスを使用し、ゲートウェイ デバイスのポートは常に指定された転送状態になります。

混合レイヤ 2 レイヤ 3 環境では、特定の VLAN のレイヤ 2 サービスおよび他の VLAN のレイヤ 3（L3）サービスを提供するために、L2 アクセス ネットワークが使用されます。アクセス ネットワークでは、L2 サービスと L3 サービスに異なる MSTI が使用されます。L2 VLAN の場合、コアまたは集約ネットワークはゲートウェイ デバイス間の L2 接続を提供します。ただし、L3 サービスでは、ゲートウェイ デバイスは L2 ネットワークを終了し、L3 ルーティングを実行します。通常、エンド ホストが適切なゲートウェイにルーティングできるように、HSRP や VRRP などの L3 冗長性メカニズムが使用されます。

このシナリオでは、単独で MSTAG を使用しても、L3 MSTI の望ましい動作は達成されません。これは、実際にはループがなくても、アクセス ネットワークのいずれかのポートがブロックされるためです。（これは、L3 VLAN のゲートウェイ デバイス間に L2 接続がないためです）。実際は、ゲートウェイ デバイスが L3 VLAN の L2 ネットワークを終了するため、望ましい動作とは、アクセス ネットワークに MSTP ルートが存在し、ゲートウェイ デバイスが単一接続を持つリーフ ノードとして表示されることです。これを行うには、MSTAG 設定を逆にします。つまり、最低品質のルートに最低品質のパスをアドバタイズするようにゲートウェイ デバイスを設定します。これは、アクセス デバイスはルートとしていずれかのアクセス デバイスを強制的に選択させるため、ポートはブロックされません。この場合、ゲートウェイ デバイスのポートは常にルート転送状態になります。MSTAG エッジ モード機能は、ゲートウェイ デバイスによってアドバタイズされるロールを指定からルートに変更することで、このシナリオをイネーブルにします。図 40 に、このシナリオを示します。

図 40　MSTAG エッジ モードのシナリオ



D：指定ポート（転送）
R：ルート ポート（転送）
A：代替ポート（ブロック）

正常な MSTAG と L2 MSTI では、トポロジ変更通知が 1 台のゲートウェイ デバイスから他のゲスト デバイスに伝播され、アクセス ネットワークに再アドバタイズされます。ただし、L3 MSTI の場合、これは望ましくありません。アクセス ネットワークに L3 MSTI のブロックがないため、トポロジ変更通知が永続的にループする可能性があります。この状況を回避するためには、MSTAG エッジ モードで、ゲートウェイ デバイスのトポロジ変更通知の処理を完全にディセーブルにします。

バンドル インターフェイスの PVSTAG

Per-VLAN スパニングツリー アクセス ゲートウェイ（PVSTAG）のサポートは、バンドル インターフェイスとともに物理インターフェイスにも拡張されています。そのため、PVST アクセス ネットワークをサポートするカスタマーの数の増加に対応できるようになりました。

物理インターフェイスでは、ブリッジ プロトコル データ ユニット（BDPU）は、インターフェイスをホストするラインカードから送信されます。ただし、バンドル インターフェイス BPDU はルート プロセッサ（RP）から送信されます。RP フェールオーバーが発生しても、バンドル インターフェイスでオーバーフローしたデータ トラフィックは影響を受けません。そのため、BPDU は、フェールオーバーが完了して新しいアクティブ RP に引き継がれるまで送信されません。遅延がある場合、ピア デバイスは BPDU 情報をタイムアウトします。これにより、イーサネット ネットワークの中断の原因になる転送ループが生じる可能性があります。そのため、RP フェールオーバーが発生した場合、ピア デバイスが BPDU 情報をタイムアウトしないようにすることが重要です。

マルチ VLAN 登録プロトコル

マルチ VLAN 登録プロトコルは IEEE 802.1ak で定義され、マルチキャストおよびブロードキャスト フレームの伝播を最適化するために MSTP ベースのネットワークで使用されます。

デフォルトでは、マルチキャストおよびブロードキャスト フレームは、スパニングツリーおよびネットワークに接続されている各エッジ（ホスト）デバイスに従って、ネットワーク内の各ポイントに伝播されます。ただし、特定の VLAN では、特定のホストだけがその VLAN のトラフィックの受信に関与する場合があります。さらに、特定のネットワーク デバイスまたは場合によってはネットワークのセグメント全体に、その VLAN のトラフィックの受信に関与する接続済みのホストがないようなことがあります。この場合、その VLAN のトラフィックを、関係のないデバイスに伝播することで、最適化が可能です。MVRP は、各ホストおよびデバイスが、接続されたピアに関与する VLAN を示すことができる、必要なプロトコル シグナリングを提供します。

MVRP がイネーブルにされたデバイスは、次の 2 つのモードで動作します。

- スタティック モード：このモードでは、デバイスは、スタティックに設定された一連の VLAN への関与を宣言する MVRP メッセージを開始します。プロトコルが、MSTP トポロジに対してまだダイナミックであることに注意してください。これは、スタティックな VLAN のセットです。
- ダイナミック モード：このモードでは、デバイスは、異なるポートで受信する MVRP メッセージを処理し、関与する VLAN のセットを決定するためにダイナミックに集約します。これは、このセットへの関与を宣言する MVRP メッセージを送信します。ダイナミック モードでは、またデバイスは受信 MVRP メッセージを使用して、接続デバイスが関与を示した VLAN だけでトラフィックが送信されるように、各ポートから送信されるトラフィックをプルーニングします。

Cisco ASR 9000 シリーズ ルータでは、スタティック モードでの動作がサポートされます。これは、MVRP-lite と呼ばれます。

マルチ スパニングツリー プロトコルの実装方法

この項では、次の手順について説明します。

- MSTP の設定
- MSTAG または REPAG の設定
- PVSTAG または PVRSTAG の設定
- MVRP-lite の設定

MSTP の設定

ここでは、MSTP を設定する手順を説明します。

- MSTP のイネーブル化
- MSTP パラメータの設定
- MSTP の確認

（注） ここでは、データのスイッチングを設定する方法については説明しません。詳細については、「*マルチ ポイント レイヤ 2 サービスの実装*」を参照してください。

MSTP のイネーブル化

デフォルトでは、STP はすべてのインターフェイス上でディセーブルです。MSTP は、各物理またはイーサネット バンドル インターフェイスの設定によって明示的にイネーブルにする必要があります。MSTP がインターフェイス上で設定されると、そのインターフェイスのサブインターフェイスはすべて自動的に MSTP イネーブルになります。

MSTP パラメータの設定

MSTP 標準は、多数の設定可能なパラメータを定義します。次にグローバル パラメータを示します。

- リージョン名およびリビジョン
- 起動遅延
- Forward Delay
- 最大経過時間またはホップ
- 転送保留カウント
- プロバイダー ブリッジ モード
- Flush Containment
- VLAN ID（スパニングツリー インスタンスごと）
- ブリッジ プライオリティ（スパニングツリー インスタンスごと）

次に、インターフェイスごとのパラメータを示します。

- 外部ポート パス コスト
- Hello Time
- Link Type

- PortFast および BPDU ガード
- ルート ガードおよびトポロジ変更ガード
- ポート プライオリティ（スパニングツリー インスタンスごと）
- 内部ポート パス コスト（スパニングツリー インスタンスごと）

インターフェイス単位の設定は、MST コンフィギュレーション サブモード内のインターフェイス サブモードで行われます。

（注） 次の項にリストされている設定手順では、設定可能なパラメータがすべて表示されます。ただし、通常、そのほとんどではデフォルト値を保持できます。

手順の概要

1. **configure**
2. **spanning-tree mst** *protocol instance identifier*
3. **bringup delay for** *interval* {**minutes | seconds**}
4. **flush containment disable**
5. **name** *name*
6. **revision** *revision*-number
7. **forward-delay** *seconds*
8. **maximum {age** *seconds* | **hops** *hops*}
9. **transmit hold-count** *count*
10. **provider-bridge**
11. **instance** *id*
12. **priority** *priority*
13. **vlan-id** *vlan-range* [,*vlan-range*][,*vlan-range*][,*vlan-range*]
14. **interface {Bundle-Ether | GigabitEthernet | TenGigE | FastEthernet}** *instance*
15. **instance** *id* **port-priority** *priority*
16. **instance** *id* **cost** *cost*
17. **external-cost** *cost*
18. **link-type** {**point-to-point | multipoint}**
19. **hello-time** *seconds*
20. **portfast** [**bpdu-guard]**
21. **guard root**
22. **guard topology-change**
23. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** RP/0/RSP0/CPU0:router# **config** Thu Jun 4 07:50:02.660 PST RP/0/RSP0/CPU0:router(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**spanning-tree mst** *protocol instance identifier* **例：** RP/0/RSP0/CPU0:router(config)# **spanning-tree mst a** RP/0/RSP0/CPU0:router(config-mstp)#	MSTP コンフィギュレーション サブモードを開始します。
ステップ 3	**bringup delay for** *interval* {**minutes** | **seconds**} **例：** RP/0/RSP0/CPU0:router(config-mstp)# **bringup delay for 10 minutes**	起動を遅らせる時間間隔を設定します。
ステップ 4	**flush containment disable** **例：** RP/0/RSP0/CPU0:router(config-mstp)# **flush containment disable**	Flush Containment をディセーブルにします。 このコマンドは、状態に関係なく、すべてのインスタンスの MAC フラッシュを実行します。
ステップ 5	**name** *name* **例：** RP/0/RSP0/CPU0:router(config-mstp)# **name m1**	MSTP 領域の名前を設定します。 デフォルト値は、IEEE Std 802 で指定する 16 進数表記を使用してテキスト文字列としてフォーマットされたスイッチの MAC アドレスです。
ステップ 6	**revision** *revision*-number **例：** RP/0/RSP0/CPU0:router(config-mstp)# **revision 10**	MSTP 領域のリビジョン レベルを設定します。 指定できる値は 0 ～ 65535 です。
ステップ 7	**forward-delay** *seconds* **例：** RP/0/RSP0/CPU0:router(config-mstp)# **forward-delay 20**	ブリッジの転送遅延パラメータを設定します。 ブリッジ転送遅延時間に使用できる秒値は、4 ～ 30 です。
ステップ 8	**maximum {age** *seconds* | **hops** *hops*} **例：** RP/0/RSP0/CPU0:router(config-mstp)# **max age 40** RP/0/RSP0/CPU0:router(config-mstp)# **max hops 30**	ブリッジの最大経過時間および最大ホップ パフォーマンス パラメータを設定します。 ブリッジの最大経過時間に使用できる秒値は、6 ～ 40 です。 ブリッジの最大ホップ数に使用できる秒値は、6 ～ 40 です。

	コマンドまたはアクション	目的
ステップ 9	**transmit hold-count** *count* **例：** RP/0/RSP0/CPU0:router(config-mstp)# **transmit hold-count 8**	伝送保留カウントのパフォーマンス パラメータを設定します。 指定できる値は 1 ～ 10 です。
ステップ 10	**provider-bridge** **例：** RP/0/RSP0/CPU0:router(config-mstp)# **provider-bridge**	プロトコルの現在のインスタンスを 802.1ad モードにします。
ステップ 11	**instance** *id* **例：** RP/0/RSP0/CPU0:router(config-mstp)# **instance 101** RP/0/RSP0/CPU0:router(config-mstp-inst)#	MSTI コンフィギュレーション サブモードを開始します。 MSTI ID に使用できる値は、0 ～ 4094 です。
ステップ 12	**priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-mstp-inst)# **priority 8192**	現在の MSTI のブリッジ プライオリティを設定します。 指定できる値は、0 ～ 61440（4096 の倍数）です。
ステップ 13	**vlan-id** *vlan-range* [,*vlan-range*][,*vlan-range*][,*vlan-range*] **例：** RP/0/RSP0/CPU0:router(config-mstp-inst)# **vlan-id 2-1005**	現在の MSTI と一連の VLAN ID を関連付けます。 VLAN のリストの範囲は、a-b、c、d、e-f、g などです。 **（注）** 各 MSTI に対してステップ 11 ～ 13 を繰り返します。
ステップ 14	**interface {Bundle-Ether | GigabitEthernet | TenGigE | FastEthernet}** *instance* **例：** RP/0/RSP0/CPU0:router(config-mstp)# **interface FastEthernet 0/0/0/1** RP/0/RSP0/CPU0:router(config-mstp-if)#	MSTP インターフェイス コンフィギュレーション サブモードを開始し、特定のポートの STP をイネーブルにします。 ラック、スロット、インスタンス、またはポート形式でインターフェイスを転送します。
ステップ 15	**instance** *id* **port-priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **instance 101 port-priority 160**	MSTI にポート プライオリティのパフォーマンス パラメータを設定します。 MSTI ID に使用できる値は、0 ～ 4094 です。 ポート プライオリティに使用できる値は、0 ～ 240（16 の倍数）です。
ステップ 16	**instance** *id* **cost** *cost* **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **instance 101 cost 10000**	現在のポートの特定のインスタンスに関する内部パス コストを設定します。 MSTI ID に使用できる値は、0 ～ 4094 です。 ポート コストに使用できる値は、1 ～ 200000000 です。 各インターフェイスの MSTI ごとにステップ 15 および 16 を繰り返します。

	コマンドまたはアクション	目的
ステップ 17	**external-cost** *cost* **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **external-cost 10000**	現在の外部ポート パス コストを設定します。 ポート コストに使用できる値は、1 ～ 200000000 です。
ステップ 18	**link-type** {**point-to-point | multipoint**} **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **link-type point-to-point**	ポートのリンク タイプをポイントツーポイントまたはマルチポイントに設定します。
ステップ 19	**hello-time** *seconds* **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **hello-time 1**	ポートの hello タイムを秒単位で設定します。 使用できる値は 1 および 2 です。
ステップ 20	**portfast** [**bpdu-guard**] **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **portfast** RP/0/RSP0/CPU0:router(config-mstp-if)# **portfast bpduguard**	ポート上で PortFast をイネーブルにし、任意で BPDU ガードをイネーブルにします。
ステップ 21	**guard root** **例：** RP/0/RSP0/CPU0:router(config-mstp-if)# **guard root**	ポート上で RootGuard 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 22	**guard topology-change** **例：** `RP/0/RSP0/CPU0:router(config-mstp-if)#` **`guard topology-change`**	ポート上で TopologyChangeGuard をイネーブルにします。 **（注）** インターフェイスごとにステップ 14 ～ 22 を繰り返します。
ステップ 23	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-mstp-if)# end` または `RP/0/RSP0/CPU0:router(config-mstp-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

MSTP の確認

次の show コマンドを使用して、MSTP の動作を確認できます。

- **show spanning-tree mst** *mst-name*
- **show spanning-tree mst** *mst-name* **interface** *interface-name*
- **show spanning-tree mst** *mst-name* **errors**
- **show spanning-tree mst** *mst-name* **configuration**
- **show spanning-tree mst** *mst-name* **bpdu interface** *interface-name*
- **show spanning-tree mst** *mst-name* **topology-change flushes**

MSTAG または REPAG の設定

ここでは、MSTAG を設定する手順を説明します。

- タグなしサブインターフェイスの設定
- MSTAG のイネーブル化
- MSTAG パラメータの設定
- MSTAG トポロジ変更の伝播の設定
- MSTAG の確認

（注） REPAG の設定手順は同じです。

ここでは、データのスイッチングを設定する方法については説明しません。詳細については、「*マルチポイント レイヤ 2 サービスの実装*」を参照してください。

タグなしサブインターフェイスの設定

物理またはバンドル イーサネット インターフェイスで MSTAG をイネーブルにするには、最初に **encapsulation untagged** コマンドを使用して、タグなしパケットと一致する L2 サブインターフェイスを設定する必要があります。L2 サブインターフェイスの設定に関する詳細については、「Cisco ASR 9000 シリーズ ルータ キャリア イーサネット モデル」を参照してください。

MSTAG のイネーブル化

MSTAG は、対応するタグなしサブインターフェイス上で明示的に設定することによって、物理インターフェイスまたはバンドル イーサネット インターフェイスでイネーブルにします。MSTAG はタグなしサブインターフェイスで設定されている場合、物理またはバンドル イーサネット インターフェイスと、その物理またはバンドル イーサネット サブインターフェイス上の他のすべてのサブインターフェイスで自動的にイネーブルになります。

MSTAG パラメータの設定

MSTAG パラメータは各インターフェイスで個別に設定され、MSTAG は各インターフェイスで完全に独立して動作します。（ルートを同じアクセス ネットワークに接続している場合を除き）異なるインターフェイスの MSTAG パラメータ間の対話はありません。

これらのパラメータは、インターフェイスごとに設定できます。

- リージョン名およびリビジョン
- ブリッジ ID
- ポート ID
- 外部ポート パス コスト
- Max Age
- プロバイダー ブリッジ モード
- Hello Time

次の MSTAG パラメータは、各スパニングツリー インスタンスのインターフェイスごとに設定可能です。

- VLAN ID
- ルート ブリッジ プライオリティおよび ID
- ブリッジ プライオリティ
- ポートのプライオリティ
- 内部ポート パス コスト

アクセス ネットワーク全体に一貫した動作を確保するには、設定時に次のガイドラインを使用する必要があります。

- アクセス ネットワーク内のデバイスのルート ブリッジ プライオリティおよび ID よりもよい（低い）ルート ブリッジ プライオリティおよび ID を（スパニングツリー インスタンスごとに）使用して両方のゲートウェイ デバイスを設定する必要があります。ゲートウェイ デバイスでは、ルート ブリッジ プライオリティおよび ID を 0 に設定することを推奨します。

（注） アクセス デバイスで検出された STP の矛盾を回避するには、両方のゲートウェイ デバイスで同じルート プライオリティおよび ID を設定する必要があります。

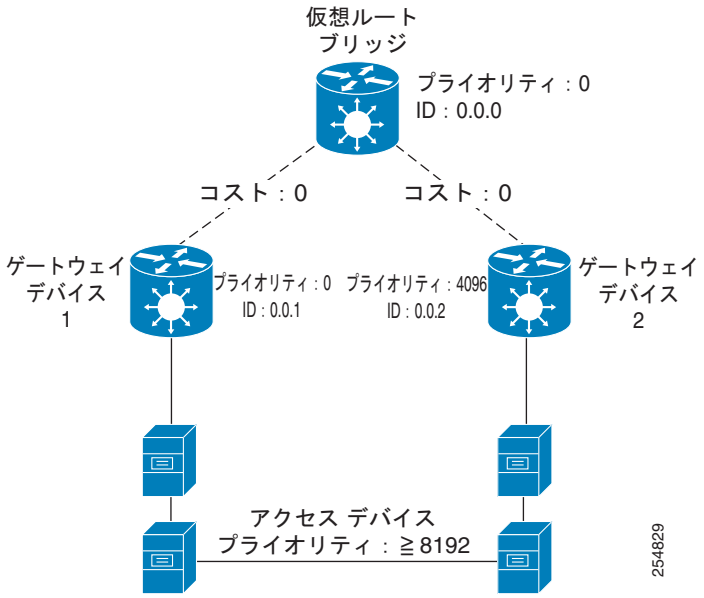
- ゲートウェイ デバイスは両方とも、ポート パス コストを 0 にして設定する必要があります。
- 各スパニングツリー インスタンスでは、ルート ブリッジ プライオリティおよび ID よりも高いが、ネットワーク内の他のデバイス（他のゲートウェイ デバイスを含む）のブリッジ プライオリティおよび ID よりも低いブリッジ プライオリティおよび ID を使用して、1 つのゲートウェイ デバイスを設定する必要があります。ブリッジ プライオリティを 0 に設定することを推奨します。
- スパニングツリー インスタンスごとに、ルート ブリッジ プライオリティおよび ID、最初のゲートウェイ デバイス ブリッジ プライオリティおよび ID よりも高いが、アクセス ネットワーク内のデバイスのブリッジ プライオリティおよび ID よりも低いブリッジ プライオリティおよび ID を使用して、2 番目のゲートウェイ デバイスを設定する必要があります。ブリッジ プライオリティを 4096 に設定することを推奨します（これは、0 よりも大きい最低許容値です）。
- ゲートウェイ デバイスよりも高いブリッジ プライオリティを使用してすべてのアクセス デバイスを設定する必要があります。8192 以上の値を使用することを推奨します。
- スパニングツリー インスタンスごとに、すべてのリンクがアップすると目的のポートがブロック状態になるように、アクセス デバイスでポート パス コストおよびその他のパラメータを設定する場合があります。

注意 MSTAG 設定のチェックはありません。設定ミスによって、アクセス デバイスの MSTP プロトコルの誤った動作が発生する可能性があります（たとえば、STP の矛盾が検出されます）。

図 41 で、上記のガイドラインについて説明します。

（注） トポロジの変更がシグナリングされると、アクセス デバイスはゲートウェイ デバイスから受信した情報を無視する場合のように、これらのガイドラインは、REPAG には適用されません。

図 41　　MSTAG のガイドライン



（注）　次の項にリストされている設定手順では、設定可能なパラメータがすべて表示されます。ただし、通常、そのほとんどではデフォルト値を保持できます。

手順の概要

1. **configure**
2. **spanning-tree mstag** *protocol instance identifier*
3. **preempt delay for** *interval* {**seconds** | **minutes** | **hours**}
4. **interface {Bundle-Ether | GigabitEthernet | TenGigE | FastEthernet}** instance.subinterface
5. **name** *name*
6. **revision** *revision*-number
7. **max age** *seconds*
8. **provider-bridge**
9. **bridge-id** *id*
10. **port-id** *id*
11. **external-cost** *cost*
12. **hello-time** *seconds*
13. **instance** *id*
14. **vlan-id** *vlan-range* [,*vlan-range*][,*vlan-range*][,*vlan-range*]
15. **priority** *priority*
16. **port-priority** *priority*
17. **cost** *cost*
18. **root-bridge** *id*

19. **root-priority** *priority*
20. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** RP/0/RSP0/CPU0:router# **configure** Thu Jun 4 07:50:02.660 PST RP/0/RSP0/CPU0:router(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**spanning-tree mstag** *protocol instance identifier* **例：** RP/0/RSP0/CPU0:router(config)# **spanning-tree mstag a** RP/0/RSP0/CPU0:router(config-mstag)#	MSTAG コンフィギュレーション サブモードを開始します。
ステップ 3	**preempt delay for** interval **{seconds | minutes | hours}** **例：** RP/0/RSP0/CPU0:router(config-mstag)# **preempt delay for 10 seconds**	プリエンプション処理を行うまでに起動 BPDU を送信する遅延時間を指定します。
ステップ 4	**interface {Bundle-Ether | GigabitEthernet | TenGigE | FastEthernet}** instance.subinterface **例：** RP/0/RSP0/CPU0:router(config-mstag)# **interface GigabitEthernet0/2/0/30.1** RP/0/RSP0/CPU0:router(config-mstag-if)#	MSTAG インターフェイス コンフィギュレーション サブモードを開始し、特定のポートの MSTAG をイネーブルにします。
ステップ 5	**name** *name* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **name** *leo*	MSTP 領域の名前を設定します。 デフォルト値は、IEEE 規格 802 で指定する 16 進数表記を使用してテキスト文字列としてフォーマットされたスイッチの MAC アドレスです。
ステップ 6	**revision** *revision*-number **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **revision** 1	MSTP 領域のリビジョン レベルを設定します。 指定できる値は 0 ～ 65535 です。

	コマンドまたはアクション	目的
ステップ 7	**max age** *seconds* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **max age 20**	ブリッジの最大経過時間のパフォーマンス パラメータを設定します。 ブリッジの最大経過時間に使用できる秒値は、6 ～ 40 です。
ステップ 8	**provider-bridge** **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **provider-bridge**	プロトコルの現在のインスタンスを 802.1ad モードにします。
ステップ 9	**bridge-id** *id* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **bridge-id 001c.0000.0011**	現在のスイッチのブリッジ ID を設定します。
ステップ 10	**port-id** *id* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **port-id 111**	現在のスイッチのポート ID を設定します。
ステップ 11	**external-cost** *cost* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **external-cost 10000**	現在の外部ポート パス コストを設定します。 ポート コストに使用できる値は、1 ～ 200000000 です。
ステップ 12	**hello-time** *seconds* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **hello-time 1**	ポートの hello タイムを秒単位で設定します。 指定できる値は 1 ～ 2 です。
ステップ 13	**instance** *id* **例：** RP/0/RSP0/CPU0:router(config-mstag-if)# **instance 1**	MSTI コンフィギュレーション サブモードを開始します。 MSTI ID に使用できる値は、0 ～ 4094 です。
ステップ 14	**edge mode** **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **edge mode**	この MSTI のアクセス ゲートウェイ エッジ モードをイネーブルにします。
ステップ 15	**vlan-id** *vlan-range* [,*vlan-range*][,*vlan-range*][,*vlan-range*] **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **vlan-id 2-1005**	現在の MSTI と一連の VLAN ID を関連付けます。 VLAN のリストの範囲は、a-b、c、d、e-f、g などです。

	コマンドまたはアクション	目的
ステップ 16	**priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **priority 4096**	現在の MSTI のブリッジ プライオリティを設定します。 指定できる値は、0 ～ 61440（4096 の倍数）です。
ステップ 17	**port-priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **port-priority 160**	MSTI にポート プライオリティのパフォーマンス パラメータを設定します。 ポート プライオリティに使用できる値は、0 ～ 240（16 の倍数）です。
ステップ 18	**cost** *cost* **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **cost 10000**	現在のポートの特定のインスタンスに関する内部パス コストを設定します。 ポート コストに使用できる値は、1 ～ 200000000 です。
ステップ 19	**root-bridge** *id* **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **root-id 001c.0000.0011**	現在のポートから送信された BPDU のルート ブリッジ ID を設定します。
ステップ 20	**root-priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# **root-priority 4096**	このポートから送信された BPDU のルート ブリッジ プライオリティを設定します。 **（注）** 各インターフェイスを設定するにはステップ 4 ～ 19 を繰り返し、インターフェイスごとに各 MSTI を設定するにはステップ 13 ～ 19 を繰り返します。
ステップ 21	**end** または **commit** **例：** RP/0/RSP0/CPU0:router(config-mstag-if-inst)# end または RP/0/RSP0/CPU0:router(config-mstag-if-inst)# commit	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

MSTAG トポロジ変更の伝播の設定

MSTAG トポロジ変更の伝播は、単に 2 台のゲートウェイ デバイスの MSTAG 対応インターフェイス間の接続を設定することによって設定されます。

1. MSTAG を上記のように設定します。使用するタグなしサブインターフェイスに留意してください。
2. ゲートウェイ デバイス間の接続を設定します。これは、MPLS 疑似回線経由で接続するか、直接物理リンクが存在する場合は VLAN サブインターフェイスになります。
3. 他のゲートウェイ デバイスへのタグなしサブインターフェイスおよびリンク（PW またはサブインターフェイス）が含まれている各ゲートウェイ デバイスでポイントツーポイント（P2P）の相互接続を設定します。

MSTAG 用に設定されたタグなしサブインターフェイスが P2P の相互接続に追加されると、MSTAG トポロジ変更の伝播が自動的にイネーブルになります。MSTAG は、トポロジの変更の検出時に信号通知するよう、その他のゲートウェイ デバイスへの相互接続によって BDPU を転送します。

MPLS 疑似回線または P2P の相互接続設定の詳細については、「ポイントツーポイント レイヤ 2 サービスの実装」を参照してください。

MSTAG の確認

次の show コマンドを使用して、MSTAG の動作を確認できます。

- **show spanning-tree mstag** *mst-name*
- **show spanning-tree mstag** *mst-name* **bpdu interface** *interface-name*
- **show spanning-tree mstag** *mst-name* **topology-change flushes**

REPAG では類似するコマンドを使用できます。

PVSTAG または PVRSTAG の設定

ここでは、PVSTAG を設定する手順を説明します。

- PVSTAG のイネーブル化
- PVSTAG パラメータの設定
- サブインターフェイスの設定
- PVSTAG の確認

PVRSTAG の設定手順は同じです。

（注） ここでは、データのスイッチングを設定する方法については説明しません。詳細については、「*マルチポイント レイヤ 2 サービスの実装*」を参照してください。

PVSTAG のイネーブル化

PVSTAG は、PVSTAG 用の物理インターフェイスおよび VLAN を明示的に設定することで、その物理インターフェイスで特定の VLAN に対してイネーブルになります。

PVSTAG パラメータの設定

次に、各 VLAN のインターフェイスごとに設定可能な PVSTAG パラメータを示します。

- ルート プライオリティおよび ID
- ルート コスト
- ブリッジ プライオリティおよび ID
- ポート プライオリティおよび ID
- Max Age
- Hello Time

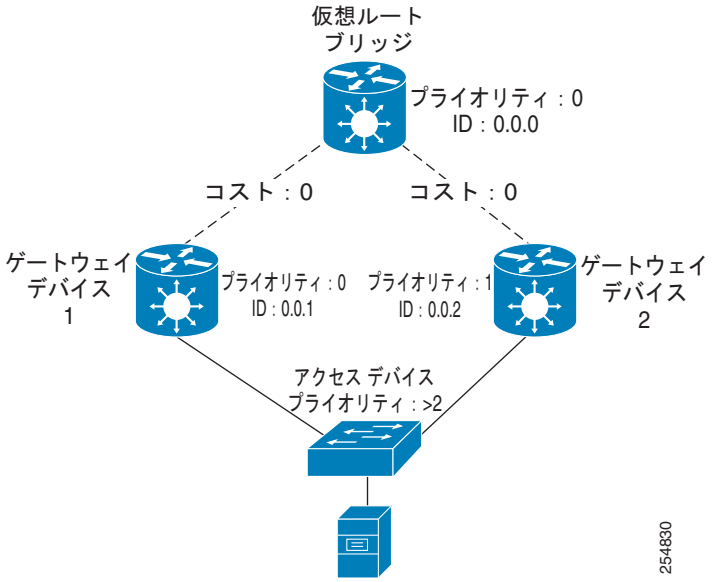
正常に動作するには、PVSTAG の設定時に次のガイドラインに従う必要があります。

- アクセス ネットワーク内のデバイスのブリッジ プライオリティおよび ID よりもよい（低い）ルート ブリッジ プライオリティおよび ID を使用して両方のゲートウェイ デバイスを設定する必要があります。ゲートウェイ デバイスでは、ルート ブリッジ プライオリティおよび ID を 0 に設定することを推奨します。
- ゲートウェイ デバイスは両方とも、ルート コストを 0 にして設定する必要があります。
- ルート ブリッジ プライオリティおよび ID よりも高いが、ネットワーク内の他のデバイス（他のゲートウェイ デバイスを含む）のブリッジ プライオリティおよび ID よりも低いブリッジ プライオリティおよび ID を使用して、1 つのゲートウェイ デバイスを設定する必要があります。ブリッジ プライオリティを 0 に設定することを推奨します。
- ルート ブリッジ プライオリティおよび ID、最初のゲートウェイ デバイス ブリッジ プライオリティおよび ID よりも高いが、アクセス ネットワーク内のデバイスのブリッジ プライオリティおよび ID よりも低いブリッジ プライオリティおよび ID を使用して、2 番目のゲートウェイ デバイスを設定する必要があります。ブリッジ プライオリティは、PVSTAG では 1、PVRSTAG では 4096 に設定することを推奨します。（PVRSTAG の場合、これは、0 よりも大きい最低許容値です）。
- ゲートウェイ デバイスよりも高いブリッジ プライオリティを使用してすべてのアクセス デバイスを設定する必要があります。PVSTAG では 2 以上の値、PVRSTAG では 8192 以上の値を使用することを推奨します。
- スパニングツリー インスタンスごとに、すべてのリンクがアップすると目的のポートがブロック状態になるように、アクセス デバイスでポート パス コストおよびその他のパラメータを設定する場合があります。

注意 PVSTAG 設定のチェックはありません。設定ミスによって、アクセス デバイスの PVST プロトコルの誤った動作が発生する可能性があります（たとえば、STP の*矛盾*が検出されます）。

図 42 で、これらのガイドラインについて説明します。

図 42　PVSTAG のガイドライン



（注）　次の項にリストされている設定手順では、設定可能なパラメータがすべて表示されます。ただし、通常、そのほとんどではデフォルト値を保持できます。

PVSTAG トポロジの制約事項

次の制約事項が PVSTAG トポロジに適用されます。

- 1 つのアクセス デバイスだけをゲートウェイ デバイスに接続できます。
- 1 つの VLAN のトポロジ変更通知は、その物理インターフェイスのすべての VLAN およびブリッジ ドメインに影響します。

手順の概要

1. **configure**
2. **spanning-tree pvstag** *protocol instance identifier*
3. **preempt delay for** *interval* {**seconds** | **minutes** | **hours**}
4. **interface** *interface-instance.subinterface*
5. **vlan** *vlan-id*
6. **root-priority** *priority*
7. **root-id** *id*
8. **root-cost** *cost*
9. **priority** *priority*
10. **bridge-id** *id*
11. **port-priority** *priority*
12. **port-id** *id*

13. **hello-time** *seconds*
14. **max age** *seconds*
15. **end**
    または
    commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** RP/0/RSP0/CPU0:router# **configure** Thu Jun 4 07:50:02.660 PST RP/0/RSP0/CPU0:router(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**spanning-tree pvstag** *protocol instance identifier* **例：** RP/0/RSP0/CPU0:router(config)# **spanning-tree pvstag a** RP/0/RSP0/CPU0:router(config-pvstag)#	PVSTAG コンフィギュレーション サブモードを開始します。
ステップ 3	**preempt delay for** interval **{seconds | minutes | hours}** **例：** RP/0/RSP0/CPU0:router(config-pvstag)# **preempt delay for 10 seconds**	プリエンプション処理を行うまでに起動 BPDU を送信する遅延時間を指定します。
ステップ 4	**interface type** *interface-path-id* or **interface Bundle-Ether** *bundle-id* **例：** RP/0/RSP0/CPU0:router(config-pvstag)# **interface GigabitEthernet0/2/0/30.1** RP/0/RSP0/CPU0:router(config-pvstag-if)# or RP/0/RSP0/CPU0:router(config-pvstag)# **interface Bundle-Ether** *100* RP/0/RSP0/CPU0:router(config-pvstag-if)#	PVSTAG インターフェイス コンフィギュレーション サブモードを開始し、特定のポートの PVSTAG をイネーブルにします。
ステップ 5	**vlan** *vlan-id* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if)# **vlan 200**	このインターフェイスで VLAN をイネーブルにして設定します。
ステップ 6	**root-priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **root-priority 4096**	このポートから送信された BPDU のルート ブリッジ プライオリティを設定します。
ステップ 7	**root-id** *id* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **root-id 0000.0000.0000**	ポートから送信された BPDU のルート ブリッジの ID を設定します。

	コマンドまたはアクション	目的
ステップ 8	**root-cost** *cost* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **root-cost 10000**	このインターフェイスから BPDU で送信するルート パス コストを設定します。
ステップ 9	**priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **priority 4096**	現在の MSTI のブリッジ プライオリティを設定します。 PVSTAG の場合、使用できる値は 0 ～ 65535 で、PVRSTAG の場合、使用できる値は 0 ～ 61440（4096 の倍数）です。
ステップ 10	**bridge-id** *id* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **bridge-id 001c.0000.0011**	現在のスイッチのブリッジ ID を設定します。
ステップ 11	**port-priority** *priority* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **port-priority 160**	MSTI にポート プライオリティのパフォーマンス パラメータを設定します。 PVSTAG の場合、ポート プライオリティに使用できる値は 0 ～ 255 で、PVRSTAG の場合、使用できる値は 0 ～ 240（16 の倍数）です。
ステップ 12	**port-id** *id* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **port-id 111**	現在のスイッチのポート ID を設定します。
ステップ 13	**hello-time** *seconds* **例：** RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# **hello-time 1**	ポートの hello タイムを秒単位で設定します。 指定できる値は 1 ～ 2 です。

	コマンドまたはアクション	目的
ステップ 14	**max age** *seconds* **例：** `RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# max age 20`	ブリッジの最大経過時間のパフォーマンス パラメータを設定します。 ブリッジの最大経過時間に使用できる秒値は、6 ～ 40 です。 **（注）** 各インターフェイスを設定するにはステップ 4 ～ 14 を繰り返し、インターフェイスごとに各 VLAN を設定するにはステップ 5 ～ 14 を繰り返します。
ステップ 15	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# end` または `RP/0/RSP0/CPU0:router(config-pvstag-if-vlan)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

サブインターフェイスの設定

インターフェイスの PVSTAG でイネーブルになっている VLAN ごとに、その VLAN のトラフィックと一致する対応するサブインターフェイスを設定する必要があります。これはデータのスイッチングと PVST BPDU の両方に使用されます。サブインターフェイスを設定するときには、次のガイドラインに従ってください。

- VLAN 1 は PVST のネイティブ VLAN として扱われます。したがって、VLAN 1 の場合は、タグなしパケット（**encapsulation untagged**）と一致するサブインターフェイスを設定する必要があります。また、VLAN 1 を明示的にタグ付けされたパケットと一致するサブインターフェイスを設定する必要が生じる場合があります（**encapsulation dot1q 1**）。
- PVST では dot1q パケットだけが許可されます。Q-in-Q および dot1ad パケットはプロトコルでサポートされていないため、これらのカプセル化で設定されたサブインターフェイスは、PVSTAG で正しく動作しません。
- VLAN の範囲と一致するサブインターフェイスは PVSTAG でサポートされます。これがデータ スイッチングのプロビジョニングで望ましい場合を除き、VLAN ごとに個別のサブインターフェイスを設定する必要はありません。
- PVSTAG は次をサポートしていません。
  - L2 モードで設定された物理インターフェイス
  - デフォルトのカプセル化（**encapsulation default**）で設定されているサブインターフェイス

– VLAN（**encapsulation dot1q any**）と一致するように設定されたサブインターフェイス

L2 サブインターフェイスの設定の詳細については、「ポイントツーポイント レイヤ 2 サービスの実装」を参照してください。

PVSTAG の確認

次の **show** コマンドを使用して、PVSTAG または PVRSTAG の動作を確認できます。

- **show spanning-tree pvstag** *mst-name*
- **show spanning-tree pvstag** *mst-name*

特に、これらのコマンドは各 VLAN に使用するサブインターフェイスを表示します。

MVRP-lite の設定

ここでは、MVRP-lite を設定する手順を説明します。

- MVRP-lite のイネーブル化
- MVRP-lite パラメータの設定
- MVRP-lite の確認

MVRP-lite のイネーブル化

MVRP ライトが設定されている場合、MSTP がイネーブルであるすべてのインターフェイスで自動的にイネーブルになります。MSTP は、MVRP をイネーブルにする前に設定する必要があります。MSTP の設定については、「MSTP の設定」（P.399）を参照してください。

MVRP-lite パラメータの設定

次に、設定可能な MVRP-lite パラメータを示します。

- 定期的な送信
- Join 時間
- Leave 時間
- Leave-all 時間

手順の概要

1. **configure**
2. **spanning-tree mst** *protocol instance name*
3. **mvrp static**
4. **periodic transmit** [**interval** *seconds*]
5. **join-time** *milliseconds*
6. **leave-time** *seconds*
7. **leaveall-time** *seconds*
8. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** RP/0/RSP0/CPU0:router# **configure** Thu Jun 4 07:50:02.660 PST RP/0/RSP0/CPU0:router(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**spanning-tree mst** *protocol instance identifier* **例：** RP/0/RSP0/CPU0:router(config)# **spanning-tree mst a** RP/0/RSP0/CPU0:router(config-mstp)#	MSTP コンフィギュレーション サブモードを開始します。
ステップ 3	**mvrp static** **例：** RP/0/RSP0/CPU0:router(config-mstp)# **mvrp static**	この MSTP プロトコル インスタンスを実行するように MVRP を設定します。
ステップ 4	**periodic transmit** [**interval** *seconds*] **例：** RP/0/RSP0/CPU0:router(config-mvrp)# **periodic transmit**	すべてのアクティブ ポートで定期的なマルチ VLAN 登録プロトコル データ ユニット（MVRPDU）を送信します。
ステップ 5	**join-time** *milliseconds* **例：** RP/0/RSP0/CPU0:router(config-mvrp)# **hello-time 1**	すべてのアクティブ ポートの Join 時間を設定します。
ステップ 6	**leave-time** *seconds* **例：** RP/0/RSP0/CPU0:router(config-mvrp)# **leave-time 20**	すべてのアクティブ ポート Leave 時間を設定します。

	コマンドまたはアクション	目的
ステップ 7	**leaveall-time** *seconds* **例：** RP/0/RSP0/CPU0:router(config-mvrp)# **leaveall-time 20**	権限をすべてのアクティブ ポートの Leave all 時間を設定します。
ステップ 8	**end** または **commit** **例：** RP/0/RSP0/CPU0:router(config-mvrp)# end または RP/0/RSP0/CPU0:router(config-mvrp)# commit	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

MVRP-lite の確認

次の show コマンドを使用して、MVRP-lite の動作を確認できます。

- **show ethernet mvrp mad**
- **show ethernet mvrp status**
- **show ethernet mvrp statistics**

MSTP の実装の設定例

ここでは、次の設定例を示します。

- MSTP の設定：例
- MSTAG の設定：例
- PVSTAG の設定：例
- MVRP-Lite の設定：例

MSTP の設定：例

次に、MSTP が単一のインターフェイスでイネーブルになっている単一スパニングツリー インスタンスの MSTP 設定例を示します。

```
config
spanning-tree mst example
    name m1
    revision 10
    forward-delay 20
    maximum hops 40
    maximum age 40
    transmit hold-count 8
    provider-bridge
    bringup delay for 60 seconds
    flush containment disable
    instance 101
        vlans-id 101-110
        priority 8192
    !
    interface GigabitEthernet0/0/0/0
        hello-time 1
        external-cost 10000
        link-type point-to-point
        portfast
        guard root
        guard topology-change
        instance 101 cost 10000
        instance 101 port-priority 160
    !
!
```

次に、スパニングツリー プロトコルの状態の概要を生成する **show spanning-tree mst** コマンドの出力例を示します。

```
# show spanning-tree mst example
Role:  ROOT=Root, DSGN=Designated, ALT=Alternate, BKP=Backup, MSTR=Master
State: FWD=Forwarding, LRN=Learning, BLK=Blocked, DLY=Bringup Delayed

Operating in dot1q mode


MSTI 0 (CIST):

  VLANS Mapped: 1-9,11-4094

  CIST Root  Priority    4096
             Address     6262.6262.6262
             This bridge is the CIST root
```

```
            Ext Cost    0

  Root ID   Priority    4096
            Address     6262.6262.6262
            This bridge is the root
            Int Cost    0
            Max Age 20 sec, Forward Delay 15 sec


  Bridge ID Priority    4096 (priority 4096 sys-id-ext 0)
            Address     6262.6262.6262
            Max Age 20 sec, Forward Delay 15 sec
            Max Hops 20, Transmit Hold count  6
Interface   Port ID           Role State Designated           Port ID
            Pri.Nbr Cost                 Bridge ID            Pri.Nbr
----------- ------- --------- ---- ----- -------------------- -------
Gi0/0/0/0   128.1   20000     DSGN FWD   4096  6262.6262.6262 128.1
Gi0/0/0/1   128.2   20000     DSGN FWD   4096  6262.6262.6262 128.2
Gi0/0/0/2   128.3   20000     DSGN FWD   4096  6262.6262.6262 128.3
Gi0/0/0/3   128.4   20000     ---- BLK   ----- -------------- -------


MSTI 1:

  VLANS Mapped: 10

  Root ID   Priority    4096
            Address     6161.6161.6161
            Int Cost    20000
            Max Age 20 sec, Forward Delay 15 sec


  Bridge ID Priority    32768 (priority 32768 sys-id-ext 0)
            Address     6262.6262.6262
            Max Age 20 sec, Forward Delay 15 sec
            Max Hops 20, Transmit Hold count  6


Interface   Port ID           Role State Designated           Port ID
            Pri.Nbr Cost                 Bridge ID            Pri.Nbr
----------- ------- --------- ---- ----- -------------------- -------
Gi0/0/0/0   128.1   20000     ROOT FWD   4096  6161.6161.6161 128.1
Gi0/0/0/1   128.2   20000     ALT  BLK   4096  6161.6161.6161 128.2
Gi0/0/0/2   128.3   20000     DSGN FWD   32768 6262.6262.6262 128.3
Gi0/0/0/3   128.4   20000     ---- BLK   ----- -------------- -------

================================================================================
```

show spanning-tree mst の出力*例*では、最初の行は、MSTP が dot1q またはプロバイダー ブリッジ モードで動作しているかどうかを示し、この情報の後に各 MSTI の詳細が表示されます。

各 MSTI について、次の情報が表示されます。

- MSTI の VLAN のリスト。
- CIST の場合、CIST ルートのプライオリティおよびブリッジ ID、および CIST ルートに到達するための外部パス コスト。またこの出力は、このブリッジが CIST ルートであるかどうかを示します。
- この MSTI のルート ブリッジのプライオリティおよびブリッジ ID、およびルートに到達するための内部パス コスト。またこの出力は、このブリッジが MSTI のルートであるかどうかを示します。
- MSTI のルート ブリッジから受信した最大経過時間および転送遅延時間。

- この MSTI のこのブリッジのプライオリティおよびブリッジ ID。
- このブリッジの最大経過時間、転送遅延、最大ホップ、および転送保留カウント（すべての MSTI で同じです）。
- MSTP 対応インターフェイスのリスト。各インターフェイスについて、次の情報が表示されます。
  - インターフェイス名。
  - この MSTI のこのインターフェイスのポート プライオリティおよびポート ID。
  - この MSTI のこのインターフェイスのポート コスト。
  - 現在のポートの役割。

    DSGN：指定：これは、この MSTI のこの LAN 上の指定ポートです。

    ROOT：ルート：この MSTI のブリッジのルート ポートです。

    ALT：代替：これは、この MSTI の代替ポートです。

    BKP：バックアップ：これは、この MSTI のバックアップ ポートです。

    MSTR：マスター：これは、CIST のルート ポートまたは代替ポートである境界ポートです。

    インターフェイスがダウンしているか、起動遅延タイマーが実行されていて、ロールがまだ割り当てられていません。
  - 現在のポート状態。

    BLK：ポートはブロックされています。

    LRN：ポートを学習中です。

    FWD：ポートは転送中です。

    DLY：起動遅延タイマーが実行中です。
  - ポートが境界ポートであり、CIST はなく、ポートが指定されていない場合は、境界ポートだけが表示され、残りの情報は表示されません。
  - ポートがアップしていないか、起動遅延タイマーが動作している場合、情報は残りのフィールドに表示されません。それ以外の場合は、インターフェイスが接続されている LAN の指定ブリッジのブリッジ プライオリティおよびブリッジ ID が表示され、その後に LAN 上の指定ポートのポート プライオリティおよびポート ID が表示されます。ポートの役割が指定されていない場合、このブリッジまたはポートの情報が表示されます。

次に、上述したように、標準コマンドよりもインターフェイス ステートに関する詳細な情報を生成する、**show spanning-tree mst** コマンドの出力例を示します。

```
# show spanning-tree mst a interface GigabitEthernet0/1/2/1
GigabitEthernet0/1/2/1
Cost: 20000
link-type: point-to-point
hello-time 1
Portfast: no
BPDU Guard: no
Guard root: no
Guard topology change: no
BPDUs sent 492, received 3
```

```
MST 3:
    Edge port:
    Boundary : internal
    Designated forwarding
    Vlans mapped to MST 3: 1-2,4-2999,4000-4094
    Port info port id 128.193 cost 200000
    Designated root address 0050.3e66.d000 priority 8193 cost 20004
    Designated bridge address 0002.172c.f400 priority 49152 port id 128.193
    Timers: message expires in 0 sec, forward delay 0, forward transitions 1
    Transitions to reach this state: 12
```

出力には、すべての MSTI に適用されるインターフェイスに関するインターフェイス情報が表示されます。

- コスト
- リンク タイプ
- hello-time
- portfast（BPDU ガードがイネーブルかどうかなど）
- ガードのルート
- ガードのトポロジ変更
- 送受信された BPDU

また、各 MSTI に固有の情報が含まれます。

- ポート ID、プライオリティ、コスト
- ルートからの BPDU 情報（ブリッジ ID、コスト、プライオリティ）
- このポートで送信される BPDU 情報（ブリッジ ID、コスト、プライオリティ）
- この状態に達するまでの状態遷移
- トポロジは、この状態になるように変更されます。
- この MSTI の Flush containment ステータス。

次に、MSTP 用に設定されているが、MSTP が動作していないインターフェイスに関する情報を生成する、**show spanning-tree mst errors** コマンドの出力例を示します。これは主に、存在しないインターフェイスに関する情報を表示します。

```
# show spanning-tree mst a errors
Interface               Error
------------------------------
GigabitEthernet1/2/3/4   Interface does not exist.
```

次に、MSTI マッピング テーブルに VLAN ID を表示する、**show spanning-tree mst configuration** の出力例を示します。また、送信された BPDU に含まれる設定ダイジェストを表示します。これは、同じ MSTP リージョン内の他のブリッジから受信したダイジェストと一致する必要があります。

```
# show spanning-tree mst a configuration
Name          leo
Revision      2702
Config Digest 9D-14-5C-26-7D-BE-9F-B5-D8-93-44-1B-E3-BA-08-CE
Instance  Vlans mapped
--------  -------------------------------
0         1-9,11-19,21-29,31-39,41-4094
1         10,20,30,40
------------------------------------------
```

次に、特定のローカル インターフェイスで出力および受信される BPDU の詳細を生成する、**show spanning-tree mst bpdu interface** の出力例を示します。

（注） 共有 LAN 上で動作する MSTP の場合は、複数の受信パケットを保存できます。

```
# show spanning-tree mst a bpdu interface GigabitEthernet0/1/2/2 direction transmit
MSTI 0 (CIST):
    Root ID : 0004.9b78.0800
    Path Cost : 83
    Bridge ID : 0004.9b78.0800
    Port ID : 12
    Hello Time : 2
    ...
```

次に、各インターフェイスの MSTI ごとに発生したトポロジ変更の詳細を表示する、**show spanning-tree mst topology-change flushes** の出力例を示します。

```
# show spanning-tree mst M topology-change flushes instance$
MSTI 1:

Interface     Last TC               Reason                           Count
------------  --------------------  -------------------------------- -----
Te0/0/0/1     04:16:05 Mar 16 2010  Role change: DSGN to ----           10
#
#
# show spanning-tree mst M topology-change flushes instance$
MSTI 0 (CIST):

Interface     Last TC               Reason                           Count
------------  --------------------  -------------------------------- -----
Te0/0/0/1     04:16:05 Mar 16 2010  Role change: DSGN to ----           10
#
```

MSTAG の設定：例

次に、単一のインターフェイスでの単一スパニングツリー インスタンスの MSTAG 設定例を示します。

```
config
interface GigabitEthernet0/0/0/0.1 l2transport
    encapsulation untagged
!
spanning-tree mstag example
    preempt delay for 60 seconds
    interface GigabitEthernet0/0/0/0.1
        name m1
        revision 10
        external-cost 0
        bridge-id 0.0.1
        port-id 1
        maximum age 40
        provider-bridge
        hello-time 1
        instance 101
            edge-mode
            vlans-id 101-110
            root-priority 0
            root-id 0.0.0
```

```
            cost 0
            priority 0
            port-priority 0
        !
    !
!
```

次に、MSTAG トポロジ変更の伝搬の追加設定例を示します。

```
l2vpn
    xconnect group example
        p2p mstag-example
            interface GigabitEthernet0/0/0/0.1
            neighbor 123.123.123.1 pw-id 100
        !
    !
!
```

次に、**show spanning-tree mstag** の出力例を示します。

```
# show spanning-tree mstag A
GigabitEthernet0/0/0/1
  Preempt delay is disabled.
  Name:            6161:6161:6161
  Revision:        0
  Max Age:         20
  Provider Bridge: no
  Bridge ID:       6161.6161.6161
  Port ID:         1
  External Cost:   0
  Hello Time:      2
  Active:          no
  BPDUs sent:      0
    MSTI 0 (CIST):
    VLAN IDs:         1-9,32-39,41-4094
    Role:             Designated
    Bridge Priority:  32768
    Port Priority:    128
    Cost:             0
    Root Bridge:      6161.6161.6161
    Root Priority:    32768
    Topology Changes: 123
  MSTI 2
    VLAN IDs:         10-31
    Role:             Designated
    Bridge Priority:  32768
    Port Priority:    128
    Cost:             0
    Root Bridge:      6161.6161.6161
    Root Priority:    32768
    Topology Changes: 123
  MSTI 10
    VLAN IDs:         40
    Role:             Root (Edge mode)
    Bridge Priority:  32768
    Port Priority:    128
    Cost:             200000000
    Root Bridge:      6161.6161.6161
    Root Priority:    61440
    Topology Changes: 0
```

次に、特定のローカル インターフェイスで出力および受信される BPDU の詳細を生成する、**show spanning-tree mstag bpdu interface** の出力例を示します。

```
RP/0/RSP0/CPU0:router#show spanning-tree mstag foo bpdu interface GigabitEthernet 0/0/0/0
Transmitted:
  MSTI 0 (CIST):
ProtocolIdentifier: 0
ProtocolVersionIdentifier: 3
BPDUType: 2
CISTFlags: Top Change Ack  0
           Agreement       1
           Forwarding      1
           Learning        1
           Role            3
           Proposal        0
           Topology Change 0
CISTRootIdentifier: priority 8, MSTI 0, address 6969.6969.6969
CISTExternalPathCost: 0
CISTRegionalRootIdentifier: priority 8, MSTI 0, address 6969.6969.6969
CISTPortIdentifierPriority: 8
CISTPortIdentifierId: 1
MessageAge: 0
MaxAge: 20
HelloTime: 2
ForwardDelay: 15
Version1Length: 0
Version3Length: 80
FormatSelector: 0
Name: 6969:6969:6969
Revision: 0
MD5Digest: ac36177f 50283cd4 b83821d8 ab26de62
CISTInternalRootPathCost: 0
CISTBridgeIdentifier: priority 8, MSTI 0, address 6969.6969.6969
CISTRemainingHops: 20
  MSTI 1:
MSTIFlags: Master          0
           Agreement       1
           Forwarding      1
           Learning        1
           Role            3
           Proposal        0
           Topology Change 0
MSTIRegionalRootIdentifier: priority 8, MSTI 1, address 6969.6969.6969
MSTIInternalRootPathCost: 0
MSTIBridgePriority: 1
MSTIPortPriority: 8
MSTIRemainingHops: 20
```

次に、インターフェイスごとに発生したトポロジ変更の詳細を表示する、**show spanning-tree mstag topology-change flushes** の出力例を示します。

```
#show spanning-tree mstag b topology-change flushes

MSTAG Protocol Instance b

Interface     Last TC              Reason                            Count
------------  -------------------  --------------------------------- -----
Gi0/0/0/1     18:03:24 2009-07-14  Gi0/0/0/1.10 egress TCN           65535
Gi0/0/0/2     21:05:04 2009-07-15  Gi0/0/0/2.1234567890 ingress TCN      2
```

PVSTAG の設定：例

次に、単一のインターフェイスでの単一 VLAN の PVSTAG 設定例を示します。

```
config
spanning-tree pvstag example
    preempt delay for 60 seconds
    interface GigabitEthernet0/0/0/0
        vlan 10
            root-priority 0
            root-id 0.0.0
            root-cost 0
            priority 0
            bridge-id 0.0.1
            port-priority 0
            port-id 1
            max age 40
            hello-time 1
        !
    !
!
```

次に、**show spanning-tree pvstag** の出力例を示します。

```
# show spanning-tree pvstag interface GigabitEthernet0/0/0/1
GigabitEthernet0/0/0/1
  VLAN 10
    Preempt delay is disabled.
    Sub-interface:    GigabitEthernet0/0/0/1.20 (Up)
    Max Age:          20
    Root Priority:    0
    Root Bridge:      0000.0000.0000
    Cost:             0
    Bridge Priority:  32768
    Bridge ID:        6161.6161.6161
    Port Priority:    128
    Port ID:          1
    Hello Time:       2
    Active:           no
    BPDUs sent:       0
    Topology Changes: 123
   VLAN 20
```

MVRP-Lite の設定：例

次に、MVRP-lite の設定例を示します。

```
config
spanning-tree mst example
    mvrp static
        periodic transmit
        join-time 200
        leave-time 30
        leaveall-time 10
    !
!
```

次に、**show ethernet mvrp mad** の出力例を示します。

```
RP/0/RSP0/CPU0:router# show ethernet mvrp mad interface GigabitEthernet 0/1/0/1
GigabitEthernet0/1/0/1
  Participant Type: Full; Point-to-Point: Yes
  Admin Control: Applicant Normal; Registrar Normal

  LeaveAll Passive (next in 5.92s); periodic disabled
  Leave in 25.70s; Join not running
  Last peer 0293.6926.9585; failed registrations: 0

VID   Applicant              Registrar
----  ---------------------  ---------
   1  Very Anxious Observer  Leaving
 283  Quiet Passive          Empty
```

次に、**show ethernet mvrp status** の出力例を示します。

```
RP/0/RSP0/CPU0:router# show ethernet mvrp status interface GigabitEthernet 0/1/0/1
GigabitEthernet0/1/0/1
  Statically declared:  1-512,768,980-1034
  Dynamically declared: 2048-3084
  Registered:           1-512
```

次に、**show ethernet mvrp statistics** の出力例を示します。

```
RP/0/RSP0/CPU0:router# show ethernet mvrp statistics interface GigabitEthernet 0/1/0/1
GigabitEthernet0/1/0/1
  MVRPDUs TX:    1245
  MVRPDUs RX:       7
  Dropped TX:       0
  Dropped RX:      42
  Invalid RX:      12
```

その他の関連資料

ここでは、Cisco ASR 9000 シリーズ ルータでのマルチ スパニングツリー プロトコル（MSTP）の実装に関する参考資料を紹介します。

関連資料

関連項目	参照先
マルチ スパニングツリー プロトコル コマンド：コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト設定、使用に関する注意事項、および例	『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』の「*Multiple Spanning Tree Protocol Commands*」

標準

標準	タイトル
IEEE 802.1Q-2005	『IEEE Standard for Local and Metropolitan Area Networks - Virtual Bridged Local Area Networks』

MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

レイヤ 2 アクセス リストの実装

イーサネット サービス アクセス コントロール リスト（ACL）は、レイヤ 2 ネットワーク トラフィック プロファイルを集合的に定義する 1 つ以上のアクセス コントロール エントリ（ACE）で構成されます。このプロファイルは、Cisco IOS XR ソフトウェア機能によって参照できます。各イーサネット サービス ACL には、送信元および宛先アドレス、サービス クラス（CoS）、または VLAN ID などの基準に基づいたアクション要素（許可または拒否）が含まれます。

このモジュールでは、Cisco ASR 9000 シリーズ アグリゲーション サービス ルータでのイーサネット サービス アクセス リストの実装に必要なタスクについて説明します。

（注） このモジュールに記載されているイーサネット サービス アクセス リスト コマンドの詳細については、マニュアル『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Command Reference*』の「*Ethernet Services (Layer 2) Access List Commands on Cisco ASR 9000 Series Routers*」を参照してください。この章で使用される他のコマンドの説明については、コマンド リファレンスのマスター索引を参照するか、またはオンラインで検索してください。

Cisco ASR 9000 シリーズ ルータでのイーサネット サービス アクセス リスト実装の機能履歴

リリース	変更内容
リリース 3.7.2	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。

内容

レイヤ 2 アクセス リスト実装の前提条件

この前提条件は、アクセス リストおよびプレフィクス リストの実装に適用されます。

このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。

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

レイヤ 2 アクセス リストの実装に関する情報

イーサネット サービス アクセス リストを実装するには、次の概念を理解している必要があります。

- 「イーサネット サービス アクセス リスト機能のハイライト」（P.434）
- 「イーサネット サービス アクセス リストの目的」（P.434）
- 「イーサネット サービス アクセス リストの機能」（P.434）
- 「イーサネット サービス アクセス リスト エントリのシーケンス番号」（P.436）

イーサネット サービス アクセス リスト機能のハイライト

イーサネット サービス アクセス リストには、次の機能のハイライトがあります。

- 特定のシーケンス番号を使用してアクセス リストのカウンタをクリアする機能。
- 別のアクセス リストに既存のアクセス リストの内容をコピーする機能。
- ユーザがシーケンス番号を permit または deny ステートメントに追加し、そのようなステートメントのシーケンスの再設定、追加、または名前付きアクセス リストからの削除を行うことができるようにします。
- パケットを転送するためにインターフェイスでパケット フィルタリングを実行します。
- イーサネット サービス ACL は、インターフェイス、VLAN サブインターフェイス、バンドル イーサネット インターフェイス、EFP、バンドル イーサネット インターフェイスを介した EFP で適用できます。イーサネット サービス ACL のアトミック置換は、これらの物理インターフェイスでサポートされています。

イーサネット サービス アクセス リストの目的

イーサネット サービス アクセス リストは、ACL ベースの転送（ABF）を使用して、ネットワークを介して移動するパケットおよび場所を制御するパケット フィルタリングを実行します。そのような制御は、着信および発信ネットワーク トラフィックを制限し、ポート レベルでネットワークにユーザおよびデバイスのアクセスを制限するために役立ちます。

イーサネット サービス アクセス リストの機能

イーサネット サービス アクセス リストは、レイヤ 2 設定に適用される、permit および deny ステートメントで構成される順序付きリストです。アクセス リストには、参照に使用される名前があります。

アクセス リストを設定して名前を付けることは可能ですが、アクセス リストを受け取るコマンドによってアクセス リストが参照されるまで、有効にはなりません。複数のコマンドから同じアクセス リストを参照できます。アクセス リストで、ルータに到達するレイヤ 2 トラフィック、またはルータ経由で送信されるレイヤ 2 トラフィックは制御できますが、ルータが送信元のトラフィックは制御できません。

イーサネット サービス アクセス リストのプロセスおよびルール

イーサネット サービス アクセス リストの設定時は、次のプロセスとルールを使用します。

- ソフトウェアは、アクセス リストの条件に対してフィルタされる各パケットの送信元アドレスや宛先アドレスをテストします。一度に 1 つの条件（permit または deny ステートメント）がテストされます。
- パケットがアクセス リストのステートメントに一致しないと、そのパケットはリスト内の次のステートメントに対してテストされます。
- パケットとアクセス リストのステートメントが一致すると、リスト内の残りのステートメントはスキップされ、パケットは一致したステートメントに指定されたとおりに許可または拒否されます。パケットが許可されるか拒否されるかは、パケットが一致する最初のエントリによって決まります。つまり、一致すると、それ以降のエントリは考慮されません。
- アクセス リストがアドレスまたはプロトコルを拒否する場合は、ソフトウェアはパケットを廃棄します。
- 各アクセス リストの最後には暗黙の deny ステートメントがあるため、一致する条件がない場合は、パケットはドロップされます。つまり、各ステートメントに対してテストするときまでにパケットを許可または拒否しないと、パケットは拒否されます。
- アクセス リストには permit ステートメントを 1 つ以上含める必要があります。そうしないと、パケットはすべて拒否されます。
- 最初に一致が見つかった後は条件のテストが終了するため、条件の順序は重要です。同じ permit ステートメントまたは deny ステートメントでも、順序が異なる場合、ある状況では通過し、別の状況では拒否されるパケットが生じる可能性があります。
- インバウンド アクセス リストは、ルータに到達するパケットを処理します。着信パケットの処理後に、アウトバウンド インターフェイスへのルーティングが行われます。インバウンド アクセス リストが効率的なのは、フィルタリング テストで拒否されたことでパケットが廃棄される場合、ルーティング検索のオーバーヘッドが抑えられるためです。パケットがテストで許可されると、そのパケットに対してルーティングの処理が実施されます。インバウンド リストの場合、許可とは、インバウンド インターフェイスでパケットの受信後に処理が続行されることを示します。拒否とは、パケットが廃棄されることを示します。
- アウトバウンド アクセス リストの場合、パケットの処理後にルータから送信されます。着信パケットはアウトバウンド インターフェイスにルーティングされてから、アウトバウンド アクセス リストで処理されます。アウトバウンド リストの場合、許可とは、出力バッファに対して送信されることを示し、拒否とは、パケットが廃棄されることを示します。
- アクセス リストは、使用中のアクセス グループによって適用されている場合には削除できません。アクセス リストを削除するには、まずアクセス リストを参照しているアクセス グループを削除してから、アクセス リストを削除します。
- アクセス リストは、**ethernet-services access-group** コマンドを使用する前に存在している必要があります。

イーサネット サービス アクセス リストを作成する際に役立つヒント

イーサネット サービス アクセス リストの作成時は、次の点に注意してください。

- アクセス リストは、インターフェイスに適用する前に作成します。
- より具体的な参照が、より全般的な参照よりも前に出現するように、アクセス リストを構成します。

送信元アドレスと宛先アドレス

送信元 MAC アドレスと宛先 MAC アドレスの 2 つのフィールドは、アクセス リストの基礎として最も一般的なフィールドです。送信元 MAC アドレスを指定して、特定のネットワーキング デバイスまたはホストからのパケットを制御します。宛先 MAC アドレスを指定して、特定のネットワーキング デバイスまたはホストに送信されるパケットを制御します。

イーサネット サービス アクセス リスト エントリのシーケンス番号

イーサネット サービス アクセス リスト エントリにシーケンス番号を適用する機能によって、アクセス リストの変更が簡単になります。アクセス リスト エントリのシーケンス番号機能を使用すると、アクセス リスト エントリにシーケンス番号を追加して、シーケンス番号を再設定できます。新しいエントリを追加する場合、アクセス リストの目的の位置に挿入されるようにシーケンス番号を選択します。必要に応じて、アクセス リストの現在のエントリを並べ替えて、新しいエントリを挿入できる場所を作成できます。

シーケンス番号の動作

次に、シーケンス番号の動作について詳細に説明します。

- シーケンス番号のないエントリを複数適用すると、最初のエントリにシーケンス番号 10 が割り当てられ、それ以降のエントリには 10 ずつ増分したシーケンス番号が割り当てられます。最大シーケンス番号は 2147483646 です。生成したシーケンス番号がこの最大値を超えると、次のメッセージが表示されます。

  ```
  Exceeded maximum sequence number.
  ```

- シーケンス番号のないエントリを 1 つ指定すると、アクセス リストの最後のシーケンス番号に 10 を加えたシーケンス番号が割り当てられ、リストの末尾に配置されます。
- ACL エントリは、トラフィック フローにもハードウェアのパフォーマンスにも影響を及ぼすことなく追加できます。
- ルート スイッチ プロセッサ（RSP）とインターフェイス カードにあるエントリのシーケンス番号が常に同期されるよう、分散サポートが提供されます。

レイヤ 2 アクセス リストの実装方法

この項では、次の手順について説明します。

- 「レイヤ 2 アクセス リスト実装の制約事項」（P.437）
- 「イーサネット サービス アクセス リストの設定」（P.438）（任意）
- 「イーサネット サービス アクセス リストの適用」（P.439）（任意）

- 「アクセス リスト エントリの並べ替え」（P.443）（任意）

レイヤ 2 アクセス リスト実装の制約事項

次の制約事項が、イーサネット サービス アクセス リストの実装に適用されます。

- イーサネット サービス アクセス リストは、管理インターフェイスではサポートされていません。
- NetIO（ソフトウェア低速パス）は、イーサネット サービス アクセス リストではサポートされません。

イーサネット サービス アクセス リストの設定

このタスクでは、イーサネット サービス アクセス リストを設定します。

手順の概要

1. **configure**
2. **ethernet-service access-list** *name*
3. [*sequence-number*] **{permit | deny}** {*src-mac-address src-mac-mask* | **any** | **host**} [{*ethertype-number*} | **vlan** *min-vlan-ID* [*max-vlan-ID*]] [**cos** *cos-value*] [**dei**] [**inner-vlan** *min-vlan-ID* [*max-vlan-ID*]] [**inner-cos** *cos-value*] [**inner-dei**]
4. 必要に応じてステップ 3 を繰り返し、計画したシーケンス番号でステートメントを追加します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。
5. **end**
   または
   commit
6. **show access-lists ethernet-services** [*access-list-name* | **maximum** | **standby** | **summary**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ethernet-service access-list** *name* **例：** `RP/0/RSP0/CPU0:router(config)# ethernet-service access-list L2ACL2`	イーサネット サービス アクセス リスト コンフィギュレーション モードを開始し、アクセス リスト L2ACL2 を設定します。
ステップ 3	[*sequence-number*] **{permit | deny}** {*src-mac-address src-mac-mask* **| any | host**} [{*ethertype-number*} | **vlan** *min-vlan-ID* [*max-vlan-ID*]] [**cos** *cos-value*] [**dei**] [**inner-vlan** *min-vlan-ID* [*max-vlan-ID*]] [**inner-cos** *cos-value*] [**inner-dei**] **例：** `RP/0/RSP0/CPU0:router(config-es-al)# 20 permit 1.2.3 3.2.1` `or` `RP/0/RSP0/CPU0:router(config-es-al)# 30 deny any dei`	パケットの通過またはドロップを決定する許可または拒否の条件を 1 つ以上指定します。
ステップ 4	必要に応じてステップ 3 を繰り返し、計画したシーケンス番号でステートメントを追加します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。	アクセス リストは変更できます。

	コマンドまたはアクション	目的
ステップ 5	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-es-acl)# end` または `RP/0/RSP0/CPU0:router(config-es-acl)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 6	**show access-lists ethernet-services** [*access-list-name* | **maximum** | **standby** | **summary**] **例：** `RP/0/RSP0/CPU0:router# show access-lists ethernet-services L2ACL1`	（任意）指定されたイーサネット サービス アクセス リストの内容を表示します。 • デフォルトとして、すべてのイーサネット アクセス リストの内容が表示されます。

次の作業

イーサネット サービス アクセス リストの作成後に、インターフェイスに適用する必要があります。アクセス リストの適用方法の詳細については、イーサネット サービス アクセス リストの適用の項を参照してください。

イーサネット サービス アクセス リストの適用

作成したアクセス リストを機能させるには、そのアクセス リストを参照する必要があります。アクセス リストは、発信または着信インターフェイスのいずれかに適用できます。ここでは、端末回線とネットワーク インターフェイスの両方に対してこのタスクを実行するためのガイドラインを示します。

着信アクセス リストでは、パケットを受信した後で、Cisco IOS XR ソフトウェアは、アクセス リストに対してパケットの送信元 MAC アドレスを検査します。アクセス リストがアドレスを許可している場合は、パケットの処理を継続します。アクセス リストがアドレスを拒否する場合は、ソフトウェアはパケットを廃棄します。

発信アクセス リストでは、パケットを受信して制御インターフェイスにルーティングした後で、ソフトウェアは、アクセス リストに対してパケットの送信元 MAC アドレスを検査します。アクセス リストがアドレスを許可している場合は、パケットを送信します。アクセス リストがアドレスを拒否する場合は、ソフトウェアはパケットを廃棄します。

（注） 空のアクセス リスト（アクセス コントロール エレメントが含まれていない）は、インターフェイスに適用できません。

インターフェイスへのアクセスの制御

このタスクでは、アクセス リストをインターフェイスに適用して、そのインターフェイスへのアクセスを制限します。アクセス リストは、発信インターフェイスまたは着信インターフェイスに適用できます。

手順の概要

1. **configure**
2. **interface** *type instance*
3. **ethernet-service access-group** *access-list-name* {**ingress** | **egress**}
4. **end**
   または
   commit

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type instance* **例：** `RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/2/0/2`	インターフェイスを設定し、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* 引数は、インターフェイス タイプを指定します。インターフェイス タイプの詳細については、疑問符（**?**）オンライン ヘルプ機能を使用してください。 • *instance* 引数は、物理インターフェイス インスタンスまたは仮想インスタンスのいずれかを指定します。 – 物理インターフェイス インスタンスの表記方法は *rack/slot/module/port* です。値を区切るスラッシュ（/）は、表記の一部として必要です。 – 仮想インターフェイス インスタンスの数値範囲は、インターフェイス タイプによって異なります。

	コマンドまたはアクション	目的
ステップ 3	**ethernet-services access-group** *access-list-name* {**ingress** | **egress**} **例：** `RP/0/RSP0/CPU0:router(config-if)# ethernet-services access-group p-in-filter ingress` `RP/0/RSP0/CPU0:router(config-if)# ethernet-services access-group p-out-filter egress`	インターフェイスへのアクセスを制御します。 • *access-list-name* 引数を使用して、特定のイーサネット サービス アクセス リストを指定します。 • **ingress** キーワードを使用すると着信パケットをフィルタリングでき、または **egress** キーワードを使用すると発信パケットをフィルタリングできます。 この例では、GigabitEthernet 0/2/0/2 から発着信されるパケットにフィルタを適用します。

	コマンドまたはアクション	目的
ステップ 4	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router(config-if)# end` または `RP/0/RSP0/CPU0:router(config-if)# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。

イーサネット サービス アクセス リストのコピー

このタスクでは、イーサネット サービス アクセス リストをコピーします。

手順の概要

1. **copy access-list ethernet-service** *source-acl destination-acl*
2. **show access-lists ethernet-services** [*access-list-name* | **maximum** | **standby** | **summary**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**copy access-list ethernet-service** *source-acl destination-acl* **例：** `RP/0/RSP0/CPU0:router# copy access-list ethernet-service list-1 list-2`	既存のイーサネット サービス アクセス リストのコピーを作成します。 • コピー元のアクセス リストの名前を指定するには、*source-acl* 引数を使用します。 • コピー元アクセス リストの内容のコピー先を指定するには、*destination-acl* 引数を使用します。 – *destination-acl* 引数は一意な名前である必要があり、アクセス リストに *destination-acl* 引数名が存在する場合は、そのアクセス リストはコピーされません。
ステップ 2	**show access-lists ethernet-services** [*access-list-name* | **maximum** | **standby** | **summary**] **例：** `RP/0/RSP0/CPU0:router# show access-lists ethernet-services list-2`	（任意）指定されたイーサネット サービス アクセス リストの内容を表示します。たとえば、コピー先の内容を検証して、宛先アクセス リスト list-2 に送信元アクセス リスト list-1 の情報がすべて含まれていることを確認できます。

アクセス リスト エントリの並べ替え

ここでは、名前付きアクセス リストのエントリにシーケンス番号を再割り当てする例を示します。アクセス リストの並べ替えは任意です。

手順の概要

1. **resequence access-list ethernet-service** *access-list-name* [*starting-sequence-number* [*increment*]]
2. **end**
   または
   commit
3. **show access-lists ethernet-services** [*access-list-name* | **maximum** | **standby** | **summary**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**resequence access-list ethernet-service** *access-list-name* [*starting-sequence-number* [*increment*]] **例：** `RP/0/RSP0/CPU0:router# resequence access-list ethernet-service L2ACL2 20 10`	（任意）目的の開始シーケンス番号およびシーケンス番号の増分を使用して、指定されたイーサネット サービス アクセス リストを並べ替えます。 • 次の例では、L2ACL2 という名前のイーサネット サービス アクセス リストを並べ替えます。開始シーケンス番号は 20、増分は 10 です。増分値を選択しないと、デフォルトの増分値 10 が使用されます。 **（注）** 並べ替えプロセス中に終了番号が許可された最大シーケンス番号を超えることがわかった場合、設定は無効になり、拒否されます。シーケンス番号は変更されません。
ステップ 2	**end** または **commit** **例：** `RP/0/RSP0/CPU0:router# end` または `RP/0/RSP0/CPU0:router# commit`	設定変更を保存します。 • **end** コマンドを実行すると、変更をコミットするように要求されます。 `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:` – **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。 – **no** と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – **cancel** と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、**commit** コマンドを使用します。
ステップ 3	**show access-lists ethernet-services** [*access-list-name* | **maximum** | **standby** | **summary**] **例：** `RP/0/RSP0/CPU0:router# show access-lists ethernet-services L2ACL2`	（任意）指定されたイーサネット サービス アクセス リストの内容を表示します。 • 出力をレビューして、アクセス リストに最新情報が含まれていることを確認します。

レイヤ 2 アクセス リストを実装するための設定例

ここでは、次の設定例を示します。

アクセス リストのエントリの並べ替え：例

次に、アクセス リストの並べ替え例を示します。並べ替えられたアクセス リストの先頭の値は 1、増分値は 2 です。後続のエントリはユーザ指定の増分値に基づいて並べられています。範囲は 1 ～ 2147483646 です。

シーケンス番号のないエントリが入力されると、デフォルトで、アクセス リストの最後のエントリのシーケンス番号に 10 を加えたシーケンス番号が割り当てられます。

```
ethernet service access-list acl_1
10 permit 1.2.3 4.5.6
20 deny 2.3.4 5.4.3
30 permit 3.1.2 5.3.4 cos 5

resequence access-list ethernet service acl_1 10 20

show access-list ethernet-service acl1_1

ipv4 access-list acl_1
  10 permit 1.2.3 4.5.6
  30 deny 2.3.4 5.4.3
  50 permit 3.1.2 5.3.4 cos 5
```

シーケンス番号を指定したエントリの追加：例

この例では、新しいエントリをイーサネット サービス アクセス リスト acl_5 に追加します。

```
ethernet-service access-list acl_5
2 permit 1.2.3 5.4.3
5 permit 2.3.4. 6.5.4 cos 3
10 permit any dei
20 permit 6.5.4 1.3.5 VLAN vlan3

configure
    ethernet-service access-list acl_5
    15 permit 1.5.7 7.5.1
    end

ethernet-service access-list acl_5
2 permit 1.2.3 5.4.3
5 permit 2.3.4. 6.5.4 cos 3
10 permit any dei
15 permit 1.5.7 7.5.1
20 permit 6.5.4 1.3.5 VLAN vlan3
```

その他の関連資料

ここでは、Cisco ASR 9000 シリーズ ルータでのイーサネット サービス アクセス リストの実装に関する参考資料を紹介します。

関連資料

関連項目	参照先
イーサネット サービス アクセス リスト コマンド：コマンド構文、コマンド モード、コマンド履歴、デフォルト設定、使用上のガイドラインおよび例	『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Command Reference*』の「*Ethernet Services (Layer 2) Access List Commands on Cisco ASR 9000 Series Routers*」

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

その他の関連資料

システムの考慮事項

このモジュールでは、Cisco ASR 9000 シリーズ ルータ 規模の制限について説明します。

（注） **show l2vpn capability** コマンドは、ルータの規模の制限を表示します。

スケール制限

表 4 で、Cisco ASR 9000 シリーズ ルータのスケール制限について説明します。

（注） 表 4 の制限は、VFI ごとに指定されます。

表 4　スケール制限

	ポート/バンドル	ライン カード			ブリッジ ドメイン			システム
		L	B	E	L	B	E	
サブインターフェイス	該当なし	32K	64K	64K	4 K	8 K	8 K	64K
ブリッジ ドメイン	該当なし	該当なし	該当なし	該当なし	該当なし	該当なし	該当なし	8 K
疑似回線	該当なし	該当なし	該当なし	該当なし	該当なし	該当なし	該当なし	64K
LAG バンドル	該当なし	該当なし	該当なし	40	該当なし	該当なし	該当なし	128
LAG サブインターフェイス	4 K	8 K	8 K	8 K	該当なし	該当なし	該当なし	16K
学習された MAC	512 K	512 K	512 K	512 K	512 K	512 K	512 K	512 K

K = 1024

ラインカード：

L：低キュー ラインカード。例：A9K-40GE-L

B：基本ラインカード。例：A9K-40GE-B

E：拡張ラインカード。例：A9K-40GE-E

（注） スケール値に達するには、サブインターフェイスはラインカードの物理ポート間で均等に割り当てる必要があります。

イーサネット ラインカードの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Ethernet Line Card Installation Guide*』の *表 1 ～ 3* を参照してください。

その他の関連資料

ここでは、Cisco ASR 9000 シリーズ ルータでのイーサネット サービス アクセス リストの実装に関する参考資料を紹介します。

関連資料

関連項目	参照先
イーサネット サービス アクセス リスト コマンド：コマンド構文、コマンド モード、コマンド履歴、デフォルト設定、使用上のガイドラインおよび例	『*Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Command Reference*』の「*Ethernet Services (Layer 2) Access List Commands on Cisco ASR 9000 Series Routers*」

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。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/en/US/support/index.html

INDEX

AR	Cisco ASR 9000 Series Aggregation Services Router Advanced System Command Reference
HR	Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Command Reference
IR	Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Command Reference
MCR	Cisco ASR 9000 Series Aggregation Services Router Multicast Command Reference
MNR	Cisco ASR 9000 Series Aggregation Services Router System Monitoring Command Reference
MPR	Cisco ASR 9000 Series Aggregation Services Router MPLS Command Reference
QR	Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Command Reference
RR	Cisco ASR 9000 Series Aggregation Services Router Routing Command Reference
SMR	Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference
SR	Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference
LSR	Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference

A

B

D

E

F

G

I

L

M

N

P

Q

R

S

V

あ

い

え

か

き

し

す

せ

そ

た

の

ひ

ふ

ほ

ま

り

る

れ

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/