Cisco ASR 9000 シリーズ アグリゲーション サービス ルータ マルチキャスト コンフィギュレーション ガイド リリース 4.2.x

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/

Text Part Number: OL-26037-02-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 Systems, Inc. All rights reserved.

目 次

はじめに

「はじめに」には、次の項があります。

マニュアルの変更履歴

次の表に、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。

リビジョン	日付	サマリー
OL-26037-02-J	2012 年 6 月	Cisco IOS XR Release 4.2.1 の機能に合わせてドキュメントを更新し再発行しました。
OL-26037-01	2011 年 12 月	このマニュアルの初版

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、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。

第 1 章

Cisco ASR 9000 シリーズ ルータ への IGMP スヌーピングを使用したレイヤ 2 マルチキャストの実装

インターネット グループ管理プロトコル（IGMP）スヌーピングは、少なくとも 1 つの関与する受信先を持つセグメントだけにレイヤ 2 のマルチキャスト フローを制限します。 このモジュールでは、Cisco ASR 9000 シリーズ ルータ への IGMP スヌーピングの実装方法について説明します。

IGMP スヌーピングの機能の履歴

リリース	変更内容
リリース 3.7.2	この機能が導入されました。
リリース 3.9.2	次の機能に対するサポートが追加されました。 • IGMP スヌーピング グループ制限およびアクセス グループ。
リリース 4.0.0	次の機能に対するサポートが追加されました。 • マルチシャーシ リンク集約（MC-LAG）を使用するマルチキャスト冗長性。

IGMP スヌーピングの前提条件

IGMP スヌーピングを実装する前に、次の前提条件を満たす必要があります。

- ネットワークは、レイヤ 2 VPN（L2VPN）で設定する必要があります。
- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 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。

  このコマンドを使用するには、適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 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場合は、AAA 管理者に連絡してください。

IGMP スヌーピングの制約事項

- IGMP スヌーピングは、L2VPN ブリッジ ドメインだけでサポートされます。
- 明示的ホスト トラッキング（IGMPv3 スヌーピング機能）はサポートされません。
- IPv6 マルチキャスト リスナー検出（MLD）スヌーピングはサポートされません。
- IGMPv1 はサポートされていません。

IGMP スヌーピングの情報

IGMP スヌーピングの概要

基本機能の説明

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。

レイヤ 3 に設定された IGMP を使用すると、IPv4 マルチキャスト ネットワーク内のホストは関与するマルチキャスト トラフィックを通知し、ルータはレイヤ 3 ネットワーク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できます。

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 List は、ブリッジ ドメインのすべてのマルチキャスト ルータ（mrouter）ポートと、指定されたルートの 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 スヌーピングはマルチキャスト ネットワークに実装され、次の属性を持ちます。

- 基本的には、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可能性があるマルチキャスト トラフィックを削減することにより、帯域幅使用量を減らします。
- 一部のオプションの設定を使用して、1 つのブリッジ ポートのホストから受信した IGMP レポートをフィルタリングし、他のブリッジ ポートのホストへの漏洩を防止することで、ブリッジ ドメイン間のセキュリティを提供します。
- オプションの設定を使用して、IGMP メンバーシップ レポート（IGMPv2）を抑制するか、アップストリーム IP マルチキャスト ルータに対して 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軽減します。

ハイ アベイラビリティ機能

すべてのハイ アベイラビリティ機能は、IGMP スヌーピングのイネーブル化以外に追加で設定することなく、IGMP スヌーピング プロセスに適用されます。 次のハイ アベイラビリティ機能がサポートされています。

- プロセスの再起動
- RP のフェールオーバー
- ステートフル スイッチオーバー（SSO）
- ノンストップ フォワーディング（NSF）：コントロール プレーンがプロセスの再起動またはルート プロセッサ（RP）のフェールオーバー後に復元している間も、転送は引き続き影響を受けません。
- ラインカードの活性挿抜（OIR）

ブリッジ ドメインのサポート

IGMP スヌーピングは、ブリッジ ドメイン レベルで動作します。 IGMP スヌーピングがブリッジ ドメインでイネーブルの場合、スヌーピング機能は、ブリッジ ドメインに属する次のポートを含むすべてのポートに適用されます。

- ブリッジ ドメインの物理ポート。
- イーサネット フロー ポイント（EFP）：EFP には VLAN、VLAN の範囲、VLAN のリスト、またはインターフェイス ポート全体を指定できます。
- VPLS ブリッジ ドメインの疑似配線（PW）。
- イーサネット バンドル：イーサネット バンドルには、IEEE 802.3ad リンク バンドルおよび Cisco EtherChannel バンドルが含まれます。 IGMP スヌーピング アプリケーションの観点では、イーサネット バンドルは単なる EFP の 1 つです。 Cisco ASR 9000 シリーズ ルータ の転送アプリケーションは、バンドルから単一のポートをランダムに指定して、マルチキャスト トラフィックを伝送します。

マルチキャスト ルータおよびホスト ポート

IGMP スヌーピングは各ポート（EFP、PW、物理ポート、EFP バンドルなど）を次のいずれかに分類します。

- マルチキャスト ルータ ポート（mrouter ポート）：マルチキャスト対応ルータが接続されているポートです。 mrouter ポートは通常動的に検出されますが、静的に設定されている場合もあります。 マルチキャスト トラフィックは、mrouter ポートが入力ポートの場合を除き、常にすべての mrouter ポートに転送されます。
- ホスト ポート：mrouter ポートでないポートはすべてホスト ポートです。

マルチキャスト ルータ検出および静的な設定

IGMP スヌーピングは、mrouter ポートを動的に検出します。 ポートを mrouter ポートとして明示的に設定することもできます。

- 検出：IGMP スヌーピングは IGMP クエリー メッセージおよび Protocol Independent Multicast Version 2（PIMv2）のハロー メッセージをスヌーピングすることで、ブリッジ ドメインのアップストリーム mrouter ポートを識別します。 PIMv2 ハロー メッセージをスヌーピングすることで、ブリッジ ドメインの IGMP 非クエリアを識別します。
- 静的設定：ポートに適用されたプロファイルで mrouter コマンドを使用して、ポートを **mrouter** ポートとして静的に設定できます。 静的設定は、シスコ以外の機器との非互換性により動的検出ができないときに役立つ場合があります。

router-guard コマンドは、IGMP クエリーや PIM メッセージなどのマルチキャスト ルータ メッセージをフィルタリングすることによって、ポートが動的に検出された mrouter ポートになること

を防止します。 **router-guard** コマンドをポートに設定した後に、スタティック mrouter として設定することができます。 同一ポートへの router-guard コマンドおよび mrouter コマンドの設定の詳細については、ルータ ガードおよびスタティック mrouter, （18 ページ）を参照してください。

IGMP スヌーピングをイネーブルにしたブリッジ ドメイン内のマルチキャスト トラフィック処理

次の表では、IGMP スヌーピングの mrouter ポートおよびホスト ポートによるトラフィック処理の動作について説明します。 表 1：IGMPv2 クエリアのマルチキャスト トラフィック処理, （5 ページ） では、IGMPv2 クエリアのトラフィック処理について説明します。 表 2：IGMPv3 クエリアのマルチキャスト トラフィック処理, （6 ページ） は IGMPv3 クエリアの場合です。

デフォルトでは、IGMP スヌーピングは IGMPv2 および IGMPv3 をサポートしています。 ブリッジ ドメインで検出された IGMP クエリアのバージョンによって、スヌーピング プロセスの動作のバージョンが決まります。 デフォルトを変更して、IGMPv3 の最小バージョンをサポートするように IGMP スヌーピングを設定した場合、IGMP スヌーピングは IGMPv2 クエリアを無視します。

表 *1*：*IGMPv2* クエリアのマルチキャスト トラフィック処理

トラフィック タイプ	mrouter ポートで受信した場合	ホスト ポートで受信した場合
IP マルチキャストの送信元トラフィック	すべての mrouter ポートと、関与を示しているホスト ポートに転送します。	すべての mrouter ポートと、関与を示しているホスト ポートに転送します。
IGMP の一般クエリー	すべてのポートに転送します。	—
IGMP グループに固有なクエリー	他のすべての mrouter ポートに転送します。	Dropped
IGMPv2 の join	レポートを検査（スヌーピング）します。 • レポート抑制がイネーブルの場合、新しいグループに対する最初の join か、既存のグループに対する一般クエリーに続く最初の join を転送します。 • レポート抑制がディセーブルの場合、すべての mrouter ポートに転送します。	レポートを検査（スヌーピング）します。 • レポート抑制がイネーブルの場合、新しいグループに対する最初の join か、既存のグループに対する一般クエリーに続く最初の join を転送します。 • レポート抑制がディセーブルの場合、すべての mrouter ポートに転送します。

トラフィック タイプ	mrouter ポートで受信した場合	ホスト ポートで受信した場合
IGMPv3 の report	無視	無視
IGMPv2 の leave	最後のメンバ クエリー処理を呼び出します。	最後のメンバ クエリー処理を呼び出します。

表 *2* : *IGMPv3* クエリアのマルチキャスト トラフィック処理

トラフィック タイプ	mrouter ポートで受信した場合	ホスト ポートで受信した場合
IPマルチキャストの送信元トラフィック	すべての mrouter ポートと、関与を示しているホスト ポートに転送します。	すべての mrouter ポートと、関与を示しているホスト ポートに転送します。
IGMP の一般クエリー	すべてのポートに転送します。	—
IGMP グループに固有なクエリー	クエリア ポートで受信した場合は、すべてのポートにフラッディングします。	—
IGMPv2 の join	IGMPv3 IS_EX{} レポートとして処理します。	IGMPv3 IS_EX{} レポートとして処理します。
IGMPv3 の report	• プロキシ レポート機能がイネーブルの場合：状態または送信元リストが変更されると、すべての mrouter ポートで状態変更レポートを生成します。 • プロキシ レポート機能がディセーブルの場合：すべての mrouter ポートに転送します。	• プロキシ レポート機能がイネーブルの場合：状態または送信元リストが変更されると、すべての mrouter ポートで状態変更レポートを生成します。 • プロキシ レポート機能がディセーブルの場合：すべての mrouter ポートに転送します。
IGMPv2 の leave	IGMPv3 IS_IN{} レポートとして処理します。	IGMPv3 IS_IN{} レポートとして処理します。

マルチシャーシ リンク集約

マルチシャーシ リンク集約（MC-LAG）機能は、デジタル加入者線アクセス マルチプレクサ（DSLAM）が Cisco ASR 9000 シリーズ ルータ にアクセスするための単純な冗長メカニズムを提供します。 冗長性は、2 つ以上の Cisco ASR 9000 シリーズ ルータ に対してデュアルホーム接続を許容することによって実現されます。

DSLAM はデュアルホーム接続デバイス（DHD）と呼ばれ、Cisco ASR 9000 シリーズ ルータは接続ポイント（PoA）と呼ばれます。 MC-LAG は冗長グループ（RG）に割り当てられます。 特定の MC-LAG を管理する Cisco ASR 9000 シリーズ ルータ（PoA）は、この RG のメンバです。 RG には複数の MC-LAG が存在する場合があります。 これは、同一の RG が他の DSLAM と MC-LAG との接続をカバーする場合があることを示します。 したがって、RG は冗長グループ ID（RGID）によって、PoA 上で一意に識別されます。 MC-LAG は一意の冗長オブジェクト ID（ROID）によって、各 PoA で識別されます。 VLAN サブインターフェイスが MC-LAG で設定されている場合は、各 VLAN サブインターフェイスに一意の ROID が存在します。

Cisco ASR 9000 シリーズ ルータの IGMP スヌーピングでは、DSLAM へのダウンストリームまたはマルチキャスト ルータへのアップストリームを監視する MC-LAG 設定をサポートしています。

（注） アクティブおよびスタンバイ POA における MC-LAG 機能の動作設定は同一である必要があります。

リンク バンドリングの設定および使用される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 Link Bundling」の章を参照してください。

IGMP スヌーピング設定プロファイルに関する情報

ブリッジ ドメインで IGMP スヌーピングをイネーブルにするには、ブリッジ ドメインにプロファイルを対応付ける必要があります。最小設定は、空のプロファイルです。プロファイルが空の場合、IGMP スヌーピングのデフォルト設定, （11 ページ）に記載されている IGMP スヌーピングのデフォルト設定オプションおよび設定値がイネーブルになります。

ブリッジ ドメインまたはブリッジ ドメインに属するポートに、IGMP スヌーピング プロファイルを適用できます。次のガイドラインでは、ポートおよびブリッジ ドメインに適用されるプロファイル間の関係について説明します。

- ブリッジ ドメインに適用されている任意の IGMP プロファイル（空のプロファイルを含む）によって、IGMP スヌーピングがイネーブルになります。 IGMP スヌーピングをディセーブルにするには、ブリッジ ドメインからプロファイルの適用を解除します。
- プロファイルが空の場合、デフォルト設定を使用して、ブリッジ ドメインおよびブリッジに属するすべてのポートに IGMP スヌーピングが設定されます。

- ブリッジ ドメインに（ブリッジ ドメイン レベルで）適用できる IGMP スヌーピング プロファイルは常に 1 つだけです。 プロファイルはブリッジに属するポートに適用でき、ポートあたり 1 つのプロファイルが適用できます。
- ポート プロファイルは、ブリッジ ドメインにプロファイルが適用されていない場合は有効になりません。
- ポート固有の設定を有効にするには、ブリッジ ドメインで IGMP スヌーピングがイネーブルになっている必要があります。
- ブリッジ ドメインに適用されたプロファイルにポート固有の設定オプションが含まれている場合は、別のポート固有プロファイルがポートに適用されていない限り、値はそのブリッジに属する mrouter ポートおよびホスト ポートを含むすべてのポートに適用されます。
- ポートにプロファイルが対応付けられていると、IGMP スヌーピングは、ブリッジ レベルのプロファイルに存在するポート設定に関係なく、そのポートを再設定します。

プロファイルの作成

プロファイルを作成するには、グローバル コンフィギュレーション モードで **igmp snooping profile** コマンドを使用します。

プロファイルの適用と解除

ブリッジ ドメインにプロファイルを適用するには、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 モードで **igmp snooping profile** コマンドを使用します。 ポートにプロファイルを適用するには、ブリッジドメインに属するインターフェイスコンフィギュレーション モードで **igmp snooping profile** コマンドを使用します。 プロファイルの適用を解除するには、適切なコンフィギュレーション モードでこのコマンドの **no** 形式を使用します。

ブリッジ ドメインまたはポートとプロファイルの対応付けを解除しても、プロファイルはそのまま存在し、後で使用できます。 プロファイルの対応付けを解除すると、次の処理が行われます。

- ブリッジ ドメインとプロファイルの対応付けを解除すると、ブリッジ ドメインで IGMP スヌーピングが非アクティブになります。
- ポートとプロファイルの対応付けを解除すると、そのポートの IGMP スヌーピング設定値は、ブリッジ ドメイン プロファイルからインスタンス化されます。

プロファイルの変更

アクティブなプロファイルは変更を加えることはできません。 アクティブなプロファイルとは、現在対応付けられているプロファイルです。

アクティブなプロファイルを変更する必要がある場合は、すべてのブリッジまたはポートとの対応付けを解除して、変更し、もう一度対応付ける必要があります。

アクティブなプロファイルを変更するもう 1 つの方法は、必要な変更を含む新しいプロファイルを作成し、ブリッジまたはポートに適用することで既存のプロファイルを置き換える方法です。これにより、IGMP スヌーピングは無効になり、新しいプロファイルのパラメータを使用して再びアクティブになります。

アクセス コントロールの設定

アクセス コントロール設定では、アクセス グループと重み付けグループの制限を設定します。

IGMP v2/v3 メッセージ フィルタリングでのアクセス グループの役割は、マルチキャスト グループ (*,G) およびマルチキャスト送信元グループ (S,G) へのホスト メンバーシップ要求を許可または拒否することです。 この役割は、IPTV チャネル パッケージへのブラック リストおよびホワイト リスト アクセスを提供するためには必須です。

重み付けグループ制限では IGMP v2/v3 グループの数が制限され、グループ内で同時に許容されるマルチキャスト チャネルの最大数を EFP および PW 単位で設定できます。

IGMP スヌーピングのアクセス グループ

レイヤ 3 IGMP ルーティングは **igmp access-group** コマンドを使用することでアクセス グループをサポートしていますが、レイヤ 3 IGMP ルーティング アクセス グループ機能は送信元グループをサポートしていないため、サポート内容はレイヤ 2 IGMP と同じではありません。

アクセス グループは、ブリッジ ドメインまたはポートに適用する IGMP スヌーピング プロファイルで参照されている拡張 IP アクセス リストを使用して指定されます。

（注） ポートレベルのアクセス グループはブリッジ ドメインレベルのアクセス グループよりも優先されます。

access-group コマンドは、受信したメンバーシップ レポートに指定されたアクセス リスト フィルタを適用するよう IGMP スヌーピングに指示します。 デフォルトでは、アクセス リストは適用されていません。

プロファイルで参照されているアクセス リストへの変更（または IGMP スヌーピング プロファイルで参照されているアクセス リストの置換）により、受信する IGMP グループ レポートおよび既存のグループ状態はただちにフィルタリングされます。このため、変更を実行するたびに、ブリッジドメインの IGMP スヌーピング プロファイルを適用解除および再適用する必要はありません。

IGMP スヌーピング グループの重み付け

IGMP v2/v3 グループの数を制限するには、グループ内で同時に許容されるマルチキャスト チャネルの最大数が EFP および PW 単位で設定可能になっている必要があり、そのうえでグループの重み付けを設定します。

IGMP スヌーピングでは、ブリッジ ポートでのメンバーシップを設定された最大数に制限しますが、IGMPv3 送信元グループをサポートし、さまざまな重み付けを個別グループまたは送信元グループに割り当てられるように機能が拡張されます。 これにより、たとえば、IPTV プロバイダー

は必要に応じて、標準画質および高解像度の IPTV ストリームを特定の加入者に関連付けることができます。

この機能は、ポートで送信される実際のマルチキャストの帯域幅を制限しません。 ただし、ポートがメンバとなる可能性がある IGMP グループと送信元グループの数を制限します。 加入者のメンバーシップ要求を適切なマルチキャスト フローに設定するのは、IPTV オペレータの責任です。

IGMP スヌーピング プロファイル コンフィギュレーション モードに属している **group policy** コマンドは、指定されたルート ポリシーを使用して新しい <*,G> または <S,G> メンバーシップ要求により追加される重みを決定するように、IGMP スヌーピングに指示します。 デフォルトは、グループの重みが設定されていない動作になります。

group limit コマンドは、ポートのグループの上限を指定します。 新しいグループまたは送信元グループによって追加される重みがこの制限を超える場合、このグループは許容されません。 （グループ ポリシーを設定せずに）グループの上限を設定した場合、<S/*,G> グループ状態にはデフォルトの重みである 1 が適用されます。

（注） デフォルトでは、各グループまたは送信元グループは、グループの上限に 1 の重みを追加します。 group policy コマンドを使用して、さまざまな重みをグループまたは送信元グループに割り当てることができます。

グループ上限ポリシーの設定は、次の条件に基づいています。

- <*,G> および <S,G> メンバーシップのグループ重み値は、BD またはポートに適用されている IGMP スヌーピング プロファイルに含まれているルート ポリシーに設定されています。
- ポート レベルの重みポリシーは、グループ制限とルート ポリシーが設定されている場合には、ブリッジ ドメイン レベルのポリシーよりも優先されます。
- ポリシーが設定されていない場合、各グループの重みは均等にカウントされ、1 になります。
- ポリシーが設定されている場合、一致するすべてのグループの重みは 1 になり、一致しないグループの重みは 0 になります。

IGMP スヌーピングのデフォルト設定

表 *3* : *IGMP* スヌーピングのデフォルト設定値

スコープ	機能	デフォルト値
ブリッジ ドメイン	IGMP スヌーピング	イネーブル化する IGMP プロファイルはブリッジ ドメインに適用されるまで、ブリッジ ドメインではディセーブルです。
	内部クエリア	未設定
	last-member-query-count	2
	last-member-query-interval	1000 ミリ秒
	minimum-version	2（IGMPv2 と IGMPv3 をサポート）
	querier query-interval	60（秒） （注） これは、非標準デフォルト値です。
	report-suppression	イネーブル（IGMPv2 のレポート抑制機能と、IGMPv3 のプロキシ レポート機能をイネーブルにします）
	querier robustness-variable	2
	ルータ アラート チェック	イネーブル
	tcn query solicit	ディセーブル
	tcn flood	イネーブル
	ttl-check	イネーブル
	unsolicited-report-timer	1000 ミリ秒

スコープ	機能	デフォルト値
ポート	immediate-leave	ディセーブル
	mrouter	スタティック mrouter は設定されていません。デフォルトで動的な検出が実行されます。
	ルータ ガード	ディセーブル
	スタティック グループ	未設定

ブリッジ ドメイン レベルでの IGMP スヌーピング設定

IGMP の最小バージョン

minimum-version コマンドは、ブリッジ ドメインの IGMP スヌーピングでサポートされる IGMP バージョンを決定します。

- minimum-version が 2 の場合、IGMP スヌーピングは IGMPv2 および IGMPv3 メッセージを受信します。 768 ビットは、デフォルト値です。
- minimum-version が 3 の場合、IGMP スヌーピングは IGMPv3 メッセージだけを受信し、IGMPv2 メッセージはすべてドロップします。

IGMPv1 はサポートされていません。 このコマンドのスコープは、ブリッジ ドメインです。 コマンドは、ポートに適用されているプロファイルでは無視されます。

システム IP アドレス

system-ip-address コマンドでは、IGMP スヌーピング用の IP アドレスを設定します。 明示的に設定しない場合、デフォルトアドレスは 0.0.0.0 です。 次の場合を除いて、デフォルトで十分です。

- 内部クエリアを設定している場合。 内部クエリアには、0.0.0.0 は使用できません。
- ブリッジが、0.0.0.0 アドレスを受け付けない IGMP ルータと通信する必要がある場合。

IGMP スヌーピングのシステム IP アドレスは、次の方法で使用されます。

- 内部クエリアは、システム IP アドレスからクエリーを送信します。 デフォルトの 0.0.0.0 以外のアドレスを設定する必要があります。
- IGMPv3 は、システム IP アドレスからプロキシ レポートを送信します。 デフォルトのアドレス 0.0.0.0 が推奨されますが、一部の IGMP ルータは受け付けない場合があります。

- ブリッジ ドメインでのトポロジ変更通知（TCN）への応答として、IGMP スヌーピングはシステム IP アドレスからグローバル脱退を送信します。 デフォルトのアドレス 0.0.0.0 が推奨されますが、一部の IGMP ルータは受け付けない場合があります。

グループ メンバーシップ インターバル、ロバストネス変数、およびクエリー間隔

グループ メンバーシップ インターバル（GMI）は、IGMP スヌーピングが古いグループ メンバーシップ状態を失効させるタイミングを制御します。 **show igmp snooping group** コマンドは、次のクエリー インターバルの後に古い状態が消去されるまで、有効期間 0 のグループを表示します。

GMI は次のように計算されます。

GMI = (robustness-variable * query-interval) + maximum-response-time

ここで、

- maximum-response-time（MRT）は時間を表します。受信先はこの時間中にメンバーシップ状態を報告する必要があります。
- robustness-variable は、GMI の計算に影響を与える整数です。
- query-interval は一般クエリーの送信間隔を表します。

GMI のコンポーネントの値は、次のように取得されます。

- MRT は IGMPv2 および IGMPv3 両方の一般クエリーでアドバタイズされます。
- クエリアが IGMPv2 を実行している場合、IGMP スヌーピングは、robustness-variable と query-interval に IGMP スヌーピングで設定された値を使用します。 これらのパラメータ値は、クエリアに設定された値と一致している必要があります。ほとんどの場合、他のシスコ ルータと対話する場合、これらの値を明示的に設定する必要はありません。通常、IGMP スヌーピングのデフォルト値は、クエリアのデフォルト値と一致しています。一致していない場合は、**querier robustness-variable** コマンドと **querier query-interval** コマンドを使用して、一致する値を設定する必要があります。
- IGMPv3 の一般クエリーは、robustness-variable と query-interval の値（それぞれ QRV と QQI）を伝えます。 IGMP スヌーピングは、クエリーからの値を使用して、IGMP スヌーピングの GMI をクエリアの GMI と一致させます。

レポート抑制機能（IGMPv2）とプロキシ レポート機能（IGMPv3）

次の IGMP スヌーピング機能は、ブリッジ ドメインのマルチキャスト トラフィックを削減します。 両方はデフォルトでイネーブルです。

- IGMPv2 レポート抑制機能：ブリッジ ドメイン クエリアが IGMPv2 を実行している場合に、現在のクエリー間隔の間に別のホストから同じ join を転送していた場合、IGMP スヌーピングはホストからの join を抑制します。 IGMP スヌーピングは、すべての mrouter ポートに最後の leave メッセージを転送します。

レポート抑制機能がイネーブルの場合にレポートが失われた場合のために、IGMP スヌーピングはIGMPv2 の join レポートを新しいグループに対して設定された querier robustness-variable で指定された回数分転送します。 querier robustness-variable コマンドを使用して、**querier robustness-variable** を設定します。

- IGMPv3 プロキシ レポート機能：ブリッジ ドメイン クエリアが IGMPv3 を実行している場合、IGMP スヌーピングはプロキシとして動作し、プロキシ レポート アドレスからレポートを生成します。 **system-ip-address** コマンドを使用して、プロキシ レポート アドレスを設定します。 デフォルト値は 0.0.0.0 です。

  プロキシ レポート機能がイネーブルの場合にレポートが失われた場合のために、IGMP スヌーピングは、状態変更レポートを robustness-variable で指定された回数分生成し、転送します。robustness-variable は、クエリアの一般クエリーの QRV 値です。 **unsolicited-report-timer** コマンドで設定された期間、レポートは不定期に転送されます。

レポート抑制機能およびプロキシ レポート機能をディセーブルにするには、**report-suppression disable** コマンドを使用します。

この項で説明するコマンドのスコープは、ブリッジドメインです。 コマンドは、ポートに適用されているプロファイルでは無視されます。

グループ脱退処理

グループ脱退オプション

ホストをマルチキャスト グループから脱退させたい場合は、そのホストで定期的な一般 IGMP クエリーを無視するか（暗黙的脱退と呼ばれます）、またはグループ固有の leave メッセージを送信します。

IGMP スヌーピングは、グループ脱退に次のように応答します。

- 最後のメンバ クエリー処理：これは、グループ脱退を処理するデフォルトの方法です。
- 即時脱退：即時脱退に対して、任意で個別のポートを設定できます。

（注） マルチホスト LAN 上でホスト単位の即時脱退機能を提供する IGMPv3 明示的ホスト トラッキングはサポートされていません。

IGMPv2 および IGMPv3 の最後のメンバ クエリー処理

最後のメンバ クエリーは、IGMP スヌーピングで使用されるデフォルトのグループ脱退処理方法です。 最後のメンバ クエリー処理では、IGMP スヌーピングは脱退メッセージを次のように処理します。

- IGMP スヌーピングは、脱退メッセージを受信するポートでグループ固有クエリーを送信して、そのインターフェイスに接続されている他のデバイスが指定されたマルチキャスト グ

ループのトラフィックに関与しているかどうかを確認します。 次の 2 つのコンフィギュレーション コマンドを使用して、脱退の要求と実際の脱退間の遅延を制御できます。

- **last-member-query-count** コマンド：IGMP スヌーピングが脱退メッセージへの応答として送信するグループ固有クエリーの数を制御します。
- **last-member-query-interval** コマンド：グループ固有クエリーの間隔を制御します。

- IGMP スヌーピングがグループ固有クエリーへの応答として IGMP join メッセージを受信しない場合、ポートに接続されている他のデバイスは、このマルチキャスト グループのトラフィックの受信に関与していないと見なし、そのマルチキャスト グループのレイヤ 2 転送テーブルのエントリからポートを削除します。
- 脱退メッセージが唯一残っているポートから送られた場合、IGMP スヌーピングはグループのエントリを削除し、マルチキャスト ルータに IGMP の脱退を生成します。

即時脱退設定

即時脱退は、任意のポートレベルの設定パラメータです。 即時脱退処理では、IGMP スヌーピングは、事前にインターフェイスに IGMP グループ固有のクエリーを送信することなく、レイヤ 2 インターフェイスを転送テーブルのエントリから即座に削除します。 IGMP 脱退メッセージを受信すると、そのポートでマルチキャスト ルータが学習されていない限り、IGMP スヌーピングは、そのマルチキャスト グループのレイヤ 2 転送テーブル エントリからインターフェイスを即座に削除します。

即時脱退処理により脱退遅延は改善されますが、この処理が適しているのは、ポートで 1 つの受信先が設定されている場合だけです。 たとえば、即時脱退は、次の状況に適しています。

- IPTV チャネル受信先などのポイントツーポイント構成
- プロキシ レポート付きのダウンストリーム DSLAM

1 つのポートに複数の受信先が存在する可能性がある場合は、ポートで即時脱退を使用しないでください。 使用すると、関与する受信機がトラフィックを受信できなくなるおそれがあります。 たとえば、即時脱退は、LAN には適していません。

即時脱退処理は、ポートレベルのオプションです。 このオプションは、ポート プロファイルでポートごとに、またはブリッジ ドメイン プロファイルで明示的に設定できます。ブリッジ ドメイン プロファイルの場合は、ブリッジに属するすべてのポートに適用されます。

トポロジ変更通知への反応

スパニングツリー プロトコル（STP）トポロジでは、トポロジ変更通知（TCN）は、STP トポロジ変更が発生したことを示します。 トポロジ変更の結果、mrouter とグループ メンバーシップを報告するホストはブリッジ ドメインに属する他の STP ポートに移行することがあります。 TCN 後、mrouter とメンバーシップの状態を再学習する必要があります。

IGMP スヌーピングは次のように TCN に反応します。

1 IGMP スヌーピングは、すべての既知のマルチキャスト ルートに設定されているフラッディングを、転送状態にある STP に参加するすべてのポートを含めるように一時的に拡張します。短期的なフラッディングにより、マルチキャスト配信はブリッジ ドメインのすべての mrouter とすべてのメンバ ホストに対して続行され、mrouter とメンバーシップの状態が再学習されます。

ただし、この TCN フラッディングの結果として、これらの追加のマルチキャスト フローにより、ダウンストリーム STP リンクがオーバーサブスクライブになる可能性があります。 このような場合は **tcn flood disable** コマンドを使用して、この機能をディセーブルにすることができます。

2 STP ルート ブリッジは、すべてのポートで（グループ 0.0.0.0 の）グローバル脱退を発行します。 この動作により、相互運用可能な IGMP クエリアは一般クエリーを送信して、再学習プロセスを促進します。

（注） グローバル脱退の送信によるクエリー要請は、シスコ固有の実装です。

3 TCN リフレッシュ期間が終了すると、IGMP スヌーピングは、マルチキャスト ルート フラッディング セットから非 mrouter および非メンバの STP ポートを除外します。 フラッディングを行う時間は、**tcn flood query count** コマンドで制御できます。 このコマンドは、TCN 後にマルチキャスト トラフィックのフラッディングに使用する IGMP 一般クエリーの数を設定するので、リフレッシュ期間に影響します。

IGMP スヌーピングのデフォルトの動作では、STP ルート ブリッジは、TCN への応答として常にグローバル脱退を発行し、非ルート ブリッジはグローバル脱退を発行しません。

tcn query solicit コマンドを使用すると、ルート ブリッジではないブリッジでも、TCN への応答として常にグローバル脱退の発行をイネーブルにできます。その場合、ルートブリッジと非ルートブリッジがグローバル脱退を発行し、両方が、TCN への応答として一般クエリーを要請します。ブリッジがルートではない場合の要請をオフにするには、コマンドの **no** 形式を使用します。

（注） **tcn query solicit** コマンドを使用する方法の 1 つは、リバース レイヤ 2 ゲートウェイ プロトコル（RL2GP）が MSTP アクセス ゲートウェイを設定するように設定されている場合です。 このシナリオで、IGMP スヌーピングはブリッジのルート ステータスまたは非ルート ステータスを認識しないため、TCN が発生すると、IGMP スヌーピングが少なくとも 1 つのブリッジで明示的に応答するように設定されていない限り、ドメイン内のどのクエリーも応答しません。

ルート ブリッジは常に、TCN への応答としてグローバル脱退を発行します。 この動作はディセーブルにできません。

内部クエリアには、TCN への反応を制御する独自の設定オプションがあります。

すべての TCN 関連設定オプションのスコープは、ブリッジ ドメイン単位です。 ポートに対応付けられたプロファイルにコマンドを使用しても効果はありません。

IGMP スヌーピングのパケット チェック

デフォルトでは、IGMP スヌーピングは次の検証を実行します。 ネットワークがこれらの検証を別の場所で実行する場合は、IGMP スヌーピング検証をディセーブルにできます。

- IGMP スヌーピングは、IGMP ヘッダーの存続可能時間（TTL）フィールドを確認し、TTL が 1 でないパケットをドロップします。 IGMP レポートおよびクエリーのヘッダーでは、TTL フィールドは常に 1 に設定されている必要があります。

  このチェックは **ttl-check disable** コマンドを使用してディセーブルにできます。この場合、IGMP スヌーピングは IGMP ヘッダーの TTL フィールドを検証することなく、すべてのパケットを処理します。

- IGMP スヌーピングは、IGMP メッセージの IP パケット ヘッダーにルータ アラート オプションがあるかどうかをチェックし、このオプションを含んでいないパケットをドロップします。

  このチェックは **router-alert-check disable** コマンドを使用してディセーブルにできます。この場合、IGMP スヌーピングはメッセージを処理する前に検証を実行しません。

スタートアップ クエリーの設定

スタートアップ クエリー機能は新しい IGMP スヌーピング プロファイル パラメータを使用して設定されます。 次のイベントに応答するように、スタートアップ クエリー処理を設定することができます。

- MC-LAG ポートがアクティブになったとき
- トポロジの変更
- ポートの起動
- 処理の開始

上記のパラメータは MC-LAG 機能に固有です。 これらはカウント、MRT、クエリー インターバルなどの既存のブリッジ ドメイン レベル パラメータとは異なります。 これらの CLI の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Multicast Command Reference*』を参照してください。

（注）

- IGMP スヌーピングが MC-LAG で正しく動作するには、両方の POA の IGMP スヌーピング設定が同じである必要があります。
- ダウンストリーム MC-LAG の場合、MC-LAG が設定され稼働している場合は、MC-LAG ポートを IGMP スヌーピング対応ブリッジドメインに追加する必要があります。
- アップストリーム MC-LAG の場合、POA がマルチキャスト ルータに適用されている場合は、トラフィックが両方の POA に供給されるようにするため、スタティック mrouter ポートを両方の POA に向いているマルチキャスト ルータに設定する必要があります。

ホスト ポート レベルの IGMP スヌーピング設定

ルータ ガードおよびスタティック mrouter

ルータ ガードは、悪意のあるユーザがホスト ポートを mrouter ポートにするのを防ぐセキュリティ機能です（この不正な動作はスプーフィングと呼ばれます）。ポートが **router-guard** コマンドで保護されていると、そのポートが mrouter としてダイナミックに検出されることはありません。ポート上でルータ ガードを設定すると、IGMP スヌーピングはポートに送信されたプロトコル パケットをフィルタリングして、マルチキャスト ルータ制御パケットの場合は破棄します。

mrouter コマンドはポートをスタティック mrouter として設定します。

たとえば次のような場合、同じポートで、**router-guard** コマンドと **mrouter** コマンドを使用して、ガードされたポートをスタティック mrouter として設定できます。

- 大量のダウンストリーム ホスト ポートが存在する場合に、動的な mrouter 検出をブロックして、スタティック mrouter を設定する場合。 この場合、ドメイン レベルでルータ ガード機能を設定します。 デフォルトでは、一般に大量のダウンストリーム ホスト ポートを含むすべてのポートに適用されます。 次に、比較的少数のアップストリーム ポートに、ルータ ガードを設定していない別のプロファイルを指定して動的な mrouter 検出を許可するか、スタティック mrouter を設定します。
- シスコ以外の機器との非互換性により動的検出を正しく行えない場合は、ルータ ガード機能を使用して動的検出をすべてディセーブルにして、mrouter を静的に設定できます。

  ポートに非互換 IGMP ルータがあるためにルータ ガード機能を使用している場合、そのポートで **mrouter** コマンドも設定して、ルータが IGMP レポートとマルチキャスト フローを受信できるようにする必要があります。

即時脱退

グループ脱退処理, （14 ページ）を参照してください。

スタティック グループ

IGMP スヌーピングは、レイヤ 2 マルチキャスト グループを動的に学習します。 レイヤ 2 マルチキャスト グループを静的に設定することもできます。

ブリッジ ドメインまたはポート用のプロファイルで **static group** コマンドを使用できます。 このオプションをブリッジ ドメインに対応付けられたプロファイルで設定すると、そのブリッジに属するすべてのポートに適用されます。

プロファイルには、複数のスタティック グループを含めることができます。 同じグループ アドレスに異なるソース アドレスを定義できます。 **source** キーワードを使用して、IGMPv3 ソース グループを設定できます。

スタティック グループ メンバーシップは、IGMP スヌーピングによるダイナミック操作より優先されます。 マルチキャスト グループ メンバーシップ リストには、スタティックとダイナミック両方のグループ定義を表示できます。

ポートでスタティック グループまたは送信元グループを設定すると、IGMP スヌーピングは、対応する <S/*,G> 転送エントリにポートを発信ポートとして追加し、IGMPv2 join または IGMPv3 report をすべての mrouter ポートに送信します。 IGMP スヌーピングは、スタティック グループがポート上で設定されている限り、一般クエリーへの応答としてメンバーシップ レポートを送信し続けます。

内部クエリア

内部クエリアを使用する場合

IP マルチキャスト ルーティングが設定されているネットワークでは、IP マルチキャスト ルータは IGMP クエリアとして機能します。 ブリッジ ドメインに外部クエリアは存在しない（マルチキャスト トラフィックをルーティングする必要がないため）が、ローカル マルチキャスト ソースが存在する状況では、内部クエリアを設定して IGMP スヌーピングを実装する必要があります。内部クエリアは、ブリッジ ドメインのホストからメンバーシップ レポートを要請し、IGMP スヌーピングがブリッジドメイン内のマルチキャストトラフィック用の制約的なマルチキャスト転送テーブルを作成できるようにします。

内部クエリアは、シスコ以外の機器での相互運用性の問題により、IGMP スヌーピングが外部クエリアと正しく連携できない場合にも役立つことがあります。 この場合、次のように対処できます。

1. 対象のポートに **router-guard** コマンドを発行して、関係のない外部クエリアが検出されるのを防ぐ。
2. ブリッジ ドメインのポートから、関連するグループ メンバーシップを学習するように内部クエリアを設定する。
3. マルチキャスト トラフィックを受信するスタティック mrouter ポートを設定する。

内部クエリアのデフォルト設定

内部クエリアの最小構成は次のとおりです。

- ブリッジ ドメインに対応付けられたプロファイルに、**internal-querier** コマンドを追加します。 デフォルト設定を表 4：内部クエリアのデフォルト設定値, （20 ページ） に示します。
- ブリッジ ドメインに対応付けられたプロファイルに、**system-ip-address** コマンドを追加して、デフォルトの 0.0.0.0 以外のアドレスを設定します。

表 *4*：内部クエリアのデフォルト設定値

コンフィギュレーション コマンド	デフォルト値
system-ip-address	0.0.0.0。デフォルトのアドレスは、内部クエリアでは無効です。
internal-querier *max-response-time*	10
internal-querier *query-interval*	60（秒） （注） これは、非標準デフォルト値です。
internal-querier *robustness-variable*	2
internal-querier tcn query *count*	2
internal-querier tcn query *interval*	10 秒
internal-querier timer *expiry* （注） これは RFC-3376 Section 8.5 で定義されている Other Querier Present Interval です。	125（秒）： robustness-variable * query-interval + 1/2(max-response-time) たとえば、すべてのコンポーネントのデフォルト値を使用した場合： (2 * 60) + 1/2 (10) = 125
internal-querier *version*	3

他の内部クエリア コマンドを削除することなく、（**internal-querier** コマンドの **no** 形式を使用して）内部クエリアをディセーブルにできます。その場合、追加の内部クエリア コマンドは無視されます。

internal-querier コマンドのスコープは、ブリッジ ドメイン単位です。 ポートに対応付けられたプロファイルにコマンドを使用しても効果はありません。

内部クエリアの処理

内部クエリアがドメインで選定されたクエリアである場合、ブリッジ ドメインのすべてのアクティブ ポートに **internal-querier query-interval** コマンドで指定された間隔で IGMP 一般クエリーを送信することで、メンバーシップ レポートを要請します。 内部クエリアは、IGMPv3 クエリーをデフォルトで送信します。 代わりに **internal-querier version** コマンドを使用して、内部クエリアが IGMPv2 メッセージを送信するように設定できます。

ローカル IGMP スヌーピング プロセスは、内部クエリアの一般クエリーに応答します。 特に、IGMPv3 プロキシ（イネーブルの場合）は、現在の状態レポートを生成し、すべての mrouter に転送します。 IGMPv2 の場合、または IGMPv3 プロキシがディセーブルになっている場合、IGMP スヌーピングはスタティック グループの状態についてのみ現在の状態レポートを生成します。

クエリーは、**system-ip-address** コマンドを使用して IGMP スヌーピング用に設定したアドレスから送信されます。 クエリーには、**internal-querier max-response-time** コマンドで設定された最大応答時間が含まれます。

internal-querier robustness-variable コマンドおよび **internal-querier query-interval** コマンドは、IGMPv2 および IGMPv3 処理の両方の値を設定します。

1 つのアクティブなクエリアの選定

ブリッジ ドメインで一度に使用できるアクティブなクエリアは 1 つだけです。 内部クエリアが、ブリッジ ドメインの他のクエリアからクエリーを受信すると、クエリアの選定が行われます。 最下位の IP アドレスが選択されます。 内部クエリアが選定されなかったクエリアの場合、IGMP スヌーピングは **internal-querier timer expiry** コマンドで設定された値でタイマーを開始します。 このタイマーの期限が、選択されたクエリアから別のクエリーを受信するまでに切れた場合、内部クエリアがアクティブなクエリアになります。

（注） デフォルトの **internal-querier timer expiry** コマンドの値は、表 4：内部クエリアのデフォルト設定値，（20 ページ） に記載されている他の設定オプションの値から取得されます。 デフォルトの計算を上書きする別の値を設定できます。

TCN への内部クエリアの反応

IGMP スヌーピングはトポロジ変更通知への応答として、グループの脱退を生成します。 IGMP スヌーピングの TCN への反応方法の詳細については、トポロジ変更通知への反応，（15 ページ）を参照してください。

内部クエリアがドメインで選定されたクエリアの場合に、グループの脱退を受信すると、次のように反応します。

- IGMP 一般クエリーをただちに生成します。

- **internal-querier tcn query interval** コマンドで設定されている時間待機し、別の IGMP 一般クエリーを生成します。
- クエリー回数が **internal querier tcn query count** コマンドで設定された値に達するまで、指定された間隔待機して、一般クエリーを送信する動作を続けます。

（注） internal querier TCN query count を 0 に設定することで、内部クエリアがグローバル脱退を無視するように設定できます。

IGMP スヌーピングの設定方法

最初の 2 つの作業は、基本的な IGMP スヌーピングの設定に必須です。 オプションの作業では、追加の IGMP スヌーピング機能を設定し、統計情報およびカウンタを表示します。

- プロファイルへのスタティック mrouter 設定の追加, （32 ページ）（任意）
- プロファイルへのルータ ガードの追加, （34 ページ）（任意）
- 即時脱退の設定, （36 ページ）（任意）
- スタティック グループの設定, （38 ページ）（任意）
- 内部クエリアの設定, （40 ページ）（任意）
- マルチキャスト転送の確認, （43 ページ）（任意）

IGMP スヌーピング プロファイルの作成

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. オプションで、デフォルト設定値を上書きするコマンドを追加します。
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile default-bd-profile`	IGMP スヌーピング プロファイル コンフィギュレーション モードを開始し、名前付きプロファイルを作成します。 デフォルト プロファイルは、IGMP スヌーピングをイネーブルにします。追加の設定をせずに新しいプロファイルをコミットするか、プロファイルに追加の設定オプションを含めることができます。 後でプロファイルに戻って、このモジュールの他の作業で記載されている手順に従って、設定を追加することもできます。
ステップ 3	オプションで、デフォルト設定値を上書きするコマンドを追加します。	ブリッジ ドメイン プロファイルを作成する場合は、次の点を考慮します。 • 空のプロファイルは、ブリッジ ドメインへの適用に適しています。空のプロファイルは、デフォルト設定値で IGMP スヌーピングをイネーブルにします。 • オプションで、デフォルト設定値を上書きするコマンドをプロファイルに追加できます。 • ブリッジ ドメイン プロファイルにポート固有の設定を含める場合、別のプロファイルがポートに適用されていない限り、設定はそのブリッジに属するすべてのポートに適用されます。 ポート固有のプロファイルを作成する場合は、次の点を考慮します。 • 空のプロファイルはポートに適用できますが、ポートの設定には影響を与えません。 • ポートにプロファイルを適用する際、IGMP スヌーピングはブリッジ ドメイン プロファイルからの設定値の継承を上書きして、ポートを再設定します。 これらの設定を保持する場合は、ポート プロファイルのコマンドを繰り返し実行する必要があります。 後でプロファイルにコマンドを追加するには、プロファイルの適用を解除し、プロファイルを変更してから再適用します。
ステップ 4	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

次の作業

プロファイルをブリッジ ドメインまたはポートに適用し、プロファイルを有効にする必要があります。 次のいずれかの作業を参照してください。

プロファイルの適用およびブリッジ ドメインでの IGMP スヌーピングのアクティブ化

ブリッジ ドメインで IGMP スヌーピングをアクティブにするには、次の手順の説明に従って、ブリッジ ドメインに IGMP スヌーピング プロファイルを適用します。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **igmp snooping profile** *profile-name*
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
7. **show igmp snooping bridge-domain detail**
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`	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	**bridge group** *bridge-group-name* 例： `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group GRP1`	名前付きブリッジ グループのレイヤ 2 VPN VPLS 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 ISP1`	名前付きブリッジ ドメインのレイヤ 2 VPN VPLS 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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# igmp snooping profile default-bd-profile`	ブリッジ ドメインに名前付き IGMP スヌーピング プロファイルを適用し、ブリッジ ドメインで IGMP スヌーピングをイネーブルにします。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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 7	**show igmp snooping bridge-domain detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping bridge-domain detail`	（任意）IGMP スヌーピングがブリッジ ドメインでイネーブルであることを確認し、ブリッジ ドメインおよびポートに適用される IGMP スヌーピング プロファイルの名前を表示します。
ステップ 8	**show l2vpn bridge-domain detail** 例： `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain`	（任意）IGMP スヌーピングがブリッジ ドメインのフォワーディング プレーン（レイヤ2）に実装されていることを確認します。

プロファイルの適用解除とブリッジ ドメインでの IGMP スヌーピングの非アクティブ化

ブリッジ ドメインで IGMP スヌーピングを非アクティブ化するには、次の手順を使用して、ブリッジ ドメインからプロファイルを削除します。

（注） ブリッジ ドメインに一度に適用できるプロファイルは 1 つだけです。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **no igmp snooping**
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
7. **show igmp snooping bridge-domain detail**
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`	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 GRP1`	名前付きブリッジ グループのレイヤ 2 VPN VPLS 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 ISP1`	名前付きブリッジ ドメインのレイヤ 2 VPN VPLS 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	**no igmp snooping** 例： `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# no igmp snooping`	ブリッジ ドメインから IGMP スヌーピング プロファイルの適用を解除し、ブリッジ ドメインで IGMP スヌーピングをディセーブルにします。 （注） 同時にブリッジ ドメインに適用できるプロファイルは 1 つだけです。 プロファイルが適用されている場合、IGMP スヌーピングはイネーブルです。 プロファイルが適用されていない場合、IGMP スヌーピングはディセーブルです。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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 7	**show igmp snooping bridge-domain detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping bridge-domain detail`	（任意）IGMP スヌーピングがブリッジ ドメインでディセーブルであることを確認します。
ステップ 8	**show l2vpn bridge-domain detail** 例： `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain`	（任意）IGMP スヌーピングがブリッジ ドメインのフォワーディング プレーン（レイヤ2）でディセーブルであることを確認します。

ブリッジに属するポートへのプロファイルの適用と解除

はじめる前に

ポート固有のプロファイルが IGMP スヌーピングの動作に影響を与えるようにするには、ブリッジ ドメインで IGMP スヌーピングがイネーブルになっている必要があります。

手順の概要

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **interface** *interface-type interface-number*
6. 次のいずれかを実行します。
   - **igmp snooping profile** *profile-name*
   - **no igmp snooping**
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show igmp snooping bridge-domain detail**
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`	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	**bridge group** *bridge-group-name* 例： `RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group GRP1`	名前付きブリッジ グループのレイヤ 2 VPN 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)#`	名前付きブリッジ ドメインのレイヤ 2 VPN 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
	`bridge-domain ISP1`	
ステップ 5	**interface** *interface-type interface-number* 例： `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface gig 1/1/1/1`	名前付きインターフェイスまたは PW のレイヤ 2 VPN VPLS ブリッジ グループ ブリッジ ドメイン 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 6	次のいずれかを実行します。 • **igmp snooping profile** *profile-name* • **no igmp snooping** 例： `RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-if)# igmp snooping profile mrouter-port-profile`	名前付き IGMP スヌーピング プロファイルをポートに適用します。 （注） ポートのプロファイルは、ブリッジに他のプロファイルが適用されていない限り、無効です。 コマンドの **no** 形式を使用して、ポートからプロファイルの適用を解除します。 ポートに適用できるプロファイルは 1 つだけです。

ステップ 7	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

	コマンドまたはアクション	目的
ステップ 8	**show igmp snooping bridge-domain detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping bridge-domain detail`	（任意）IGMP スヌーピングがブリッジ ドメインでイネーブルであることを確認し、ブリッジ ドメインおよびポートに適用される IGMP スヌーピング プロファイルの名前を表示します。
ステップ 9	**show l2vpn bridge-domain detail** 例： `RP/0/RSP0/CPU0:router# show l2vpn bridge-domain`	（任意）IGMP スヌーピングがブリッジ ドメインのフォワーディング プレーン（レイヤ 2）に実装されていることを確認します。

プロファイルへのスタティック mrouter 設定の追加

はじめる前に

ポート固有のプロファイルが IGMP スヌーピングの動作に影響を与えるようにするには、ブリッジ ドメインで IGMP スヌーピングがイネーブルになっている必要があります。

（注） スタティック mrouter ポート設定はポートレベルのオプションであり、ポートを対象としたプロファイルに追加する必要があります。ブリッジドメインを対象としたプロファイルに mrouter ポート設定を追加することは推奨しません。

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **mrouter**
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
5. **show igmp snooping profile** *profile-name* **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile mrouter-port-profile`	IGMP スヌーピング プロファイル コンフィギュレーションモードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。
ステップ 3	**mrouter** 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# mrouter`	スタティック mrouter ポートとしてポートを設定します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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

	コマンドまたはアクション	目的
		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 igmp snooping profile** *profile-name* **detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping profile mrouter-port-profile detail`	（任意）名前付きプロファイルの設定を表示します。

次の作業

スタティック mrouter 設定を完了するには、ポートにプロファイルを適用します。 ブリッジに属するポートへのプロファイルの適用と解除, （29 ページ）を参照してください。

プロファイルへのルータ ガードの追加

マルチキャスト ルーティング プロトコル メッセージをポート上で受信しないようにして、ポートが動的 mrouter ポートになることを防止するには、次の手順を実行します。 ルータ ガードとスタティック mrouter コマンドの両方が同じポートで設定されることに注意してください。 詳細については、ルータ ガードおよびスタティック mrouter, （18 ページ）を参照してください。

はじめる前に

ポート固有のプロファイルが IGMP スヌーピングの動作に影響を与えるようにするには、ブリッジ ドメインで IGMP スヌーピングがイネーブルになっている必要があります。

（注） ルータ ガード設定はポートレベルのオプションであり、ポートを対象としたプロファイルに追加する必要があります。 ブリッジ ドメインを対象としたプロファイルにルータ ガード設定を追加することは推奨しません。 設定すると、IGMP クエリアを含むすべての mrouter がブリッジ ドメインでは検出されなくなります。

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **router-guard**
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
5. **show igmp snooping profile** *profile-name* **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile host-port-profile`	IGMP スヌーピング プロファイル コンフィギュレーションモードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。
ステップ 3	**router-guard** 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# router-guard`	動的検出からポートを保護します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **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。

	コマンドまたはアクション	目的
		◦ 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 5	**show igmp snooping profile** *profile-name* **detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping profile host-port-profile detail`	（任意）名前付きプロファイルの設定を表示します。

次の作業

ルータ ガード設定を完了するには、ポートにプロファイルを適用します。ブリッジに属するポートへのプロファイルの適用と解除, （29 ページ）を参照してください。

即時脱退の設定

IGMP スヌーピング プロファイルに IGMP スヌーピング即時脱退オプションを追加する手順は、次のとおりです。

はじめる前に

ポート固有のプロファイルが IGMP スヌーピングの動作に影響を与えるようにするには、ブリッジ ドメインで IGMP スヌーピングがイネーブルになっている必要があります。

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **immediate-leave**
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
5. **show igmp snooping profile** *profile-name* **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile host-port-profile`	IGMP スヌーピング プロファイル コンフィギュレーション モードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。
ステップ 3	**immediate-leave** 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# immediate-leave`	immediate-leave オプションをイネーブルにします。 • ブリッジドメインに適用されたプロファイルにこのオプションを追加すると、そのブリッジに属するすべてのポートに適用されます。 • ポートに適用されたプロファイルにこのオプションを追加すると、このオプションはそのポートに適用されます。
ステップ 4	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# 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

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	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。 • 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 5	**show igmp snooping profile** *profile-name* **detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping profile host-port-profile detail`	（任意）名前付きプロファイルの設定を表示します。

次の作業

即時脱退の設定を完了するには、ブリッジ ドメインまたはポートにプロファイルを適用します。次のいずれかの項を参照してください。

スタティック グループの設定

IGMP スヌーピング プロファイルに 1 つ以上のスタティック グループまたは IGMPv3 送信元グループを追加するには、次の手順を実行します。

はじめる前に

ポート固有のプロファイルが IGMP スヌーピングの動作に影響を与えるようにするには、ブリッジ ドメインで IGMP スヌーピングがイネーブルになっている必要があります。

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **static-group** *group-addr* [**source** *source-addr*]
4. スタティック グループをさらに追加する場合は、必要に応じて前の手順を繰り返します。
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
6. **show igmp snooping profile** *profile-name* **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile host-port-profile`	IGMP スヌーピング プロファイル コンフィギュレーション モードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。
ステップ 3	**static-group** *group-addr* [**source** *source-addr*] 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# static-group 239.1.1.1 source 10.0.1.1`	スタティック グループを設定します。 • ブリッジ ドメインに適用されたプロファイルにこのオプションを追加すると、そのブリッジに属するすべてのポートに適用されます。 • ポートに適用されたプロファイルにこのオプションを追加すると、このオプションはそのポートに適用されます。
ステップ 4	スタティック グループをさらに追加する場合は、必要に応じて前の手順を繰り返します。	（任意）追加のスタティック グループを追加します。
ステップ 5	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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 6	**show igmp snooping profile** *profile-name* **detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping profile host-port-profile detail`	（任意）名前付きプロファイルの設定を表示します。

次の作業

スタティック グループ設定を完了するには、ブリッジ ドメインまたはポートにプロファイルを適用します。 次のいずれかの項を参照してください。

内部クエリアの設定

はじめる前に

この手順を有効にするには、IGMP スヌーピングがそのブリッジ ドメインでイネーブルになっている必要があります。

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **system-ip-address** *ip-addr*
4. **internal-querier**
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
6. **show igmp snooping profile** *profile-name* **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile internal-querier-profile`	IGMP スヌーピング プロファイル コンフィギュレーション モードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。
ステップ 3	**system-ip-address** *ip-addr* 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# system-ip-address 10.1.1.1`	内部クエリアが使用する IP アドレスを設定します。 デフォルトの system-ip-address の値（0.0.0.0）は、内部クエリアでは無効です。 IP アドレスを明示的に設定する必要があります。
ステップ 4	**internal-querier** 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# internal-querier`	すべてのオプションにデフォルト値を使用して、内部クエリアをイネーブルにします。
ステップ 5	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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 6	**show igmp snooping profile** *profile-name* **detail** 例： `RP/0/RSP0/CPU0:router# show igmp snooping profile internal-querier-profile detail`	（任意）名前付きプロファイルの設定を表示します。

次の作業

内部クエリアの設定を完了するには、ブリッジ ドメインにプロファイルを適用します。

プロファイルの適用およびブリッジ ドメインでの IGMP スヌーピングのアクティブ化, （24 ページ）を参照してください。

マルチキャスト転送の確認

手順の概要

1. **configure**
2. **show l2vpn forwarding bridge-domain** [*bridge-group-name*:*bridge-domain-name*] **mroute ipv4** [**detail**] [**hardware** {**ingress** | **egress**}] **location** *node-id*
3. **show l2vpn forwarding bridge-domain** [*bridge-group-name*:*bridge-domain-name*] **mroute ipv4 summary location** *node-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	**show l2vpn forwarding bridge-domain** [*bridge-group-name*:*bridge-domain-name*] **mroute ipv4** [**detail**] [**hardware** {**ingress**	**egress**}] **location** *node-id* 例： `RP/0/RSP0/CPU0:router# show l2vpn forwarding bridge-domain bridgeGroup1:ABC mroute ipv4 detail location 0/3/CPU0`	フォワーディング プレーンの転送テーブルに変換されるマルチキャスト ルートを表示します。 特定のブリッジ グループまたはブリッジ ドメインに表示を制限するには、任意の引数を使用します。 これらのルートが期待したルートではない場合は、コントロール プレーンの設定を確認し、対応する IGMP スヌーピング プロファイルを訂正してください。
ステップ 3	**show l2vpn forwarding bridge-domain** [*bridge-group-name*:*bridge-domain-name*] **mroute ipv4 summary location** *node-id* 例： `RP/0/RSP0/CPU0:router# show l2vpn forwarding bridge-domain bridgeGroup1:ABC mroute ipv4 summary location 0/3/CPU0`	フォワーディング プレーンの転送テーブルに保存されているマルチキャスト ルートの要約レベルの情報を表示します。 特定のブリッジ ドメインに表示を制限するには、任意の引数を使用します。

グループ制限の設定

この手順では、次の作業について説明します。

ルート ポリシーの設定

手順の概要

1. **configure**
2. **route-policy** *policy-name*
3. **end-policy**
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**route-policy** *policy-name* 例： `RP/0/RSP0/CPU0:router(config)# route-policy sky`	定義されている名前でルート ポリシーを設定します。
ステップ 3	**end-policy** 例： `RP/0/RSP0/CPU0:router(config-rpl)# end-policy`	ルートポリシーの設定を終了します。
ステップ 4	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# 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。

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	◦ 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。

グループ上限の設定

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **group policy** *policy-name*
4. **group limit** *range*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile`	IGMP スヌーピング プロファイル コンフィギュレーション モードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。

	コマンドまたはアクション	目的
	name1	
ステップ 3	**group policy** *policy-name* 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# group policy policy1`	設定されたルートポリシーがグループの重みを設定するように指定します。
ステップ 4	**group limit** *range* 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# group limit 100`	ポートで許容されているグループ（または送信元グループ）の数を制限します。

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

アクセスグループの設定

この作業では、メンバーシップ レポートを受信するために、IGMP スヌープに指定されたアクセス リスト フィルタを適用するよう指示します。

ユーザはアクセスグループを設定する前にアクセス リストを作成し、設定する必要があります。標準アクセス リストおよび拡張アクセス リストを作成し設定する詳細な設定手順については、『Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Configuration Guide』を参照してください。

手順の概要

1. **configure**
2. **igmp snooping profile** *profile-name*
3. **access-group** *acl-name*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**igmp snooping profile** *profile-name* 例： `RP/0/RSP0/CPU0:router(config)# igmp snooping profile name1`	IGMP スヌーピング プロファイル コンフィギュレーション モードを開始し、新しいプロファイルを作成するか、または既存のプロファイルにアクセスします。
ステップ 3	**access-group** *acl-name* 例： `RP/0/RSP0/CPU0:router(config-igmp-snooping-profile)# access-group acl1`	グループ メンバーシップ フィルタを設定します。
ステップ 4	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

IGMP スヌーピングの設定例

次に、Cisco ASR 9000 シリーズ ルータ のレイヤ 2 VPLS ブリッジ ドメインで IGMP スヌーピングをイネーブルにする例を示します。

ブリッジに属する物理インターフェイスでの IGMP スヌーピングの設定：例

1 2 つのプロファイルを作成します。

```
igmp snooping profile bridge_profile
!
igmp snooping profile port_profile
```

```
   mrouter
!
```

2 L2 転送用の 2 つの物理インターフェイスを設定します。

```
interface GigabitEthernet0/8/0/38
   negotiation auto
   l2transport
   no shut
   !
!
interface GigabitEthernet0/8/0/39
   negotiation auto
   l2transport
   no shut
   !
!
```

3 ブリッジ ドメインにインターフェイスを追加します。 ブリッジ ドメインに bridge_profile を適用し、イーサネット インターフェイスのいずれかに port_profile を適用します。 2 番目のイーサネット インターフェイスは、ブリッジ ドメイン プロファイルから IGMP スヌーピング設定属性を継承します。

```
l2vpn
  bridge group bg1
     bridge-domain bd1
     igmp snooping profile bridge_profile
     interface GigabitEthernet0/8/0/38
       igmp snooping profile port_profile
     interface GigabitEthernet0/8/0/39
     !
  !
!
```

4 設定されたブリッジ ポートを確認します。

```
show igmp snooping port
```

ブリッジに属する VLAN インターフェイスでの IGMP スヌーピングの設定：例

1 2 つのプロファイルを設定します。

```
igmp snooping profile bridge_profile
igmp snooping profile port_profile
   mrouter
!
```

2 L2 転送用の VLAN インターフェイスを設定します。

```
interface GigabitEthernet0/8/0/8
   negotiation auto
   no shut
   !
!
interface GigabitEthernet0/8/0/8.1 l2transport
   encapsulation dot1q 1001
   mtu 1514
```

```
   !
!
interface GigabitEthernet0/8/0/8.2 l2transport
   encapsulation dot1q 1002
   mtu 1514
   !
!
```

3 プロファイルを適用し、ブリッジ ドメインにインターフェイスを追加します。 インターフェイスのいずれかにプロファイルを適用します。 他のインターフェイスは、ブリッジ ドメイン プロファイルから IGMP スヌーピング設定属性を継承します。

```
l2vpn
  bridge group bg1
     bridge-domain bd1
     igmp snooping profile bridge_profile
     interface GigabitEthernet0/8/0/8.1
       igmp snooping profile port_profile
     interface GigabitEthernet0/8/0/8.2
     !
  !
!
```

4 設定されたブリッジ ポートを確認します。

```
show igmp snooping port
```

ブリッジに属するイーサネット バンドルでの IGMP スヌーピングの設定：例

1 この例では、バンドルのフロントエンドが事前に設定されていることを前提にしています。たとえば、バンドル設定が次の3つのスイッチインターフェイスから構成されているとします。

```
    interface Port-channel1
    !
interface GigabitEthernet0/0/0/0
    !
interface GigabitEthernet0/0/0/1
!
    interface GigabitEthernet0/0/0/2
       channel-group 1 mode on
    !
    interface GigabitEthernet0/0/0/3
       channel-group 1 mode on
    !
```

2 2 つの IGMP スヌーピング プロファイルを設定します。

```
       igmp snooping profile bridge_profile
       !
       igmp snooping profile port_profile
          mrouter
       !
```

3 バンドルのメンバ リンクとしてインターフェイスを設定します。

```
       interface GigabitEthernet0/0/0/0
         bundle id 1 mode on
```

```
  negotiation auto
!
interface GigabitEthernet0/0/0/1
  bundle id 1 mode on
  negotiation auto
!
interface GigabitEthernet0/0/0/2
  bundle id 2 mode on
  negotiation auto
!
interface GigabitEthernet0/0/0/3
  bundle id 2 mode on
  negotiation auto
!
```

4 L2 転送用のバンドル インターフェイスを設定します。

```
interface Bundle-Ether 1
          l2transport
          !
       !
       interface Bundle-Ether 2
          l2transport
          !
       !
```

5 インターフェイスをブリッジ ドメインに追加し、IGMP スヌーピング プロファイルを適用します。

```
       l2vpn
         bridge group bg1
            bridge-domain bd1
            igmp snooping profile bridge_profile
            interface bundle-Ether 1
              igmp snooping profile port_profile
            interface bundle-Ether 2
            !
         !
      !
```

6 設定されたブリッジ ポートを確認します。

```
show igmp snooping port
```

ブリッジに属する VFI での IGMP スヌーピングの設定：例

次に、ブリッジ ドメインに属する仮想転送インスタンス（VFI）に IGMP スヌーピングを設定する例を示します。 トポロジは 2 つのルータ（PE1 および PE2）から構成され、ブリッジ ポートとしてアクセス回線（AC）と疑似配線（PW）を持っています。

PE1 の設定

1 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile prof1
!
igmp snooping profile prof2
   mrouter
!
```

2 インターフェイスを設定します。

```
interface Loopback0
  ipv4 address 10.1.1.1 255.255.255.255
!
interface GigabitEthernet0/2/0/9
   ipv4 address 10.10.10.1 255.255.255.0
   negotiation auto
!
interface GigabitEthernet0/2/0/39
   negotiation auto
   l2transport
!
```

3 Open Shortest Path First（OSPF）を設定します。

```
router ospf 1
  log adjacency changes
  router-id 10.1.1.1
  area 0
    interface Loopback0
    !
    interface GigabitEthernet0/2/0/9
    !
  !
!
```

4 ラベル配布プロトコル（LDP）を設定します。

```
mpls ldp
   router-id 10.1.1.1
   log neighbor
   !
   interface GigabitEthernet0/2/0/9
   !
!
```

5 ブリッジ ドメインを設定し、ブリッジ上でIGMP スヌーピングをイネーブルにして、ブリッジ ドメインにインターフェイスを追加します。

```
l2vpn
  pw-class atom-dyn
  encapsulation mpls
     protocol ldp
  !
!

bridge group bg1
    bridge-domain bd1
    igmp snooping profile prof1
    interface GigabitEthernet0/2/0/39
       igmp snooping profile prof2
    vfi mplscore
       neighbor 10.2.2.2 pw-id 101
       pw-class atom-dyn
       !
    !
  !
!
```

6 設定されたブリッジ ポートを確認します。

```
show igmp snooping port
```

PE2 の設定

1 IGMP プロファイルを設定します。

```
igmp snooping profile bridge_profile
!
igmp snooping profile port_profile
   mrouter
!
```

2 インターフェイスを設定します。

```
interface Loopback0
  ipv4 address 10.2.2.2 255.255.255.255
!
interface GigabitEthernet0/2/0/9
   ipv4 address 10.10.10.1 255.255.255.0
   negotiation auto
!
interface GigabitEthernet0/2/0/39
   negotiation auto
   l2transport
!
```

3 OSPF を設定します。

```
router ospf 1
  log adjacency changes
  router-id 10.2.2.2
  area 0
    interface Loopback0
    !
    interface GigabitEthernet0/2/0/9
    !
  !
!
```

4 LDP を設定します。

```
mpls ldp
   router-id 10.2.2.2
   log neighbor
   !
   interface GigabitEthernet0/2/0/9
   !
!
```

5 インターフェイスをブリッジ ドメインに追加し、IGMP スヌーピング プロファイルを適用します。

```
   l2vpn
      pw-class atom-dyn
      encapsulation mpls
         protocol ldp
      !
    !

    bridge group bg1
        bridge-domain bd1
        igmp snooping profile bridge_profile
        interface GigabitEthernet0/2/0/39
           igmp snooping profile port_profile
        vfi mplscore
           neighbor 10.1.1.1 pw-id 101
           pw-class atom-dyn
```

```
            !
          !
        !
      !
```

6 設定されたブリッジ ポートを確認します。

```
show igmp snooping port
```

IGMP アクセスグループの設定

次の例では、<*,G> グループ（225.0.0.0/24 および 228.0.0.0/24）のユーザ メンバーシップだけを許容するリストを設定し、L2VPN ブリッジ ポートに適用します。 <S,G> メンバーシップを許可する 2 番目のアクセス リストを定義します。 このアクセス リストをブリッジ ポートに適用します。

```
interface gig 0/2/0/1.1 l2transport
...
!
ipv4 access-list iptv-basic-white-list
 10 permit ipv4 any 225.0.0.0/24
 20 permit ipv4 any 228.0.0.0/24
!
ipv4 access-list iptv-premium-white-list
 10 permit ipv4 192.168.0.1 232.0.1.0/24
 20 permit ipv4 192.168.0.1 232.0.2.0/24
!
igmp snooping profile iptv
 access-group iptv-white-list
!
igmp snooping profile iptv2
 access-group iptv-premium-white-list
!
l2vpn
 bridge group vz
bridge domain vz-iptv
 igmp snooping profile iptv
 interface gig 0/2/0/1.1
 interface gig 0/2/0/1.2
  igmp snooping profile iptv2
 interface gig 0/2/0/1.3
 ...
!
```

また、IGMP ルーティングでは igmp **access-group** コマンドを使用することでアクセスグループをサポートします。 IGMP ルーティングでは簡易 IP アクセスグループを使用して、グループ アドレス フィルタを指定します。 送信元グループ フィルタおよびグループ フィルタをサポートするには、IGMP スヌーピングに拡張 IP アクセス リストが必要になります。

（注） アクセスグループはスタティック グループおよび送信元グループには適用されません。

MCLAG での IGMP スヌーピングの設定：例

ケース 1：ダウンストリーム MCLAG

トポロジ：PE に順番に接続する、2 つの POA に接続する DHD。

DHD：

1 POA1 および POA2 へのバンドルを設定します。 このデバイスは、2 つの POA の存在をマスクされています。 バンドルは、1 つの POA に接続されていると判断します。

```
interface Bundle-Ether10
 description interface towards POAs
 lacp switchover suppress-flaps 100
 bundle maximum-active links 1
l2transport
 !

!

interface GigabitEthernet0/0/0/28
 description interface towards POA1
 bundle id 10 mode active
!

interface GigabitEthernet0/0/0/29
 description interface towards POA2
 bundle id 10 mode active
!
```

2 デバイスに送信された join は、バンドル上の POA に転送する必要があります。 そのため、L2VPN BD（スヌーピングなし）内の着信ポート（ホスト ポート）とバンドルを設定します。

```
RP/0/RSP0/CPU0:router:DHD# show running-config l2vpn

l2vpn
 bridge group bg1
  bridge-domain bg1_bd1
   interface Bundle-Ether10
   !
interface GigabitEthernet0/0/0/10
   !

  !

 !

!
```

POA1：

1 インターフェイスを設定します（OSPF および MPLS LDP 用）

```
interface Loopback0

ipv4 address 20.20.20.20 255.255.255.255

!
```

```
interface GigabitEthernet0/2/0/1
description interface towards POA2
ipv4 address 10.0.0.1 255.255.255.0

 negotiation auto

!

interface GigabitEthernet0/2/0/8

 description interface towards PE

 ipv4 address 10.0.1.1 255.255.255.0

 negotiation auto

!
```

2 OSPF と MPLS LDP を設定します。

```
router ospf 1
router-id 20.20.20.20
nsf cisco
area 0
interface Loopback0

  !
interface GigabitEthernet0/2/0/1
  !
interface GigabitEthernet0/2/0/8

  !

 !

!
mpls ldp
router-id 20.20.20.20
graceful-restart
interface GigabitEthernet0/2/0/1
 !
interface GigabitEthernet0/2/0/8

 !

!
```

3 DHD への MCLAG バンドルを設定します。

```
interface Bundle-Ether10
description interface towards DHD
lacp switchover suppress-flaps 100
mlacp iccp-group 1
mlacp switchover recovery-delay 60
mlacp port-priority 1
mac-address 0.aaaa.1111
bundle wait-while 0
l2transport

 !

!
interface GigabitEthernet0/2/0/29
bundle id 10 mode active
!
```

4 MCLAG の冗長グループを設定します。

```
redundancy

 iccp
group 1
mlacp node 1
mlacp system mac 0000.aaaa.0000
mlacp system priority 1
member
neighbor 30.30.30.30
!
backbone
interface GigabitEthernet0/2/0/8
  !

  !

 !

!
```

5 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile p1
ttl-check disable
router-alert-check disable

!
```

6 PE 方向の DHD と PW に対する MCLAG バンドルを含む L2VPN BD 内の IGMP スヌーピングをイネーブルにします。

```
l2vpn
bridge group bg1
bridge-domain bg1_bd1
igmp snooping profile p1
interface Bundle-Ether10
!
vfi bg1_bd1_vfi
neighbor 40.40.40.40 pw-id 1

    !

   !

  !

 !

!
```

POA2：

1 インターフェイスを設定します（OSPF および MPLS LDP 用）

```
interface Loopback0

ipv4 address 30.30.30.30 255.255.255.255
!
interface GigabitEthernet0/0/0/1
description interface towards POA1
ipv4 address 10.0.0.2 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/0/0/8
```

```
description interface towards PE
ipv4 address 10.0.2.1 255.255.255.0
negotiation auto
!
```

2 OSPF と MPLS LDP を設定します。

```
router ospf 1
router-id 30.30.30.30
nsf cisco
area 0
interface Loopback0
!
interface GigabitEthernet0/0/0/1
!
interface GigabitEthernet0/0/0/8
!
!

!
mpls ldp
router-id 30.30.30.30
graceful-restart
interface GigabitEthernet0/0/0/1
!
interface GigabitEthernet0/0/0/8

 !

!
```

3 DHD への MCLAG バンドルを設定します。

```
interface Bundle-Ether10
description interface towards DHD
lacp switchover suppress-flaps 100
mlacp iccp-group 1
mlacp switchover recovery-delay 60
mlacp port-priority 2
mac-address 0.aaaa.1111
bundle wait-while 0
l2transport

 !

!
interface GigabitEthernet0/0/0/28
bundle id 10 mode active

!
```

4 MCLAG の冗長グループを設定します。

```
redundancy
iccp
group 1
mlacp node 2
mlacp system mac 0000.aaaa.0000
mlacp system priority 1
member
neighbor 20.20.20.20
!
backbone
interface GigabitEthernet0/0/0/8
   !
  !
 !
!
```

5 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile p1
ttl-check disable
router-alert-check disable
!
```

6 PE 方向の DHD と PW に対する MCLAG バンドルを含む L2VPN BD 内の IGMP スヌーピングをイネーブルにします。

```
l2vpn
bridge group bg1
bridge-domain bg1_bd1
igmp snooping profile p1
interface Bundle-Ether10
!
vfi bg1_bd1_vfi
neighbor 40.40.40.40 pw-id 1

    !

   !

  !

 !

!
```

PE：

1 インターフェイスを設定します。

```
interface Loopback0
ipv4 address 40.40.40.40 255.255.255.255
!
interface GigabitEthernet0/0/0/8
description interface towards POA1
ipv4 address 10.0.1.2 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/0/0/9
description interface towards POA2
ipv4 address 10.0.2.2 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/0/0/20
description interface towards Multicast Router
l2transport
!
!
```

2 OSPF と MPLS LDP を設定します。

```
router ospf 1
router-id 40.40.40.40
nsf cisco
area 0
interface Loopback0
!
interface GigabitEthernet0/0/0/8
!
interface GigabitEthernet0/0/0/9
!
!
```

```
!
mpls ldp
router-id 40.40.40.40
graceful-restart

interface GigabitEthernet0/0/0/8
!
interface GigabitEthernet0/0/0/9
!
!
```

3 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile p1
ttl-check disable
router-alert-check disable
!
```

4 マルチキャスト ルータ方向の POA とポートの両方に対する PW を含む L2VPN BD 内の IGMP スヌーピングをイネーブルにします。

```
l2vpn
bridge group bg1
bridge-domain bg1_bd1
igmp snooping profile p1
interface GigabitEthernet0/0/0/20
!
vfi bg1_bd1_vfi
neighbor 20.20.20.20 pw-id 1
!
neighbor 30.30.30.30 pw-id 1
!
!
!
```

ケース 2：アップストリーム MCLAG

トポロジ：マルチキャスト ルータは 2 つの POA に接続されており、順番に PE マルチキャスト ルータに接続します。

1 POA へのバンドルを設定します。

```
interface Bundle-Ether10
description interface towards POAs
ipv4 address 100.0.0.1 255.255.255.0
lacp switchover suppress-flaps 100
bundle maximum-active links 1
!
interface GigabitEthernet0/0/0/28
description interface towards POA1
bundle id 10 mode active
!
interface GigabitEthernet0/0/0/29
description interface towards POA2
bundle id 10 mode active
!
```

2 バンドル インターフェイス上でマルチキャスト ルーティングをイネーブルにします。

```
multicast-routing
address-family ipv4
interface Bundle-Ether10
enable
!
```

```
!
!
```

POA1：

1 インターフェイスを設定します（OSPF および MPLS LDP 用）。

```
interface Loopback0
ipv4 address 20.20.20.20 255.255.255.255
!
interface GigabitEthernet0/2/0/1
description interface towards POA2
ipv4 address 10.0.0.1 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/2/0/8
description interface towards PE
ipv4 address 10.0.1.1 255.255.255.0
negotiation auto
!
```

2 OSPF と MPLS LDP を設定します。

```
router ospf 1
router-id 20.20.20.20
nsf cisco
area 0
interface Loopback0
!
interface GigabitEthernet0/2/0/1
!
interface GigabitEthernet0/2/0/8
!
!
!
mpls ldp
router-id 20.20.20.20
graceful-restart
interface GigabitEthernet0/2/0/1
!
interface GigabitEthernet0/2/0/8
!
!
```

3 DHD への MCLAG バンドルを設定します。

```
interface Bundle-Ether10
description interface towards DHD
lacp switchover suppress-flaps 100
mlacp iccp-group 1
mlacp switchover recovery-delay 60
mlacp port-priority 1
mac-address 0.aaaa.1111
bundle wait-while 0
l2transport
!
!
interface GigabitEthernet0/2/0/29
bundle id 10 mode active
!
```

4 MCLAG の冗長グループを設定します。

```
redundancy
iccp
```

```
group 1
mlacp node 1
mlacp system mac 0000.aaaa.0000
mlacp system priority 1
member
neighbor 30.30.30.30
!
backbone
interface GigabitEthernet0/2/0/8
!
!
!
!
```

5 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile p1
ttl-check disable
router-alert-check disable
!
```

6 PE 方向の DHD と PW に対する MCLAG バンドルを含む L2VPN BD 内の IGMP スヌーピングをイネーブルにします。

```
l2vpn
bridge group bg1
bridge-domain bg1_bd1
igmp snooping profile p1
interface Bundle-Ether10
!
vfi bg1_bd1_vfi
neighbor 40.40.40.40 pw-id 1
!
!
!
!
!
```

POA2：

1 インターフェイスを設定します（OSPF および MPLS LDP 用）。

```
interface Loopback0
ipv4 address 30.30.30.30 255.255.255.255
!
interface GigabitEthernet0/0/0/1
description interface towards POA1
ipv4 address 10.0.0.2 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/0/0/8
description interface towards PE
ipv4 address 10.0.2.1 255.255.255.0
negotiation auto
!
```

2 OSPF と MPLS LDP を設定します。

```
router ospf 1
router-id 30.30.30.30
nsf cisco
area 0
interface Loopback0
!
interface GigabitEthernet0/0/0/1
!
interface GigabitEthernet0/0/0/8
```

```
!
!
!
mpls ldp
router-id 30.30.30.30
graceful-restart
interface GigabitEthernet0/0/0/1
!
interface GigabitEthernet0/0/0/8
!
!
```

3 DHD への MCLAG バンドルを設定します。

```
interface Bundle-Ether10
description interface towards DHD
lacp switchover suppress-flaps 100
mlacp iccp-group 1
mlacp switchover recovery-delay 60
mlacp port-priority 2
mac-address 0.aaaa.1111
bundle wait-while 0
l2transport
!
!
interface GigabitEthernet0/0/0/28
bundle id 10 mode active
!
```

4 MCLAG の冗長グループを設定します。

```
redundancy
iccp
group 1
mlacp node 2
mlacp system mac 0000.aaaa.0000
mlacp system priority 1
member
neighbor 20.20.20.20
!
backbone
interface GigabitEthernet0/0/0/8
!
!
!
```

5 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile p1
ttl-check disable
router-alert-check disable
!
```

6 PE 方向の DHD と PW に対する MCLAG バンドルを含む L2VPN BD 内の IGMP スヌーピングをイネーブルにします。

```
l2vpn
bridge group bg1
bridge-domain bg1_bd1
igmp snooping profile p1
interface Bundle-Ether10
!
vfi bg1_bd1_vfi
neighbor 40.40.40.40 pw-id 1
!
!
!
!
```

PE：

1 インターフェイスを設定します。

```
interface Loopback0
ipv4 address 40.40.40.40 255.255.255.255
!
interface GigabitEthernet0/0/0/8
description interface towards POA1
ipv4 address 10.0.1.2 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/0/0/9
description interface towards POA2
ipv4 address 10.0.2.2 255.255.255.0
negotiation auto
!
interface GigabitEthernet0/0/0/20
description interface towards Host
l2transport
!
!
```

2 OSPF と MPLS LDP を設定します。

```
router ospf 1
router-id 40.40.40.40
nsf cisco
area 0
interface Loopback0
!
interface GigabitEthernet0/0/0/8
!
interface GigabitEthernet0/0/0/9
!
!
!
mpls ldp
router-id 40.40.40.40
graceful-restart
interface GigabitEthernet0/0/0/8
!
interface GigabitEthernet0/0/0/9
!
!
```

3 IGMP スヌーピング プロファイルを設定します。

```
igmp snooping profile p1
ttl-check disable
router-alert-check disable
!
igmp snooping profile p2
mrouter
!
```

4 ホスト方向の POA とポートの両方に対する PW を含む L2VPN BD 内の IGMP スヌーピングをイネーブルにします。 両方の POA への PW にスタティック mrouter ポートを設定します。

```
l2vpn
bridge group bg1
bridge-domain bg1_bd1
igmp snooping profile p1
interface GigabitEthernet0/0/0/20
!
vfi bg1_bd1_vfi
neighbor 20.20.20.20 pw-id 1
```

```
igmp snooping profile p2
!
neighbor 30.30.30.30 pw-id 1
igmp snooping profile p2
!
!
!
!
```

その他の参考資料

関連資料

関連項目	参照先
MPLS VPLS ブリッジの設定	『*Cisco ASR 9000 Series Aggregation Services Router MPLS Configuration Guide*』の「Implementing Virtual Private LAN Services on Cisco IOS XR Software」モジュール
スタートアップ情報	『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』
EFP と EFP バンドルの設定	『*Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration 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
MIB は、IGMP スヌーピングをサポートしません。	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml 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。

RFC

RFC	タイトル
RFC-4541	『Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches』

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 サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。 Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

第 2 章

Cisco IOS XR ソフトウェアでのレイヤ 3 マルチキャスト ルーティングの実装

このモジュールでは、Cisco IOS XR Software を実行している Cisco ASR 9000 シリーズ ルータ でレイヤ 3 マルチキャスト ルーティングを実装する方法について説明します。

マルチキャスト ルーティングは、単一の情報ストリームを場合によっては数千もの企業や家庭に同時に配信することでトラフィックを軽減する、帯域幅節約型のテクノロジーです。 マルチキャスト ルーティングを活用する用途には、ビデオ会議、企業通信、遠隔学習に加えて、ソフトウェア、株価、およびニュースの配信などがあります。

このマニュアルには、IPv4 および IPv6 マルチキャスト ルーティング設定作業と Cisco IOS XR Software の概念についての知識が必要です。

マルチキャスト ルーティングはホストが、ユニキャスト送信のように単一のホストではなく、すべてのホストのサブセットに対してグループ送信として、またはブロードキャスト伝送のようにすべてのホストにパケットを送信できます。 ホストのサブセットは**グループ メンバ**と呼ばれ、224.0.0.0 ～ 239.255.255.255 の IP クラス D アドレス範囲に含まれる 1 つのマルチキャスト グループ アドレスによって識別されます。

マルチキャスト ルーティングに関する詳細な概念情報およびこのモジュールに記載されているマルチキャスト ルーティング コマンドの詳細な説明については、関連資料, （278 ページ）を参照してください。 設定作業を実行する手順の中で出現する可能性のあるその他のコマンドについて記載されたマニュアルを特定するには、『*Cisco ASR 9000 Series Aggregation Services Router Commands Master List*』をオンラインで検索してください。

Cisco ASR 9000 シリーズ ルータ上でのマルチキャスト ルーティング設定機能の履歴

リリース	変更内容
リリース 3.7.2	この機能が導入されました。

リリース	変更内容
リリース 3.9.0	次の機能のサポートが追加されました。 • フローベースのマルチキャスト専用高速再ルーティング（MoFRR）。 • IGMP VRF オーバーライド。
リリース 3.9.1	マルチキャスト VPN 機能のサポートが追加されました。（IPv4 アドレス ファミリの場合）
リリース 4.0.0	Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 ラインカードおよび MVPN ハブ アンド スポークトポロジ上で、IPv4 マルチキャスト ルーティング、マルチキャスト VPN ベーシック、および InterAS オプション A のサポートが追加されました。
リリース 4.0.1	IPv6 マルチキャスト ルーティングのサポートが追加されました。
リリース 4.1.0	グローバル コンテキストのみでの（VRF を除く）ポイントツーマルチポイント トラフィック エンジニアリングを使用したラベル スイッチド マルチキャストのサポートが追加されました。
リリース 4.2.1	次の機能のサポートが追加されました。 • MLDP（マルチキャスト ラベル配布プロトコル）を使用したラベル スイッチド マルチキャスト。 • IPv6 アドレス ファミリのマルチキャスト VPN。 • サテライト nV のサポート。 • マルチキャスト VPN での InterAS のサポート。

- マルチキャスト ルーティングを実装するための前提条件，69 ページ
- マルチキャスト ルーティングの実装に関する情報，69 ページ
- マルチキャスト ルーティングの実装方法，113 ページ
- マルチキャスト専用高速再ルーティング（MoFRR），205 ページ
- ポイントツーマルチポイント トラフィック エンジニアリング ラベル スイッチド マルチキャスト，210 ページ
- IGMP VRF オーバーライドの設定，216 ページ

- ソフトウェアでマルチキャスト ルーティングを実装するための設定例，225 ページ
- その他の参考資料，278 ページ

マルチキャストルーティングを実装するための前提条件

- マルチキャスト ルーティング ソフトウェアのパッケージ インストレーション エンベロープ（PIE）をインストールし、アクティブ化する必要があります。 任意の PIE インストールの詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』を参照してください。
- MLDP の場合、MPLS PIE はインストールする必要があります。
- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。 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。
- このコマンドを使用するには、適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。 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場合は、AAA 管理者に連絡してください。
- IPv4 および IPv6 マルチキャスト ルーティングの設定作業と概要に関する知識が必要です。
- ユニキャスト ルーティングは動作可能でなければなりません。
- マルチキャスト VPN をイネーブルにするには、VPN ルーティングおよび転送（VRF）インスタンスを設定する必要があります。 VRF の設定の詳細については、『Cisco ASR 9000 Series Aggregation Services Router MPLS Layer 3 VPN Configuration Guide』を参照してください。

マルチキャスト ルーティングの実装に関する情報

Cisco IOS XR ソフトウェア マルチキャスト ルーティングの実装でサポートされている主要なプロトコルと機能

表 5：Cisco ASR 9000 シリーズ ルータ上の IPv4 および IPv6 でサポートされる機能，（69 ページ）に、Cisco IOS XR Software の IPv4 および IPv6 マルチキャスト ルーティングでサポートされる機能をリストします。

表 *5*：*Cisco ASR 9000* シリーズ ルータ上の *IPv4* および *IPv6* でサポートされる機能

機能	IPv4 サポート	IPv6 サポート
ダイナミック ホスト登録	Yes（IGMP v1/2/3）	Yes

機能	IPv4 サポート	IPv6 サポート
ホスト、グループ、およびチャネルの明示的なトラッキング	Yes（IGMP v3）	Yes
PIM-SM[2]	Yes	Yes
PIM-SSM	Yes	Yes
PIM-SSM マッピング	Yes	Yes
Auto-RP	Yes	No
マルチキャスト VPN	Yes	Yes
InterAS オプション A	Yes	Yes
BSR	Yes	Yes
BGP	Yes	Yes
MSDP	Yes	No
マルチキャスト NSF	Yes	Yes
OOR の処理	Yes	Yes

2

マルチキャスト ルーティングの機能概要

従来の IP 通信では、ホストはパケットを単一のホスト（ユニキャスト伝送）またはすべてのホスト（ブロードキャスト伝送）に送信できます。 マルチキャストは、第三の方式を提供するものであり、ホストが単一のデータ ストリームをすべてのホストのサブセット（グループ伝送）にほぼ同時に送信できるようにします。 IP ホストはグループ メンバと呼ばれます。

グループ メンバに伝送されるパケットは、単一のマルチキャスト グループ アドレスによって識別されます。 IP ユニキャスト パケットと同様、マルチキャスト パケットは、ベストエフォート型の信頼性を使用してグループに配信されます。

マルチキャスト環境は、送信側と受信側で構成されます。 どのホストも、グループのメンバであるかどうかにかかわらず、グループに送信できます。 ただし、グループのメンバだけがメッセージを受信します。

マルチキャスト アドレスがマルチキャスト グループの受信先として選択されます。 送信者は、データグラムの宛先アドレスとしてグループのすべてのメンバに到達するためにそのグループ アドレスを使用します。

マルチキャスト グループ内のメンバーシップはダイナミックです。ホストはいつでも加入および脱退できます。 マルチキャスト グループ内のメンバの場所または数に制約はありません。 ホストは、一度に複数のマルチキャスト グループのメンバにすることができます。

マルチキャスト グループのアクティブ状態および所属メンバは、グループや時間によって変化し、 マルチキャスト グループを長時間または短時間アクティブにすることもできます。 グループのメンバーシップはいつでも変更可能です。 メンバを含むグループにアクティビティがない場合もあります。

ルータは、直接接続されたサブネットにグループのメンバがあるかどうかを学習するため、インターネットグループ管理プロトコル（IGMP）（IPv4）およびマルチキャストリスナー検出（MLD）（IPv6）を使用します。 ホストは、IGMP または MLD レポート メッセージを送信することにより、マルチキャスト グループに参加します。

多くのマルチメディアアプリケーションには複数の参加者が含まれます。 マルチキャストはその性質上この通信パラダイムに適しています。

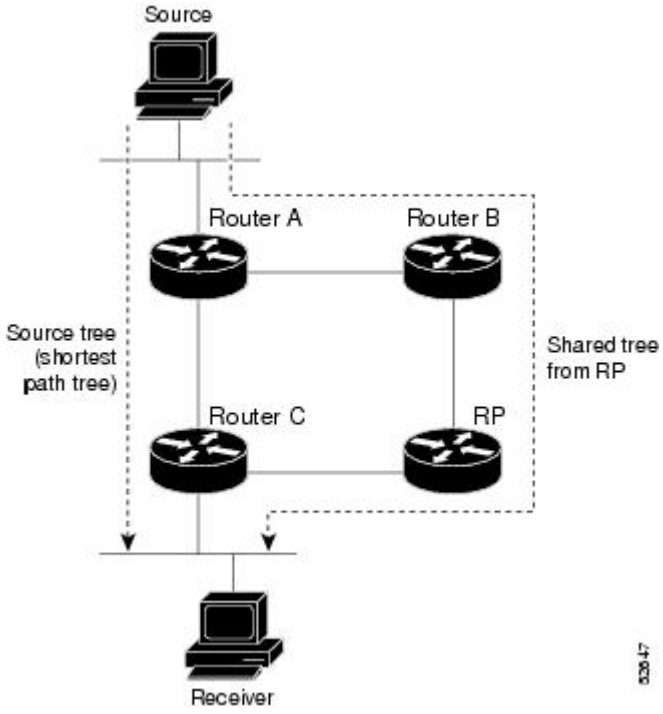
Cisco IOS XR Software マルチキャスト ルーティング実装

Cisco IOS XR Software は、マルチキャスト ルーティングを実装するために次のプロトコルをサポートしています。

- IGMP は、同じ LAN 上のホストとルータ間で、ホストがメンバのマルチキャスト グループを追跡するために使用されます。
- スパース モードの Protocol Independent Multicast（PIM-SM）は、相互に転送されるマルチキャスト パケット、および直接接続されている LAN に転送されるマルチキャスト パケットを追跡するためにルータ間で使用されます。
- Source-Specific Multicast の Protocol Independent Multicast（PIM-SSM）は PIM-SM と類似していますが、IP マルチキャストアドレスを宛先とした特定の送信元アドレス（または特定の送信元アドレスを除くすべてのアドレス）からのパケットを受信する対象をレポートする機能を別途備えています。
- PIM-SSM は IGMPv3 および MLDv2 によって実現されます。 ホストは IGMPv3 および MLDv2 を使用して特定の送信元への関心を示すことができます。 SSM は、動作のためにランデブーポイント（RP）を必要としません。
- 双方向 PIM は、IP マルチキャスト向けルーティング プロトコルの 1 組の Protocol Independent Multicast 形式です。 PIM-BIDIR は、各 PIM ドメイン内の多対多のアプリケーションで使用するよう設計されています。

この図は、マルチキャスト環境で動作している IGMP と PIM-SM を示しています。

図 *1*：*Cisco IOS XR Software* でサポートされるマルチキャスト ルーティング プロトコル



PIM-SM および PIM-SSM

Protocol Independent Multicast（PIM）は、マルチキャスト データ パケットの転送に使用されるマルチキャスト配信ツリーを作成するために使用されるマルチキャスト ルーティング プロトコルです。 PIM は、Multicast Open Shortest Path First（MOSPF）やディスタンス ベクトル マルチキャスト ルーティング プロトコル（DVMRP）などの他のマルチキャスト プロトコルとは異なり、ルーティング テーブルから「独立した」効率的な IP ルーティング プロトコルです。

Cisco IOS XR Softwareは、スパース モードでの Protocol Independent Multicast（PIM-SM）および Source-Specific Multicast での Protocol Independent Multicast（PIM-SSM）をサポートしているため、これらのモードはルータ上で同時に動作できます。

PIM-SMおよびPIM-SSMは、プロトコル メカニズムを大幅に簡略化して配置を容易にすることにより、一対多のアプリケーションをサポートします。

PIM-SM の処理

スパース モードの PIM は、トラフィックに対して明示的な要求がある場合を除いて、各マルチキャストに関与しているルータの数が比較的少なく、これらのルータがグループのマルチキャスト パケットを転送しないときに、マルチキャスト ネットワークで使用されます。

PIM-SM の詳細については、PIM スパース モード, （76 ページ）を参照してください。

PIM-SSM の処理

Source-Specific Multicast 動作の PIM は、受信側から提供されたマルチキャスト グループの送信元アドレスから得た情報を使用して、トラフィックの送信元フィルタリングを実行します。

- デフォルトでは、PIM-SSM は、IPv4 の場合は 232.0.0.0/8 のマルチキャスト グループ範囲で動作し、IPv6 の場合は ff3x::/32（x は有効な範囲）のマルチキャスト グループ範囲で動作します。 これらの値を設定するには、**ssm range** コマンドを使用します。
- PIM-SM 用に設定されているネットワークに SSM を配置する場合、SSM 機能をサポートするラストホップ ルータのみを Cisco IOS XR Software でアップグレードする必要があります。
- SSM 範囲内の MSDP SA メッセージは、受け入れ、生成、転送のいずれも実行されません。

PIM-SM および PIM-SSM の制約事項

SSM との相互運用性

SSM 範囲のアドレスの PIM-SM 動作は、PIM-SSM に変更されます。 このモードでは、ルータで生成されるのは PIM (S,G) の join と prune のメッセージだけであり、(S,G) の RP 共有ツリーや (*,G) の共有ツリー メッセージは生成されません。

IGMP Version

隣接マルチキャスト ルータにマルチキャスト メンバーシップを報告するには、ホストは IGMP を使用し、サブネット上のルータはすべて、同じバージョンの IGMP で設定する必要があります。

Cisco IOS XR Software が動作するルータは自動的にバージョン 1 システムを検出しません。 ルータ IGMP コンフィギュレーション サブモードで **version** コマンドを使用し、IGMP バージョンを設定する必要があります。

インターネット グループ管理プロトコル

Cisco IOS XR Software は、IPv4 上でインターネット グループ管理プロトコル（IGMP）のサポートを提供します。

IGMP は、ホストが関心を持っているマルチキャスト トラフィックを示し、ルータがネットワーク全体でマルチキャスト トラフィックのフローを制御および制限するための方法を提供します。ルータは、IGMP および MLD メッセージ（つまり、ルータのクエリーおよびホスト レポート）を使用して状態を構築します。

同じ送信元からのマルチキャストデータ ストリームを受信する一連のクエリーおよびホストは、マルチキャスト グループと呼ばれます。 ホストでは、IGMP および MLD メッセージを使用して、マルチキャスト グループに加入し、マルチキャスト グループを脱退します。

（注） IGMP メッセージはクラス D の IP アドレスであるグループ アドレスを使用します。 クラス D アドレスの上位 4 ビットは 1110 です。 ホスト グループ アドレスの範囲は、224.0.0.0 ～ 239.255.255.255 です。 アドレス 224.0.0.0 は、どのグループにも割り当てられません。 アドレス 224.0.0.1 は、サブネットのシステムすべてに割り当てられます。 アドレス 224.0.0.2 は、サブネットのルータすべてに割り当てられます。

IGMP のバージョン

IGMP バージョン 1、2、および 3 の要点は次のとおりです。

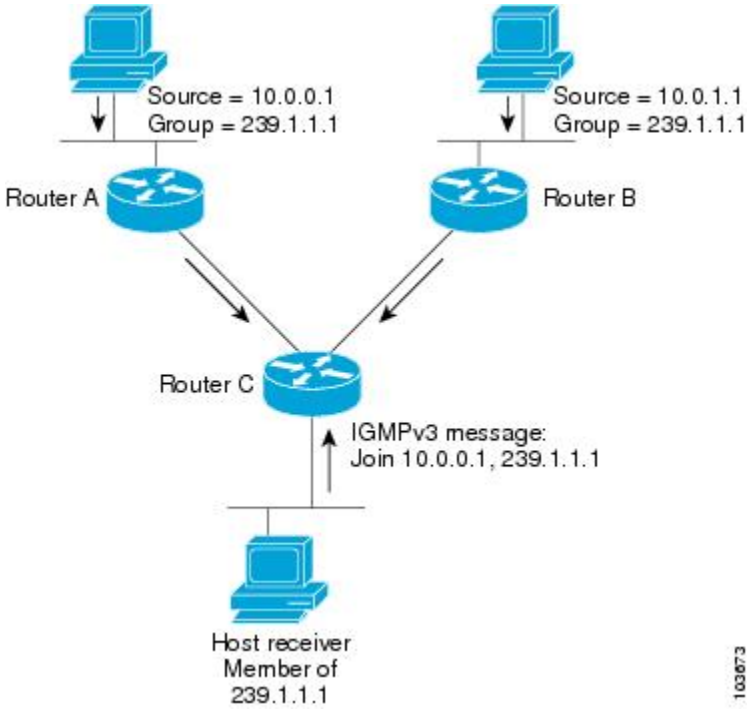
- IGMP バージョン 1 は、どのマルチキャスト グループがアクティブであるかをマルチキャスト ルータが決定できる基本的なクエリー応答メカニズムと、ホストがマルチキャスト グループに加入および脱退できるようにするためのその他のプロセスを提供します。
- IGMP バージョン 2 では、IGMP が拡張され、IGMP クエリー タイムアウトや最大クエリー応答時間などの機能を使用できます。 RFC 2236 を参照してください。
- IGMP バージョン 3 では、マルチキャスト グループのすべての送信元からのトラフィックを要求する代わりに、特定の送信元とグループのペアの参加または脱退が可能です。

IGMP のルーティング例

図 2：IGMPv3 シグナリング，（75 ページ） に、グループ 239.1.1.1 に対してマルチキャスト通信を行う 2 つの送信元 10.0.0.1 および 10.0.1.1 を示します。 レシーバは、グループ 239.1.1.1 宛のトラフィックのうち、送信元 10.0.0.1 からのトラフィックを受信し、送信元 10.0.1.1 からのトラフィックを受信しません。 ホストは、参加する送信元とグループ (S,G) のリストと、参加しない送信元とグループ (S,G) のリストを含む IGMPv3 メッセージを送信する必要があります。 ルータ C は、送信元 10.0.1.1 からのトラフィックをプルーニングするためにこの情報を使用して、送信元 10.0.0.1 のトラフィックだけが

ルータ C に渡されるようにできます。

図 2：***IGMPv3*** シグナリング



（注） IGMP を設定する場合は、サブネット上のすべてのシステムが同じ IGMP バージョンをサポートすることを確認します。 ルータは自動的にバージョン 1 システムを検出しません。 使用しているホストでバージョン 3 がサポートされていない場合は、ルータをバージョン 2 に設定してください。

プロトコル独立マルチキャスト

Protocol Independent Multicast（PIM）は、マルチキャスト ルーティング アップデートを送受信するように設計されたルーティング プロトコルです。 マルチキャストが適切に動作するためには、送信元または RP へのユニキャスト パスを認識している必要があります。 PIM は、ユニキャスト ルーティング プロトコルを使用してこのリバース パス転送（RPF）情報を取得します。 PIM という名前が示すとおり、使用されるユニキャスト プロトコルとは独立して動作します。 PIM は RPF 情報についてルーティング情報ベース（RIB）に依存します。 マルチキャスト Subsequent Address Family Identifier（SAFI）がボーダー ゲートウェイ プロトコル（BGP）で設定されているか、マルチキャストがそのまま設定されている場合、別個のマルチキャスト ユニキャスト RIB は、BGP マルチキャスト SAFI ルート、そのままの情報、およびユニキャスト RIB 内の IGP 情報を使用して作成および設定されます。 そうでない場合、PIM はユニキャスト SAFI RIB から情報を直接取得

します。 マルチキャスト ユニキャスト データベースとユニキャスト データベースは、どちらも PIM の範囲外です。

Cisco IOS XR の PIM の実装は、『RFC 4601 Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification』に基づいています。 詳細については、RFC 4601 およびインターネット技術特別調査委員会（IETF）インターネット ドラフト『Protocol Independent Multicast (PIM): Motivation and Architecture』を参照してください。

（注） Cisco IOS XR Software は PIM-SM、PIM-SSM、および PIM バージョン 2 だけをサポートします。 ネイバーから受信する PIM バージョン 1 hello メッセージは拒否されます。

PIM スパース モード

通常、スパース モードの PIM（PIM-SM）動作は、マルチキャスト ネットワークで比較的少数のルータがマルチキャストに関連する場合に使用されます。 ルータは、トラフィックの明示的な要求がない場合、グループのマルチキャスト パケットを転送しません。 要求は、ツリーのルート ノードに向けてホップバイホップで送信される PIM join メッセージを使用して行われます。 PIM-SM のツリーのルート ノードは、共有ツリーの場合はランデブー ポイント（RP）、最短パス ツリー（SPT）の場合はマルチキャスト送信元に直接接続されているファーストホップ ルータになります。 RP はマルチキャスト グループを追跡し、マルチキャスト パケットを送信する送信元は送信元のファーストホップ ルータによって RP に登録されます。

PIM join がツリーの上位方向に送信されると、要求されたマルチキャスト トラフィックがツリーの下位方向に転送されるように、パス上のルータがマルチキャスト転送ステートを設定します。 マルチキャスト トラフィックが不要になったら、ルータはルート ノードに向けてツリーの上位方向に PIM prune メッセージを送信し、不必要なトラフィックをプルーニング（削除）送信します。 この PIM prune がホップごとにツリーを上位方向に移動する際、各ルータはその転送状態を適切に更新します。 最終的に、マルチキャスト グループまたは送信元に関連付けられている転送ステートは削除されます。 また、prune が明示的に送信されない場合、以降の join メッセージがないと、PIM ステートがタイムアウトし削除されます。

PIM-SM は、WAN リンクの最後に潜在的なメンバがあるマルチキャスト ネットワークに最も適しています。

PIM 送信元固有マルチキャスト

送信元がわかっている多くのマルチキャスト構成では、プロトコル独立型マルチキャスト送信元特定マルチキャスト（PIM-SSM）マッピングは、その単純さから、使用すべき明白なマルチキャスト ルーティング プロトコルの選択です。 PIM-SSM のメリットを享受できる一般的なマルチキャスト構成としては、ETTH スペースなどのエンターテインメント型のソリューションや、静的な転送に完全に依存する金融機関での展開が挙げられます。

PIM-SSM は PIM-SM から派生したものです。 ただし、PIM-SM では、PIM join メッセージに応じて特定のグループに送信するすべての送信元のデータ伝送が可能なのに対し、SSM 機能は、受信先が明示的に加入した送信元からのトラフィックのみをレシーバへ転送します。 PIM join および

prune はトラフィックの送信元に直接送信されるため、RP と共有ツリーは不要で拒否されます。SSM が、帯域利用率を最適化し、不要なインターネット ブロードキャスト トラフィックを拒否するために使用されます。 送信元は、IGMPv3 メンバーシップレポートを使用して対象の受信先により提供されます。

SSM では、データグラムは (S,G) チャネルに基づいて配信されます。 1 つの (S,G) チャネルのトラフィックは、IP 宛先アドレスとして IP ユニキャスト送信元アドレス S とマルチキャスト グループ アドレス G を持つデータグラムで構成されています。 システムは、(S,G) チャネルのメンバになることによって、トラフィックを受信します。 シグナリングは不要ですが、受信先は特定の送信元からのトラフィックを受信する場合は (S,G) チャネルに加入し、トラフィックを受信しない場合はチャネルから脱退する必要があります。 チャネル加入シグナリングでは、IGMP を使用してモード メンバーシップ レポートを含めます。これは、IGMP バージョン 3（IGMPv3）でのみサポートされています。

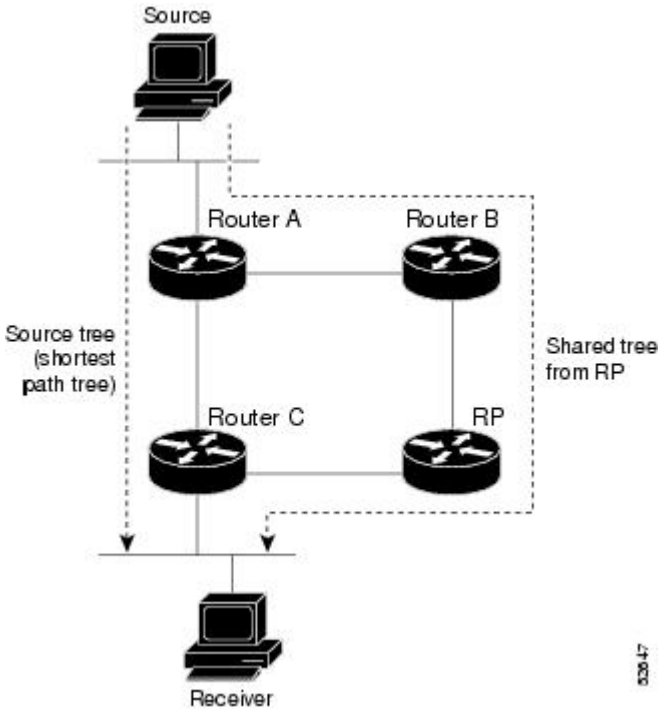
IGMPv3 で SSM を使用するには、マルチキャスト ルータ、アプリケーションが稼働しているホスト、そしてアプリケーション自体が SSM をサポートしている必要があります。 Cisco IOS XR Software では、224.0.0.0 ～ 239.255.255.255 の IP マルチキャスト アドレス範囲の任意のサブセットの SSM 設定を許可します。 SSM 範囲が定義されると、（アプリケーションが明示的な (S,G) チャネル加入を使用するように変更されていない限り）SSM 範囲内でアドレスを使用しようとする場合に既存の IP マルチキャスト レシーバ アプリケーションはトラフィックを受信しません。

PIM 共有ツリーおよび送信元ツリー（最短パス ツリー）

PIM-SM では、特定のグループにデータを送信する送信元と、そのグループに join を送信する受信先をブリッジングするために、ランデブー ポイント（RP）が使用されます。 状態の初期設定では、対象の受信先は、RP をルートとする単一のデータ配布ツリーを通じて、送信側からグループへのデータを受信します。 このタイプの配布ツリーは、共有ツリーまたはランデブーポイント ツリー（RPT）と呼ばれます（図 3：共有ツリーおよび送信元ツリー（最短パス ツリー），（78

ページ） を参照）。 送信側からのデータは、RP に配信され、その共有ツリーに加入しているグループ メンバに配布されます。

図 3：共有ツリーおよび送信元ツリー（最短パス ツリー）



spt-threshold infinity コマンドが設定されていない場合、この初期状態は、トラフィックがリーフ ルータ（受信先ホストに最も近い指定ルータ）で受信されるとすぐに別の状態になります。 リーフ ルータが RPT 上の RP からトラフィックを受信すると、ルータはトラフィックを送信する送信元で開始されるデータ配信ツリーに切り替えを開始します。 このタイプの配信ツリーは、**最短パス ツリー**または**送信元ツリー**と呼ばれます。 デフォルトでは、Cisco IOS XR Software が送信元から最初のデータ パケットを受信すると、送信元ツリーに切り替わります。

次に、共有ツリーから送信元ツリーに切り替わるプロセスの詳細を示します。

1. 受信先がグループに加入します。リーフ ルータ C が RP に join メッセージを送信します。
2. RP がルータ C へのリンクを発信インターフェイス リストに登録します。
3. 送信元がデータを送信します。ルータ A が Register にデータをカプセル化し、それを RP に送信します。
4. RP が共有ツリーの下位方向のルータ C にデータを転送し、送信元に join メッセージを送信します。 この時点で、データは RP に 2 回（カプセル化された状態で 1 回、ネイティブの状態で 1 回）着信する可能性があります。
5. データがネイティブ状態（カプセル化されていない状態）で RP に着信すると、RP は register-stop メッセージをルータ A に送信します。

6 デフォルトでは、ルータ C は、最初のデータ パケットを受信した時点で、送信元に join メッセージを送信します。

7 ルータ C が (S,G) でデータを受信すると、ルータ C は共有ツリーの上位方向にある送信元に prune メッセージを送信します。

8 RP が (S,G) の発信インターフェイスからルータ C へのリンクを削除します。 RP が送信元への prune メッセージをトリガーします。

送信元および RP に join および prune メッセージが送信されます。 これらのメッセージはホップバイホップで送信され、送信元または RP に向かうパス上の各 PIM ルータによって処理されます。 register および register-stop メッセージは、ホップバイホップで送信されません。 これらのメッセージは、送信元に直接接続されている指定ルータと、そのグループの RP の間で、直接ユニキャスト通信を使用して交換されます。

ヒント **spt-threshold infinity** コマンドを使用すると、最短パス ツリー（SPT）に切り替わらないようにルータを設定できます。

multicast-intact

multicast-intact 機能を使用すると、Interior Gateway Protocol（IGP）ショートカットがルータに設定されアクティブな場合に、マルチキャスト ルーティング（PIM）を実行できます。 Open Shortest Path First バージョン 2（OSPFv2）と Intermediate System-to-Intermediate System（IS-IS）の両方が multicast-intact 機能をサポートしています。 マルチプロトコル ラベル スイッチング トラフィック エンジニアリング（MPLS-TE）と IP マルチキャストの共存は、Cisco IOS XR Software により、**mpls traffic-eng multicast-intact** IS-IS または OSPF ルータ コマンドを使用してサポートされます。 IS-IS および OSPF コマンドを使用して multicast intact を設定する方法の詳細については『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』を参照してください。

IGP の multicast-intact は、マルチキャスト ルーティング プロトコル（PIM）と IGP ショートカットがルータで設定されている場合にイネーブルにできます。 IGP ショートカットは IGP に公開される MPLS トンネルです。 IGP は、トンネルの出力ルータの下流にある宛先にこれらのトンネル上で IP トラフィックをルーティングします（SPF の観点から）。 PIM は PIM join を伝搬するために IGP ショートカットを使用できません。これは、リバース パス転送（RPF）が単方向トンネルで動作できないためです。

IGP の multicast-intact をイネーブルにすると、IGP は PIM で使用するための、並行または代替の等コスト ネクストホップをパブリッシュします。 これらのネクスト ホップは **mcast-intact ネクスト ホップ**と呼ばれます。 mcast-intact ネクスト ホップは次の属性を持ちます。

- IGP のショートカットが含まれていないことが保証されます。
- ユニキャスト ルーティングに使用されませんが、PIM によってのみ PIM 送信元への IPv4 ネクスト ホップの検索に使用されます。
- 転送情報ベース（FIB）にパブリッシュされません。

- multicast-intact が IGP でイネーブルになっている場合、リンクステート アドバタイズメントで学習したすべての IPv4 宛先は、設定された等価コスト mcast-intact ネクストホップとともに RIB にパブリッシュされます。 この属性は、ネイティブ ネクストホップに IGP のショートカットがない場合でも適用されます。
- IS-IS では、ネイティブおよび mcast-intact ネクスト ホップを一緒にカウントすることで、max-paths 制限が適用されます。 （OSPFv2 では動作が若干異なります）。

指定ルータ

Cisco ルータは、LAN セグメント上に複数のルータが存在する場合、PIM-SM を使用してマルチキャスト トラフィックを転送し、選択プロセスに従って指定ルータ（DR）を選択します。

指定ルータは、PIM register メッセージ、PIM join メッセージ、および PIM prune メッセージを RP に送信し、ホスト グループ メンバーシップに関する情報を通知します。

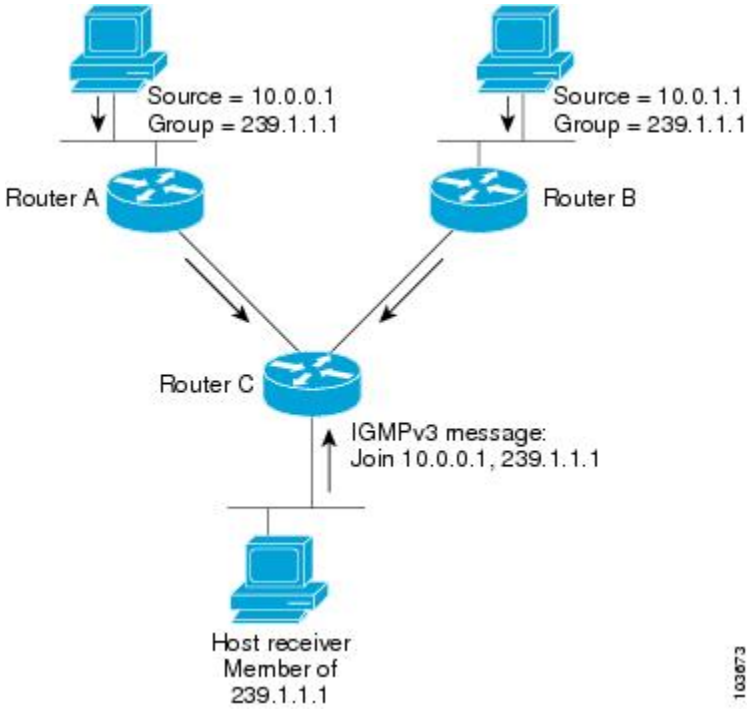
LAN 上に複数の PIM-SM ルータが存在する場合は、指定ルータを選択して、接続されているホストに対するマルチキャスト トラフィックの重複を回避する必要があります。 **dr-priority** コマンドを使用して DR の選択を強制することを選択しない限り、最も大きい IP アドレスの PIM ルータが LAN の DR になります。 DR プライオリティ オプションを使用すると、LAN セグメント上の各ルータの DR プライオリティ（デフォルトのプライオリティ = 1）を指定して、最もプライオリティの高いルータが DR として選択されるようにすることができます。 LAN セグメント上のすべてのルータのプライオリティが同じ場合にも、最上位 IP アドレスを持つルータが選択されます。

図 4：マルチアクセス セグメントでの代表ルータの選択, （81 ページ） に、マルチアクセス セグメントでの動作を示します。 ルータ A（10.0.0.253）およびルータ B（10.0.0.251）は、ホスト A（10.0.0.1）をグループ A のアクティブな受信先として使用する共通のマルチアクセス イーサネット セグメントに接続されます。 明示的な join モデルが使用されるため、DR として動作しているルータ A のみが、グループ A の共有ツリーを構築するために RP に join を送信します。 ルータ B も RP への (*,G) join の送信を許可されている場合は、パラレル パスが作成され、ホスト A が重複マルチキャスト トラフィックを受信します。 ホスト A がグループにマルチキャスト トラフィックを送信し始めたら、DR は register メッセージを RP に送信する役割を担います。 両方のルータに役割が割り当てられている場合は、RP が重複マルチキャスト パケットを受信します。

DR で障害が発生した場合、PIM-SM はルータ A の障害を検出し、フェールオーバー DR を選択する手段を提供します。 DR（ルータ A）が動作不能になると、ルータ A との隣接ルータとの隣接関係がタイムアウトしたときに、ルータ B はその状況を検出します。 ルータ B はホスト A から IGMP メンバーシップ レポートを受けているため、このインターフェイスでグループ A の IGMP ステートをすでに持ち、新しい DR になると即座に RP に join を送信します。 この段階で、ルータ B を使用する共有ツリーの新しいブランチの下位方向へのトラフィック フローが再び確立されます。 また、ホスト A がトラフィックを送信していた場合、ルータ B は、ホスト A から次のマルチキャスト パケットを受信した直後に、新しい登録プロセスを開始します。 このアクションがトリガーとなって、RP は、ルータ B を使用する新しいブランチを介して、ホスト A への SPT に加入します。

ヒント　2 つの PIM ルータが直接接続されている場合、これらのルータはネイバーになります。 PIM ネイバーを表示するには、EXEC モードで **show pim neighbor** コマンドを使用します。

図 *4*：マルチアクセス セグメントでの代表ルータの選択



（注）　DR 選択プロセスは、マルチアクセス LAN のみで必要です。 ホストに直接接続されているラストホップ ルータが DR です。

ランデブー ポイント

PIM がスパース モードで設定されている場合は、ランデブー ポイント（RP）として動作する 1 つ以上のルータを選択する必要があります。 ランデブー ポイントは、図 3：共有ツリーおよび送信元ツリー（最短パス ツリー），（78 ページ） に示すように、共有配信ツリーの選択したポイントに配置された単一の共通ルートです。 ランデブー ポイントは各ボックスで静的に設定するか、ダイナミック メカニズムによって学習できます。

PIM DR は、共有ツリーの下位方向に配布するために、直接接続されているマルチキャスト送信元からランデブー ポイントにデータを転送します。 データは次の 2 つの方法のいずれかを使用してランデブー ポイントに転送されます。

- register パケットにカプセル化され、DR として動作しているファーストホップ ルータによってランデブー ポイントに直接ユニキャストで送信されます
- ランデブー ポイント自体が送信元ツリーに参加している場合、リバース パス転送, （84 ページ）で説明するように、RPF 転送アルゴリズムによってマルチキャストで転送されます。

ランデブーポイントアドレスは、パケットをグループに送信するホストの代わりに、ファーストホップ ルータで PIM register メッセージを送信するために使用されます。 また、ラストホップ ルータでも、PIM join および prune メッセージをランデブー ポイントに送信してグループ メンバーシップについて通知するために使用されます。 すべてのルータ（ランデブー ポイント ルータを含む）でランデブー ポイント アドレスを設定する必要があります。

1 つの PIM ルータを複数のグループのランデブー ポイントにすることができます。 1 つの PIM ドメイン内で一度に使用できるランデブー ポイント アドレスは、1 つだけです。 アクセス リストで指定されている条件によって、ルータがどのグループのランデブー ポイントであるかが判別されます。

ランデブー ポイントとして動作する PIM ルータを手動で設定するか、Auto-RP または BSR を設定することで、ランデブー ポイントがグループから RP へのマッピングを自動的に学習するように指定できます。 （詳細については、後述するAuto-RP, （82 ページ）のセクションと、PIM ブートストラップ ルータ, （83 ページ）を参照してください）。

Auto-RP

自動ルート処理（Auto-RP）は、PIM ネットワークにおけるグループから RP へのマッピングの配信を自動化する機能です。 この機能には次の利点があります。

- ネットワーク内で複数の RP を使用し、複数のグループ範囲を処理する作業が簡単になります。
- 異なる RP 間で負荷を分割できます。
- グループに加入するホストの場所に従った RP の調整を容易にします。
- 接続の問題の原因となる、矛盾した手動 RP 設定を回避できます。

複数の RP を使用して、異なるグループ範囲にサービスを提供したり、互いにホット バックアップとしての役割を果たしたりできます。 Auto-RP が機能するよう、特定のグループ範囲の RP として動作できることを通知できるように、候補 RP としてルータを設定します。 また、RP 通知メッセージを候補 RP から受信して競合を解決する RP マッピング エージェントとしてルータが指定されている必要があります。 RP マッピング エージェントは、グループから RP への一貫したマッピングを残りのすべてのルータに送信します。 これにより、すべてのルータは、サポート対象のグループに使用する RP を自動的に決定します。

ヒント　デフォルトでは、特定のグループ アドレスが静的な RP 設定によるグループから RP へのマッピングに含まれており、かつ Auto-RP または PIM BSR を使用して検出される場合、Auto-RP または PIM BSR の範囲が優先されます。 デフォルトを無効にし、RP マッピングのみを使用するには、**rp-address override** キーワードを使用します。

（注）　PIM をスパース モードに設定し、Auto-RP を設定しない場合は、RP を静的に設定する必要があります（スタティック RP の設定と下位互換性の許可, （120 ページ）を参照）。 ルータ インターフェイスがスパース モードに設定されている場合、Auto-RP グループに対してすべてのルータが 1 つのスタティック RP アドレスで設定されているときは、引き続き Auto-RP グループを使用できます。

（注）　Auto-RP は VRF インターフェイスではサポートされていません。 Auto-RP Lite を使用すると、CE ルータで Auto-RP を設定できます。 これにより、VRF インターフェイスを持つ PE ルータが Auto-RP 検出を中継し、コアを通じて、最終的にリモート CE にメッセージを送信できます。 Auto-RP は IPv4 アドレス ファミリのみでサポートされます。

PIM ブートストラップ ルータ

PIM ブートストラップ ルータ（BSR）は、Auto-RP プロセスを簡素化する、フォールト トレラントで自動的な RP 検出と配信メカニズムを提供します。 この機能はデフォルトでイネーブルになり、ルータはグループから RP へのマッピングを動的に学習できます。

PIM は BSR を使用して各グループ プレフィックスの RP 設定情報を検出し、PIM ドメイン内のすべてのルータにアナウンスします。 これは、Auto-RP によって行われるのと同じ機能ですが、BSR は PIM バージョン 2 仕様の一部です。 BSR メカニズムは、Cisco ルータ上の Auto-RP と相互運用します。

シングル ポイント障害を回避するために、1 つの PIM ドメインに複数の候補 BSR を設定できます。 BSR は候補 BSR の中から自動的に選択されます。 候補はブートストラップ メッセージを使用して最もプライオリティの高い BSR を検出します。 プライオリティの高い候補は、PIM ドメイン内のすべての PIM ルータに、BSR であると通知を送信します。

候補 RP として設定されたルータは、BSR に、各自が担当するグループ範囲をユニキャストします。 BSR はブートストラップ メッセージにこの情報を含め、ドメイン内のすべての PIM ルータに広めます。 この情報に基づいて、すべてのルータが特定の RP にマルチキャスト グループをマッピングできます。 ルータがブートストラップ メッセージを受信する限り、RP マップは最新になります。

リバース パス転送

リバース パス転送（RPF）は、マルチキャスト データグラムの転送に使用されるアルゴリズムです。 これは、次のように機能します。

- ルータで送信元へのユニキャスト パケットの送信に使用しているインターフェイスでデータグラムを受信すると、パケットは RPF インターフェイスに着信しています。
- パケットが RPF インターフェイスに着信した場合、ルータは、マルチキャスト ルーティング テーブル エントリの発信インターフェイス リストに存在するインターフェイスにパケットを転送します。
- パケットが RPF インターフェイスに着信しない場合、パケットはループを回避するためにサイレントにドロップされています。

PIM では、送信元ツリーと RP をルートとする共有ツリーの両方を使用してデータグラムを転送します。RPF チェックは、次のようにそれぞれ異なる方法で実行されます。

- PIM ルータのマルチキャスト ルーティング テーブル内に (S,G) エントリがある場合（送信元ツリー ステートである場合）、マルチキャスト パケットの送信元の IP アドレスに対して RPF チェックが実行されます。
- PIM ルータに明示的な送信元ツリー ステートがない場合、共有ツリー ステートと見なされます。 ルータは、メンバがグループに加入したときにわかる RP のアドレスに対して RPF チェックを実行します。

空間モード PIM では、RPF ルックアップ機能を使用して、join および prune の送信先を決定します。 (S,G) Join メッセージ（送信元ツリー ステート）は送信元に向け送信されます。 (*,G) Join メッセージ（共有ツリー ステート）は RP に向け送信されます。

マルチキャスト VPN

マルチキャスト VPN（MVPN）を使用すると、MPLS ネットワーク経由でマルチキャスト サポートを動的に提供できます。 MVPN では、プロバイダーが VPN でマルチキャスト トラフィックをサポートできるようにするのに役立つ、追加の一連のプロトコルと手順が導入されています。

MCAST VPN トラフィックをコア ネットワーク経由で転送する方法には、次の 2 つがあります。

- Rosen GRE（ネイティブ）：MVPN は、一意のマルチキャスト配信ツリー（MDT）転送で GRE を使用することにより、コア ネットワーク内のネイティブな IP マルチキャストのスケーラビリティを可能にします。 MVPN では、VPN ルーティング/転送テーブル（VRF）にマルチキャスト ルーティング情報が導入されており、マルチキャスト VRF が作成されます。 Rosen GRE では、プロバイダーのコアで PIM プロトコルがイネーブルになるように、MCAST カスタマー パケット（c パケット）がプロバイダー MCAST パケット（p パケット）にカプセル化され、コア内の p パケットの転送には mrib/mfib が使用されます。
- MLDP の場合（Rosen、パーティション）：MVPN を使用すると、サービス プロバイダーは MPLS VPN 環境でマルチキャスト トラフィックを設定およびサポートできます。 このタイ

プでは、個々の VPN ルーティングおよび転送（VRF）インスタンスでのマルチキャスト パケットのルーティングおよび転送がサポートされ、サービス プロバイダーのバックボーン全体にわたって VPN マルチキャスト パケットを転送するためのメカニズムも提供されます。MLDP の場合は、通常のラベル スイッチ パス転送が使用されるため、コアが PIM プロトコルを実行する必要はありません。 このシナリオでは、c パケットは MPLS ラベル内にカプセル化され、転送はユニキャストの場合と同様に MPLS ラベル スイッチド パス（LSP）に基づいて行われます。

上のどちらのタイプでも、MVPN サービスにより、ソースとレシーバが異なるサイトに配置された Protocol Independent Multicast（PIM）ドメインを構築できます。

複数の分散したサイトがあるカスタマーにレイヤ 3 マルチキャスト サービスを提供する場合は、サービス プロバイダーはプロバイダー ネットワーク経由でカスタマーのマルチキャスト トラフィックを伝送するセキュアかつスケーラブルなメカニズムを求めます。 マルチキャスト VPN（MVPN）は、BGP/MPLS VPN のようなネイティブ マルチキャスト テクノロジーを使用して共有サービス プロバイダー バックボーンを介して、このようなサービスを提供します。

マルチキャスト VPN は、すべてのイーサネット ベースのラインカードに加えて、Cisco IOS XR ソフトウェア リリース 4.0 から Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 カードでもサポートされます。 Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 を使用すると、Cisco ASR 9000 シリーズ ルータ は、イーサネット ネットワーク用に主に設計されたルータで複数レガシー サービス（TDM や ATM のような）をサポートできます。 Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 は QFP ベースであるため、Cisco ASIC が提供する柔軟性およびサービス スケールと、Cisco IOS XR ソフトウェアの信頼性を備えています。

MVPN は、マルチキャスト ドメイン（MD）の概念を採用するときに MPLS VPN テクノロジーをエミュレートします。その際、プロバイダー エッジ（PE）ルータは、同一カスタマー VPN に接続している他の PE ルータとの仮想 PIM ネイバー接続を確立します。 これらの PE ルータはプロバイダー ネットワーク上のセキュアな仮想マルチキャスト ドメインを形成します。 マルチキャスト トラフィックは、専用プロバイダーネットワークを通過しているかのように、サイト間をコア ネットワーク上で伝送されます。

VPN ルーティングおよび転送（VRF）インスタンスごとに個別のマルチキャスト ルーティングおよび転送テーブルが保持され、トラフィックは、サービス プロバイダーのバックボーン全体にわたって VPN トンネル経由で送信されます。

マルチキャスト VPN ルーティングおよび転送

ある VPN のトラフィックと別の VPN のトラフィックを分離できるように、VPN ごとに専用のマルチキャスト ルーティングおよび転送テーブルが作成されます。

VPN 固有のマルチキャスト ルーティングおよび転送データベースは、**MVRF** と呼ばれます。 PE ルータで、MVRF は、マルチキャストが VRF 用にイネーブルにされたときに作成されます。Protocol Independent Multicast（PIM）プロトコルとインターネット グループ管理プロトコル（IGMP）プロトコルは MVRF のコンテキストで動作し、MVRF プロトコル インスタンスによって作成されたすべてのルートは、対応する MVRF に関連付けられます。 VPN 固有のプロトコル ステートを保持する VRF に加え、PE ルータにはグローバル VRF インスタンスが常に保持され、プロバイダー ネットワークのすべてのルーティングおよび転送情報が含まれます。

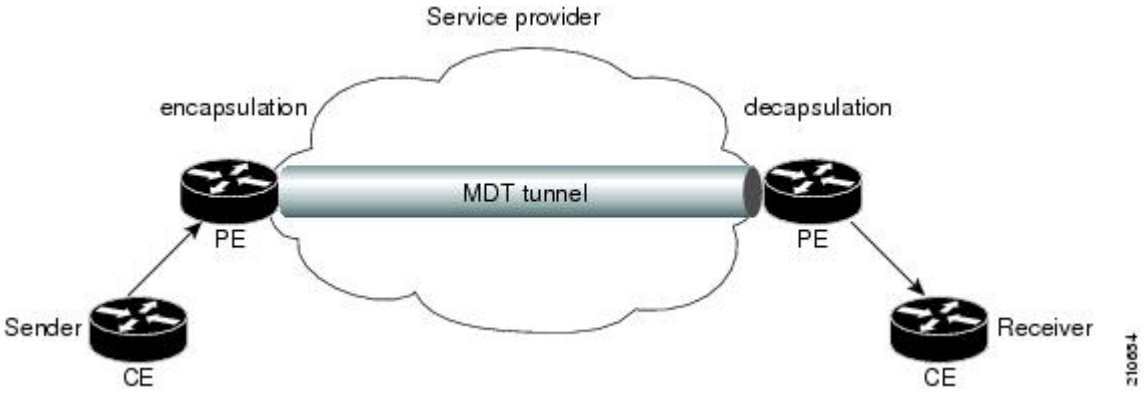
マルチキャスト配信ツリー トンネル

マルチキャスト配信ツリーは（MDT）、プロバイダー ネットワークを介して複数のカスタマー サイトにまたがることができ、1 つの送信元から複数の受信先にトラフィックを転送できます。MLDP については、MDT トンネル ツリーは、Labeled MDT（LMDT）とも呼ばれます。

入力 PE ルータでカスタマー エッジ（CE）ルータから送信されたマルチキャスト パケットの安全なデータ転送は、プロバイダー ヘッダーにパケットをカプセル化し、コアを介してパケットを送信することによって実現されます。 出力 PE ルータでは、カプセル化されたパケットはカプセル化が解除されて、CE 受信ルータに送信されます。

マルチキャスト配信ツリー（MDT）トンネルはポイントツーマルチポイントです。 MDT トンネル インターフェイスは、MVRF がマルチキャスト ドメインにアクセスするために使用するインターフェイスです。 これは MVRF とグローバル MVRF をつなぐ通路と見なすことができます。MDT トンネル インターフェイスに送信されるパケットは、複数の受信ルータで受信されます。MDT トンネル インターフェイスに送信されたパケットはカプセル化され、MDT トンネル インターフェイスから受信したパケットはカプセル化が解除されます。

図 *5*：*MDT* トンネル インターフェイス上での仮想 *PIM* ピア接続



プロバイダー ヘッダーにマルチキャスト パケットをカプセル化することにより、PE ルータがパケットの送信元を引き続き認識せずに済みます。プロバイダー エラーを通過するすべての VPN パケットはネイティブ マルチキャスト パケットとして認識され、コア ネットワーク内のルーティング情報に基づいてルーティングされます。 MVPN をサポートするには、PE ルータは、ネイティブ マルチキャスト ルーティングのみをサポートする必要があります。

MVPN は、まばらに受信先が分散した広帯域アプリケーション用の最適化された VPN トラフィックの転送をサポートします。 専用のマルチキャスト グループを特定の送信元からのパケットのカプセル化に使用でき、該当する受信先に接続されている PE ルータだけにトラフィックを送信するように最適化された MDT を作成できます。 これは**データ MDT** と呼ばれます。

マルチキャスト VPN での InterAS のサポート

マルチキャスト VPN Inter-AS サポート機能によって、サービス プロバイダーは、複数の自律システムにまたがる VPN サイトにマルチキャスト接続を提供できます。 この機能は MLDP プロファ

イルに追加され、それにより、マルチキャスト VPN（MVPN）に使用されるマルチキャスト配信ツリー（MDT）が複数の自律システムにまたがることができるようになります。

次の 2 つのタイプの MVPN Inter-AS 導入シナリオがあります。

- シングルプロバイダー方式の Inter-AS：内部ネットワークが複数の自律システムで構成されたサービス プロバイダー。
- イントラプロバイダー方式の Inter-AS：Inter-AS サポートを提供するためにネットワークの調整が必要な複数のサービス プロバイダー。

2 つの自律システム間でマルチキャスト VPN を確立するには、MDT のデフォルト トンネルを 2 台の PE ルータ間で設定する必要があります。 PE ルータは、設定された MDT デフォルト グループを結合することでこれを達成します。 この MDT デフォルト グループは PE ルータで設定され、VPN ごとに一意です。 PIM は、グループのモード（PIM SSM、またはスパース モード）に基づいて join を送信します。

MVPN Inter-AS サポートの利点

MVPN Inter-AS サポート機能には、サービス プロバイダーにとって次の利点があります。

- MPLS レイヤ 3 VPN サービスにおいてマルチキャストが複数のサービス プロバイダーにわたる必要がある顧客にとって、マルチキャストのカバレッジを向上させる。
- 企業の合併や買収の場合など、既存の MVPN サービスを別の MVPN サービスと統合する。

InterAS オプション A

InterAS オプション A は基本的なマルチキャスト VPN の設定オプションです。 このオプションでは、PE ルータは部分的に各自律システム（AS）で自律システム境界ルータ（ASBR）の役割を担います。 各 AS のこのような PE ルータは、複数の VRF 処理サブインターフェイスで直接接続されます。 MPLS ラベル配布プロトコルは、これらの InterAS ピアリング PE ルータ間で実行する必要はありません。 ただし、IGP または BGP プロトコルが VRF の下のルート配布で使用できます。

オプション A モデルは、異なる自律システムの PE ルータ間の直接の接続を想定しています。 PE ルータは複数の物理または論理インターフェイスによって接続され、各インターフェイスは特定の VPN に関連しています（VRF インスタンスを通して）。 したがって、各 PE ルータは隣接 PE ルータをカスタマー エッジ（CE）ルータと同様に扱います。 各自律システムでのルート再配布には標準的なレイヤ 3 MPLS VPN メカニズムが使用されます。つまり、各 PE は、外部 BGP（eBGP）を使用して相互にラベルなし IPv4 アドレスを配布します。

（注） オプション A を使用すると、サービス プロバイダーが各自律システムを隔離できます。 これは、2つのネットワーク間のルーティング交換やセキュリティの制御を向上します。ただし、オプション A は、すべての AS 間接続オプションで最もスケーラブルでないオプションと考えられています。

InterAS オプション B

InterAS オプション B は、ASBR 間の VPNv4 ルートの交換をイネーブルにするモデルです。 このモデルではまた、BGP MVPN アドレス ファミリも配布します。 このモデルでは、PE ルータが内部 BGP（iBGP）を使用して、ラベル付き VPNv4 ルートを ASBR か、または ASBR がクライアントになっているルート リフレクタのどちらかに再配布します。 これらの ASBR は、マルチプロトコル eBGP（MP-eBGP）を使用して、VPNv4 ルートをローカル自律システムにアドバタイズします。 MP-eBGP は、VPNv4 プレフィックスおよびラベル情報をサービス プロバイダーの境界を超えてアドバタイズします。 アドバタイズする ASBR ルータは、VPNv4 ルートをアドバタイズする前に、ローカル自律システム内の発信元の PE ルータおよび VPN 宛先に到達するために使用する 2 レベルのラベル スタックを、ローカルに割り当てられたラベルに置き換えます。 この置き換えが実行されるのは、2 つのサービス プロバイダーの間でアドバタイズされるすべてのルートのネクスト ホップ属性が ASBR ルータのピアリング アドレスにリセットされ、それによって ASBR ルータがアドバタイズされたルートのラベル スイッチド パス（LSP）の終端地点になるためです。 入力 PE ルータと出力 PE ルータの間の LSP を保持するために、ASBR ルータは、ローカル VPN ネットワーク内のルートのラベル スタックを識別するために使用されるローカル ラベルを割り当てます。 この新しく割り当てられたラベルは、隣接するサービス プロバイダーからプレフィックスに向けて送信されるパケットに設定されます。

（注） オプション B では、サービス プロバイダーは、オプション A よりも高度に変更できるという利点が追加された両方の自律システムを隔離できます。

InterAS オプション B モデルでは、BGP-AD プロファイルのみがサポートされています。

- BGP-AD を使用した MLDP MS-PMSI MP2MP（プロファイル 4）
- BGP-AD を使用した、または使用しない Rosen GRE（プロファイル 9）

（注） プロファイル 9 は、IGP にルート アドレスをリークする場合のみサポートされています。

（注） BGP-AD を使用した MLDP MS-PMSI MP2MP（プロファイル 5）はサポートされていません。

InterAS オプション C

InterAS オプション C を使用すると、マルチホップ eBGP ピアリング セッションを使用してルータ リフレクタ（RR）間で VPNv4 ルートを交換できます。 このモデルでは、異なる自律システムの RR 間の VPNv4 ルートの MP-eBGP 交換が、対応する ASBR ルータ間のこれらのルートの交換のためのネクスト ホップと結合されます。 このモデルではまた、VPNv4 とともに BGP MVPN アドレス ファミリも配布します。 このモデルでは、VPNv4 ルートが保持されることも、ASBR によって配布されることも許可されません。 ASBR は、PE ルータへのラベル付き IPv4 ルートを自身の自律システム内に保持し、eBGP を使用してこれらのルートを他の自律システムに配布しま

す。 中継自律システムでは、ASBR が eBGP を使用してラベル付き IPv4 ルートを渡すため、入力 PE ルータから出力 PE ルータへの LSP が作成されます。

オプション C モデルでは、異なる自律システムの RR が直接接続されていないため、マルチホップ機能を使用して MP-eBGP ピアリング セッションを確立できるようにしています。 RR はまた、VPNv4 ルートのネクストホップ属性を隣接する自律システムにアドバタイズするときに、これらの自律システムがアドバタイズする宛先へのトラフィックを引き込まないため、これらの属性をリセットしません。それにより、ネクストホップの交換のイネーブル化が必須になります。 これらは、単なるソース PE とレシーバ PE の間の中継ステーションです。 このため、VPNv4 の PE ルータ ネクスト ホップ アドレスは、ASBR ルータ間で交換されます。 自律システム間のこれらのアドレスの交換は、自律システム間で PE ルータ /32 アドレスを再配布するか、または BGP ラベル配布を使用することによって実現されます。

（注） オプション C は通常、各自律システムが、グローバルな自律システムを備えたグローバルなレイヤ 3 MPLS VPN サービス プロバイダーなどのより包括的な機関に同じく属している場合にのみ配備されます。

InterAS オプション C モデルでは、次のプロファイルがサポートされています。

- BGP-AD を使用しない Rosen MLDP（プロファイル 1）
- BGP-AD を使用した MLDP MS-PMSI MP2MP（プロファイル 4）
- BGP-AD を使用した MLDP MS-PMSI MP2MP（プロファイル 5）
- MLDP VRF インバンド シグナリング（プロファイル 6）
- BGP-AD を使用した Rosen GRE（プロファイル 9）

MVPN 上の IPv6 接続

Cisco ASR 9000 シリーズ ルータの、Cisco IOS XR Software リリース 4.2.1 では、デフォルト VRF の IPv4 のみのコア ネットワーク上のカスタマー サイト間で IPv6 接続がサポートされます。 VPN PE ルータは、2 つのアドレス ファミリの間で相互動作し、IPv4 でカプセル化された MDT と IPv6 カスタマー ルート間で制御および転送を行います。 IPv6 ユーザは、BGP を介して IPv6 over IPv4 マルチキャスト VPN サポートを設定できます。

詳細については、IPv6 マルチキャスト VPN の設定例, （236 ページ）を参照してください。

Cisco IOS XR Software で、MVPNv6 は MDT グループが設定された MVPNv4 とは異なる別個のデータを持つことができます。 ただし MVPNv6 と MVPNv4 には、同じデフォルト MDT グループが設定されている必要があります。

次の設定例に、Cisco IOS XR Softwareリリース 4.2.1 の MVPNv6 データ mdt を示します。

```
vrf cisco-sjc1
  address-family ipv4
   mdt data 226.8.3.0/24 threshold 5
   mdt default ipv4 226.8.0.1
  !
  address-family ipv6
```

```
 mdt data 226.8.4.0/24 threshold 5
 mdt default ipv4 226.8.0.1
!
```

BGP 要件

PE ルータでは、MVPN を認識する必要があり、MVPN に関する情報をリモート PE にシグナリングできる唯一のルータです。 特定の VRF 内の RPF PE ピアの取得に PE ルータは BGP ピアリング アドレス情報を使用するため、すべての PE ルータは互いに直接またはルート リフレクタを介して BGP 関係を持っていることが不可欠です。

BGP コネクタ属性を使用してトンネルを確立するため、PIM-SSM MDT トンネルは、設定された BGP MDT アドレス ファミリなしで設定できません。

マルチキャスト VPN に対する BGP サポートについては、『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』の「Implementing BGP on Cisco IOS XR Software」モジュールを参照してください。

MVPN スタティック P2MP TE

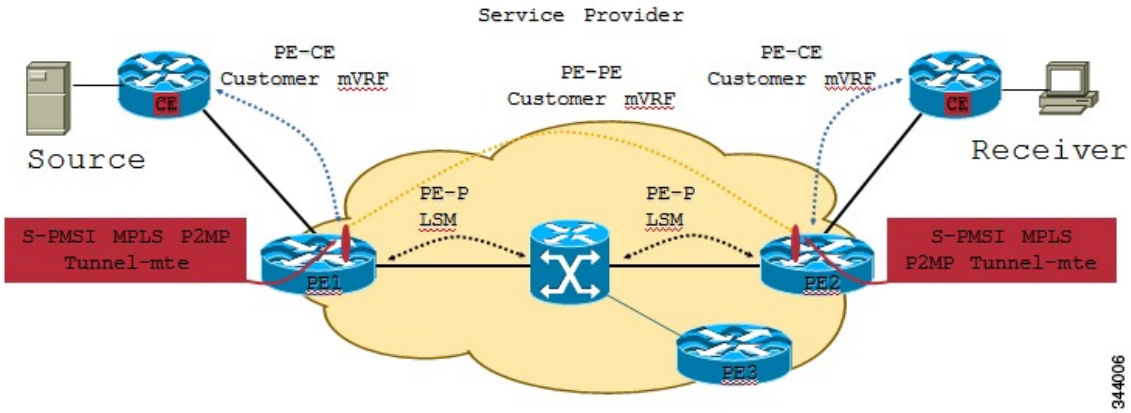
この機能では、ポイントツーマルチポイント トラフィック エンジニアリング（P2MP-TE）を使用したマルチキャストでのマルチキャスト VPN（MVPN）のサポートについて説明します。 現在、Cisco IOS-XR ソフトウェアではグローバル テーブルで P2MP-TE のみがサポートされており、グローバル テーブル内の (S,G) ルートを P2MP-TE トンネルにマッピングできます。 ただし、この機能では現在、サービス プロバイダーが P2MP-TE トンネルを使用して VRF マルチキャスト トラフィックを伝送できます。 VRF (S,G) トラフィックを P2MP-TE トンネルにマッピングするためにスタティック マッピングが使用され、VRF ベースの P2MP FEC を含む P2MP BGP 不透明値を MDT Selective Provider Multicast Service Interface（S-PMSI）として送信するために BGP-AD が使用されます。

P2MP-TE を使用したマルチキャストでの MVPN サポートの利点は次のとおりです。

- 帯域予約、帯域幅共有、転送レプリケーション、明示的ルーティング、高速再ルーティング（FRR）などのトラフィック エンジニアリングをサポートします。

- 複数のマルチキャスト ストリームのトンネルへのマッピングをサポートします。

図 *6*：マルチキャスト *VRF*



PE1 ルータでは、マルチキャスト S,G（ビデオ）トラフィックは VRF インターフェイス上で受信されます。マルチキャスト S,G ルートは、P2MP-TE トンネルに静的にマッピングされます。ヘッドエンドは次に、PMSI トンネル属性（PTA）で P2MP-TE トンネルをコア ツリーとして指定して、S,G ごとに S-PMSI（タイプ 3）BGP-AD ルートを発信します。この PTA のタイプは RSVP-TE P2MP LSP に設定され、PTA トンネル識別子の形式は、RSVP-TE P2MP LSP SESSION オブジェクトで伝送される <拡張トンネル ID、予約済み、トンネル ID、P2MP ID> に設定されます。複数の S,G A-D ルートに同じ PMSI トンネル属性を設定できます。

テールエンド PE（PE2、PE3）は、これらの（すべてのヘッドエンド PE によって送信された）S-PMSI アップデートを受信してキャッシュします。コア全体にアップストリーム マルチキャスト ホップ（UMH）があるときに、VRF 内に S,G Join が存在する場合、この PE は UMH からの S-PMSI アナウンスを探します。P2MP-TE PTA で S-PMSI ルートが見つかった場合、PE は、その VRF にトンネルのテール ラベルを関連付けます。パケットが P2MP-TE トンネルに到達すると、テールエンドがそのラベルを削除し、「関連付けられた」VRF 内で S,G ルックアップを実行します。一致が見つかった場合、パケットはその発信情報に従って転送されます。

マルチトポロジ ルーティング

マルチトポロジ ルーティングを使用すると、重複しないパスに流れること望ましい場合（たとえば、重複ビデオ ストリームをブロードキャストする場合）に、ネットワーク トラフィック フローを操作できます。

マルチトポロジ ルーティング テクノロジーの中心となるのが、ルータ空間インフラストラクチャ（RSI）です。RSI がルーティング テーブルのグローバル設定を管理します。これらのテーブルは、論理ルータの VRF テーブルに階層構造で編成されています。デフォルトでは、RSI がデフォルト VRF で IPv4 と IPv6 両方のユニキャストおよびマルチキャストのテーブルを作成します。マルチトポロジ ルーティングを使用すると、デフォルト VRF の名前付きトポロジを設定できます。

PIM は、送信元へのリバース パス転送（RPF）パスを検索するためのトポロジを選択するため、送信元またはグループアドレスに対するマッチングをサポートするルーティングポリシーを使用します。ポリシーを設定しない場合、既存の動作（デフォルトテーブルを選択する）が有効なままになります。

現在、IS-IS および PIM ルーティング プロトコルのみがマルチトポロジをイネーブルにしたネットワークをサポートしています。

マルチキャスト VPN エクストラネット ルーティング

マルチキャスト VPN（MVPN）エクストラネットルーティングを使用すると、サービスプロバイダーは、企業サイトから他の企業サイトにマルチキャスト VRF 上で IP マルチキャスト コンテンツを配信できます。 つまり、この機能は VRF 境界をシームレスにホップしてマルチキャスト コンテンツをエンドツーエンドで配信するための機能を提供します。

ユニキャストエクストラネットは、VRF 間で一致するルート ターゲットを設定するだけで実現できます。 しかし、マルチキャスト エクストラネットは、次のものに加えて VRF 間でルート ルックアップを解決するための設定が必要です。

- VRF 間でのマルチキャスト トポロジ マップの維持。
- VRF 間でトラフィックを転送するためのマルチキャスト配信ツリーの維持。

エクストラネットに関する情報

エクストラネットは企業外部のユーザに拡張された企業イントラネットの一部と見なすことができます。 VPN は、製品の販売や、強いビジネス パートナーシップの維持など、他の企業やカスタマーとビジネスを行うための手段の 1 つとして使用されます。 エクストラネットは、1 つ以上の企業サイトを外部のビジネス パートナーまたはサプライヤに接続し、企業のビジネス情報または業務の選択した部分を安全に共有するための VPN です。

MVPN エクストラネット ルーティングは、次のようなビジネス上の問題を解決するために使用できます。

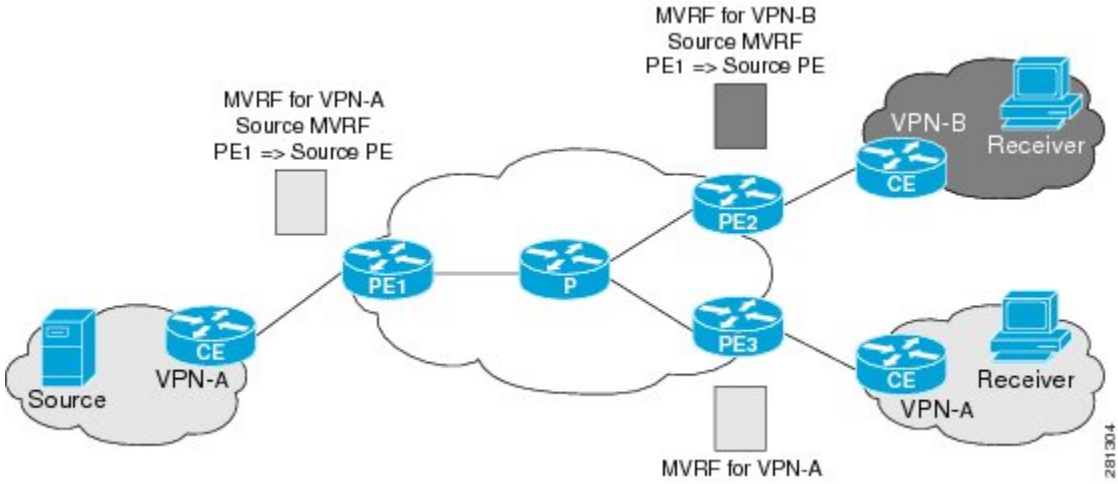
- 企業間の非効率的なコンテンツ配信。
- サービス プロバイダーまたはコンテンツ プロバイダーから企業 VPN カスタマーへの非効率的なコンテンツ配信。

MVPN エクストラネット ルーティングは IPv4 および IPv6 アドレス ファミリをサポートします。

エクストラネット ネットワークでは、PE ルータが VRF 間のトラフィック（図 7：エクストラネット MVPN のコンポーネント，（93 ページ） で「P」でラベル付けされたもの）を通過させることが必要です。 エクストラネット ネットワークは IPv4 または IPv6 を実行できますが、コア ネットワークは IPv4 マルチキャストのみを常に実行します。

エクストラネットのコンポーネント

図 *7*：エクストラネット *MVPN* のコンポーネント



MVRF：マルチキャスト VPN ルーティングおよび転送（VRF）インスタンス。 MVRF はマルチキャスト対応 VRF です。 VRF は、IP ルーティング テーブル、取得されたルーティング テーブル、そのルーティングテーブルを使用する一連のインターフェイス、ルーティングテーブルに登録されるものを決定する一連のルールおよびルーティングプロトコルで構成されています。 一般に、VRF には、プロバイダー エッジ（PE）ルータに付加されるカスタマー VPN サイトが定義されたルーティング情報が格納されています。

ソース MVRF：直接接続されたカスタマー エッジ（CE）ルータを使用して送信元に到達できる MVRF。

レシーバ MVRF：受信先が 1 つまたは複数の CE デバイスを介して接続される MVRF。

ソース PE：直接接続された CE ルータの背後にマルチキャスト送信元が存在する PE ルータ。

レシーバ PE：直接接続された CE ルータの背後に 1 つ以上の該当する受信先を持つ PE ルータ。

エクストラネット MVPN ルーティング トポロジに関する情報

ピアツーピア VPN のユニキャスト ルーティングでは、BGP ルーティング プロトコルがプロバイダー エッジ（PE）ルータ間で VPN IPv4 および IPv6 カスタマー ルートをアドバタイズするために使用されます。 ただし、MVPN エクストラネット ピアツーピア ネットワークでは、PIM RPF が、RPF ネクスト ホップが同じ VRF と別の VRF のどちらにあるか、およびそのソース VRF が PE にローカルまたはリモートのいずれであるかを判断するために使用されます。

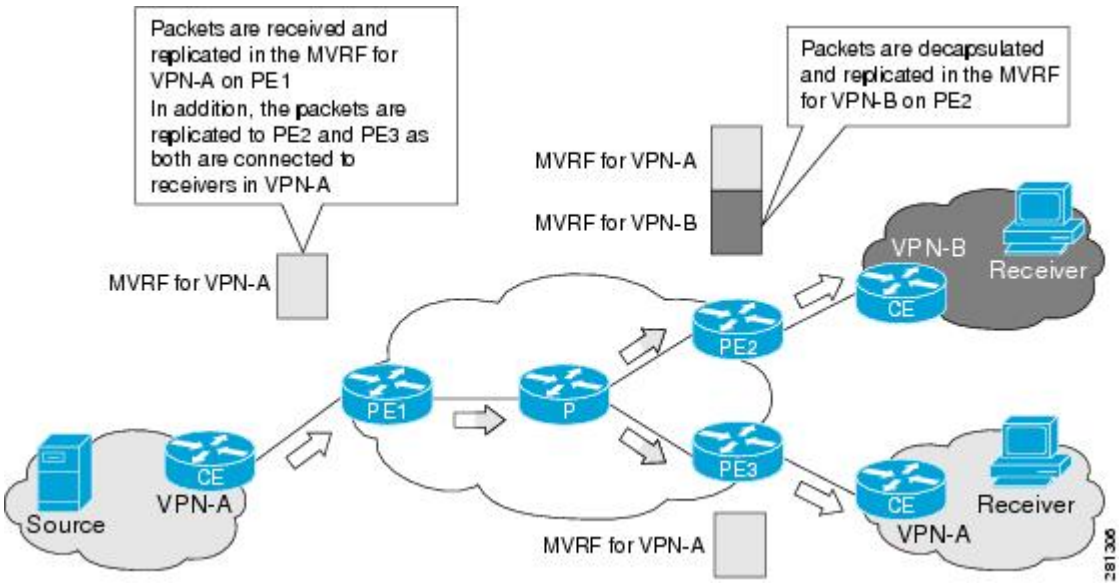
レシーバ PE ルータ上のソース MVRF

レシーバ PE ルータでソース MVRF を設定することによって企業 VPN カスタマーにエクストラネット MVPN サービスを提供するには、次の手順を実行します。

- 直接接続された CE ルータの背後のエクストラネット サイトに 1 つ以上の受信先が存在する受信 PE ルータで、マルチキャスト発信元に接続されたサイトと同じデフォルト MDT グループが存在する MVRF を設定します。
- レシーバ PE ルータで、ソース MVRF からレシーバ MVRF へのルートをインポートするために同じユニキャスト ルーティング ポリシーを設定します。

RPF ネクスト ホップの元となる MVRF がローカルの場合（レシーバ PE ルータでのソース MVRF）、レシーバ VRF の参加の状態は、ソース VRF のデフォルトマルチキャスト配信ツリー（MDT）を使用してコアに伝搬します。 図 8：レシーバ PE ルータでのソース MVRF, （94 ページ） に、ソース MVRF がレシーバ PE ルータ上で設定された（レシーバ MVRF トポロジでの送信元）エクストラネット MVPN トポロジのマルチキャスト トラフィックのフローを示します。 MVRF は、PE2（レシーバ PE ルータ）上の VPN-A および VPN-B 用に設定されます。 PE1（送信元 PE ルータ）の背後のマルチキャスト送信元は、VPN-A の MVRF にマルチキャスト ストリームを送信し、PE2（VPN-B のレシーバ PE ルータ）と、PE3（VPN-A のレシーバ PE ルータ）の背後に対象となる受信先があります。 PE1 は VPN-A の MVRF の送信元からパケットを受信すると、PE2 と PE3 にパケットを複製し転送します。 VPN-A の PE2 で受信したパケットはカプセル化が解除され、VPN-B のレシーバに複製されます。

図 *8*：レシーバ *PE* ルータでのソース *MVRF*



ソース PE ルータ上のレシーバ MVRF

ソース PE ルータでレシーバ MVRF を設定することによって企業 VPN カスタマーにエクストラネット MVPN サービスを提供するには、次の手順を実行します。

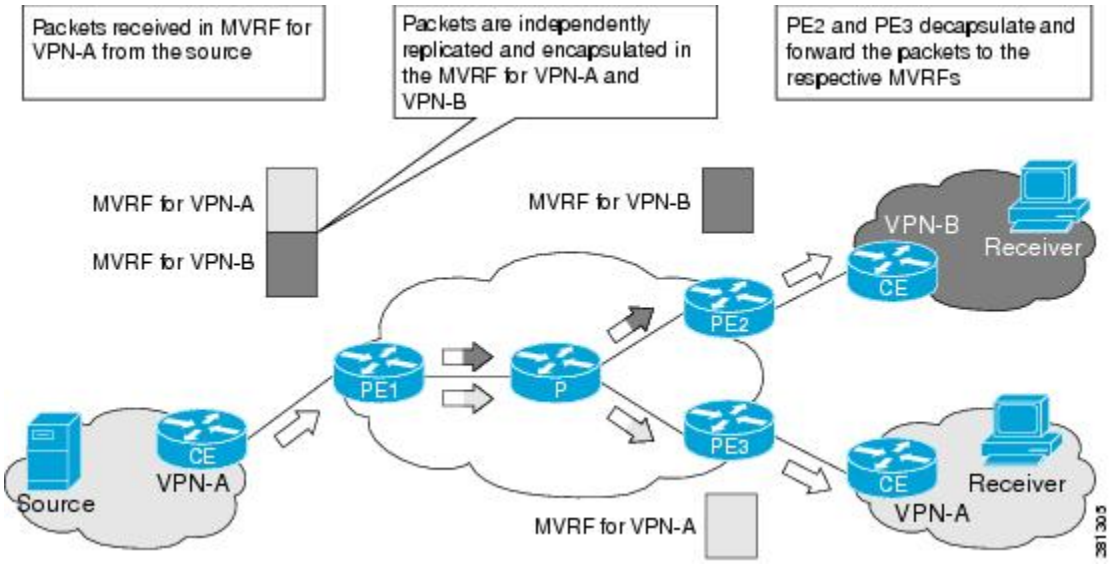
- 各エクストラネット サイトでは、MVRF がソース PE でまだ設定されていない場合、レシーバ MVRF と同じデフォルト MDT グループが割り当てられているソース PE ルータで追加の MVRF を設定します。

- レシーバ MVRF の設定では、ソース MVRF からレシーバ MVRF にルートをインポートするため、ソースおよびレシーバ PE ルータに同じユニキャスト ルーティング ポリシーを設定します。

RPF ネクストホップの発信元 MVRF がリモート（ソース PE ルータ上のレシーバ MVRF）の場合、レシーバ VRF の参加状態は、各レシーバの MDT 経由でコアに伝播されます。

図 9：ソース PE ルータ レシーバのレシーバ MVRF, （95 ページ） に、レシーバ MVRF がソース PE ルータ上で設定されたエクストラネット MVPN トポロジのマルチキャスト トラフィックのフローを示します。 MVRF は、PE1（ソース PE ルータ）上の VPN-A および VPN-B 用に設定されます。 PE1 の背後のマルチキャスト送信元は、VPN-A の MVRF にマルチキャスト ストリームを送信し、PE2 と PE3（それぞれ VPN-B と VPN-A のレシーバ PE ルータ）の背後に対象となる受信先があります。 PE1 は、VPN-A の MVRF の送信元からパケットを受信すると、VPN-A および VPN-B の MVRF でパケットを個別に複製およびカプセル化し、パケットを転送します。 この送信元からのパケットを受信すると、PE2 と PE3 はパケットのカプセル化を解除し、それぞれの MVRF に転送します。

図 *9*：ソース *PE* ルータ レシーバのレシーバ *MVRF*



詳細については、MVPN エクストラネット ルーティングの設定, （192 ページ） およびMVPN エクストラネット ルーティングの設定例, （248 ページ） も参照してください。

エクストラネットでの RPF ポリシー

RPF ポリシーは、レシーバ VRF での RPF ルックアップをバイパスし、指定されたソース VRF に参加状態を静的に伝播するように、レシーバ VRF で設定できます。 このようなポリシーは、マルチキャスト グループ範囲、マルチキャスト送信元範囲、または RP アドレスに基づいてソース VRF が選択されるように設定できます。

エクストラネットでの RPF ポリシーの設定の詳細については、ソース VRF に join を伝播するためのレシーバ VRF での RPL ポリシーの設定例, （250 ページ）およびソース VRF に join を伝播するためのソース PE ルータ上のレシーバ VRF での RPL ポリシーの設定例, （252 ページ）を参照してください。

マルチキャスト VPN ハブ アンド スポーク トポロジ

ハブ アンド スポーク トポロジは、2 つのサイト カテゴリ（ハブ サイトとスポーク サイト）の相互接続です。 サイト間でアドバタイズされるルートは、制限されたハブ アンド スポーク方法で接続を実現します。 残りのネットワーク（つまり、他のハブおよびスポーク）はハブの背後に隠れているように見えるため、スポークはハブのみ相互通信します。

ハブ アンド スポーク トポロジは、次の理由で適用できます。

- VPN カスタマーのスポーク サイトは、サーバ ファームなどのサービスをホストする中央の（またはハブ）サイトからのトラフィックをすべて受信します。
- VPN カスタマーのスポーク サイトは、中央サイトを介してスポーク サイト間のすべての接続が必要です。 つまり、ハブ サイトがスポーク間接続の中継ポイントになります。
- VPN カスタマーのスポーク サイトでは、スポーク サイト間の接続は不要です。 ハブは、すべてのサイトからのトラフィックを送受信できますが、スポーク サイトはハブ サイトとしかトラフィックを送受信できません。

ハブ アンド スポーク トポロジの実現

ハブアンドスポーク実装は、MVPNエクストラネット用に構築されたインフラストラクチャを利用します。 通常のMVPNは、パケットが任意のサイトから別のサイトに流れることができるモデルに従います。 ただし、ハブ アンド スポーク MVPN は登録に基づいてトラフィック フローを制限します。

サイトは、VPN アクセス用に PE-CE リンクによって PE ルータに接続されているサーバ ファームなど、CE ルータと他のデバイスのグループがある地理的位置にあると考えることができます。 各サイトを個別の VRF に配置するか、複数のサイトを PE ルータ上の 1 つの VRF にまとめることができます。

独立したVRFに各サイトをプロビジョニングすることによって、ユニキャストおよびマルチキャストのハブアンドスポーク実装を簡素化できます。 このような構成は、その性質上、あるスポーク サイトから別のスポーク サイトへのトラフィックの漏れからの保護を提供します。 Cisco IOS XR ソフトウェアのハブ アンド スポークの実装は、1 つの VRF に 1 つのサイトが対応するモデルに従います。 ルートのインポートまたはエクスポートが設定されている方法に基づいて、どのサイトもハブ サイトまたはスポーク サイトとして指定できます。 複数のハブ アンド スポーク サイトを特定の PE ルータにまとめることができます。

ユニキャストのハブアンドスポーク接続は、ハブサイトだけからルートをインポートするスポーク サイトおよびすべてのサイトからルートをインポートするハブサイトによって実現されます。 スポーク サイトがルートを交換していないため、スポーク サイト間のトラフィックは許可されま

せん。 スポーク間接続が必要な場合、ハブはあるスポーク サイトから学習したルートを他のスポーク サイトに再挿入することもできます。

MVPN ハブ アンド スポークは、コア トンネルを、ハブ サイトから送信されたトラフィック用とスポーク サイトから送信されたトラフィック用に分割することで実現されます。 MDT ハブはすべてのハブ サイトから発信されるトラフィックを伝送するトンネルであり、MDT スポークはすべてのスポーク サイトから送信されたトラフィックを伝送します。 このようなトンネルのエンドポイントは、ハブ アンド スポーク トポロジに参加するすべての PE で設定されます。 スポーク サイトがマルチキャスト ソースまたは RP をホストしない場合、MDT スポークのプロビジョニングはこのようなすべてのルータで完全に回避できます。

これらのトンネルがプロビジョニングされると、マルチキャスト トラフィック パスが次のようにポリシー ルーティングされます。

1. ハブ サイトは、MDT ハブだけにトラフィックを送信します。
2. スポーク サイトは、MDT スポークだけにトラフィックを送信します。
3. ハブ サイトは、両方のトンネルからトラフィックを受信します。
4. スポーク サイトは MDT ハブだけからトラフィックを受信します。

これらの規則により、ハブおよびスポークは相互にトラフィックを送受信できますが、スポーク間の直接通信は存在しません。 必要に応じて、スポーク間マルチキャストは、ハブ サイトでトラフィックを折り返すことで通過できます。

これらの拡張は、Cisco IOS XR ソフトウェア リリース 4.0 のマルチキャスト ハブ アンド スポーク トポロジに対して行われます。

- Auto-RP および BSR は、エクストラネットを介して接続された VRF 全体でサポートされます。 これは静的 RP だけの使用に限定されなくなりました。
- MP-BGP は、プレフィックスのネクスト ホップ情報を RIB に渡すときに、一致するインポート ルート ターゲットをパブリッシュできます。
- ルート ポリシーは IP アドレス範囲の代わりに拡張コミュニティのルート ターゲットを使用できます。
- ハブ アンド スポークのデータ mdt を実装できるように、エクストラネット v4 データ mdt のサポートが追加されました。

ラベル スイッチド マルチキャスト（LSM）マルチキャスト ラベル配布プロトコル（mLDP）ベースのマルチキャスト VPN（mVPN）のサポート

ラベル スイッチド マルチキャスト（LSM）はラベル カプセル化を使用してマルチキャストをサポートする MPLS テクノロジーの拡張機能です。 CRS の次世代 MVPN は、MPLS ネットワークを介して P2MP および MP2MP LSP を構築するために使用できるマルチキャスト ラベル配布プロ

トコル（mLDP）に基づいています。 これらの LSP は、グローバル テーブルまたは VPN のコンテキストで IPv4 と IPv6 の両方のマルチキャスト パケット転送に使用できます。

LSM MLDP based MVPN の利点

LSM には、コア内のカスタマー トラフィックを転送するために現在使用されている GRE コア トンネルと比較した場合、次の利点があります。

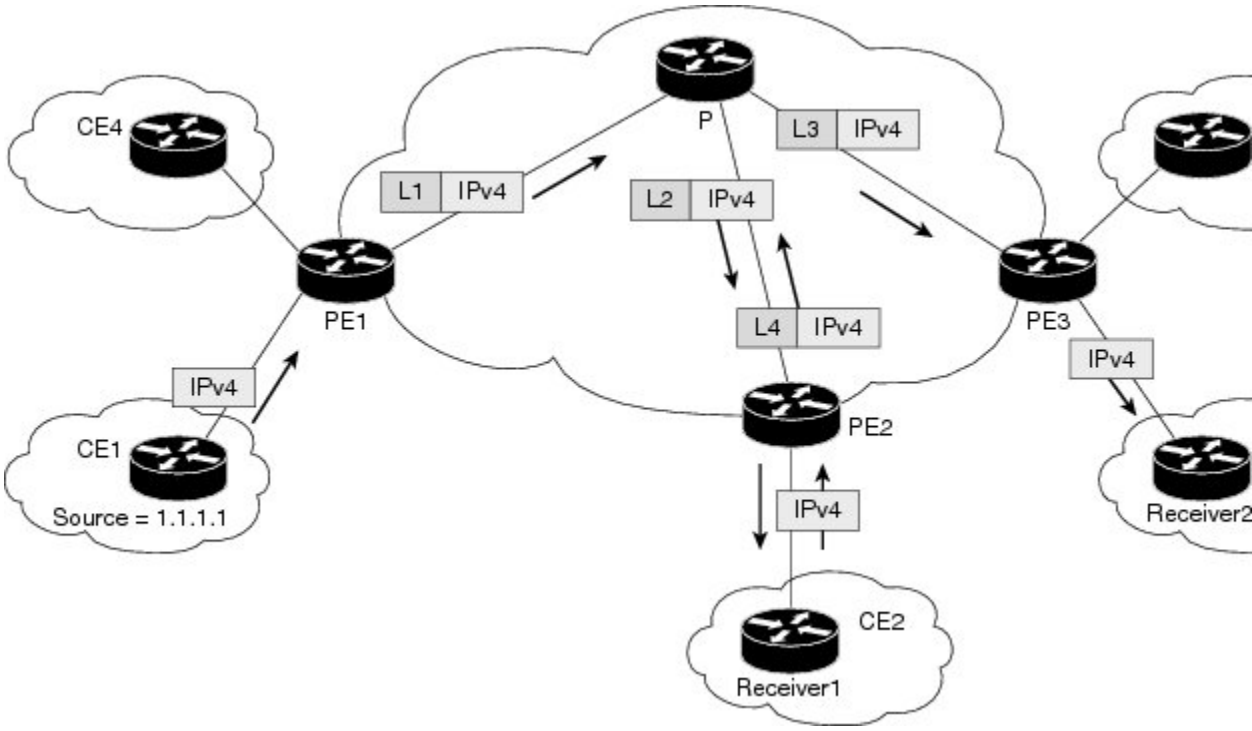
- IP マルチキャスト パケットを転送するための MPLS インフラストラクチャを活用し、ユニキャストとマルチキャストのための共通のデータ プレーンを提供します。
- MPLS の利点を高速再ルーティング（FRR）などの IP マルチキャストに適用します。
- PIM に関連した複雑さを解消します。

MLDP MVPN の設定

MLDP MVPN の設定により、MPLS を使用した IPv4 マルチキャスト パケット配信をイネーブルにします。 この設定では、MPLS ラベルを使用して、デフォルトおよびデータ マルチキャスト配信ツリー（MDT）を構築します。 MPLS レプリケーションは、コア ネットワークの転送メカニズムとして使用されます。 MLDP MVPN の設定を有効にするには、MPLS MLDP のグローバル設定がイネーブルであることを確認します。 MVPNエクストラネットサポートを設定するには、レシーバ プロバイダー エッジ（PE）ルータにソースのマルチキャスト VPN ルーティングおよび転送

（mVRF）を設定するか、ソース PE にレシーバの mVRF を設定します。 MLDP MVPN は、イントラネットとエクストラネットの両方に対してサポートされます。

図 *10*：*MLDP based MPLS* ネットワーク



P2MP および MP2MP ラベル スイッチド パス

mLDP は、MPLS コアにマルチキャスト ルーティング プロトコルが存在しなくても、MPLS ネットワーク内にマルチポイント ラベル スイッチド パス（MP LSP）を設定できるアプリケーションです。 mLDP は、他のマルチキャスト ツリー構築プロトコルと対話したり、それらを使用することなく、P2MP または MP2MP LSP を構築します。 MP LSP およびユニキャスト IP ルーティングに対する LDP 拡張を使用すると、mLDP は MP LSP を設定できます。 設定できる MP LSP のタイプには、ポイントツーマルチポイント（P2MP）とマルチポイントツーマルチポイント（MP2MP）のタイプの LSP の 2 つがあります。

P2MP LSP を使用すると、1 つのルート（入力ノード）からのトラフィックを複数のリーフ（出力ノード）に配信できます。ここで、各 P2MP ツリーは 2 タプル（ルート ノード アドレス、P2MP LSP 識別子）で一意に識別されます。 P2MP LSP は、1 つのルート ノード、0 個以上の中継ノード、および 1 つ以上のリーフ ノードで構成されます。ここで通常、ルート ノードとリーフ ノードは PE であり、中継ノードは P ルータです。 P2MP LSP の設定はレシーバから起動され、mLDP P2MP FEC を使用してシグナリングされます。ここで、LSP 識別子は MP Opaque Value 要素で表されます。 MP Opaque Value は、入力 LSR とリーフ LSR が認識している情報を伝送しますが、中継 LSR で解釈する必要はありません。 特定の入力ノードをルートとする、それぞれ独自の識別子を持つ MP LSP が複数存在する可能性があります。

MP2MP LSP を使用すると、複数の入力ノードからのトラフィックを複数の出力ノードに配信できます。ここで、MP2MP ツリーは 2 タプル（ルート ノード アドレス、MP2MP LSP 識別子）で一意に識別されます。 MP2MP LSP の場合は、入力ノードから送信されたパケットを、送信ノードを除くすべての出力ノードが受信します。

MP2MP LSP は P2MP LSP と同様ですが、各リーフ ノードが入力ノードと出力ノードの両方として機能します。 MP2MP LSP を構築するには、ダウンストリーム パスとアップストリーム パスを次のように設定できます。

- ダウンストリーム パスは、通常の P2MP LSP のように設定します。
- アップストリーム パスは、アップストリーム ルータに向けられた P2P LSP のように設定しますが、ダウンストリーム ラベルをダウンストリーム P2MP LSP から継承するようにします。

mLDP ベースのマルチキャスト VPN 内のパケット フロー

着信するパケットごとに、MPLS は複数の外側ラベルを作成します。 ソース ネットワークからのパケットは、レシーバ ネットワークへのパス上で複製されます。 CE1 ルータは、ネイティブの IP マルチキャスト トラフィックを送信します。 PE1 ルータは着信マルチキャスト パケットにラベルを付加し、MPLS コア ネットワークへのラベル付きパケットを複製します。 パケットは、コア ルータ（P）に到達すると、MP2MP のデフォルト MDT または P2MP のデータ MDT に対応する適切なラベル付きで複製され、すべての出力 PE に送信されます。 パケットが出力 PE に到達すると、ラベルが削除され、IP マルチキャスト パケットは VRF インターフェイスに複製されます。

mLDP ベースのマルチキャスト VPN の実現

mLDP によって構築されたラベル スイッチド パス（LSP）は、アプリケーションの要件や性質に応じて、次のようないくつかの方法で使用できます。

- インバンドシグナリングを使用したグローバルテーブル中継マルチキャスト用の P2MP LSP。
- MI-PMSI（Multidirectional Inclusive Provider Multicast Service Instance）に基づいた MVPN 用の P2MP/MP2MP LSP（Rosen ドラフト）。
- MS-PMSI（Multidirectional Selective Provider Multicast Service Instance）に基づいた MVPN 用の P2MP/MP2MP LSP（パーティション化 E-LAN）。

Cisco ASR 9000 シリーズ ルータは、MLDP の実装のための次の重要な機能を実行します。

1. VRF マルチキャスト IP パケットの GRE/ラベルによるカプセル化、およびコア インターフェイスへの複製（インポジション ノード）。
2. マルチキャスト ラベル パケットの異なるラベルによる別のインターフェイスへの複製（中間ノード）。
3. ラベル パケットのカプセル化解除、および VRF インターフェイスへの複製（ディスポジション ノード）。

mLDP プロファイルの特性

ここでは、さまざまな mLDP プロファイルの特性を示します。

Rosen-mLDP（BGP-AD なし）

このプロファイルの特性は次のとおりです。

- コアでは MP2MP mLDP ツリーが使用されています。
- VPN-ID が VRF 識別子として使用されています。
- デフォルト MDT に基づいた設定。
- IPv4 および IPv6 トラフィックに使用されるのと同じデフォルト MDT コア ツリー。
- PIM によって（デフォルト MDT 経由で）送信されるデータ MDT アナウンス。
- マルチキャスト トラフィックは SM、SSM、または Bidir のいずれかです。
- Inter-AS オプション A、B、および C がサポートされています。 コネクタ属性は VPN-IP ルートでアナウンスされます。

MS-PMSI-mLDP-MP2MP（BGP-AD なし）

このプロファイルの特性は次のとおりです。

- コアでは MP2MP mLDP ツリーが使用されています。
- IPv4 および IPv6 トラフィックとは異なる MS-PMSI コア ツリー。
- マルチキャスト トラフィックは SM または SSM のいずれかです。
- エクストラネット、ハブ アンド スポークがサポートされています。
- Inter-AS オプション A、B、および C がサポートされています。 コネクタ属性は VPN-IP ルートでアナウンスされます。

BGP-AD を使用した Rosen-GRE

このプロファイルの特性は次のとおりです。

- コアでは PIM ツリーが使用されています。 使用されているデータ カプセル化方式は GRE です。
- コアでは SM、SSM、または Bidir が使用されています。
- 設定はデフォルト MDT に基づいています。
- マルチキャスト トラフィックは SM または SSM のいずれかです。
- コアの MoFRR がサポートされています。
- エクストラネット、ハブ アンド スポーク、CsC、Customer-RP 検出（Embedded-RP、AutoRP、および BSR）がサポートされています。

- Inter-AS オプション A、B、および C がサポートされています。 VRF-Route-Import EC は VPN-IP ルートでアナウンスされます。

BGP-AD を使用した MS-PMSI-mLDP-MP2MP

このプロファイルの特性は次のとおりです。

- コアでは MP2MP mLDP ツリーが使用されています。
- マルチキャスト トラフィックは SM または SSM のいずれかです。
- エクストラネット、ハブアンドスポーク、CsC、Customer-RP 検出（Embedded-RP、AutoRP、および BSR）がサポートされています。
- Inter-AS オプション A、B、および C がサポートされています。 VRF-Route-Import EC は VPN-IP ルートでアナウンスされます。

BGP-AD を使用した MS-PMSI-mLDP-P2MP

このプロファイルの特性は次のとおりです。

- コアでは P2MP mLDP ツリーが使用されています。
- マルチキャスト トラフィックは SM または SSM のいずれかです。
- エクストラネット、ハブアンドスポーク、CsC、Customer-RP 検出（Embedded-RP、AutoRP、および BSR）がサポートされています。
- Inter-AS オプション A、B、および C がサポートされています。 VRF-Route-Import EC は VPN-IP ルートでアナウンスされます。

VRF インバンド シグナリング（BGP-AD なし）

このプロファイルの特性は次のとおりです。

- コアでは P2MP mLDP ツリーが使用されています。
- コアの MoFRR がサポートされています。
- VRF-S,G ルートごとに 1 つのコア ツリーが構築されます。 コアを経由した RPF の到達可能性により、VRF に (*,G) ルートを含めることはできません。
- マルチキャスト トラフィックは SM S,G または SSM のいずれかです。

MLDP の実装および OAM の概念の詳細については、『Cisco IOS XR MPLS Configuration Guide for the Cisco ASR 9000 シリーズ ルータ』を参照してください。

サポートされる MVPN プロファイルの要約

この表は、サポートされる MVPN プロファイルの要約を示しています。

プロファイル番号	名前	不透明値	BDP-AD	データ MDT
0	Rosen GRE	N/A	N/A	デフォルト MDT 経由の PIM TLV
1	Rosen MLDP	タイプ 2 - ルート アドレス：VPN-ID:0-n	N/A	デフォルト MDT 経由の PIM TLV
2	MS-PMSI（パーティション）MLDP MP2MP	シスコ独自 - ソース-PE:RD:0	N/A	N/A
3	BGP-AD を使用した Rosen GRE	N/A	• Intra-AS MI- PMSI • データ MDT の S-PMSI	PIM または BGP-AD（ノブ制御）
4	BGP-AD を使用した MS-PMSI（パーティション）MLDP MP2MP	タイプ 1 - ソース-PE:Global-ID	• 空の PTA を使用した I-PMSI • パーティション MDT の MS-PMSI • データ MDT の S-PMSI • S-PMSI cust RP 検出ツリー	BGP-AD
5	BGP-AD を使用した MS-PMSI（パーティション）MLDP P2MP	タイプ 1 - ソース-PE:Global-ID	• 空の PTA を使用した I-PMSI • パーティション MDT の MS-PMSI • データ MDT の S-PMSI • S-PMSI cust RP 検出ツリー	BGP-AD
6	VRF インバンド MLDP	RD:S,G	N/A	N/A

プロファイル番号	名前	不透明値	BDP-AD	データ MDT
7	グローバル インバンド	S,G	N/A	N/A
8	グローバル P2MP TE	N/A	N/A	N/A
9	BGP-AD を使用した Rosen MLDP	タイプ 2 - ルート アドレス：VPN - ID:0 -n	• Intra-AS MI- PMSI • データ MDT の S-PMSI	PIM または BGP-AD（ノブ制御）

MLDP MVPN の設定プロセス（イントラネット）

次の手順は、イントラネットのための MLDP MVPN の各種の設定プロセスの広範囲の概要を示しています。

（注） さまざまな MVPN プロファイルの詳細な要約については、サポートされる MVPN プロファイルの要約，（102 ページ）を参照してください。

- MPLS MLDP のイネーブル化
  - configure
  - mpls ldp mldp
- VRF エントリの設定
  - configure
  - vrf vrf_name
  - address-family ipv4/ipv6 unicast
  - import route-target route-target-ext-community
  - export route-target route-target-ext-community
- VPN ID の設定
  - configure
  - vrf *vrf_name*
  - vpn id *vpn_id*

VPN ID の設定の手順は、プロファイル 1 および 9（Rosen MLDP）に必要です。

- MVPN ルーティング/転送インスタンスの設定
  - configure
  - multicast-routing vrf *vrf_name*
  - address-family ipv4
  - mdt default mldp ipv4 *root-node*

  プロファイル 1（MLDP Rosen）の場合は **mdt default mldp ipv4** コマンドが、プロファイル 4/5（BGP-AD を使用した MS-PMSI）の場合は **mdt partitioned mldp ipv4 mp2mp/p2mp** コマンドが設定されます。
- ルート識別子の設定
  - configure
  - router bgp *AS Number*
  - vrf *vrf_name*
  - rd *rd_value*
- データ MDT の設定（任意）
  - configure
  - multicast-routing vrf *vrf_name*
  - address-family ipv4
  - mdt data <1-255>
- BGP MDT アドレス ファミリの設定
  - configure
  - router bgp *AS Number*
  - address-family ipv4 mdt
- BGP vpnv4 アドレス ファミリの設定
  - configure
  - router bgp *AS Number*
  - address-family vpnv4 unicast
- BGP IPv4 VRF アドレス ファミリの設定
  - configure
  - router bgp *AS Number*
  - vrf *vrf_name*

- address-family ipv4 unicast

- VRF の PIM SM/SSM モードの設定
  - configure
  - router pim
  - vrf *vrf_name*
  - address-family ipv4
  - rpf topology route-policy *rosen_mvpn_mldp*

  プロファイルごとに、異なるルート ポリシーが設定されます。
- ルート ポリシーの設定
  - route-policy *rosen_mvpn_mldp*
  - set core-tree *tree-type*
  - pass
  - end-policy

  プロファイル 1（MLDP Rosen）の場合は mldp-rosen コア ツリー タイプが、プロファイル 4/5（BGP-AD を使用した MS-PMSI）の場合は mldp-partitioned-mp2mp/p2mp コア ツリー タイプが設定されます。

（注） 上の手順の設定は、各設定で使用されるプロファイルによって異なります。 各プロファイルの詳細な例については、LSM based MLDP の設定例, （268 ページ）を参照してください。

Multicast Source Discovery Protocol

Multicast Source Discovery Protocol（MSDP）は、複数の PIM スパース モード ドメインを接続するためのメカニズムです。 MSDP を使用すると、さまざまなドメイン内のすべての Rendezvous Point（RP; ランデブー ポイント）に、グループのマルチキャスト送信元を通知できます。 各 PIM-SM ドメインは自身の RP を使用するため、他のドメインの RP に依存する必要はありません。

PIM-SM ドメインの RP は、他のドメインの MSDP 対応ルータとの MSDP ピアリング関係を持ちます。 各ピアリング関係は、下位のルーティング システムによって維持される TCP 接続上で行われます。

MSDP スピーカーは、Source Active（SA）メッセージとも呼ばれるメッセージを交換します。 RP は、一般に PIM register メッセージを通じてローカル アクティブ ソースについて学習するとき、MSDP プロセスが SA メッセージの register をカプセル化し、ピアに情報を転送します。 メッセージには、マルチキャスト フローの送信元およびグループの情報と、カプセル化されたデータが格納されます。 ネイバー RP にマルチキャスト グループのローカル加入者がある場合、RP は S,G ルートをインストールし、SA メッセージに含まれるカプセル化データを転送し、送信元に向けて

PIM join を逆に送信します。 このプロセスは、マルチキャスト パスをドメイン間で構築する方法について説明します。

（注） 最適な MSDP ドメイン間動作のために BGP またはマルチプロトコル BGP を設定することをお勧めしますが、Cisco IOS XR Software の実装では必須とは見なされません。 BGP またはマルチプロトコル BGP と MSDP とともに使用する方法については、インターネット技術特別調査委員会（IETF）インターネット ドラフト『Multicast Source Discovery Protocol (MSDP)』に記載されている MSDP RPF ルールを参照してください。

マルチキャスト ノンストップ フォワーディング

マルチキャスト用の Cisco IOS XR Software ノンストップ フォワーディング（NSF）機能を使用すると、マルチキャスト パケット転送のハイ アベイラビリティ（HA）が向上します。 NSF は、コントロール プレーンのハードウェアまたはソフトウェア障害により、ルータを通過する既存のパケット転送が中断されるのを防ぎます。

マルチキャスト転送情報ベース（MFIB）の内容は、コントロールプレーン障害時に変化しないよう固定されます。 その後、隣接ルータが問題のあるルータで PIM hello ネイバー隣接がタイムアウトする前に、PIM は通常のプロトコル処理と状態を回復しようとします。 この動作は、NSF 対応ルータがネイバーに転送されるのを防ぎます。この機能がない場合、ネイバーはタイム アウト隣接によって障害を検出します。 MFIB 内のルートは NSF が開始された後に古いとマーキングされ、トラフィックは NSF 完了まで（それらのルートに基づいて）転送され続けます。 完了すると、MRIB が MFIB に通知し、MFIB が現在の MRIB ルート情報と MFIB を同期するマークアンドスイープを実行します。

マルチキャスト コンフィギュレーション サブモード

Cisco IOS XR Software では、コントロール プレーンの CLI 設定がプロトコル固有のサブモードに移動されており、マルチキャスト機能を多数のインターフェイスでイネーブル化、ディセーブル化、設定するためのメカニズムが提供されます。

Cisco IOS XR Software では、サブモードで使用できるコマンドのほとんどを、グローバル コンフィギュレーション モードで 1 つのコマンド文字列として実行できます。

たとえば、**ssm** コマンドは、次のようにマルチキャスト ルーティング コンフィギュレーション サブモードから実行できます。

```
RP/0/RSP0/CPU0:router(config)# multicast-routing
RP/0/RSP0/CPU0:router(config-mcast-ipv4)# ssm range
```

また、次のように、グローバル コンフィギュレーション モードから同じコマンドを実行できます。

```
RP/0/RSP0/CPU0:router(config)# multicast-routing ssm range
```

次のマルチキャストプロトコル固有サブモードは、これらのコンフィギュレーションサブモードで使用できます。

マルチキャスト ルーティング コンフィギュレーション サブモード

Cisco IOS XR ソフトウェア リリース 3.7.2 以降のリリースでは、マルチキャスト PIE（asr9k-mcast-p.pie）がインストールされている場合、明示的な設定を行わなくても基本的なマルチキャスト サービスが自動的に開始されます。 自動的に開始されるマルチキャスト サービスは次のとおりです。

- MFWD
- MRIB
- PIM
- IGMP

これ以外のマルチキャストサービスを開始するには、明示的に設定する必要があります。たとえば、MSDP プロセスを開始するためには、**router msdp** コマンドを入力し、MSDP プロセスを明示的に設定する必要があります。

multicast-routing ipv4 または **multicast-routing ipv6** コマンドを実行すると、すべてデフォルトのマルチキャスト コンポーネント（PIM、IGMP、MLD、MFWD、および MRIB）が自動的に開始され、CLI プロンプトが「config-mcast-ipv4」または「config-mcast-ipv6」に変わり、マルチキャスト ルーティング コンフィギュレーション サブモードが開始されたことが示されます。

PIM コンフィギュレーション サブモード

router pim コマンドを発行すると、CLI プロンプトが「config-pim-ipv4」に変わり、デフォルト PIM アドレス ファミリ コンフィギュレーション サブモードが開始されたことが示されます。IPv6 の PIM アドレス ファミリ コンフィギュレーション サブモードを開始するには、Enter キーを押す前に **address-family ipv6** キーワードと **router pim** コマンドを入力します。

IGMP コンフィギュレーション サブモード

router igmp コマンドを実行すると、CLI プロンプトが「config-igmp」に変わり、IGMP コンフィギュレーション サブモードが開始されたことが示されます。

MLD コンフィギュレーション サブモード

router mld コマンドを実行すると、CLI プロンプトが「config-mld」に変わり、MLD コンフィギュレーション サブモードが開始されたことが示されます。

MSDP コンフィギュレーション サブモード

router msdp コマンドを実行すると、CLI プロンプトが「config-msdp」に変わり、ルータ MSDP コンフィギュレーション サブモードが開始されたことが示されます。

インターフェイス設定の継承の概要

Cisco IOS XR Software では、すべてのインターフェイスによって継承できるマルチキャスト ルーティング サブモード内でコマンド コンフィギュレーションを適用することで、多数のインターフェイスに対してコマンドを設定できます。 継承メカニズムを無効にするには、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、明示的に別のコマンド パラメータを入力します。

たとえば、次の設定でルータのすべての既存および新しい PIM インターフェイスが 420 秒の hello 間隔パラメータを使用することをすばやく指定できます（ルータ PIM コンフィギュレーション モードで）。 ただし、Packet-over-SONET/SDH（POS）インターフェイス 0/1/0/1 ではグローバル インターフェイス コンフィギュレーションが無効になり、210 秒の hello 間隔時間が使用されます。

```
RP/0/RSP0/CPU0:router(config)# router pim
RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# hello-interval 420
RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# interface pos 0/1/0/1
RP/0/RSP0/CPU0:router(config-pim-ipv4-if)# hello-interval 210
```

次に、継承メカニズムを使用するコマンドのリストを示します（適切なルータ サブモードで指定）。

```
router pim
  dr-priority
  hello-interval
  join-prune-interval

multicast-routing
  version
  query-interval
  query-max-response-time
  explicit-tracking
router mld
  interface all disable
  version
  query-interval
  query-max-response-time
  explicit-tracking

router msdp
  connect-source
  sa-filter
  filter-sa-request list
  remote-as
  ttl-threshold
```

インターフェイス設定の継承の無効化の概要

他の場所で示されているように、Cisco IOS XR Software では、すべてのインターフェイスによって継承できるマルチキャストルーティングサブモード内でコンフィギュレーションを適用することで、複数のインターフェイスを設定できます。

特定のインターフェイスまたはすべてのインターフェイスで継承機能を無効にするには、マルチキャスト ルーティング コンフィギュレーション モードのアドレス ファミリ IPv4 または IPv6 サブモードを開始し、**interface-inheritance disable** コマンドと **interface** *type interface-path-id* または **interface all** コマンドを入力します。 これにより、PIM または IGMP プロトコルは、マルチキャスト ルーティングを拒否し、指定したインターフェイスのみでマルチキャスト転送を許可します。ただし、ルーティングは指定した個々のインターフェイスで明示的にイネーブルにできます。

次の設定は、PIM および IGMP 全般でマルチキャストルーティング インターフェイス継承をディセーブルにしますが、転送はイネーブルなままになります。 例では、GigabitEthernet 0/6/0/3 の IGMP でのインターフェイスのイネーブル化を示します。

```
RP/0/RSP0/CPU0:router# multicast-routing  address-family ipv4
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# interface all enable
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# interface-inheritance disable

!

!
RP/0/RSP0/CPU0:router(config)# router igmp
RP/0/RSP0/CPU0:router(config-igmp)# vrf default
RP/0/RSP0/CPU0:router(config-igmp)# interface GigabitEthernet0/6/0/0
RP/0/RSP0/CPU0:router(config-igmp-name-if)# router enable
```

関連情報については、インターフェイスのイネーブル化とディセーブル化の概要, （110 ページ）を参照してください。

インターフェイスのイネーブル化とディセーブル化の概要

Cisco IOS XR Software マルチキャスト ルーティング機能がルータで設定されている場合、デフォルトでは、イネーブルになっているインターフェイスはありません。

単一のインターフェイスまたは複数のインターフェイスのマルチキャスト ルーティングおよびプロトコルをイネーブルにするには、マルチキャスト ルーティング コンフィギュレーション モードで **interface** コマンドを入力し、インターフェイスを明示的にイネーブルにする必要があります。

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を設定するには、マルチキャストルーティング コンフィギュレーション モードで **interface all** コマンドを入力します。 完全にマルチキャストルーティングをイネーブルにする任意のインターフェイスは、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で特にイネーブルにする（またはデフォルトにする）必要があり、PIM および IGMP/MLD コンフィギュレーション モードでディセーブルにしないでください。

たとえば、次の設定では、すべてのインターフェイスがマルチキャスト ルーティング コンフィギュレーション サブモードから明示的に設定されています。

```
RP/0/RSP0/CPU0:router(config)# multicast-routing
RP/0/RSP0/CPU0:router(config-mcast)# interface all enable
```

マルチキャスト ルーティング コンフィギュレーション サブモードからグローバルに設定されたインターフェイスをディセーブルにするには、次の例に示すように、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ます。

```
RP/0/RSP0/CPU0:router(config-mcast)# interface GigabitEthernet0pos 0/1/0/0
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4-if)# disable
```

Multicast Routing Information Base（マルチキャスト ルーティング情報ベース）

マルチキャスト ルーティング情報ベース（MRIB）は、1 つ以上のマルチキャスト ルーティング プロトコルを実行している論理ネットワークを記述するプロトコル独立型マルチキャスト ルーティング テーブルです。 テーブルには個別のマルチキャスト ルーティング プロトコルでインストールされた汎用マルチキャスト ルートが含まれます。 ルータが設定されている論理ネットワーク（VPN）ごとに MRIB があります。 MRIB はマルチキャスト ルーティング プロトコル間でルートを再配信しません。同等のものから優先されるマルチキャスト ルートを選択し、任意のマルチキャスト ルートの選択された属性の変更をクライアントに通知します。

マルチキャスト転送情報ベース

マルチキャスト転送情報ベース（MFIB）は、プロトコル独立型マルチキャスト フォワーディング システムで、指定されたネットワークで認識されている発信元またはグループのペアごとに、一意のマルチキャスト フォワーディング エントリが格納されています。 ルータが設定されている論理ネットワーク（VPN）ごとに、個別の MFIB があります。 各 MFIB エントリは、指定された発信元またはグループのペアを、リバース フォワーディング（RPF）チェックの場合は着信インターフェイス（IIF）に、マルチキャスト フォワーディングの場合は発信インターフェイス リスト（olist）に解決します。

MSDP MD5 パスワード認証

MSDP MD5 パスワード認証は、2 つの Multicast Source Discovery Protocol（MSDP）ピア間の TCP 接続上で Message Digest 5（MD5）シグネチャの保護を提供するための拡張です。 この機能は、TCP 接続ストリームに導入されるスプーフィングされた TCP セグメントの脅威に対して MSDP を保護することにより、追加のセキュリティを提供します。

MSDP MD5 パスワード認証は MSDP ピア間の TCP 接続上で送信された各セグメントを検証します。 **password clear** コマンドは、2 つの MSDP ピア間の TCP 接続の MD5 認証をイネーブルにす

るために使用されます。 2 つの MSDP ピア間で MD5 認証がイネーブルになると、ピア間の TCP 接続で送信された各セグメントが確認されます。

（注） MSDP MD5 認証は、両方の MSDP ピアで同じパスワードを設定する必要があります。そうしないと、ピア間の接続はイネーブルになりません。 「password encrypted」コマンドは、保存されている実行コンフィギュレーションに適用する場合にのみ使用されます。 MSDP MD5 認証を設定すると、このコマンドを使用して設定を復元できます。

MSDP MD5 パスワード認証は、信頼性とセキュリティを向上させるために業界標準の MD5 アルゴリズムを採用しています。

IGMP インターフェイスでの VRF の上書き

次世代の集約またはコア ネットワークのユーザネットワーク インターフェイス上のすべてのユニキャスト トラフィックは、特定の VRF にマッピングする必要があります。 これらは、ネットワーク間の側の MPLS VPN にマッピングする必要があります。 これには、この特定の VRF で物理インターフェイスの設定が必要です。

この機能により、ユーザとユーザを結ぶインターフェイス経由で受信する IGMP パケットから、グローバル マルチキャスト ルーティング テーブル中のマルチキャスト ルートへのマッピングが可能になります。 これにより、特定の VRF 内のインターフェイスが、あるマルチキャスト ルートに対して、テーブル中の発信インターフェイス リストの一部になることができます。

デフォルト（グローバル）VRF では、デフォルト以外の VRF インターフェイス経由で受信した IGMP パケットが処理され、後で IGMP はインターフェイス関連のマルチキャスト ステート（ルートとインターフェイス）を MRIB に配信します。 これは、インターフェイスが属する VRF ではなくデフォルト VRF を介して行われます。 MRIB、PIM、MSDP および MFIB はデフォルト VRF によってこのインターフェイスのマルチキャスト ステートを処理します。

設定されたインターフェイスで特定の (S,G) の IGMP join を受信すると、IGMP は VRF 固有のデータベースにこの情報を保存します。 ただし、アップデートを MRIB に送信する際に、IGMP はデフォルト VRF を通じてこのルートを送信します。 MRIB は、この (S,G) を、デフォルト マルチキャスト ルーティング テーブルの OLIST メンバとして、インターフェイスとともにプログラムします。

同様に、PIM が MRIB からの IGMP ルートに関する情報を要求するとき、MRIB はデフォルト VRF のコンテキストで PIM にこのアップデートを送信します。

この機能は特に次の点をサポートしています。

- デフォルト以外の VRF インターフェイス上の IGMP 要求の、デフォルト VRF のマルチキャスト ルーティング テーブルへのマッピング。
- 実行時の VRF オーバーライド機能のイネーブル化およびディセーブル化。
- ルーティング ポリシー設定は個々のインターフェイス単位でできないため、グローバル VRF（デフォルト）レベルでのルーティング ポリシー設定。

- 物理イーサネット、VLAN サブインターフェイス、バンドル、およびバンドル上の VLAN を含む、すべてのレイヤ 3 およびレイヤ 2 インターフェイス タイプ上での IGMP VRF オーバーライドの有効化および無効化。
- VRF オーバーライド機能が動作している場合でも、同じ規模のマルチキャスト ルートと OLIST インターフェイスが現在プラットフォームでサポートされています。

サテライト nV のサポート

マルチキャスト コンポーネント（IGMP、IGMP スヌーピング、PIM、MRIB/LMRIB、MFIB、L2FIB を含む）が、新しいサテライト-イーサ インターフェイス タイプを認識するように、また論理または物理タイプを照会して保持するように機能強化されました。

（注）
- サテライト-イーサ インターフェイスは、サテライトのアップリンク インターフェイスに基づいて論理インターフェイスまたは物理インターフェイスのどちらかになります。サテライト インターフェイスのこの属性は、実行時に決定される必要があります。
- 論理的なサテライト-イーサ インターフェイスの場合、発信インターフェイスの選択（このインターフェイスがバンドル インターフェイスであるかのようにアップリンク インターフェイスのメンバを選択すること）は、システムが計算するハッシュ値に基づいて行われます。

サテライト nV の機能の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router Interfaces and Hardware Component Configuration Guide*』を参照してください。

マルチキャスト ルーティングの実装方法

このセクションでは、基本的なマルチキャスト設定の作成のための手順と、マルチキャスト ネットワークでのルータを最適化、デバッグ、および検出するのに役立つオプションのタスクについて説明します。

- マルチキャスト VPN の設定, （136 ページ）（任意）

PIM-SM および PIM-SSM の設定

手順の概要

1. **configure**
2. **multicast-routing** [**address-family** {**ipv4** | **ipv6**}]
3. **interface all enable**
4. exit
5. **router igmp mld**
6. **version** {**1** | **2** | **3**}
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show pim** [**ipv4** | **ipv6**] **group-map** [*ip-address-name*] [**info-source**]
9. **show pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **topology** [*source-ip-address* [*group-ip-address*] | **entry-flag** *flag* | **interface-flag** | **summary**] [**route-count**]

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** | **ipv6**}] 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 がデフォルトでイネーブルです。
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。

	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(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、ルータを送信元コンフィギュレーション モードに戻します。
ステップ 5	**router igmp mld** 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	次のいずれかのコマンドを使用します。
	• **end**
	• **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

	コマンドまたはアクション	目的
		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 8	**show pim** [**ipv4** | **ipv6**] **group-map** [*ip-address-name*] [**info-source**] 例： `RP/0/RSP0/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** | **ipv6**] **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。

レガシー マルチキャストの配置で使用する PIM-SSM の設定

レガシー マルチキャスト対応ネットワークに PIM-SSM を配置すると、ネットワークに接続されるさまざまなデバイスで使用されているマルチキャスト グループ管理プロトコルの変更が必要になるため、問題が発生します。 その場合、ホスト、ルータおよびスイッチをすべてアップグレードする必要があります。

PIM-SSM配置のレガシーホストとスイッチをサポートするために、Cisco ASR 9000 シリーズ ルータ は、設定可能なマッピング機能を提供します。 SSM グループ範囲内のグループのレガシー グループ メンバーシップ レポートは、その一連の(S,G)チャネルのサービスを提供する送信元のセットにマッピングされます。

この設定は 2 つの作業からなります。

PIM-SSM マッピングの制約事項

PIM-SSM マッピングは SSM グループ範囲を変更しません。 代わりに、レガシー デバイスは、SSM グループ範囲内の目的のグループのグループ メンバーシップを報告する必要があります。

スタティック SSM マッピングのアクセス コントロール リストのセットの設定

この作業では、アクセス コントロール リスト（ACL）のセットを設定します。各 ACL は、1 つ以上の送信元にマッピングする SSM グループのセットを表します。

手順の概要

1. **configure**
2. **ipv4 access-list** *acl-name*
3. [*sequence-number*] **permit** *source* [*source-wildcard*]
4. ステップ 3, （117 ページ） を繰り返し、ACL にエントリを追加します。
5. セットの一部にするすべての ACL が入力されるまでステップ 2, （117 ページ） からステップ 4, （117 ページ） を繰り返します。
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**ipv4 access-list** *acl-name* 例： `RP/0/RSP0/CPU0:router(config)# ipv4 access-list mc3`	IPv4 ACL コンフィギュレーション サブモードを開始し、IPv4 アクセス リストの名前を作成します。
ステップ 3	[*sequence-number*] **permit** *source* [*source-wildcard*] 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# permit 1 host 232.1.1.2 any`	指定されたアクセス リスト セットの一部としてソースを認識するため、アクセス リストの条件を設定します。各 ACL は、マッピングする一連の SSM グループを記述します。
ステップ 4	ステップ 3, （117 ページ） を繰り返し、ACL にエントリを追加します。	—
ステップ 5	セットの一部にするすべての ACL が入力されるまでステップ 2, （117 ページ） か	—

	コマンドまたはアクション	目的
	らステップ 4, （117 ページ） を繰り返します。	

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

SSM マッピングの一連の送信元の設定

この作業では、アクセス リスト（ACL）で記述される、SSM グループによってマッピングされる複数の送信元を設定します。

手順の概要

1. **configure**
2. **router igmp** [**vrf** *vrf-name*]
3. **ssm map static** *source-address access-list*
4. SSM マッピングのセットに含める送信元アドレスの個数だけ ステップ 3, （119 ページ） を繰り返します。
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
6. **show igmp** [**vrf** *vrf-name*] **ssm map** [*group-address*][*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	**router igmp** [**vrf** *vrf-name*] 例： `RP/0/RSP0/CPU0:router(config)# router igmp vrf vrf20`	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 3	**ssm map static** *source-address access-list* 例： `RP/0/RSP0/CPU0:router(config-igmp)# ssm map static 232.1.1.1 mc2`	指定したアクセス リストによって記述された SSM グループをマッピングする複数の送信元の一部として送信元を設定します。
ステップ 4	SSM マッピングのセットに含める送信元アドレスの個数だけ ステップ 3, （119 ページ） を繰り返します。	—
ステップ 5	次のいずれかのコマンドを使用します。 • **end** • **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]:`

	コマンドまたはアクション	目的

	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		◦ 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。
		• 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 6	**show igmp** [**vrf** *vrf-name*] **ssm map** [*group-address*][*detail*]
	例：
	`RP/0/RSP0/CPU0:router# show igmp vrf vrf20 ssm map 232.1.1.1`
	`232.1.1.1 is static with 1 source`
	または
	`RP/0/RSP0/CPU0:router# show igmp vrf vrf20 ssm map`
	`232.1.1.0 is static with 3 sources`
	`232.1.1.1 is static with 1 source`
		（任意）マッピング状態を照会します。
		• マッピング用に 1 個のアドレスを指定した場合、そのアドレスの状態のみが返されます。
		• マッピング用にアドレスを指定しない場合、すべての送信元の状態が返されます。

スタティック RP の設定と下位互換性の許可

PIM がスパース モードで設定されている場合は、マルチキャスト グループのランデブー ポイント（RP）として動作する 1 つ以上のルータを選択する必要があります。 RP は、共有配布ツリーの選択ポイントに配置された単一の共通ルートです。 RPは各ルータで静的に設定するか、Auto-RP または BSR によって学習できます。

ここでは、静的な RP を設定します。 RP の詳細については、ランデブー ポイント，（81 ページ）を参照してください。 Auto-RP の設定方法については、グループから RP へのマッピングを自動化するための Auto-RP の設定，（123 ページ）を参照してください。

手順の概要

1. **configure**
2. **router pim** [**address-family** {**ipv4** | **ipv6**}]
3. **rp-address** *ip-address* [*group-access-list*]**override**]
4. **old-register-checksum**
5. **exit**
6. {**ipv4** | **ipv6**} **access-list** *name*
7. [*sequence-number*] **permit** *source* [*source-wildcard*]
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**router pim** [**address-family** {**ipv4**	**ipv6**}] 例： `RP/0/RSP0/CPU0:router(config)# router pim`	PIM コンフィギュレーション モード、または PIM アドレス ファミリ 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	**rp-address** *ip-address* [*group-access-list*]**override**] 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# rp-address 172.16.6.22 rp-access`	マルチキャスト グループに RP を割り当てます。 • **group-access-list-number** の値を指定する場合、**ipv4 access-list** コマンドを使用して、アクセス リストを設定する必要があります。
ステップ 4	**old-register-checksum** 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4)# old-register-checksum`	（任意）古いレジスタ チェックサム方式が使用される RP の下位互換性を許可します。

	コマンドまたはアクション	目的
ステップ 5	**exit** 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4)# exit`	PIM コンフィギュレーション モードを終了し、ルータを送信元コンフィギュレーション モードに戻します。
ステップ 6	{**ipv4** | **ipv6**} **access-list** *name* 例： `RP/0/RSP0/CPU0:router(config)# ipv4 access-list rp-access`	（任意）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、RP アクセス リストを設定します。 • 「rp-access」という名前のアクセス リストが、マルチキャスト グループ 239.1.1.0 0.0.255.255 を許可します。
ステップ 7	[*sequence-number*] **permit** *source* [*source-wildcard*] 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# permit 239.1.1.0 0.0.255.255`	（任意）「rp-access」リストのマルチキャスト グループ 239.1.1.0 0.0.255.255 を許可します。 ヒント ステップ 6, （122 ページ） とステップ 7, （122 ページ） のコマンドは 1 つのコマンド ストリングに統合でき、グローバル コンフィギュレーション モードから ipv4 access-list rp-access permit 239.1.1.0 0.0.255.255 のように実行します。

ステップ 8	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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。

グループから RP へのマッピングを自動化するための Auto-RP の設定

この作業では、Auto-RP メカニズムを、ネットワークでグループから RP へのマッピングの配信を自動化するように設定します。 Auto-RP を実行するネットワークで、1 台以上のルータが RP 候補として動作し、他のルータが RP マッピング エージェントとして動作している必要があります。 Cisco ASR 9000 シリーズ ルータの VRF インターフェイスは Auto-RP 候補 RP にはできません。

Auto-RP の詳細については、Auto-RP, （82 ページ）を参照してください。

手順の概要

1. **configure**
2. **router pim** [**address-family ipv4**]
3. **auto-rp candidate-rp** *type instance* **scope** *ttl-value* [**group-list** *access-list-name*] [**interval** *seconds*]
4. **auto-rp mapping-agent** *type number* **scope** *ttl-value* [**interval** *seconds*]
5. **exit**
6. **ipv4 access-list** *name*
7. [*sequence-number*] **permit** *source* [*source-wildcard*]
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **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。

	コマンドまたはアクション	目的
ステップ 2	**router pim** [**address-family ipv4**] 例： `RP/0/RSP0/CPU0:router(config)# router pim`	PIM コンフィギュレーション モード、または PIM アドレス ファミリ 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	**auto-rp candidate-rp** *type instance* **scope** *ttl-value* [**group-list** *access-list-name*] [**interval** *seconds*] 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4)# auto-rp candidate-rp GigabitEthernet0/1/0/1 scope 31 group-list 2`	CISCO-RP-ANNOUNCE マルチキャスト グループ（224.0.1.39）にメッセージを送信する RP 候補を設定します。 • 次に、最大ホップ数が 31 であるすべての PIM 対応インターフェイスから RP アナウンスを送信する例を示します。 ルータを RP として識別するために使用される IP アドレスは、GigabitEthernet インターフェイス 0/1/0/1 に関連付けられた IP アドレスです。 • アクセス リスト 2 はこのルータが RP として機能しているグループを示しています。 • **group-list** を指定する場合、任意の **access-list** コマンドを設定する必要があります。
ステップ 4	**auto-rp mapping-agent** *type number* **scope** *ttl-value* [**interval** *seconds*] 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4)# auto-rp mapping-agent GigabitEthernet0/1/0/1 scope 20`	ルータを指定したインターフェイス上の RP マッピング エージェントとして設定します。 • ルータが RP マッピング エージェントとして設定され、CISCO-RP-ANNOUNCE（224.0.1.39）グループを通じた RP からグループへのマッピングを決定した後、ルータは、既知のグループ CISCO-RP-DISCOVERY（224.0.1.40）に Auto-RP 検出メッセージでマッピングを送信します。 • PIM DR はこの既知のグループをリッスンし、使用する RP を決定します。 • 次に、Auto-RP 検出メッセージを 20 ホップに制限する例を示します。
ステップ 5	**exit** 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4)# exit`	PIM コンフィギュレーション モードを終了し、ルータを送信元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 6	**ipv4 access-list** *name* 例： `RP/0/RSP0/CPU0:router(config)# ipv4 access-list 2`	（任意）RP アクセス リストを定義します。
ステップ 7	[*sequence-number*] **permit** *source* [*source-wildcard*] 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# permit 239.1.1.1 0.0.0.0`	（任意）RP アクセス リストのマルチキャスト グループ 239.1.1.1 を許可します。 ヒント　ステップ 6, （125 ページ） とステップ 7, （125 ページ） のコマンドは 1 つのコマンド ストリングに統合でき、グローバル コンフィギュレーション モードから ipv4 access-list rp-access permit 239.1.1.1 0.0.0.0 のように実行します。

ステップ 8	次のいずれかのコマンドを使用します。
	
	• **end**
	
	• **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。
		
		• 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。

ブートストラップ ルータの設定

このタスクでは、1 つ以上の候補ブートストラップ ルータ（BSR）と BSR マッピング エージェントを設定します。 また、ネットワーク バックボーンの一部で候補 BSR を接続し、検出します。

BSR の詳細については、PIM ブートストラップ ルータ，（83 ページ）を参照してください。

手順の概要

1. **configure**
2. **router pim** [**address-family** {**ipv4** | **ipv6**}]
3. **bsr candidate-bsr** *ip-address* [**hash-mask-len** *length*] [**priority** *value*]
4. **bsr candidate-rp** *ip-address* [**group-list** *access-list* **interval** *seconds*] [**priority** *value*]
5. **interface** *type interface-path-id*
6. **bsr-border**
7. **exit**
8. **exit**
9. {**ipv4** | **ipv6**} **access-list** *name*
10. 次のいずれかを実行します。
    - [*sequence-number*] **permit** *source* [*source-wildcard*]
    - [*sequence-number*] **permit** *source-prefix dest-prefix*
11. 次のいずれかのコマンドを使用します。
    - **end**
    - **commit**
12. **clear pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **bsr**
13. **show pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **bsr candidate-rp**
14. **show pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **bsr election**
15. **show pim** [**vrf** *vrf-name*][**ipv4** | **ipv6**] **bsr rp-cache**
16. **show pim** [**vrf** *vrf-name*][**ipv4** | **ipv6**] **group-map** [*ip-address-name*] [**info-source**]

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	**router pim** [**address-family** {**ipv4** | **ipv6**}] 例： `RP/0/RSP0/CPU0:router(config)# router pim`	PIM コンフィギュレーション モード、またはアドレス ファミリ 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	**bsr candidate-bsr** *ip-address* [**hash-mask-len** *length*] [**priority** *value*] 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# bsr candidate-bsr 10.0.0.1 hash-mask-len 30`	ルータが BSR として候補であることをアナウンスするよう設定します。
ステップ 4	**bsr candidate-rp** *ip-address* [**group-list** *access-list* **interval** *seconds*] [**priority** *value*] 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# bsr candidate-rp 172.16.0.0 group-list 4 bidir`	ルータが自身を PIM バージョン 2 の候補 RP として BSR にアドバタイズするよう設定します。 • グループ リスト 4 の設定については、ステップ 9, （128 ページ） を参照してください。
ステップ 5	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# interface GigE 0/1/0/0`	（任意）PIM プロトコル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 6	**bsr-border** 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4-if)# bsr-border`	（任意）Protocol Independent Multicast（PIM）ルータ インターフェイスでのブートストラップ ルータ（BSR）メッセージの転送を停止します。
ステップ 7	**exit** 例： `RP/0/RSP0/CPU0:router(config-pim-ipv4-if)# exit`	（任意）PIM 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、ルータを PIM コンフィギュレーション モードに戻します。
ステップ 8	**exit** 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# exit`	PIM コンフィギュレーション モードを終了し、ルータをグローバル 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 9	{**ipv4** | **ipv6**} **access-list** *name* 例： `RP/0/RSP0/CPU0:router(config)# ipv4 access-list 4`	（任意）BSR に候補グループ リストを定義します。 • アクセス リスト番号 4 は候補 RP アドレス 172.16.0.0 に関連付けられたグループ プレフィックスを指定します。 （ステップ 4, （127 ページ） を参照）。 • この RP は、プレフィックスが 239 であるグループを処理します。
ステップ 10	次のいずれかを実行します。 • [*sequence-number*] **permit** *source* [*source-wildcard*] • [*sequence-number*] **permit** *source-prefix dest-prefix* 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# permit 239.1.1.1 0.255.255.255`	（任意）候補グループ リストのマルチキャスト グループ 239.1.1.1 を許可します。 ヒント ステップ 6, （127 ページ） とステップ 7, （127 ページ） のコマンドは 1 つのコマンド ストリングに統合でき、グローバル コンフィギュレーション モードから ipv4 access-list rp-access permit 239.1.1.1 0.255.255.255 のように実行します。

ステップ 11	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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 12	**clear pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **bsr** 例： `RP/0/RSP0/CPU0:router# clear pim bsr`	（任意）PIM RP グループ マッピング キャッシュから BSR エントリを削除します。
ステップ 13	**show pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **bsr candidate-rp** 例： `RP/0/RSP0/CPU0:router# show pim bsr candidate-rp`	（任意）BSR の PIM 候補 RP 情報を表示します。
ステップ 14	**show pim** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **bsr election** 例： `RP/0/RSP0/CPU0:router# show pim bsr election`	（任意）BSR の PIM 候補選択情報を表示します。
ステップ 15	**show pim** [**vrf** *vrf-name*][**ipv4** | **ipv6**] **bsr rp-cache** 例： `RP/0/RSP0/CPU0:router# show pim bsr rp-cache`	（任意）BSR の PIM RP キャッシュ情報を表示します。
ステップ 16	**show pim** [**vrf** *vrf-name*][**ipv4** | **ipv6**] **group-map** [*ip-address-name*] [**info-source**] 例： `RP/0/RSP0/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。

ルートごとのレートの計算

この手順は、VRF ファミリ単位でマルチキャスト ハードウェア転送レート カウンタをイネーブルにします。

手順の概要

1. **configure**
2. **multicast-routing** [**vrf** *vrf-name*] [**address-family** {**ipv4** | **ipv6**}]
3. **rate-per-route**
4. **interface** {*type interface-path-id* | **all**} **enable**
5. 次のいずれかを実行します。
   - **accounting per-prefix**
   - **accounting per-prefix forward-only**
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
7. **show mfib** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **route** [**rate** | **statistics**] [***** | *source-address*] [*group-address* [/*prefix-length*] [**detail** | **old-output**] | **summary**] [**location** *node-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	**multicast-routing** [**vrf** *vrf-name*] [**address-family** {**ipv4**	**ipv6**}] 例： `RP/0/RSP0/CPU0:router(config)# multicast-routing address-family ipv4`	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 がデフォルトでイネーブルです。
ステップ 3	**rate-per-route** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# rate-per-route`	特定のルートの (S,G) ごとのレート計算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	**interface** {*type interface-path-id* | **all**} **enable**
	
	例：
	`RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# interface all enable`
	
	または
	
	`RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# interface FastEthernet0/3/3/1 enable`
		すべてのインターフェイスでマルチキャスト ルーティングをイネーブルにします。

ステップ 5	次のいずれかを実行します。 • **accounting per-prefix** • **accounting per-prefix forward-only** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv)# accounting per-prefix`	• ハードウェアに存在するプレフィックス単位のカウンタをイネーブルにします。これにより、既存および新規のすべての (S,G) ルートにおいて、入力ルートには転送カウンタ、パント カウンタ、およびドロップ カウンタが割り当てられ、出力ルートには転送カウンタとパント カウンタが割り当てられます。 (*,G) ルートには単一カウンタが割り当てられます。 • accounting per-prefix：既存および新規のすべての (S,G) ルートにおいて、入力の 3 つのカウンタ（転送カウンタ、パント カウンタ、ドロップ カウンタ）、および出力の 2 つのカウンタ（転送カウンタとパント カウンタ）をイネーブルにします。 (*,G) ルートには単一カウンタが割り当てられます。 • accounting per-prefix forward-only：ハードウェアの統計情報リソースを節約するため、ハードウェアで入力、出力でそれぞれ 1 つずつのカウンタをイネーブルにします。（マルチキャスト VPN ルーティング構成、またはルート集約型の構成を使用するラインカードに推奨されます）。
ステップ 6	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# 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

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	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。 • 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 7	**show mfib** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **route** [**rate** | **statistics**] [***** | *source-address*] [*group-address* [*/prefix-length*] [**detail** | **old-output**] | **summary**] [**location** *node-id*] 例： `RP/0/RSP0/CPU0:router# show mfib vrf 12 route statistics location 0/1/cpU0`	マルチキャスト転送情報ベース（MFIB）テーブルのルート エントリを表示します。 • **rate** キーワードを *source-* および *group-address* とともに使用した場合、コマンドは、マルチキャスト転送情報ベース（MFIB）テーブル内のすべてのラインカードのルートあたりの累積レートを表示します。 • **statistics** キーワードが使用されている場合、このコマンドはマルチキャスト転送情報ベース（MFIB）テーブル内の 1 つのラインカードについてルートあたりのレートを表示します。

マルチキャスト ノンストップ フォワーディングの設定

このタスクでは、ネットワーク障害や、ソフトウェアのアップグレードとダウングレードを軽減するために、マルチキャスト パケット転送のノンストップ フォワーディング（NSF）機能を設定します。

NSF ライフタイムのデフォルト値を使用することを強くお勧めしますが、任意のステップ 4, （134 ページ） からステップ 9, （135 ページ） では、Protocol Independent Multicast（PIM）およびインターネット グループ管理プロトコル（IGMP）またはマルチキャスト リスナー検出（MLD）の NSF タイムアウト値を変更できます。 これらのコマンドは、PIM および IGMP または MLD がデフォルト以外の間隔か、join および prune 操作のクエリー間隔を使用して設定されている場合に使用します。

通常、IGMP NSF と PIM NSF のライフタイム値を同じに設定するか、クエリーまたは join クエリー間隔を超えるように設定します。 たとえば IGMP クエリー時間を 120 秒に設定する場合、IGMP NSF ライフタイムを 120 秒以上に設定します。

NSF がルータでイネーブルになった後、Cisco IOS XR Software コントロール プレーンが収束および再接続しない場合、マルチキャスト パケットの転送は最大 15 分継続され、その後パケット転送が停止されます。

はじめる前に

NSF がマルチキャスト ネットワークで動作するためには、PIM がリバース パス転送（RPF）情報を取得するユニキャスト プロトコル（IS-IS、OSPF、および BGP など）でも NSF をイネーブルにする必要があります。 ユニキャスト プロトコルに NSF を設定する方法については、該当するコンフィギュレーション モジュールを参照してください。

手順の概要

1. **configure**
2. **multicast-routing** [**address-family** {**ipv4** | **ipv6**}]
3. **nsf** [**lifetime** *seconds*]
4. **exit**
5. **router pim** [**address-family** {**ipv4** | **ipv6**}]
6. **nsf lifetime** *seconds*
7. **exit**
8. **router** {**igmp** | **mld**}
9. **nsf lifetime** *seconds*
10. 次のいずれかのコマンドを使用します。
    - **end**
    - **commit**
11. **show** {**igmp nsf**
12. **show mfib** [**ipv4** | **ipv6**] **nsf** [**location** *node-id*]
13. **show mrib** [**ipv4** | **ipv6**] **nsf**
14. **show pim** [**ipv4** | **ipv6**] **nsf**

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	**multicast-routing** [**address-family** {**ipv4** | **ipv6**}] 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 がデフォルトでイネーブルです。
ステップ 3	**nsf** [**lifetime** *seconds*] 例： `RP/0/RSP0/CPU0:router(config-mcast)# nsf`	マルチキャスト ルーティング システムの NSF 機能をオンにします。
ステップ 4	**exit** 例： `RP/0/RSP0/CPU0:router(config-mcast)# exit`	（任意）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、ルータを送信元コンフィギュレーション モードに戻します。
ステップ 5	**router pim** [**address-family** {**ipv4** | **ipv6**}] 例： `RP/0/RSP0/CPU0:router(config)# router pim address-family ipv4`	（任意）PIM アドレス ファミリ 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 6	**nsf lifetime** *seconds* 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# nsf lifetime 30`	（任意）PIM プロセスでマルチキャスト転送ルート エントリの NSF タイムアウト値を設定します。 （注） デフォルト以外の値に PIM hello 間隔を設定した場合は、PIM NSF ライフタイムを hello ホールド タイムよりも小さい値に設定します。 通常、ホールド タイム フィールドの値はインターバル値の 3.5 倍となります。PIM hello インターバルが 30 秒の場合、ホールドタイムは 120 秒となります。
ステップ 7	**exit** 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv4)# exit`	（任意）PIM コンフィギュレーション モードを終了し、ルータを送信元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	**router** {**igmp** | **mld**} 例： `RP/0/RSP0/CPU0:router(config)# router igmp`	（任意）ルータ IGMP または MLD コンフィギュレーション モードを開始します。
ステップ 9	**nsf lifetime** *seconds* 例： `RP/0/RSP0/CPU0:router(config-igmp)# nsf lifetime 30`	（任意）IGMP プロセスでマルチキャスト転送ルート エントリの NSF タイムアウト値を設定します。

ステップ 10	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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 11	**show** {**igmp nsf** 例： `RP/0/RSP0/CPU0:router# show igmp nsf`	（任意）IGMP での NSF の動作状態を表示します。

	コマンドまたはアクション	目的
ステップ 12	**show mfib** [**ipv4** | **ipv6**] **nsf** [**location** *node-id*] 例： `RP/0/RSP0/CPU0:router# show mfib nsf`	（任意）MFIB ラインカードでの NSF の動作状態を表示します。
ステップ 13	**show mrib** [**ipv4** | **ipv6**] **nsf** 例： `RP/0/RSP0/CPU0:router# show mrib nsf`	（任意）MRIB での NSF の動作状態を表示します。
ステップ 14	**show pim** [**ipv4** | **ipv6**] **nsf** 例： `RP/0/RSP0/CPU0:router# show pim nsf`	（任意）PIM での NSF の動作状態を表示します。

マルチキャスト VPN の設定

- マルチキャスト ルーティングの VPN のイネーブル化, （138 ページ）（必須）
- 「Configuring BGP to Advertise VRF Routes for Multicast VPN from PE to PE」（必須）

  『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』のモジュール「Implementing BGP on Cisco IOS XR Software」を参照してください。
- BGP での MDT アドレス ファミリ セッションの PE 間プロトコルとしての設定（PIM-SM MDT グループでは任意、PIM-SSM MDT グループでは必須）

  『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』の「Configuring an MDT Address Family Session in BGP」の項を参照してください。
- プロバイダー エッジとカスタマー エッジの間のプロトコルの設定（任意）

  『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』の「Configuring BGP as a PE-CE Protocol」、「Configuring OSPF as a PE-to-CE Protocol」および「Configuring EIGRP as a PE-to CE Protocol」の項を参照してください。
- PIM VRF インスタンスの指定, （141 ページ）（任意）

マルチキャスト VPN の前提条件

- PIM およびマルチキャスト転送はマルチキャスト トラフィックで使用されるすべてのインターフェイスで設定する必要があります。 MVPN では、次のインターフェイスの PIM とマルチキャスト転送をイネーブルにする必要があります。
  - バックボーンに接続されているプロバイダー エッジ（PE）ルータの物理インターフェイス
  - BGP ピアリングの送信元アドレスに使用するインターフェイス。
  - PIM ランデブー ポイントに設定されているインターフェイス。

（注） PIM およびマルチキャスト転送は、マルチキャスト ルーティング コンフィギュレーション モードでイネーブルになります。 PIM プロトコルをイネーブルにするために、ルータ PIM モードでの追加設定は必要ありません。

- マルチキャスト トラフィックの転送で使用するために設計された VPN のインターフェイスでは、PIM およびマルチキャスト転送をイネーブルにする必要があります。
- マルチキャスト トラフィックの送受信を行うすべてのルータでは、BGP を設定して動作させる必要があります。
- MVPN を有効にするには、BGP 設定に VPN IPv4 VPN アドレス ファミリ（AFI）を含める必要があります。 マルチキャスト ルーティングのマルチキャスト VPN の制約事項, （137 ページ）を参照してください。 （『Cisco IOS XR Routing Configuration Guide』の「Enabling BGP Routing」の項も参照してください）。
- マルチキャスト ドメインのすべての PE ルータでは、MVPN をサポートする Cisco IOS XR Software イメージを実行する必要があります。
- マルチキャスト転送はグローバル IPv4 アドレス ファミリに対して設定する必要があります。
- 各マルチキャスト SM VRF ドメインには、PIM ランデブー ポイント（RP）定義が関係付けられていることが必要です。 Auto-RP とブートストラップ ルータ（BSR）を使用して、カスタマー エッジ（CE）デバイス上での MVPN サービスの RP を設定できます（MVPN が RP を動的に学習するため）。 VRF インターフェイスは PE デバイス上のリスナーとして使用できます。

  スタティック RP サービスをイネーブルにするには、ドメイン内の各デバイスをこの目的のために設定する必要があります。

マルチキャスト ルーティングのマルチキャスト VPN の制約事項

- VRF 単位の MDT ソースの設定は IPv4 のみでサポートされます。

- MDT グループ アドレスは、同じ VRF の両方のアドレス ファミリに対して同じにする必要があります。

マルチキャスト ルーティングの VPN のイネーブル化

ここでは、IPv4 のマルチキャスト VPN ルーティングをイネーブルにします。

MDT グループ アドレスが、MDT の仮想的な PIM の「ネイバーシップ」を構成するために、プロバイダーエッジ（PE）ルータによって使用されます。 これにより、PE が、VRF 内の他の PE と、LAN を共有しているかのように通信できるようになります。

カスタマー VRF トラフィックを送信するときに、PE はトラフィックを自身の (S,G) 状態にカプセル化します。ここで、G は MDT グループ アドレス、S は PE の MDT 送信元です。 PE ネイバーの (S,G) MDT と結合することにより、PE ルータはその VRF のカプセル化されたマルチキャスト トラフィックを受信できます。

つまり、VRF 自体は多くのグループに送信する多くのマルチキャスト送信元がありますが、プロバイダー ネットワークは VRF ごとに 1 つのグループ、つまり MDT グループの状態のみをインストールする必要があります。

手順の概要

1. **configure**
2. **multicast-routing**
3. **address-family ipv4**
4. **nsf**
5. **mdt source** *type interface-path-id*
6. **interface all enable**
7. **vrf** *vrf-name*
8. **vrf** *vrf_A* [**address-family** {**ipv4**}]
9. **mdt default** *mdt-group-address*
10. **mdt data** *mdt-group-address/prefix-length* **threshold** *threshold acl-name*
11. **interface all enable**
12. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**multicast-routing** 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。
ステップ 3	**address-family ipv4** 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4 アドレス ファミリ サブモードを開始します。
ステップ 4	**nsf** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# nsf`	ノンストップ フォワーディング（NSF）を、マルチキャスト プロセスに障害が発生した場合に転送状態を維持するように設定します。
ステップ 5	**mdt source** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# mdt source GigE 0/1/0/0`	MDT 送信元アドレスを指定します。 （注） MDT 送信元インターフェイスの名前は、BGP ピアリングに使用するものと同じ名前にする必要があります。
ステップ 6	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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。個々のインターフェイスをイネーブルにすることもできます。 **注意** リバース パス転送（RPF）障害の可能性を回避するには、マルチキャスト トラフィックを伝送することがあるインターフェイスを予防的にイネーブルにする必要があります。

	コマンドまたはアクション	目的
ステップ 7	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-)# vrf vrf_A`	VPN ルーティングおよび転送（VRF）インスタンスを設定し、VRF 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	**vrf** *vrf_A* [**address-family** {**ipv4**}]	IPv4 アドレス ファミリの仮想ルーティングおよび転送インスタンスを指定します。
ステップ 9	**mdt default** *mdt-group-address* 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf_A-ipv4)# mdt default 239.23.2.1`	マルチキャスト配信ツリー（MDT）デフォルト グループ アドレスを指定します。
ステップ 10	**mdt data** *mdt-group-address/prefix-length* **threshold** *threshold acl-name* 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf_A-ipv4)# mdt data 239.23.3.0/24 threshold 1200 acl-A`	（IPv4 MVPN 構成のみ）データ MDT トラフィックで使用されるマルチキャスト グループのアドレス範囲を指定します。 （注） このグループ範囲は、MDT デフォルト グループと重複してはなりません。 これは任意のコマンドです。トラフィックがデータ MDT グループを使用して送信される、デフォルトのしきい値は 1 kbps です。ただし、必要に応じて、より大きなしきい値を設定できます。 また、必要に応じてデータ MDT グループを介してトンネリングされるグループの数を制限するアクセス リストを設定できます。アクセス リストに含まれていないグループからのトラフィックは、デフォルト MDT グループを使用してトンネリングされ続けます。
ステップ 11	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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 12	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

PIM VRF インスタンスの指定

スパース モードの Protocol Independent Multicast（PIM-SM）を MVPN で設定する場合は、ランデブー ポイント（RP）も設定する必要があります。 ここでは、オプションの PIM VPN インスタンスを指定します。

手順の概要

1. **configure**
2. **router pim vrf** *vrf-name* **address-family** {**ipv4** | **ipv6**}
3. **rp-address** *ip-address* [*group-access-list-name*] **override**]
4. 次のいずれかのコマンドを使用します。
   • **end**
   • **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	**router pim vrf** *vrf-name* **address-family** {**ipv4** | **ipv6**} 例： `RP/0/RSP0/CPU0:router(config)# router pim vrf vrf_A address-family ipv4`	PIM アドレス ファミリ コンフィギュレーション サブモードを開始し、IPv4 または IPv6 アドレス ファミリの PIM VRF を設定します。
ステップ 3	**rp-address** *ip-address* [*group-access-list-name*] **override**] 例： `RP/0/RSP0/CPU0:router(config-pim-vrf_A-ipv4)# rp-address 10.0.0.0`	PIM ランデブー ポイント（RP）アドレスを設定します。 • **group-access-list-name** には、特定の RP にマッピングするグループのアクセス リストを指定します。 • **bidir** = 双方向 RP を指定します。 • **override** は、スタティック RP 設定が自動 RP およびブートストラップ ルータ（BSR）を上書きすることを指定します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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。

IGMP VRF インスタンスの指定

手順の概要

1. **configure**
2. **router igmp**
3. **vrf** *vrf-name*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**router igmp** 例： `RP/0/RSP0/CPU0:router(config)# router igmp`	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 3	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-igmp)# vrf vrf_B`	VRF インスタンスを設定します。
ステップ 4	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

VRF ごとの MDT 送信元の設定

このオプション機能はデフォルト VRF で設定された BGP ピアリング内のループバックを通じてすべてのユニキャスト トラフィックをルーティングする、マルチキャスト VPN ネットワーク トポロジのデフォルト ルーティング メカニズムを変更できます。 代わりに、デフォルト VRF ではなく、特定の VRF を使用して MDT 送信元を指定できるループバックを設定できます。 これは、現在動作を上書きし、MDT グループの一部として BGP を更新します。 BGP は、MDT SAFI および VPN IPv4 アップデートの送信元とコネクタ属性を変更します。

MDT送信元が設定されていないVRFに対して、デフォルトVRFのMDT送信元が適用されます。また、VRF の MDT 送信元が未設定の場合、MDT 送信元のデフォルトの VRF 設定が有効になります。

（注） 次の設定では、デフォルト VRF はステップ 3 での明示的な参照を必要としません。

手順の概要

1. **configure**
2. **multicast-routing**
3. **mdt source loopback** *interface-path-id*
4. **vrf** *vrf-name* **mdt source loopback** *interface-path-id*
5. 上記手順を、他の VRF を作成するために必要な回数だけ繰り返します。
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
7. **show pim vrf all mdt interface**

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**
	例：
	`RP/0/RSP0/CPU0:router(config)# multicast-routing`
	`RP/0/RSP0/CPU0:router(config-mcast)#`
		IP 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 3	**mdt source loopback** *interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-mcast)# mdt source loopback 0`	デフォルトの VRF を使用して、MVPN の MDT の送信元アドレスを設定するために使用されるインターフェイスを設定します。 （注） デフォルトの VRF に明示的なコマンドは不要です。つまり暗黙的なコマンドになります。
ステップ 4	**vrf** *vrf-name* **mdt source loopback** *interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-mcast)# vrf 101 mdt source loopback 1`	デフォルト VRF を上書きするため、ループバックで特定の VRF を指定することで 2 番目のインターフェイスを設定します。
ステップ 5	上記手順を、他の VRF を作成するために必要な回数だけ繰り返します。	—

	コマンドまたはアクション	目的
	例： `RP/0/RSP0/CPU0:router(config-mcast)# vrf 102 mdt source loopback 2`	

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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 7	**show pim vrf all mdt interface**
	例：
	`RP/0/RSP0/CPU0:router# show pim vrf all mdt interface`
	`multicast-routing`
	` vrf default address-family ipv4`
	`  mdt source Loopback0`
	`!`
	` vrf 101 address-family ipv4`
	`  mdt default ipv4 239.1.1.1`
	`  mdt source Loopback1`
	`!`
	` vrf 102 address-family ipv4`
	`  mdt default ipv4 239.1.1.2`
	`  mdt source Loopback2`
	` !`
	` vrf 103 address-family ipv4`
		すべての MDT データ ストリームを表示します。
		この例では、ループバック 1 は、VRF ごとの MDT 送信元です。

	コマンドまたはアクション	目的
	mdt default ipv4 239.1.1.3 !	

ラベル スイッチド マルチキャストの設定

LSM MLDP based MVPN の展開には、デフォルトの MDT と 1 つ以上のデータ MDT の設定が含まれます。 各マルチキャスト ドメインに対してデフォルトのスタティック MDT が確立されます。デフォルト MDT は、PE ルータがマルチキャスト ドメインにある他の PE ルータに、マルチキャスト データとコントロール メッセージを送信するために使用するパスを定義します。 デフォルト MDT は、単一の MP2MP LSP を使用してコア ネットワークに作成されます。

また、LSP MLDP based MVPN は、高帯域幅の送信用にデータ MDT の動的な作成をサポートします。 レートの高いデータ ソースの場合、ストリームに属さない PE への帯域幅を無駄に廃棄しないよう、デフォルト MDT からのトラフィックをオフロードするため、P2MP LSP を使用してデータ MDT が作成されます。 イントラネットとエクストラネットの両方に MLDP MVPN を設定できます。 この設定の項では Rosen ベースの MLDP のプロファイルについて説明します。 その他の MLDP プロファイルの設定例については、LSM based MLDP の設定例, （268 ページ）を参照してください。

（注） MLDP based MVPN を設定する前に、コア側のインターフェイスで MPLS がイネーブルであることを確認します。 MPLS の設定の詳細については、『Cisco IOS XR MPLS Configuration Guide』を参照してください。 または、コア ルータで BGP および任意の Interior Gateway Protocol（OSPF または ISIS）がイネーブルであることを確認します。 BGP およびルート ポリシーの設定の詳細については、『Cisco IOS XR Routing Configuration Guide』を参照してください。

ラベル スイッチド マルチキャ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. **mpls ldp mldp**
3. **vrf** *vrf_name*
4. **address-family** [**ipv4** | **ipv6**] **unicast**
5. **import route-target** [*xx.yy.nn* | *as-number:nn* | *ip-address:nn*]
6. **export route-target** [*xx.yy.nn* | *as-number:nn* | *ip-address:nn*]
7. **vpn id** *vpn-id*
8. **multicast-routing vrf** *vrf_name*
9. **mdt default mldp ipv4** *root-node*
10. **mdt data** *mdt-group-address* **threshold** *value*
11. **router bgp**
12. **rd** *route-distinguisher*
13. **address-family ipv4 mdt**
14. **address-family vpnv4 unicast**
15. **router pim**
16. **vrf** *vrf_name*
17. **address-family** [**ipv4** | **ipv6**]
18. **rpf topology route-policy** *route_policy_name*
19. **route-policy** *route_policy_name*
20. **set core-tree** *tree_type*
21. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**mpls ldp mldp** 例： `RP/0/RSP0/CPU0:router(config)# mpls ldp mldp`	MPLS MLDP サ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 3	**vrf** *vrf_name* 例： `RP/0/RSP0/CPU0:router(config-ldp-mldp)# vrf vrf1`	VRF インスタンスを設定します。 vrf-name 引数は、VRF に割り当てる名前です。
ステップ 4	**address-family** [**ipv4** | **ipv6**] **unicast** 例： `RP/0/RSP0/CPU0:router(config-vrf)# address-family ipv4 unicast`	アドレス ファミリ サブモードを開始します。
ステップ 5	**import route-target** [*xx.yy.nn* | *as-number:nn* | *ip-address:nn*] 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# route-target import 100:102`	任意で次のいずれかで表現される選択したルート ターゲットをインポートします。 • xx.yy:nn 形式の、ルート ターゲットの 4 バイト AS 番号。 範囲は 0 ～ 65535.0 ～ 65535:0 ～ 65535 です。 • ルート ターゲット AS 番号（nn 形式）。 範囲は 0 ～ 65535 です。 • ルート ターゲットの IP アドレス（A.B.C.D. 形式）。
ステップ 6	**export route-target** [*xx.yy.nn* | *as-number:nn* | *ip-address:nn*] 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# route-target export 100:102`	任意で次のいずれかで表現される選択したルート ターゲットをエクスポートします。 • xx.yy:nn 形式の、ルート ターゲットの 4 バイト AS 番号。 範囲は 0 ～ 65535.0 ～ 65535:0 ～ 65535 です。 • ルート ターゲット AS 番号（nn 形式）。 範囲は 0 ～ 65535 です。 • ルート ターゲットの IP アドレス（A.B.C.D. 形式）。
ステップ 7	**vpn id** *vpn-id* 例： `RP/0/RSP0/CPU0:router(config-vrf)# vpn id 10:3`	VRF で VPN ID を設定または更新します。

	コマンドまたはアクション	目的
ステップ 8	**multicast-routing vrf** *vrf_name* 例： `RP/0/RSP0/CPU0:router(config)# multicast-routing vrf vrf1`	指定されたVRFのマルチキャストルーティングをイネーブルにします。
ステップ 9	**mdt default mldp ipv4** *root-node* 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf)# mdt default mldp ipv4 2.2.2.2`	VRF に MLDP MDT を設定します。 ルート ノードは、プロバイダー ネットワーク内の任意のルータ（ソース PE、レシーバPE、またはコアルータ）にあるループバックまたは物理インターフェイスのIPアドレスにすることができます。 ルート ノード アドレスは、ネットワーク内のすべてのルータから到達できる必要があります。シグナリングが発生するルータは、ルートノードとして機能します。 デフォルトMDTを各PEルータで設定しないと、PEルータは、この特定 MVRF のマルチキャスト トラフィックを受信できません。 （注） デフォルトでは、MPLS MLDPはイネーブルになります。 ディセーブルにするには、**no mpls ldp mldp** コマンドを使用します。 （注） LSPVIF トンネルが **mdt default mldp** *root-node* コマンドの結果として作成されます。
ステップ 10	**mdt data** *mdt-group-address* **threshold** *value* 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf)# mdt data threshold 20`	データ MDT にしきい値を設定します。
ステップ 11	**router bgp** 例： `RP/0/RSP0/CPU0:router(config)# router bgp`	BGP 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	**rd** *route-distinguisher* 例： `RP/0/RSP0/CPU0:router(config-vrf)# rd 10:3`	ルーティング テーブルと転送テーブルを作成します。route-distinguisher引数で、8バイトの値をIPv4プレフィックスに追加して VPN IPv4 プレフィックスを作成することを指定します。 RD 値は、次のいずれかの形式で入力できます。 • 16ビット自律システム番号。たとえば、101:3と指定します。

	コマンドまたはアクション	目的
		• 32 ビット IP アドレス：16 ビット数値。 たとえば、192.168.122.15:1 と指定します。
ステップ 13	**address-family ipv4 mdt** 例： `RP/0/RSP0/CPU0:router(config)# address-family ipv4 mdt`	BGP MDT アドレス ファミリを設定します。
ステップ 14	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config)# address-family vpnv4 unicast`	BGP VPNv4 アドレス ファミリを設定します。
ステップ 15	**router pim** 例： `RP/0/RSP0/CPU0:router(config)# router pim`	PIM コンフィギュレーション モードを開始します。
ステップ 16	**vrf** *vrf_name* 例： `RP/0/RSP0/CPU0:router(config-pim)# vrf vrf1`	VRF インスタンスを指定します。
ステップ 17	**address-family** [**ipv4** | **ipv6**] 例： `RP/0/RSP0/CPU0:router(config-pim-vrf1)# address-family ipv4`	アドレス ファミリ サブモードを開始します。
ステップ 18	**rpf topology route-policy** *route_policy_name* 例： `RP/0/RSP0/CPU0:router(config-pim-vrf1-af)# rpf topology route-policy rosen_mvpn_mldp`	RPF トポロジ テーブルに特定のルーティング ポリシーを割り当てます。
ステップ 19	**route-policy** *route_policy_name* 例： `RP/0/RSP0/CPU0:router(config)# route-policy route1`	プロファイルのルート ポリシーを設定します。 ルート ポリシーの設定の詳細については、『*Cisco IOS XR Routing Configuration Guide*』を参照してください。

	コマンドまたはアクション	目的
ステップ 20	**set core-tree** *tree_type* 例： `RP/0/RSP0/CPU0:router(config-rpl)# set core-tree mldp-rosen`	ルート ポリシーの MDT タイプを指定します。

ステップ 21	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

LSM mLDP based MVPN の設定の検証

次のコマンドを使用して、LSM mLDP based MVPN のイントラネットの設定を確認します。

• MLDP ネイバーをチェックするには、**show mpls mldp neighbor** コマンドを使用します。

```
Router# show mpls mldp neighbors
mLDP neighbor database
 MLDP peer ID      : 1.0.0.1:0, uptime 15:36:30 Up,
  Capabilities     : GR, Typed Wildcard FEC, P2MP, MP2MP, MBB
  Target Adj       : No
  Upstream count   : 0
  Branch count     : 0
```

```
 LDP GR           : Enabled
                  : Instance: 1
 Label map timer  : never
 Policy filter in : None
 Path count       : 1
 Path(s)          : 11.11.11.10       GigabitEthernet0/2/0/0 LDP
 Adj list         : 11.11.11.10       GigabitEthernet0/2/0/0
 Peer addr list   : 8.39.21.2
                  : 1.0.0.1
                  : 1.1.1.1
                  : 1.2.2.1
                  : 1.3.3.1
                  : 1.4.4.1
                  : 1.5.5.1
                  : 1.6.6.1
                  : 1.7.7.1
                  : 1.8.8.1
                  : 1.9.9.1
                  : 1.10.10.1
                  : 1.11.11.1
                  : 1.12.12.1
                  : 1.13.13.1
                  : 1.14.14.1
                  : 1.15.15.1
                  : 1.16.16.1
                  : 1.17.17.1
                  : 1.18.18.1
                  : 1.19.19.1
                  : 1.20.20.1
                  : 1.21.21.1
                  : 1.22.22.1
                  : 1.23.23.1
                  : 1.24.24.1
                  : 1.25.25.1
                  : 1.26.26.1
                  : 1.27.27.1
                  : 1.28.28.1
                  : 1.29.29.1
                  : 1.30.30.1
                  : 11.11.11.10
                  : 111.113.1.5
                  : 111.112.1.1
                  : 8.39.21.222

MLDP peer ID      : 3.0.0.1:0, uptime 15:36:31 Up,
 Capabilities     : GR, Typed Wildcard FEC, P2MP, MP2MP, MBB
 Target Adj       : No
 Upstream count   : 334
 Branch count     : 328
 LDP GR           : Enabled
                  : Instance: 1
 Label map timer  : never
 Policy filter in : None
 Path count       : 1
 Path(s)          : 11.113.1.2        GigabitEthernet0/2/0/3 LDP
 Adj list         : 11.113.1.2        GigabitEthernet0/2/0/3
 Peer addr list   : 8.39.15.2
                  : 3.0.0.1
                  : 189.189.189.189
                  : 13.13.13.18
                  : 11.113.1.2
                  : 22.113.1.2
                  : 111.113.1.6
                  : 112.113.1.6
```

- PIM ネイバーをチェックするには、**show pim vrf vrf-name neighbor** コマンドを使用します。

```
Router# show pim vrf A1_MIPMSI neighbor
PIM neighbors in VRF A1_MIPMSI

Neighbor Address             Interface              Uptime    Expires  DR pri  s
```

```
101.2.2.101*                Loopback2              15:54:43  00:00:02 1 (DR) BP
101.0.0.101*                LmdtA1/MIPMSI          15:54:43  00:00:02 1      B
102.0.0.102                 LmdtA1/MIPMSI          03:52:08  00:00:02 1      B
103.0.0.103                 LmdtA1/MIPMSI          15:28:13  00:00:02 1 (DR) B
60.3.0.1                    Multilink0/2/1/0/3     15:54:39  00:01:21 1      B
60.3.0.2*                   Multilink0/2/1/0/3     15:54:43  00:00:02 1 (DR) BP
60.1.0.5                    Serial0/2/2/0/1:1.16   15:54:42  00:01:42 1      B
60.1.0.6*                   Serial0/2/2/0/1:1.16   15:54:43  00:00:02 1 (DR) BP
60.2.0.1                    Serial0/5/0/0/1        15:54:42  00:01:17 1      B
60.2.0.2*                   Serial0/5/0/0/1        15:54:43  00:00:02 1 (DR) BP
```

- 特定の VRF のマルチキャスト ルートをチェックするには、**show mrib vrf vrf_name route** コマンドを使用します。

```
Router# show mrib vrf A1_MIPMSI route
IP Multicast Routing Information Base
Entry flags: L - Domain-Local Source, E - External Source to the Domain,
    C - Directly-Connected Check, S - Signal, IA - Inherit Accept,
    IF - Inherit From, D - Drop, MA - MDT Address, ME - MDT Encap,
    MD - MDT Decap, MT - MDT Threshold Crossed, MH - MDT interface handle
    CD - Conditional Decap, MPLS - MPLS Decap, MF - MPLS Encap, EX - Extranet
    MoFE - MoFRR Enabled, MoFS - MoFRR State
Interface flags: F - Forward, A - Accept, IC - Internal Copy,
    NS - Negate Signal, DP - Don't Preserve, SP - Signal Present,
    II - Internal Interest, ID - Internal Disinterest, LI - Local Interest,
    LD - Local Disinterest, DI - Decapsulation Interface
    EI - Encapsulation Interface, MI - MDT Interface, LVIF - MPLS Encap,
    EX - Extranet, A2 - Secondary Accept

(*,224.0.0.0/24) Flags: D
  Up: 15:57:19


(*,224.0.1.39) Flags: S
  Up: 15:57:19


(*,224.0.1.40) Flags: S
  Up: 15:57:19

  Outgoing Interface List
    Serial0/5/0/0/1 Flags: II LI, Up: 15:57:12

(*,225.0.0.0/19) RPF nbr: 101.2.2.101 Flags: L C
  Up: 15:57:19

  Outgoing Interface List
    Decapstunnel98 Flags: NS DI, Up: 15:57:10

(*,225.0.32.0/19) RPF nbr: 102.0.0.102 Flags: C
  Up: 15:57:19


(*,225.0.32.1) RPF nbr: 102.0.0.102 Flags: C
  Up: 04:08:30

  Incoming Interface List
    LmdtA1/MIPMSI Flags: A LMI, Up: 04:08:30
  Outgoing Interface List
    Serial0/2/2/0/1:1.16 Flags: F NS, Up: 04:08:30

(*,225.0.32.2) RPF nbr: 102.0.0.102 Flags: C
  Up: 04:08:30

  Incoming Interface List
    LmdtA1/MIPMSI Flags: A LMI, Up: 04:08:30
  Outgoing Interface List
    Serial0/2/2/0/1:1.16 Flags: F NS, Up: 04:08:30

(*,225.0.32.3) RPF nbr: 102.0.0.102 Flags: C
  Up: 04:08:30
```

```
  Incoming Interface List
    LmdtA1/MIPMSI Flags: A LMI, Up: 04:08:30
  Outgoing Interface List
    Serial0/2/2/0/1:1.16 Flags: F NS, Up: 04:08:30

(*,225.0.32.4) RPF nbr: 102.0.0.102 Flags: C
  Up: 04:08:30

  Incoming Interface List
    LmdtA1/MIPMSI Flags: A LMI, Up: 04:08:30
  Outgoing Interface List
    Serial0/2/2/0/1:1.16 Flags: F NS, Up: 04:08:30
```

- MPLS フォワーディング ステータスをチェックするには、**show mpls forwarding** コマンドを使用します。

```
Router# show mpls forwarding
Local  Outgoing    Prefix             Outgoing     Next Hop        Bytes
Label  Label       or ID              Interface                    Switched
------ ----------- ------------------ ------------ --------------- ------------
16000  16255       MLDP LSM ID: 0x1   Gi0/2/0/3    11.113.1.2      348727240
16001  16254       MLDP LSM ID: 0x3   Gi0/2/0/3    11.113.1.2      348727234
16002  16253       MLDP LSM ID: 0x5   Gi0/2/0/3    11.113.1.2      348727234
16003  16252       MLDP LSM ID: 0x7   Gi0/2/0/3    11.113.1.2      348727234
16004  16251       MLDP LSM ID: 0x9   Gi0/2/0/3    11.113.1.2      421876882
16005  16250       MLDP LSM ID: 0xb   Gi0/2/0/3    11.113.1.2      348726916
```

MVPN スタティック P2MP-TE の設定

スタティック ポイントツーマルチポイント（P2MP）トラフィック エンジニアリング（TE）用のマルチキャスト VPN を設定するには、次の手順を実行します。

入力 PE の MVPN P2MP の設定

入力 PE の MVPN P2MP 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. **multicast-routing**
3. **address-family** {**ipv4**|**ipv6**}
4. **mdt source** *type interface-path-id*
5. **interface all enable**
6. **vrf** *vrf-name*
7. **address-family** {**ipv4** | **ipv6**}
8. **bgp auto-discovery rsvpte**
9. **mdt static p2mp-te tunnel-mte** *value*
10. **interface all enable**
11. **router igmp**
12. **vrf** *name*
13. **interface** *type interface-path-id*
14. **static-group** *ip_group_address source-address*
15. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**multicast-routing** 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。
ステップ 3	**address-family** {**ipv4**|**ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4 または IPv6 アドレス ファミリ 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 4	**mdt source** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# mdt source Loopback 0`	MDT 送信元アドレスを指定します。 （注） MDT送信元インターフェイスの名前は、BGP ピアリングに使用するものと同じ名前にする必要があります。
ステップ 5	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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。個々のインターフェイスをイネーブルにすることもできます。
ステップ 6	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-mcast)# vrf vrf1`	VPN ルーティングおよび転送（VRF）インスタンスを設定し、VRF 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	**address-family** {**ipv4** | **ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf1)# address-family ipv4`	IPv4（または IPv6）アドレス ファミリ サブモードを開始します。
ステップ 8	**bgp auto-discovery rsvpte** 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf1-ipv4)# bgp auto-discovery rsvp-te`	RSVP-TE I-PMSI コア ツリーをイネーブルにします。
ステップ 9	**mdt static p2mp-te tunnel-mte** *value* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# mdt static p2mp-te tunnel-mte 1`	スタティック p2mp-te mpls トラフィック エンジニアリング P2MP トンネル インターフェイスを指定します。
ステップ 10	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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。個々のインターフェイスをイネーブルにすることもできます。

	コマンドまたはアクション	目的
ステップ 11	**router igmp** 例： `RP/0/RSP0/CPU0:router(config)# router igmp`	ルータ igmp を設定し、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 12	**vrf** *name* 例： `RP/0/RSP0/CPU0:router(config-igmp)# vrf vrf1`	指定されたアクセス リスト セットの一部としてソースを認識するため、アクセス リストの条件を設定します。各 ACL は、マッピングする一連の SSM グループを記述します。
ステップ 13	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-igmp)# interface tunnel-mte1`	MPLS トラフィック エンジニアリング P2MP トンネル インターフェイスを設定します。
ステップ 14	**static-group** *ip_group_address source-address* 例： `RP/0/RSP0/CPU0:router(config-igmp-default-if)# static-group 232.1.1.1 192.1.1.2`	IGMP のスタティック マルチキャスト グループを設定します。

ステップ 15	次のいずれかのコマンドを使用します。
	• **end**
	• **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

	コマンドまたはアクション	目的
		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。

MVPN P2MP BGP の設定

MVPN P2MP BGP を設定するには、次の作業を実行します。

手順の概要

1. **configure**
2. **router bgp 100**
3. **bgp router-id** *ip_address*
4. **address-family** {**ipv4** | **ipv6**} **unicast**
5. **address-family** {**vpnv4** | **vpnv6**} **unicast**
6. **address-family** {**ipv4** | **ipv6**} **mvpn**
7. **neighbor** *address*
8. **remote-as** *2-byte AS number*
9. **update-source interface** *type interface-path-id*
10. **address-family** {**ipv4** | **ipv6**} **unicast**
11. **address-family** {**vpnv4** | **vpnv6**} **unicast**
12. **address-family** {**ipv4** | **ipv6**} **mvpn**
13. **vrf** *name*
14. **rd** *x.y format*
15. **bgp router-id** *ip_address*
16. **address-family** {**ipv4** | **ipv6**} **unicast**
17. **redistribute connected**
18. **address-family** {**ipv4** | **ipv6**} **mvpn**
19. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**router bgp 100** 例： `RP/0/RSP0/CPU0:router(config)# router bgp 100`	BGP ルーティング プロセスを設定します。
ステップ 3	**bgp router-id** *ip_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 12.33.42.34`	BGP プロトコルのルータ ID を設定します。
ステップ 4	**address-family** {**ipv4** | **ipv6**} **unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast`	ユニキャスト用の IPv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 5	**address-family** {**vpnv4** | **vpnv6**} **unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャストの VPNv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 6	**address-family** {**ipv4** | **ipv6**} **mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN 用の IPv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 7	**neighbor** *address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 1.3.45.6`	隣接ルータを指定します。

	コマンドまたはアクション	目的
ステップ 8	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 100`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 9	**update-source interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 1`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 10	**address-family** {**ipv4** | **ipv6**} **unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast`	ユニキャスト用の IPv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 11	**address-family** {**vpnv4** | **vpnv6**} **unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャストの VPNv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 12	**address-family** {**ipv4** | **ipv6**} **mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN 用の IPv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 13	**vrf** *name* 例： `RP/0/RSP0/CPU0:router(config-bgp)# vrf vrf1`	VRF を設定します。
ステップ 14	**rd** *x.y format* 例： `RP/0/RSP0/CPU0:router(config-bgp-vrf)# rd 1:1`	ルート識別子を設定します。

	コマンドまたはアクション	目的
ステップ 15	**bgp router-id** *ip_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 12.33.42.34`	BGP プロトコルのルータ ID を設定します。
ステップ 16	**address-family** {**ipv4** | **ipv6**} **unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast`	ユニキャスト用の IPv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。
ステップ 17	**redistribute connected** 例： `RP/0/RSP0/CPU0:router(config-bgp-af)# redistribute connected`	接続ルートを経由して、別のルーティング プロトコルからの情報を再配布します。
ステップ 18	**address-family** {**ipv4** | **ipv6**} **mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN 用の IPv4 アドレス ファミリを設定し、アドレス ファミリ コマンド モードを開始します。

ステップ 19	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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。

出力 PE の MVPN P2MP の設定

出力 PE の MVPN P2MP 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. **multicast-routing**
3. **address-family** {**ipv4**|**ipv6**}
4. **mdt source** *type interface-path-id*
5. **interface all enable**
6. **vrf** *vrf-name*
7. **address-family** {**ipv4** | **ipv6**}
8. **core-tree-protocol rsvp-te group-list** *name*
9. **interface all enable**
10. **ipv4 access-list** *acl-name*
11. [*sequence-number*] **permit ipv4 host** *source_address* **host** [*destination_address*]
12. [*sequence-number*] **permit ipv4 any host** *destination_address*
13. 次のいずれかのコマンドを使用します。
    - **end**
    - **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。

	コマンドまたはアクション	目的
ステップ 2	**multicast-routing** 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。
ステップ 3	**address-family** {**ipv4**|**ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4 または IPv6 アドレス ファミリ サブモードを開始します。
ステップ 4	**mdt source** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# mdt source Loopback 1`	MDT 送信元アドレスを指定します。 （注） MDT送信元インターフェイスの名前は、BGP ピアリングに使用するものと同じ名前にする必要があります。
ステップ 5	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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。 個々のインターフェイスをイネーブルにすることもできます。
ステップ 6	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-mcast)# vrf vrf1`	VPN ルーティングおよび転送（VRF）インスタンスを設定し、VRF 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	**address-family** {**ipv4** | **ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf1)# address-family ipv4`	IPv4（または IPv6）アドレス ファミリ サブモードを開始します。
ステップ 8	**core-tree-protocol rsvp-te group-list** *name* 例： `RP/0/RSP0/CPU0:router(config-mcast-vrf1-ipv4)# core-tree-protocol rsvp-te group-list mvpn_acl`	RSVP-TE をコア ツリー プロトコルとして設定し、コア ツリー プロトコル 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 9	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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。 個々のインターフェイスをイネーブルにすることもできます。
ステップ 10	**ipv4 access-list** *acl-name* 例： `RP/0/RSP0/CPU0:router(config)# ipv4 access-list mvpn_acl`	IPv4 ACL コンフィギュレーション サブモードを開始し、IPv4 アクセス リストの名前を作成します。
ステップ 11	[*sequence-number*] **permit ipv4 host** *source_address* **host** [*destination_address*] 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# permit 1 host 232.1.1.2 any`	指定されたアクセス リスト セットの一部として送信元を認識するように、アクセス リストの条件を設定します。
ステップ 12	[*sequence-number*] **permit ipv4 any host** *destination_address* 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# 20 permit ipv4 any host 232.1.1.2`	指定されたアクセス リスト セットの一部として送信元を認識するように、アクセス リストの条件を設定します。

ステップ 13	次のいずれかのコマンドを使用します。
	• **end**
	• **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。

	コマンドまたはアクション	目的
		◦ 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。

MVPN InterAS オプションの設定

さまざまな MVPN InterAS オプションを設定するには、次の手順を実行します。

PE ルータでの MVPN InterAS オプション B または C の設定

PE ルータで MVPN InterAS オプション B または C を設定するには、次の手順を実行します。

手順の概要

1. **configure**
2. **vrf vpn1**
3. **address-family ipv4 unicast**
4. **import route-target** *2-byte AS number*
5. **export route-target** *2-byte AS number*
6. **router bgp** *2-byte AS number*
7. **bgp router-id** *ipv4 address*
8. **address-family ipv4 unicast**
9. **allocate-label all**
10. **address-family vpnv4 unicast**
11. **address-family ipv4 mvpn**
12. **neighbor** *neighbor_address*
13. **remote-as** *2-byte AS number*
14. **update-source Loopback** *0-655335*
15. **address-family ipv4 labeled-unicast**
16. **address-family vpnv4 unicast**
17. **inter-as install**
18. **address-family ipv4 mvpn**
19. **vrf** *vpn1*
20. **rd** *2-byte AS number*
21. **address-family ipv4 unicast**
22. **route-target download**
23. **address-family ipv4 mvpn**
24. **inter-as install**
25. **mpls ldp**
26. **router-id** *ip address*
27. **mldp recursive-fec**
28. **interface** *type interface-path-id*
29. **multicast-routing**
30. **address-family ipv4**
31. **mdt source** *type interface-path-id*
32. **interface all enable**
33. **vrf vpn1**
34. **address-family ipv4**
35. **bgp auto-discovery mldp inter-as**
36. **mdt partitioned mldp ipv4 mp2mp**
37. **interface all enable**
38. **router pim**
39. **vrf vrf1**

40. **address-family ipv4**
41. **rpf toplogy route-policy** *policy_name*
42. **route-policy** *policy_name*
43. **set core-tree mldp-partitioned-mp2mp**
44. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**vrf vpn1** 例： `RP/0/RSP0/CPU0:router(config)# vrf vpn1`	VRF を設定し、VRF 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	**address-family ipv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-vrf)# address-family ipv4 unicast`	ユニキャスト トポロジ用の IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ サブモードを開始します。
ステップ 4	**import route-target** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# import route-target 20:1`	インポート ルート ターゲット拡張コミュニティの 2 バイトの AS 番号を指定します。
ステップ 5	**export route-target** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# export route-target 10:1`	エクスポート ルート ターゲット拡張コミュニティの 2 バイトの AS 番号を指定します。

	コマンドまたはアクション	目的
ステップ 6	**router bgp** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config)# router bgp 100`	ルータ BGP を設定し、ルータ BGP 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	**bgp router-id** *ipv4 address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 10.10.10.1`	IPv4 アドレスを使用して BGP ルータ ID を設定します。 （注） ステップ 1 ～ 7 は、オプション B と C の両方に共通です。
ステップ 8	**address-family ipv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast`	ユニキャスト トポロジを使用して IPv4 アドレス ファミリを設定します。
ステップ 9	**allocate-label all** 例： `RP/0/RSP0/CPU0:router(config-bgp)# allocate-label all`	すべてのプレフィックスにラベルを割り当てます。 （注） ステップ 8 および 9 は、オプション C の設定の一部です。
ステップ 10	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。
ステップ 11	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。
ステップ 12	**neighbor** *neighbor_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.10.10.02`	ネイバー アドレスを使用して隣接ルータを指定および設定します。

	コマンドまたはアクション	目的
ステップ 13	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 100`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 14	**update-source Loopback** *0-655335* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0`	ループバック インターフェイスを使用して、ルーティング アップデートの送信元を指定します。 （注） ステップ 10 ～ 14 は、オプション B と C の両方に共通です。
ステップ 15	**address-family ipv4 labeled-unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 labeled-unicast`	ラベル付きユニキャスト トポロジを使用して IPv4 アドレス ファミリを設定します。 （注） ステップ 15 は、オプション C の設定でのみ実行されます。
ステップ 16	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。 （注） ステップ 16 は、オプション B と C の両方に共通です。
ステップ 17	**inter-as install** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# inter-as install`	Inter-AS オプションをインストールします。 （注） ステップ 17 は、オプション B の設定専用です。
ステップ 18	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。
ステップ 19	**vrf** *vpn1* 例： `RP/0/RSP0/CPU0:router(config-bgp)# vrf vpn1`	VRF を設定し、VRF 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 20	**rd** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-vrf)# rd 10:1`	2 バイトの AS 番号を使用してルート識別子を設定します。
ステップ 21	**address-family ipv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-vrf)# address-family ipv4 unicast`	ユニキャスト トポロジ用の IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ サブモードを開始します。
ステップ 22	**route-target download** 例： `RP/0/RSP0/CPU0:router(config-bgp-vrf-af)# route-target download`	ルート ターゲットを RIB にインストールして設定します。
ステップ 23	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp-vrf)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。 （注） ステップ 18 ～ 23 は、オプション B と C の両方に共通です。
ステップ 24	**inter-as install** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# inter-as install`	Inter-AS オプションをインストールします。 （注） ステップ 24 は、オプション C の設定専用です。
ステップ 25	**mpls ldp** 例： `RP/0/RSP0/CPU0:router(config)# mpls ldp`	MPLS ラベル配布プロトコル（LDP）を設定します。 （注） ステップ 25 ～ 44 は、オプション B と C の両方に共通です。
ステップ 26	**router-id** *ip address* 例： `RP/0/RSP0/CPU0:router(config-ldp)# router-id 10.10.10.1`	IP アドレスを使用してルータ 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 27	**mldp recursive-fec** 例： `RP/0/RSP0/CPU0:router(config-ldp)# mldp recursive-fec`	mLDP 再帰 FEC のサポートを設定します。
ステップ 28	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-ldp)# interface GigabitEthernet 0/1/0/0`	GigabitEthernet/IEEE 802.3 インターフェイスを設定します。
ステップ 29	**multicast-routing** 例： `RP/0/RSP0/CPU0:router(config)# multicast-routing`	IP マルチキャスト転送をイネーブルにし、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。
ステップ 30	**address-family ipv4** 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ サブモードを開始します。
ステップ 31	**mdt source** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# mdt source Loopback 0`	MVPN を設定し、MDT 送信元アドレスを設定するために使用されるインターフェイスを指定します。
ステップ 32	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-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。 個々のインターフェイスをイネーブルにすることもできます。
ステップ 33	**vrf vpn1** 例： `RP/0/RSP0/CPU0:router(config-mcast)# vrf vpn1`	VRF を設定し、VRF 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 34	**address-family ipv4** 例： `RP/0/RSP0/CPU0:router(config-mcast-vpn1)# address-family ipv4`	IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ サブモードを開始します。
ステップ 35	**bgp auto-discovery mldp inter-as** 例： `RP/0/RSP0/CPU0:router(config-mcast-vpn1-ipv4)# bgp auto-discovery mldp inter-as`	BGP MVPN 自動検出をイネーブルにします。
ステップ 36	**mdt partitioned mldp ipv4 mp2mp** 例： `RP/0/RSP0/CPU0:router(config-mcast-vpn1-ipv4)# mdt partitioned mldp ipv4 mp2mp`	IPv4 コアの MLDP MP2MP 信号パーティション化配信ツリーをイネーブルにします。 （注） この設定は、使用されているコア ツリーオプションによって異なります。たとえば、上のステップでは MLDP MP2MP コア ツリーをイネーブルにします。 代わりに、P2MP コア ツリーを選択した場合、この設定は MLDP P2MP コア ツリーをイネーブルにします。
ステップ 37	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-vpn1-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。 個々のインターフェイスをイネーブルにすることもできます。
ステップ 38	**router pim** 例： `RP/0/RSP0/CPU0:router(config)# router pim`	ルータ PIM を設定し、PIM コンフィギュレーション モードを開始します。
ステップ 39	**vrf vrf1** 例： `RP/0/RSP0/CPU0:router(config-pim)# vrf vrf1`	VRF を設定し、VRF コンフィギュレーション モードを開始します。
ステップ 40	**address-family ipv4** 例： `RP/0/RSP0/CPU0:router(config-pim-vrf1)# address-family ipv4`	IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ 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 41	**rpf toplogy route-policy** *policy_name* 例： `RP/0/RSP0/CPU0:router(config-pim-vrf1-ipv4)# rpf topology route-policy MSPMSI_MP2MP`	RPF トポロジを選択するようにルート ポリシーを設定します。
ステップ 42	**route-policy** *policy_name* 例： `RP/0/RSP0/CPU0:router(config)# route-policy MSPMSI_MP2MP`	RPF トポロジを選択するようにルート ポリシーを設定します。
ステップ 43	**set core-tree mldp-partitioned-mp2mp** 例： `RP/0/RSP0/CPU0:router(config-rpl)# set core-tree mldp-partitioned-mp2mp`	MLDP パーティション化 MP2MP コア マルチキャスト配信ツリーのタイプを設定します。

ステップ 44	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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。

ASBR ルータでの MVPN InterAS オプション B または C の設定

ASBR ルータで MVPN InterAS オプション B または C を設定するには、次の手順を実行します。

はじめる前に

ASBR ルータでの MVPN InterAS オプション B または C の設定を開始する前に、次の手順を実行します。

```
prefix-set IGP_leaks
  10.10.10.1/32,
  10.10.10.2/32,
  10.10.10.3/32
  end-set
!
route-policy IGP_INTER_AS_C_OUT
  if destination in IGP_leaks then
    pass
  else
    drop
  endif
  end-policy
!
```

手順の概要

1. **configure**
2. **router static**
3. **address-family ipv4 unicast** *destination prefix interface-type interface-path-id*
4. **router bgp** *2-byte AS number*
5. **bgp router-id** *ipv4 address*
6. **address-family vpnv4 unicast**
7. **retain route-target all**
8. **address-family ipv4 mvpn**
9. **retain route-target all**
10. **address-family ipv4 unicast**
11. **redistribute ospf** *router_tag*
12. **route-policy** *policy_name*
13. **allocate-label all**
14. **neighbor** *neighbor_address*
15. **remote-as** *2-byte AS number*
16. **update-source** *interface 0-655335*
17. **address-family vpnv4 unicast**
18. **address-family ipv4 labeled-unicast**
19. **route-policy** *policy_name* **in**
20. **route-policy** *policy_name* **out**
21. **neighbor** *neighbor_address*
22. **remote-as** *2-byte AS number*
23. **update-source Loopback** *0-655335*
24. **address-family vpnv4 unicast**
25. **address-family ipv4 labeled-unicast**
26. **next-hop-self**
27. **address-family ipv4 mvpn**
28. **next-hop-self**
29. **mpls ldp**
30. **router-id** *ip address*
31. **mldp recursive-fec**
32. **interface** *type interface-path-id*
33. **discovery transport-address** *ip_address*
34. **interface** *type interface-path-id*
35. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**router static** 例： `RP/0/RSP0/CPU0:router(config)# router static`	スタティック ルーティング プロセスをイネーブルにします。
ステップ 3	**address-family ipv4 unicast** *destination prefix interface-type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-static)# address-family ipv4 unicast 3.3.3.3/32 GigabitEthernet 0/1/0/1`	送信先プレフィックスを使用して、ユニキャスト トポロジの IPv4 アドレス ファミリを設定します。
ステップ 4	**router bgp** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config)# router bgp 100`	ルータ BGP を設定し、ルータ BGP 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	**bgp router-id** *ipv4 address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 10.10.10.1`	IPv4アドレスを使用してBGPルータIDを設定します。 （注） ステップ 1 ～ 5 は、オプション B と C の両方に共通です。
ステップ 6	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用してVPNv4アドレス ファミリを設定します。
ステップ 7	**retain route-target all** 例： `RP/0/RSP0/CPU0:router(config-bgp-af)# retain route-target all`	少なくとも 1 つのルート ターゲットを含む、受信したアップデートを受け入れるか、または保持します。

	コマンドまたはアクション	目的
ステップ 8	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。
ステップ 9	**retain route-target all** 例： `RP/0/RSP0/CPU0:router(config-bgp-af)# retain route-target all`	少なくとも 1 つのルート ターゲットを含む、受信したアップデートを受け入れるか、または保持します。 （注） ステップ 6 ～ 9 は、オプション B の設定専用です。
ステップ 10	**address-family ipv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast`	ユニキャスト トポロジ用の IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ サブモードを開始します。
ステップ 11	**redistribute ospf** *router_tag* 例： `RP/0/RSP0/CPU0:router(config-bgp-af)# redistribute ospf 100`	別のルーティング プロトコルからの情報を再配布します。
ステップ 12	**route-policy** *policy_name* 例： `RP/0/RSP0/CPU0:router(config)# route-policy IGP_INTER_AS_C_OUT`	RPF トポロジを選択するようにルート ポリシーを設定します。
ステップ 13	**allocate-label all** 例： `RP/0/RSP0/CPU0:router(config-bgp)# allocate-label all`	すべてのプレフィックスにラベルを割り当てます。 （注） ステップ 10 および 13 は、オプション C の設定の一部です。
ステップ 14	**neighbor** *neighbor_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.10.10.02`	ネイバー アドレスを使用して隣接ルータを指定および設定します。

	コマンドまたはアクション	目的
ステップ 15	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 100`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。 （注） ステップ 14 および 15 は、オプション B と C の両方に共通です。
ステップ 16	**update-source** *interface 0-655335* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source GigabitEthernet 0/1/0/1`	GigabitEthernet インターフェイスを使用して、ルーティング アップデートのソースを指定します。
ステップ 17	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。 （注） ステップ 16 および 17 は、オプション B の設定専用です。
ステップ 18	**address-family ipv4 labeled-unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 labeled-unicast`	ラベル付きユニキャスト トポロジを使用して IPv4 アドレス ファミリを設定します。 （注） ステップ 18 は、オプション C の設定でのみ実行されます。
ステップ 19	**route-policy** *policy_name* **in** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# route-policy pass-all in`	着信ルートにルート ポリシーを適用します。
ステップ 20	**route-policy** *policy_name* **out** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# route-policy pass-all out`	発信ルートにルート ポリシーを適用します。 （注） ステップ 19 および 20 は、オプション B と C の両方に共通です。
ステップ 21	**neighbor** *neighbor_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.10.10.02`	ネイバー アドレスを使用して隣接ルータを指定および設定します。

	コマンドまたはアクション	目的
ステップ 22	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 100`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 23	**update-source Loopback** *0-655335* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0/1/0/1`	ループバック インターフェイスを使用して、ルーティング アップデートの送信元を指定します。
ステップ 24	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。
ステップ 25	**address-family ipv4 labeled-unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 labeled-unicast`	ラベル付きユニキャスト トポロジを使用して IPv4 アドレス ファミリを設定します。 （注） ステップ 25 は、オプション C の設定でのみ実行されます。
ステップ 26	**next-hop-self** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# next-hop-self`	このネイバーのネクスト ホップ計算をディセーブルにします。 （注） ステップ 21 ～ 26 は、オプション C にのみ適用されるステップ 25 を除き、オプション B と C の両方に共通です。
ステップ 27	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。
ステップ 28	**next-hop-self** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# next-hop-self`	このネイバーのネクスト ホップ計算をディセーブルにします。 （注） ステップ 25 および 26 は、オプション B の設定にのみ適用されます。

	コマンドまたはアクション	目的
ステップ 29	**mpls ldp** 例： `RP/0/RSP0/CPU0:router(config)# mpls ldp`	MPLS ラベル配布プロトコル（LDP）を設定します。 （注） ステップ 27 ～ 33 は、オプション B と C の両方に共通です。
ステップ 30	**router-id** *ip address* 例： `RP/0/RSP0/CPU0:router(config-ldp)# router-id 10.10.10.1`	IP アドレスを使用してルータ ID を設定します。
ステップ 31	**mldp recursive-fec** 例： `RP/0/RSP0/CPU0:router(config-ldp)# mldp recursive-fec`	mLDP 再帰 FEC のサポートを設定します。
ステップ 32	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-ldp)# interface GigabitEthernet 0/1/0/0`	GigabitEthernet/IEEE 802.3 インターフェイスを設定します。
ステップ 33	**discovery transport-address** *ip_address* 例： `RP/0/RSP0/CPU0:router(config-ldp-if)# discovery transport-address 3.3.3.2`	インターフェイス LDP トランスポート アドレスを指定して、インターフェイス LDP 検出パラメータを設定します。
ステップ 34	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-ldp)# interface GigabitEthernet 0/1/0/1`	GigabitEthernet/IEEE 802.3 インターフェイスを設定します。
ステップ 35	次のいずれかのコマンドを使用します。 • **end** • **commit** 例： `RP/0/RSP0/CPU0:router(config)# 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]:`

	コマンドまたはアクション	目的
	または `RP/0/RSP0/CPU0:router(config)# commit`	◦ 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。 • 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。

MVPN InterAS オプション C の RR の設定

MVPN InterAS オプション C の RR を設定するには、この手順を実行します。

手順の概要

1. **configure**
2. **router bgp** *2-byte AS number*
3. **bgp router-id** *ipv4 address*
4. **address-family ipv4 unicast**
5. **allocate-label all**
6. **address-family vpnv4 unicast**
7. **address-family ipv4 mvpn**
8. **neighbor** *neighbor_address*
9. **remote-as** *2-byte AS number*
10. **update-source Loopback** *0-655335*
11. **address-family ipv4 labeled-unicast**
12. **route-reflector-client**
13. **address-family vpnv4 unicast**
14. **route-reflector-client**
15. **address-family ipv4 mvpn**
16. **route-reflector-client**
17. **neighbor** *neighbor_address*
18. **remote-as** *2-byte AS number*
19. **update-source Loopback** *0-655335*
20. **address-family ipv4 labeled-unicast**
21. **route-reflector-client**
22. **neighbor** *neighbor_address*
23. **remote-as** *2-byte AS number*
24. **update-source Loopback** *0-655335*
25. **address-family vpnv4 unicast**
26. **route-policy** *policy_name* **in**
27. **route-policy** *policy_name* **out**
28. **next-hop-unchanged**
29. 次のいずれかのコマンドを使用します。
    - **end**
    - **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	**router bgp** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config)# router bgp 100`	ルータ BGP を設定し、ルータ BGP 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	**bgp router-id** *ipv4 address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 10.10.10.1`	IPv4 アドレスを使用して BGP ルータ ID を設定します。
ステップ 4	**address-family ipv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast`	ユニキャスト トポロジ用の IPv4 アドレス ファミリを設定し、IPv4 アドレス ファミリ サブモードを開始します。
ステップ 5	**allocate-label all** 例： `RP/0/RSP0/CPU0:router(config-bgp-af)# allocate-label all`	すべてのプレフィックスにラベルを割り当てます。
ステップ 6	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。
ステップ 7	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。

	コマンドまたはアクション	目的
ステップ 8	**neighbor** *neighbor_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.10.10.1`	ネイバー アドレスを使用して隣接ルータを指定および設定します。
ステップ 9	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 100`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 10	**update-source Loopback** *0-655335* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0`	ループバック インターフェイスを使用して、ルーティング アップデートの送信元を指定します。
ステップ 11	**address-family ipv4 labeled-unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 labeled-unicast`	ラベル付きユニキャスト トポロジを使用して IPv4 アドレス ファミリを設定します。
ステップ 12	**route-reflector-client** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# route-reflector-client`	ルート リフレクタ クライアントとしてネイバーを設定します。
ステップ 13	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp)# address-family vpnv4 unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。
ステップ 14	**route-reflector-client** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# route-reflector-client`	ルート リフレクタ クライアントとしてネイバーを設定します。

	コマンドまたはアクション	目的
ステップ 15	**address-family ipv4 mvpn** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 mvpn`	MVPN を使用して IPv4 アドレス ファミリを設定します。
ステップ 16	**route-reflector-client** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# route-reflector-client`	ルートリフレクタ クライアントとしてネイバーを設定します。
ステップ 17	**neighbor** *neighbor_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.10.10.2`	ネイバー アドレスを使用して隣接ルータを指定および設定します。
ステップ 18	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 100`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 19	**update-source Loopback** *0-655335* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0`	ループバック インターフェイスを使用して、ルーティング アップデートの送信元を指定します。
ステップ 20	**address-family ipv4 labeled-unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 labeled-unicast`	ラベル付きユニキャスト トポロジを使用して IPv4 アドレス ファミリを設定します。
ステップ 21	**route-reflector-client** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# route-reflector-client`	ルートリフレクタ クライアントとしてネイバーを設定します。

	コマンドまたはアクション	目的
ステップ 22	**neighbor** *neighbor_address* 例： `RP/0/RSP0/CPU0:router(config-bgp)# neighbor 20.20.20.3`	ネイバー アドレスを使用して隣接ルータを指定および設定します。
ステップ 23	**remote-as** *2-byte AS number* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 200`	指定した 2 バイトの AS 番号を使用してリモート AS を設定します。
ステップ 24	**update-source Loopback** *0-655335* 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# update-source Loopback 0`	ループバック インターフェイスを使用して、ルーティング アップデートの送信元を指定します。
ステップ 25	**address-family vpnv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family ipv4 labeled-unicast`	ユニキャスト トポロジを使用して VPNv4 アドレス ファミリを設定します。
ステップ 26	**route-policy** *policy_name* **in** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# route-policy pass-all in`	着信ルートにルート ポリシーを適用します。
ステップ 27	**route-policy** *policy_name* **out** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# route-policy pass-all out`	発信ルートにルート ポリシーを適用します。
ステップ 28	**next-hop-unchanged** 例： `RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# next-hop-unchanged`	ネクスト ホップが eBGP ピアにアドバタイズされるまでは、そのまま維持されるように（上書きされないように）指示します。
ステップ 29	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

マルチトポロジ ルーティングの設定

この一連の手順では、リバース パス転送（RPF）のパス選択のために PIM で使用されるマルチトポロジ ルーティングを設定します。

- 「Configuring a Global Topology and Associating It with an Interface」（必須）

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

- 「Enabling an IS-IS Topology」（必須）

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

- 「Placing an Interface in a Topology in IS-IS」（必須）

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

- 「Configuring a Routing Policy」（必須）

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

マルチトポロジ ルーティングの設定に関する制約事項

- 現在、デフォルト VRF のみがマルチトポロジ ソリューションでサポートされます。
- プロトコル独立型マルチキャスト（PIM）と Intermediate System-to-Intermediate System（IS-IS）ルーティング プロトコルのみが現在サポートされています。
- トポロジの選択は、SM と SSM の両方で (S,G) ルート送信元のみ制限されます。 スタティックおよびIS-IS は、マルチトポロジ配置をサポートする唯一の Interior Gateway Protocol（IGP）です。

  ランデブー ポイントやブートストラップ ルータ（BSR）などの非 (S,G) ルート送信元の場合や、ルートポリシーが設定されていない場合、現在のポリシーのデフォルトは有効なままになります。 つまり、ユニキャスト デフォルトかマルチキャスト デフォルト テーブルのいずれかが、次のいずれかに基づいてすべての送信元に対して選択されます。

  - Open Shortest Path First（OSPF）
  - Intermediate System-to-Intermediate System（IS-IS）
  - マルチプロトコル ボーダー ゲートウェイ プロトコル（MBGP）

（注） ルーティング ポリシー言語（RPL）で **address-family** {**ipv4** | **ipv6**} コマンドを使用するときに **multicast** キーワードと **unicast** キーワードの両方を使用できますが、グローバルに設定できるのはマルチキャスト SAFI のトポロジだけです。

マルチトポロジ ルーティングに関する情報

マルチトポロジ ネットワークの設定には、次の作業が必要です。

- 「Configuring a Global Topology and Associating It with an Interface」（必須）

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

- 「Enabling an IS-IS Topology」（必須）

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

- 「Placing an Interface in a Topology in IS-IS」（必須）

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

- 「Configuring a Routing Policy」（必須）

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

PIM での RPF トポロジの設定

手順の概要

1. **configure**
2. **router pim address-family** {**ipv4** | **ipv6**}
3. **rpf topology route-policy** *policy-name*
4. **exit**
5. **multicast-routing address-family** {**ipv4** | **ipv6**}
6. **interface all enable**
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show pim [vrf** *vrf-name*] [**ipv4** | **ipv6**] [{**unicast** | **multicast** | **safi-all**} **topology** {*table-name* | **all**}] **rpf** [*ip-address* | **hash** | **summary** | **route-policy**]

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	**router pim address-family** {**ipv4**	**ipv6**}
		
		例：
		`RP/0/RSP0/CPU0:router(config)#`
		`RP/0/RSP0/CPU0:router(config-pim-default-ipv6)#`
			選択した IP プレフィックスの PIM アドレス ファミリ 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 3	**rpf topology route-policy** *policy-name* 例： `RP/0/RSP0/CPU0:router(config-pim-default-ipv)# rpf topology route-policy mtpolicy`	RPF トポロジ テーブルに特定のルーティング ポリシーを割り当てます。

ステップ 4	**exit**
	
	例：
	`RP/0/RSP0/CPU0:router(config-pim-default-ipv6)# exit`
	`RP/0/RSP0/CPU0:router(config)#`
		PIM アドレス ファミリ コンフィギュレーション サブモードを終了し、グローバル コンフィギュレーション モードに戻ります。

ステップ 5	**multicast-routing address-family** {**ipv4** | **ipv6**} 例： `RP/0/RSP0/CPU0:router(config)# multicast-routing address-family ipv`	マルチキャスト アドレス ファミリ 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 6	**interface all enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv)# 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 7	次のいずれかのコマンドを使用します。
	• **end**
	• **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

	コマンドまたはアクション	目的
		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 8	**show pim [vrf** *vrf-name*] [**ipv4** | **ipv6**] [{**unicast** | **multicast** | **safi-all**} **topology** {*table-name* | **all**}] **rpf** [*ip-address* | **hash** | **summary** | **route-policy**] 例： `RP/0/RSP0/CPU0:router# show pim vrf mtt rpf ipv multicast topology all rpf`	1 つ以上のテーブルの PIM RPF エントリを示します。

MVPN エクストラネットルーティングの設定

ソース VRF からレシーバ VRF へのユニ キャスト ルートをインポートするには、レシーバ VRF のインポート ルート ターゲットはソース VRF のエクスポート ルート ターゲットと一致する必要があります。 また、エクストラネット ソースレシーバ スイッチオーバーが発生する PE 上のすべての VRF は、それらの PE 上の BGP ルータ コンフィギュレーションに追加する必要があります。

MVPN エクストラネット ルーティングを設定するには、以下の必須および任意の作業をこの順序で実行します。

- 「Configuring a Routing Policy」（次の作業を実行する場合にのみ必要）

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

MVPN エクストラネットルーティングの詳細については、MVPN 上の IPv6 接続, （89 ページ）を参照してください。 2 つの使用可能な MVPN エクストラネット トポロジ ソリューションのエンドツーエンド設定例については、MVPN エクストラネット ルーティングの設定例, （248 ページ）を参照してください。

MVPN エクストラネット ルーティングの前提条件

- PIM-SM および PIM-SSM がサポートされています。 PIM モードが一致するソースおよびレシーバ VRF でマルチキャスト グループ範囲を設定する必要があります。

- 特定のマルチキャスト グループ範囲に対して現在スタティック RP の設定のみがサポートされるため、ソースおよびレシーバ MVRF に両方同じ RP を設定する必要があります。
- MVPN 上の IPv6 接続, （89 ページ） トポロジ モデルでは、データ MDT のカプセル化の範囲は、集約なしでエクストラネット ストリームを提供するのに十分大きくなければなりません。 これにより、複数の VRF に流れるエクストラネット トラフィックが、1 つのデータ MDT だけで伝送されるのを防ぎます。
- ソース VRF とソース PE ルータのみでデータ MDT の設定が必要になります。

MVPN エクストラネット ルーティングの制約事項

- PIM-DM はサポートされません。
- Cisco IOS XR Software ソフトウェアは、IPv4 コア マルチキャスト ルーティング上の IPv4 エクストラネット マルチキャスト ルーティングのみをサポートします。
- エクストラネット スイッチオーバーが発生し、ソース VRF にインターフェイスがない、「ソース PE ルータ上のレシーバ VRF」モデルの PE を除き、すべての PE を RP として設定できます。 これは、ソース VRF は先頭ホップから受信したデータ パケットをシグナリングする物理インターフェイスを持っている必要があるためです。
- Cisco IOS XR Software は、現在エクストラネット上で VRF トラフィックの 1 つのカプセル化だけをサポートします。 これは、マルチキャスト ルートの発信転送インターフェイス リストで、1 つのカプセル化インターフェイス（または MDT）のみが許可されることを意味します。 特定のストリームに、同じソース VRF に加入する複数のレシーバ VRF がある場合、最初のレシーバ VRF だけがトラフィックを受信します。他のレシーバ VRF の join は廃棄されます。

（注） この制限は、トポロジ モデル MVPN 上の IPv6 接続, （89 ページ）のみに適用されます。

VPN ルート ターゲットの設定

この手順は、トポロジごとに VPN ルート ターゲットを設定する方法を示します。

（注） レシーバ VRF がソース VRF のプレフィックスにユニキャストで到達可能となるように、ルート ターゲットを設定する必要があります。 これらの設定手順は、ソース VRF プレフィックスがレシーバ VRF にすでにインポートされている場合は省略できます。

手順の概要

1. **configure**
2. **vrf** *source-vrf*
3. **address-family** [**ipv4** | **ipv6**} **unicast**
4. **import route-target** [*xx.yy:nn* | *as-number:nn* | *ip-address:nn*]
5. **export route-target** [*xx.yy:nn* | *as-number:nn* | *ip-address:nn*]
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
7. **configure**
8. **vrf** *receiver-vrf*
9. ステップ 3 ～ 6 を繰り返します。

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	**vrf** *source-vrf*
	
	例：
	
	`RP/0/RSP0/CPU0:router(config)# vrf green`
	`RP/0/RSP0/CPU0:router(config-vrf)#`
		ソース PE ルータの VRF インスタンスを設定します。

ステップ 3	**address-family** [**ipv4**	**ipv6**} **unicast** 例： `RP/0/RSP0/CPU0:router(config-vrf)# address-family ipv4 unicast`	ユニキャスト IPv4 または IPv6 アドレス ファミリを指定し、アドレス ファミリ コンフィギュレーション サブモードを開始します。 （注） IPv4 アドレッシングのみがエクストラネットでサポートされています。

ステップ 4	**import route-target** [*xx.yy:nn*	*as-number:nn*	*ip-address:nn*]
			
			例：
			
			`RP/0/RSP0/CPU0:router(config-vrf-af)# import route-target 234:222`
			`RP/0/RSP0/CPU0:router(config-vrf-af)#`
				任意で次のいずれかで表現される選択したルート ターゲットをインポートします。
				
				• *xx.yy:nn* 形式の、ルート ターゲットの 4 バイト AS 番号。範囲は 0 ～ 65535.0 ～ 65535:0 ～ 65535 です。
				• ルート ターゲット AS 番号（*nn* 形式）。範囲は 0 ～ 65535 です。

	コマンドまたはアクション	目的
	`import route-target 100:100`	• ルート ターゲットの IP アドレス（*A.B.C.D.* 形式）。
ステップ 5	**export route-target** [*xx.yy:nn* | *as-number:nn* | *ip-address:nn*] 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# export route-target 100:100`	任意で次のいずれかで表現される選択したルート ターゲットをエクスポートします。 • *xx.yy:nn* 形式の、ルート ターゲットの 4 バイト AS 番号。範囲は 0 ～ 65535.0 ～ 65535:0 ～ 65535 です。 • ルート ターゲット AS 番号（*nn* 形式）。範囲は 0 ～ 65535 です。 • ルート ターゲットの IP アドレス（*A.B.C.D.* 形式）。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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 7	**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 8	**vrf** *receiver-vrf*
	
	例：
	`RP/0/RSP0/CPU0:router(config)# vrf red`
	`RP/0/RSP0/CPU0:router(config-vrf)#`
		レシーバ PE ルータの VRF インスタンスを設定します。

ステップ 9	ステップ 3 ～ 6 を繰り返します。	—

PIM-SM ドメインと MSDP の相互接続

別のドメインの MSDP 対応ルータとの MSDP ピアリング関係を設定するには、ローカル ルータに、MSDP ピアを設定します。

ドメインに BGP ピアを設定しないか設定できない場合、すべての Source-Active（SA）メッセージを受け入れるデフォルト MSDP ピアを定義できます。

最後に、MSDP メッシュ グループ内の複数のルータで論理 RP を設定するときに、送信元 ID を変更できます。

はじめる前に

すべての MSDP ピアのアドレスが BGP またはマルチプロトコル BGP で認識されていない場合、MSDP のデフォルト ピアリングを設定する必要があります。

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **ipv4 address** *address mask*
4. **exit**
5. **router msdp**
6. **default-peer** *ip-address* [**prefix-list** *list*]
7. **originator-id** *type interface-path-id*
8. **peer** *peer-address*
9. **connect-source** *type interface-path-id*
10. **mesh-group** *name*
11. **remote-as** *as-number*
12. 次のいずれかのコマンドを使用します。
    - **end**
    - **commit**
13. **show msdp** [**ipv4**] **globals**
14. **show msdp** [**ipv4**] **peer** [*peer-address*]
15. **show msdp** [**ipv4**] **rpf** *rpf-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	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config)# interface loopback 0`	（任意）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、インターフェイスの IPv4 アドレスを定義します。 （注） この手順は、プライマリ アドレスが TCP 接続の送信元 IP アドレスとなるインターフェイスのタイプおよび番号を指定する場合に必要です。
ステップ 3	**ipv4 address** *address mask* 例： `RP/0/RSP0/CPU0:router(config-if)# ipv4`	（任意）インターフェイスの IPv4 アドレスを定義します。 （注） この手順は、プライマリ アドレスが TCP 接続の送信元 IP アドレスとなるインターフェイスのタイプおよび番号を指定する場合にのみ必要です。**connect-source** コマンドの設定については、オプションを参照してください。

	コマンドまたはアクション	目的
	`address 10.0.1.3 255.255.255.0`	
ステップ 4	**exit** 例： `RP/0/RSP0/CPU0:router(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、ルータをグローバル コンフィギュレーション モードに戻します。
ステップ 5	**router msdp** 例： `RP/0/RSP0/CPU0:router(config)# router msdp`	MSDP プロトコル コンフィギュレーション モードを開始します。
ステップ 6	**default-peer** *ip-address* [**prefix-list** *list*] 例： `RP/0/RSP0/CPU0:router(config-msdp)# default-peer 172.23.16.0`	（任意）すべての MSDP SA メッセージの受信元となるデフォルト ピアを定義します。
ステップ 7	**originator-id** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-msdp)# originator-id GigabitEthernet0/1/1/0`	（任意）Source-Active（SA）メッセージのソースの MSDP スピーカーがインターフェイスの IP アドレスを SA メッセージ内で RP アドレスとして使用できるようにします。
ステップ 8	**peer** *peer-address* 例： `RP/0/RSP0/CPU0:router(config-msdp)# peer 172.31.1.2`	MSDP ピア コンフィギュレーション モードを開始し、MSDP ピアを設定します。 • BGP ネイバーとしてルータを設定します。 • この MSDP ピアとともに BGP ピアも使用する場合は、MSDP と BGP で同一の IP アドレスを使用する必要があります。 MSDP ピア間に BGP またはマルチプロトコル BGP パスがある場合は、MSDP ピアとともに BGP またはマルチプロトコル BGP を実行する必要はありません。
ステップ 9	**connect-source** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config-msdp-peer)# connect-source loopback 0`	（任意）MSDP接続に使用される送信元アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 10	**mesh-group** *name* 例： `RP/0/RSP0/CPU0:router(config-msdp-peer)# mesh-group internal`	（任意）MSDP ピアをメッシュ グループのメンバとして設定します。
ステップ 11	**remote-as** *as-number* 例： `RP/0/RSP0/CPU0:router(config-msdp-peer)# remote-as 250`	（任意）このピアのリモート自律システム番号を設定します。

ステップ 12	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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 13	**show msdp** [**ipv4**] **globals** 例： `RP/0/RSP0/CPU0:router# show msdp globals`	MSDP のグローバル変数を表示します。

	コマンドまたはアクション	目的
ステップ 14	**show msdp** [**ipv4**] **peer** [*peer-address*] 例： `RP/0/RSP0/CPU0:router# show msdp peer 172.31.1.2`	MSDP ピアに関する詳細情報を表示します。
ステップ 15	**show msdp** [**ipv4**] **rpf** *rpf-address* 例： `RP/0/RSP0/CPU0:router# show msdp rpf 172.16.10.13`	RPF ルックアップを表示します。

MSDP ピア ルータの送信元情報の制御

MSDP ピア ルータは、送信、転送、受信、キャッシュ、カプセル化される送信元情報を制御するようにカスタマイズできます。

Source-Active（SA）メッセージを送信する場合、送信元情報の送信先を、情報を要求している送信元に基づいて制御できます。

SA メッセージを転送する場合、次のことを行うことができます。

- すべての送信元とグループのペアのフィルタリング
- 特定の送信元とグループのペアだけが通過するように、拡張アクセス リストを指定
- ルート マップの一致条件に基づくフィルタリング

SA メッセージを受信する場合、次のことを行うことができます。

- MSDP ピアからのすべての着信 SA メッセージのフィルタリング
- 特定の送信元とグループのペアが通過するように、拡張アクセス リストを指定
- ルート マップの一致条件に基づくフィルタリング

また、Time To Live（TTL）を使用して、各送信元の最初の SA メッセージにカプセル化されるデータを制御できます。 たとえば、内部トラフィックの TTL を 8 ホップに制限したとします。 他のグループを外部に送信する場合は、これらのパケットの TTL を 8 ホップより大きく設定して送信します。

デフォルトでは、新しいメンバがグループに加入してマルチキャスト トラフィックを受信する必要が生じた場合、MSDP はピアに SA メッセージを自動的に送信します。 指定された MSDP ピアへの SA 要求を設定する必要はなくなりました。

手順の概要

1. **configure**
2. **router msdp**
3. **sa-filter** {**in** | **out**} {*ip-address* | *peer-name*} [**list** *access-list-name*] [**rp-list** *access-list-name*]
4. **cache-sa-state** [**list** *access-list-name*] [**rp-list** *access-list-name*]
5. **ttl-threshold** *ttl-value*
6. **exit**
7. **ipv4 access-list** *name* [*sequence-number*] **permit** *source* [*source-wildcard*]
8. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**router msdp** 例： `RP/0/RSP0/CPU0:router(config)# router msdp`	MSDP プロトコル コンフィギュレーション モードを開始します。
ステップ 3	**sa-filter** {**in**	**out**} {*ip-address*	*peer-name*} [**list** *access-list-name*] [**rp-list** *access-list-name*] 例： `RP/0/RSP0/CPU0:router(config-msdp)# sa-filter out router.cisco.com list 100`	指定の MSDP ピアから受信するメッセージの着信または発信フィルタ リストを設定します。 • **list** および **rp-list** キーワードの両方を指定した場合、送信 Source-Active（SA）メッセージ内の任意の送信元とグループ (S,G) のペアが通過するためには、すべての条件に当てはまる必要があります。 • **ipv4 access-list** コマンドを ステップ 7, （202 ページ） で設定する必要があります。 • すべての一致条件を満たす場合、ルート マップに **permit** が指定されていれば、ルートはフィルタを通過します。 **deny** が指定されていれば、ルートはフィルタリングされます。

	コマンドまたはアクション	目的
		• 次の例では、アクセス リスト 100 を通過する (S,G) ペアだけが SA メッセージに格納され、router.cisco.com という名前のピアに転送されるように設定します。
ステップ 4	**cache-sa-state** [**list** *access-list-name*] [**rp-list** *access-list-name*] 例： `RP/0/RSP0/CPU0:router(config-msdp)# cache-sa-state 100`	受信した Source-Active（SA）メッセージから送信元とグループのペアを作成し、アクセス リストを通じてペアを制御します。
ステップ 5	**ttl-threshold** *ttl-value* 例： `RP/0/RSP0/CPU0:router(config-msdp)# ttl-threshold 8`	（任意）SA メッセージで MSDP ピアに送信されるマルチキャスト データを制限します。 • IP ヘッダーの TTL が *ttl-value* 引数以上であるマルチキャスト パケットだけが、IP アドレスまたは名前により指定された MSDP ピアに送信されます。 • TTL によりマルチキャスト データ トラフィックを検査する場合、このコマンドを使用します。 たとえば、内部トラフィックの TTL を 8 に制限したとします。 その他のグループが外部の場所に移動できるようにするには、8 よりも大きい TTL を使用してパケットを送信します。 • 次の例では、TTL しきい値を 8 ホップに設定します。
ステップ 6	**exit** 例： `RP/0/RSP0/CPU0:router(config-msdp)# 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	**ipv4 access-list** *name* [*sequence-number*] **permit** *source* [*source-wildcard*] 例： `RP/0/RSP0/CPU0:router(config)# ipv4 access-list 100 20 permit 239.1.1.1 0.0.0.0`	SA フィルタリングによって使用される IPv4 アクセス リストを定義します。 • この例では、アクセス リスト 100 がマルチキャスト グループ 239.1.1.1 を許可します。 • ステップ 3, （201 ページ） で、SA フィルタリング用にキーワード **list** が設定される場合、**ipv4 access-list** コマンドが必要です。
ステップ 8	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

MSDP MD5 パスワード認証の設定

手順の概要

1. **configure**
2. **router msdp**
3. **peer** *peer-address*
4. **password** {**clear** | **encrypted**} *password*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
6. **show mfib** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **hardware route** {* | *source-address* | *group-address*[/*prefix-length*]} **location** *node-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	**router msdp** 例： `RP/0/RSP0/CPU0:router(config)# router msdp`	MSDP 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	**peer** *peer-address* 例： `RP/0/RSP0/CPU0:router(config-msdp)# peer 10.0.5.4`	MSDP ピアを設定します。
ステップ 4	**password** {**clear** | **encrypted**} *password* 例： `RP/0/RSP0/CPU0:router(config-msdp-peer)# password encrypted a34bi5m`	パスワードを設定します。

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **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
		• 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 6	**show mfib** [**vrf** *vrf-name*] [**ipv4** | **ipv6**] **hardware route** {***** | *source-address* | *group-address*[/*prefix-length*]} **location** *node-id* 例： `RP/0/RSP0/CPU0:router# show mfib hardware route * location 0/1/cpu0`	マルチキャスト QoS および関連パラメータで設定されたマルチキャスト ルートを表示します。

マルチキャスト専用高速再ルーティング（MoFRR）

MoFRR を使用すると、マルチキャスト ルータでマルチキャスト トラフィックの高速再ルーティングが可能になります。 MoFRR では、ノードまたはリンク障害時に（トポロジのマージ ポイントで）、ネットワークのパケット損失が最小限になります。 MoFRR は、マルチキャスト ルーティング プロトコルに対する単純な拡張により機能します。

MoFRR では、受信側からのマルチキャスト join メッセージをプライマリ パス上の送信元に向けて転送し、受信側からのセカンダリ マルチキャスト join メッセージをバックアップ パス上の送信元に向けて転送します。 データ パケットは、プライマリ パスとセカンダリ パスから受信されます。 冗長なパケットは、リバース パス転送（RPF）チェックを使用してトポロジのマージ ポイントで廃棄されます。プライマリ パスで障害が検出されると、パケットが受け入れられるインターフェイスをセカンダリ インターフェイスに変更することによりローカルで修復が実行されるため、プライマリ パスのノードまたはリンク障害の場合にコンバージェンス時間が短縮されます。

現在 MoFRR は等コスト マルチパス（ECMP）トポロジのみでサポートされます。 XML サポートは MoFRR で使用できます。

MoFRR の動作モード

- RIB ベースの MoFRR：Cisco CRS および XR12000 シリーズ ルータをサポートします。RIB のバージョンはソフトウェア レベルで設定され、ルーティング コンバージェンスに基づきます。 RIB イベントは、スイッチオーバーのトリガーとして使用されます。
- フローベースの MoFRR：Cisco ASR 9000 シリーズ アグリゲーション サービス ルータをサポートします。 フローベースの MoFRR では、プライマリおよびセカンダリ、RPF インターフェイスがフォワーディング プレーンに公開され、ハードウェア レベルでスイッチオーバーが発生します。

フローベースの MoFRR では、プライマリ ストリームのパケット カウントを監視することで、より高速なコンバージェンスが可能になります。 アクティビティが 30 ms の間検出されない場合、バックアップ ストリームへのスイッチオーバーがトリガーされ、トラフィック損失は 50 ms 以内になります。

制約事項

これらの制約事項は、MoFRR 配置で Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 ラインカードが Cisco ASR 9000 シリーズ ルータ シャーシで使用されている場合に適用されます。

1. Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 は、マルチキャスト送信元に戻るプライマリまたはバックアップ（ECMP パス）パスとして、入力インターフェイスで使用できません。
2. Cisco ASR 9000 シリーズ SPA インターフェイス プロセッサ 700 上に出力インターフェイスがあると、マルチキャスト ストリームが短い時間（入力での Trident プライマリ パスから Trident バックアップ パスへの切り替えの間）だけ重複する可能性があります。

MoFRR の設定

RIB ベースの MoFRR

手順の概要

1. **configure**
2. **router pim**
3. **mofrr rib** *acl-name*
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **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。

	コマンドまたはアクション	目的
ステップ 2	**router pim** 例： `RP/0/RSP0/CPU0:router(config)# router pim`	PIM 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	**mofrr rib** *acl-name* 例： `RP/0/RSP0/CPU0:router(pim)# mofrr rib acl1`	ACL 名を入力します。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

フローベースの MoFRR

手順の概要

1. **configure**
2. **ipv4 access-list** *acl-name*
3. *sequence number* [**permit**|**deny**] **ipv4 host** *address* [**host** *address* | **any**]
4. **exit**
5. **router pim**
6. **mofrr** *acl-name*
7. 次のいずれかのコマンドを使用します。
   - **end**
   - **commit**
8. **show mfib hardware route summary location**

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	**ipv4 access-list** *acl-name* 例： `RP/0/RSP0/CPU0:router (config)# ipv4 access-list flow_mofrr`	IPv4 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、指定したアクセス リストを設定します。
ステップ 3	*sequence number* [**permit**|**deny**] **ipv4 host** *address* [**host** *address*	**any**] 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl) #10 permit ipv4 host 20.0.0.2 any`	作成した IPv4 アクセス リストで許可または拒否する 1 つ以上の条件を指定します。
ステップ 4	**exit** 例： `RP/0/RSP0/CPU0:router(config-ipv4-acl)# exit`	MoFRR ACL の設定を保存し、IPv4 ACL コンフィギュレーション モードを終了します。 ここでは 2 回 **exit** を実行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 5	**router pim** 例： `RP/0/RSP0/CPU0:router(config)# router pim`	PIM 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	**mofrr** *acl-name* 例： `RP/0/RSP0/CPU0:router(pim)# mofrr flow_mofrr`	ハードウェアのスイッチオーバー トリガーを使用して指定したアクセス リスト ソース グループの MoFRR をイネーブルにします。 これは、IPv4 のみでサポートされます。

ステップ 7	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		
		• 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 8	**show mfib hardware route summary location** 例： `RP/0/RSP0/CPU0:router#` **show mfib hardware route 4**	イネーブルになっている MoFRR ルートの数を表示します。

ポイントツーマルチポイントトラフィックエンジニアリング ラベル スイッチド マルチキャスト

IPマルチキャストは、IPTVのブロードキャストおよびコンテンツ提供サービスに従来使用されていました。 MPLS-TE（トラフィック エンジニアリング）は、次のような利点から、IPマルチキャスト技術を速い速度で置き換えています。

- リンクまたはノードに障害が発生した場合の高速再ルーティングと復元
- 帯域幅保証
- 明示的なパス設定とオフライン計算

MPLS は、ポイントツーポイント パスをサポートします。 ただし、マルチキャスト サービスに MPLS を使用するには、ポイントツーマルチポイント パスを処理するように MPLS を拡張する必要があります。 ポイントツーマルチポイント（P2MP）のラベル スイッチド パス（LSP）をシグナリングするための信頼できるソリューションはポイントツーマルチポイント TE LSP です。 このソリューションは、P2MP TE LSP を確立するためのシグナリング プロトコルとして、リソース予約プロトコル トラフィック エンジニアリング（RSVP-TE）拡張を使用します。

ポイントツーマルチポイント LSP（P2MP）

P2MP LSP は単方向です。 ネイティブ IP マルチキャストの場合、マルチキャスト転送は常にアクセプタンス チェックを実行する必要があります。 このチェックでは、すべてのマルチキャスト パケットに RPF チェックを実行し、パケットが送信元の方向に正しいインターフェイスに着信したことを確認します。 ただし、MPLS 転送を使用したアクセプタンス チェックは、ユニキャストまたはアップストリーム ラベルの場合は異なることがあります。

マルチキャスト シグナリング プロトコルによっては、ラベル付きパケットは、P および PE ルータで、マルチキャスト ルーティングに従って物理インターフェイスにマルチキャスト パケットを転送するために、追加の L3 検索が必要な場合があります。 この場合、受信したマルチキャスト パケットの着信インターフェイスとしての着信 P2MP LSP も、L3 検索中にマルチキャスト フォワーディング プレーンで使用できる必要があります。 RSVP-TE および P2MP LSP 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。

P2MP のマルチキャスト ルーティング プロトコルのサポート

すべてのマルチキャスト ルーティング プロトコルは、P2MP TE LSP をサポートします。 入力ノードで、マルチキャスト プロトコルは静的加入の設定を使用し、マルチキャスト トラフィックと P2MP TE LSP 間のマッピングを作成する必要があります。 出力ノードでは、マルチキャスト プロ

トコルはMPLS コアから受信したマルチキャスト パケットに対して特別な RPF チェックを行い、カスタマー相対インターフェイスに転送する必要があります。 RPF チェックは static-rpf の設定に基づいて行われます。 P2MP TE LSP を介して転送されるこれらのマルチキャスト グループは、PIM-SSM の場合は static-rpf の設定で指定できます。

トンネルインターフェイス上のマルチキャスト転送のイネーブル化（入力ノード）

この設定は、指定したインターフェイス上のマルチキャスト パケットの転送を許可するために使用されます。

手順の概要

1. **configure**
2. **multicast-routing**
3. **address-family** {**ipv4**|**ipv6**}
4. **interface tunnel-mte** *range*
5. **enable** | **disable**
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**multicast-routing** 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。
ステップ 3	**address-family** {**ipv4**|**ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4 または IPv6 アドレス ファミリ 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 4	**interface tunnel-mte** *range* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# interface tunnel-mte 100`	範囲を指定します。 範囲は 0 ～ 65535 です。
ステップ 5	**enable** | **disable** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# enable`	enable が設定されている場合、MFIB がインターフェイス上でマルチキャスト パケットを転送します。disable が設定されている場合、MFIB はインターフェイス上でのマルチキャストパケットの転送を停止します。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

出力ノードとバド ノードでの P2MP の設定

静的リバース パス転送（RPF）の設定

手順の概要

1. **configure**
2. **multicast-routing**
3. **address-family** {**ipv4** | **ipv6**}
4. **static-rpf** *address range prefix*
5. **mpls address**
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**multicast-routing** 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。
ステップ 3	**address-family** {**ipv4**	**ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4（または IPv6）アドレス ファミリ サブモードを開始します。
ステップ 4	**static-rpf** *address range prefix* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# static-rpf 10.1.1.1 32`	送信元とプレフィックス長を入力します。

	コマンドまたはアクション	目的
ステップ 5	**mpls address** 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# mpls 10.2.2.2`	MPLS P2MP トンネルのソース PE アドレスを入力します。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

コア ツリー プロトコルの設定

手順の概要

1. **configure**
2. **multicast-routing**
3. **address-family** {**ipv4** | **ipv6**}
4. **core-tree-protocol rsvp-te group-list** *name*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**multicast-routing** 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。
ステップ 3	**address-family** {**ipv4**	**ipv6**} 例： `RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv4`	IPv4（または IPv6）アドレス ファミリ サブモードを開始します。
ステップ 4	**core-tree-protocol rsvp-te group-list** *name* 例： `RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# core-tree-protocol rsvp-te group-list acl1`	コアツリープロトコル コンフィギュレーション モードを開始します。
ステップ 5	次のいずれかのコマンドを使用します。	設定変更を保存します。

	コマンドまたはアクション	目的

	• **end**
	• **commit**
	
	例：
	`RP/0/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		• **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。
		• 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。

IGMP VRF オーバーライドの設定

この手順は、次の作業で構成されます。

VRF 定義の指定

手順の概要

1. **configure**
2. **vrf** *vrf-name*
3. **address-family ipv4 unicast**
4. **import route-target 1:1**
5. **export route-target 1:1**
6. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config)# vrf name1`	VRF 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	**address-family ipv4 unicast** 例： `RP/0/RSP0/CPU0:router(config-vrf)# address-family ipv4 unicast`	IPv4 の AFI 設定。 これはユニキャスト トポロジのみでサポートされます。
ステップ 4	**import route-target 1:1** 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# import route-target 1:1`	VRF のイン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 5	**export route-target 1:1** 例： `RP/0/RSP0/CPU0:router(config-vrf-af)# export route-target 1:1`	VRF のエクスポートをイネーブルにします。

ステップ 6	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

デフォルトとデフォルト以外の VRF のマルチキャスト ルーティングのイネーブル化

ここでは、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。 VRF オーバーライド機能では、マルチキャスト ルーティングは、デフォルトとデフォルト以外の VRF の両方でイネーブルにする必要があります。

手順の概要

1. **configure**
2. **multicast-routing vrf** [*vrf-name* | *default]*
3. **interface** {*type interface-path-id* | **all**} **enable**
4. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**multicast-routing vrf** [*vrf-name*	*default]* 例： `RP/0/RSP0/CPU0:router(config)# multicast-routing vrf green`	指定した VRF のマルチキャスト コンフィギュレーション モードが開始されます。 マルチキャスト ルーティングのデフォルトのコンフィギュレーション モードはデフォルト VRF であることに注意してください（デフォルト以外の VRF 名が指定されていない場合）。
ステップ 3	**interface** {*type interface-path-id*	**all**} **enable** 例： `RP/0/RSP0/CPU0:router(config-mcast-green)# interface all enable`	新規および既存の 1 つまたは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	次のいずれかのコマンドを使用します。
	• **end**
	• **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。

	コマンドまたはアクション	目的
		◦ 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。

デフォルト以外の VRF インスタンスのインターフェイス設定

手順の概要

1. **configure**
2. **interface** *type interface-path-id*
3. **vrf** *vrf-name*
4. **ipv4 address** *address mask*
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **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。

	コマンドまたはアクション	目的
ステップ 2	**interface** *type interface-path-id* 例： `RP/0/RSP0/CPU0:router(config)# interface tengige 0/1/0/0`	PIM アドレス ファミリ IPv4 サブモードを開始します。
ステップ 3	**vrf** *vrf-name* 例： `RP/0/RSP0/CPU0:router(config-if)# vrf name1`	インターフェイスの VRF を設定します。
ステップ 4	**ipv4 address** *address mask* 例： `RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0`	インターフェイスの IPv4 アドレスを設定します。

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

ルート ポリシーの設定

手順の概要

1. **configure**
2. **route-policy** *policy-name*
3. **set rpf-topology vrf default**
4. **end-policy**
5. 次のいずれかのコマンドを使用します。
   - **end**
   - **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	**route-policy** *policy-name* 例： `RP/0/RSP0/CPU0:router(config)# route-policy policy1`	ルート ポリシーを定義します。
ステップ 3	**set rpf-topology vrf default** 例： `RP/0/RSP0/CPU0:router(config-rpl)# set rpf-topology vrf default`	デフォルト VRF の PIM RPF トポロジ属性を設定します。
ステップ 4	**end-policy** 例： `RP/0/RSP0/CPU0:router(config-rpl)# end-policy`	ルート ポリシー定義設定を終了します。

	コマンドまたはアクション	目的

ステップ 5	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

IGMP レポートを受信する VRF に対する PIM 設定へのルート ポリシーの関連付け

手順の概要

1. **configure**
2. **router pim vrf** *vrf-name* **address-family ipv4**
3. **rpf-topology route-policy** *policy-name*
4. 次のいずれかのコマンドを使用します。
   • **end**
   • **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	**router pim vrf** *vrf-name* **address-family ipv4**	PIM アドレス ファミリ IPv4 サブモードを開始します。
ステップ 3	**rpf-topology route-policy** *policy-name* 例： `RP/0/RSP0/CPU0:router(config-rpl)# rpf-topology extranet-igmp-reports`	以前に定義されたルート ポリシーを、IGMP レポートを受信するデフォルト以外の VRF に関連付けます。

ステップ 4	次のいずれかのコマンドを使用します。
	• **end**
	• **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。
		• 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。

ソフトウェアでマルチキャストルーティングを実装するための設定例

ここでは、次の設定例について説明します。

ルートごとのレートの計算例

次に、特定の送信元とグループ アドレス ロケーションの、ルートごとのレートに基づくハードウェア カウンタからの出力例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# multicast-routing vrf vpn12 address-family ipv4
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# rate-per-route
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# interface all enable
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# accounting per-prefix
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# commit
RP/0/RSP0/CPU0:router(config-mcast-default-ipv4)# exit
RP/0/RSP0/CPU0:router(config-mcast)# exit
RP/0/RSP0/CPU0:router(config)# exit
RP/0/RSP0/CPU0:router# show mfib route rate

IP Multicast Forwarding Rates Source Address, Group Address HW Forwarding Rates: bps In/pps
 In/bps Out/pps Out

(*,224.0.0.0/24)
bps_in /pps_in /bps_out /pps_out
N/A / N/A / N/A / N/A

(*,224.0.1.39)
bps_in /pps_in /bps_out /pps_out
N/A / N/A / N/A / N/A

(*,224.0.1.40)
bps_in /pps_in /bps_out /pps_out
N/A / N/A / N/A / N/A

(*,232.0.0.0/8)
bps_in /pps_in /bps_out /pps_out
N/A / N/A / N/A / N/A
(10.0.70.2,225.0.0.0)
bps_in /pps_in /bps_out /pps_out
22649 / 50 / 22951 / 50

(10.0.70.2,225.0.0.1)
bps_in /pps_in /bps_out /pps_out
22649 / 50 / 22951 / 50

(10.0.70.2,225.0.0.2)
bps_in /pps_in /bps_out /pps_out
22649 / 50 / 22951 / 50

(10.0.70.2,225.0.0.3)
bps_in /pps_in /bps_out /pps_out
22649 / 50 / 22951 / 50

(10.0.70.2,225.0.0.4)
bps_in /pps_in /bps_out /pps_out
22649 / 50 / 22951 / 50
```

```
(10.0.70.2,225.0.0.5)
bps_in /pps_in /bps_out /pps_out
22649 / 50 / 22951 / 50

(10.0.70.2,225.0.0.6)
bps_in /pps_in /bps_out /pps_out
```

Auto-RP メッセージのソフトウェアでの転送の防止例

次に、Auto-RP メッセージが GigabitEthernet インターフェイス 0/3/0/0 から送信されるのを防止する例を示します。 この例はまた、GigabitEthernet インターフェイス 0/3/0/0 上のトラフィックを含めるために、アクセス リスト 111 が Auto-RP 候補によって使用され、アクセス リスト 222 が **boundary** コマンドによって使用されることも示しています。

```
ipv4 access-list 111
 10 permit 224.1.0.0 0.0.255.255 any
 20 permit 224.2.0.0 0.0.255.255 any
!
!Access list 111 is used by the Auto-RP candidate.
!
ipv4 access-list 222
 10 deny any host 224.0.1.39
 20 deny any host 224.0.1.40
!
!Access list 222 is used by the boundary command to contain traffic (on
GigabitEthernet0/3/0/0) that is sent to groups 224.0.1.39 and 224.0.1.40.
!
router pim
 auto-rp mapping-agent loopback 2 scope 32 interval 30
 auto-rp candidate-rp loopback 2 scope 15 group-list 111 interval 30
multicast-routing
 interface GigabitEthernet0/3/0/0
 boundary 222
!
```

ソフトウェア上の MSDP での継承例

次の MSDP コマンドは、ルータ MSDP コンフィギュレーション モードで設定すると、すべての MSDP ピアによって継承できます。 さらに、継承機能を無効にするには、コマンドを、特定のピアのピア コンフィギュレーション モードで設定できます。

- **connect-source**
- **sa-filter**
- **ttl-threshold**

コマンドがルータ msdp モードとピア コンフィギュレーション モードの両方で設定されている場合、ピアの設定が優先されます。

次の例では、ルータ A の MSDP がアドレス範囲 226/8（IP アドレス 172.16.0.2 を除く）のすべてのピア グループの Source-Active（SA）アナウンスをフィルタし、送信元 RP 172.16.0.3 から 172.16.0.2 に送信された SA をフィルタします。

MSDP ピア（172.16.0.1、172.16.0.2、および 172.17.0.1）は、ピアリングを設定するために、ルータ A のループバック 0 アドレスを使用します。 ただし、ピア 192.168.12.2 は、ルータ A とピアリングするために、GigabitEthernet インターフェイスで設定された IPv4 アドレスを使用します。

ルータ A

```
!
ipv4 access-list 111
 10 deny ip host 172.16.0.3 any
 20 permit any any
!

ipv4 access-list 112
 10 deny any 226.0.0.0 0.255.255.255
 30 permit any any
!
router msdp
 connect-source loopback 0
 sa-filter in rp-list 111
 sa-filter out rp-list 111
 peer 172.16.0.1
!
peer 172.16.0.2
 sa-filter out list 112
!
peer 172.17.0.1
!
peer 192.168.12.2
 connect-source GigabitEthernet0/2/0/0
!
```

IPv4 マルチキャスト VPN の設定例

Cisco ASR 9000 シリーズ ルータ は、IPv4 アドレッシングのみをサポートしています。

このエンドツーエンド設定例は、カスタマー エッジ（CE）ルータとプロバイダー エッジ（PE）ルータの間でトラフィックをブロードキャストするために 2 つの異なるルーティング プロトコル（OSPF と BGP）を使用して、マルチキャスト VPN トポロジ（図 11：CE4PE1PE2 CE3MVPN 構成のトポロジ, （227 ページ））を確立する方法を示します。

図 *11*：*CE4PE1PE2 CE3MVPN* 構成のトポロジ

CE4---------- PE1 ---------- PE2 ---------- CE3

詳しい設定情報については、このモジュールのマルチキャスト VPN の設定, （136 ページ）と、『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』の関連設定情報を参照してください。

OSPF を使用して CE と PE 間のルートをアドバタイズするように MVPN を設定する例

PE1：

```
!
vrf vpn1
```

```
 address-family ipv4 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 2.2.2.2 255.255.255.255
!
interface GigabitEthernet0/5/0/0
 vrf vpn1
 ipv4 address 101.1.1.1 255.255.255.0
!
interface TenGigE0/6/0/0
 ipv4 address 12.1.1.1 255.255.255.0
!
mpls ldp
 router-id 1.1.1.1
 interface TenGigE0/6/0/0
 !
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  rate-per-route
  interface all enable
  accounting per-prefix
 !
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
!
router bgp 100
 bgp router-id 1.1.1.1
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 9.9.9.9
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   redistribute ospf 1
  !
 !
!
router ospf 1
 vrf vpn1
  router-id 2.2.2.2
  redistribute bgp 100
  area 0
```

```
   interface Loopback1
   !
   interface GigabitEthernet0/5/0/0
   !
  !
 !
!
router ospf 100
 router-id 1.1.1.1
 area 0
  interface Loopback0
  !
  interface TenGigE0/6/0/0
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 2.2.2.2
 log neighbor changes
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

PE2：

```
!
vrf vpn1
 address-family ipv4 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 9.9.9.9 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 10.10.10.10 255.255.255.255
!
interface GigabitEthernet0/2/2/7
 vrf vpn1
 ipv4 address 122.1.1.1 255.255.255.0
 negotiation auto
!
interface TenGigE0/3/0/0
 ipv4 address 12.1.1.2 255.255.255.0
!
mpls ldp
 router-id 9.9.9.9
 interface TenGigE0/3/0/0
 !
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  rate-per-route
  interface all enable
  accounting per-prefix
 !
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
```

```
  accounting per-prefix
 !
!
router bgp 100
 bgp router-id 9.9.9.9
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 1.1.1.1
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   redistribute ospf 1
  !
 !
!
router ospf 1
 vrf vpn1
  router-id 10.10.10.10
  redistribute bgp 100
  area 0
   interface Loopback1
   !
   interface GigabitEthernet0/2/2/7
   !
  !
 !
!
router ospf 100
 router-id 9.9.9.9
 area 0
  interface Loopback0
  !
  interface TenGigE0/3/0/0
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 2.2.2.2
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

CE4：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのコンフィギュレーション マニュアルを参照してください。

```
!
interface Loopback0
 ipv4 address 101.101.101.101 255.255.255.255
!
interface GigabitEthernet0/0/0/0
 ipv4 address 101.1.1.2 255.255.255.0
!
interface GigabitEthernet0/0/0/3
```

```
 ipv4 address 11.1.1.1 255.255.255.0
!
multicast-routing
 address-family ipv4
  interface all enable
 !
!
router ospf 1
 router-id 101.101.101.101
 area 0
  interface Loopback0
  !
  interface GigabitEthernet0/0/0/0
  !
  interface GigabitEthernet0/0/0/3
  !
 !
!
router pim vrf default address-family ipv4
 rp-address 2.2.2.2
 interface Loopback0
 !
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/3
 !
!
end
```

CE3：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのコンフィギュレーション マニュアルを参照してください。

```
interface Loopback0
 ipv4 address 122.122.122.122 255.255.255.255
!

interface GigabitEthernet0/1/3/0
 ipv4 address 22.1.1.1 255.255.255.0
!

interface GigabitEthernet0/2/3/0
 ipv4 address 122.1.1.2 255.255.255.0

multicast-routing
 address-family ipv4
  interface all enable
!
router ospf 1
 router-id 122.122.122.122
 area 0
  interface Loopback0
  !
  interface GigabitEthernet0/1/3/0
  !
  interface GigabitEthernet0/2/3/0
  !
 !
!
router pim vrf default address-family ipv4
 rp-address 2.2.2.2
 interface Loopback0
 !
 interface GigabitEthernet0/1/3/0
 !
 interface GigabitEthernet0/2/3/0
 !
!
```

```
end
```

BGP を使用して CE と PE 間のルートをアドバタイズするように MVPN を設定する例

PE1：

```
vrf vpn1
 address-family ipv4 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 2.2.2.2 255.255.255.255
!
interface GigabitEthernet0/5/0/0
 vrf vpn1
 ipv4 address 101.1.1.1 255.255.255.0
!
interface TenGigE0/6/0/0
 ipv4 address 12.1.1.1 255.255.255.0
!
mpls ldp
 router-id 1.1.1.1
 interface TenGigE0/6/0/0
 !
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  rate-per-route
  interface all enable
  accounting per-prefix
 !
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 100
 bgp router-id 1.1.1.1
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 9.9.9.9
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
```

```
  !
  address-family vpnv4 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   redistribute connected
  !
  neighbor 101.1.1.2
   remote-as 400
   address-family ipv4 unicast
    route-policy pass-all in
    route-policy pass-all out
   !
  !
 !
!
router ospf 100
 router-id 1.1.1.1
 area 0
  interface Loopback0
  !
  interface TenGigE0/6/0/0
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 2.2.2.2
 log neighbor changes
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

PE2：

```
!
vrf vpn1
 address-family ipv4 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 9.9.9.9 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 10.10.10.10 255.255.255.255
!
interface GigabitEthernet0/2/2/7
 vrf vpn1
 ipv4 address 122.1.1.1 255.255.255.0
 negotiation auto
!
interface TenGigE0/3/0/0
 ipv4 address 12.1.1.2 255.255.255.0
!
mpls ldp
 router-id 9.9.9.9
 interface TenGigE0/3/0/0
 !
```

```
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  rate-per-route
  interface all enable
  accounting per-prefix
 !
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 100
 bgp router-id 9.9.9.9
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 1.1.1.1
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   redistribute connected
  !
  neighbor 122.1.1.2
   remote-as 500
   address-family ipv4 unicast
    route-policy pass-all in
    route-policy pass-all out
   !
  !
 !
!
router ospf 100
 router-id 9.9.9.9
 area 0
  interface Loopback0
  !
  interface TenGigE0/3/0/0
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 2.2.2.2
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

CE4：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのコンフィギュレーション マニュアルを参照してください。

```
interface Loopback0
 ipv4 address 101.101.101.101 255.255.255.255
!
interface GigabitEthernet0/0/0/0
 ipv4 address 101.1.1.2 255.255.255.0
!
interface GigabitEthernet0/0/0/3
 ipv4 address 11.1.1.1 255.255.255.0
!
multicast-routing
 address-family ipv4
  interface all enable
 !
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 400
 bgp router-id 101.101.101.101
 address-family ipv4 unicast
  redistribute connected
 !
 neighbor 101.1.1.1
  remote-as 100
  address-family ipv4 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
 !
!
router pim vrf default address-family ipv4
 rp-address 2.2.2.2
 interface Loopback0
 !
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/3
 !
!
end
```

CE3：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのコンフィギュレーション マニュアルを参照してください。

```
interface Loopback0
 ipv4 address 122.122.122.122 255.255.255.255
!

interface GigabitEthernet0/1/3/0
 ipv4 address 22.1.1.1 255.255.255.0
!

interface GigabitEthernet0/2/3/0
 ipv4 address 122.1.1.2 255.255.255.0

multicast-routing
 address-family ipv4
  interface all enable
```

```
!
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 500
 bgp router-id 122.122.122.122
 address-family ipv4 unicast
  redistribute connected
 !
 neighbor 122.1.1.1
  remote-as 100
  address-family ipv4 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
 !
!
!
router pim vrf default address-family ipv4
 rp-address 2.2.2.2
 interface Loopback0
 !
 interface GigabitEthernet0/1/3/0
 !
 interface GigabitEthernet0/2/3/0
 !
!
end
```

IPv6 マルチキャスト VPN の設定例

Cisco XR 12000 シリーズ ルータでは、IPv4 アドレス指定のプロトコルが常に使用される必要がある MVPN のコアの場合を除き、IPv4 と IPv6 両方 MVPN がサポートされます。

図 12：CE1PE1PE2 CE2MVPN 構成のトポロジの例, （236 ページ）のエンドツーエンドの設定例は、カスタマー エッジ（CE）ルータとプロバイダー エッジ（PE）ルータ間のブロードキャスト トラフィックに 2 種類のルーティング プロトコル（EIGRP または BGP）を使用して、IPv6 マルチキャスト VPN トポロジを確立する方法を示しています。

図 *12*：*CE1PE1PE2 CE2MVPN* 構成のトポロジの例

CE1---------- PE1 ---------- PE2 ---------- CE2

MVPN の詳細については、このマニュアルのマルチキャスト VPN の設定, （136 ページ）、および『Cisco IOS XR Routing Configuration Guide』の関連設定情報を参照してください。 IPv4 アドレスだけを使用する MVPN の設定例については、IPv4 マルチキャスト VPN の設定例, （227 ページ）を参照してください。

IPv6 マルチキャスト VPN を、プロトコルとして EIGRP を持つ CE から PE 間のルートをアドバタイズするように設定する例

CE1：

Cisco IOS XR Software を使用した CE ルータの設定の詳細については、適切な Cisco IOS ソフトウェア マニュアルを参照してください。

```
interface Loopback0
 ipv4 address 101.101.101.101 255.255.255.255
!
interface GigabitEthernet0/5/0/0
 ipv6 address 2013::90:1:1:2/126
!
interface GigabitEthernet0/5/0/1
 ipv6 address 2013::102:1:1:2/96
!
multicast-routing
 address-family ipv6
  interface all enable
 !
!
route-policy pass-all
  pass
end-policy
!
router eigrp 1
 address-family ipv6
  router-id 101.101.101.101
  default-metric 1000 100 250 100 1500
  redistribute connected
  interface GigabitEthernet0/5/0/1
  !
 !
!
router pim vrf default address-family ipv6
 rp-address ::192:168:10:1
!
end
```

PE1：

```
!
vrf vpn1
 address-family ipv6 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 192.168.10.1 255.255.255.255
 ipv6 address ::192:168:10:1/128
!
interface GigabitEthernet0/4/0/1
 vrf vpn1
```

```
 ipv6 address 2013::102:1:1:1/96
!
interface FastEthernet0/5/1/0
 ipv4 address 12.1.1.1 255.255.255.0
!
route-policy pass-all
  pass
end-policy
!
router ospf 100
 router-id 1.1.1.1
 area 0
  interface Loopback0
  !
  interface FastEthernet0/5/1/0
  !
 !
!
router bgp 100
 bgp router-id 1.1.1.1
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv6 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 9.9.9.9
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family ipv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   maximum-paths ebgp 3
   redistribute connected
  !
  address-family ipv6 unicast
   maximum-paths ebgp 3
   redistribute eigrp 1
  !
 !
!
mpls ldp
 router-id 1.1.1.1
 interface FastEthernet0/5/1/0
 !
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  interface all enable
 !
 vrf vpn1 address-family ipv6
  mdt default ipv4 232.1.1.1
  interface all enable
 !
```

```
 address-family ipv4
  mdt source Loopback0
  interface all enable
 !
 address-family ipv6
  interface all enable
 !
!
router eigrp 1
 vrf vpn1
  address-family ipv6
   router-id 1.1.1.1
   default-metric 1000 100 250 100 1000
   autonomous-system 1
   redistribute bgp 100
   interface Loopback1
   !
   interface GigabitEthernet0/4/0/1
    site-of-origin 1:1
   !
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 192.168.10.1
!
router pim vrf vpn1 address-family ipv6
 rp-address ::192:168:10:1
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

PE2：

```
!
vrf vpn1
 address-family ipv6 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 9.9.9.9 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 10.10.10.10 255.255.255.255
!
interface FastEthernet0/4/1/0
 ipv4 address 12.1.1.2 255.255.255.0
!
mpls ldp
 router-id 9.9.9.9
 interface FastEthernet0/4/1/0
 !
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  interface all enable
 !
 vrf vpn1 address-family ipv6
  mdt default ipv4 232.1.1.1
```

```
  interface all enable
 !
 address-family ipv4
  multipath
  mdt source Loopback0
  interface all enable
 !
 address-family ipv6
  interface all enable
 !
!
route-policy pass-all
  pass
end-policy
!
router eigrp 2
 vrf vpn1
  address-family ipv6
   router-id 9.9.9.9
   default-metric 1000 100 250 100 1000
   autonomous-system 2
   redistribute bgp 100
   interface GigabitEthernet0/4/0/1
    site-of-origin 2:2
   !
  !
 !
!
router bgp 100
 bgp router-id 9.9.9.9
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv6 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 1.1.1.1
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family ipv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   maximum-paths ebgp 3
   redistribute connected
  !
  address-family ipv6 unicast
   maximum-paths ebgp 3
   redistribute eigrp 2
  !
 !
!
router ospf 100
 router-id 9.9.9.9
 area 0
  interface Loopback0
  !
```

```
  interface FastEthernet0/4/1/0
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 192.168.10.1
!
router pim vrf vpn1 address-family ipv6
 rp-address ::192:168:10:1
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

CE2：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのマニュアルを参照してください。

```
!
interface Loopback0
 ipv4 address 122.122.122.122 255.255.255.255
!
interface GigabitEthernet0/5/0/0
 ipv6 address 2013::80:1:1:2/126
!
interface GigabitEthernet0/5/0/1
 ipv6 address 2013::122:1:1:2/96
!
multicast-routing
 address-family ipv6
  interface all enable
 !
!
route-policy pass-all
  pass
end-policy
!
router eigrp 2
 address-family ipv6
  router-id 122.122.122.122
  default-metric 1000 100 250 100 1000
  redistribute connected
  interface GigabitEthernet0/5/0/1
  !
 !
!
router pim vrf default address-family ipv6
 dr-priority 2
 rp-address ::192:168:10:1
!
end
```

IPv6 マルチキャスト VPN を、プロトコルとして BGP を持つ CE から PE 間のルートをアドバタイズするように設定する例

CE1：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのマニュアルを参照してください。

```
!
interface Loopback0
 ipv4 address 101.101.101.101 255.255.255.255
!
interface GigabitEthernet0/5/0/0
 ipv6 address 2013::90:1:1:2/126
!
interface GigabitEthernet0/5/0/1
 ipv6 address 2013::102:1:1:2/96
!
multicast-routing
 address-family ipv4
  interface all enable
 !
 address-family ipv6
  interface all enable
 !
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 1
 bgp router-id 101.101.101.101
 address-family ipv6 unicast
  redistribute connected
 !
 neighbor 2013::102:1:1:1
  remote-as 100
  address-family ipv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
 !
!
router pim vrf default address-family ipv6
 rp-address ::192:168:10:1
!
end
```

PE1：

```
!
vrf vpn1
 address-family ipv6 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
```

```
!
interface Loopback1
 vrf vpn1
 ipv4 address 192.168.10.1 255.255.255.255
 ipv6 address ::192:168:10:1/128
!
interface GigabitEthernet0/4/0/1
 vrf vpn1
 ipv6 address 2013::102:1:1:1/96
!
interface FastEthernet0/5/1/0
 ipv4 address 12.1.1.1 255.255.255.0
!
route-policy pass-all
  pass
end-policy
!
router static
 address-family ipv4 unicast
  223.0.0.0/8 5.9.0.1
 !
!
router ospf 100
 router-id 1.1.1.1
 area 0
  interface Loopback0
  !
  interface FastEthernet0/5/1/0
  !
 !
!
router bgp 100
 bgp router-id 1.1.1.1
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv6 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 9.9.9.9
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family ipv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   maximum-paths ebgp 3
   redistribute connected
  !
  address-family ipv6 unicast
   maximum-paths ebgp 3
   redistribute connected
  !
  neighbor 2013::102:1:1:2
   remote-as 1
   address-family ipv6 unicast
    route-policy pass-all in
```

```
    route-policy pass-all out
   !
  !
 !
!
mpls ldp
 router-id 1.1.1.1
 interface FastEthernet0/5/1/0
 !
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  interface all enable
 !
 vrf vpn1 address-family ipv6
  mdt default ipv4 232.1.1.1
  interface all enable
 !
 address-family ipv4
  multipath
  mdt source Loopback0
  interface all enable
 !
 address-family ipv6
  interface all enable
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 192.168.10.1
!
router pim vrf vpn1 address-family ipv6
 rp-address ::192:168:10:1
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

PE2：

```
!
vrf vpn1
 address-family ipv6 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
!
interface Loopback0
 ipv4 address 9.9.9.9 255.255.255.255
!
interface Loopback1
 vrf vpn1
 ipv4 address 10.10.10.10 255.255.255.255
!
interface GigabitEthernet0/4/0/1
 vrf vpn1
 ipv6 address 2013::122:1:1:1/96
!
interface FastEthernet0/4/1/0
 ipv4 address 12.1.1.2 255.255.255.0
!
mpls ldp
 router-id 9.9.9.9
 interface FastEthernet0/4/1/0
 !
```

```
!
multicast-routing
 vrf vpn1 address-family ipv4
  mdt data 233.1.0.0/16 threshold 3
  mdt default ipv4 232.1.1.1
  interface all enable
 !
 vrf vpn1 address-family ipv6
  mdt default ipv4 232.1.1.1
  interface all enable
 !
 address-family ipv4
  multipath
  mdt source Loopback0
  interface all enable
 !
 address-family ipv6
  interface all enable
 !
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 100
 bgp router-id 9.9.9.9
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 address-family ipv6 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 1.1.1.1
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family ipv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf vpn1
  rd 1:1
  address-family ipv4 unicast
   maximum-paths ebgp 3
   redistribute connected
  !
  address-family ipv6 unicast
   maximum-paths ebgp 3
   redistribute connected
  !
  neighbor 2013::122:1:1:2
   remote-as 2
   address-family ipv6 unicast
    route-policy pass-all in
    route-policy pass-all out
   !
  !
 !
!
router ospf 100
```

```
 router-id 9.9.9.9
 area 0
  interface Loopback0
  !
  interface FastEthernet0/4/1/0
  !
 !
!
router pim vrf vpn1 address-family ipv4
 rp-address 192.168.10.1
!
router pim vrf vpn1 address-family ipv6
 rp-address ::192:168:10:1
!
router pim vrf default address-family ipv4
 rp-address 1.1.1.1
!
end
```

CE2：

Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのマニュアルを参照してください。

```
!
interface Loopback0
 ipv4 address 122.122.122.122 255.255.255.255
!
interface GigabitEthernet0/5/0/0
 ipv6 address 2013::80:1:1:2/126
!
interface GigabitEthernet0/5/0/1
 ipv6 address 2013::122:1:1:2/96
!
multicast-routing
 address-family ipv4
  interface all enable
 !
 address-family ipv6
  interface all enable
 !
!
!
route-policy pass-all
  pass
end-policy
!
router bgp 2
 bgp router-id 122.122.122.122
 address-family ipv6 unicast
  redistribute connected
 !
 neighbor 2013::122:1:1:1
  remote-as 100
  address-family ipv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  !
 !
!
router pim vrf default address-family ipv6
 dr-priority 2
 rp-address ::192:168:10:1
!
end
```

MVPN スタティック P2MP TE の設定例

入力 PE での MVPN P2MP の設定例

```
multicast-routing
 address-family ipv4
  mdt source Loopback0
  interface all enable
 !
 vrf vrf1
  address-family ipv4
   bgp auto-discovery rsvpte
   mdt static p2mp-te tunnel-mte1
   interface all enable
  !
router igmp
 vrf vrf1
  interface tunnel-mte1
   static-group 232.1.1.1 192.1.1.2
  !
```

MVPN P2MP BGP の設定例

```
router bgp 100
 bgp router-id 110.110.110.110
 address-family ipv4 unicast
 address-family vpnv4 unicast
 address-family ipv4 mvpn
 !
 neighbor 130.130.130.130
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  address-family vpnv4 unicast
  address-family ipv4 mvpn
 !
 vrf vrf1
  rd 1:1
  bgp router-id 110.110.110.110
  address-family ipv4 unicast
   redistribute connected
  address-family ipv4 mvpn
  !
 !
```

出力 PE での MVPN P2MP の設定例

```
multicast-routing
 address-family ipv4
  mdt source Loopback0
  interface all enable
 !
 vrf vrf1
  address-family ipv4
   core-tree-protocol rsvp-te group-list mvpn-acl
   interface all enable
  !

ipv4 access-list mvpn-acl
 10 permit ipv4 host 192.1.1.2 host 232.1.1.1
```

```
 20 permit ipv4 any host 232.1.1.2
```

MVPN エクストラネット ルーティングの設定例

次の例は、MVPN エクストラネット ルーティングを設定する 2 つの方法を示しています。

設定作業全体については、MVPNエクストラネットルーティングの設定, （192ページ）を参照してください。

レシーバ PE ルータでのソース MVRF の設定例

次に、レシーバ PE ルータでソース MVRF を指定して、MVPN エクストラネット ルーティングを設定する例を示します。

ソース PE ルータとレシーバ PE ルータの両方を設定する必要があります。

ルート ターゲットを使用したソース PE ルータの設定

```
interface Loopback5
 ipv4 address 201.5.5.201 255.255.255.255
!
interface Loopback22
 vrf provider-vrf
 ipv4 address 201.22.22.201 255.255.255.255
!
interface GigabitEthernet0/6/0/0
 vrf provider-vrf
 ipv4 address 10.10.10.1 255.255.0.0
!
vrf provider-vrf
 address-family ipv4 unicast
 import route-target
 1100:1
 !
export route-target
 1100:1
 !
!
router bgp 1
 regular BGP MVPN config
vrf provider-vrf
 rd 1100:1
 address-family ipv4 unicast
 redistribute connected

 !
!
multicast-routing
 vrf provider-vrf address-family ipv4
 mdt data 226.1.4.0/24 threshold 3
 log-traps
 mdt default ipv4 226.0.0.4
 rate-per-route
 interface all enable
 accounting per-prefix
 !
!
address-family ipv4
 nsf
 mdt source Loopback5
 interface all enable
 !
```

```
!
router pim vrf provider-vrf address-family ipv4
 rp-address 201.22.22.201
!
```

ルート ターゲットを使用したレシーバ PE ルータの設定

```
interface Loopback5
 ipv4 address 202.5.5.202 255.255.255.255
!
interface GigabitEthernet0/3/0/2
 vrf receiver-vrf
 ipv4 address 20.20.20.1 255.255.0.0
!
vrf provider-vrf
 address-family ipv4 unicast
 import route-target
 1100:1
 !
 export route-target
 1100:1
 !
!
vrf receiver-vrf
 address-family ipv4 unicast
 import route-target
 1100:1
 1101:1
 !
 export route-target
 1101:1
 !
!
multicast-routing
 vrf provider-vrf address-family ipv4
 log-traps
 mdt default ipv4 226.0.0.4
 rate-per-route
 interface all enable
 accounting per-prefix
!

 vrf receiver_vrf address-family ipv4
 log-traps
 mdt default ipv4 226.0.0.5
 rate-per-route
 interface all enable
 accounting per-prefix
!
address-family ipv4
 nsf
 mdt source Loopback5
 interface all enable
!
router pim vrf provider-vrf address-family ipv4
 rp-address 201.22.22.201
!

router pim vrf receiver_vrf address-family ipv4
 rp-address 201.22.22.201
!
router bgp 1
 regular BGP MVPN config
vrf provider-vrf
 rd 1100:1
 address-family ipv4 unicast
 redistribute connected
 !

 vrf receiver_vrf
```

```
 rd 1101:1
 address-family ipv4 unicast
 redistribute connected
!
```

ソース VRF に join を伝播するためのレシーバ VRF での RPL ポリシーの設定例

ルート ターゲットの設定に加えて、指定されたソース VRF に join を伝播するように、ルーティング ポリシー言語（RPL）ポリシーをレシーバ PE ルータ上のレシーバ VRF で設定できます。 ただし、この設定は任意です。

次の設定例は、レシーバ VRF が「provider_vrf_1」または「provider_vrf_2」を選択して PIM join を伝播するポリシーを示しています。

この例では、provider_vrf_1 は 227.0.0.0 ～ 227.255.255.255 の範囲内のマルチキャスト ストリームに使用され、provider_vrf_2 は 228.0.0.0 ～ 228.255.255.255 の範囲のストリームに使用されます。

```
route-policy extranet_streams_from_provider_vrf
 if destination in (227.0.0.0/32 ge 8 le 32) then
  set rpf-topology vrf provider_vrf_1
 elseif destination in (228.0.0.0/32 ge 8 le 32) then
  set rpf-topology vrf provider_vrf_2
 else
  pass
 endif
end-policy
!
router pim vrf receiver_vrf address-family ipv4
 rpf topology route-policy extranet_streams_from_provider_vrf
!
```

ソース PE ルータでのレシーバ MVRF の設定例

次に、ソース PE ルータでレシーバ MVRF を指定して、MVPN エクストラネット ルーティングを設定する例を示します。

（注） ソース PE ルータとレシーバ PE ルータの両方を設定する必要があります。

ルート ターゲットを使用したソース PE ルータの設定

```
interface Loopback5
 ipv4 address 202.5.5.202 255.255.255.255
!
interface GigabitEthernet0/3/0/2
 vrf provider-vrf
 ipv4 address 20.20.20.1 255.255.0.0
!
vrf provider-vrf
 address-family ipv4 unicast
 import route-target
 1100:1
 !
 export route-target
 1100:1
 !
!
```

```
vrf receiver-vrf
 address-family ipv4 unicast
 import route-target
 1100:1
 1101:1
 !
 export route-target
 1101:1
 !
!

router bgp 1
 regular BGP MVPN config
vrf provider-vrf
 rd 1100:1
 address-family ipv4 unicast
 redistribute connected
 !

 vrf receiver-vrf
 rd 1101:1
 address-family ipv4 unicast
 redistribute connected
 !
!

multicast-routing
 vrf provider-vrf address-family ipv4
 log-traps
 mdt default ipv4 226.0.0.4
 rate-per-route
 interface all enable
 accounting per-prefix
!

 vrf receiver_vrf address-family ipv4
 log-traps
 mdt default ipv4 226.0.0.5
 rate-per-route
 interface all enable
 accounting per-prefix
!
address-family ipv4
 nsf
 mdt source Loopback5
 interface all enable
!
router pim vrf provider-vrf address-family ipv4
 rp-address 201.22.22.201
!
router pim vrf receiver_vrf address-family ipv4
 rp-address 201.22.22.201
!
```

ルート ターゲットを使用したレシーバ PE ルータの設定

```
interface Loopback5
 ipv4 address 201.5.5.201 255.255.255.255
!
interface Loopback22
 vrf receiver_vrf
 ipv4 address 201.22.22.201 255.255.255.255
!
interface GigabitEthernet0/6/0/0
 vrf receiver_vrf
 ipv4 address 10.10.10.1 255.255.0.0
!

vrf receiver_vrf
```

```
 address-family ipv4 unicast
 import route-target
 1100:1
 1101:1
 !
 export route-target
 1101:1
 !
!

router bgp 1
 regular BGP MVPN config
vrf receiver_vrf
 rd 1101:1
 address-family ipv4 unicast
 redistribute connected
 !

multicast-routing
 vrf receiver_vrf address-family ipv4
 log-traps
 mdt default ipv4 226.0.0.5
 rate-per-route
 interface all enable
 accounting per-prefix
 !
address-family ipv4
 nsf
 mdt source Loopback5
 interface all enable
!

router pim vrf receiver_vrf address-family ipv4
 rp-address 201.22.22.201
!
```

ソース VRF に join を伝播するためのソース PE ルータ上のレシーバ VRF での RPL ポリシーの設定例

ルート ターゲットの設定に加えて、指定されたソース VRF に join を伝播するように、RPL ポリシーをソース PE ルータ上のレシーバ VRF で設定できます。 ただし、この設定は任意です。

次の設定は、レシーバ VRF が「provider_vrf_1」または「provider_vrf_2」を選択して PIM join を伝播するポリシーを示しています。 マルチキャスト ストリームのランデブー ポイント（RP）が 201.22.22.201 の場合は provider_vrf_1 が選択され、マルチキャスト ストリームの RP が 202.22.22.201 の場合には provider_vrf_2 が選択されます。

代わりに、ソース VRF に join を伝播するためのレシーバ VRF での RPL ポリシーの設定例, （250 ページ）に示すようにマルチキャスト グループベース ポリシーを設定することもできます。

```
route-policy extranet_streams_from_provider_rp
 if source in (201.22.22.201) then
  set rpf-topology vrf provider_vrf_1
 else if source in (202.22.22.201) then
  set rpf-topology vrf provider_vrf_2
 else
  pass
 endif
end-policy
!
router pim vrf receiver_vrf address-family ipv4
 rpf topology route-policy extranet_streams_from_provider_rp
 rp-address 201.22.22.201 grange_227
 rp-address 202.22.22.201 grange_228
!
```

マルチキャスト ハブ アンド スポーク トポロジの設定例

次の例は、マルチキャスト ハブ アンド スポークを設定する 2 つの方法を示しています。

図 *13*：*CE1 PE1PE3 CE3* マルチキャスト ハブ アンド スポーク トポロジの設定例

CE1---------- PE1 ---------- PE3 ---------- CE3

CE1、PE1、および PE3 は、すべて Cisco IOS XR ソフトウェア上にあり、CE3 には VRF インターフェイス上で Auto-RP を設定するために Cisco IOS ソフトウェアがあります。 Cisco IOS ソフトウェアを使用した CE ルータの設定については、該当する Cisco IOS ソフトウェアのマニュアルを参照してください。

ハブ アンド スポーク Non-Turnaround の設定例

A1-Hub-1 (bsr RP) A1-Hub-4 (auto-rp RP)

A1-Spoke-3

BSR と Auto-RP リレーを使用した Non-Turnaround の場合

PE1：

```
vrf A1-Hub-1
address-family ipv4 unicast
import route-target

   1000:10

   1001:10

  !

  export route-target

   1000:10

  !

 !

vrf A1-Hub-Tunnel
address-family ipv4 unicast

  import route-target

   1000:10

  !

 !

!

vrf A1-Spoke-Tunnel
address-family ipv4 unicast

  import route-target
```

```
   1001:10
  !
 !
!
router pim
 vrf A1-Hub-1
  address-family ipv4
   rpf topology route-policy A1-Hub-Policy
   bsr relay vrf A1-Hub-Tunnel
   bsr candidate-bsr 201.10.10.201 hash-mask-len 30 priority 4
   bsr candidate-rp 201.10.10.201 group-list A1_PE1_RP_grange priority 4 interval 60
   auto-rp relay vrf A1-Hub-Tunnel
  !
 !
!
router pim
 vrf A1-Hub-Tunnel
  address-family ipv4
  !
 !
!
multicast-routing
 vrf A1-Hub-1
  address-family ipv4
   log-traps
   multipath
   rate-per-route
   interface all enable
   accounting per-prefix
  !
 !
!
multicast-routing
 vrf A1-Hub-Tunnel
  address-family ipv4
   mdt data 226.202.1.0/24 threshold 10
   log-traps
```

```
   mdt default ipv4 226.202.0.0
   rate-per-route
   accounting per-prefix
  !
 !
!
multicast-routing
 vrf A1-Spoke-Tunnel
  address-family ipv4
   mdt mtu 2000
   mdt data 226.202.2.0/24 threshold 5
   log-traps
   mdt default ipv4 226.202.0.1
   rate-per-route
   accounting per-prefix
  !
 !
!
router bgp 1
 vrf A1-Hub-1
  rd 1000:1
  address-family ipv4 unicast
   route-target download
   redistribute connected
   redistribute eigrp 20 match internal external metric 1000
  !
 !
!
router bgp 1
 vrf A1-Hub-Tunnel
  rd 1002:1
  address-family ipv4 unicast
   redistribute connected
  !
 !
!
router bgp 1
```

```
 vrf A1-Spoke-Tunnel
  rd 1002:2
  address-family ipv4 unicast
   redistribute connected
  !
 !
!
route-policy A1-Hub-Policy
  if extcommunity rt matches-any (1000:10) then
    set rpf-topology vrf A1-Hub-Tunnel
  elseif extcommunity rt matches-any (1001:10) then
    set rpf-topology vrf A1-Spoke-Tunnel
  else
    pass
  endif
end-policy
!
route-policy A1-Spoke-Policy
  if extcommunity rt matches-any (1000:10) then
    set rpf-topology vrf A1-Hub-Tunnel
  else
    pass
  endif
end-policy
!
```

PE3 :

```
vrf A1-Hub-4
address-family ipv4 unicast
import route-target
   1000:10
   1001:10
  !
  export route-target
   1000:10
  !
 !
!
```

```
vrf A1-Spoke-2
address-family ipv4 unicast
import route-target

   1000:10
  !

  export route-target

   1001:10

  !

 !

!

vrf A1-Hub-Tunnel
address-family ipv4 unicast
import route-target

   1000:10

  !

 !

!

vrf A1-Spoke-Tunnel
address-family ipv4 unicast
import route-target

   1001:10

  !

 !

!

router pim

 vrf A1-Hub-4

  address-family ipv4

   rpf topology route-policy A1-Hub-Policy

   bsr relay vrf A1-Hub-Tunnel listen

   auto-rp relay vrf A1-Hub-Tunnel

  !

 !

!

router pim

 vrf A1-Spoke-2

  address-family ipv4

   rpf topology route-policy A1-Spoke-Policy

   bsr relay vrf A1-Hub-Tunnel listen

   auto-rp relay vrf A1-Hub-4

  !
```

```
 !
!
multicast-routing
 vrf A1-Hub-4
  address-family ipv4
   log-traps
   rate-per-route
   interface all enable
   accounting per-prefix
  !
 !
!
multicast-routing
 vrf A1-Spoke-2
  address-family ipv4
   log-traps
   rate-per-route
   interface all enable
   accounting per-prefix
  !
 !
!
multicast-routing
 vrf A1-Hub-Tunnel
  address-family ipv4
   mdt data 226.202.1.0/24 threshold 10
   log-traps
   mdt default ipv4 226.202.0.0
   rate-per-route
   accounting per-prefix
  !
 !
!
multicast-routing
 vrf A1-Spoke-Tunnel
  address-family ipv4
   mdt data 226.202.2.0/24 threshold 5
```

```
   log-traps
   mdt default ipv4 226.202.0.1
   rate-per-route
   accounting per-prefix
  !
 !
!
router bgp 1
 vrf A1-Hub-4
  rd 1000:4
  address-family ipv4 unicast
   route-target download
   redistribute connected
   redistribute eigrp 4 match internal external metric 1000
  !
 !
!
router bgp 1
 vrf A1-Spoke-2
  rd 1001:2
  address-family ipv4 unicast
   route-target download
   redistribute connected
   redistribute eigrp 6 match internal external metric 1000
  !
 !
router bgp 1
 vrf A1-Hub-Tunnel
  rd 1002:1
  address-family ipv4 unicast
   redistribute connected
  !
 !
!
router bgp 1
 vrf A1-Spoke-Tunnel
  rd 1002:2
  address-family ipv4 unicast
```

```
   redistribute connected
  !
 !
!
route-policy A1-Hub-Policy
  if extcommunity rt matches-any (1000:10) then
    set rpf-topology vrf A1-Hub-Tunnel
  elseif extcommunity rt matches-any (1001:10) then
    set rpf-topology vrf A1-Spoke-Tunnel
  else
    pass
  endif
end-policy
!
route-policy A1-Spoke-Policy
  if extcommunity rt matches-any (1000:10) then
    set rpf-topology vrf A1-Hub-Tunnel
  else
    pass
  endif
end-policy
!
```

CE1：

```
vrf A1-Hub-1
 address-family ipv4 unicast
  import route-target
   1000:10
   1001:10
  !
  export route-target
   1000:10
  !
 !
!
multicast-routing
 vrf A1-Hub-1
  address-family ipv4
```

```
   log-traps
   rate-per-route
   interface all enable
   accounting per-prefix
  !
 !
!
No router pim configuration required
```

CE3：**Auto-RP** が設定されています（**VRF** インターフェイス上の **Auto-RP** は **Cisco IOS XR** ソフトウェアでサポートされていないため、これは **Cisco IOS** ソフトウェアの例です）

```
ip vrf A1-Hub-4
 rd 1000:4
 route-target export 1000:10
 route-target import 1000:10
 route-target import 1001:10
!
ip vrf A1-Spoke-2
 rd 1001:2
 route-target export 1001:10
 route-target import 1000:10
!
ip multicast-routing vrf A1-Hub-4
ip multicast-routing vrf A1-Spoke-2

interface Loopback10
 ip vrf forwarding A1-Hub-4
 ip address 103.10.10.103 255.255.255.255
 ip pim sparse-mode
!
ip pim vrf A1-Hub-4 autorp listener
ip pim vrf A1-Hub-4 send-rp-announce Loopback10 scope 32
ip pim vrf A1-Hub-4 send-rp-discovery Loopback10 scope 32
```

Turnaround を使用したハブ アンド スポークの例

マルチキャストの Turnaround では、ハブ サイトへの 2 インターフェイス接続が必要です。

CE をターンアラウンドルータとして設定するには、CE が 2 つのインターフェイスを介して各 PE に接続され、各インターフェイスは **hub-x-in vrf** および **hub-x-out vrf** という個別のハブ サイト VRF に配置されます。 hub-x-in vrf はレシーバのスポーク サイトからハブ トンネルを介して受信した join を伝送し、hub-x-out vrf は、次の 4 つの基本ルールに違反せずに、スポーク トンネルを介して送信元スポーク サイトに向かって同じ join を伝送します。 送信元スポークは hub-x-out へのスポーク トンネルにトラフィックを送信し、このトンネルが、hub-x-in インターフェイス上のハブ トンネルに方向転換します。

1. ハブ サイトは MDTHub だけにトラフィックを送信します。
2. スポーク サイトは MDTspoke だけにトラフィックを送信します。
3. ハブ サイトは両方のトンネルからトラフィックを受信します。
4. スポーク サイトは MDTHub からだけトラフィックを受信します。

A2-Spoke-1 A2-Hub-2

A2-Spoke-2 A2-Hub-3in

A2-Hub-2out

A2-Spoke-3（スポークに Auto-RP があります）

図 *14*：*Turnaround* を使用した *CE1 PE1 PE2 CE2* マルチキャスト ハブ アンド スポーク トポロジの例

CE1---------- PE1 ---------- PE2 ---------- CE2

ハブ サイトによってエクスポートされたルートはハブ サイトとスポーク サイトでインポートされます。 スポーク サイトによってエクスポートされたルートは **hub-x-out** と **hub-x-in** の両方でインポートされ、ハブ サイトはハブ VRF ルート ターゲットによってスポーク ルートを逆にコアにエクスポートします。 これにより、1 つのスポーク サイトから発信されたルートが、他のすべてのスポーク サイトによって学習されますが、ネクストホップは **hub-x-out** になります。 たとえば、Spoke2 は Spoke1 の RPF を、ネクストホップ **A2-Hub-3in** で到達可能と見なします。 これは、マルチキャスト トラフィックの所要の方向転換の実現に役立つルート リークの基本的な違いです。

PE1：

```
vrf A2-Spoke-1
 address-family ipv4 unicast
  import route-target
   4000:1
   4000:2
   4000:3
   4000:4
  !
  export route-target
   4001:1
```

```
  !
 !
!
```

vrf A2-Spoke-2

```
address-family ipv4 unicast
  import route-target
   4000:1
   4000:2
   4000:3
   4000:4
  !
  export route-target
   4001:2
  !
 !
!
```

PE2：

```
vrf A2-Hub-2
 address-family ipv4 unicast
  import route-target
   4000:1
   4000:2
   4000:3
   4000:4
   4001:1
   4001:2
   4001:3
   4001:4
  !
  export route-target
   4000:2
  !
 !
!
```

```
vrf A2-Hub-3out
 address-family ipv4 unicast
  import route-target
   4000:1
   4000:2
   4000:3
   4000:4
   4001:1   --------à exports the spoke routes into CE2 into vrf default
   4001:2   --------à exports the spoke routes into CE2 into vrf default
   4001:3   --------à exports the spoke routes into CE2 into vrf default
   4001:4   --------à exports the spoke routes into CE2 into vrf default
  !
  export route-target
   4000:4
  !
 !
!
vrf A2-Hub-3in
 address-family ipv4 unicast
  import route-target
   4000:1
   4000:2
   4000:3
   4000:4
  !
  export route-target
   4000:3--------à selected spoke routes (in the prefix-set below) can be re-exported with
 hub route target so other spokes can reach them via A2-Hub-3in
  !
 !
!
prefix-set A2-Spoke-family
  112.31.1.0/24,
  112.32.1.0/24,
  152.31.1.0/24,
  132.30.1.0/24,
  102.9.9.102/32,
```

```
  103.31.31.103/32,
  183.31.1.0/24,
  183.32.1.0/24
end-set
!
route-policy A2-Spoke-family
  if destination in A2-Spoke-family then
    pass
  else
    drop
  endif
end-policy
!


router bgp 1
 vrf A2-Hub-3in
  rd 4000:3
  address-family ipv4 unicast
   route-target download
   redistribute connected
  !
  neighbor 113.113.114.9
   remote-as 12
   address-family ipv4 unicast
```

route-policy A2-Spoke-family in ------スポーク サイトによる RPF A2-Hub-3in でのインポートが可能となるように、選択したスポーク ルートをハブ ルート ターゲットでリークします。

```
    route-policy pass-all out
   !
  !
 !
!
router bgp 1
 vrf A2-Hub-3out
  rd 4000:4
  address-family ipv4 unicast
   route-target download
```

```
   redistribute connected
  !
 !
!
router bgp 1
 vrf A2-Hub-2
  rd 4000:2
  address-family ipv4 unicast
   route-target download
   redistribute connected
   redistribute eigrp 20 match internal external metric 1000
  !
 !
!
multicast-routing
 vrf A2-Hub-2
  address-family ipv4
   log-traps
   rate-per-route
   interface all enable
   accounting per-prefix
  !
 !
!
multicast-routing
 vrf A2-Hub-3in
  address-family ipv4
   log-traps
   rate-per-route
   interface all enable
   accounting per-prefix
  !
 !
!

multicast-routing
 vrf A2-Hub-3out
  address-family ipv4
```

```
   log-traps
   rate-per-route
   interface all enable
   accounting per-prefix
  !


 !
!
router pim
 vrf A2-Hub-2
  address-family ipv4
   rpf topology route-policy A2-Hub-Policy
   bsr relay vrf A2-Spoke-3 listen
   auto-rp relay vrf A2-Hub-Tunnel
  !
 !
!
router pim
vrf A2-Hub-3in
address-family ipv4
rpf topology route-policy A2-Hub-Policy
!
!
!
router pim
vrf A2-Hub-3out
address-family ipv4
rpf topology route-policy A2-Hub-Policy
!
!
!

route-policy A2-Hub-Policy
if extcommunity rt matches-any (4000:1, 4000:2, 4000:3, 4000:4) then
set rpf-topology vrf A2-Hub-Tunnel
elseif extcommunity rt matches-any (4001:1, 4001:2, 4001:3, 4001:4) then
set rpf-topology vrf A2-Spoke-Tunnel
else
pass
endif
end-policy

!
```

任意の CE-PE プロトコルを使用できます。 この例では、A2-Hub-3out が、CE2 への EIGRP を通過するすべてのハブ アンド スポーク ルートをエクスポートします。

A2-Hub-3in は、ルート ポリシー A2-Spoke-family を使用して、選択したスポーク ルートを BGP を通じて PE2 に再インポートします。

```
router eigrp 20
vrf A2-Hub-3out
address-family ipv4
default-metric 1000 1 255 1 1500
```

```
autonomous-system 20
redistribute bgp 1
interface GigabitEthernet0/1/0/1.13
hold-time 60

   !

  !

 !

!
```

CE2：

ここで A2-Hub-3in および A2-Hub-3out インターフェイスは、VRF のデフォルトにあり、ハブ サイトの VRF にはありません。

```
interface GigabitEthernet0/12/1/0.12
description To PE2 or vrf A2-Hub-3in
ipv4 address 113.113.114.9 255.255.255.252
dot1q vlan 3001

!
interface GigabitEthernet0/12/1/0.13
description To PE2 or vrf A2-Hub-3out
ipv4 address 113.113.114.13 255.255.255.252
dot1q vlan 3002
!
router bgp 12
nsr
bgp graceful-restart

 address-family ipv4 unicast
redistribute connected
redistribute eigrp 20
!
neighbor 113.113.114.10   --à this is the A2-Hub-3in neighbor on PE2.
remote-as 1
address-family ipv4 unicast
route-policy pass-all in
route-policy pass-all out
  !
 !
!
```

LSM based MLDP の設定例

次の例では、MLDP based MVPN を設定するための複数のプロファイルについて説明します。

BGP アドバタイズメントを使用しない Rosen MLDP

```
vrf 1
 vpn id 1:1
 address-family ipv4 unicast
  import route-target
   1:1
  !
  export route-target
   1:1
  !
 !
```

```
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
route-policy mldp-1
  set core-tree mldp-default
end-policy
!
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
 !
!
router bgp 100 mvpn
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mdt
  !
 !
 vrf 1
  rd 1:1
  address-family ipv4 unicast
   redistribute connected
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
!
multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
 vrf 1
  address-family ipv4
   interface all enable
   mdt default mldp ipv4 1.1.1.1
   accounting per-prefix
  !
 !
router pim
 vrf 1
  address-family ipv4
   rpf topology route-policy mldp-1
   rp-address 10.1.1.1
```

```
  !
 !
```

BGP アドバタイズメントを使用した Rosen MLDP

```
vrf 101
  vpn id 101:101
 address-family ipv4 unicast
  import route-target
   101:101
  !
  export route-target
   101:101
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback101
 vrf 101
 ipv4 address 10.1.101.1 255.255.255.255
!
route-policy mldp-101
  set core-tree mldp-default
end-policy
!
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface GigabitEthernet0/0/2/0
  !
  interface GigabitEthernet0/3/2/1
  !
  interface GigabitEthernet0/3/2/2
  !
 !
 mpls traffic-eng router-id Loopback0
!
router bgp 100 mvpn
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mvpn
 !
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mvpn
  !
 !
 vrf 101
  rd 101:101
  address-family ipv4 unicast
   redistribute connected
  !
```

```
  address-family ipv4 mvpn
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
 !
!
multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
!
router pim
 vrf 101
  address-family ipv4
   rpf topology route-policy mldp-101
   vpn-id 101
   rp-address 10.1.101.1
  !
 !
```

VRF のインバンド プロファイル

```
vrf 250
 address-family ipv4 unicast
  import route-target
   250:250
  !
  export route-target
   250:250
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback250
 vrf 250
 ipv4 address 10.1.250.1 255.255.255.255
!
route-policy mldp-250
  set core-tree mldp-inband
end-policy
!
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface GigabitEthernet0/0/2/0
  !
  interface GigabitEthernet0/3/2/1
  !
  interface GigabitEthernet0/3/2/2
  !
 !
```

```
 mpls traffic-eng router-id Loopback0
!
router bgp 100
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
 !
 vrf 250
  rd 250:250
  address-family ipv4 unicast
   redistribute connected
  !
  address-family ipv4 mvpn
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
 !
!
multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
 vrf 250
  address-family ipv4
   mdt mldp in-band-signaling
   interface all enable
  !
 !
router pim
 vrf 250
  address-family ipv4
   rpf topology route-policy mldp-250
   rp-address 10.1.250.1
  !
 !
```

BGP-AD を使用しないパーティション化 MDT MP2MP

```
vrf 251
 address-family ipv4 unicast
  import route-target
   251:251
  !
  export route-target
   251:251
```

```
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback251
 vrf 251
 ipv4 address 10.11.1.1 255.255.255.255
!
route-policy mldp-251
  set core-tree mldp-partitioned-mp2mp
end-policy
!
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface GigabitEthernet0/0/2/0
  !
  interface GigabitEthernet0/3/2/1
  !
  interface GigabitEthernet0/3/2/2
  !
 !
 mpls traffic-eng router-id Loopback0
!
router bgp 100
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
!
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
 !
 vrf 251
  rd 251:251
  address-family ipv4 unicast
   redistribute connected
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
 !
!
multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
```

```
  interface all enable
  accounting per-prefix
 !
 vrf 251
  address-family ipv4
   mdt partitioned mldp ipv4 mp2mp
   interface all enable
  !
 !
router pim
 vrf 251
  address-family ipv4
   rpf topology route-policy mldp-251
   rp-address 10.11.1.1
  !
 !
```

BGP-AD を使用したパーティション化 MDT MP2MP

```
vrf 301
 address-family ipv4 unicast
  import route-target
   301:301
  !
  export route-target
   301:301
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback301
 vrf 301
 ipv4 address 10.11.51.1 255.255.255.255
!
route-policy mldp-301
  set core-tree mldp-partitioned-mp2mp
end-policy
!
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface GigabitEthernet0/0/2/0
  !
  interface GigabitEthernet0/3/2/1
  !
  interface GigabitEthernet0/3/2/2
  !
 !
 mpls traffic-eng router-id Loopback0
!
router bgp 100
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mvpn
 !
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
```

```
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mvpn
  !
 !
 vrf 301
  rd 301:301
  address-family ipv4 unicast
   redistribute connected
  !
  address-family ipv4 mvpn
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
 !
!

multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
 vrf 301
  address-family ipv4
   bgp auto-discovery mldp
   mdt partitioned mldp ipv4 mp2mp
   interface all enable
  !
 !
router pim
 vrf 301
  address-family ipv4
   rpf topology route-policy mldp-301
   rp-address 10.11.51.1
  !
 !
```

BGP アドバタイズメントを使用した Multidirectional Selective Provider Multicast Service Instance mLDP-P2MP

```
vrf 401
 address-family ipv4 unicast
  import route-target
   401:401
  !
  export route-target
   401:401
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback401
 vrf 401
```

```
 ipv4 address 10.11.151.1 255.255.255.255
!
route-policy mldp-401
  set core-tree mldp-partitioned-p2mp
end-policy
!
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface GigabitEthernet0/0/2/0
  !
  interface GigabitEthernet0/3/2/1
  !
  interface GigabitEthernet0/3/2/2
  !
 !
 mpls traffic-eng router-id Loopback0
!
router bgp 100
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mvpn
 !
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mvpn
  !
 !
 vrf 401
  rd 401:401
  address-family ipv4 unicast
   redistribute connected
  !
  address-family ipv4 mvpn
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
 !
!
multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
```

```
 vrf 401
  address-family ipv4
   bgp auto-discovery mldp
   mdt partitioned mldp ipv4 p2mp
   interface all enable
  !
 !
router pim
 vrf 401
  address-family ipv4
   rpf topology route-policy mldp-401
   rp-address 10.11.151.1
  !
```

BGP アドバタイズメントを使用した Rosen-GRE

```
vrf 501
  address-family ipv4 unicast
  import route-target
   501:501
  !
  export route-target
   501:501
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface Loopback501
 vrf 501
 ipv4 address 10.111.1.1 255.255.255.255
!

<no route policy?>

 vrf 501
  rd 501:501
  address-family ipv4 unicast
   redistribute connected
  !
  address-family ipv4 mvpn
  !
 !
router ospf 1
 address-family ipv4 unicast
 area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface GigabitEthernet0/0/2/0
  !
  interface GigabitEthernet0/3/2/1
  !
  interface GigabitEthernet0/3/2/2
  !
 !
 mpls traffic-eng router-id Loopback0
!
router bgp 100
 address-family ipv4 unicast
  redistribute connected
 !
 address-family vpnv4 unicast
 !
 address-family vpnv6 unicast
 !
 address-family ipv4 mvpn
 !
```

```
 neighbor 5.5.5.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  address-family vpnv6 unicast
  !
  address-family ipv4 mvpn
  !
 !
 vrf 501
  rd 501:501
  address-family ipv4 unicast
   redistribute connected
  !
  address-family ipv4 mvpn
  !
 !
mpls traffic-eng
 interface GigabitEthernet0/0/2/0
 !
!
mpls ldp
 router-id 1.1.1.1
 graceful-restart
 mldp
  logging internal
 !
 <all core-facing interfaces>
 !
!
multicast-routing
 address-family ipv4
  nsf
  mdt source Loopback0
  interface all enable
  accounting per-prefix
 !
 vrf 501
  address-family ipv4
   bgp auto-discovery pim
   mdt default ipv4 232.1.1.1
   interface all enable
  !
 !
router pim
 vrf 501
  address-family ipv4
   rp-address 10.111.1.1
  !
 !
```

その他の参考資料

関連資料

関連項目	参照先
マルチキャスト コマンド リファレンス	『*Cisco ASR 9000 Series Aggregation Services Router Multicast Command Reference*』

関連項目	参照先
スタートアップ資料	『*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*』
モジュラ Quality of Service コマンド リファレンス	『*Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Command Reference*』
ルーティング コマンド リファレンスおよびコンフィギュレーション マニュアル	『*Cisco ASR 9000 Series Aggregation Services Router Routing Command Reference*』 『*Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide*』
ユーザ グループとタスク ID に関する情報	『*Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide*』

索 引

A

B

C

D

G

I

L

M

N

O

P

Q

R

S

T

U

V

い

き

く

け

こ

さ

し

す

せ

そ

た

て

と

な

に

は

ひ

ふ

へ

ほ

ま

も

ゆ

り

る

れ

ろ