



Cisco IOS XE Amsterdam 17.2.x (Catalyst 9300 スイッチ) マルチ プロトコルラベルスイッチング (MPLS) コンフィギュレーションガイド

初版 : 2020 年 3 月 30 日

最終更新 : 2020 年 5 月 25 日

シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先 : シスコ コンタクトセンター

0120-092-255 (フリーコール、携帯・PHS含む)

電話受付時間 : 平日 10:00~12:00、13:00~17:00

<http://www.cisco.com/jp/go/contactcenter/>

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020 Cisco Systems, Inc. All rights reserved.



目次

第 1 章

マルチプロトコル ラベル スイッチング (MPLS) の設定 1

Multiprotocol Label Switching : マルチプロトコル ラベル スイッチング 1

マルチプロトコル ラベル スイッチングの制約事項 1

マルチプロトコル ラベル スイッチングに関する情報 1

マルチプロトコル ラベル スイッチングの機能の説明 2

ラベル スイッチング機能 2

ラベル バインディングの配布 2

MPLS レイヤ 3 VPN 3

MPLS QoS EXP の分類とマーキング 3

マルチプロトコル ラベル スイッチングの設定方法 4

MPLS スイッチング用のスイッチの設定 4

MPLS 転送用のスイッチの設定 5

マルチプロトコル ラベル スイッチングの設定の確認 6

MPLS スイッチングの構成の確認 7

MPLS 転送の構成の確認 7

マルチプロトコル ラベル スイッチングに関するその他の参考資料 9

マルチプロトコル ラベル スイッチングの機能履歴 9

第 2 章

MPLS レイヤ 3 VPN の設定 11

MPLS レイヤ 3 VPNs 11

MPLS バーチャルプライベート ネットワークの前提条件 11

MPLS バーチャルプライベート ネットワークの制約事項 12

MPLS バーチャルプライベート ネットワークに関する情報 14

MPLS バーチャルプライベート ネットワークの定義 14

MPLS バーチャルプライベート ネットワークの仕組み	15
MPLS バーチャルプライベート ネットワークの主要コンポーネント	15
MPLS バーチャルプライベート ネットワークの利点	16
MPLS バーチャルプライベート ネットワークの設定方法	18
コア ネットワークの設定	19
MPLS バーチャルプライベート ネットワーク カスタマーの接続	20
バーチャルプライベート ネットワークの設定の確認	23
MPLS バーチャルプライベート ネットワーク サイト間の接続の確認	23
MPLS バーチャルプライベート ネットワーク (VPN) の設定例	25
例：RIP を使用した MPLS バーチャルプライベート ネットワークの設定	26
例：スタティック ルートを使用した MPLS バーチャルプライベート ネットワークの設定	27
例：BGP を使用した MPLS バーチャルプライベート ネットワークの設定	28
その他の参考資料	30
MPLS バーチャルプライベート ネットワークの機能履歴	30

第 3 章

eBGP および iBGP マルチパスの設定 33

MPLS-VPN における eBGP および iBGP に対する BGP マルチパス ロードシェアリング	33
MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの前提条件	33
MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの制約事項	34
MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングについて	34
eBGP と iBGP 間のマルチパス ロードシェアリング	34
BGP MPLS ネットワークにおける eBGP および iBGP のマルチパス ロードシェアリング	35
eBGP および iBGP の両方に対するマルチパス ロードシェアリングの利点	36
MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの設定方法	36
eBGP および iBGP の両方に対するマルチパス ロードシェアリングの設定	36
eBGP および iBGP の両方に対するマルチパス ロードシェアリングの設定の確認	38

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリング機能の設定例 38

eBGP および iBGP のマルチパス ロードシェアリングの設定例 38

MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの機能情報 39

第 4 章

EIGRP MPLS VPN PE-CE Site of Origin の設定 41

EIGRP MPLS VPN PE-CE Site of Origin 41

EIGRP MPLS VPN PE-CE Site of Origin の前提条件 41

EIGRP MPLS VPN PE-CE Site of Origin の制約事項 42

EIGRP MPLS VPN PE-CE Site of Origin について 42

EIGRP MPLS VPN PE-CE Site of Origin サポートの概要 42

バックドア リンクに対する Site of Origin のサポート 42

Site of Origin 拡張コミュニティとルータとの相互運用 43

Site of Origin を EIGRP に伝送する BGP VPN ルートの再配布 44

EIGRP MPLS VPN PE-CE Site of Origin サポート機能の利点 44

EIGRP MPLS VPN PE-CE Site of Origin サポートの設定方法 44

Site of Origin 拡張コミュニティの設定 44

SoO 拡張コミュニティの設定の確認 47

EIGRP MPLS VPN PE-CE SoO の設定例 47

Site of Origin 拡張コミュニティの設定例 47

Site of Origin 拡張コミュニティの確認の例 48

EIGRP MPLS VPN PE-CE Site of Origin の機能履歴 49

第 5 章

Ethernet-over-MPLS (EoMPLS) および疑似回線冗長性の設定 51

Ethernet-over-MPLS の設定 51

Ethernet-over-MPLS の前提条件 51

Ethernet-over-MPLS の制約事項 52

Ethernet-over-MPLS ポートモードの制約事項 52

EoMPLS VLAN モードの制約事項 52

Ethernet-over-MPLS に関する情報 53

	Ethernet-over-MPLS の設定方法	54
	Ethernet-over-MPLS ポートモードの設定	54
	Ethernet-over-MPLS VLAN モードの設定	58
	Ethernet-over-MPLS の設定例	63
	疑似回線冗長性 の設定	68
	疑似回線冗長性の前提条件	68
	疑似回線冗長性の制約事項	68
	疑似回線冗長性ポートモードの制約事項	68
	疑似回線冗長性 VLAN モードの制約事項	69
	疑似回線冗長性について	69
	疑似回線冗長性 の設定方法	70
	疑似回線冗長性ポートモードの設定	70
	疑似回線冗長性 VLAN モードの設定	75
	疑似回線冗長性 の設定例	82
	Ethernet-over-MPLS および疑似回線冗長性の機能情報	85
<hr/>		
第 6 章	MPLS を介した IPv6 プロバイダー エッジ (6PE) の設定	87
	6PE の前提条件	87
	6PE の制約事項	87
	6PE について	87
	6PE の設定	88
	6PE の設定例	91
	MPLS を介した IPv6 プロバイダーエッジ (6PE) の機能履歴	93
<hr/>		
第 7 章	MPLS を介した IPv6 VPN プロバイダー エッジ (6VPE) の設定	95
	6VPE の設定	95
	6VPE の制約事項	95
	6VPE について	95
	6VPE の設定例	96
	MPLS を介した IPv6 VPN プロバイダーエッジ (6VPE) の機能履歴	100

第 8 章

MPLS VPN InterAS オプションの設定 101

- MPLS VPN InterAS オプションに関する情報 101
 - ASE および ASBR 101
 - MPLS VPN InterAS オプション 102
 - InterAS オプション B 102
- MPLS VPN InterAS オプションの設定方法 105
 - MPLS VPN InterAS オプション B の設定 105
 - ネクストホップセルフ方式を使用した InterAS オプション B の設定 105
 - Redistribute Connected 方式を使用した InterAS オプション B の設定 111
- MPLS VPN InterAS オプションの設定の確認 115
- MPLS VPN InterAS オプションの設定例 116
 - ネクストホップセルフ方式 116
 - IGP Redistribute Connected Subnet 方式 122
- MPLS VPN InterAS オプションに関するその他の参考資料 128
- MPLS VPN InterAS オプションの機能履歴 128

第 9 章

MPLS over GRE の設定 131

- MPLS over GRE の前提条件 131
- GRE を介した MPLS の制約事項 131
- MPLS over GRE に関する情報 132
 - PE-to-PE トンネリング 132
 - P-to-PE トンネリング 133
 - P-to-P トンネリング 133
- GRE を介した MPLS の設定方法 134
 - MPLS over GRE トンネルインターフェイスの設定 134
- MPLS over GRE の設定例 135
 - 例 : PE-to-PE トンネリング 135
 - 例 : P-to-PE トンネリング 137
 - 例 : P-to-P トンネリング 138
- MPLS over GRE に関するその他の参考資料 139

MPLS over GRE の機能情報 139

第 10 章

GRE を介した MPLS レイヤ 2 VPN の設定 141

GRE を介した MPLS レイヤ 2 VPN に関する情報 141

トンネリング設定のタイプ 141

PE-to-PE トンネリング 141

P-to-PE トンネリング 142

P-to-P トンネリング 143

GRE を介した MPLS レイヤ 3 VPN の設定方法 143

GRE を介した MPLS レイヤ 2 VPN の設定例 144

例：非 MPLS ネットワークにまたがる GRE トンネルの設定 144

GRE を介した MPLS レイヤ 2 VPN の設定に関するその他の参考資料 145

GRE を介した MPLS レイヤ 2 VPN の設定の機能情報 145

第 11 章

GRE を介した MPLS レイヤ 3 VPN の設定 147

GRE を介した MPLS レイヤ 3 VPN の前提条件 147

GRE を介した MPLS レイヤ 3 VPN の制約事項 148

GRE を介した MPLS レイヤ 3 VPN に関する情報 148

トンネリング設定のタイプ 148

PE-to-PE トンネリング 148

P-to-PE トンネリング 149

P-to-P トンネリング 150

GRE を介した MPLS レイヤ 3 VPN の設定方法 150

GRE を介した MPLS レイヤ 3 VPN の設定例 151

例：GRE を介した MPLS レイヤ 3 VPN (PE-to-PE トンネリング) の設定 151

例：GRE を介した MPLS レイヤ 3 VPN (P-to-PE トンネリング) の設定 154

GRE を介した MPLS レイヤ 3 VPN の設定の機能情報 157

第 12 章

MPLS QoS : EXP の分類およびマーキング 159

MPLS EXP の分類とマーキング 159

MPLS EXP の分類とマーキングの前提条件 159

MPLS EXP の分類とマーキングの制約事項	159
MPLS EXP の分類とマーキングに関する情報	160
MPLS EXP の分類とマーキングの概要	160
MPLS 実験フィールド	160
MPLS EXP の分類とマーキングのメリット	161
MPLS EXP の分類とマーキングの方法	161
MPLS カプセル化パケットの分類	161
最も外側のラベルでの MPLS EXP のマーキング	162
ラベルスイッチドパケットでの MPLS EXP のマーキング	164
条件付きマーキングの設定	165
MPLS EXP の分類とマーキングの設定例	167
例：MPLS カプセル化パケットの分類	167
例：最も外側のラベルでの MPLS EXP のマーキング	168
例：ラベルスイッチドパケットの MPLS EXP のマーキング	169
例：条件付きマーキングの設定	169
その他の参考資料	170
QoS MPLS EXP の機能履歴	170

 第 13 章

MPLS スタティックラベルの設定	171
MPLS スタティック ラベル	171
MPLS スタティック ラベルの前提条件	171
MPLS スタティック ラベルの制限事項	171
MPLS スタティック ラベルに関する情報	172
MPLS スタティック ラベルの概要	172
MPLS スタティック ラベルの利点	172
MPLS スタティック ラベルの設定方法	172
MPLS スタティック プレフィックス ラベル バインディングの設定	172
MPLS スタティック Prefix/Label バインディングの確認	173
MPLS スタティック ラベルの監視とメンテナンス	174
MPLS スタティック ラベルの設定例	175
例：MPLS スタティック Prefix/Label の設定	175

その他の参考資料	176
MPLS スタティックラベルの機能履歴	177

第 14 章

仮想プライベート LAN サービス (VPLS) および VPLS BGP ベースの自動検出の設定	179
VPLS の制約事項	179
VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport に関する情報	180
VPLS の概要	180
フルメッシュ構成について	180
VPLS BGP ベースの自動検出について	181
Flow Aware Transport 疑似回線について	182
Cisco Catalyst 6000 シリーズ スイッチと Cisco Catalyst 9000 シリーズ スイッチ間の相互運用性	183
VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport の設定方法	183
CE デバイスへのレイヤ 2 PE デバイスインターフェイスの設定	184
CE デバイスからのタグ付きトラフィックを受け取る PE デバイスの 802.1Q トランクの設定	184
CE デバイスからのタグなしトラフィックを受け取る PE デバイスの 802.1Q アクセスポートの設定	185
PE デバイスでのレイヤ 2 VLAN インスタンスの設定	186
VPLS の設定	187
Xconnect モードでの VPLS の設定	187
プロトコル CLI モードでの VPLS の設定	190
VPLS BGP ベースの自動検出の設定	197
VPLS BGP ベースの自動検出のイネーブル化	197
VPLS 自動検出を有効にする BGP の設定	198
プロトコル CLI モードでの VPLS BGP ベースの自動検出の設定	200
VPLS および VPLS BGP ベースの自動検出の設定例	204
例：Xconnect モードでの VPLS の設定	204
例：Xconnect モードで設定された VPLS の確認	205
例：テンプレートを使用した VPLS Flow Aware Transport の設定 (プロトコル CLI モード)	207
例：VPLS BGP 自動検出の設定	208

例：VPLS BGP 自動検出の確認	208
VPLS および VPLS BGP ベースの自動検出の機能情報	209

第 15 章

MPLS VPN ルートターゲット書き換えの設定	211
MPLS VPN ルート ターゲット書き換えの前提条件	211
MPLS VPN ルート ターゲット書き換えの制約事項	211
MPLS VPN ルート ターゲット書き換えに関する情報	211
ルートターゲット置換ポリシー	212
ルートマップおよびルート ターゲットの置換	212
MPLS VPN ルート ターゲット書き換えの設定方法	213
ルートターゲット置換ポリシーの設定	213
ルートターゲット置換ポリシーの適用	217
特定の BGP ネイバーへのルート マップの割り当て	217
ルートターゲット置換ポリシーの確認	219
MPLS VPN ルート ターゲット書き換えの設定例	220
例：ルート ターゲット置換ポリシーの適用	220
例：特定の BGP ネイバーへのルート マップの割り当て	220
MPLS VPN ルートターゲット書き換えの機能履歴	221

第 16 章

MPLS VPN-Inter-AS-IPv4 BGP ラベル配布の設定	223
MPLS VPN Inter-AS IPv4 BGP ラベル配布	223
MPLS VPN Inter-AS IPv4 BGP ラベル配布	224
MPLS VPN Inter-AS IPv4 BGP ラベル配布に関する情報	224
MPLS VPN Inter-AS IPv4 BGP ラベル配布の概要	224
BGP ルーティング情報	225
BGP においてルートとともに MPLS ラベルが送信される方法	226
ルートマップを使用したルートのフィルタリング	226
MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定方法	226
IPv4 ルートおよび MPLS ラベルを交換する ASBR の設定	227
VPNv4 ルートを交換するルートリフレクタの設定	229
自律システム内でリモートルートを反映するルートリフレクタの設定	232

ルートマップの作成	234
着信ルート用のルートマップの設定	235
発信ルート用のルートマップの設定	236
ASBR へのルートマップの適用	238
MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定の確認	240
ルート リフレクタ設定の確認	240
CE1 に CE2 のネットワーク到達可能性情報があることの確認	241
PE1 に CE2 のネットワーク層到達可能性情報があることの確認	242
PE2 に CE2 のネットワーク到達可能性情報があることの確認	244
ASBR の設定の確認	245
MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定例	247
BGP を使用して MPLS VPN サービスプロバイダー経由でルートおよび MPLS ラベルを配布する Inter-AS の設定例	247
例：ルートリフレクタ 1 (MPLS VPN サービスプロバイダー)	247
設定例：ASBR1 (MPLS VPN サービスプロバイダー)	249
設定例：ルートリフレクタ 2 (MPLS VPN サービスプロバイダー)	250
設定例：ASBR2 (MPLS VPN サービスプロバイダー)	251
設定例：BGP を使用して非 MPLS VPN サービスプロバイダー経由でルートおよび MPLS ラベルを配布する Inter-AS	253
設定例：ルートリフレクタ 1 (非 MPLS VPN サービスプロバイダー)	253
設定例：ASBR1 (非 MPLS VPN サービスプロバイダー)	255
設定例：ルートリフレクタ 2 (非 MPLS VPN サービスプロバイダー)	256
設定例：ASBR2 (非 MPLS VPN サービスプロバイダー)	257
設定例：ASBR3 (非 MPLS VPN サービスプロバイダー)	258
設定例：ルートリフレクタ 3 (非 MPLS VPN サービスプロバイダー)	260
設定例：ASBR4 (非 MPLS VPN サービスプロバイダー)	260
MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定の機能履歴	262
<hr/>	
第 17 章	シームレス MPLS の設定 265
	シームレス MPLS に関する情報 265
	シームレス MPLS の概要 265

シームレス MPLS のアーキテクチャ	266
シームレス MPLS の設定方法	267
PE ルータでのシームレス MPLS の設定	267
ルートリフレクタでのシームレス MPLS の設定	269
シームレス MPLS の設定例	273
例：PE ルータ 1 でのシームレス MPLS の設定	273
例：ルートリフレクタ 1 でのシームレス MPLS の設定	274
例：PE ルータ 2 でのシームレス MPLS の設定	274
例：ルートリフレクタ 2 でのシームレス MPLS の設定	275
シームレス MPLS の機能履歴	275



第 1 章

マルチプロトコル ラベル スイッチング (MPLS) の設定

- [マルチプロトコル ラベル スイッチング \(1 ページ\)](#)
- [マルチプロトコル ラベル スイッチングの制約事項 \(1 ページ\)](#)
- [マルチプロトコル ラベル スイッチングに関する情報 \(1 ページ\)](#)
- [マルチプロトコル ラベル スイッチングの設定方法 \(4 ページ\)](#)
- [マルチプロトコル ラベル スイッチングの設定の確認 \(6 ページ\)](#)
- [マルチプロトコル ラベル スイッチングに関するその他の参考資料 \(9 ページ\)](#)
- [マルチプロトコル ラベル スイッチングの機能履歴 \(9 ページ\)](#)

マルチプロトコル ラベル スイッチング

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

マルチプロトコル ラベル スイッチングの制約事項

- マルチプロトコルラベルスイッチング (MPLS) フラグメンテーションはサポートされていません。
- MPLS 最大伝送ユニット (MTU) はサポートされていません。

マルチプロトコル ラベル スイッチングに関する情報

マルチプロトコルラベルスイッチング (MPLS) は、レイヤ 3 (ネットワーク層) ルーティングの実績のある拡張性とレイヤ 2 (データリンク層) スイッチングのパフォーマンスおよび機能を組み合わせたものです。MPLSにより、既存のネットワークインフラストラクチャを犠牲にすることなく、サービスを差別化する機会を提供しながら、ネットワーク使用率の急激な増加の課題に対処できるようになります。MPLS アーキテクチャは柔軟性があり、レイヤ 2 テク

ノロジーを任意に組み合わせて使用することができます。MPLSのサポートは、すべてのレイヤ3プロトコルに対して提供され、今日のネットワークで一般的に提供されているものよりもはるかに優れたスケーリングが可能です。

マルチプロトコル ラベルスイッチングの機能の説明

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

ラベルスイッチング機能

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

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

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

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

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

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

ラベルバインディングの配布

ネットワーク内の各ラベルスイッチングルータ（LSR）は、転送同等クラスを表すためにどのラベル値を使用するかについて独立したローカルな決定を行います。このアソシエーションは、ラベルバインディングと呼ばれます。各LSRは、自身が行ったラベルバインディングを

ネイバーに通知します。このようにネイバー スイッチにラベル バインディングを認識させる処理は、次のプロトコルによって促進されます。

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

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

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



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

MPLS レイヤ 3 VPN

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

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

MPLS QoS EXP の分類とマーキング

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

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

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

機能制限

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

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

マルチプロトコル ラベル スイッチング の設定方法

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

MPLS スイッチング用のスイッチの設定

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



(注) `ip unnumbered` コマンドは MPLS 設定ではサポートされていません。

手順の概要

1. `enable`
2. `configure terminal`
3. `ip cef distributed`
4. `mpls label range minimum-value maximum-value`
5. `mpls label protocol ldp`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip cef distributed 例 : Device(config)# ip cef distributed	スイッチでシスコ エクスプレス フォワーディングをイネーブルにします。
ステップ 4	mpls label range minimum-value maximum-value 例 : Device(config)# mpls label range 16 4096	パケット インターフェイス上で MPLS アプリケーションで使用可能なローカルラベルの範囲を設定します。
ステップ 5	mpls label protocol ldp 例 : Device(config)# mpls label protocol ldp	プラットフォームの Label Distribution Protocol を指定します。

MPLS 転送用のスイッチの設定

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



(注) **ip unnumbered** コマンドは MPLS 設定ではサポートされていません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface type slot/subslot /port**
4. **mpls ip**
5. **mpls label protocol ldp**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface type slots/subslot /port 例 : Device(config)# interface gigabitethernet 1/0/0 Device(config)# interface vlan 1000	ギガビット イーサネット インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。スイッチ仮想インターフェイス (SVI) の場合の例を次に示します。
ステップ 4	mpls ip 例 : Device(config-if)# mpls ip	ルーテッド物理インターフェイス (ギガビットイーサネット)、スイッチ仮想インターフェイス (SVI)、またはポート チャネルに沿った IPv4 パケットの MPLS 転送を有効にします。
ステップ 5	mpls label protocol ldp 例 : Device(config-if)# mpls label protocol ldp	インターフェイスの Label Distribution Protocol を指定します。 (注) MPLS LDP は、Virtual Routing and Forwarding (VRF) インターフェイスで有効にすることはできません。
ステップ 6	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

マルチプロトコル ラベル スイッチング の設定の確認

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

MPLS スイッチングの構成の確認

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

手順の概要

1. **show ip cef summary**

手順の詳細

show ip cef summary

例 :

```
Device# show ip cef summary

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

MPLS 転送の構成の確認

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



-
- (注) MPLS MTU 値は、デフォルトではポートまたはスイッチの IP MTU 値と同等です。MPLS の MTU 設定はサポートされていません。
-

手順の概要

1. **show mpls interfaces detail**
2. **show running-config interface**
3. **show mpls forwarding**

手順の詳細

ステップ 1 show mpls interfaces detail

例 :

```

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

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

```

```

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

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

```

ステップ2 show running-config interface

例 :

```

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

Building configuration...

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

```

```

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

Building configuration...

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

```

ステップ3 show mpls forwarding

例 :

```

For physical (Gigabit Ethernet) interface:
Device# show mpls forwarding-table

```

```

Local      Outgoing  Prefix          Bytes Label  Outgoing  Next Hop
Label      Label     or Tunnel Id   Switched     interface
500        No Label  12ckt (3)      0            Gi3/0/22  point2point
501        No Label  12ckt (1)      12310411816789 none       point2point
502        No Label  12ckt (2)      0            none      point2point
503        566      15.15.15.15/32 0            Po5       192.1.1.2
504        530      7.7.7.7/32     538728528   Po5       192.1.1.2
505        573      6.6.6.10/32   0            Po5       192.1.1.2
506        606      6.6.6.6/32    0            Po5       192.1.1.2
507        explicit-n 1.1.1.1/32    0            Po5       192.1.1.2
556        543      19.10.1.0/24  0            Po5       192.1.1.2
567        568      20.1.1.0/24   0            Po5       192.1.1.2
568        574      21.1.1.0/24   0            Po5       192.1.1.2
574        No Label  213.1.1.0/24[V] 0            aggregate/vpn113
575        No Label  213.1.2.0/24[V] 0            aggregate/vpn114
576        No Label  213.1.3.0/24[V] 0            aggregate/vpn115
577        No Label  213:1:1::/64   0            aggregate
594        502      103.1.1.0/24  0            Po5       192.1.1.2
595        509      31.1.1.0/24   0            Po5       192.1.1.2
596        539      15.15.1.0/24  0            Po5       192.1.1.2
597        550      14.14.1.0/24  0            Po5       192.1.1.2
633        614      2.2.2.0/24    0            Po5       192.1.1.2
634        577      90.90.90.90/32 873684     Po5       192.1.1.2
635        608      154.1.1.0/24  0            Po5       192.1.1.2
636        609      153.1.1.0/24  0            Po5       192.1.1.2
Device# end

```

マルチプロトコルラベルスイッチングに関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	『 <i>Command Reference (Catalyst 9300 Series Switches)</i> 』の「マルチプロトコルラベルスイッチング (MPLS) コマンド」の項を参照してください。

マルチプロトコルラベルスイッチングの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	マルチプロトコルラベルスイッチング	マルチプロトコルラベルスイッチングは、レイヤ3（ネットワーク層）ルーティングの実績のある拡張性とレイヤ2（データリンク層）スイッチングのパフォーマンスおよび機能を組み合わせたものです。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 2 章

MPLS レイヤ 3 VPN の設定

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

- [MPLS レイヤ 3 VPNs \(11 ページ\)](#)

MPLS レイヤ 3 VPNs

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

MPLS バーチャルプライベートネットワークの前提条件

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

MPLS バーチャル プライベート ネットワークの制約事項

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

MPLS 環境でサポートされるスタティック ルート

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

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

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

- **ip route destination-prefix mask interface1 next-hop1**
- **ip route destination-prefix mask interface2 next-hop2**

TFIB を使用する MPLS 環境でサポートされないスタティック ルート

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

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

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

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

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

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

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

MPLS VPN 環境でサポートされるスタティック ルート

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

- **ip route vrf vrf-name destination-prefix mask next-hop-address**
- **ip route vrf vrf-name destination-prefix mask interface next-hop-address**
- **ip route vrf vrf-name destination-prefix mask interface1 next-hop1**

- **ip route vrf vrf-name destination-prefix mask interface2 next-hop2**

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

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

MPLS VPN 環境でスタティックルートを設定し、スタティックな非再帰ルートと特定のアウトバウンドインターフェイスを使用するロードシェアリングを有効にする場合、次の **ip route** コマンドがサポートされます。

- **ip route destination-prefix mask interface1 next-hop1**
- **ip route destination-prefix mask interface2 next-hop2**

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

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

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

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

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

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

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

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

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

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

MPLS VPN 環境でスタティックルートを設定し、ネクストホップが CE 側のグローバルテーブルに存在し、スタティックな非再帰ルートと特定のアウトバウンドインターフェイスを使用するロードシェアリングを有効にする場合、次の `ip route` コマンドがサポートされます。

- `ip route destination-prefix mask interface1 nexthop1`
- `ip route destination-prefix mask interface2 nexthop2`

MPLS バーチャル プライベート ネットワークに関する情報

この項では、MPLS バーチャルプライベート ネットワークについて説明します。

MPLS バーチャル プライベート ネットワークの定義

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

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

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

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

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

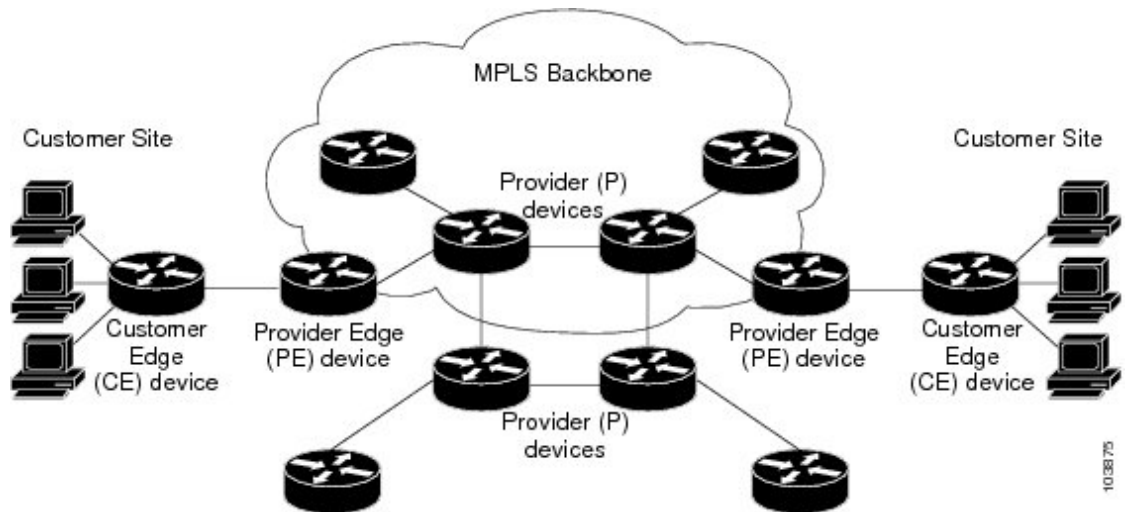
MPLS VPN のさまざまな部分について、次に説明します。

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

- カスタマー (C) デバイス : ISP または企業ネットワークのデバイス。
- CE デバイス : ネットワーク上の PE デバイスに接続する、ISP のネットワーク上のエッジデバイス。CE デバイスは、PE デバイスとインターフェイスする必要があります。

次の図に、基本的な MPLS VPN を示します。

図 1: 基本的 MPLS VPN 用語



MPLS バーチャル プライベート ネットワークの仕組み

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

- カスタマーエッジ (CE) デバイスとルーティングアップデートを交換する。
- CE ルーティング情報を VPNv4 ルートに変換する。
- マルチプロトコルボーダーゲートウェイプロトコル (MP-BGP) を介して、他の PE デバイスと VPNv4 ルートを交換する。

ここでは、MPLS VPN の機能について説明します。

MPLS バーチャル プライベート ネットワークの主要コンポーネント

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

- VPN ルートターゲットコミュニティ : VPN ルートターゲットコミュニティは、VPN コミュニティのすべてのメンバのリストです。VPN ルートターゲットは、各 VPN コミュニティメンバに設定する必要があります。
- VPN コミュニティプロバイダーエッジ (PE) デバイスのマルチプロトコル BGP (MP-BGP) ピアリング : MP-BGP は、VPN コミュニティのすべてのメンバーに Virtual

Route Forwarding (VRF) 到達可能性情報を伝播します。MP-BGP ピアリングは、VPN コミュニティのすべての PE デバイスで設定されている必要があります。

- MPLS 転送：MPLS は、VPN サービス プロバイダー ネットワーク上のすべての VPN コミュニティ メンバ間のすべてのトラフィックを転送します。

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

MPLS バーチャル プライベート ネットワークの利点

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

コネクションレス型サービス

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

集中型サービス

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

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

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

拡張性

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

MPLS VPN のその他の拡張性の問題は、PE デバイス間の VPN ルートのパーティショニングに起因します。また、コアネットワークでの PE デバイスとプロバイダー (P) デバイス間での VPN ルートおよび内部ゲートウェイプロトコル (IGP) ルートのさらなるパーティショニングに起因します。

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

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

セキュリティ

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

セキュリティは、次の領域で提供されます。

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

作成の容易さ

VPN を最大限に活用するには、カスタマーは、新しい VPN とユーザーコミュニティを簡単に作成する必要があります。MPLS VPN はコネクションレスであるため、特定のポイントツーポイント接続マップまたはトポロジは必要ありません。イントラネットやエクストラネットに

サイトを追加して、非公開ユーザー グループを形成できます。この方法でVPNを管理すると、指定された任意のサイトを複数のVPNのメンバにできるため、イントラネットやエクストラネットを構築する場合の柔軟性が最大限に高められます。

柔軟なアドレッシング

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

統合 QoS サポート

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

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

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

直接的な移行

サービスプロバイダーは、VPNサービスを迅速に展開するために、直接的な移行パスを使用します。MPLS VPN の独自の長所として、IP、ATM、フレームリレー、およびハイブリッドネットワークを含む、複数のネットワークアーキテクチャ上に構築できることを挙げることができます。

CE デバイス上で MPLS をサポートする必要がないため、エンドカスタマーの移行作業は簡単になります。お客様のイントラネットを変更する必要はありません。

MPLS バーチャル プライベート ネットワークの設定方法

次の項では、MPLS バーチャルプライベート ネットワークを設定する手順について説明します。

コア ネットワークの設定

次の項では、コア ネットワークを設定する手順について説明します。

MPLS バーチャル プライベート ネットワーク カスタマーのニーズの評価

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

手順の概要

1. ネットワークのサイズを識別します。
2. コアにおけるルーティング プロトコルを識別します。
3. MPLS VPN ハイ アベイラビリティのサポートが必要であるかどうかを判断します。
4. MPLS VPN コアで Border Gateway Protocol (BGP) ロードシェアリングおよび冗長パスが必要であるかどうかを決定します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	ネットワークのサイズを識別します。	必要となるデバイスとポートの数を決定するために、次の内容を識別します。 <ul style="list-style-type: none"> • サポートする必要があるカスタマーの数 • カスタマーごとに必要となる VPN の数 • 各 VPN に存在する、仮想ルーティングおよび転送インスタンスの数
ステップ 2	コアにおけるルーティング プロトコルを識別します。	コア ネットワークで必要なルーティング プロトコルを決定します。
ステップ 3	MPLS VPN ハイ アベイラビリティのサポートが必要であるかどうかを判断します。	MPLS VPN ノンストップフォワーディングおよびグレースフルリスタートは、選択デバイスおよび Cisco IOS ソフトウェア リリースでサポートされています。Cisco サポートに問い合わせ、正確な要件およびハードウェア サポートを確認してください。
ステップ 4	MPLS VPN コアで Border Gateway Protocol (BGP) ロードシェアリングおよび冗長パスが必要であるかどうかを決定します。	設定手順については、『 <i>MPLS Layer 3 VPNs Inter-AS and CSC Configuration Guide</i> 』の「Load Sharing MPLS VPN Traffic」モジュールを参照してください。

コアにおける MPLS の設定

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

- MPLS ラベル配布プロトコル (LDP)。設定については、『*MPLS Label Distribution Protocol Configuration Guide*』の「MPLS Label Distribution Protocol (LDP)」モジュールを参照してください。

MPLS バーチャルプライベート ネットワーク カスタマーの接続

次の項では、MPLS バーチャルプライベート ネットワーク カスタマーの接続について説明します。

カスタマーの接続を可能にするための、PE デバイスでの VRF の定義

次の手順を使用して、IPv4 の仮想ルーティングおよび転送 (VRF) 設定を定義します。IPv4 と IPv6 の VRF を定義するには、MPLS レイヤ 3 VPN コンフィギュレーションガイド [英語] の「IPv6 VPN over MPLS」モジュールの「Configuring a Virtual Routing and Forwarding Instance for IPv6」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **vrf definition vrf-name**
4. **rd route-distinguisher**
5. **address-family ipv4 | ipv6**
6. **route-target {import | export | both} route-target-ext-community**
7. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vrf definition vrf-name 例 : Device(config)# vrf definition vrf1	バーチャルプライベートネットワーク (VRF) 名を割り当て、VRF コンフィギュレーションモードを開始することにより、Virtual Routing and Forwarding (VPN) ルーティングインスタンスを定義します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>vrf-name</i> 引数は、VRF に割り当てる名前です。
ステップ 4	rd route-distinguisher 例 : Device(config-vrf)# rd 100:1	ルーティング テーブルと転送テーブルを作成します。 <ul style="list-style-type: none"> • <i>route-distinguisher</i> 引数によって、8 バイトの値が IPv4 プレフィックスに追加され、VPN IPv4 プレフィックスが作成されます。ルート識別子 (RD) は、次のいずれかの形式で入力できます。 <ul style="list-style-type: none"> • 16 ビットの AS 番号 : 32 ビットの番号。 101:3 など。 • 32 ビットの IP アドレス : 16 ビットの番号。 10.0.0.1:1 など。
ステップ 5	address-family ipv4 ipv6 例 : Device(config-vrf)# address-family ipv6	IPv4 または IPv6 アドレスファミリモードを開始します。
ステップ 6	route-target {import export both} <i>route-target-ext-community</i> 例 : Device(config-vrf-af)# route-target both 100:1	VRF 用にルートターゲット拡張コミュニティを作成します。 <ul style="list-style-type: none"> • import キーワードを使用すると、ターゲット VPN 拡張コミュニティからルーティング情報がインポートされます。 • export キーワードを使用すると、ルーティング情報がターゲット VPN 拡張コミュニティにエクスポートされます。 • both キーワードを使用すると、ターゲット VPN 拡張コミュニティとの間でルーティング情報がインポートおよびエクスポートされます。 • <i>route-target-ext-community</i> 引数により、<i>route-target</i> 拡張コミュニティ属性が、インポートやエクスポートの <i>route-target</i> 拡張コミュニティの VRF リストに追加されます。
ステップ 7	exit 例 : Device(config-vrf)# exit	(任意) 終了して、グローバル コンフィギュレーション モードに戻ります。

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

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

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **vrf forwarding** *vrf-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>type number</i> 例 : Device(config)# interface GigabitEthernet 0/0/1	設定するインターフェイスを指定し、インターフェイス コンフィギュレーションモードを開始します。 • <i>type</i> 引数で、設定するインターフェイスのタイプを指定します。 • <i>number</i> 引数には、ポート、コネクタ、またはインターフェイス カード番号を指定します。
ステップ 4	vrf forwarding <i>vrf-name</i> 例 : Device(config-if)# vrf forwarding vrf1	指定したインターフェイスまたはサブインターフェイスに VRF を関連付けます。 • <i>vrf-name</i> 引数は、VRF に割り当てる名前です。
ステップ 5	end 例 : Device(config-if)# end	(任意) 終了して、特権 EXEC モードに戻ります。

PE デバイスと CE デバイス間でのルーティング プロトコルの設定

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

バーチャル プライベート ネットワーク の設定の確認

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

手順の概要

1. show ip vrf

手順の詳細

show ip vrf

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

MPLS バーチャル プライベート ネットワーク サイト間の接続の確認

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

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

手順の概要

1. **enable**
2. **ping** [protocol] {host-name | system-address}
3. **trace** [protocol] [destination]
4. **show ip route** [ip-address [mask] [longer-prefixes]] | protocol [process-id]] | [list [access-list-name | access-list-number]

手順の詳細

ステップ 1 enable

特権 EXEC モードをイネーブルにします。

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

ステップ 2 ping [protocol] {host-name | system-address}

AppleTalk、コネクションレス型モード ネットワーク サービス (CLNS)、IP、Novell、Apollo、Virtual Integrated Network Service (VINES)、DECnet、または Xerox Network Service (XNS) ネットワークでの基本的なネットワーク接続を診断します。ping コマンドを使用して、CE デバイス間の接続を確認します。

ステップ 3 trace [protocol] [destination]

パケットがその宛先に送信される時に取るルートを検出します。trace コマンドは、2 つのデバイスが通信できない場合に問題の箇所を分離するのに役立ちます。

ステップ 4 show ip route [ip-address [mask] [longer-prefixes]] | protocol [process-id] | [list [access-list-name | access-list-number]

ルーティング テーブルの現在の状態を表示します。ip-address 引数を使用して、CE1 に CE2 へのルートが含まれていることを確認します。CE1 から学習したルートを確認します。CE2 へのルートがリストされていることを確認します。

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

手順の概要

1. enable
2. show ip route vrf vrf-name [prefix]
3. show ip cef vrf vrf-name [ip-prefix]

手順の詳細

ステップ 1 enable

特権 EXEC モードをイネーブルにします。

ステップ 2 show ip route vrf vrf-name [prefix]

Virtual Route Forwarding (VRF) インスタンスに関連付けられている IP ルーティングテーブルを表示します。ローカル カスタマー エッジ (CE) デバイスとリモート カスタマー エッジ (CE) デバイスのループバック アドレスが、プロバイダー エッジ (PE) でデバイスのルーティング テーブルに存在することを確認します。

ステップ 3 show ip cef vrf vrf-name [ip-prefix]

VRF に関連付けられている Cisco Express Forwarding 転送テーブルを表示します。次のように、リモート CE デバイスのプレフィックスが、シスコ エクスプレス フォワーディング テーブルに存在することを確認します。

MPLS バーチャルプライベートネットワーク (VPN) の設定例

次の項では、MPLS バーチャルプライベートネットワークを設定する手順について説明します。

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

PE の設定	CE の設定
<pre> vrf vpn1 rd 100:1 route-target export 100:1 route-target import 100:1 ! ip cef mpls ldp router-id Loopback0 force mpls label protocol ldp ! interface Loopback0 ip address 10.0.0.1 255.255.255.255 ! interface GigabitEthernet 1/0/1 vrf forwarding vpn1 ip address 192.0.2.3 255.255.255.0 no cdp enable interface GigabitEthernet 1/0/1 ip address 192.0.2.2 255.255.255.0 mpls label protocol ldp mpls ip ! router rip version 2 timers basic 30 60 60 120 ! address-family ipv4 vrf vpn1 version 2 redistribute bgp 100 metric transparent network 192.0.2.0 distribute-list 20 in no auto-summary exit-address-family ! router bgp 100 no synchronization bgp log-neighbor changes neighbor 10.0.0.3 remote-as 100 neighbor 10.0.0.3 update-source Loopback0 no auto-summary ! address-family vpnv4 neighbor 10.0.0.3 activate neighbor 10.0.0.3 send-community extended bgp scan-time import 5 exit-address-family ! address-family ipv4 vrf vpn1 redistribute connected redistribute rip no auto-summary no synchronization exit-address-family </pre>	<pre> ip cef mpls ldp router-id Loopback0 force mpls label protocol ldp ! interface Loopback0 ip address 10.0.0.9 255.255.255.255 ! interface GigabitEthernet 1/0/1 ip address 192.0.2.1 255.255.255.0 no cdp enable router rip version 2 timers basic 30 60 60 120 redistribute connected network 10.0.0.0 network 192.0.2.0 no auto-summary </pre>

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

PE の設定	CE の設定
<pre> vrf vpn1 rd 100:1 route-target export 100:1 route-target import 100:1 ! ip cef mpls ldp router-id Loopback0 force mpls label protocol ldp ! interface Loopback0 ip address 10.0.0.1 255.255.255.255 ! interface GigabitEthernet 1/0/1 vrf forwarding vpn1 ip address 192.0.2.3 255.255.255.0 no cdp enable ! interface GigabitEthernet 1/0/1 ip address 192.168.0.1 255.255.0.0 mpls label protocol ldp mpls ip ! router ospf 100 network 10.0.0. 0.0.0.0 area 100 network 192.168.0.0 255.255.0.0 area 100 ! router bgp 100 no synchronization bgp log-neighbor changes neighbor 10.0.0.3 remote-as 100 neighbor 10.0.0.3 update-source Loopback0 no auto-summary ! address-family vpnv4 neighbor 10.0.0.3 activate neighbor 10.0.0.3 send-community extended bgp scan-time import 5 exit-address-family ! address-family ipv4 vrf vpn1 redistribute connected redistribute static no auto-summary no synchronization exit-address-family ! ip route vrf vpn1 10.0.0.9 255.255.255.255 192.0.2.2 ip route vrf vpn1 192.0.2.0 255.255.0.0 192.0.2.2 </pre>	<pre> ip cef ! interface Loopback0 ip address 10.0.0.9 255.255.255.255 ! interface GigabitEthernet 1/0/1 ip address 192.0.2.2 255.255.0.0 no cdp enable ! ip route 10.0.0.9 255.255.255.255 192.0.2.3 3 ip route 198.51.100.0 255.255.255.0 192.0.2.3 3 </pre>

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

PE の設定	CE の設定
	<pre> router bgp 5000 bgp log-neighbor-changes neighbor 5.5.5.6 remote-as 5001 neighbor 5.5.5.6 ebgp-multihop 2 neighbor 5.5.5.6 update-source Loopback5 neighbor 35.2.2.2 remote-as 5001 neighbor 35.2.2.2 ebgp-multihop 2 neighbor 35.2.2.2 update-source Loopback1 neighbor 3500::1 remote-as 5001 neighbor 3500::1 ebgp-multihop 2 neighbor 3500::1 update-source Loopback1 ! address-family ipv4 redistribute connected neighbor 5.5.5.6 activate neighbor 35.2.2.2 activate no neighbor 3500::1 activate exit-address-family ! address-family ipv6 redistribute connected neighbor 3500::1 activate exit-address-family Device-RP(config)# </pre>

PE の設定	CE の設定
<pre> router bgp 5001 bgp log-neighbor-changes bgp graceful-restart bgp sso route-refresh-enable bgp refresh max-eor-time 600 redistribute connected neighbor 102.1.1.1 remote-as 5001 neighbor 102.1.1.1 update-source Loopback1 neighbor 105.1.1.1 remote-as 5001 neighbor 105.1.1.1 update-source Loopback10 neighbor 160.1.1.2 remote-as 5002 ! address-family vpnv4 neighbor 102.1.1.1 activate neighbor 102.1.1.1 send-community both neighbor 105.1.1.1 activate neighbor 105.1.1.1 send-community extended exit-address-family ! address-family vpnv6 neighbor 102.1.1.1 activate neighbor 102.1.1.1 send-community extended neighbor 105.1.1.1 activate neighbor 105.1.1.1 send-community extended exit-address-family ! address-family ipv4 vrf full redistribute connected neighbor 20.1.1.1 remote-as 5000 neighbor 20.1.1.1 ebgp-multihop 2 neighbor 20.1.1.1 update-source Loopback2 neighbor 20.1.1.1 activate neighbor 20.1.1.1 send-community both exit-address-family ! address-family ipv6 vrf full redistribute connected neighbor 2000::1 remote-as 5000 neighbor 2000::1 ebgp-multihop 2 neighbor 2000::1 update-source Loopback2 neighbor 2000::1 activate exit-address-family ! address-family ipv4 vrf orange network 87.1.0.0 mask 255.255.252.0 network 87.1.1.0 mask 255.255.255.0 redistribute connected neighbor 40.1.1.1 remote-as 7000 neighbor 40.1.1.1 ebgp-multihop 2 neighbor 40.1.1.1 update-source Loopback3 neighbor 40.1.1.1 activate neighbor 40.1.1.1 send-community extended neighbor 40.1.1.1 route-map orange-lp in maximum-paths eibgp 2 exit-address-family ! address-family ipv6 vrf orange redistribute connected maximum-paths eibgp 2 neighbor 4000::1 remote-as 7000 neighbor 4000::1 ebgp-multihop 2 </pre>	

PE の設定	CE の設定
<pre>neighbor 4000::1 update-source Loopback3 neighbor 4000::1 activate exit-address-family ! address-family ipv4 vrf sona redistribute connected neighbor 160.1.1.2 remote-as 5002 neighbor 160.1.1.2 activate neighbor 160.1.1.4 remote-as 5003 neighbor 160.1.1.4 activate exit-address-family</pre>	

その他の参考資料

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	『 <i>Command Reference (Catalyst 9300 Series Switches)</i> 』の「MPLS コマンド」の項を参照してください。
Cisco Express Forwarding の設定	『 <i>Cisco Express Forwarding Configuration Guide</i> 』の「Configuring Basic Cisco Express Forwarding」モジュール
LDP の設定	『 <i>MPLS Label Distribution Protocol Configuration Guide</i> 』の「MPLS Label Distribution Protocol (LDP)」モジュール

MPLS バーチャルプライベートネットワークの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	MPLS バーチャル プライベート ネットワーク	MPLS バーチャル プライベート ネットワーク (VPN) は、マルチプロトコル ラベル スイッチング (MPLS) プロバイダー コア ネットワークによって相互接続された一連のサイトで構成されます。各カスタマー サイトでは、1 つ以上のカスタマー エッジ (CE) デバイスが、1 つ以上のプロバイダー エッジ (PE) デバイスに接続されます。
Cisco IOS XE Gibraltar 16.11.1	MPLS レイヤ 3 VPN の BGP PE-CE サポート	プロバイダー エッジ (PE) デバイスとカスタマー エッジ (CE) デバイス間のルーティングプロトコルとしての BGP のサポートが導入されました。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 3 章

eBGP および iBGP マルチパスの設定

- MPLS-VPN における eBGP および iBGP に対する BGP マルチパス ロードシェアリング (33 ページ)
- MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングについて (34 ページ)
- MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの設定方法 (36 ページ)
- MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリング機能の設定例 (38 ページ)
- MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの機能情報 (39 ページ)

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

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

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

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

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

アドレス ファミリのサポート

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

メモリ消費の制約事項

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

パス数の制限

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

サポートされていないコマンド

`ip unnumbered` コマンドは MPLS 設定ではサポートされていません。

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

eBGP と iBGP 間のマルチパス ロードシェアリング

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



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

マルチパス全体でのロードバランシングは CEF によって実行されます。CEF ロードバランシングは、パケット単位のラウンドロビンまたはセッション単位 (送信元と宛先のペア) を基準として設定されます。CEF の設定の詳細については、[Cisco IOS IP スイッチング コンフィギュレーション ガイド \[英語\]](#)、[IP スイッチング Cisco Express Forwarding コンフィギュレーション](#)

ガイド [英語] を参照してください。MPLS VPN における eBGP および iBGP に対する BGP マルチパス ロードシェアリング機能は、IPv4 VRF アドレスファミリーおよび IPv6 VRF アドレスファミリー コンフィギュレーションモードで有効になります。この機能が有効にされると、VRF にインポートされた eBGP パスまたは iBGP パスあるいはその両方でロードバランシングを実行できます。マルチパスの数は VRF 単位で設定されます。別々の VRF マルチパス設定は、固有ルート識別子によって分離されます。

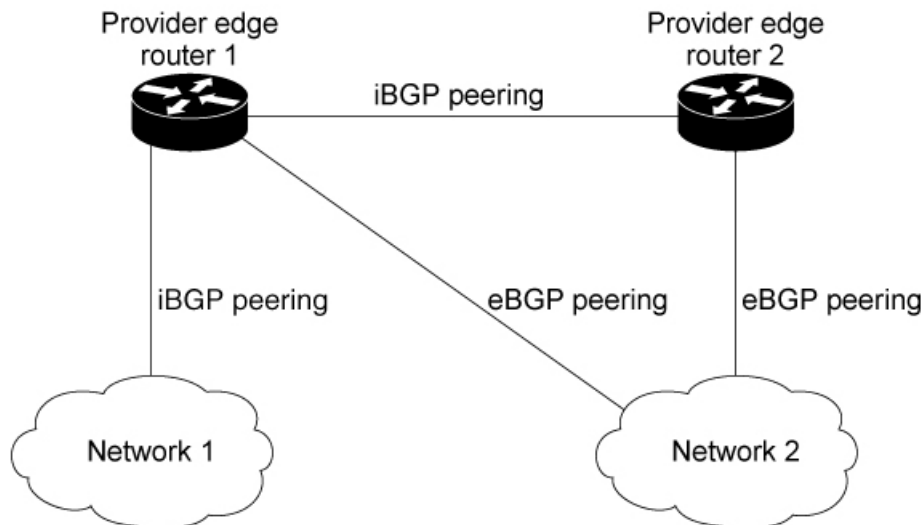


(注) MPLS VPN における eBGP および iBGP に対する BGP マルチパス ロードシェアリング機能は、設定されたアウトバウンドルーティング ポリシーのパラメータの範囲内で動作します。

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

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

図 2: サービスプロバイダー BGP MPLS ネットワーク



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

355811

では、PE ルータ 1 が eBGP パスを使用してロードシェアリングします。これは、IP トラフィックと iBGP パスが MPLS トラフィックとして送信されるためです。



- (注)
- ローカル CE とローカル PE 間の eBGP セッションはサポートされていません。
 - ローカル PE からリモート CE への eBGP セッションはサポートされています。
 - eiBGP マルチパスは、プレフィックス単位のラベル割り当てモードでのみサポートされません。他のラベル割り当てモードではサポートされません。

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

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

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

ここでは、次の手順について説明します。

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

手順の概要

1. **enable**
2. **configure { terminal | memory | network }**
3. **router bgp as-number**
4. **neighbor { ip-address | ipv6-address | peer-group-name }**
5. **address-family ipv4 vrfvrf-name**
6. **address-family ipv6 vrfvrf-name**
7. **neighbor { ip-address | ipv6-address | peer-group-name } update-source interface-type interface-name**
8. **neighbor { ip-address | ipv6-address | peer-group-name } activate**
9. **maximum-paths eibgp [import-number]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure { terminal memory network } 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp as-number 例： Device(config)# router bgp 40000	ルータ コンフィギュレーションモードを開始して、BGP ルーティング プロセスを作成または設定します。
ステップ 4	neighbor { ip-address ipv6-address peer-group-name } 例： Device(config-router)# neighbor group192	直接接続されていないネットワーク上の外部ピアからの BGP 接続を受け入れ、またそのピアへの BGP 接続を試みます。
ステップ 5	address-family ipv4 vrfvrf-name 例： Device(config-router)# address-family ipv4 vrf RED	ルータをアドレス ファミリ コンフィギュレーション モードにします。 • 別々の VRF マルチパス設定は、固有ルート識別子によって分離されます。
ステップ 6	address-family ipv6 vrfvrf-name 例： Device(config-router)# address-family ipv6 vrf RED	ルータをアドレス ファミリ コンフィギュレーション モードにします。 • 別々の VRF マルチパス設定は、固有ルート識別子によって分離されます。
ステップ 7	neighbor { ip-address ipv6-address peer-group-name } update-source interface-type interface-name 例： Device(config-router)# neighbor FE80::1234:BFF:FE0E:A471 update-source Gigabitethernet 1/0/0	ピアリングが発生するリンクローカルアドレスを指定します。
ステップ 8	neighbor { ip-address ipv6-address peer-group-name } activate 例： (config-router)# neighbor group192 activate	設定されたアドレスファミリに対してネイバーまたは受信範囲ピア グループをアクティブにします。
ステップ 9	maximum-paths eibgp [import-number] 例： (config-router-af)# maximum-paths eibgp 2	ルーティングテーブルにインストールできるパラレルの iBGP ルートおよび eBGP ルートの数を設定します。

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

手順の概要

1. **enable**
2. **show ip bgp neighbors**
3. **show ip bgp vpnv4 vrfvrf name**
4. **show ip route vrfvrf-name**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show ip bgp neighbors 例： Device# show ip bgp neighbors	ネイバーへの TCP 接続および BGP 接続についての情報を表示します。
ステップ 3	show ip bgp vpnv4 vrfvrf name 例： Device# show ip bgp vpnv4 vrf RED	VPN アドレス情報を BGP テーブルから表示します。このコマンドは、VRF が BGP によって受信されたことを確認するために使用します。
ステップ 4	show ip route vrfvrf-name 例： Device# show ip route vrf RED	VRF インスタンスに関連する IP ルーティングテーブルを表示します。show ip route vrf コマンドは、該当する VRF がルーティングテーブルにあることを確認するために使用します。

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

次に、この機能の設定方法および確認方法の例を示します。

eBGP および iBGP のマルチパス ロードシェアリングの設定例

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

```
Device(config)# router bgp 40000
Device(config-router)# address-family ipv4 vrf RED
```

```
Device(config-router-af)# maximum-paths eibgp 2
Device(config-router-af)# end
```

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

```
Device(config)#router bgp 40000
Device(config-router)# address-family ipv6 vrf RED
Device(config-router-af)# maximum-paths eibgp 2
Device(config-router-af)# end
```

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

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

表 1: MPLS-VPN における eBGP および iBGP の両方に対する BGP マルチパス ロードシェアリングの機能情報

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



第 4 章

EIGRP MPLS VPN PE-CE Site of Origin の設定

- [EIGRP MPLS VPN PE-CE Site of Origin \(41 ページ\)](#)
- [EIGRP MPLS VPN PE-CE Site of Origin について \(42 ページ\)](#)
- [EIGRP MPLS VPN PE-CE Site of Origin サポートの設定方法 \(44 ページ\)](#)
- [EIGRP MPLS VPN PE-CE SoO の設定例 \(47 ページ\)](#)
- [EIGRP MPLS VPN PE-CE Site of Origin の機能履歴 \(49 ページ\)](#)

EIGRP MPLS VPN PE-CE Site of Origin

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

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

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

- この機能は、PE と CE 間の EIGRP に対する MPLS VPN Support 機能をサポートするために導入されており、この機能は、EIGRP MPLS VPN の作成後に設定する必要があります。
- EIGRP MPLS VPN 対応に設定されているすべての PE ルータは、SoO の拡張コミュニティをサポートする Cisco IOS XE Gibraltar 16.11.1 以降のリリースを実行している必要があります。

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

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

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

ここでは、EIGRP MPLS VPN PE-CE Site of Origin について説明します。

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

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

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

バックドア リンクに対する Site of Origin のサポート

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

VPN リンクの障害がない場合はバックドア ルータを介したルートが選択されないように、メトリックはバックドア リンク上に設定されます。

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

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

Site of Origin 拡張コミュニティとルータとの相互運用

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

- BGP または CE ルータから受信したルートには、受信側インターフェイス上の SoO 値と一致する SoO 値が含まれている場合：受信側インターフェイス上に設定されている SoO 値と一致する関連 SoO 値とともにルートを受信した場合、そのルートは別の PE ルータまたはバックドアリンクから学習したルートであるため、フィルタリングされます。この動作は、ルーティング ループを回避するために設計されています。
- CE ルータから受信したルートが一致しない SoO 値で設定されている場合：あるルートが、関連付けられている SoO 値とともに受信され、その値が、受信インターフェイス上で設定されている SoO 値と一致しない場合、そのルートは、BGP へ再配布されるように EIGRP トポロジテーブルに追加されます。ルートがすでに EIGRP トポロジテーブルにインストールされているが、別の SoO 値と関連付けられている場合は、そのルートが BGP へ再配布されるときに、トポロジテーブルの SoO 値が使用されます。
- CE ルータから受信したルートに SoO 値が含まれていない場合：受信したルートに SoO 値がない場合、そのルートは EIGRP トポロジテーブルに受け入れられます。ルートが BGP へ再配布される前に、ネクストホップ CE ルータに到達するために使用されるインターフェイスの SoO 値がそのルートに付加されます。

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

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

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

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

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

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

ここでは、EIGRP MPLS VPN PE-CE Site of Origin サポートの設定方法について説明します。

Site of Origin 拡張コミュニティの設定

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

始める前に

- ネットワークコア (またはサービス プロバイダー バックボーン) にボーダー ゲートウェイ プロトコル (BGP) が設定されていることを確認する。
- この機能を設定する前に、EIGRP MPLS VPN を設定する。

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

手順の概要

1. **enable**
2. **configure terminal**
3. **route-map map-name {permit|deny} [sequence-number]**
4. **set extcommunity sooextended-community-value**
5. **exit**
6. **interface type number**
7. **no switchport**
8. **vrf forwarding vrf-name**
9. **ip vrf sitemap route-map-name**
10. **ip address ip-address subnet-mask**
11. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	route-map map-name { permit deny } [<i>sequence-number</i>] 例： Device(config)# route-map Site-of-Origin permit 10	ルートマップ コンフィギュレーション モードを開始して、ルート マップを作成します。 • この手順でルート マップが作成され、SoO 拡張コミュニティが適用されるようになります。
ステップ 4	set extcommunity sooextended-community-value 例： Device(config-route-map)# set extcommunity soo 100:1	BGP 拡張コミュニティ属性を設定します。 • soo キーワードには、Site of Origin 拡張コミュニティ属性を指定します。 • extended-community-value 引数には、設定する値を指定します。この値では、次のいずれかの形式を使用できます。 • 自律システム番号: ネットワーク番号

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • IP アドレス: ネットワーク番号 <p>自律システム番号とネットワーク番号、または IP アドレスとネットワーク番号の区切りにはコロンを使用します。</p>
ステップ 5	exit 例 : Device(config-route-map)# exit	ルートマップ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 6	interface type number 例 : Device(config)# interface GigabitEthernet 1/0/1	特定のインターフェイスを設定するため、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	no switchport 例 : Device(config-if)# no switchport	インターフェイスをレイヤ2ポートとして動作することを停止し、シスコルーテッド (レイヤ3) ポートにします。
ステップ 8	vrf forwarding vrf-name 例 : Device(config-if)# vrf forwarding VRF1	<p>VRF をインターフェイスまたはサブインターフェイスに関連付けます。</p> <ul style="list-style-type: none"> • この手順で設定された VRF 名は、プロバイダー エッジとカスタマー エッジ間の EIGRP に対する MPLS VPN Support 機能を備えた EIGRP MPLS VPN に対して作成された VRF 名と一致している必要があります。
ステップ 9	ip vrf sitemap route-map-name 例 : Device(config-if)# ip vrf sitemap Site-of-Origin	<p>VRF をインターフェイスまたはサブインターフェイスに関連付けます。</p> <ul style="list-style-type: none"> • この手順で設定されたルート マップ名は、手順 3 で、SoO 拡張コミュニティを適用するために作成されたルート マップ名と一致している必要があります。
ステップ 10	ip address ip-address subnet-mask 例 : Device(config-if)# ip address 10.0.0.1 255.255.255.255	<p>インターフェイスの IP アドレスを設定します。</p> <ul style="list-style-type: none"> • IP アドレスは、VRF フォワーディングをイネーブルにした後で再設定する必要があります。
ステップ 11	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

次のタスク

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

SoO 拡張コミュニティの設定の確認

手順の概要

- enable
- show ip bgp vpnv4 {all | rdroute-distinguisher | vrfvrf-name} [ip-prefix/length]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show ip bgp vpnv4 {all rdroute-distinguisher vrfvrf-name} [ip-prefix/length] 例： Device# ip bgp vpnv4 vrf SOO-1 20.2.1.1/32	VPN アドレス情報を BGP テーブルから表示します。 • show ip bgp vpnv4 コマンドと all キーワードを使用して、指定したルートが、SoO 拡張コミュニティ属性で設定されていることを検証します。

EIGRP MPLS VPN PE-CE SoO の設定例

ここでは、EIGRP MPLS VPN PE-CE SoO の設定例を紹介します。

Site of Origin 拡張コミュニティの設定例

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

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

Site of Origin 拡張コミュニティの確認の例

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

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

カスタマー エッジ デバイス show コマンド

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

プロバイダー エッジ デバイス show コマンド

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

EIGRP MPLS VPN PE-CE Site of Origin の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	EIGRP MPLS VPN PE-CE Site of Origin	EIGRP MPLS VPN PE-CE Site of Origin 機能によって、マルチプロトコルラベルスイッチング (MPLS) バーチャルプライベート ネットワーク (VPN) トラフィックを、Enhanced Interior Gateway Routing Protocol (EIGRP) ネットワークに対してサイト単位でフィルタリングする機能が追加されます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 5 章

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

- [Ethernet-over-MPLS の設定 \(51 ページ\)](#)
- [疑似回線冗長性の設定 \(68 ページ\)](#)
- [Ethernet-over-MPLS および疑似回線冗長性の機能情報 \(85 ページ\)](#)

Ethernet-over-MPLS の設定

ここでは、Ethernet over Multiprotocol Label Switching (EoMPLS) の設定方法について説明します。

Ethernet-over-MPLS の前提条件

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

- プロバイダーエッジ (PE) デバイスが IP によって相互に到達できるように、コアに IP ルーティングを設定します。
- PE デバイス間にラベルスイッチパス (LSP) が存在するように、コアに MPLS を設定します。
- 接続回線で Xconnect を設定する前に、**no switchport**、**no keepalive**、および **no ip address** コマンドを設定します。
- ロードバランシングの場合、**port-channel load-balance** コマンドの設定は必須です。
- EoMPLS VLAN モードを有効にするには、サブインターフェイスがサポートされている必要があります。

Ethernet-over-MPLS の制約事項

次の項では、EoMPLS ポートモードおよび EoMPLS VLAN モードの制約事項を示します。

Ethernet-over-MPLS ポートモードの制約事項

- イーサネット フロー ポイントはサポートされていません。
- Quality of Service (QoS) : お客様の Differentiated Services Code Point (DSCP; DiffServ コードポイント) の再マーキングは、Virtual Private Wire Service (VPWS) および EoMPLS ではサポートされません。
- 明示的 null の仮想回線接続検証 (VCCV) ping はサポートされていません。
- レイヤ 2 プロトコルトンネリング CLI はサポートされていません。
- Flow Aware Transport (FAT) 疑似回線冗長性は、プロトコル CLI モードでのみサポートされています。サポートされているロードバランシングパラメータは、送信元 IP、送信元 MAC アドレス、宛先 IP、および宛先 MAC アドレスです。
- MPLS QoS は、パイプモードと均一モードでのみサポートされています。デフォルトモードはパイプモードです。
- レガシー Xconnect モードとプロトコル CLI (インターフェイス疑似回線設定) モードはどちらもサポートされています。
- Xconnect と MACSec を同じインターフェイスに設定することはできません。
- MACSec は CE デバイスで設定し、Xconnect は PE デバイスで設定する必要があります。
- CE デバイス間で MACSec セッションを使用できる必要があります。
- デフォルトでは、EoMPLS PW は Cisco Discovery Protocol やスパニングツリープロトコル (STP) などのすべてのプロトコルをトンネリングします。EoMPLS PW は L2 プロトコル トンネリング CLI の一環として選択的なプロトコル トンネリングを実行できません。
- Link Aggregation Control Protocol (LACP) および Port Aggregation Protocol (PAgP) パケットは、ローカル PE によって処理されるため、Ethernet-over-MPLS 疑似回線を介して転送されません。

EoMPLS VLAN モードの制約事項

- 各 PE デバイスで同じインターワーキングタイプが設定されていない場合、仮想回線は機能しません。
- タグなしトラフィックは、着信トラフィックとしてはサポートされません。
- マルチプレクサユーザーネットワーク インターフェイス (MUX UNI) がサポートされていないため、レイヤ 2 サブインターフェイスでは Xconnect モードを有効にできません。
- Xconnect モードは、ポート間トランスポートのメインインターフェイスで有効になっている場合、サブインターフェイスには設定できません。

- FAT は、プロトコル CLI モードでのみ設定できます。
- VLAN モード EoMPLS では、CE デバイスによってクリアされた dot1q で暗号化されたパケットのみが PE デバイスによって処理されます。
- QoS : カスタマー DSCP 再マーキングは VPWS と EoMPLS ではサポートされていません。
- MPLS QoS は、パイプモードと均一モードでサポートされています。デフォルトモードはパイプモードです。
- VLAN モードの EoMPLS では、CE からの Cisco Discovery Protocol パケットは PE で処理されますが、EoMPLS 仮想回線では伝送されません。一方、ポートモードでは、CE からの Cisco Discovery Protocol パケットは仮想回線で伝送されます。
- イーサネットおよび VLAN インターワーキングタイプのみがサポートされています。
- L2 プロトコル トンネリング CLI はサポートされていません。
- Link Aggregation Control Protocol (LACP) および Port Aggregation Protocol (PAgP) パケットは、ローカル PE によって処理されるため、Ethernet-over-MPLS 疑似回線を介して転送されません。

Ethernet-over-MPLS に関する情報

EoMPLS は、Any Transport over MPLS (AToM) トランスポートタイプの 1 つです。EoMPLS は、イーサネットプロトコルデータユニット (PDU) を MPLS パケットにカプセル化し、MPLS ネットワーク上で転送することにより機能します。各 PDU は単一パケットとして転送されます。

次のモードがサポートされています。

- ポートモード : ポートのすべてのトラフィックが MPLS ネットワーク上の単一の仮想回線を共有できるようにします。ポートモードは仮想回線タイプ 5 を使用します。
- VLAN モード : MPLS ネットワーク上の単一の仮想回線を介して、送信元 802.1Q VLAN から宛先 802.1Q VLAN にイーサネットトラフィックを転送します。VLAN モードは仮想回線タイプ 5 をデフォルトとして使用します (dot1q タグは転送されません)。ただし、リモート PE がサブインターフェイスベース (VLAN ベース) の EoMPLS の仮想回線タイプ 5 をサポートしていない場合は、仮想回線タイプ 4 (dot1 タグを転送) を使用します。

EoMPLS ポートモードと EoMPLS VLAN モード間のインターワーキング : EoMPLS ポートモードがローカル PE で設定され、EoMPLS VLAN モードがリモート PE で設定されている場合、カスタマーエッジ (CE) レイヤ 2 スイッチポートインターフェイスは、ポートモード側で access として設定する必要があります。また、スパンニングツリープロトコルは、CE デバイスの VLAN モード側で無効にする必要があります。

PE 間のすべての中間リンクの最大伝送ユニット (MTU) が、入力 PE で受信される最大のレイヤ 2 パケットを伝達できる必要があります。

Ethernet-over-MPLS の設定方法

EoMPLS は、ポートモードまたは VLAN モードで設定できます。

Ethernet-over-MPLS ポートモードの設定

EoMPLS ポートモードは、Xconnect モードまたはプロトコル CLI 方式のいずれかを使用して設定できます。

Xconnect モード

Xconnect モードで EoMPLS ポートモードを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **no switchport**
5. **no ip address**
6. **no keepalive**
7. **xconnect peer-device-id vc-id encapsulation mpls**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device(config)# interface TenGigabitEthernet1/0/36	トランクとして設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	no switchport 例： Device(config-if) # no switchport	物理ポートに限り、レイヤ 3 モードを開始します。
ステップ 5	no ip address 例： Device(config-if) # no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	no keepalive 例： Device(config-if) # no keepalive	デバイスがキープアライブメッセージを送信しないことを確認します。
ステップ 7	xconnect peer-device-id vc-id encapsulation mpls 例： Device(config-if) # xconnect 10.1.1.1 962 encapsulation mpls	接続回線を疑似回線仮想回線 (VC) にバインドします。このコマンドの構文は、その他のレイヤ 2 トランスポートの場合と同じです。
ステップ 8	end 例： Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

プロトコル CLI 方式

プロトコル CLI モードで EoMPLS ポートモードを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **port-channel load-balance dst-ip**
4. **interface interface-id**
5. **no switchport**
6. **no ip address**
7. **no keepalive**

8. **exit**
9. **interface pseudowire number**
10. **encapsulation mpls**
11. **neighbor peer-ip-addr vc-id**
12. **l2vpn xconnect context context-name**
13. **member interface-id**
14. **member pseudowire number**
15. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	port-channel load-balance dst-ip 例： Device(config)# port-channel load-balance dst-ip	負荷分散方式を宛先 IP アドレスに設定します。
ステップ 4	interface interface-id 例： Device(config)# interface TenGigabitEthernet1/0/21	トランクとして設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	no switchport 例： Device(config-if)# no switchport	物理ポートに限り、レイヤ3モードを開始します。
ステップ 6	no ip address 例：	物理ポートに割り当てられている IP アドレスがないことを確認します。

	コマンドまたはアクション	目的
	Device(config-if)# no ip address	
ステップ 7	no keepalive 例 : Device(config-if)# no keepalive	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 8	exit 例 : Device(config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、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 9	interface pseudowire number 例 : Device(config)# interface pseudowire 17	指定した値で疑似回線インターフェイスを確立して、疑似回線コンフィギュレーションモードを開始します。
ステップ 10	encapsulation mpls 例 : Device(config-if)# encapsulation mpls	トンネリング カプセル化を指定します。
ステップ 11	neighbor peer-ip-addr vc-id 例 : Device(config-if)# neighbor 10.10.0.10 17	レイヤ 2 VPN (L2VPN) 疑似回線のピア IP アドレスと仮想回線 (VC) ID を指定します。
ステップ 12	l2vpn xconnect context context-name 例 : Device(config-if)# l2vpn xconnect context vpws17	L2VPN クロスコネク トコンテキストを作成して、Xconnect コンテキスト コンフィギュレーションモードを開始します。
ステップ 13	member interface-id 例 :	L2VPN クロスコネク トを形成するインターフェイスを指定します。

	コマンドまたはアクション	目的
	<pre>Device(config-if-xconn)# member TenGigabitEthernet1/0/21</pre>	
ステップ 14	<p>member pseudowire number</p> <p>例 :</p> <pre>Device(config-if-xconn)# member pseudowire 17</pre>	L2VPN クロスコネクトを形成する疑似回線インターフェイスを指定します。
ステップ 15	<p>end</p> <p>例 :</p> <pre>Device(config-if-xconn)# end</pre>	Xconnect インターフェイス コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

Ethernet-over-MPLS VLAN モードの設定

EoMPLS VLAN モードは、Xconnect モードまたはプロトコル CLI 方式のいずれかを使用して設定できます。

Xconnect モード

Xconnect モードで EoMPLS VLAN モードを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **no switchport**
5. **no ip address**
6. **no keepalive**
7. **exit**
8. **interface interface-id.subinterface**
9. **encapsulation dot1Q vlan-id**
10. **xconnect peer-ip-addr vc-id encapsulation mpls**
11. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device (config)# interface TenGigabitEthernet1/0/36	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 4	no switchport 例 : Device (config-if)# no switchport	物理ポートに限り、レイヤ3モードを開始します。
ステップ 5	no ip address 例 : Device (config-if)# no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	no keepalive 例 : Device (config-if)# no keepalive	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 7	exit 例 : Device (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、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 8	interface interface-id.subinterface 例 : Device (config)# interface TenGigabitEthernet1/0/36.1105	設定するサブインターフェイスを定義して、サブインターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 9	encapsulation dot1Q <i>vlan-id</i> 例 : Device(config-subif) # encapsulation dot1Q 1105	サブインターフェイス上で、トラフィックの IEEE 802.1Q カプセル化をイネーブルにします。
ステップ 10	xconnect <i>peer-ip-addr vc-id encapsulation mpls</i> 例 : Device(config-subif) # xconnect 10.0.0.1 1105 encapsulation mpls	接続回線を疑似接続 VC にバインドします。このコマンドの構文は、その他のレイヤ 2 トランスポートの場合と同じです。
ステップ 11	end 例 : Device(config-subif-xconn) # end	特権 EXEC モードに戻ります。

プロトコル CLI 方式

プロトコル CLI モードで EoMPLS VLAN モードを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **port-channel load-balance *dst-ip***
4. **interface *interface-id***
5. **no switchport**
6. **no ip address**
7. **no keepalive**
8. **exit**
9. **interface *interface-id.subinterface***
10. **encapsulation dot1Q *vlan-id***
11. **exit**
12. **interface pseudowire *number***
13. **encapsulation mpls**
14. **neighbor *peer-ip-addr vc-id***
15. **l2vpn xconnect context *context-name***
16. **member *interface-id.subinterface***
17. **member pseudowire *number***
18. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	port-channel load-balance dst-ip 例： Device(config)# port-channel load-balance dst-ip	負荷分散方式を宛先 IP アドレスに設定します。
ステップ 4	interface interface-id 例： Device(config)# interface TenGigabitEthernet1/0/36	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 5	no switchport 例： Device(config-if)# no switchport	物理ポートに限り、レイヤ3モードを開始します。
ステップ 6	no ip address 例： Device(config-if)# no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 7	no keepalive 例： Device(config-if)# no keepalive	デバイスがキープアライブ メッセージを送信しないことを確認します。

	コマンドまたはアクション	目的
ステップ 8	exit 例： Device(config-if) # exit	インターフェイス コンフィギュレーション モードを終了し、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 9	interface interface-id.subinterface 例： Device(config) # interface TenGigabitEthernet1/0/36.1105	設定するサブインターフェイスを定義して、サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 10	encapsulation dot1Q vlan-id 例： Device(config-subif) # encapsulation dot1Q 1105	サブインターフェイス上で、トラフィックの IEEE 802.1Q カプセル化をイネーブルにします。
ステップ 11	exit 例： Device(config-subif) # exit	サブインターフェイス コンフィギュレーション モードを終了し、インターフェイス コンフィギュレーション モードに戻ります。
ステップ 12	interface pseudowire number 例： Device(config) # interface pseudowire 17	指定した値で疑似回線インターフェイスを確立して、疑似回線コンフィギュレーション モードを開始します。
ステップ 13	encapsulation mpls 例： Device(config-if) # encapsulation mpls	トンネリング カプセル化を指定します。
ステップ 14	neighbor peer-ip-addr vc-id 例： Device(config-if) # neighbor 10.10.0.10 17	L2VPN 疑似回線のピア IP アドレスと VC ID 値を指定します。

	コマンドまたはアクション	目的
ステップ 15	l2vpn xconnect context <i>context-name</i> 例： Device(config-if)# l2vpn xconnect context vpws17	L2VPN クロスコネクトコンテキストを作成して、Xconnect コンテキスト コンフィギュレーション モードを開始します。
ステップ 16	member interface-id.subinterface 例： Device(config-if-xconn)# member TenGigabitEthernet1/0/36.1105	L2VPN クロスコネクトを形成するサブインターフェイスを指定します。
ステップ 17	member pseudowire number 例： Device(config-if-xconn)# member pseudowire 17	L2VPN クロスコネクトを形成する疑似回線インターフェイスを指定します。
ステップ 18	end 例： Device(config-if-xconn)# end	Xconnect コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

Ethernet-over-MPLS の設定例

図 3: EoMPLS トポロジ

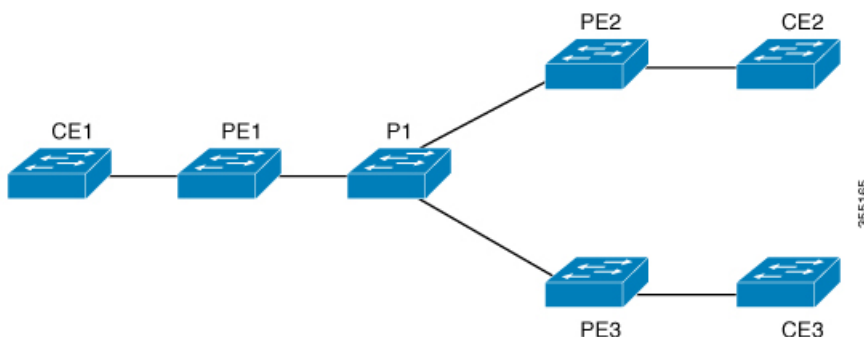


表 2: EoMPLS ポートモードの設定

PE の設定	CE の設定
<pre> mpls ip mpls label protocol ldp mpls ldp graceful-restart mpls ldp router-id loopback 1 force interface Loopback1 ip address 10.1.1.1 255.255.255.255 ip ospf 100 area 0 router ospf 100 router-id 10.1.1.1 nsf system mtu 9198 port-channel load-balance dst-ip ! interface gigabitethernet 2/0/39 no switchport no ip address no keepalive ! interface pseudowire101 encapsulation mpls neighbor 10.10.10.10 101 load-balance flow ip dst-ip load-balance flow-label both l2vpn xconnect context pw101 member pseudowire101 member gigabitethernet 2/0/39 ! interface tengigabitethernet 3/0/10 switchport trunk allowed vlan 142 switchport mode trunk channel-group 42 mode active ! interface Port-channel42 switchport trunk allowed vlan 142 switchport mode trunk ! interface Vlan142 ip address 10.11.11.11 255.255.255.0 ip ospf 100 area 0 mpls ip mpls label protocol ldp ! </pre>	<pre> interface gigabitethernet 1/0/33 switchport trunk allowed vlan 912 switchport mode trunk spanning-tree portfast trunk ! interface Vlan912 ip address 10.91.2.3 255.255.255.0 ! </pre>

表 3: EoMPLS VLAN モードの設定

PE の設定	CE の設定
<pre> interface tengigabitethernet 1/0/36 no switchport no ip address no keepalive exit ! interface tengigabitethernet 1/0/36.1105 encapsulation dot1Q 1105 exit ! interface pseudowire1105 encapsulation mpls neighbor 10.10.0.10 1105 exit ! l2vpn xconnect context vm1105 member tengigabitethernet 1/0/36.1105 member pseudowire1105 end ! </pre>	<pre> interface fortygigabitethernet 1/9 switchport switchport mode trunk switchport trunk allowed vlan 1105 mtu 9216 end ! </pre>

表 4: EoMPLS ポートモードと EoMPLS VLAN モードの設定間のインターワーキング

PE の設定 : ポートモード	CE の設定 : ポートモード
<pre> interface tengigabitethernet 1/0/37 no switchport no ip address no keepalive exit ! interface pseudowire1105 encapsulation mpls neighbor 10.11.11.11 1105 exit ! l2vpn xconnect context vm1105 member tengigabitethernet 1/0/37 member pseudowire1105 end ! </pre>	<pre> interface fortygigabitethernet1/10 switchport switchport mode access switchport access vlan 1105 end no spanning-tree vlan 1105 ! </pre>

PE の設定 : VLAN モード	CE の設定 : VLAN モード
<pre>interface tengigabitethernet 1/0/36 no switchport no ip address no keepalive exit ! interface tengigabitethernet 1/0/36.1105 encapsulation dot1Q 1105 exit ! interface pseudowire1105 encapsulation mpls neighbor 10.10.0.10 1105 exit ! l2vpn xconnect context vme1105 member tengigabitethernet 1/0/36.1105 member pseudowire1105 end !</pre>	<pre>interface fortygigabitethernet 1/9 switchport switchport mode trunk switchport trunk allowed vlan 1105 mtu 9216 end no spanning-tree vlan 1105 !</pre>

EoMPLS ポートモードと EoMPLS VLAN モード間のインターワーキングのもう 1 つのシナリオは、両方の CE デバイスで次のコマンドを設定することです。

- **switchport mode trunk**
- **switchport trunk allowed vlan *vlan-id***
- **spanning-tree vlan *vlan-id***

送信されたトラフィックが二重 VLAN タグ付きでない場合、データトラフィックは両方の CE デバイスで STP を無効化することで流れます。

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

```
Device# show mpls l2 vc vcid 1105 detail
Local interface: TenGigabitEthernet1/0/36.1105 up, line protocol up, Eth VLAN 1105 up
Interworking type is Ethernet
Destination address: 10.0.0.1, VC ID: 1105, VC status: up
Output interface: Po10, imposed label stack {33 10041}
Preferred path: not configured
Default path: active
Next hop: 10.10.0.1
Create time: 00:04:09, last status change time: 00:02:13
Last label FSM state change time: 00:02:12
Signaling protocol: LDP, peer 10.0.0.1:0 up
Targeted Hello: 10.0.0.10(LDP Id) -> 10.0.0.1, LDP is UP
Graceful restart: configured and enabled
Non stop routing: not configured and not enabled
Status TLV support (local/remote) : enabled/supported
LDP route watch : enabled
Label/status state machine : established, LruRru
Last local dataplane status rcvd: No fault
Last BFD dataplane status rcvd: Not sent
Last BFD peer monitor status rcvd: No fault
Last local AC circuit status rcvd: No fault
Last local AC circuit status sent: No fault
```



```

Last local PW i/f circ status rcvd: No fault
Last local LDP TLV      status sent: No fault
Last remote LDP TLV    status rcvd: No fault
Last remote LDP ADJ    status rcvd: No fault
MPLS VC labels: local 124, remote 10041
Group ID: local 336, remote 352
MTU: local 9198, remote 9198
Remote interface description:
MAC Withdraw: sent:1, received:0
Sequencing: receive disabled, send disabled
Control Word: On (configured: autosense)
SSO Descriptor: 10.0.0.1/1105, local label: 124
Dataplane:
SSM segment/switch IDs: 9465983/446574 (used), PWID: 109
VC statistics:
transit packet totals: receive 0, send 0
transit byte totals:   receive 0, send 0
transit packet drops:  receive 0, seq error 0, send 0

```

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

```

Device# show l2vpn atom vc vcid 1105 detail
pseudowire100109 is up, VC status is up PW type: Ethernet
Create time: 00:04:17, last status change time: 00:02:22
Last label FSM state change time: 00:02:20
Destination address: 10.0.0.1 VC ID: 1105
Output interface: Po10, imposed label stack {33 10041}
Preferred path: not configured
Default path: active
Next hop: 10.10.0.1
Member of xconnect service TenGigabitEthernet1/0/36.1105-1105, group right
Associated member TenGigabitEthernet1/0/36.1105 is up, status is up
Interworking type is Ethernet
Service id: 0x1f000037
Signaling protocol: LDP, peer 10.0.0.1:0 up
Targeted Hello: 10.0.0.10(LDP Id) -> 10.0.0.1, LDP is UP
Graceful restart: configured and enabled
Non stop routing: not configured and not enabled
Pwid FEC (128), VC ID: 1105
Status TLV support (local/remote)      : enabled/supported
LDP route watch                        : enabled
Label/status state machine             : established, LruRru
Local dataplane status received        : No fault
BFD dataplane status received          : Not sent
BFD peer monitor status received       : No fault
Status received from access circuit    : No fault
Status sent to access circuit          : No fault
Status received from pseudowire i/f    : No fault
Status sent to network peer            : No fault
Status received from network peer      : No fault
Adjacency status of remote peer        : No fault
Sequencing: receive disabled, send disabled
Bindings
Parameter      Local                               Remote
-----
Label          124                                           10041
Group ID       336                                           352
Interface
MTU            9198                                           9198
Control word on (configured: autosense) on
PW type        Ethernet
VCCV CV type   0x02                                           0x02
                LSPV [2]
VCCV CC type   0x06                                           0x06

```

```

RA [2], TTL [3]
Status TLV enabled RA [2], TTL [3] supported
SSO Descriptor: 10.0.0.1/1105, local label: 124
Dataplane:
SSM segment/switch IDs: 9465983/446574 (used), PWID: 109
Rx Counters
0 input transit packets, 0 bytes
0 drops, 0 seq err
0 MAC withdraw
Tx Counters
0 output transit packets, 0 bytes
0 drops
1 MAC withdraw

```

次に、**show mpls forwarding-table** コマンドの出力例を示します。

```
Device# show mpls forwarding-table 10.0.0.1
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Outgoing interface	Next Hop
2049	33	10.0.0.1/32	38540	Hu2/0/30/2.1	10.0.0.2
	33	10.0.0.1/32	112236	Hu2/0/30/2.2	10.0.0.6
	33	10.0.0.1/32	46188	Hu2/0/30/2.3	10.0.0.8

疑似回線冗長性の設定

ここでは、疑似回線の冗長性を設定する方法について説明します。

疑似回線冗長性の前提条件

- 接続回線で Xconnect モードを設定する前に、**no switchport**、**no keepalive**、および **no ip address** コマンドを設定します。
- ロードバランシングの場合、**port-channel load-balance** コマンドを設定します。
- 疑似回線冗長性 VLAN モードを有効にするには、サブインターフェイスがサポートされている必要があります。

疑似回線冗長性の制約事項

ここでは、疑似回線冗長性ポートモードおよび疑似回線冗長性 VLAN モードの制約事項について説明します。

疑似回線冗長性ポートモードの制約事項

- Ethernet Flow Point (EFP) および Internet Group Management Protocol (IGMP) スヌーピングはサポートされません。
- コアネットワークでの ECMP ロードバランシングのフローラベルは、カスタマーの送信元 IP、宛先 IP、送信元 MAC、および宛先 MAC に基づきます。

- MPLS QoS は、パイプおよび均一モードでサポートされています。デフォルトモードはパイプモードです。
- QoS : カスタマー DSCP 再マーキングは VPWS と EoMPLS ではサポートされていません。
- 明示的 null の VCCV ping はサポートされていません。
- **ip unnumbered** コマンドは MPLS 設定ではサポートされていません。
- 複数のバックアップ疑似回線はサポートされていません。
- PW 冗長グループのスイッチオーバーはサポートされていません。

疑似回線冗長性 VLAN モードの制約事項

- 各 PE デバイスで同じインターワーキングタイプが設定されていない場合、仮想回線は機能しません。
- タグなしトラフィックは、着信トラフィックとしてはサポートされません。
- マルチプレクサ ユーザーネットワーク インターフェイス (MUX UNI) がサポートされていないため、レイヤ 2 サブインターフェイスでは Xconnect モードを有効にできません。
- Xconnect モードは、ポート間トランスポートのメインインターフェイスで有効になっている場合、サブインターフェイスには設定できません。
- Flow Aware Transport (FAT) は、プロトコル CLI モードでのみ設定できます。
- MACsec は、疑似回線冗長性 VLAN モードではサポートされません。
- QoS : カスタマー DSCP 再マーキングは VPWS と疑似回線冗長性ではサポートされていません。
- MPLS QoS は、パイプモードと均一モードでのみサポートされています。デフォルトモードはパイプモードです。
- VLAN モードの類似回線冗長性では、CE からの Cisco Discovery Protocol パケットは PE で処理されますが、類似回線冗長性の仮想回線では伝送されません。一方、ポートモードでは、CE からの Cisco Discovery Protocol パケットは仮想回線で伝送されます。
- イーサネットおよび VLAN インターワーキングタイプのみがサポートされています。
- L2 プロトコル トンネリング CLI はサポートされていません。

疑似回線冗長性について

L2VPN 疑似回線冗長性機能を使用すると、ネットワーク内の障害を検出して、サービスの提供を続行可能な別のエンドポイントにレイヤ 2 サービスを再ルーティングするようにネットワークを設定できます。この機能により、リモート PE デバイスで発生した障害、または PE デバイスと CE デバイス間のリンクで発生した障害から回復できます。

PE 間のすべての中間リンクの最大伝送ユニット (MTU) が、入力 PE で受信される最大のレイヤ 2 パケットを伝達できる必要があります。

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

疑似回線冗長性の設定方法

疑似回線冗長性は、ポートモードまたは VLAN モードで設定できます。

疑似回線冗長性ポートモードの設定

疑似回線冗長性ポートモードは、Xconnect モードまたはプロトコル CLI 方式のいずれかを使用して設定できます。

Xconnect モード

Xconnect モードで疑似回線冗長性ポートモードを設定するには、次の手順を実行します。



(注) ロードバランスを有効にするには、「Ethernet-over-MPLSの設定方法」セクションの Xconnect モードの手順から該当する **load-balance** コマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **no switchport**
5. **no ip address**
6. **no keepalive**
7. **xconnect peer-device-id vc-id encapsulation mpls**
8. **backup peer peer-router-ip-addr vcid vc-id [priority value]**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 :	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
	Device# configure terminal	
ステップ 3	interface interface-id 例 : Device(config)# interface GigabitEthernet1/0/44	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 4	no switchport 例 : Device(config-if)# no switchport	物理ポートに限り、レイヤ 3 モードを開始します。
ステップ 5	no ip address 例 : Device(config-if)# no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	no keepalive 例 : Device(config-if)# no keepalive	デバイスがキープアライブメッセージを送信しないことを確認します。
ステップ 7	xconnect peer-device-id vc-id encapsulation mpls 例 : Device(config-if)# xconnect 10.1.1.1 117 encapsulation mpls	接続回線を疑似接続 VC にバインドします。このコマンドの構文は、その他のレイヤ 2 トランスポートの場合と同じです。
ステップ 8	backup peer peer-router-ip-addr vcid vc-id [priority value] 例 : Device(config-if)# backup peer 10.11.11.11 118 priority 9	疑似回線 VC の冗長ピアを指定します。
ステップ 9	end 例 :	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

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

プロトコル CLI 方式

プロトコル CLI モードで疑似回線冗長性ポートモードを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **port-channel load-balance dst-ip**
4. **interface *interface-id***
5. **no switchport**
6. **no ip address**
7. **no keepalive**
8. **exit**
9. **interface pseudowire *number-active***
10. **encapsulation mpls**
11. **neighbor *active-peer-ip-addr vc-id***
12. **exit**
13. **interface pseudowire *number-standby***
14. **encapsulation mpls**
15. **neighbor *standby-peer-ip-addr vc-id***
16. **l2vpn xconnect context *context-name***
17. **member *interface-id***
18. **member pseudowire *number-active* group *group-name* [priority *value*]**
19. **member pseudowire *number-standby* group *group-name* [priority *value*]**
20. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	port-channel load-balance dst-ip 例 : Device (config) # port-channel load-balance dst-ip	負荷分散方式を宛先 IP アドレスに設定します。
ステップ 4	interface interface-id 例 : Device (config) # interface TenGigabitEthernet1/0/36	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 5	no switchport 例 : Device (config-if) # no switchport	物理ポートに限り、レイヤ3モードを開始します。
ステップ 6	no ip address 例 : Device (config-if) # no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 7	no keepalive 例 : Device (config-if) # no keepalive	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 8	exit 例 : Device (config-if) # exit	インターフェイス コンフィギュレーション モードを終了します。
ステップ 9	interface pseudowire number-active 例 : Device (config) # interface pseudowire 17	指定した値でアクティブ状態の疑似回線インターフェイスを確立して、疑似回線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 10	encapsulation mpls 例： Device(config-if)# encapsulation mpls	トンネリング カプセル化を指定します。
ステップ 11	neighbor active-peer-ip-addr vc-id 例： Device(config-if)# neighbor 10.10.0.10 17	L2VPN 疑似回線のアクティブ状態のピア IP アドレスと VC ID 値を指定します。
ステップ 12	exit 例： Device(config-if)# exit	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 13	interface pseudowire number-standby 例： Device(config)# interface pseudowire 18	指定した値でスタンバイ状態の疑似回線インターフェイスを確立して、疑似回線コンフィギュレーションモードを開始します。
ステップ 14	encapsulation mpls 例： Device(config-if)# encapsulation mpls	トンネリング カプセル化を指定します。
ステップ 15	neighbor standby-peer-ip-addr vc-id 例： Device(config-if)# neighbor 10.10.0.11 18	L2VPN 疑似回線のスタンバイ状態のピア IP アドレスと VC ID 値を指定します。
ステップ 16	l2vpn xconnect context context-name 例： Device(config-if)# l2vpn xconnect context vpws17	L2VPN クロスコネクトコンテキストを作成し、VLAN モードの EoMPLS 接続回線をアクティブ状態およびスタンバイ状態の疑似回線インターフェイスに接続します。

	コマンドまたはアクション	目的
ステップ 17	member interface-id 例 : <pre>Device(config-if-xconn)# member TenGigabitEthernet1/0/36</pre>	L2VPN クロスコネクトを形成するインターフェイスを指定します。
ステップ 18	member pseudowire number-active group group-name [priority value] 例 : <pre>Device(config-if-xconn)# member pseudowire 17 group pwr10</pre>	L2VPN クロスコネクトを形成するアクティブ状態の疑似回線インターフェイスを指定します。
ステップ 19	member pseudowire number-standby group group-name [priority value] 例 : <pre>Device(config-if-xconn)# member pseudowire 18 group pwr10 priority 6</pre>	L2VPN クロスコネクトを形成するスタンバイ状態の疑似回線インターフェイスを指定します。
ステップ 20	end 例 : <pre>Device(config-if-xconn)# end</pre>	Xconnect コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

疑似回線冗長性 VLAN モードの設定

疑似回線冗長性 VLAN モードは、Xconnect モードまたはプロトコル CLI 方式のいずれかを使用して設定できます。

Xconnect モード

Xconnect モードで疑似回線冗長性 VLAN モードを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **no switchport**
5. **no ip address**
6. **no keepalive**
7. **exit**

8. **interface** *interface-id.subinterface*
9. **encapsulation dot1Q** *vlan-id*
10. **xconnect** *peer-ip-addr vc-id encapsulation mpls*
11. **backup peer** *peer-ip-addr vc-id [priority value]*
12. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface TenGigabitEthernet1/0/36	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 4	no switchport 例： Device(config-if)# no switchport	物理ポートに限り、レイヤ3モードを開始します。
ステップ 5	no ip address 例： Device(config-if)# no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 6	no keepalive 例： Device(config-if)# no keepalive	デバイスがキープアライブ メッセージを送信しないことを確認します。

	コマンドまたはアクション	目的
ステップ 7	exit 例： Device (config-if) # exit	インターフェイス コンフィギュレーション モードを終了し、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 8	interface interface-id.subinterface 例： Device (config) # interface TenGigabitEthernet1/0/36.1105	設定するサブインターフェイスを定義して、サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 9	encapsulation dot1Q vlan-id 例： Device (config-subif) # encapsulation dot1Q 1105	サブインターフェイス上で、トラフィックの IEEE 802.1Q カプセル化をイネーブルにします。
ステップ 10	xconnect peer-ip-addr vc-id encapsulation mpls 例： Device (config-subif) # xconnect 10.0.0.1 1105 encapsulation mpls	接続回線を疑似接続 VC にバインドします。このコマンドの構文は、その他のレイヤ 2 トランスポートの場合と同じです。
ステップ 11	backup peer peer-ip-addr vc-id [priority value] 例： Device (config-subif-xconn) # backup peer 10.10.10.10 1105 priority 8	疑似回線 VC の冗長ピアを指定します。
ステップ 12	end 例： Device (config-subif-xconn) # end	Xconnect コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

プロトコル CLI 方式

プロトコル CLI モードで疑似回線冗長性 VLAN モードを設定するには、次の手順を実行します。

手順の概要

1. enable

2. **configure terminal**
3. **port-channel load-balance dst-ip**
4. **interface *interface-id***
5. **no switchport**
6. **no ip address**
7. **no keepalive**
8. **exit**
9. **interface *interface-id.subinterface***
10. **encapsulation dot1Q *vlan-id***
11. **exit**
12. **interface pseudowire *number-active***
13. **encapsulation mpls**
14. **neighbor *active-peer-ip-addr vc-id***
15. **exit**
16. **interface pseudowire *number-standby***
17. **encapsulation mpls**
18. **neighbor *standby-peer-ip-addr vc-id***
19. **l2vpn xconnect context *context-name***
20. **member *interface-id.subinterface***
21. **member pseudowire *number-active* group *group-name* [priority *value*]**
22. **member pseudowire *number-standby* group *group-name* [priority *value*]**
23. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	port-channel load-balance dst-ip 例： Device(config)# port-channel load-balance dst-ip	負荷分散方式を宛先 IP アドレスに設定します。

	コマンドまたはアクション	目的
ステップ 4	interface <i>interface-id</i> 例 : Device (config) # interface TenGigabitEthernet1/0/36	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 5	no switchport 例 : Device (config-if) # no switchport	物理ポートに限り、レイヤ3モードを開始します。
ステップ 6	no ip address 例 : Device (config-if) # no ip address	物理ポートに割り当てられている IP アドレスがないことを確認します。
ステップ 7	no keepalive 例 : Device (config-if) # no keepalive	デバイスがキープアライブ メッセージを送信しないことを確認します。
ステップ 8	exit 例 : Device (config-if) # exit	インターフェイス コンフィギュレーション モードを終了します。
ステップ 9	interface <i>interface-id.subinterface</i> 例 : Device (config) # interface TenGigabitEthernet1/0/36.1105	設定するサブインターフェイスを定義して、サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 10	encapsulation dot1Q <i>vlan-id</i> 例 : Device (config-subif) # encapsulation dot1Q 1105	サブインターフェイス上で、トラフィックの IEEE 802.1Q カプセル化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 11	exit 例： Device(config-subif)# exit	サブインターフェイスコンフィギュレーションモードを終了します。
ステップ 12	interface pseudowire number-active 例： Device(config)# interface pseudowire 17	指定した値でアクティブ状態の疑似回線インターフェイスを確立して、疑似回線コンフィギュレーションモードを開始します。
ステップ 13	encapsulation mpls 例： Device(config-if)# encapsulation mpls	トンネリング カプセル化を指定します。
ステップ 14	neighbor active-peer-ip-addr vc-id 例： Device(config-if)# neighbor 10.10.0.10 17	レイヤ 2 VPN (L2VPN) 疑似回線のアクティブなピア IP アドレスと仮想回線 (VC) ID を指定します。
ステップ 15	exit 例： Device(config-if)# exit	インターフェイス設定モードを終了します。
ステップ 16	interface pseudowire number-standby 例： Device(config)# interface pseudowire 18	指定した値でスタンバイ状態の疑似回線インターフェイスを確立して、疑似回線コンフィギュレーションモードを開始します。
ステップ 17	encapsulation mpls 例： Device(config-if)# encapsulation mpls	トンネリング カプセル化を指定します。
ステップ 18	neighbor standby-peer-ip-addr vc-id 例：	L2VPN 疑似回線のスタンバイ状態のピア IP アドレスと VC ID 値を指定します。

	コマンドまたはアクション	目的
	Device(config-if)# neighbor 10.10.0.11 18	
ステップ 19	l2vpn xconnect context context-name 例 : Device(config-if)# l2vpn xconnect context vpws17	L2VPN クロスコネクトコンテキストを作成し、VLAN モードの EoMPLS 接続回線をアクティブ状態およびスタンバイ状態の疑似回線インターフェイスに接続します。
ステップ 20	member interface-id.subinterface 例 : Device(config-if-xconn)# member TenGigabitEthernet1/0/36.1105	L2VPN クロスコネクトを形成するインターフェイスを指定します。
ステップ 21	member pseudowire number-active group group-name [priority value] 例 : Device(config-if-xconn)# member pseudowire 17 group pwr10	L2VPN クロスコネクトを形成するアクティブ状態の疑似回線インターフェイスを指定します。
ステップ 22	member pseudowire number-standby group group-name [priority value] 例 : Device(config-if-xconn)# member pseudowire 18 group pwr10 priority 6	L2VPN クロスコネクトを形成するスタンバイ状態の疑似回線インターフェイスを指定します。
ステップ 23	end 例 : Device(config-if-xconn)# end	Xconnect コンフィギュレーションモードを終了して、特権 EXEC モードに戻ります。

疑似回線冗長性の設定例

表 5: 疑似回線冗長性ポートモードの設定

PE の設定	CE の設定
<pre> mpls ip mpls label protocol ldp mpls ldp graceful-restart mpls ldp router-id loopback 1 force ! interface Loopback1 ip address 10.1.1.1 255.255.255.255 ip ospf 100 area 0 router ospf 100 router-id 10.1.1.1 nsf ! interface gigabitethernet 2/0/39 no switchport no ip address no keepalive ! interface pseudowire101 encapsulation mpls neighbor 10.10.10.10 101 ! interface pseudowire102 encapsulation mpls neighbor 10.10.10.11 101 l2vpn xconnect context pw101 member pseudowire101 group pwgrp1 priority 1 member pseudowire102 group pwgrp1 priority 15 member GigabitEthernet2/0/39 ! interface tengigabitethernet 3/0/10 switchport trunk allowed vlan 142 switchport mode trunk channel-group 42 mode active ! interface Port-channel42 switchport trunk allowed vlan 142 switchport mode trunk ! interface Vlan142 ip address 10.11.11.11 255.255.255.0 ip ospf 100 area 0 mpls ip mpls label protocol ldp ! </pre>	<pre> interface gigabitethernet 1/0/33 switchport trunk allowed vlan 912 switchport mode trunk spanning-tree portfast trunk ! interface Vlan912 ip address 10.91.2.3 255.255.255.0 ! </pre>

表 6: 疑似回線冗長 VLAN モードの設定

PE の設定	CE の設定
<pre> interface tengigabitethernet 1/0/36 no switchport no ip address no keepalive exit ! interface tengigabitethernet 1/0/36.1105 encapsulation dot1Q 1105 exit ! interface pseudowire1105 encapsulation mpls neighbor 10.10.0.10 1105 exit ! interface pseudowire1106 encapsulation mpls neighbor 10.10.0.11 1106 ! l2vpn xconnect context vme1105 member tengigabitethernet 1/0/36.1105 member pseudowire1105 group pwr10 member pseudowire1106 group pwr10 priority 6 end ! </pre>	<pre> interface fortygigabitethernet 1/9 switchport switchport mode trunk switchport trunk allowed vlan 1105 mtu 9216 end ! </pre>

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

```

Device# show mpls l2 vc vcid 1105 detail
Local interface: TenGigabitEthernet1/0/36.1105 up, line protocol up, Eth VLAN 1105 up
  Interworking type is Ethernet
  Destination address: 10.11.11.11, VC ID: 1105, VC status: standby
    Output interface: Po10, imposed label stack {1616}
    Preferred path: not configured
    Default path: active
    Next hop: 10.10.0.1
  Create time: 00:04:09, last status change time: 00:02:13
  Last label FSM state change time: 00:02:15
  Signaling protocol: LDP, peer 10.11.11.11:0 up
  Targeted Hello: 10.10.0.10(LDP Id) -> 10.11.11.11, LDP is UP
  Graceful restart: configured and enabled
  Non stop routing: not configured and not enabled
  Status TLV support (local/remote)   : enabled/supported
    LDP route watch                   : enabled
    Label/status state machine        : established, LrdRru
    Last local dataplane status rcvd: No fault
    Last BFD dataplane status rcvd: Not sent
    Last BFD peer monitor status rcvd: No fault
    Last local AC circuit status rcvd: DOWN(standby)
    Last local AC circuit status sent: No fault
    Last local PW i/f circ status rcvd: No fault
    Last local LDP TLV status sent: DOWN(standby)
    Last remote LDP TLV status rcvd: No fault
    Last remote LDP ADJ status rcvd: No fault
  MPLS VC labels: local 125, remote 1616
  Group ID: local 336, remote 0

```

```

MTU: local 9198, remote 9198
Remote interface description:
MAC Withdraw: sent:1, received:0
Sequencing: receive disabled, send disabled
Control Word: On (configured: autosense)
SSO Descriptor: 10.11.11.11/1105, local label: 125
Dataplane:
  SSM segment/switch IDs: 96143/450671 (used), PWID: 110
VC statistics:
  transit packet totals: receive 0, send 0
  transit byte totals:  receive 0, send 0
  transit packet drops:  receive 0, seq error 0, send 0

```

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

```

Device# show l2vpn atom vc vcid 1105 detail
pseudowire100110 is up, VC status is standby PW type: Ethernet
  Create time: 00:04:17, last status change time: 00:02:22
  Last label FSM state change time: 00:02:24
  Destination address: 10.11.11.11 VC ID: 1105
  Output interface: Po10, imposed label stack {1616}
  Preferred path: not configured
  Default path: active
  Next hop: 10.0.0.1
Member of xconnect service TenGigabitEthernet1/0/36.1105-1105, group right
  Associated member TenGigabitEthernet1/0/36.1105 is up, status is up
  Interworking type is Ethernet
  Service id: 0x1f000037
Signaling protocol: LDP, peer 10.11.11.11:0 up
  Targeted Hello: 10.0.0.10(LDP Id) -> 10.11.11.11, LDP is UP
  Graceful restart: configured and enabled
  Non stop routing: not configured and not enabled
  PWid FEC (128), VC ID: 1105
  Status TLV support (local/remote)           : enabled/supported
  LDP route watch                             : enabled
  Label/status state machine                  : established, LrdRru
  Local dataplane status received             : No fault
  BFD dataplane status received               : Not sent
  BFD peer monitor status received            : No fault
  Status received from access circuit         : DOWN(standby)
  Status sent to access circuit               : No fault
  Status received from pseudowire i/f         : No fault
  Status sent to network peer                 : DOWN(standby)
  Status received from network peer           : No fault
  Adjacency status of remote peer            : No fault
Sequencing: receive disabled, send disabled
Bindings
  Parameter      Local                               Remote
  -----
  Label          125                                           1616
  Group ID      336                                           0
  Interface
  MTU           9198                                           9198
  Control word  on (configured: autosense)                   on
  PW type       Ethernet                                       Ethernet
  VCCV CV type  0x02                                           0x02
                LSPV [2]                               LSPV [2]
  VCCV CC type  0x06                                           0x02
                RA [2], TTL [3]                         RA [2]
  Status TLV    enabled                                       supported
SSO Descriptor: 10.11.11.11/1105, local label: 125
Dataplane:
  SSM segment/switch IDs: 96143/450671 (used), PWID: 110
Rx Counters

```

```

0 input transit packets, 0 bytes
0 drops, 0 seq err
0 MAC withdraw
Tx Counters
0 output transit packets, 0 bytes
0 drops
1 MAC withdraw

```

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

```
Device# show mpls l2transport vc 101
```

Local intf	Local circuit	Dest address	VC ID	Status
TenGigabitEthernet1/0/36.1105	Eth VLAN 1105	10.0.0.1	1105	UP
TenGigabitEthernet1/0/36.1105	Eth VLAN 1105	10.11.11.11	1105	STANDBY

Ethernet-over-MPLS および疑似回線冗長性の機能情報

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

表 7: Ethernet-over-MPLS および疑似回線冗長性の機能情報

機能名	リリース	機能情報
Ethernet-over-MPLS および疑似回線冗長性	Cisco IOS XE Everest 16.6.1	この機能が導入されました。 ポートモードのサポートが導入されました。
Ethernet over MPLS および疑似回線冗長性	Cisco IOS XE Gibraltar 16.12.1	VLAN モードのサポートが導入されました。
Macsec over EoMPLS	Cisco IOS XE Amsterdam 17.1.1	VLAN モード EoMPLS の場合、CE デバイスで <code>macsec dot1q-in-clear 1</code> コマンドによって設定されたパケットのみが PE デバイスで処理されます。



第 6 章

MPLS を介した IPv6 プロバイダー エッジ (6PE) の設定

- [6PE の前提条件 \(87 ページ\)](#)
- [6PE の制約事項 \(87 ページ\)](#)
- [6PE について \(87 ページ\)](#)
- [6PE の設定 \(88 ページ\)](#)
- [6PE の設定例 \(91 ページ\)](#)
- [MPLS を介した IPv6 プロバイダーエッジ \(6PE\) の機能履歴 \(93 ページ\)](#)

6PE の前提条件

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

6PE の制約事項

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

6PE について

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

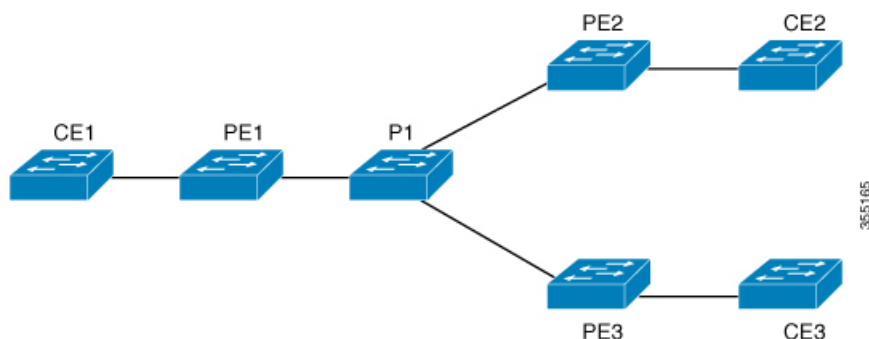
6PE 実装時は、プロバイダー エッジ ルータが 6PE をサポートするようにアップグレードされますが、残りのコア ネットワークに影響することはありません (IPv6 非対応)。転送が IP

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

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

次の図に 6PE トポロジを示します。

図 4: 6PE トポロジ



6PE の設定

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

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

6PE を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 unicast-routing**
4. **router bgp *as-number***
5. **bgp router-id interface *interface-id***

6. **bgp log-neighbor-changes**
7. **bgp graceful-restart**
8. **neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **remote-as** *as-number*
9. **neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **update-source** *interface-type* *interface-number*
10. **address-family ipv6**
11. **redistribute protocol** *as-number* **match** { **internal** | **external 1** | **external 2**
12. **neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **activate**
13. **neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **send-label**
14. **exit-address-family**
15. **end**

手順の詳細

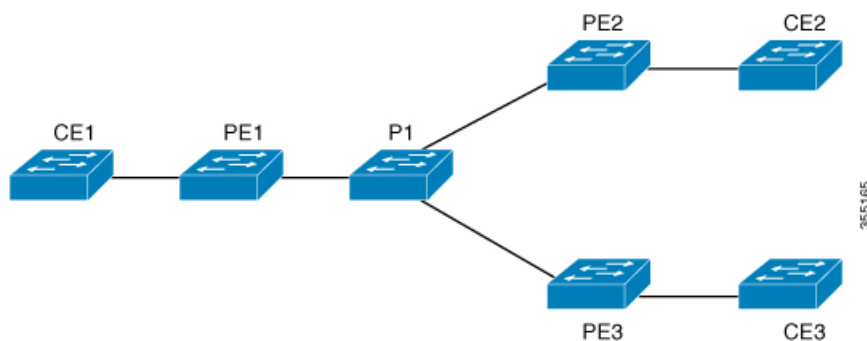
	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 unicast-routing 例 : Device(config)# ipv6 unicast-routing	IPv6 ユニキャスト データグラムの転送を有効にします。
ステップ 4	router bgp as-number 例 : Device(config)# router bgp 65001	ルータが存在する自律システム (AS) を識別する番号を入力します。 <i>as-number</i> : 自律システム番号。2 バイトの番号の範囲は 1 ~ 65535 です。4 バイトの番号の範囲は 1.0 ~ 65535.65535 です。
ステップ 5	bgp router-id interface interface-id 例 : Device(config-router)# bgp router-id interface Loopback1	ローカル ボーダー ゲートウェイ プロトコル (BGP) ルーティングプロセスの固定ルータ ID を設定します。
ステップ 6	bgp log-neighbor-changes 例 :	BGP ネイバーリセットのロギングを有効にします。

	コマンドまたはアクション	目的
	Device(config-router)# bgp log-neighbor-changes	
ステップ 7	bgp graceful-restart 例 : Device(config-router)# bgp graceful-restart	すべての Border Gateway Protocol (BGP) ネイバーで BGP グレースフルリスタート機能をグローバルで有効にします。
ステップ 8	neighbor { ip-address ipv6-address peer-group-name } remote-as as-number 例 : Device(config-router)# neighbor 33.33.33.33 remote-as 65001	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。 <ul style="list-style-type: none"> • <i>ip-address</i> : ルーティング情報を交換するピア ルータの IP アドレス。 • <i>ipv6-address</i> : ルーティング情報を交換するピア ルータの IPv6 アドレス。 • <i>peer-group-name</i> : BGP ピア グループの名前。 • <i>remote-as</i> : リモート自律システムを指定します。 • <i>as-number</i> : ネイバーが属する自律システムの 1 ~ 65535 の範囲内の番号。
ステップ 9	neighbor { ip-address ipv6-address peer-group-name } update-source interface-type interface-number 例 : Device(config-router)# neighbor 33.33.33.33 update-source Loopback1	BGP セッションが TCP 接続の動作インターフェイスを使用できるように設定します。
ステップ 10	address-family ipv6 例 : Device(config-router)# address-family ipv6	標準 IPv6 アドレスプレフィックスを使用する BGP などのルーティングセッションを設定するために、アドレスファミリ コンフィギュレーションモードを開始します。
ステップ 11	redistribute protocol as-number match { internal external 1 external 2 } 例 : Device(config-router-af)# redistribute ospf 11 match internal external 1	ルートを1つのルーティングドメインから他のルーティングドメインに再配布します。
ステップ 12	neighbor { ip-address ipv6-address peer-group-name } activate 例 :	BGP ネイバーとの情報交換を有効にします。

	コマンドまたはアクション	目的
	Device(config-router-af)# neighbor 33.33.33.33 activate	
ステップ 13	neighbor { ip-address ipv6-address peer-group-name } send-label 例 : Device(config-router-af)# neighbor 33.33.33.33 send-label	隣接 BGP ルータに BGP ルートを含む MPLS ラベルを送信します。
ステップ 14	exit-address-family 例 : Device(config-router-af)# exit-address-family	BGP アドレス ファミリ サブモードを終了します。
ステップ 15	end 例 : Device(config)# end	特権 EXEC モードに戻ります。

6PE の設定例

図 5: 6PE トポロジ



PE の設定

```

router ospfv3 11
ip routing
ipv6 unicast-routing
address-family ipv6 unicast
redistribute bgp 65001
exit-address-family
!
router bgp 65001
bgp router-id interface Loopback1
bgp log-neighbor-changes
bgp graceful-restart
neighbor 33.33.33.33 remote-as 65001
neighbor 33.33.33.33 update-source Loopback1
!
address-family ipv4
neighbor 33.33.33.33 activate
!
address-family ipv6
redistribute ospf 11 match internal external 1 external 2 include-connected
neighbor 33.33.33.33 activate
neighbor 33.33.33.33 send-label
neighbor 33.33.33.33 send-community extended
!

```

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

```

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

```

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

```

sh ipv route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route

      B - BGP, R - RIP, I1 - ISIS L1, I2 - ISIS L2
      IA - ISIS interarea, IS - ISIS summary, D - EIGRP, EX - EIGRP
external
      ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr -
Redirect
      RL - RPL, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
      OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2

```

```

    la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
1A - LISP away
C  10:1:1:2::/64 [0/0]
    via Vlan4, directly connected
L  10:1:1:2::1/128 [0/0]
    via Vlan4, receive
LC 11:11:11:11::11/128 [0/0]
    via Loopback1, receive
B  30:1:1:2::/64 [200/0]
    via 33.33.33.33%default, indirectly connected
B  40:1:1:2::/64 [200/0]
    via 44.44.44.44%default, indirectly connected

```

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

```

BGP table version is 112, local router ID is 11.11.11.11
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
                r RIB-failure, S Stale, m multipath, b backup-path, f
RT-Filter,
                x best-external, a additional-path, c RIB-compressed,
                t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

```

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	10:1:1:2::/64	::	0		32768	?
*>i	30:1:1:2::/64	::FFFF:33.33.33.33		0	100	0 ?
*>i	40:1:1:2::/64	::FFFF:44.44.44.44		0	100	0 ?
*>i	173:1:1:2::/64	::FFFF:33.33.33.33		2	100	0 ?

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

```

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

```

MPLS を介した IPv6 プロバイダーエッジ (6PE) の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	MPLS を介した IPv6 プロバイダーエッジ (6PE)	MPLS を介した IPv6 プロバイダーエッジ (6PE) は、IPv4 MPLS を介したグローバル IPv6 到達可能性を提供し、他のすべてのデバイスに 1 つの共有ルーティングテーブルを提供します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 7 章

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

- [6VPE の設定 \(95 ページ\)](#)

6VPE の設定

次の項では、スイッチでの 6VPE の設定について説明します。

6VPE の制約事項

- Inter-AS および Carrier Supporting Carrier (CSC) はサポートされていません。
- VRF ルートリーキングはサポートされていません。
- eBGP は CE-PE としてサポートされていません。
- EIGRP、OSPFv3、RIP、ISIS、スタティックルートは、CE-PE としてサポートされていません。
- サポートされている MPLS ラベル割り当てモードは VRF 単位とプレフィックス単位です。プレフィックス単位がデフォルトのモードです。
- IP フラグメンテーションは、レイヤ 3 VPN の Per-Prefix モードではサポートされていません。
- DHCPv6 は、ポート単位の信頼が有効になっている 6VPE トポロジではサポートされません。

6VPE について

6VPE は IPv4 バックボーンを使用して VPN IPv6 サービスを提供するメカニズムです。使用可能な IPv4 MPLS バックボーンを利用することで、MPLS コア内でのデュアルスタッキングが不要になります。つまり、運用コストを削減し、6PE アプローチのセキュリティ上の制限に対処

します。6VPE は、通常の IPv4 MPLS-VPN プロバイダー エッジ とほぼ同じですが、VRF 内に IPv6 サポート が追加されています。これは、VPN メンバー デバイス 用に、論理的に分割されたルーティング テーブル エントリ を提供します。

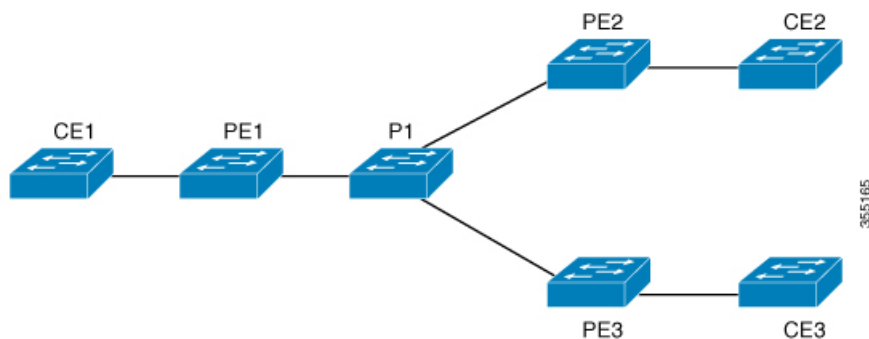
MPLS ベースの 6VPE ネットワークのコンポーネント

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

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

6VPE の設定例

図 6: 6VPE トポロジ



PE の設定

PE の設定

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


PE の設定

```
neighbor 55.55.55.55 send-community both
exit-address-family
!
```

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

```
Local Outgoing Prefix Bytes Label Outgoing Next Hop
Label Label or Tunnel Id Switched interface
29 No Label A:A:A:565::/64[V] \ 0 aggregate/VRF601
32 No Label A:B5:1:5::/64[V] 2474160 V1601 FE80::200:7BFF:FE62:2636
33 No Label A:B5:1:4::/64[V] 2477978 V1601 FE80::200:7BFF:FE62:2636
35 No Label A:B5:1:3::/64[V] 2477442 V1601 FE80::200:7BFF:FE62:2636
36 No Label A:B5:1:2::/64[V] 2476906 V1601 FE80::200:7BFF:FE62:2636
37 No Label A:B5:1:1::/64[V] 2476370 V1601 FE80::200:7BFF:FE62:2636
```

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

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

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

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

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

MPLS を介した IPv6 VPN プロバイダーエッジ (6VPE) の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	MPLS を介した IPv6 VPN プロバイダーエッジ (6VPE)	MPLS を介した IPv6 VPN プロバイダーエッジ (6VPE) は IPv4 バックボーンを使用して VPN IPv6 サービスを提供するメカニズムです。使用可能な IPv4 MPLS バックボーンを利用することで、MPLS コア内でのデュアルスタッキングが不要になります。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 8 章

MPLS VPN InterAS オプションの設定

- [MPLS VPN InterAS オプションに関する情報](#) (101 ページ)
- [MPLS VPN InterAS オプションの設定方法](#) (105 ページ)
- [MPLS VPN InterAS オプションの設定の確認](#) (115 ページ)
- [MPLS VPN InterAS オプションの設定例](#) (116 ページ)
- [MPLS VPN InterAS オプションに関するその他の参考資料](#) (128 ページ)
- [MPLS VPN InterAS オプションの機能履歴](#) (128 ページ)

MPLS VPN InterAS オプションに関する情報

MPLS VPN InterAS オプションは、異なる MPLS VPN サービスプロバイダー間で VPN を相互接続するさまざまな方法を提供します。これにより、お客様のサイトを複数のキャリアネットワーク（自律システム）に存在させ、サイト間でのシームレスな VPN 接続が可能になります。

ASE および ASBR

自律システム（AS）とは、共通のシステム管理グループによって管理され、単一の明確に定義されたプロトコルを使用している単一のネットワークまたはネットワークのグループのことです。多くの場合、VPN は異なる地理的領域の異なる AS に拡張されます。一部の VPN は、複数のサービスプロバイダーにまたがって拡張する必要があり、それらはオーバーラッピング VPN と呼ばれます。VPN の複雑さや場所に関係なく、AS 間の接続はお客様に対してシームレスである必要があります。

AS 境界ルータ（ASBR）は、複数のルーティングプロトコルを使用して接続された AS 内のデバイスであり、外部ルーティングプロトコル（eBGP など）またはスタティックルートを使用するか、あるいは両方を使用して、他の ASBR とルーティング情報を交換します。

異なるサービスプロバイダーからの個別の AS は、VPN IP アドレスの形式で情報を交換することによって通信し、次のプロトコルを使用してルーティング情報を共有します。

- AS 内では、ルーティング情報は iBGP を使用して共有されます。

iBGP は、各 VPN および各 AS 内の IP プレフィックスのネットワーク層情報を配布します。

- AS 間では、ルーティング情報は eBGP を使用して共有されます。

eBGP を使用することで、サービスプロバイダーは別の AS 間でのルーティング情報のループフリー交換を保証するインターネットルーティングシステムを設定できます。eBGP の主な機能は、AS ルートのリストに関する情報を含む、AS 間のネットワーク到達可能性情報を交換することです。AS は、eBGP ボーダーエッジルータを使用してラベルスイッチング情報を含むルートを配布します。各ボーダーエッジルータでは、ネクストホップおよび MPLS ラベルが書き換えられます。

MPLS VPN InterAS オプションの設定はサポートされており、異なるボーダーエッジルータで接続されている 2 つ以上の AS を含む MPLS VPN であるプロバイダー間 VPN を含めることができます。AS は eBGP を使用してルートを交換し、iBGP やルーティング情報は AS 間で交換されません。

MPLS VPN InterAS オプション

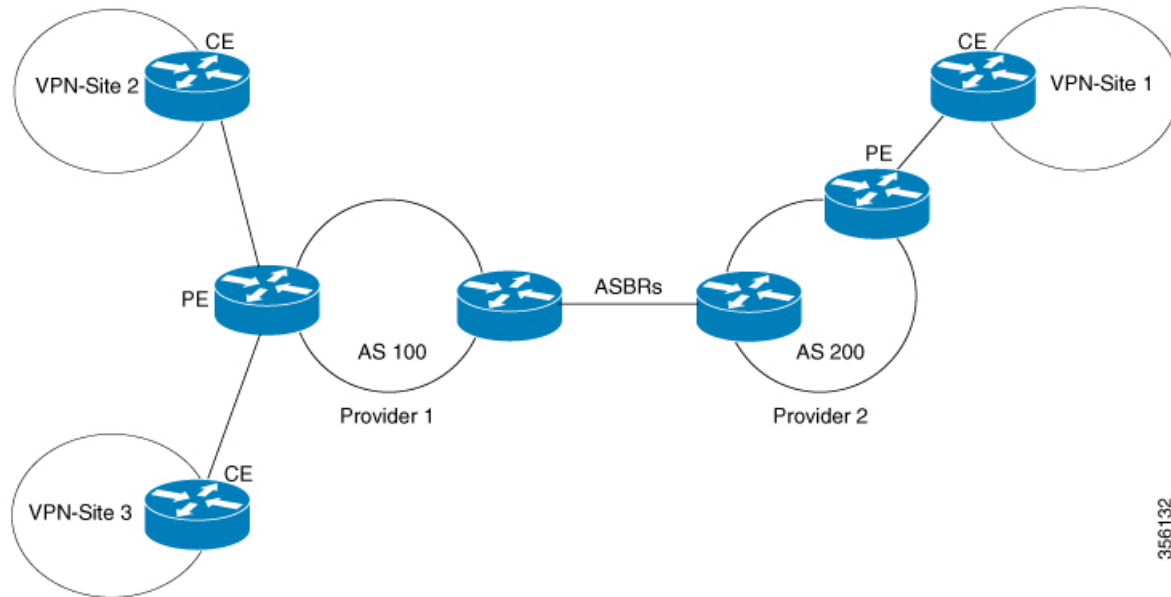
RFC4364 で定義されている次のオプションは、異なる AS 間の MPLS VPN 接続を提供します。

- InterAS オプション B : このオプションは、ASBR 間の VPNv4 ルート配布を提供します。

InterAS オプション B

InterAS オプション B ネットワークでは、ASBR ポートは、MPLS トラフィックを受信できる 1 つ以上のインターフェイスによって接続されます。このオプションを使用すると、ASBR は eBGP セッションを使用して相互にピアリングします。ASBR は PE ルータとしても機能し、AS 内のすべての PE ルータとピアリングします。ASBR は VRF を保持しませんが、他の AS に渡す必要がある PE ルータからの VPNv4 ルートのすべてまたはサブセットを保持します。VPNv4 ルートは、route-distinguisher を使用して ASBR で一意に維持され、ルートターゲットを使用してフィルタリングされます。ASBR は、eBGP を使用して VPNv4 ルートと VPN ラベルを交換します。

図 7: InterAS オプション B のトポロジ



ASBR 間で VPNv4 ルートのネクストホップを配布するための 2 つの方法がサポートされています。2 つの ASBR を接続するリンクで LDP または IGP を有効にする必要はありません。ASBR 上の直接接続されたインターフェイス間の MP-eBGP セッションにより、インターフェイスはラベル付きパケットを転送できます。直接接続された BGP ピアに対してこの MPLS 転送を保証するには、ASBR に接続するインターフェイスで `mpls bgp forwarding` コマンドを設定する必要があります。このコマンドは、直接接続されたインターフェイスの IOS に実装されています。最大 200 の BGP ネイバーを設定できます。

- **ネクストホップセルフ方式**：ネクストホップを他の ASBR から学習したすべての VPNv4 ルートのローカル ASBR のネクストホップに変更します。
- **Redistribute Connected Subnet 方式**：`redistribute connected subnets` コマンドを使用して、リモート ASBR のネクストホップアドレスをローカル IGP に再配布します。つまり、VPNv4 ルートがローカル AS に再配布されても、ネクストホップは変更されません。

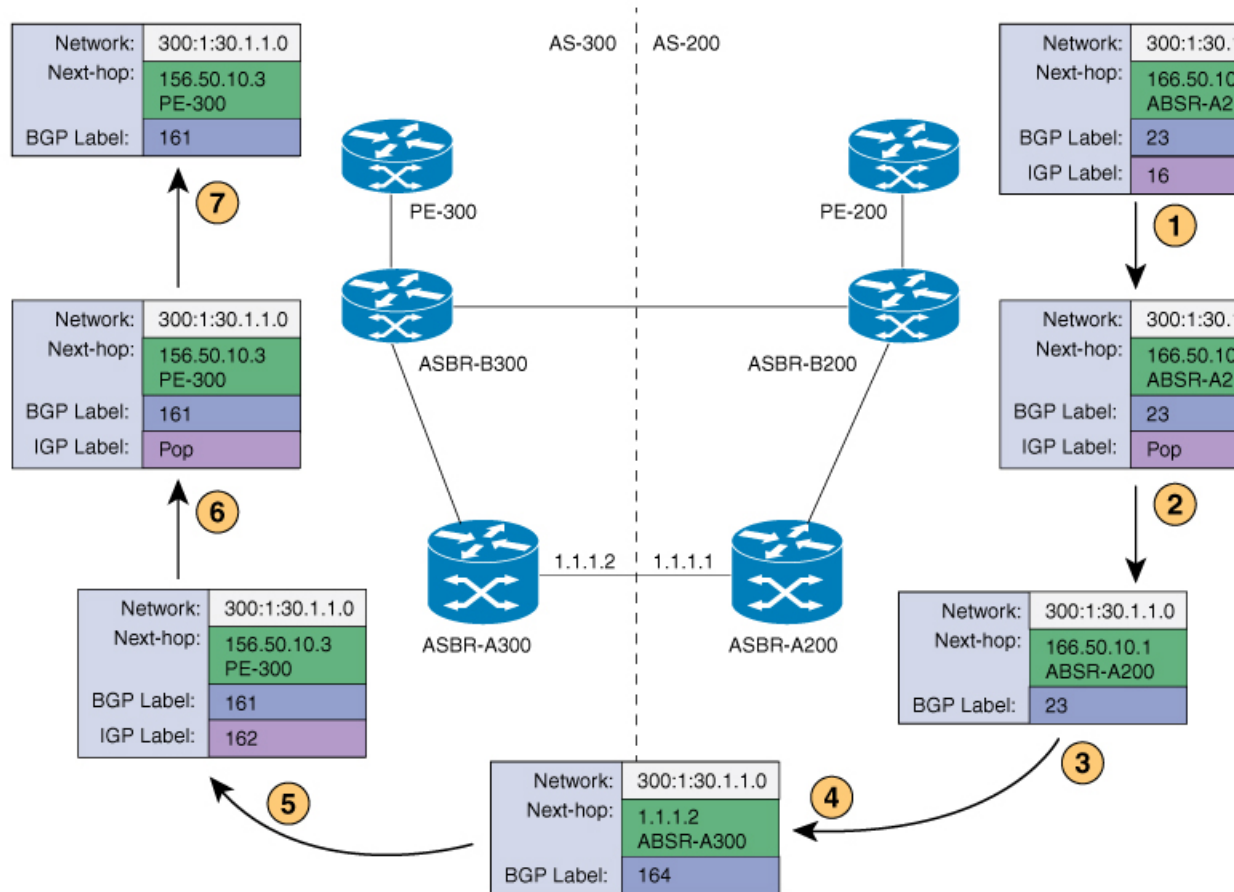


(注) 等コストパス（リモート AS への ECMP）が複数ある場合は、ASBR 上のリモートループバックに対する MPLS スタティック ラベルバインディングを設定する必要があります。そのように設定しないと、パケットが損失する場合があります。

次に説明するラベルスイッチパス転送の項では、AS200 はネクストホップセルフ方式で設定されており、AS300 は Redistribute Subnet 方式で設定されています。

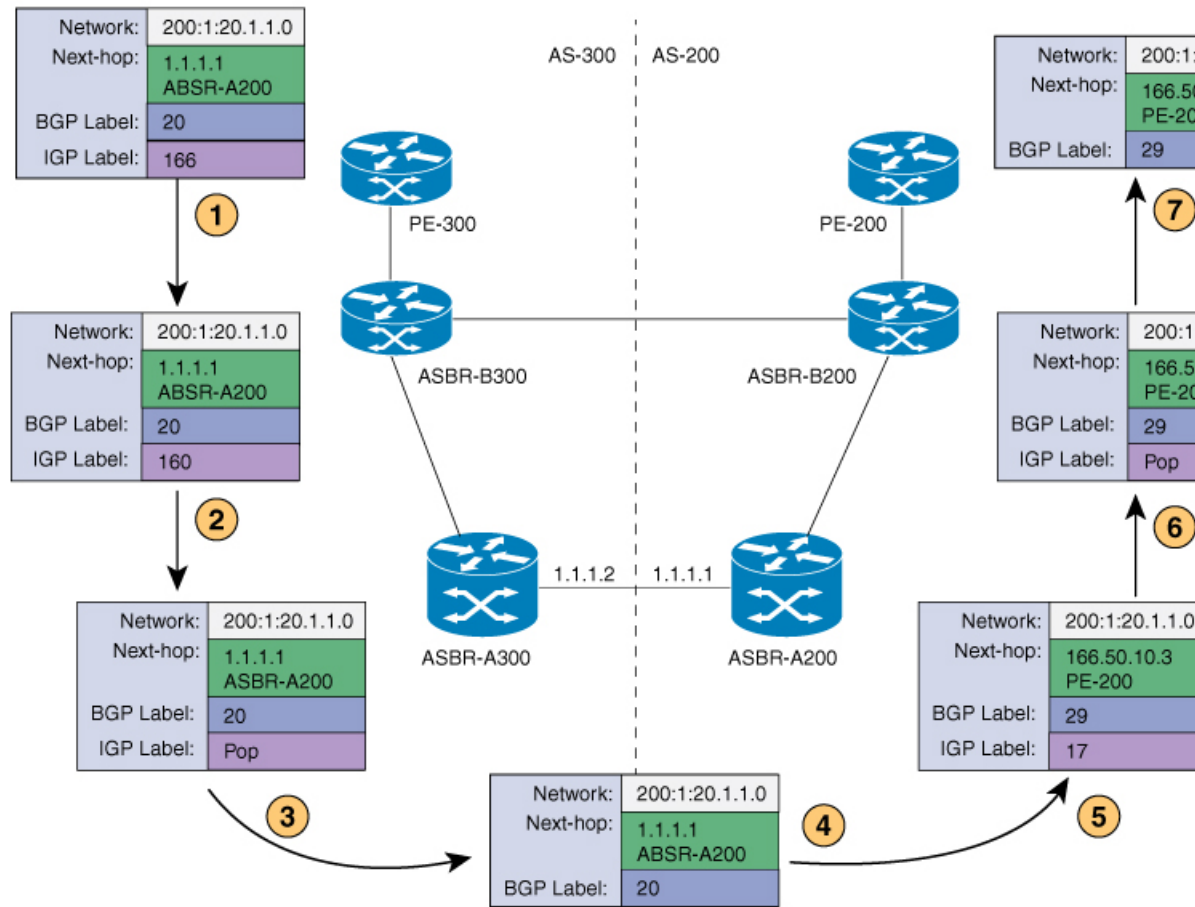
ネクストホップセルフ方式

次の図に、ネクストホップセルフ方式のラベル転送パスを示します。パケットが AS 200 の PE-200 から AS 300 の PE-300 に到達するときに、ラベルがスタックにプッシュ、スワップ、およびポップされます。ステップ 5 で、ASBR-A300 はラベル付きフレームを受信し、ラベル 164 をラベル 161 に置き換え、IGP ラベル 162 をラベルスタックにプッシュします。



Redistribute Connected Subnet 方式

次の図に、Redistribute Connected Subnet 方式のラベル転送パスを示します。パケットが AS 300 の PE-300 から AS 200 の PE-200 に移動するときに、ラベルがスタックにプッシュ、スワップ、およびポップされます。ステップ 5 で、ASBR-A200 は BGP ラベル 20 のフレームを受信し、ラベル 29 と交換し、ラベル 17 をプッシュします。



MPLS VPN InterAS オプションの設定方法

次の項では、MPLS VPN InterAS オプションの設定方法について説明します。

MPLS VPN InterAS オプション B の設定

ネクストホップセルフ方式を使用した InterAS オプション B の設定

ネクストホップセルフ方式を使用して ASBR で InterAS オプション B を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router ospf process-id**
4. **router-id ip-address**

5. **nsr**
6. **nsf**
7. **redistribute bgp** *autonomous-system-number*
8. **passive-interface** *interface-type interface-number*
9. **network** *ip-address wildcard-mask* **area** *area-id*
10. **exit**
11. **router bgp** *autonomous-system-number*
12. **bgp router-id** *ip-address*
13. **bgp log-neighbor changes**
14. **no bgp default ipv4-unicast**
15. **no bgp default route-target filter**
16. **neighbor** *ip-address* **remote-as** *as-number*
17. **neighbor** *ip-address* **update-source** *interface-type interface-number*
18. **neighbor** *ip-address* **remote-as** *as-number*
19. **address-family** *ipv4*
20. **neighbor** *ip-address* **activate**
21. **neighbor** *ip-address* **send-label**
22. **exit address-family**
23. **address-family** *vpn4*
24. **neighbor** *ip-address* **activate**
25. **neighbor** *ip-address* **send-community** **extended**
26. **neighbor** *ip-address* **next-hop-self**
27. **neighbor** *ip-address* **activate**
28. **neighbor** *ip-address* **send-community** **extended**
29. **exit address-family**
30. **bgp router-id** *ip-address*
31. **bgp log-neighbor changes**
32. **neighbor** *ip-address* **remote-as** *as-number*
33. **neighbor** *ip-address* **update-source** *interface-type interface-number*
34. **address-family** *vpn4*
35. **neighbor** *ip-address* **activate**
36. **neighbor** *ip-address* **send-community** **extended**
37. **exit address-family**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 :	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
	Device# configure terminal	
ステップ 3	router ospf process-id 例 : Device (config)# router ospf 1	OSPF ルーティングプロセスを設定し、プロセス番号を割り当てます。
ステップ 4	router-id ip-address 例 : Device (config)# router-id 4.1.1.1	固定ルータ ID を指定します。
ステップ 5	nsr 例 : Device (config-router)# nsr	OSPF ノンストップルーティング (NSR) を設定します。
ステップ 6	nsf 例 : Device (config-router)# nsf	OSPF ノンストップフォワーディング (NSF) を設定します。
ステップ 7	redistribute bgp autonomous-system-number 例 : Device (config-router)# redistribute bgp 200	BGP 自律システムからルートを OSPF ルーティングプロセスに再配布します。
ステップ 8	passive-interface interface-type interface-number 例 : Device (config-router)# passive-interface GigabitEthernet 1/0/10 Device (config-router)# passive-interface Tunnel0	インターフェイスの Open Shortest Path First (OSPF) ルーティングアップデートを無効にします。
ステップ 9	network ip-address wildcard-mask aread area-id 例 : Device (config-router)# network 4.1.1.0 0.0.0.0.255 area 0	OSPF を実行するインターフェイスを定義し、そのインターフェイスに対するエリア ID を定義します。
ステップ 10	exit 例 : Device (config-router)# exit	ルータ コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 11	router bgp <i>autonomous-system-number</i> 例 : Device(config)# router bgp 200	BGP ルーティングプロセスを設定します。
ステップ 12	bgp router-id <i>ip-address</i> 例 : Device(config-router)# bgp router-id 4.1.1.1	BGP ルーティングプロセスの固定ルータ ID を設定します。
ステップ 13	bgp log-neighbor changes 例 : Device(config-router)# bgp log-neighbor changes	BGP ネイバーリセットのロギングを有効にします。
ステップ 14	no bgp default ipv4-unicast 例 : Device(config-router)# no bgp default ipv4-unicast	アドレスファミリー IPv4 のルーティング情報のアドレスバタイズメントを無効にします。
ステップ 15	no bgp default route-target filter 例 : Device(config-router)# no bgp default route-target filter	BGP の route-target コミュニティフィルタリングを無効にします。
ステップ 16	neighbor <i>ip-address</i> remote-as <i>as-number</i> 例 : Device(config-router)# neighbor 4.1.1.3 remote-as 200	エントリを BGP ネイバーテーブルに設定します。
ステップ 17	neighbor <i>ip-address</i> update-source <i>interface-type interface-number</i> 例 : Device(config-router)# neighbor 4.1.1.3 update-source Loopback0	Cisco IOS ソフトウェアで、BGP セッションによる TCP 接続の特定の動作インターフェイスを使用できるようになります。
ステップ 18	neighbor <i>ip-address</i> remote-as <i>as-number</i> 例 : Device(config-router)# neighbor 4.1.1.3 remote-as 300	エントリを BGP ネイバーテーブルに設定します。

	コマンドまたはアクション	目的
ステップ 19	address-family ipv4 例 : Device (config-router) # address-family ipv4	標準 IP バージョン 4 アドレスプレフィックスを使用する BGP ルーティングセッションを設定するために、アドレス ファミリ コンフィギュレーション モードを開始します。
ステップ 20	neighbor ip-address activate 例 : Device (config-router-af) # neighbor 10.32.1.2 activate	BGP ネイバーとの情報交換を有効にします。
ステップ 21	neighbor ip-address send-label 例 : Device (config-router-af) # neighbor 10.32.1.2 send-label	隣接 BGP ルータに BGP ルートを含む MPLS ラベルを送信します。
ステップ 22	exit address-family 例 : Device (config-router-af) # exit address-family	BGP アドレス ファミリ サブモードを終了します。
ステップ 23	address-family vpv4 例 : Device (config-router) # address-family vpv4	アドレス ファミリ コンフィギュレーション モードでデバイスを設定して、標準 VPNv4 アドレスプレフィックスを使用する、BGP などのルーティングセッションを設定します。
ステップ 24	neighbor ip-address activate 例 : Device (config-router-af) # neighbor 4.1.1.3 activate	BGP ネイバーとの情報交換を有効にします。
ステップ 25	neighbor ip-address send-community extended 例 : Device (config-router-af) # neighbor 4.1.1.3 send-community extended	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 26	neighbor ip-address next-hop-self 例 : Device (config-router-af) # neighbor 4.1.1.3 next-hop-self	ルータを BGP スピーキングネイバーのネクストホップとして設定します。これは、ネクストホップセルフ方式を実装するコマンドです。
ステップ 27	neighbor ip-address activate 例 :	BGP ネイバーとの情報交換を有効にします。

	コマンドまたはアクション	目的
	Device(config-router-af) # neighbor 10.30.1.2 activate	
ステップ 28	neighbor ip-address send-community extended 例 : Device(config-router-af) # neighbor 10.30.1.2 send-community extended	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 29	exit address-family 例 : Device(config-router-af) # exit address-family	BGP アドレス ファミリ サブモードを終了します。
ステップ 30	bgp router-id ip-address 例 : Device(config-router) # bgp router-id 4.1.1.3	BGP ルーティングプロセスの固定ルータ ID を設定します。
ステップ 31	bgp log-neighbor changes 例 : Device(config-router) # bgp log-neighbor changes	BGP ネイバーリセットのロギングを有効にします。
ステップ 32	neighbor ip-address remote-as as-number 例 : Device(config-router) # neighbor 4.1.1.1 remote-as 200	エントリを BGP ネイバーテーブルに設定します。
ステップ 33	neighbor ip-address update-source interface-type interface-number 例 : Device(config-router) # neighbor 4.1.1.1 update-source Loopback0	Cisco IOS ソフトウェアで、BGP セッションによる TCP 接続の特定の動作インターフェイスを使用できるようになります。
ステップ 34	address-family vpnv4 例 : Device(config-router) # address-family vpnv4	アドレスファミリ コンフィギュレーションモードでデバイスを設定して、標準 VPNv4 アドレスプレフィックスを使用する、BGP などのルーティングセッションを設定します。
ステップ 35	neighbor ip-address activate 例 : Device(config-router-af) # neighbor 4.1.1.1 activate	BGP ネイバーとの情報交換を有効にします。

	コマンドまたはアクション	目的
ステップ 36	neighbor ip-address send-community extended 例 : Device (config-router-af) # neighbor 4.1.1.1 send-community extended	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 37	exit address-family 例 : Device (config-router-af) # exit address-family	BGP アドレス ファミリ サブモードを終了します。

Redistribute Connected 方式を使用した InterAS オプション B の設定

Redistribute Connected 方式を使用して ASBR で InterAS オプション B を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router ospf process-id**
4. **router-id ip-address**
5. **nsr**
6. **nsf**
7. **redistribute connected**
8. **passive-interface interface-type interface-number**
9. **network ip-address wildcard-mask aread area-id**
10. **exit**
11. **router bgp autonomous-system-number**
12. **bgp router-id ip-address**
13. **bgp log-neighbor changes**
14. **no bgp default ipv4-unicast**
15. **no bgp default route-target filter**
16. **neighbor ip-address remote-as as-number**
17. **neighbor ip-address update-source interface-type interface-number**
18. **neighbor ip-address remote-as as-number**
19. **address-family vpnv4**
20. **neighbor ip-address activate**
21. **neighbor ip-address send-community extended**
22. **neighbor ip-address activate**
23. **neighbor ip-address send-community extended**
24. **exit address-family**
25. **mpls ldp router-id interface-id [force]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router ospf process-id 例： Device(config)# router ospf 1	OSPF ルーティングプロセスを設定し、プロセス番号を割り当てます。
ステップ 4	router-id ip-address 例： Device(config)# router-id 5.1.1.1	固定ルータ ID を指定します。
ステップ 5	nsr 例： Device(config-router)# nsr	OSPF ノンストップルーティング (NSR) を設定します。
ステップ 6	nsf 例： Device(config-router)# nsf	OSPF ノンストップフォワーディング (NSF) を設定します。
ステップ 7	redistribute connected 例： Device(config-router)# redistribute connected	リモート ASBR のネクストホップアドレスをローカル IGP に再配布します。これは、Redistribute Connected 方式を実装するコマンドです。
ステップ 8	passive-interface interface-type interface-number 例： Device(config-router)# passive-interface GigabitEthernet 1/0/10 Device(config-router)# passive-interface Tunnel0	インターフェイスの Open Shortest Path First (OSPF) ルーティングアップデートを無効にします。

	コマンドまたはアクション	目的
ステップ 9	network <i>ip-address wildcard-mask</i> area <i>area-id</i> 例 : Device(config-router)# network 5.1.1.0 0.0.0.0.255 area 0	OSPF を実行するインターフェイスを定義し、そのインターフェイスに対するエリア ID を定義します。
ステップ 10	exit 例 : Device(config-router)# exit	ルータ コンフィギュレーション モードを終了します。
ステップ 11	router bgp <i>autonomous-system-number</i> 例 : Device(config)# router bgp 300	BGP ルーティング プロセスを設定します。
ステップ 12	bgp router-id <i>ip-address</i> 例 : Device(config-router)# bgp router-id 5.1.1.1	BGP ルーティングプロセスの固定ルータ ID を設定します。
ステップ 13	bgp log-neighbor changes 例 : Device(config-router)# bgp log-neighbor changes	BGP ネイバーリセットのログギングを有効にします。
ステップ 14	no bgp default ipv4-unicast 例 : Device(config-router)# no bgp default ipv4-unicast	アドレスファミリ IPv4 のルーティング情報のアドバタイズメントを無効にします。
ステップ 15	no bgp default route-target filter 例 : Device(config-router)# no bgp default route-target filter	BGP の route-target コミュニティフィルタリングを無効にします。
ステップ 16	neighbor <i>ip-address</i> remote-as <i>as-number</i> 例 : Device(config-router)# neighbor 5.1.1.3 remote-as 300	エントリを BGP ネイバーテーブルに設定します。
ステップ 17	neighbor <i>ip-address</i> update-source <i>interface-type</i> <i>interface-number</i> 例 :	Cisco IOS ソフトウェアで、BGP セッションによる TCP 接続の特定の動作インターフェイスを使用できるようにします。

Redistribute Connected 方式を使用した InterAS オプション B の設定

	コマンドまたはアクション	目的
	Device(config-router)# neighbor 4.1.1.3 update-source Loopback0	
ステップ 18	neighbor ip-address remote-as as-number 例： Device(config-router)# neighbor 10.30.1.2 remote-as 200	エントリを BGP ネイバーテーブルに設定します。
ステップ 19	address-family vpnv4 例： Device(config-router)# address-family vpnv4	アドレスファミリー コンフィギュレーションモードでデバイスを設定して、標準 VPNv4 アドレスプレフィックスを使用する、BGP などのルーティングセッションを設定します。
ステップ 20	neighbor ip-address activate 例： Device(config-router-af)# neighbor 5.1.1.3 activate	BGP ネイバーとの情報交換を有効にします。
ステップ 21	neighbor ip-address send-community extended 例： Device(config-router-af)# neighbor 5.1.1.3 send-community extended	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 22	neighbor ip-address activate 例： Device(config-router-af)# neighbor 10.30.1.1 activate	BGP ネイバーとの情報交換を有効にします。
ステップ 23	neighbor ip-address send-community extended 例： Device(config-router-af)# neighbor 10.30.1.2 send-community extended	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 24	exit address-family 例： Device(config-router-af)# exit address-family	BGP アドレスファミリー サブモードを終了します。
ステップ 25	mpls ldp router-id interface-id [force] 例：	LDP ルータ ID を決定する優先インターフェイスを指定します。

コマンドまたはアクション	目的
Device(config-router)# mpls ldp router-id Loopback0 force	

MPLS VPN InterAS オプションの設定の確認

InterAS オプション B の設定情報を確認するには、次のいずれかの作業を行います。

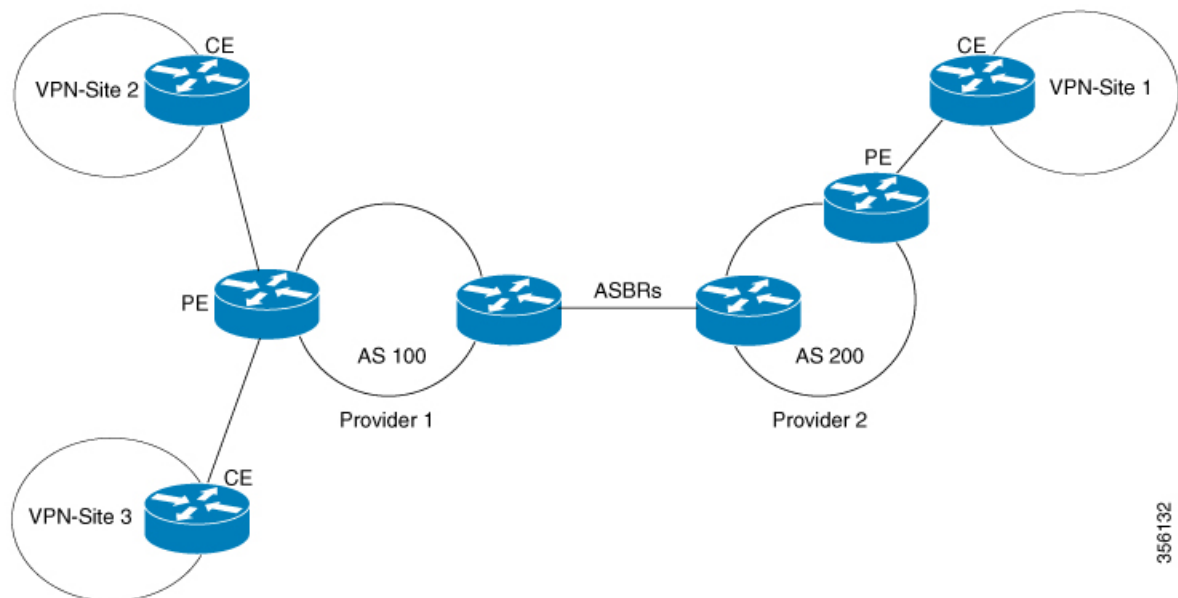
コマンド	目的
ping ip-address source interface-type	デバイスのアクセシビリティをチェックします。ループバック インターフェイスを使用して CE1 と CE2 間の接続を確認するには、このコマンドを使用します。
show bgp vpnv4 unicast labels	着信および発信 BGP ラベルを表示します。
show mpls forwarding-table	MPLS ラベル転送情報ベースの内容を表示します。
show ip bgp	BGP ルーティングテーブル内のエントリを表示します。
show { ip ipv6 } bgp [vrf vrf-name]	VRF での BGP に関する情報を表示します。
show ip route [ip-address [mask]] [protocol] vrf vrf-name	ルーティング テーブルの現在の状態を表示します。ip-address 引数を使用して、CE1 に CE2 へのルートが含まれていることを確認します。CE1 から学習したルートを確認します。CE2 へのルートがリストされていることを確認します。
show { ip ipv6 } route vrf vrf-name	VRF に関連付けられた IP ルーティング テーブルを表示します。ローカル CE ルータとリモート CE ルータのループバックアドレスが、PE ルータのルーティングテーブルに存在することを確認します。
show running-config bgp	BGP の実行コンフィギュレーションを表示します。
show running-config vrf vrf-name	VRF の実行コンフィギュレーションを表示します。
show vrf vrf-name interface interface-type interface-id	VRF に対して設定されるルート識別子 (RD) およびインターフェイスを検証します。

コマンド	目的
<code>trace destination [vrf vrf-name]</code>	パケットがその宛先に送信される時に取るルートを検出します。 trace コマンドは、2つのルータが通信できない場合に問題の箇所を分離するのに役立ちます。

MPLS VPN InterAS オプションの設定例

ネクストホップセルフ方式

図 8: ネクストホップセルフ方式を使用した *InterAS* オプション B のトポロジ



356132

PE1 - P1 - ASBR1 の設定

PE1	P1	ASBR1
	<pre>interface Loopback0 ip address 4.1.1.2 255.255.255.255 ip ospf 1 area 0 interface GigabitEthernet1/0/4 no switchport ip address 10.10.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp ! interface GigabitEthernet1/0/23 no switchport ip address 10.20.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp</pre>	<pre>interface Loopback0 ip address 4.1.1.1 255.255.255.255 ip ospf 1 area 0 interface GigabitEthernet1/0/10 no switchport ip address 10.30.1.1 255.255.255.0 mpls bgp forwarding interface GigabitEthernet1/0/23 no switchport ip address 10.20.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp router ospf 1 router-id 4.1.1.1 nsr nsf redistribute bgp 200 passive-interface GigabitEthernet1/0/10 passive-interface Tunnel0 network 4.1.1.0 0.0.0.255 area 0 router bgp 200 bgp router-id 4.1.1.1 bgp log-neighbor-changes no bgp default ipv4-unicast no bgp default route-target filter neighbor 4.1.1.3 remote-as 200 neighbor 4.1.1.3 update-source Loopback0 neighbor 10.30.1.2 remote-as 300 ! address-family ipv4 neighbor 10.30.1.2 activate neighbor 10.30.1.2 send-label exit-address-family ! address-family vpnv4 neighbor 4.1.1.3 activate neighbor 4.1.1.3 send-community extended neighbor 4.1.1.3 next-hop-self neighbor 10.30.1.2 activate neighbor 10.30.1.2 send-community extended exit-address-family</pre>

PE1	P1	ASBR1
<pre> vrf definition Mgmt-vrf ! address-family ipv4 exit-address-family ! address-family ipv6 exit-address-family ! vrf definition vrf1 rd 200:1 route-target export 200:1 route-target import 200:1 route-target import 300:1 ! address-family ipv4 exit-address-family interface Loopback0 ip address 4.1.1.3 255.255.255.255 ip ospf 1 area 0 ! interface Loopback1 vrf forwarding vrf1 ip address 192.1.1.1 255.255.255.255 ip ospf 200 area 0 ! interface GigabitEthernet2/0/4 no switchport ip address 10.10.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp interface GigabitEthernet2/0/9 description to-IXIA-1:p8 no switchport vrf forwarding vrf1 ip address 192.2.1.1 255.255.255.0 ip ospf 200 area 0 router ospf 200 vrf vrf1 router-id 192.1.1.1 nsr nsf redistribute connected redistribute bgp 200 network 192.1.1.1 0.0.0.0 area 0 network 192.2.1.0 0.0.0.255 area 0 router ospf 1 router-id 4.1.1.3 nsr nsf redistribute connected router bgp 200 bgp router-id 4.1.1.3 bgp log-neighbor-changes neighbor 4.1.1.1 remote-as </pre>		

PE1	P1	ASBR1
<pre>200 neighbor 4.1.1.1 update-source Loopback0 ! address-family vpnv4 neighbor 4.1.1.1 activate neighbor 4.1.1.1 send-community extended exit-address-family ! address-family ipv4 vrf vrf1 redistribute connected redistribute ospf 200 maximum-paths ibgp 4 exit-address-family</pre>		

ASBR2 – P2 – PE2 の設定

表 8:

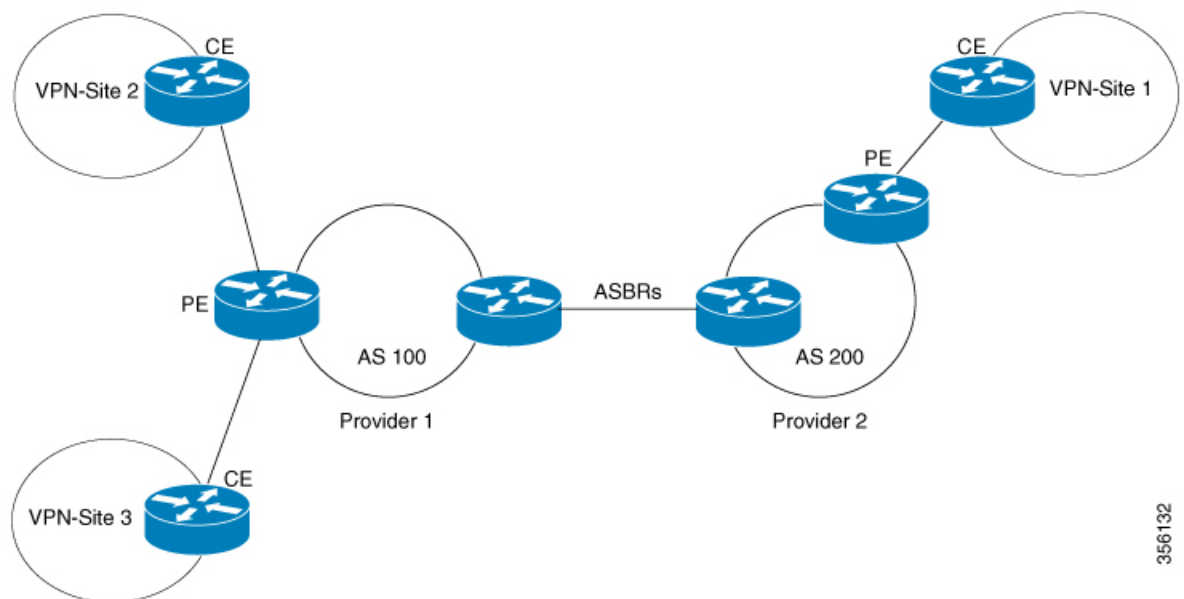
PE2	P2	ASBR2
	<pre>interface Loopback0 ip address 5.1.1.2 255.255.255.255 ip ospf 1 area 0 interface GigabitEthernet1/0/1 no switchport ip address 10.50.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp interface GigabitEthernet2/0/3 no switchport ip address 10.40.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp</pre>	<pre>interface Loopback0 ip address 5.1.1.1 255.255.255.255 ip ospf 1 area 0 ! interface GigabitEthernet1/0/37 no switchport ip address 10.30.1.2 255.255.255.0 mpls bgp forwarding interface GigabitEthernet1/0/47 no switchport ip address 10.40.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp router ospf 1 router-id 5.1.1.1 nsr nsf passive-interface GigabitEthernet1/0/37 passive-interface Tunnel0 network 5.1.1.0 0.0.0.255 area 0 ! router bgp 300 bgp router-id 5.1.1.1 bgp log-neighbor-changes no bgp default ipv4-unicast no bgp default route-target filter neighbor 5.1.1.3 remote-as 300 neighbor 5.1.1.3 update-source Loopback0 neighbor 10.30.1.1 remote-as 200 ! address-family ipv4 neighbor 10.30.1.1 activate neighbor 10.30.1.1 send-label exit-address-family ! address-family vpnv4 neighbor 5.1.1.3 activate neighbor 5.1.1.3 send-community extended neighbor 5.1.1.3 next-hop-self neighbor 10.30.1.1 activate neighbor 10.30.1.1 send-community extended exit-address-family</pre>

PE2	P2	ASBR2
<pre> vrf definition vrf1 rd 300:1 route-target export 300:1 route-target import 300:1 route-target import 200:1 ! address-family ipv4 exit-address-family interface Loopback0 ip address 5.1.1.3 255.255.255.255 ip ospf 1 area 0 ! interface Loopback1 vrf forwarding vrf1 ip address 193.1.1.1 255.255.255.255 ip ospf 300 area 0 interface GigabitEthernet1/0/1 no switchport ip address 10.50.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp ! interface GigabitEthernet1/0/2 no switchport vrf forwarding vrf1 ip address 193.2.1.1 255.255.255.0 ip ospf 300 area 0 router ospf 300 vrf vrf1 router-id 193.1.1.1 nsr nsf redistribute connected redistribute bgp 300 network 193.1.1.1 0.0.0.0 area 0 network 193.2.1.0 0.0.0.255 area 0 ! router ospf 1 router-id 5.1.1.3 nsr nsf redistribute connected router bgp 300 bgp router-id 5.1.1.3 bgp log-neighbor-changes neighbor 5.1.1.1 remote-as 300 neighbor 5.1.1.1 update-source Loopback0 ! address-family ipv4 neighbor 5.1.1.1 activate neighbor 5.1.1.1 send-label exit-address-family </pre>		

PE2	P2	ASBR2
<pre> ! address-family vpv4 neighbor 5.1.1.1 activate neighbor 5.1.1.1 send-community extended exit-address-family ! address-family ipv4 vrf vrf1 redistribute connected redistribute ospf 300 maximum-paths ibgp 4 exit-address-family </pre>		

IGP Redistribute Connected Subnet 方式

図 9: *Redistribute Connected Subnet* 方式を使用した *InterAS* オプション B のトポロジ



356132

PE1 - P1 - ASBR1 の設定

PE1	P1	ASBR1
	<pre> interface Loopback0 ip address 4.1.1.2 255.255.255.255 ip ospf 1 area 0 interface GigabitEthernet1/0/4 no switchport ip address 10.10.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp ! interface GigabitEthernet1/0/23 no switchport ip address 10.20.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp </pre>	<pre> router ospf 1 router-id 4.1.1.1 nsr nsf redistribute connected passive-interface GigabitEthernet1/0/10 passive-interface Tunnel0 network 4.1.1.0 0.0.0.255 area 0 router bgp 200 bgp router-id 4.1.1.1 bgp log-neighbor-changes no bgp default ipv4-unicast no bgp default route-target filter neighbor 4.1.1.3 remote-as 200 neighbor 4.1.1.3 update-source Loopback0 neighbor 10.30.1.2 remote-as 300 ! address-family vpnv4 neighbor 4.1.1.3 activate neighbor 4.1.1.3 send-community extended neighbor 10.30.1.2 activate neighbor 10.30.1.2 send-community extended exit-address-family mpls ldp router-id Loopback0 force </pre>

PE1	P1	ASBR1
<pre> vrf definition Mgmt-vrf ! address-family ipv4 exit-address-family ! address-family ipv6 exit-address-family ! vrf definition vrf1 rd 200:1 route-target export 200:1 route-target import 200:1 route-target import 300:1 ! address-family ipv4 exit-address-family interface Loopback0 ip address 4.1.1.3 255.255.255.255 ip ospf 1 area 0 ! interface Loopback1 vrf forwarding vrf1 ip address 192.1.1.1 255.255.255.255 ip ospf 200 area 0 ! interface GigabitEthernet2/0/4 no switchport ip address 10.10.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp interface GigabitEthernet2/0/9 description to-IXIA-1:p8 no switchport vrf forwarding vrf1 ip address 192.2.1.1 255.255.255.0 ip ospf 200 area 0 router ospf 200 vrf vrf1 router-id 192.1.1.1 nsr nsf redistribute connected redistribute bgp 200 network 192.1.1.1 0.0.0.0 area 0 network 192.2.1.0 0.0.0.255 area 0 router ospf 1 router-id 4.1.1.3 nsr nsf redistribute connected router bgp 200 bgp router-id 4.1.1.3 bgp log-neighbor-changes neighbor 4.1.1.1 remote-as </pre>		

PE1	P1	ASBR1
<pre>200 neighbor 4.1.1.1 update-source Loopback0 ! address-family vpnv4 neighbor 4.1.1.1 activate neighbor 4.1.1.1 send-community extended exit-address-family ! address-family ipv4 vrf vrf1 redistribute connected redistribute ospf 200 maximum-paths ibgp 4 exit-address-family</pre>		

ASBR2 – P2 – PE2 の設定

PE2	P2	ASBR2
	<pre> interface Loopback0 ip address 5.1.1.2 255.255.255.255 ip ospf 1 area 0 interface GigabitEthernet1/0/1 no switchport ip address 10.50.1.1 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp interface GigabitEthernet2/0/3 no switchport ip address 10.40.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp </pre>	<pre> router ospf 1 router-id 5.1.1.1 nsr nsf redistribute connected passive-interface GigabitEthernet1/0/10 passive-interface Tunnel0 network 5.1.1.0 0.0.0.255 area 0 router bgp 300 bgp router-id 5.1.1.1 bgp log-neighbor-changes no bgp default ipv4-unicast no bgp default route-target filter neighbor 5.1.1.3 remote-as 300 neighbor 5.1.1.3 update-source Loopback0 neighbor 10.30.1.1 remote-as 200 ! address-family vpnv4 neighbor 5.1.1.3 activate neighbor 5.1.1.3 send-community extended neighbor 10.30.1.1 activate neighbor 10.30.1.1 send-community extended exit-address-family mpls ldp router-id Loopback0 force </pre>

PE2	P2	ASBR2
<pre> vrf definition vrf1 rd 300:1 route-target export 300:1 route-target import 300:1 route-target import 200:1 ! address-family ipv4 exit-address-family interface Loopback0 ip address 5.1.1.3 255.255.255.255 ip ospf 1 area 0 ! interface Loopback1 vrf forwarding vrf1 ip address 193.1.1.1 255.255.255.255 ip ospf 300 area 0 interface GigabitEthernet1/0/1 no switchport ip address 10.50.1.2 255.255.255.0 ip ospf 1 area 0 mpls ip mpls label protocol ldp ! interface GigabitEthernet1/0/2 no switchport vrf forwarding vrf1 ip address 193.2.1.1 255.255.255.0 ip ospf 300 area 0 router ospf 300 vrf vrf1 router-id 193.1.1.1 nsr nsf redistribute connected redistribute bgp 300 network 193.1.1.1 0.0.0.0 area 0 network 193.2.1.0 0.0.0.255 area 0 ! router ospf 1 router-id 5.1.1.3 nsr nsf redistribute connected router bgp 300 bgp router-id 5.1.1.3 bgp log-neighbor-changes neighbor 5.1.1.1 remote-as 300 neighbor 5.1.1.1 update-source Loopback0 ! address-family ipv4 neighbor 5.1.1.1 activate neighbor 5.1.1.1 send-label </pre>		

PE2	P2	ASBR2
<pre> exit-address-family ! address-family vpnv4 neighbor 5.1.1.1 activate neighbor 5.1.1.1 send-community extended exit-address-family ! address-family ipv4 vrf vrf1 redistribute connected redistribute ospf 300 maximum-paths ibgp 4 exit-address-family </pre>		

MPLS VPN InterAS オプションに関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	『 <i>Command Reference (Catalyst 9300 Series Switches)</i> 』の「MPLS コマンド」の項を参照してください。

MPLS VPN InterAS オプションの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.11.1	MPLS VPN InterAS オプション B	InterAS オプションは、iBGP および eBGP ピアリングを使用して、異なる AS 内の VPN が相互に通信できるようにします。InterAS オプション B ネットワークでは、ASBR ポートは、MPLS トラフィックを受信できる 1 つ以上のインターフェイスによって接続されます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 9 章

MPLS over GRE の設定

- [MPLS over GRE の前提条件](#) (131 ページ)
- [GRE を介した MPLS の制約事項](#) (131 ページ)
- [MPLS over GRE に関する情報](#) (132 ページ)
- [GRE を介した MPLS の設定方法](#) (134 ページ)
- [MPLS over GRE の設定例](#) (135 ページ)
- [MPLS over GRE に関するその他の参考資料](#) (139 ページ)
- [MPLS over GRE の機能情報](#) (139 ページ)

MPLS over GRE の前提条件

次のルーティングプロトコルが正しく設定され、動作していることを確認します。

- ラベル配布プロトコル (LDP) : MPLS ラベル配布の場合。
- コアデバイス P1-P2 間のルーティングプロトコル (ISIS または OSPF)
- PE1-P1 と PE2-P2 間の MPLS
- 入力トラフィックは MPLS ネットワークから IP コアに入り、出力トラフィックは IP コアを出て MPLS ネットワークに入るため、プロトコル境界を通過するときに QoS グループ値を使用して QoS ポリシーを定義することをお勧めします。

GRE を介した MPLS の制約事項

- GRE トンネリング :
 - L2VPN over mGRE および L3VPN over mGRE はサポートされていません。
 - トンネル送信元は、ループバックインターフェイスまたはレイヤ3インターフェイスにのみできます。これらのインターフェイスは、物理インターフェイスまたは EtherChannel のいずれかです。

- トンネルインターフェイスは、スタティックルート、Enhanced Interior Gateway Routing Protocol (EIGRP)、および Open Shortest Path First (OSPF) ルーティングプロトコルをサポートしています。
- GRE オプション：シーケンシング、チェックサム、およびソースルートはサポートされていません。
- IPv6 Generic Routing Encapsulation (GRE) はサポートされていません。
- Carrier Supporting Carrier (CSC) はサポートされていません。

MPLS over GRE に関する情報

MPLS over GRE 機能は、非 MPLS ネットワーク経由でマルチプロトコル ラベル スイッチング (MPLS) パケットのトンネリングを行うためのメカニズムを提供します。この機能を使用すると、非 MPLS ネットワーク間の Generic Routing Encapsulation (GRE) トンネルを作成できます。MPLS パケットは、GRE トンネルパケット内でカプセル化され、カプセル化されたパケットは、GRE トンネルを経由して非 MPLS ネットワークを通ります。GRE トンネルパケットを非 MPLS ネットワークの反対側で受信すると、GRE トンネルパケット ヘッダーが削除され、内部の MPLS パケットが最終的な宛先に転送されます。GRE トンネルのエンドポイント間のコアネットワークは ISIS または OSPF ルーティングプロトコルを使用しますが、GRE トンネルは OSPF または EIGRP を使用します。

PE-to-PE トンネリング

プロバイダーエッジ間 (PE-to-PE) トンネリング設定によって、非 MPLS ネットワーク間の複数のカスタマーネットワークをスケーラブルな方法で接続できます。この設定を使用して、複数のカスタマーネットワーク宛のトラフィックは、単一の Generic Routing Encapsulation (GRE) トンネルから多重化されます。



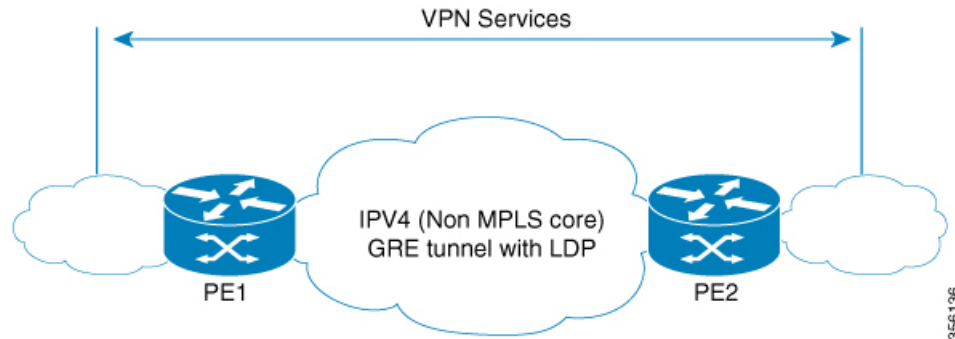
- (注) 類似したスケーラブルではない代替方法は、別個の GRE トンネルから各カスタマーネットワークに接続することです (たとえば、1つのカスタマー ネットワークを各 GRE トンネルに接続します)。

非 MPLS ネットワークのいずれかの側にある PE デバイスは、(非 MPLS ネットワーク内で動作している) ルーティングプロトコルを使用して、非 MPLS ネットワークのもう一方の側にある PE デバイスについて学習します。PE デバイス間に確立された学習ルートは、メインまたはデフォルトのルーティング テーブルに格納されます。

反対方向の PE デバイスは、OSPF または EIGRP を使用して、PE デバイスの背後にあるカスタマーネットワークに関連付けられたルートについて学習します。これらの学習ルートは、非 MPLS ネットワークには認識されません。

次の図は、非 MPLS ネットワークにまたがる GRE トンネルを介した、ある PE デバイスから別の PE デバイスへのエンドツーエンド IP コアを示しています。

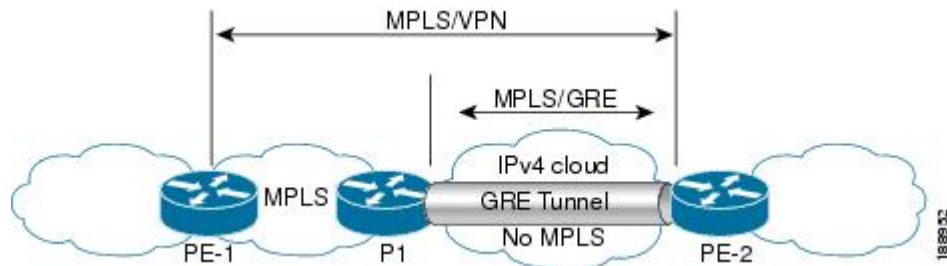
図 10: PE-to-PE トンネリング



P-to-PE トンネリング

Provider-to-Provider Edge (P-to-PE) トンネリング設定によって、非マルチプロトコルラベルスイッチング (MPLS) ネットワークで PE デバイス (P1) を MPLS セグメント (PE-2) に接続できます。この設定では、非 MPLS ネットワークの一方の側宛の MPLS トラフィックは、単一の Generic Routing Encapsulation (GRE) トンネル経由で送信されます。

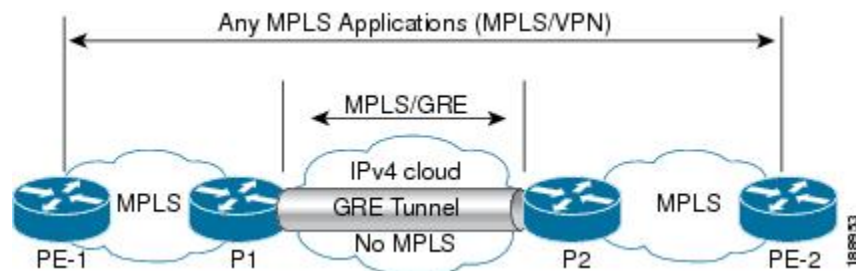
図 11: P-to-PE トンネリング



P-to-P トンネリング

下図に示すように、Provider-to-Provider (P-to-P) 設定によって、非マルチプロトコルラベルスイッチング (MPLS) ネットワークで 2 つの MPLS セグメント (P1 から P2) を接続できます。この設定では、非 MPLS ネットワークの一方の側宛の MPLS トラフィックは、単一の Generic Routing Encapsulation (GRE) トンネル経由で送信されます。

図 12: P-to-P トンネリング



GRE を介した MPLS の設定方法

次の項では、GRE を介した MPLS のさまざまな設定手順について説明します。

MPLS over GRE トンネル インターフェイスの設定

MPLS over GRE 機能を設定するには、非 MPLS ネットワークにまたがる GRE トンネルを作成する必要があります。次の手順は、GRE トンネルの両方の終端にあるデバイスで実行する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface tunnel *tunnel-number***
4. **ip address *ip-address mask***
5. **tunnel source *source-address***
6. **tunnel destination *destination-address***
7. **mpls ip**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	interface tunnel <i>tunnel-number</i> 例 : Device(config)# interface tunnel 1	トンネルインターフェイスを作成します。続いて、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	ip address <i>ip-address mask</i> 例 : Device(config-if)# ip address 10.0.0.1 255.255.255.0	トンネル インターフェイスに IP アドレスを割り当てます。
ステップ 5	tunnel source <i>source-address</i> 例 : Device(config-if)# tunnel source 10.1.1.1	トンネル送信元 IP アドレスを指定します。
ステップ 6	tunnel destination <i>destination-address</i> 例 : Device(config-if)# tunnel destination 10.1.1.2	トンネル宛先 IP アドレスを指定します。
ステップ 7	mpls ip 例 : Device(config-if)# mpls ip	トンネルの物理インターフェイスでマルチプロトコルラベルスイッチング (MPLS) を有効にします。
ステップ 8	end 例 : Device(config-if)# end	特権 EXEC モードに戻ります。

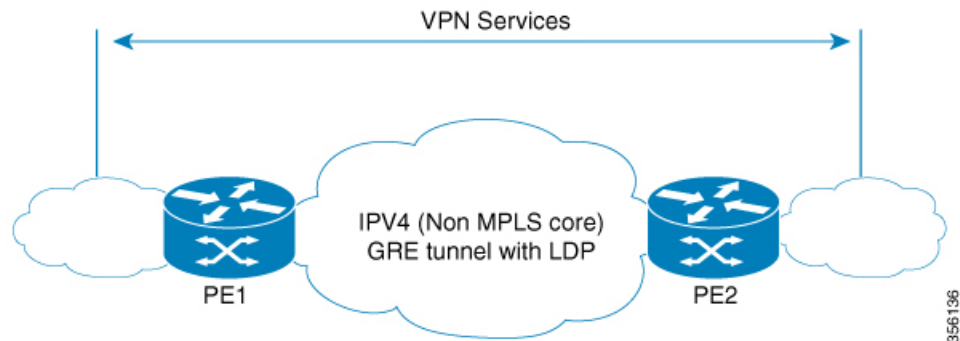
MPLS over GRE の設定例

次の項では、GRE を介した MPLS のさまざまな設定例について説明します。

例 : PE-to-PE トンネリング

次に、2つのプロバイダーエッジ (PE) デバイスでの基本的な MPLS 設定を示します。PE-to-PE トンネリングは、GRE トンネルを使用して非 MPLS ネットワーク経由でトラフィックを送信します。

図 13: PE-to-PE トンネリングのトポロジ



PE1 の設定

```

!
mpls ip
!
interface loopback 10
ip address 11.2.2.2 255.255.255.255
ip router isis
!
interface GigabitEthernet 1/1/1
ip address 1.1.1.1 255.255.255.0
ip router isis
!
interface Tunnel 1
ip address 10.0.0.1 255.255.255.0
ip ospf 1 are 0
tunnel source 11.2.2.2
tunnel destination 11.1.1.1
mpls ip
!
interface Vlan701
ip address 65.1.1.1 255.255.255.0
ip ospf 1 area 0
!

```

PE2 の設定

```

!
mpls ip
!
interface loopback 10
ip address 11.1.1.1 255.255.255.255
ip router isis
!
interface GigabitEthernet 1/1/1
ip address 2.1.1.1 255.255.255.0
ip router isis
!
interface Tunnel 1
ip address 10.0.0.2 255.255.255.0
ip ospf 1 are 0
tunnel source 11.1.1.1
tunnel destination 11.2.2.2
mpls ip

```

```

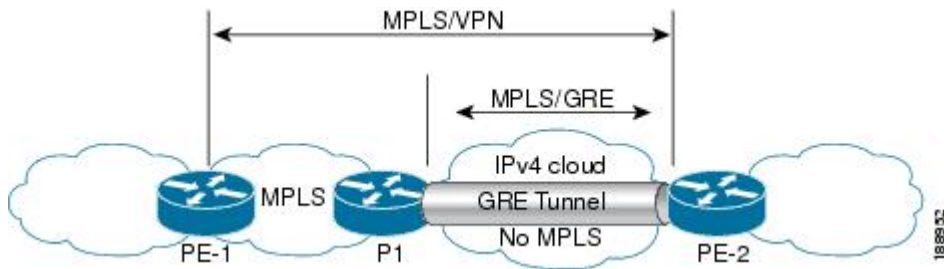
!
interface Vlan701
ip address 75.1.1.1 255.255.255.0
ip ospf 1 area 0
!

```

例：P-to-PE トンネリング

次に、2つのプロバイダー（P）デバイス（P-to-PE トンネリング）での基本的な MPLS 設定を示します。P-to-PE トンネリングでは、GRE トンネルを使用して非 MPLS ネットワーク経由でトラフィックが送信されます。

図 14: P-to-PE トンネリングのトポロジ



PE1 の設定

```

!
mpls ip
!
interface GigabitEthernet 1/1/1
ip address 3.1.1.2 255.255.255.0
ip ospf 1 are 0
mpls ip
!
interface Vlan701
ip address 75.1.1.1 255.255.255.0
ip ospf 1 area 0
!

```

P1 の設定

```

!
mpls ip
!
interface loopback 10
ip address 11.2.2.2 255.255.255.255
ip router isis
!
interface GigabitEthernet 1/1/1
ip address 1.1.1.1 255.255.255.0
ip router isis
!
interface GigabitEthernet 1/1/2
ip address 3.1.1.1 255.255.255.0
ip ospf 1 are 0
mpls ip
!

```

例 : P-to-P トンネリング

```

interface Tunnel 1
ip address 10.0.0.1 255.255.255.0
ip ospf 1 are 0
tunnel source 11.2.2.2
tunnel destination 11.1.1.1
mpls ip
!
```

PE2 の設定

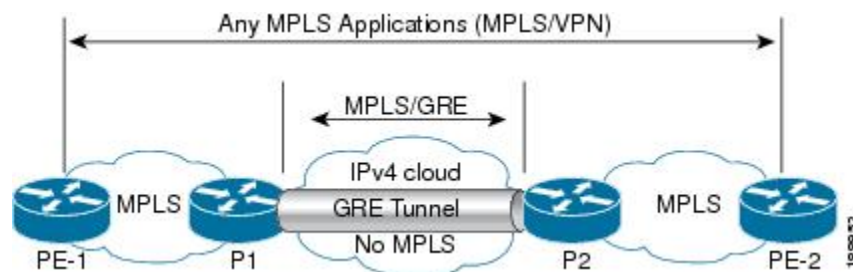
```

!
mpls ip
!
interface loopback 10
ip address 11.1.1.1 255.255.255.255
ip router isis
!
interface GigabitEthernet 1/1/1
ip address 2.2.1.1 255.255.255.0
ip router isis
!
interface Tunnel 1
ip address 10.0.0.2 255.255.255.0
ip ospf 1 are 0
tunnel source 11.1.1.1
tunnel destination 11.2.2.2
mpls ip
!
interface Vlan701
ip address 75.1.1.1 255.255.255.0
ip ospf 1 area 0
!
```

例 : P-to-P トンネリング

次に、2つのプロバイダー (P) デバイス (P-to-PE トンネリング) での基本的な MPLS 設定の例を示します。P-to-PE トンネリングでは、GRE トンネルを使用して非 MPLS ネットワーク経由でトラフィックが送信されます。

図 15: P-to-P トンネリングのトポロジ



P1 の設定

```

!
interface Loopback10
ip address 10.1.1.1 255.255.255.255
```



```

ip router isis
!
interface Tunnell0
ip address 10.10.10.1 255.255.255.252
ip ospf 1 area 0
mpls ip
tunnel source 10.1.1.1
tunnel destination 10.2.1.1

```

P2 の設定

```

!
interface Tunnell0
ip address 10.10.10.2 255.255.255.252
ip ospf 1 area 0
mpls ip
tunnel source 10.2.1.1
tunnel destination 10.1.1.1
!
interface Loopback10
ip address 10.2.1.1 255.255.255.255
ip router isis

```

MPLS over GRE に関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	『 <i>Command Reference (Catalyst 9300 Series Switches)</i> 』の「MPLS コマンド」の項を参照してください。

MPLS over GRE の機能情報

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

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

表 9: MPLS over GRE の機能情報

機能名	リリース	変更内容
MPLS over GRE	Cisco IOS XE Gibraltar 16.11.1	<p>GRE を介した MPLS 機能は、Generic Routing Encapsulation (GRE) トンネルを作成することで、非 MPLS ネットワーク経由でマルチプロトコル ラベル スイッチング (MPLS) パケットのトンネリングを行うためのメカニズムを提供します。MPLS パケットは、GRE トンネル パケット内でカプセル化され、カプセル化されたパケットは、GRE トンネルを経由して非 MPLS ネットワークを通ります。GRE トンネル パケットを非 MPLS ネットワークの反対側で受信すると、GRE トンネル パケット ヘッダーが削除され、内部の MPLS パケットが最終的な宛先に転送されます。</p>



第 10 章

GRE を介した MPLS レイヤ 2 VPN の設定

- [GRE を介した MPLS レイヤ 2 VPN に関する情報](#) (141 ページ)
- [GRE を介した MPLS レイヤ 3 VPN の設定方法](#) (143 ページ)
- [GRE を介した MPLS レイヤ 2 VPN の設定例](#) (144 ページ)
- [GRE を介した MPLS レイヤ 2 VPN の設定に関するその他の参考資料](#) (145 ページ)
- [GRE を介した MPLS レイヤ 2 VPN の設定の機能情報](#) (145 ページ)

GRE を介した MPLS レイヤ 2 VPN に関する情報

GRE を介した MPLS レイヤ 2 VPN 機能は、非 MPLS ネットワーク経由でマルチプロトコル ラベルスイッチング (MPLS) パケットのトンネリングを行うためのメカニズムを提供します。この機能を使用すると、非 MPLS ネットワーク間の Generic Routing Encapsulation (GRE) トンネルを作成できます。MPLS パケットは、GRE トンネルパケット内でカプセル化され、カプセル化されたパケットは、GRE トンネルを経由して非 MPLS ネットワークを通ります。GRE トンネルパケットを非 MPLS ネットワークの反対側で受信すると、GRE トンネルパケットヘッダーが削除され、内部の MPLS パケットが最終的な宛先に転送されます。

GRE を介した MPLS レイヤ 2 VPN を設定するには、仮想プライベート LAN サービス (VPLS) または EoMPLS (Ethernet over MPLS) を設定する必要があります。

トンネリング設定のタイプ

次の項では、サポートされているさまざまなタイプのトンネリング設定について説明します。

PE-to-PE トンネリング

プロバイダーエッジ間 (PE-to-PE) トンネリング設定によって、非 MPLS ネットワーク間の複数のカスタマーネットワークをスケーラブルな方法で接続できます。この設定を使用して、複数のカスタマーネットワーク宛のトラフィックは、単一の GRE トンネルから多重化されます。

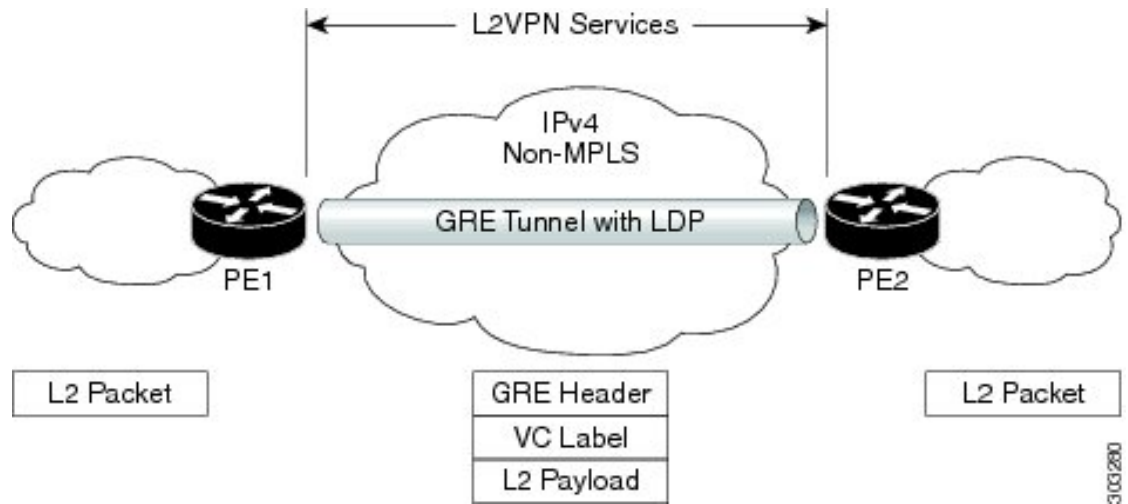
非 MPLS ネットワークのいずれかの側にある PE デバイスは、(非 MPLS ネットワーク内で動作している) ルーティングプロトコルを使用して、非 MPLS ネットワークのもう一方の側に

ある PE デバイスについて学習します。PE デバイス間に確立された学習ルートは、メインまたはデフォルトのルーティング テーブルに格納されます。

反対方向の PE デバイスは、ボーダー ゲートウェイ プロトコル (BGP) を使用して、PE デバイスの背後にあるカスタマーネットワークに関連付けられたルートについて学習します。これらの学習ルートは、非 MPLS ネットワークには認識されません。

図 16: PE-to-PE トンネリング (142 ページ) は、非 MPLS ネットワークにまたがる GRE トンネルを介した、PE デバイス間のエンドツーエンド IP コアを示しています。

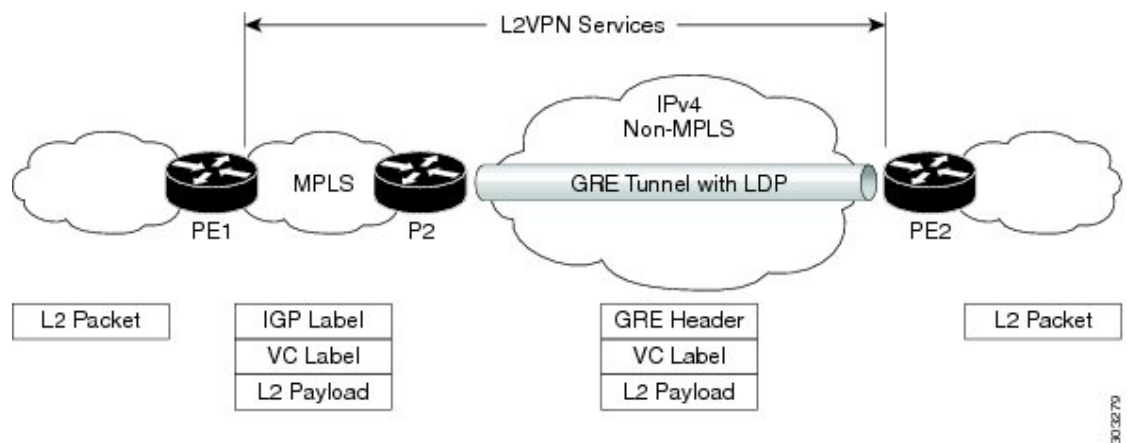
図 16: PE-to-PE トンネリング



P-to-PE トンネリング

図 17: P-to-PE トンネリング (142 ページ) に、非 MPLS ネットワーク上で 2 つの MPLS セグメント (P2 から PE2) を接続する方法を示します。この設定では、非 MPLS ネットワークの一方の側宛の MPLS トラフィックは、単一の GRE トンネル経由で送信されます。

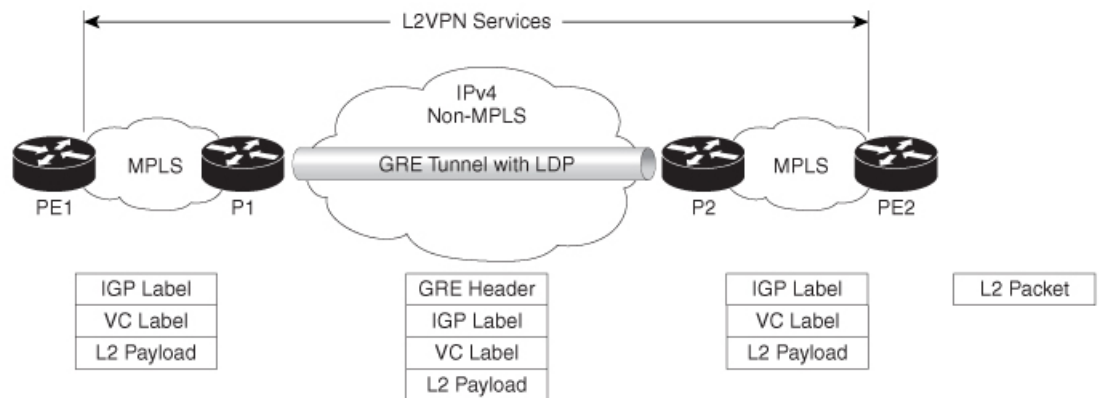
図 17: P-to-PE トンネリング



P-to-P トンネリング

図 18: P-to-P トンネリング (143 ページ) に、非 MPLS ネットワーク上で 2 つの MPLS セグメント (P1 ~ P2) を接続する方法を示します。この設定では、非 MPLS ネットワークの一方の側宛の MPLS トラフィックは、単一の GRE トンネル経由で送信されます。

図 18: P-to-P トンネリング



356234

GRE を介した MPLS レイヤ 3 VPN の設定方法

GRE を介した MPLS 機能を設定するには、非 MPLS ネットワークにまたがる GRE トンネルを作成する必要があります。GRE トンネルの両端にあるデバイスで、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface tunnel tunnel-number 例： Device(config)# interface tunnel 1	トンネルインターフェイスを作成します。続いて、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	ip address ip-address mask 例： Device(config-if)# ip address 10.0.0.1 255.255.255.0	トンネル インターフェイスに IP アドレスを割り当てます。

	コマンドまたはアクション	目的
ステップ 5	tunnel source <i>source-address</i> 例： Device(config-if)# tunnel source 10.1.1.1	トンネルの送信元 IP アドレスを設定します。
ステップ 6	tunnel destination <i>destination-address</i> 例： Device(config-if)# tunnel destination 10.1.1.2	トンネルの宛先 IP アドレスを設定します。
ステップ 7	mpls ip 例： Device(config-if)# mpls ip	トンネルの物理インターフェイスでの MPLS を有効にします。
ステップ 8	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。

GRE を介した MPLS レイヤ 2 VPN の設定例

次の項では、GRE を介した MPLS レイヤ 2 VPN の設定例を示します。

例：非 MPLS ネットワークにまたがる GRE トンネルの設定

次に、非 MPLS ネットワークにまたがる汎用的な GRE トンネルの設定例を示します。

次に、PE1 デバイスのトンネルの設定例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel 1
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# tunnel source 10.0.0.1
Device(config-if)# tunnel destination 10.0.0.2
Device(config-if)# ip ospf 1 area 0
Device(config-if)# mpls ip
```

次に、PE2 デバイスのトンネルの設定例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel 1
Device(config-if)# ip address 10.1.1.2 255.255.255.0
Device(config-if)# tunnel source 10.0.0.2
Device(config-if)# tunnel destination 10.0.0.1
Device(config-if)# ip ospf 1 area 0
Device(config-if)# mpls ip
```

GRE を介した MPLS レイヤ 2 VPN の設定に関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
VPLS の設定	詳細については、「VPLSに関する情報」を参照してください。
Ethernet-over-MPLS (EoMPLS) および疑似回線冗長性 (PWR) の設定	詳細については、次を参照してください。 Ethernet-over-MPLS の設定方法 (54 ページ)

GRE を介した MPLS レイヤ 2 VPN の設定の機能情報

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

表 10: GRE を介した MPLS レイヤ 2 VPN の設定に関する機能情報

機能名	リリース	機能情報
GRE を介した MPLS レイヤ 2 VPN	Cisco IOS XE Gibraltar 16.12.1	GRE を介した MPLS レイヤ 2 VPN 機能は、非 MPLS ネットワーク経由で MPLS パケットのトンネリングを行うためのメカニズムを提供します。



第 11 章

GRE を介した MPLS レイヤ 3 VPN の設定

- [GRE を介した MPLS レイヤ 3 VPN の前提条件](#) (147 ページ)
- [GRE を介した MPLS レイヤ 3 VPN の制約事項](#) (148 ページ)
- [GRE を介した MPLS レイヤ 3 VPN に関する情報](#) (148 ページ)
- [GRE を介した MPLS レイヤ 3 VPN の設定方法](#) (150 ページ)
- [GRE を介した MPLS レイヤ 3 VPN の設定例](#) (151 ページ)
- [GRE を介した MPLS レイヤ 3 VPN の設定の機能情報](#) (157 ページ)

GRE を介した MPLS レイヤ 3 VPN の前提条件

- マルチプロトコル ラベル スイッチング (MPLS) バーチャルプライベート ネットワーク (VPN) が設定されていることを確認します。
- 次のルーティングプロトコルが設定されていることを確認します。
 - Label Distribution Protocol (LDP; ラベル配布プロトコル) : MPLS ラベル配布の場合。
 - マルチプロトコル ボーダー ゲートウェイ プロトコル (MP-BGP) : VPN ルートとラベル配布の場合。
- プロトコル境界を横断する QoS ポリシーを定義するには、Quality of Service (QoS) グループ値を使用することを推奨します。入力トラフィックは MPLS ネットワークから IP コアに入り、出力トラフィックは IP コアを出て MPLS ネットワークに入るため、QoS グループ値が必要です。
- Generic Routing Encapsulation (GRE) トンネルを設定する前に、IP アドレスを指定してループバック インターフェイス (Virtual Routing Forwarding (VRF) に接続されていない) インターフェイスを設定します。IPv4 アドレスを持つこのダミー ループバック インターフェイスは、IPv4 転送用に内部で作成されたトンネル インターフェイスを有効にします。VRF に接続されておらず IPv4 アドレスが設定されているインターフェイスがシステムに 1 つ以上ある場合は、ループバック インターフェイスを設定する必要はありません。

GRE を介した MPLS レイヤ 3 VPN の制約事項

GRE を介した MPLS レイヤ 3 VPN 機能では、次のものはサポートされません。

- トンネルインターフェイスに設定されている QoS サービスポリシー



(注) トンネルインターフェイスに設定されている QoS サービスポリシーはサポートされませんが、物理インターフェイスまたはサブインターフェイスに設定されている QoS サービスポリシーはサポートされます。

- シーケンシング、チェックサム、およびソースルートなどの GRE オプション
- IPv6 GRE の設定
- Carrier Supporting Carrier (CSC) などの拡張機能

GRE を介した MPLS レイヤ 3 VPN に関する情報

GRE を介した MPLS レイヤ 3 VPN 機能は、非 MPLS ネットワーク経由で MPLS パケットのトンネリングを行うためのメカニズムを提供します。この機能を使用すると、非 MPLS ネットワーク間の GRE トンネルを作成できます。MPLS パケットは、GRE トンネル packets 内でカプセル化され、カプセル化されたパケットは、GRE トンネルを経由して非 MPLS ネットワークを通ります。GRE トンネル packets を非 MPLS ネットワークの反対側で受信すると、GRE トンネル packets ヘッダーが削除され、内部の MPLS パケットが最終的な宛先に転送されます。

トンネリング設定のタイプ

次の項では、サポートされているさまざまなタイプのトンネリング設定について説明します。

PE-to-PE トンネリング

プロバイダーエッジ間 (PE-to-PE) トンネリング設定によって、非 MPLS ネットワーク間の複数のカスタマーネットワークをスケーラブルな方法で接続できます。この設定を使用して、複数のカスタマーネットワーク宛のトラフィックは、単一の GRE トンネルから多重化されます。

図 19: PE-to-PE トンネリング (149 ページ) に示すように、PE デバイスは、VRF 番号を非 MPLS ネットワークの各側にあるカスタマーエッジ (CE) デバイスに割り当てます。

PE デバイスは、ボーダーゲートウェイプロトコル (BGP)、Open Shortest Path First (OSPF)、または Routing Information Protocol (RIP) などのルーティングプロトコルを、CE デバイスの

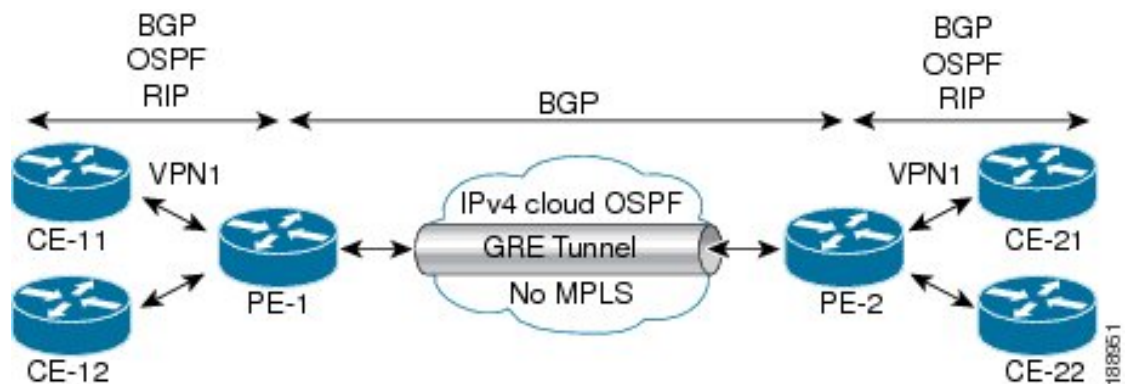
背後にある IP ネットワークを学習するために使用します。CE デバイスの背後にある IP ネットワークへのルートは、関連する CE デバイスの VRF ルーティングテーブルに格納されます。

非 MPLS ネットワークの一方の側にある PE デバイスは（非 MPLS ネットワーク内で動作している）ルーティングプロトコルを使用して、非 MPLS ネットワークのもう一方の側にある PE デバイスについて学習します。PE デバイス間に確立された学習ルートは、メインまたはデフォルトのルーティングテーブルに格納されます。

反対方向の PE デバイスは、BGP を使用して、PE デバイスの背後にあるカスタマーネットワークに関連付けられたルートについて学習します。これらの学習ルートは、非 MPLS ネットワークには認識されません。

図 19: PE-to-PE トンネリング (149 ページ) は、非 MPLS ネットワークにまたがる GRE トンネル経由で BGP ネイバー（反対方向の PE デバイス）へのスタティックルートを定義する BGP を示しています。BGP ネイバーによって学習されたルートには GRE トンネルのネクストホップが含まれているため、すべてのカスタマーネットワークトラフィックが GRE トンネルを使用して送信されます。

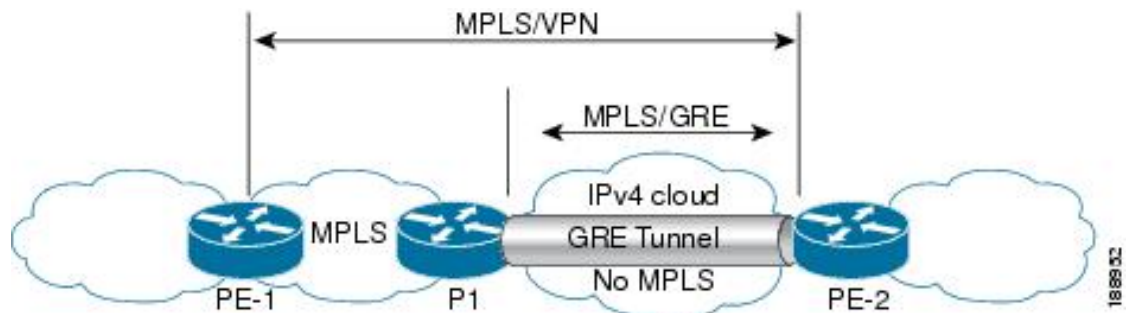
図 19: PE-to-PE トンネリング



P-to-PE トンネリング

図 20: P-to-PE トンネリング (149 ページ) に、非 MPLS ネットワーク上で 2 つの MPLS セグメント (P2 から PE2) を接続する方法を示します。この設定では、非 MPLS ネットワークの一方の側宛の MPLS トラフィックは、単一の GRE トンネル経由で送信されます。

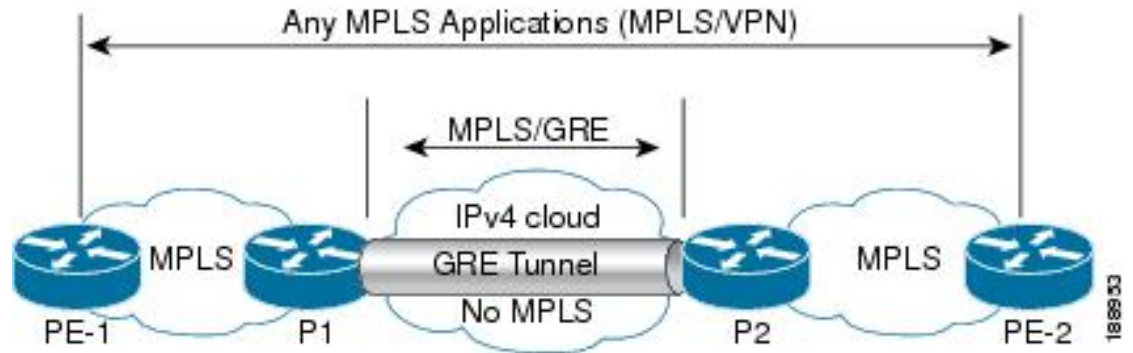
図 20: P-to-PE トンネリング



P-to-P トンネリング

図 21 : P-to-P トンネリング (150 ページ) に、非 MPLS ネットワーク上で 2 つの MPLS セグメント (P1 ~ P2) を接続する方法を示します。この設定では、非 MPLS ネットワークの一方の側宛の MPLS トラフィックは、単一の GRE トンネル経由で送信されます。

図 21 : P-to-P トンネリング



GRE を介した MPLS レイヤ 3 VPN の設定方法

GRE を介した MPLS 機能を設定するには、非 MPLS ネットワークにまたがる GRE トンネルを作成する必要があります。GRE トンネルの両端にあるデバイスで、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface tunnel tunnel-number 例 : Device(config)# interface tunnel 1	トンネルインターフェイスを作成します。続いて、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	ip address ip-address mask 例 : Device(config-if)# ip address 10.0.0.1 255.255.255.0	トンネル インターフェイスに IP アドレスを割り当てます。

	コマンドまたはアクション	目的
ステップ 5	tunnel source <i>source-address</i> 例： Device(config-if)# tunnel source 10.1.1.1	トンネルの送信元 IP アドレスを設定します。
ステップ 6	tunnel destination <i>destination-address</i> 例： Device(config-if)# tunnel destination 10.1.1.2	トンネルの宛先 IP アドレスを設定します。
ステップ 7	mpls ip 例： Device(config-if)# mpls ip	トンネルの物理インターフェイスでの MPLS を有効にします。
ステップ 8	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。

GRE を介した MPLS レイヤ 3 VPN の設定例

次の項では、GRE を介した MPLS レイヤ 3 VPN のさまざまな設定例を示します。

例：GRE を介した MPLS レイヤ 3 VPN（PE-to-PE トンネリング）の設定

次に、PE1 から PE2 へのレイヤ 3 VPN および GRE トンネルを設定する例を示します（[図 19：PE-to-PE トンネリング（149 ページ）](#) を参照）。

次に、PE1 にループバック インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Loopback10
Device(config-if)# ip address 209.165.200.225 255.255.255.255
Device(config-if)# end
```

次に、PE2 にループバック インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Loopback3
Device(config-if)# ip address 209.165.202.129 255.255.255.255
Device(config-if)# end
```

次に、PE1 の IGP でループバックをアドバタイズする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# router ospf 10
```

例：GRE を介した MPLS レイヤ 3 VPN (PE-to-PE トンネリング) の設定

```
Device(config-router)# router-id 198.51.100.10
Device(config-router)# end
```

次に、GRE トンネルを設定し、トンネルで異なる IGP インスタンスを設定し、PE1 のトンネルで MPLS を有効にする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel13
Device(config-if)# ip address 203.0.113.200 255.255.255.248
Device(config-if)# ip ospf 11 area 0
Device(config-if)# mpls ip
Device(config-if)# tunnel source 209.165.200.225
Device(config-if)# tunnel destination 209.165.202.129
Device(config-if)# end
```

次に、GRE トンネルを設定し、トンネルで異なる IGP インスタンスを設定し、PE2 のトンネルで MPLS を有効にする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel31
Device(config-if)# ip address 203.0.113.201 255.255.255.248
Device(config-if)# ip ospf 11 area 0
Device(config-if)# mpls ip
Device(config-if)# tunnel source 209.165.202.129
Device(config-if)# tunnel destination 209.165.200.225
Device(config-if)# end
```

次に、トンネルに設定された IGP インスタンスで BGP の PE1 ループバック IP をアドバタイズする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# router ospf 11
Device(config-router)# router-id 198.51.100.11
Device(config-router)# network 192.0.1.1 0.0.0.0 area 0
Device(config-router)# end
```

次に、トンネルに設定された IGP インスタンスで BGP の PE2 ループバック IP をアドバタイズする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# router ospf 11
Device(config-router)# router-id 203.0.113.201
Device(config-router)# network 192.0.1.1 0.0.0.0 area 0
Device(config-router)# end
```

次に、CE1 が接続されている PE1 に VRF を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# vrf definition vrf-1
Device (config-vrf)# rd 1:1
Device (config-vrf)# address-family ipv4
Device (config-vrf-af)# route-target import 1:2
Device (config-vrf-af)# route-target export 1:1
Device(config-vrf)# end
```

次に、CE2 が接続されている PE2 に VRF を設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# vrf definition vrf-1
Device (config-vrf)# rd 2:2
Device (config-vrf)# address-family ipv4
Device (config-vrf-af)# route-target import 1:1
Device (config-vrf-af)# route-target export 1:2
Device (config-vrf)# end
```

次に、PE1-CE1 インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# int po14.1
Device (config-subif)# encapsulation dot1Q 10
Device (config-subif)# vrf forwarding vrf-1
Device (config-subif)# ip address 14.2.1.1 255.255.255.0
Device (config-subif)# end
```

次に、PE2-CE2 インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# int po24.1
Device (config-subif)# encapsulation dot1Q 10
Device (config-subif)# vrf forwarding vrf-1
Device (config-subif)# ip address 24.2.1.1 255.255.255.0
Device (config-subif)# end
```

次に、PE1-CE1 外部ボーダー ゲートウェイ プロトコル（EBGP）を設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# router bgp 65040
Device (config-router)# address-family ipv4 vrf vrf-1
Device (config-router-af)# neighbor 14.2.1.2 remote-as 65041
Device (config-router-af)# neighbor 14.2.1.2 activate
Device (config-router-af)# exit-address-family
Device (config-router)# end
```

次に、PE2-CE2 EBGP を設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# router bgp 65040
Device (config-router)# address-family ipv4 vrf vrf-1
Device (config-router-af)# neighbor 24.2.1.2 remote-as 65041
Device (config-router-af)# neighbor 24.2.1.2 activate
Device (config-router-af)# exit-address-family
Device (config-router)# end
```

次に、PE1 に PE1-PE2 MP-BGP を設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# router bgp 65040
Device (config-router)# neighbor 192.0.2.1 remote-as 65040
Device (config-router)# neighbor 192.0.2.1 update-source Loopback0
Device (config-router)# address-family ipv4
Device (config-router-af)# neighbor 192.0.2.1 activate
Device (config-router-af)# exit
Device (config-router)# address-family vpnv4
Device (config-router-af)# neighbor 192.0.2.1 activate
```

例：GRE を介した MPLS レイヤ 3 VPN (P-to-PE トンネリング) の設定

```
Device (config-router-af)# neighbor 192.0.2.1 send-community both
Device (config-router-af)# exit
Device (config-router)# end
```

例：GRE を介した MPLS レイヤ 3 VPN (P-to-PE トンネリング) の設定

次に、PE デバイス (PE1 と PE2) および MPLS セグメント (P1) でレイヤ 3 VPN を設定し、PE1 から P1、PE2 への GRE トンネルを設定する例を示します (図 20: P-to-PE トンネリング (149 ページ) を参照)。

次に、PE1 の GRE トンネルにループバック インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Loopback4
Device(config-if)# ip address 209.165.200.230 255.255.255.255
Device(config-if)# end
```

次に、P1 の GRE トンネルにループバック インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Loopback100
Device(config-if)# ip address 209.165.200.235 255.255.255.255
Device(config-if)# end
```

次に、PE1-P1 からインターフェイスを設定し、IGP を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Port-channel11
Device(config-if)# no switchport
Device(config-if)# ip address 209.165.201.1 255.255.255.248
Device(config-if)# ip ospf 10 area 0
Device(config-if)# end
```

次に、P1-PE1 からインターフェイスを設定し、IGP を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Port-channel1
Device(config-if)# no switchport
Device(config-if)# ip address 209.165.201.2 255.255.255.248
Device(config-if)# ip broadcast-address 209.165.201.31
Device(config-if)# ip ospf 10 area 0
Device(config-if)# end
```

次に、PE1 の IGP でループバックをアドバタイズする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# router ospf 10
Device(config-router)# router-id 198.51.100.10
Device(config-router)# network 209.165.200.230 0.0.0.0 area 0
Device(config-router)# end
```

次に、P1 の IGP でループバックをアドバタイズする例を示します。


```
Device> enable
Device# configure terminal
Device(config)# router ospf 10
Device(config-router)# router-id 198.51.100.20
Device(config-router)# network 209.165.200.235 0.0.0.0 area 0
Device(config-router)# end
```

次に、GRE トンネルを設定し、トンネルで IGP インスタンスを設定し、PE1 のトンネルで MPLS を有効にする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel111
Device(config-if)# ip address 209.165.202.140 255.255.255.248
Device(config-if)# ip ospf 11 area 0
Device(config-if)# mpls ip
Device(config-if)# tunnel source 209.165.200.230
Device(config-if)# tunnel destination 209.165.200.235
Device(config-if)# end
```

次に、GRE トンネルを設定し、トンネルで IGP インスタンスを設定し、P1 のトンネルで MPLS を有効にする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel111
Device(config-if)# ip address 209.165.202.141 255.255.255.248
Device(config-if)# ip ospf 11 area 0
Device(config-if)# mpls ip
Device(config-if)# tunnel source 209.165.200.235
Device(config-if)# tunnel destination 209.165.200.230
Device(config-if)# end
```

次に、PE1 上のトンネルの IGP インスタンスで BGP の PE ループバック IP をアドバタイズする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel111
Device(config)# router ospf 11
Device(config-router)# router-id 198.51.100.11
Device(config-router)# network 192.0.1.1 0.0.0.0 area 0
Device(config-router)# end
```

次に、PE2-P1 からインターフェイスを設定し、IGP および MPLS を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Port-channel12
Device(config-if)# no switchport
Device(config-if)# ip address 209.165.201.1 255.255.255.248
Device(config-if)# ip ospf 11 area 0
Device(config-if)# mpls ip
Device(config-if)# end
```

次に、P1-PE2 からインターフェイスを設定し、IGP を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface Port-channel12
Device(config-if)# no switchport
Device(config-if)# ip address 209.165.201.2 255.255.255.248
```

例：GRE を介した MPLS レイヤ 3 VPN (P-to-PE トンネリング) の設定

```
Device(config-if)# ip ospf 11 area 0
Device(config-if)# mpls ip
Device(config-if)# end
```

次に、CE1 が接続されている PE1 で VRF を作成する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# vrf definition vrf-1
Device (config-vrf)# rd 1:1
Device (config-vrf)# address-family ipv4
Device (config-vrf-af)# route-target import 1:2
Device (config-vrf-af)# route-target export 1:1
Device (config-vrf-af)# exit
Device (config-vrf)# end
```

次に、CE2 が接続されている PE2 で VRF を作成する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# vrf definition vrf-1
Device (config-vrf)# rd 2:2
Device (config-vrf)# address-family ipv4
Device (config-vrf-af)# route-target import 1:1
Device (config-vrf-af)# route-target export 1:2
Device (config-vrf-af)# exit
Device (config-vrf)# end
```

次に、PE1-CE1 インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# int po14.1
Device (config-subif)# encapsulation dot1Q 10
Device (config-subif)# vrf forwarding vrf-1
Device (config-subif)# ip address 14.2.1.1 255.255.255.0
Device (config-subif)# exit
Device (config)# end
```

次に、PE2-CE2 インターフェイスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# int po24.1
Device (config-subif)# encapsulation dot1Q 10
Device (config-subif)# vrf forwarding vrf-1
Device (config-subif)# ip address 24.2.1.1 255.255.255.0
Device (config-subif)# exit
Device (config)# end
```

次に、PE1-CE1 EBGP を設定する例を示します。

```
Device> enable
Device# configure terminal
Device (config)# router bgp 65040
Device (config-router)# address-family ipv4 vrf vrf-1
Device (config-router-af)# neighbor 14.2.1.2 remote-as 65041
Device (config-router-af)# neighbor 14.2.1.2 activate
Device (config-router-af)# exit-address-family
Device (config-router)# end
```

次に、PE2-CE2 EBGP を設定する例を示します。

```
Device> enable
Device# configure terminal
```

```

Device (config)# router bgp 65040
Device (config-router)# address-family ipv4 vrf vrf-1
Device (config-router-af)# neighbor 24.2.1.2 remote-as 65041
Device (config-router-af)# neighbor 24.2.1.2 activate
Device (config-router-af)# exit-address-family
Device (config-router)# end

```

次に、PE1 に PE1-PE2 MP-BGP を設定する例を示します。

```

Device> enable
Device# configure terminal
Device (config)# router bgp 65040
Device (config-router)# neighbor 192.0.2.1 remote-as 65040
Device (config-router)# neighbor 192.0.2.1 update-source Loopback0
Device (config-router)# address-family ipv4
Device (config-router-af)# neighbor 192.0.2.1 activate
Device (config-router-af)# exit
Device (config-router)# address-family vpnv4
Device (config-router-af)# neighbor 192.0.2.1 activate
Device (config-router-af)# neighbor 192.0.2.1 send-community both
Device (config-router-af)# exit
Device (config-router)# end

```

次に、PE2 に PE2-PE1 MP-BGP を設定する例を示します。

```

Device> enable
Device# configure terminal
Device (config)# router bgp 65040
Device (config-router)# neighbor 192.0.1.1 remote-as 65040
Device (config-router)# neighbor 192.0.1.1 update-source Loopback0
Device (config-router)# address-family ipv4
Device (config-router-af)# neighbor 192.0.1.1 activate
Device (config-router-af)# exit
Device (config-router)# address-family vpnv4
Device (config-router-af)# neighbor 192.0.1.1 activate
Device (config-router-af)# neighbor 192.0.1.1 send-community both
Device (config-router-af)# exit
Device (config-router)# end

```

GRE を介した MPLS レイヤ 3 VPN の設定の機能情報

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

表 11: GRE を介した MPLS レイヤ 3 VPN の設定に関する機能情報

機能名	リリース	機能情報
GRE を介した MPLS レイヤ 3 VPN	Cisco IOS XE Gibraltar 16.12.1	GRE を介した MPLS レイヤ 3 VPN 機能は、非 MPLS ネットワーク経由で MPLS パケットのトンネリングを行うためのメカニズムを提供します。



第 12 章

MPLS QoS : EXP の分類およびマーキング

- [MPLS EXP の分類とマーキング \(159 ページ\)](#)

MPLS EXP の分類とマーキング

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

MPLS EXP の分類とマーキングの前提条件

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

MPLS EXP の分類とマーキングの制約事項

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

- パケットが MPLS でカプセル化されている場合は、IP などの他のプロトコルの MPLS ペイロードをチェックして分類またはマーキングすることはできません。MPLS EXP マーキングのみが MPLS によってカプセル化されたパケットに影響します。

MPLS EXP の分類とマーキングに関する情報

このセクションでは、MPLS EXP の分類およびマーキングに関する情報を示します。

MPLS EXP の分類とマーキングの概要

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

- **トラフィックの分類**

分類プロセスでマーキングするトラフィックが選択されます。分類は、トラフィックを複数の優先順位レベル、つまり、サービスクラスに分割することによりこのプロセスを実施します。トラフィック分類は、クラスベースの QoS プロビジョニングのプライマリコンポーネントです。詳細については、『Classifying Network Traffic』モジュールを参照してください。

- **トラフィックのポリシングとマーキング**

ポリシングでは、設定されたレートを上回るトラフィックが廃棄されるか、別のドロップレベルにマーキングされます。トラフィックのマーキングは、パケットフローを特定してそれらを区別する方法です。パケットマーキングを利用すれば、ネットワークを複数の優先プライオリティレベルまたはサービスクラスに分割することができます。詳細については、『Marking Network Traffic』モジュールを参照してください。

MPLS 実験フィールド

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

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



- (注) **set ip dscp** により設定されたポリシーマップは、プロバイダーエッジデバイスではサポートされません。MPLS ラベルインポジションノードのポリシーアクションは、**set mpls experimental imposition** 値に基づく必要があります。ただし、入力インターフェイスと出力インターフェイスの両方がレイヤ 3 ポートである場合、アクション **set ip dscp** が指定されたポリシーマップはサポートされます。

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

MPLS EXP の分類とマーキングのメリット

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

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

MPLS EXP の分類とマーキングの方法

このセクションでは、MPLS EXP を分類およびマーキングする方法に関する情報を示します。

MPLS カプセル化パケットの分類

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

手順の概要

1. **enable**
2. **configure terminal**
3. **class-map [match-all | match-any] class-map-name**
4. **match mpls experimental topmost mpls-exp-value**
5. **end**

手順の詳細

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

最も外側のラベルでの MPLS EXP のマーキング

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

最も外側のラベルでの MPLS EXP のマーキング

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

始める前に

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



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



(注) プロバイダーエッジのイーグレスポリシーは、入力時の再マーキングポリシーがある場合にのみ、MPLS EXP クラスの一致により機能します。入力時のプロバイダーエッジは IP インターフェイスであり、デフォルトでは DSCP 値のみが信頼されています。入力時の再マーキングポリシーを設定しない場合、キューイングのラベルは MPLS EXP 値ではなく DSCP 値に基づいて生成されます。ただし、中継プロバイダールータは MPLS インターフェイス上で動作するため、入力時の再マーキングポリシーを設定しなくても機能します。



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

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-map-name*
5. **set mpls experimental imposition** *mpls-exp-value*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	policy-map <i>policy-map-name</i> 例： Device(config)# policy-map mark-up-exp-2	作成されるポリシー マップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 • ポリシー マップ名を入力します。
ステップ 4	class <i>class-map-name</i> 例： Device(config-pmap)# class prec012	トラフィックを指定したクラスにマッチングするために使用するクラス マップを作成し、クラス マップ コンフィギュレーション モードを開始します。

ラベルスイッチドパケットでの MPLS EXP のマーキング

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> クラス マップ名を入力します。
ステップ 5	set mpls experimental imposition <i>mpls-exp-value</i> 例 : Device(config-pmap-c)# set mpls experimental imposition 2	上部のラベルの MPLS EXP フィールドの値を設定します。
ステップ 6	end 例 : Device(config-pmap-c)# end	(任意) 特権 EXEC モードに戻ります。

ラベルスイッチドパケットでの MPLS EXP のマーキング

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

始める前に



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

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-map-name*
5. **set mpls experimental topmost** *mpls-exp-value*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	policy-map policy-map-name 例 : Device(config)# policy-map mark-up-exp-2	作成されるポリシー マップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 • ポリシー マップ名を入力します。
ステップ 4	class class-map-name 例 : Device(config-pmap)# class-map exp012	トラフィックを指定したクラスにマッチングするために使用するクラス マップを作成し、クラス マップ コンフィギュレーション モードを開始します。 • クラス マップ名を入力します。
ステップ 5	set mpls experimental topmost mpls-exp-value 例 : Device(config-pmap-c)# set mpls experimental topmost 2	出力インターフェイスの最上位ラベルの MPLS EXP フィールド値を設定します。
ステップ 6	end 例 : Device(config-pmap-c)# end	(任意) 特権 EXEC モードに戻ります。

条件付きマーキングの設定

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

始める前に



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

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map policy-map-name**

4. **class** *class-map-name*
5. **police** *cir bps bc pir bps be*
6. **conform-action** **transmit**
7. **exceed-action** **set-mpls-exp-topmost-transmit** **dscp** **table** *dscp-table-value*
8. **violate-action** **drop**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none">パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	policy-map <i>policy-map-name</i> 例 : Device(config)# policy-map ip2tag	作成されるポリシー マップの名前を指定し、ポリシー マップ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none">ポリシー マップ名を入力します。
ステップ 4	class <i>class-map-name</i> 例 : Device(config-pmap)# class iptcp	トラフィックと指定されたクラスを照合するために使用するクラス マップを作成し、ポリシーマップ クラス コンフィギュレーション モードを開始します。 <ul style="list-style-type: none">クラス マップ名を入力します。
ステップ 5	police <i>cir bps bc pir bps be</i> 例 : Device(config-pmap-c)# police cir 1000000 pir 2000000	分類するトラフィック用のポリサーを定義し、ポリシーマップクラスポリシング コンフィギュレーション モードを開始します。
ステップ 6	conform-action transmit 例 : Device(config-pmap-c-police)# conform-action transmit 3	ポリサーで指定された値に適合するパケットに対して実行するアクションを定義します。 <ul style="list-style-type: none">この例では、パケットが認定情報レート (cir) に適合する場合または適合バースト (bc) サイズ以内の場合に、MPLS EXP フィールドが 3 に設定されます。

	コマンドまたはアクション	目的
ステップ 7	exceed-action set-mpls-exp-topmost-transmit dscp table dscp-table-value 例 : <pre>Device(config-pmap-c-police)# exceed-action set-mpls-exp-topmost-transmit dscp table dscp2exp</pre>	ポリサーで指定された値を上回るパケットに対して実行するアクションを定義します。
ステップ 8	violate-action drop 例 : <pre>Device(config-pmap-c-police)# violate-action drop</pre>	レートが最大情報レート (pir) を超えており、bc と be の範囲外のパケットに対して実行するアクションを定義します。 <ul style="list-style-type: none"> 違反アクションを指定する前に、超過アクションを指定する必要があります。 この例では、パケット レートが pir レートを超えており、bc と be の範囲外の場合に、パケットがドロップされます。
ステップ 9	end 例 : <pre>Device(config-pmap-c-police)# end</pre>	(任意) 特権 EXEC モードに戻ります。

MPLS EXP の分類とマーキングの設定例

このセクションでは、MPLS EXP の分類とマーキングの設定例を示します。

例 : MPLS カプセル化パケットの分類

MPLS EXP クラス マップの定義

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

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

ポリシー マップの定義とポリシー マップの入インターフェイスへの適用

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

```
Device(config)# policy-map change-exp-3-to-2
Device(config-pmap)# class exp3
```

例 : 最も外側のラベルでの MPLS EXP のマーキング

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

ポリシー マップの定義とポリシー マップの出力インターフェイスへの適用

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

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

例 : 最も外側のラベルでの MPLS EXP のマーキング

MPLS EXP インポジションポリシー マップの定義

次の例では、転送されたパケットの IP precedence 値に基づいて MPLS EXP インポジション値を 2 に設定するポリシー マップを定義します。

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# class-map prec012
Device(config-cmap)# match ip prec 0 1 2
Device(config-cmap)# exit
Device(config)# policy-map mark-up-exp-2
Device(config-pmap)# class prec012
Device(config-pmap-c)# set mpls experimental imposition 2
Device(config-pmap-c)# exit
Device(config-pmap)# exit
```

MPLS EXP インポジションポリシー マップをメイン インターフェイスに適用する

次に、ポリシーマップをギガビットイーサネット インターフェイス 0/0/0 に適用する例を示します。

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

例 : ラベルスイッチドパケットの MPLS EXP のマーキング

MPLS EXP ラベルスイッチドパケット ポリシー マップの定義

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

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

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

次に、ポリシー マップのメイン インターフェイスへの適用例を示します。

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

例 : 条件付きマーキングの設定

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

```
Device(config)# policy-map ip2tag
Device(config-pmap)# class iptcp
Device(config-pmap-c)# police cir 1000000 pir 2000000
Device(config-pmap-c-police)# conform-action transmit
Device(config-pmap-c-police)# exceed-action set-mpls-exp-imposition-transmit 2
Device(config-pmap-c-police)# violate-action drop
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface GigabitEthernet 0/0/1
Device(config-if)# service-policy input ip2tag
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
QoS コマンド	『Cisco IOS Quality of Service Solutions Command Reference』

QoS MPLS EXP の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	QoS MPLS EXP	QoS EXP Matching 機能を使用すると、マルチプロトコル ラベル スイッチング (MPLS) Experimental ビット (EXP ビット) フィールドを変更することで、ネットワークトラフィックを分類、マーキング、およびキューイングできます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 13 章

MPLS スタティックラベルの設定

- [MPLS スタティック ラベル \(171 ページ\)](#)

MPLS スタティック ラベル

このマニュアルでは、Cisco MPLS スタティック ラベル機能について説明します。MPLS スタティック ラベル機能は、次のものをスタティックに設定できるようにします。

- ラベルと IPv4 プレフィックス間のバインディング
- LFIB 相互接続エントリの内容

MPLS スタティック ラベルの前提条件

MPLS スタティック ラベルをイネーブルにするには、次の Cisco IOS 機能をネットワークでサポートする必要があります。

- マルチプロトコル ラベル スイッチング (MPLS)
- Cisco Express Forwarding; シスコ エクスプレス フォワーディング

MPLS スタティック ラベルの制限事項

- MPLS スタティック ラベルのトラブルシューティング プロセスは複雑です。
- MPLS VPN のプロバイダーエッジ (PE) ルータには、ラベルをカスタマー ネットワーク プレフィックス (VPN IPv4 プレフィックス) にスタティックにバインドするためのメカニズムは存在しません。
- MPLS スタティッククロスコネクトはサポートされていません。
- MPLS スタティック ラベルはラベル制御非同期転送モード (lc-atm) ではサポートされていません。

- MPLS スタティック バインディングは、ローカルプレフィックスではサポートされません。

MPLS スタティック ラベルに関する情報

MPLS スタティック ラベルの概要

一般的に、ラベルスイッチングルータ (LSR) は、ラベルスイッチパケットに使用するラベルを動的に学習します。これは、次のようなラベル配布プロトコルによって行われます。

- ラベルをネットワークアドレスにバインドするために使用される Internet Engineering Task Force (IETF) 標準である、Label Distribution Protocol (LDP; ラベル配布プロトコル)
- トラフィック エンジニアリング (TE) のラベル配布に使用されるリソース予約プロトコル (RSVP)
- マルチプロトコル ラベル スwitching (MPLS) パーチャルプライベート ネットワーク (VPN) のラベル配布に使用されるボーダー ゲートウェイ プロトコル (BGP)

学習したラベルをパケットのラベル スwitching に使用するために、LSR はそのラベルをラベル転送情報ベース (LFIB) にインストールします。

MPLS スタティック ラベル機能は、次のものをスタティックに設定できるようにします。

- ラベルと IPv4 プレフィックス間のバインディング
- LFIB 相互接続エントリの内容

MPLS スタティック ラベルの利点

ラベルと IPv4 プレフィックス間のスタティック バインディング

ラベルと IPv4 プレフィックス間のスタティックバインディングを設定して、LDP ラベル配布を実装しないネイバルルータ経由の MPLS ホップバイホップ転送をサポートできます。

MPLS スタティック ラベルの設定方法

MPLS スタティック プレフィックス ラベルバインディングの設定

MPLS スタティック Prefix/Label バインディングを設定するには、グローバルコンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mpls label range** *min-label max-label* [**static** *min-static-label max-static-label*]

4. mpls static binding ipv4 prefix mask [input|output nexthop] label

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mpls label range min-label max-label [static min-static-label max-static-label] 例： Device(config)# mpls label range 200 100000 static 16 199	MPLS スタティック ラベル機能で使用するラベルの範囲を指定します。 (デフォルトではスタティック割り当て用に予約されたラベルはありません)。
ステップ 4	mpls static binding ipv4 prefix mask [input output nexthop] label 例： Device(config)# mpls static binding ipv4 10.0.0.0 255.0.0.0 55	IPv4 プレフィックスに対するラベルのスタティック バインディングを指定します。 指定したバインディングは、ルーティングの要求時に自動的に MPLS 転送テーブルにインストールされます。

MPLS スタティック Prefix/Label バインディングの確認

MPLS スタティック Prefix/Label バインディングの設定を確認するには、次の手順を実行します。

手順の概要

1. **show mpls label range** コマンドを入力します。出力には、新しいラベル範囲はリロードが行われるまで有効にならないことが示されます。
2. 設定されたスタティック Prefix/Label バインディングを表示するには、**show mpls static binding ipv4** コマンドを入力します。
3. MPLS 転送で現在使用されているスタティック Prefix/Label バインディングを確認するには、**show mpls forwarding-table** コマンドを使用します。

手順の詳細

ステップ1 `show mpls label range` コマンドを入力します。出力には、新しいラベル範囲はリロードが行われるまで有効にならないことが示されます。

例：

```
Device# show mpls label range

Downstream label pool: Min/Max label: 16/100000
  [Configured range for next reload: Min/Max label: 200/100000]
Range for static labels: Min/Max/Number: 16/199
```

リロード後に実行される `show mpls label range` コマンドの次の出力には、新しいラベル範囲が有効になっていることが示されます。

例：

```
Device# show mpls label range

Downstream label pool: Min/Max label: 200/100000
Range for static labels: Min/Max/Number: 16/199
```

ステップ2 設定されたスタティック Prefix/Label バインディングを表示するには、`show mpls static binding ipv4` コマンドを入力します。

例：

```
Device# show mpls static binding ipv4
10.17.17.17/32: Incoming label: 251 (in LIB)
  Outgoing labels:
    10.0.0.1          18
10.18.18.18/32: Incoming label: 201 (in LIB)
  Outgoing labels:
    10.0.0.1 implicit-null
```

ステップ3 MPLS 転送で現在使用されているスタティック Prefix/Label バインディングを確認するには、`show mpls forwarding-table` コマンドを使用します。

例：

```
Device# show mpls forwarding-table
Local  Outgoing  Prefix          Bytes tag  Outgoing     Next Hop
tag    tag or VC  or Tunnel Id   switched  interface
201    Pop tag    10.18.18.18/32  0         PO1/1/0     point2point
        2/35      10.18.18.18/32  0         AT4/1/0.1   point2point
251    18        10.17.17.17/32  0         PO1/1/0     point2point
```

MPLS スタティック ラベルの監視とメンテナンス

MPLS スタティック ラベルをモニタおよびメンテナンスするには、次のコマンドを1つ以上使用します。

手順の概要

1. **enable**
2. **show mpls forwarding-table**
3. **show mpls label range**
4. **show mpls static binding ipv4**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Devie> enable	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	show mpls forwarding-table 例： Device# show mpls forwarding-table	MPLS LFIB の内容を表示します。
ステップ 3	show mpls label range 例： Device# show mpls label range	スタティックラベル範囲に関する情報が表示されます。
ステップ 4	show mpls static binding ipv4 例： Device# show mpls static binding ipv4	設定されているスタティック Prefix/Label バインディングに関する情報を表示します。

MPLS スタティック ラベルの設定例

例：MPLS スタティック Prefix/Label の設定

次の出力では、動的に割り当てられたラベル 16 ～ 100000 から 200 ～ 100000 に使用される範囲が **mpls label range** コマンドによって再設定されます。また、16 ～ 199 のスタティックラベル範囲が設定されます。

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# mpls label range 200 100000 static 16 199
% Label range changes take effect at the next reload.
Router(config)# end
```

次の出力では、新しいラベルの範囲はリロードが発生するまで適用されないことが **show mpls label range** コマンドによって示されています。

```
Device# show mpls label range
```

```
Downstream label pool: Min/Max label: 16/100000
 [Configured range for next reload: Min/Max label: 200/100000]
Range for static labels: Min/Max/Number: 16/199
```

次の出力では、リロード後に実行される **show mpls label range** コマンドによって、新しいラベルの範囲が有効になっていることが示されています。

```
Device# show mpls label range
```

```
Downstream label pool: Min/Max label: 200/100000
Range for static labels: Min/Max/Number: 16/199
```

次の出力では、**mpls static binding ipv4** コマンドによってスタティック Prefix/Label バインディングが設定されています。さまざまなプレフィックスの着信（ローカル）と発信（リモート）のラベルも設定されています。

```
Device# configure terminal
```

```
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# mpls static binding ipv4 10.0.0.0 255.0.0.0 55
Device(config)# mpls static binding ipv4 10.0.0.0 255.0.0.0 output 10.0.0.66 2607
Device(config)# mpls static binding ipv4 10.6.0.0 255.255.0.0 input 17
Device(config)# mpls static binding ipv4 10.0.0.0 255.0.0.0 output 10.13.0.8 explicit-null
Device(config)# end
```

次の出力では、**show mpls static binding ipv4** コマンドによってスタティック Prefix/Label バインディングが表示されています。

```
Device# show mpls static binding ipv4
```

```
10.0.0.0/8: Incoming label: none;
  Outgoing labels:
10.13.0.8          explicit-null
10.0.0.0/8: Incoming label: 55 (in LIB)
  Outgoing labels:
    10.0.0.66          2607
10.66.0.0/16: Incoming label: 17 (in LIB)
  Outgoing labels: None
```

その他の参考資料

関連資料

関連項目	マニュアルタイトル
MPLS コマンド	『 <i>Multiprotocol Label Switching Command Reference</i> 』

標準

標準	タイトル
この機能がサポートする新しい規格または変更された規格はありません。既存の規格のサポートは、この機能によって変更されていません。	--

MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャセットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

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

シスコのテクニカル サポート

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

MPLS スタティックラベルの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	MPLS スタティックラベル	MPLS スタティックラベル機能は、ラベルと IPv4 プレフィックス間のバインディングを静的に設定できるようにします。 次のコマンドが導入または変更されました。 debug mpls static binding 、 mpls label range 、 mpls static binding ipv4 、 show mpls label range 、 show mpls static binding ipv4

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 14 章

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

- [VPLS の制約事項 \(179 ページ\)](#)
- [VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport に関する情報 \(180 ページ\)](#)
- [VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport の設定方法 \(183 ページ\)](#)
- [VPLS および VPLS BGP ベースの自動検出の設定例 \(204 ページ\)](#)
- [VPLS および VPLS BGP ベースの自動検出の機能情報 \(209 ページ\)](#)

VPLS の制約事項

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

VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport に関する情報

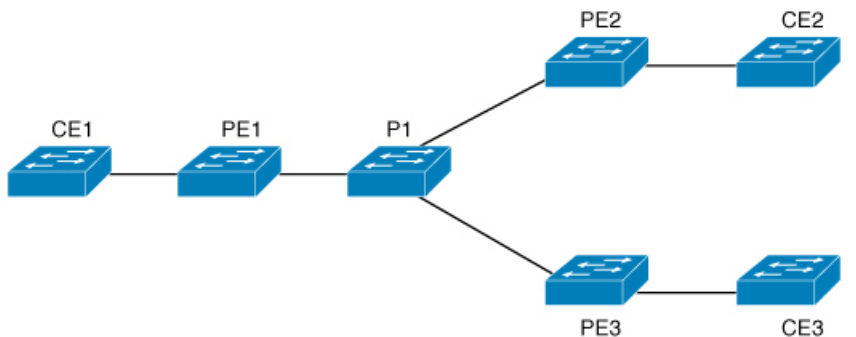
次の項では、VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport について説明します。

VPLS の概要

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

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

図 22: VPLS トポロジ



フルメッシュ構成について

フルメッシュ構成では、VPLSに参加するすべてのプロバイダーエッジ (PE) デバイス間でトンネルラベルスイッチパス (LSP) のフルメッシュが必要です。フルメッシュ構成では、シグナリングのオーバーヘッドと、PE デバイス上でプロビジョニング対象の VC に対するパケット複製の要件が多くなります。

フルメッシュ構成の場合、参加している各 PE デバイスに仮想転送インスタンス (VFI) が必要です。VFI には、VPLS ドメインの VPN ID、そのドメインの他の PE デバイスのアドレス、トンネルシグナリングのタイプ、各ピア PE デバイスのカプセル化のメカニズムが含まれます。

VPLS インスタンスは、エミュレート VC の相互接続によって形成される一連の VFI を構成します。VPLS インスタンスは、パケット交換ネットワーク上の論理ブリッジを形成します。VPLS インスタンスには、一意の VPN ID が割り当てられます。

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

フルメッシュ構成では、PE デバイスが単一のブロードキャストドメインを維持できます。そのため、接続回線でブロードキャスト、マルチキャスト、または未知のユニキャストパケットを受信すると、PE デバイスは、他のすべての接続回線およびエミュレート回線のパケットを、その VPLS インスタンスに参加している他のすべての CE デバイスへに送信します。CE デバイスでは、VPLS インスタンスを、エミュレート LAN として認識します。

プロバイダーコアでのパケットループの問題を回避するために、PE デバイスは、エミュレート VC に「スプリットホライズン」の原則を適用します。スプリットホライズンの原則により、エミュレート VC でパケットを受信したパケットは、他のいずれのエミュレート VC にも転送されなくなります。

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

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

特定の PE デバイスの VPLS インスタンスは、特定の物理または論理ポートに着信するイーサネットフレームを受信し、イーサネットスイッチによる動作同様に、MAC アドレステーブルに入力します。PE デバイスは、この MAC アドレスを使用して、リモートサイトにある別の PE デバイスに配布するために、このようなフレームを適切な LSP に切り替えます。

MAC アドレスが MAC アドレステーブルにない場合、PE デバイスは、イーサネットフレームを複製し、イーサネットフレームが入力された入力ポートを除く、その VPLS インスタンスに関連付けられたすべての論理ポートにフラッドします。PE デバイスは、特定のポートでパケットを受信したときに MAC アドレステーブルを更新し、一定期間使用されていないアドレスを削除します。

VPLS BGP ベースの自動検出について

VPLS 自動検出を使用すると、各 PE デバイスで、同じ VPLS ドメインの一部である他の PE デバイスを検出できます。VPLS 自動検出は、PE デバイスが VPLS ドメインに追加、またはドメインから削除されたタイミングも追跡します。VPLS 自動検出を有効にすると、VPLS ドメインを手動で設定したり、PE デバイスが追加または削除されたときに設定を維持したりする必要がなくなります。VPLS 自動検出は、ボーダー ゲートウェイ プロトコル (BGP) を使用して、VPLS メンバーを検出し、VPLS ドメイン内の擬似回線 (PW) をセットアップおよび解除します。

BGP では、エンドポイントプロビジョニング情報を保存する際にレイヤ 2 VPN ルーティング情報ベース (RIB) が使用されます。これは、レイヤ 2 VFI が設定されるたびにアップデートされます。プレフィックスおよびパス情報はレイヤ 2 VPN データベースに保存され、ベストパスが BGP により決定されるようになります。BGP により、更新メッセージですべての BGP ネ

イーサネットにエンドポイントプロビジョニング情報が配布される場合、レイヤ2 VPN ベースのサービスをサポートするために、このエンドポイント情報を使用して疑似回線メッシュが設定されます。

BGP 自動検出のメカニズムにより、VPLS 機能に必要な不可欠なレイヤ2 VPN サービスの設定が簡易化されます。VPLS は、高速イーサネットを使用した堅牢でスケーラブルな IP MPLS ネットワークによる大規模な LAN として、地理的に分散した拠点間を接続することで柔軟なサービスの展開を実現します。

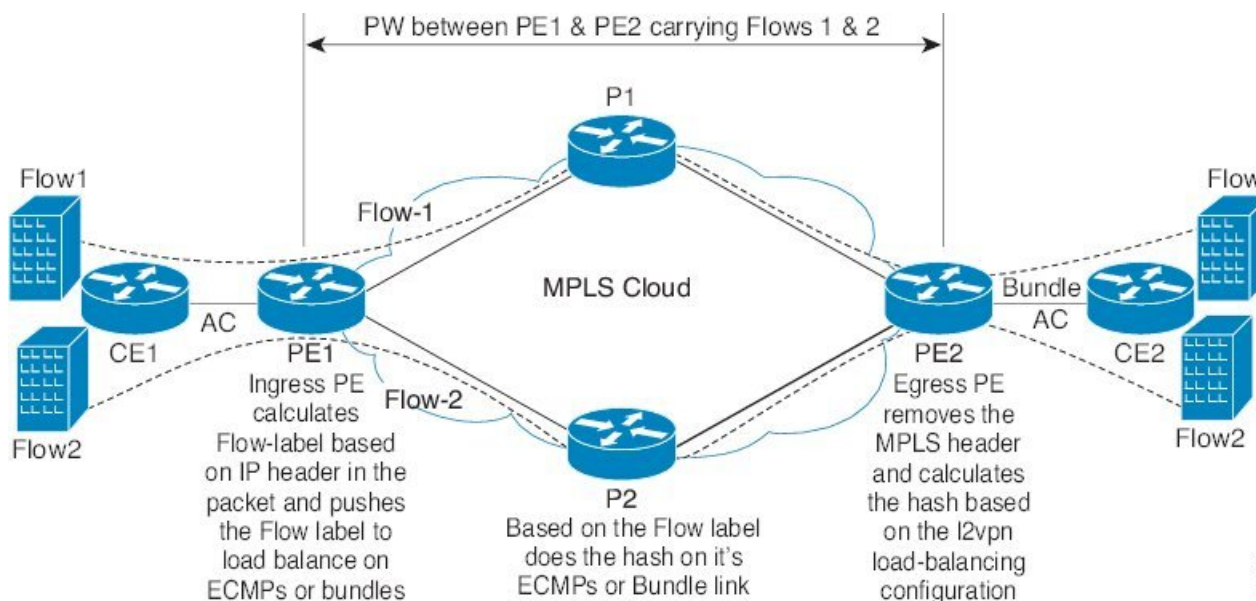
Flow Aware Transport 疑似回線について

デバイスは通常、ラベルスタックの最低ラベル（特定の疑似回線のすべてのフローに対して同じラベル）に基づいてトラフィックをロードバランスします。このとき、非対称ロードバランシングが発生することがあります。このコンテキストでは、フローは同じ送信元/宛先ペアを持つパケットのシーケンスを示します。パケットは、送信元プロバイダーエッジ (PE) デバイスから宛先 PE デバイスに転送されます。

Flow Aware Transport PW は、PW 内の個々のフローを識別する機能を提供します。また、それらのフローを使用してトラフィックをロードバランスする機能をデバイスに提供します。Equal Cost Multipath (ECMP; 等コストマルチパス) が使用されている場合、Flow Aware Transport PW はコア内のトラフィックのロードバランスに使用されます。PW に伝送される個々のパケットフローに基づいてフローラベルが作成され、最低ラベルとしてパケットに挿入されます。デバイスは、フローラベルをロードバランシングに使用でき、コア内の ECMP パスまたはリンクがバンドルされたパスでより適切なトラフィックの分配が行われます。

図 23 : Flow Aware Transport PW と、ECMP およびバンドルされたリンクへ分配される 2つのフローに、Flow Aware Transport PW と、ECMP およびバンドルされたリンクへ分配される 2つのフローの例を示します。

図 23 : Flow Aware Transport PW と、ECMP およびバンドルされたリンクへ分配される 2つのフロー



追加のラベルは、仮想回線 (VC) のフロー情報を含むスタック (フローラベルと呼ばれる) に追加されます。フローラベルは、PW 内のフローを区別する一意の ID で、送信元/宛先 MAC アドレスと送信元/宛先 IP アドレスから取得されます。フローラベルにはラベルスタック (EOS) ビットセットの末尾が含まれ、VC ラベルの後ろや、コントロールワード (存在する場合) の前に挿入されます。入力 PE は、フローラベルを計算し、転送します。Flow Aware Transport PW コンフィギュレーションは、フローラベルを有効にします。出力 PE は、決定が行われないように、フローラベルを廃棄します。

すべてのコアデバイスが、Flow Aware Transport PW でフローラベルに基づいてロードバランシングを実行します。これにより、ECMP とリンクバンドルへのフローの分配が可能になります。

Flow Aware Transport PW は、ポートチャネルロードバランシングアルゴリズムのみに基づいて動作します。

Cisco Catalyst 6000 シリーズスイッチと Cisco Catalyst 9000 シリーズスイッチ間の相互運用性

次の項では、Cisco Catalyst 6000 シリーズスイッチと Cisco Catalyst 9000 シリーズスイッチ間でフローラベルを送受信できるようにする方法について説明します。

Flow Aware Transport PW (Advanced VPLS を使用) で設定された Cisco Catalyst 6000 シリーズスイッチでは、フローラベルのネゴシエーションはサポートされていません。Cisco Catalyst 6000 シリーズスイッチが Cisco Catalyst 9000 シリーズスイッチなどのリモート PE デバイスと相互運用可能な場合、Cisco Catalyst 9000 シリーズスイッチはデータトラフィックのフローラベルを送受信できません。Cisco Catalyst 9000 シリーズスイッチで **load-balance flow-label both static** コマンドを設定すると、Cisco Catalyst 6000 シリーズスイッチがフローラベルのネゴシエーションをサポートしていない場合でも、Cisco Catalyst 9000 シリーズスイッチでフローラベルを送受信できます。

次に、フローラベルの送受信を有効にする設定例を示します。

```
Device> enable
Device# configure terminal
Device(config)# template type pseudowire mpls
Device(config-template)# encapsulation mpls
Device(config-template)# load-balance flow ip dst-ip
Device(config-template)# load-balance flow-label both static
Device(config-template)# end
```

VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport の設定方法

次の項では、VPLS、VPLS BGP ベースの自動検出、および Flow Aware Transport に関する設定情報について説明します。

CE デバイスへのレイヤ 2 PE デバイスインターフェイスの設定

CE デバイスへのレイヤ 2 PE デバイスインターフェイスを設定する必要があります。次の項では、VPLS を設定する前に完了する必要があるさまざまな設定作業について説明します。

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

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device(config)# interface TenGigabitEthernet1/0/24	トランクとして設定するインターフェイスを定義し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 4	no ip address ip_address mask [secondary] 例 : Device(config-if)# no ip address	IP 処理をディセーブルにして、インターフェイスコンフィギュレーションモードを開始します。
ステップ 5	switchport 例 : Device(config-if)# switchport	レイヤ 2 スイッチドインターフェイスのスイッチング特性を変更します。
ステップ 6	switchport trunk encapsulation dot1q 例 : Device(config-if)# switchport trunk encapsulation dot1q	スイッチ ポートのカプセル化形式を 802.1Q に設定します。
ステップ 7	switchport trunk allow vlan vlan_ID 例 :	許可 VLAN のリストを設定します。

	コマンドまたはアクション	目的
	Device(config-if) # switchport trunk allow vlan 2129	
ステップ 8	switchport mode trunk 例 : Device(config-if) # switchport mode trunk	トランキング VLAN レイヤ 2 インターフェイスへのインターフェイスを設定します。
ステップ 9	end 例 : Device(config-if) # end	特権 EXEC モードに戻ります。

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

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device(config) # interface TenGigabitEthernet1/0/24	トランクとして設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	no ip address ip_address mask [secondary] 例 : Device(config-if) # no ip address	IP 処理をディセーブルにします。
ステップ 5	switchport 例 :	レイヤ 2 スイッチドインターフェイスのスイッチング 特性を変更します。

PE デバイスでのレイヤ 2 VLAN インスタンスの設定

	コマンドまたはアクション	目的
	Device(config-if)# switchport	
ステップ 6	switchport mode access 例 : Device(config-if)# switchport mode access	インターフェイスタイプを、非トランキング、タグなし、シングル VLAN レイヤ 2 インターフェイスとして設定します。
ステップ 7	switchport access vlan vlan_ID 例 : Device(config-if)# switchport access vlan 2129	インターフェイスがアクセスモードのときに VLAN を設定します。
ステップ 8	end 例 : Device(config-if)# end	特権 EXEC モードに戻ります。

PE デバイスでのレイヤ 2 VLAN インスタンスの設定

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

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan vlan-id 例 : Device(config)# vlan 2129	特定の VLAN を設定します。
ステップ 4	interface vlan vlan-id 例 :	この VLAN にインターフェイスを設定します。

	コマンドまたはアクション	目的
	Device(config-vlan)# interface vlan 2129	
ステップ 5	end 例 : Device(config-vlan)# end	特権 EXEC モードに戻ります。

VPLS の設定

VPLS は、Xconnect モードまたはプロトコル CLI 方式を使用して設定できます。次の項では、VPLS の設定方法について説明します。

Xconnect モードでの VPLS の設定

次の項では、Xconnect モードでの VPLS の設定について説明します。

PE デバイス上での MPLS の設定

PE デバイスで MPLS を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mpls ip 例 : Device(config)# mpls ip	MPLS ホップバイホップ転送を設定します。
ステップ 4	mpls label protocol ldp 例 : Device(config)# mpls label protocol ldp	プラットフォームの Label Distribution Protocol (LDP; ラベル配布プロトコル) を指定します。

PE デバイスでの VFI の設定

	コマンドまたはアクション	目的
ステップ 5	mpls ldp logging neighbor-changes 例 : Device(config)# mpls ldp logging neighbor-changes	(任意) ネイバーの変更の記録を指定します。
ステップ 6	end 例 : Device(config)# end	特権 EXEC モードに戻ります。

PE デバイスでの VFI の設定

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

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	l2 vfi vfi-name manual 例 : Device(config)# l2 vfi 2129 manual	レイヤ 2 VFI 手動コンフィギュレーション モードをイネーブルにします。
ステップ 4	vpn id vpn-id 例 : Device(config-vfi)# vpn id 2129	VPLS ドメインの VPN ID を設定します。このレイヤ 2 Virtual Routing Forwarding (VRF) にバインドされたエミュレート VC でシグナリングにこの VPN ID が使用されます。 (注) <i>vpn-id</i> は <i>vlan-id</i> と同じです。
ステップ 5	neighbor router-id {encapsulation mpls} 例 :	リモートピアリングルータ ID と、エミュレート VC をセットアップするために使用されるトンネルカプ

	コマンドまたはアクション	目的
	Device(config-vfi)# neighbor remote-router-id encapsulation mpls	セル化タイプまたは疑似回線 (PW) プロパティを指定します。
ステップ 6	end 例 : Device(config-vfi)# end	特権 EXEC モードに戻ります。

PE デバイスでの VFI への接続回線の関連付け

VFI を定義したら、1 つ以上の接続回線に関連付ける必要があります。

接続回線を VFI に関連付けるには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface vlan vlan-id 例 : Device(config)# interface vlan 2129	動的なスイッチ仮想インターフェイス (SVI) を作成するか、使用します。 (注) <i>vlan-id</i> は <i>vpn-id</i> と同じです。
ステップ 4	no ip address 例 : Device(config-if)# no ip address	IP 処理をディセーブルにします。 (IP アドレスを設定する場合は、VLAN のレイヤ 3 インターフェイスを設定できます)。
ステップ 5	xconnect vfi vfi-name 例 : Device(config-if)# xconnect vfi 2129	VLAP ポートにバインドするレイヤ 2 VFI を指定します。
ステップ 6	end 例 :	特権 EXEC モードに戻ります。

■ プロトコル CLI モードでの VPLS の設定

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

プロトコル CLI モードでの VPLS の設定

次の項では、プロトコル CLI モードでの VPLS の設定について説明します。

プロトコル CLI モードでの VPLS の設定

プロトコル CLI モードで VPLS を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	l2vpn vfi context vfi-name 例： Device(config)# l2vpn vfi context vpls1	レイヤ 2 VPN VFI コンテキストを確立して、レイヤ 2 VFI コンフィギュレーションモードを開始します。
ステップ 4	vpn id vpn-id 例： Device(config-vfi)# vpn id 10	VPLS ドメインの VPN ID を設定します。
ステップ 5	member ip-address encapsulation mpls 例： Device(config-vfi)# member 2.2.2.2 encapsulation mpls	ポイントツーポイントレイヤ 2 VPN VFI 接続を形成するデバイスを指定します。
ステップ 6	exit 例： Device(config-vfi)# exit	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	次のいずれかを選択します。 <ul style="list-style-type: none"> • vlan configuration <i>vlan-id</i> • interface vlan <i>vlan-id</i> 例 : Device(config)# vlan configuration 100 OR Device(config)# interface vlan 100	VLAN またはインターフェイスに適用する設定を適用し、VLAN またはインターフェイス コンフィギュレーション モードを開始します。
ステップ 8	member vfi <i>vfi-name</i> 例 : Device(config-vlan-config)# member vfi vpls1	VFI インスタンスを VLAN またはインターフェイス にバインドします。
ステップ 9	end 例 : Device(config-vlan-config)# end	特権 EXEC モードに戻ります。

疑似回線インターフェイスを使用した VPLS Flow Aware Transport の設定 (プロトコル CLI モード)

疑似回線インターフェイスを使用して VPLS Flow Aware Transport を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface pseudowire <i>number</i> 例 : Device(config)# interface pseudowire 1001	指定した名前 で PW を確立して、疑似回線インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	encapsulation mpls 例 :	トンネリング カプセル化を MPLS として指定します。

	コマンドまたはアクション	目的
	Device(config-if)# encapsulation mpls	
ステップ 5	neighbor peer-address vcid-value 例 : Device(config-if)# neighbor 10.1.1.200 200	レイヤ 2 VPN PW のピア IP アドレスと VC ID 値を指定します。
ステップ 6	load-balance flow 例 : Device(config-if)# load-balance flow	ロードバランシングがフロー単位で実行されるように、PW 機能を使用したロードバランシングを有効にします。
ステップ 7	load-balance flow-label 例 : Device(config-if)# load-balance flow-label both	MPLS PW 機能の Flow Aware Transport を有効にして、フローラベルの使用方法を指定します。
ステップ 8	exit 例 : Device(config-if)# exit	特権 EXEC モードに戻ります。
ステップ 9	l2vpn vfi context vfi-name 例 : Device(config)# l2vpn vfi context vpls1	レイヤ 2 VPN VFI コンテキストを確立して、レイヤ 2 VFI コンフィギュレーションモードを開始します。
ステップ 10	vpn id vpn-id 例 : Device(config-vfi)# vpn id 10	VPLS ドメインの VPN ID を設定します。
ステップ 11	member pseudowire number 例 : Device(config-vfi)# member pseudowire 1001	疑似回線インターフェイスを VFI のメンバーとして追加します。
ステップ 12	exit 例 : Device(config-vfi)# exit	特権 EXEC モードに戻ります。
ステップ 13	次のいずれかを選択します。 <ul style="list-style-type: none"> • vlan configuration vlan-id • interface vlan vlan-id 	VLAN またはインターフェイスに適用する設定を適用し、VLAN またはインターフェイス コンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
	例 : Device(config)# vlan configuration 100 OR Device(config)# interface vlan 100	
ステップ 14	member vfi vfi-name 例 : Device(config-vlan-config)# member vfi vpls1	VFI インスタンスを VLAN またはインターフェイスにバインドします。
ステップ 15	end 例 : Device(config-vlan-config)# end	特権 EXEC モードに戻ります。

テンプレートを使用した VPLS Flow Aware Transport の設定 (プロトコル CLI モード)

テンプレートを使用して VPLS Flow Aware Transport を設定すると、複数の PW が同じ設定を共有できます。

テンプレートを使用して VPLS Flow Aware Transport を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	template type pseudowire [template-name] 例 : Device(config)# template type pseudowire mpls	レイヤ 2 PW の名前を指定し、擬似回線テンプレート コンフィギュレーション モードを開始します。
ステップ 4	encapsulation mpls 例 : Device(config-template)# encapsulation mpls	トンネリング カプセル化を MPLS として指定します。

	コマンドまたはアクション	目的
ステップ 5	load-balance flow 例 : Device(config-template)# load-balance flow	ロードバランシングがフロー単位で実行されるように、PW 機能を使用したロードバランシングを有効にします。
ステップ 6	load-balance flow-label 例 : Device(config-template)# load-balance flow-label both	MPLS PW 機能の Flow Aware Transport を有効にして、フローラベルの使用方法を指定します。
ステップ 7	exit 例 : Device(config-template)# exit	特権 EXEC モードに戻ります。
ステップ 8	l2vpn vfi context vfi-name 例 : Device(config)# l2vpn vfi context vpls1	レイヤ 2 VPN VFI コンテキストを確立して、レイヤ 2 VFI コンフィギュレーションモードを開始します。
ステップ 9	vpn id vpn-id 例 : Device(config-vfi)# vpn id 10	VPLS ドメインの VPN ID を設定します。
ステップ 10	member ip-address template template-name 例 : Device(config-vfi)# member 102.102.102.102 template mpls	ポイントツーポイントレイヤ 2 VPN VFI 接続を形成するデバイスを指定します。 <ul style="list-style-type: none"> • ip-address : VFI ネイバーの IP アドレス。 • template-name : テンプレート方式としてテンプレート名 mpls を指定します。 template
ステップ 11	exit 例 : Device(config-vfi)# exit	特権 EXEC モードに戻ります。
ステップ 12	次のいずれかを選択します。 <ul style="list-style-type: none"> • vlan configuration vlan-id • interface vlan vlan-id 例 : Device(config)# vlan configuration 100	VLAN またはインターフェイスに適用する設定を適用し、VLAN またはインターフェイス コンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
	OR Device(config)# interface vlan 100	
ステップ 13	member vfi vfi-name 例 : Device(config-vlan-config)# member vfi vpls1	VFI インスタンスを VLAN またはインターフェイスにバインドします。
ステップ 14	end 例 : Device(config-vlan-config)# end	特権 EXEC モードに戻ります。

疑似回線とテンプレートを使用した VPLS Flow Aware Transport の設定 (プロトコル CLI モード)

PW とテンプレートの両方を使用して VPLS Flow Aware Transport を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	template type pseudowire [template-name] 例 : Device(config)# template type pseudowire mpls	レイヤ 2 PW の名前を指定し、疑似回線テンプレート コンフィギュレーション モードを開始します。
ステップ 4	encapsulation mpls 例 : Device(config-template)# encapsulation mpls	トンネリング カプセル化を MPLS として指定します。
ステップ 5	load-balance flow 例 : Device(config-template)# load-balance flow	ロードバランシングがフロー単位で実行されるように、PW 機能を使用したロードバランシングを有効にします。

	コマンドまたはアクション	目的
ステップ 6	load-balance flow-label 例 : Device(config-template)# load-balance flow-label both	MPLS PW 機能の Flow Aware Transport を有効にして、フローラベルの使用方法を指定します。
ステップ 7	exit 例 : Device(config-template)# exit	特権 EXEC モードに戻ります。
ステップ 8	interface pseudowire number 例 : Device(config)# interface pseudowire 1001	指定した名前でも PW を確立して、疑似回線インターフェイス コンフィギュレーション モードを開始します。
ステップ 9	source template type pseudowire [template-name] 例 : Device(config-if)# source template type pseudowire mpls	mpls という名前のタイプ疑似回線のソーステンプレートを設定します。
ステップ 10	neighbor peer-address vcid-value 例 : Device(config-if)# neighbor 10.1.1.200 200	レイヤ 2 VPN PW のピア IP アドレスと VC ID 値を指定します。
ステップ 11	exit 例 : Device(config-if)# exit	特権 EXEC モードに戻ります。
ステップ 12	l2vpn vfi context vfi-name 例 : Device(config)# l2vpn vfi context vpls1	レイヤ 2 VPN VFI コンテキストを確立して、レイヤ 2 VFI コンフィギュレーション モードを開始します。
ステップ 13	vpn id vpn-id 例 : Device(config-vfi)# vpn id 10	VPLS ドメインの VPN ID を設定します。
ステップ 14	member pseudowire number 例 : Device(config-vfi)# member pseudowire 1001	疑似回線インターフェイスを VFI のメンバーとして追加します。

	コマンドまたはアクション	目的
ステップ 15	exit 例： Device(config-vfi)# exit	特権 EXEC モードに戻ります。
ステップ 16	次のいずれかを選択します。 <ul style="list-style-type: none"> • vlan configuration <i>vlan-id</i> • interface vlan <i>vlan-id</i> 例： Device(config)# vlan configuration 100 OR Device(config)# interface vlan 100	VLAN またはインターフェイスに適用する設定を適用し、VLAN またはインターフェイス コンフィギュレーション モードを開始します。
ステップ 17	member vfi <i>vfi-name</i> 例： Device(config-vlan-config)# member vfi vpls1	VFI インスタンスを VLAN またはインターフェイスにバインドします。
ステップ 18	end 例： Device(config-vlan-config)# end	特権 EXEC モードに戻ります。

VPLS BGP ベースの自動検出の設定

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

VPLS BGP ベースの自動検出のイネーブル化

VPLS BGP ベースの自動検出を有効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例：	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。

VPLS 自動検出を有効にする BGP の設定

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	l2 vfi vfi-name autodiscovery 例 : Device(config)# l2 vfi 2128 autodiscovery	PE デバイス上で VPLS 自動検出を有効にして、L2 VFI コンフィギュレーションモードを開始します。
ステップ 4	vpn id vpn-id 例 : Device(config-vfi)# vpn id 2128	VPLS ドメインの VPN ID を設定します。
ステップ 5	end 例 : Device(config-vfi)# end	特権 EXEC モードに戻ります。

VPLS 自動検出を有効にする BGP の設定

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

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーションモードを開始します。
ステップ 3	router bgp autonomous-system-number 例 : Device(config)# router bgp 1000	指定したルーティングプロセスのルータ コンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 4	<p>no bgp default ipv4-unicast</p> <p>例 :</p> <pre>Device(config-router)# no bgp default ipv4-unicast</pre>	<p>BGP ルーティングプロセスで使用される IPv4 ユニキャストアドレスファミリーを無効にします。</p> <p>(注) IPv4 ユニキャストアドレスファミリーのルーティング情報は、neighbor remote-as router コマンドを使用して設定された各 BGP ルーティングセッションに対して、デフォルトでアドバタイズされます。ただし、neighbor remote-as コマンドを設定する前に、no bgp default ipv4-unicast コマンドを設定した場合は除きます。既存のネイバーコンフィギュレーションは影響されません。</p>
ステップ 5	<p>bgp log-neighbor-changes</p> <p>例 :</p> <pre>Device(config-router)# bgp log-neighbor-changes</pre>	<p>BGP ネイバーリセットのロギングを有効にします。</p>
ステップ 6	<p>neighbor remote-as { ip-address peer-group-name } remote-as autonomous-system-number</p> <p>例 :</p> <pre>Device(config-router)# neighbor 44.254.44.44 remote-as 1000</pre>	<p>指定された自律システム内のネイバーの IP アドレスまたはピア グループ名を、ローカル デバイスの IPv4 マルチプロトコル BGP ネイバー テーブルに追加します。</p> <ul style="list-style-type: none"> • <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致する場合、ネイバーは内部ネイバーになります。 • <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致しない場合、ネイバーは外部ネイバーになります。
ステップ 7	<p>neighbor { ip-address peer-group-name } update-source interface-type interface-number</p> <p>例 :</p> <pre>Device(config-router)# neighbor 44.254.44.44 update-source Loopback300</pre>	<p>(任意) ルーティング テーブル アップデートを受信するための特定のソースまたはインターフェイスを選択するようにデバイスを設定します。</p>
ステップ 8	<p>他の BGP ネイバーを設定する場合は、ステップ 6 と 7 を繰り返します。</p>	<p>インターフェイス コンフィギュレーション モードを終了します。</p>

	コマンドまたはアクション	目的
ステップ 9	address-family l2vpn [vpls] 例 : Device(config-router)# address-family l2vpn vpls	レイヤ 2 VPN アドレスファミリーを指定し、アドレスファミリー コンフィギュレーションモードを開始します。 オプションの vpls キーワードは、VPLS エンドポイントプロビジョニング情報が BGP ピアに配布されるように指定します。
ステップ 10	neighbor { ip-address peer-group-name } activate 例 : Device(config-router-af)# neighbor 44.254.44.44 activate	BGP ネイバーとの情報交換を有効にします。
ステップ 11	neighbor { ip-address peer-group-name } send-community { both standard extended } 例 : Device(config-router-af)# neighbor 44.254.44.44 send-community both	コミュニティ属性が BGP ネイバーに送信されるように指定します。
ステップ 12	ステップ 10 と 11 を繰り返して、L2VPN アドレスファミリー内の他の BGP ネイバーをアクティブにします。	
ステップ 13	exit-address-family 例 : Device(config-router-af)# exit-address-family	アドレスファミリー コンフィギュレーションモードを終了し、ルータ コンフィギュレーションモードに戻ります。
ステップ 14	end 例 : Device(config-router)# end	ルータ コンフィギュレーションモードを終了して、特権 EXEC モードに戻ります。

プロトコル CLI モードでの VPLS BGP ベースの自動検出の設定

次の項では、プロトコル CLI モードでの VPLS BGP ベースの自動検出の設定について説明します。

プロトコル CLI モードでの VPLS BGP ベースの自動検出の設定

プロトコル CLI モードで VPLS BGP ベースの自動検出を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	l2vpn vfi context vfi-name 例： Device(config)# l2vpn vfi context vpls1	レイヤ 2 VPN VFI コンテキストを確立して、レイヤ 2 VFI コンフィギュレーション モードを開始します。
ステップ 4	vpn id vpn-id 例： Device(config-vfi)# vpn id 10	VPLS ドメインの VPN ID を設定します。
ステップ 5	autodiscovery bgp signaling ldp 例： Device(config-vfi)# autodiscovery bgp signaling ldp	BGP シグナリングと LDP シグナリングを有効にします。
ステップ 6	exit 例： Device(config-vfi-autodiscovery)# exit	特権 EXEC モードに戻ります。
ステップ 7	exit 例： Device(config-vfi)# exit	特権 EXEC モードに戻ります。
ステップ 8	次のいずれかを選択します。 <ul style="list-style-type: none">• vlan configuration vlan-id• interface vlan vlan-id	VLAN またはインターフェイスに適用する設定を適用し、VLAN またはインターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	例 : Device(config)# vlan configuration 100 OR Device(config)# interface vlan 100	
ステップ 9	member vfi vfi-name 例 : Device(config-vlan-config)# member vfi vpls1	VFI インスタンスを VLAN またはインターフェイスにバインドします。
ステップ 10	end 例 : Device(config-vlan-config)# end	特権 EXEC モードに戻ります。

テンプレートを使用した VPLS BGP ベースの自動検出 Flow Aware Transport の設定 (プロトコル CLI モード)

テンプレートを使用して VPLS BGP ベースの自動検出 Flow Aware Transport を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	template type pseudowire [template-name] 例 : Device(config)# template type pseudowire mpls	レイヤ 2 PW の名前を指定し、擬似回線テンプレート コンフィギュレーション モードを開始します。
ステップ 4	encapsulation mpls 例 : Device(config-template)# encapsulation mpls	トンネリング カプセル化を MPLS として指定します。
ステップ 5	load-balance flow 例 :	ロードバランシングがフロー単位で実行されるように、PW 機能を使用した Any Transport over MPLS

	コマンドまたはアクション	目的
	Device(config-template)# load-balance flow	(AToM) ロードバランシング機能を有効にします。
ステップ 6	load-balance flow-label 例 : Device(config-template)# load-balance flow-label both	MPLS PW 機能の Flow Aware Transport を有効にして、フローラベルの使用方法を指定します。
ステップ 7	exit 例 : Device(config-template)# exit	特権 EXEC モードに戻ります。
ステップ 8	l2vpn vfi context vfi-name 例 : Device(config)# l2vpn vfi context vpls1	レイヤ 2 VPN VFI コンテキストを確立して、レイヤ 2 VFI コンフィギュレーション モードを開始します。
ステップ 9	vpn id vpn-id 例 : Device(config-vfi)# vpn id 10	VPLS ドメインの VPN ID を設定します。
ステップ 10	autodiscovery bgp signaling ldp template name 例 : Device(config-vfi)# autodiscovery bgp signaling ldp template mpls	BGP シグナリングと LDP シグナリングを有効にします。
ステップ 11	exit 例 : Device(config-vfi)# exit	特権 EXEC モードに戻ります。
ステップ 12	次のいずれかを選択します。 <ul style="list-style-type: none">• vlan configuration vlan-id• interface vlan vlan-id 例 : Device(config)# vlan configuration 100 OR Device(config)# interface vlan 100	VLAN またはインターフェイスに適用する設定を適用し、VLAN またはインターフェイス コンフィギュレーション モードを開始します。
ステップ 13	member vfi vfi-name 例 :	VFI インスタンスを VLAN またはインターフェイスにバインドします。

	コマンドまたはアクション	目的
	Device(config-vlan-config) # member vfi vpls1	
ステップ 14	end 例 : Device(config-vlan-config) # end	特権 EXEC モードに戻ります。

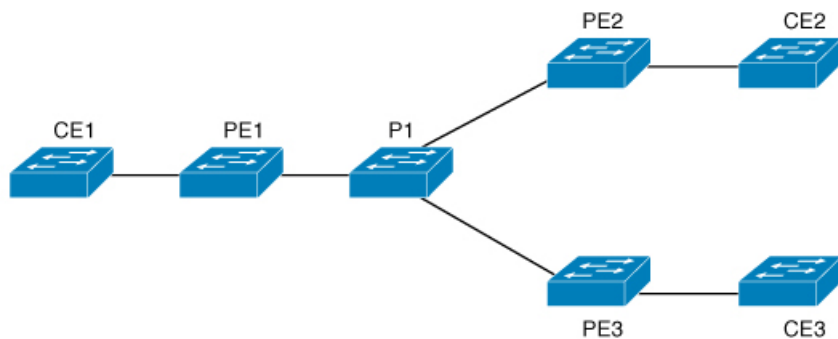
VPLS および VPLS BGP ベースの自動検出の設定例

この項では、VPLS および VPLS BGP ベースの自動検出の設定例を示します。

例 : Xconnect モードでの VPLS の設定

次に、PE1 および PE2 デバイスで VPLS を設定する例を示します。

図 24: VPLS トポロジ



PE1 の設定

```

Device> enable
Device# configure terminal
Device(config)# pseudowire-class vpls2129
Device(config-if)# encapsulation mpls
Device(config-if)# exit
Device(config)# 12 vfi 2129 manual
Device(config-vfi)# vpn id 2129
Device(config-vfi)# neighbor 44.254.44.44 pw-class vpls2129
Device(config-vfi)# neighbor 188.98.89.98 pw-class vpls2129
Device(config-vfi)# exit
Device(config)# interface TenGigabitEthernet1/0/24
Device(config-if)# switchport trunk allowed vlan 2129
Device(config-if)# switchport mode trunk
Device(config-if)# exit
Device(config)# interface vlan 2129
Device(config-vlan-config)# no ip address
Device(config-vlan-config)# xconnect vfi 2129

```

例 : Xconnect モードで設定された VPLS の確認

次に、**show mpls 12transport vc detail** コマンドの出力例を示します。このコマンドの出力には、仮想回線に関する情報が表示されます。

```

Device# show mpls 12transport vc detail
Local interface: VFI 2129 vfi up
Interworking type is Ethernet
Destination address: 44.254.44.44, VC ID: 2129, VC status: up
Output interface: Gi1/0/9, imposed label stack {18 17}
Preferred path: not configured
Default path: active
Next hop: 177.77.177.2
Create time: 19:09:33, last status change time: 09:24:14
Last label FSM state change time: 09:24:14
Signaling protocol: LDP, peer 44.254.44.44:0 up
Targeted Hello: 1.1.1.72(LDP Id) -> 44.254.44.44, LDP is UP
Graceful restart: configured and enabled
Non stop routing: not configured and not enabled
Status TLV support (local/remote) : enabled/supported
LDP route watch : enabled
Label/status state machine : established, LruRru
Last local dataplane status rcvd: No fault
Last BFD dataplane status rcvd: Not sent
Last BFD peer monitor status rcvd: No fault
Last local AC circuit status rcvd: No fault
Last local AC circuit status sent: No fault
Last local PW i/f circ status rcvd: No fault
Last local LDP TLV status sent: No fault
Last remote LDP TLV status rcvd: No fault
Last remote LDP ADJ status rcvd: No fault
MPLS VC labels: local 512, remote 17
Group ID: local n/a, remote 0
MTU: local 1500, remote 1500
Remote interface description:
Sequencing: receive disabled, send disabled

```

例: Xconnect モードで設定されたVPLSの確認

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

次に、**show l2vpn atom vc** コマンドの出力例を示します。このコマンドの出力には、ATM over MPLS が VC に設定されていることが示されます。

```
Device# show l2vpn atom vc detail

pseudowire100005 is up, VC status is up PW type: Ethernet
  Create time: 19:25:56, last status change time: 09:40:37
  Last label FSM state change time: 09:40:37
  Destination address: 44.254.44.44 VC ID: 2129
  Output interface: Gi1/0/9, imposed label stack {18 17}
  Preferred path: not configured
  Default path: active
  Next hop: 177.77.177.2
Member of vfi service 2129
  Bridge-Domain id: 2129
  Service id: 0x32000003
Signaling protocol: LDP, peer 44.254.44.44:0 up
  Targeted Hello: 1.1.1.72(LDP Id) -> 44.254.44.44, LDP is UP
  Graceful restart: configured and enabled
  Non stop routing: not configured and not enabled
  PWid FEC (128), VC ID: 2129
  Status TLV support (local/remote)           : enabled/supported
    LDP route watch                           : enabled
    Label/status state machine                 : established, LruRru
    Local dataplane status received            : No fault
    BFD dataplane status received              : Not sent
    BFD peer monitor status received           : No fault
    Status received from access circuit        : No fault
    Status sent to access circuit              : No fault
    Status received from pseudowire i/f        : No fault
  Status sent to network peer                  : No fault
    Status received from network peer          : No fault
    Adjacency status of remote peer           : No fault
  Sequencing: receive disabled, send disabled
Bindings
  Parameter      Local                               Remote
  -----
  Label          512                                           17
  Group ID       n/a                                           0
  Interface
  MTU            1500                                           1500
  Control word   off                                           off
  PW type        Ethernet                                         Ethernet
  VCCV CV type   0x02                                           0x02
                  LSPV [2]                               LSPV [2]
  VCCV CC type   0x06                                           0x06
                  RA [2], TTL [3]                           RA [2], TTL [3]
  Status TLV     enabled                                         supported
SSO Descriptor: 44.254.44.44/2129, local label: 512
Dataplane:
  SSM segment/switch IDs: 20498/20492 (used), PWID: 2
Rx Counters
```

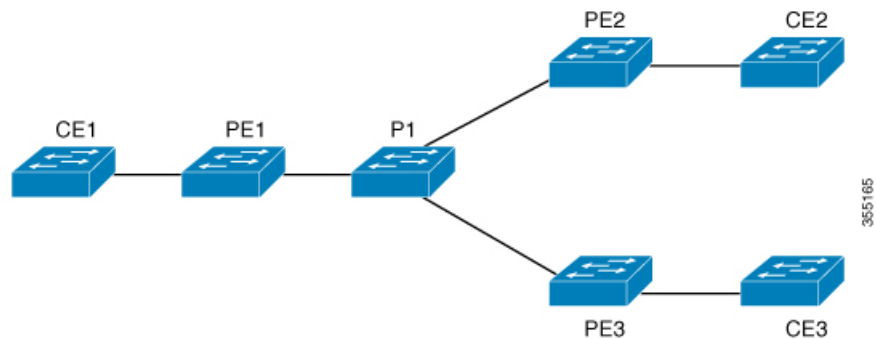
```

0 input transit packets, 0 bytes
0 drops, 0 seq err
Tx Counters
0 output transit packets, 0 bytes
0 drops
    
```

例：テンプレートを使用した VPLS Flow Aware Transport の設定 (プロトコル CLI モード)

次に、PE1 および PE2 デバイスで VPLS を設定する例を示します。

図 25: VPLS トポロジ



PE1 の設定

```

Device> enable
Device# configure terminal
Device(config)# template type pseudowire mpls
Device(config-template)# encapsulation mpls
Device(config-template)# load-balance flow ip dst-ip
Device(config-template)# load-balance flow-label both
Device(config-template)# exit
Device(config)# interface Loopback0
Device(config-if)# ip address 1.1.1.30 255.255.255.255
Device(config-if)# ip ospf 1 area 0
Device(config-if)# exit
Device(config)# interface TwentyFiveGigE1/0/9
Device(config-if)# no switchport
Device(config-if)# ip address 80.0.0.30 255.255.255.0
Device(config-if)# ip ospf 1 area 0
Device(config-if)# mpls ip
Device(config-if)# exit
Device(config)# l2vpn vfi context foo
Device(config-vfi)# vpn id 2129
Device(config-vfi)# member 1.1.1.20 template mpls
Device(config-vfi)# exit
Device(config)# interface TwentyFiveGigE1/0/2
Device(config-if)# switchport mode access
Device(config-if)# switchport access vlan 100
Device(config-if)# exit
Device(config)# interface vlan 100
Device(config-vlan-config)# member vfi foo
Device(config-vlan-config)# end
    
```

例 : VPLS BGP 自動検出の設定

次に、PE デバイスで VPLS を設定する例を示します。

```

Device> enable
Device# configure terminal
Device(config)# router bgp 1000
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# bgp graceful-restart
Device(config-router)# neighbor 44.254.44.44 remote-as 1000
Device(config-router)# neighbor 44.254.44.44 update-source Loopback300
Device(config-router)# address-family l2vpn vpls
Device(config-router-af)# neighbor 44.254.44.44 activate
Device(config-router-af)# neighbor 44.254.44.44 send-community both
Device(config-router-af)# exit-address-family
Device(config-router-af)# end
Device(config)# 12 vfi 2128 autodiscovery
Device(config-vfi)# vpn id 2128
Device(config-vfi)# exit
Device(config)# interface vlan 2128
Device(config-vlan-config)# no ip address
Device(config-vlan-config)# xconnect vfi 2128
!
```

例 : VPLS BGP 自動検出の確認

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

```

Device# show platform software fed sw 1 matm macTable vlan 2000

VLAN  MAC                Type      Seq#    macHandle          siHandle          diHandle
      *a_time *e_time  ports
2000  2852.6134.05c8  0X8002   0       0xffbba312c8      0xffbb9ef938     0x5154
      0          0       Vlan2000
2000  0000.0078.9012  0X1     32627  0xffbb665ec8      0xffbb60b198     0xffbb653f98
      300      278448  Port-channel11
2000  2852.6134.0000  0X1     32651  0xffba15e1a8      0xff454c2328     0xffbb653f98
      300      63     Port-channel11
2000  0000.0012.3456  0X2000001 32655  0xffba15c508      0xff44f9ec98     0x0
      300      1       2000:33.33.33.33
Total Mac number of addresses:: 4
*a_time=aging_time(secs) *e_time=total_elapsed_time(secs)
Type:
MAT_DYNAMIC_ADDR      0x1     MAT_STATIC_ADDR      0x2
MAT_CPU_ADDR          0x4     MAT_DISCARD_ADDR     0x8
MAT_ALL_VLANS         0x10    MAT_NO_FORWARD       0x20
MAT_IPMULT_ADDR       0x40    MAT_RESYNC            0x80
MAT_DO_NOT_AGE        0x100   MAT_SECURE_ADDR      0x200
MAT_NO_PORT           0x400   MAT_DROP_ADDR        0x800
MAT_DUP_ADDR          0x1000  MAT_NULL_DESTINATION 0x2000
MAT_DOTIX_ADDR        0x4000  MAT_ROUTER_ADDR      0x8000
MAT_WIRELESS_ADDR     0x10000 MAT_SECURE_CFG_ADDR  0x20000
MAT_OPQ_DATA_PRESENT  0x40000 MAT_WIRED_TUNNEL_ADDR 0x80000
MAT_DLR_ADDR          0x100000 MAT_MRP_ADDR          0x200000
```

```
MAT_MSRP_ADDR      0x400000  MAT_LISP_LOCAL_ADDR  0x800000
MAT_LISP_REMOTE_ADDR 0x1000000 MAT_VPLS_ADDR      0x2000000
```

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

```
Device# show bgp l2vpn vpls all

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

VPLS および VPLS BGP ベースの自動検出の機能情報

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

表 12: VPLS および VPLS BGP ベースの自動検出の機能情報

機能名	リリース	機能情報
VPLS および VPLS BGP ベースの自動検出の設定	Cisco IOS XE Everest 16.5.1a	VPLSにより、企業は、サービスプロバイダーから提供されるインフラストラクチャを介して、複数サイトからのイーサネットベースのLANをまとめてリンクできます。 VPLS自動検出を使用すると、各 PE デバイスで、同じ VPLS ドメインの一部である他の PE デバイスを検出できます。
VPLS レイヤ 2 スヌーピング : IGMP (IPv4)	Cisco IOS XE Amsterdam 17.1.1	IGMP スヌーピングは、VPLS が設定されたネットワークでサポートされます。



第 15 章

MPLS VPN ルート ターゲット 書き換えの 設定

- [MPLS VPN ルート ターゲット 書き換えの前提条件 \(211 ページ\)](#)
- [MPLS VPN ルート ターゲット 書き換えの制約事項 \(211 ページ\)](#)
- [MPLS VPN ルート ターゲット 書き換えに関する情報 \(211 ページ\)](#)
- [MPLS VPN ルート ターゲット 書き換えの設定方法 \(213 ページ\)](#)
- [MPLS VPN ルート ターゲット 書き換えの設定例 \(220 ページ\)](#)
- [MPLS VPN ルート ターゲット 書き換えの機能履歴 \(221 ページ\)](#)

MPLS VPN ルート ターゲット 書き換えの前提条件

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

MPLS VPN ルート ターゲット 書き換えの制約事項

ルート ターゲットの書き換えは、単一 AS トポロジにのみ実装できます。

`ip unnumbered` コマンドは MPLS 設定ではサポートされていません。

MPLS VPN ルート ターゲット 書き換えに関する情報

この項では、MPLS VPN ルートターゲット書き換えについて説明します。

ルート ターゲット置換ポリシー

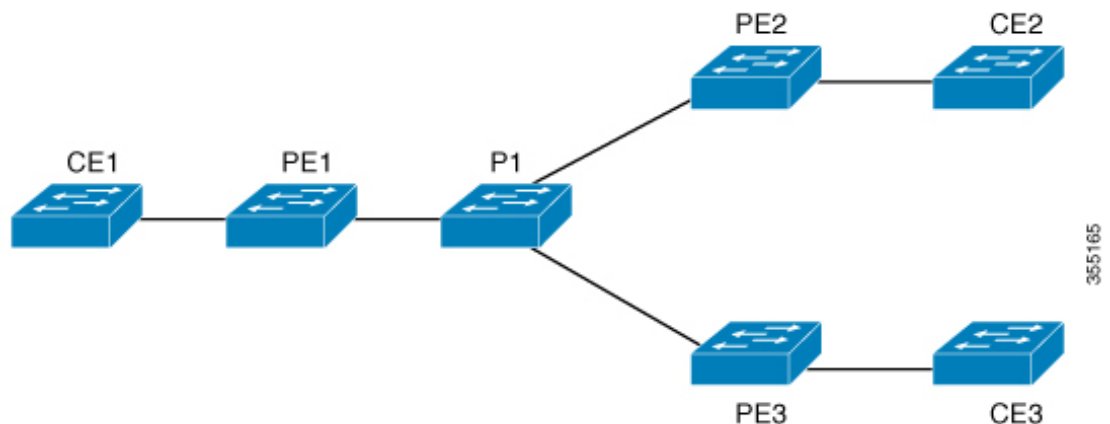
ピアのルーティング ポリシーには、インバウンドまたはアウトバウンドのルーティング テーブルアップデートに影響する可能性のある設定がすべて含まれています。インバウンドおよびアウトバウンドの Border Gateway Protocol (BGP) アップデートに対してルート ターゲットの置換を有効にすると、MPLS VPN ルート ターゲット書き換え機能がルーティング テーブルアップデートに影響する可能性があります。BGP バーチャルプライベート ネットワーク IP バージョン4 (VPNv4) のアップデートでは、ルートターゲットが拡張コミュニティ属性として送信されます。ルートターゲット拡張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

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

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

- PE1 は、VRF カスタマー A の RT 65000:1 をインポートおよびエクスポートして、RT 65000:1 のすべてのインバウンド VPNv4 プレフィックスを RT 65000:2 に書き換えるように設定されています。
- PE2 は、VRF カスタマー B の RT 65000:2 をインポートおよびエクスポートして、RT 65000:2 のすべてのインバウンド VPNv4 プレフィックスを RT 65000:1 に書き換えるように設定されています。

図 26: 単一の MPLS VPN 自律システム トポロジのプロバイダー エッジ (PE) デバイスでのルート ターゲットの置換



ルート マップおよびルート ターゲットの置換

MPLS VPN ルート ターゲット書き換え機能によって Border Gateway Protocol (BGP) インバウンド/アウトバウンドルートマップ機能が拡張され、ルートターゲットの置換がイネーブルになります。ルートマップ コンフィギュレーション モードで入力した `set extcomm-list delete` コ

マンドを使用すると、拡張コミュニティリストに基づいてルートターゲット拡張コミュニティ属性を削除できます。

MPLS VPN ルート ターゲット書き換えの設定方法

次の項では、MPLS VPN ルートターゲット書き換えの設定手順について説明します。

ルート ターゲット置換ポリシーの設定

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

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

手順の概要

1. **enable**
2. **configure terminal**
3. **ip extcommunity-list** {*standard-list-number* | *expanded-list-number*} {**permit** | **deny**} [*regular-expression*] [**rt** | **soo** *extended-community-value*]
4. **route-map** *map-name* [**permit** | **deny**] [*sequence-number*]
5. **match extcommunity** {*standard-list-number* | *expanded-list-number*}
6. **set extcomm-list** *extended-community-list-number* **delete**
7. **set extcommunity** { **rt** *extended-community-value* [**additive**] | **soo** *extended-community-value*}
8. **end**
9. **show route-map** *map-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip extcommunity-list { <i>standard-list-number</i> <i>expanded-list-number</i> } { permit deny } [<i>regular-expression</i>] [rt soo <i>extended-community-value</i>]	拡張コミュニティ アクセス リストを作成し、リストへのアクセスを制御します。

	コマンドまたはアクション	目的
	<p>例 :</p> <pre>Device(config)# ip extcommunity-list 1 permit rt 65000:2</pre>	<ul style="list-style-type: none"> • <i>standard-list-number</i> 引数は 1 ~ 99 の整数で、拡張コミュニティの 1 つまたは複数の許可グループまたは拒否グループを指定します。 • <i>expanded-list-number</i> 引数は 100 ~ 500 の整数で、拡張コミュニティの 1 つまたは複数の許可グループまたは拒否グループを指定します。拡張リストには正規表現を設定できますが、標準リストには設定できません。 • permit キーワードを指定すると、条件が一致した場合にアクセスが許可されます。 • deny キーワードを指定すると、条件が一致した場合にアクセスが拒否されます。 • <i>regular-expression</i> 引数には、マッチングを行う入力ストリングパターンを指定します。拡張された拡張コミュニティリストを使用してルートターゲットのマッチングを行う場合は、正規表現にパターン RT: を追加します。 • rt キーワードには、ルートターゲット拡張コミュニティ属性を指定します。rt キーワードは標準拡張コミュニティリストにのみ設定できます。拡張された拡張コミュニティリストには設定できません。 • soo キーワードには、Site of Origin (SOO) 拡張コミュニティ属性を指定します。soo キーワードは標準拡張コミュニティリストだけに設定できます。拡張された拡張コミュニティリストには設定できません。 • <i>extended-community-value</i> 引数には、ルートターゲットまたは Site of Origin を指定します。この値には次の組み合わせのいずれかを指定できます。 <ul style="list-style-type: none"> • <code>autonomous-system-number:network-number</code> • <code>ip-address:network-number</code> <p>自律システム番号とネットワーク番号、または IP アドレスとネットワーク番号の区切りにはコロンを使用します。</p>

	コマンドまたはアクション	目的
ステップ 4	<p>route-map <i>map-name</i> [permit deny] [<i>sequence-number</i>]</p> <p>例 :</p> <pre>Device(config)# route-map rtrewrite permit 10</pre>	<p>ルーティングプロトコル間でルートを再配布する条件を定義するか、ポリシールーティングをイネーブルにしてルートマップコンフィギュレーションモードを開始します。</p> <ul style="list-style-type: none"> • <i>map-name</i> 引数では、ルート マップに意味のある名前を定義します。 redistribute ルータ コンフィギュレーションコマンドはこの名前を使用して、このルートマップを参照します。複数のルートマップで同じマップ名を共有できます。 • このルートマップの一致基準が満たされた場合、 permit キーワードが指定されていると、設定アクションに従ってルートが再配布されます。ポリシールーティングの場合、パケットはポリシーに従ってルーティングされます。 <p>一致基準が満たされなかった場合、 permit キーワードが指定されていると、同じマップタグを持つ次のルートマップがテストされます。あるルートが、同じ名前を共有するルート マップセットの一致基準のいずれをも満たさない場合、そのセットによる再配布は行われません。</p> <p>デフォルトは permit キーワードです。</p> <ul style="list-style-type: none"> • ルートマップの一致基準が満たされた場合でも、 deny キーワードが指定されているとルートは再配布されません。ポリシールーティングの場合、パケットはポリシーに従ってルーティングされません。また、同じマップタグ名を共有するルート マップは、これ以上検証されません。パケットがポリシールーティングの対象にならない場合、通常の転送アルゴリズムが使用されます。 • <i>sequence-number</i> 引数は、同じ名前を設定済みのルートマップのリストにおける新しいルートマップの位置を示す番号です。このコマンドの no 形式を指定すると、ルートマップの位置が削除されます。
ステップ 5	<p>match extcommunity {<i>standard-list-number</i> <i>expanded-list-number</i>}</p> <p>例 :</p>	<p>Border Gateway Protocol (BGP) 拡張コミュニティ リスト属性とマッチングします。</p>

	コマンドまたはアクション	目的
	Device(config-route-map)# match extcommunity 1 例 : Device(config-route-map)# match extcommunity 101	<ul style="list-style-type: none"> • <i>standard-list-number</i> 引数は 1 ~ 99 の番号で、拡張コミュニティ属性の 1 つまたは複数の許可グループまたは拒否グループを指定します。 • <i>expanded-list-number</i> 引数は 100 ~ 500 の番号で、拡張コミュニティ属性の 1 つまたは複数の許可グループまたは拒否グループを指定します。
ステップ 6	set extcomm-list extended-community-list-number delete 例 : Device(config-route-map)# set extcomm-list 1 delete	インバウンドまたはアウトバウンド BGP バーチャルプライベートネットワークバージョン 4 (VPNv4) アップデートの拡張コミュニティ属性からルートターゲットを削除します。 <ul style="list-style-type: none"> • <i>extended-community-list-number</i> 引数には、拡張コミュニティ リスト番号を指定します。
ステップ 7	set extcommunity { rt extended-community-value [additive] soo rt extended-community-value } 例 : Device(config-route-map)# set extcommunity rt 65000:1 additive	BGP 拡張コミュニティ属性を設定します。 <ul style="list-style-type: none"> • rt キーワードには、ルートターゲット拡張コミュニティ属性を指定します。 • soo キーワードには、Site of Origin 拡張コミュニティ属性を指定します。 • <i>extended-community-value</i> 引数には、設定値を指定します。この値には次の組み合わせのいずれかを指定できます。 <ul style="list-style-type: none"> • autonomous-system-number:network-number • ip-address:network-number 自律システム番号とネットワーク番号、または IP アドレスとネットワーク番号の区切りにはコロンを使用します。 <ul style="list-style-type: none"> • additive キーワードを指定すると、既存のルートターゲットを置換することなく、既存のルートターゲットリストにルートターゲットが追加されます。
ステップ 8	end 例 : Device(config-route-map)# end	(任意) 特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	show route-map map-name 例 : Device# show route-map extmap	(任意) マッチングと設定されたエントリが正しいことを確認します。 <ul style="list-style-type: none"> • <i>map-name</i> 引数には、特定のルート マップの名前を指定します。

ルート ターゲット置換ポリシーの適用

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

特定の BGP ネイバーへのルート マップの割り当て

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp as-number**
4. **neighbor {ip-address | peer-group-name} remote-as as-number**
5. **address-family vpv4 [unicast]**
6. **neighbor {ip-address | peer-group-name} activate**
7. **neighbor {ip-address | peer-group-name} send-community [both | extended | standard]**
8. **neighbor {ip-address | peer-group-name} route-map map-name {in | out}**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp as-number 例 : Device(config)# router bgp 100	Border Gateway Protocol (BGP) ルーティングプロセスを設定し、デバイスでルータ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • <i>as-number</i> 引数は、デバイスを他の BGP デバイスに対して識別し、転送するルーティング情報

	コマンドまたはアクション	目的
		<p>にタグを設定する自律システムの番号を示します。</p> <p>指定できる範囲は 0 ~ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ~ 65535 です。</p>
ステップ 4	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} remote-as <i>as-number</i></p> <p>例 :</p> <pre>Device(config-router)# neighbor 172.10.0.2 remote-as 200</pre>	<p>BGP ネイバーテーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。</p> <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、ネイバーの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、BGP ピア グループの名前を指定します。 • <i>as-number</i> 引数には、ネイバーが属している自律システムを指定します。
ステップ 5	<p>address-family vpnv4 [unicast]</p> <p>例 :</p> <pre>Device(config-router)# address-family vpnv4</pre>	<p>アドレス ファミリ コンフィギュレーション モードを開始して、標準バーチャルプライベートネットワークバージョン 4 (VPNv4) アドレスプレフィックスを使用する、BGP などのルーティングセッションを設定します。</p> <ul style="list-style-type: none"> • unicast キーワード (任意) は、VPNv4 ユニキャストアドレスプレフィックスを指定します。
ステップ 6	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} activate</p> <p>例 :</p> <pre>Device(config-router-af)# neighbor 172.16.0.2 activate</pre>	<p>ネイバー BGP デバイスとの情報交換を有効にします。</p> <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、ネイバーの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、BGP ピア グループの名前を指定します。
ステップ 7	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} send-community [both extended standard]</p> <p>例 :</p> <pre>Device(config-router-af)# neighbor 172.16.0.2 send-community extended</pre>	<p>コミュニティ属性が BGP ネイバーに送信されるように指定します。</p> <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、BGP 対応ネイバーの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、BGP ピア グループの名前を指定します。 • both キーワードを指定すると、標準および拡張コミュニティ属性が送信されます。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • extended キーワードを指定すると、拡張コミュニティ属性が送信されます。 • standard キーワードを指定すると、標準コミュニティ属性が送信されます。
ステップ 8	neighbor { <i>ip-address</i> <i>peer-group-name</i> } route-map <i>map-name</i> { in out } 例 : <pre>Device(config-router-af)# neighbor 172.16.0.2 route-map extmap in</pre>	着信ルートまたは発信ルートにルートマップを適用します。 <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、ネイバーの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、BGP ピア グループまたはマルチプロトコル ピア グループの名前を指定します。 • <i>map-name</i> 引数には、ルート マップの名前を指定します。 • in キーワードを指定すると、着信ルートにルートマップが適用されます。 • out キーワードを指定すると、発信ルートにルートマップが適用されます。
ステップ 9	end 例 : <pre>Device(config-router-af)# end</pre>	(任意) 特権 EXEC モードに戻ります。

ルート ターゲット置換ポリシーの確認

手順の概要

1. **enable**
2. **show ip bgp vpnv4 vrf vrf-name**
3. **exit**

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例 :

```
Device> enable
Device#
```

ステップ 2 show ip bgp vpnv4 vrf vrf-name

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

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

例：

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

ステップ 3 exit

ユーザー EXEC モードに戻ります。

例：

```
Device# exit
Device>
```

MPLS VPN ルート ターゲット書き換えの設定例

次の項では、MPLS VPN ルートターゲット書き換えの設定例について説明します。

例：ルート ターゲット置換ポリシーの適用

例：特定の BGP ネイバーへのルート マップの割り当て

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

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

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

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

MPLS VPN ルートターゲット書き換えの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	MPLS VPN ルートターゲット書き換え	インバウンドおよびアウトバウンドの Border Gateway Protocol (BGP) アップデートに対してルートターゲットの置換を有効にすると、MPLS VPN ルートターゲット書き換え機能がルーティングテーブルアップデートに影響する可能性があります。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 16 章

MPLS VPN-Inter-AS-IPv4 BGP ラベル配布の設定

- [MPLS VPN Inter-AS IPv4 BGP ラベル配布 \(223 ページ\)](#)
- [MPLS VPN Inter-AS IPv4 BGP ラベル配布 \(224 ページ\)](#)
- [MPLS VPN Inter-AS IPv4 BGP ラベル配布に関する情報 \(224 ページ\)](#)
- [MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定方法 \(226 ページ\)](#)
- [ルートマップの作成 \(234 ページ\)](#)
- [MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定の確認 \(240 ページ\)](#)
- [MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定例 \(247 ページ\)](#)
- [MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定の機能履歴 \(262 ページ\)](#)

MPLS VPN Inter-AS IPv4 BGP ラベル配布

この機能を使用すると、バーチャルプライベートネットワーク (VPN) サービスプロバイダーネットワークを設定できます。このネットワークでは、自律システム境界ルータ (ASBR) が、プロバイダーエッジ (PE) ルータのマルチプロトコル ラベル スイッチング (MPLS) ラベル付きの IPv4 ルートを交換します。ルートリフレクタ (RR) は、マルチホップ、マルチプロトコル外部ボーダーゲートウェイプロトコル (EBGP) を使用して VPNv4 ルートを交換します。この設定では、ASBR にすべての VPNv4 ルートを格納する必要がなくなります。ルートリフレクタを使用して VPNv4 ルートを格納し、PE ルータに転送すると、拡張性が向上します。

MPLS VPN—Inter-AS—IPv4 BGP ラベル配布機能には、次の利点があります。

- ルートリフレクタを使用して VPNv4 ルートを格納すると拡張性が向上する：この設定は、ASBR がすべての VPNv4 ルートを保持し、VPNv4 ラベルに基づいてルートを転送する設定よりも拡張性が優れています。この設定では、ルートリフレクタが VPNv4 ルートを保持することで、ネットワーク境界での設定が簡素化されます。
- 非 VPN コアネットワークが VPN トラフィックの中継ネットワークとして機能できる：非 MPLS VPN サービスプロバイダーを介して、MPLS ラベル付きの IPv4 ルートを転送できます。

- 隣接 LSR 間の他のラベル配布プロトコルが不要になる：隣接する 2 つのラベルスイッチルータ (LSR) が BGP ピアでもある場合、BGP で MPLS ラベルの配布を実行できます。これら 2 つの LSR 間で、他のラベル配布プロトコルは必要ありません。
- 自律システム (AS) の境界を越えた IPv4 ルートのロードバランシングを可能にする EBGP マルチパスのサポートが含まれています。

MPLS VPN Inter-AS IPv4 BGP ラベル配布

この機能には、次の制約事項があります。

- EBGP マルチホップが設定されたネットワークでは、非隣接デバイス間にラベルスイッチパス (LSP) を設定する必要があります (RFC 3107)。
- PE デバイスでは、BGP ラベル配布をサポートするイメージを実行する必要があります。実行できない場合は、PE デバイス間で EBGP を実行できません。
- ASBR 上の Point-to-Point Protocol (PPP) カプセル化は、この機能ではサポートされていません。
- BGP スピーカーを接続する物理インターフェイスは、Cisco Express Forwarding (CEF) または分散型 CEF と MPLS をサポートしている必要があります。

MPLS VPN Inter-AS IPv4 BGP ラベル配布に関する情報

MPLS VPN Inter-AS IPv4 BGP ラベル配布を設定するには、次の情報が必要です。

MPLS VPN Inter-AS IPv4 BGP ラベル配布の概要

この機能を使用すると、VPN サービス プロバイダー ネットワークを設定して、MPLS ラベル付き IPv4 ルートを交換できます。次のように VPN サービス プロバイダー ネットワークを設定できます。

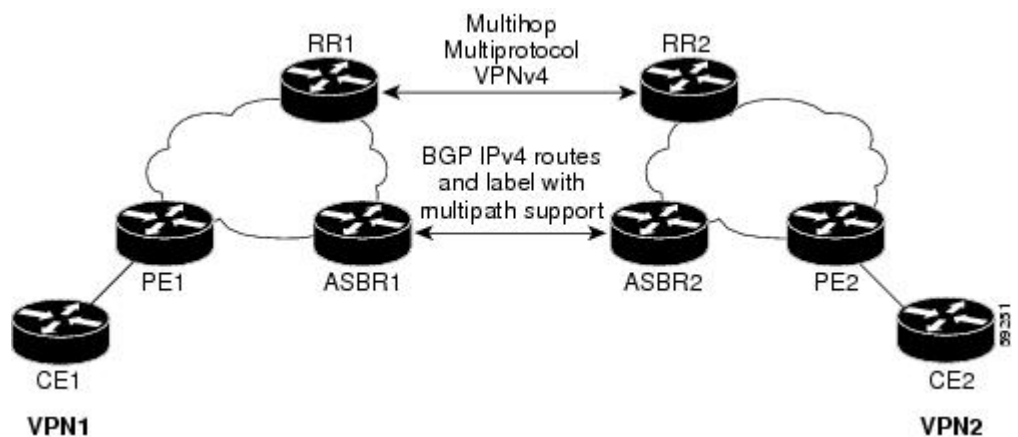
- ルートリフレクタは、マルチホップ、マルチプロトコル EBGP を使用して VPNv4 ルートを交換します。この設定では、自律システムをまたがってネクストホップ情報および VPN ラベルが維持されます。
- ローカル PE ルータ (図 1 の PE1 など) は、リモート PE ルータ (PE2) のルートおよびラベル情報を認識する必要があります。この情報は、次のいずれかの方法で PE ルータおよび ASBR 間で交換できます。
 - 内部ゲートウェイ プロトコル (IGP) と Label Distribution Protocol (LDP; ラベル配布プロトコル) : ASBR は、EBGP から学習した IPv4 ルートおよび MPLS ラベルを IGP や LDP に再配布できます。その逆も可能です。

- 内部ボーダー ゲートウェイ プロトコル (IBGP) IPv4 ラベル配布 : ASBR および PE ルータは、直接 IBGP セッションを使用して、VPNv4 と IPv4 ルートおよび MPLS ラベルを交換できます。

または、ルート リフレクタが、ASBR から学習した IPv4 ルートおよび MPLS ラベルを VPN の PE ルータに反映できます。これは、ASBR が IPv4 ルートおよび MPLS ラベルをルートリフレクタと交換できるようにすることで実現されます。ルートリフレクタは、VPNv4 ルートも VPN の PE ルータに反映します (最初の箇条書き項目を参照)。たとえば、VPN1 では、RR1 は、学習した VPNv4 ルート、および ASBR1 から学習した IPv4 ルートと MPLS ラベルを PE1 に反映します。ルートリフレクタを使用して VPNv4 ルートを格納し、それらのルートを PE ルータおよび ASBR 経由で転送することで、スケーラブルな構成が可能になります。

- ASBR は、EBGP を使用して PE ルータの IPv4 ルートと MPLS ラベルを交換します。これにより、CSC 境界全体のロードバランシングが可能になります。

図 27: EBGP および IBGP を使用してルートと MPLS ラベルを配布する VPN



BGP ルーティング情報

BGP ルーティング情報には、次の項目が含まれています。

- 宛先の IP アドレスであるネットワーク番号 (プレフィックス)。
- 自律システム (AS) パス : ルートがローカルルータに到達するために通過する他の AS のリスト。リスト内の最初の自律システムがローカルルータに最も近いシステムです。リスト内の最後の自律システムはローカルルータから最も遠いシステムであり、通常は、ルートの始点となる自律システムです。
- ネクスト ホップなどの、自律システム パスについての他の情報を提供するパス属性。

BGP においてルートとともに MPLS ラベルが送信される方法

BGP（EBGP および IBGP）でルート配布する場合、そのルートにマッピングされている MPLS ラベルも配布できます。ルートの MPLS ラベルマッピング情報は、そのルートに関する情報を含む BGP 更新メッセージによって伝送されます。ネクストホップが変わらない場合は、ラベルも維持されます。

両方の BGP ルータで `neighbor send-label` コマンドを発行すると、ルートとともに MPLS ラベルを送信できることがルータ間で相互にアドバタイズされます。ルータ間で MPLS ラベルを送信可能であると正常にネゴシエーションされると、それらのルータからのすべての発信 BGP アップデートに MPLS ラベルが追加されます。

ルートマップを使用したルートのフィルタリング

両方のルータが MPLS ラベルを使用してルート配布するように設定されている場合、すべてのルートがマルチプロトコル拡張を使用して符号化され、すべてのルートに MPLS ラベルが付いています。ルートマップを使用して、ルータ間の MPLS ラベルの配布を制御できます。ルートマップで指定できるルートは次のとおりです。

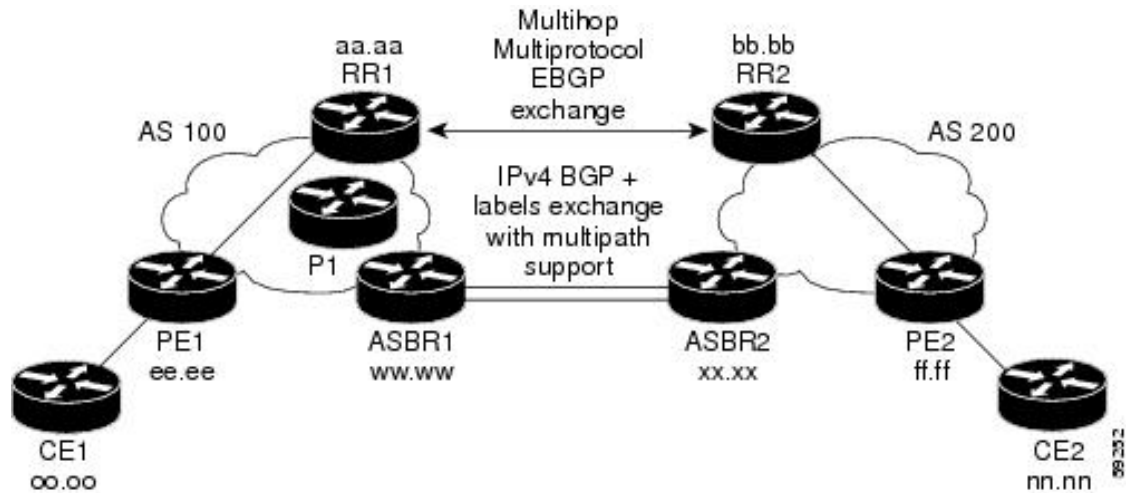
- MPLS ラベルを配布するルータの場合、MPLS ラベルを使用して配布するルートを指定できます。
- MPLS ラベルを受信するルータの場合、受け入れるルートおよび BGP テーブルにインストールするルートを指定できます。

MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定方法

以下の図は、次の設定を示しています。

- この設定は、2つの VPN で構成されています。
- ASBR は、MPLS ラベル付きの IPv4 ルートを交換します。
- ルートリフレクタは、マルチホップ MPLS EBGP を使用して VPNv4 ルートを交換します。
- ルートリフレクタは、その自律システム内の他のルータに IPv4 ルートおよび VPN4 ルートを反映します。

図 28: IPv4 ルートおよび MPLS ラベルを交換する 2つの VPN サービス プロバイダーの設定



IPv4 ルートおよび MPLS ラベルを交換する ASBR の設定

次のタスクを実行して、ASBR を設定し、MPLS ラベル付きの BGP ルートを配布できるようにします。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp *as-number***
4. **neighbor { *ip-address* | *peer-group-name* } remote-as *as-number***
5. **address-family ipv4 [*multicast* | *unicast* | *vrfvrf-name*]**
6. **maximum-paths *number-paths***
7. **neighbor { *ip-address* | *peer-group-name* } activate**
8. **neighbor *ip-address* send-label**
9. **exit-address-family**
10. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 :	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
	Device# configure terminal	
ステップ 3	router bgp <i>as-number</i> 例： Device(config)# router bgp 100	ルータ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • as-number : 他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。有効値の範囲は 1 ~ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ~ 65535 です。
ステップ 4	neighbor {<i>ip-address</i> <i>peer-group-name</i>} remote-as <i>as-number</i> 例： Device(config)# neighbor 209.165.201.2 remote-as 200	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。 <ul style="list-style-type: none"> • ip-address 引数には、ネイバーの IP アドレスを指定します。 • peer-group-name 引数には、BGP ピアグループの名前を指定します。 • as-number 引数には、ネイバーが属している自律システムを指定します。
ステップ 5	address-family ipv4 [multicast unicast vrf <i>vrf-name</i>] 例： Device(config-router)# address-family ipv4	標準 IPv4 アドレスプレフィックスを使用する BGP などのルーティングセッションを設定するために、アドレスファミリ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • multicast キーワードでは、IPv4 マルチキャスト アドレス プレフィックスを指定します。 • unicast キーワードでは、IPv4 ユニキャスト アドレス プレフィックスを指定します。 • vrf vrf-name キーワードおよび引数では、後続の IPv4 アドレス ファミリ コンフィギュレーション モード コマンドに関連付ける VPN ルーティング および 転送 (VRF) インスタンスの名前を指定します。
ステップ 6	maximum-paths <i>number-paths</i> 例： Device(config-router)# maximum-paths 2	(任意) IP ルーティング プロトコル がサポート できる 並列 ルートの 最大数 を 制御 します。 number-paths 引数には、IP ルーティング プロトコル がルーティング テーブル にインストール する 平行 ルートの 最大数 を 1 ~ 6 の範囲 で 指定 します。

	コマンドまたはアクション	目的
ステップ 7	neighbor {ip-address peer-group-name} activate 例： <pre>Device(config-router-af)# neighbor 209.165.201.2 activate</pre>	ネイバー ルータとの情報交換をイネーブルにします。 <ul style="list-style-type: none"> • ip-address 引数には、ネイバーの IP アドレスを指定します。 • peer-group-name 引数には、BGP ピア グループの名前を指定します。
ステップ 8	neighbor ip-address send-label 例： <pre>Device(config-router-af)# neighbor 10.0.0.1 send-label</pre>	BGP ルートとともに MPLS ラベルをネイバー BGP ルータに送信できるように BGP ルータを設定します。 <ul style="list-style-type: none"> • ip-address 引数には、ネイバー ルータの IP アドレスを指定します。
ステップ 9	exit-address-family 例： <pre>Device(config-router-af)# exit-address-family</pre>	アドレスファミリサブモードを終了します。
ステップ 10	end 例： <pre>Device(config-router-af)# end</pre>	(任意) 終了して、特権 EXEC モードに戻ります。

VPNv4 ルートを交換するルートリフレクタの設定

始める前に

ルートリフレクタでマルチホップ、マルチプロトコル EBGp を使用して VPNv4 ルートを交換できるようにするには、次の手順を実行します。

また、この手順では、自律システム間でネクストホップ情報および VPN ラベルが維持されるように指定します。この手順では、例として RR1 を使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp as-number**
4. **neighbor {ip-address|peer-group-name} remote-as as-number**
5. **address-family vpnv4 [unicast]**
6. **neighbor {ip-address|peer-group-name} ebgp-multihop [ttl]**
7. **neighbor {ip-address|peer-group-name} activate**
8. **neighbor {ip-address|peer-group-name} next-hop unchanged**

9. **exit-address-family**

10. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>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。</p>
ステップ 3	<p>router bgp as-number</p> <p>例 :</p> <pre>Device(config)# router bgp 100</pre>	<p>ルータ コンフィギュレーション モードを開始します。</p> <ul style="list-style-type: none"> • as-number : 他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。有効値の範囲は 1 ~ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ~ 65535 です。 <p>自律システム番号によって、他の自律システム内のルータで RR1 が特定されます。</p>
ステップ 4	<p>neighbor {ip-address peer-group-name} remote-as as-number</p> <p>例 :</p> <pre>Device(config)# neighbor 192.0.2.1 remote-as 200</pre>	<p>BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。</p> <ul style="list-style-type: none"> • ip-address 引数には、ネイバーの IP アドレスを指定します。 • peer-group-name 引数には、BGP ピアグループの名前を指定します。 • as-number 引数には、ネイバーが属している自律システムを指定します。
ステップ 5	<p>address-family vpnv4 [unicast]</p> <p>例 :</p> <pre>Device(config-router)# address-family vpnv4</pre>	<p>アドレスファミリ コンフィギュレーション モードを開始して、標準仮想プライベート ネットワークバージョン 4 (VPNv4) アドレスプレフィックスを使用する、BGP などのルーティングセッションを設定します。</p>

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • unicast キーワード (任意) は、VPNv4 ユニキャストアドレスプレフィックスを指定します。
ステップ 6	neighbor { <i>ip-address</i> <i>peer-group-name</i> } ebgp-multihop [<i>ttl</i>] 例 : Device(config-router-af)# neighbor 192.0.2.1 ebgp-multihop 255	直接接続されていないネットワーク上の外部ピアからの BGP 接続を受け入れ、またそのピアへの BGP 接続を試みます。 <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、BGP 対応ネイバーの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、BGP ピアグループの名前を指定します。 • <i>ttl</i> 引数には、1 ~ 255 ホップの範囲の存続可能時間を指定します。
ステップ 7	neighbor { <i>ip-address</i> <i>peer-group-name</i> } activate 例 : Device(config-router-af)# neighbor 192.0.2.1 activate	ネイバー ルータとの情報交換をイネーブルにします。 <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、ネイバーの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、BGP ピアグループの名前を指定します。
ステップ 8	neighbor { <i>ip-address</i> <i>peer-group-name</i> } next-hop unchanged 例 : Device(config-router-af)# neighbor 10.0.0.2 next-hop unchanged	外部 BGP (EBGP) マルチホップピアで、ネクストホップを変更せずに伝播できるようにします。 <ul style="list-style-type: none"> • <i>ip-address</i> 引数には、ネクストホップの IP アドレスを指定します。 • <i>peer-group-name</i> 引数には、ネクストホップである BGP ピアグループの名前を指定します。
ステップ 9	exit-address-family 例 : Device(config-router-af)# exit-address-family	アドレスファミリサブモードを終了します。
ステップ 10	end 例 : Device(config-router-af)# end	(任意) 終了して、特権 EXEC モードに戻ります。

自律システム内でリモートルートを反映するルートリフレクタの設定

RR が ASBR から学習した IPv4 ルートおよびラベルを自律システム内の PE ルータに反映できるようにするには、次の手順を実行します。

これは、ASBR および PE ルータを RR のルートリフレクタ クライアントにすることによって実現されます。また、この手順では、RR で VPNv4 ルートを反映できるようにする方法についても説明します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp *as-number***
4. **address-family ipv4 [multicast | unicast | vrfvrf-name]**
5. **neighbor {ip-address | peer-group-name} activate**
6. **neighbor ip-address route-reflector-client**
7. **neighbor ip-address send-label**
8. **exit-address-family**
9. **address-family vpnv4 [unicast]**
10. **neighbor {ip-address | peer-group-name} activate**
11. **neighbor ip-address route-reflector-client**
12. **exit-address-family**
13. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp <i>as-number</i> 例： Device(config)# router bgp 100	ルータ コンフィギュレーション モードを開始します。 • <i>as-number</i> ：他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。有効値の範囲は 1～65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512～65535 です。

	コマンドまたはアクション	目的
		自律システム番号によって、他の自律システム内のルータで RR1 が特定されます。
ステップ 4	address-family ipv4 [multicast unicast vrf vrf-name] 例 : Device(config-router)# address-family ipv4	標準 IPv4 アドレス プレフィックスを使用する BGP などのルーティングセッションを設定するために、アドレスファミリ コンフィギュレーションモードを開始します。 <ul style="list-style-type: none"> • multicast キーワードでは、IPv4 マルチキャストアドレスプレフィックスを指定します。 • unicast キーワードでは、IPv4 ユニキャストアドレスプレフィックスを指定します。 • vrf vrf-name キーワードおよび引数では、後続の IPv4 アドレスファミリ コンフィギュレーションモードコマンドに関連付ける VPN ルーティングおよび転送 (VRF) インスタンスの名前を指定します。
ステップ 5	neighbor { ip-address peer-group-name } activate 例 : Device(config-router-af)# neighbor 203.0.113.1 activate	ネイバー ルータとの情報交換をイネーブルにします。 <ul style="list-style-type: none"> • ip-address 引数には、ネイバーの IP アドレスを指定します。 • peer-group-name 引数には、BGP ピア グループの名前を指定します。
ステップ 6	neighbor ip-address route-reflector-client 例 : Device(config-router-af)# neighbor 203.0.113.1 route-reflector-client	ルータを BGP ルートルリフレクタとして設定し、指定したネイバーをそのクライアントとして設定します。 <ul style="list-style-type: none"> • ip-address 引数には、クライアントとして識別される BGP ネイバーの IP アドレスを指定します。
ステップ 7	neighbor ip-address send-label 例 : Device(config-router-af)# neighbor 203.0.113.1 send-label	BGP ルートとともに MPLS ラベルをネイバー BGP ルータに送信できるように BGP ルータを設定します。 <ul style="list-style-type: none"> • ip-address 引数には、ネイバー ルータの IP アドレスを指定します。
ステップ 8	exit-address-family 例 :	アドレスファミリサブモードを終了します。

	コマンドまたはアクション	目的
	Device(config-router-af)# exit-address-family	
ステップ 9	address-family vpnv4 [unicast] 例： Device(config-router)# address-family vpnv4	アドレスファミリ コンフィギュレーションモードを開始して、標準 VPNv4 アドレスプレフィックスを使用する、BGP などのルーティングセッションを設定します。 <ul style="list-style-type: none"> • unicast キーワード (任意) は、VPNv4 ユニキャスト アドレスプレフィックスを指定します。
ステップ 10	neighbor {ip-address peer-group-name} activate 例： Device(config-router-af)# neighbor 203.0.113.1 activate	ネイバー ルータとの情報交換をイネーブルにします。 <ul style="list-style-type: none"> • ip-address 引数には、ネイバーの IP アドレスを指定します。 • peer-group-name 引数には、BGP ピア グループの名前を指定します。
ステップ 11	neighbor ip-address route-reflector-client 例： Device(config-router-af)# neighbor 203.0.113.1 route-reflector-client	RR がネイバー ルータに IBGP ルートを渡せるようにします。
ステップ 12	exit-address-family 例： Device(config-router-af)# exit-address-family	アドレスファミリ サブモードを終了します。
ステップ 13	end 例： Device(config-router-af)# end	(任意) 終了して、特権 EXEC モードに戻ります。

ルートマップの作成

ルートマップを使用すると、MPLS ラベルを使用して配布するルートを指定できます。また、ルータが受信し、BGP テーブルに追加する MPLS ラベル付きのルートも指定することもできます。

ルートマップはアクセスリストと連動します。ルートをアクセスリストに入力し、ルートマップを設定するときにアクセスリストを指定します。

次の手順を実行すると、ASBR 使用して、ルートマップで指定されているルートとともに MPLS ラベルを送信できます。また、ASBR はルートマップで指定されたルートのみを受け入れません。

着信ルート用のルートマップの設定

着信ルートをフィルタリングするルートマップを作成するには、次の作業を実行します。アクセスリストを作成し、ルータで受け入れて BGP テーブルに追加させるルートを指定します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp *as-number***
4. **route-map *route-map name* [permit|deny] [*sequence-number*]**
5. **match ip address {*access-list-number*|*access-list-name*} [...*access-list-number*]|...*access-list-name*]**
6. **match mpls-label**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp <i>as-number</i> 例： Device(config)# router bgp 100	ルータ コンフィギュレーション モードを開始します。 • as-number ：他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。有効値の範囲は 1 ～ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ～ 65535 です。 自律システム番号によって、他の自律システム内のルータで RR1 が特定されます。
ステップ 4	route-map <i>route-map name</i> [permit deny] [<i>sequence-number</i>] 例： Device(config-router)# route-map IN permit 11	指定した名前でもルートマップを作成します。 • permit キーワードを指定すると、すべての条件が満たされた場合にアクションが実行されます。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • deny キーワードを指定すると、すべての条件が満たされた場合にアクションが実行されません。 • sequence-number 引数を指定すると、ルートマップに優先順位付けできます。複数のルートマップが存在し、それらにプライオリティを設定する場合、それぞれに番号を割り当てます。最初に最も低い番号のルートマップが実装され、次に2番めに低い番号のルートマップが実装され、それ以降も同様です。
ステップ 5	match ip address { <i>access-list-number</i> <i>access-list-name</i> } [... <i>access-list-number</i> ... <i>access-list-name</i>] 例： Device(config-route-map)# match ip address 2	標準アクセスリストまたは拡張アクセスリストで許可された宛先ネットワーク番号アドレスを含むすべてのルートを配するか、またはパケットに対してポリシールーティングを実行します。 <ul style="list-style-type: none"> • access-list-number 引数は、標準アクセスリストまたは拡張アクセスリストの番号です。1～199の整数を指定できます。 • access-list-name 引数は、標準アクセスリストまたは拡張アクセスリストの名前です。1～199の整数を指定できます。
ステップ 6	match mpls-label 例： Device(config-route-map)# match mpls-label	ルートがルートマップで指定された条件を満たす場合、MPLS ラベルを含むルートが再配布されます。
ステップ 7	end 例： Device(config-router-af)# end	(任意) 終了して、特権 EXEC モードに戻ります。

発信ルート用のルートマップの設定

発信ルートをフィルタリングするルートマップを作成するには、次の作業を実行します。アクセスリストを作成し、MPLS ラベルを使用してルータに配布させるルートを指定します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **route-map** *route-map name* [**permit** | **deny**] [*sequence-number*]

5. **match ip address** {*access-list-number* | *access-list-name*} [...*access-list-number* | ...*access-list-name*]
6. **set mpls-label**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp <i>as-number</i> 例： Device(config)# router bgp 100	ルータ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • as-number : 他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。有効値の範囲は 1 ~ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ~ 65535 です。 AS 番号によって、他の自律システム内のルータへの RRI が特定されます。
ステップ 4	route-map <i>route-map name</i> [permit deny] [<i>sequence-number</i>] 例： Device(config-router)# route-map OUT permit 10	指定した名前でもルートマップを作成します。 <ul style="list-style-type: none"> • permit キーワードを指定すると、すべての条件が満たされた場合にアクションが実行されません。 • deny キーワードを指定すると、すべての条件が満たされた場合にアクションが実行されません。 • sequence-number 引数を指定すると、ルートマップに優先順位付けできます。複数のルートマップが存在し、それらにプライオリティを設定する場合、それぞれに番号を割り当てます。最初に最も低い番号のルートマップが実装され、次に 2 番めに低い番号のルートマップが実装され、それ以降も同様です。

	コマンドまたはアクション	目的
ステップ 5	match ip address { <i>access-list-number</i> <i>access-list-name</i> } [... <i>access-list-number</i> ... <i>access-list-name</i>] 例： Device(config-route-map)# match 10.0.0.2 1	標準アクセスリストまたは拡張アクセスリストで許可された宛先ネットワーク番号アドレスを含むすべてのルートを配するか、またはパケットに対してポリシー ルーティングを実行します。 <ul style="list-style-type: none"> • access-list-number 引数は、標準アクセスリストまたは拡張アクセスリストの番号です。1～199の整数を指定できます。 • access-list-name 引数は、標準アクセスリストまたは拡張アクセスリストの名前です。1～199の整数を指定できます。
ステップ 6	set mpls-label 例： Device(config-route-map)# set mpls-label	ルートがルートマップで指定された条件を満たす場合、MPLS ラベルを使用してルートを配布できるようにします。
ステップ 7	end 例： Device(config-router-af)# end	(任意) 終了して、特権 EXEC モードに戻ります。

ASBR へのルートマップの適用

ASBR でルートマップを使用できるようにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **address-family ipv4** [**multicast** | **unicast** | **vrf** *vrf-name*]
5. **neighbor ip-address route-map** *route-map-name* **out**
6. **neighbor ip-address send-label**
7. **exit-address-family**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。

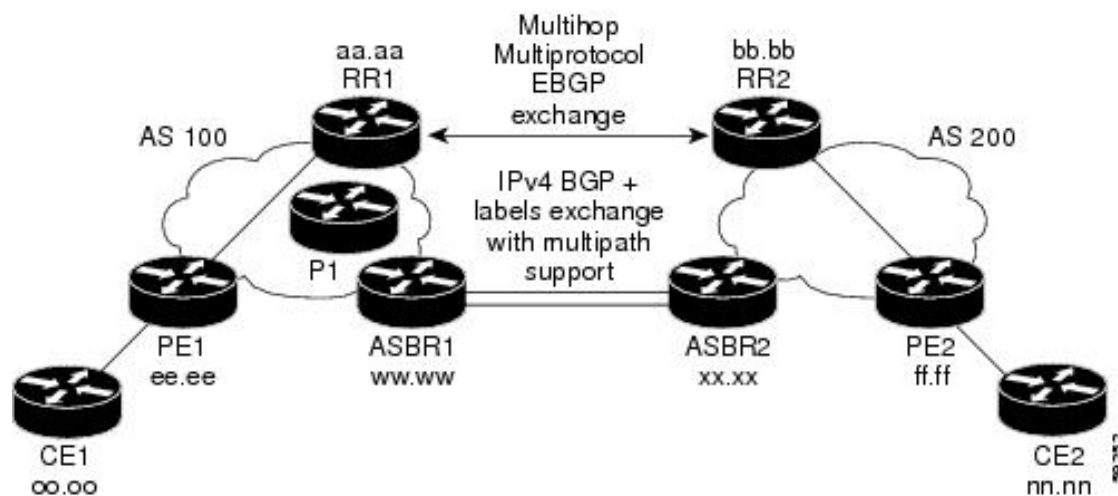
	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	router bgp as-number 例 : Device(config)# router bgp 100	ルータ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • as-number : 他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。有効値の範囲は 1 ~ 65535 です。内部ネットワークで使用できるプライベート自律システム番号の範囲は、64512 ~ 65535 です。 自律システム番号によって、他の自律システム内のルータで RRI が特定されます。
ステップ 4	address-family ipv4 [multicast unicast vrf vrf-name] 例 : Device(config-router)# address-family ipv4	標準 IPv4 アドレス プレフィックスを使用する BGP などのルーティングセッションを設定するために、アドレス ファミリ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • multicast キーワードでは、IPv4 マルチキャスト アドレス プレフィックスを指定します。 • unicast キーワードでは、IPv4 ユニキャスト アドレス プレフィックスを指定します。 • vrf vrf-name キーワードおよび引数では、後続の IPv4 アドレス ファミリ コンフィギュレーション モード コマンドに関連付ける VPN ルーティングおよび転送 (VRF) インスタンスの名前を指定します。
ステップ 5	neighbor ip-address route-map route-map-name out 例 : Device (config-router-af) # neighbor 209.165.200.225 route-map OUT out	着信ルートにルートマップを適用します。 <ul style="list-style-type: none"> • ip-address 引数では、ルートマップを適用するルートを指定します。 • route-map-name 引数では、ルートマップの名前を指定します。 • out キーワードでは、発信ルートにルートマップを適用します。

	コマンドまたはアクション	目的
ステップ 6	neighbor ip-address send-label 例： Device(config-router-af)# neighbor 209.165.200.225 send-label	ルートとともに MPLS ラベルを送信するルータの機能をアドバタイズします。 • ip-address 引数では、ルートとともに MPLS ラベルを送信できるルータを指定します。
ステップ 7	exit-address-family 例： Device(config-router-af)# exit-address-family	アドレスファミリサブモードを終了します。
ステップ 8	end 例： Device(config-router-af)# end	(任意) 終了して、特権 EXEC モードに戻ります。

MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定の確認

設定については、次の図を参照してください。

図 29: IPv4 ルートおよび MPLS ラベルを交換する 2 つの VPN サービス プロバイダーの設定



ルートリフレクタを使用して VPNv4 ルートを配布し、ASBR を使用して IPv4 ラベルを配布する場合は、次の手順に従って設定を確認します。

ルート リフレクタ設定の確認

ルートリフレクタ設定を確認するには、次の作業を実行します。

手順の概要

1. **enable**
2. **show ip bgp vpnv4 {all | rd route-distinguisher | vrf vrf-name} [summary] [labels]**
3. **disable**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	show ip bgp vpnv4 {all rd route-distinguisher vrf vrf-name} [summary] [labels] 例： Device# show ip bgp vpnv4 all summary 例： Device# show ip bgp vpnv4 all labels	（任意）BGP テーブルからの VPN アドレス情報を表示します。 <ul style="list-style-type: none"> • ルートリフレクタ間にマルチホップ、マルチプロトコル、EBGP セッションが存在し、ルートリフレクタ間で VPNv4 ルートが交換されていることを確認するには、all キーワードと summary キーワードを指定して、show ip bgp vpnv4 コマンドを使用します。 • コマンド出力の最後の 2 行に、次の情報が表示されます。 <ul style="list-style-type: none"> • プレフィックスが PE1 から学習されて RR2 に渡されていること。 • プレフィックスが RR2 から学習されて PE1 に渡されていること。 • ルートリフレクタ間で VPNv4 ラベル情報が交換されていることを確認するには、all キーワードと labels キーワードを指定して、show ip bgp vpnv4 コマンドを使用します。
ステップ 3	disable 例： Device# disable	（任意）終了して、ユーザー EXEC モードに戻ります。

CE1 に CE2 のネットワーク到達可能性情報があることの確認

ルータ CE1 がルータ CE2 の NLRI を持っていることを確認するには、次の作業を実行します。

PE1にCE2のネットワーク層到達可能性情報があることの確認

手順の概要

1. **enable**
2. **show ip route** [*ip-address* [*mask*] [**longer prefixes**]] | [*protocol* [*process-id*]] | [**list** *access-list-number* | *access-list-name*]
3. **disable**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show ip route [<i>ip-address</i> [<i>mask</i>] [longer prefixes]] [<i>protocol</i> [<i>process-id</i>]] [list <i>access-list-number</i> <i>access-list-name</i>] 例： Device# show ip route 209.165.201.1	ルーティングテーブルの現在の状態を表示します。 • ip-address 引数を指定して show ip route コマンドを使用して、CE1にCE2へのルートが含まれていることを確認します。 • show ip route コマンドを使用して、CE1が学習したルートを確認します。CE2へのルートがリストされていることを確認します。
ステップ 3	disable 例： Device# disable	(任意) 終了して、ユーザー EXEC モードに戻ります。

PE1にCE2のネットワーク層到達可能性情報があることの確認

ルータ PE1 がルータ CE2 の NLRI を持っていることを確認するには、次の作業を実行します。

手順の概要

1. **enable**
2. **show ip route vrf** *vrf-name* [**connected**] [*protocols* [*as-number*] [*tag*] [*output-modifiers*]] [**list** *number* [*output-modifiers*]] [**profile**] [**static** [*output-modifiers*]] [**summary** [*output-modifiers*]] [**supernets-only** [*output-modifiers*]] [**traffic engineering** [*output-modifiers*]]
3. **show ip bgp vpnv4** { **all** | **rd** *route-distinguisher* | **vrf** *vrf-name* } { *ip-prefix/length* [**longer-prefixes**] [*output-modifiers*]] [*network-address* [*mask*] [**longer-prefixes**] [*output-modifiers*]] [**cidr-only**] [*community*] [**community-list**] [**dampened-paths**] [**filter-list**] [**flap-statistics**] [**inconsistent-as**] [**neighbors**] [**path** [*line*]] [**peer-group**] [**quote-regexp**] [**regexp**] [**summary**] [**tags**]]
4. **show ip cef** [*vrf vrf-name*] [*network* [*mask*]] [**longer-prefixes**] [**detail**]
5. **show mpls forwarding-table** [{ *network* { *mask* | *length* } | **labels** *label* [-*label*] | **interface** *interface* | **next-hop** *address* | **lsp-tunnel** [*tunnel-id*] }] [**detail**]

6. **show ip bgp** [*network*] [*network-mask*] [**longer-prefixes**]
7. **show ip bgp vpnv4** {**all** | **rd route-distinguisher** | **vrf vrf-name**} [**summary**] [**labels**]
8. **disable**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	show ip route vrf vrf-name [connected] [<i>protocols</i>] [<i>as-number</i>] [<i>tag</i>] [<i>output-modifiers</i>]] [list number] [<i>output-modifiers</i>]] [profile] [static] [<i>output-modifiers</i>]] [summary] [<i>output-modifiers</i>]] [supernets-only] [<i>output-modifiers</i>]] [traffic engineering] [<i>output-modifiers</i>]] 例： Device# show ip route vrf vpn1 209.165.201.1	（任意）VRF に関連付けられている IP ルーティングテーブルを表示します。 <ul style="list-style-type: none"> • show ip route vrf コマンドを使用して、ルータ PE1 がルータ CE2 (nn.nn.nn.nn) からルートを学習していることを確認します。
ステップ 3	show ip bgp vpnv4 { all rd route-distinguisher vrf vrf-name } { <i>ip-prefix/length</i> } [longer-prefixes] [<i>output-modifiers</i>]] [<i>network-address</i>] [<i>mask</i>] [longer-prefixes] [<i>output-modifiers</i>]] [cidr-only] [<i>community</i>] [community-list] [dampened-paths] [filter-list] [flap-statistics] [inconsistent-as] [neighbors] [path] [<i>line</i>]] [peer-group] [quote-regexp] [regexp] [summary] [tags] 例： Device# show ip bgp vpnv4 vrf vpn1 209.165.201.1	（任意）BGP テーブルからの VPN アドレス情報を表示します。 <ul style="list-style-type: none"> • ルータ PE2 がルータ CE2 の BGP ネクストホップであることを確認するには、vrf または all キーワード指定して show ip bgp vpnv4 コマンドを使用します。
ステップ 4	show ip cef [vrf vrf-name] [<i>network</i>] [<i>mask</i>]] [longer-prefixes] [detail] 例： Device# show ip cef vrf vpn1 209.165.201.1	（任意）転送情報ベース（FIB）のエントリを表示するか、または FIB のサマリーを表示します。 <ul style="list-style-type: none"> • show ip cef コマンドを使用して、Cisco Express Forwarding (CEF) エントリが正しいことを確認します。
ステップ 5	show mpls forwarding-table [{ <i>network</i> { <i>mask</i> <i>length</i> } labels <i>label</i> [- <i>label</i>] interface <i>interface</i> next-hop address lsp-tunnel [<i>tunnel-id</i>] }] [detail] 例： Device# show mpls forwarding-table	（任意）MPLS 転送情報ベース（LFIB）の内容を表示します。 <ul style="list-style-type: none"> • show mpls forwarding-table コマンドを使用して、BGP ネクストホップルータ（自律システム境界）の IGP ラベルを確認します。

PE2 に CE2 のネットワーク到達可能性情報があることの確認

	コマンドまたはアクション	目的
ステップ 6	show ip bgp [network] [network-mask] [longer-prefixes] 例： Device# show ip bgp 209.165.202.129	(任意) BGP ルーティングテーブルのエントリを表示します。 • show ip bgp コマンドを使用して、リモート出力 PE ルータ (PE2) のラベルを確認します。
ステップ 7	show ip bgp vpnv4 {all rd route-distinguisher vrf vrf-name} [summary] [labels] 例： Device# show ip bgp vpnv4 all labels	(任意) BGP テーブルからの VPN アドレス情報を表示します。 • PE2 からアドバタイズされた CE2 の VPN ラベルを確認するには、 show ip bgp vpnv4 all summary コマンドを使用します。
ステップ 8	disable 例： Device# disable	(任意) 終了して、ユーザー EXEC モードに戻ります。

PE2 に CE2 のネットワーク到達可能性情報があることの確認

PE2 が CE2 にアクセスできることを確認するには、次の作業を実行します。

手順の概要

1. **enable**
2. **show ip route vrf vrf-name** [connected] [protocol [as-number] [tag] [output-modifiers]] [list number [output-modifiers]] [profile] [static [output-modifiers]] [summary [output-modifiers]] [supernets-only [output-modifiers]] [traffic-engineering [output-modifiers]]
3. **show mpls forwarding-table** [vrf vpn-name] [{network {mask | length} | labels label [-label] | interface interface | next-hop address | lsp-tunnel [tunnel-id] }] [detail]
4. **show ip bgp vpnv4** {all | rd route-distinguisher | vrf vrf-name} [summary] [labels]
5. **show ip cef** [vrf vrf-name] [network [mask]] [longer-prefixes] [detail]
6. **disable**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	<p>show ip route vrf vrf-name [connected] [<i>protocol</i> [<i>as-number</i>] [<i>tag</i>] [<i>output-modifiers</i>]] [list number [<i>output-modifiers</i>]] [profile] [static [<i>output-modifiers</i>]] [summary [<i>output-modifiers</i>]] [supernets-only [<i>output-modifiers</i>]] [traffic-engineering [<i>output-modifiers</i>]]</p> <p>例： Device# show ip route vrf vpn1 209.165.201.1</p>	<p>(任意) VRF に関連付けられている IP ルーティングテーブルを表示します。</p> <ul style="list-style-type: none"> CE2 の VPN ルーティングおよび転送テーブルを確認するには、show ip route vrf コマンドを使用します。出力にはネクストホップ情報が表示されます。
ステップ 3	<p>show mpls forwarding-table [vrf vpn-name] [{<i>network</i> {<i>mask</i> <i>length</i> } labels label [-<i>label</i>] interface interface next-hop address lsp-tunnel [<i>tunnel-id</i>]]} [detail]</p> <p>例： Device# show mpls forwarding-table vrf vpn1 209.165.201.1</p>	<p>(任意) LFIB の内容を表示します。</p> <ul style="list-style-type: none"> CE2 の VPN ルーティングおよび転送テーブルを確認するには、vrf キーワードを指定して show mpls forwarding-table コマンドを使用します。出力に、CE2 のラベルと発信インターフェイスが表示されます。
ステップ 4	<p>show ip bgp vpnv4 {all rd route-distinguisher vrf vrf-name} [summary] [labels]</p> <p>例： Device# show ip bgp vpnv4 all labels</p>	<p>(任意) BGP テーブルからの VPN アドレス情報を表示します。</p> <ul style="list-style-type: none"> マルチプロトコル BGP テーブル内の CE2 の VPN ラベルを確認するには、all および labels キーワードを指定して show ip bgp vpnv4 コマンドを使用します。
ステップ 5	<p>show ip cef [vrf vrf-name] [<i>network</i> [<i>mask</i>]] [longer-prefixes] [detail]</p> <p>例： Device# show ip cef <vrf-name> 209.165.201.1</p>	<p>(任意) 転送情報ベース (FIB) のエントリを表示するか、または FIB のサマリーを表示します。</p> <ul style="list-style-type: none"> CE2 の CEF エントリを確認するには、show ip cef コマンドを使用します。コマンド出力に、CE2 のローカルラベルと発信インターフェイスが表示されます。
ステップ 6	<p>disable</p> <p>例： Device# disable</p>	<p>(任意) 終了して、ユーザー EXEC モードに戻ります。</p>

ASBR の設定の確認

ASBR 間で、ルートマップの指定に従って MPLS ラベル付きの IPv4 ルートまたはラベルなしの IPv4 ルートが交換されていることを確認するには、次の作業を実行します。

手順の概要

1. **enable**
2. **show ip bgp** [*network*] [*network-mask*] [**longer-prefixes**]
3. **show ip cef** [**vrf vrf-name**] [*network* [*mask*]] [**longer-prefixes**] [**detail**]
4. **disable**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show ip bgp [<i>network</i>] [<i>network-mask</i>] [longer-prefixes] 例： Device# show ip bgp 209.165.202.129 例： Device# show ip bgp 192.0.2.1	(任意) BGP ルーティングテーブルのエントリを表示します。 • show ip bgp コマンドを使用して、次のことを確認します。 • ASBR1 が ASBR2 から PE2 の MPLS ラベルを受信していること。 • ASBR1 がラベルなしの RR2 の ASBR2 IPv4 ルートを受信していること。コマンド出力に MPLS ラベル情報が表示されない場合、MPLS ラベルなしでルートが受信されています。 • ASBR2 が ASBR1 に PE2 の MPLS ラベルを配布していること。 • ASBR2 が ASBR1 に RR2 のラベルを配布していないこと。
ステップ 3	show ip cef [vrf vrf-name] [<i>network</i> [<i>mask</i>]] [longer-prefixes] [detail] 例： Device# show ip cef 209.165.202.129 例： Device# show ip cef 192.0.2.1	(任意) 転送情報ベース (FIB) のエントリを表示するか、または FIB のサマリーを表示します。 • ASBR1 および ASBR2 から show ip cef コマンドを使用して、次のことを確認します。 • PE2 の CEF エントリが正しいこと。 • RR2 の CEF エントリが正しいこと。
ステップ 4	disable 例：	(任意) 終了して、ユーザー EXEC モードに戻ります。

コマンドまたはアクション	目的
Device# disable	

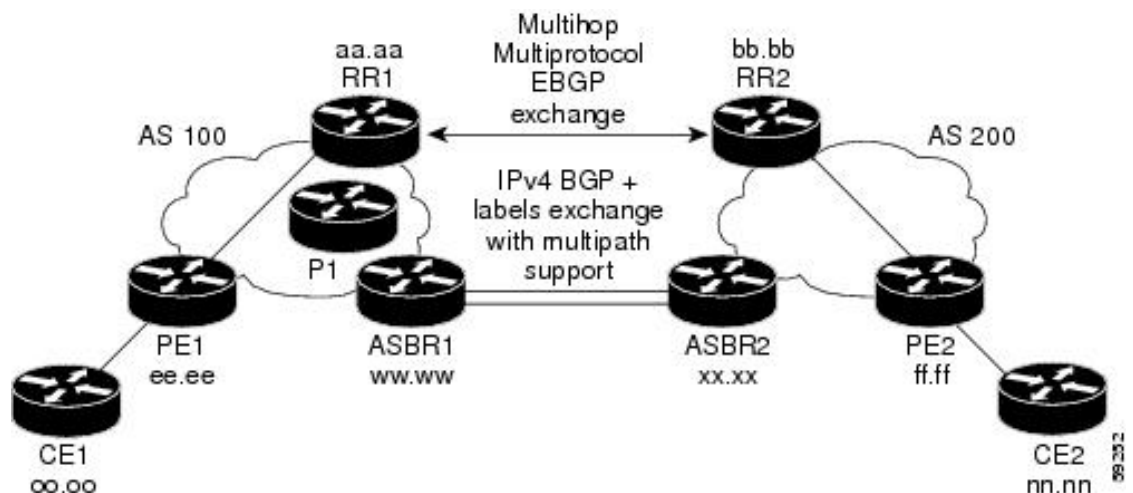
MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定例

MPLS VPN Inter-AS IPv4 BGP ラベル配布機能の設定例には、次のものがあります。

BGP を使用して MPLS VPN サービスプロバイダー経由でルートおよび MPLS ラベルを配布する Inter-AS の設定例

次の図に、2つの MPLS VPN サービスプロバイダーを示します。サービスプロバイダーは、ルートリフレクタ間で VPNv4 ルートを配布します。サービスプロバイダーは、ASBR 間で MPLS ラベル付きの IPv4 ルートを配布します。

図 30: MPLS VPN サービスプロバイダー間での IPv4 ルートと MPLS ラベルの配布



設定例では、リモートの RR と PE からローカルの RR と PE に、VPNv4 ルートおよび MPLS ラベル付きの IPv4 ルートを配布するために使用できる次の 2 つの技術を示しています。

- 自律システム 100 は、RR を使用して、リモート RR から学習した VPNv4 ルートを配布します。また、RR は、IPv4 ラベルを使用して、ASBR1 から学習したリモート PE アドレスとラベルを配布します。
- 自律システム 200 では、ASBR2 が学習した IPv4 ルートが IGP に再配布されます。

この項では、次の設定例を示します。

例：ルートリフレクタ 1 (MPLS VPN サービスプロバイダー)

RR1 の設定例では、次のことが指定されています。

- RR1 は、マルチプロトコル、マルチホップ EBGP を使用して、RR2 と VPNv4 ルートを交換します。
- VPNv4 ネクストホップ情報および VPN ラベルは、自律システム間で保存されます。
- RR1 から PE1 に次の内容が反映されます。
 - RR2 から学習した VPNv4 ルート
 - ASBR1 から学習した IPv4 ルートおよび MPLS ラベル

```

ip subnet-zero
ip cef
!
interface Loopback0
 ip address 10.0.0.1 255.255.255.255
 no ip directed-broadcast
!
interface Serial1/2
 ip address 209.165.201.8 255.0.0.0
 no ip directed-broadcast
 clockrate 124061
!
router ospf 10
 log-adjacency-changes
 auto-cost reference-bandwidth 1000
 network 10.0.0.1 0.0.0.0 area 100
 network 209.165.201.9 0.255.255.255 area 100
!
router bgp 100
 bgp cluster-id 1
 bgp log-neighbor-changes
 timers bgp 10 30
 neighbor 203.0.113.1 remote-as 100
 neighbor 203.0.113.1 update-source Loopback0
 neighbor 209.165.200.225 remote-as 100
 neighbor 209.165.200.225 update-source Loopback0
 neighbor 192.0.2.1 remote-as 200
 neighbor 192.0.2.1 ebgp-multihop 255
 neighbor 192.0.2.1 update-source Loopback0
 no auto-summary
!
address-family ipv4
 neighbor 203.0.113.1 activate
 neighbor 203.0.113.1 route-reflector-client                !IPv4+labels session to PE1

 neighbor 203.0.113.1 send-label
 neighbor 209.165.200.225 activate
 neighbor 209.165.200.225 route-reflector-client          !IPv4+labels session
to ASBR1
 neighbor 209.165.200.225 send-label
 no neighbor 192.0.2.1 activate
 no auto-summary
 no synchronization
 exit-address-family
!
address-family vpnv4
 neighbor 203.0.113.1 activate
 neighbor 203.0.113.1 route-reflector-client                !VPNv4 session with PE1
 neighbor 203.0.113.1 send-community extended
 neighbor 192.0.2.1 activate
 neighbor 192.0.2.1 next-hop-unchanged                    !MH-VPNv4 session with RR2

```

```

neighbor 192.0.2.1 send-community extended                !with next hop unchanged

exit-address-family
!
ip default-gateway 3.3.0.1
no ip classless
!
snmp-server engineID local 00000009020000D0584B25C0
snmp-server community public RO
snmp-server community write RW
no snmp-server ifindex persist
snmp-server packetsize 2048
!
end

```

設定例 : ASBR1 (MPLS VPN サービスプロバイダー)

ASBR1 は、ASBR2 と IPv4 ルートおよび MPLS ラベルを交換します。

この例では、ASBR1 で、次のルートマップを使用してルートがフィルタリングされています。

- OUT というルート マップでは、ASBR1 において、PE1 ルート (ee.aa) はラベルを付けて配布し、RR1 ルート (aa.aa) はラベルを付けずに配布する必要があることが指定されています。
- IN というルート マップでは、ASBR1 にラベル付きの PE2 ルート (ff.bb) とラベルなしの RR2 ルート (bb.bb) を受け入れさせるように指定しています。

```

ip subnet-zero
mpls label protocol tdp
!
interface Loopback0
 ip address 209.165.200.225 255.255.255.255
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet0/2
 ip address 209.165.201.6 255.0.0.0
 no ip directed-broadcast
 no ip mroute-cache
!
interface Ethernet0/3
 ip address 209.165.201.18 255.0.0.0
 no ip directed-broadcast
 no ip mroute-cache
 mpls label protocol ldp
 mpls ip
!router ospf 10
 log-adjacency-changes
 auto-cost reference-bandwidth 1000
 redistribute connected subnets
 passive-interface Ethernet0/2
 network 209.165.200.225 0.0.0.0 area 100
 network 209.165.201.9 0.255.255.255 area 100

router bgp 100
 bgp log-neighbor-changes
 timers bgp 10 30
 neighbor 10.0.0.1 remote-as 100
 neighbor 10.0.0.1 update-source Loopback0

```

設定例：ルータリフレクタ 2 (MPLS VPN サービスプロバイダー)

```

neighbor 209.165.201.2 remote-as 200
no auto-summary
!
address-family ipv4
  redistribute ospf 10
  neighbor 10.0.0.1 activate
  neighbor 10.0.0.1 send-label
  neighbor 209.165.201.2 activate
  neighbor 209.165.201.2 advertisement-interval 5
  neighbor 209.165.201.2 send-label
  neighbor 209.165.201.2 route-map IN in
  neighbor 209.165.201.2 route-map OUT out
  neighbor 209.165.201.3 activate
  neighbor 209.165.201.3 advertisement-interval 5
  neighbor 209.165.201.3 send-label
  neighbor 209.165.201.3 route-map IN in
  neighbor 209.165.201.3 route-map OUT out
no auto-summary
no synchronization
exit-address-family
!
ip default-gateway 3.3.0.1
ip classless
!
access-list 1 permit 203.0.113.1 log
access-list 2 permit 209.165.202.129 log
access-list 3 permit 10.0.0.1 log
access-list 4 permit 192.0.2.1 log

route-map IN permit 10
  match ip address 2
  match mpls-label
!
route-map IN permit 11
  match ip address 4
!
route-map OUT permit 12
  match ip address 3
!
route-map OUT permit 13
  match ip address 1
  set mpls-label
!
end

```

設定例：ルータリフレクタ 2 (MPLS VPN サービスプロバイダー)

RR2 は、マルチホップ、マルチプロトコル EBGP を使用して、RR1 と VPNv4 ルートを交換します。また、この設定では、自律システム間でネクストホップ情報および VPN ラベルが維持されるように指定されています。

```

ip subnet-zero
ip cef
!
interface Loopback0
  ip address 192.0.2.1 255.255.255.255
  no ip directed-broadcast
!
interface Serial1/1
  ip address 209.165.201.10 255.0.0.0
  no ip directed-broadcast
  no ip mroute-cache
!

```



```

router ospf 20
  log-adjacency-changes
  network 192.0.2.1 0.0.0.0 area 200
  network 209.165.201.20 0.255.255.255 area 200
!
router bgp 200
  bgp cluster-id 1
  bgp log-neighbor-changes
  timers bgp 10 30
  neighbor 10.0.0.1 remote-as 100
  neighbor 10.0.0.1 ebgp-multihop 255
  neighbor 10.0.0.1 update-source Loopback0
  neighbor 209.165.202.129 remote-as 200
  neighbor 209.165.202.129 update-source Loopback0
  no auto-summary
!
  address-family vpnv4
  neighbor 10.0.0.1 activate
  neighbor 10.0.0.1 next-hop-unchanged                !Multihop VPNv4 session with RR1
  neighbor 10.0.0.1 send-community extended          !with next-hop-unchanged
  neighbor 209.165.202.129 activate
  neighbor 209.165.202.129 route-reflector-client    !VPNv4 session with PE2
  neighbor 209.165.202.129 send-community extended
  exit-address-family
!
ip default-gateway 3.3.0.1
no ip classless
!
end

```

設定例 : ASBR2 (MPLS VPN サービスプロバイダー)

ASBR2 は、ASBR1 と IPv4 ルートおよび MPLS ラベルを交換します。ただし、ASBR1 とは異なり、ASBR2 は RR を使用して IPv4 ルートおよび MPLS ラベルを PE2 に反映しません。ASBR2 は、ASBR1 から学習した IPv4 ルートおよび MPLS ラベルを IGP に再配布します。これで、PE2 がこれらのプレフィックスに到達できるようになります。

```

ip subnet-zero
ip cef
!
mpls label protocol tdp
!
interface Loopback0
  ip address 209.165.200.226 255.255.255.255
  no ip directed-broadcast
!
interface Ethernet1/0
  ip address 209.165.201.2 255.0.0.0
  no ip directed-broadcast
  no ip mroute-cache
!
interface Ethernet1/2
  ip address 209.165.201.4 255.0.0.0
  no ip directed-broadcast
  no ip mroute-cache
  mpls label protocol tdp
  mpls ip
!
router ospf 20
  log-adjacency-changes
  auto-cost reference-bandwidth 1000
  redistribute connected subnets

```

```

redistribute bgp 200 subnets          ! Redistributing the routes learned from
passive-interface Ethernet1/0         ! ASBR1(EBGP+labels session) into IGP
network 209.165.200.226 0.0.0.0 area 200 ! so that PE2 will learn them
network 209.165.201.5 0.255.255.255 area 200
!
router bgp 200
  bgp log-neighbor-changes
  timers bgp 10 30
  neighbor 192.0.2.1 remote-as 200
  neighbor 192.0.2.1 update-source Loopback0
  neighbor 209.165.201.6 remote-as 100
  no auto-summary
!
address-family ipv4
  redistribute ospf 20                ! Redistributing IGP into BGP
  neighbor 209.165.201.6 activate      ! so that PE2 & RR2 loopbacks
  neighbor 209.165.201.6 advertisement-interval 5 ! will get into the BGP-4 table.
  neighbor 209.165.201.6 route-map IN in
  neighbor 209.165.201.6 route-map OUT out
  neighbor 209.165.201.6 send-label
  neighbor 209.165.201.7 activate
  neighbor 209.165.201.7 advertisement-interval 5
  neighbor 209.165.201.7 route-map IN in
  neighbor 209.165.201.7 route-map OUT out
  neighbor 209.165.201.7 send-label
  no auto-summary
  no synchronization
  exit-address-family
!
address-family vpnv4
  neighbor 192.0.2.1 activate
  neighbor 192.0.2.1 send-community extended
  exit-address-family
!
ip default-gateway 3.3.0.1
ip classless
!
access-list 1 permit 209.165.202.129 log          !Setting up the access lists
access-list 2 permit 203.0.113.1 log
access-list 3 permit 192.0.2.1 log
access-list 4 permit 10.0.0.1 log

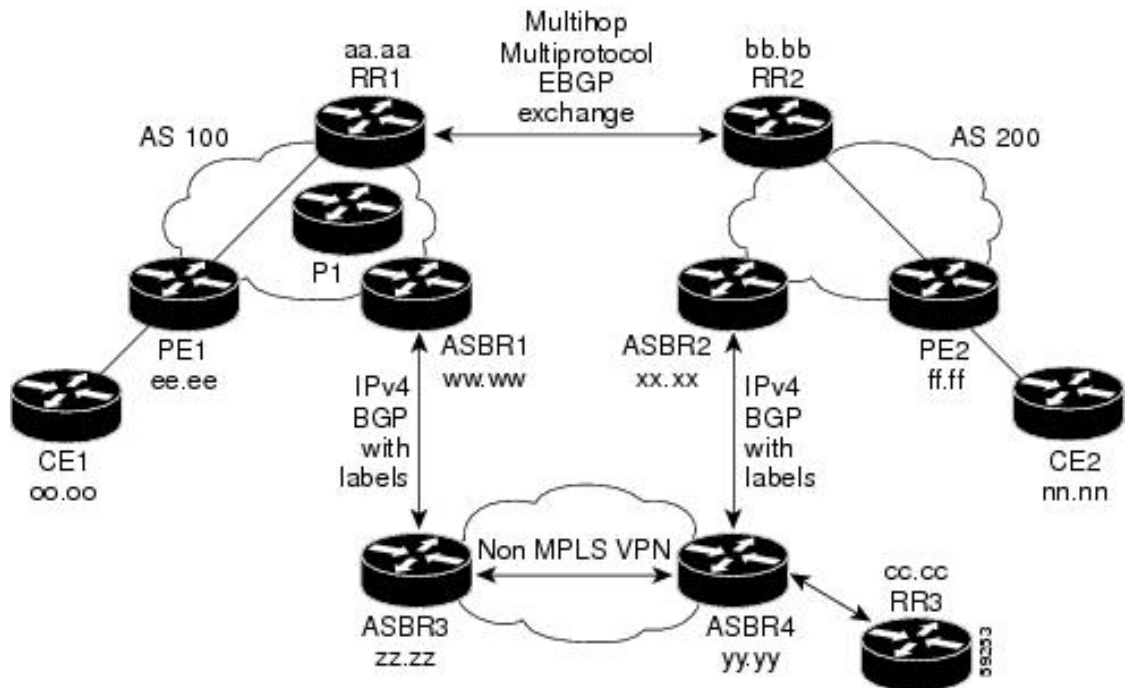
route-map IN permit 11                          !Setting up the route maps
  match ip address 2
  match mpls-label
!
route-map IN permit 12
  match ip address 4
!
route-map OUT permit 10
  match ip address 1
  set mpls-label
!
route-map OUT permit 13
  match ip address 3
end

```

設定例：BGP を使用して非 MPLS VPN サービスプロバイダー経由でルートおよび MPLS ラベルを配布する Inter-AS

次の図に、非 MPLS VPN サービスプロバイダー経由で接続された 2 つの MPLS VPN サービスプロバイダーを示します。ネットワークの中間にある自律システムは、Label Distribution Protocol (LDP; ラベル配布プロトコル) または Tag Distribution Protocol (TDP) を使用して MPLS ラベルを配布するバックボーン自律システムとして設定されます。また、TDP や LDP の代わりにトラフィック エンジニアリング トンネルを使用して、非 MPLS VPN サービスプロバイダーで LSP を構築できます。

図 31: 非 MPLS VPN サービスプロバイダー経由でのルートと MPLS ラベルの配布



ここでは、BGP を使用して非 MPLS VPN サービスプロバイダー経由でルートおよび MPLS ラベルを配布する Inter-AS の次の設定例について説明します。

設定例：ルータリフレクタ 1 (非 MPLS VPN サービスプロバイダー)

RR1 の設定例では、次のことが指定されています。

- RR1 は、マルチプロトコル、マルチホップ EBGP を使用して、RR2 と VPNv4 ルートを交換します。
- VPNv4 ネクストホップ情報および VPN ラベルは、自律システム間で保存されます。
- RR1 から PE1 に次の内容が反映されます。
 - RR2 から学習した VPNv4 ルート
 - ASBR1 から学習した IPv4 ルートおよび MPLS ラベル

```

ip subnet-zero
ip cef
!
interface Loopback0
ip address 10.0.0.1 255.255.255.255
no ip directed-broadcast
!
interface Serial1/2
ip address 209.165.201.8 255.0.0.0
no ip directed-broadcast
clockrate 124061
!
router ospf 10
log-adjacency-changes
auto-cost reference-bandwidth 1000
network 10.0.0.1 0.0.0.0 area 100
network 209.165.201.9 0.255.255.255 area 100
!
router bgp 100
bgp cluster-id 1
bgp log-neighbor-changes
timers bgp 10 30
neighbor 203.0.113.1 remote-as 100
neighbor 203.0.113.1 update-source Loopback0
neighbor 209.165.200.225 remote-as 100
neighbor 209.165.200.225 update-source Loopback0
neighbor 192.0.2.1 remote-as 200
neighbor 192.0.2.1 ebgp-multihop 255
neighbor 192.0.2.1 update-source Loopback0
no auto-summary
!
address-family ipv4
neighbor 203.0.113.1 activate
neighbor 203.0.113.1 route-reflector-client                !IPv4+labels session to PE1

neighbor 203.0.113.1 send-label
neighbor 209.165.200.225 activate
neighbor 209.165.200.225 route-reflector-client                !IPv4+labels session
to ASBR1
neighbor 209.165.200.225 send-label
no neighbor 192.0.2.1 activate
no auto-summary
no synchronization
exit-address-family
!
address-family vpnv4
neighbor 203.0.113.1 activate
neighbor 203.0.113.1 route-reflector-client                !VPNv4 session with PE1
neighbor 203.0.113.1 send-community extended
neighbor 192.0.2.1 activate
neighbor 192.0.2.1 next-hop-unchanged                    !MH-VPNv4 session with RR2
neighbor 192.0.2.1 send-community extended                with next-hop-unchanged

exit-address-family
!
ip default-gateway 3.3.0.1
no ip classless
!
snmp-server engineID local 00000009020000D0584B25C0
snmp-server community public RO
snmp-server community write RW
no snmp-server ifindex persist
snmp-server packetsize 2048

```

```
!
end
```

設定例 : ASBR1 (非 MPLS VPN サービスプロバイダー)

ASBR1 は、ASBR2 と IPv4 ルートおよび MPLS ラベルを交換します。

この例では、ASBR1 で、次のルートマップを使用してルートがフィルタリングされています。

- OUT というルート マップでは、ASBR1 において、PE1 ルート (ee.aa) はラベルを付けて配布し、RR1 ルート (aa.aa) はラベルを付けずに配布する必要があることが指定されています。
- IN というルート マップでは、ASBR1 にラベル付きの PE2 ルート (ff.aa) とラベルなしの RR2 ルート (bb.bb) を受け入れさせるように指定しています。

```
ip subnet-zero
ip cef distributed
mpls label protocol tdp
!
interface Loopback0
 ip address 209.165.200.225 255.255.255.255
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
interface Serial3/0/0
 ip address 209.165.201.7 255.0.0.0
 no ip directed-broadcast
 ip route-cache distributed
!
interface Ethernet0/3
 ip address 209.165.201.18 255.0.0.0
 no ip directed-broadcast
 no ip mroute-cache
 mpls label protocol ldp
 mpls ip
!
router ospf 10
 log-adjacency-changes
 auto-cost reference-bandwidth 1000
 redistribute connected subnets
 passive-interface Serial3/0/0
 network 209.165.200.225 0.0.0.0 area 100
 network dd.0.0.0 0.255.255.255 area 100

router bgp 100
 bgp log-neighbor-changes
 timers bgp 10 30
 neighbor 10.0.0.1 remote-as 100
 neighbor 10.0.0.1 update-source Loopback0
 neighbor kk.0.0.1 remote-as 200
 no auto-summary
!
 address-family ipv4
 redistribute ospf 10
 neighbor 10.0.0.1 activate ! Redistributing IGP into BGP
 neighbor 10.0.0.1 send-label ! so that PE1 & RR1 loopbacks
 neighbor 209.165.201.3 activate ! get into BGP table
 neighbor 209.165.201.3 advertisement-interval 5
 neighbor 209.165.201.3 send-label
```

設定例：ルータリフレクタ 2（非 MPLS VPN サービスプロバイダー）

```

neighbor 209.165.201.3 route-map IN in      ! Accepting routes specified in route map
IN
neighbor 209.165.201.3 route-map OUT out   ! Distributing routes specified in route map
OUT
no auto-summary
no synchronization
exit-address-family
!
ip default-gateway 3.3.0.1
ip classless
!
access-list 1 permit 203.0.113.1 log
access-list 2 permit 209.165.202.129 log
access-list 3 permit 10.0.0.1 log
access-list 4 permit 192.0.2.1 log
!
route-map IN permit 10
match ip address 2
match mpls-label
!
route-map IN permit 11
match ip address 4
!
route-map OUT permit 12
match ip address 3
!
route-map OUT permit 13
match ip address 1
set mpls-label
!
end

```

設定例：ルータリフレクタ 2（非 MPLS VPN サービスプロバイダー）

RR2 は、マルチホップ、マルチプロトコル EBGP を使用して、RR1 と VPNv4 ルートを交換します。また、この設定では、自律システム間でネクストホップ情報および VPN ラベルが維持されるように指定されています。

```

ip subnet-zero
ip cef
!
interface Loopback0
ip address 192.0.2.1 255.255.255.255
no ip directed-broadcast
!
interface Serial1/1
ip address 209.165.201.10 255.0.0.0
no ip directed-broadcast
no ip mroute-cache
!
router ospf 20
log-adjacency-changes
network 192.0.2.1 0.0.0.0 area 200
network 209.165.201.20 0.255.255.255 area 200
!
router bgp 200
bgp cluster-id 1
bgp log-neighbor-changes
timers bgp 10 30
neighbor 10.0.0.1 remote-as 100
neighbor 10.0.0.1 ebgp-multihop 255
neighbor 10.0.0.1 update-source Loopback0
neighbor 209.165.202.129 remote-as 200

```

```

neighbor 209.165.202.129 update-source Loopback0
no auto-summary
!
address-family vpnv4
neighbor 10.0.0.1 activate
neighbor 10.0.0.1 next-hop-unchanged                !MH vpnv4 session with RR1
neighbor 10.0.0.1 send-community extended          !with next-hop-unchanged
neighbor 209.165.202.129 activate
neighbor 209.165.202.129 route-reflector-client    !vpn4 session with PE2
neighbor 209.165.202.129 send-community extended
exit-address-family
!
ip default-gateway 3.3.0.1
no ip classless
!
end

```

設定例 : ASBR2 (非 MPLS VPN サービスプロバイダー)

ASBR2 は、ASBR1 と IPv4 ルートおよび MPLS ラベルを交換します。ただし、ASBR1 とは異なり、ASBR2 は RR を使用して IPv4 ルートおよび MPLS ラベルを PE2 に反映しません。ASBR2 は、ASBR1 から学習した IPv4 ルートおよび MPLS ラベルを IGP に再配布します。これで、PE2 がこれらのプレフィックスに到達できるようになります。

```

ip subnet-zero
ip cef
!
mpls label protocol tdp
!
interface Loopback0
ip address 209.165.200.226 255.255.255.255
no ip directed-broadcast
!
interface Ethernet0/1
ip address 209.165.201.11 255.0.0.0
no ip directed-broadcast
!
interface Ethernet1/2
ip address 209.165.201.4 255.0.0.0
no ip directed-broadcast
no ip mroute-cache
mpls label protocol tdp
mpls ip
!
router ospf 20
log-adjacency-changes
auto-cost reference-bandwidth 1000
redistribute connected subnets
redistribute bgp 200 subnets                !redistributing the routes learned from
passive-interface Ethernet0/1                !ASBR2 (EBGP+labels session) into IGP
network 209.165.200.226 0.0.0.0 area 200      !so that PE2 will learn them
network 209.165.201.5 0.255.255.255 area 200
!
router bgp 200
bgp log-neighbor-changes
timers bgp 10 30
neighbor 192.0.2.1 remote-as 200
neighbor 192.0.2.1 update-source Loopback0
neighbor 209.165.201.21 remote-as 100
no auto-summary
!
address-family ipv4                            ! Redistributing IGP into BGP

```

```

redistribute ospf 20                                ! so that PE2 & RR2 loopbacks
neighbor 209.165.201.21 activate                    ! will get into the BGP-4 table
neighbor 209.165.201.21 advertisement-interval 5
neighbor 209.165.201.21 route-map IN in
neighbor 209.165.201.21 route-map OUT out
neighbor 209.165.201.21 send-label
no auto-summary
no synchronization
exit-address-family
!
address-family vpnv4
neighbor 192.0.2.1 activate
neighbor 192.0.2.1 send-community extended
exit-address-family
!
ip default-gateway 3.3.0.1
ip classless
!
access-list 1 permit 209.165.202.129 log
access-list 2 permit 203.0.113.1 log
access-list 3 permit 192.0.2.1 log
access-list 4 permit 10.0.0.1 log
!
route-map IN permit 11
match ip address 2
match mpls-label
!
route-map IN permit 12
match ip address 4
!
route-map OUT permit 10
match ip address 1
set mpls-label
!
route-map OUT permit 13
match ip address 3
!
end

```

設定例 : ASBR3 (非 MPLS VPN サービスプロバイダー)

ASBR3 は、非 MPLS VPN サービスプロバイダーに属しています。ASBR3 は、ASBR1 との間で IPv4 ルートおよび MPLS ラベルを交換します。また、ASBR3 は、ASBR1 から学習したルートを RR3 経由で ASBR3 に渡します。



(注) IBGP を使用してルートおよびラベルを配布する場合は、学習した EBGP ルートを IBGP に再配布しないでください。このような設定はサポートされていません。

```

ip subnet-zero
ip cef
!
interface Loopback0
ip address 209.165.200.227 255.255.255.255
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
!
ip routing

```



```
mpls label protocol ldp
mpls ldp router-id Loopback0 force

interface GigabitEthernet1/0/1
ip address 209.165.201.12 255.0.0.0

interface TenGigabitEthernet1/1/1
no switchport
ip address 209.165.201.3 255.0.0.0
load-interval 30
mpls ip

!
router ospf 30
log-adjacency-changes
auto-cost reference-bandwidth 1000
redistribute connected subnets
network 209.165.200.227 0.0.0.0 area 300
network 209.165.201.13 0.255.255.255 area 300
!
router bgp 300
bgp log-neighbor-changes
timers bgp 10 30
neighbor 10.0.0.3 remote-as 300
neighbor 10.0.0.3 update-source Loopback0
neighbor 209.165.201.7 remote-as 100
no auto-summary
!
address-family ipv4
neighbor 10.0.0.3 activate          ! IBGP+labels session with RR3
neighbor 10.0.0.3 send-label
neighbor 209.165.201.7 activate    ! EBGP+labels session with ASBR1
neighbor 209.165.201.7 advertisement-interval 5
neighbor 209.165.201.7 send-label
neighbor 209.165.201.7 route-map IN in
neighbor 209.165.201.7 route-map OUT out
no auto-summary
no synchronization
exit-address-family
!
ip classless
!
access-list 1 permit 203.0.113.1 log
access-list 2 permit 209.165.202.129 log
access-list 3 permit 10.0.0.1 log
access-list 4 permit 192.0.2.1 log
!
route-map IN permit 10
match ip address 1
match mpls-label
!
route-map IN permit 11
match ip address 3
!
route-map OUT permit 12
match ip address 2
set mpls-label
!
route-map OUT permit 13
match ip address 4
!
ip default-gateway 3.3.0.1
ip classless
```

設定例：ルータリフレクタ 3（非 MPLS VPN サービスプロバイダー）

```
!
end
```

設定例：ルータリフレクタ 3（非 MPLS VPN サービスプロバイダー）

RR3 は、MPLS ラベル付きの IPv4 ルートを ASBR3 および ASBR4 に反映する非 MPLS VPN RR です。

```
ip subnet-zero
mpls label protocol tdp
mpls traffic-eng auto-bw timers
no mpls ip
!
interface Loopback0
 ip address 10.0.0.3 255.255.255.255
 no ip directed-broadcast
!
interface POS0/2
 ip address 209.165.201.15 255.0.0.0
 no ip directed-broadcast
 no ip route-cache cef
 no ip route-cache
 no ip mroute-cache
 crc 16
 clock source internal
!
router ospf 30
 log-adjacency-changes
 network 10.0.0.3 0.0.0.0 area 300
 network 209.165.201.16 0.255.255.255 area 300
!
router bgp 300
 bgp log-neighbor-changes
 neighbor 209.165.201.2 remote-as 300
 neighbor 209.165.201.2 update-source Loopback0
 neighbor 209.165.200.227 remote-as 300
 neighbor 209.165.200.227 update-source Loopback0
 no auto-summary
!
address-family ipv4
 neighbor 209.165.201.2 activate
 neighbor 209.165.201.2 route-reflector-client
 neighbor 209.165.201.2 send-label ! IBGP+labels session with ASBR3
 neighbor 209.165.200.227 activate
 neighbor 209.165.200.227 route-reflector-client
 neighbor 209.165.200.227 send-label ! IBGP+labels session with ASBR4
 no auto-summary
 no synchronization
 exit-address-family
!
ip default-gateway 3.3.0.1
ip classless
!
end
```

設定例：ASBR4（非 MPLS VPN サービスプロバイダー）

ASBR4 は、非 MPLS VPN サービスプロバイダーに属しています。ASBR4 と ASBR3 は、RR3 経由で IPv4 ルートと MPLS ラベルを交換します。



- (注) IBGP を使用してルートおよびラベルを配布する場合は、学習した EBGP ルートを IBGP に再配布しないでください。このような設定はサポートされていません。

```

ip subnet-zero
ip cef distributed
!
interface Loopback0
 ip address 209.165.201.2 255.255.255.255
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet0/2
 ip address 209.165.201.21 255.0.0.0
 no ip directed-broadcast
 no ip mroute-cache
!
ip routing
mpls label protocol ldp
mpls ldp router-id Loopback0 force

interface GigabitEthernet1/0/1
ip address 209.165.201.17 255.0.0.0

interface TenGigabitEthernet1/1/1
no switchport
ip address 209.165.201.14 255.0.0.0
load-interval 30
mpls ip

!
router ospf 30
 log-adjacency-changes
 auto-cost reference-bandwidth 1000
 redistribute connected subnets
 passive-interface Ethernet0/2
 network 209.165.201.2 0.0.0.0 area 300
 network 209.165.201.16 0.255.255.255 area 300
 network 209.165.201.13 0.255.255.255 area 300
!
router bgp 300
 bgp log-neighbor-changes
 timers bgp 10 30
 neighbor 10.0.0.3 remote-as 300
 neighbor 10.0.0.3 update-source Loopback0
 neighbor 209.165.201.11 remote-as 200
 no auto-summary
!
 address-family ipv4
  neighbor 10.0.0.3 activate
  neighbor 10.0.0.3 send-label
  neighbor 209.165.201.11 activate
  neighbor 209.165.201.11 advertisement-interval 5
  neighbor 209.165.201.11 send-label
  neighbor 209.165.201.11 route-map IN in
  neighbor 209.165.201.11 route-map OUT out
 no auto-summary
 no synchronization
 exit-address-family

```

```

!
ip classless
!
access-list 1 permit 209.165.202.129 log
access-list 2 permit 203.0.113.1 log
access-list 3 permit 192.0.2.1 log
access-list 4 permit 10.0.0.1 log
!
route-map IN permit 10
  match ip address 1
  match mpls-label
!
route-map IN permit 11
  match ip address 3
!
route-map OUT permit 12
  match ip address 2
  set mpls-label
!
route-map OUT permit 13
  match ip address 4
!
ip default-gateway 3.3.0.1
ip classless
!
end

```

MPLS VPN Inter-AS IPv4 BGP ラベル配布の設定の機能履歴

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

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

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.11.1	MPLS VPN Inter-AS IPv4 BGP ラベル配布	この機能を使用すると、バーチャルプライベートネットワーク (VPN) サービスプロバイダーネットワークを設定できます。このネットワークでは、自律システム境界ルータ (ASBR) が、プロバイダーエッジ (PE) ルータのマルチプロトコルラベルスイッチング (MPLS) ラベル付きの IPv4 ルートを交換します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。



第 17 章

シームレス MPLS の設定

- [シームレス MPLS に関する情報 \(265 ページ\)](#)
- [シームレス MPLS の設定方法 \(267 ページ\)](#)
- [シームレス MPLS の設定例 \(273 ページ\)](#)
- [シームレス MPLS の機能履歴 \(275 ページ\)](#)

シームレス MPLS に関する情報

次の項では、シームレス MPLS について説明します。

シームレス MPLS の概要

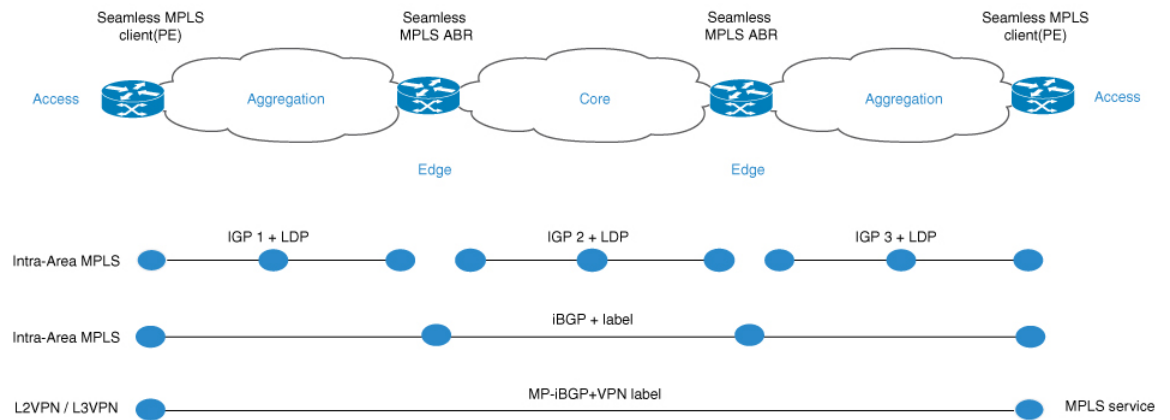
シームレス MPLS は、複数のネットワークを単一の MPLS ドメインに統合するための、非常に柔軟でスケーラブルなアーキテクチャを提供します。これは、既存の既知のプロトコルに基づいています。

大規模な MPLS ネットワークでは、ネットワークのさまざまな部分に複数のタイプのプラットフォームとサービスを配置できます。このようなネットワークは、通常、コアエリアと集約エリアなどのエリアに分割され、各エリアに異なる内部ゲートウェイプロトコル (IGP) があります。あるエリアの IGP プレフィックスを別のエリアに配布することはできません。IGP プレフィックスを配布できない場合、エンドツーエンドのラベルスイッチパス (LSP) は確立できません。これは、ネットワークの拡張性に影響します。

シームレス MPLS では、エンドツーエンド LSP を確立することで、拡張性が向上します。シームレス MPLS は、プロバイダーエッジ (PE) ルータのループバックプレフィックスを転送するために、IGP ではなくボーダー ゲートウェイ プロトコル (BGP) を使用します。BGP は、プレフィックスをエンドツーエンドで配布します。これにより、あるドメインの IGP プレフィックスを別のドメインにインストールする必要がなくなります。

シームレス MPLS は、サービスプレーンとトランスポートプレーンの分離を導入し、エンドツーエンドのサービスに依存しないトランスポートを提供します。これにより、ネットワークトランスポート ノードでサービス固有の設定が不要になります。

シームレス MPLS のアーキテクチャ



図は、3つの異なるエリア（1つのコアエリアと2つの集約エリア）があるネットワークを示しています。各エリアでは独自のIGPが実行され、エリア境界ルータ（ABR）ではエリア間の再配布は行われません。エンドツーエンドMPLS LSPを提供するためには、BGPを使用する必要があります。BGPは、ドメイン全体にラベルを付けてPEルータのループバックをアドバタイズし、エンドツーエンドLSPを提供します。BGPはPEとABRの間に導入されます。

シームレスMPLSは、BGPを使用してエンドツーエンドMPLS LSPを提供します。BGPはPEとABRの間に導入されます。BGPはIPv4プレフィックスとラベルを送信します。BGPは、ドメイン全体にラベルを付けてPEルータのループバックをアドバタイズし、エンドツーエンドLSPを提供します。

ネットワークでIGPを使用する場合、プレフィックスのネクストホップアドレスはPEルータのループバックプレフィックスです。このプレフィックスは、ネットワークの他の部分で使用されているIGPには認識されません。ネクストホップアドレスは、IGPプレフィックスへの再帰には使用できません。これを回避するために、プレフィックスはBGPで伝送されます。ABRはルートリフレクタ（RR）として設定されます。RRは、反映されたiBGPプレフィックスの場合でも、ネクストホップをRR自体に設定するように設定されます。

次の2つのシナリオが考えられます。

- ABRは、ABRによってネットワークの集約部分にアドバタイズされる（BGPによって反映される）プレフィックスのネクストホップをABR自体に設定しません。ABRは、コアIGPから集約IGPにABRのループバックプレフィックスを再配布する必要があります。（コアからの）ABRループバックプレフィックスのみを集約部分にアドバタイズする必要があります。リモート集約部分からのPEルータのループバックプレフィックスは不要です。
- ABRは、ABRによって集約部分にアドバタイズされた（BGPによって反映された）プレフィックスのネクストホップをABR自体に設定します。このため、ABRはABRのループバックプレフィックスをコアIGPから集約IGPに再配布する必要はありません。

いずれのシナリオでも、ABR は、ABR によってネットワークの集約部分からコア部分にアドバタイズされた（BGP によって反映された）プレフィックスのネクストホップを ABR 自体に設定します。

シームレス MPLS の設定方法

次の項では、シームレス MPLS の設定方法について説明します。

PE ルータでのシームレス MPLS の設定

次の手順を使用して、PE ルータでシームレス MPLS を設定できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface loopback slot/port**
4. **ip address ip-address subnet-mask**
5. **interface ethernet slot/port**
6. **no ip address**
7. **xconnect peer-ip-address vcid encapsulation mpls**
8. **router ospf process-id**
9. **network ip-address wild-mask area area-id**
10. **network ip-address wild-mask area area-id**
11. **router bgp autonomous-system-number**
12. **bgp log neighbor changes**
13. **address-family ipv4**
14. **network network-number mask network-mask**
15. **no bgp default ipv4 unicast**
16. **no bgp default route-target filter**
17. **neighbor ip-address remote-as autonomous-system-number**
18. **neighbor ip-address update-source interface-type interface-number]**
19. **neighbor ip-address send-label**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例：	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
	Device# configure terminal	
ステップ 3	interface loopback slot/port 例： Device(config-if)# interface Loopback0	ループバックインターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	ip address ip-address subnet-mask 例： Device(config-if) ip address 10.100.1.4 255.255.255.255	インターフェイスの IP アドレスを入力します。
ステップ 5	interface ethernet slot/port 例： Device(config-if)# interface Ethernet1/0	イーサネットインターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	no ip address 例： Device(config-if)# no ip address	IP アドレス定義を削除します。
ステップ 7	xconnect peer-ip-address vcid encapsulation mpls 例： Device(config-if)# xconnect 10.100.1.5 100 encapsulation mpls	カプセル化するためのトンネリング方式として MPLS を指定します。
ステップ 8	router ospf process-id 例： Device(config)# router ospf 2	OSPF ルーティング プロセスを設定します。
ステップ 9	network ip-address wild-mask area area-id 例： Device(config-router)# network 10.2.0.0 0.0.255.255 area 0	OSPF を実行するインターフェイスを定義し、それらのインターフェイスに対するエリア ID を定義します。
ステップ 10	network ip-address wild-mask area area-id 例： Device(config-router)# network 10.100.1.4 0.0.0.0 area 0	OSPF を実行するインターフェイスを定義し、それらのインターフェイスに対するエリア ID を定義します。
ステップ 11	router bgp autonomous-system-number 例： Device(config)# router bgp 1	BGP ルーティング プロセスを設定します。
ステップ 12	bgp log neighbor changes 例：	BGP ネイバーリセットのロギングを有効にします。

	コマンドまたはアクション	目的
	Device(config-router)# <code>bgp log neighbor changes</code>	
ステップ 13	address-family ipv4 例 : Device(config-router)# <code>address-family ipv4</code>	アドレスファミリ コンフィギュレーションモードを開始します。
ステップ 14	network network-number mask network-mask 例 : Device(config-router-af)# <code>network 10.100.1.4 mask 255.255.255.255</code>	BGP およびマルチプロトコル BGP ルーティングプロセスによってアドバタイズされるネットワークを指定します。
ステップ 15	no bgp default ipv4 unicast 例 : Device(config-router-af)# <code>no bgp default ipv4 unicast</code>	ピアリングセッションを確立するためのデフォルトの IPv4 ユニキャストアドレスファミリを無効にします。
ステップ 16	no bgp default route-target filter 例 : Device(config-router-af)# <code>no bgp default route-target filter</code>	BGP の route-target コミュニティフィルタリングを無効にします。
ステップ 17	neighbor ip-address remote-as autonomous-system-number 例 : Device(config-router-af)# <code>neighbor 10.100.1.1 remote-as 1</code>	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。
ステップ 18	neighbor ip-address update-source interface-type interface-number] 例 : Device(config-router-af)# <code>neighbor 10.100.1.1 update-source Loopback0</code>	BGP セッションが、TCP 接続の動作インターフェイスを使用できるようにします。
ステップ 19	neighbor ip-address send-label 例 : Device(config-router-af)# <code>neighbor 10.100.1.1 send-label</code>	BGP ルートとともに MPLS ラベルをネイバー BGP ルータに送信できるように BGP ルータを設定します。

ルータリフレクタでのシームレス MPLS の設定

次の手順を使用して、ルータリフレクタでシームレス MPLS を設定できます。

手順の概要

1. `enable`
2. `configure terminal`

3. **interface loopback** *slot/port*
4. **ip address** *ip-address subnet-mask*
5. **router ospf** *process-id*
6. **network** *ip-address wild-mask area area-id*
7. **network** *ip-address wild-mask area area-id*
8. **exit**
9. **router ospf** *process-id*
10. **redistribute ospf** *instance-tag route-map map-name*
11. **network** *ip-address wild-mask area area-id*
12. **exit**
13. **router bgp** *autonomous-system-number*
14. **bgp log neighbor changes**
15. **address-family ipv4**
16. **neighbor** *ip-address remote-as autonomous-system-number*
17. **neighbor** *ip-address update-source interface-type interface-number*]
18. **neighbor** *ip-address***next-hop-self all**
19. **neighbor** *ip-address***send-label**
20. **neighbor** *ip-address remote-as autonomous-system-number*
21. **neighbor** *ip-address update-source interface-type interface-number*]
22. **neighbor** *ip-address***route-reflector-client**
23. **neighbor** *ip-address***next-hop-self all**
24. **neighbor** *ip-address***send-label**
25. **exit**
26. **ip prefix-list** *name seq number permit prefix*
27. **route-map** *name permit sequence-number*
28. **match ip address prefix-list** *prefix-list-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface loopback <i>slot/port</i> 例： Device(config-if)# interface Loopback0	ループバック インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	ip address <i>ip-address subnet-mask</i> 例 : Device(config-if)# ip address 10.100.1.1 255.255.255.255	インターフェイスの IP アドレスを入力します。
ステップ 5	router ospf <i>process-id</i> 例 : Device(config)# router ospf 1	OSPF ルーティング プロセスを設定します。
ステップ 6	network <i>ip-address wild-mask area area-id</i> 例 : Device(config-router)# network 10.1.0.0 0.0.255.255 area 0	OSPF を実行するインターフェイスを定義し、それらのインターフェイスに対するエリア ID を定義します。
ステップ 7	network <i>ip-address wild-mask area area-id</i> 例 : Device(config-router)# 10.100.1.1 0.0.0.0 area 0	OSPF を実行するインターフェイスを定義し、それらのインターフェイスに対するエリア ID を定義します。
ステップ 8	exit 例 : Device(config-router)#exit	コンフィギュレーション モードを終了します。
ステップ 9	router ospf <i>process-id</i> 例 : Device(config)# router ospf 2	OSPF ルーティング プロセスを設定します。
ステップ 10	redistribute ospf <i>instance-tag route-map map-name</i> 例 : Device(config-router)# redistribute ospf 1 subnets match internal route-map ospf1-into-ospf2	1 つのルーティングドメインから OSPF にルートを注入します。
ステップ 11	network <i>ip-address wild-mask area area-id</i> 例 : Device(config-router)# network 10.2.0.0 0.0.255.255 area 0	OSPF を実行するインターフェイスを定義し、それらのインターフェイスに対するエリア ID を定義します。
ステップ 12	exit 例 : Device(config-router)#exit	コンフィギュレーション モードを終了します。
ステップ 13	router bgp <i>autonomous-system-number</i> 例 : Device(config)# router bgp 1	BGP ルーティング プロセスを設定します。

	コマンドまたはアクション	目的
ステップ 14	bgp log neighbor changes 例 : Device(config-router)# bgp log neighbor changes	BGP ネイバー リセットのロギングを有効にします。
ステップ 15	address-family ipv4 例 : Device(config-router)# address family ipv4	アドレスファミリー 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	neighbor ip-address remote-as autonomous-system-number 例 : Device(config-route-af)# neighbor 10.100.1.2 remote-as 1	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。
ステップ 17	neighbor ip-address update-source interface-type interface-number] 例 : Device(config-router-af)# neighbor 10.100.1.2 update-source Loopback0	BGP セッションが、TCP 接続の動作インターフェイスを使用できるようにします。
ステップ 18	neighbor ip-address next-hop-self all 例 : Device(config-router-af)# neighbor 10.100.1.2 next-hop-self all	ルータを BGP スピーキングネイバーまたはピアグループのネクストホップとして設定します。
ステップ 19	neighbor ip-address send-label 例 : Device(config-router-af)# neighbor 10.100.1.2 send-label	BGP ルートとともに MPLS ラベルをネイバー BGP ルータに送信できるように BGP ルータを設定します。
ステップ 20	neighbor ip-address remote-as autonomous-system-number 例 : Device(config-router-af)# neighbor 10.100.1.4 remote-as 1	BGP ネイバー テーブルまたはマルチプロトコル BGP ネイバー テーブルにエントリを追加します。
ステップ 21	neighbor ip-address update-source interface-type interface-number] 例 : Device(config-router-af)# neighbor 10.100.1.4 update-source Loopback0	BGP セッションが、TCP 接続の動作インターフェイスを使用できるようにします。
ステップ 22	neighbor ip-address route-reflector-client 例 : Device(config-router-af)# neighbor 10.100.1.4 route-reflector-client	ルータを BGP ルータリフレクタとして設定し、指定したネイバーをそのクライアントとして設定します。

	コマンドまたはアクション	目的
ステップ 23	neighbor ip-address next-hop-self all 例 : Device(config-router-af)# neighbor 10.100.1.4 next-hop-self all	ルータを BGP スピーキングネイバーまたはピアグループのネクストホップとして設定します。
ステップ 24	neighbor ip-address send-label 例 : Device(config-router-af)# neighbor 10.100.1.4 send-label	BGP ルートとともに MPLS ラベルをネイバー BGP ルータに送信できるように BGP ルータを設定します。
ステップ 25	exit 例 : Device(config-router)#exit	コンフィギュレーション モードを終了します。
ステップ 26	ip prefix-list name seq number permit prefix 例 : Device(config)# ip prefix-list prefix-list-ospf1-into-ospf2 seq 5 permit 10.100.1.1/32	IP パケットまたはルートと照合するプレフィックスリストを作成します。
ステップ 27	route-map name permit sequence-number 例 : Device(config)# route-map ospf1-into-ospf2 permit 10	ルートマップのエントリを作成します。ルートマップ コンフィギュレーション モードを開始します。
ステップ 28	match ip address prefix-list prefix-list-name 例 : Device(config-route-map)# match ip address prefix-list prefix-list-ospf1-into-ospf2	プレフィックスリストで許可された宛先 IP ネットワーク番号アドレスを含むルートを配布します。

シームレス MPLS の設定例

次の項に、シームレス MPLS の設定例を示します。

例 : PE ルータ 1 でのシームレス MPLS の設定

次に、PE ルータ 1 でシームレス MPLS を設定する例を示します。

```
Device(config-if)#interface Loopback0
Device(config-if)#ip address 10.100.1.4 255.255.255.255
!
Device(config-if)# interface Ethernet1/0
Device(config-if)# no ip address
Device(config-if)# xconnect 10.100.1.5 100 encapsulation mpls
!
Device(config)# router ospf 2
```

例：ルートリフレクタ 1 でのシームレス MPLS の設定

```

Device(config-router)# network 10.2.0.0 0.0.255.255 area 0
Device(config-router)# network 10.100.1.4 0.0.0.0 area 0
!
Device(config)#router bgp 1
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# address family ipv4
Device(config-router-af)# network 10.100.1.4 mask 255.255.255.255
Device(config-router-af)# no bgp default ipv4 unicast
Device(config-router-af)# no bgp default route-target filter
Device(config-router-af)# neighbor 10.100.1.1 remote-as 1
Device(config-router-af)# neighbor 10.100.1.1 update-source Loopback0
Device(config-router-af)# neighbor 10.100.1.1 send-label

```

例：ルートリフレクタ 1 でのシームレス MPLS の設定

次に、ルートリフレクタ 1 にシームレス MPLS を設定する例を示します。

```

Device(config-if)# interface Loopback0
Device(config-if)# ip address 10.100.1.1 255.255.255.255
Device(config)# router ospf 1
Device(config-router)# network 10.1.0.0 0.0.255.255 area 0
Device(config-router)# network 10.100.1.1 0.0.0.0 area 0
!
Device(config)# router ospf 2
Device(config-router)# redistribute ospf 1 subnets match internal route-map
ospf1-into-ospf2
Device(config-router)# network 10.2.0.0 0.0.255.255 area 0
!
Device(config)# router bgp 1
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# address family ipv4
Device(config-router-af)# neighbor 10.100.1.2 remote-as 1
Device(config-router-af)# neighbor 10.100.1.2 update-source Loopback0
Device(config-router-af)# neighbor 10.100.1.2 next-hop-self all
Device(config-router-af)# neighbor 10.100.1.2 send-label
Device(config-router-af)# neighbor 10.100.1.4 remote-as 1
Device(config-router-af)# neighbor 10.100.1.4 update-source Loopback0
Device(config-router-af)# neighbor 10.100.1.4 route-reflector-client
Device(config-router-af)# neighbor 10.100.1.4 next-hop-self all
Device(config-router-af)# neighbor 10.100.1.4 send-label

Device(config)# ip prefix-list prefix-list-ospf1-into-ospf2 seq 5 permit 10.100.1.1/32

Device(config)# route-map ospf1-into-ospf2 permit 10
Device(config-route-map)# match ip address prefix-list prefix-list-ospf1-into-ospf2

```

例：PE ルータ 2 でのシームレス MPLS の設定

次に、PE ルータ 2 でシームレス MPLS を設定する例を示します。

```

Device(config-if)#interface Loopback0
Device(config-if)#ip address 10.100.1.5 255.255.255.255
!
Device(config-if)# interface Ethernet1/0
Device(config-if)# no ip address
Device(config-if)# xconnect 10.100.1.4 100 encapsulation mpls
!
Device(config)# router ospf 3
Device(config-router)# network 10.3.0.0 0.0.255.255 area 0
Device(config-router)# network 10.100.1.5 0.0.0.0 area 0
!

```



```
Device(config)#router bgp 1
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# address family ipv4
Device(config-router-af)# network 10.100.1.5 mask 255.255.255.255
Device(config-router-af)# no bgp default ipv4 unicast
Device(config-router-af)# no bgp default route-target filter
Device(config-router-af)# neighbor 10.100.1.2 remote-as 1
Device(config-router-af)# neighbor 10.100.1.2 update-source Loopback0
Device(config-router-af)# neighbor 10.100.1.2 send-label
```

例：ルータリフレクタ 2 でのシームレス MPLS の設定

次に、ルータリフレクタ 2 にシームレス MPLS を設定する例を示します。

```
Device(config-if)# interface Loopback0
Device(config-if)# ip address 10.100.1.2 255.255.255.255
Device(config)# router ospf 1
Device(config-router)# network 10.1.0.0 0.0.255.255 area 0
Device(config-router)# network 10.100.1.2 0.0.0.0 area 0
!
Device(config)# router ospf 3
Device(config-router)# redistribute ospf 1 subnets match internal route-map
ospf1-into-ospf3
Device(config-router)# network 10.3.0.0 0.0.255.255 area 0
!
Device(config)# router bgp 1
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# address family ipv4
Device(config-router-af)# neighbor 10.100.1.1 remote-as 1
Device(config-router-af)# neighbor 10.100.1.1 update-source Loopback0
Device(config-router-af)# neighbor 10.100.1.1 next-hop-self all
Device(config-router-af)# neighbor 10.100.1.1 send-label
Device(config-router-af)# neighbor 10.100.1.5 remote-as 1
Device(config-router-af)# neighbor 10.100.1.5 update-source Loopback0
Device(config-router-af)# neighbor 10.100.1.5 route-reflector-client
Device(config-router-af)# neighbor 10.100.1.5 next-hop-self all
Device(config-router-af)# neighbor 10.100.1.5 send-label

Device(config)# ip prefix-list prefix-list-ospf1-into-ospf3 seq 5 permit 10.100.1.1/32

Device(config)# route-map ospf1-into-ospf3 permit 10
Device(config-route-map)# match ip address prefix-list prefix-list-ospf1-into-ospf3
```

シームレス MPLS の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.12.1	シームレスな MPLS	シームレス MPLS は、複数のネットワークを単一の MPLS ドメインに統合するための、非常に柔軟でスケーラブルなアーキテクチャを提供します。これは、既存の既知のプロトコルに基づいています。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。