



Cisco NCS 560 シリーズルータ（IOS XR リリース 6.6.x）セグメントルーティングコンフィギュレーションガイド

初版：2019年5月30日

シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

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

お問い合わせ先：シスコ コンタクトセンター

0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

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

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

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 NON-INFRINGEMENT 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.

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: www.cisco.com/go/trademarks. 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)

© 2019 Cisco Systems, Inc. All rights reserved.



目次

第 1 章	セグメント ルーティングについて 1
	スコープ 1
	必要性 2
	利点 3
	セグメント ルーティングを展開するためのワークフロー 3

第 2 章	セグメント ルーティング グローバル ブロックおよびセグメント ルーティング ローカル ブロック の設定 5
	セグメント ルーティング グローバル ブロックについて 5
	セグメント ルーティング ローカル ブロックについて 6
	非デフォルト セグメント ルーティング グローバル ブロック 範囲の設定 8
	非デフォルト セグメント ルーティング ローカル ブロック 範囲の設定 9

第 3 章	IS-IS プロトコル用のセグメント ルーティングの設定 11
	IS-IS プロトコル用のセグメント ルーティングの有効化 11
	IS-IS 対応ループバック インターフェイスでのプレフィックス SID の設定 13
	隣接関係 SID の設定 15
	レイヤ 2 隣接関係 SID の設定 17
	帯域幅ベースのローカル UCMP の設定 20
	IS-IS マルチドメインプレフィックス SID とドメイン ステッチング : 例 21
	IS-IS マルチドメインプレフィックス SID の設定 22
	共通ルータ ID の設定 23
	IS-IS リンクステート データの配布 24

第 4 章	OSPF プロトコル用のセグメント ルーティングの設定	25
	OSPF プロトコル用のセグメント ルーティングの有効化	25
	OSPF 対応ループバック インターフェイスでのプレフィックス SID の設定	27

第 5 章	BGP 用のセグメント ルーティングの設定	31
	BGP 用のセグメント ルーティング	31
	BGP プレフィックス セグメント識別子の設定	32
	セグメント ルーティング出力ピア エンジニアリングの設定	33
	BGP リンク ステートの設定	34
	例 : SR-EPE および BGP-LS の設定	35

第 6 章	SR-TE ポリシーの設定	39
	SR-TE ポリシーの設定	39
	自動ルート インクルード	42
	カラーのみのステアリング	43
	アドレスファミリに依存しないステアリング	44
	SRTE ポリシーを使用したスタティック ルート トラフィック ステアリング	44
	BGP SR-TE	45
	明示的 BGP SR-TE の設定	46
	CO フラグの設定	48
	バインドセグメントの使用	49

第 7 章	セグメント ルーティング パス計算要素の設定	51
	SR-PCE について	51
	SR-PCE の設定	52
	ディスジョイント ポリシーの設定 (オプション)	54
	トラフィック管理の PCE 開始 SR ポリシー	56
	セグメント ルーティング オンデマンドネクストホップ	57
	PCEP 接続の ACL サポート	60

第 8 章	トポロジに依存しないループフリー代替 (TI-LFA) の設定 61
	IS-IS 用の TI-LFA の設定 61
	OSPF 用の TI-LFA の設定 63
	TI-LFA の設定と確認：例 64

第 9 章	セグメント ルーティング マイクロループ回避の設定 69
	セグメント ルーティング マイクロループ回避について 69
	IS-IS 向けセグメント ルーティング マイクロループ回避の設定 69

第 10 章	セグメント ルーティング マッピング サーバの設定 71
	セグメント ルーティング マッピング サーバ 71
	セグメント ルーティング マッピング サーバの制限事項 72
	セグメント ルーティングと LDP の相互運用性 72
	例：セグメント ルーティング LDP の相互運用性 73
	マッピング サーバの設定 75
	マッピング アドバタイズメントの有効化 77
	IS-IS 向けマッピング アドバタイズメントの設定 77
	OSPF 向けマッピング アドバタイズメントの設定 78
	マッピング クライアントの有効化 79

第 11 章	セグメント ルーティング トラフィック マトリックスの使用 81
	セグメント ルーティング トラフィック マトリックス 81
	トラフィック コレクタ プロセス 81
	トラフィック コレクタの設定 82
	トラフィック情報の表示 83

第 12 章	セグメント ルーティング OAM の使用 87
	BGP および IGP プレフィックス SID 用の MPLS Ping および Traceroute 87
	例：プレフィックス SID の MPLS Ping、Traceroute、および ツリー トレース 88
	MPLS LSP ping および traceroute Nil FEC ターゲット 90

例 : Nil_FEC ターゲットの LSP Ping および Traceroute 91

セグメント ルーティング Ping 92

セグメント ルーティング Traceroute 94

IPv6 OAM を介したセグメント ルーティング 97



第 1 章

セグメント ルーティングについて

この章では、セグメント ルーティングの概念およびセグメント ルーティングを設定するためのワークフローについて説明します。

- [スコープ \(1 ページ\)](#)
- [必要性 \(2 ページ\)](#)
- [利点 \(3 ページ\)](#)
- [セグメント ルーティングを展開するためのワークフロー \(3 ページ\)](#)

スコープ

セグメント ルーティングは、送信元のルーティング パラダイムに基づいてネットワーク上でパケットを転送する方法です。送信元はパスを選択し、パケットヘッダーでセグメントの番号付きリストとしてエンコードします。セグメントは、任意のタイプの命令の識別子です。例えば、トポロジセグメントは、宛先へのネクスト ホップを識別します。各セグメントを識別するセグメント ID (SID) は、フラットな 20 ビットの符号なし整数からなります。

セグメント

内部ゲートウェイプロトコル (IGP) は、2つのタイプのセグメント、プレフィックスセグメントと隣接関係セグメントを配布します。各ルータ (ノード) と各リンク (隣接関係) には、関連付けられたセグメント識別子 (SID) があります。

- プレフィックス SID は、IP プレフィックスに関連付けられます。プレフィックス SID は、ラベルのセグメント ルーティング グローバル ブロック (SRGB) の範囲から手動で設定され、IS-IS または OSPF によって配布されます。プレフィックスセグメントは、その宛先への最短パスに沿ってトラフィックを誘導します。ノード SID は、特定のノードを識別する特別なタイプのプレフィックス SID です。ノードのループバックアドレスをプレフィックスとして使用して、ループバック インターフェイスの下に設定されます。

プレフィックスセグメントはグローバルセグメントであるため、プレフィックス SID はセグメント ルーティング ドメイン内でグローバルに一意です。

- 隣接関係セグメントは、隣接ルータへの出力インターフェイスなどの特定の隣接関係を表す、隣接関係 SID と呼ばれるラベルによって識別されます。隣接関係 SID は、動的ラベル

の範囲から動的に割り当てることも、ラベルのセグメントルーティング ローカル ブロック (SRLB) の範囲から手動で設定することもできます。隣接関係 SID は、IS-IS または OSPF によって配布されます。隣接関係セグメントは、トラフィックを特定の隣接関係に誘導します。

隣接関係セグメントはローカルセグメントであるため、隣接関係 SID は特定のルータに対してローカルに一意です。

番号付きリストでプレフィックス (ノード) と隣接関係セグメント ID を組み合わせることにより、ネットワーク内で任意のパスを構築できます。各ホップにおいて、先頭のセグメントがネクストホップを識別するために使用されます。セグメントはパケットヘッダーの先頭に順番にスタックされます。先頭のセグメントに別のノードの ID が含まれている場合、受信ノードは等コストマルチパス (ECMP) を使用してパケットをネクストホップに移動させます。ID が受信ノードの ID である場合、ノードは先頭のセグメントをポップし、次のセグメントに必要なタスクを実行します。

データプレーン

セグメントルーティングは、マルチプロトコルラベルスイッチング (MPLS) アーキテクチャに直接適用することができ、フォワーディングプレーンは変更されません。セグメントは MPLS ラベルとしてエンコードされます。セグメントの番号付きリストはラベルのスタックとしてエンコードされます。処理するセグメントは、スタックの一番上にあります。関連するラベルは、セグメントの完成後にスタックからポップされます。

サービス

セグメントルーティングは、レイヤ 3 VPN (L3VPN)、仮想プライベートワイヤサービス (VPWS)、仮想プライベート LAN サービス (VPLS)、イーサネット VPN (EVPN) など、MPLS の豊富なマルチサービス機能と統合されています。

トラフィック エンジニアリング用のセグメントルーティング

トラフィック エンジニアリング用のセグメントルーティング (SR-TE) は、送信元と宛先のペア間のトンネルを通じて行われます。トラフィック エンジニアリング用のセグメントルーティングでは、送信元ルーティングの概念が使用されます。送信元はパスを計算し、パケットヘッダーでセグメントとしてエンコードします。各セグメントは、送信元から宛先までのエンドツーエンドのパスであり、プロバイダー コア ネットワークのルータに、IGP によって計算された最短パスではなく指定されたパスに従うように指示します。宛先はトンネルの存在を認識しません。

必要性

トラフィック エンジニアリング用のセグメントルーティング (SR-TE) では、ネットワークはアプリケーション単位およびフロー単位の状態を維持する必要はありません。代わりに、パケットで提供されている転送指示に従うだけです。

SR-TEは、すべてのセグメントレベルでECMPを使用することにより、従来のMPLS-TEネットワークよりも効果的にネットワーク帯域幅を利用します。単一のインテリジェントソースを使用し、残りのルータをネットワーク経由で必要なパスを計算するタスクから解放します。

利点

- **SDN 対応**：セグメントルーティングはSDN向けに構築され、Application Engineered Routing (AER) の基礎となります。SRは、アプリケーションがネットワークの行動を指示できるビジネスモデル用のネットワークを準備します。SRは、分散されたインテリジェンスと集中化された最適化およびプログラミングの間の適切なバランスを提供します。
- **最小構成**：TEのセグメントルーティングでは、送信元ルータで最小構成が必要です。
- **ロードバランシング**：RSVP-TEとは異なり、セグメントルーティングのロードバランシングは、Equal Cost Multipath (ECMP; 等コストマルチパス) の存在下で実行できます。
- **Fast Reroute (FRR) をサポート**：Fast Rerouteにより、パス障害の50ミリ秒以内に事前設定されたバックアップパスの有効化が可能になります。
- **プラグアンドプレイ展開**：セグメントルーティングトンネルは、既存のMPLSコントロールプレーンおよびデータプレーンと相互運用可能で、既存の展開に実装できます。

セグメントルーティングを展開するためのワークフロー

セグメントルーティングを展開するには、次のワークフローに従います。

1. セグメントルーティンググローバルブロック (SRGB) の設定
2. IGPでのセグメントルーティングおよびノードSIDの有効化
3. SR-TEポリシーの設定



第 2 章

セグメント ルーティング グローバル ブロック および セグメント ルーティング ローカル ブロック の設定

ローカル ラベルの割り当てはラベル スイッチング データベース (LSD) によって管理されます。セグメント ルーティング グローバル ブロック (SRGB) およびセグメント ルーティング ローカル ブロック (SRLB) は LSD のセグメント ルーティング に対して保持されるラベル値です。

- [セグメント ルーティング グローバル ブロック について \(5 ページ\)](#)
- [セグメント ルーティング ローカル ブロック について \(6 ページ\)](#)
- [非デフォルト セグメント ルーティング グローバル ブロック 範囲 の設定 \(8 ページ\)](#)
- [非デフォルト セグメント ルーティング ローカル ブロック 範囲 の設定 \(9 ページ\)](#)

セグメント ルーティング グローバル ブロック について

SRGB ラベル値は SR 対応 ノード へのプレフィックス セグメント 識別子 (SID) として割り当てられ、ドメイン全体でグローバルな意味を持ちます。



(注) 範囲から割り当てられた値はドメイン全体で重要な意味を持つため、ドメイン内のすべてのルータに同じ値の範囲を設定することをお勧めします。

デフォルトの SRGB の範囲は 16000 ~ 23999 です。



(注) SR 対応ルータでは、実行中のシステムで SR が有効になっているときにデフォルトの SRGB ラベル値 (16000 ~ 23999) を使用できるように、動的ラベル範囲のデフォルトの開始値が 16000 ~ 24000 に増加します。動的ラベルの範囲が開始値 16000 で設定されている場合、実行中のシステムで SR が有効になっている場合は、デフォルトの SRGB ラベル値がすでに使用されている可能性があります。したがって、SR を有効にした後にルータをリロードして、現在割り当てられているラベルを解放し、SRGB を割り当てる必要があります。

また、SR を有効にした後で SRGB の範囲を増やす必要がある場合は、ルータをリロードして、現在割り当てられているラベルを解放し、新しい SRGB を割り当てる必要があります。

セグメントルーティングの設定を簡易に保ち、セグメントルーティングの問題のトラブルシューティングを容易にするため、ドメイン内の各ノードではデフォルトの SRGB 範囲を使用することをお勧めします。ただし、異なる範囲を定義する必要がある場合があります。次に例を示します。

- 別のベンダーのノードがデフォルトの SRGB とは異なるラベル範囲をサポートしていて、すべてのノードで同じ SRGB を使用したい場合。
- デフォルトの範囲が小さすぎる場合。
- 範囲が重複しない限り、IS-IS および OSPF プロトコルに別々の SRGB を指定する場合。

制約事項

- Cisco IOS XR リリース 6.2.x 以前では、LSD ラベル値 0 ~ 15999 が予約されています。Cisco IOS XR リリース 6.3.1 以降では、LSD ラベル値 0 ~ 14999 が予約されています。
- Cisco IOS XR リリース 6.2.x 以前では、最大 SRGB サイズは 65536 です。Cisco IOS XR リリース 6.3.1 以降では、最大 SRGB サイズは 262,143 です。
- SRGB の上限は、プラットフォームの能力を超えることはできません。



(注) 事前に予約されていないラベル値は、動的割り当てに使用できます。

SR を使用しない場合は、SRGB を無効にできます。

セグメントルーティングローカルブロックについて

セグメントルーティングローカルブロック (SRLB) は、隣接関係セグメント識別子 (adj-SID) の手動割り当てのために保存されているラベル値の範囲です。これらのラベルはローカルで重要であり、ラベルを割り当てるノードでのみ有効です。デフォルトの SRLB の範囲は 15000 ~ 15999 です。



- (注) SRLB を使用して手動で割り当てられない隣接関係 SIDは、動的ラベルの範囲から動的に割り当てられます。

セグメントルーティングの設定を簡易に保ち、セグメントルーティングの問題のトラブルシューティングを容易にするため、デフォルトの SRLB 範囲を使用することをお勧めします。ただし、異なる範囲を定義する必要がある場合があります。次に例を示します。

- 別のベンダーのノードがデフォルトの SRLB とは異なるラベル範囲をサポートしていて、すべてのノードで同じ SRLB を使用したい場合。
- デフォルトの範囲が小さすぎる場合。

新しい SRLB 範囲を定義すると、ラベルの競合が発生する可能性があります（たとえば、新しい SRLB 範囲でラベルがすでに静的または動的に割り当てられている場合など）。この場合、新しい SRLB 範囲は受け入れられますが、適用はされません（保留中）。以前の SRLB 範囲（アクティブ）は、次のいずれかを行うまで引き続き使用されます。

- ルータをリロードして、現在割り当てられているラベルを解放し、新しい SRLB を割り当てる。
- **clear segment-routing local-block discrepancy all** コマンドを使用してラベルの競合をクリアする。

制約事項

- LSD ラベル値 0 ~ 14999 は予約されています。
- SRLB のサイズは 262,143 を超えることはできません。
- SRLB の上限は、プラットフォームの能力を超えることはできません。



- (注) SRLB（セグメントルーティング ローカルブロック）の不整合および割り当て失敗エラーは、SRLB および SRGB（セグメントルーティング グローバルブロック）にデフォルト以外の値が設定され、**commit-replace** の後に設定の再適用が続く場合に発生します。SR ラベルが適切にプログラムされていないため、この問題はデータ転送に影響を及ぼします。

この問題を回避するには、**clear segment-routing local-block discrepancy all** コマンドを使用してラベルの競合をクリアします。

SR を使用しない場合は、SRLB を無効にできます。

非デフォルトセグメントルーティンググローバルブロック範囲の設定

このタスクでは、デフォルト以外の SRGB 範囲を設定する方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	[router { isis instance-id ospf process_name}] 例 : RP/0/RP0/CPU0:router (config) # router isis 1	(任意) IS-IS および OSPF プロトコル用に個別の SRGB を設定する場合は、 router isis instance-id または router ospf process_name コマンドを入力します。
ステップ 3	segment-routing global-block starting_value ending_value 例 : RP/0/RP0/CPU0:router (config-isis) # segment-routing global-block 18000 19999	SRGB 範囲に開始値として含める最小値を入力します。SRGB 範囲に終了値として含める最大値を入力します。
ステップ 4	commit	

SRGB 設定を確認します。

```
RP/0/RP0/CPU0:router# show mpls label table detail
Table Label   Owner                               State Rewrite
-----
<...snip...>
0      18000   ISIS (A):1                          InUse No
  Lbl-blk SRGB, vers:0, (start_label=18000, size=2000)
0      24000   ISIS (A):1                          InUse Yes
  (SR Adj Segment IPv4, vers:0, index=1, type=0, intf=Gi0/0/0/0, nh=10.0.0.2)
```

次のタスク

プレフィックス SID を設定し、セグメントルーティングを有効にします。

非デフォルトセグメントルーティング ローカル ブロック範囲の設定

このタスクでは、デフォルト以外の SRLB 範囲を設定する方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	segment-routing local-block <i>starting_value</i> <i>ending_value</i> 例 : RP/0/RP0/CPU0:router(config)# segment-routing lcoal-block 30000 30999	SRLB 範囲に開始値として含める最小値を入力します。SRLB 範囲に終了値として含める最大値を入力します。
ステップ 3	commit	

SRLB 設定を確認します。

```
RP/0/RP0/CPU0:router# show mpls label table detail
Table Label   Owner                               State Rewrite
-----
<...snip...>
0      13      LSD(A)                               InUse Yes
0      30000   LSD(A)                               InUse No
( Lbl-blk SRLB, vers:0, (start_label=30000, size=1000, app_notify=0)
0      30002   Static(A)                            InUse Yes
```

SRLB の不一致を表示し解決します。

```
RP/0/RP0/CPU0:router# show segment-routing local-block inconsistencies
Tue Aug 15 13:53:30.555 EDT
SRLB inconsistencies range: Start/End: 30000/30009

RP/0/RP0/CPU0:router# show mpls lsd private | i SRLB
Tue Aug 15 13:53:50.874 EDT
SRLB Lbl Mgr:
  Current Active SRLB block      = [15000, 15999]
  Configured Pending SRLB block = [30000, 30009]

RP/0/RP0/CPU0:router# clear segment-routing local-block discrepancy all
Tue Aug 15 13:59:46.897 EDT

RP/0/RP0/CPU0:router# show mpls lsd private | i SRLB
```

```
Tue Aug 15 13:59:55.370 EDT
SRLB Lbl Mgr:
  Current Active SRLB block    = [30000, 30009]
  Configured Pending SRLB block = [0, 0]

RP/0/RP0/CPU0:router# show mpls label table detail private
Tue Aug 15 14:00:26.023 EDT
Table Label  Owner                               State Rewrite
-----
0          0          LSD(A)                               InUse  Yes
0          1          LSD(A)                               InUse  Yes
0          2          LSD(A)                               InUse  Yes
0          13         LSD(A)                               InUse  Yes
0          30000       LSD(A)                               InUse  No
(Lbl-blk SRLB, vers:0, (start_label=30000, size=1000, app_notify=0)
```

次のタスク

隣接関係 SID を設定し、セグメントルーティングを有効にします。



第 3 章

IS-IS プロトコル用のセグメントルーティングの設定

Integrated Intermediate System-to-Intermediate System (IS-IS)、インターネットプロトコルバージョン 4 (IPv4) は、標準ベースの内部ゲートウェイプロトコル (IGP) です。Cisco IOS XR ソフトウェアは、国際標準化機構 (ISO) /International Engineering Consortium (IEC) 10589 および RFC 1995 に記載されている IP ルーティング機能を実装し、IP バージョン 6 (IPv6) 向けに標準拡張のシングルトポロジおよびマルチトポロジ IS-IS を追加しています。

このモジュールは、IS-IS のセグメントルーティングを有効にするために使用される設定情報を提供します。

- [IS-IS プロトコル用のセグメントルーティングの有効化 \(11 ページ\)](#)
- [IS-IS 対応ループバック インターフェイスでのプレフィックス SID の設定 \(13 ページ\)](#)
- [隣接関係 SID の設定 \(15 ページ\)](#)
- [帯域幅ベースのローカル UCMP の設定 \(20 ページ\)](#)
- [IS-IS マルチドメインプレフィックス SID とドメインステッチング: 例 \(21 ページ\)](#)

IS-IS プロトコル用のセグメントルーティングの有効化

IS-IS コントロールプレーン上のセグメントルーティングは、次をサポートしています。

- レベル 1、レベル 2、およびマルチレベルのルーティング
- ループバック インターフェイス上のホストプレフィックスのプレフィックス SID
- 隣接関係用の隣接関係 SID
- MPLS penultimate hop popping (PHP) と明示的な NULL シグナリング

ここでは、IS-IS 用のセグメントルーティングを有効にする方法について説明します。

始める前に

ルータで IS-IS のセグメントルーティングをイネーブルにする前に、ネットワークで MPLS Cisco IOS XR ソフトウェア機能をサポートする必要があります。



(注) ネットワークのトラフィック エンジニアリング部分にあるすべての IS-IS ルータ上で、次のタスク リストのコマンドを入力する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router isis instance-id 例 : RP/0/RP0/CPU0:router (config) # router isis isp	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 (注) is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実行されるルーティングのレベルを変更できます。
ステップ 3	metric-style wide [level { 1 2 }] 例 : RP/0/RP0/CPU0:router (config-isis-af) # metric-style wide level 1	レベル 1 エリアでワイドリンク メトリックのみを生成して受け入れるようにルータを設定します。
ステップ 4	segment-routing mpls 例 : RP/0/RP0/CPU0:router (config-isis-af) # segment-routing mpls	セグメント ルーティングは、次の操作で有効になります。 <ul style="list-style-type: none"> • IS-IS がアクティブなすべてのインターフェイスで MPLS 転送が有効化される。 • 転送プレーン内のすべての既知プレフィックス SID が、リモートルータによってアドバタイズされた、またはローカルまたはリモート マッピング サーバを介して学習されたプレフィックス SID を使用してプログラミングされる。 • ローカルで設定されたプレフィックス SID がアドバタイズされる。
ステップ 5	exit	

	コマンドまたはアクション	目的
	例 : RP/0/RP0/CPU0:router(config-isis-af) # exit RP/0/RP0/CPU0:router(config-isis) # exit	
ステップ 6	commit	

次のタスク

プレフィックス SID を設定します。

IS-IS対応ループバックインターフェイスでのプレフィックス SID の設定

プレフィックス SID は、IP プレフィックスに関連付けられます。プレフィックス SID は、ラベルのセグメントルーティング グローバル ブロック (SRGB) の範囲から手動で設定されます。プレフィックス セグメントは、その宛先への最短パスに沿ってトラフィックを誘導します。ノード SID は、特定のノードを識別する特別なタイプのプレフィックス SID です。ノードのループバック アドレスをプレフィックスとして使用して、ループバック インターフェイスの下に設定されます。

プレフィックス SID は、セグメント ルーティング ドメイン内でグローバルに一意です。

このタスクでは、IS-IS 対応ループバック インターフェイスでプレフィックス セグメント識別子 (SID) のインデックスまたは絶対値を設定する方法について説明します。

始める前に

セグメントルーティングが対応するアドレスファミリで有効になっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router isis instance-id 例 : RP/0/RP0/CPU0:router(config) # router isis 1	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 <ul style="list-style-type: none"> • is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実

	コマンドまたはアクション	目的
		行されるルーティングのレベルを変更できます。
ステップ 3	interface Loopback instance 例 : <pre>RP/0/RP0/CPU0:router(config-isis)# interface Loopback0</pre>	ループバック インターフェイスとインスタンスを指定します。
ステップ 4	prefix-sid { index SID-index absolute SID-value } [n-flag-clear] [explicit-null] 例 : <pre>RP/0/RP0/CPU0:router(config-isis-if-af)# prefix-sid index 1001 RP/0/RP0/CPU0:router(config-isis-if-af)# prefix-sid absolute 17001</pre>	<p>インターフェイスのプレフィックス SID インデックスまたは絶対値を設定します。</p> <p>SRGB+インデックスの下限に基づいてプレフィックス SID を作成するには、各ノードに index SID-index を指定します。</p> <p>SRGB内に特定のプレフィックス SID を作成するには、各ノードに absolute SID-value を指定します。</p> <p>デフォルトでは、n-flag がプレフィックス SID に設定され、ノード SID であることを示します。特定のプレフィックス SID (たとえば、Anycast プレフィックス SID) の場合は、n-flag-clear キーワードを入力します。IS-IS は、プレフィックス SID サブタイプ/長さ/値 (TLV) に N フラグを設定しません。</p> <p>penultimate-hop-popping (PHP) を無効にし、明示的なヌル ラベルを追加するには、explicit-null キーワードを入力します。IS-IS は、プレフィックス SID サブ TLV に E フラグを設定します。</p>
ステップ 5	commit	

プレフィックス SID 設定を確認します。

隣接関係 SID の設定

隣接関係 SID (Adj-SID) は、隣接ノードへの隣接関係に関連付けられています。隣接関係 SID は、トラフィックを特定の隣接関係に誘導します。隣接関係 SID はローカルな意味を持ち、それらを割り当てるノードでのみ有効です。

隣接関係 SID は、動的ラベルの範囲から動的に割り当てることも、ラベルのセグメントルーティング ローカルブロック (SRLB) の範囲から手動で設定することもできます。

動的に割り当てられる隣接関係 SID には特別な構成は必要ありませんが、いくつかの制限があります。

- 動的に割り当てられた Adj-SID 値は、割り当てられるまで認識されず、情報が IGP によってフラッディングされるまでコントローラは Adj-SID 値を認識しません。
- 動的に割り当てられた Adj-SID は永続的ではなく、リロードまたはプロセスの再起動後に再割り当てすることができます。
- 各リンクには一意の Adj-SID が割り当てられているため、複数のリンクで同じ Adj-SID を共有することはできません。

手動で割り当てられた Adj-SID は、リロードおよび再起動後も永続的です。同じネイバーまたは異なるネイバーへの複数の隣接関係にプロビジョニングできます。Adj-SID が保護されることを指定できます。Adj-SID がプライマリ インターフェイスで保護されていて、バックアップパスが利用可能な場合、バックアップパスがインストールされます。デフォルトでは、手動 Adj-SID は保護されていません。

隣接関係 SID は、既存の IS-IS Adj-SID サブ TLV を使用してアドバタイズされます。S フラグと P フラグは、手動で割り当てられた Adj-SID に対して定義されています。

```

0 1 2 3 4 5 6 7
+---+---+---+---+---+---+
|F|B|V|L|S|P| |
+---+---+---+---+---+---+

```

表 1: 隣接関係セグメント識別子 (Adj-SID) のフラグサブ TLV フィールド

フィールド	説明
S (セット)	このフラグは、同じ Adj-SID 値が複数のインターフェイスにプロビジョニングされている場合に設定されます。
P (永続的)	このフラグは、Adj-SID が永続的 (手動割り当て) の場合に設定されます。

手動で割り当てられた Adj-SID は、ポイントツーポイント (P2P) インターフェイスでサポートされています。

ここでは、インターフェイスに Adj-SID を設定する方法について説明します。

始める前に

セグメントルーティングが対応するアドレスファミリで有効になっていることを確認します。

show mpls label table detail コマンドを使用して、SRLB の範囲を確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router isis instance-id 例： RP/0/RP0/CPU0:router (config) # router isis 1	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 • is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実行されるルーティングのレベルを変更できます。
ステップ 3	interface type interface-path-id 例： RP/0/RP0/CPU0:router (config-isis) # interface GigabitEthernet0/0/0/7	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。
ステップ 4	point-to-point 例： RP/0/RP0/CPU0:router (config-isis-if) # point-to-point	インターフェイスがポイントツーポイント インターフェイスになるように指定します。
ステップ 5	adjacency-sid {index adj-SID-index absolute adj-SID-value } [protected] 例： RP/0/RP0/CPU0:router (config-isis-if-af) # adjacency-sid index 10 RP/0/RP0/CPU0:router (config-isis-if-af) # adjacency-sid absolute 15010	インターフェイスの Adj-SID インデックスまたは絶対値を設定します。 SRLB + インデックスの下限に基づいて Adj-SID を作成するには、各リンクに index adj-SID-index を指定します。 SRLB 内に特定の Adj-SID を作成するには、各リンクに absolute adj-SID-value を指定します。 Adj-SID が protected であるかを指定します。各プライマリ パスについて、Adj-SID がプライマリ インターフェイスで保護されていて、バックアップ パスが利用可能な場合、バックアップ パス

	コマンドまたはアクション	目的
		がインストールされます。デフォルトでは、手動 Adj-SID は保護されていません。
ステップ 6	commit	

Adj-SID 設定を確認します。

ラベルが MPLS Forwarding Information Base (LFIB) に追加されていることを確認します。

```
RP/0/RP0/CPU0:router# show mpls forwarding labels 15010
Mon Jun 12 02:50:12.172 PDT
Local   Outgoing   Prefix           Outgoing   Next Hop       Bytes
Label   Label      or ID            Interface  Next Hop       Switched
-----
15010   Pop        SRLB (idx 10)    Gi0/0/0/3  10.0.3.3       0
        Pop        SRLB (idx 10)    Gi0/0/0/7  10.1.0.5       0
        16004     SRLB (idx 10)    Gi0/0/0/7  10.1.0.5       0 (!)
        16004     SRLB (idx 10)    Gi0/0/0/3  10.0.3.3       0 (!)
```

レイヤ2隣接関係 SID の設定

通常、隣接関係 SID (Adj-SID) はネイバー ノードへのレイヤ3隣接に関連付けられ、トラフィックを特定の隣接関係に誘導します。複数の物理インターフェイスがバンドルインターフェイスを形成するレイヤ2バンドルインターフェイスを使用する場合、個々のレイヤ2バンドルメンバーは IGP には表示されません。バンドルインターフェイスのみが表示されます。

個々のレイヤ2バンドルインターフェイスにレイヤ2 Adj-SID を設定できます。この設定により、個々のバンドルメンバーリンクの可用性を追跡し、運用管理および保守 (OAM) のために個々のバンドルメンバーリンクを介してセグメントルーティング転送を確認することができます。

レイヤ2 Adj-SID は動的に割り当てることも、手動で設定することもできます。

- IGP は、レイヤ2バンドルメンバーごとに動的ラベル範囲からレイヤ2 Adj-SID を動的に割り当てます。動的レイヤ2 Adj-SID は永続的ではなく、レイヤ2バンドルリンクが稼働および停止したときに再割り当てできます。
- 手動で設定されたレイヤ2 Adj-SID は、レイヤ2バンドルリンクが稼働および停止している場合は永続的です。レイヤ2 Adj-SID は、ラベルのセグメントルーティングローカルブロック (SRLB) から割り当てられます。ただし、レイヤ2 Adj-SID の設定値が利用可能な SRLB 内に収まらない場合、レイヤ2 Adj-SID は転送情報ベース (FIB) にはプログラムされません。

[Restrictions (機能制限)]

- Adj-SID 転送にはネクストホップが必要です。これは IPv4 アドレスまたは IPv6 アドレスのいずれかで指定できますが、両方を指定することはできません。そのため、手動で設定されたレイヤ2 Adj-SID は address-family ごとに設定されます。

- 手動で設定されたレイヤ 2 Adj-SID は、1 つのレイヤ 2 バンドル メンバー リンクにのみ関連付けることができます。
- レイヤ 2 Adj-SID に使用される SID 値はレイヤ 3 Adj-SID と共有することはできません。
- レイヤ 2 Adj-SID を使用した SR-TE はサポートされていません。

ここでは、インターフェイスにレイヤ 2 Adj-SID を設定する方法について説明します。

始める前に

セグメントルーティングが対応するアドレスファミリで有効になっていることを確認します。

show mpls label table detail コマンドを使用して、SRLB の範囲を確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	segment-routing 例： RP/0/RP0/CPU0:Router(config)# segment-routing	セグメントルーティング コンフィギュレーション モードを開始します。
ステップ 3	adjacency-sid 例： RP/0/RP0/CPU0:Router(config-sr)# adjacency-sid	隣接関係 SID 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	interface type interface-path-id 例： RP/0/RP0/CPU0:Router(config-sr-adj)# interface GigabitEthernet0/0/0/3	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。
ステップ 5	address-family { ipv4 ipv6 } [unicast] 例： RP/0/RP0/CPU0:Router(config-sr-adj-intf)# address-family ipv4 unicast	IPv4 または IPv6 アドレス ファミリを指定して、ルータ 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 6	l2-adjacency sid {index adj-SID-index absolute adj-SID-value } [next-hop { ipv4_address ipv6_address }] 例： RP/0/RP0/CPU0:Router(config-sr-adj-intf-af)#	インターフェイスの Adj-SID インデックスまたは絶対値を設定します。 SRLB+インデックスの下限に基づいて Adj-SID を作成するには、各リンクに index adj-SID-index を指定します。

	コマンドまたはアクション	目的
	<pre>l2-adjacency sid absolute 15015 next-hop 10.1.1.4</pre>	<p>SRLB 内に特定の Adj-SID を作成するには、各リンクに absolute adj-SID-value を指定します。</p> <p>ポイントツーポイントインターフェイスの場合は、ネクストホップを指定する必要はありません。ただし、ネクストホップを指定した場合は、指定されたネクストホップがネイバーアドレスと一致する場合にのみ、レイヤ2 Adj-SID が使用されます。</p> <p>LAN インターフェイスの場合は、ネクストホップ IPv4 または IPv6 アドレスを設定する必要があります。ネクストホップを設定しない場合、レイヤ2 Adj-SID は LAN インターフェイスには使用されません。</p>
ステップ 7	commit	
ステップ 8	end	
ステップ 9	<p>router isis instance-id</p> <p>例 :</p> <pre>RP/0/RP0/CPU0:Router(config)# router isis isp</pre>	<p>指定したルーティングインスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータコンフィギュレーションモードにします。</p>
ステップ 10	<p>address-family { ipv4 ipv6 } [unicast]</p> <p>例 :</p> <pre>RP/0/RP0/CPU0:Router(config-isis)# address-family ipv4 unicast</pre>	<p>IPv4 または IPv6 アドレスファミリを指定して、ルータ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。</p>
ステップ 11	<p>segment-routing bundle-member-adj-sid</p> <p>例 :</p> <pre>RP/0/RP0/CPU0:Router(config-isis-af)#</pre>	<p>動的レイヤ2 Adj-SID をプログラムし、手動と動的の両方のレイヤ2 Adj-SID をアドバタイズします。</p>

	コマンドまたはアクション	目的
	<code>segment-routing bundle-member-adj-sid</code>	(注) このコマンドは、手動 L2 Adj-SID のプログラミングには必要ありませんが、動的レイヤ 2 Adj-SID のプログラミングには必要で、手動と動的の両方のレイヤ 2 Adj-SID をアドバタイズする必要があります。

設定を確認します。

```
Router# show mpls forwarding detail | i "Pop|Outgoing Interface|Physical Interface"
Tue Jun 20 06:53:51.876 PDT
. . .
15001 Pop          SRLB (idx 1)      BE1          10.1.1.4      0
    Outgoing Interface: Bundle-Ether1 (ifhandle 0x000000b0)
    Physical Interface: GigabitEthernet0/0/0/3 (ifhandle 0x000000b0)

Router# show running-config segment-routing
Tue Jun 20 07:14:25.815 PDT
segment-routing
 adjacency-sid
  interface GigabitEthernet0/0/0/3
   address-family ipv4 unicast
    12-adjacency-sid absolute 15001
  !
 !
 !
 !
```

帯域幅ベースのローカル UCMP の設定

帯域幅ベースのローカル非等コストマルチパス (UCMP) を使用すると、ローカルリンクの帯域幅に基づいて、等コストマルチパス (ECMP) のパス間で UCMP 機能をローカルで有効にできます。

帯域幅ベースのローカル UCMP は、IS-IS によってインストールされたプレフィックス、セグメントルーティング隣接関係 SID、およびセグメントルーティングラベルクロスコネクタに対して実行され、有効な帯域幅を持つ物理インターフェイスまたは仮想インターフェイスでサポートされます。

たとえば、リンクまたはラインカードのアップ/ダウンイベントのためにバンドルインターフェイスの容量が変化した場合、利用可能なプロビジョニング済みバンドルメンバーに関係なく、トラフィックは引き続き影響を受けるバンドルインターフェイスを使用します。障害により一部のバンドルメンバーが利用できなかった場合、この動作によりトラフィックでバンドルインターフェイスが過負荷状態になる可能性があります。バンドル容量の変更に対処するため

に、帯域幅ベースのローカル UCMP は、バンドル容量が変更されたときにローカルリンクの帯域幅を使用してトラフィックの負荷を分散します。

始める前に

手順

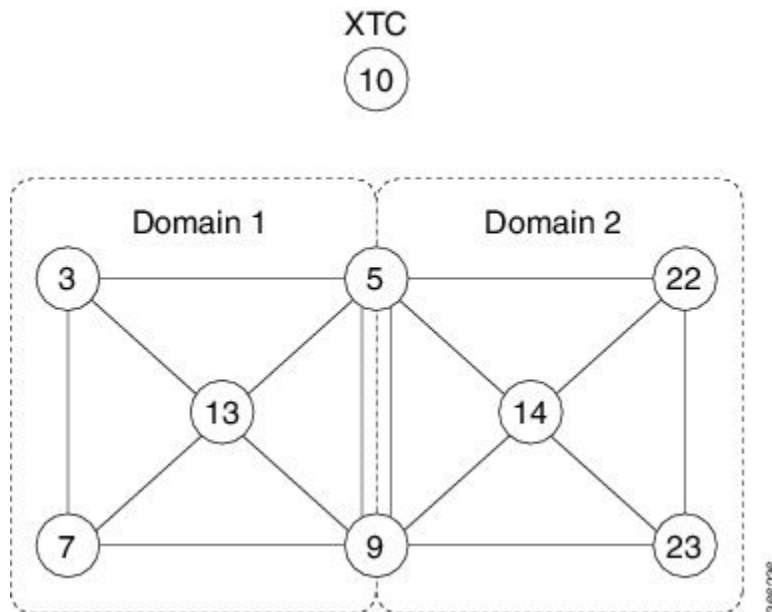
	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router isis instance-id 例 : RP/0/RP0/CPU0:router(config)# router isis 1	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実行されるルーティングのレベルを変更できます。
ステップ 3	apply-weight ecmp-only bandwidth 例 : RP/0/RP0/CPU0:router(config-isis-af)# apply-weight ecmp-only bandwidth	ローカルリンクの帯域幅に基づいて、ECMP パス間で UCMP 機能をローカルで有効にします。
ステップ 4	commit	

IS-IS マルチドメイン プレフィックス SID とドメインステッチング : 例

IS-IS マルチドメインプレフィックス SID とドメインステッチングでは、ドメイン ボーダー ノード用に同じループバック インターフェイス上に複数の IS-IS インスタンスを設定できます。複数の IS-IS インスタンスの下にループバック インターフェイスとプレフィックス SID を指定すると、プレフィックスとプレフィックス SID が異なるドメインに到達できるようになります。

この例では、次のトポロジを使用しています。ノード 5 とノード 9 は、2 つの IS-IS ドメイン (Domain1 と Domain2) の間のボーダー ノードです。ノード 10 は、IOS XR トラフィック コントローラ (XTC) として設定されています。

図 1: マルチドメイントポロジ



IS-IS マルチドメインプレフィックス SID の設定

各ボーダー ノードで複数の IS-IS インスタンスの下にループバック インターフェイスとプレフィックス SID を指定します。

Example: Border Node 5

```
router isis Domain1
 interface Loopback0
   address-family ipv4 unicast
   prefix-sid absolute 16005

router isis Domain2
 interface Loopback0
   address-family ipv4 unicast
   prefix-sid absolute 16005
```

Example: Border Node 9

```
router isis Domain1
 interface Loopback0
   address-family ipv4 unicast
   prefix-sid absolute 16009

router isis Domain2
 interface Loopback0
   address-family ipv4 unicast
   prefix-sid absolute 16009
```

ボーダー ノード 5 および 9 はそれぞれ 2 つの IS-IS インスタンス (Domain1 および Domain2) を実行し、両方のドメインで Loopback0 プレフィックスとプレフィックス SID をアドバタイズします。

両方のドメイン内のノードは、同じプレフィックスとプレフィックスSIDを使用してボーダーノードに到達できます。例えば、ノード3およびノード22は、プレフィックスSID 16005を使用してノード5に到達できます。

共通ルータ ID の設定

各ボーダーノードで、各IS-ISインスタンスの下に共通のTEルータIDを設定します。

Example: Border Node 5

```
router isis Domain1
 address-family ipv4 unicast
  router-id loopback0
```

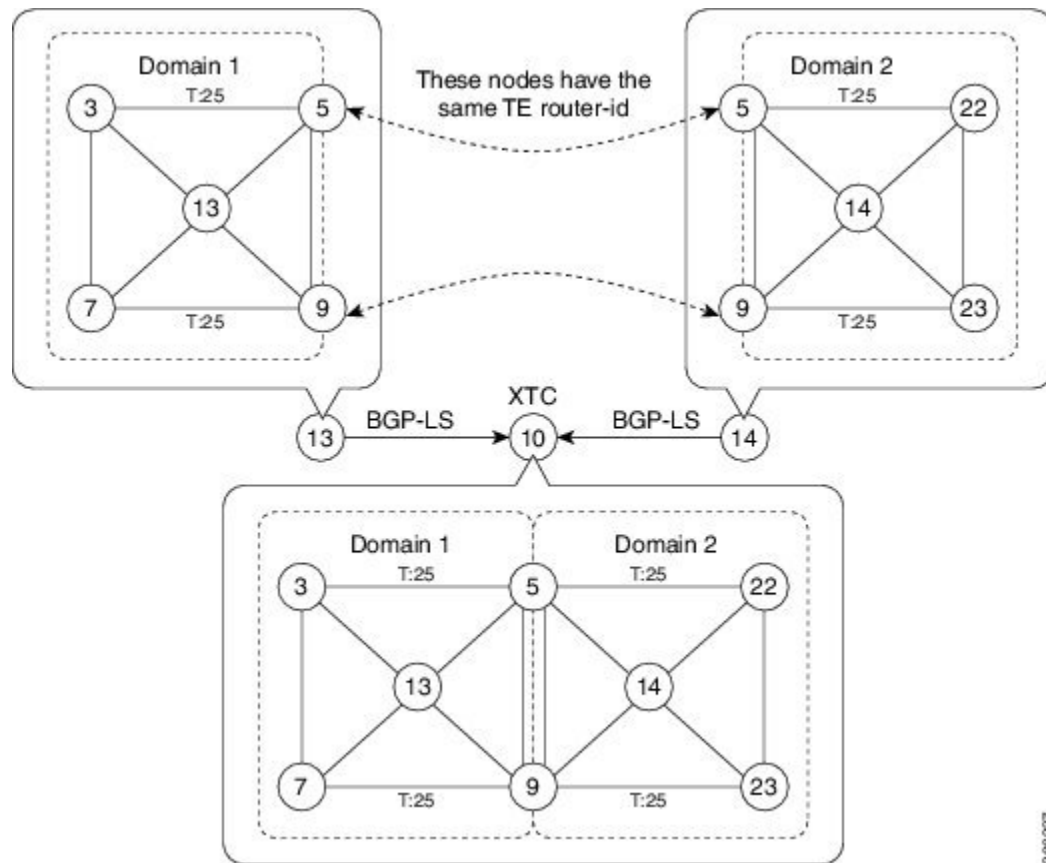
```
router isis Domain2
 address-family ipv4 unicast
  router-id loopback0
```

Example: Border Node 9

```
router isis Domain1
 address-family ipv4 unicast
  router-id loopback0
```

```
router isis Domain2
 address-family ipv4 unicast
  router-id loopback0
```

IS-IS リンクステート データの配布



ノード 13 およびノード 14 で BGP リンクステート (BGP-LS) を設定して、ローカルドメインをノード 10 に報告します。

Example: Node 13
 router isis Domain1
 distribute link-state id

Example: Node 14
 router isis Domain2
 distribute link-state id

Link-state ID は 32 から始まります。IGP ドメインごとに 1 つの ID が必要です。SR-TE TED が特定の IGP ドメインに属していることを識別するために、異なるドメイン ID が必要です。

ノード 13 とノード 14 はそれぞれ、BGP-LS のローカルドメインをノード 10 に報告します。

ノード 10 は、共通のアドバタイズされた TE ルータの ID によってボーダーノード (ノード 5 および 9) を識別してから、これらのボーダーノード上のドメインを結合 (ステッチ) してエンドツーエンドのパス計算を行います。



第 4 章

OSPF プロトコル用のセグメントルーティングの設定

Open Shortest Path First (OSPF) は、Internet Engineering Task Force (IETF) の OSPF ワーキンググループによって開発された内部ゲートウェイプロトコル (IGP) です。OSPF は特に IP ネットワーク向けに設計されており、IP サブネット化、および外部から取得したルーティング情報のタグgingをサポートしています。OSPF を使用するとパケット認証も可能になり、パケットを送受信するときに IP マルチキャストが使用されます。

このモジュールは、OSPF のセグメントルーティングを有効にするための設定情報を提供します。



(注) で OSPF を実装する場合の詳細については、『』の「*Implementing OSPF*」モジュールを参照してください。

- [OSPF プロトコル用のセグメントルーティングの有効化 \(25 ページ\)](#)
- [OSPF 対応ループバック インターフェイスでのプレフィックス SID の設定 \(27 ページ\)](#)

OSPF プロトコル用のセグメントルーティングの有効化

OSPF コントロールプレーン上のセグメントルーティングは、次をサポートしています。

- OSPFv2 のコントロールプレーン
- マルチエリア
- ループバック インターフェイス上のホストプレフィックスの IPv4 プレフィックス SID
- 隣接関係用の隣接関係 SID
- MPLS penultimate hop popping (PHP) と明示的な NULL シグナリング

ここでは、OSPF でセグメントルーティング MPLS および MPLS 転送を有効にする方法について説明します。セグメントルーティングは、インスタンス、エリア、またはインターフェイスレベルで設定できます。

始める前に

ルータで OSPF のセグメントルーティングをイネーブルにする前に、ネットワークで MPLS Cisco IOS XR ソフトウェア機能をサポートする必要があります。



- (注) ネットワークのトラフィックエンジニアリング部分にあるすべての OSPF ルータ上で、次のタスクリストのコマンドを入力する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router ospf process-name 例： RP/0/RP0/CPU0:router (config) # router ospf 1	指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーションモードでルータを配置します。
ステップ 3	segment-routing mpls 例： RP/0/RP0/CPU0:router (config-ospf) # segment-routing mpls	ルーティング プロセス上の MPLS データプレーンと、ルーティングプロセスのすべてのエリアとインターフェイスを使用して、セグメントルーティングを有効にします。 ルーティング プロセスのすべてのインターフェイスでセグメントルーティングの転送を有効にし、OSPF が受信した SID を転送テーブルにインストールします。
ステップ 4	area 0 例： RP/0/RP0/CPU0:router (config-ospf) # area 0	エリア 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	segment-routing mpls 例： RP/0/RP0/CPU0:router (config-ospf-ar) # segment-routing mpls	(任意) エリア上の MPLS データプレーンとそのエリア内のすべてのインターフェイスを使用して、セグメントルーティングを有効にします。エリアのすべてのインターフェイスでセグメントルー

	コマンドまたはアクション	目的
		ティングの転送を有効にし、OSPF が受信した SID を転送テーブルにインストールします。
ステップ 6	exit 例 : <pre>RP/0/RP0/CPU0:router(config-ospf-ar)# exit RP/0/RP0/CPU0:router(config-ospf)# exit</pre>	
ステップ 7	commit	

次のタスク

プレフィックス SID を設定します。

OSPF対応ループバックインターフェイスでのプレフィックス SID の設定

プレフィックス SID は、IP プレフィックスに関連付けられます。プレフィックス SID は、ラベルのセグメント ルーティング グローバルブロック (SRGB) の範囲から手動で設定されます。プレフィックスセグメントは、その宛先への最短パスに沿ってトラフィックを誘導します。ノードSIDは、特定のノードを識別する特別なタイプのプレフィックスSIDです。ノードのループバック アドレスをプレフィックスとして使用して、ループバック インターフェイスの下に設定されます。

プレフィックス SID は、セグメント ルーティング ドメイン内でグローバルに一意です。

このタスクでは、OSPF 対応ループバック インターフェイスでプレフィックスセグメント識別子 (SID) のインデックスまたは絶対値を設定する方法について説明します。

始める前に

インスタンス、エリア、またはインターフェイスでセグメントルーティングが有効になっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router ospf <i>process-name</i> 例 :	指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ

	コマンドまたはアクション	目的
	RP/0/RP0/CPU0:router (config) # router ospf 1	コンフィギュレーション モードでルータを配置します。
ステップ 3	area value 例 : RP/0/RP0/CPU0:router (config-ospf) # area 0	エリア 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	interface Loopback interface-instance 例 : RP/0/RP0/CPU0:router (config-ospf-ar) # interface Loopback0 passive	ループバック インターフェイスとインスタンスを指定します。
ステップ 5	prefix-sid { index SID-index absolute SID-value } [n-flag-clear] [explicit-null] 例 : RP/0/RP0/CPU0:router (config-ospf-ar) # prefix-sid index 1001 RP/0/RP0/CPU0:router (config-ospf-ar) # prefix-sid absolute 17001	<p>インターフェイスのプレフィックス SID インデックスまたは絶対値を設定します。</p> <p>SRGB+インデックスの下限に基づいてプレフィックス SID を作成するには、各ノードに index SID-index を指定します。</p> <p>SRGB内に特定のプレフィックス SID を作成するには、各ノードに absolute SID-value を指定します。</p> <p>デフォルトでは、n-flag がプレフィックス SID に設定され、ノード SID であることを示します。特定のプレフィックス SID (たとえば、Anycast プレフィックス SID) の場合は、n-flag-clear キーワードを入力します。OSPF は、プレフィックス SID サブタイプ/長さ/値 (TLV) に N フラグを設定しません。</p> <p>penultimate-hop-popping (PHP) を無効にし、明示的なヌル ラベルを追加するには、explicit-null キーワードを入力します。OSPF は、プレフィックス SID サブ TLV に E フラグを設定します。</p>
ステップ 6	commit	

プレフィックス SID 設定を確認します。

```
RP/0/RP0/CPU0:router# show ospf database opaque-area 7.0.0.1 self-originate
```

```
OSPF Router with ID (10.0.0.1) (Process ID 1)
      Type-10 Opaque Link Area Link States (Area 0)
<...>
  Extended Prefix TLV: Length: 20
    Route-type: 1
    AF          : 0
    Flags       : 0x40
    Prefix      : 10.0.0.1/32

  SID sub-TLV: Length: 8
    Flags       : 0x0
    MTID        : 0
    Algo        : 0
    SID Index : 1001
```




第 5 章

BGP 用のセグメント ルーティングの設定

ボーダー ゲートウェイ プロトコル (BGP) は、自律システム間にループフリーのドメイン間ルーティングを作成可能な外部ゲートウェイプロトコル (EGP) です。自律システムは、単一の技術管理に基づくルータのまとまりです。自律システム内のルータは、複数の内部ゲートウェイプロトコル (IGP) を使用して自律システム内のルーティング情報を交換し、EGP を使用して自律システム外でパケットをルーティングします。

このモジュールは、BGP のセグメント ルーティングを有効にするために使用される設定情報を提供します。



(注) で BGP を実装する場合の詳細については、『』の「*Implementing BGP*」モジュールを参照してください。

- [BGP 用のセグメント ルーティング \(31 ページ\)](#)
- [BGP プレフィックス セグメント 識別子の設定 \(32 ページ\)](#)
- [セグメント ルーティング 出力ピア エンジニアリングの設定 \(33 ページ\)](#)
- [BGP リンク ステートの設定 \(34 ページ\)](#)
- [例 : SR-EPE および BGP-LS の設定 \(35 ページ\)](#)

BGP 用のセグメント ルーティング

従来の BGP ベースのデータセンター (DC) ファブリックでは、パケットは自律システムの各ノードにホップバイホップで転送されます。トラフィックは、外部 BGP (eBGP) マルチパス ECMP に沿ってのみ送信されます。トラフィック エンジニアリングを行うことはできません。

MPLS ベースの DC ファブリックでは、ノード間の eBGP セッションは、BGP ラベル付きユニキャスト (BGP-LU) ネットワーク層到達可能性情報 (NLRI) を交換します。MPLS ベースの DC ファブリックを使用すると、ファブリック内の任意のリーフ (トップオブブラックまたは境界ルータ) が単一のラベルを使用して他のリーフと通信できるため、従来の BGP ベースの DC ファブリックよりもパケット転送パフォーマンスが高くなり、カプセル化のオーバーヘッドが少なくなります。ただし、各ラベル値はホップごとに異なる可能性があるため、MPLS ベースの DC ファブリックはトラブルシューティングが難しく、構成が複雑です。

BGP は、セグメントルーティングプレフィックス SID インデックスを伝送するように拡張されました。BGP-LU は、各ノードが他のリーフノードの BGP プレフィックス SID を学習するのに役立ち、送信元と宛先の間で ECMP を使用できます。BGP のセグメントルーティングによって、ファブリックの構成、操作、およびトラブルシューティングが簡素化されます。BGP のセグメントルーティングでは、BGP プレフィックス SID を使用してデータセンターでトラフィックステアリング機能を有効にできます。

BGP プレフィックス セグメント識別子の設定

BGP プレフィックスに関連付けられたセグメントは、BGP プレフィックス SID と呼ばれます。BGP プレフィックス SID は、セグメントルーティングまたは BGP ドメイン内でグローバルです。これは、BGP によって計算された ECMP 対応のベストパス上のパケットを関連するプレフィックスに転送する命令を識別します。BGP プレフィックス SID は、ラベルのセグメントルーティンググローバルブロック (SRGB) の範囲から手動で設定されます。

各 BGP スピーカーは、**segment-routing global-block** コマンドを使用して SRGB で設定する必要があります。SRGB の詳細については、「[セグメントルーティンググローバルブロックについて](#)」の項を参照してください。



(注) 範囲から割り当てられた値はドメイン全体で重要な意味を持つため、ドメイン内のすべてのルータに同じ値の範囲を設定することをお勧めします。

BGP プレフィックス SID を割り当てるには、最初に **set label-index index** 属性を使用してルーティングポリシーを作成し、次にそのインデックスをノードに関連付けます。

例

次の例に、SRGB を設定し、\$SID パラメータと **set label-index** 属性を使用して BGP ルートポリシーを作成し、プレフィックス SID インデックスをノードに関連付ける方法を示します。

```
RP/0/RP0/CPU0:router(config)# segment-routing global-block 16000 23999

RP/0/RP0/CPU0:router(config)# route-policy SID($SID)
RP/0/RP0/CPU0:router(config-rpl)# set label-index $SID
RP/0/RP0/CPU0:router(config-rpl)# end policy

RP/0/RP0/CPU0:router(config)# router bgp 1
RP/0/RP0/CPU0:router(config-bgp)# bgp router-id 1.1.1.1
RP/0/RP0/CPU0:router(config-bgp)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-bgp-af)# network 1.1.1.3/32 route-policy SID(3)
RP/0/RP0/CPU0:router(config-bgp-af)# allocate-label all
RP/0/RP0/CPU0:router(config-bgp-af)# commit
RP/0/RP0/CPU0:router(config-bgp-af)# end

RP/0/RP0/CPU0:router# show bgp 1.1.1.3/32
BGP routing table entry for 1.1.1.3/32
Versions:
  Process                bRIB/RIB    SendTblVer
```

```

Speaker                74          74
  Local Label: 16003
Last Modified: Sep 29 19:52:18.155 for 00:07:22
Paths: (1 available, best #1)
  Advertised to update-groups (with more than one peer):
    0.2
  Path #1: Received by speaker 0
  Advertised to update-groups (with more than one peer):
    0.2
  3
  99.3.21.3 from 99.3.21.3 (1.1.1.3)
    Received Label 3
    Origin IGP, metric 0, localpref 100, valid, external, best, group-best
    Received Path ID 0, Local Path ID 1, version 74
    Origin-AS validity: not-found
    Label Index: 3

```

セグメントルーティング出力ピアエンジニアリングの設定

セグメントルーティング出力ピアエンジニアリング (EPE) はコントローラを使用して、セグメントルーティングドメイン内の入力プロバイダーエッジまたはコンテンツソース (ノード) に、特定の出口プロバイダーエッジ (ノード) および特定の外部インターフェイスを使用して宛先に到達するよう指示します。BGP ピア SID は、ソースルーティングされたドメイン間パスを表すために使用されます。

コントローラは、BGP-LS EPE ルートを介して、BGP ピア SID と出力境界ルータの外部トポロジを学習します。コントローラは、BGP ラベル付きユニキャスト (BGP-LU) を使用して出口ノードとピアノードを経由して宛先にトラフィックを誘導するように入力ノードをプログラミングできます。

EPE 機能は、EPE 出力境界ルータおよび EPE コントローラでのみ必要です。

このタスクでは、EPE 出口ノードでセグメントルーティング EPE を設定する方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	router bgp <i>as-number</i> 例 : RP/0/RP0/CPU0:router(config)# router bgp 1	BGP AS 番号を指定し、BGP コンフィギュレーションモードを開始します。このモードでは、BGP ルーティングプロセスを設定できます。
ステップ 2	neighbor <i>ip-address</i> 例 :	BGP ルーティングのためにルータをネイバー コンフィギュレーションモード

	コマンドまたはアクション	目的
	RP/0/RP0/CPU0:router (config-bgp) # neighbor 192.168.1.3	にして、ネイバーの IP アドレスを BGP ピアとして設定します。
ステップ 3	remote-as as-number 例 : RP/0/RP0/CPU0:router (config-bgp-nbr) # remote-as 3	ネイバーを作成し、リモート自律システム番号を割り当てます。
ステップ 4	egress-engineering 例 : RP/0/RP0/CPU0:router (config-bgp-nbr) # egress-engineering	eBGP ピア用に EPE を使用して出力ノードを設定します。

BGP リンク ステートの設定

BGP リンクステート (LS) は、BGP を介して内部ゲートウェイ プロトコル (IGP) リンクステート データベースを伝えるために定義されたアドレスファミリー識別子 (AFI) およびサブアドレスファミリー識別子 (SAFI) です。BGPLS は、ネットワーク トポロジ情報を トポロジサーバおよびアプリケーション層トラフィック最適化 (ALTO) サーバに提供します。BGP LS では、集約、情報の非表示、および抽象化に対するポリシーベースの制御が可能です。BGP LS は、IS-IS および OSPFv2 をサポートしています。



(注) IGP は、リモートピアからの BGPLS データを使用しません。BGP は、ルータの他のコンポーネントに受信した BGPLS データをダウンロードしません。

セグメントルーティングの場合、次の属性が BGP LS に追加されています。

- ノード : セグメントルーティング機能 (SRGB 範囲を含む) およびアルゴリズム
- リンク : 隣接関係 SID と LAN 隣接関係 SID
- プレフィックス : プレフィックス SID およびセグメントルーティング マッピング サーバ (SRMS) のプレフィックス範囲

次の例は、リンクステート情報を BGP ネイバーと交換する方法を示しています。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router (config) # router bgp 1
RP/0/RP0/CPU0:router (config-bgp) # neighbor 10.0.0.2
RP/0/RP0/CPU0:router (config-bgp-nbr) # remote-as 1
```



```
RP/0/RP0/CPU0:router(config-bgp-nbr)# address-family link-state link-state
RP/0/RP0/CPU0:router(config-bgp-nbr-af)# exit
```

IGP の拡張機能

所定の BGP ノードは、複数の独立したルーティング ドメインに接続できます。BGP への IGP リンクステート配布が OSPF プロトコルと ISIS プロトコルの両方に追加され、そのノードは、これらの複数のドメインにまたがるまたはドメインを含むパスを構築するアプリケーションに同様の方法でこの情報を渡すことができます。

BGP を使用して ISIS リンクステート データを配布するには、ルータ コンフィギュレーション モードで **distribute bgp-ls** コマンドを使用します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# distribute bgp-ls instance-id 32 level 2 throttle 5
```

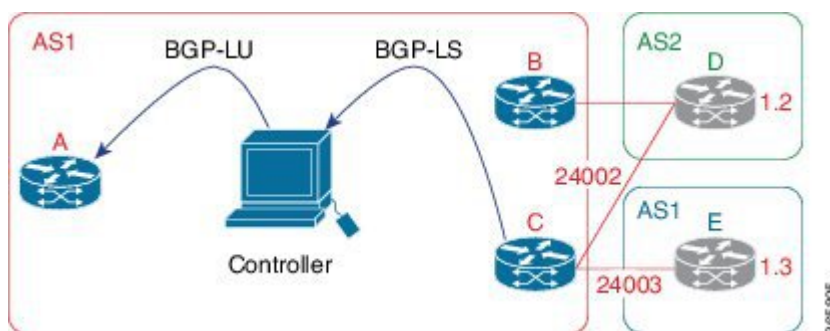
BGP を使用して OSPFv2 および OSPFv3 リンクステート データを配布するには、ルータ コンフィギュレーション モードで **distribute bgp-ls** コマンドを使用します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 100
RP/0/RP0/CPU0:router(config-ospf)# distribute bgp-ls instance-id 32 throttle 10
```

例：SR-EPE および BGP-LS の設定

次の図では、入口ノード A および出口ノード B および C を備えた自律システム AS1 でセグメントルーティングが有効になっています。この例で、出口ノード C に EPE を設定します。

図 2: トポロジ



手順

ステップ 1 eBGP ピア D および E 用に EPE を使用してノード C を設定します。

例：

```

RP/0/RP0/CPU0:router_C(config)# router bgp 1
RP/0/RP0/CPU0:router_C(config-bgp)# neighbor 192.168.1.3
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# remote-as 3
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# description to E
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# egress-engineering
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# address-family ipv4 unicast
RP/0/RP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_in in
RP/0/RP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_out out
RP/0/RP0/CPU0:router_C(config-bgp-nbr-af)# exit
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# exit
RP/0/RP0/CPU0:router_C(config-bgp)# neighbor 192.168.1.2
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# remote-as 2
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# description to D
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# egress-engineering
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# address-family ipv4 unicast
RP/0/RP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_in in
RP/0/RP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_out out
RP/0/RP0/CPU0:router_C(config-bgp-nbr-af)# exit
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# exit

```

ステップ2 BGP-LSを使用してピア ノードSIDをコントローラにアドバタイズするようにノードCを設定します。

例：

```

RP/0/RP0/CPU0:router_C(config-bgp)# neighbor 172.29.50.71
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# remote-as 1
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# description to EPE_controller
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# address-family link-state link-state
RP/0/RP0/CPU0:router_C(config-bgp-nbr)# exit
RP/0/RP0/CPU0:router_C(config-bgp)# exit

```

ステップ3 設定をコミットします。

例：

```

RP/0/RP0/CPU0:router_C(config)# commit

```

ステップ4 設定を確認します。

例：

```

RP/0/RP0/CPU0:router_C# show bgp egress-engineering

Egress Engineering Peer Set: 192.168.1.2/32 (10b87210)
  Nexthop: 192.168.1.2
  Version: 2, rn_version: 2
  Flags: 0x00000002
  Local ASN: 1
  Remote ASN: 2
  Local RID: 1.1.1.3
  Remote RID: 1.1.1.4
  First Hop: 192.168.1.2
  NHID: 3
  Label: 24002, Refcount: 3
  rpc_set: 10b9d408

Egress Engineering Peer Set: 192.168.1.3/32 (10be61d4)
  Nexthop: 192.168.1.3
  Version: 3, rn_version: 3

```

```

Flags: 0x00000002
Local ASN: 1
Remote ASN: 3
Local RID: 1.1.1.3
Remote RID: 1.1.1.5
First Hop: 192.168.1.3
NHID: 4
Label: 24003, Refcount: 3
rpc_set: 10be6250

```

出力は、ノード C が各 eBGP ピアに対してピア SID を割り当てたことを示しています。

例 :

```

RP/0/RP0/CPU0:router_C# show mpls forwarding labels 24002 24003
Local   Outgoing   Prefix      Outgoing   Next Hop    Bytes
Label   Label      or ID      Interface  Interface   Switched
-----
24002   Unlabelled No ID      Te0/0/0/1  192.168.1.2  0
24003   Unlabelled No ID      Te0/0/0/2  192.168.1.3  0

```

出力は、ノード C が転送情報ベース (FIB) にピア ノード SID をインストールしたことを示しています。



第 6 章

SR-TE ポリシーの設定

このモジュールでは、トラフィック エンジニアリング (SR-TE) ポリシーのセグメントルーティングの詳細、SR-TE ポリシーの設定方法、および SR-TE ポリシーへのトラフィックの誘導方法について説明します。

- [SR-TE ポリシーの設定 \(39 ページ\)](#)
- [BGP SR-TE \(45 ページ\)](#)
- [バインドセグメントの使用 \(49 ページ\)](#)

SR-TE ポリシーの設定

トラフィック エンジニアリングを実現するためのセグメントルーティング (SR-TE) では、ネットワークを介してトラフィックを誘導する「ポリシー」を使用します。SR-TE ポリシーパスは、セグメント ID (SID) リストと呼ばれるパスを指定するセグメントのリストとして表されます。各セグメントは、送信元から宛先までのエンドツーエンドのパスであり、ネットワークのルータに、IGP によって計算された最短パスに従うのではなく指定されたパスに従うように指示します。パケットが SR-TE ポリシーへと誘導される場合、SID リストはヘッドエンドによってパケットにプッシュされます。残りのネットワークは、SID リストに埋め込まれた命令を実行します。

SR-TE ポリシーは、順序付きリスト (ヘッドエンド、カラー、エンドポイント) として識別されます。

- ヘッドエンド : SR-TE ポリシーがインスタンス化される場所
- カラー : 同じノードペアへの2つ以上のポリシーを区別する数値 (ヘッドエンド-エンドポイント)
- エンドポイント : SR-TE ポリシーの宛先

すべての SR-TE ポリシーにはカラー値があります。同じノードペア間の各ポリシーには、一意のカラー値が必要です。

SR-TE ポリシーは、1つ以上の候補パスを使用します。候補パスは、単一セグメントリスト (SID リスト) または重み付け SID リストのセット (重み付け等コスト マルチパス (WECMP)) です。候補パスは動的または明示的のどちらかです。

動的パスは、最適化の目的と一連の制約に基づいています。ヘッドエンドはソリューションを計算し、結果としてSIDリストまたはSIDリストのセットを生成します。トポロジが変更されると、新しいパスが計算されます。ヘッドエンドにトポロジーに関する十分な情報がない場合、ヘッドエンドは計算をパス計算エンジン（PCE）に委任できます。PCEとしてXTCを設定する詳細については、「IOS XR トラフィック コントローラ（XTC）の設定」の章を参照してください。

明示的なパスは、指定されたSIDリストまたはSIDリストのセットです。

SR-TE ポリシーは、RIB/FIB 内で単一の（選択された）パスを開始します。これが優先される有効な候補パスです。

候補パスには次の特性があります。

- 優先順位があります：2つのポリシーに同じ {color, endpoint} があり、優先順位が異なる場合は、優先順位が最も高いポリシーが選択されます。
- 単一のバインド SID（BSID）に関連付けられます：同じ BSID を持つ異なる SR ポリシーがある場合、BSID 競合が発生します。この場合、最初にインストールされたポリシーが BSID を取得し、選択されます。
- 使用可能な場合に有効になります。

パスが有効で、その設定がそのポリシーのすべての候補パスの中でベストの場合にそのパスが選択されます。



(注) 送信元のプロトコルは、パス選択ロジックには関係ありません。

設定例

ローカル SR-TE ポリシーを設定するには、次の設定を完了する必要があります。

1. セグメント リストを作成します。
2. ポリシーを作成します。

ローカル SR-TE ポリシーの設定

```
/* Enter the global configuration mode and create the SR-TE segment lists */
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# segment-list name Plist-1
Router(config-sr-te-sl)# index 1 mpls label 400102
Router(config-sr-te-sl)# index 2 mpls label 400106
Router(config-sr-te-sl)# exit

Router(config-sr-te)# segment-list name Plist-2
Router(config-sr-te-sl)# index 1 mpls label 400222
Router(config-sr-te-sl)# index 2 mpls label 400106
Router(config-sr-te-sl)# exit
```

```

/* Create the SR-TE policy */
Router(config-sr-te)# policy P1
Router(config-sr-te-policy)# binding-sid mpls 15001
Router(config-sr-te-policy)# color 1 end-point ipv4 6.6.6.6
Router(config-sr-te-policy)# candidate-paths
Router(config-sr-te-policy-path)# preference 10
Router(config-sr-te-pp-index)# explicit segment-list Plist-1
Router(config-sr-te-pp-info)# weight 2
Router(config-sr-te-pp-info)# exit

Router(config-sr-te-pp-index)# explicit segment-list Plist-2
Router(config-sr-te-pp-info)# weight 2
Router(config-sr-te-pp-info)# commit
Router(config-sr-te-pp-info)# end
Router(config)#

```

実行コンフィギュレーション

```

Router# show running-configuration
segment-routing
traffic-eng
  segment-list name Plist-1
    index 1 mpls label 400102
    index 2 mpls label 400106
  !
  segment-list name Plist-2
    index 1 mpls label 400222
    index 2 mpls label 400106
  !
  policy P1
    binding-sid mpls 15001
    color 1 end-point ipv4 6.6.6.6
    candidate-paths
      preference 10
      explicit segment-list Plist-1
        weight 2
      !
      explicit segment-list Plist-2
        weight 2
    !
  !
  !
  !
  !
  !
  !

```

確認

```

Router# show segment-routing traffic-eng policy name srte_c_1_ep_6.6.6.6
Sat Jul  8 12:25:34.114 UTC
SR-TE policy database
-----
Name: P1 (Color: 1, End-point: 6.6.6.6)
Status:
  Admin: up Operational: up for 00:06:21 (since Jul  8 12:19:13.198)
Candidate-paths:
  Preference 10:
    Explicit: segment-list Plist-1 (active)

```

```

Weight: 2
  400102 [Prefix-SID, 2.1.1.1]
  400106
Explicit: segment-list Plist-2 (active)
Weight: 2
  400222 [Prefix-SID, 22.11.1.1]
  400106
Attributes:
  Binding SID: 15001
  Allocation mode: explicit
  State: programmed
  Policy selected: yes
  Forward Class: 0

```

自動ルート インクルード

自動ルート インクルードを使用して SR-TE ポリシーを設定すると、最短以外のパスを介して特定の IGP (IS-IS、OSPF) プレフィックスを誘導し、そのプレフィックスのトラフィックを SR-TE ポリシーに転送することができます。自動ルート インクルードは、指定された宛先またはプレフィックスに自動ルート アナウンス機能を適用します。

自動ルート SR-TE ポリシーはプレフィックスを IGP に追加します。これにより、エンドポイントのプレフィックスまたはエンドポイントのダウンストリームのプレフィックスが SR-TE ポリシーを使用する資格があるかどうか決定されます。プレフィックスが適格な場合、IGP はプレフィックスが自動ルート インクルード設定にリストされているかどうかを確認します。プレフィックスが含まれている場合、IGP は発信パスとして SR-TE ポリシーを使用してプレフィックス ルートをダウンロードします。

自動ルート インクルードは、次の 3 つのメトリック タイプをサポートします。

- デフォルト (メトリックなし) : SR-TE ポリシーを介したパスは最短パスメトリックを継承します。
- 絶対メトリック : ポリシー エンドポイントへの最短パスメトリックは設定された絶対メトリックに置き換えられます。自動ルートが含まれるプレフィックスへのメトリックは絶対メトリックに変更されます。
- 相対メトリック : ポリシー エンドポイントへの最短パスメトリックは設定された相対値 (プラスまたはマイナス) を使用して変更されます。



(注) IGP パス上のロードバランシングを防止するために、IGP が自動ルート設定した宛先 (**autoroute metric relative -1** など) に対して考慮する値よりも低いメトリックを指定できます。

設定例

```

Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)#policy P1

```



```

Router(config-sr-te-policy)# color 20 end ipv4 1.1.1.2
Router(config-sr-te-policy)# autoroute include ipv4 1.1.1.21/32
Router(config-sr-te-policy)# autoroute include ipv4 1.1.1.23/32
Router(config-sr-te-policy)# autoroute metric constant 1
Router(config-sr-te-policy)# candidate-paths
Router(config-sr-te-policy-path)# preference 100
Router(config-sr-te-pp-index)# explicit segment-list Plist-1

```

カラーのみのステアリング

カラーのみのステアリングは、エンドポイントに関係なく、特定のカラーでポリシーが作成されるトラフィック ステアリング メカニズムです。

NULL エンドポイント（IPv4 NULL の場合は 0.0.0.0、IPv6 NULL エンドポイントの場合は ::0）を使用する特定のカラーに SR-TE ポリシーを作成できます。つまり、その色に基づいてトラフィックを誘導できる単一のポリシーと、特定の色の拡張コミュニティを持つ宛先が異なるルート（ネクストホップ）の NULL エンドポイントを持つことができます。



- (注) NULL エンドポイントを使用したすべての SR-TE ポリシーには、明示パスオプションが必要です。ポリシーの宛先が存在しないため、ポリシーにはダイナミック パスオプション（パスがヘッドエンドまたは PCE によって計算される）を設定することはできません。

また、オーバーレイ ルートのカラー拡張コミュニティでカラーのみ (CO) フラグを指定することもできます。CO フラグを使用すると、エンドポイントのサブアドレス ファミリ識別子 (SAFI)（IPv4 または IPv6）に関係なく、一致するカラーの SR ポリシーを選択できます。[CO フラグの設定 \(48 ページ\)](#) を参照してください。

カラーのみのステアリングの設定

```

Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy P1
Router(config-sr-te-policy)# color 1 end-point ipv4 0.0.0.0

Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy P2
Router(config-sr-te-policy)# color 2 end-point ipv6 ::0

Router# show running-configuration
segment-routing
traffic-eng
policy P1
  color 1 end-point ipv4 0.0.0.0
  !
policy P2
  color 2 end-point ipv6 ::
  !

```

```
!
!
end
```

アドレスファミリーに依存しないステアリング

アドレスファミリーに依存しないステアリングでは、SR-TE ポリシーを使用して、ラベル付きとラベルなしの両方の IPv4 および IPv6 トラフィックを誘導します。この機能には、IPv4 エンドポイント ポリシーを介した IPv6 カプセル化 (IPv6 caps) のサポートが必要です。

IPv4 NULL エンドポイントの IPv6 caps は、ポリシーが XR トラフィック コントローラ (XTC) で作成されたときに自動的に有効になります。各ポリシーのバインディング SID (BSID) 状態通知には、IPv6 caps のステータス (有効または無効) を XTC クライアントに通知する「`ipv6_caps`」フラグが含まれます。

特定のカラーと IPv4 NULL エンドポイントを使用する SR-TE ポリシーは複数の候補パスを使用できます。候補パスのいずれかで IPv6 caps が有効になっている場合は、残りのすべての候補パスで IPv6 caps が有効になっている必要があります。同じカラーとエンドポイントのすべての候補パスで IPv6 caps が有効になっていない場合、トラフィックが破棄される可能性があります。

ローカル ポリシーで `ipv6 disable` コマンドを使用すると、特定のカラーと IPv4 NULL エンドポイントの IPv6 caps を無効にできます。このコマンドは、同じカラーと IPv4 NULL エンドポイントを共有するすべての候補パスで IPv6 caps を無効にします。

IPv6 カプセル化の無効化

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy P1
Router(config-sr-te-policy)# color 1 end-point ipv4 0.0.0.0
Router(config-sr-te-policy)# ipv6 disable
```

SRTE ポリシーを使用したスタティック ルート トラフィック ステアリング

以前のリリースでは、セグメントルーティング ラベル スイッチド パス (SR-LSP) をスタティック ルートに関連付けることしかできませんでした。SRTE ポリシーを使用したスタティック ルート トラフィック ステアリング機能を使用すると、MPLS および IPv6 データプレーンのスタティック ルートを設定するときに、セグメントルーティング (SR) ポリシーをインターフェイス タイプとして指定できます。

スタティック ルートの設定に関する詳細については、『*Routing Configuration Guide for Cisco NCS 540 Series Routers*』の「Implementing Static Routes」の章を参照してください。

設定例

```
Router(config)# router static
Router (config-static)# address-family ipv4 unicast

//configure administrative distance
Router (config-static-afi)# 1.1.1.1/32 sr-policy policy1 110

//Configure load metric
Router (config-static-afi)# 1.1.1.1/32 sr-policy policy1 metric 5

//Install the route in RIB regardless of reachability
Router (config-static-afi)# 1.1.1.1/32 sr-policy policy1 permanent
```

実行コンフィギュレーション

```
configure
router static
  address-family ipv4 unicast
    1.1.1.1/32 sr-policy policy1 110
    1.1.1.1/32 sr-policy policy1 metric 5
    1.1.1.1/32 sr-policy policy1 permanent
  !
!
```

BGP SR-TE

SR-TE は、データセンター（DC）のオペレータがさまざまなレベルの Service Level Assurance（SLA）を提供するために使用できます。BGP（BGP SR-TE）を使用して SR-TE パスを設定すると、この目的のために新しいプロトコルを導入することなく、DC ネットワーク操作が簡素化されます。

明示的 BGP SR-TE

明示的 BGP SR-TE は、各明示パスに対応する SID を持つ明示パスの一覧を含む SR-TE ポリシー（固有色 ID で識別される）を使用します。BGP スピーカーは明示的 SR-TE ポリシーをリモートピアに信号で伝え、特定の特性と明示パスを持つ TE トンネルの設定がトリガーされます。受信側では、明示パスに対応する TE トンネルが BGP によって設定されます。BGP 更新で言及された宛先のパケットは、ポリシーによって記述された明示パスに従います。各ポリシーは複数の明示パスを含むことができ、TE はパスごとにトンネルを作成します。



(注) ルーティングポリシーとルーティングポリシー言語（RPL）の詳細については、『*Routing Configuration Guide for Cisco NCS 540 Series Routers*』の「Implementing Routing Policy」の章を参照してください。

明示的 BGP SR-TE の設定

明示的な BGP SR-TE を設定するには、次の作業を実行します。

始める前に

ヘッドエンド ルータには、次の設定を適用する必要があります。

```
Router(config)# ipv4 unnumbered mpls traffic-eng Loopback0
Router(config)# mpls traffic-eng
Router(config-mpls-te)# auto-tunnel p2p tunnel-id min number max number
```

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	extcommunity-set opaque name 例： RP/0/RP0/CPU0:router(config)# extcommunity-set opaque color1	カラー拡張コミュニティセットを定義します。
ステップ 3	name 例： RP/0/RP0/CPU0:router(config-ext)# 1	カラー拡張コミュニティセットを定義します。
ステップ 4	end-set 例： RP/0/RP0/CPU0:router(config-ext)# end-set	拡張コミュニティセットの定義を終了します。
ステップ 5	route-policy route-policy-name 例： RP/0/RP0/CPU0:router(config)# route-policy color RP/0/RP0/CPU0:router(config-rpl)# if destination in (5.5.5.1/32) then RP/0/RP0/CPU0:router(config-rpl-if)# set extcommunity color color1 RP/0/RP0/CPU0:router(config-rpl-if)# endif RP/0/RP0/CPU0:router(config-rpl)# end-policy	ルートポリシーを作成し、ルートポリシーコンフィギュレーションモードを開始します。このモードでは、カラー拡張コミュニティ値を使用してプレフィックスをマークするルートポリシーを定義できます。

	コマンドまたはアクション	目的
ステップ 6	end-policy 例 : RP/0/RP0/CPU0:router (config-rpl) # end-policy	ルート ポリシーの定義を終了して、ルートポリシーコンフィギュレーションモードを終了します。
ステップ 7	router bgp as-number 例 : RP/0/RP0/CPU0:router (config) # router bgp 1	BGP AS 番号を指定し、BGP コンフィギュレーションモードを開始します。このモードでは、BGP ルーティングプロセスを設定できます。
ステップ 8	bgp router-id ip-address 例 : RP/0/RP0/CPU0:router (config-bgp) # bgp router-id 10.10.0.2	指定したルータ ID で、ローカルルータを設定します。
ステップ 9	address-family {ipv4 ipv6} sr-policy 例 : RP/0/RP0/CPU0:router (config-bgp) # address-family ipv4 sr-policy	IPv4 または IPv6 のいずれかのアドレスファミリを指定し、アドレスファミリのコンフィギュレーションサブモードを開始します。
ステップ 10	exit	
ステップ 11	neighbor ip-address 例 : RP/0/RP0/CPU0:router (config-bgp) # neighbor 10.10.0.1	BGP ルーティングのためにルータをネイバーコンフィギュレーションモードにして、ネイバーの IP アドレスを BGP ピアとして設定します。
ステップ 12	remote-as as-number 例 : RP/0/RP0/CPU0:router (config-bgp-nbr) # remote-as 1	ネイバーを作成し、リモート自律システム番号を割り当てます。
ステップ 13	address-family {ipv4 ipv6} unicast 例 : RP/0/RP0/CPU0:router (config-bgp-nbr) # address-family ipv4 unicast	IPv4 または IPv6 のいずれかのアドレスファミリを指定し、アドレスファミリのコンフィギュレーションサブモードを開始します。

	コマンドまたはアクション	目的
ステップ 14	route-policy route-policy-name {in out} 例： RP/0/RP0/CPU0:router (config-bgp-nbr-af) # route-policy color out	指定したポリシーをIPv4ユニキャストルートに適用します。
ステップ 15	send-extended-community-ebgp 例： RP/0/RP0/CPU0:router (config-bgp-nbr-af) # send-extended-community-ebgp	拡張コミュニティ属性を外部ボーダークラウドプロトコル (eBGP) ネイバーに送信します。

CO フラグの設定

BGP ベースのステアリング メカニズムでは、BGP のカラーおよびネクストホップと SR-TE ポリシーのそれらが照合されます。ポリシーが存在しない場合、BGP は、関連付けられたカラー、エンドポイント、および明示パスを使用して SR-TE ポリシーを作成するように XTC に要求します。カラーのみのステアリング (NULL エンドポイント) の場合、BGP のカラー拡張コミュニティの一部としてカラーのみ (CO) フラグを設定できます。



(注) カラーのみのステアリング (NULL エンドポイント) の詳細については、[カラーのみのステアリング \(43 ページ\)](#) を参照してください。

ステアリング メカニズムの動作は CO フラグの次の値に基づいています。

co-flag 00	<ol style="list-style-type: none"> 1. BGP のネクストホップおよびカラー <N, C> が SR-TE ポリシーのネクストホップとカラー <N, C> に一致します。 2. ポリシーが存在しない場合は、ネクストホップ N の IGP パスが選択されます。
-------------------	---

co-flag 01	<ol style="list-style-type: none"> 1. BGP のネクストホップおよびカラー <N, C> が SR-TE ポリシーのネクストホップとカラー <N, C> に一致します。 2. ポリシーが存在しない場合は、N と カラー C と同じアドレスファミリーを持つ NULL エンドポイントを使用する SR-TE ポリシーが選択されます。 3. N と同じアドレスファミリーを持つ NULL エンドポイントを使用するポリシーが存在しない場合は、NULL エンドポイントとカラー C を使用する SR-TE ポリシーが選択されます。 4. 一致が見つからない場合は、ネクストホップ N の IGP パスが選択されます。
-------------------	--

設定例

```

Router(config)# extcommunity-set opaque overlay-color
Router(config-ext)# 1 co-flag 01
Router(config-ext)# end-set
Router(config)#
Router(config)# route-policy color
Router(config-rpl)# if destination in (5.5.5.1/32) then
Router(config-rpl-if)# set extcommunity color overlay-color
Router(config-rpl-if)# endif
Router(config-rpl)# pass
Router(config-rpl)# end-policy
Router(config)#

```

バインドセグメントの使用

バインドセグメントは、SR-TE ポリシーを識別するローカルセグメントです。各 SR-TE ポリシーは、バインドセグメント ID (BSID) に関連付けられています。BSID は、SR-TE ポリシーがインスタンス化されるときに SR-TE ポリシーごとに自動的に割り当てられるローカルラベルです。



(注) Cisco IOS XR 6.3.2 以降のリリースでは、SR-TE ポリシーに明示的な BSID を指定できます。次の「**明示的バインド SID**」の項を参照してください。

BSID を使用して、トラフィックを SR-TE ポリシーにドメイン境界を超えて誘導し、シームレスなエンドツーエンドのドメイン間 SR-TE ポリシーを作成できます。各ドメインはローカルの SR-TE ポリシーを制御します。リモートドメインのヘッドエンドとは独立して、ローカルの SR-TE ポリシーを検証し、必要に応じて再ルーティングすることができます。バインドセグメントを使用すると、リモートドメインのトポロジの変更からヘッドエンドが分離されます。

トップラベルとして BSID で受信されたパケットは、BSID に関連付けられている SR-TE ポリシーに誘導されます。BSID ラベルがポップされると、SR-TE ポリシーの SID リストがプッシュされます。

BSID は次の場合に使用できます。

- マルチドメイン（ドメイン間、自律システム間）：BSID を使用して、ドメイン境界を越えてトラフィックを誘導し、シームレスなエンドツーエンドのドメイン間 SR-TE ポリシーを作成できます。
- 単一ドメイン内の大規模：ヘッドエンドは、SR-TE ポリシーの別のレイヤ内でエンドツーエンド（エッジツーエッジ）の SR-TE ポリシーをネストすることにより、階層型 SR-TE ポリシーを使用できます（アグリゲーションからアグリゲーションまで）。SR-TE ポリシーは、BSID を使用する別のポリシーのレイヤ内にネストされ、シームレスなエンドツーエンドの SR-TE ポリシーが作成されます。
- ラベルスタック圧縮：SR-TE ポリシーに必要なラベルスタックのサイズがプラットフォーム機能を超えている場合、SR-TE ポリシーは、バインドセグメントを使用して他の SR-TE ポリシーにシームレスにステッチしたり、ネストすることができます。

明示的なバインド SID

SR-TE ポリシーに指定した BSID 値を使用するように要求するには、**binding-sid explicit {fallback-dynamic | enforce-srlb}** コマンドを使用します。明示的な BSID は、セグメントルーティングローカルブロック（SRLB）またはラベルのダイナミックレンジから割り当てられます。

SR-TE ポリシーへのこの BSID の要求と取得はベストエフォートで行われます。要求された BSID が利用できない場合（利用可能な SRLB に属していない、または別のアプリケーションまたは SR-TE ポリシーによってすでに使用されている場合）、ポリシーはダウン状態のままです。

BSID 値が利用できない場合は、BSID 割り当ての動作を指定することができます。

- 動的割り当てへのフォールバック：BSID が利用できない場合、BSID は動的に割り当てられ、ポリシーが起動します。

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# binding-sid explicit fallback-dynamic
```

- 厳格な SRLB 適用：BSID が SRLB 内にはない場合、ポリシーはダウン状態のままです。

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# binding-sid explicit enforce-srlb
```




第 7 章

セグメント ルーティング パス計算要素の 設定

セグメント ルーティング パス計算要素 (SR-PCE) は、既存の IOS-XR PCEP 機能に追加機能を拡張してステートフル PCE 機能を提供します。SR-PCE は、MPLS データ プレーンと IPv4 コントロール プレーンでサポートされています。



(注) SR-PCE をインストールするには、Cisco IOS XRv 9000 ルータのインスタンスをインストールする必要があります。詳細については、『[Cisco IOS XRv 9000 Router Installation and Configuration Guide](#)』を参照してください。

- [SR-PCE について \(51 ページ\)](#)
- [SR-PCE の設定 \(52 ページ\)](#)
- [トラフィック管理の PCE 開始 SR ポリシー \(56 ページ\)](#)
- [セグメント ルーティング オンデマンド ネクストホップ \(57 ページ\)](#)
- [PCEP 接続の ACL サポート \(60 ページ\)](#)

SR-PCE について

パス計算要素プロトコル (PCEP) は、パス計算クライアント (PCC) が PCC を起点とするヘッドエンドラベルスイッチドパス (LSP) の制御を PCE ピアに報告し委任できる一連の手順を記述しています。PCE は、PCC が制御している LSP のパラメータの更新と変更を PCC に要求することができます。また、ステートフルモデルでは、PCC は PCE が計算を開始することを許可でき、PCE はネットワーク全体のオーケストレーションを実行できます。

SR-PCE は、IGP (OSPF または IS-IS) または BGP リンクステート (BGP-LS) 経由でトポロジ情報を学習します。

SR-PCE は、以下の方法を使用してパスを計算できます。

- **TE メトリック** : SR-PCE は TE メトリックを使用してパス計算を行い、累積 TE メトリックを最適化します。

- IGP メトリック：SR-PCE は IGP メトリックを使用してパス計算を行い、到達可能性を最適化します。
- LSP ディスジョイントネス：SR-PCE はパス計算アルゴリズムを使用して、ディスジョイント LSP のペアを計算します。ディスジョイントパスの起点は、同じヘッドエンドまたは異なるヘッドエンドです。ディスジョイントレベルとは、2つの計算されたパスで共有すべきではないリソースのタイプを指します。SR-PCE は、次のディスジョイントパス計算をサポートしています。
 - リンク：リンクが計算されたパスで共有されないことを指定します。
 - ノード：ノードが計算されたパス上で共有されないことを指定します。
 - SRLG：同じ SRLG 値を持つリンクが計算されたパスで共有されないことを指定します。
 - SRLG ノード：SRLG とノードが計算されたパス上で共有されないことを指定します。

所定のディスジョイントグループ ID で最初の要求が受信されると、最初の LSP が計算され、最初の送信元から最初の宛先への最短パスがエンコードされます。2つ目の LSP 要求が同じディスジョイントグループ ID で受信されると、両方の要求で受信された情報を使用して2つのディスジョイントパス（1つは最初の送信元から最初の宛先へのパス、もう1つは2つ目の送信元から2つ目の宛先へのパス）が計算されます。両方のパスが同時に計算されます。

SR-PCE の設定

このタスクでは、SR-PCE を設定する方法について説明します。

始める前に

必要に応じて、Cisco IOS XRv 9000 ルータのインスタンスをインストールして設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	pce 例： RP/0/RP0/CPU0:router(config)# pce	PCE を有効にし、PCE コンフィギュレーション モードを開始します。
ステップ 3	address ipv4 address 例：	PCE IPv4 アドレスを設定します。

	コマンドまたはアクション	目的
	RP/0/RP0/CPU0:router(config-pce)# address ipv4 192.168.0.1	
ステップ 4	state-sync ipv4 address 例 : RP/0/RP0/CPU0:router(config-pce)# state-sync ipv4 192.168.0.3	リモートピアに状態同期を設定します。
ステップ 5	tcp-buffer size 例 : RP/0/RP0/CPU0:router(config-pce)# tcp-buffer 1024000	各 PCEP セッションの送受信 TCP バッファ サイズをバイト単位で設定します。デフォルトのバッファサイズは 256000 です。有効な範囲は 204800 ~ 1024000 です。
ステップ 6	password {clear encrypted} password 例 : RP/0/RP0/CPU0:router(config-pce)# password encrypted pwd1	すべての PCEP ピアの TCP 認証を有効にします。設定されたパスワードと一致する MAC を含まない PCC から来る TCP セグメントはすべて拒否されます。パスワードが暗号化されているか、またはクリアテキストであるかを指定します。
ステップ 7	segment-routing {strict-sid-only te-latency} 例 : RP/0/RP0/CPU0:router(config-pce)# segment-routing strict-sid-only	厳格な SID または TE のレイテンシを使用するようにセグメントルーティング アルゴリズムを設定します。 (注) この設定はグローバルで、このコントローラからパスを要求するすべての LSP に適用されます。
ステップ 8	timers 例 : RP/0/RP0/CPU0:router(config-pce)# timers	タイマー コンフィギュレーション モードを開始します。
ステップ 9	keepalive time 例 : RP/0/RP0/CPU0:router(config-pce-timers)# keepalive 60	ローカルで生成されたキープアライブ メッセージのタイマー値を設定します。デフォルトの時間は 30 秒です。

	コマンドまたはアクション	目的
ステップ 10	minimum-peer-keepalive <i>time</i> 例： RP/0/RP0/CPU0:router (config-pce-timers)# minimum-peer-keepalive 30	セッション確立中にリモートピアが PCEOPEN メッセージで提案できる最小の許容キープアライブタイマーを設定します。デフォルトの時間は 20 秒です。
ステップ 11	reoptimization <i>time</i> 例： RP/0/RP0/CPU0:router (config-pce-timers)# reoptimization 30	再最適化タイマーを設定します。デフォルトタイマーは 60 秒です。
ステップ 12	exit 例： RP/0/RP0/CPU0:router (config-pce-timers)# exit	タイマーコンフィギュレーションモードを終了し、PCE コンフィギュレーションモードに戻ります。

ディスジョイントポリシーの設定（オプション）

次のタスクでは、PCEP 要求に PCEP 関連グループ ID オブジェクトを含まない PCC によってシグナリングされた LSP のペアのディスジョイントネスを計算するように SR-PCE を設定する方法について説明します。これは、PCC がこの PCEP オブジェクトをサポートしていない場合、またはネットワークオペレータが LSP ディスジョイント設定を一元管理する場合の展開に便利です。

手順

	コマンドまたはアクション	目的
ステップ 1	disjoint-path 例： RP/0/RP0/CPU0:router (config-pce)# disjoint-path	ディスジョイン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 2	group-id <i>value</i> type { link node srlg srlg-node } [sub-id <i>value</i>] 例： RP/0/RP0/CPU0:router (config-pce-disjoint)# group-id 1 type node sub-id 1	ディスジョイントグループ ID を設定し、ディスジョイントネスの優先レベル（2つのパスで共有されるべきでないリソースのタイプ）を定義します。 <ul style="list-style-type: none"> • link：リンクが計算されたパス上で共有されないことを指定します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • node : ノードが計算されたパス上で共有されないことを指定します。 • srlg : 同じ SRLG 値を持つリンクが計算されたパスで共有されないことを指定します。 • srlg-node : SRLG とノードが計算されたパス上で共有されないことを指定します。 <p>要求されたディスジョイントネス レベルを満たすパスのペアが見つからない場合、パスは自動的に下位レベルにフォールバックされます。</p> <ul style="list-style-type: none"> • 要求されたディスジョイントネス レベルが SRLG またはノードの場合、リンクディスジョイントパスが計算されます。 • 要求されたディスジョイントネス レベルがリンクの場合、または SRLG またはノードのディスジョイントネスからの最初のフォールバックが失敗した場合は、2つの最短パスをエンコードするセグメントのリストが、ディスジョイントネスの制約なしで計算されます。
ステップ 3	strict 例 : <pre>RP/0/RP0/CPU0:router (config-pce-disjoint)# strict</pre>	(オプション) 優先レベルのディスジョイントネスの自動フォールバック動作を防止します。要求されたディスジョイントネス レベルを満たすパスのペアが見つからない場合、ディスジョイントの計算は終了し、新しいパスは提供されません。既存のパスは変更されません。
ステップ 4	lsp {1 2} pcc ipv4 address lsp-name lsp_name [shortest-path] 例 : <pre>RP/0/RP0/CPU0:router (config-pce-disjoint)# lsp 1 pcc ipv4 192.168.0.1 lsp-name rtrA_t1 shortest-path RP/0/RP0/CPU0:router (config-pce-disjoint)# lsp 2 pcc ipv4 192.168.0.5 lsp-name</pre>	ディスジョイント グループに LSP を追加します。 shortest-path キーワードは、ディスジョイントパスの1つに、送信元から宛先までの最短パスに従うよう強制します。このオプションは、指定された最初の LSP にのみ適用できます。

	コマンドまたはアクション	目的
	<code>rtrE_t2</code>	

トラフィック管理の PCE 開始 SR ポリシー

SR-TE ポリシーは、リンクの輻輳を軽減したり、ネットワーク タッチ ポイントの数を最小限に抑えたりするようにパス計算要素 (PCE) で設定することができます。



- (注) PCE 開始 SR-TE ポリシーは PCE コンフィギュレーション モードで入力されます。SR-TE ポリシーの設定に関する詳細については、[SR-TE ポリシーの設定 \(39 ページ\)](#) を参照してください。

PCE は、トラフィック需要やリンク使用率などのネットワーク情報を収集します。PCE はリンクが輻輳していると判断すると、輻輳の原因となっている 1 つ以上のフローを特定します。PCE は適切なパスを見つけ、ネットワークの別の部分に輻輳を移動せずに、そのフローを転送するように SR-TE ポリシーを展開します。リンクの輻輳がない場合、ポリシーは削除されます。

ネットワーク タッチ ポイントの数を最小限に抑えるために、ネットワーク サービス オーケストレータ (NSO) などのアプリケーションは PCE に SR-TE ポリシーを作成するように要求できます。PCE は、PCC-PCE 通信プロトコル (PCEP) を使用して SR-TE ポリシーを展開します。

1. PCE は PCInitiate メッセージを PCC に送信します。
2. PCInitiate メッセージが有効な場合、PCC は PCRpt メッセージを送信します。それ以外の場合は、PCErr メッセージが送信されます。
3. PCInitiate メッセージが受け入れられると、PCE は PCUpd メッセージを送信して SR-TE ポリシーを更新します。

SR-TE ポリシーを使用して複数の PCE を設定すると、ハイアベイラビリティを実現できます。ヘッドエンド (PCC) が 1 つの PCE との接続を失った場合、別の PCE が SR-TE ポリシーの制御を引き継ぐことになります。

設定例

PCE 開始 SR-TE ポリシーを設定するには、次の設定を完了する必要があります。

1. PCE コンフィギュレーション モードを開始します。
2. セグメント リストを作成します。
3. ポリシーを作成します。

```
/* Enter PCE configuration mode and create the SR-TE segment lists */
Router# configure
Router(config)# pce

/* Create the SR-TE segment lists */
Router(config-pce)# segment-routing
Router(config-pce-sr)# traffic-eng
Router(config-pce-sr-te)# segment-list name addr2a
Router(config-pce-sr-te-sl)# index 1 address ipv4 14.14.14.4
Router(config-pce-sr-te-sl)# exit

/* Create the SR-TE policy */
Router(config-pce-sr-te)# peer ipv4 1.1.1.1
Router(config-pce-sr-te)# policy P1
Router(config-pce-sr-te-policy)# color 2 end-point ipv4 2.2.2.2
Router(config-pce-sr-te-policy)# candidate-paths
Router(config-pce-sr-te-policy-path)# preference 50
Router(config-pce-sr-te-pp-index)# explicit segment-list addr2a
Router(config-pce-sr-te-pp-info)# end
Router(config)#
```

ランニング コンフィギュレーション

```
pce
segment-routing
  traffic-eng
    segment-list name addr2a
      index 1 address ipv4 14.14.14.4
    !
  peer ipv4 1.1.1.1
  policy P1
    color 2 end-point ipv4 2.2.2.2
    candidate-paths
      preference 50
      explicit segment-list addr2a
    !
  !
```

セグメントルーティングオンデマンドネクストホップ

大規模な集約ネットワークは数千のリーフノードを使用することができます。ただし、特定のリーフノードでは、他のいくつかのリーフノードのサービスのみを使用できます。そのため、リーフノード間にサービスが存在しない限り、2つのリーフノード間のパスを計算して保存することは現実的ではありません。オンデマンドのネクストホップ (ODN) 機能により、オペレータが介入せずに、必要に応じてリーフノードがオンデマンドでリーフノード間のパスを自動的に計算できるようになります。ODN は、運用の簡素化と効率的な大規模ネットワーク集約につながります。ODN 機能を使用すると、リーフノード (分散) によってリーフ内ドメインパスが計算されます。一方、リーフ間ドメインパスは PCE によって計算されます。これは、PCE または WAE/PCE の組み合わせ (集中型) になります。

ODN 機能は MPLS セグメントルーティング (SR) ポリシーに基づいています。SR ポリシーはそのカラー (C) およびエンドポイント (EP) によって定義されます。オンデマンド SR ポ

リシーは、IPv4、IPv6、L3VPNv4/L3VPNv6、6PE、EVPNなどのBGPベースのサービスを伝送するために、BGPによって動的に作成されます。SRポリシーはバインドセグメントID (BSID)に関連付けられます。

SRポリシーでは、BGPのようなアプリケーションによるトラフィックステアリングが許可されています。トラフィックステアリングが許可されたSRポリシーは承認済みポリシーと呼ばれます。トラフィックステアリングの目的で定義された一連のカラーは、許可されたカラーと呼ばれます。

適切なODN操作を行うには、BGPなどのアプリケーションが承認済みポリシーと許可されたカラーの両方を認識している必要があります。また、サービスの伝送に最適な他のポリシーがない場合は、アプリケーションがSRポリシーを動的に作成できるようにする必要があります。

リーフノードがパスの計算に対応していない場合は、既知のステートフルPCEを使用してパスを取得します。パスはPCEに委任されます。

設定例

ODNを設定するには、次の設定を完了する必要があります。

1. ステアリングおよびオンデマンドポリシーのカラーを許可します。
2. PCEの設定
3. PCCの設定

ステアリングおよびオンデマンドポリシーのカラーの許可

```
Router(config)# segment-routing
Router(config-sr)# traffic-eng

/*Enable authorized on-demand color, with dynamic path computation using PCE*/
Router(config-sr-te)# on-demand color 4
Router(config-sr-te-color)# dynamic pce
Router(config-sr-te-color-dyn-mpls-pce)# exit

/*Configure metric for use in path computation*/
Router(config-sr-te-color-dyn)# metric type igp

/*Configure disjoint-path type*/
Router(config-sr-te-color-dyn)# disjoint-path group-id 111 type srlg-node sub-id 222
Router(config-sr-te-color-dyn)# exit
Router(config-sr-te-color)# exit
Router(config-sr-te)# exit
Router(config-sr)#

/*Disable authorization on manual policy*/
Router(config-sr-te)# policy bgp_AP_1
Router(config-sr-te-policy)# steering bgp disable
Router(config-sr-te-policy)# exit
Router(config-sr-te)# exit
Router(config-sr)# exit
Router(config)#
```


PCE の設定

```
/*Configure PCE IPv4 address*/
Router(config)# pce
Router(config-pce)# address ipv4 <address>
```

PCC の設定

PCE への接続を確立するように PCC を設定します。PCC と PCE 間で TCP 接続（PCEP メッセージの交換用）を確立できるように、PCC アドレスと PCE アドレスをルーティング可能にする必要があります。

PCE には任意の優先順位を付与することができます。PCC が複数の PCE に接続されている場合、PCC は最も低い優先順位値の PCE を選択します。タイがある場合は、最高位の IP アドレスの PCE がコンピューティングパス用に選択されます。

```
/*Configure the PCC to establish a connection to the PCE*/
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# pcc
Router(config-sr-te-pcc)# source-address ipv4 <local-source-address>
Router(config-sr-te-pcc)# pce address ipv4 <PCE-address>[precedence <value>]

/* Specify how long a PCC-delegated policy stays up after connections to PCE servers are
lost*/
Router(config-sr-te-pcc)# timers delegation-timeout 0

/*Specify how often keepalive messages are sent from PCC to its peers*/
Router(config-sr-te-pcc)# timers keepalive 20

/*Specify how long the remote peers wait before bringing down the PCEP session if no
PCEP messages are received from this PCC*/
Router(config-sr-te-pcc)# timers deadtimer 80
```

検証

```
Router# show segment-routing traffic-eng policy
Tue Aug 15 15:18:38.403 EDT

SR-TE policy database
-----

Name: bgp_AP_1 (Color: 4, End-point: 192.168.0.2)
Status:
  Admin: up Operational: up for 00:04:19 (since Aug 15 15:14:19.555)
Candidate-paths:
  Preference 100:
    Dynamic (pce 192.168.1.5) (active)
      Weight: 0
      24002 [Adjacency-SID, 11.11.11.1 - 11.11.11.2]
Attributes:
  Binding SID: 24011
  Allocation mode: dynamic
  State: programmed
  Policy selected: yes
  Forward Class: 0
```

```
Distinguisher: 0
Auto-policy info:
Creator: BGP
IPv6 caps enable: no
```

PCEP 接続の ACL サポート

PCE プロトコル (PCEP) (RFC5440) は TCP/IP を介して実行されているクライアントサーバモデルであり、サーバ (PCE) がポートをオープンし、クライアント (PCC) が接続を開始します。ピアが TCP 接続を確立すると、その接続上で PCE セッションを作成します。

PCEP 接続の ACL サポート機能は、アクセスコントロールリスト (ACL) を使用して PCE サーバを保護し、クライアントの送信元アドレスに基づいて TCP 接続が作成されたときに IPv4 PCC ピアを制限する方法を提供します。クライアントが TCP 接続を開始すると、ACL が参照され、クライアントの送信元アドレスが比較されます。ACL はアドレスを許可または拒否し、TCP 接続を続行するかどうかを選択できます。

ACL 設定情報については、『*IP Addresses and Services Configuration Guide for Cisco NCS 540 Series Routers*』の「Understanding Access Lists」の章を参照してください。

ACL を PCE に適用するには、`pce peer-filter ipv4 access-list acl_name` コマンドを使用します。



第 8 章

トポロジに依存しないループフリー代替 (TI-LFA) の設定

トポロジに依存しないループフリー代替 (TI-LFA) は、セグメントルーティングを使用して、他の高速再ルーティング技術が保護を提供できないトポロジでリンク保護を提供します。TI-LFA の目的は、リンク障害によるトポロジ変更後にルータがコンバージェンスする間に結果として生じるパケット損失を減らすことです。急速な障害修復 (50 ミリ秒未満) は、分散ネットワーク コンバージェンス プロセスが完了するまで、ループフリーで安全に使用できる事前計算済みのバックアップ パスを使用することによって達成されます。



(注) TI-LFA は IPv4 のみをサポートします。

TI-LFA はリンク保護を提供します。リンクはコンバージェンス後のバックアップ パスの計算中に除外されます。

- [IS-IS 用の TI-LFA の設定 \(61 ページ\)](#)
- [OSPF 用の TI-LFA の設定 \(63 ページ\)](#)
- [TI-LFA の設定と確認：例 \(64 ページ\)](#)

IS-IS 用の TI-LFA の設定

このタスクでは、リンクの障害に関するトラフィックフローを収束させるために、プレフィックスごとのトポロジに依存しないループフリー代替 (TI-LFA) の計算を有効にする方法について説明します。

始める前に

次のトポロジ要件を満たしていることを確認してください。

- ルータ インターフェイスがトポロジごとに設定されている。
- ルータが IS-IS で設定されている。

- セグメント ルーティング LSP が設定されている。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router isis instance-id 例 : RP/0/RP0/CPU0:router(config)# router isis 1	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 (注) is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実行されるルーティングのレベルを変更できます。
ステップ 3	interface type interface-path-id 例 : RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet0/0/0/1	インターフェイス設定モードを開始します。
ステップ 4	address-family ipv4 [unicast] 例 : RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast	IPv4 アドレス ファミ리를指定し、ルータ 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 5	fast-reroute per-prefix 例 : RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix	プレフィックスごとの Fast Reroute を有効にします。
ステップ 6	fast-reroute per-prefix ti-lfa 例 : RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix ti-lfa	プレフィックスごとの TI LFA Fast Reroute リンク保護を有効にします。

TI-LFA がセグメント ルーティング用に正常に設定されました。

OSPF 用の TI-LFA の設定

このタスクでは、リンクの障害に関するトラフィックフローを収束させるために、プレフィックスごとのトポロジに依存しないループフリー代替 (TI-LFA) の計算を有効にする方法について説明します。



- (注) TI-LFA は、インスタンス、エリア、またはインターフェイスで設定できます。インスタンスまたはエリアに設定すると、インスタンスまたはエリア内のすべてのインターフェイスが設定を継承します。

始める前に

次のトポロジ要件を満たしていることを確認してください。

- ルータ インターフェイスがトポロジごとに設定されている。
- ルータが OSPF で設定されている。
- セグメント ルーティング LSP が設定されている。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router ospf process-name 例 : RP/0/RP0/CPU0:router(config)# router ospf 1	指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。
ステップ 3	area area-id 例 : RP/0/RP0/CPU0:router(config-ospf)# area 1	エリア 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	interface type interface-path-id 例 : RP/0/RP0/CPU0:router(config-ospf-ar)# interface GigabitEthernet0/0/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	fast-reroute per-prefix 例 :	プレフィックスごとの Fast Reroute を有効にします。

	コマンドまたはアクション	目的
	RP/0/RP0/CPU0:router (config-ospf-ar-if) # fast-reroute per-prefix	
ステップ 6	fast-reroute per-prefix ti-lfa 例 : RP/0/RP0/CPU0:router (config-ospf-ar-if) # fast-reroute per-prefix ti-lfa	プレフィックスごとの TI LFA Fast Reroute リンク保護を有効にします。

TI-LFA がセグメント ルーティング用に正常に設定されました。

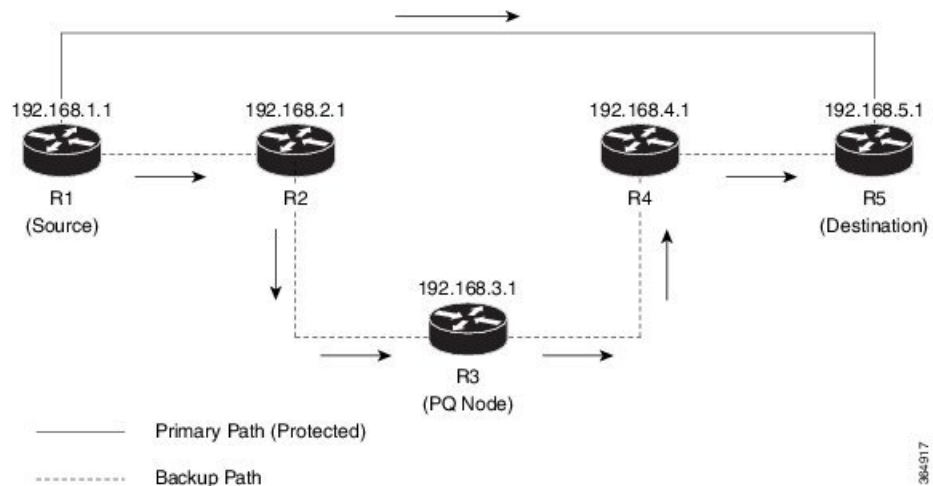
TI-LFA の設定と確認 : 例

この例では、単一またはディスジョイントの PQ ノードを使用してセグメントルーティング TE トンネルに TI-LFA を設定します。

次の図は、この例で使用されている 2 つのトポロジを示しています。

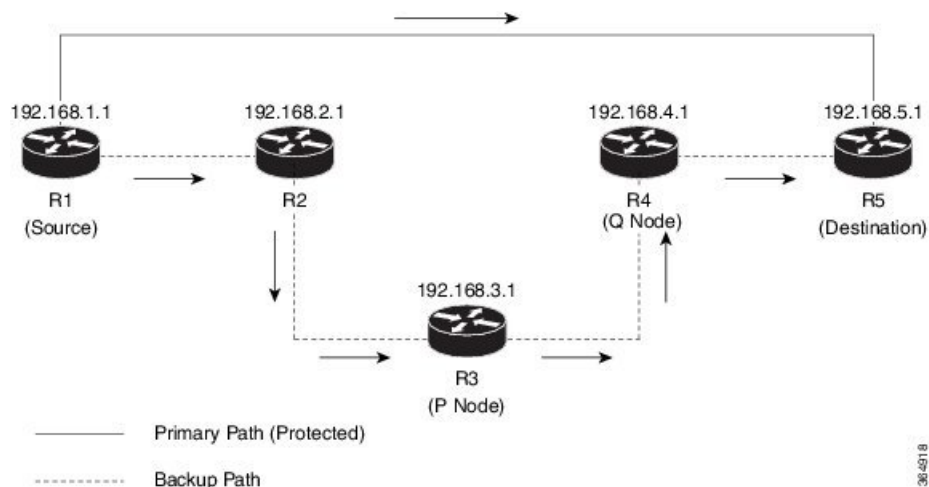
- トポロジ 1 は単一の PQ ノードを使用するため、2 つの SID を持ちます。送信元ルータ R1 から PQ ノードを経由して宛先ルータ R5 に送信されます。

図 3: トポロジ 1: 単一の PQ ノード



- トポロジ 2 はディスジョイント PQ ノードを使用するため、3 つの SID で構成されます。送信元ルータ R1 から P ノードおよび Q ノードを介して宛先ルータ R5 に送信されます。

図 4: トポロジ 2: ディスジョイント PQ ノード



手順

ステップ 1 宛先ルータ (R5) に接続する送信元ルータ (R1) インターフェイスで IS-IS または OSPF 用に TI-LFA を設定します。

• IS-IS の場合

```
RP/0/RP0/CPU0:R1 (config)# router isis 1
RP/0/RP0/CPU0:R1 (config-isis)# interface TenGigE0/0/0/2
RP/0/RP0/CPU0:R1 (config-isis-if)# point-to-point
RP/0/RP0/CPU0:R1 (config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:R1 (config-isis-if)# fast-reroute per-prefix
RP/0/RP0/CPU0:R1 (config-isis-if)# fast-reroute per-prefix ti-lfa
RP/0/RP0/CPU0:R1 (config-isis-if)# exit
```

• OSPF の場合

```
RP/0/RP0/CPU0:R1 (config)# router ospf 1
RP/0/RP0/CPU0:R1 (config-ospf)# area 0
RP/0/RP0/CPU0:R1 (config-ospf-ar) interface TenGigE0/0/0/2
RP/0/RP0/CPU0:R1 (config-isis-ar-if) # fast-reroute per-prefix
RP/0/RP0/CPU0:R1 (config-isis-ar-if) # fast-reroute per-prefix ti-lfa
RP/0/RP0/CPU0:R1 (config-isis-ar-if) # exit
```

(注) この例では、特定のインターフェイスで TI-LFA を設定しました。TI-LFA はインスタンスまたはエリアに設定できます。インスタンスまたはエリア内のすべてのインターフェイスが設定を継承します。

ステップ 2 R1 を R5 に接続するセグメントルーティング トンネル インターフェイス上で自動ルート通知を設定します。

例 :

```
RP/0/RP0/CPU0:R1(config)# interface tunnel-tel
RP/0/RP0/CPU0:R1(config-if)# ipv4 unnumbered Loopback0
RP/0/RP0/CPU0:R1(config-if)# autoroute announce
RP/0/RP0/CPU0:R1(config-if)# destination 192.168.5.1
RP/0/RP0/CPU0:R1(config-if)# path-option 1 dynamic segment-routing
```

ダイナミックセグメントルーティングパスオプションは、セグメントルーティングに隣接関係 SID を使用するように設定されています。

ステップ 3 ネットワークのセグメントルーティンググローバルブロック (SRGB) を定義します。

例 :

```
RP/0/RP0/CPU0:R1(config-isis)# segment-routing global-block 50000 60000
```

セグメントルーティングがプレフィックス SID とともに使用されるように設定する場合、セグメントルーティングラベルは定義されたグローバルブロックから割り当てられます。この例では、隣接関係 SID を動的に使用するように送信元ルータを設定しているため、ラベル割り当てでは SRGB は使用されません。

ステップ 4 設定をコミットします。

例 :

```
RP/0/RP0/CPU0:R1# commit
```

ステップ 5 送信元ルータと宛先ルータ間のパス上の IP FRR 保護を確認します。

次の出力は、単一の PQ ノードトポロジの出力です。

例 :

```
RP/0/RP0/CPU0:R1# show mpls traffic-eng forwarding tunnels 1 detail
Tunnel      Outgoing      Outgoing      Next Hop      Bytes
Name        Label         Interface     Interface     Switched
-----
tel         (SR) Pop      Te0/0/0/1.100 10.15.1.2     31340256

Updated: Aug 28 10:21:27.763
Path Flags: 0x400 [ BKUP-IDX:1 (0x0) ]
Version: 12635036, Priority: 2
Label Stack (Top -> Bottom): { Imp-Null }
NHID: 0x0, Encap-ID: N/A, Path idx: 0, Backup path idx: 1, Weight: 0
MAC/Encaps: 18/18, MTU: 1496
Packets Switched: 26616

          50103 Te0/0/0/11.100 10.12.2.2      0          (!)
Updated: Aug 28 10:21:27.763
Path Flags: 0x100 [ BKUP, NoFwd ]
Version: 12635036, Priority: 2
Label Stack (Top -> Bottom): { 50103 50105 }
NHID: 0x0, Encap-ID: N/A, Path idx: 1, Backup path idx: 0, Weight: 0
MAC/Encaps: 18/26, MTU: 1496
Packets Switched: 0
(!): FRR pure backup
```



```
Interface Handle: 0x08000120, Local Label: 24002
Forwarding Class: 0, Weight: 0
Packets/Bytes Switched: 34727459/40968290594
```

次の出力は、ディスジョイント PQ ノードの出力です。

例:

```
RP/0/RP0/CPU0:R1# show mpls forwarding tunnels 1 detail

Tunnel      Outgoing   Outgoing   Next Hop   Bytes
Name        Label      Interface  Next Hop   Switched
-----
tt1         (SR) Pop   Te0/0/0/1.100  10.15.1.2  65361590
Updated: Aug 31 07:52:17.630
Path Flags: 0x400 [ BKUP-IDX:1 (0x0) ]
Version: 42799904, Priority: 2
Label Stack (Top -> Bottom): { Imp-Null }
NHID: 0x0, Encap-ID: N/A, Path idx: 0, Backup path idx: 1, Weight: 0
MAC/Encaps: 18/18, MTU: 1496
Packets Switched: 55581

          50103   Te0/0/0/2  10.12.100.2  0          (!)
Updated: Aug 31 07:52:17.630
Path Flags: 0x100 [ BKUP, NoFwd ]
Version: 42799904, Priority: 2
Label Stack (Top -> Bottom): { 50103 50104 50105 }
NHID: 0x0, Encap-ID: N/A, Path idx: 1, Backup path idx: 0, Weight: 0
MAC/Encaps: 14/26, MTU: 1500
Packets Switched: 0
(!): FRR pure backup

Interface Handle: 0x08000120, Local Label: 24029
Forwarding Class: 0, Weight: 0
Packets/Bytes Switched: 97227973/114534209178
```

単一の PQ ノードの出力は、R1 から R5 へのプライマリパス（トランジットルータがないためラベルがポップされています）が、ラベル 50103（ルータ R1 から PQ ノード、ルータ R3 へのパスの場合）および 50105（ルータ R3 から宛先ルータ R5 へのパスの場合）が付いたバックアップパスによって保護されていることを確認しています。

ディスジョイント PQ ノードの出力は、R1 から R5 へのプライマリパス（トランジットルータがないためラベルがポップされています）が、ラベル 50103（ルータ R1 から P ノード、ルータ R3 へのパスの場合）、50104（P ノード（ルータ R3）から Q ノード、ルータ R4 へのパスの場合）、および 50105（ルータ R4 から宛先ルータ R5 へのパスの場合）が付いたバックアップパスによって保護されていることを確認しています。

(注) ダイナミックセグメントルーティングがルータ上で設定されている場合、隣接関係 SID が使用され、ラベルは SRGB から選択されません（この例では、ローカルラベル 24002 で示されています）。

TI-LFA がセグメントルーティング用に正常に設定されました。



第 9 章

セグメント ルーティング マイクロループ回避の設定

セグメント ルーティング マイクロループ回避機能により、IS-IS などのリンクステート ルーティング プロトコルを使用して、トポロジ変更後のネットワーク コンバージェンス中に発生するマイクロループを防止または回避することができます。

- [セグメント ルーティング マイクロループ回避について \(69 ページ\)](#)
- [IS-IS 向けセグメント ルーティング マイクロループ回避の設定 \(69 ページ\)](#)

セグメント ルーティング マイクロループ回避について

マイクロループは、トポロジの変更（リンク ダウン、リンク アップ、またはメトリック変更 イベント）後にネットワークで発生する短いパケットループです。マイクロループは、ネットワーク内の異なるノードの非同時コンバージェンスによって引き起こされます。ノードが収束し、まだ収束していないネイバーノードにトラフィックを送信すると、これら2つのノード間でトラフィックがループし、パケット損失、ジッター、および順不同パケットが発生する可能性があります。

セグメント ルーティング マイクロループ回避機能は、トポロジの変更後にマイクロループが可能かどうかを検出します。新しいトポロジでマイクロループが発生する可能性がある場合、ノードが計算した場合、ノードはセグメントのリストを使用して宛先へのループフリー SR-TE ポリシーパスを作成します。RIB 更新遅延タイマーの有効期限が切れた後、SR-TE ポリシーは通常の転送パスに置き換えられます。

IS-IS 向けセグメント ルーティング マイクロループ回避の設定

このタスクでは、セグメント ルーティング マイクロループ回避を有効にし、IS-IS のルーティング情報ベース（RIB）更新遅延値を設定する方法について説明します。

始める前に

次のトポロジ要件を満たしていることを確認してください。

- ルータ インターフェイスがトポロジごとに設定されている。
- ルータが IS-IS で設定されている。
- IS-IS のセグメント ルーティングが設定されている。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	router isis instance-id 例 : RP/0/RP0/CPU0:router (config) # router isis 1	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実行されるルーティングのレベルを変更できます。
ステップ 3	address-family ipv4 [unicast] 例 : RP/0/RP0/CPU0:router (config-isis) # address-family ipv4 unicast	IPv4 アドレス ファミリを指定し、ルータ 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 4	microloop avoidance segment-routing 例 : RP/0/RP0/CPU0:router (config-isis-af) # microloop avoidance segment-routing	セグメント ルーティング マイクロループ回避を有効にします。
ステップ 5	microloop avoidance rib-update-delay delay-time 例 : RP/0/RP0/CPU0:router (config-isis-af) # microloop avoidance rib-update-delay 3000	転送テーブルを更新する前に、ノードがマイクロループ回避ポリシーを使用する時間を指定します。 <i>delay-time</i> の単位はミリ秒です。値の範囲は 1 ~ -60000 です。デフォルト値は 5000 です。



第 10 章

セグメント ルーティング マッピング サーバの設定

マッピングサーバは LDP とセグメント ルーティング間のインターワーキングの主要コンポーネントです。LDP ノードと相互接続できるように SR 対応ノードを有効にします。マッピングサーバは、他の非 SR 対応ノードの代わりに IGP でプレフィックスと SID のマッピングをアドバタイズします。

- [セグメント ルーティング マッピング サーバ \(71 ページ\)](#)
- [セグメント ルーティングと LDP の相互運用性 \(72 ページ\)](#)
- [マッピング サーバの設定 \(75 ページ\)](#)
- [マッピング アドバタイズメントの有効化 \(77 ページ\)](#)
- [マッピング クライアントの有効化 \(79 ページ\)](#)

セグメント ルーティング マッピング サーバ

Cisco IOS XR セグメント ルーティングのマッピングサーバ機能では、プレフィックス SID が既知のプレフィックスの一部またはすべてに集中的に割り当てられます。ルータは、マッピングサーバ、マッピングクライアント、またはその両方として動作する必要があります。

- マッピングサーバとして機能するルータでは、ユーザは、SID マッピング エントリを設定して、一部またはすべてのプレフィックスに対しプレフィックス SID を指定できます。これにより、ローカル SID マッピング ポリシーが作成されます。ローカル SID マッピング ポリシーには、重複しない SID マッピング エントリが含まれています。マッピングサーバは、ローカル SID マッピング ポリシーをマッピングクライアントにアドバタイズします。
- マッピングクライアントとして機能するルータは、マッピングサーバからリモートで受信した SID を受信して解析し、リモート SID マッピング エントリを作成します。
- マッピングサーバおよびマッピングクライアントとして機能するルータは、リモートで学習されローカルに設定されたマッピングエントリを使用して、重複しない一貫したアクティブなマッピングポリシーを構築します。IGP インスタンスは、アクティブなマッピン

グポリシーを使用して、一部またはすべてのプレフィックスのプレフィックス SID を計算します。

マッピングサーバは、マッピング エントリの挿入および削除を自動的に管理して、重複しない一貫した SID マッピング エントリを含むアクティブなマッピング ポリシーを常に生成します。

- ローカルに設定されたマッピング エントリは、互いに重複してはいけません。
- マッピングサーバは、ローカルに設定されたマッピング ポリシーと、特定の IGP インスタンスからリモートで学習されたマッピング エントリを入力として受け取り、その IGP インスタンスの設定ルールに従って重複するマッピング エントリの中から単一のマッピング エントリを選択します。その結果、重複しない一貫したマッピング エントリで構成されるアクティブなマッピング ポリシーが作成されます。
- 定常状態では、少なくとも同じエリアまたはレベルにあるすべてのルータは、同一のアクティブなマッピング ポリシーを持っている必要があります。

セグメントルーティング マッピング サーバの制限事項

- ネットワーク内のマッピングサーバの位置は重要ではありません。ただし、マッピング アドバタイズメントは通常の IGP アドバタイズメント メカニズムを使用して IGP に配布されるため、マッピングサーバにはネットワークへの IGP 隣接関係が必要です。
- マッピングサーバの役割は非常に重要です。冗長性を確保するには、ネットワーク内に複数のマッピングサーバを設定する必要があります。
- マッピングサーバ機能は、1つの IS-IS インスタンスを通じて学習された SID マッピング エントリが、プレフィックスのプレフィックス SID を決定するために別の IS-IS インスタンスによって使用されるというシナリオをサポートしていません。たとえば、「ルータの isis 1」によってリモートルータから学習されたマッピング エントリを使用して、「ルータ isis 2」によって FIB に学習、アドバタイズ、またはダウンロードされたプレフィックスのプレフィックス SID を計算することはできません。マッピングサーバは IS-IS 領域ごとに必要です。
- セグメントルーティング マッピングサーバは現在、Virtual Routing and Forwarding (VRF) をサポートしていません。

セグメントルーティングと LDP の相互運用性

IGP では、セグメントルーティング (SR) が Label Distribution Protocol (LDP; ラベル配布プロトコル) と相互運用するためのメカニズムが提供されます。セグメントルーティングのコントロールプレーンは、LDP と共存します。

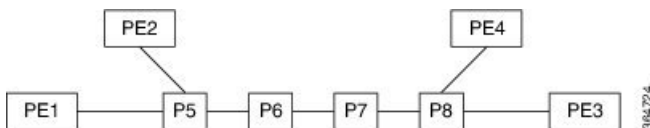
SR のセグメントルーティング マッピングサーバ (SRMS) 機能は、SR をサポートしていないネットワークの LDP 部分で、宛先に SID をアドバタイズするために使用されます。SRMS

は、そのような宛先へのセグメント識別子 (SID) マッピングエントリを維持およびアドバタイズします。IGP は SRMS マッピングエントリを伝播し、SRMS と相互に作用して、フォワーディングプレーンのプログラミング時に SID 値を決定します。IGP は、転送情報ベース (FIB) をプログラムするために使用されるプレフィックスと対応するラベルをルーティング情報ベース (RIB) にインストールします。

例：セグメントルーティング LDP の相互運用性

セグメントルーティング (SR) と Label Distribution Protocol (LDP; ラベル配布プロトコル) を混在させたネットワークを考えてみましょう。相互運用性を促進することにより、連続するマルチプロトコルラベルスイッチング (MPLS) LSP (ラベル付きスイッチドパス) を確立できます。SR ドメイン内の 1 つ以上のノードがセグメントルーティングマッピングサーバ (SRMS) として機能します。SRMS は、非 SR 対応ノードに代わって SID マッピングをアドバタイズします。各 SR 対応ノードは、個々のノードを明示的に構成することなく、非 SR 対応ノードに割り当てられた SID について学習します。

次の図に示すようなネットワークを考えてみましょう。このネットワークは、LDP 対応ノードと SR 対応ノードの両方を組み合わせたものです。



この混在ネットワークでは、

- ノード P6、P7、P8、PE4、および PE3 は LDP に対応しています。
- ノード PE1、PE2、P5、P6 は SR に対応しています。
- ノード PE1、PE2、P5、および P6 は、セグメントルーティンググローバルブロック (SRGB) が (100, 200) に設定されています。
- ノード PE1、PE2、P5、および P6 は、ノードセグメントがそれぞれ 101、102、105、106 に設定されています。

サービスフローは、連続する MPLS トンネル上で PE1 から PE3 まで確立する必要があります。これには、SR と LDP の相互運用が必要です。

LDP から SR へ

LDP から SR へのトラフィックフロー (右から左へ) では、次のような流れとなります。

1. PE3 は、nhop が PE1 であるサービスルート进行学习します。PE3 には、FEC PE1 用に nhop P8 からの LDP ラベルバインドがあります。PE3 はパケット P8 を転送します。
2. P8 には、FEC PE1 用に nhop P7 からの LDP ラベルバインドがあります。P8 はパケットを P7 に転送します。
3. P7 には、FEC PE1 用に nhop P6 からの LDP ラベルバインドがあります。P7 はパケットを P6 に転送します。

4. P6には、FEC PE1用のnhop P5からのLDPバインドがありません。しかし、P6には、IGPルート PE1へのSRノードセグメントがあります。P6はパケットをP5に転送し、等価ノードセグメント101によってローカルLDPラベルをFEC PE1と交換します。このプロセスは、ラベルのマージと呼ばれます。
5. P5は、PE1が最後から2番目のポップフラグがセットされたノードセグメント101をアドバタイズしたと仮定して101をポップし、PE1に転送します。
6. PE1は、トンネリングされたパケットを受信し、サービスラベルを処理します。

エンドツーエンドのMPLSトンネルは、PE3からP6までのLDP LSPと、P6からPE1までの関連ノードセグメントから確立されます。

SR から LDP へ

オペレータがセグメントルーティングマッピングサーバ (SRMS) としてP5を設定し、マッピング (P7, 107)、(P8, 108)、(PE3, 103) および (PE4, 104) をアドバタイズすると仮定します。PE3がSR対応だった場合、オペレータはPE3にノードセグメント103を設定している可能性があります。PE3は非SR対応であるため、オペレータはそのポリシーをSRMSで設定します。SRMSは非SR対応ノードに代わってマッピングをアドバタイズします。冗長性のために、複数のSRMSサーバをネットワークにプロビジョニングできます。マッピングサーバのアドバタイズメントは、SR対応ノードによってのみ認識されます。SR対応ルータは、ノードセグメントがノード自体によってアドバタイズされた場合と全く同じ方法で、関連するノードセグメントをMPLSデータプレーンにインストールします。

SRからLDPへのトラフィックフロー (左から右へ) では、次のような流れとなります。

1. PE1は、PE3がノードセグメント103をアドバタイズした場合と全く同じ方法で、ノードセグメント103をnhop P5でインストールします。
2. P5は103を103と交換し、P6に転送します。
3. IGPルートPE3に対するP6のnhopは非SR対応です。(P7はSR機能をアドバタイズしません)。ただし、P6には同じFECに対してそのnhopからのLDPラベルバインドがあります。(たとえば、LDPラベル1037)。P6は103を1037と交換し、P7に転送します。このプロセスをラベルマージと呼びます。
4. P7はこのラベルをP8から受け取ったLDPラベルと交換し、P8に転送します。
5. P8はLDPラベルをポップし、PE3に転送します。
6. PE3はパケットを受信し、必要に応じて処理します。

エンドツーエンドのMPLS LSPは、PE1からP6までのSRノードセグメントと、P6からPE3までのLDP LSPから確立されます。

マッピングサーバの設定

これらのタスクを実行して、マッピングサーバを設定し、プレフィックス SID マッピング エントリをアクティブなローカルマッピングポリシーに追加します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	segment-routing 例 : RP/0/RP0/CPU0:router(config)# segment-routing	セグメントルーティングを有効にします。
ステップ 3	mapping-server 例 : RP/0/RP0/CPU0:router(config-sr)# mapping-server	マッピングサーバコンフィギュレーションモードを有効にします。
ステップ 4	prefix-sid-map 例 : RP/0/RP0/CPU0:router(config-sr-ms)# prefix-sid-map	プレフィックス SID マッピングコンフィギュレーションモードを有効にします。 (注) 双方向プレフィックス SID は、IS-IS の下で直接、またはマッピングサーバ経由で有効にできます。
ステップ 5	address-family ipv4 ipv6 例 : 次の例に、ipv4 用のアドレスファミリーを示します。 RP/0/RP0/CPU0:router(config-sr-ms-map)# address-family ipv4 次の例に、ipv6 用のアドレスファミリーを示します。 RP/0/RP0/CPU0:router(config-sr-ms-map)# address-family ipv6	IS-IS 用のアドレスファミリーを設定します。

	コマンドまたはアクション	目的
ステップ 6	<p><code>ip-address/prefix-length first-SID-value range range</code></p> <p>例 :</p> <pre>RP/0/RP0/CPU0:router (config-sr-ms-map-af) # 10.1.1.1/32 10 range 200 RP/0/RP0/CPU0:router (config-sr-ms-map-af) # 20.1.0.0/16 400 range 300</pre>	<p>アクティブなローカル マッピング ポリシーに SID マッピング エントリを追加します。設定された例では、</p> <ul style="list-style-type: none"> • プレフィックス 10.1.1.1/32 にはプレフィックス SID 10 が割り当てられ、プレフィックス 10.1.1.2/32 にはプレフィックス SID 11 が割り当てられ、プレフィックス 10.1.1.199/32 にはプレフィックス SID 200 が割り当てられています。 • プレフィックス 20.1.0.0/16 にはプレフィックス SID 400 が割り当てられ、プレフィックス 20.2.0.0/16 にはプレフィックス SID 401 が割り当てられ、以下同様となります。
ステップ 7	commit	
ステップ 8	<p>show segment-routing mapping-server prefix-sid-map [ipv4 ipv6] [detail]</p> <p>例 :</p> <pre>RP/0/RP0/CPU0:router# show segment-routing mapping-server prefix-sid-map ipv4 Prefix SID Index Range Flags 20.1.1.0/24 400 300 10.1.1.1/32 10 200 Number of mapping entries: 2 RP/0/RP0/CPU0:router# show segment-routing mapping-server prefix-sid-map ipv4 detail Prefix 20.1.1.0/24 SID Index: 400 Range: 300 Last Prefix: 20.2.44.0/24 Last SID Index: 699 Flags: 10.1.1.1/32 SID Index: 10 Range: 200 Last Prefix: 10.1.1.200/32 Last SID Index: 209 Flags: Number of mapping entries: 2</pre>	<p>ローカルで設定されたプレフィックス/SID マッピングに関する情報を表示します。</p> <p>(注) IS-IS用のアドレスファミリを指定します。</p>

	コマンドまたはアクション	目的

次のタスク

IGP でローカル SID マッピング ポリシーのアダプタイズメントを有効にします。

マッピングアダプタイズメントの有効化

スタティック マッピング ポリシーの設定に加えて、IGP でマッピングのアダプタイズメントを有効にする必要があります。

IGP がローカルに設定されたプレフィックス SID マッピングをアダプタイズできるようにするには、次の手順を実行します。

IS-IS 向けマッピングアダプタイズメントの設定

手順

	コマンドまたはアクション	目的
ステップ 1	router isis instance-id 例 : RP/0/RP0/CPU0:router(config)# router isis 1	指定したルーティング インスタンスの IS-IS ルーティングをイネーブルにし、ルータをルータ コンフィギュレーション モードにします。 <ul style="list-style-type: none"> • is-type ルータ コンフィギュレーション コマンドを使用して、特定のルーティング インスタンスによって実行されるルーティングのレベルを変更できます。
ステップ 2	address-family { ipv4 ipv6 } [unicast] 例 : 次に、IPv4 アドレス ファミリの例を示します。 RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast	IPv4 または IPv6 アドレス ファミリを指定して、ルータ 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 3	segment-routing prefix-sid-map advertise-local 例 :	ローカルに設定されたプレフィックス SID マッピングをアダプタイズするように IS-IS を設定します。

	コマンドまたはアクション	目的
	<pre>RP/0/RP0/CPU0:router(config-isis-af)# segment-routing prefix-sid-map advertise-local</pre>	
ステップ 4	commit	
ステップ 5	show isis database verbose 例 : <pre>RP/0/RP0/CPU0:router# show isis database verbose <...removed...> SID Binding: 10.1.1.1/32 F:0 M:0 S:0 D:0 A:0 Weight:0 Range:200 SID: Start:10, Algorithm:0, R:0 N:0 P:0 E:0 V:0 L:0 SID Binding: 20.1.1.0/24 F:0 M:0 S:0 D:0 A:0 Weight:0 Range:300 SID: Start:400, Algorithm:0, R:0 N:0 P:0 E:0 V:0 L:0</pre>	IS-IS プレフィックス SID マッピング アドバタイズメントと TLV を表示します。

OSPF 向けマッピング アドバタイズメントの設定

手順

	コマンドまたはアクション	目的
ステップ 1	router ospf process-name 例 : <pre>RP/0/RP0/CPU0:router(config)# router ospf 1</pre>	指定したルーティング インスタンスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。
ステップ 2	segment-routing prefix-sid-map advertise-local 例 : <pre>RP/0/RP0/CPU0:router(config-ospf)# segment-routing prefix-sid-map advertise-local</pre>	ローカルに設定されたプレフィックス SID マッピングをアドバタイズするように OSPF を設定します。
ステップ 3	commit	
ステップ 4	show ospf database opaque-area 例 : <pre>RP/0/RP0/CPU0:router# show ospf</pre>	OSPF プレフィックス SID マッピング アドバタイズメントと TLV を表示します。

	コマンドまたはアクション	目的
	<pre> database opaque-area <...removed...> Extended Prefix Range TLV: Length: 24 AF : 0 Prefix : 10.1.1.1/32 Range Size: 200 Flags : 0x0 SID sub-TLV: Length: 8 Flags : 0x60 MTID : 0 Algo : 0 SID Index : 10 </pre>	

マッピングクライアントの有効化

デフォルトでは、マッピングクライアント機能は有効になっています。

segment-routing prefix-sid-map receive disable コマンドを使用して、マッピングクライアント機能を無効にできます。

segment-routing prefix-sid-map receive コマンドを使用して、マッピングクライアント機能を再び有効にできます。

次に、IS-IS 用にマッピングクライアントを有効にする例を示します。

```

RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# segment-routing prefix-sid-map receive

```

次に、OSPF 用にマッピングクライアントを有効にする例を示します。

```

RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# segment-routing prefix-sid-map receive

```




第 11 章

セグメントルーティングトラフィックマトリックスの使用

このモジュールでは、セグメントルーティングトラフィックマトリックス (SR-TM) およびトラフィックコレクタプロセスに関する情報を提供し、TM ボーダーとトラフィックコレクタを設定し、トラフィック情報を表示する方法について説明します。

- [セグメントルーティングトラフィックマトリックス \(81 ページ\)](#)
- [トラフィックコレクタプロセス \(81 ページ\)](#)
- [トラフィックコレクタの設定 \(82 ページ\)](#)
- [トラフィック情報の表示 \(83 ページ\)](#)

セグメントルーティングトラフィックマトリックス

ネットワークのトラフィックマトリックスは、ネットワークを出入りする集約トラフィックフローの説明、測定、または推定です。

セグメントルーティングトラフィックマトリックス (SR-TM) は、ユーザがルータ上のトラフィックパターンを理解できるように設計されています。トラフィックマトリックスボーダーはネットワークを 2 つの部分に分割します。内部 (境界内にあるインターフェイス) と外部 (境界外のインターフェイス) です。デフォルトでは、すべてのインターフェイスは内部です。インターフェイスは外部として設定できます。

トラフィックコレクタプロセス

トラフィックコレクタは、プレフィックスカウンタ、トンネルカウンタ、TM カウンタなどのルータコンポーネントからパケットおよびバイトの統計情報を収集します。これは、外部インターフェイスからネットワークへのトラフィックがセグメントルーティングのプレフィックス SID を宛先とする場合に増加します。トラフィックコレクタは統計情報の履歴を保持し、プロセスの再起動、フェールオーバー、および ISSU の間でも維持されます。履歴は設定可能な期間保持されます。

Pcounter

Pcounter は、カウンタの packets と bytes のペアです。トンネルごとに 1 つの Pcounter があります。プレフィックス SID ごとに 2 つの Pcounter があります。

- ベース Pcounter : プレフィックス SID 転送情報ベース (FIB) エントリで切り替えられる packets
- TmPcounter : 外部インターフェイスからの packets で、プレフィックス SID FIB エントリで切り替えられる packets

トラフィックコレクタは、すべてのプレフィックス SID のベース Pcounter と TmPcounter、およびすべてのトンネルインターフェイスの Pcounter を定期的に収集します。

各 Pcounter について、トラフィックコレクタは最後の間隔で転送された packets 数と bytes 数を計算します。トラフィックコレクタは、各 Pcounter の間隔ごとの統計の履歴を保持します。履歴の各エントリには次のものが含まれます。

- 間隔の開始時刻と終了時刻
- 間隔中に転送された packets 数
- 間隔中に転送された bytes 数

トラフィックコレクタの設定

トラフィックコレクタを設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure	
ステップ 2	traffic-collector 例 : RP/0/RP0/CPU0:router(config)# traffic-collector	トラフィックコレクタを有効にし、ルータをトラフィックコレクタコンフィギュレーションモードにします。
ステップ 3	statistics collection-interval value 例 : RP/0/RP0/CPU0:router(config-tc)# statistics collection-interval 5	(オプション) トラフィックコレクタがデータを収集して投稿する頻度を分単位で設定します。有効な値は 1、2、3、4、5、6、10、12、15、20、30、および 60 です。デフォルトの間隔は、1 秒です。

	コマンドまたはアクション	目的
ステップ 4	statistics history-size value 例 : <pre>RP/0/RP0/CPU0:router(config-tc)# statistics history-size 10</pre>	(オプション) 履歴データベースに保持されているエントリの数を指定します。有効な値は 1 ~ 10 です。デフォルトは 5 分です。 (注) エントリ数は、平均パケットレートと平均バイトレートの計算方法に影響します。レートは履歴の範囲で計算され、リアルタイムでの平均ではありません。
ステップ 5	statistics history-timeout value 例 : <pre>RP/0/RP0/CPU0:router(config-tc)# statistics history-timeout 24</pre>	(オプション) プレフィックス SID または tunnel-te インターフェイスが削除されると、history-timeout は、プレフィックス SID およびトンネル統計情報が削除される前に履歴に保持される時間の長さを時間単位で設定します。最小は 1 時間で、最大は 720 時間です。デフォルト値は 48 です。 (注) 履歴タイムアウトをディセーブルにするには、0 を入力します。(履歴は保持されません)。
ステップ 6	interface type l3-interface-address 例 : <pre>RP/0/RP0/CPU0:router(config-tc)# interface TenGigE 0/0/0/3</pre>	外部トラフィックを処理するインターフェイスを識別します。外部トラフィックに対しては、L3 インターフェイスのみがサポートされています。
ステップ 7	commit	

これでトラフィック コレクタの設定が完了します。

トラフィック情報の表示

次の show コマンドは、インターフェイスとトンネルに関する情報を表示します。



(注) 次の **show** コマンドのコマンド構文の詳細については、『*Segment Routing Command Reference Guide*』を参照してください。

- 設定済みの外部インターフェイスを表示します。

```
RP/0/RP0/CPU0:router# show traffic-collector external-interface
Interface                Status
-----
Te0/0/0/3                Enabled
Te0/0/0/4                Enabled
```

- プレフィックス SID のカウンタ履歴データベースを表示します。

```
RP/0/RP0/CPU0:router# show traffic-collector ipv4 counters prefix 1.1.1.10/32 detail
Prefix: 1.1.1.10/32 Label: 16010 State: Active
Base:
Average over the last 5 collection intervals:
Packet rate: 9496937 pps, Byte rate: 9363979882 Bps

History of counters:
23:01 - 23:02: Packets 9379529, Bytes: 9248215594
23:00 - 23:01: Packets 9687124, Bytes: 9551504264
22:59 - 23:00: Packets 9539200, Bytes: 9405651200
22:58 - 22:59: Packets 9845278, Bytes: 9707444108
22:57 - 22:58: Packets 9033554, Bytes: 8907084244
TM Counters:
Average over the last 5 collection intervals:
Packet rate: 9528754 pps, Byte rate: 9357236821 Bps

History of counters:
23:01 - 23:02: Packets 9400815, Bytes: 9231600330
23:00 - 23:01: Packets 9699455, Bytes: 9524864810
22:59 - 23:00: Packets 9579889, Bytes: 9407450998
22:58 - 22:59: Packets 9911734, Bytes: 9733322788
22:57 - 22:58: Packets 9051879, Bytes: 8888945178
```

この出力には、指定されたプレフィックス SID の平均 Pcounter（パケット、バイト）、Pcounter 履歴、ベースおよび TM の収集間隔が示されます。

- トンネルのカウンタ履歴データベースを表示します。

```
RP/0/RP0/CPU0:router# show traffic-collector counters tunnels tunnel-te 1 detail
Tunnel: tt1 State: Active
Average over the last 5 collection intervals:
Packet rate: 9694434 pps, Byte rate: 9597489858 Bps

History of counters:
23:14 - 23:15: Packets 9870522 , Bytes: 9771816780
23:13 - 23:14: Packets 9553048 , Bytes: 9457517520
23:12 - 23:13: Packets 9647265 , Bytes: 9550792350
23:11 - 23:12: Packets 9756654 , Bytes: 9659087460
23:10 - 23:11: Packets 9694434 , Bytes: 9548235180
```

この出力には、トンネルの平均 Pcounter（パケット、バイト）、Pcounter 履歴、および収集間隔が示されます。



第 12 章

セグメント ルーティング OAM の使用

セグメントルーティング保守運用管理 (OAM) は、ネットワークの障害検出とトラブルシューティングに役立ちます。これを使用することで、サービス プロバイダーはラベルスイッチドパス (LSP) をモニタしてフォワーディングの問題を迅速に隔離できます。セグメントルーティング OAM 機能では、BGP プレフィックス SID、Nil-FEC (転送等価クラス) LSP Ping および Traceroute 機能のサポートを提供します。

- [BGP および IGP プレフィックス SID 用の MPLS Ping および Traceroute \(87 ページ\)](#)
- [例：プレフィックス SID の MPLS Ping、Traceroute、およびツリートレース \(88 ページ\)](#)
- [MPLS LSP ping および traceroute Nil FEC ターゲット \(90 ページ\)](#)
- [例：Nil_FEC ターゲットの LSP Ping および Traceroute \(91 ページ\)](#)
- [セグメントルーティング Ping \(92 ページ\)](#)
- [セグメントルーティング Traceroute \(94 ページ\)](#)
- [IPv6 OAM を介したセグメント ルーティング \(97 ページ\)](#)

BGP および IGP プレフィックス SID 用の MPLS Ping および Traceroute

プレフィックス SID 用の MPLS Ping および Traceroute の操作は、次のようなさまざまな BGP および IGP シナリオでサポートされています。

- IS-IS レベルまたは OSPF エリア内
- IS-IS レベルまたは OSPF エリア間
- IS-IS から OSPF へ、および OSPF から IS-IS へのルート再配布
- エニーキャスト プレフィックス SID
- BGP と LDP によってシグナリングされた LSP の組み合わせ

MPLS LSP ping 機能を使用して、LSP に沿った入力ラベルスイッチルータ (LSR) と出力 LSR 間の接続を確認します。MPLS LSP ping は、Internet Control Message Protocol (ICMP) のエコー要求メッセージと応答メッセージと同様に、LSP の検証に MPLS エコーの要求メッセージと応

答メッセージを使用します。MPLS エコー要求パケットの宛先 IP アドレスは、ラベルスタックの選択に使用されるアドレスとは異なります。宛先 IP アドレスは 127.x.y.z/8 アドレスとして定義され、LSP が壊れている場合は IP パケットがそれ自体の宛先へ IP を切り替えないようにします。

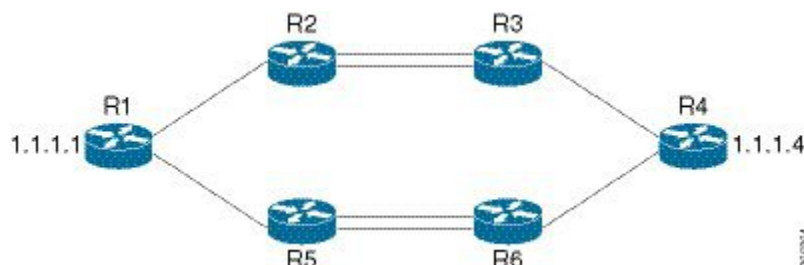
MPLS LSP traceroute 機能を使用して、LSP の障害ポイントを隔離します。これはホップバイホップ エラーのローカリゼーションとパス トレースに使用されます。MPLS LSP traceroute 機能は、エコー要求を送送するパケットの存続可能時間 (TTL) 値の期限切れに依存します。MPLS エコー要求メッセージが中継ノードを見つけると TTL 値をチェックし、期限が切れている場合はコントロールプレーンにパケットが渡されます。それ以外の場合は、メッセージが転送されます。エコー メッセージがコントロールプレーンに渡されると、要求メッセージの内容に基づいて応答メッセージが生成されます。

MPLS LSP ツリートレース (traceroute マルチパス) 操作は、BGP および IGP プレフィックス SID でもサポートされています。MPLS LSP ツリートレースでは、LSP のすべての可能な等コスト マルチパス (ECMP) ルーティング パスを検出して宛先プレフィックス SID に到達する手段が提供されます。エコー要求パケットにエンコードされたマルチパスデータを使用して、ロードバランシング情報が照会されます。これにより、発信者は各 ECMP の実行を許可される場合があります。パケット TTL が応答ノードで期限切れになると、ノードはダウンストリームパスのリストとマルチパス情報を返します。これにより、オペレータは MPLS エコー応答内の各パスを実行できるようになります。この操作は、すべての ECMP が検出されて検証されるまで、TTL 値が増加しながら各パスのホップごとに繰り返し実行されます。

MPLS エコー要求パケットは、ターゲット FEC スタック サブ TLV を伝送します。ターゲット FEC サブ TLV は、レスポндаによって FEC 検証のために使用されます。BGP および IGP IPv4 プレフィックス サブ TLV がターゲット FEC スタック サブ TLV に追加されました。IGP IPv4 プレフィックス サブ TLV には、プレフィックス SID、プレフィックス長、およびプロトコル (IS-IS または OSPF) が含まれています。BGP IPv4 プレフィックス サブ TLV には、プレフィックス SID とプレフィックス長が含まれています。

例：プレフィックス SID の MPLS Ping、Traceroute、およびツリートレース

これらの例では、次のトポロジを使用しています。



プレフィックス SID の MPLS Ping

```
RP/0/RP0/CPU0:router-arizona# ping mpls ipv4 1.1.1.4/32
Thu Dec 17 01:01:42.301 PST
```

```
Sending 5, 100-byte MPLS Echos to 1.1.1.4,
      timeout is 2 seconds, send interval is 0 msec:
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
```

プレフィックス SID の MPLS Traceroute

```
RP/0/RP0/CPU0:router-arizona# traceroute mpls ipv4 1.1.1.4/32
Thu Dec 17 14:45:05.563 PST
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
 0 12.12.12.1 MRU 4470 [Labels: 16004 Exp: 0]
L 1 12.12.12.2 MRU 4470 [Labels: 16004 Exp: 0] 3 ms
L 2 23.23.23.3 MRU 4470 [Labels: implicit-null Exp: 0] 3 ms
! 3 34.34.34.4 11 ms
```

プレフィックス SID の MPLS ツリー トレース

```
RP/0/RP0/CPU0:router-arizona# traceroute mpls multipath ipv4 1.1.1.4/32
Thu Dec 17 14:55:46.549 PST
```

Starting LSP Path Discovery for 1.1.1.4/32

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```

LL!
Path 0 found,
  output interface TenGigE0/0/0/0 nexthop 12.12.12.2 source 12.12.12.1 destination
127.0.0.0 L!
Path 1 found,
  output interface TenGigE0/0/0/0 nexthop 12.12.12.2 source 12.12.12.1 destination
127.0.0.2 LL!
Path 2 found,
  output interface TenGigE0/0/0/1 nexthop 15.15.15.5 source 15.15.15.1 destination
127.0.0.1 L!
Path 3 found,
  output interface TenGigE0/0/0/1 nexthop 15.15.15.5 source 15.15.15.1 destination
127.0.0.0

Paths (found/broken/unexplored) (4/0/0)
Echo Request (sent/fail) (10/0)
Echo Reply (received/timeout) (10/0)
Total Time Elapsed 53 ms

```

MPLS LSP ping および traceroute Nil FEC ターゲット

Nil-FEC LSP ping および traceroute の操作は、通常の MPLS ping および traceroute の拡張機能です。

Nil-FEC LSP Ping/Traceroute 機能は、セグメントルーティングと MPLS スタティックをサポートしています。また、他のすべての LSP タイプに対する追加の診断ツールとしても機能します。この機能は、オペレータに以下を指定することを許可することで、オペレータがラベルスタックを自由にテストできるようにします。

- ラベルスタック
- 発信インターフェイス
- ネクストホップアドレス

セグメントルーティングの場合、ルーティングパスに沿った各セグメントノードラベルおよび隣接関係ラベルは、イニシエータのラベルスイッチルータ (LSR) からのエコー要求メッセージのラベルスタックに入れられます。MPLS データプレーンは、このパケットをラベルスタックターゲットに転送し、ラベルスタックターゲットはエコーメッセージを送り返します。

次の表に、ping および traceroute コマンドの構文を示します。

表 2: LSP ping および traceroute Nil FEC コマンド

コマンド構文
ping mpls nil-fec labels {label[,label]} [output { interface tx-interface} [nexthop nexthop-ip-addr]]
traceroute mpls nil-fec labels {label[,label]} [output { interface tx-interface} [nexthop nexthop-ip-addr]]

例 : Nil_FEC ターゲットの LSP Ping および Traceroute

これらの例では、次のトポロジを使用しています。

```
Node loopback IP address: 172.18.1.3   172.18.1.4   172.18.1.5   172.18.1.7
Node label:                16004         16005         16007
Nodes:                      Arizona ---- Utah ----- Wyoming ---- Texas

Interface:                  GigabitEthernet0/0/0/1   GigabitEthernet0/0/0/1
Interface IP address:       10.1.1.3                 10.1.1.4
```

```
RP/0/RP0/CPU0:router-utah# show mpls forwarding
```

```
Tue Jul  5 13:44:31.999 EDT
Local  Outgoing  Prefix          Outgoing      Next Hop      Bytes
Label  Label       or ID           Interface     Interface     Switched
-----
16004  Pop          No ID           Gi0/0/0/1    10.1.1.4     1392
      Pop          No ID           Gi0/0/0/2    10.1.2.2     0
16005  16005       No ID           Gi0/0/0/0    10.1.1.4     0
      16005       No ID           Gi0/0/0/1    10.1.2.2     0
16007  16007       No ID           Gi0/0/0/0    10.1.1.4     4752
      16007       No ID           Gi0/0/0/1    10.1.2.2     0
24000  Pop          SR Adj (idx 0)  Gi0/0/0/0    10.1.1.4     0
24001  Pop          SR Adj (idx 2)  Gi0/0/0/0    10.1.1.4     0
24002  Pop          SR Adj (idx 0)  Gi0/0/0/1    10.1.2.2     0
24003  Pop          SR Adj (idx 2)  Gi0/0/0/1    10.1.2.2     0
24004  Pop          No ID           tt10          point2point   0
24005  Pop          No ID           tt11          point2point   0
24006  Pop          No ID           tt12          point2point   0
24007  Pop          No ID           tt13          point2point   0
24008  Pop          No ID           tt30          point2point   0
```

Ping Nil FEC ターゲット

```
RP/0/RP0/CPU0:router-arizona# ping mpls nil-fec labels 16005,16007 output interface
GigabitEthernet 0/0/0/1 nexthop 10.1.1.4 repeat 1
```

```
Sending 1, 72-byte MPLS Echos with Nil FEC labels 16005,16007,
timeout is 2 seconds, send interval is 0 msec:
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
'X' - unknown return code, 'x' - return code 0
```

```
Type escape sequence to abort.
```

```
!
```

```
Success rate is 100 percent (1/1), round-trip min/avg/max = 1/1/1 ms
Total Time Elapsed 0 ms
```

Traceroute Nil FEC ターゲット

```
RP/0/RP0/CPU0:router-arizona# traceroute mpls nil-fec labels 16005,16007 output interface
GigabitEthernet 0/0/0/1 nexthop 10.1.1.4
```

```
Tracing MPLS Label Switched Path with Nil FEC labels 16005,16007, timeout is 2 seconds
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
'X' - unknown return code, 'x' - return code 0
```

```
Type escape sequence to abort.
```

```
0 10.1.1.3 MRU 1500 [Labels: 16005/16007/explicit-null Exp: 0/0/0]
L 1 10.1.1.4 MRU 1500 [Labels: implicit-null/16007/explicit-null Exp: 0/0/0] 1 ms
L 2 10.1.1.5 MRU 1500 [Labels: implicit-null/explicit-null Exp: 0/0] 1 ms
! 3 10.1.1.7 1 ms
```

セグメントルーティング Ping

MPLS LSP ping 機能を使用して、LSP の入力と出力の間の接続を確認します。MPLS LSP ping は、Internet Control Message Protocol (ICMP) のエコー要求メッセージと応答メッセージと同様に、LSP の検証に MPLS エコーの要求メッセージと応答メッセージを使用します。セグメントルーティング ping は、セグメントルーティング コントロールプレーン上で接続性検証を実行するための MPLS LSP ping の拡張機能です。



(注) セグメントルーティング ping は、発信デバイスがセグメントルーティングを実行している場合にのみ使用できます。

セグメントルーティング ping の操作は、セグメントルーティング コントロールプレーンが発信者側で使用可能な場合（優先されていない場合でも）にのみ開始できます。これにより、パス上でトラフィックを誘導する前に、SR パスを検証できます。セグメントルーティング ping は、汎用 FEC タイプまたは SR コントロールプレーン FEC タイプ（SR-OSPF、SR-ISIS）のいずれかを使用できます。複数のデバイスが MPLS コントロールプレーンを実行している（LDP など）、または SR FEC を認識していない混合ネットワークでは、汎用 FEC タイプを使用することで、デバイスがエコー要求を正常に処理して応答することができます。デフォルトでは、汎用 FEC タイプがセグメントルーティング ping エコー要求のターゲット FEC スタックで使用されます。汎用 FEC は、特定のコントロールプレーンに結合されていません。そのため、アドバタイジングプロトコルが不明の場合、またはエコー要求のパス中に変更される可能性がある場合に、パス検証を行うことができます。ターゲット FEC を指定する必要がある場合は、FEC タイプを OSPF、IS-IS、または BGP として選択できます。これにより、セグメントルーティング コントロールプレーンを実行し、セグメントルーティング IGP FEC を理解できるデバイスだけがエコー要求に応答することが保証されます。

設定例

次の例に、セグメントルーティング コントロール プレーンの接続性をテストするためにセグメントルーティング ping を使用する方法を示します。最初の例では、FEC のタイプは指定されていません。他の例に示すように、FEC タイプを指定することもできます。

```
RP/0/RP0/CPU0:router# ping sr-mpls 10.1.1.2/32
```

```
Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,  
timeout is 2 seconds, send interval is 0 msec:
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,  
'L' - labeled output interface, 'B' - unlabeled output interface,  
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,  
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,  
'P' - no rx intf label prot, 'p' - premature termination of LSP,  
'R' - transit router, 'I' - unknown upstream index,  
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
```

```
RP/0/RP0/CPU0:router# ping sr-mpls 10.1.1.2/32 fec-type generic
```

```
Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,  
timeout is 2 seconds, send interval is 0 msec:
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,  
'L' - labeled output interface, 'B' - unlabeled output interface,  
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,  
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,  
'P' - no rx intf label prot, 'p' - premature termination of LSP,  
'R' - transit router, 'I' - unknown upstream index,  
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

```
RP/0/RP0/CPU0:router# ping sr-mpls 10.1.1.2/32 fec-type igp ospf
```

```
Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,  
timeout is 2 seconds, send interval is 0 msec:
```

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,  
'L' - labeled output interface, 'B' - unlabeled output interface,  
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,  
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,  
'P' - no rx intf label prot, 'p' - premature termination of LSP,  
'R' - transit router, 'I' - unknown upstream index,  
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

```
RP/0/RP0/CPU0:router# ping sr-mpls 10.1.1.2/32 fec-type igp isis
```

```

Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,
    timeout is 2 seconds, send interval is 0 msec:

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
        'L' - labeled output interface, 'B' - unlabeled output interface,
        'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
        'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
        'P' - no rx intf label prot, 'p' - premature termination of LSP,
        'R' - transit router, 'I' - unknown upstream index,
        'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms

RP/0/RP0/CPU0:router# ping sr-mpls 10.1.1.2/32 fec-type bgp

Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,
    timeout is 2 seconds, send interval is 0 msec:

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
        'L' - labeled output interface, 'B' - unlabeled output interface,
        'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
        'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
        'P' - no rx intf label prot, 'p' - premature termination of LSP,
        'R' - transit router, 'I' - unknown upstream index,
        'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms

```

セグメントルーティング Traceroute

MPLS LSP traceroute を使用して、LSP の障害ポイントを隔離します。これはホップバイホップエラーのローカリゼーションとパス トレースに使用されます。MPLS LSP traceroute 機能は、エコー要求を送信するパケットの存続可能時間 (TTL) 値の期限切れに依存します。MPLS エコー要求メッセージが中継ノードを見つけると TTL 値をチェックし、期限が切れている場合はコントロールプレーンにパケットが渡されます。それ以外の場合は、メッセージが転送されます。エコーメッセージがコントロールプレーンに渡されると、要求メッセージの内容に基づいて応答メッセージが生成されます。セグメントルーティング traceroute 機能は、MPLS LSP traceroute 機能をセグメントルーティング ネットワークに拡張します。

セグメントルーティング ping と同様に、セグメントルーティング traceroute 操作は、セグメントルーティング コントロールプレーンが発信者側で使用可能な場合 (優先されていない場合でも) にのみ開始できます。セグメントルーティング traceroute は、汎用 FEC タイプまたは SR コントロールプレーン FEC タイプ (SR-OSPF、SR-ISIS) のいずれかを使用できます。デフォルトでは、汎用 FEC タイプがセグメントルーティング traceroute エコー要求のターゲット FEC スタックで使用されます。ターゲット FEC を指定する必要がある場合は、FEC タイプを OSPF、IS-IS、または BGP として選択できます。これにより、セグメントルーティング コン

トロールプレーンを実行し、セグメントルーティング IGP FEC を理解できるデバイスだけがエコー要求に応答することが保証されます。

MPLS ネットワーク内のルータにロード バランシングが存在すると、MPLS トラフィックをターゲットルータに伝送するための代替パスが提供されます。マルチパスセグメントルーティング **traceroute** 機能は、入力ルータと出力ルータ間で LSP のすべての可能なパスを検出する手段を提供します。

設定例

次の例に、セグメントルーティング **traceroute** を使用して、指定された IPv4 プレフィックス SID アドレスの LSP をトレースする方法を示します。最初の例では、FEC のタイプは指定されていません。他の例に示すように、FEC タイプを指定することもできます。

```
RP/0/RP0/CPU0:router# traceroute sr-mpls 10.1.1.2/32

Tracing MPLS Label Switched Path to 10.1.1.2/32, timeout is 2 seconds

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

 0 10.12.12.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.12.12.2 3 ms

RP/0/RP0/CPU0:router# traceroute sr-mpls 10.1.1.2/32 fec-type generic

Tracing MPLS Label Switched Path to 10.1.1.2/32, timeout is 2 seconds

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

 0 10.12.12.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.12.12.2 2 ms

RP/0/RP0/CPU0:router# traceroute sr-mpls 10.1.1.2/32 fec-type igp ospf

Tracing MPLS Label Switched Path to 10.1.1.2/32, timeout is 2 seconds

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
0 10.12.12.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.12.12.2 2 ms
```

```
RP/0/RP0/CPU0:router# traceroute sr-mpls 10.1.1.2/32 fec-type igp isis
```

Tracing MPLS Label Switched Path to 10.1.1.2/32, timeout is 2 seconds

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
0 10.12.12.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.12.12.2 2 ms
```

```
RP/0/RP0/CPU0:router#traceroute sr-mpls 10.1.1.2/32 fec-type bgp
```

Tracing MPLS Label Switched Path to 10.1.1.2/32, timeout is 2 seconds

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
0 10.12.12.1 MRU 1500 [Labels: implicit-null/implicit-null Exp: 0/0]
! 1 10.12.12.2 2 ms
```

次の例に、マルチパス **traceroute** を使用して、IPv4 プレフィックス SID に可能なすべてのパスを検出する方法を示します。

```
RP/0/RP0/CPU0:router# traceroute sr-mpls multipath 10.1.1.2/32
```

Starting LSP Path Discovery for 10.1.1.2/32

```
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
```

Type escape sequence to abort.

```
!
Path 0 found,
output interface GigabitEthernet0/0/0/2 nexthop 10.13.13.2
```

```

source 10.13.13.1 destination 127.0.0.0
!
Path 1 found,
  output interface Bundle-Ether1 nexthop 10.12.12.2
source 10.12.12.1 destination 127.0.0.0

Paths (found/broken/unexplored) (2/0/0)
Echo Request (sent/fail) (2/0)
Echo Reply (received/timeout) (2/0)
Total Time Elapsed 14 ms

```

IPv6 OAM を介したセグメントルーティング

IPv6 データプレーンを介したセグメントルーティング (SRv6) の実装では、新しいタイプのルーティング拡張ヘッダーが追加されます。そのため、ping や traceroute などの既存の ICMPv6 メカニズムを SRv6 ネットワークで使用できます。ping と traceroute 操作が SRv6 ネットワーク内の IPv6 対応または SRv6 対応ノードに対して動作する方法に変更はありません。

制約事項および使用上の注意事項

SRv6 OAM には、次の制限が適用されます。

- SRv6 SID への ping はサポートされていません。

例 : SRv6 OAM

次に、SRv6 ネットワークで ping を使用する例を示します。

```

RP/0/RP0/CPU0:Router# ping ipv6 2001::33:33:33:33
Mon Sep 17 20:04:10.068 UTC
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001::33:33:33:33, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

```

次に、SRv6 ネットワークで traceroute を使用する例を示します。

```

RP/0/RP0/CPU0:Router# traceroute ipv6 2001::33:33:33:33 probe 1 timeout 0 srv6
Fri Sep 14 15:59:25.170 UTC
Type escape sequence to abort.
Tracing the route to 2001::33:33:33:33
 1  2001::22:22:22:22[IP tunnel: DA=cafe:0:0:a4:1::: SRH =(2001::33:33:33:33 ,SL=1)]
 2  msec
 2  2001::2:2:2:2[IP tunnel: DA=cafe:0:0:a4:1::: SRH =(2001::33:33:33:33 ,SL=1)] 2
msec
 3  2001::44:44:44:44 2 msec
 4  2001::33:33:33:33 3 msec

```

次に、SRH を使用しない SRv6 ネットワークで traceroute を使用する例を示します。

```

RP/0/RSP1/CPU0:Router# traceroute ipv6 2001::44:44:44:44 srv6
Wed Jan 16 14:35:27.511 UTC
Type escape sequence to abort.
Tracing the route to 2001::44:44:44:44
 1  2001::2:2:2:2 3 msec 2 msec 2 msec

```

```
2 2001::44:44:44:44 3 msec 3 msec 3 msec
```

次に、VRF で指定した IP アドレスに対して ping を使用する例を示します。

```
RP/0/RP0/CPU0:Router# ping 10.15.15.1 vrf red
Mon Sep 17 20:07:10.085 UTC
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.15.15.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

次に、VRF で指定した IP アドレスに対して traceroute を使用する例を示します。

```
RP/0/RP0/CPU0:Router# traceroute 10.15.15.1 vrf red
Mon Sep 17 20:07:18.478 UTC

Type escape sequence to abort.
Tracing the route to 10.15.15.1
 0 10.15.15.1 3 msec 2 msec 2 msec
```

次に、VRF の CE1 (4.4.4.5) から CE2 (5.5.5.5) に対して traceroute を使用する例を示します。

```
RP/0/RP0/CPU0:Router# traceroute 5.5.5.5 vrf a
Wed Jan 16 15:08:46.264 UTC

Type escape sequence to abort.
Tracing the route to 5.5.5.5
 0 14.14.14.1 5 msec 1 msec 1 msec
 1 15.15.15.1 3 msec 2 msec 2 msec
 2 15.15.15.2 2 msec * 3 msec
```