MPLS トラフィック エンジニアリングおよび拡張機能の設定

MPLS トラフィック エンジニアリングおよび拡張機能の前提条件

MPLS TE を有効にする前に、ネットワークが次の Cisco IOS 機能をサポートしていることを確認します。

- Multiprotocol Label Switching：マルチプロトコル ラベル スイッチング
- IP シスコ エクスプレス フォワーディング
- Intermediate System-to-Intermediate System（IS-IS）または Open Shortest Path First（OSPF）

MPLS トラフィック エンジニアリングおよび拡張機能の制約事項

- MPLS TE 高速再ルーティングはサポートされていません。

- MPLS TE は、1 つの IGP プロセスまたはインスタンスのみをサポートします。複数の IGP プロセスまたはインスタンスはサポートされず、複数の IGP プロセスまたはインスタンスでは MPLS TE を設定できません。
- MPLS TE 機能では、番号なし IP アドレスリンクを介したルーティングおよびシグナリングはサポートされていません。このため、このようなリンク上には、この機能を設定しないでください。
- 明示パスを指定するとき、転送アドレス（トラフィックを次のルータに転送するインターフェイスのアドレス）をネクストホップアドレスとして指定すると、明示パスが使用されない場合があります。転送アドレスを使用すると、そのエントリをパス計算のルーズホップとして扱うことができます。受信アドレス（送信側ルータからのトラフィックを受信するインターフェイスのアドレス）をネクストホップアドレスとして使用することを推奨します。

次の例では、スイッチ S3 からスイッチ S1 にトラフィックが送信されます。スイッチ S1 とルータ S2 の間で a,b および x,y とマーク付けされたパスはパラレルパスです。

```
S1(a)----(b)S2(c)--(d)S3
  (x)----(y)
```

転送アドレス（アドレス d および b）を使用して S3 から S1 への明示パスを設定すると、トンネルは明示パスの代わりにパラレルパス（x,y）を介してトラフィックを再ルーティングする場合があります。明示パスがトンネルによって必ず使用されるようにするには、次の例に示すように、 **next-address** コマンドの一部として受信アドレスを指定します。

```
ip explicit-path name path1
  next-address (c)
  next-address (a)
```

MPLS トラフィック エンジニアリングおよび拡張機能について

続くセクションでは、MPLS TE および拡張機能について説明します。

MPLS トラフィック エンジニアリングおよび拡張機能の概要

MPLS は、レイヤ 2 テクノロジーとレイヤ 3 テクノロジーを統合したものです。従来のレイヤ 2 機能をレイヤ 3 で使用可能にすることで、MPLS はトラフィック エンジニアリングを可能にしています。したがって、レイヤ 2 ネットワークの上にレイヤ 3 ネットワークを重ねることによってのみ可能になる機能を、1 層のネットワーク内で実現できます。

トラフィック エンジニアリングは、サービスプロバイダーと ISP バックボーンに不可欠です。このようなバックボーンは、伝送容量の大量使用をサポートしている必要があります。また、リンク障害やノード障害に耐えられるように、復元力が高いネットワークである必要があります。

MPLS トラフィック エンジニアリングによって、統合型のトラフィック エンジニアリングが可能になります。MPLS を使用すると、レイヤ 3 にトラフィック エンジニアリング機能が統合され、バックボーンの容量とトポロジによる制約を前提に、IP トラフィックのルーティングが最適化されます。

MPLS トラフィック エンジニアリングでは、次の機能がサポートされています。

- パケットを適切なトラフィックフローに自動的にマッピングするように、IS-IS や OSPF などの標準的な内部ゲートウェイプロトコル（IGP）を拡張します。
- トランスポート トラフィックは MPLS 転送を使用してネットワーク全体に伝送されます。
- ネットワーク全体のトラフィックフローのルートを決定します。その決定は、トラフィックフローに必要なリソースおよびネットワークで使用可能なリソースに基づいています。
- 制約ベースルーティングを使用します。制約ベースルーティングでは、トラフィックフローのパスはそのフローのリソース要件（制約）を満たす最短パスになります。MPLS トラフィック エンジニアリングでは、トラフィック フローには、帯域幅要件、メディア要件、プライオリティ（他のフローのプライオリティと比較）などがあります。
- トポロジが変更されたことによって発生した新しい制約に対応することによって、リンク障害またはノード障害から回復します。
- マルチホップ ラベルスイッチド パス（LSP）を通過する MPLS 転送を使用して、パケットを転送します。
- 次のようなバックボーントポロジで LSP のルーティングおよびシグナリング機能を使用します。
  - そのバックボーントポロジと使用可能なリソースを認識している。
  - バックボーン上の LSP のルートを決定するとき、リンク帯域幅とトラフィックフローのサイズが考慮される。
  - 複数のプライマリパスがオフラインで事前に計算されている場合も、障害に対するバックボーンの復元力を高めるダイナミック適応メカニズムが備えられている。
  - IGP（IS-IS または OSPF）最短パス優先（SPF）計算の拡張機能が備えられており、どのトラフィックをどの LSP を介して送信するかが自動的に計算される。

MPLS トラフィック エンジニアリングの利点

WAN 接続は、ISP 予算において高価な項目です。トラフィック エンジニアリングにより、ISP はネットワーク トラフィックをルーティングして、スループットと遅延の観点でユーザに最善のサービスを提供できるようになります。トラフィック エンジニアリングでは、サービス プロバイダーの効率を高めることによって、ネットワークのコストを削減します。

現在、一部の ISP は、オーバーレイ モデルを基礎としてサービスを提供しています。このモデルでは、送信施設はレイヤ 2 スイッチングによって管理されます。ルータはフル メッシュの仮想トポロジだけを認識し、ほとんどの宛先が 1 ホップ離れて出現します。明示的なレイヤ 2 転

送レイヤを使用する場合、トラフィックが使用可能な帯域幅を使用する方法を正確に制御できます。ただし、オーバーレイモデルには、数多くのデメリットがあります。MPLSトラフィックエンジニアリングでは、個別のネットワークを稼働させることも、スケーラブルでない完全メッシュのルータ相互接続を使用することもなく、オーバーレイ モデルのトラフィック エンジニアリングの利点が得られます。

MPLS トラフィック エンジニアリングのしくみ

MPLS TE では、RSVP を使用して、バックボーン上で LSP を自動的に確立および維持します。LSP で使用されるパスは、LSP リソース要件とネットワーク リソース（帯域幅など）によって決まります。

使用可能なリソースは、リンクステートベースの IGP に対する拡張機能を使用してフラッディングされます。

トラフィック エンジニアリング トンネルは、必要なリソースと使用可能なリソースの調和に基づいて LSP ヘッドで計算されます（制約ベースルーティング）。IGP は、これらの LSP にトラフィックを自動的にルーティングします。通常、MPLS TE バックボーンを通過するパケットは、入力ポイントと出力ポイントを接続する単一の LSP 上を伝送されます。

MPLS トラフィック エンジニアリングは、次の Cisco IOS メカニズムに基づいて構築されています。

- IP トンネル インターフェイス

  レイヤ 2 の観点では、MPLS トンネル インターフェイスは LSP のヘッドを表します。これは、帯域幅要件、メディア要件、プライオリティなどの一連のリソース要件を使用して設定されます。

  レイヤ 3 の観点では、LSP トンネル インターフェイスはトンネル宛先への単一方向仮想リンクのヘッドエンドです。

- MPLS トラフィック エンジニアリング パス計算モジュール

  この計算モジュールは LSP ヘッドで動作します。このモジュールは、LSP で使用するパスを決定します。パス計算では、フラッディングされたトポロジおよびリソース情報を含むリンクステート データベースが使用されます。

- トラフィック エンジニアリング拡張を備えた RSVP

  RSVP は各 LSP ホップで動作し、計算されたパスに基づいて LSP のシグナリングおよび維持のために使用されます。

- MPLS トラフィック エンジニアリング リンク管理モジュール

  このモジュールは、各 LSP ホップで動作します。RSVP シグナリングメッセージに対するリンクコールアドミッションを実行し、フラッディングされるトポロジおよびリソース情報のブックキーピングを行います。

- リンクステート IGP（トラフィック エンジニアリング拡張機能を備えた IS-IS または OSPF）

これらのIGPは、リンク管理モジュールからトポロジおよびリソース情報をグローバルにフラッディングするために使用されます。

- リンクステート IGP（IS-IS または OSPF）で使用される SPF 計算の拡張

  IGPは、トンネル宛先に基づいて適切なLSPトンネルにトラフィックを自動的にルーティングします。また、スタティック ルートを使用して、LSPトンネルにトラフィックを誘導することもできます。

- ラベル スイッチング フォワーディング

  この転送メカニズムは、レイヤ 2 と類似の機能をルータに提供し、RSVP シグナリングによって確立された LSP の複数のホップを経由してトラフィックを誘導できるようにします。

バックボーンのエンジニアリングを行う方法の1つは、すべての入力デバイスからすべての出力デバイスまでのトンネルのメッシュを定義することです。MPLS TE パス計算モジュールおよびシグナリングモジュールは、これらのトンネルのLSPで使用されるパスを、リソースの可用性とネットワークの動的な状態に基づいて決定します。入力デバイスで動作するIGPは、どの出力デバイスにどのトラフィックを送信するかを決定し、入力から出力へのトンネルにそのトラフィックを誘導します。

入力デバイスから出力デバイスへのフローが大きいため、単一のリンクに収まらなくなる可能性があります。また、このフローは単一のトンネルでは伝送できません。こうしたシナリオでは、特定の入力および出力の間に複数のトンネルを設定し、それらの間でフローの負荷が分担されるようにすることができます。

トンネルへのトラフィックのマッピング

このセクションでは、トラフィックがトンネルにどのようにマッピングされるかについて説明します。従来のホップバイホップ リンクステート ルーティング プロトコルが MPLS TE 機能とどのように相互作用するかについて説明します。このセクションでは、最短パス優先（SPF）アルゴリズム（Dijkstra アルゴリズムとも呼ばれることもある）がどのように拡張されるかについて説明します。この拡張により、リンクステートIGPは、MPLSトラフィックエンジニアリングが確立するトンネルを介してトラフィックを自動的に転送できます。

統合IS-ISまたはOSPFなどのリンクステートプロトコルでは、SPFアルゴリズムを使用して、ネットワーク内のヘッドエンドノードからすべてのノードへの最短パスツリーを計算します。ルーティング テーブルは、この最短パス ツリーを基に作成されます。ルーティング テーブルには、宛先と先頭ホップに関する情報のセットが順番に格納されています。ルータで通常のホップバイホップルーティングが実行されている場合、最初のホップはルータに接続された物理インターフェイス上に存在します。

新しいトラフィック エンジニアリング アルゴリズムでは、ネットワーク内の 1 つまたは複数のノードへの明示ルートを計算します。送信元ルータは、これらの明示ルートを論理インターフェイスとして認識します。このマニュアルの中では、これらの明示ルートはLSPによって表され、トラフィック エンジニアリング トンネル（TE トンネル）と呼ばれます。

次の各項では、リンクステートIGPがこれらのショートカットをどのように使用し、これらのTE トンネルを指すルートをルーティング テーブルにどのようにインストールするかについて説明します。これらのトンネルは明示ルートを使用します。TEトンネルで使用されるパスは、トンネルのヘッドエンドルータによって制御されます。エラーがない場合、TEトンネルはループしないことが保証されていますが、ルータが TE トンネルの使用方法に同意している必要があります。このようにしない場合、トラフィックは複数のトンネルでループする可能性があります。

MPLS TE トンネルの明示パスを指定する場合、明示パス内にネクストホップ ルータのリンク アドレスまたはノード アドレスを指定できます。リンク アドレスとノード アドレスを混在させて指定することもできます。リンク アドレスとノード アドレスを混在させて指定する場合でも制約はありません。

新しいテクノロジーへの IS-IS ネットワークの移行

RFC 1142 で規定されている IS-IS には、MPLS TE の拡張が含まれています。IS-IS 上で MPLS トラフィック エンジニアリングを実行したり、これらの他の拡張を利用したりするには、この新しいテクノロジーに IS-IS ネットワークを移行する必要があります。このセクションでは、これらの拡張について説明します。ここでは、既存のIS-IS ネットワークを標準的なISO 10589 プロトコルからRFC 1142 で規定されているIS-IS のバージョンに移行する 2 つの方法を示します。既存の IS-IS ネットワーク上で MPLS TE を実行するには、RFC 1142 で規定されている IS-IS のバージョンへの移行が必要です。ただし、OSPF 上で MPLS TE を実行する場合は、同様のネットワークの移行は不要です。

IS-IS ルーティング プロトコルの拡張

IS-IS ルーティング プロトコルの拡張は、次の目的に使用できます。

- リンク メトリックの 6 ビット制限を削除します。
- エリア間 IP ルートを許可します。
- IS-IS でトラフィック エンジニアリング用に異なる種類の情報を伝送できるようにします。今後、さらに拡張が必要になる場合があります。

これらの目的に役立つように、次の 2 つの新しいタイプ、長さ、値（TLV）オブジェクトが定義されています。

- TLV 22 はリンク（厳密には隣接）を表します。これは、ISO 10589（TLV 2）の「IS ネイバー オプション」と同じ目的に役立ちます。
- TLV 135 は到達可能な IP プレフィックスを表します。これは、RFC 1195（TLV 128 および 130）の IP ネイバー オプションに似ています。

（注） これら 2 つの TLV（22 および 135）は簡略化して「新スタイルの TLV」と呼ばれます。TLV 2、128、および 130 は「旧スタイルの TLV」と呼ばれます。

新しい TLV には両方とも固定長部分があり、それに任意のサブ TLV が続きます。これらの新しい TLV のメトリック領域は 6 ビットから 24 または 32 ビットに拡張されています。サブ TLV では、リンクおよびプレフィックスに新しいプロパティを追加できます。トラフィック エンジニアリングは、この機能を使用してリンクに新しいプロパティを追加できる最初のテクノロジーです。

IS-IS ネットワークを新しいテクノロジーに移行するためのソリューション 1

旧スタイルの TLV を新スタイルの TLV に移行する場合、同じ情報を 2 回（旧スタイルの TLV で 1 回と新スタイルの TLV で 1 回）アドバタイズできます。この操作により、すべてのデバイスでアドバタイズ内容が認識されます。

このアプローチを使用した場合、次の 3 つのデメリットがあります。

- LSP のサイズ：移行中、LSP は元のサイズの約 2 倍に増大します。このことは、大規模な LSP データベース（LSPDB）を使用するネットワークで問題になる場合があります。次の理由により、LSP データベースが大きくなる場合があります。
  - 多数のデバイスがあるため LSP が増大する。
  - ルータごとに、数多くのネイバーまたは IP プレフィックスが存在する。多くの情報をアドバタイズするデバイスにより、LSP がフラグメント化されます。
- 予測不可能な結果：大規模なネットワークでは、このアプローチによって予測不可能な結果が生じることがあります。大規模なネットワークを移行する場合、LSP フラッディングおよび SPF スケーリングに関する制限が強制されます。
- あいまいさ：デバイスが旧スタイルの TLV と新スタイルの TLV で異なる情報を検出すると、デバイスでどのような処理を実行するかが不明確になる可能性があります。
  - ネットワークが不安定になる何らかの特別な状況が発生することがある。このとき、いずれかの実装がどの程度状況に対応できるかを試してはならない。
  - トラフィック エンジニアリング拡張機能によって、LSP が頻繁に再フラッディングされるようになる可能性がある。

このような問題の大部分は、次の項目を使用して簡単に解決できます。

- LSP 内の旧スタイルの TLV と新スタイルの TLV のすべての情報
- 最も小さいリンクメトリックを持つ隣接（隣接が複数回アドバタイズされる場合）

同じ情報を 2 回アドバタイズする主な利点として、新スタイルの TLV がネットワーク内のすべてのデバイスによって認識される前に、ネットワーク管理者が新スタイルの TLV を使用できることを挙げることができます。

ソリューション 1 での移行アクション

旧スタイルの TLV を使用する IS-IS を新スタイルの TLV に移行する場合は、次のアクションを実行します。

- すべてのデバイスが古いソフトウェアを実行している場合、旧スタイルの TLV だけをアドバタイズおよび使用します。
- 一部のデバイスを新しいソフトウェアにアップグレードします。
- 旧スタイルの TLV と新スタイルの TLV の両方をアドバタイズするように、新しいソフトウェアを使用する一部のデバイスを設定します。これらのルータでは、両方のスタイルの TLV を受け入れます。旧スタイルの TLV だけを引き続きアドバタイズおよび使用するように、（古いソフトウェアを使用する）他のデバイスを設定します。
- ネットワークの一部でトラフィック エンジニアリングをテストします。
- ネットワーク全体を移行する必要がある場合は、両方のスタイルの TLV をアドバタイズして受け入れるように、残りのすべてのデバイスをアップグレードおよび設定します。
- 新スタイルの TLV だけをアドバタイズして受け入れるように、すべてのデバイスを設定します。
- 63 よりも大きいメトリックを設定します。

IS-IS ネットワークを新しいテクノロジーに移行するためのソリューション 2

デバイスは同時に 1 つのスタイルの TLV だけをアドバタイズしますが、移行時には両方のスタイルの TLV を認識できます。このアプローチには、主に次の 2 つの利点があります。

- LSP は、移行時にほぼ同じサイズになります。
- 1 つの LSP 内で同じ情報が 2 回アドバタイズされると、あいまいさがなくなります。

この方法は、より広いメトリックを使用する（つまり、IS-IS を実行しているルータで新スタイルの TLV だけを生成して受け入れるようにする）ように、ネットワーク全体（またはエリア全体）を移行する場合に役立ちます。

この方法のデメリットとして、いずれかのデバイスが新スタイルの TLV のアドバタイズを開始するには、すべてのデバイスが新スタイルの TLV を認識している必要があることを挙げることができます。この方法は、2つめの問題、つまり、ネットワーク管理者がトラフィック エンジニアリング用に新スタイルの TLV を使用し、一部のデバイスが旧スタイルの TLV だけを認識できる場合の問題の解決には役立ちません。

2 つめのソリューションでの移行アクション

2 つめのソリューションを使用する場合は、次のアクションを実行できます。

- すべてのデバイスが古いソフトウェアを実行している場合、旧スタイルの TLV だけをアドバタイズおよび使用します。
- すべてのデバイスを新しいソフトウェアにアップグレードします。
- 旧スタイルの TLV をアドバタイズするが、両方のスタイルの TLV を受け入れるように、すべてのデバイスを 1 つずつ設定します。
- 新スタイルの TLV をアドバタイズするが、両方のスタイルの TLV を受け入れるように、すべてのデバイスを 1 つずつ設定します。
- 新スタイルの TLV だけをアドバタイズして受け入れるように、すべてのデバイスを 1 つずつ設定します。
- 63 よりも大きいメトリックを設定します。

TLV コンフィギュレーション コマンド

metric-style コマンドを使用して、デバイスに受け入れられる TLV のタイプを設定できます。デバイスが IS-IS コンフィギュレーション モードの場合、**metric-style** コマンドで次のキーワードを設定できます。

- **metric-style narrow** ：デバイスが旧スタイルの TLV だけを生成して受け入れるようにします。
- **metric-style transition** ：デバイスが旧スタイルと新スタイル両方の TLV を生成して受け入れるようにします。
- **metric-style wide** ：デバイスが新スタイルの TLV だけを生成して受け入れるようにします。

metric-style コマンドを使用する場合、次の移行スキームのいずれかを使用できます。

- narrow > transition > wide
- narrow > narrow transition > wide transition > wide

Cisco IOS XE ソフトウェアでの実装

Cisco IOS XE では両方の移行ソリューションを導入できます。それぞれの環境に適したソリューションを選択してください。テストネットワークにはソリューション 1 が適しています（「IS-IS ネットワークを新しいテクノロジーに移行するためのソリューション 1 （7 ページ）」を参照）。どちらのソリューションでも完全移行を行えますが、手順と設定はソリューション 1 の方が簡単です。移行中に LSP データベースのサイズが急激に増加する恐れがある大規模なネットワークでは、ソリューション 2 を使用してください（「IS-IS ネットワークを新しいテクノロジーに移行するためのソリューション 2 （8 ページ）」を参照）。

MPLS トラフィック エンジニアリングおよび拡張機能の設定方法

続くセクションでは、MPLS トラフィック エンジニアリングおよび拡張機能の設定手順について説明します。

トンネルをサポートするためのデバイスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip cef** 例： `Device(config)# ip cef`	標準的なシスコ エクスプレス フォワーディング動作をイネーブルにします。
ステップ 4	**mpls traffic-eng tunnels** 例： `Device(config)# mpls traffic-eng tunnels`	デバイスで MPLS トラフィック エンジニアリング トンネルをイネーブルにします。
ステップ 5	**exit** 例： `Device(config)# exit`	特権 EXEC モードに戻ります。

RSVP ベースのトンネル シグナリングおよび IGP フラッディングをサポートするためのインターフェイスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type slot / subslot / port* [*subinterface-number*] 例： `Device(config)# interface Port-channel 114`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**mpls traffic-eng tunnels** 例： `Device(config-if)# mpls traffic-eng tunnels`	インターフェイスで MPLS トラフィック エンジニアリング トンネルをイネーブルにします。
ステップ 5	**ip rsvp bandwidth** *bandwidth* 例： `Device(config-if)# ip rsvp bandwidth 1000`	インターフェイスで RSVP をイネーブルにし、予約する帯域幅の量を指定します。
ステップ 6	**exit** 例： `Device(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 7	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

MPLS トラフィック エンジニアリング用の IS-IS の設定

（注） MPLS トラフィック エンジニアリングは、1 つの IGP プロセスまたはインスタンスのみをサポートします。複数の IGP プロセスまたはインスタンスはサポートされていません。MPLS トラフィック エンジニアリングを複数の IGP プロセスまたはインスタンスで設定することはできません。

MPLS トラフィック エンジニアリング用に IS-IS を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**router isis** 例： `Device(config)# router isis`	IS-IS ルーティングをイネーブルにし、IS-IS プロセスを指定します。デバイスはコンフィギュレーション モードになります。
ステップ 4	**mpls traffic-eng level** 例： `Device(config-router)# mpls traffic-eng level-1`	IS-IS レベル 1 で MPLS トラフィック エンジニアリングをオンにします。
ステップ 5	**mpls traffic-eng level** 例： `Device(config-router)# mpls traffic-eng level-2`	IS-IS レベル 2 で MPLS トラフィック エンジニアリングをオンにします。
ステップ 6	**mpls traffic-eng router-id** *type number* 例： `Device(config-router)# mpls traffic-eng router-id loopback 0`	ノードのトラフィック エンジニアリング ルータ識別子が、インターフェイス loopback0 に関連付けられている IP アドレスになるように指定します。

	コマンドまたはアクション	目的
ステップ 7	**metric-style wide** 例： `Device(config-router)# metric-style wide`	新スタイルのタイプ、長さ、値（TLV）オブジェクトだけを生成して受け入れるようにルータを設定します。

MPLS トラフィック エンジニアリング用の OSPF の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**router ospf** *process-id* 例： `Device(config)# router ospf 200`	IP 用の OSPF ルーティング プロセスを設定し、ルータ コンフィギュレーション モードを開始します。 • *process-id* 引数の値は、OSPF ルーティング プロセスの内部で使用される識別パラメータです。ローカルで割り当てられ、任意の正の整数を使用できます。OSPF ルーティング プロセスごとに固有の値を割り当てます。
ステップ 4	**mpls traffic-eng area** *number* 例： `Device(config-router)# mpls traffic-eng area 0`	指定された OSPF エリアに対して MPLS TE をオンにします。
ステップ 5	**mpls traffic-eng router-id loopback0** 例： `Device(config-router)# mpls traffic-eng router-id loopback0`	ノードの TE ルータ識別子が、インターフェイス loopback0 に関連付けられている IP アドレスになるように指定します。

	コマンドまたはアクション	目的
ステップ 6	**exit** 例： `Device(config-router)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**exit** 例： `Device(config)# exit`	特権 EXEC モードに戻ります。

MPLS トラフィック エンジニアリング トンネルの設定

MPLS TE トンネルの優先明示パスを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface tunnel** *number* 例： `Device(config)# interface Tunnel0`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。 • *number* 引数はトンネルの番号です。
ステップ 4	**ip unnumbered** *type number* 例： `Device(config-if)# ip unnumbered loopback0`	明示的な IP アドレスをインターフェイスに割り当てずにインターフェイス上の IP 処理をイネーブルにします。 • *type* 引数および *number* 引数では、ルータに IP アドレスが割り当てられている別のインターフェイスのタイプと番号を指定します。番号付けされていない別のインターフェイスは指定できません。

	コマンドまたはアクション	目的
		• MPLS トラフィック エンジニアリング トンネル インターフェイスは単一方向リンクを表すため、番号なしにする必要があります。
ステップ 5	**tunnel destination** *ip-address* 例： `Device(config-if)# tunnel destination 192.168.4.4`	トンネル インターフェイスの宛先を指定します。 • *ip-address* 引数では、宛先デバイスの MPLS トラフィック エンジニアリング ルータ ID を指定する必要があります。
ステップ 6	**tunnel mode mpls traffic-eng** 例： `Device(config-if)# tunnel mode mpls traffic-eng`	トンネル カプセル化モードを MPLS トラフィック エンジニアリングに設定します。
ステップ 7	**tunnel mpls traffic-eng bandwidth** *bandwidth* 例： `Device(config-if)# tunnel mpls traffic-eng bandwidth 250`	MPLS トラフィック エンジニアリング トンネルの帯域幅を設定します。 • *bandwidth* 引数は、MPLS トラフィック エンジニアリング トンネルで確保する kbps 単位の数値です。範囲は 1 ～ 4294967295 です。 （注） トンネルに自動帯域幅が設定されている場合は、**tunnel mpls traffic-eng bandwidth** コマンドを使用して、トンネルの初期帯域幅を設定します。
ステップ 8	**tunnel mpls traffic-eng path-option** *number* {**dynamic** | **explicit** {**name** *path-name* | **identifier** *path-number*}} [**lockdown**] 例： `Device(config-if)# tunnel mpls traffic-eng path-option 10 explicit identifier 321`	指定した IP 明示パス、またはトラフィック エンジニアリング トポロジ データベースからダイナミックに計算されたパスを使用するように、トンネルを設定します。 • *number* 引数は、このパス オプションの優先度です。複数のパス オプションを設定する場合、より低い数値のオプションが優先されます。有効値は 1 ～ 1000 です。

	コマンドまたはアクション	目的
		• **dynamic** キーワードは、LSP のパスがダイナミックに計算されることを示します。 • **explicit** キーワードは、LSP のパスが IP 明示パスの場合に指定します。 • **name** *path-name* のキーワードと引数のペアは、トンネルがこのオプションで使用する IP 明示パスのパス名です。 • **identifier** *path-number* のキーワードと引数のペアは、トンネルがこのオプションで使用する IP 明示パスのパス番号です。有効な範囲は 1 ～ 65535 です。 • **lockdown** キーワードは、LSP を再最適化できないようにする場合に指定します。 （注） 明示パスが現在使用可能でない場合は、ダイナミックパスが使用されます。
ステップ 9	**exit** 例： `Device(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。

IGP で使用できる MPLS トラフィック エンジニアリング トンネルの設定

IGP で使用できる MPLS TE トンネルを設定するには、次の手順を実行します。

手順

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

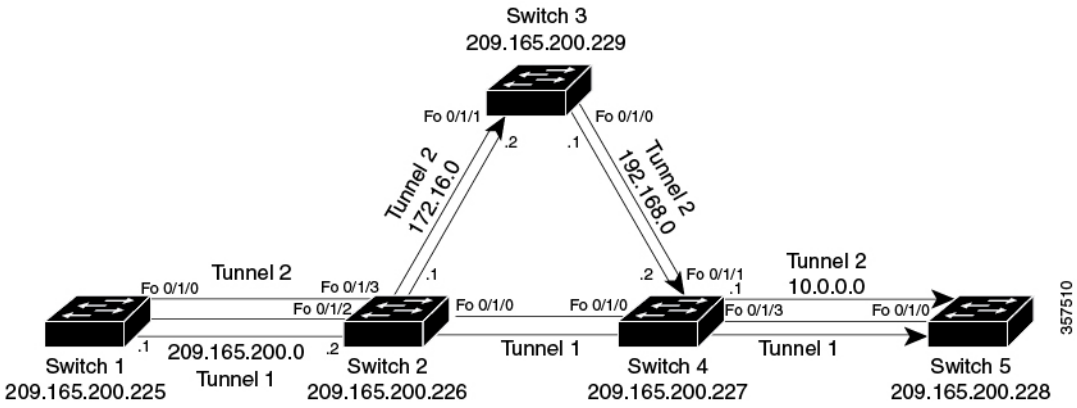
	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface tunnel** *number* 例： `Device(config)# interface tunnel10`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip unnumbered** *type number* 例： `Device(config-if)# ip unnumbered loopback 0`	トンネル インターフェイスに IP アドレスを割り当てます。 MPLS トラフィック エンジニアリング トンネル インターフェイスは単一方向リンクを表すため、番号なしにする必要があります。
ステップ 5	**tunnel destination** *ip-address* 例： `Device(config-if)# tunnel destination 10.20.1.1`	トンネルの宛先を指定します。 *ip-address* キーワードは、ホスト宛先の IP アドレス（ドット付き 10 進表記）です。
ステップ 6	**tunnel mode mpls traffic-eng** 例： `Device(config-if)# tunnel mode mpls traffic-eng`	トンネル カプセル化モードを MPLS トラフィック エンジニアリングに設定します。
ステップ 7	**tunnel mpls traffic-eng bandwidth** *bandwidth* 例： `Device(config-if)# tunnel mpls traffic-eng bandwidth 1000`	MPLS トラフィック エンジニアリング トンネルの帯域幅を設定します。
ステップ 8	**tunnel mpls traffic-eng path-option** *number* {**dynamic** | **explicit** {**name** *path-name*} | **identifier** *path-number*} [**lockdown**] 例： `Device(config-if)# tunnel mpls traffic-eng path-option 1 explicit identifier 1`	指定した IP 明示パス、またはトラフィック エンジニアリング トポロジ データベースからダイナミックに計算されたパスを使用するように、トンネルを設定します。 明示パスが現在使用できない場合は、ダイナミック パスが使用されます。

	コマンドまたはアクション	目的
ステップ 9	**exit** 例： `Device(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。

MPLS トラフィック エンジニアリングおよび拡張機能の設定例

次の図は、MPLS トポロジの例を示しています。この例では、ポイントツーポイントの発信インターフェイスを指定しています。続く各セクションでは、MPLS トラフィック エンジニアリング、および図 3 に示す基本的なトンネル設定を実装するときに入力するコンフィギュレーション コマンドの例を示します。

図 *1: MPLS* トラフィック エンジニアリング トンネルの設定例



例：IS-IS を使用した MPLS トラフィック エンジニアリングの設定

次に、MPLS TE を設定し、IS-IS ルーティングをイネーブルにするときに入力するコマンドの例を示します（図 1 を参照）。

（注） ネットワークのトラフィック エンジニアリング対象部分にあるすべてのルータで次のコマンドを入力します。

デバイス 1：MPLS トラフィック エンジニアリングの設定

MPLS トラフィック エンジニアリングを設定するには、次のコマンドを入力します。

```
ip cef
mpls traffic-eng tunnels
interface loopback 0
ip address 10.0.0.0 255.255.255.254
ip router isis
interface Fo 1/0/0
ip address 209.165.200.1 255.255.0.0
ip router isis
mpls traffic-eng tunnels
ip rsvp bandwidth 1000
```

デバイス 1：IS-IS 設定

IS-IS ルーティングをイネーブルにするには、次のコマンドを入力します。

```
router isis
network 47.0000.0011.0011.00
is-type level-1
metric-style wide
mpls traffic-eng router-id loopback0
mpls traffic-eng level-1
```

例：OSPF を使用した MPLS トラフィック エンジニアリングの設定

次に、MPLS トラフィック エンジニアリングを設定し、OSPF ルーティングをイネーブルにするときに入力するコマンドの例を示します（図 1 を参照）。

（注） ネットワークのトラフィック エンジニアリング対象部分にあるすべてのルータで次のコマンドを入力します。

デバイス 1：MPLS トラフィック エンジニアリングの設定

MPLS トラフィック エンジニアリングを設定するには、次のコマンドを入力します。

```
ip cef
mpls traffic-eng tunnels
interface loopback 0
ip address 209.165.200.225 255.255.255.255
interface Fo 1/0/0
ip address 209.165.200.1 255.255.0.0
mpls traffic-eng tunnels
  ip rsvp bandwidth 1000
```

デバイス 1：OSPF 設定

OSPF をイネーブルにするには、次のコマンドを入力します。

```
router ospf 0
network 209.165.200.0.0.0.255.255 area 0
mpls traffic-eng router-id Loopback0
mpls traffic-eng area 0
```

例：MPLS トラフィック エンジニアリング トンネルの設定

次に、ダイナミック パス トンネルとそのトンネルの明示パスを設定する例を示します。MPLS トラフィック エンジニアリング トンネルを設定する前に、指定のルータ（この場合、ルータ 1）で適切なグローバルコマンドおよびインターフェイスコマンドを入力します。

デバイス 1：ダイナミックパストンネルの設定

ダイナミックパスを使用するようにトンネルを設定するには、次のコマンドを入力します。

```
interface tunnel1
ip unnumbered loopback 0
tunnel destination 209.165.200.228
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng bandwidth 100
tunnel mpls traffic-eng priority 1 1
tunnel mpls traffic-eng path-option 1 dynamic
```

デバイス 1：ダイナミックパストンネルの確認

トンネルがアップ状態になっていることを確認するには、次のコマンドを入力します。

```
show mpls traffic-eng tunnels
show ip interface tunnel1
```

デバイス 1：明示パスの設定

明示パスを設定するには、次のコマンドを入力します。

```
ip explicit-path identifier 1
next-address 209.165.200.1
next-address 172.16.0.1
next-address 192.168.0.1
next-address 10.0.0.1
```

デバイス 1：明示パストンネルの設定

明示パスを使用するようにトンネルを設定するには、次のコマンドを入力します。

```
interface tunnel2
ip unnumbered loopback 0
tunnel destination 209.165.200.228
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng bandwidth 100
tunnel mpls traffic-eng priority 1 1
tunnel mpls traffic-eng path-option 1 explicit identifier 1
```

デバイス 1：明示パストンネルの確認

トンネルがアップ状態になっていることを確認するには、次のコマンドを入力します。

```
show mpls traffic-eng tunnels
show ip interface tunnel2
```

例：トンネル経由の拡張 SPF ルーティングの設定

ここでは、IGP での拡張 SPF 計算でトンネルが考慮されるようにして、適切なネットワークプレフィックスに対するトンネル経由のルートをインストールするコマンドを示します。

デバイス 1：IGP 拡張 SPF による考慮の設定

IGP が拡張最短パス優先（SPF）計算でトンネル（トンネルがアップ状態の場合）を使用するように指定するには、次のコマンドを入力します。

```
interface tunnel1
tunnel mpls traffic-eng autoroute announce
```

デバイス 1：ルートとトラフィックの確認

トンネルがアップ状態になっており、トラフィックがトンネル経由でルーティングされていることを確認するには、次のコマンドを入力します。

```
#show mpls traffic-eng tunnels tu12001 brief
Signalling Summary:
LSP Tunnels Process: running
Passive LSP Listener: running
RSVP Process: running
Forwarding: enabled
auto-tunnel:
p2p Disabled (0), id-range:62336-64335

Periodic reoptimization: every 3600 seconds, next in 694 seconds
Periodic FRR Promotion: Not Running
Periodic auto-bw collection: every 300 seconds, next in 94 seconds
SR tunnel max label push: 2 primary path labels (2 repair path labels)
TUNNEL NAME DESTINATION UP IF DOWN IF STATE/PROT
tu12001 2.2.2.2 - Po114 up/up
```

その他の参考資料

ここでは、MPLS トラフィック エンジニアリングおよび拡張機能に関する関連資料について説明します。

関連資料

関連項目	マニュアル タイトル
IS-IS コマンド	『*Cisco IOS IP Routing Protocols Command Reference*』
OSPF コマンド	『*Cisco IOS IP Routing Protocols Command Reference*』
MPLS TE コマンド	『*Cisco IOS Multiprotocol Label Switching Command Reference*』
RSVP コマンド	『*Cisco IOS Quality of Service Solutions Command Reference*』

MIB

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

RFC

RFC	タイトル
1142	『*IS-IS*』
1195	『*Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*』
2205	『*Resource ReSerVation Protocol (RSVP)*』
2328	『*OSPF Version 2*』
2370	『*The OSPF Opaque LSA Option*』

シスコのテクニカル サポート

説明	リンク
シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。 お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service（Field Notice からアクセス）、Cisco Technical Services Newsletter、Really Simple Syndication（RSS）フィードなどの各種サービスに加入できます。 シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。	http://www.cisco.com/en/US/support/index.html

MPLS トラフィック エンジニアリングおよび拡張機能の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Bengaluru 17.6.1	MPLS トラフィック エンジニアリングおよび拡張機能	マルチプロトコル ラベル スイッチング（MPLS）には、レイヤ 2 テクノロジーとレイヤ 3テクノロジーが統合されています。従来のレイヤ 2 機能をレイヤ 3 で使用可能にすることで、MPLS はトラフィック エンジニアリングを可能にしています。したがって、1層のネットワーク内で、今までレイヤ 2 ネットワークの上にレイヤ 3 ネットワークを重ねることによって初めて実現できていた機能を提案できます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator にアクセスするには、https://cfnng.cisco.com/にアクセスします。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。