MPLS 組み込み管理および MIB コンフィギュレーション ガイド

初版：2012 年 11 月 05 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。

目 次

第 1 章

最初にお読みください

Cisco IOS XE 16 に関する重要な情報

有効な 2 つのリリースとしての Cisco IOS XE リリース 3.7.0E（Catalyst スイッチ用）および Cisco IOS XE リリース 3.17S（アクセスおよびエッジ ルーティング用）が、1 つのバージョンの統合されたリリース（Cisco IOS XE 16）へと展開（マージ）されています。これにより、スイッチングおよびルーティング ポートフォリオの広範なアクセスおよびエッジ製品が盛り込まれた 1 つのリリースが実現しました。

（注） 技術構成ガイドの機能情報の表に、機能の導入時期を記載しています。他のプラットフォームがその機能をサポートした時期については、記載があるものも、ないものもあります。特定の機能が使用しているプラットフォームでサポートされているかどうかを判断するには、製品のランディング ページに掲載された技術構成ガイドを参照してください。技術的構成ガイドが製品のランディング ページに表示される場合は、その機能がお使いのプラットフォームでサポートされていることを示します。

第 2 章

インターフェイス MIB への MPLS の機能拡張

このマニュアルでは、MPLS レイヤをサポートするための既存のインターフェイス MIB（RFC 2233）へのマルチプロトコル ラベル スイッチング（MPLS）機能拡張について説明します。このレイヤは、MPLS に固有のカウンタと統計情報を提供します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

インターフェイス MIB への MPLS の機能拡張の前提条件

- ラベル スイッチング ルータ（LSR）上に簡易ネットワーク管理プロトコル（SNMP）をインストールしてイネーブルにする必要があります。
- LSR で MPLS をイネーブルにする必要があります。
- インターフェイス上で MPLS IP をイネーブルにするか、インターフェイス上で MPLS トラフィック エンジニアリング（TE）トンネルをイネーブルにする必要があります。

インターフェイス MIB への MPLS の機能拡張の制約事項

- MPLS レイヤのリンク アップおよびリンク ダウン トラップは、このリリースではサポートされません。
- このリリースでは、SNMP SET コマンドを使用した書き込み機能は MPLS レイヤに対してサポートされません。
- 廃棄やマルチキャストなどの一部のカウンタは、基礎となる物理層で増加します。したがって、MPLS レイヤには到達しないため、0 と等しくなります。
- インターフェイス MIB の MPLS レイヤ インターフェイスの大容量カウンタには 64 ビットのカウンタ データが含まれています。以前のバージョンでは、大容量カウンタには 32 ビットのカウンタ データが表示されました。

次の MIB オブジェクトが影響を受けます。

-
  - ifHCInOctets
  - ifHCOutOctets
  - ifHCInUcastPkts
  - ifHCOutUcastPkts

64 ビット値が値 232 よりも小さい場合、32 ビット値と 64 ビット値は同一です。

カウンタが増加して 232 よりも大きい値になったあとは、両方のカウンタが異なります。64 ビット値は次の式で計算されます。

X * (232) + Y

値は次のとおりです。

-
  - X は、32 ビット カウンタが循環した回数です。
  - Y は、循環が発生したあとのカウンタの残余値です。Y 値は 32 ビット値と等しくなります。

大容量カウンタ値を32ビット値と比較した場合、カウンタ値が等しくない期間があります。カウンタが32ビットハードウェアカウンタをポーリングして正しいカウンタ値を計算しているとき、64 ビット値の算出は 32 ビット値よりも遅くなります。ポーリングと計算の間隔においては、次の大容量カウンタ値のカウンタが一致しないことがあります。

- • ifInOctets
  • ifOutOctets
  • ifInUcastPkts
  • ifOutUcastPkts

矛盾する値は、トラフィックがインターフェイスを絶えず流れ、MIB ウォークが実行される場合に発生することがあります。32ビット値は、その時点で正確です。64ビット値は、生成にポーリング計算が必要なため若干遅れます。インターフェイス上でのトラフィックの流れが停止し、ポーリング期間が経過したあと、2 つのカウンタは同一の正しい値になります。

遅延時間は次の要因に依存します。

- • インターフェイス MIB で使用されるポーリング間隔。ポーリング間隔の時間が短いほど、値が正確になります。
  • インターフェイス MIB のサイズ。MIB が大きいとウォークに時間がかかり、その時点で見つかった値に影響を及ぼす場合があります。
  • 64 ビット値の生成に必要な計算の回数。MPLS 対応インターフェイスの数により、計算が必要な 64 ビット カウンタ値の数が増えます。

インターフェイス MIB への MPLS の機能拡張に関する情報

インターフェイス MIB への MPLS の機能拡張の機能設計

インターフェイス MIB（IF MIB）は、SNMP ベースでインターフェイスを管理します。IF MIB の各エントリは、インデックス作成、統計情報、および基礎となる物理インターフェイス、サブインターフェイス、および Cisco ソフトウェア内に存在するレイヤ 2 プロトコル間のスタッキング関係を確立します。

機能拡張では、MPLS レイヤが IF MIB に、インターフェイス上の MPLS としてカプセル化されるトラフィックの統計情報を提供するレイヤ 2 プロトコルとして追加されます。この構造では、MPLS カプセル化トラフィック カウンタや MPLS 最大伝送単位（MTU）などの MPLS 固有のデータは基礎となる物理または仮想インターフェイスの最上位に存在し、MPLS 以外のデータからの分離を可能にします。

機能拡張によって、インデックス作成、統計情報、およびifStackTableを使用したスタッキング関係の表示も可能になります。MPLS レイヤ インターフェイスは、MPLS トラフィックを実際に転送している基礎となる物理または仮想インターフェイスの上にスタックされます。MPLS トラフィック エンジニアリング トンネルは、これらの MPLS レイヤの上にスタックされます。

IF MIB では、複数のタイプのインターフェイスがサポートされます。MPLS カプセル化トラフィックのプロトコル統計情報を提供する仮想インターフェイスが追加されました。このインターフェイスは、ファスト イーサネット（fe0/1/0）や ATM（at1/1.1）などの Cisco インターフェイスまたはサブインターフェイスの上にスタックされます。

Cisco ソフトウェアは、インターフェイス コンフィギュレーション モードで **mpls ip** コマンドを発行して MPLS が有効になっている場合に、MPLS をサポートできる各インターフェイスの上に、対応する MPLS レイヤを作成します。

インターフェイス コンフィギュレーション モードで **mpls traffic-eng tunnels** コマンドを使用して MPLS TE を有効にした場合は、インターフェイス レイヤも作成できます。

（注） MPLS IP または MPLS TE をイネーブルにするには、これらのコマンドをグローバル コンフィギュレーション モードで発行する必要もあります。

IF MIB エントリは、インターフェイス上で MPLS IP または MPLS TE トンネルをイネーブルにした場合に作成されます。MPLS IP と MPLS TE の両方をディセーブルにすると、エントリが削除されます。

ifStackTable オブジェクト

次の表に、ifStackTable オブジェクトの定義を示します。

表 *1*：*ifStackTable* オブジェクトと定義

オブジェクト	定義
ifStackHigherLayer	関係の上位サブレイヤに対応する ifIndex の値。つまり、ifStackLowerLayer の対応するインスタンスによって識別されるサブレイヤの最上位で実行されるサブレイヤです。 （注） インデックス オブジェクトは、MIB ウォークでアクセスできません。この値は、ifStackTable 内の各オブジェクトのオブジェクト識別子（OID）の一部です。

オブジェクト	定義
ifStackLowerLayer	関係の下位サブレイヤに対応する ifIndex の値。つまり、ifStackHigherLayer の対応するインスタンスによって識別されるサブレイヤの下で実行されるサブレイヤです。 （注） インデックス オブジェクトは、MIB ウォークでアクセスできません。この値は、ifStackTable 内の各オブジェクトの OID の一部です。
ifStackStatus	ifStackTable の行を作成および削除するために使用します。MPLS の場合ステータスは常に active(1) です。

ifRcvAddressTable オブジェクト

次の表に、ifRcvAddressTable オブジェクトの定義を示します。

（注） MPLS レイヤのエントリは、ifRcvAddressTable には表示されません。

表 *2*：*ifRcvAddressTable* オブジェクトおよび説明

オブジェクト	定義
ifRcvAddressAddress	システムがこのエントリのインターフェイスでパケットとフレームを受け入れるアドレス。 （注） インデックス オブジェクトは、MIB ウォークでアクセスできません。この値は、ifRcvAddressTable 内の各オブジェクトの OID の一部です。
ifRcvAddressStatus	ifRcvAddressTable の行を作成および削除するために使用します。
ifRcvAddressType	ifRcvAddressTable 内の各エントリに使用されるストレージのタイプ。

インターフェイス MIB のスカラー オブジェクト

IF MIB では、次のスカラー オブジェクトがサポートされます。

- ifStackLastChange：インターフェイス スタック全体が最後に変更された時点の sysUpTime の値。インターフェイス スタックの変更とは、ifStackStatus のいずれかのインスタンスの値の作成、削除、または変更として定義されます。ローカル ネットワーク管理サブシステムの前回の初期化以降にインターフェイス スタックが変更されていない場合、このオブジェクトには 0 値が含まれます。
- ifTableLastChange：ifTable のエントリが最後に作成または削除された時点での sysUpTime 値。ローカル ネットワーク管理サブシステムの前回の初期化以降にエントリの数が変更されていない場合、このオブジェクトには 0 値が含まれます。

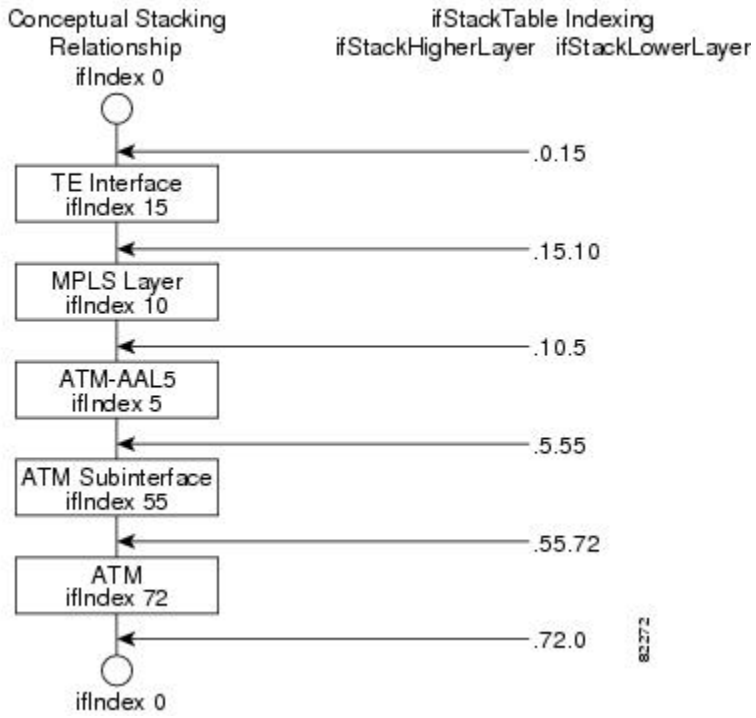
MPLS レイヤ インターフェイスのスタッキング関係

IF MIB 内の ifStackTable は、ifTable のエントリとして表されるインターフェイスとサブインターフェイス間の概念的なスタッキング関係を提供します。

ifStackTable は、リンクされたリストのようにインデックスが作成されます。各エントリは、2 つのインターフェイス間の関係を示し、上位と下位のインターフェイスの ifIndex を提供します。エントリがチェーンして、スタッキング関係全体を示します。各エントリは、スタックの最上位および最下位で 0 の ifIndex でスタックが終了するまで次々にリンクします。たとえば、以下の図では、インデックス .10.5 は ifIndex 10 が ifIndex 5 の上にスタックされていることを示します。ス

タックの最上位と最下位には、0 個のエントリがあります。この図では、インデックス .0.15 および .72.0 がそれぞれスタックの最上位と最下位になっています。

図 *1*：*ifStackTable* 内の *ATM* スタッキング関係の例



以下の表に、上記の図に示されているレイヤ関係の ifStackTable のインデックスを記載します。

（注） この表のエントリの順序は、SNMP 順序付けのルールに従わなければならない MIB ウォークとは異なる場合があります。

表 *3*：レイヤ関係

レイヤ関係（降順）	ifStackHigherLayer/ifStackLowerLayer
最上位レイヤとしての TE インターフェイス	.0.15
MPLS レイヤの上にスタックされた TE インターフェイス	.15.10
ATM-AAL5 の上にスタックされた MPLS レイヤ	.10.5

レイヤ関係（降順）	ifStackHigherLayer/ifStackLowerLayer
ATM サブインターフェイスの上にスタックされた ATM-AAL5 レイヤ	.5.55
ATM の上にスタックされた ATM サブインターフェイス	.55.72
最下位レイヤとしての ATM	.72.0

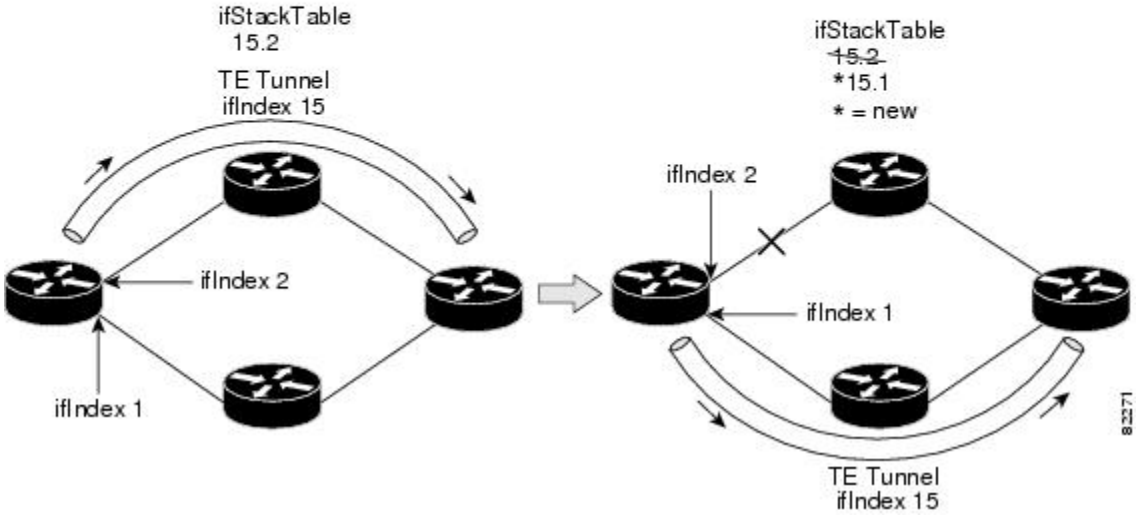
トラフィック エンジニアリング トンネルのスタッキング関係

MPLS TE トンネルは、Cisco ソフトウェアおよび IF MIB では仮想インターフェイスとして表現されます。正しくシグナルされている場合、TE トンネルはトラフィックを物理インターフェイス上の MPLS 経由で渡します。このプロセスは、TE トンネルを基礎となるインターフェイスにスタックされている MPLS レイヤ上にスタックするよう指示します。

TE トンネルは、さまざまなエラーまたはネットワーク条件に応じてパスを変更することもできます。これらの変更は、RSVP-TE シグナリング プロトコルを使用して行われます。変更が発生した場合、トンネルは別の MPLS インターフェイスに切り替えることができます。シグナリング パスが存在しない場合は、パスが選択されないため、MPLS インターフェイスは使用されません。

TE トンネルは IF MIB ifTable エントリとして表されるため、ifStackTable には、TE トンネルに対応するエントリも含まれます。TE トンネルが正常にシグナルされた場合、ifStackTable には、トンネル インターフェイスと 1 つの MPLS インターフェイス間のリンクも含まれます。TE トンネルに対応するシグナルされたパスがないこともあるため、TE トンネルの ifStackTable エントリに対応する下位レイヤがない場合もあります。この場合、下位レイヤ変数には値 0 が含まれます。

以下の図に、ルーティングされる前の TE トンネル（左）とルーティングされた後の TE トンネル（右）および ifStackTable に対する影響を示しています。ifIndex 2 が失敗した場合、TE トンネルは ifIndex1 経由でリルートされ、15.2 エントリが ifStackTable から削除されて 15.1 エントリが追加されます。



MPLS ラベル スイッチング ルータ MIB の機能拡張

MPLS-LSR-MIB テーブル内のすべての ifIndex 参照は、基礎となる物理または仮想インターフェイスの ifIndex から MPLS レイヤの ifIndex に変更されました。

次の表に、具体的な変更内容を示します。

表 *4*：機能拡張された *MPLS-LSR-MIB ifIndex* オブジェクト

テーブル	**ifIndex**
MPLS インターフェイス コンフィギュレーション テーブル（mplsInterfaceConfTable）	mplsInterfaceConfIndex
MPLS 着信セグメント テーブル（mplsInSegmentTable）	mplsInSegmentIfIndex
MPLS 相互接続テーブル（mplsXCTable）	mplsInSegmentIfIndex
MPLS 発信セグメント テーブル（mplsOutSegmentTable）	mplsOutSegmentIfIndex

mplsInterfaceConfTable の次のオブジェクトが影響を受けます。

- mplsInterfaceOutPackets：MPLS カプセル化出力パケットのみをカウントします。
- mplsInterfaceInPackets：MPLS カプセル化入力パケットのみをカウントします。

インターフェイス MIB への MPLS の機能拡張の利点

改善されたアカウンティング機能

MPLS レイヤを表示すると、非 MPLS カプセル化トラフィック（IP パケットなど）を含まない MPLS カプセル化トラフィック カウンタを確認できます。したがって、カウンタは MPLS 関連統計情報でより役立ちます。

TE トンネル インターフェイス

TE トンネル インターフェイスでは、スタッキング関係は現在使用中の基礎となる MPLS インターフェイスを反映し、TE トンネルで再最適化およびリルートが実行されると、動的に変化します。

MPLS 固有の情報

MPLS レイヤは、次のような MPLS 固有の情報を示します。

- MPLS がイネーブルになっているかどうか
- MPLS カウンタ
- MPLS MTU
- MPLS の動作状態

インターフェイス MIB への MPLS の機能拡張の設定方法

SNMP エージェントのイネーブル化

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro***number*]
5. **end**
6. **write memory**
7. **show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 SNMP の情報が表示されない場合は、次のステップに進みます。 SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro***number*] 例： `Router(config)# snmp-server community public ro`	MPLS ラベル配布プロトコル（LDP）MIB に対して読み取り専用（ro）のコミュニティ ストリングを設定します。 • *string* 引数は、パスワードのように機能し、MPLS ネットワーク内のラベル スイッチング ルータ（LSR）上の SNMP 機能へのアクセスを許可します。 • オプションの **ro** キーワードでは、MPLS LDP MIB 内のオブジェクトへの読み取り専用（ro）アクセスを設定します。
ステップ 5	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**write memory** 例： `Router# write memory`	変更した SNMP 設定をルータの NVRAM に書き込み、SNMP 設定を永続的に保存します。
ステップ 7	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 snmp-server という文が表示される場合は、ルータで SNMP がイネーブルになっています。

	コマンドまたはアクション	目的
		SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。

インターフェイス MIB への MPLS の機能拡張の設定例

インターフェイス MIB への MPLS の機能拡張：例

次に、SNMP エージェントをイネーブルにする例を示します。

```
Router# configure terminal
Router(config)# snmp-server community
```

次の例では、SNMPv1 および SNMPv2C がイネーブルになっています。この設定では、任意の SNMP マネージャがコミュニティ ストリング *public* を使用して、読み取り専用権限ですべてのオブジェクトにアクセスできます。

```
Router(config)# snmp-server community public
```

次の例では、comaccess コミュニティ ストリングを指定するアクセス リスト 4 のメンバに対してすべてのオブジェクトの読み取り専用アクセスを許可しています。その他の SNMP マネージャは、どのオブジェクトにもアクセスできません。

```
Router(config)# snmp-server community comaccess ro 4
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
SNMP コマンド	『*Cisco IOS Network Management Command Reference*』
SNMP コンフィギュレーション	『*Network Management Configuration Guide*』の「*Configuring SNMP Support*」
MPLS トラフィック エンジニアリング MIB（MPLS TE MIB）に対する SNMP エージェント サポートの説明	MPLS トラフィック エンジニアリング（TE）MIB

標準

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

MIB

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

RFC

RFC	タイトル
RFC 1156	『*Management Information Base for Network Management of TCP/IP-based internets*』
RFC 1157	『*A Simple Network Management Protocol (SNMP)*』
RFC 1213	『*Management Information Base for Network Management of TCP/IP-based internets: MIB-II*』
RFC 1229	『*Extensions to the Generic-Interface MIB*』
RFC 2233	『*Interfaces MIB*』

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

インターフェイス MIB への MPLS の機能拡張に関する機能情報

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

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

表 *5*：インターフェイス *MIB* への *MPLS* の機能拡張に関する機能情報

機能名	リリース	機能情報
インターフェイス MIB への MPLS の機能拡張	12.0(23)S 12.3(8)T 12.2(33)SRA 12.2(33)SXH 12.2(33)SB Cisco IOS XE Release 2.1	このマニュアルでは、MPLS レイヤをサポートするための既存のインターフェイス MIB（RFC 2233）へのマルチプロトコルラベルスイッチング（MPLS）機能拡張について説明します。このレイヤは、MPLS に固有のカウンタと統計情報を提供します。 この機能は、Cisco IOS Release 12.0(23)S で導入されました。 この機能は、Cisco IOS Release 12.3(8)T に統合されました。 この機能は、Cisco IOS Release 12.2(33)SRA に統合されました。 この機能は、Cisco IOS Release 12.2(33)SXH に統合されました。 この機能は、Cisco IOS Release 12.2(33)SB に統合されました。 Cisco IOS XE Release 2.1 では、この機能は Cisco ASR 1000 シリーズ アグリゲーション サービスルータに実装されました。 以下のコマンドが導入または変更されました。**snmp-server community**。

用語集

ATM：Asynchronous Transfer Mode（非同期転送モード）。セルリレーの国際規格です。複数のサービスタイプ（音声、ビデオ、データなど）が固定長（53バイト）のセルで転送されます。固定長セルの場合は、ハードウェアでセルを処理できるため、伝送遅延が短縮されます。高速の送信メディア（E3、SONET、T3 など）を利用するには、ATM を指定します。

ATM-AAL5：ATM Adaptation Layer 5（ATM アダプテーション層 5）。ITU-T が推奨する 4 つの AAL の 1 つ。AAL5 は、コネクション型可変ビット レート（VBR）サービスをサポートしており、主に Classical IP over ATM および LAN エミュレーション（LANE）トラフィックの転送に使用されます。AAL5 では、Simple and Efficient AAL（SEAL）を使用し、現在の AAL 推奨のうち最も複雑さが低くなっています。交換における帯域幅オーバーヘッドが低く処理要件が単純なため、帯域幅容量の削減とエラー回復機能が実現されます。

カプセル化：特定のプロトコル ヘッダーにデータをラップすること。たとえば、イーサネット データは、ネットワークで送信される前に、特定のイーサネット ヘッダーでラップされます。また、異種ネットワークをブリッジングする場合は、一方のネットワークからのフレーム全体が、もう一方のネットワークのデータ リンク層プロトコルで使用されるヘッダーに単純に配置されます。

IETF：Internet Engineering Task Force（インターネット技術特別調査委員会）。インターネットおよび IP プロトコル スイートの標準を開発している、80 を超えるワーキング グループで構成される委員会です。

インターフェイス：ISO モデルの隣接レイヤ間の境界。

ラベル：パケットの転送を判断するために使用する短い固定長の識別子。

ラベルスイッチング：ネットワーク層ルーティングアルゴリズムに基づくラベル交換アルゴリズムを使用した IP（またはその他のネットワーク層）パケットの転送を説明するために使用される用語。このようなパケットの転送では、完全一致アルゴリズムが使用され、ラベルが書き換えられます。

LSR：Label Switching Router（ラベル スイッチング ルータ）。各パケット内にカプセル化されている固定長ラベルの値に基づいてマルチプロトコル ラベル スイッチング（MPLS）パケットを転送するデバイスです。

MIB：Management Information Base（管理情報ベース）。簡易ネットワーク管理プロトコル（SNMP）などの、ネットワーク管理プロトコルが使用および維持するネットワーク管理情報のデータベース。MIB オブジェクトの値を変更または検索するには、通常はネットワーク管理システムを介して、SNMP コマンドを使用します。MIB オブジェクトはツリー構造であり、ツリーにはパブリック（標準）ブランチとプライベート（独自）ブランチを含みます。

MPLS：Multiprotocol Label Switching（マルチプロトコル ラベル スイッチング）。ネットワークを介してパケット（フレーム）を転送する方式。ネットワークのエッジにあるルータがラベルをパケット（フレーム）に適用できるようにします。ネットワーク コア内の ATM スイッチまたは既存のルータは、最小限のルックアップ オーバーヘッドでラベルに従ってパケットを切り替えることができます。

MPLS インターフェイス：マルチプロトコル ラベル スイッチング（MPLS）トラフィックが有効になっているインターフェイス。

MTU：Maximum Transmission Unit（最大伝送ユニット）。特定のインターフェイスで処理できる最大パケット サイズ（バイト単位）。

NMS：Network Management System（ネットワーク管理システム）。ネットワークの少なくとも一部分の管理に責任を負うシステム。NMS は、一般的に適度にパワーのある装備の整ったコンピュータで、エンジニアリング ワークステーションなどです。NMS はエージェントと通信して、ネットワーク統計情報やリソースを追跡し続けるのに役立ちます。

OID：Object Identifier（オブジェクト識別子）。値は特定の MIB モジュールで定義されます。イベント MIB では、ユーザまたは NMS が指定されたオブジェクトを監視し、存在、しきい値、および Boolean テストに基づいてイベント トリガーを設定できます。トリガーが起動されると、つまり、オブジェクト上の指定されたテストによって true 値が返されると、イベントが発生します。トリガーを作成するには、ユーザまたはネットワーク管理システム（NMS）がイベント MIB の mteTriggerTable にトリガー エントリを設定します。このトリガー エントリでは、監視するオブジェクトの OID を指定します。各トリガーエントリ タイプについて、対応するテーブル（存在、しきい値、および Boolean テーブル）に、テストの実行に必要な情報が入力されます。トリガーがアクティブ化（起動）されたときにシンプル ネットワーク管理プロトコル（SNMP）Set が実行されるか、通知が目的のホストに送信されるか、またはその両方が行われるように MIB を設定できます。

SNMP：Simple Network Management Protocol（シンプル ネットワーク管理プロトコル）。TCP/IP ネットワークでほぼ独占的に使用されている管理プロトコル。SNMP によって、ネットワーク デバイスを監視および制御し、設定、統計情報収集、パフォーマンス、およびセキュリティを管理する手段が提供されます。

トラフィック エンジニアリング トンネル：トラフィック エンジニアリングに使用されるラベル スイッチド トンネル。このようなトンネルは、通常のレイヤ 3 ルーティング以外の方法で設定します。レイヤ 3 ルーティングでトンネルが使用するパス以外のパスでトラフィックを転送するために使用します。

トラップ：シンプル ネットワーク管理プロトコル（SNMP）エージェントによってネットワーク管理ステーション、コンソール、または端末に送信されるメッセージ。これにより、重大なイベントが発生したことが示されます。トラップは通知要求よりも信頼性が低くなります。これは、トラップの受信時に、受信者が確認応答を送信しないためです。送信側は、トラップが受信されたかどうかを判断できません。

トンネル：2 つのピア間（ルータ間など）のセキュアな通信パス。

第 3 章

MPLS LSP Ping Traceroute for LDP/TE および LSP Ping for VCCV

MPLS LSP Ping/Traceroute for LDP/TE および LSP Ping for VCCV 機能を使用すると、サービス プロバイダーはラベル スイッチド パス（LSP）を監視したり、マルチプロトコル ラベル スイッチング（MPLS）転送の問題を迅速に隔離したりできます。

この機能には、次の機能が含まれます。

- MPLS LSP ping。IPv4 ラベル配布プロトコル（LDP）のプレフィックス、リソース予約プロトコル（RSVP）トラフィック エンジニアリング（TE）、および Any Transport over MPLS（AToM）Forward Equivalence Class（FEC）の LSP 接続をテストします。
- MPLS LSP traceroute。IPv4 LDP プレフィックスと RSVP TE プレフィックスの LSP をトレースします。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV の前提条件

MPLS LSP Ping/Traceroute for LDP/TE および LSP Ping for VCCV 機能を使用する前に、次のことを行う必要があります。

- MPLS ネットワークの基本動作を決定する。次に例を示します。
  - 予想される MPLS Experimental（EXP）の処理。
  - LSP の予想される最大サイズ パケットまたは最大伝送単位（MTU）。
  - トポロジは、予想されるラベル スイッチド パス、および LSP のリンク数。ロード バランシング用のパスなど、ラベル スイッチド パケットのパスをトレースします。
- MPLS と MPLS アプリケーションの使用方法を理解する。次の作業が必要です。
  - LDP の設定方法
  - AToM の概念
- ラベル スイッチング、転送、ロード バランシング

ping mpls コマンドまたは **trace mpls** コマンドを使用する前に、ネットワーク内のすべての受信側ルータが認識できる形式で MPLS エコー パケットを符号化およびデコードするようにルータが設定されていることを確認する必要があります。

MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV の制約事項

- MPLS LSP traceroute を使用して、AToM パケットがたどるパスをトレースすることはできません。MMPLS LSP traceroute は AToM ではサポートされません。（MPLS LSP ping は AToM でサポートされます。）ただし、MPLS LSP traceroute を使用して、AToM によって使用される Interior Gateway Protocol（IGP）LSP をトラブルシューティングすることはできます。
- MPLS LSP ping を使用して、MPLS バーチャル プライベート ネットワーク（VPN）を検証またはトレースすることはできません。
- MPLS LSP traceroute を使用して、存続可能時間（TTL）隠蔽を使用する LSP をトラブルシューティングすることはできません。
- MPLS は、宛先単位およびパケット単位の（ラウンドロビン）ロードバランシングをサポートします。パケット単位のロードバランシングが有効な場合は、MPLS LSP traceroute を使用しないでください。これは、中継ルータでの LSP traceroute が、直接接続されているアップストリーム ルータからの前のエコー応答で提供された情報の整合性をチェックすることになるためです。ラウンドロビンを使用していると、TTL に送られるパケットを特定のルータで期限切れにするような方法で、エコー要求パケットが辿るパスを制御することはできません。そのため、LSP traceroute の実行中に整合性検査が失敗し、整合性検査エラーの戻りコードが返されることがあります。
- プラットフォームでは、MPLS エコー要求パケットに応答できるように、LSP ping と traceroute がサポートされている必要があります。
- MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV 機能がパス全体でイネーブルになっている場合を除き、パス上のいずれかのノードで要求が失敗した場合、応答を受け取ることはできません。
- ネットワーク内に異なるドラフトバージョンを組み合わせて実装する場合は、特定の制限があります。ドラフトのバージョンは、シスコの実装と互換性がある必要があります。LSP ping ドラフトの作成方法により、タイプ、長さ、値（TLV）形式が変更になったため、十分なバージョン情報がなければ、旧バージョンは新バージョンと互換性がない可能性があります。シスコの実装では、これを補うために、特定のバージョンであることを前提として、エコーパケットを符号化およびデコードするように送信側ルータと応答側ルータを設定できるようになっています。
- MPLS LSP traceroute を使用する場合は、ネットワークで TTL 隠蔽を使用しないでください。

MPLS LSP Ping Traceroute for LDP/TE および LSP Ping for VCCV に関する情報

MPLS LSP Ping Traceroute for LDP TE および LSP Ping for VCCV の機能

Internet Control Message Protocol（ICMP）ping および traceroute は、転送が失敗する場合の根本原因の診断によく使用されます。ただし、これらは LSP 障害の特定には適していません。これは、LSP 切断発生時には ICMP パケットを IP 経由で宛先に転送できるためです。

MPLS LSP Ping/Traceroute for LDP/TE および LSP Ping for VCCV 機能は、次の理由で LSP 切断の特定に適しています。

- MPLS エコー要求パケットは、IP 経由で転送できません。これは、IP TTL が 1 に設定され、宛先 IP アドレス フィールドは 127/8 アドレスに設定されるためです。
- チェック対象の FEC は IP 宛先アドレス フィールドに保存されません（ICMP の場合）。

MPLS エコー要求パケットと応答パケットは、LSP をテストします。下流のルータでは、次の 2 つの方法でパケットを受信できます。

- 以前に Internet Engineering Task Force（IETF）インターネット ドラフト『Detecting MPLS Data Plane Failures』（draft-ietf-mpls-lsp-ping-03.txt）に基づいていた MPLS エコー要求とエコー応答のシスコ実装。
- このドキュメントで説明する、IETF RFC 4379『Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures』に基づく機能
  - エコー要求の出力インターフェイス制御
  - エコー要求のトラフィック ペーシング
  - エコー要求のスタック末尾の明示的ヌル ラベル シム
  - エコー要求の request-dsmap 機能
  - Request-fec チェック
  - 深度制限の報告

MPLS LSP ping の動作

MPLS LSP ping では、MPLS エコー要求パケットとエコー応答パケットを使用して LSP を検証します。MPLS LSP ping を使用すると、**pingmpls** コマンドで適切なキーワードと引数を使用することによって、IPv4 LDP、AToM、および IPv4 RSVP FEC を検証できます。

MPLS エコー要求パケットは、検証対象の LSP に関連付けられた適切なラベル スタックを使用してターゲット ルータに送信されます。ラベル スタックを使用すると、パケットは LSP 自体を介して転送されます。

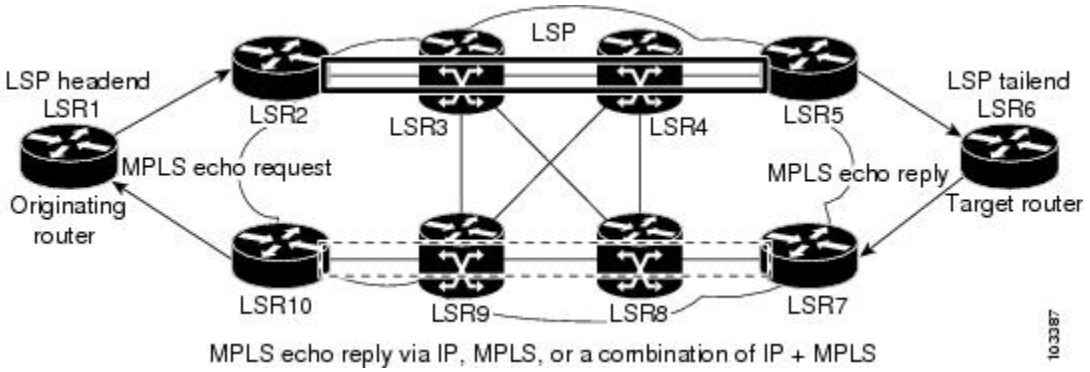
MPLS エコー要求パケットの宛先 IP アドレスは、ラベル スタックの選択に使用されるアドレスとは異なります。宛先 IP アドレスは、127.*x* .*y* .*z* /8 アドレスとして定義されます。127.*x* .*y* .*z* /8 アドレスを使用すると、LSP が切断された場合に IP パケットが宛先に IP スイッチングされるのを防ぐことができます。

MPLS エコー応答は、MPLS エコー要求に応じて送信されます。応答は IP パケットとして送信され、IP、MPLS、または両方のスイッチング タイプの組み合わせを使用して転送されます。MPLS エコー応答パケットの送信元アドレスは、エコー応答を生成するルータから取得されたアドレスです。宛先アドレスは、MPLS エコー要求パケットを送信したルータの送信元アドレスです。

MPLS エコー応答の宛先ポートは、エコー要求の送信元ポートに設定されます。

次の図に、MPLS LSP ping のエコー要求とエコー応答のパスを示します。

図 *2* : *MPLS LSP ping* のエコー要求とエコー応答のパス



LSR1 で LSR6 の FEC に対する MPLS LSP ping 要求を開始すると、次の表に示すような結果になります。

表 *6* : *MPLS LSP ping* の例

ステップ	ルータ	アクション
1.	LSR1	ターゲット ルータ LSR6 の FEC に対する MPLS LSP ping 要求を開始し、MPLS エコー要求を LSR2 に送信します。
2.	LSR2	MPLS エコー要求パケットを受信し、中継ルータ LSR3 と LSR4 を経由して最後から 2 番めのルータ LSR5 に転送します。

ステップ	ルータ	アクション
3.	LSR5	MPLS エコー要求を受信し、MPLS ラベルをポップしてパケットを IP パケットとして LSR6 に転送します。
4.	LSR6	IP パケットを受信し、MPLS エコー要求を処理して、代替ルート経由で MPLS エコー応答を LSR1 に送信します。
5.	LSR7 ～ LSR10	MPLS エコー応答を受信し、送信元ルータ LSR1 に転送します。
6.	LSR1	MPLS エコー要求に対する MPLS エコー応答を受信します。

MPLS LSP Traceroute の動作

MPLS LSP traceroute では、MPLS エコー要求パケットとエコー応答パケットを使用して LSP を検証します。MPLS LSP traceroute を使用すると、**trace mpls** コマンドで適切なキーワードと引数を使用することによって、IPv4 LDP と IPv4 RSVP FEC を検証できます。

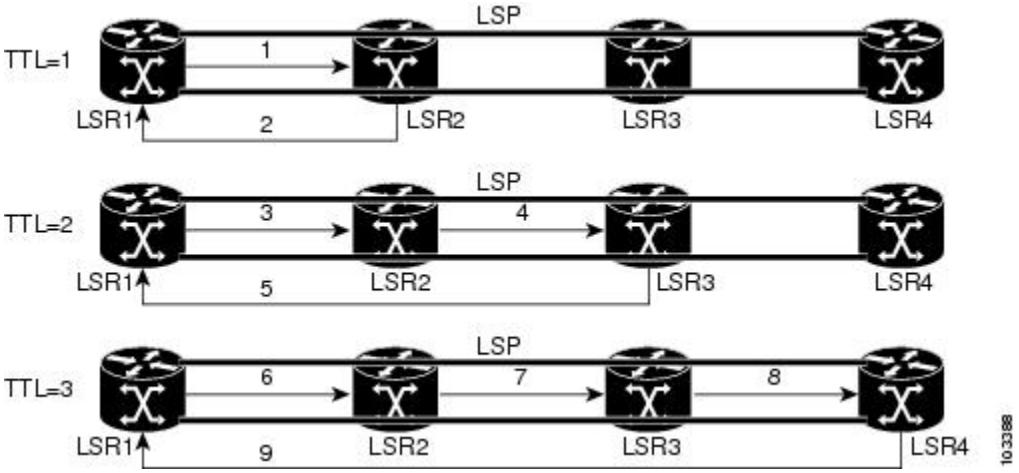
MPLS LSP Traceroute 機能は、TTL 設定を使用して LSP に沿って TTL を強制的に期限切れにします。MPLS LSP Traceroute は、連続した各ホップのダウンストリーム マッピングを検出するために、自身の MPLS エコー要求の TTL 値（TTL = 1、2、3、4）を付加的に増加させます。LSP traceroute の成否は、TTL = 1 のラベル付きパケットの受信時に MPLS エコー要求を処理する中継ルータに依存します。Cisco ルータでは、TTL が期限切れになると、パケットが処理のためにルートプロセッサ（RP）に送信されます。中継ルータは、TTL 期限の切れた MPLS パケットに対し、中継ホップの情報を持つ MPLS エコー応答を戻します。

MPLS エコー応答の宛先ポートは、エコー要求の送信元ポートに設定されます。

（注） ルータがトラフィック エンジニアリング トンネルを通過する IPV4 FEC をトレースすると、中間ルータは、中間ルータで LDP が実行されていない場合に U（到達不能）を返すことがあります。

次の図に、LSR1 から LSR4 までの LSP の MPLS LSP traceroute の例を示します。

図 *3*： ***MPLS LSP Traceroute*** の例



LSR1 から LSR4 の FEC に対する LSP traceroute を入力すると、次の表に示すような結果になります。

表 *7*： ***MPLS LSP Traceroute*** の例

ステップ	ルータ	**MPLS** パケットタイプと説明	ルータのアクション（受信または送信）
1.	LSR1	MPLS エコー要求：ターゲット FEC は LSR4 とダウンストリーム マッピングを指す	• ラベル スタックの TTL を 1 に設定する • 要求を LSR2 に送信する
2.	LSR2	MPLS エコー応答	• TTL = 1 のパケットを受信する • User Datagram Protocol（UDP）パケットを MPLS エコー要求として処理する • ダウンストリーム マッピングを検索し、着信ラベルに基づいて独自のダウンストリーム マッピングを付加して LSR1 に応答する

ステップ	ルータ	MPLS パケットタイプと説明	ルータのアクション（受信または送信）
3.	LSR1	MPLS エコー要求：ターゲット FEC は同じで、LSR2 からのエコー応答で受信したダウンストリーム マッピングを含む	• ラベル スタックの TTL を 2 に設定する • 要求を LSR2 に送信する
4.	LSR2	MPLS エコー要求	• TTL=2 のパケットを受信する • TTL を減らす • エコー要求を LSR3 に転送する
5.	LSR3	MPLS 応答パケット	• TTL=1 のパケットを受信する • UDP パケットを MPLS エコー要求として処理する • ダウンストリーム マッピングを検索し、着信ラベルに基づいて独自のダウンストリーム マッピングを付加して LSR1 に応答する
6.	LSR1	MPLS エコー要求：ターゲット FEC は同じで、LSR3 からのエコー応答で受信したダウンストリーム マッピングを含む	• パケットの TTL を 3 に設定する • 要求を LSR2 に送信する
7.	LSR2	MPLS エコー要求	• TTL=3 のパケットを受信する • TTL を減らす • エコー要求を LSR3 に転送する

ステップ	ルータ	MPLS パケットタイプと説明	ルータのアクション（受信または送信）
8.	LSR3	MPLS エコー要求	• TTL=2 のパケットを受信する • TTL を減らす • エコー要求を LSR4 に転送する
9.	LSR4	MPLS エコー応答	• TTL=1 のパケットを受信する • UDP パケットを MPLS エコー要求として処理する • ダウンストリーム マッピングを検索し、ルータがターゲット FEC の出力ルータであることも確認する • LSR1 に応答する

MPLS LSP ping および MPLS LSP traceroute を使用した MPLS ネットワーク管理

MPLS ネットワークを管理するには、LSP をモニタリングして MPLS 転送の問題を迅速に隔離できる必要があります。そのためには、LSP の動作を評価したり、LSP によるユーザ トラフィックの伝送の失敗を検出したりする方法が必要です。

MPLS LSP ping を使用すると、IPv4 LDP プレフィックス宛てのパケットの転送に使用される LSP や AToM PW FEC を確認できます。MPLS LSP traceroute を使用すると、IPv4 LDP プレフィックス宛てのパケットの伝送に使用される LSP をトレースできます。

MPLS エコー要求は、検証する LSP 経由で送信されます。TTL の期限切れまたは LSP の切断が発生すると、中継ルータはエコー要求を目的の宛先に到達する前に処理します。ルータは説明的な応答コードを含む MPLS エコー応答をエコー要求の送信元に返します。

成功したエコー要求は LSP の出口で処理されます。エコー応答は IP パス、MPLS パス、または両方のパスの組み合わせを経由してエコー要求の送信元に返送されます。

Any Transport over MPLS 仮想回線接続

AToM 仮想回線接続性検証（VCCV）を使用すると、送信元のプロバイダー エッジ（PE）ルータから AToM PW の帯域内で制御パケットを送信できます。伝送は宛先 PE ルータで代行受信され、カスタマー エッジ（CE）ルータには転送されません。この機能により、MPLS LSP ping を使用して AToM 仮想回線（VC）の PW セクションをテストできます。

LSP ping を使用すると、FEC 128 または FEC 129 による AToM VC のセットアップを検証できます。FEC 128 ベースの AToM VC をセットアップするには、シグナリングに LDP を使用するか、2 つのエンドポイントでシグナリング コンポーネントを使用しないで静的な疑似回線設定を使用します。Cisco ソフトウェアでは、MPLS ping の発行中、FEC 128 と FEC 129 の静的な疑似回線は区別されず、同じコマンドが使用されます。

AToM VCCV は次のコンポーネントで構成されます。

- VC ラベルのシグナリング中に AToM VCCV 機能がアドバタイズされるシグナリング対象のコンポーネント
- AToM VC ペイロードが制御パケットとして処理されるスイッチング コンポーネント

AToM VCCV シグナリング

AToM VC セットアップの手順の 1 つは、AToM VC エンドポイント間での VC ラベルと AToM VCCV 機能のシグナリングまたは通信です。各エンドポイントの AToM VCCV ディスポジション機能を通信するために、ルータは IETF インターネット ドラフト『*Pseudo Wire (PW) Virtual Circuit Connection Verification (VCCV)*』（draft-ieft-pwe3-vccv-01）で定義されているオプション パラメータを使用します。

AToM VCCV ディスポジション機能は、次のように分類されます。

- アプリケーション：MPLS LSP ping と ICMP ping は、パケットを制御用に AToM PW の帯域内で送信するために AToM VCCV によってサポートされるアプリケーションです。
- スイッチング モード：AToM VCCV で制御トラフィックとデータ トラフィックを区別するために使用されるスイッチング モードとして、タイプ 1 とタイプ 2 があります。

次の表に、AToM VCCV のタイプ 1 とタイプ 2 のスイッチング モードを示します。

表 *8*：タイプ *1* とタイプ *2* の *AToM VCCV* スイッチング モード

スイッチング モード	説明
タイプ 1	AToM 制御ワードのプロトコル ID（PID）フィールドを使用して、AToM VCCV パケットを識別します。

スイッチング モード	説明
タイプ 2	VC ラベルの上の MPLS ルータ アラート ラベルを使用して、AToM VCCV パケットを識別します。

AToM VCCV スイッチング タイプの選択

Cisco ルータでは、AToM VC 制御チャネルを介して MPLS LSP ping パケットを送信するときに、使用可能な場合は常にタイプ 1 スイッチングを使用します。タイプ 2 スイッチングは、AToM 制御ワードをサポートまたは解釈しない VC タイプと実装に対応します。

以下の表に、AToM VC によってアドバタイズおよび選択される AToM VCCV スイッチング モードを示します。

表 *9*：*AToM VC* によってアドバタイズおよび選択される *AToM VCCV* スイッチング モード

アドバタイズされるタイプ	選択されるタイプ
AToM VCCV はサポートされない	--
タイプ 1 AToM VCCV スイッチング	タイプ 1 AToM VCCV スイッチング
タイプ 2 AToM VCCV スイッチング	タイプ 2 AToM VCCV スイッチング
タイプ 1 およびタイプ 2 AToM VCCV スイッチング	タイプ 1 AToM VCCV スイッチング

AToM VC は、AToM VCCV ディスポジション機能を両方向、つまり送信元ルータ（PE1）から宛先ルータ（PE2）へ、PE2 から PE1 へアドバタイズします。

2 つのエンドポイントの AToM VCCV 機能が異なる場合、AToM VC は異なるスイッチング タイプを使用することがあります。PE1 がタイプ 1 およびタイプ 2 AToM VCCV スイッチングをサポートし、PE2 がタイプ 2 AToM VCCV スイッチングだけをサポートしている場合は、次の 2 とおりの結果になります。

- PE1 から PE2 に送信された LSP ping パケットは、タイプ 2 スイッチングでカプセル化される。
- PE2 から PE1 に送信された LSP ping パケットは、タイプ 1 スイッチングを使用する。

ピアにアドバタイズされた AToM VCCV 機能やピアから受信した AToM VCCV 機能を確認するには、PE ルータで **show mpls l2transport binding** コマンドを入力します。

LSP ping または LSP traceroute を処理するルータから提供される情報

次の表に、LSP ping または LSP traceroute パケットを処理するルータから、要求の成否について送信者に返される文字について説明します。

ping mpls verbose コマンドを入力することでも、MPLS LSP ping 操作の戻りコードを表示できます。

表 *10*：エコー応答の戻りコード

出力コード	エコーの戻りコード	意味
x	0	戻りコードなし。
M	1	不正なエコー要求。
m	2	サポートされていない TLV。
!	3	成功。
F	4	FEC マッピングなし。
D	5	DS マップの不一致。
I	6	不明なアップストリーム インターフェイス インデックス。
U	7	予備。
L	8	ラベル付けされた出力インターフェイス。
B	9	ラベル付けされていない出力インターフェイス。
f	10	FEC の不一致。
N	11	ラベル エントリなし。
P	12	受信インターフェイスのラベル プロトコルなし。
p	13	LSP の終了が不完全。
X	unknown	未定義の戻りコード。

（注） エコーの戻りコード 6 と 7 は、バージョン 3（draft-ieft-mpls-ping-03）でのみ受け入れられます。

IP で MPLS エコー要求パケットが転送されない

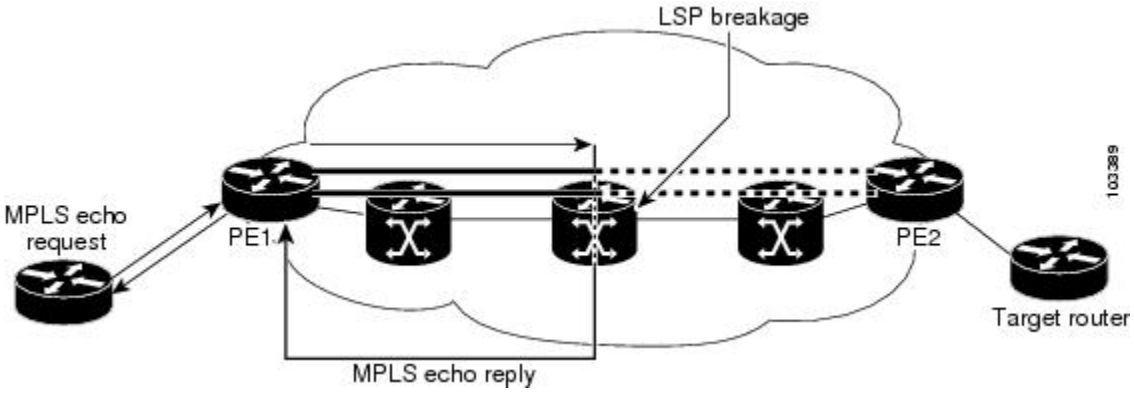
LSP ping 中に送信された MPLS エコー要求パケットは、IP によって転送されません。MPLS エコー要求パケットの IP ヘッダーの宛先アドレス フィールドは 127.*x.y.z*/8 アドレスです。ルータは 127.*x.y.z*/8 アドレスを使用したパケットを転送しません。127.*x.y.z*/8 アドレスは、ローカル ホストのアドレスに対応します。

127.*x .y .z* アドレスを UDP パケットの宛先アドレスとして使用することが重要です。これは、中継ルータが LSP のラベル スイッチングを行わない場合、MPLS エコー要求パケットは、このアドレスをターゲット ルータにすることができないためです。127.*x .y .z* アドレスを使用すると、LSP の切断を検出できます。中継ルータでは、次のことが発生します。

- 中継ルータで LSP の切断が発生した場合、MPLS エコー パケットは転送されず、ルータによって使用されます。
- LSP が切断されていない場合、MPLS エコー パケットはターゲット ルータに到達し、LSP の終点で処理されます。

次の図に、中継ルータが LSP でパケットのラベル スイッチングに失敗した場合の MPLS エコー要求と応答のパスを示します。

図 *4*：中継ルータがパケットのラベル スイッチングに失敗した場合のパス



（注） AToM ペイロードは IP パケットではない可能性があるため、ペイロードには中継ルータで使用可能なフォワーディング情報が格納されません。MPLS VPN パケットは IP パケットですが、MPLS ネットワークのエンドポイントの Virtual Routing and Forwarding（VRF）インスタンスには宛先 IP アドレスだけが重要であるため、MPLS VPN パケットには中継ルータで使用可能なフォワーディング情報が格納されません。

MPLS LSP と ping または traceroute 実装間の互換性

バージョン 3（draft-ietf-mpls-ping-03）よりもあとの LSP ping ドラフトでは、多数の TLV 形式の変更が行われていますが、ドラフトのバージョン同士が必ずしも相互運用するとはかぎりません。

新しいシスコの実装がドラフト バージョン 3 のシスコの実装やシスコ以外の実装と相互運用できるようにするには、グローバル コンフィギュレーション モードを使用して、エコー パケットをドラフト バージョン 3 の実装によって認識される形式でデコードします。

特に設定がなければ、シスコの実装では、IETF の実装がベースにしているバージョンを想定して、エコー要求の符号化とデコードを行います。

TLV バージョンの問題によって発生するエラーが応答ルータから報告されないようにするには、コア内のすべてのルータを設定する必要があります。同じドラフト バージョンで MPLS エコー パケットを符号化およびデコードしてください。たとえば、ネットワークで RFC 4379（シスコ バージョン 4）の実装が実行され、1 つのルータがバージョン 3（シスコ リビジョン 3）にだけ対応している場合は、ネットワーク内のすべてのルータをリビジョン 3 モードで動作するように設定します。

MPLS エコー要求とエコー応答のシスコの実装は、IETF RFC 4379 に基づいています。この RFC よりもあとの IEFT ドラフト（ドラフト 3、4、5、6、および 7）では、TLV 形式に相違があります。エコー パケットでは、ある TLV 形式と別の TLV 形式を区別できないため、これらの相違を識別できません。これらのリリース間の相互運用を可能にするために、**ping mpls** コマンドと **trace mpls** コマンドに **revision** キーワードが追加されました。**revision** キーワードを使用すると、Cisco IOS XE リリースは既存のドラフトの変更と今後の IETF LSP ping ドラフト バージョンからの変更に対応できるようになります。

（注） revision オプションの代わりに、**mpls oam** グローバル コンフィギュレーション コマンドを使用することを推奨します。

（注） cisco.com ではリビジョン 2 をサポートするイメージを入手できません。TLV の符号化とデコードのモードを設定する場合は、バージョン 3 以降をサポートするイメージだけを使用することを推奨します。MPLS マルチパス LSP traceroute を使用するには、シスコ リビジョン 4 以降が必要です。

CiscoVendorExtensions

シスコ バージョン 3（draft-ietf-mpls-ping-03.txt）の実装では、ignore-if-not-understood TLV スペースにベンダー拡張 TLV が定義されています。これは次の目的で使用されます。

- TLV バージョンを追跡する機能を提供する。
- 試験的な応答 TOS 機能を提供する。

最初の機能は、エコー パケットの符号化とデコードの動作を設定するために、グローバル コンフィギュレーション コマンドよりも前に定義されました。設定されたデコード動作は、エコーパケット内の TLV バージョン情報によって上書きされます。グローバル コンフィギュレーション機能の導入後、TLV バージョンにこの TLV を使用する必要はなくなりました。

2 番めの機能は、応答 DSCP を制御します。ドラフト バージョン 8 では、応答 TOS TLV が定義されているため、応答 DSCP を使用する必要はなくなりました。

エコー パケットのデフォルト動作をカスタマイズして MPLS LSP と ping または traceroute 実装間の互換性を有効にします。

エコー応答で特定のサービス クラスを要求するための DSCP オプション

Cisco ソフトウェアに、応答 DiffServ コード ポイント（DSCP）オプションが追加されました。このオプションを使用すると、エコー応答における特定のサービスクラス（CoS）を要求できます。

応答 DSCP オプションは、IETF draft-ietf-mpls-lsp-ping-03.txt の試験モードでサポートされます。シスコは、応答 TOS TLV を使用するのではなく、応答 DSCP オプションのベンダー固有の拡張を実装しました。応答 TOS TLV は、RFC 4379 の **reply dscp** コマンドと同じ目的を果たします。このドラフトは、応答 DSCP を制御するための標準化された方法を示します。

（注） ドラフト バージョン 8 よりも前のバージョンでは、シスコは応答 DSCP オプションをシスコのベンダー拡張 TLV を使用した試験的な機能として実装しました。ルータがドラフトバージョン 3 の実装の MPLS エコー パケットを符号化するように設定されている場合は、ドラフトバージョン 8 で定義された応答 TOS TLV の代わりに、シスコのベンダー拡張 TLV が使用されます。

MPLS LSP ping と LSP Traceroute のエコー要求に対する応答モード

応答モードは、**ping mpls** コマンドまたは **trace mpls** コマンドによって送信された MPLS エコー要求に対する応答ルータの応答方法を制御します。エコー要求パケットには、次の 2 つの応答モードがあります。

- ipv4：IPv4 UDP パケットで応答（デフォルト）

- router-alert：ルータ アラートを含む IPv4 UDP パケットで応答

（注） ipv4 および router-alert 応答モードを互いに組み合わせて使用すると、false negative を防ぐことができます。ipv4 モードで応答を受信できない場合は、router-alert 応答モードでテストを送信すると役に立ちます。両方のモードで失敗する場合は、リターンパスに何か問題があります。唯一考えられる問題は、応答 TOS が正しく設定されていないことです。

IPv4 応答モード

IPv4 パケットは、LSP の完全性を定期的にポーリングする場合に、**ping mpls** コマンドまたは **trace mpls** コマンドで使用される最も一般的な応答モードです。このオプションは、パケットが IP ホップと MPLS ホップのいずれを通過して MPLS エコー要求の送信元に到達するかを明示的に制御するものではありません。**reply mode ipv4** キーワードを使用した場合に、送信元（ヘッドエンド）ルータが MPLS エコー要求に対する応答を受信できないときは、**reply mode router-alert** キーワードを使用します。

router-alert 応答モード

router-alert 応答モードを使用すると、ルータ アラート オプションが IP ヘッダーに追加されます。IP ヘッダーに IP ルータ アラート オプションを含む IP パケット、または最も外側のラベルとしてルータ アラート ラベルを含む MPLS パケットがルータに到達すると、ルータはパケットを処理するためにルート プロセッサ（RP）レベルにパント（リダイレクト）します。これにより、Cisco ルータはパケットが宛先に戻るときに各中間ホップでパケットを処理します。これにより、ハードウェアとラインカード フォワーディングの不整合が回避されます。router-alert 応答モードは、応答がホップバイホップで移動するため、IPv4 モードよりもコストがかかります。処理速度も遅いため、送信元が応答を受信するまで比較的長い時間がかかります。

以下の表に、IP ルータ アラート オプションを含む IP パケットと MPLS パケットがルータ スイッチング パス プロセスによって処理される方法を示します。

表 *11*： パス プロセスによる *IP* および *MPLS* ルータ アラート パケットの処理

着信パケット	通常のスイッチング アクション	プロセス スイッチング アクション	発信パケット
IP パケット：IP ヘッダーにルータ アラート オプションが含まれる	IP ヘッダーにルータ アラート オプションが含まれていると、パケットはプロセス スイッチング パスにパントされる。	パケットをそのまま転送する。	IP パケット：IP ヘッダーにルータ アラート オプションが含まれる
		パケットをそのまま転送する。	MPLS パケット

着信パケット	通常のスイッチング アクション	プロセス スイッチング アクション	発信パケット
MPLSパケット：最も外側のラベルにルータ アラートが含まれる	ルータ アラート ラベルが最も外側のラベルである場合、パケットはプロセス スイッチング パスにパントされる。	最も外側のルータ アラート ラベルを削除し、パケットを IP パケットとして転送する。	IP パケット：IP ヘッダーにルータ アラート オプションが含まれる
		最も外側のルータ アラート ラベルを保持し、MPLS パケットを転送する。	MPLSパケット：最も外側のラベルにルータ アラートが含まれる

LSP の切断

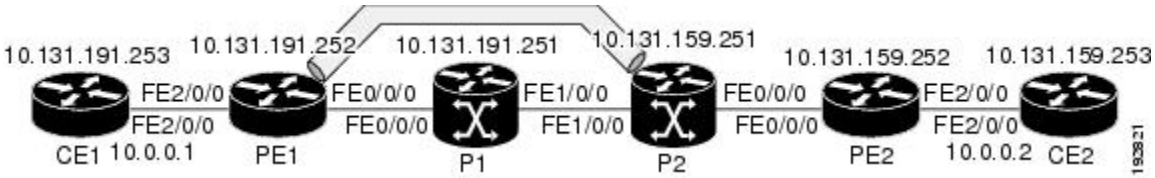
ネットワークで MPLS パケットの転送に問題が発生した場合は、LSP が切断されている場所を特定できます。ここでは、LSP での MTU ディスカバリについて説明します。

最後から 2 番めのホップの非タグ付き出力インターフェイスは、LSP 経由の IP パケットの転送に影響しません。これは、転送判断が最後から 2 番めのホップで着信ラベルを使用して行われるためです。ただし、非タグ付き出力インターフェイスを使用すると、AToM と MPLS VPN のトラフィックが最後から 2 番めのホップでドロップされます。

MPLS LSP ping の実行中、MPLS エコー要求パケットは IP パケット属性が「do not fragment」に設定された状態で送信されます。つまり、パケットの IP ヘッダーに Don't Fragment（DF）ビットが設定されます。これにより、MPLS エコー要求を使用して、フラグメンテーションなしでパケットが LSP を通過できるようにするための MTU をテストできます。

次の図に、LDP によってアドバタイズされたラベルで構成されている 1 つの LSP（PE1 から PE2 まで）のサンプル ネットワークを示します。

図 *5*：*LSP* のサンプル ネットワーク：*LDP* によってアドバタイズされたラベル



MPLS Traceroute 機能を使用して LSP をトレースすることによって、各ホップの最大受信ユニット（MRU）を確認できます。MRU は、LSP 経由で転送できる、ラベル付けされたパケットの最大サイズです。

MPLS LSP Ping Traceroute for LDP/TE および LSP Ping for VCCV の設定方法

MPLS LSP と ping または traceroute 実装間の互換性のイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **mplsoam**
4. **echo revision** {**3** | **4**}
5. **echo vendor-extension**
6. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mplsoam** 例： `Router(config)# mpls oam`	エコー パケットのデフォルト動作をカスタマイズするために、MPLS OAM 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	**echo revision** {**3** | **4**} 例： `Router(config-mpls)# echo revision 4`	エコー パケットのデフォルト値のリビジョン番号を指定します。 • 3：draft-ietf-mpls-ping-03（リビジョン 2） • 4：RFC 4379 準拠（デフォルト）

	コマンドまたはアクション	目的
ステップ 5	**echo vendor-extension** 例： `Router(config-mpls)# echo vendor-extension`	エコー パケットで TLV のシスコ独自の拡張を送信します。
ステップ 6	**exit** 例： `Router(config-mpls)# 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。

MPLS LSP ping と MPLS LSP traceroute を使用した LDP IPv4 FEC の検証

手順の概要

1. **enable**
2. 次のいずれかを実行します。
   - **ping mpls ipv4***destination-address/destination-mask-length* [**repeat***count*] [**exp***exp-bits*] [**verbose**]
   - **trace mpls ipv4***destination-address/destination-mask-length*
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを実行します。 • **ping mpls ipv4***destination-address/destination-mask-length* [**repeat***count*] [**exp***exp-bits*] [**verbose**] • **trace mpls ipv4***destination-address/destination-mask-length*	検証する LDP IPv4 プレフィックスの FEC を選択します。

	コマンドまたはアクション	目的

	例：
	`Router# ping mpls ipv4 10.131.191.252/32 exp 5 repeat 5 verbose`
	例：
	`Router# trace mpls ipv4 10.131.191.252/32`	

ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP ping と MPLS LSP traceroute を使用したレイヤ 2 FEC の検証

手順の概要

1. **enable**
2. **ping mpls pseudowire***ipv4-address***vc-id***vc-id*
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**ping mpls pseudowire***ipv4-address***vc-id***vc-id* 例： `Router# ping mpls pseudowire 10.131.191.252 vc-id 333`	検証するレイヤ 2 FEC を選択します。

	コマンドまたはアクション	目的
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

DSCP を使用した、エコー応答における特定のサービス クラスの要求

手順の概要

1. **enable**
2. 次のいずれかを実行します。
   - **ping mpls** {**ipv4***destination-address/destination-mask-length* | **pseudowire***ipv4-address***vc-id***vc-id*} [**reply dscp***dscp-value*]
   - **trace mpls ipv4***destination-address/destination-mask-length* [**reply dscp***dscp-value*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを実行します。 • **ping mpls** {**ipv4***destination-address/destination-mask-length* | **pseudowire***ipv4-address***vc-id***vc-id*} [**reply dscp***dscp-value*] • **trace mpls ipv4***destination-address/destination-mask-length* [**reply dscp***dscp-value*] 例： `Router# ping mpls ipv4 10.131.191.252/32 reply dscp 50`	エコー応答の DSCP 値を制御します。

	コマンドまたはアクション	目的
	例： `Router# trace mpls ipv4 10.131.191.252/32 reply dscp 50`	
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS エコー要求に対する応答ルータの応答方法の制御

手順の概要

1. **enable**
2. 次のいずれかを実行します。
   - **ping mpls** {**ipv4***destination-address*/*destination-mask-length* | **pseudowire***ipv4-address***vc-id***vc-id*} **reply mode** {**ipv4** | **router-alert**}
   - **trace mpls ipv4***destination-address*/*destination-mask***reply mode** {**ipv4** | **router-alert**}
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを実行します。 • **ping mpls** {**ipv4***destination-address*/*destination-mask-length* | **pseudowire***ipv4-address***vc-id***vc-id*} **reply mode** {**ipv4** | **router-alert**}	MPLS LSP 接続をチェックします。 または パケットが宛先に転送されるときに実際にたどる MPLS LSP ルートを検出します。

	コマンドまたはアクション	目的

	• **trace mpls ipv4***destination-address*/*destination-mask***reply mode** {**ipv4** | **router-alert**}
	
	例：
	`Router# ping mpls ipv4 10.131.191.252/32 reply mode ipv4`
	
	例：
	`Router# trace mpls ipv4 10.131.191.252/32 reply mode router-alert`
		（注） 応答モードを指定するには、**ipv4** キーワードまたは **router-alert** キーワードとともに **reply mode** キーワードを入力する必要があります。

ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP ping を使用したループの検出

MPLS LSP ping 機能を使用する場合、UDP 宛先アドレス範囲、繰り返しオプション、またはスイープ オプションを指定すると、ループが発生する可能性があります。

MPLS LSP ping を使用してループを検出するには、次の手順を実行します。

手順の概要

1. **enable**
2. **ping mpls** {**ipv4***destination-address/destination-mask* [**destination***address-startaddress-endincrement* | [**pseudowire***ipv4-address***vc-id***vc-idaddress-endincrement*]} [**repeat***count*] [**sweep***minimummaximumsize-increment*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**ping mpls** {**ipv4***destination-address/destination-mask* [**destination***address-startaddress-endincrement* | [**pseudowire***ipv4-address***vc-id***vc-idaddress-endincrement*]} [**repeat***count*] [**sweep***minimummaximumsize-increment*] 例： `Router# ping mpls ipv4 10.131.159.251/32 destination 127.0.0.1 127.0.0.2 1 repeat 2 sweep 1450 1475 25`	MPLS LSP 接続をチェックします。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP traceroute を使用したループの検出

MPLS LSP Traceroute 機能を使用する場合、UDP 宛先アドレス範囲オプションと存続可能時間オプションを指定すると、ループが発生する可能性があります。

デフォルトでは、最大 TTL は 30 に設定されます。したがって、traceroute のターゲットに到達しない場合は（LSP に問題がある場合に発生する可能性があります）、traceroute の出力に 30 行が含まれることがあります。LSP に問題が発生した場合は、エントリが重複する可能性があります。トレースが最後に到達するポイントのルータ アドレスは、出力が 30 行になるまで繰り返されます。重複したエントリは無視できます。

手順の概要

1. **enable**
2. **tracemplsipv4***destination-address /destination-mask* [**destination***address-startaddress-endaddressincrement*] [**ttl***maximum-time-to-live*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**tracemplsipv4***destination-address /destination-mask* [**destination***address-startaddress-endaddressincrement*] [**ttl***maximum-time-to-live*] 例： `Router# trace mpls ipv4 10.131.159.251/32 destination 127.0.0.1 127.0.0.3 1 ttl 5`	パケットが宛先に転送されるときにたどる MPLS LSP ルートを検出します。ここに挙げた例は、ループがどのように発生するかを示しています。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

次の作業

暗黙的ヌルとタグ付けされたパケットの追跡

手順の概要

1. **enable**
2. **trace mpls ipv4***destination-address*/*destination-mask*
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**trace mpls ipv4***destination-address*/*destination-mask* 例： `Router# trace mpls ipv4 10.131.159.252/32`	パケットが宛先に転送されるときに実際にたどる MPLS LSP ルートを検出します。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

非タグ付きパケットの追跡

手順の概要

1. **enable**
2. **show mpls forwarding-table***destination-address/destination-mask*
3. **show mpls ldp discovery**
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show mpls forwarding-table***destination-address/destination-mask* 例： `Router# show mpls forwarding-table 10.131.159.252/32`	MPLS ラベル転送情報ベース（LFIB）のコンテンツを表示し、LDP が正しく設定されているかどうかを示します。

	コマンドまたはアクション	目的
ステップ 3	**show mpls ldp discovery** 例： `Router# show mpls ldp discovery`	LDP ディスカバリ プロセスのステータスを表示し、LDP が正しく設定されているかどうかを示します。
ステップ 4	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

パケットを送信できない原因の特定

Q 戻りコードは、パケットを送信できなかったことを意味します。この問題は、処理メモリの不足が原因である場合がありますが、コマンド ラインで入力された FEC 情報に一致する LSP が見つからなかったために発生した可能性があります。

LSP のパスの問題を修正できるように、パケットが転送されなかった原因を特定する必要があります。そのためには、ルーティング情報ベース（RIB）、転送情報ベース（FIB）、ラベル情報ベース（LIB）、および MPLS LFIB を調べます。これらのルーティングまたはフォワーディングベースに FEC のエントリがない場合は、戻りコードが Q になります。

パケットを送信できなった原因を特定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **show ip route** [*ip-address* [**mask**]]
3. **show mpls forwarding-table** [*network* {*mask* | *length*} | **labels***label*[-*label*] | **interface***interface* | **next-hop***address* | **lsp-tunnel** [*tunnel-id*]]
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**show ip route** [*ip-address* [**mask**]] 例： `Router# show ip route 10.0.0.1`	ルーティング テーブルの現在の状態を表示します。 MPLS エコー応答が Q を返すと、ルーティング情報データベースでトラブルシューティングが実行されます。
ステップ 3	**show mpls forwarding-table** [*network* {*mask* | *length*} | **labels***label*[-*label*] | **interface***interface* | **next-hop***address* | **lsp-tunnel** [*tunnel-id*]] 例： `Router# show mpls forwarding-table 10.0.0.1/32`	MPLS LFIB のコンテンツを表示します。MPLS エコー応答が Q を返すと、ラベル情報データベースと MPLS 転送情報データベースでトラブルシューティングが実行されます。
ステップ 4	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

IPv4 LDP LSP でロード バランシングがイネーブルになっている場合の LSP 切断の検出

ICMP ping または trace は、送信元ルータからターゲット ルータまでの 1 本のパスをたどります。送信元ルータからの IP パケットのラウンド ロビン ロード バランシングでは、ターゲット IP アドレスへのさまざまな出力パスを検出します。

MPLS ping と traceroute の場合、ネットワークにターゲット ルータへの複数のパスが存在するときに、Cisco ルータはロード バランシングに IP ヘッダー内の送信元アドレスと宛先アドレスを使用します。MPLS のシスコの実装では、IP ペイロードの宛先アドレスをチェックしてロード バランシングを実行する場合があります（チェックのタイプはプラットフォームによって異なります）。

IPv4 LDP LSP でロード バランシングがイネーブルになっている場合に LSP の切断を検出するには、次の手順を実行します。

手順の概要

1. **enable**
2. **ping mpls ipv4***destination-address/destination-mask-length* [**destination***address-startaddress-endincrement*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**ping mpls ipv4***destination-address/destination-mask-length* [**destination***address-startaddress-endincrement*] 例： `Router# ping mpls ipv4 10.131.159.251/32 destination 127.0.0.1/8`	ロード バランシングのパスをチェックします。 127.*z.y.x* /8 宛先アドレスを入力します。
ステップ 3	**exit** 例： `Router#` **exit**	ユーザ EXEC モードに戻ります。

エコーパケットがルータから発信されるときに経由するインターフェイスの指定

エコー パケットがルータから発信されるときに経由するインターフェイスを制御できます。パス出力情報は、LSP ping と traceroute への入力として使用されます。

エコー要求の出力インターフェイス制御機能を使用すると、LSP の詳細なデバッグや評価を行うパスをエコー パケットが通過することを強制できます。この機能は、PE ルータが MPLS クラウドに接続し、切断されたリンクがある場合に役立ちます。特定のリンクを介してトラフィックを誘導できます。この機能は、ネットワークの問題のトラブルシューティングにも役立ちます。

エコー要求の出力インターフェイスを指定するには、次の手順を実行します。

手順の概要

1. **enable**
2. 次のいずれかのコマンドを入力します。
   - **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} [**output interface***tx-interface*]
   - **trace mpls ipv4***destination-address/destination-mask*
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

ステップ 2	次のいずれかのコマンドを入力します。
	• **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} [**output interface***tx-interface*]
	• **trace mpls ipv4***destination-address/destination-mask*
	
	例：
	`Router# ping mpls ipv4 10.131.159.251/32 output interface fastethernet0/0/0`
	
	例：
	`Router# trace mpls ipv4 10.131.159.251/32 output interface fastethernet0/0/0`
		MPLS LSP 接続をチェックします。
		または
		パケットが宛先に転送されるときに実際にたどる MPLS LSP ルートを検出します。
		（注） この作業では、**outputinterface** キーワードを指定する必要があります。

ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

パケット伝送のペーシング

エコー要求のトラフィック ペーシングを使用すると、受信側ルータがパケットをドロップしないように、パケット伝送をペーシングできます。エコー要求のトラフィック ペーシングを実行するには、次の手順を実行します。

手順の概要

1. **enable**
2. 次のいずれかを実行します。
   - **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} [**interval***ms*]]
   - **trace mpls ipv4***destination-address/destination-mask*
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

ステップ 2	次のいずれかを実行します。
	
	• **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} [**interval***ms*]]
	• **trace mpls ipv4***destination-address/destination-mask*
	
	例：
	
	`Router# ping mpls ipv4 10.131.159.251/32 interval 2`
	
	例：
	
	`Router# trace mpls ipv4 10.131.159.251/32`
		MPLS LSP 接続をチェックします。
		
		または
		
		パケットが宛先に転送されるときにたどる MPLS LSP ルートを検出します。
		
		（注） この作業では、**ping mpls** コマンドを使用する場合、**interval** キーワードを指定する必要があります。

	コマンドまたはアクション	目的
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

エコー要求の request-dsmap を使用した中継ルータに対するダウンストリーム情報の問い合わせ

エコー要求の request-dsmap のトラブルシューティング機能を TTL フラグと組み合わせて使用すると、中継ルータに選択的に問い合わせることができます。エラーがある場合は、以前の各エラーに対して **lsp traceroute** コマンドを入力する必要はありません。失敗したホップにだけ集中できます。

ダウンストリーム マッピングのフラグ フィールドの request-dsmap フラグと非準拠ルータの追跡方法を指定する手順によって、ワイルドカード ダウンストリーム マップ（DSMAP）で任意に MPLS エコー要求パケットの存続可能時間（TTL）を期限切れにすることができます。

ラベルのないエコー要求の DSMAP を受信した場合、送信元に検証する DSMAP がなかったことを意味します。エコー要求の DSMAP TLV のダウンストリーム ルータ ID フィールドが ALLROUTERs アドレス（224.0.0.2）に設定され、ラベルがない場合、送信元ルータは任意に中継ルータに対して DSMAP 情報を問い合わせることができます。

ping mpls コマンドを使用すると、トラブルシューティングやダウンストリーム ルータに対する DSMAP の問い合わせのために、ワイルドカード DSMAP を使用して中継ルータの MPLS エコー要求の TTL を期限切れにすることができます。デフォルトでは、DSMAP には IPv4 ビットマップ ハッシュキーがあります。ハッシュキー 0（なし）を選択することもできます。**ping mpls** コマンドの目的は、送信元ルータが中継ルータのエコー要求の TTL を選択的に期限切れにして、中継ルータにダウンストリーム情報を問い合わせることができるようにすることです。マルチパス（ハッシュキー）タイプも選択できるようにすると、送信元ルータは、マルチパス LSP traceroute の場合と同様に、中継ルータに対してロード バランシング情報を問い合わせることができます。このとき、送信元ルータと各エコー要求の TTL が期限切れになるルータ間を通過するすべての後続ノードに問い合わせる必要はありません。エコー要求は、TTL設定と組み合わせて使用します。これは、エコー要求に使用した LSP の出口にエコー要求が到達しても、応答ルータは DSMAP を返さないためです。

エラーがある場合に失敗したホップにだけ集中できるように、中継ルータにダウンストリーム情報を問い合わせるには、次の手順を実行します。

手順の概要

1. **enable**
2. **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} [**dsmap** [**hashkey** {**none** | **ipv4 bitmap***bitmap-size*}]]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} [**dsmap** [**hashkey** {**none** | **ipv4 bitmap***bitmap-size*}]] 例： `Router# ping mpls ipv4 10.161.251/32 dsmap hashkey ipv4 bitmap 16`	MPLS LSP 接続をチェックします。 （注） この作業では、**dsmap** キーワードと **hashkey** キーワードを指定する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

ルータに対する DSMAP の問い合わせ

ルータは、ソフトウェアまたはハードウェアのフォワーディング レイヤに DSMAP TLV で返す必要がある深度制限を問い合わせることができます。フォワーディングによって値が提供されない場合、デフォルトは 255 です。

深度制限を確認するには、**ping mpls** コマンドで **dsmap** キーワードと **ttl** キーワードを指定します。中継ルータは、DSMAP についての問い合わせを受けます。深度制限は、エコー応答の DSMAP とともに返されます。値が 0 の場合、IP ヘッダーはロード バランシングに使用されています。別の値の場合、IP ヘッダーはその数のラベルまでロード バランスを行います。

ルータに DSMAP を問い合わせるには、次の手順を実行します。

手順の概要

1. **enable**
2. **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} **ttl***time-to-live***dsmap**
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} **ttl***time-to-live***dsmap** 例： `Router# ping mpls ipv4 10.131.159.252/32 ttl 1 dsmap`	MPLS LSP 接続をチェックします。 （注） **ttl** キーワードと **dsmap** キーワードを指定する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

中継ルータによるターゲット FEC スタックの検証の要求

MPLS エコー要求は、特定の LSP をテストします。テスト対象の LSP は、FEC スタックで識別されます。

中継ルータにターゲット FEC スタックの検証を要求するには、送信元ルータから **ping mpls** コマンドと **trace mpls** コマンドに **flags fec** キーワードを入力して V フラグを設定します。デフォルトでは、エコー要求パケットは V フラグが 0 に設定されて送信されます。

中継ルータにターゲット FEC スタックの検証を要求するには、次の手順を実行します。

手順の概要

1. **enable**
2. 次のいずれかを実行します。
   - **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} **flags fec**
   - **trace mpls ipv4***destination-address/destination-mask***flags fec**
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

ステップ 2	次のいずれかを実行します。
	• **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} **flags fec**
	• **trace mpls ipv4***destination-address/destination-mask***flags fec**
	
	例：
	`Router# ping mpls ipv4 10.131.159.252/32 flags fec`
	
	例：
	`Router# trace mpls ipv4 10.131.159.252/32 flags fec`
		MPLS LSP 接続をチェックします。
		または
		パケットが宛先に転送されるときに実際にたどる MPLS LSP ルートを検出します。
		（注） **flags fec** キーワードを入力する必要があります。

ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

LSP ping のイネーブル化による非タグ付きインターフェイスを起因とする LSP 切断の検出

IPv4 FEC を伝送する LSP の MPLS LSP ping と traceroute では、ラベルが要求されていない場合でも、明示的ヌル ラベルを MPLS ラベル スタックに強制的に追加できます。これにより、LSP ping で非タグ付きインターフェイスを起因とする LSP の切断を検出できます。LSP ping は、LSP が MPLS トラフィックを送信できない場合は LSP が動作していると報告しません。

明示的ヌル ラベルが MPLS ラベル スタックに追加されるのは、MPLS エコー要求パケットが LSP ping の宛先に直接接続されている非タグ付きインターフェイスから転送された場合、または MPLS エコー要求パケットの IP TTL 値が 1 に設定されている場合です。

lsp ping コマンドを入力すると、LSP が IP トラフィックを伝送できるかどうかがテストされます。最後から 2 番めのホップの非タグ付き出力インターフェイスで発生したエラーは検出されません。明示的ヌル シムを使用すると、LSP の MPLS トラフィック伝送能力をテストできます。

LSP ping をイネーブルにして、非タグ付きインターフェイスを起因とする LSP の切断を検出するには、次の手順に示すように、**ping mpls** コマンドまたは **trace mpls** コマンドで **force-explicit-null** キーワードを指定します。

手順の概要

1. **enable**
2. 次のいずれかを実行します。
   - **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} **force-explicit-null**
   - **trace mpls ipv4***destination-address/destination-mask***force-explicit-null**
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを実行します。 • **ping mpls** {**ipv4***destination-address/destination-mask* | **pseudowire***ipv4-address***vc-id***vc-id*} **force-explicit-null**	MPLS LSP 接続をチェックします。 または パケットが宛先に転送されるときに実際にたどる MPLS LSP ルートを検出します。

	コマンドまたはアクション	目的

	• **trace mpls ipv4***destination-address/destination-mask***force-explicit-null**
	
	例：
	`Router# ping mpls ipv4 10.131.191.252/32 force-explicit null`
	
	例：
	`Router# trace mpls ipv4 10.131.191.252/32 force-explicit-null`
		（注） **force-explicit-null** キーワードを入力する必要があります。

ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

ピアにアドバタイズされた AToM VCCV 機能やピアから受信した AToM VCCV 機能の表示

ピアにアドバタイズされた AToM VCCV 機能やピアから受信した AToM VCCV 機能を表示するには、次の手順を実行します。

手順の概要

1. **enable**
2. **show mpls l2transport binding**
3. **exit**

手順の詳細

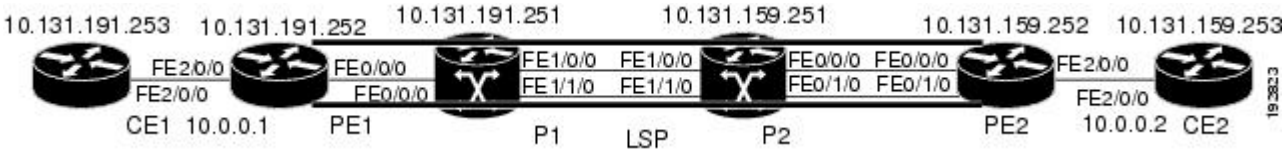
	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**show mpls l2transport binding** 例： `Router# show mpls l2transport binding`	VC ラベルのバインディング情報を表示します。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV の設定例

MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV 機能の例は、以下の図に示すサンプル トポロジに基づいています。

図 *6*：設定例のサンプル トポロジ



ここでは、次の設定例を示します。

MPLS LSP と ping または traceroute 実装間の互換性のイネーブル化：例

次に、RFC 4379 の解釈がシスコとは異なるベンダー実装と相互運用するように MPLS マルチパス LSP traceroute を設定する例を示します。

```
configure terminal
!
mpls oam
 echo revision 4
 no echo vendor-extension
 exit
```

デフォルトのエコー リビジョン番号は 4 です。これは IEFT ドラフト 11 に対応します。

MPLS LSP ping を使用したレイヤ 2 FEC の検証：例

次に、レイヤ 2 FEC を検証する例を示します。

```
Router# ping mpls pseudowire 10.10.10.15 108 vc-id 333

Sending 5, 100-byte MPLS Echos to 10.10.10.15,
      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,
   '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 = 28/32/40 ms PE-802#
```

MPLS LSP ping と MPLS LSP traceroute を使用した LDP IPv4 FEC の検証：例

次に、**ping mpls** コマンドを使用して IPv4 LDP LSP の接続をテストする例を示します。

```
Router# ping mpls ipv4 10.131.191.252/32 repeat 5 exp 5 verbose
Sending 5, 100-byte MPLS Echos to 10.131.191.252, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
!     10.131.191.230, return code 3
!     10.131.191.230, return code 3
!     10.131.191.230, return code 3
!     10.131.191.230, return code 3
!     10.131.191.230, return code 3
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/10
```

DSCPを使用した、エコー応答における特定のサービスクラスの要求：例

次に、DSCP を使用してエコー応答における特定の CoS を要求する例を示します。

```
Router# ping mpls ipv4 10.131.159.252/32 reply dscp 50
 <0-63>   Differentiated services codepoint value
 af11     Match packets with AF11 dscp (001010)
 af12     Match packets with AF12 dscp (001100)
 af13     Match packets with AF13 dscp (001110)
 af21     Match packets with AF21 dscp (010010)
 af22     Match packets with AF22 dscp (010100)
 af23     Match packets with AF23 dscp (010110)
 af31     Match packets with AF31 dscp (011010)
 af32     Match packets with AF32 dscp (011100)
 af33     Match packets with AF33 dscp (011110)
```

```
 af41     Match packets with AF41 dscp (100010)
 af42     Match packets with AF42 dscp (100100)
 af43     Match packets with AF43 dscp (100110)
 cs1      Match packets with CS1(precedence 1) dscp (001000)
 cs2      Match packets with CS2(precedence 2) dscp (010000)
 cs3      Match packets with CS3(precedence 3) dscp (011000)
 cs4      Match packets with CS4(precedence 4) dscp (100000)
 cs5      Match packets with CS5(precedence 5) dscp (101000)
 cs6      Match packets with CS6(precedence 6) dscp (110000)
 cs7      Match packets with CS7(precedence 7) dscp (111000)
 default  Match packets with default dscp (000000)
 ef       Match packets with EF dscp (101110)
```

MPLS エコー要求に対する応答ルータの応答方法の制御：例

次に、ipv4 応答モードを使用して MPLS LSP 接続をチェックする例を示します。

```
Router# ping mpls ipv4 10.131.191.252/32 reply mode ipv4
```

MPLS LSP ping で発生する可能性があるループの防止：例

以下に、**ping mpls** コマンドを使用した場合のループ動作の例を示します。

```
Router# ping mpls
 ipv4
 10.131.159.251/32 destination 127.0.0.1 127.0.0.2 1 repeat 2
sweep 1450 1475 25
Sending 2, [1450..1500]-byte MPLS Echos to 10.131.159.251/32,
     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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
Destination address 127.0.0.1
!
!
Destination address 127.0.0.2
!
!
Destination address 127.0.0.1
!
!
Destination address 127.0.0.2
!
!
```

ping mpls コマンドは、終了アドレスに到達するまで、各宛先アドレスについて、パケットサイズの範囲ごとに送信されます。この例では、宛先アドレス 127.0.0.5 に到達するまでループは同じように続行されます。シーケンスは、その回数が **repeat***count* キーワードと引数で指定した値に到達するまで続行されます。この例では、リピート回数は 2 です。MPLS LSP ping のループ シーケンスは次のようになります。

```
repeat  = 1
  destination address 1 (address-start
)
    for (size from sweep minimum
```

```
 to maximum
, counting by size-increment
)
      send an lsp ping
  destination address 2 (address-start
 +
address-
increment
)
    for (size from sweep minimum
 to maximum
, counting by size-increment
)
      send an lsp ping
  destination address 3 (address-start
 +
address-
increment
 +
address-
increment
)
     for (size from sweep minimum
 to maximum
, counting by size-increment
)
      send an lsp ping
  .
  .
  .
  until destination address = address-end
  .
  .
  .
  until repeat = count 2
```

MPLS LSP traceroute で発生する可能性があるループの防止：例

以下に、**trace mpls** コマンドを使用した場合に発生するループの例を示します。

```
Router# trace mpls ipv4 10.131.159.251/32 destination 127.0.0.1 127.0.0.3 1 ttl 5
Tracing MPLS Label Switched Path to 10.131.159.251/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
Destination address 127.0.0.1
  0 10.131.191.230 MRU 1500 [Labels: 19 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
! 2 10.131.159.225 40 ms
Destination address 127.0.0.2
  0 10.131.191.230 MRU 1500 [Labels: 19 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
! 2 10.131.159.225 40 ms
Destination address 127.0.0.3
  0 10.131.191.230 MRU 1500 [Labels: 19 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
! 2 10.131.159.225 48 ms
```

mpls trace コマンドは、宛先の *end-address* 引数で指定されたアドレスに到達するまで、各宛先アドレスについて、1 から最大 TTL（**ttl***maximum-time-to-live* キーワードと引数）までの TTL ごとに

送信されます。この例では、最大 TTL は 5 で、終了宛先アドレスは 127.0.0.3 です。MPLS LSP traceroute のループ シーケンスは次のようになります。

```
destination address 1 (address-start
)
  for (ttl from 1 to maximum-time-to-live
)
    send an lsp trace
destination address 2 (address-start
 + address-increment
)
  for (ttl from 1 to 5
)
    send an lsp trace
destination address 3 (address-start
 + address-increment
 + address-increment
)
  for (ttl from 1 to
maximum-time-to-live)
    send an lsp trace
.
.
.
until destination address = 4
```

次に、トレース中に IP アドレスが 10.6.1.6 のルータで LSP の問題が発生した場合の例を示します。

```
Router# traceroute mpls ipv4 10.6.7.4/32
Tracing MPLS Label Switched Path to 10.6.7.4/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.6.1.14 MRU 4470 [Labels: 22 Exp: 0]
R 1 10.6.1.5 MRU 4470 [Labels: 21 Exp: 0] 2 ms
R 2 10.6.1.6 4 ms                  <------ Router address repeated for 2nd to 30th TTL.
R 3 10.6.1.6 1 ms
R 4 10.6.1.6 1 ms
R 5 10.6.1.6 3 ms
R 6 10.6.1.6 4 ms
R 7 10.6.1.6 1 ms
R 8 10.6.1.6 2 ms
R 9 10.6.1.6 3 ms
R 10 10.6.1.6 4 ms
R 11 10.6.1.6 1 ms
R 12 10.6.1.6 2 ms
R 13 10.6.1.6 4 ms
R 14 10.6.1.6 5 ms
R 15 10.6.1.6 2 ms
R 16 10.6.1.6 3 ms
R 17 10.6.1.6 4 ms
R 18 10.6.1.6 2 ms
R 19 10.6.1.6 3 ms
R 20 10.6.1.6 4 ms
R 21 10.6.1.6 1 ms
R 22 10.6.1.6 2 ms
R 23 10.6.1.6 3 ms
R 24 10.6.1.6 4 ms
R 25 10.6.1.6 1 ms
R 26 10.6.1.6 3 ms
R 27 10.6.1.6 4 ms
R 28 10.6.1.6 1 ms
R 29 10.6.1.6 2 ms
R 30 10.6.1.6 3 ms                    <------ TTL 30.
```

ネットワーク内の最大ホップ数がわかっている場合は、**trace mpls ttl***maximum-time-to-live* コマンドを使用して、TTL を小さい値に設定できます。次の例では、上記の例と同じ **traceroute** コマンドを使用していますが、TTL が 5 に設定されています。

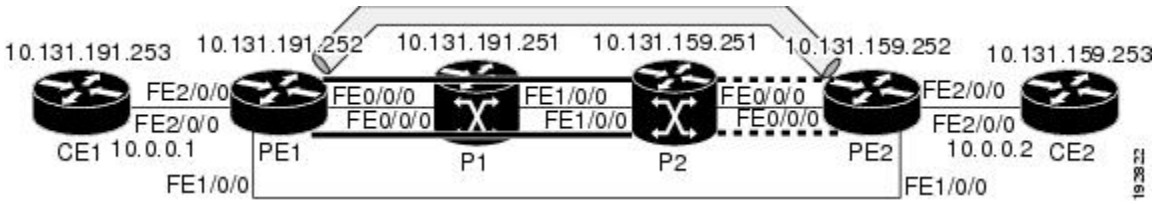
```
Router# traceroute mpls ipv4 10.6.7.4/32 ttl 5
Tracing MPLS Label Switched Path to 10.6.7.4/32, timeout is 2 seconds
Codes:
    '!' - success, 'Q' - request not sent, '.' - timeout,
    'L' - labeled output interface, 'B' - unlabeled output interface,
    'D' - DS Map mismatch, 'F' - no FEC mapping, '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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.6.1.14 MRU 4470 [Labels: 22 Exp: 0]
R 1 10.6.1.5 MRU 4474 [No Label] 3 ms
R 2 10.6.1.6 4 ms                  <------ Router address repeated for 2nd to 5th TTL.
R 3 10.6.1.6 1 ms
R 4 10.6.1.6 3 ms
R 5 10.6.1.6 4 ms
```

LSP ping または traceroute を使用したトラブルシューティング：例

ICMP の **ping** コマンドと **trace** コマンドは、多くの場合、エラーの根本原因の診断に使用されます。LSP が切断されている場合、パケットは IP フォワーディングによってターゲット ルータに到達することがあるため、ICMP の ping と traceroute 機能は、MPLS 転送の問題の検出では信頼性がありません。MPLS LSP ping または traceroute と AToM VCCV 機能は、この診断とトラブルシューティングの機能を MPLS ネットワークに拡張し、IP と MPLS の転送テーブル間の不整合（ある場合）、MPLS 制御とデータ プレーンにおける不整合、および応答パスの問題を処理します。

以下の図に、LDP LSP のサンプル トポロジを示します。

図 *7*：*LDP LSP* のサンプル トポロジ



ここでは、次の内容について説明します。

サンプル トポロジの設定

以降の項では、トラブルシューティング例のサンプル トポロジの設定を示します（上記の図を参照）。6 つのサンプル ルータ設定があります。

ルータ CE1 の設定

次に、CE1 ルータの設定を示します。

```
!
version 2.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE1
!
boot-start-marker
boot-end-marker
!
enable password lab
!
clock timezone EST -5
ip subnet-zero
!
!
!
interface Loopback0
 ip address 10.131.191.253 255.255.255.255
 no ip directed-broadcast
 no clns route-cache
!
!
interface FastEthernet2/0/0
 no ip address
 no ip directed-broadcast
 no keepalive
 no cdp enable
 no clns route-cache
!
interface FastEthernet2/0/0.1
 encapsulation dot1Q 1000
 ip address 10.0.0.1 255.255.255.0
 no ip directed-broadcast
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
end
```

ルータ PE1 の設定

次に、PE1 ルータの設定を示します。

```
!
version 2.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE1
!
boot-start-marker
boot-end-marker
!
logging snmp-authfail
enable password lab
!
```

```
clock timezone EST -5
ip subnet-zero
ip cef
no ip domain-lookup
!
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0 force
mpls label protocol ldp
!
!
!
interface Loopback0
 ip address 10.131.191.252 255.255.255.255
 no clns route-cache
!
interface FastEthernet0/0/0
 ip address 10.131.191.230 255.255.255.252
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.246 255.255.255.252
 shutdown
 no clns route-cache
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
interface FastEthernet2/0/0
 no ip address
 no cdp enable
 no clns route-cache
!
interface FastEthernet2/0/0.1
 encapsulation dot1Q 1000
 xconnect 10.131.159.252 333 encapsulation mpls
!
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.244 0.0.0.3 area 0
 network 10.131.191.228 0.0.0.3 area 0
 network 10.131.191.232 0.0.0.3 area 0
 network 10.131.191.252 0.0.0.0 area 0
!
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
!
end
```

ルータ P1 の設定

次に、P1 ルータの設定を示します。

```
version 2.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P1
!
boot-start-marker
boot-end-marker
```

```
!
logging snmp-authfail
enable password lab
!
clock timezone EST -5
ip subnet-zero
ip cef
no ip domain-lookup
!
!
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0 force
mpls label protocol ldp
!
!
!
 no clns route-cache
!
interface Loopback0
 ip address 10.131.191.251 255.255.255.255
 no clns route-cache
!
interface FastEthernet0/0/0
 ip address 10.131.191.229 255.255.255.252
 no clns route-cache
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.226 255.255.255.252
 no clns route-cache
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
interface FastEthernet1/1/0
 ip address 10.131.159.222 255.255.255.252
 no clns route-cache
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.220 0.0.0.3 area 0
 network 10.131.159.224 0.0.0.3 area 0
 network 10.131.191.228 0.0.0.3 area 0
 network 10.131.191.251 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
end
```

ルータ P2 の設定

次に、P2 ルータの設定を示します。

```
!
version 2.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
```

```
!
hostname P2
!
boot-start-marker
boot-end-marker
!
enable password lab
!
clock timezone EST -5
ip subnet-zero
ip cef
no ip domain-lookup
!
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0 force
mpls label protocol ldp
!
!
!
interface Loopback0
 ip address 10.131.159.251 255.255.255.255
 no ip directed-broadcast
!
interface FastEthernet0/0/0
 ip address 10.131.159.229 255.255.255.252
 no ip directed-broadcast
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
interface FastEthernet0/1/0
 ip address 10.131.159.233 255.255.255.252
no ip directed-broadcast
ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.225 255.255.255.252
no ip directed-broadcast
ip rsvp bandwidth 1500 1500
ip rsvp signalling dscp 0
!
interface FastEthernet1/1/0
 ip address 10.131.159.221 255.255.255.252
no ip directed-broadcast
ip rsvp signalling dscp 0
!
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.220 0.0.0.3 area 0
 network 10.131.159.224 0.0.0.3 area 0
 network 10.131.159.228 0.0.0.3 area 0
 network 10.131.159.232 0.0.0.3 area 0
 network 10.131.159.251 0.0.0.0 area 0
!
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
end
```

ルータ PE2 の設定

次に、PE2 ルータの設定を示します。

```
!
version 2.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE2
!
boot-start-marker
boot-end-marker
!
logging snmp-authfail
enable password lab
!
clock timezone EST -5
ip subnet-zero
ip cef
no ip domain-lookup
!
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0 force
mpls label protocol ldp
!
!
!
interface Loopback0
 ip address 10.131.159.252 255.255.255.255
 no clns route-cache
!
interface FastEthernet0/0/0
 ip address 10.131.159.230 255.255.255.252
 no clns route-cache
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet0/1/0
 ip address 10.131.159.234 255.255.255.252
 no clns route-cache
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.245 255.255.255.252
 mpls ip
 no clns route-cache
!
interface FastEthernet3/0/0
 no ip address
 no cdp enable
 no clns route-cache
!
interface FastEthernet3/0/0.1
 encapsulation dot1Q 1000
 no snmp trap link-status
 no cdp enable
 xconnect 10.131.191.252 333 encapsulation mpls
!
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.122.0 0.0.0.3 area 0
 network 10.131.159.228 0.0.0.3 area 0
 network 10.131.159.232 0.0.0.3 area 0
 network 10.131.159.236 0.0.0.3 area 0
 network 10.131.159.244 0.0.0.3 area 0
 network 10.131.159.252 0.0.0.0 area 0
```

```
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
!
end
```

ルータ CE2 の設定

次に、CE2 ルータの設定を示します。

```
!
version 2.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE2
!
boot-start-marker
boot-end-marker
!
enable password lab
!
clock timezone EST -5
ip subnet-zero
ip cef
no ip domain-lookup
!
!
interface Loopback0
 ip address 10.131.159.253 255.255.255.255
 no ip directed-broadcast
 no clns route-cache
!
interface FastEthernet3/0/0
 no ip address
 no ip directed-broadcast
 no keepalive
 no cdp enable
 no clns route-cache
!
interface FastEthernet3/0/0.1
 encapsulation dot1Q 1000
 ip address 10.0.0.2 255.255.255.0
 no ip directed-broadcast
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
end
```

LSP が正しく設定されているかどうかの確認

この項の **show** コマンドの出力を使用して、LSP が正しく設定されているかどうかを確認します。

show mpls forwarding-table コマンドは、トンネル 1 が MPLS 転送テーブルにあることを示しています。

```
PE1# show mpls forwarding-table 10.131.159.252
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
22     18
[T] 10.131.159.252/32 0          Tu1        point2point
[T]     Forwarding through a TSP tunnel.
        View additional tagging info with the 'detail' option
```

PE 1 で発行された **trace mpls** コマンドは、最も外側のラベルが 16 でスタック末尾のラベルが 18 であるパケットが PE1 から PE2 に転送されることを確認します。

```
PE1# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
   0 10.131.191.252 MRU 1496 [Labels: 16/18 Exp: 0/0] L 1 10.131.191.229
   MRU 1508 [Labels: 18 Exp: 0] 0 ms L 2 10.131.159.225
   MRU 1504 [Labels: implicit-null Exp: 0] 0 ms ! 3 10.131.159.234 20 ms
   PE1#
```

感嘆符（!）で示されているように、PE2 に対する MPLS LSP traceroute は成功しています。

LSP 切断の検出

この項のコマンドの出力を使用して、LSP の切断を検出します。

次の **show mpls ldp discovery** コマンドの出力に示されているように、ルータ PE1 と P2 の間に LDP ターゲット セッションが確立されています。

```
PE1# show mpls ldp discovery
 Local LDP Identifier:
    10.131.191.252:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
        Tunnel1 (ldp): Targeted -> 10.131.159.251
    Targeted Hellos:
        10.131.191.252 -> 10.131.159.252 (ldp): active/passive, xmit/recv
            LDP Id: 10.131.159.252:0
        10.131.191.252 -> 10.131.159.251 (ldp): active, xmit/recv
LDP Id: 10.131.159.251:0
```

P2 ルータで、次のコマンドをグローバル コンフィギュレーション モードで入力します。

```
P2(config)# no mpls ldp discovery targeted-hello accept
```

LDP設定の変更により、TE トンネルのヘッドエンドとテールエンド間のターゲット LDP セッションがダウンします。P2 で学習された IPv4 プレフィックスのラベルは、PE1 にアドバタイズされません。したがって、P2 から到達可能なすべての IP プレフィックスには、PE1 から MPLS ではなく IP を経由する場合にだけ到達可能です。つまり、PE1 のトンネル 1 を経由したそれらのプレフィックス宛てのパケットは、P2 で IP スイッチングされます（これは望ましくありません）。

次の **show mpls ldp discovery** コマンドは、LDP ターゲット セッションがダウンしていることを示しています。

```
PE1# show mpls ldp discovery
 Local LDP Identifier:
    10.131.191.252:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
        Tunnel1 (ldp): Targeted -> 10.131.159.251
    Targeted Hellos:
        10.131.191.252 -> 10.131.159.252 (ldp): active/passive, xmit/recv
            LDP Id: 10.131.159.252:0
        10.131.191.252 -> 10.131.159.251 (ldp): active, xmit
```

PE1 ルータで **show mpls forwarding-table** コマンドを入力します。次の表示は、LDP 設定が変更された結果、発信パケットが非タグ付きになったことを示しています。

```
PE1# show mpls forwarding-table 10.131.159.252
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
22     Untagged[T] 10.131.159.252/32 0          Tu1        point2point
[T]     Forwarding through a TSP tunnel.
        View additional tagging info with the 'detail' option
```

PE1 ルータで **ping mpls** コマンドを入力すると、次のように表示されます。

```
PE1# ping mpls ipv4 10.131.159.252/32 repeat 1
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
      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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
R
Success rate is 0 percent (0/1)
```

この **ping mpls** コマンドは失敗しています。R は、MPLS エコー応答の送信元にルーティング エントリがあり、MPLS FEC がないことを示します。**ping mpls** コマンドで **verbose** キーワードを入力すると、MPLS LSP エコー応答の送信元アドレスと戻りコードが表示されます。応答ルータに対する telnet とフォワーディングやラベルのテーブルの検査によって、切断が発生した場所を特定できる必要があります。切断はアップストリーム ルータで発生する可能性があるため、隣接するアップストリーム ルータも調べる必要があります。

```
PE1# ping mpls ipv4 10.131.159.252/32 repeat 1 verbose
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
      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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
R   10.131.159.225, return code 6
Success rate is 0 percent (0/1)
```

または、LSP **traceroute** コマンドを使用して、切断の原因となったデバイスを特定します。次の例では、TTL の後続の値が 2 よりも大きい場合、同じルータ（10.131.159.225）が応答し続けます。これは、TTL にかかわらず、MPLS エコー要求はそのルータによって処理され続けることを意味

します。ラベル スタックの検査によって、P1 が最後のラベルをポップし、パケットを IP パケットとして P2 に転送することがわかります。これは、パケットが P2 によって処理され続ける理由を説明するものです。MPLS エコー要求パケットは、IP ヘッダーの宛先アドレスを使用して転送できません。これは、アドレスが 127/8 アドレスに設定されているためです。

```
PE1# trace mpls ipv4 10.131.159.252/32 ttl 5
Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.191.230 MRU 1496 [Labels: 22/19 Exp: 0/0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 40 ms
R 2 10.131.159.229 MRU 1504 [implicit-null] 28 ms
! 3 10.131.159.230 40 ms
pe1#
```

LSP での MTU ディスカバリ：例

次の例は、LSP が LDP によって作成されたラベルで構成されている場合に **trace mpls** コマンドを実行した結果を示しています。

```
PE1# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.191.230 MRU 1496 [Labels: 22/19 Exp: 0/0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 40 ms
R 2 10.131.159.229 MRU 1504 [implicit-null] 28 ms
! 3 10.131.159.230 40 ms
pe1#
```

show mpls forwarding detail コマンドを使用すると、各ホップの LSP の MRU を確認できます。

```
PE1# show mpls forwarding 10.131.159.252 detail
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
22     19          10.131.159.252/32 0          Tu1        point2point
        MAC/Encaps=14/22, MRU=1496, Tag Stack{22 19}, via Et0/0
        AABBCC009700AABBCC0098008847 0001600000013000
        No output feature configured
```

LSP に収容されるエコー要求の大きさを確認するには、まず **show interface***interface-name* コマンドを使用して、IP MTU のサイズを計算します。

```
PE1# show interface e0/0
FastEthernet0/0/0 is up, line protocol is up
  Hardware is Lance, address is aabb.cc00.9800 (bia aabb.cc00.9800)
  Internet address is 10.131.191.230/30
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:01, output hang never
```

```
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     377795 packets input, 33969220 bytes, 0 no buffer
     Received 231137 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     441772 packets output, 40401350 bytes, 0 underruns
     0 output errors, 0 collisions, 10 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

show interface*interface-name* の例では、IP MTU は 1500 バイトです。MTU の数値からラベル スタックに対応するバイト数を引きます。**show mpls forwarding** コマンドの出力は、タグ スタックが 1 つのラベル（21）で構成されていることを示しています。したがって、LSP で送信できる最も大きい MPLS エコー要求パケットは、1500 - (2 X 4) = 1492 になります。

これを検証するには、次の **mpls ping** コマンドを使用します。

```
PE1# ping mpls ipv4 10.131.159.252/32 sweep 1492 1500 1 repeat 1
Sending 1, [1492..1500]-byte MPLS Echos to 10.131.159.252/32,
     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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
!QQQQQQQQ
Success rate is 11 percent (1/9), round-trip min/avg/max = 40/40/40 ms
```

このコマンドでは、サイズ範囲が 1492 ～ 1500 バイトのエコー パケットが宛先アドレスに送信されます。感嘆符（!）で示されているように、1492 バイトのパケットだけが正常に送信されています。バイト サイズが 1493 ～ 1500 のパケットは、Q で示されているように、送信元で抑制されました。

指定サイズのペイロードをテストできるように、MPLS エコー要求をパディングできます。パディング TLV は、MPLS エコー要求を使用して LSP でサポート可能な MTU を検出する場合に役立ちます。MTU ディスカバリは、フラグメント化できない非 IP ペイロードを含む AToM のようなアプリケーションにはきわめて重要です。

暗黙的ヌルとタグ付けされたパケットの追跡：例

次の例では、トンネル 1 はシャットダウンされ、LDP ラベルで構成された LSP だけが確立されます。暗黙的ヌルは、P2 ルータと PE2 ルータの間でアドバタイズされます。PE1 ルータで MPLS LSP traceroute コマンドを入力すると、次のように出力され、パケットが暗黙的ヌル ラベル付きで P2 から PE2 に転送されることが示されます。アドレス 10.131.159.229 は、PE2 ルータに対する P2 のファスト イーサネット 0/0/0 出力インターフェイス用に設定されています。

```
PE1# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.191.230 MRU 1496 [Labels: 22/19 Exp: 0/0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 40 ms
R 2 10.131.159.229 MRU 1504 [implicit-null] 28 ms
! 3 10.131.159.230 40 ms
pe1#
```

非タグ付きパケットの追跡：例

非タグ付きの例は、MPLS VPN の問題の原因となる可能性がある IGP LSP に有効な設定です。

P2 ルータで発行された **show mpls forwarding-table** コマンドと **show mpls ldp discovery** コマンドは、LDP が正しく設定されていることを示しています。

```
P2# show mpls forwarding-table 10.131.159.252
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
19     Pop tag     10.131.159.252/32 0          fe0/0/0    10.131.159.230
P2# show mpls ldp discovery
 Local LDP Identifier:
    10.131.159.251:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0/0 (ldp): xmit/recv
            LDP Id: 10.131.159.252:0
        FastEthernet1/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
```

show mpls ldp discovery コマンドの出力には、PE2 を P2 に接続するファスト イーサネット インターフェイス 0/0/0 がパケットを送受信していることが示されます。

ファスト イーサネット インターフェイス 0/0/0 に対して **no mpls ip** コマンドを入力すると、P2 ルータと PE2 ルータ間の LDP セッションが確立されない可能性があります。PE ルータで入力した **show mpls ldp discovery** コマンドは、PE2 との MPLS LDP セッションがダウンしていることを示しています。

```
P2# show mpls ldp discovery
 Local LDP Identifier:
    10.131.159.251:0
    Discovery Sources:
    Interfaces:

FastEthernet0/0/0 (ldp): xmit
        FastEthernet1/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
```

PE2 との MPLS LDP セッションがダウンすると、**show mpls forwarding-table** コマンドで示されるように、10.131.159.252 への LSP が非タグ付きになります。

```
P2# show mpls forwarding-table 10.131.159.252/32
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
19     Untagged    10.131.159.252/32 864        fe0/0/0    10.131.159.230
```

非タグ付きの例では、次に示すように、MPLS LSP traceroute の応答に No Label のタグが付いたパケットが含まれます。P2 から PE2 への出力インターフェイス（この例ではファスト イーサネッ

ト 0/0/0）に対して **mpls ip** コマンドを入力して、インターフェイス P2 から PE2 への MPLS LSP セッションを再確立する必要があります。

```
PE1# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.191.230 MRU 1500 [Labels: 20 Exp: 0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 80 ms
R 2 10.131.159.229 MRU 1504 [No Label] 28 ms       <----No MPLS session from P2 to PE2.
! 3 10.131.159.230 40 ms
```

パケットを送信できない原因の特定：例

次に、MPLS エコー要求が送信されない場合の **ping mpls** コマンドの例を示します。返された Q によって、伝送エラーが示されています。

```
PE1# ping mpls ipv4 10.0.0.1/32
Sending 5, 100-byte MPLS Echos to 10.0.0.1/32,
      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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
QQQQQ
Success rate is 0 percent (0/5)
```

次の **show mpls forwarding-table** コマンドと **show ip route** コマンドは、IPv4 アドレス（10.0.0.1）が LFIB または RIB ルーティングテーブルにないことを示しています。したがって、MPLS エコー要求は送信されません。

```
PE1# show mpls forwarding-table 10.0.0.1
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
PE1# show ip route 10.0.0.1
% Subnet not in table
```

IPv4 LSP でロード バランシングがイネーブルになっている場合の LSP 切断の検出：例

次の例では、宛先が同じパスが複数あります。これらの例の出力は、送信元ルータとターゲットルータ間でロード バランシングが行われていることを示しています。

PE1 ルータのファスト イーサネット インターフェイス 1/0/0 を動作させるには、PE1 ルータで次のコマンドを入力します。

```
PE1# configure terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
PE1(config)# interface fastethernet 1/0/0
PE1(config-if)# no shutdown
PE1(config-if)# end
*Dec 31 19:14:10.034: %LINK-3-UPDOWN: Interface FastEthernet1/0/0, changed state to up
*Dec 31 19:14:11.054: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0/0,
changed state to upend
PE1#
*Dec 31 19:14:12.574: %SYS-5-CONFIG_I: Configured from console by console
*Dec 31 19:14:19.334: %OSPF-5-ADJCHG: Process 1, Nbr 10.131.159.252 on FastEthernet1/0/0
from LOADING to FULL, Loading Done
PE1#
```

次の **show mpls forwarding-table** コマンドは、プレフィックス 10.131.159.251/32 の発信インターフェイスとネクスト ホップを表示します。

```
PE1# show mpls forwarding-table 10.131.159.251/32
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
21     19          10.131.159.251/32 0          fe0/0/0    10.131.191.229
       20          10.131.159.251/32 0          fe1/0/0    10.131.159.245
```

宛先 UDP アドレスが 127.0.0.1 の 10.131.159.251/32 に対する次の **ping mpls** コマンドは、選択したパスのパス インデックスが 0 であることを示しています。

```
Router# ping mpls ipv4
 10.131.159.251/32 destination
 127.0.0.1/32
Sending 1, 100-byte MPLS Echos to 10.131.159.251/32,
      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, '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 = 40/40/40 ms
PE1#
*Dec 29 20:42:40.638: LSPV: Echo Request sent on IPV4 LSP, load_index 2,
pathindex 0, size 100
*Dec 29 20:42:40.638: 46 00 00 64 00 00 40 00 FF 11 9D 03 0A 83 BF FC
*Dec 29 20:42:40.638: 7F 00 00 01 94 04 00 00 0D AF 0D AF 00 4C 14 70
*Dec 29 20:42:40.638: 00 01 00 00 01 02 00 00 1A 00 00 1C 00 00 00 01
*Dec 29 20:42:40.638: C3 9B 10 40 A3 6C 08 D4 00 00 00 00 00 00 00 00
*Dec 29 20:42:40.638: 00 01 00 09 00 01 00 05 0A 83 9F FB 20 00 03 00
*Dec 29 20:42:40.638: 13 01 AB CD AB CD AB CD AB CD AB CD AB CD AB CD
*Dec 29 20:42:40.638: AB CD AB CD
*Dec 29 20:42:40.678: LSPV: Echo packet received: src 10.131.159.225,
dst 10.131.191.252, size 74
*Dec 29 20:42:40.678: AA BB CC 00 98 01 AA BB CC 00 FC 01 08 00 45 C0
*Dec 29 20:42:40.678: 00 3C 32 D6 00 00 FD 11 15 37 0A 83 9F E1 0A 83
*Dec 29 20:42:40.678: BF FC 0D AF 0D AF 00 28 D1 85 00 01 00 00 02 02
*Dec 29 20:42:40.678: 03 00 1A 00 00 1C 00 00 00 01 C3 9B 10 40 A3 6C
*Dec 29 20:42:40.678: 08 D4 C3 9B 10 40 66 F5 C3 C8
```

宛先 UDP アドレスが 127.0.0.3 の 10.131.159.251/32 に対する次の **ping mpls** コマンドは、選択したパスのパス インデックスが 1 であることを示しています。

```
PE1# ping mpls ipv4 10.131.159.251/32 destination 127.0.0.3/32
Sending 1, 100-byte MPLS Echos to 10.131.159.251/32,
      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, '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 = 40/40/40 ms
PE1#
*Dec 29 20:43:09.518: LSPV: Echo Request sent on IPV4 LSP, load_index 13,
pathindex 1, size 100
*Dec 29 20:43:09.518: 46 00 00 64 00 00 40 00 FF 11 9D 01 0A 83 BF FC
*Dec 29 20:43:09.518: 7F 00 00 03 94 04 00 00 0D AF 0D AF 00 4C 88 58
*Dec 29 20:43:09.518: 00 01 00 00 01 02 00 00 38 00 00 1D 00 00 00 01
*Dec 29 20:43:09.518: C3 9B 10 5D 84 B3 95 84 00 00 00 00 00 00 00 00
*Dec 29 20:43:09.518: 00 01 00 09 00 01 00 05 0A 83 9F FB 20 00 03 00
*Dec 29 20:43:09.518: 13 01 AB CD AB CD AB CD AB CD AB CD AB CD AB CD
*Dec 29 20:43:09.518: AB CD AB CD
*Dec 29 20:43:09.558: LSPV: Echo packet received: src 10.131.159.229,
dst 10.131.191.252, size 74
*Dec 29 20:43:09.558: AA BB CC 00 98 01 AA BB CC 00 FC 01 08 00 45 C0
*Dec 29 20:43:09.558: 00 3C 32 E9 00 00 FD 11 15 20 0A 83 9F E5 0A 83
*Dec 29 20:43:09.558: BF FC 0D AF 0D AF 00 28 D7 57 00 01 00 00 02 02
*Dec 29 20:43:09.558: 03 00 38 00 00 1D 00 00 00 01 C3 9B 10 5D 84 B3
*Dec 29 20:43:09.558: 95 84 C3 9B 10 5D 48 3D 50 78
```

選択された実際のパスを確認するには、**packet** キーワードと **data** キーワードを指定して debug **debug mpls lspv** コマンドを入力します。

（注） ロード バランシング アルゴリズムは、IP ヘッダーの送信元アドレスと宛先アドレスに基づくハッシュによって、パケットを使用可能な出力パスに均一に分散しようとします。**destination** キーワードに *address-start*、*address-end*、および *address-increment* 引数を指定しても、期待どおりの結果が得られない場合があります。

エコーパケットがルータから発信されるときに経由するインターフェイスの指定：例

次に、上流のルータからロード バランシングをテストする例を示します。

```
Router# ping mpls ipv4 10.131.161.251/32 ttl 1 repeat 1 dsmap hashkey ipv4 bitmap 8

Sending 1, 100-byte MPLS Echos to 10.131.161.251/32,
     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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
L
Echo Reply received from 10.131.131.2
  DSMAP 0, DS Router Addr 10.131.141.130, DS Intf Addr 10.131.141.130
    Depth Limit 0, MRU 1500 [Labels: 54 Exp: 0]
    Multipath Addresses:
      127.0.0.3       127.0.0.5       127.0.0.7       127.0.0.8

  DSMAP 1, DS Router Addr 10.131.141.2, DS Intf Addr 10.131.141.2
    Depth Limit 0, MRU 1500 [Labels: 40 Exp: 0]
    Multipath Addresses:
      127.0.0.1       127.0.0.2       127.0.0.4       127.0.0.6
```

次に、エコー応答の送信元アドレスが 2 ホップ離れていると判断し、アップストリームにアドバタイズされた rx ラベルをチェックして、中継ルータが適切な結果を報告したことを検証する例を示します。

```
Success rate is 0 percent (0/1)
Router# trace mpls ipv4 10.131.161.251/32 destination 127.0.0.6 ttl 2
Tracing MPLS Label Switched Path to 10.131.161.251/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.131.1 10.131.131.2 MRU 1500 [Labels: 37 Exp: 0]
L 1 10.131.131.2 10.131.141.2 MRU 1500 [Labels: 40 Exp: 0] 0 ms, ret code 8
L 2 10.131.141.2 10.131.150.2 MRU 1504 [Labels: implicit-null Exp: 0] 0 ms, ret code 8
Router#
Router# telnet 10.131.141.2
Trying 10.131.141.2 ... Open
User Access Verification
Password:
Router> enable
The following example shows how the output interface
 keyword forces an LSP traceroute out FastEthernet interface 0/0/0:
Router# show mpls forwarding-table 10.131.159.251
Local  Outgoing      Prefix            Bytes Label   Outgoing   Next Hop
Label  Label or VC   or Tunnel Id      Switched      interface
20     19            10.131.159.251/32 0             fe1/0/0    10.131.159.245
       18            10.131.159.251/32 0             fe0/0/0    10.131.191.229
Router# trace mpls ipv4 10.131.159.251/32

Tracing MPLS Label Switched Path to 10.131.159.251/32, timeout is 2 seconds
Type escape sequence to abort.
  0 10.131.159.246 MRU 1500 [Labels: 19 Exp: 0]
L 1 10.131.159.245 MRU 1504 [Labels: implicit-null Exp: 0] 4 ms
! 2 10.131.159.229 20 ms
Router# trace mpls ipv4 10.131.159.251/32 output-interface fastethernet0/0/0
Tracing MPLS Label Switched Path to 10.131.159.251/32, timeout is 2 seconds
Type escape sequence to abort.
  0 10.131.191.230 MRU 1500 [Labels: 18 Exp: 0]
L 1 10.131.191.229 MRU 1504 [Labels: implicit-null Exp: 0] 0 ms
! 2 10.131.159.225 1 ms
```

パケット伝送のペーシング：例

次に、パケットの伝送ペースの例を示します。

```
Router# ping mpls ipv4 10.5.5.5/32 interval 100

Sending 5, 100-byte MPLS Echos to 10.5.5.5/32,
     timeout is 2 seconds, send interval is 100 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,
  '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 = 28/29/36 ms PE-802
```

中継ルータに対するダウンストリーム情報の問い合わせ：例

次に、2 本の出力パスを持つルータに問い合わせた場合のサンプル出力を示します。

```
Router# ping mpls ipv4 10.161.251/32 ttl 4 repeat 1 dsmap hashkey ipv4 bitmap 16

Sending 1, 100-byte MPLS Echos to 10.131.161.251/32,
     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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
L
Echo Reply received from 10.131.131.2
  DSMAP 0, DS Router Addr 10.131.141.130, DS Intf Addr 10.131.141.130
    Depth Limit 0, MRU 1500 [Labels: 54 Exp: 0]
    Multipath Addresses:
      127.0.0.3       127.0.0.6       127.0.0.9       127.0.0.10
      127.0.0.12      127.0.0.13      127.0.0.14      127.0.0.15
      127.0.0.16
  DSMAP 1, DS Router Addr 10.131.141.2, DS Intf Addr 10.131.141.2
    Depth Limit 0, MRU 1500 [Labels: 40 Exp: 0]
    Multipath Addresses:
      127.0.0.1       127.0.0.2       127.0.0.4       127.0.0.5
      127.0.0.7       127.0.0.8       127.0.0.11
Success rate is 0 percent (0/1)
```

マルチパスアドレスにより、パケットは出力ラベルスタックを使用してルータに中継されます。出力パスの本数の確認には **ping mpls** コマンドが有用ですが、ルータが 2 ホップ以上離れている場合は、ルータでこれらのアドレスを使用して問い合わせ対象のルータにパケットを中継できるとはかぎりません。このような状況になるのは、IP ヘッダーの宛先アドレスが変更されたために、送信元ルータと応答ルータの間にあるルータによって、パケットが異なる方法でロードバランシングされることがあるためです。ロードバランシングは、IP ヘッダーの送信元アドレスの影響を受けます。次に、アップストリームルータからのロードバランシング報告をテストする例を示します。

```
Router# ping mpls ipv4 10.131.161.251/32 ttl 1 repeat 1 dsmap hashkey ipv4 bitmap 8

Sending 1, 100-byte MPLS Echos to 10.131.161.251/32,
     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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
L
Echo Reply received from 10.131.131.2
  DSMAP 0, DS Router Addr 10.131.141.130, DS Intf Addr 10.131.141.130
    Depth Limit 0, MRU 1500 [Labels: 54 Exp: 0]
    Multipath Addresses:
      127.0.0.3       127.0.0.5       127.0.0.7       127.0.0.8

  DSMAP 1, DS Router Addr 10.131.141.2, DS Intf Addr 10.131.141.2
    Depth Limit 0, MRU 1500 [Labels: 40 Exp: 0]
    Multipath Addresses:
      127.0.0.1       127.0.0.2       127.0.0.4       127.0.0.6
To validate that the transit router reported the proper results, determine the Echo Reply
sender address that is two hops away and consistently check the rx label that is advertised
 upstream. The following is sample output:
```

```
Success rate is 0 percent (0/1)
Router# trace mpls ipv4 10.131.161.251/32 destination 127.0.0.6 ttl 2
Tracing MPLS Label Switched Path to 10.131.161.251/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.131.1 10.131.131.2 MRU 1500 [Labels: 37 Exp: 0]
L 1 10.131.131.2 10.131.141.2 MRU 1500 [Labels: 40 Exp: 0] 0 ms, ret code 8
L 2 10.131.141.2 10.131.150.2 MRU 1504 [Labels: implicit-null Exp: 0] 0 ms, ret code 8
Router#
Router# telnet 10.131.141.2

Trying 10.131.141.2 ... Open
User Access Verification
Password:
Router> enable
Router# show mpls forwarding-table 10.131.161.251

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
40     Pop tag     10.131.161.251/32 268        fe1/0/0    10.131.150.2
Router#
```

ルータに対する DSMAP の問い合わせ：例

次に、ソフトウェアとハードウェアのフォワーディング レイヤに DSMAP TLV で返す必要がある深度制限を問い合わせる例を示します。

```
Router# ping mpls ipv4 10.131.159.252/32 ttl 1 dsmap
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
     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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
L
Echo Reply received from 10.131.191.229
  DSMAP 0, DS Router Addr 10.131.159.225, DS Intf Addr 10.131.159.225
    Depth Limit 0, MRU 1508 [Labels: 18 Exp: 0]
    Multipath Addresses:
      127.0.0.1       127.0.0.2       127.0.0.3       127.0.0.4
      127.0.0.5       127.0.0.6       127.0.0.7       127.0.0.8
      127.0.0.9       127.0.0.10      127.0.0.11      127.0.0.12
      127.0.0.13      127.0.0.14      127.0.0.15      127.0.0.16
      127.0.0.17      127.0.0.18      127.0.0.19      127.0.0.20
      127.0.0.21      127.0.0.22      127.0.0.23      127.0.0.24
      127.0.0.25      127.0.0.26      127.0.0.27      127.0.0.28
      127.0.0.29      127.0.0.30      127.0.0.31      127.0.0.32
Success rate is 0 percent (0/1)
```

中継ルータによるターゲット FEC スタックの検証の要求：例

次に、中継ルータで、テスト対象の LSP を識別するターゲット FEC スタックを検証する例を示します。

```
Router# trace mpls ipv4 10.5.5.5/32 flags fec

Tracing MPLS Label Switched Path to 10.5.5.5/32, 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 label entry,
  '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 10.2.3.2 10.2.3.3 MRU 1500 [Labels: 19 Exp: 0] L 1 10.2.3.3 10.3.4.4 MRU 1500 [Labels:
 19 Exp: 0] 40 ms, ret code 8 L 2 10.3.4.4 10.4.5.5 MRU 1504 [Labels: implicit-null Exp:
0] 32 ms, ret code 8 ! 3 10.4.5.5 40 ms, ret code 3
Router# ping mpls ipv4 10.5.5.5/32

Sending 5, 100-byte MPLS Echos to 10.5.5.5/32
     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 label entry,
  '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.
!    size 100, reply addr 10.4.5.5, return code 3
!    size 100, reply addr 10.4.5.5, return code 3
!    size 100, reply addr 10.4.5.5, return code 3
!    size 100, reply addr 10.4.5.5, return code 3
!    size 100, reply addr 10.4.5.5, return code 3
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/32 ms
```

LSP ping のイネーブル化による、非タグ付きインターフェイスを起因とする LSP 切断の検出：例

次に、明示的ヌル ラベル シムがある場合に、ラベル スタックの最後に追加されるラベルの例を示します。

```
Switch# trace mpls ipv4 10.131.159.252/32 force-explicit-null

Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.191.252 MRU 1492 [Labels: 16/18/explicit-null Exp: 0/0/0]
L 1 10.131.191.229 MRU 1508 [Labels: 18/explicit-null Exp: 0/0] 0 ms
L 2 10.131.159.225 MRU 1508 [Labels: explicit-null Exp: 0] 0 ms
! 3 10.131.159.234 4 ms
```

次に、明示的ヌル ラベル シムがない場合のコマンド出力例を示します。

```
Switch# trace mpls ipv4 10.131.159.252/32

Tracing MPLS Label Switched Path to 10.131.159.252/32, 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, 'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
  0 10.131.191.252 MRU 1496 [Labels: 16/18 Exp: 0/0]
L 1 10.131.191.229 MRU 1508 [Labels: 18 Exp: 0] 4 ms
L 2 10.131.159.225 MRU 1504 [Labels: implicit-null Exp: 0] 4 ms
! 3 10.131.159.234 4 ms
```

ピアにアドバタイズされた AToM VCCV 機能やピアから受信した AToM VCCV 機能の表示：例

次に、ルータ PE1 が AToM VCCV タイプ 1 とタイプ 2 の両方のスイッチング機能をアドバタイズし、リモート ルータ PE2 がタイプ 2 のスイッチング機能だけをアドバタイズする例を示します。

```
Router# show mpls l2transport binding

  Destination Address: 10.131.191.252,  VC ID: 333
    Local Label:  16
        Cbit: 1,    VC Type: FastEthernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV Capabilities: Type 1, Type 2 <----- Locally advertised VCCV capabilities
    Remote Label: 19
        Cbit: 1,    VC Type: FastEthernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV Capabilities: Type 2           <-----Remotely advertised VCCV capabilities
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
MPLS コマンド	『Cisco IOS Multiprotocol Label Switching Command Reference』
MPLS 転送プロファイル コンフィギュレーション資料	MPLS トランスポート プロファイル

標準および RFC

標準/RFC	タイトル
draft-ietf-mpls-tp-te-mib-02.txt	MPLS-TP トラフィックエンジニアリング（TE）Management Information Base (MIB)

シスコのテクニカル サポート

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

MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV の機能情報

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

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

表 *12*：*MPLS LSP ping/traceroute for LDP/TE* および *LSP ping for VCCV* の機能情報

機能名	リリース	機能情報
MPLS 組み込み管理機能 LSP ping/traceroute for LDP	Cisco IOS XE Release 2.1	この機能は、Cisco ASR 1000 シリーズのアグリゲーションサービスルータで導入されました。

機能名	リリース	機能情報
MPLS 組み込み管理機能 LSP ping/traceroute for LDP および Resource Reservation Protocol（RSVP）IPv4 Forwarding Equivalence Classes（FEC）	Cisco IOS XE Release 2.3	MPLS 組み込み管理機能 LSP ping/traceroute for LDP 機能が変更されて RSVP IPv4 FEC をサポートするようになりました。
MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV	Cisco IOS XE Release 2.3	MPLS LSP ping/traceroute for LDP/TE および LSP ping for VCCV 機能を使用すると、サービス プロバイダーはラベル スイッチド パスを監視したり、MPLS 転送の問題を迅速に隔離したりできます。 次のコマンドが導入または変更されました。**debug mpls lspv**、**echo**、**mpls oam**、**ping mpls**、**show mpls oam**、**echo statistics**、**trace mpls**。

用語集

FEC：Forward Equivalence Class。転送のために同等に処理できるパケットのセット。したがって、1 つのラベルへのバインディングに適しています。たとえば、1 つのアドレス プレフィックス宛てのパケットのセットやフロー内のパケットなどがあります。

フロー：一組のホスト間、または一組のホスト上にある一組のトランスポートプロトコルポート間で転送されるパケットのセット。たとえば、同じ送信元アドレス、送信元ポート、宛先アドレス、および宛先ポートを持つパケットは、フローと見なされることがあります。

フローは、ネットワークの 2 つのエンドポイント間で（たとえば、ある LAN ステーションから別の LAN ステーションへ）転送されるデータのストリームでもあります。単一の回線上で複数のフローを転送できます。

フラグメンテーション：元のパケットサイズをサポートできないネットワーク メディアを介してパケットを送信するときに、パケットを小さい単位に分割するプロセス。

ICMP：Internet Control Message Protocol。エラーを報告し、IP パケット処理に関連するその他の情報を提供するネットワーク層インターネット プロトコル。RFC 792 に記載されています。

LFIB：Label Forwarding Information Base（ラベル転送情報ベース）。宛先および着信ラベルが発信インターフェイスおよびラベルに関連付けられている転送を管理するデータ構造および手段。

localhost：ホスト ルータ（デバイス）を表す名前。localhost は、予約済みのループバック IP アドレス（127.0.0.1）を使用します。

LSP：Label Switched Path（ラベル スイッチド パス）。MPLS がパケットを転送する 2 つのルータ間の接続。

LSPV：Label Switched Path Verification。LSP ping のサブプロセス。MPLS エコー要求と応答を符号化およびデコードします。また、MPLS エコー要求と応答を送受信するために、IP、MPLS、および ATOM スイッチングとやり取りします。MPLS エコー要求の発信元ルータでは、LSPV によって、エコー応答が受信されていない未処理のエコー要求のデータベースが保持されます。

MPLS ルータ アラート ラベル：MPLS ラベル 1。ルータ アラート ラベルを含む MPLS パケットは、処理のためにルータによってルート プロセッサ（RP）の処理レベルにリダイレクトされます。これにより、これらのパケットはハードウェア ルーティングテーブルにおけるフォワーディング エラーを回避できます。

MRU：Maximum Receive Unit（最大受信ユニット）。LSP を介して転送できる、ラベル付きパケットの最大サイズ（バイト単位）。

MTU：Maximum Transmission Unit（最大伝送ユニット）。特定のインターフェイスが送受信できる最大パケット サイズ（バイト単位）。

パント：ルータ アラートを含むパケットを処理のためにラインカードまたはインターフェイスからルート プロセッサ（RP）のレベル処理にリダイレクトします。

PW：疑似回線。パケット スイッチド ネットワークを介して、エミュレートされた回線の重要な要素を、あるプロバイダー エッジ（PE）ルータから別の PE ルータに伝送するトンネルの形式。

RP：ルート プロセッサ。Cisco 7000 シリーズ ルータのプロセッサ モジュール。CPU、システム ソフトウェア、およびルータで使用されるほとんどのメモリ コンポーネントが含まれます。スーパーバイザリ プロセッサと呼ばれることもあります。

RSVP：Resource Reservation Protocol。IP ネットワーク上でリソースの予約をサポートするためのプロトコル。IP エンド システム上で動作しているアプリケーションは、RSVP を使用して、受信するパケット ストリームの特性（帯域幅、ジッタ、最大バーストなど）を他のノードに示すことができます。RSVP は IPv6 に依存します。リソース予約設定プロトコルとも呼ばれます。

TLV：Type, Length, Value（タイプ、長さ、値）。Cisco Discovery Protocol アドレスに含まれる情報のブロックです。

TTL 隠蔽：存続可能時間は、設定可能なパラメータで、パケットが宛先に到達するまでに通過するホップの最大数を示します。

UDP：User Datagram Protocol（ユーザ データグラムプロトコル）。TCP/IP プロトコル スタックのコネクションレス型トランスポート層プロトコルです。UDP は、確認応答や配信保証を行わずにデータグラムを交換する単純なプロトコルです。そのため、エラー処理と再伝送を他のプロトコルで処理する必要があります。UDP は RFC 768 で定義されています。

第 4 章

MPLS LSP ping、traceroute、AToM VCCV

マルチプロトコルラベルスイッチング（MPLS）の導入の増加に伴い、送信できるトラフィックタイプが増えるため、サービスプロバイダーがラベルスイッチドパス（LSP）を監視し、MPLS 転送の問題を迅速に特定できることが、サービスを提供する上で重要です。MPLS LSP Ping、Traceroute、および AToM VCCV 機能を利用することで、このような課題に対応できるようになります。

MPLS LSP Ping、Traceroute、および AToM VCCV 機能は、LSP がユーザ トラフィックの配信に失敗したことを検出できます。

- MPLS LSP Ping を使用して、IPv4 Label Distribution Protocol（LDP）プレフィックス、トラフィック エンジニアリング（TE）Forwarding Equivalence Class（FEC）、および AToM FEC の LSP 接続をテストできます。
- MPLS LSP Traceroute を使用して、IPv4 LDP プレフィックスと TE トンネル FEC の LSP をトレースできます。
- Any Transport over MPLS Virtual Circuit Connection Verification（AToM VCCV）では、MPLS LSP ping を使用して、AToM 仮想回線（VC）の疑似回線（PW）セクションをテストできます。

Internet Control Message Protocol（ICMP）ping および trace は、転送が失敗する場合の根本原因の診断によく使用されます。MPLS LSP Ping、Traceroute、および AToM VCCV 機能は、この診断とトラブルシューティングの機能を MPLS ネットワークに拡張し、IP と MPLS の転送テーブル間の不整合、MPLS 制御とデータ プレーンにおける不整合、および応答パスの問題の特定を支援します。

MPLS LSP Ping、Traceroute、および AToM VCCV 機能は、MPLS エコー要求パケットと応答パケットを使用して LSP をテストします。MPLS エコー要求とエコー応答のシスコ実装は、Internet Engineering Task Force（IETF）インターネット ドラフト『*Detecting MPLS Data Plane Failures*』に基づいています。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS LSP ping、traceroute、AToM VCCV の前提条件

MPLS LSP ping、traceroute、および AToM VCCV 機能を使用する前に、次のことを行う必要があります。

- マルチプロトコル ラベル スイッチング（MPLS）ネットワークの基本動作を決定します。次に例を示します。
  - 期待される MPLS EXP 試験（EXP）処理。
  - ラベル スイッチド パスに予想されるパケットの最大サイズまたは最大伝送ユニット（MTU）。
  - 使用されるトポロジ。予想されるラベル スイッチド パス。ラベル スイッチング パス（LSP）のリンク数。ロード バランシング用のパスなど、ラベル スイッチド パケットのパスをトレースします。
- トラフィック エンジニアリング、Any Transport over MPLS（AToM）、Label Distribution Protocol（LDP）を含め、MPLS および MPLS アプリケーションの使用方法を理解します。それには、次のことを理解する必要があります。
  - LDP の設定方法
  - AToM の概念
  - TE トンネルのトラブルシューティング方法
- ラベル スイッチング、転送、ロード バランシング

MPLS LSP ping、traceroute、AToM VCCV の制約事項

- MPLS LSP traceroute を使用して、Any Transport over Multiprotocol Label Switching（AToM）パケットがたどるパスをトレースすることはできません。MMPLS LSP traceroute は AToM ではサポートされません。（MPLS LSP ping は AToM でサポートされます。）ただし、MPLS LSP traceroute を使用して、AToM によって使用される Interior Gateway Protocol（IGP）LSP をトラブルシューティングすることはできます。
- MPLS LSP ping または traceroute を使用して、MPLS バーチャル プライベート ネットワーク（VPN）を検証またはトレースすることはできません。
- MPLS LSP traceroute を使用して、存続可能時間（TTL）隠蔽を使用するラベル スイッチング パス（LSP）をトラブルシューティングすることはできません。

MPLS LSP Ping、Traceroute、および AToM VCCV に関する情報

MPLS LSP ping の動作

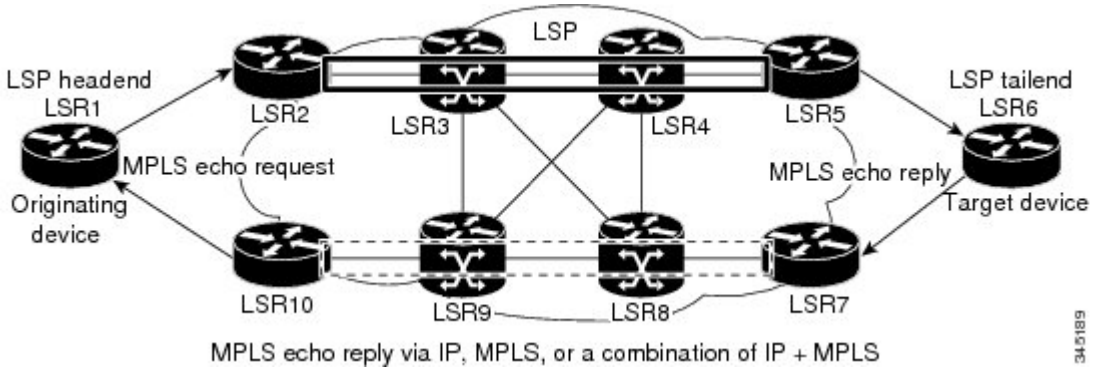
MPLS LSP ping は、ラベル スイッチド パス（LSP）の検証に、マルチプロトコル ラベル スイッチング（MPLS）エコー要求および応答パケットを使用します。MPLS エコー要求と MPLS エコー応答は、送信元ポートと宛先ポートが 3503 に設定された User Datagram Protocol（UDP）パケットです。

MPLS エコー要求パケットは、検証対象の LSP に関連付けられた適切なラベル スタックを使用してターゲット デバイスに送信されます。ラベル スタックを使用すると、パケットは LSP のインバンドにスイッチングされます（LSP 自体を介して転送されます）。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。UDP パケットの宛先アドレスは、127.*x* .*y* .*z* /8 アドレスとして定義されます。これにより、LSP が切断された場合に IP パケットが宛先に IP スイッチングされるのを防ぐことができます。

MPLS エコー応答は、MPLS エコー要求に応じて送信されます。応答は IP パケットとして送信され、IP、MPLS、または両方のスイッチング タイプの組み合わせを使用して転送されます。MPLS エコー応答パケットの送信元アドレスは、エコー応答を生成するデバイスから取得されたアドレスです。宛先アドレスは、MPLS エコー要求パケットのデバイスの送信元アドレスです。

次の図に、MPLS LSP Ping のエコー要求とエコー応答のパスを示します。

図 *8*：***MPLS LSP ping*** のエコー要求とエコー応答のパス



LSR1 で LSR6 の Forwarding Equivalence Class（FEC）に対する MPLS LSP ping 要求を開始すると、次の表に示すような結果になります。

表 *13*：***MPLS LSP ping*** の例

ステップ	デバイス	アクション
1	LSR1	ターゲット デバイス LSR6 の FEC に対する MPLS LSP ping 要求を開始し、MPLS エコー要求を LSR2 に送信します。
1	LSR2	MPLS エコー要求パケットを受信し、中継デバイス LSR3 と LSR4 を経由して最後から 2 番めのデバイス LSR5 に転送します。
1	LSR5	MPLS エコー要求を受信し、MPLS ラベルをポップしてパケットを IP パケットとして LSR6 に転送します。
1	LSR6	IP パケットを受信し、MPLS エコー要求を処理して、代替ルート経由で MPLS エコー応答を LSR1 に送信します。
1	LSR7 ～ LSR10	MPLS エコー応答を受信し、LSR1（発信元デバイス）に転送します。

ステップ	デバイス	アクション
1	LSR1	MPLS エコー要求に対する MPLS エコー応答を受信します。

MPLS LSP ping を使用して、コマンドに適切なキーワードと引数を指定することで、IPv4 Label Distribution Protocol（LDP）、Any Transport over MPLS（AToM）、および IPv4 Resource Reservation Protocol（RSVP）FEC を検証できます。

```
ping mpls
 {ipv4

destination-address destination-mask
 | pseudowire

ipv4-address
 vc-id

| traffic-eng

tunnel-interface tunnel-number
}
```

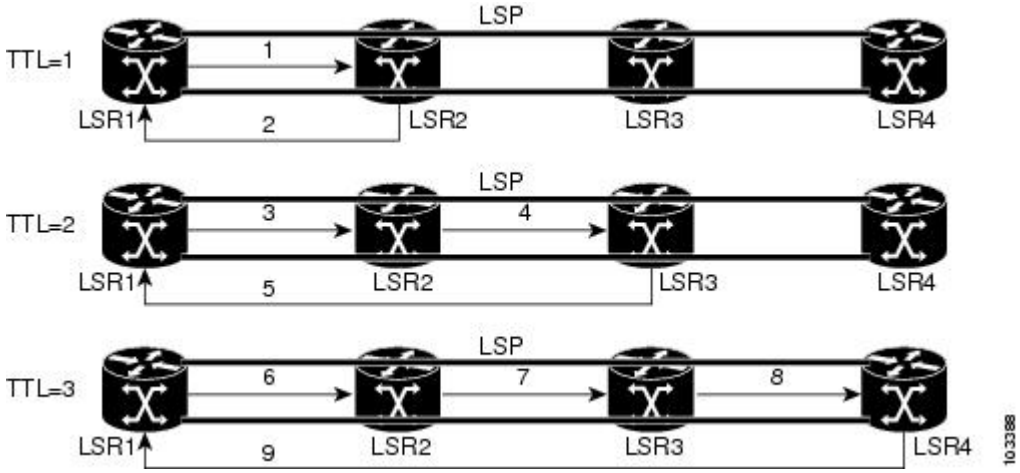
MPLS LSP Traceroute の動作

MPLS LSP Traceroute は、ラベル スイッチド パス（LSP）の検証に、マルチプロトコル ラベル スイッチング（MPLS）エコー要求および応答パケットも使用します。エコー要求とエコー応答は、送信元ポートと宛先ポートが 3503 に設定された User Datagram Protocol（UDP）パケットです。

MPLS LSP Traceroute 機能は存続可能時間（TTL）設定を使用し、LSP に沿って TTL を強制的に期限切れにします。MPLS LSP Traceroute は、連続した各ホップのダウンストリーム マッピングを検出するために、自身の MPLS エコー要求の TTL 値（TTL = 1、2、3、4、...）を付加的に増加させます。LSP traceroute の成否は、TTL = 1 のラベル付きパケットの受信時に MPLS エコー要求を処理する中継デバイスに依存します。Cisco デバイスでは、TTL が期限切れになると、パケットが処理のためにルート プロセッサ（RP）に送信されます。中継デバイスは、TTL 期限の切れた MPLS パケットに応じて中継ホップの情報を持つ MPLS エコー応答を戻します。

次の図に、LSR1 から LSR4 までの LSP の MPLS LSP Traceroute の例を示します。

図 *9*： ***MPLS LSP Traceroute*** の例



LSR1 から LSR4 の Forwarding Equivalence Class（FEC）に対する LSP traceroute を入力すると、次の表に示すような結果になります。

表 *14*： ***MPLS LSP Traceroute*** の例

ステップ	デバイス	**MPLS** パケットタイプと説明	デバイス アクション
1	LSR1	MPLS エコー要求：ターゲット FEC は LSR4 とダウンストリーム マッピングを指す。	• ラベル スタックの TTL を 1 に設定する。 • 要求を LSR2 に送信する。
1	LSR2	MPLS エコー応答。	TTL＝1 のパケットを受信する。 • UDP パケットを MPLS エコー要求として処理する。 • ダウンストリーム マッピングを検索し、着信ラベルに基づいて独自のダウンストリーム マッピングを付加して LSR1 に応答し、応答を送信する。

ステップ	デバイス	MPLS パケットタイプと説明	デバイス アクション
1	LSR1	MPLS エコー要求：ターゲット FEC は同じで、LSR2 からのエコー応答で受信したダウンストリーム マッピングを含む。	• ラベル スタックの TTL を 2 に設定する。 • 要求を LSR2 に送信する。
1	LSR2	MPLS エコー要求。	TTL = 2 のパケットを受信する。 • TTL を減らす。 • エコー要求を LSR3 に転送する。
1	LSR3	MPLS 応答パケット。	TTL = 1 のパケットを受信する。 • UDP パケットを MPLS エコー要求として処理する。 • ダウンストリーム マッピングを検索し、着信ラベルに基づいて独自のダウンストリーム マッピングを付加して LSR1 に応答する。
1	LSR1	MPLS エコー要求：ターゲット FEC は同じで、LSR3 からのエコー応答で受信したダウンストリーム マッピングを含む。	• パケットの TTL を 3 に設定する。 • 要求を LSR2 に送信する。
1	LSR2	MPLS エコー要求。	TTL = 3 のパケットを受信する。 • TTL を減らす。 • エコー要求を LSR3 に転送する。

ステップ	デバイス	MPLS パケットタイプと説明	デバイス アクション
1	LSR3	MPLS エコー要求。	TTL = 2 のパケットを受信する。 • TTL を減らす。 • エコー要求を LSR4 に転送する。
1	LSR4	MPLS エコー応答。	TTL = 1 のパケットを受信する。 • UDP パケットを MPLS エコー要求として処理する。 • ダウンストリーム マッピングを検索し、デバイスがターゲット FEC の出力デバイスであることも確認する。 • LSR1 に応答する。

MPLS LSP Traceroute を使用すると、**trace mpls** コマンドで適切なキーワードと引数を使用することによって、IPv4 Label Distribution Protocol（LDP）と IPv4 RSVP FEC を検証できます。

```
trace mpls ipv4 {destination-address destination-mask | traffic-eng
 tunnel-interface tunnel-number}
```

デフォルトでは、TTL は 30 に設定されます。したがって、LSP の問題が発生している場合でも、traceroute の出力には常に 30 行が含まれます。LSP の問題が発生すると、出力に重複エントリが含まれます。トレースが最後に到達するポイントのデバイス アドレスは、出力が 30 行になるまで繰り返されます。重複したエントリは無視できます。次に、トレース中に IP アドレスが 10.6.1.6 のデバイスで LSP の問題が発生した場合の例を示します。

```
Device# traceroute mpls ipv4 10.6.7.4/32
Tracing MPLS Label Switched Path to 10.6.7.4/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.6.1.14 MRU 4470 [Labels: 22 Exp: 0]
R 1 10.6.1.5 MRU 4470 [Labels: 21 Exp: 0] 2 ms
R 2 10.6.1.6 4 ms                 <------ Router address repeated for 2nd to 30th TTL.
R 3 10.6.1.6 1 ms
R 4 10.6.1.6 1 ms
R 5 10.6.1.6 3 ms
R 6 10.6.1.6 4 ms
R 7 10.6.1.6 1 ms
R 8 10.6.1.6 2 ms
R 9 10.6.1.6 3 ms
R 10 10.6.1.6 4 ms
```

```
R 11 10.6.1.6 1 ms
R 12 10.6.1.6 2 ms
R 13 10.6.1.6 4 ms
R 14 10.6.1.6 5 ms
R 15 10.6.1.6 2 ms
R 16 10.6.1.6 3 ms
R 17 10.6.1.6 4 ms
R 18 10.6.1.6 2 ms
R 19 10.6.1.6 3 ms
R 20 10.6.1.6 4 ms
R 21 10.6.1.6 1 ms
R 22 10.6.1.6 2 ms
R 23 10.6.1.6 3 ms
R 24 10.6.1.6 4 ms
R 25 10.6.1.6 1 ms
R 26 10.6.1.6 3 ms
R 27 10.6.1.6 4 ms
R 28 10.6.1.6 1 ms
R 29 10.6.1.6 2 ms
R 30 10.6.1.6 3 ms                    <------ TTL 30.
```

ネットワーク内の最大ホップ数がわかっている場合は、**trace mpls ttl** *maximum-time-to-live* コマンドを使用して、TTL を小さい値に設定できます。次の例では、上記の例と同じ **traceroute** コマンドを使用していますが、TTL が 5 に設定されています。

```
Device# traceroute mpls ipv4 10.6.7.4/32 ttl 5
Tracing MPLS Label Switched Path to 10.6.7.4/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.6.1.14 MRU 4470 [Labels: 22 Exp: 0]
R 1 10.6.1.5 MRU 4474 [No Label] 3 ms
R 2 10.6.1.6 4 ms                  <------ Router address repeated for 2nd to 5th TTL.
R 3 10.6.1.6 1 ms
R 4 10.6.1.6 3 ms
R 5 10.6.1.6 4 ms
```

Any Transport over MPLS 仮想回線接続の検証

AToM 仮想回線接続性検証（AToM VCCV）を使用すると、送信元のプロバイダー エッジ（PE）ルータから AToM 疑似回線（PW）の帯域内で制御パケットを送信できます。伝送は宛先 PE デバイスで代行受信され、カスタマーエッジ（CE）デバイスには転送されません。この機能により、MPLS LSP Ping を使用して AToM 仮想回線（VC）の PW セクションをテストできます。

AToM VCCV は次のコンポーネントで構成されます。

- VC ラベルのシグナリング中に AToM VCCV 機能がアドバタイズされるシグナリング対象のコンポーネント
- AToM VC ペイロードが制御パケットとして処理されるスイッチング コンポーネント

AToM VCCV シグナリング

Any Transport over Multiprotocol Label Switching（AToM）仮想回線（VC）設定手順の 1 つに、AToM VC エンドポイント間での VC ラベルと AToM Virtual Circuit Connection Verification（VCCV）機能のシグナリングがあります。デバイスは、インターネット ドラフト（*draft-ieft-pwe3-vccv-01.txt*）

で定義されているオプション パラメータを使用して、各エンドポイントの AToM VCCV ディスポジション機能と通信します。

AToM VCCV ディスポジション機能は、次のように分類されます。

- アプリケーション：MPLS LSP ping と Internet Control Message Protocol（ICMP）ping は、パケットを制御用に AToM PW の帯域内で送信するために AToM VCCV によってサポートされるアプリケーションです。
- スイッチング モード：AToM VCCV で制御トラフィックとデータ トラフィックを区別するために使用されるスイッチング モードとして、タイプ 1 とタイプ 2 があります。

次の表に、AToM VCCV のタイプ 1 とタイプ 2 のスイッチング モードを示します。

表 *15*：タイプ *1* とタイプ *2* の *AToM VCCV* スイッチング モード

スイッチング モード	説明
タイプ 1	AToM 制御ワードのプロトコル ID（PID）フィールドを使用して、AToM VCCV パケットを識別します。
タイプ 2	VC ラベルの上の MPLS ルータ アラート ラベルを使用して、AToM VCCV パケットを識別します。

AToM VCCV スイッチング タイプの選択

Cisco デバイスでは、Any Transport over Multiprotocol Label Switching（AToM）仮想回線（VC）制御チャネルを介して MPLS LSP ping パケットを送信するときに、使用可能な場合は常にタイプ 1 スイッチングを使用します。タイプ 2 スイッチングは、AToM 制御ワードをサポートまたは解釈しない VC タイプと実装に対応します。

以下の表に、AToM VC によってアドバタイズおよび選択される AToM Virtual Circuit Connection Verification（VCCV）スイッチング モードを示します。

表 *16*：*AToM* 仮想回線によってアドバタイズおよび選択される *AToM VCCV* スイッチング モード

アドバタイズされるタイプ	選択されるタイプ
AToM VCCV はサポートされない	–
タイプ 1 AToM VCCV スイッチング	タイプ 1 AToM VCCV スイッチング
タイプ 2 AToM VCCV スイッチング	タイプ 2 AToM VCCV スイッチング

アドバタイズされるタイプ	選択されるタイプ
タイプ 1 およびタイプ 2 AToM VCCV スイッチング	タイプ 1 AToM VCCV スイッチング

AToM VC は、AToM VCCV ディスポジション機能を両方向、つまり送信元デバイス（PE1）から宛先デバイス（PE2）へ、PE2 から PE1 へアドバタイズします。

2 つのエンドポイントの AToM VCCV 機能が異なる場合、AToM VC は異なるスイッチング タイプを使用することがあります。PE1 がタイプ 1 およびタイプ 2 AToM VCCV スイッチングをサポートし、PE2 がタイプ 2 AToM VCCV スイッチングだけをサポートしている場合は、次の 2 とおりの結果になります。

- PE1 から PE2 に送信された LSP ping パケットは、タイプ 2 スイッチングでカプセル化される。
- PE2 から PE1 に送信された LSP ping パケットは、タイプ 1 スイッチングを使用する。

ピアにアドバタイズされた AToM VCCV 機能やピアから受信した AToM VCCV 機能を確認するには、PE デバイスで **show mpls l2transport binding** コマンドを入力します。次に例を示します。

```
Device# show mpls l2transport binding

  Destination Address: 10.131.191.252,  VC ID: 333
    Local Label:  16
        Cbit: 1,    VC Type: FastEthernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV Capabilities: Type 1, Type 2
    Remote Label: 19
        Cbit: 1,    VC Type: FastEthernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV Capabilities: Type 1
```

ping mpls および trace mpls のコマンド オプション

MPLS LSP Ping および Traceroute コマンドのオプションは、**ping mpls** コマンドと **trace mpls** コマンドのキーワードおよび引数として指定されます。

ping mpls コマンドには、次のコマンド構文でオプションを指定します。

```
ping mpls ipv4{destination-address/destination-mask [destination address-start
address-end increment] [ttl time-to-live] | pseudowire ipv4-address
vc-id vc-id [destination address-start address-end increment] |
traffic-eng tunnel-interface tunnel-number [ttl time-to-live]} [source
source-address] [repeat count] [timeout seconds][{size
 packet-size} | {sweep minimum maximum size-Increment}] [pad pattern]
[reply mode {ipv4|router-alert}] [interval msec]
[exp exp-bits] [verbose]
```

trace mpls コマンドには、次のコマンド構文でオプションを指定します。

```
trace mpls {ipv4 destination-address/destination-mask [destination
address-start [address-end [address-increment]]] | traffic-eng tunnel tunnel-interface-number}
[source source-address] [timeout seconds] [reply mode reply-mode]
```

```
[ttl maximum-time-to-live] [exp exp-bits]
```

検証対象の FEC の選択

ラベル スイッチド パス（LSP）はラベルで構成されています。デバイスはこれらのラベルを学習するために、Label Distribution Protocol（LDP）、トラフィック エンジニアリング（TE）、Any Transport over Multiprotocol Label Switching（AToM）、または他の MPLS アプライアンスを使用します。MPLS LSP ping および traceroute を使用して、特定の Forwarding Equivalence Class（FEC）のトラフィックを転送するために使用される LSP を検証できます。以下の表に、検証に使用する LSP を選択するに指定できる、**ping mpls** コマンドと **traceroute mpls** コマンドのキーワードと引数を記載します。

表 *17*：検証に使用する *LSP* の選択

FEC タイプ	ping mpls のキーワードおよび引数	traceroute mpls のキーワードおよび引数
LDP IPv4 プレフィックス	**ipv4***destination-address destination-mask*	**ipv4***destination-address destination-mask*
MPLS TE トンネル	**traffic-eng***tunnel-interfacetunnel-number*	**traffic-eng***tunnel-interfacetunnel-number*
AToM VC	**pseudowire***ipv4-address***vc-id***vc-id*	このリリースでは、MPLS LSP traceroute は AToM トンネル LSP タイプをサポートしていません。

MPLS LSP ping および traceroute に対する応答モードのオプション

応答モードを使用して、MPLS LSP ping または MPLS LSP traceroute コマンドによって送信されたマルチプロトコル ラベル スイッチング（MPLS）エコー要求に対する応答側デバイスの応答方法を制御します。以下の表に、応答モードのオプションを記載します。

表 *18*：応答側デバイスの応答モードのオプション

オプション	説明
ipv4	IPv4 User Datagram Protocol（UDP）パケットで応答します（デフォルト）。ラベル スイッチド パス（LSP）の整合性を定期的にポーリングする場合、この応答モードが MPLS LSP ping および MPLS LSP traceroute コマンドで最も一般的に選択されます。 このオプションは、パケットが IP ホップと MPLS ホップのいずれを通過して MPLS エコー要求の送信元に到達するかを明示的に制御するものではありません。 ヘッドエンドデバイスが応答を受信できない場合は、router-alert オプションを選択し、ルータ アラートを設定して IPv4 UDP パケットで応答します。 応答側デバイスは応答パケットの IP プレシデンスを 6 に設定します。 このオプションを実装するには、**reply mode ipv4** キーワードを使用します。
router-alert	デバイス アラートを設定して IPv4 UDP パケットで応答します。この応答モードでは、IP ヘッダーにルータ アラート オプションが追加されます。これにより、パケットが宛先に戻るときに、各中間ホップでパケットにシスコデバイスによる特殊な処理が適用されます。 この応答モードにはコストがかかるため、ipv4 オプションの「IPv4 UDP パケットで応答」では応答を取得できない場合にだけ、router-alert オプションを使用してください。 このオプションを実装するには、**reply mode router-alert** キーワードを使用します。

IPv4 UDP パケットで応答するということは、デバイスが MPLS エコー要求に対する応答で IPv4 UDP パケットを送信しなければならないことを意味します。ipv4 応答モードを選択する場合、MPLS エコー要求の送信元に到達するために、パケットで IP を使用するか MPLS ホップを使用するかを明示的に制御する必要はありません。このモードは通常、LSP をテストして検証する場合に使用します。

デバイス アラートが含まれる IPv4 UDP で応答すると、パケットは宛先に戻され、各中間ホップでルート プロセッサ（RP）のプロセル スイッチングによって処理されます。これにより、ハードウェアやラインカードの転送テーブルの不一致をバイパスします。送信元（ヘッドエンド）デバイスが MPLS エコー要求に対する応答を受信できない場合には、このオプションを選択してください。

応答側デバイスに IP ルータ アラート オプションを使用したエコー応答を送信させるには、次のいずれかのコマンドを使用します。

```
ping mpls
 {ipv4 destination-address/destination-mask | pseudowire ipv4-address
 vc-idvc-id | traffic-engtunnel-interface tunnel-number}
reply mode  router-alert
```

または

```
trace mpls
 {ipv4destination-address/destination-mask
 | traffic-eng tunnel-interface tunnel-number
} reply mode router-alert
```

ただし、ルータ アラートを使用した応答により、送信元デバイスに応答を返すプロセスにオーバーヘッドが追加されます。この方法はルータ アラートなしの応答よりもコストがかかるため、応答を受信できない場合に限って使用してください。つまり、MPLS LSP ping または MPLS LSP traceroute では、送信元（ヘッドエンド）デバイスが MPLS エコー要求に対する応答を受信できない場合にのみ、ルータ アラート ラベルを設定した応答を使用する必要があります。

リターン パスでの IP MPLS ルータ アラートを伴うパケットの処理

IP ヘッダーに IP ルータ アラート オプションを含む IP パケット、または最も外側のラベルとしてルータ アラート ラベルを含むマルチプロトコル ラベル スイッチング（MPLS）パケットがデバイスに到達すると、デバイスはパケットを処理するためにルート プロセッサ（RP）プロセス レベルにパント（リダイレクト）します。これにより、これらのパケットはハードウェア ルーティング テーブルにおけるフォワーディング エラーを回避できます。以下の表に、IP ルータ アラート オプションを含む IP パケットと MPLS パケットがデバイス スイッチング パス プロセスによって処理される方法を示します。

表 *19*：スイッチング パス プロセスによる *IP* および *MPLS* ルータ アラート パケットの処理

着信パケット	通常のスイッチング アクション	プロセス スイッチング アクション	発信パケット
IP パケット：IP ヘッダーにルータ アラート オプションが含まれる	IP ヘッダーにルータ アラート オプションが含まれていると、パケットはプロセス スイッチング パスにパントされる。	パケットをそのまま転送する。	IP パケット：IP ヘッダーにルータ アラート オプションが含まれる
	IP ヘッダーにルータ アラート オプションが含まれていると、パケットはプロセス スイッチング パスにパントされる。	ルータ アラートを最も外側のラベルとして追加し、MPLS パケットとして転送する。	MPLS パケット：最も外側のラベルにルータ アラートが含まれる
MPLS パケット：最も外側のラベルにルータ アラートが含まれる	ルータ アラート ラベルが最も外側のラベルである場合、パケットはプロセス スイッチング パスにパントされる。	最も外側のルータ アラート ラベルを削除し、IP ルータ アラート オプションを IP パケットに追加して、IP パケットとして転送する。	IP パケット：IP ヘッダーにルータ アラート オプションが含まれる
	ルータ アラート ラベルが最も外側のラベルである場合、パケットはプロセス スイッチング パスにパントされる。	最も外側のルータ アラート ラベルを保持し、MPLS パケットを転送する。	MPLS パケット：最も外側のラベルにルータ アラートが含まれる

その他の MPLS LSP ping および traceroute コマンド オプション

以下の表に、**ping mpls** コマンド、または **ping mpls** コマンドと **trace mpls** コマンドの両方でキーワードまたは引数として指定できる、MPLS LSP ping および traceroute コマンドのその他のオプションを記載します。**ping mpls** コマンドでのみ使用できるオプションについては、MPLS LSP ping 機能専用と記載しています。

表 *20*：その他の *MPLS LSP ping* および *traceroute* と *AToM VCCV* オプション

オプション	説明
データグラム サイズ	ラベル スタックが適用されたパケットのサイズ。**size***packet-size* キーワードおよび引数で指定します。デフォルトのサイズは 100 です。 MPLS LSP ping 機能専用。

オプション	説明
パディング	MPLS エコー要求（ラベル スタックが適用された User Datagram Protocol（UDP）パケット）が指定のサイズに満たない場合、パディング（Time Length Value（TLV）パッド）を使用してデータグラムが埋められます。**pad***pattern* キーワードおよび引数で指定します。 MPLS LSP ping 機能専用。
スイープ サイズ範囲	開始サイズから終了サイズまでの範囲のさまざまなサイズのパケットを送信できるようにするためのパラメータ。このパラメータは、Internet Control Message Protocol（ICMP）の ping スイープ パラメータと同様です。スイープ範囲の下限として指定できる値は、ラベル スイッチド パス（LSP）タイプによって異なります。スイープ サイズ範囲を指定できるのは、**ping mpls** コマンドを使用する場合です。**sweep***minimum maximum size-increment* キーワードおよび引数を使用します。 MPLS LSP ping 機能専用。
繰り返し回数	同じパケットを再送信する回数。デフォルト値は 5 回です。繰り返し回数を指定できるのは、**ping mpls** コマンドを使用する場合です。**repeat***count* キーワードおよび引数を使用します。 MPLS LSP ping 機能専用。
MPLS エコー要求の送信元アドレス	送信者のルーティング可能なアドレス。デフォルトのアドレスは loopback0 です。マルチプロトコル ラベル スイッチング（MPLS）エコー応答では、このアドレスが宛先アドレスとして使用されます。**source***source-address* キーワードおよび引数を使用します。 MPLS LSP ping および traceroute 機能で使用します。

オプション	説明
UDP 宛先アドレス	有効な 127/8 アドレス。単一の *x.y.z* を指定するか、0.0.0 ～ *x.y.z* の範囲を指定できます。ここで、*x.y.z* は 0 ～ 255 の値で、127.*x.y.z* に対応します。**destination** {*address* | *address-start address-end increment*} キーワードおよび引数を使用します。 MPLS パケットを宛先デバイスに転送する際は、UDP パケット内の MPLS エコー要求の宛先アドレスは使用されません。エコー要求を転送するために使用されるラベルスタックが、MPLS パケットを宛先デバイスにルーティングします。UDP パケットの宛先アドレスが転送に使用される場合、127/8 アドレスにより、パケットが localhost（アドレスを処理するデバイスのデフォルトのループバック アドレス）にルーティングされることが保証されます。 さらに、IP ペイロードの宛先アドレスがロードバランシングに使用される場合は、ロードバランシングを調整するために宛先アドレスが使用されます。 MPLS LSP ping 機能を使用した IPv4 および Any Transport over MPLS（AToM）Forward Equivalence Class（FEC）と MPLS LSP traceroute 機能を使用した IPv4 FEC で使用します。

オプション	説明
存続可能時間（TTL）	設定可能なパラメータであり、パケットが宛先に到達するまでに通過するホップの最大数を示します。パケットがデバイスを通過するたびに、パケット内の存続可能時間（TTL）フィールドの値が 1 ずつ減算されます。 MPLS LSP ping の場合、TTL が経過すると、パケットが破棄されて MPLS エコー応答が発信側デバイスに返されます。**ttl***time-to-live* キーワードおよび引数を使用します。 MPLS LSP traceroute の場合、TTL は最大存続期間として、宛先デバイスまでのダウンストリームのホップ数を検出するために使用されます。MPLS LSP traceroute は自身の MPLS エコー要求の TTL 値を徐々に増分することで（TTL = 1、2、3、4...）、このホップ数を検出します。**ttl***time-to-live* キーワードおよび引数を使用します。
タイムアウト	MPLS 要求パケットのタイムアウト（秒単位）を制御するために指定できるパラメータ。指定できる範囲は 0 ～ 3600 秒です。デフォルトは 2 です。 **timeout***seconds* キーワードおよび引数で指定します。 MPLS LSP ping および traceroute 機能で使用します。
間隔	連続した MPLS エコー要求の間隔（ミリ秒単位）を指定するために使用できるパラメータ。デフォルトは 0 です。 **interval***msec* キーワードおよび引数で指定します。

オプション	説明
EXP ビット	MPLS エコー応答の優先順位を指定するために使用される、MPLSヘッダー内の3つの試験ビット（これらのビットは一般に EXP ビットと呼ばれます）。指定できる範囲は 0 ～ 7 で、デフォルトは 0 です。 **exp***exp-bits* キーワードおよび引数で指定します。 MPLS LSP ping および traceroute 機能で使用します。
詳細	MPLS エコー応答の追加情報（送信元アドレスと戻りコード）を提供するためのオプション。MPLS LSP ping 機能では、このオプションを **verbose** キーワードで実装します。 MPLS LSP ping 機能専用。

上記の表に記載されている MPLS LSP ping オプションを実装するには、次の構文を使用します。

```
ping mpls
{ipv4 destination-address destination-mask [destination address-start address-end increment]

 [ttl time-to-live] | pseudowire ipv4-address
 vc-id vc-id
[destination address-start address-end increment] | traffic-eng tunnel-interface
tunnel-number
[ttl time-to-live]}
[source source-address] [repeat count]
[{size packet-size} | {sweep minimum maximum size-Increment}]
[pad pattern]
[timeout seconds] [intervalmsec]
[exp exp-bits] [verbose
```

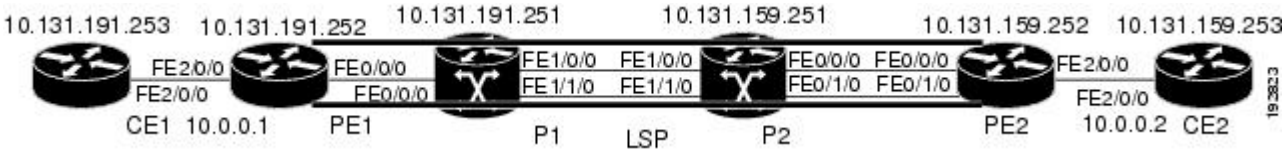
上記の表に記載されている MPLS LSP traceroute オプションを実装するには、次の構文を使用します。

```
trace mpls
 {ipv4 destination-address destination-mask
 [destination address-start address-end address-increment] | traffic-eng tunnel-interface
tunnel-number}
[source source-address] [timeout seconds]
[ttl maximum-time-to-live]
[exp exp-bits]
```

オプションの相互作用とループ

この項と以降の項で取り上げる MPLS LSP の ping および traceroute と AToM VCCV 機能の使用例は、次の図に示すサンプル トポロジに基づいています。

図 *10*：設定例のサンプル トポロジ



一部の MPLS LSP の ping および traceroute と AToM VCCV オプションの相互作用により、ループが発生する可能性があります。**ping mpls** コマンドと **trace mpls** コマンドで発生する可能性があるループについては、次の項を参照してください。

MPLS LSP ping で発生する可能性があるループ

MPLS LSP ping 機能で繰り返し回数オプション、スイープ サイズ範囲オプション、または User Datagram Protocol（UDP）宛先アドレス範囲オプションを指定すると、ループが発生する可能性があります。

```
ping mpls
 {ipv4 destination-address/destination-mask
[destination address-start address-end increment] | pseudowire ipv4-address
 vc-id vc-id
 [destination address-start address-end increment] |
traffic-eng tunnel-interface tunnel-number}
[repeat count]
[sweep minimum maximum size-increment]
```

以下に、**ping mpls** コマンドで次のキーワードと引数を使用した場合のループ動作の例を示します。

```
Device# ping mpls
 ipv4
 10.131.159.251/32 destination 127.0.0.1 127.0.0.1 0.0.0.1 repeat 2
sweep 1450 1475 25
Sending 2, [1450..1500]-byte MPLS Echos to 10.131.159.251/32,
     timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
Destination address 127.0.0.1
!
!
Destination address 127.0.0.1
!
!
Destination address 127.0.0.1
!
!
Destination address 127.0.0.1
!
!
```

mpls ping コマンドは、終了アドレスに到達するまで、各宛先アドレスについて、パケット サイズの範囲ごとに送信されます。この例では、宛先アドレス 127.0.0.1 に到達するまでループは同じように続行されます。シーケンスは、その回数が **repeat***count* キーワードと引数で指定した値に到達するまで続行されます。この例では、リピート回数は 2 です。MPLS LSP ping のループ シーケンスは次のようになります。

```
repeat  = 1
  destination address 1 (address-start
)
    for (size from sweep
 minimum
 to maximum
, counting by size-increment
)
      send an lsp ping
  destination address 2 (address-start
 +
address-
increment
)
    for (size from sweep
 minimum
 to maximum
, counting by size-increment
)
      send an lsp ping
  destination address 3 (address-start
 +
address-
increment
 +
address-
increment
)
     for (size from sweep
 minimum
 to maximum
, counting by size-increment
)
      send an lsp ping
.
.
.
  until destination address = address-end
.
.
.
until repeat = count
```

MPLS LSP traceroute で発生する可能性があるループ

MPLS LSP traceroute 機能を使用する場合、User Datagram Protocol（UDP）宛先アドレス範囲オプションと存続可能時間オプションを指定すると、ループが発生する可能性があります。

```
trace mpls
 {ipv4

destination-address destination-mask
 [destination

address-start
address-end

address-increment
] | traffic-eng
```

```
 tunnel-interface

tunnel-number
[ttl
 maximum-
time-to-live
]
```

以下に、**trace mpls** コマンドで次のキーワードと引数を使用した場合のループ動作の例を示します。

```
Device# trace mpls
ipv4
 10.131.159.251/32 destination 127.0.0.1 127.0.0.1 1 ttl 5
Tracing MPLS Label Switched Path to 10.131.159.251/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
Destination address 127.0.0.1
  0 10.131.191.230 MRU 1500 [Labels: 19 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
! 2 10.131.159.225 40 ms
Destination address 127.0.0.2
  0 10.131.191.230 MRU 1500 [Labels: 19 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
! 2 10.131.159.225 40 ms
Destination address 127.0.0.3
  0 10.131.191.230 MRU 1500 [Labels: 19 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
! 2 10.131.159.225 48 ms
```

mpls trace コマンドは、宛先の *end-address* 引数で指定されたアドレスに到達するまで、各宛先アドレスについて、1 から最大 TTL（**ttl***maximum-time-to-live* キーワードと引数）までの TTL ごとに送信されます。この例では、最大 TTL は 5 で、終了宛先アドレスは 127.0.0.1 です。MPLS LSP Traceroute のループ シーケンスは次のようになります。

```
destination address 1 (address-start
)
  for (ttl
 from 1 to maximum-time-to-live
)
    send an lsp trace
destination address 2 (address-start
 + address-increment
)
  for (ttl
 from 1 to maximum-time-to-live
)
    send an lsp trace
destination address 3 (address-start
 + address-increment
 + address-increment
)
  for (ttl
 from 1 to
maximum-time-to-live)
    send an lsp trace
.
.
.
until destination address = address-end
```

IP で転送されない MPLS エコー要求パケット

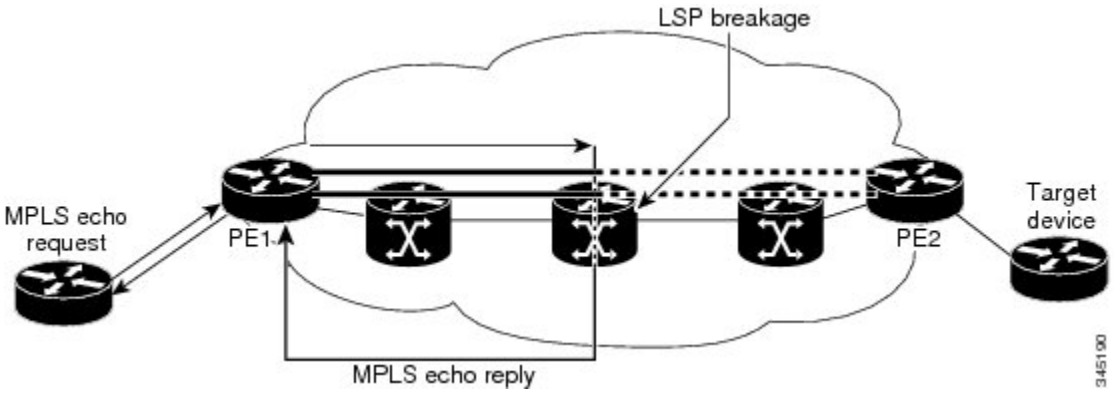
ラベル スイッチド パス（LSP）ping 中に送信されたマルチプロトコル ラベル スイッチング（MPLS）エコー要求パケットが、IP により転送されることはありません。MPLS エコー要求パケットの IP ヘッダーの宛先アドレス フィールドは 127.*x.y.z*/8 アドレスです。デバイスは 127.*x.y.z*/8 アドレスを使用したパケットを転送しません。127.*x.y.z*/8 アドレスは、ローカル ホストのアドレスに対応します。

127.*x .y .z* アドレスを User Datagram Protocol（UDP）パケットの宛先アドレスとして使用することが重要です。これは、中継デバイスが LSP のラベル スイッチングを行わない場合、MPLS エコー要求パケットは、このアドレスをターゲット デバイスにすることができないためです。これにより、LSP 切断を検出できます。

- 中継デバイスで LSP の切断が発生した場合、MPLS エコー パケットは転送されませんが、デバイスによって使用されます。
- LSP が切断されていない場合、MPLS エコー パケットはターゲット デバイスに到達し、LSP の終点で処理されます。

次の図に、中継デバイスが LSP でパケットのラベル スイッチングに失敗した場合の MPLS エコー要求と応答のパスを示します。

図 *11*：中継デバイスがパケットのラベル スイッチングに失敗した場合のパス



（注） Any Transport over MPLS（AToM）ペイロードは IP パケットではない可能性があるため、このペイロードには中継デバイスで使用可能なフォワーディング情報が格納されません。MPLS バーチャル プライベート ネットワーク（VPN）パケットは IP パケットですが、MPLS ネットワークのエンドポイントの Virtual Routing and Forwarding（VRF）インスタンスには宛先 IP アドレスだけが重要であるため、MPLS VPN パケットには中継デバイスで使用可能なフォワーディング情報が格納されません。

LSP ping または LSP traceroute を処理するデバイスから提供される情報

次の表に、LSP ping または LSP traceroute パケットを処理するデバイスから、要求の成否について送信者に返される文字について説明します。

ping mpls verbose コマンドを入力することでも、MPLS LSP ping 操作の戻りコードを表示できます。

表 *21*：*LSP ping* および *traceroute* の応答特性

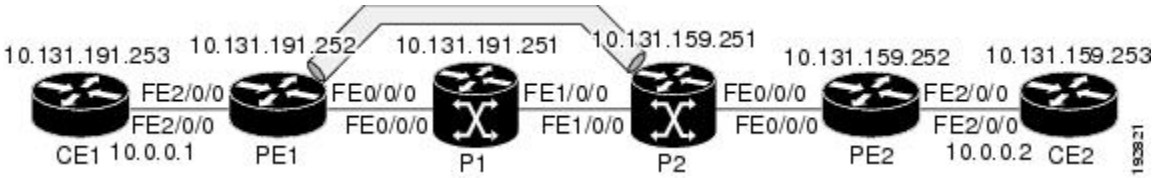
文字	意味
ピリオド（.）	ターゲットデバイスが応答する前にタイムアウトが発生しました。
U	ターゲット デバイスに到達できません。
R	マルチプロトコル ラベル スイッチング（MPLS）エコー要求を処理するデバイスはダウンストリームデバイスですが、宛先ではありません。
感嘆符（!）	応答デバイスは宛先の出力です。
Q	エコー要求が正常に送信されませんでした。これが返される理由としては、メモリが不十分であることが考えられますが、より可能性の高い理由は、Forwarding Equivalence Class（FEC）情報に一致するラベル スイッチドパス（LSP）が存在しないことです。
C	エコー要求の形式が正しくないため、応答デバイスがエコー要求を拒否しました。

LSP での MTU ディスカバリ

MPLS LSP ping の実行中、マルチプロトコル ラベル スイッチング（MPLS）エコー要求パケットは IP パケット属性が「do not fragment」に設定された状態で送信されます。つまり、パケットの IP ヘッダーに DF ビットが設定されます。これにより、MPLS エコー要求を使用して、フラグメンテーションなしでパケットがラベル スイッチド パス（LSP）を通過できるようにするための MTU をテストできます。

以下の図に、LDPによってアドバタイズされたラベルで構成されている1つのLSP（PE1からPE2まで）のサンプル ネットワークを示します。

図 *12*：*LSP* のサンプル ネットワーク：*LDP* によってアドバタイズされたラベル



MPLS Traceroute機能を使用してLSPをトレースすることによって、各ホップの最大受信ユニット（MRU）を確認できます。MRU は、LSP 経由で転送できる、ラベル付けされたパケットの最大サイズです。次の例は、LSP が Label Distribution Protocol（LDP）によって作成されたラベルで構成されている場合に **trace mpls** コマンドを実行した結果を示しています。

```
Device# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.131.191.230 MRU 1496 [Labels: 22/19 Exp: 0/0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 40 ms
R 2 10.131.159.229 MRU 1504 [implicit-null] 28 ms
! 3 10.131.159.230 40 ms
```

show forwarding detail コマンドを使用すると、各ホップの LSP の MRU を確認できます。

```
Device# show mpls forwarding 10.131.159.252 detail

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
22     19          10.131.159.252/32 0          Tu1        point2point
        MAC/Encaps=14/22, MRU=1496, Tag Stack{22 19}, via Et0/0
        AABBCC009700AABBCC0098008847 0001600000013000
        No output feature configured
```

LSPに収容できるエコー要求の最大サイズを確認するには、まず **show interface***type number* コマンドを使用して、IP MTU を調べます。

```
Device# show interface e0/0

FastEthernet0/0/0 is up, line protocol is up
  Hardware is Lance, address is aabb.cc00.9800 (bia aabb.cc00.9800)
  Internet address is 10.131.191.230/30
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load ½55
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:01, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     377795 packets input, 33969220 bytes, 0 no buffer
     Received 231137 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     441772 packets output, 40401350 bytes, 0 underruns
     0 output errors, 0 collisions, 10 interface resets
```

```
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

show interface*type number* の例では、IP MTU は 1500 バイトです。MTU の数値からラベル スタックに対応するバイト数を引きます。**show mpls forwarding** コマンドの出力では、タグ スタックは 1 つのラベル（21）で構成されています。したがって、上記の図に示されている LSP で送信できる最も大きい MPLS エコーパケット要求は、1500 - (2 X 4) = 1492 になります。

これを検証するには、次の **ping mpls** コマンドを使用します。

```
Device# ping mpls ipv4 10.131.159.252/32 sweep 1492 1500 1 repeat 1
Sending 1, [1492..1500]-byte MPLS Echos to 10.131.159.252/32,
     timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
!QQQQQQQQ
Success rate is 11 percent (1/9), round-trip min/avg/max = 40/40/40 ms
```

このコマンドの出力で感嘆符（!）によって示されているように 1492 バイトのパケットだけが正常に送信されていることがわかります。バイト サイズが 1493 ～ 1500 のパケットは、Q で示されているように、送信元で抑制されました。

指定サイズのペイロードをテストできるように、MPLSエコー要求をパディングできます。パディング TLV は、MPLS エコー要求を使用して LSP でサポート可能な MTU を検出する場合に役立ちます。MTU ディスカバリは、フラグメント化できない非 IP ペイロードを含む AToM のようなアプリケーションにはきわめて重要です。

LSP ネットワーク管理

マルチプロトコル ラベル スイッチング（MPLS）ネットワークを管理するには、ラベル スイッチド パス（LSP）を監視して MPLS 転送の問題を迅速に隔離できる必要があります。そのためには、LSP の動作を評価したり、ラベル スイッチド パスによるユーザ トラフィックの伝送の失敗を検出したりする方法が必要です。

MPLS LSP Ping を使用すると、IPv4 Label Distribution Protocol（LDP）プレフィックス、トラフィック エンジニアリング（TE）トンネル、および Any Transport over MPLS pseudowire Forwarding Equivalence Class（AToM PW FEC）宛てのパケットの転送に使用される LSP を確認できます。MPLS LSP Traceroute を使用すると、IPv4 LDP プレフィックスおよび TE トンネル FEC 宛てのパケットの伝送に使用される LSP をトレースできます。

MPLS エコー要求は、検証する LSP 経由で送信されます。TTL の期限切れまたは LSP の切断が発生すると、中継デバイスはエコー要求を目的の宛先に到達する前に処理し、説明的な応答コードを含む MPLS エコー応答をエコー要求の送信元に返します。

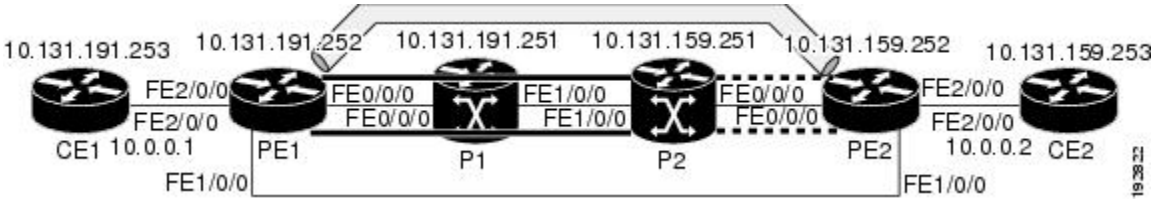
成功したエコー要求は LSP の出口で処理されます。エコー応答は IP パス、MPLS パス、または両方のパスの組み合わせを経由してエコー要求の送信元に返送されます。

ICMP ping および trace コマンドとトラブルシューティング

Internet Control Message Protocol（ICMP）の **ping** コマンドと **trace** コマンドは、多くの場合、エラーの根本原因の診断に使用されます。ラベル スイッチド パス（LSP）が切断されている場合、パケットは IP フォワーディングによってターゲット デバイスに到達することがあるため、ICMP の ping と traceroute は、マルチプロトコル ラベル スイッチング（MPLS）転送の問題の検出では信頼性がありません。MPLS LSP Ping または Traceroute と AToM VCCV 機能は、この診断とトラブルシューティングの機能を MPLS ネットワークに拡張し、IP と MPLS の転送テーブル間の不整合、MPLS 制御とデータ プレーンにおける不整合、および応答パスの問題を処理します。

次の図に、Label Distribution Protocol（LDP）LSP およびトラフィック エンジニアリング（TE）トンネル LSP を使用したトポロジの例を示します。

図 *13* : ***LDP LSP*** および ***TE*** トンネル ***LSP*** を使用したトポロジの例



ここでは、次の内容について説明します。

MPLS LSP Ping および Traceroute による LSP 切断の検出

サンプル トポロジの設定

以降の項では、トラブルシューティング例のサンプル トポロジの設定を示します（上記の図を参照）。6 つのサンプル デバイス設定があります。

デバイス CE1 の設定

```
version 12.0
!
hostname ce1
!
enable password lab
!
interface Loopback0
 ip address 10.131.191.253 255.255.255.255
 no ip directed-broadcast
!
interface FastEthernet2/0/0
 ip address 10.0.0.1 255.255.255.255
 no ip directed-broadcast
 no keepalive
 no cdp enable
!
end
```

デバイス PE1 の設定

```
version 12.0
!
hostname pe1
!
ip cef
mpls label protocol ldp
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers frequency 0
mpls ldp discovery targeted-hello accept
!
interface Loopback0
 ip address 10.131.191.252 255.255.255.255
 no ip directed-broadcast
!
interface Tunnel1
 ip unnumbered Loopback0
 no ip directed-broadcast
 mpls label protocol ldp
 mpls ip
 tunnel destination 10.131.159.255
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth  512
 tunnel mpls traffic-eng path-option 1 dynamic
!
interface Tunnel2
 ip unnumbered Loopback0
 no ip directed-broadcast
 shutdown
 mpls label protocol ldp
 mpls ip
 tunnel destination 10.131.159.255
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  100
 tunnel mpls traffic-eng path-option 1 dynamic
!
interface FastEthernet0/0/0
 ip address 10.131.191.230 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.246 255.255.255.255
 no ip directed-broadcast
 no shutdown
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet2/0/0
 no ip address
 no ip directed-broadcast
 no cdp enable
 xconnect 10.131.159.252 333 encapsulation mpls
!
interface FastEthernet3/0/0
 no ip address
 no ip directed-broadcast
 shutdown
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.244 0.0.0.3 area 0
```

```
 network 10.131.191.228 0.0.0.3 area 0
 network 10.131.191.232 0.0.0.3 area 0
 network 10.131.191.252 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip classless

end
```

デバイス P1 の設定

```
version 12.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname p1
!
enable password lab
!
ip cef
mpls label protocol ldp
mpls ldp logging neighbor-changes
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers frequency 0
mpls ldp discovery targeted-hello accept
!
interface Loopback0
 ip address 10.131.191.251 255.255.255.255
 no ip directed-broadcast
!
interface FastEthernet0/0/0
 ip address 10.131.191.229 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.226 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.224 0.0.0.3 area 0
 network 10.131.191.228 0.0.0.3 area 0
 network 10.131.191.251 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
end
```

デバイス P2 の設定

```
version 12.0
hostname p2
!
ip cef
mpls label protocol ldp
mpls ldp logging neighbor-changes
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers frequency 0
```

```
mpls ldp discovery directed-hello accept
!
!
interface Loopback0
 ip address 10.131.159.251 255.255.255.255
 no ip directed-broadcast
!
interface FastEthernet0/0/0
 ip address 10.131.159.229 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.225 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.224 0.0.0.3 area 0
 network 10.131.159.228 0.0.0.3 area 0
 network 10.131.159.251 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
end
```

デバイス PE2 の設定

```
version 12.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname pe2
!
logging snmp-authfail
enable password lab
!
clock timezone EST -5
ip subnet-zero
ip cef
no ip domain-lookup
mpls label protocol ldp
mpls ldp logging neighbor-changes
mpls ldp explicit-null
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers frequency 0
tag-switching tdp discovery directed-hello accept
frame-relay switching
!
!
interface Loopback0
 ip address 10.131.159.252 255.255.255.255
 no ip directed-broadcast
!
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 10.131.191.252
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng path-option 5 explicit name as1pe-long-path
!
interface FastEthernet0/0/0
```

```
 ip address 10.131.159.230 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet1/0/0
 ip address 10.131.159.245 255.255.255.255
 no ip directed-broadcast
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface FastEthernet2/0/0
 no ip address
 no ip directed-broadcast
 no cdp enable
 xconnect 10.131.191.252 333 encapsulation mpls
!
interface FastEthernet3/0/0
 no ip address
 no ip directed-broadcast
!
interface Serial4/0/0
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial5/0/0
 no ip address
 no ip directed-broadcast
 shutdown
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.122.0 0.0.0.3 area 0
 network 10.131.159.228 0.0.0.3 area 0
 network 10.131.159.232 0.0.0.3 area 0
 network 10.131.159.244 0.0.0.3 area 0
 network 10.131.159.252 0.0.0.0 area 0
!
ip classless
!
!
ip explicit-path name as1pe-long-path enable
 next-address 10.131.159.229
 next-address 10.131.159.226
 next-address 10.131.191.230
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password lab
 login
!
end
```

デバイス CE2 の設定

```
version 12.0
!
hostname ce2
!
```

```
enable password lab
!
interface Loopback0
 ip address 10.131.159.253 255.255.255.255
 no ip directed-broadcast
!
interface FastEthernet2/0/0
 ip address 10.0.0.2 255.255.255.255
 no ip directed-broadcast
 no keepalive
 no cdp enable
!
end
```

LSP が正しく設定されているかどうかの確認

show mpls forwarding-table コマンドは、トンネル 1 がマルチプロトコル ラベル スイッチング（MPLS）転送テーブルにあることを示しています。

```
Device# show mpls forwarding-table 10.131.159.252

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
22     19
     [T] 10.131.159.252/32 0          Tu1
        point2point
[T]     Forwarding through a TSP tunnel.
        View additional tagging info with the 'detail' option
```

PE1 で入力した **show mpls traffic-eng tunnels tunnel 1** コマンドは、トンネル 1 に関する情報を表示し、トンネル 1 が外部ラベル 22 を設定してパケットを転送していることを確認します。

```
Device# show mpls traffic-eng tunnels tunnel 1

Name: PE1_t1                           (Tunnel1) Destination: 10.131.159.251
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, type dynamic (Basis for Setup, path weight 20)
  Config Parameters:
    Bandwidth: 512      kbps (Global)  Priority: 2  2   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 512      bw-based
    auto-bw: disabled
  Active Path Option Parameters:
    State: dynamic path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  InLabel  :  -
  OutLabel : FastEthernet0/0/0, 22
  RSVP Signalling Info:
       Src 10.131.191.252, Dst 10.131.159.251, Tun_Id 1, Tun_Instance 28
    RSVP Path Info:
      My Address: 10.131.191.230
      Explicit Route: 10.131.191.229 10.131.159.226 10.131.159.225 10.131.159.251
      Record   Route:   NONE
      Tspec: ave rate=512 kbits, burst=1000 bytes, peak rate=512 kbits
    RSVP Resv Info:
      Record   Route:   NONE
      Fspec: ave rate=512 kbits, burst=1000 bytes, peak rate=512 kbits
  Shortest Unconstrained Path Info:
    Path Weight: 20 (TE)
    Explicit Route: 10.131.191.230 10.131.191.229 10.131.159.226 10.131.159.225
                    10.131.159.251
  History:
    Tunnel:
      Time since created: 9 days, 14 hours, 12 minutes
      Time since path change: 2 minutes, 18 seconds
    Current LSP:
      Uptime: 2 minutes, 18 seconds
```

```
    Prior LSP:
      ID: path option 1 [3]
      Removal Trigger: tunnel shutdown
```

PE 1 で発行された **trace mpls** コマンドは、最も外側のラベルが 22 でスタック末尾のラベルが 19 であるパケットが PE1 から PE2 に転送されることを確認します。

```
Device# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.131.191.230 MRU 1496 [Labels: 22/19
 Exp: 0/0]
R 1 10.131.159.226 MRU 1504 [Labels: 19 Exp: 0] 40 ms
R 2 10.131.159.229 MRU 1504 [implicit-null] 28 ms
! 3 10.131.159.230 40 ms
```

感嘆符（!）で示されているように、PE2 に対する MPLS LSP traceroute は成功しています。

LSP 切断の検出

次の **show mpls ldp discovery** コマンドの出力に示されているように、デバイス PE1 と P2 の間に Label Distribution Protocol（LDP）ターゲット セッションが確立されています。

```
Device# show mpls ldp discovery

 Local LDP Identifier:
    10.131.191.252:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
        Tunnel1 (ldp): Targeted -> 10.131.159.251
    Targeted Hellos:
        10.131.191.252 -> 10.131.159.252 (ldp): active/passive, xmit/recv
            LDP Id: 10.131.159.252:0
        10.131.191.252 -> 10.131.159.251 (ldp): active, xmit/recv
            LDP Id: 10.131.159.251:0
```

P2 デバイスで、次のコマンドをグローバル コンフィギュレーション モードで入力します。

```
Device# no mpls ldp discovery targeted-hello accept
```

LDP 設定の変更により、トラフィック エンジニアリング（TE）トンネルのヘッドエンドとテールエンド間のターゲット LDP セッションがダウンします。P2 で学習された IPv4 プレフィックスのラベルは、PE1 にアドバタイズされません。したがって、P2 から到達可能なすべての IP プレフィックスには、PE1 から MPLS ではなく IP を経由する場合にだけ到達可能です。つまり、PE1 のトンネル 1 を経由したそれらのプレフィックス宛てのパケットは、P2 で IP スイッチングされます（これは望ましくありません）。

次の **show mpls ldp discovery** コマンドは、LDP ターゲット セッションがダウンしていることを示しています。

```
Device# show mpls ldp discovery

 Local LDP Identifier:
    10.131.191.252:0
    Discovery Sources:
    Interfaces:
       FastEthernet0/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
        Tunnel1 (ldp): Targeted -> 10.131.159.251
```

```
    Targeted Hellos:
        10.131.191.252 -> 10.131.159.252 (ldp): active/passive, xmit/recv
            LDP Id: 10.131.159.252:0
        10.131.191.252 -> 10.131.159.251 (ldp): active, xmit
```

PE1 デバイスで **show mpls forwarding-table** コマンドを入力します。次の表示は、LDP 設定が変更された結果、発信パケットが非タグ付きになったことを示しています。

```
Device# show mpls forwarding-table 10.131.159.252

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
22     Untagged[T]
 10.131.159.252/32 0          Tu1        point2point
[T]     Forwarding through a TSP tunnel.
        View additional tagging info with the 'detail' option
```

PE1 デバイスで **ping mpls** コマンドを入力すると、次のように表示されます。

```
Device# ping mpls ipv4 10.131.159.252/32 repeat 1
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
R
Success rate is 0 percent (0/1)
```

この **ping mpls** コマンドは失敗しています。R は、マルチプロトコル ラベル スイッチング（MPLS）エコー応答の送信元にルーティング エントリがあり、MPLS Forwarding Equivalence Class（FEC）がないことを示します。**ping mpls verbose** コマンドを入力すると、MPLS ラベル スイッチド パス（LSP）エコー応答の送信元アドレスと戻りコードが表示されます。応答デバイスに対する telnet とフォワーディングやラベルのテーブルの検査によって、問題を解決できる必要があります。切断はアップストリームデバイスで発生する可能性があるため、隣接するアップストリームデバイスも調べる必要があります。

```
Device# ping mpls ipv4 10.131.159.252/32 repeat 1 verbose
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
R   10.131.159.225, return code 6
Success rate is 0 percent (0/1)
```

または、LSP **traceroute** コマンドを使用して、切断の原因となったルータを特定します。次の例では、TTL の後続の値が 2 よりも大きい場合、同じデバイス（10.131.159.225）が応答し続けます。これは、TTL にかかわらず、MPLS エコー要求はそのデバイスによって処理され続けることを意味します。ラベル スタックの検査によって、P1 が最後のラベルをポップし、パケットを IP パケットとして P2 に転送することがわかります。これは、パケットが P2 によって処理され続ける理由を説明するものです。MPLS エコー要求パケットは、IP ヘッダーの宛先アドレスを使用して転送できません。これは、アドレスが 127/8 アドレスに設定されているためです。

```
Device# trace mpls ipv4 10.131.159.252/32 ttl 5
Tracing MPLS Label Switched Path to 10.131.159.252/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.131.191.230 MRU 1500 [Labels: 22 Exp: 0]
R 1 10.131.159.226 MRU 1504 [implicit-null] 40 ms
R 2 10.131.159.225 40 ms
```

```
R 3 10.131.159.225 40 ms
R 4 10.131.159.225 40 ms
R 5 10.131.159.225 40 ms
```

MPLS LSP traceroute でトラックされる非タグ付き：例

このトラブルシューティングの項では、MPLS LSP Traceroute を使用して、暗黙の Null でタグ付けされているパケットとタグ付けされていないパケットで発生する可能性のある問題を判別する方法を示します。

最後から 2 番めのホップのタグなし出力インターフェイスは、ラベル スイッチド パス（LSP）経由の IP パケットの転送に影響しません。これは、転送判断が最後から 2 番めのホップで着信ラベルを使用して行われるためです。タグなしのケースでは、Any Transport over Multiprotocol Label Switching（AToM）および MPLS バーチャル プライベート ネットワーク（VPN）トラフィックが最後から 2 番めのホップでドロップされます。

暗黙的ヌルのトラブルシューティング：例

次の例では、トンネル 1 はシャットダウンされ、Label Distribution Protocol（LDP）ラベルで構成されたラベル スイッチド パス（LSP）だけが確立されます。暗黙的ヌルは、P2 デバイスと PE2 デバイスの間でアドバタイズされます。PE1 デバイスで MPLS LSP traceroute を入力すると、次の結果が表示されます。

```
Device# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.131.191.230 MRU 1500 [Labels: 20 Exp: 0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 80 ms
R 2 10.131.159.229 MRU 1504 [implicit-null] 28 ms
! 3 10.131.159.230 40 ms
```

この出力には、パケットが暗黙的ヌル ラベル付きで P2 から PE2 に転送されていることが示されてます。アドレス 10.131.159.229 は、PE2 デバイスに対する P2 のファスト イーサネット 0/0/0 出力インターフェイス用に設定されています。

非タグ付きのトラブルシューティング：例

非タグ付きの例は、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）の問題の原因となる可能性がある内部ゲートウェイ プロトコル（IGP）ラベル スイッチド パス（LSP）に有効な設定です。

P2 デバイスで **show mpls forwarding-table** コマンドと **show mpls ldp discovery** コマンドを発行すると、ラベル配布プロトコル（LDP）は適切に設定されていることが示されます。

```
Device# show mpls forwarding-table 10.131.159.252

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
19     Pop tag     10.131.159.252/32 0          Et0/0      10.131.159.230
Device# show mpls ldp discovery
 Local LDP Identifier:
    10.131.159.251:0
```

```
    Discovery Sources:
    Interfaces:
        FastEthernet0/0/0 (ldp): xmit/recv
            LDP Id: 10.131.159.252:0
        FastEthernet1/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
```

show mpls ldp discovery コマンドの出力には、PE2 を P2 に接続するファスト イーサネット インターフェイス 0/0/0 がパケットを送受信していることが示されます。

ファスト イーサネット 0/0/0 に対して **no mpls ip** コマンドを入力すると、P2 デバイスと PE2 デバイス間の LDP セッションが確立されない可能性があります。PE デバイスで入力した **show mpls ldp discovery** コマンドは、PE2 デバイスとの MPLS LDP セッションがダウンしていることを示しています。

```
Device# show mpls ldp discovery

 Local LDP Identifier:
    10.131.159.251:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0/0 (ldp): xmit
        FastEthernet1/0/0 (ldp): xmit/recv
            LDP Id: 10.131.191.251:0
```

PE2 との MPLS LDP セッションがダウンすると、**show mpls forwarding-table** コマンドで示されるように、10.131.159.252 への LSP が非タグ付きになります。

```
Device# show mpls forwarding-table 10.131.159.252

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
19     Untagged
    10.131.159.252/32 864        Et0/0      10.131.159.230
```

非タグ付きの例では、次に示すように、MPLS LSP traceroute の応答に No Label のタグが付いたパケットが含まれます。

```
Device# trace mpls ipv4 10.131.159.252/32
Tracing MPLS Label Switched Path to 10.131.159.252/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
  0 10.131.191.230 MRU 1500 [Labels: 20 Exp: 0]
R 1 10.131.159.226 MRU 1500 [Labels: 19 Exp: 0] 80 ms
R 2 10.131.159.229 MRU 1504 [No Label] 28 ms
! 3 10.131.159.230 40 ms
```

MPLS LSP ping および traceroute で返される Q

Q 戻りコードは常に、パケットを送信できなかったことを意味します。この問題は、メモリの不足が原因で発生することがありますが、コマンドラインで入力された Forwarding Equivalence Class（FEC）情報に一致するラベル スイッチド パス（LSP）が見つからなかったために発生した可能性があります。

パケットが転送されなかった原因を判別する必要があります。このためには、ルーティング情報ベース（RIB）、転送情報ベース（FIB）、ラベル情報ベース（LIB）、および MPLS ラベル転送情報ベース（LFIB）を調べます。いずれかのルーティング/転送ベースに FEC のエントリがない場合に、Q が戻されます。

次の表に、MPLS エコー要求から Q が戻される場合のトラブルシューティングに使用できるコマンドのリストを示します。

表 *22*：*Q* のトラブルシューティング

データベース	コンテンツを表示するコマンド
ルーティング情報ベース	**show ip route**
ラベル情報ベースおよびMPLS転送情報ベース	**show mpls forwarding-table detail**

次に、戻される Q が示すように MPLS エコー要求が送信されない **ping mpls** コマンドの例を示します。

```
Device# ping mpls ipv4 10.0.0.1/32
Sending 5, 100-byte MPLS Echos to 10.0.0.1/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
QQQQQ
Success rate is 0 percent (0/5)
```

show mpls forwarding-table コマンドと **show ip route** コマンドは、アドレスがいずれかのルーティング テーブルにないことを示します。

```
Device# show mpls forwarding-table 10.0.0.1

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
Device# show ip route 10.0.0.1

% Subnet not in table
```

IPv4 アドレス（10.0.0.1）が LFIB または RIB ルーティング テーブルのいずれかにないため、MPLS エコー要求が送信されません。

IPv4 LDP LSP のロード バランシング

Internet Control Message Protocol（ICMP）ping または trace は、送信元デバイスからターゲット デバイスまでの 1 本のパスをたどります。ターゲットの IP アドレスへの複数の出力パスを検出するには、送信元デバイスからの IP パケットのラウンド ロビン ロード バランシングを使用します。

MPLS Ping と Traceroute の場合、ネットワークにターゲット デバイスへの複数のパスが存在するときに、Cisco デバイスはロード バランシングに IP ヘッダー内の送信元アドレスと宛先アドレスを使用します。MPLS のシスコの実装では、IP ペイロードの宛先アドレスをチェックしてロード バランシングを実行する場合があります（このチェックはプラットフォームによって異なります）。

ロード バランシング パスを確認するには、**ping mpls ipvr***ip-address address-mask***destination***address-start address-end address-increment* コマンドで 127.*z.y.x* /8 宛先アドレスを使用します。次の例は、同一の宛先までの複数のパスがたどられることを示しています。こ

れは、送信元デバイスとターゲットデバイス間でロードバランシングが発生することを示しています。

PE1 デバイスのファスト イーサネット インターフェイス 1/0/0 が動作していることを確認するには、PE1 デバイスで次のコマンドを入力します。

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface fastethernet 1/0/0
Device(config-if)# no shutdown
Device(config-if)# end
*Dec 31 19:14:10.034: %LINK-3-UPDOWN: Interface FastEthernet1/0/0, changed state to up
*Dec 31 19:14:11.054: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0/0,
changed state to upend
PE1#
*Dec 31 19:14:12.574: %SYS-5-CONFIG_I: Configured from console by console
*Dec 31 19:14:19.334: %OSPF-5-ADJCHG: Process 1, Nbr 10.131.159.252 on FastEthernet1/0/0
from LOADING to FULL, Loading Done
PE1#
```

次の **show mpls forwarding-table** コマンドは、プレフィックス 10.131.159.251/32 の発信インターフェイスとネクスト ホップを表示します。

```
Device# show mpls forwarding-table 10.131.159.251

Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
21     19          10.131.159.251/32 0          FE0/0/0 10.131.191.229
       20          10.131.159.251/32 0          FE1/0/0 10.131.159.245
```

宛先 UDP アドレスが 127.0.0.1 の 10.131.159.251/32 に対する次の **ping mpls** コマンドは、選択したパスのパス インデックスが 0 であることを示します。

```
Device# ping mpls ipv4
 10.131.159.251/32 destination
 127.0.0.1 repeat 1
Sending 1, 100-byte MPLS Echos to 10.131.159.251/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 40/40/40 ms
PE1#
*Dec 29 20:42:40.638: LSPV: Echo Request sent on IPV4 LSP, load_index 2,
pathindex 0
, size 100
*Dec 29 20:42:40.638: 46 00 00 64 00 00 40 00 FF 11 9D 03 0A 83 BF FC
*Dec 29 20:42:40.638: 7F 00 00 01 94 04 00 00 0D AF 0D AF 00 4C 14 70
*Dec 29 20:42:40.638: 00 01 00 00 01 02 00 00 1A 00 00 1C 00 00 00 01
*Dec 29 20:42:40.638: C3 9B 10 40 A3 6C 08 D4 00 00 00 00 00 00 00 00
*Dec 29 20:42:40.638: 00 01 00 09 00 01 00 05 0A 83 9F FB 20 00 03 00
*Dec 29 20:42:40.638: 13 01 AB CD AB CD AB CD AB CD AB CD AB CD AB CD
*Dec 29 20:42:40.638: AB CD AB CD
*Dec 29 20:42:40.678: LSPV: Echo packet received: src 10.131.159.225,
dst 10.131.191.252, size 74
*Dec 29 20:42:40.678: AA BB CC 00 98 01 AA BB CC 00 FC 01 08 00 45 C0
*Dec 29 20:42:40.678: 00 3C 32 D6 00 00 FD 11 15 37 0A 83 9F E1 0A 83
*Dec 29 20:42:40.678: BF FC 0D AF 0D AF 00 28 D1 85 00 01 00 00 02 02
*Dec 29 20:42:40.678: 03 00 1A 00 00 1C 00 00 00 01 C3 9B 10 40 A3 6C
*Dec 29 20:42:40.678: 08 D4 C3 9B 10 40 66 F5 C3 C8
```

宛先 UDP アドレスが 127.0.0.1 の 10.131.159.251/32 に対する次の **ping mpls** コマンドは、選択したパスのパス インデックスが 1 であることを示します。

```
Device# ping mpls ipv4 10.131.159.251/32 dest 127.0.0.1 repeat 1
```

```
Sending 1, 100-byte MPLS Echos to 10.131.159.251/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 40/40/40 ms
*Dec 29 20:43:09.518: LSPV: Echo Request sent on IPV4 LSP, load_index 13,
pathindex 1
, size 100
*Dec 29 20:43:09.518: 46 00 00 64 00 00 40 00 FF 11 9D 01 0A 83 BF FC
*Dec 29 20:43:09.518: 7F 00 00 03 94 04 00 00 0D AF 0D AF 00 4C 88 58
*Dec 29 20:43:09.518: 00 01 00 00 01 02 00 00 38 00 00 1D 00 00 00 01
*Dec 29 20:43:09.518: C3 9B 10 5D 84 B3 95 84 00 00 00 00 00 00 00 00
*Dec 29 20:43:09.518: 00 01 00 09 00 01 00 05 0A 83 9F FB 20 00 03 00
*Dec 29 20:43:09.518: 13 01 AB CD AB CD AB CD AB CD AB CD AB CD AB CD
*Dec 29 20:43:09.518: AB CD AB CD
*Dec 29 20:43:09.558: LSPV: Echo packet received: src 10.131.159.229,
dst 10.131.191.252, size 74
*Dec 29 20:43:09.558: AA BB CC 00 98 01 AA BB CC 00 FC 01 08 00 45 C0
*Dec 29 20:43:09.558: 00 3C 32 E9 00 00 FD 11 15 20 0A 83 9F E5 0A 83
*Dec 29 20:43:09.558: BF FC 0D AF 0D AF 00 28 D7 57 00 01 00 00 02 02
*Dec 29 20:43:09.558: 03 00 38 00 00 1D 00 00 00 01 C3 9B 10 5D 84 B3
*Dec 29 20:43:09.558: 95 84 C3 9B 10 5D 48 3D 50 78
```

選択された実際のパスを確認するには、**debug mpls lspv packet data** コマンドを使用します。

（注） ハッシュ アルゴリズムは非決定的です。したがって、**destination** キーワードに *address-start*、*address-end*、および *address-increment* 引数を指定しても、期待どおりの結果が得られない場合があります。

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
MPLS コマンド	『Cisco IOS Multiprotocol Label Switching Command Reference』
スイッチング サービス コマンド	『Cisco IOS IP Switching Command Reference』
MPLS VPN の概念と設定作業	『*MPLS: Layer 3 VPNs Configuration Guide*』（『*Multiprotocol Label Switching Configuration Guide Library*』の一部）

標準および RFC

標準/RFC	タイトル
draft-ietf-mpls-lsp-ping-03.txt	『*Detecting MPLS Data Plane Failures*』
draft-ietf-pwe3-vccv-01.txt	『*Pseudo-Wire (PW) Virtual Circuit Connection Verification (VCCV)*』
RFC 2113	*IP Router Alert Option*

シスコのテクニカル サポート

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

MPLS LSP ping、traceroute、AToM VCCV の機能情報

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

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

表 *23*：***MPLS LSP ping*、*traceroute*、*AToM VCCV*** の機能情報

機能名	リリース	機能情報
MPLS LSP ping、traceroute、AToM VCCV	12.0(27)S 12.2(28)SB 12.2(33)SXH Cisco IOS XE Release 2.3 Cisco IOS XE Release 3.5S	MPLS LSP ping を使用することで、IPv4 Label Distribution Protocol（LDP）プレフィックス、トラフィック エンジニアリング（TE）Forwarding Equivalence Class（FEC）、および Any Transport over MPLS（AToM）FEC のラベル スイッチド パス（LSP）接続をテストできます。MPLS LSP Traceroute を使用して、IPv4 LDP プレフィックスと TE トンネル FEC の LSP をトレースできます。AToM VCCV により、MPLS LSP ping を使用して AToM 仮想回線（VC）の疑似回線（PW）セクションをテストできます。 Cisco IOS Release 12.2(28)SB で、この機能が拡張されて Cisco 10000 シリーズ ルータをサポートするようになりました。 この機能が Cisco IOS Release 12.2(33)SXH および Cisco IOS XE Release 2.3 に統合されました。 Cisco IOS XE Release 3.5S では、Cisco ASR 903 ルータのサポートが追加されました。 次のコマンドが導入または変更されました。**debug mpls lspv**、**ping mpls**、**trace mpls**。

用語集

FEC：Forward Equivalence Class。転送のために同等に処理できるパケットのセット。したがって、1 つのラベルへのバインディングに適しています。たとえば、1 つのアドレス プレフィックス宛てのパケットのセットや任意のフローなどがあります。

フロー：一般に、一組のホスト間、または一組のホスト上にある一組のトランスポート プロトコル ポート間で転送されるパケットのセット。たとえば、同じ送信元アドレス、送信元ポート、宛先アドレス、および宛先ポートを持つパケットは、フローと見なされることがあります。

フローは、ネットワークの 2 つのエンドポイント間で（たとえば、ある LAN ステーションから別の LAN ステーションへ）転送されるデータのストリームでもあります。単一の回線上で複数のフローを転送できます。

フラグメンテーション：元のパケット サイズをサポートできないネットワーク メディアを介してパケットを送信するときに、パケットを小さい単位に分割するプロセス。

ICMP：Internet Control Message Protocol。エラーを報告し、IP パケット処理に関連するその他の情報を提供するネットワーク層インターネット プロトコル。RFC 792 に記載されています。

LFIB：Label Forwarding Information Base（ラベル転送情報ベース）。宛先および着信ラベルが発信インターフェイスおよびラベルに関連付けられている転送を管理するデータ構造および手段。

localhost：デバイスのホスト名を表す名前。localhost は、予約済みのループバック IP アドレス（127.0.0.1）を使用します。

LSP：Label Switched Path（ラベル スイッチド パス）。MPLS を使用してパケットを転送する 2 つのデバイス間の接続。

LSPV：Label Switched Path Verification。LSP ping サブプロセスであり、MPLS エコー要求とエコー応答を符号化および復号化し、MPLS エコー要求とエコー応答を送受信するために IP、MPLS、および AToM スイッチングとやり取りします。MPLS エコー要求発信元デバイスでは、対応するエコー応答が受信されていない未処理のエコー要求が格納されているデータベースを維持します。

MPLS ルータ アラート ラベル：MPLS ラベル 1。ルータ アラート ラベルを含む MPLS パケットは、処理のためにデバイスによって Route ルート プロセッサ（PR）の処理レベルにリダイレクトされます。これにより、これらのパケットはハードウェア ルーティング テーブルにおけるフォワーディング エラーを回避できます。

MRU：Maximum Receive Unit（最大受信ユニット）。LSP を介して転送できる、ラベル付きパケットの最大サイズ（バイト単位）。

MTU：Maximum Transmission Unit（最大伝送ユニット）。特定のインターフェイスで処理できる最大パケット サイズ（バイト単位）。

パント：ルータ アラートを含むパケットを処理のためにラインカードまたはインターフェイスからルート プロセッサ（RP）のレベル処理にリダイレクトします。

PW：pseudowire（疑似回線）。パケット スイッチド ネットワークを介して、エミュレートされた回線の重要な要素を、あるプロバイダー エッジ（PE）デバイスから別の PE デバイスに伝送するメカニズム。

RP：ルート プロセッサ。Cisco 7000 シリーズ ルータのプロセッサ モジュールで、CPU、システム ソフトウェア、およびデバイスで使用されるメモリ コンポーネントの大半が含まれます。スーパーバイザリ プロセッサと呼ばれることもあります。

RSVP：Resource Reservation Protocol。IP ネットワーク上でリソースの予約をサポートするためのプロトコル。IP エンド システム上で動作しているアプリケーションは、RSVP を使用して、受信するパケット ストリームの特性（帯域幅、ジッタ、最大バーストなど）を他のノードに示すことができます。RSVP は IPv6 に依存します。リソース予約設定プロトコルとも呼ばれます。

UDP：User Datagram Protocol。TCP/IP プロトコル スタックのコネクションレス型トランスポート層プロトコルです。UDP は、確認応答や配信保証なしでデータグラムを交換する単純なプロトコルです。エラー処理と再送信は、他のプロトコルで処理する必要があります。UDP は RFC 768 で定義されています。

第 5 章

MPLS EM - MPLS LSP マルチパス ツリー トレース

MPLS EM - MPLS LSP マルチパス ツリー トレース機能は、出力ルータと入力ルータ間でラベル スイッチドパス（LSP）の可能な等コストマルチパス（ECMP）ルーティングパスをすべて検出する手段を提供します。これらのパスは、検出後、マルチプロトコル ラベル スイッチング（MPLS）LSP ping または traceroute を使用して定期的に再テストできます。この機能は、IPv4 LSP のトレース用の MPLS LSP traceroute 機能に対する拡張です。

MPLS EM - MPLS LSP マルチパス ツリー トレース機能を使用して、IPv4 LSP のすべてのパスを検出できます。

MPLS EM - MPLS LSP マルチパス ツリー トレース機能の実装は、RFC 4379、『Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures』に基づいています。

MPLS LSP ping および traceroute の使用の詳細については、『MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV』フィーチャ モジュールを参照してください。

Cisco MPLS Embedded Management（EM）は、障害、設定、アカウンティング、パフォーマンス、セキュリティ（FCAPS）モデルに従ったMPLSベースのネットワークの開発、操作、アドミニストレーション、および管理を容易にする標準と付加価値サービスのセットです。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS EM - MPLS LSP マルチパス ツリー トレースの前提条件

MPLS EM—MPLS LSP マルチパス ツリー トレース機能を使用するための前提条件は次のとおりです。

- 『MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV』マニュアルで説明されている MPLS LSP ping または traceroute の概念と使用方法を理解している必要があります。
- ネットワーク内のルータでは、RFC 4379、『Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures』に基づく実装を使用している必要がある。
- MPLS ネットワークについて次のことを理解している必要があります。
  - トポロジ
  - ネットワーク内のリンクの数
  - LSP の予期される数と実際の数
- ラベル スイッチング、転送、ロード バランシング

MPLS EM - MPLS LSP マルチパス ツリー トレースの制約事項

- MPLS LSP ping および LSP traceroute 機能に適用されるすべての制約事項が MPLS EM - MPLS LSP マルチパス ツリー トレース機能にも適用されます。
  - MPLS LSP マルチパス ツリー トレース機能を使用して、AToM パケットが辿るパスをトレースすることはできません。MPLS LSP マルチパス ツリー トレース機能は、AToM ではサポートされません。（MPLS LSP ping は AToM でサポートされます。）ただし、

MPLS LSP マルチパス ツリー トレース機能を使用して、AToM によって使用される Interior Gateway Protocol（IGP）LSP をトラブルシューティングすることはできます。

- MPLS LSP マルチパス ツリー トレース機能を使用して、MPLS バーチャル プライベート ネットワーク（VPN）を検証またはトレースすることはできません。MPLS コア内のすべてのルータで『Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures』の RFC 4379 実装がサポートされていないかぎり、複数の LSP パスは検出されません。

- 存続可能時間（TTL）の非表示をサポートするネットワークでの MPLS LSP マルチパス ツリー トレースの動作は想定されていません。

MPLS EM - MPLS LSP マルチパス ツリー トレースに関する情報

MPLS LSP マルチパス ツリー トレースの概要

MPLS 展開の数が増えると、MPLS ネットワークで伝送されるトラフィック タイプの数が増えることがあります。また、MPLS ネットワーク内のラベル スイッチング ルータ（LSR）上のロード バランシングによって、ターゲット ルータに MPLS トラフィックを伝送するための代替パスが提供されます。サービス プロバイダーがサービスを提供するには、LSP を監視し、MPLS 転送の問題を迅速に特定できることが不可欠です。

MPLS EM—MPLS LSP マルチパス ツリー トレース機能がリリースされる前は、プロバイダー エッジ（PE）ルータ間のすべてのパスを自動的に検出する手段はありませんでした。PE 間の転送の問題のトラブルシューティングには手間がかかりました。

MPLS EM—MPLS LSP マルチパス ツリー トレース機能のリリースにより、中継ルータで IPv4 ロード バランシングを使用するマルチベンダー ネットワーク内で入力 PE ルータから出力 PE ルータへのすべてのパスを自動的に検出できるようになっています。PE 間のパスが検出されたら、MPLS LSP ping と MPLS LSP traceroute を使用して、定期的にこれらのパスをテストします。

MPLS EM—MPLS LSP マルチパス ツリー トレース機能を使用するには、RFC 4379 に基づく Cisco RFC 準拠の実装が要件となります。RFC 379 をサポートする Cisco ソフトウェア リリースを使用していない場合、MPLS LSP マルチパス ツリー トレースはすべての PE から PE へのパスの検出操作を行いません。

MPLS LSP マルチパス ツリー トレースによる IPv4 ロード バランシング パスの検出

中継ルータでの IPv4 ロード バランシングは、着信ラベル スタックと、IP ヘッダー内の送信元および宛先アドレスに基づきます。出ラベル スタックと IP ヘッダー送信元アドレスは、トレースされる各ブランチに対して一定です。

送信元 LSR に対して MPLS LSP マルチパス ツリー トレースを実行する場合、ルータは IP ヘッダーの宛先アドレスのセットを探してすべての可能な出力パスを使用する必要があります。送信元 LSR は、MPLS エコー要求で中継ルータにビットマップを送信することでパス ディスカバリを開始します。中継ルータは、エコー応答のダウンストリーム マップ（DS Map）内のビットマップのサブセットを含む MPLS エコー要求で情報を返します。送信元ルータは、エコー応答の情報を使用して次のルータに問い合わせることができます。送信元ルータは、パス上のすべてのルータに共通の 1 つのビットマップ設定が見つかるまで、後続の各ルータに問い合わせます。共通ビットを検出するため、ルータは TTL 有効期限を使用して、ルータに問い合わせます。

たとえば、送信元ルータで次のコマンドを入力することでパス ディスカバリを開始できます。

```
Router# trace mpls multipath ipv4 10.131.101.129/32 hashkey ipv4 bitmap 16
```

このコマンドは、ターゲット ルータの IP アドレスを 10.131.101.192 255.255.255.255 に設定し、次のように設定します。

- デフォルト ハッシュ キー タイプを 8 に設定します。これにより、IPv4 アドレス プレフィックスとビット マスク アドレス セットがエコー応答の DS マップで返されるように要求されます。
- ビットマップ サイズを 16 に設定します。このことは、MPLS LSP マルチパス ツリー トレースが送信元ルータとターゲット ルータ間の LSP のすべてのパスの検出で、16 個のアドレス（127.0.0.1 から開始）を使用することを意味します。

trace mpls multipath ipv4 10.131.101.129/32 コマンドを入力すると、MPLS LSP マルチパス ツリー トレースはデフォルト ハッシュ タイプ 8 または IP v4 とデフォルト ビットマップ サイズ 32 を使用します。ビットマップ サイズの選択は、ネットワーク内のルータの台数によって決まります。ルートの数が膨大な場合は、大きなビットマップ サイズを選択する必要があります。

マルチパス LSP ツリー トレースを処理するルータによって送信されるエコー応答戻りコード

以下の表で、マルチパス LSP ツリー トレース パケットを処理しているルータが、要求の失敗または成功について送信者に返す文字について説明します。

表 *24*：エコー応答の戻りコード

出力コード	エコーの戻りコード	意味
ピリオド（.）	—	ターゲット ルータが応答する前にタイムアウトが発生した。
x	0	戻りコードなし。
M	1	不正な形式の要求。
m	2	サポートされていないタイプ、長さ、値（TLV）。

出力コード	エコーの戻りコード	意味
!	3	成功。
F	4	Forward Equivalence Class（FEC）マッピングはない。
D	5	DS マップの不一致。
R	6	ターゲットでないダウンストリーム ルータ。
U	7	予備。
L	8	ラベル付けされた出力インターフェイス。
B	9	ラベル付けされていない出力インターフェイス。
f	10	FEC の不一致。
N	11	ラベル エントリなし。
P	12	受信インターフェイスのラベル プロトコルなし。
p	13	LSP の終了が不完全。
X	unknown	未定義の戻りコード。

MPLS 組み込み管理設定

ping mpls、**trace mpls**、または **trace mpls multipath** コマンドを使用する前に、ネットワーク内のすべての受信側ルータが認識できる形式で MPLS エコー パケットを符号化および復号化するようにルータが設定されていることを確認することを検討してください。

バージョン 3（draft-ietf-mpls-ping-03）よりも後の LSP ping ドラフトでは、多数の TLV 形式の変更が行われていますが、異なるドラフトに基づく実装は、適切に相互運用されない可能性があります。

新しいシスコの実装がドラフト バージョン 3 のシスコの実装やシスコ以外の実装と相互運用できるようにするには、グローバル コンフィギュレーション モード（MPLS OAM コンフィギュレーション）を使用することで、エコー パケットをドラフト バージョン 3 の実装によって指定される形式で符号化および復号化します。

特に設定がなければ、シスコの実装では、Internet Engineering Task Force（IETF）の実装がベースにしているバージョンを想定して、エコー要求の符号化とデコードを行います。

以前のリビジョン 1 および 3 イメージとのシームレスな相互運用を可能にするために、MPLS 操作、管理、メンテナンス（OAM）コンフィギュレーション モード パラメータを使用して、リビジョン4イメージのデフォルトの動作がネットワーク内でリビジョン1またはリビジョン3イメージに準拠または互換になるように強制できます。

TLV バージョンの問題によって発生するエラーが応答ルータから報告されないようにするには、コア内のすべてのルータを設定する必要があります。同じドラフト バージョンでMPLSエコー パケットを符号化およびデコードしてください。たとえば、ネットワークでRFC 4379（シスコ リビジョン 4）の実装が実行され、1 つのルータがバージョン 3（シスコ リビジョン 3）にだけ対応している場合は、ネットワーク内のすべてのルータをリビジョン 3 モードで動作するように設定します。

シスコ リビジョン4がデフォルト バージョンです。デフォルト バージョンは、ルータ上のイメージによってサポートされる最新の LSP ping バージョンです。

MPLS EM - MPLS LSP マルチパス ツリー トレースの設定方法

MPLS エコー パケットのデフォルトの動作のカスタマイズ

MPLS エコー パケットのデフォルトの動作をカスタマイズするには、次の作業を実行します。『Detecting MPLS Data Plane Failures』（RFC 4379）の新しいバージョンを、このドラフトの前のバージョンを実行しているネットワークに導入できるようにするには、デフォルトのエコー パケット エンコードおよびデコードの動作をカスタマイズする必要があります。

はじめる前に

MPLS LSP マルチパス ツリー トレースを使用するには、RFC 4379（リビジョン 4）が必要です。

手順の概要

1. **enable**
2. **configure terminal**
3. **mpls oam**
4. **echo revision** {**3** | **4**}
5. [**no**] **echo vendor-extension**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mpls oam** 例： `Router(config)# mpls oam`	MPLS OAM コンフィギュレーション モードを開始し、エコー パケットのデフォルトの動作をカスタマイズします。

ステップ 4	**echo revision** {**3** | **4**}
	
	例：
	
	`Router(config-mpls)# echo revision 4`
		エコー パケットのデフォルトの動作をカスタマイズします。
		
		• **revision** キーワードで、次のいずれかにエコー パケット属性を設定します。
		  • **3** = draft-ietf-mpls-ping-03（リビジョン 2）
		  • **4** = RFC 4379 準拠（デフォルト）
		
		（注） MPLS LSP マルチパス ツリー トレース機能を使用するには、リビジョン 4 に設定する必要があります。

ステップ 5	[**no**] **echo vendor-extension** 例： `Router(config-mpls)# echo vendor-extension`	エコー パケットのデフォルトの動作をカスタマイズします。 • **vendor-extension** キーワードは、エコー パケットで TLV のシスコ固有の拡張を送信します。 • コマンドの **no** 形式では、別のベンダーの非準拠実装でサポートされない可能性のあるシスコ ベンダーの拡張 TLV をディセーブルにできます。 ルータのデフォルトは **echo vendor-extension** です。
ステップ 6	**end** 例： `Router(config-mpls)# end`	特権 EXEC モードに戻ります。

MPLS LSP マルチパス ツリー トレースの設定

MPLS マルチパス LSP traceroute を設定するには、次の作業を実行します。この作業は、出力ルータから入力ルータへのすべての LSP を検出するのに役立ちます。

はじめる前に

draft-ietf-mpls-lsp-ping-11 に基づく Cisco LSP ping または traceroute の実装では、MPLS エコー要求の送信者の形式を検出できる場合があります。ただし、エコー要求またはエコー応答にシスコ拡張 TLV が含まれていない場合もあります。不正な TLV 形式を想定してエコー パケットがデコードされるケースによる複雑さを回避するには、ネットワーク内のすべてのルータを同じモードで動作するように設定します。

MPLS LSP マルチパス ツリー トレースを成功させるには、使用するルータ内での実装が、すべてのコア ルータ上で RFC 4379 をサポートする必要があります。

ネットワーク内のすべてのルータで RFC-4379 がサポートされている一方、シスコのベンダー TLV を正しく処理できない別のベンダーの実装が存在する場合は、RFC 準拠またはより新しい設定がサポートされるルータに、シスコ ベンダー TLV 拡張をディセーブルにするコマンドが含まれている必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **mpls oam**
4. **echo revision 4**
5. [**no**] **echo vendor-extension**
6. **end**
7. **trace mpls multipath ipv4***destination-ip-address*/*destinationmask-length*
8. **debug mpls lspv multipath**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**mpls oam** 例： `Router(config)# mpls oam`	MPLS OAM 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	**echo revision 4** 例： `Router(config-mpls)# echo revision 4`	エコーパケットのデフォルトの動作をカスタマイズします。 • **revision 4** キーワードでは、エコー パケット属性をデフォルトのリビジョン 4（RFC 4379 準拠）を設定します。 （注） MPLS LSP マルチパス ツリー トレース機能を使用するには、リビジョン 4 に設定する必要があります。
ステップ 5	[**no**] **echo vendor-extension** 例： `Router(config-mpls) echo vendor-extension`	（任意）エコーパケットのデフォルトの動作をカスタマイズします。 • **vendor-extension** キーワードは、エコーパケットで TLV のシスコ固有の拡張を送信します。 • コマンドの **no** 形式では、別のベンダーの非準拠実装でサポートされない可能性のあるシスコベンダーの拡張 TLV をディセーブルにできます。 ルータのデフォルトは **echo vendor-extension** です。
ステップ 6	**end** 例： `Router(config-mpls)# end`	特権 EXEC モードに戻ります。
ステップ 7	**trace mpls multipath ipv4***destination-ip-address*/*destinationmask-length* 例： `Router# trace mpls multipath ipv4 10.131.161.251/32`	出力ルータから入力ルータへのすべての LSP を検出します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-ip-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。

	コマンドまたはアクション	目的
ステップ 8	**debug mpls lspv multipath** 例： `Router# debug mpls lspv multipath`	MPLS LSP マルチパス ツリー トレース機能に関連するマルチパス情報を表示します。

MPLS LSP マルチパス ツリー トレースを使用した IPv4 ロード バランシング パスの検出

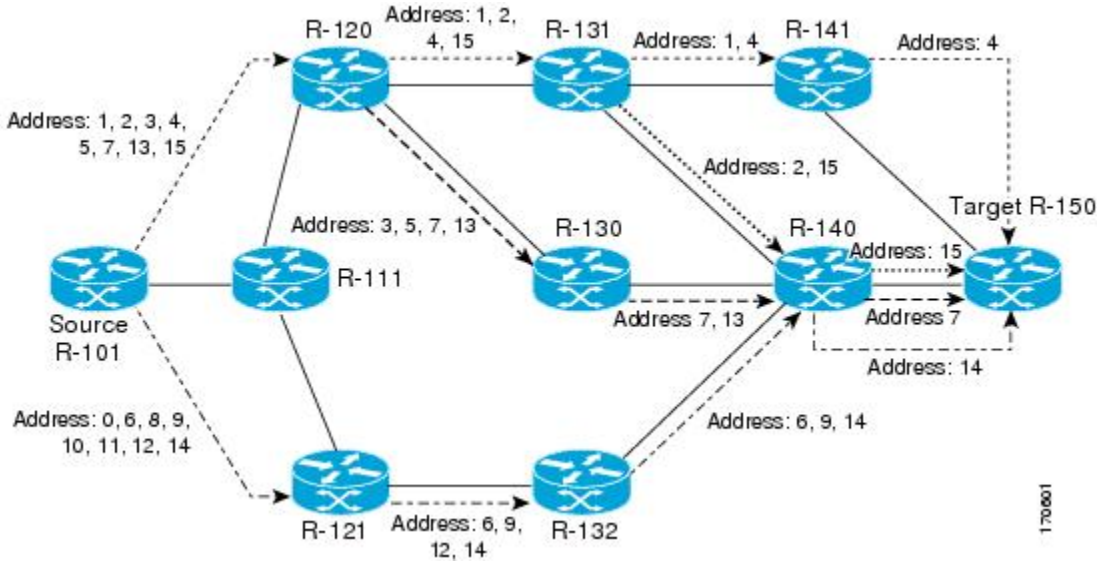
MPLS LSP マルチパス ツリー トレースを使用して IPv4 ロードバランシング パスを検出するには、次の作業を実行します。

Cisco ルータでは、着信ラベル スタックと、IP ヘッダー内の送信元および宛先アドレスに基づいて MPLS パケットのロード バランシングを行います。出ラベル スタックと IP ヘッダー送信元アドレスは、トレースされる各パスに対して一定です。ルータは IP ヘッダーの宛先アドレスのセットを探してすべての可能な出力パスを使用する必要があります。それには、127.x.y.z/8 アドレス空間の網羅的な検索が必要になる場合があります。送信元 LSR からターゲットまたは宛先 LSR へのすべてのパスを MPLS LSP マルチパス ツリー トレースで検出したあとで、MPLS LSP traceroute を使用してこれらのパスを監視できます。

以下の図に、MPLS LSP マルチパス ツリー トレースによってサンプル ネットワーク内の LSP パスがどのように検出されるかを示します。以下の図では、ビットマップ サイズは 16 で、番号 0 ～ 15 は、MPLS LSP マルチパス ツリー トレースが送信元 LSR R-101 からターゲット LSR R-150 へのすべてのパスの検出に使用するビットマップ化されたアドレスを表します。以下の図に、**trace**

mpls multipath コマンドによってサンプル ネットワーク内のすべての LSP パスがどのように検出されるかを示します。

図 *14*：サンプル ネットワークでの *MPLS LSP* マルチパス ツリー トレースによるパス ディスカバリ



手順の概要

1. **enable**
2. **configure terminal**
3. **mpls oam**
4. **echo revision 4**
5. **end**
6. **trace mpls multipath ipv4***destination-address*/*destination-mask-length***hashkey ipv4 bitmap***bitmap-size*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。

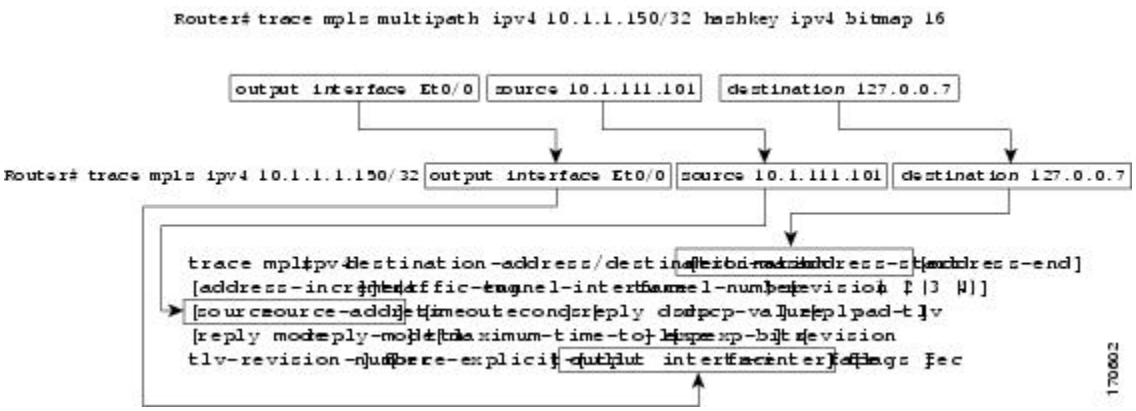
	コマンドまたはアクション	目的
ステップ 3	**mpls oam** 例： `Router(config)# mpls oam`	MPLS OAM コンフィギュレーション モードを開始し、エコー パケット属性をリビジョン 4（RFC 4379 準拠）に設定します。
ステップ 4	**echo revision 4** 例： `Router(config-mpls)# echo revision 4`	エコーパケットのデフォルトの動作をカスタマイズします。 • **revision 4** キーワードでは、エコー パケット属性をデフォルトのリビジョン 4（RFC 4379 準拠）を設定します。 （注） MPLS LSP マルチパス ツリー トレース機能を使用するには、リビジョン 4 に設定する必要があります。
ステップ 5	**end** 例： `Router(config-mpls)# end`	特権 EXEC モードに戻ります。
ステップ 6	**trace mpls multipath ipv4***destination-address*/*destination-mask-length***hashkey ipv4 bitmap***bitmap-size* 例： `Router# trace mpls multipath ipv4 10.131.161.251/32 hashkey ipv4 bitmap 16`	出力ルータから入力ルータへのすべての MPLS LSP を検出します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **hashkey ipv4** キーワードは、ハッシュキー タイプを IPv4 アドレスに設定します。 • **bitmap***bitmap-size* キーワードと引数では、マルチパス ディスカバリのビットマップ サイズを設定します。

MPLS LSP traceroute を使用した MPLS LSP マルチパス ツリー トレースで検出された LSP パスのモニタ

MPLS LSP マルチパス ツリー トレースにより検出された LSP パスを、MPLS LSP traceroute を使用して監視するには、次の作業を実行します。出力を **trace mpls multipath** コマンドから直接取得し、定期的に **trace mpls** コマンドに追加して、パスがまだ動作していることを検証できます。

次の図に、**trace mpls multipath** コマンドと **trace mpls** コマンドの出力の対応を示します。

図 *15*：*trace mpls multipath* コマンド出力と *trace mpls* コマンドの対応



MPLS LSP マルチパス ツリー トレースで検出する各パスをこの方法で定期的にテストして、ネットワーク内の LSP パスを監視できます。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length***hashkey ipv4 bitmap***bitmap-size*
3. **trace mpls ipv4***destination-address*/*destination-mask-length* [**output interface***tx-interface*] [**source***source-address*] [**destination***address-start*
4. **exit**

手順の詳細

ステップ 1 **enable**

このコマンドを使用して、特権EXECモードをイネーブルにします。パスワードを入力します（要求された場合）。次に例を示します。

例：

```
Router> enable
Router#
```

ステップ 2 **trace mpls multipath ipv4***destination-address*/*destination-mask-length***hashkey ipv4 bitmap***bitmap-size*

このコマンドを使用して、出力ルータから入力ルータへのすべての MPLS LSP を検出します。次に例を示します。

例：

```
Router# trace mpls multipath ipv4 10.1.1.150/32 hashkey ipv4 bitmap 16

Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 468 ms
```

例の **trace mpls multipath ipv4** コマンドの出力は、MPLS LSP マルチパス ツリー トレースによるパス ディスカバリの結果を示します。この例では、コマンドはビットマップ サイズを 16 に設定します。パス ディスカバリは、プレフィックスおよびマスク 10.1.1.150/32 で送信元ルータからターゲット ルータへの LSP パスを探すときに、16 のビットマップ化されたアドレスを使用した MPLS LSP マルチパス ツリー トレースによって開始されます。MPLS LSP マルチパス ツリー トレース機能は、127.0.0.1 の 127.x.y.z/8 アドレス空間を使用して開始します。

ステップ 3 **trace mpls ipv4***destination-address*/*destination-mask-length* [**output interface***tx-interface*] [**source***source-address*] [**destination***address-start*

このコマンドを使用して、**trace mpls multipath ipv4** コマンドの入力時に検出されたパスがまだ動作していることを検証します。たとえば、ステップ 2 の **trace mpls multipath ipv4** コマンドのパス 0 の出力は次のとおりです。

例：

```
output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
```

パス 0 の出力を **trace mpls** コマンドに入力した場合は、次の結果が表示されます。

例：

```
Router# trace mpls ipv4 10.1.1.150/32 output interface Fe0/0/0 source 10.1.111.101 destination
127.0.0.0

Tracing MPLS Label Switched Path to 10.1.1.150/32, 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 label entry,
  '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 10.1.111.101 MRU 1500 [Labels: 33 Exp: 0]
L 1 10.1.111.111 MRU 1500 [Labels: 34 Exp: 0] 40 ms
L 2 10.2.121.121 MRU 1500 [Labels: 34 Exp: 0] 32 ms
L 3 10.3.132.132 MRU 1500 [Labels: 32 Exp: 0] 16 ms
L 4 10.4.140.240 MRU 1504 [Labels: implicit-null Exp: 0] 20 ms
! 5 10.5.150.50 20 ms
```

出力を **trace mpls multipath** コマンドから直接取得し、定期的に **trace mpls** コマンドに追加して、パスがまだ動作していることを検証できます（上記の図を参照）。

ステップ 4 **exit**

このコマンドを使用して、ユーザ EXEC モードに戻ります。次に例を示します。

例：

```
Router# exit
Router>
```

DSCP を使用した、エコー応答における特定のサービス クラスの要求

応答 Diffserv コード ポイント（DSCP）オプションを使用して、エコー応答で特定のサービス クラス（CoS）を要求できます。

応答 DSCP オプションは、IETF draft-ietf-mpls-lsp-ping-03.txt の試験モードでサポートされます。シスコは、応答 TOS TLV を使用するのではなく、応答 DSCP オプションのベンダー固有の拡張を実装しました。応答 TOS TLV は、IETF draft-ietf-mpls-lsp-ping-11.txt の **reply dscp** コマンドと同じ目的を果たします。このドラフトは、応答 DSCP を制御するための標準化された方法を示します。

（注） RFC 4379 よりも前のバージョンでは、シスコは応答 DSCP オプションをシスコのベンダー拡張 TLV を使用した試験的な機能として実装しました。ルータがドラフト バージョン 3 の実装の MPLS エコー パケットを符号化するように設定されている場合は、ドラフト バージョン 8 で定義された応答 TOS TLV の代わりに、シスコのベンダー拡張 TLV が使用されます。

DSCP を使用してエコー応答における特定の CoS を要求するには、次の手順を実行します。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**reply dscp***dscp-value*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**reply dscp***dscp-value*] 例： `Router# trace mpls multipath ipv4 10.131.191.252/32 reply dscp 50`	入力ルータから出力ルータへのすべての MPLS LSP を検出し、エコー応答の DSCP 値を制御します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **reply dscp***dscp-value* キーワードと引数は、エコー応答の DSCP 値です。応答 TOS TLV は、IETF draft-ietf-mpls-lsp-ping-11.txt の **reply dscp** コマンドと同じ目的を果たします。 （注） DSCP 値を指定するには、**reply dscp***dscp-value* キーワードと引数を入力する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS エコー要求に対する応答ルータの応答方法の制御

この項では、応答ルータが MPLS エコー要求に応答する方法の制御に関する情報と手順について説明します。エコー要求応答の応答モードを設定する前に、次の情報を理解する必要があります。

MPLS LSP マルチパス ツリー トレースのエコー要求に対する応答モード

応答モードでは、応答ルータが **trace mpls multipath** コマンドによって送信された MPLS エコー要求に応答する方法を制御します。エコー要求パケットには、次の 2 つの応答モードがあります。

- ipv4：IPv4 User Datagram Protocol（UDP）パケットで応答する（デフォルト）。
- router-alert：ルータ アラートを含む IPv4 UDP パケットで応答する

（注） 見逃しを防ぐために ipv4 および router-alert 応答モードを相互に使用します。ipv4 モードを介して応答を受信できない場合は、router-alert reply 応答モードでテストを送信します。両方のモードで失敗する場合は、リターン パスに何か問題があります。問題は、不適切な ToS 設定が原因の可能性があります。

IPv4 UDP 応答モード：IPv4 UDP 応答モードは、LSP の完全性を定期的にポーリングする場合に、**trace mpls multipath** コマンドで使用される最も一般的な応答モードです。このオプションは、パケットが IP ホップと MPLS ホップのいずれを通過して MPLS エコー要求の送信元に到達するかを明示的に制御するものではありません。**reply mode ipv4** キーワードを使用した場合に、送信元（ヘッドエンド）ルータが MPLS エコー要求に対する応答を受信できないときは、**reply mode router-alert** キーワードを使用します。

router-alert 応答モード：router-alert 応答モードを使用すると、ルータ アラート オプションが IP ヘッダーに追加されます。IP ヘッダーに IP ルータ アラート オプションを含む IP パケット、または最も外側のラベルとしてルータ アラート ラベルを含む MPLS パケットがルータに到達すると、ルータはパケットを処理するためにルート プロセッサ（RP）プロセス レベルにパント（リダイレクト）します。これにより、各中間ルータの RP は宛先に戻るときに各中間ホップでパケットを明確に処理します。これにより、ハードウェアとラインカード フォワーディングの不整合が回避されます。router-alert 応答モードは、各ホップで応答にプロセスレベルの RP 処理が必要となるため、IPv4 モードよりも低速になります。

以下の表に、発信パケットが IP パケットまたは MPLS パケットの場合に、IP ルータ アラートを備えた着信 IP パケットがルータ スイッチング パス プロセスによってどのように処理されるかを示します。この表には、発信パケットが IP パケットまたは MPLS パケットの場合に、ルータ アラート オプションを使用した MPLS パケットがルータ スイッチング パス プロセスによってどのように処理されるかも示しています。

表 *25*： パス プロセスによる *IP* および *MPLS* ルータ アラート パケットの処理

着信パケット	発信パケット	通常のスイッチング アクション	プロセス スイッチング アクション
IP パケット：IP ヘッダーにルータ アラート オプションが含まれる	IP パケット：IP ヘッダーにルータ アラート オプションが含まれる	IP ヘッダーにルータ アラート オプションが含まれていると、パケットはプロセス スイッチング パスにパントされる。	パケットをそのまま転送する。
	MPLS パケット		パケットをそのまま転送する。
MPLSパケット：最も外側のラベルにルータ アラートが含まれる	IP パケット：IP ヘッダーにルータ アラート オプションが含まれる	ルータ アラート ラベルが最も外側のラベルである場合、パケットはプロセス スイッチング パスにパントされる。	最も外側のルータ アラート ラベルを削除し、パケットを IP パケットとして転送する。
	MPLSパケット：最も外側のラベルにルータ アラートが含まれる		最も外側のルータ アラート ラベルを保持し、MPLS パケットを転送する。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length***reply mode** {**ipv4** | **router-alert**}
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length***reply mode** {**ipv4**	**router-alert**}	入力ルータから出力ルータへのすべての MPLS LSP を検出し、応答モードを指定します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。

	コマンドまたはアクション	目的
	例： `Router# trace mpls multipath ipv4 10.131.191.252/32 reply mode router-alert`	• *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **reply mode** キーワードでは、応答モードを指定するために次のいずれかのキーワードを入力する必要があります。 　• **ipv4** キーワード：IPv4 UDP パケットで応答します（デフォルト）。 　• **router-alert** キーワード：ルータ アラートを含む IPv4 UDP パケットで応答します。 （注）　応答モードを指定するには、**ipv4** キーワードまたは **router-alert** キーワードとともに **reply mode** キーワードを入力する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP マルチパス ツリー トレースのためにルータから発信されるエコー パケットの出力インターフェイスの指定

MPLS LSP マルチパス ツリー トレース機能のためにルータから発信されるエコー パケットの出力インターフェイスを指定するには、次の作業を実行します。この作業で、特定のインターフェイスを介して到達可能な LSP をテストできます。

エコー要求出力インターフェイス コントロール：エコーパケットがルータから発信されるときに経由するインターフェイスを制御できます。パス出力情報は、LSP ping と traceroute への入力として使用されます。

エコー要求の出力インターフェイス制御機能を使用すると、LSP の詳細なデバッグや評価を行うパスをエコー パケットが通過することを強制できます。この機能は、PE ルータが MPLS クラウドに接続し、切断されたリンクがある場合に役立ちます。特定のリンクを介してトラフィックを誘導できます。この機能は、ネットワークの問題のトラブルシューティングにも役立ちます。

MPLS LSP マルチパス ツリー トレースのためにルータから発信されるエコーパケットの出力インターフェイスの指定

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**output interface***tx-interface*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**output interface***tx-interface*] 例： `Router# trace mpls multipath ipv4 10.131.159.251/32 output interface fastethernet0/0/0`	入力ルータから出力ルータへのすべての MPLS LSP を検出し、エコー パケットがルータから発信されるときに通過するインターフェイスを指定します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **output interface***tx-interface* キーワードおよび引数は、MPLS エコー要求に対する出力インターフェイスを指定します。 （注） **output interface** キーワードを指定する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP マルチパス ツリー トレースの MPLS エコー要求パケット送信ペースの設定

MPLS LSP マルチパス ツリー トレース機能の MPLS エコー要求パケット送信ペースを設定するには、次の作業を実行します。エコー要求トラフィック ペーシングを使用すると、受信側ルータがパケットをドロップしないように、パケットの送信ペースを設定できます。ネットワーク上のトラフィックが大量である場合は、受信側ルータによってパケットがドロップされないように、間隔のサイズを増やすことができます。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**interval***milliseconds*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**interval***milliseconds*] 例： `Router# trace mpls multipath ipv4 10.131.159.251/32 interval 100`	出力ルータから入力ルータへのすべての MPLS LSP を検出し、連続する MPLS エコー要求間の時間をミリ秒単位で設定します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **interval***milliseconds* キーワードと引数では、連続する MPLS エコー要求間の時間をミリ秒単位で設定します。デフォルトは 0 ミリ秒です。 （注） パケットの送信をペーシングするには、**interval** キーワードを指定する必要があります。

	コマンドまたはアクション	目的
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP マルチパス ツリー トレースによる LSP 切断検出のイネーブル化

MPLS 設定がないインターフェイスを原因とする LSP 中断を MPLS LSP マルチパス ツリー トレースで検出できるようにするには、次の作業を実行します。インターフェイスが MPLS 用に設定されていない場合は、MPLS パケットを転送できません。

明示的ヌル ラベル シムによる、LSP が MPLS トラフィックを伝送する機能のテスト：IPv4 FEC を伝送する LSP の MPLS LSP マルチパス ツリー トレースでは、ラベルが要求されていない場合でも、明示的ヌル ラベルを MPLS ラベル スタックに強制的に追加できます。これにより、MPLS 用に設定されていないインターフェイスを原因とする LSP 中断を MPLS LSP マルチパス ツリー トレースで検出できます。MPLS LSP マルチパス ツリー トレースで MPLS トラフィックを送信できない場合、LSP が機能しているとレポートしません。

明示的ヌル ラベルが MPLS ラベル スタックに追加されるのは、MPLS エコー要求パケットが、MPLS LSP マルチパス ツリー トレースの宛先に直接接続されている MPLS に対して設定されていないインターフェイスから転送された場合、または MPLS エコー要求パケットの IP TTL 値が 1 に設定されている場合です。

trace mpls multipath コマンドを入力する場合は、出力ルータから入力ルータへのすべての MPLS LSP パスを探します。最後から 2 番めのホップの MPLS 用に設定されていない出力インターフェイスでの障害は検出されません。明示的ヌル シムを使用すると、LSP の MPLS トラフィック伝送能力をテストできます。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length***force-explicit-null**
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length***force-explicit-null** 例： `Router# trace mpls multipath ipv4 10.131.191.252/32 force-explicit-null`	出力ルータから入力ルータへのすべての MPLS LSP を検出し、明示的ヌル ラベルを MPLS ラベル スタックに強制的に追加します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **force-explicit-null** キーワードにより、ラベルが要求されていない場合でも、明示的ヌル ラベルを MPLS ラベル スタックに強制的に追加されます。 （注） **force-explicit-null** キーワードを入力して、MPLS 用に設定されていないインターフェイスが原因の LSP 中断を MPLS LSP マルチパス ツリー トレースで検出できるようにする必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

中継ルータへの MPLS LSP マルチパス ツリー トレースのターゲット FEC スタックの検証の要求

中継ルータに MPLS LSP マルチパス ツリー トレース機能のターゲット FEC スタックを検証するよう要求するには、次の作業を実行します。

MPLS エコー要求は、特定の LSP をテストします。テスト対象の LSP は、FEC スタックで識別されます。

MPLS LSP マルチパス ツリー トレースの実行中、エコー パケット検証ルールは、中継ルータがターゲット FEC スタック TLV を検証することを要求しません。ターゲット FEC スタックのチェックを実行するには、適切な受信ラベルを含むダウンストリーム マップ TLV がエコー要求に存在する必要があります。

中継ルータによるターゲット FEC スタックの検証を要求するには、**trace mpls multipath** コマンドに **flags fec** キーワードを入力して、送信元ルータから V フラグを設定します。デフォルトでは、エコー要求パケットは V フラグが 0 に設定されて送信されます。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**flags fec**] [**ttl***maximum-time-to-live*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**flags fec**] [**ttl***maximum-time-to-live*] 例： `Router# trace mpls multipath ipv4 10.131.159.252/32 flags fec ttl 5`	出力ルータから入力ルータへのすべての MPLS LSP を検出し、中継ルータによるターゲット FEC スタックの検証を要求します。 • **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。

	コマンドまたはアクション	目的
		• **flags fec** キーワードは、中継ルータでターゲット FEC スタック検証が実行されることを要求します。 • **ttl***maximum-time-to-live* キーワードと引数のペアでは、最大ホップ カウントを指定します。 （注） 中継ルータがターゲット FEC スタックを検証するためには、**flags fec** および **ttl** キーワードを入力する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS LSP マルチパス ツリー トレースのタイムアウト試行回数の設定

MPLS LSP マルチパス ツリー トレース機能のタイムアウト試行回数を設定するには、次の作業を実行します。

未処理のエコー要求が対応するエコー応答の待機でタイムアウトになった場合に再試行が行われます。

手順の概要

1. **enable**
2. **trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**retry-count***retry-count-value*]
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**trace mpls multipath ipv4***destination-address*/*destination-mask-length* [**retry-count***retry-count-value*]	MPLS LSP マルチパス ツリー トレース中の再試行回数を設定します。

	コマンドまたはアクション	目的
	例： `Router# trace mpls multipath ipv4 10.131.159.252/32 retry-count 4`	• **ipv4** キーワードでは、宛先タイプを LDP IPv4 アドレスとして指定します。 • *destination-address* 引数は、テストするターゲットのアドレス プレフィックスです。 • *destination-mask-length* 引数は、ターゲット アドレスのネットワーク マスク内のビット数です。この引数の前には / キーワードが必要です。 • **retry-count***retry-count-value* キーワードと引数では、タイムアウトの発生後の再試行回数を設定します。 「0」の rertry-count 値は無制限の再試行を意味します。0 ～ 10 の retry-count 値をお勧めします。10 では小さすぎる場合、再試行値を 10 よりも大きい値に増やすことがあります。デフォルトの retry-count 値は 3 です。 （注） タイムアウト後の再試行回数を設定するには、**retry-count** キーワードを入力する必要があります。
ステップ 3	**exit** 例： `Router# exit`	ユーザ EXEC モードに戻ります。

MPLS EM - MPLS LSP マルチパス ツリー トレースの設定例

MPLS エコー パケットのデフォルトの動作のカスタマイズ：例

次に、RFC 4379 をシスコと同じようには解釈しないベンダー実装と MPLS LSP マルチパス ツリー トレース機能が相互運用するように MPLS エコー パケットの動作をカスタマイズする例を示します。

```
configure terminal
!
mpls oam
 echo revision 4
 no echo vendor-extension
 end
```

完全を期すために、この例には **echo revision** コマンドが含まれています。デフォルトのエコー リビジョン番号は 4 です。これは RFC 4379 に対応します。

MPLS LSP マルチパス ツリー トレースの設定例

次に、RFC 4379 の解釈がシスコとは異なるベンダー実装と相互運用するように MPLS LSP マルチパス ツリー トレース機能を設定する例を示します。

```
configure terminal
!
mpls oam
 echo revision 4
 no echo vendor-extension
 end
!
trace mpls multipath ipv4 10.131.161.151/32
```

完全を期すために、この例には **echo revision** コマンドが含まれています。デフォルトのエコー リビジョン番号は 4 です。これは RFC 4379 に対応します。

MPLS LSP マルチパス ツリー トレースを使用した IPv4 ロード バランシング パスの検出の例

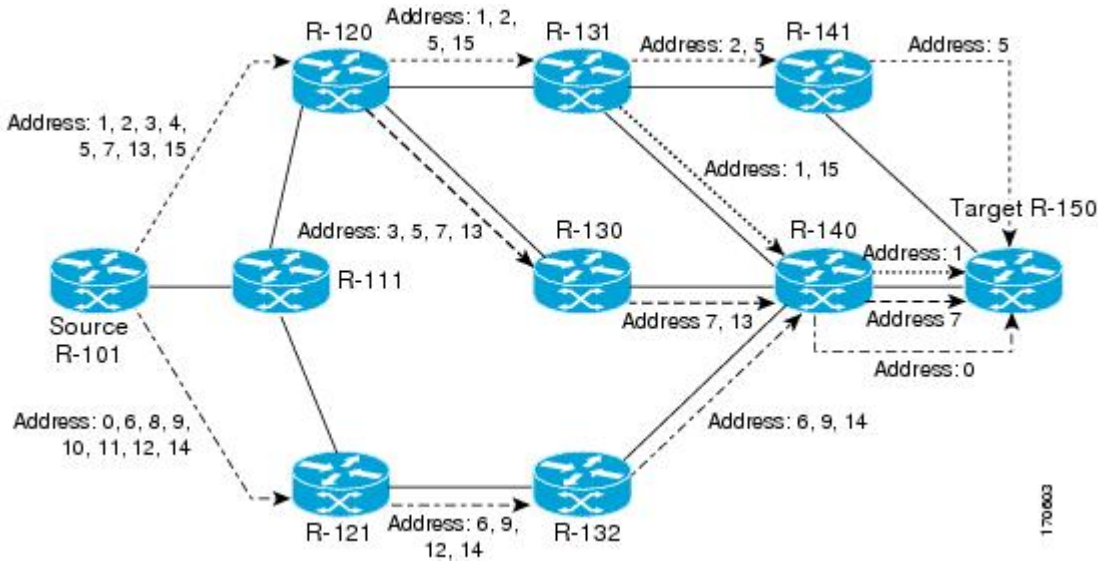
次に、MPLS LSP マルチパス ツリー トレース機能を使用して IPv4 ロード バランシング パスを検出する例を示します。この例は、次の図に示すサンプル ネットワークに基づいています。この例では、ビットマップ サイズは 16 に設定されます。したがって、パス ディスカバリは、16 のビットマップ化されたアドレスを使用する MPLS LSP マルチパス ツリー トレース機能で開始されます。ここでは、プレフィックスおよびマスク 10.1.1.150/32 を使用して、送信元ルータ R-101 からターゲット ルータ R-150 への LSP パスを探します。MPLS LSP マルチパス ツリー トレース機能は、127.0.0.0 の 127.x.y.z/8 アドレス空間を使用して開始します。

```
Router# trace mpls multipath
ipv4 10.1.1.150/32 hashkey ipv4 bitmap 16
Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
```

```
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 468 ms
```

例の **trace mpls multipath** コマンドの出力は、MPLS LSP マルチパス ツリー トレース機能によるパス ディスカバリの結果を示します。

図 ***16*** : サンプル ネットワークでの ***MPLS LSP*** マルチパス ツリー トレースによるパス ディスカバリ



DSCPを使用した、エコー応答における特定のサービスクラスの要求：例

次に、DSCP を使用してエコー応答における特定の CoS を要求する例を示します。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 reply dscp 50
Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 448 ms
```

MPLS エコー要求に対する応答ルータの応答方法の制御：例

次に、応答ルータが MPLS エコー要求に応答する方法を制御する例を示します。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 reply mode router-alert
Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0
 Total Time Elapsed 708 ms
```

MPLS LSP マルチパス ツリー トレースのためにルータから発信されるエコー パケットの出力インターフェイスの指定の例

次に、MPLS LSP マルチパス ツリー トレース機能のためにルータから発信されるエコー パケットの出力インターフェイスを指定する例を示します。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 output interface fastethernet0/0/0

Tracing MPLS Label Switched Path to 10.1.1.150/32, 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 label entry,
  '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 10.1.111.101 MRU 1500 [Labels: 33 Exp: 0]
L
  1 10.1.111.111 MRU 1500 [Labels: 33 Exp: 0] 40 ms
L
  2 10.2.120.120 MRU 1500 [Labels: 33 Exp: 0] 20 ms
L
  3 10.3.131.131 MRU 1500 [Labels: 34 Exp: 0] 20 ms
L
  4 10.4.141.141 MRU 1504 [Labels: implicit-null Exp: 0] 20 ms !
  5 10.5.150.150 16 ms
```

MPLS LSP マルチパス ツリー トレースの MPLS エコー要求パケット送信ペースの設定の例

次に、MPLS LSP マルチパス ツリー トレース機能の MPLS エコー要求パケット送信ペースを設定する例を示します。連続する MPLS エコー要求間の時間は、最初の例では 300 ミリ秒に設定され、2 番めの例では 400 ミリ秒に設定されています。

```
Router# trace mpls multipath ipv4 10.131.159.252/32 interval 300
Starting LSP Multipath Traceroute for 10.131.159.252/32
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 label entry,
  '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 Et1/0 source 10.2.3.2 destination 127.0.0.0
Paths (found/broken/unexplored) (1/0/0)
 Echo Request (sent/fail) (3/0)
 Echo Reply (received/timeout) (3/0)
 Total Time Elapsed 1604 ms
Router# trace mpls multipath ipv4 10.131.159.252/32 interval 400
Starting LSP Multipath Traceroute for 10.131.159.252/32
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 label entry,
  '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 Et1/0 source 10.2.3.2 destination 127.0.0.0
Paths (found/broken/unexplored) (1/0/0)
 Echo Request (sent/fail) (3/0)
 Echo Reply (received/timeout) (3/0)
 Total Time Elapsed 1856 ms
```

間隔値を大きくすると経過時間が長くなることに注意してください。

MPLS LSP マルチパス ツリー トレースの有効化の例

次に、MPLS 設定がないインターフェイスを原因とする LSP 中断を MPLS LSP マルチパス ツリー トレース機能が検出できるようにする例を示します。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 force-explicit-null

Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
```

```
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 460 ms
```

この例は、**verbose** キーワードをコマンドに追加した場合に提供される追加情報を示しています。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 force-explicit-null verbose
Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
 '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
  0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
  1 10.1.111.111 10.2.121.121 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
  2 10.2.121.121 10.3.132.132 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 1
L
  3 10.3.132.132 10.4.140.240 MRU 1500 [Labels: 32/explicit-null Exp: 0/0] ret code 8
multipaths 1
L
  4 10.4.140.240 10.5.150.50 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
 1 !
  5 10.5.150.50, ret code 3 multipaths 0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
  0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
  1 10.1.111.111 10.2.120.120 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
  2 10.2.120.120 10.3.131.131 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
  3 10.3.131.131 10.4.141.141 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
  4 10.4.141.141 10.5.150.150 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
 1
!
5 10.5.150.150, ret code 3 multipaths 0
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
  0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
 1 10.1.111.111 10.2.120.120 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
 2 10.2.120.120 10.3.131.131 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
```

```
L
 3 10.3.131.131 10.4.140.140 MRU 1500 [Labels: 32/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
 4 10.4.140.140 10.5.150.50 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
1 ! 5 10.5.150.50, ret code 3 multipaths 0
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
  0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
  1 10.1.111.111 10.2.120.120 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
  2 10.2.120.120 10.3.130.130 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
  3 10.3.130.130 10.4.140.40 MRU 1500 [Labels: 32/explicit-null Exp: 0/0] ret code 8
multipaths 1
L
  4 10.4.140.40 10.5.150.50 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
1
!
  5 10.5.150.50, ret code 3 multipaths 0
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 492 ms
```

中継ルータへのMPLS LSPマルチパス トレースのターゲットFECスタックの検証の要求の例

次に、中継ルータに MPLS LSP マルチパス ツリー トレース機能のターゲット FEC スタックを検証するよう要求する例を示します。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 flags fec ttl 5


Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 464 ms
```

ターゲット FEC スタック検証は、**trace mpls multipath** コマンドで **flags fec** キーワードが指定されている場合に出力ルータで常に実行されます。

MPLSLSPマルチパスツリートレースのタイムアウト試行回数の設定：例

次に、MPLS LSP マルチパス ツリー トレース機能のタイムアウト試行回数を 4 に設定する例を示します。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 retry-count 4


Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLLL!
Path 0 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1
L!
Path 2 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.5
LL!
Path 3 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 460 ms
```

次の出力には、**trace mpls multipath** コマンドで 1 つの探索されていないパス、1 つの正常なパス、および 1 つの分断したパスが見つかったことが示されています。

```
Router# trace mpls multipath ipv4 10.1.1.150/32 retry-count 4

Starting LSP Multipath Traceroute for 10.1.1.150/32
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 label entry,
  '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.
LLL....
Path 0 Unexplorable,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.0
LLL!
Path 1 found,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.1 B
Path 2 Broken,
 output interface Fe0/0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (1/1/1)
 Echo Request (sent/fail) (12/0)
 Echo Reply (received/timeout) (8/4)
 Total Time Elapsed 7868 ms
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
MPLS ベースの機能	• *MPLS* ラベル配布プロトコル（*LDP*） • 『*MPLS Label Switching Router MIB*』 • 『*MPLS Scalability Enhancements for the LSC LSR*』 • 『*MPLS Scalability Enhancements for the ATM LSR*』 • 『*MPLS Traffic Engineering (TE)—Automatic Bandwidth Adjustment for MPLS TE Tunnels*』 • 『*MPLS Traffic Engineering (TE)—Scalability Enhancements*』 • 『*MPLS Class of Service Enhancements*』 • 『*RFC 2233 Interfaces MIB*』

標準

規格	タイトル
draft-ietf-mpls-te-mib-05	『MPLS Traffic Engineering Management Information Base Using SMIv2』

MIB

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

RFC

RFC	タイトル
RFC 2026 RFC 3812	「*The Internet Standards Process*」 『Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)』

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

関連資料

関連項目	マニュアル タイトル
MPLS LSP ping または traceroute の概念と設定作業	MPLS LSP Ping/Traceroute for LDP/TE および LSP Ping for VCCV
MPLS コマンド	『*Cisco IOS Multiprotocol Label Switching Command Reference*』

標準

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

MIB

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

RFC

RFC	タイトル
RFC 2113	*IP Router Alert Option*
RFC 3443	『*Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks*』
RFC 4377	『*Operations and Management (OAM) Requirements for Multi-Protocol Label Switched (MPLS) Networks*』
RFC 4378	『*A Framework for Multi-Protocol Label Switching (MPLS) Operations and Management (OAM)*』
RFC 4379	『*Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures*』

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 EM - MPLS LSP マルチパス ツリー トレースの機能情報

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

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

表 *26*：***MPLS EM—MPLS LSP*** マルチパス ツリー トレースの機能情報

機能名	リリース	機能情報
MPLS EM - MPLS LSP マルチパス ツリー トレース	Cisco IOS XE Release 2.3	MPLS EM—MPLS LSP マルチパス ツリー トレース機能は、出力ルータと入力ルータ間でラベル スイッチド パス（LSP）として使用可能なすべてのパスを検出する手段となります。これらのパスは、検出後、マルチプロトコル ラベル スイッチング（MPLS）LSP ping または traceroute を使用して定期的に再テストできます。この機能は、IPv4 LSP のトレース用の MPLS LSP traceroute 機能に対する拡張です。 MPLS Embedded Management（EM）は、障害、設定、アカウンティング、パフォーマンス、セキュリティ（FCAPS）モデルに従った MPLS ベースのネットワークの開発、操作、アドミニストレーション、および管理を容易にする標準と付加価値サービスのセットです。 Cisco IOS XE Release 2.3 では、Cisco ASR 1000 シリーズ Aggregation Services Router にこの機能が実装されました。
		次のコマンドが導入または変更されました。**debug mpls lspv**、**echo**、**mpls oam**、**trace mpls**、**trace mpls multipath**。

用語集

ECMP：等コスト マルチパス。パケット転送に使用できる複数の等コスト ルーティング パス。

FEC：Forward Equivalence Class。転送のために同等に処理できるパケットのセット。したがって、1 つのラベルへのバインディングに適しています。たとえば、1 つのアドレス プレフィックス宛てのパケットのセットやフロー内のパケットなどがあります。

フロー：一組のホスト間、または一組のホスト上にある一組のトランスポートプロトコルポート間で転送されるパケットのセット。たとえば、同じ送信元アドレス、送信元ポート、宛先アドレス、および宛先ポートを持つパケットは、フローと見なされることがあります。

フローは、ネットワークの 2 つのエンドポイント間で（たとえば、ある LAN ステーションから別の LAN ステーションへ）転送されるデータのストリームでもあります。単一の回線上で複数のフローを転送できます。

localhost：ホスト ルータ（デバイス）を表す名前。localhost は、予約済みのループバック IP アドレス（127.0.0.1）を使用します。

LSP：Label Switched Path（ラベル スイッチド パス）。マルチプロトコル ラベル スイッチング（MPLS）がパケットを転送する 2 つのルータ間の接続。

LSPV：Label Switched Path Verification。LSP ping のサブプロセス。マルチプロトコル ラベル スイッチング（MPLS）エコー要求と応答を符号化およびデコードします。また、MPLS エコー要求と応答を送受信するために、IP、MPLS、および AToM スイッチングとやり取りします。MPLS エコー要求の発信元ルータでは、LSPV によって、エコー応答が受信されていない未処理のエコー要求のデータベースが保持されます。

MPLS ルータ アラート ラベル：マルチプロトコル ラベル スイッチング（MPLS）ラベル 1。ルータ アラート ラベルを含む MPLS パケットは、処理のためにルータによって Route ルート プロセッサ（PR）の処理レベルにリダイレクトされます。これにより、これらのパケットはハードウェア ルーティング テーブルにおけるフォワーディング エラーを回避できます。

OAM：Operation, Administration, and Maintenance（保守運用管理）。

パント：ルータ アラートを含むパケットを処理のためにラインカードまたはインターフェイスからルート プロセッサ（RP）のレベル処理にリダイレクトします。

RP：ルート プロセッサ。このプロセッサ モジュールには、CPU、システム ソフトウェア、およびルータで使用されるほとんどのメモリ コンポーネントが含まれています。

TTL：存続可能時間。設定可能なパラメータであり、パケットが宛先に到達するまでに通過するホップの最大数を示します。

TLV：Type, Length, Value（タイプ、長さ、値）。Cisco Discovery Protocol アドレスに含まれる情報のブロックです。

UDP：User Datagram Protocol。TCP/IP プロトコル スタックのコネクションレス型トランスポート層プロトコルです。UDP は、確認応答や配信保証を行わずにデータグラムを交換する単純なプロトコルです。そのため、エラー処理と再伝送を他のプロトコルで処理する必要があります。UDP は RFC 768 で定義されています。

XDR：eXternal Data Representation（外部データ表現）。Sun Microsystems によって開発された、マシンに依存しないデータ構造の規格。ルート プロセッサ（RP）とライン カード間のメッセージ伝送に使用されます。

第 6 章

MPLS ラベル配布プロトコル MIB

このドキュメントでは、Cisco ソフトウェアで提供される、MPLS ラベル配布プロトコル管理情報ベース（MPLS LDP MIB）の Simple Network Management Protocol（SNMP）エージェント サポートについて説明します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS LDP MIB の制約事項

MPLS LDP MIB は、MIB オブジェクトに対するアクセス権が読み取り専用（RO）に制限されます。ただし、SNMP エージェントによる書き込みが可能な拡張 MIB オブジェクト mplsLdpSessionUpDownTrapEnable は例外です。

このオブジェクトの値を true に設定すると、ラベル スイッチド ルータ（LSR）で mplsLdpSessionUp 通知と mplsLdpSessionDown 通知の両方がイネーブルになります。逆に、このオブジェクトの値を false に設定すると、これらの通知がいずれもディセーブルになります。mplsLdpSessionUpDownTrapEnable オブジェクトの値は、MPLS LDP MIB ホスト上の NVRAM に保存されます。

イベント通知については、「MPLS LDP MIB の通知の生成イベント」を参照してください。

ほとんどの MPLS LDP MIB オブジェクトは、LDP ピアのディスカバリ（Hello）プロセス、および以降の LDP ピア間 LDP セッションのパラメータや確立のネゴシエーション中に自動的に設定されます。

MPLS LDP MIB に関する情報

MPLS LDP の概要

マルチプロトコル ラベル スイッチング（MPLS）は、パケット転送テクノロジーであり、パケットでラベルと呼ばれる短い固定長の値を使用して、ラベル スイッチング ルータ（LSR）による MPLS ネットワークでのパケット転送のネクスト ホップを判別します。

基本的な MPLS の原則は、MPLS ネットワーク内の LSR は、パケット転送操作に使用するラベルの定義で一致している必要があるということです。ラベルの同意は、Label Distribution Protocol（LDP）で定義されている手順によって MPLS ネットワークで行われます。

LDP 操作は検出（Hello）プロセスで開始し、このプロセスで LDP エンティティ（ローカル LSR）がネットワーク内の協働 LDP ピアを検出し、これらの間での基本操作プロシージャをネゴシエートします。この検出プロセスによりピアが認識および特定されると、Hello 隣接が生成されます。Hello 隣接は、ローカル LSR とその LDP ピアの間でラベル バインド情報が交換される状況を表します。次に LDP 機能は、ラベルバインド情報の交換を実現するため 2 つの LSR の間でアクティブな LDP セッションを確立します。このプロセスが MPLS ネットワーク内のすべての LSR に関して完了すると、通信ネットワーク デバイス間のエンドツーエンドのパケット伝送経路を構成するラベル スイッチド パス（LSP）が確立されます。

LDP により、LSR はラベル バインド情報を収集し、MPLS ネットワーク内の他のデバイスに配布および解放します。これにより、ネットワーク内で通常ルーティング パスに沿ったパケットのホップバイホップ転送が有効になります。

MPLS LDP MIB の概要

MPLS LDP MIB は、Cisco ソフトウェアにおけるラベル スイッチング機能について標準の SNMP ベースのネットワーク管理を実行できるようするために実装されました。この機能を使用するには、ネットワーク内の指定したネットワーク管理ステーション（NMS）で SNMP エージェントコードを実行する必要があります。NMS は、MPLS LDP MIB 内のネットワーク管理オブジェクトとユーザの対話の媒体となります。

SNMP エージェントは、Cisco ソフトウェアと互換性のある階層構造を持ち、MPLS LDP MIB 内のオブジェクト、さらに Cisco ソフトウェアによってサポートされる豊富なラベル スイッチング機能一式とのネットワーク管理インターフェイスを提供します。

SNMP エージェントにより、標準の SNMP **get** 操作を使用して MPLS LDP MIB オブジェクトにアクセスでき、さまざまなネットワーク管理タスクを実行できます。MPLS LDP MIB のすべてのオブジェクトは、Internet Engineering Task Force（IETF）ドラフト MIB（*draft-ietf-mpls-ldp-mib-08.txt*）に定義されている規則に従います。このドラフト MIB は、構造的および標準的な方法でネットワーク管理オブジェクトを定義します。このドラフト MIB は今後標準規格となるべく継続的に作業が行われています。したがって、MPLS LDP MIB は、この IETF ドキュメントの発展を追って実装されます。

IETF ドラフト MIB と Cisco ソフトウェア内の同等機能の実装はわずかに異なるため、MPLS LDP MIB オブジェクトと Cisco ソフトウェアの内部データ構造の間でいくつかの軽微な変換が必要となります。このような変換は SNMP エージェントにより実行されます。SNMP エージェントは、NMS ワークステーション上で、優先度が低いプロセスとしてバックグラウンドで実行されます。

MPLS LDP MIB の機能は次のとおりです。

- MPLS LDP MIB は、LDP セッションのステータスの変化を伝えるイベント通知メッセージを生成して送信できます。
- SNMP CLI コマンドを使用してイベント通知メッセージを有効および無効にできます。
- ネットワーク管理の目的でイベント通知メッセージが送信される NMS ワークステーションの名前または IP アドレスを指定できます。
- NMS の不揮発性メモリ（NVRAM）に、イベント通知メッセージに関連する設定を保存できます。

MPLS LDP MIB の構造は、抽象構文記法 1（ASN.1）に準拠しているため、高度に構造化された理想的なネットワーク管理オブジェクト データベースを形成します。

標準の SNMP アプリケーションを使用して、標準の SNMP GET 操作によって MPLS LDP MIB から情報を取得して表示できます。同様に、SNMP GETNEXT 操作によって MIB の情報を走査して表示することができます。

（注） MPLS LDP MIB の実装時点では、この MIB には Internet Assigned Numbers Authority（IANA）の Experimental OID が割り当てられていなかったため、シスコでは Cisco Experimental OID 番号を使用してこの MIB を実装しました（ciscoExperiment 1.3.6.1.4.1.9.10 mplsLdpMIB 1.3.6.1.4.1.9.10.65）。MPLS LDP MIB に IANA Experimental OID 番号が割り当てられる場合には、シスコは ciscoExperimental OID が設定されたこの MIB のすべてのオブジェクトを廃止し、IANA Experimental OID に移行します。

MPLS LDP MIB を使用する利点

MPLS LDP MIB には次の利点があります。

- MPLS ネットワーク内のピア デバイス間の LDP セッションの確立
- 次のような LDP エンティティの操作に関連する MIB パラメータの取得：
  - 既知の LDP ディスカバリ ポート
  - 最大伝送ユニット（MTU）
  - 提示されるキープアライブ タイマー インターバル
  - ループ検出
  - セッション確立しきい値
  - ラベルの形成に使用される VPI/VCI ペアの範囲
- LDP の動作に関連する次のような統計情報の収集：
  - LDP エンティティに対して確立されたセッションの合計数
  - LDP エンティティに対して試行されたセッションの合計数
- hello 隣接の残り時間の監視
- 次のような LDP ピアの特性とステータスの監視：
  - LDP ピアのインターネットワーク層アドレスのタイプ
  - LDP ピアの実際のインターネットワーク層アドレス
  - LDP ピアのデフォルト MTU
  - LDP ピアがキープアライブ インターバルの値として提示する秒数
  - LDP ピアが認識する VPI/VCI ラベル範囲の確立
- 次のような LDP セッションの特性とステータスの監視：
  - LDP セッションが使用している LDP のバージョンの確認

- LDP セッションのキープアライブ保留時間の確認
- LDP セッションの状態の確認（セッションがアクティブかどうか）
- LDP セッションによって使用される VPI/VCI ペアの範囲の確認
- LDP セッションの最後のアクティブ インターフェイスの確認

MPLS LDP MIB 要素の説明

MPLS LDP MIB に含まれている要素を次に示します。

- LDP エンティティ：ラベル スペースの交換を目的としており、LDP インスタンスに関連します。
- LDP ピア：リモート LDP エンティティ（つまり非ローカル LSR）を意味します。
- LDP セッション：ローカル LSR とリモート LDP ピアの間のアクティブな LDP プロセスを意味します。
- Hello 隣接：MPLS ネットワーク内の 2 つの LSR が相互に隣接している（つまり LDP ピアである）状態であることを示す LDP ディスカバリ プロセスの結果を意味します。

Hello 隣接は、MPLS ネットワーク内の 2 つの LSR 間で有効なコンテキストを構成します。隣接は、ラベル バインド情報の交換に使用されます。

これらの MPLS LDP MIB 要素について、以降で個々の見出しの下で簡潔に説明します。

実際には MPLS LDP MIB が提供するネットワーク管理データベースでは、ネットワーク内の MPLS LDP 操作の現在の状況を反映し、内部のさまざまな MIB オブジェクトへのリアルタイム アクセスをサポートしています。このネットワーク管理情報データベースにアクセスするには、MPLS/LDP 稼働環境で NMS から標準の SNMP コマンドを使用します。

MPLS LDP MIB は、次のネットワーク管理アクティビティをサポートしています。

- LDP 動作に関連する MPLS LDP MIB パラメータの取得
- LDP ピアの特性とステータスの監視
- LDP ピア間の LDP セッションのステータスの監視
- ネットワークにおける Hello 隣接の監視
- LDP セッションに関する統計情報の収集

LDP エンティティ

LDP エンティティは、オブジェクト名 *mplsLdpEntityLdpId* を含む LDP 識別子によって一意に識別されます。このオブジェクトは、ルータ ID（4 オクテット）とインターフェイス番号（2 オクテット）で構成されます。ルータ ID は LSR に割り当てられている IP アドレスをエンコードします。インターフェイス番号は、LSR 内で使用可能な特定のラベル スペースを示します。

LDP エンティティは、LDP ピアへの配布のターゲットであるラベル スペースを表します。インターフェイス固有の LDP エンティティの場合、ラベル スペースは 1 つの LDP セッションにより 1 つの LDP ピアに配布されます。

プラットフォーム全体の LDP エンティティは、複数の LDP ピアに関連付けることができます。この場合、ラベル スペースは各ピアに関連する個別の LDP セッションにより複数の LDP ピアに配布されます。

LDP Peers

LSR に、別の LSR または複数の LSR にアドバタイズするラベル スペースがある場合、ラベル スペース情報を受信する LSR ごとに 1 つの LDP セッションが存在します。ラベル スペース情報の受信側は LDP ピアと呼ばれます。

インターフェイス単位のラベル スペースは、1 つの LDP セッションにより 1 つの LDP ピアにアドバタイズされます。プラットフォーム単位のラベル スペースは、複数の LDP セッションにより複数の LDP ピアにアドバタイズされます。

プラットフォーム単位の LDP ピアが複数存在する可能性がある場合、LDP エンティティはその一意の LDP タグだけでなく、LDP インデックスによっても識別されます。この場合、ラベル スペースは同一ですが、LDP インデックスによって、ラベル スペースを複数の LDP ピアに配布する LDP セッションが区別されます。

LDP セッション

ローカル エンティティとリモート ピアの間の LDP セッションは、ラベル スペースを配布します。LDP ピアと LDP セッションの間の関係は常に 1 対 1 です。単一 LDP セッションは、1 つ以上のネットワーク リンク経由で単一 LDP ピアと通信する Label Distribution Protocol インスタンスです。プラットフォーム全体のローカル LDP エンティティの場合、複数の LDP セッションと、これに対応する数のリモート LDP ピアが存在する可能性があります。

LDP Hello 隣接

LDP セッションは、1 つ以上のネットワーク リンク経由でピア プロトコル インスタンスと通信する LDP インスタンスです。LDP が実行されるリンクごとに LDP hello 隣接が存在します。1 つの LDP ピアへのリンクが複数ある場合には常に、複数のリンク隣接が存在します。たとえばプラットフォーム全体のラベル スペースの場合、ラベル スペースのアドバタイズ先となる LSR ごとに個別の LDP ピア/LDP セッション関係があります。

MPLS LDP MIB オブジェクトのカテゴリ

MPLS LDP MIB には、IETF ドラフト ドキュメント *draft-ietf-mpls-ldp-08.txt* で定義されているように、MPLS Label Distribution Protocol の管理対象オブジェクトの定義が含まれています。

MPLS LDP MIB の管理対象オブジェクトの構造は、次のカテゴリに従っています。

- MPLS LDP Textual Conventions

- MPLS LDP Objects
- MPLS Label Distribution Protocol Entity Objects
- LDP Entity Objects for Generic Labels
- LDP Entity Objects for ATM
- MPLS LDP Entity Configured ATM Label Range Table
- MPLS Entity Objects for Frame Relay
- Frame Relay Label Range Components
- MPLS LDP Entity Statistics Table
- MPLS LDP Entity Peer Table
- MPLS LDP Hello Adjacency Table
- MPLS LDP Sessions Table
- MPLS LDP ATM Session Information
- MPLS LDP Frame Relay Session Information
- MPLS LDP Session Statistics Table
- Address Message/Address Withdraw Message Information
- MPLS LDP LIB Table
- MPLS LDP FEC Table
- Notifications
- Module Conformance Statement

MPLS LDP MIB 通知の生成イベント

snmp-server enable traps mpls ldp コマンドを発行して MPLS LDP MIB 通知機能を有効にすると、Cisco ソフトウェア内で特定のイベントが発生したことを伝える通知メッセージが生成され、指定された NSM に送信されます。

LDP ステータス移行とイベント通知をアナウンスする MPLS LDP MIB オブジェクトには、次のメッセージが含まれています。

- mplsLdpSessionUp：このメッセージは、LDP エンティティ（ローカル LSR）によって別の LDP エンティティ（ネットワーク内の隣接 LDP ピア）との LDP セッションが確立されると生成されます。
- mplsLdpSessionDown：このメッセージは、ローカル LSR とその隣接 LDP ピア間の LDP セッションが終了すると生成されます。

アップ通知とダウン通知は、LDP セッションにおける最後のアクティブ インターフェイスを示します。

- mplsLdpPathVectorLimitMismatch：このメッセージは、ローカル LSR によって、その隣接ピアである LSR との LDP セッションが確立され、2 つの LSR でパス ベクトル制限が異なる場合に生成されます。

パス ベクトル制限の値の範囲は 0 ～ 255 です。値が 0 の場合、ループ検出はオフです。0 以外の 255 までの値の場合、ループ検出はオンで、さらにネットワーク内のループ状態が検知されるまでに LDP メッセージが通過できるホップの最大数が示されます。

ネットワーク内のすべての LDP 対応ルータに同じパス ベクトル制限を設定することを推奨します。mplsLdpPathVectorLimitMismatch オブジェクトが MPLS-LDP-MIB に存在するのは、LDP 動作に関わっている 2 つのルータのパス ベクトル制限が異なる場合に NMS に警告メッセージを送信するためです。

- mplsLdpFailedInitSessionThresholdExceeded：このメッセージは、ローカル LSR と隣接 LDP ピアが、それらの間に LDP セッションを確立しようとして失敗し、その試行回数が指定数を超えた場合に生成されます。デフォルトの試行回数は 8 回です。このデフォルト値は Cisco ソフトウェアで実装され、CLI および SNMP エージェントが変更することはできません。

デバイス間での非互換性が原因で、ローカル LSR と LDP ピア間の LDP セッションの確立に 8 回失敗しました。このため、この通知メッセージが生成されます。

一般に、Cisco ルータは複数のプラットフォームで同じ機能をサポートします。したがって、Cisco LSR 間で最も発生する可能性が高い非互換は、それぞれの ATM VPI/VCI ラベル範囲のミスマッチです。

たとえば、LSR に有効なラベルの範囲を指定し、その範囲が隣接 LDP ピアの範囲と重ならない場合、ルータは LDP ピアとの LDP セッションを 8 回確立しようとします。その後、mplsLdpFailedInitSessionThresholdExceeded 通知が生成され、情報メッセージとして NMS に送信されます。

運用上、ラベル範囲が重ならない LSR は、8 回のリトライ制限を超えても、引き続きそれらの LSR 間の LDP セッションを確立しようとします。そのような場合、LDP しきい値超過通知によって、ネットワーク内に注意すべき状態があることがネットワーク管理者に知らされます。

RFC 3036『LDP Specification』に、MPLS ネットワーク内の Cisco ルータやその他の LSR 間に存在する可能性がある非互換について詳しく記載されています。このような非互換の例を次に示します。

- - LDP セッションのセットアップを試行する LSR の間で、ATM VPI/VCI 範囲（前述）が重複していない、またはフレーム リレー DLCI 範囲が重複していない。
  - ラベル配布方式がサポートされていない。
  - プロトコル データ ユニット（PDU）サイズが異なる。
  - LDP 機能のサポートが異なる。

MPLS LDP MIB の設定方法

MPLS LDP MIB に対する SNMP エージェントのイネーブル化

デフォルトでは、MPLS LDP MIB の SNMP エージェントはディセーブルになっています。ホスト NMS ワークステーション上の SNMP エージェントをイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [*acl-number*]
5. **do copy running-config startup-config**
6. **exit**
7. **show running-config** [**interface** | **map-class**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： `Router# show running-config`	SNMP エージェントがすでに実行中かどうかが判別される実行コンフィギュレーションが表示されます。 • SNMP の情報が表示されない場合は、次のステップに進みます。SNMP 情報が表示される場合、必要に応じて、情報を変更できます。
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [*acl-number*]	SNMP プロトコルへのアクセスを許可するようにコミュニティ アクセス ストリングを設定します。

	コマンドまたはアクション	目的
	例： `Router(config)# snmp-server community comaccess ro`	• *string* 引数はパスワードのように機能し、SNMP プロトコルへのアクセスを許可します。 • **Viewview-name***view-name* キーワードと引数のペアには、以前に定義されたビューの名前を指定します。ビューには、コミュニティで使用できるオブジェクトが定義されています。 • **ro** キーワードは、読み取り専用アクセスを指定します。MIB オブジェクトを取得できるのは、許可された管理ステーションだけです。 • **rw** キーワードは、読み取りと書き込みアクセスを指定します。MIB オブジェクトの取得と変更の両方を実行できるのは、許可された管理ステーションです。 • *acl-number* 引数は、1 ～ 99 の整数で、コミュニティ ストリングを使用した SNMP エージェントへのアクセスが許可される IP アドレスのアクセス リストを指定します。
ステップ 5	**do copy running-config startup-config** 例： `Router(config)# do copy running-config startup-config`	変更された設定をスタートアップ コンフィギュレーション ファイルとして不揮発性メモリ（NVRAM）に保存します • **do** コマンドを使用すると、コンフィギュレーション モードで EXEC レベルのコマンドを実行できます。
ステップ 6	**exit** 例： `Router(config)# exit`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** [**interface** | **map-class**] 例： `Router# show running-config | include smnp-server`	（任意）ルータ上の現在の設定情報、特定のインターフェイスの情報、またはマップクラス情報を表示します。 • **show running-config** コマンドを使用すると、**snmp-server** のステートメントが出力に表示されることをチェックできます。

ルータによる SNMP トラップ送信の設定

トラップをホストに送信するようにルータを設定するには、この作業を実行します。

snmp-server host コマンドを使用して、トラップを受信するホストを指定します。**snmp-server enable traps** コマンドでは、指定したトラップのトラップ生成メカニズムをグローバルにイネーブルにします。

ホストでトラップを受信するには、そのホストに **snmp-server host** コマンドを設定する必要があります。また通常は、**snmp-server enable traps** コマンドでトラップをグローバルにイネーブルにされていることも必要です。

（注） **snmp-server host** コマンド自体を使用して *community-string* 引数を設定できますが、**snmp-server community** コマンドを使用してこのストリングを定義してから **snmp-server host** コマンドを使用することを推奨します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server host***host-addr* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [*notification-type*] [**vrf***vrf-name*]
4. **snmp-server enable traps mpls ldp** [**session-down**] [**session-up**] [**pv-limit**] [**threshold**]
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server host***host-addr* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [*notification-type*] [**vrf***vrf-name*] 例： `Router(config)# snmp-server host 172.20.2.160 traps comaccess mpls-ldp`	SNMP 通知操作の受信者を指定します。 • *host-addr* 引数には、ホスト（ターゲット受信者）の名前またはインターネット アドレスを指定します。 • **traps** キーワードを指定すると、このホストに SNMP トラップが送信されます。これはデフォルトです。 • **informs** キーワードを指定すると、このホストに SNMP 応答要求が送信されます。

	コマンドまたはアクション	目的

		• **version** キーワードは、トラップの送信に使用する SNMP のバージョンを指定します。最も安全なモデルはバージョン 3 です。このバージョンでは、**priv** キーワードを使用してパケットを暗号化できるためです。**version** キーワードを使用する場合は、次のいずれかを指定する必要があります。
		  • **1**：SNMPv1。情報の場合は、このオプションを使用できません。
		  • **2c**：SNMPv2C。
		  • **3**：SNMPv3。**version 3** キーワードのあとに 3 つのオプション キーワード（**auth**、**noauth**、**priv**）を指定できます。
		• *community-string* 引数は、通知操作で送信される、パスワードに似たコミュニティ文字列です。
		• **udp-port***port* キーワードと引数のペアには、使用するホストの UDP ポートを指定します。デフォルトは 162 です。
		• *notification-type* 引数には、ホストに送信する通知のタイプを指定します。タイプが指定されていない場合、すべての通知が送信されます。
		• **vrf***vrf-name* キーワードと引数のペアには、SNMP 通知の送信に使用する VRF テーブルを指定します。

ステップ 4	**snmp-server enable traps mpls ldp** [**session-down**] [**session-up**] [**pv-limit**] [**threshold**] 例： `Router(config)# snmp-server enable traps mpls ldp session-down session-up`	ルータで MPLS VPN 固有の SNMP 通知（トラップと応答要求）を送信できるようにします。 • **session-down** キーワードを指定すると、LDP セッションのダウン通知（mplsLdpSessionDown）を制御（イネーブルまたはディセーブルに）できます。このメッセージは、ルータとその隣接 LDP ピア間の LDP セッションが終了すると生成されます。 • **session-up** キーワードを使用すると、LDP セッションのアップ通知（mplsLdpSessionUp）を制御（イネーブルまたはディセーブルに）できます。この通知は、ルータによって別の LDP エンティティ（ネットワーク内の隣接 LDP ピア）との LDP セッションが確立されると生成されます。 • **pv-limit** キーワードを使用すると、パス ベクトル（PV）制限の通知（mplsLdpPathVectorLimitMismatch）を制御（イネーブルまたはディセーブルに）できます。この通知は、ルータがその隣接ピアである LSR と LDP セッションを確立し、2 つの LSR でパス ベクトル制限が異なる場合に生成されます。 • **threshold** キーワードを使用すると、PV 制限の通知（mplsLdpFailedInitSessionThresholdExceeded）を制御（イネーブルまた

	コマンドまたはアクション	目的
		はディセーブルに）できます。この通知は、ルータと LDP ピア間の LDP セッションの確立に 8 回失敗すると生成されます。デバイス間に何らかの非互換がある場合にもセッションの確立に失敗することがあります。
ステップ 5	**exit** 例： `Router(config)# exit`	（任意）終了して、特権 EXEC モードに戻ります。

SNMP エージェントのステータスの確認

ホスト NMS ワークステーション上で SNMP エージェントがイネーブルにされたことを確認するには、次の手順を実行します。

手順の概要

1. **enable**
2. **show running-config**
3. **exit**

手順の詳細

ステップ 1 **enable**

このコマンドを使用して、特権EXECモードをイネーブルにします。パスワードを入力します（要求された場合）。次に例を示します。

例：

```
Router> enable
Router#
```

ステップ 2 **show running-config**

このコマンドを使用して、ホスト NMS 上の実行コンフィギュレーションを表示し、SNMP 情報の出力を確認します。次に例を示します。

例：

```
Router# show running-config
.
.
```

```
.
snmp-server community public RO
snmp-server community private RO
```

上記のような形式の出力に snmp-server のステートメントが含まれる場合、SNMP エージェントはホスト NMS ワークステーションでイネーブルにされていることになります。

ステップ 3 **exit**

このコマンドを使用して、ユーザ EXEC モードに戻ります。次に例を示します。

例：

```
Router# exit
Router>
```

MPLS LDP MIB の設定例

SNMP エージェントのイネーブル化：例

次に、ホスト NMS 上で SNMP エージェントをイネーブルにする例を示します。

```
Router# configure terminal
Router(config)# snmp-server community
```

次に、ホスト NMS 上の SNMPv1 と SNMPv2C をイネーブルにする例を示します。設定では、コミュニティ ストリング public を使用して、SNMP エージェントが読み取り専用アクセス権ですべての MPLS LDP MIB オブジェクトにアクセスすることを許可しています。

```
Router(config)# snmp-server community public
```

次に、comaccess コミュニティ ストリングを指定するアクセス リスト 4 のメンバに、すべての MPLS LDP MIB オブジェクトへの読み取り専用アクセスを許可する例を示します。その他の SNMP エージェントは MPLS LDP MIB オブジェクトにアクセスできません。

```
Router(config)# snmp-server community comaccess ro 4
```

次に、セッション アップおよびセッション ダウンの LDP 通知をイネーブルにする例を示します。

```
Router(config)# snmp-server enable traps mpls ldp session-up
Router(config)# snmp-server enable traps mpls ldp session-down
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
MPLS LDP の設定作業	MPLS ラベル配布プロトコル（LDP）
MPLS LDP コマンド：コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco IOS Multiprotocol Label Switching Command Reference*』
SNMP コマンド	『*Cisco IOS Network Management Command Reference*』
SNMP コンフィギュレーション	『*Network Management Configuration Guide*』の「*Configuring SNMP Support*」

標準

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

MIB

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

RFC

RFC	タイトル
RFC 3036	『*LDP Specification*』

RFC	タイトル
RFC 3037	『*LDP Applicability*』

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 LDP MIB の機能情報

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

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

表 *27* : ***MPLS LDP MIB*** の機能情報

機能名	リリース	機能情報
MPLS LDP MIB	12.0(11)ST 12.2(2)T 12.0(21)ST 12.2(13)T 12.0(30)S 12.2(27)SBC 12.2(28)SB 12.2(33)SRA 12.2(33)SXH Cisco IOS XE Release 2.1	

機能名	リリース	機能情報
		MPLS LDP MIB は、Cisco ソフトウェアにおけるラベルスイッチング機能について標準の SNMP ベースのネットワーク管理を実行できるようするために実装されました。 Cisco IOS Release 12.0(11)ST で、Cisco 7200、Cisco 7500、および Cisco 12000 シリーズ ルータで MPLS LDP MIB を使用する際に SNMP エージェントをサポートするために、この機能が導入されました。 Cisco IOS Release 12.2(2)T で、Cisco 7200 および Cisco 7500 シリーズ ルータで MPLS LDP MIB を使用する際に SNMP エージェントをサポートするために、この機能が統合されました。 Cisco IOS Release 12.0(21)ST で、**snmp-server enable traps mpls ldp** コマンドが導入されました。 **snmp-server enable traps mpls ldp** コマンドが Cisco IOS Release 12.2(13)T に統合されました。 この機能は、Cisco IOS Release 12.0(30)S に統合されました。 この機能は、Cisco IOS Release 12.2(27)SBC に統合されました。 この機能は、Cisco IOS Release 12.2(28)SB に統合されました。 この機能は、Cisco IOS Release 12.2(33)SRA に統合されました。 この機能は、Cisco IOS Release 12.2(33)SXH に統合されまし

機能名	リリース	機能情報
		た。 この機能が Cisco IOS XE Release 2.1 に統合され、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに実装されました。 次のコマンドが導入または変更されました。**snmp-server enable traps mpls ldp**
MPLS LDP—MIB 通知	Cisco IOS XE Release 2.1	この機能は、重要な MPLS LDP イベントに対して SNMP トラップを提供します。 この機能は、Cisco ASR 1000 シリーズのアグリゲーションサービスルータで導入されました。 次のコマンドが導入または変更されました。**snmp-server enable traps mpls ldp**

第 7 章

MPLS ラベル配布プロトコル MIB バージョン 8 アップグレード

MPLS ラベル配布プロトコル（LDP）MIB バージョン 8 アップグレード機能により、Internet Engineering Task Force（IETF）ドラフト バージョン 8 をサポートするよう LDP MIB が拡張されます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS LDP MIB バージョン 8 アップグレードの前提条件

- ラベル スイッチ ルータ（LSR）上に簡易ネットワーク管理プロトコル（SNMP）をインストールしてイネーブルにする必要があります。
- LSR でマルチプロトコル ラベル スイッチング（MPLS）がイネーブルになっている。
- LDP が LSR でイネーブルになっている必要があります。

MPLS LDP MIB バージョン 8 アップグレードの制約条件

MPLS LDP MIB のこの実装では、MIB オブジェクトに対するアクセス権が読み取り専用（RO）に制限されます。ただし、SNMP エージェントによる書き込みが可能になるように拡張された MIB オブジェクト *mplsLdpSessionUpDownTrapEnable* は例外です。

このオブジェクトの値を true に設定すると、LSR で *mplsLdpSessionUp* 通知と *mplsLdpSessionDown* 通知の両方がイネーブルになります。逆に、このオブジェクトの値を false に設定すると、これらの通知がいずれもディセーブルになります。

イベント通知については、「MPLS LDP MIB バージョン 8 アップグレードでの MPLS LDP MIB の通知生成イベント」を参照してください。

ほとんどの MPLS LDP MIB オブジェクトは、LDP ピアのディスカバリ（hello）プロセス、および以降の LDP ピア間 LDP セッションのパラメータや確立のネゴシエーション中に自動的に設定されます。

次のテーブルは、この機能では実装されていません。

- mplsLdpEntityFrParmsTable
- mplsLdpEntityConfFrLRTable
- mplsLdpFrameRelaySesTable
- mplsFecTable
- mplsLdpSesInLabelMapTable
- mplsXCsFecsTable
- mplsLdpSesPeerAddrTable

MPLS LDP MIB バージョン 8 アップグレードに関する情報

MPLS LDP MIB バージョン 8 アップグレードの機能設計

MPLS は、パケット転送テクノロジーであり、パケットでラベルと呼ばれる短い固定長の値を使用して、ラベル スイッチ ルータ（LSR）による MPLS ネットワークでのパケット転送のネクストホップを指定します。

基本的な MPLS の原則は、MPLS ネットワーク内の LSR は、パケット転送操作に使用するラベルの定義で一致している必要があるということです。ラベルの同意は、LDP で定義されている手順によって MPLS ネットワークで行われます。

LDP 操作では最初に検出（hello）プロセスが実行されます。このプロセスでは、LDP エントリ（ローカル LSR）がネットワーク内で連携する LDP ピアを検出し、この両者が基本操作プロシージャをネゴシエートします。この検出プロセスによりピアが認識および特定されると、hello 隣接が確立されます。hello 隣接は、ローカル LSR とその LDP ピアの間でラベル バインド情報が交換される状況を表します。次に LDP は 2 つの LSR の間でアクティブな LDP セッションを確立し、ラベルバインド情報が交換できるようにします。このプロセスが MPLS ネットワーク内のすべての LSR に関して完了すると、通信ネットワーク デバイス間のエンドツーエンドのパケット伝送経路を構成するラベル スイッチド パス（LSP）が確立されます。

LDP により、LSR はラベル バインド情報を収集し、MPLS ネットワーク内の他のデバイスに配布および解放します。これにより、ネットワーク内で通常ルーティング パスに沿ったパケットのホップバイホップ転送が有効になります。

MPLS LDP MIB は、Cisco ソフトウェアにおけるラベル スイッチング機能について標準の SNMP ベースのネットワーク管理を実行できるようするために実装されました。この機能を使用するには、ネットワーク内の指定したネットワーク管理ステーション（NMS）で SNMP エージェント コードを実行する必要があります。NMS は、MPLS LDP MIB 内のネットワーク管理オブジェクトとユーザの対話の媒体となります。

SNMP エージェント コードは、Cisco ソフトウェアと互換性のある階層構造を持ち、MPLS LDP MIB 内のオブジェクト、さらに Cisco ソフトウェアによってサポートされる豊富なラベル スイッチング機能一式とのネットワーク管理インターフェイスを提供します。

SNMP エージェントにより、SNMP GET 操作を使用して MPLS LDP MIB オブジェクトにアクセスでき、これらのオブジェクトを使用してさまざまなネットワーク管理タスクを実行できます。MPLS LDP MIB のすべてのオブジェクトは、IETF ドラフト MIB（*draft-ietf-mpls-ldp-mib-08.txt*）に定義されている規則に従います。このドラフト MIB は、構造的および標準的な方法でネットワーク管理オブジェクトを定義します。このドラフト MIB は改訂が進んでおり、今後標準となる予定です。そのため、MPLS LDP MIB は、この IETF ドキュメントの今後の改訂を追跡できる方法で実装されます。

ただし、IETF ドラフト MIB と、これに相当する Cisco 機能の実装には若干の違いがあります。その結果、MPLS LDP MIB オブジェクトと内部 Cisco データ構造の間での小規模な変換が必要となります。このような変換は SNMP エージェントにより実行されます。SNMP エージェントは NMS ワークステーション上で、優先度が低いプロセスとしてバックグラウンドで実行されます。

豊富な Cisco ラベル スイッチング機能によって、WAN で伝送された大量のトラフィックについて統合型の管理が可能になります。これらの機能はレイヤ 3 ネットワーク サービスに統合され、インターネット サービスプロバイダーのバックボーンを通過する大量のトラフィックのルーティングが最適化されます。同時に、リンクやノードの障害に対するネットワークの耐性が確保されます。

MPLS Label Distribution Protocol MIB バージョン 8 アップグレードでは次の機能がサポートされています。

- Tag Distribution Protocol（TDP）（このプロトコルは一部のソフトウェア リリースでサポートされていない可能性があります）。
- LDP セッションのステータスの変更を伝えるイベント通知メッセージの生成と送信
- 既存の SNMP CLI コマンドの拡張によってイベント通知メッセージの有効化または無効化
- 動作環境で、ネットワーク管理の目的で Cisco イベント通知メッセージが送信される NMS ワークステーションの名前または IP アドレスの指定
- NMS の NVRAM へのイベント通知メッセージに関連する設定の保管

MPLS LDP MIB の構造は、抽象構文記法 1（ASN.1）に準拠しているため、この MIB は高度に構造化された理想的なネットワーク管理オブジェクト データベースを形成します。

標準の SNMP アプリケーションを使用して、標準の SNMP GET 操作および GETNEXT 操作によって MPLS LDP MIB から情報を取得して表示できます。

（注） MPLS LDP MIB の実装時点では、この MIB には Internet Assigned Numbers Authority（IANA）の Experimental Object ID（OID）が指定されなかったため、シスコでは ciscoExperimental OID 番号を使用してこの MIB を実装しました（ciscoExperimental 1.3.6.1.4.1.9.10 mplsLdpMIB 1.3.6.1.4.1.9.10.65 ）。MPLS LDP MIB に IANA Experimental OID 番号が割り当てられる場合、シスコは ciscoExperimental OID が設定されたこの MIB のすべてのオブジェクトを置き替え、これらのオブジェクトを IANA Experimental OID に移行します。

MPLS LDP MIB バージョン 8 の機能拡張

MPLS LDP MIB のバージョン 8 には、次の機能拡張が含まれています。

- TDP サポート（このプロトコルは一部のソフトウェアでサポートされていない可能性があります）。
- アップグレードされたオブジェクト
- セッション数に基づかない新しいインデックス
- バーチャル プライベート ネットワーク（VPN）の複数 SNMP コンテキスト サポート

MPLS LDP MIB バージョン 8 アップグレードの利点

- TDP と LDP をサポートする（TDP は一部のソフトウェア リリースでサポートされていない可能性があります）。
- MPLS ネットワーク内のピア デバイス間の LDP セッションを確立する。
- 次のような LDP エンティティの操作に関連する MIB パラメータを取得する。
  - 既知の LDP ディスカバリ ポート
  - 最大伝送ユニット（MTU）
  - 提示されるキープアライブ タイマー インターバル
  - ループ検出
  - セッション確立しきい値
  - ラベルの形成に使用される仮想パス識別子/仮想チャネル識別子（VPI/VCI）ペアの範囲
- LDP の動作に関連する統計情報（エラー数など）を収集する。
- hello 隣接の残り時間を監視する。
- 次のような LDP ピアの特性とステータスを監視する。
  - LDP ピアのインターネットワーク層アドレス
  - LDP ピアのループ検出
  - LDP ピアのデフォルト MTU
  - LDP ピアがキープアライブ インターバルの値として提示する秒数
- 次のような LDP セッションの特性とステータスを監視する。
  - エラー数の表示
  - LDP セッションが使用している LDP のバージョンの確認
  - LDP セッションのキープアライブ保留時間の確認
  - LDP セッションの状態の確認（セッションがアクティブかどうか）
  - プラットフォーム全体およびインターフェイス固有のセッションのラベル範囲の表示
  - ATM パラメータの表示

MPLS LDP MIB バージョン 8 アップグレードの MPLS LDP MIB 要素の説明

MPLS LDP MIB に関連する LDP 動作には、次の機能要素が関与します。

- LDP エンティティ：ラベル スペースの交換を目的としており、LDP インスタンスに関連します。発生する可能性があるセッションを記述します。
- LDP ピア：リモート LDP エンティティ（つまり非ローカル LSR）を意味します。
- LDP セッション：ローカル LSR とリモート LDP ピアの間のアクティブな LDP プロセスを意味します。
- Hello 隣接：MPLS ネットワーク内の 2 つの LSR の状態が、相互に隣接している（つまり LDP ピアである）ことを示す LDP ディスカバリ プロセスの結果を意味します。ネイバーが検出されると、そのネイバーは hello 隣接になります。hello 隣接との LDP セッションを確立できます。セッションの確立後に、LSR 間でラベル バインドが交換されます。

これらの MPLS LDP MIB 要素について、以降で個々の見出しの下で簡潔に説明します。

実際には、MPLS LDP MIB は、データベース内の各種 MIB オブジェクトへのリアルタイム アクセスをサポートするネットワーク管理データベースを提供します。このデータベースは、ネットワークでの MPLS LDP の現在の状態を反映します。このネットワーク管理情報データベースにアクセスするには、MPLS/LDP 稼働環境で NMS から標準の SNMP コマンドを発行します。

MPLS LDP MIB は、次のネットワーク管理アクティビティをサポートしています。

- LDP 動作に関連する MPLS LDP MIB パラメータの取得
- LDP ピアの特性とステータスの監視
- LDP ピア間の LDP セッションのステータスの監視
- ネットワークにおける Hello 隣接の監視
- LDP セッションに関する統計情報の収集

LDP エンティティ

LDP エンティティは、mplsLdpEntityLdpId と mplsLdpEntityIndex（次の図を参照）で構成される LDP ID によって一意に識別されます。

- mplsLdpEntityLdpId は、ローカル LSR ID（4 オクテット）とラベル スペース ID（2 オクテット）で構成されます。ラベル スペース ID は、LSR 内で使用可能な特定のラベル スペースを識別します。
- mplsLdpEntityIndex は、ピアのアクティブな hello 隣接の IP アドレス（ピア LSR に割り当てられた IP アドレスの 32 ビット表現）で構成されます。

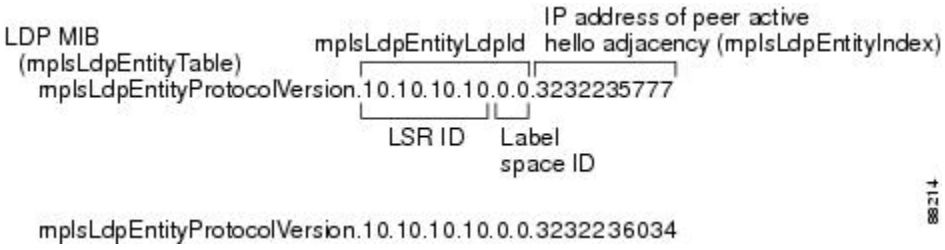
mplsldpEntityProtocolVersion は、mplsLdpEntityTable のサンプル オブジェクトです。

この図は、次のインデックスを示しています。

- mplsLdpEntityLdpId = 10.10.10.10.0.0
- LSR ID = 10.10.10.10
- ラベル スペース ID = 0.0

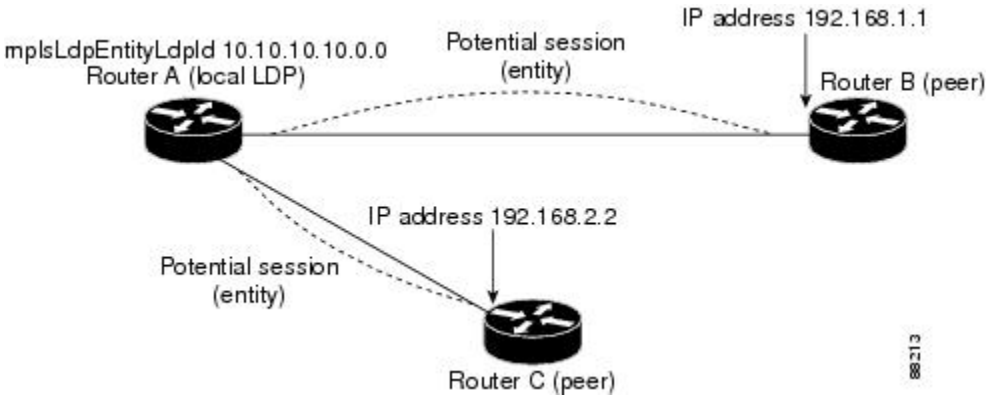
mplsLdpEntityLdpId または LDP ID は、LSR ID とラベル スペース ID で構成されます。

- ピアのアクティブな hello 隣接の IP アドレスまたは mplsLdpEntityIndex = 3232235777（ピアのアクティブな hello 隣接に割り当てられた ID の 32 ビット表現）。



LDP エンティティは、LDP ピアとの間でセッションが発生する可能性があるラベル スペースを表します。LDP エンティティは、hello 隣接が LDP ピアから hello メッセージを受信すると設定されます。

次の図では、ルータ A と 2 つのリモート ピア（ルータ B および C）の間でセッションが発生している可能性があります。mplsLdpEntityLdpId は 10.10.10.10.0.0、ピアのアクティブな hello 隣接の IP アドレス（mplsLdpEntityIndex）は 3232235777（ルータ B の IP アドレス 192.168.1.1 の 32 ビット表現）です。



LDP セッションおよびピア

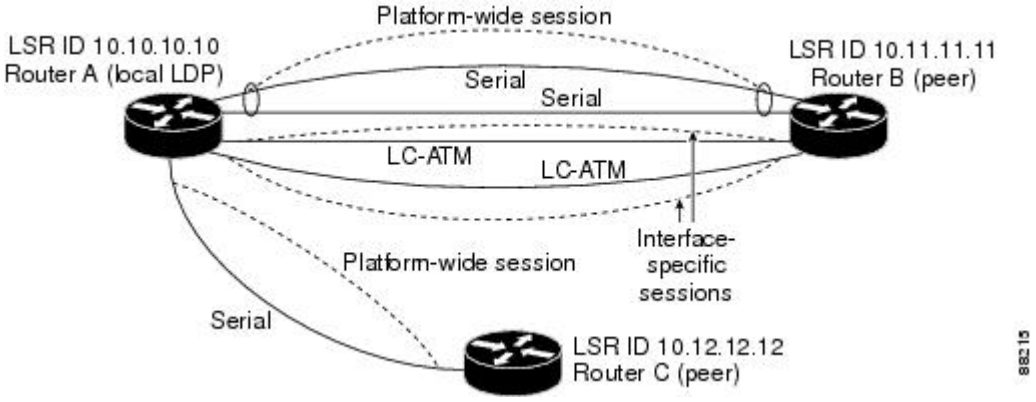
LDP セッションは、ラベル スペース配布の目的でローカル エンティティとリモート ピアの間で確立されます。LDP ピアと LDP セッションの間の関係は常に 1 対 1 です。単一 LDP セッションは、1 つ以上のネットワーク リンク経由で単一 LDP ピアと通信する LDP インスタンスです。

LDP では、次のタイプのセッションがサポートされています。

- インターフェイス固有：インターフェイス固有のセッションでは、ラベル スペースの配布にインターフェイス リソースを使用します。たとえば、各ラベル制御 ATM（LC-ATM）インターフェイスでは、ラベル スペースの配布に独自の VPI および VCI を使用します。設定によって、LDP プラットフォームでサポートされるインターフェイス固有のセッションの数は、ゼロ、1 つ、または複数です。各 LC-ATM インターフェイスには、専用のインターフェイス固有ラベル スペースと、ゼロ以外のラベル スペース ID があります。
- プラットフォーム全体：LDP プラットフォームでは、1 つのプラットフォーム全体のセッションがサポートされます。プラットフォーム全体のセッションは、同じグローバル ラベル スペースを共有できるすべてのインターフェイスによって使用されます。シスコのプラットフォームでは、LC-ATM を除くすべてのインターフェイス タイプでプラットフォーム全体のセッションが使用され、ラベル スペース ID はゼロです。

2 つのピア間でセッションが確立されると、mplsLdpPeerTable と mplsLdpSessionTable にエントリが作成されます。これは、これらのテーブルのインデックスが同一であるためです。

次の図では、ルータ A には 2 つのリモート ピア（ルータ B および C）があります。ルータ A とルータ B の間では、2 つのシリアル インターフェイスで構成される 1 つのプラットフォーム全体のセッションがあり、ルータ C との間ではもう 1 つのプラットフォーム全体のセッションがあります。ルータ A とルータ B の間には 2 つのインターフェイス固有セッションもあります。



次の図に、mplsLdpPeerTable と上記の図の mplsLdpSessionTable に対応するエントリを示します。

次の図では、mplsLdpSesState はルータ A 上の mplsLdpSessionTable のサンプル オブジェクトです。4 つの mplsLdpSesState サンプル オブジェクトが示されています（上から下）。最初のオブジェクトは、2 つのシリアル インターフェイスに関連付けられているプラットフォーム全体のセッションを表します。次の 2 つのオブジェクトは、ルータ A と B 上にある LC-ATM インターフェイスのインターフェイス固有のセッションを表しています。これらのインターフェイス固有のセッションには、ゼロ以外のピア ラベル スペース ID があります。最後のオブジェクトは、次のピア、ルータ C とのプラットフォーム全体のセッションを表しています。

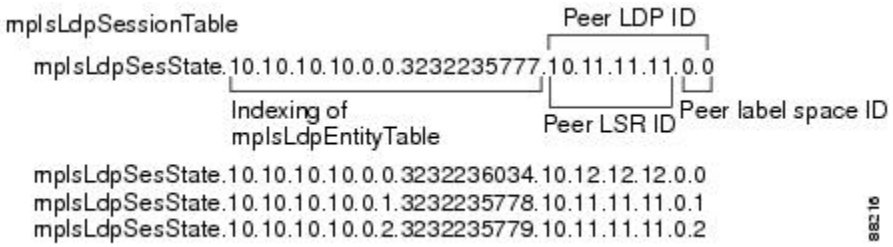
インデックスは mplsLdpEntityTable のエントリに基づいています。mplsLdpEntityTable のインデックスで始まり、次の値が付加されます。

- ピア LDP ID = 10.11.11.11.0.0

ピア LDP ID は、LSR ID（4 オクテット）とピア ラベル スペース ID（2 オクテット）で構成されます。

- ピア LSR ID = 10.11.11.11
- ピア ラベル スペース ID = 0.0

ピア ラベル スペース ID は、LSR 内で使用可能な特定のピア ラベル スペースを識別します。



LDP Hello 隣接

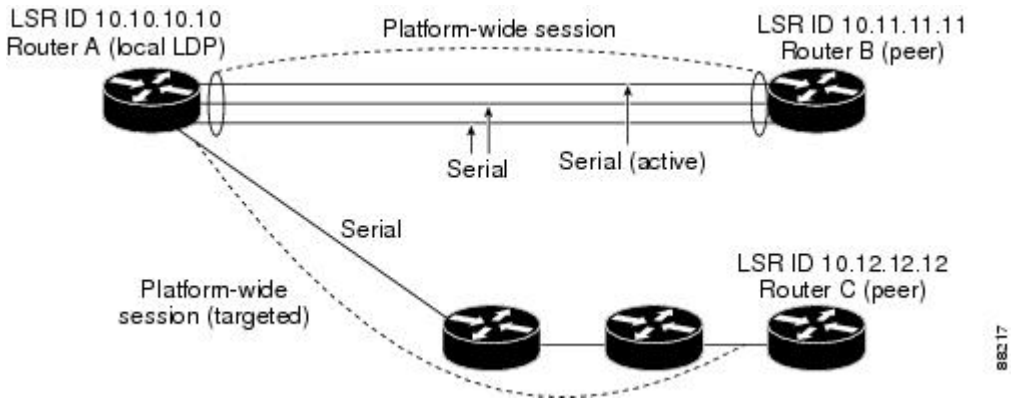
LDP hello 隣接は、ルータとそのピアの間のネットワーク リンクです。LDP hello 隣接によって、2 つの隣接ピアはラベル バインディング情報を交換できます。

LDP が実行されるリンクごとに LDP hello 隣接が存在します。ルータとそのピア間のセッション（プラットフォーム全体のセッションなど）に複数のリンクがある場合は、常に複数の LDP hello 隣接が存在します。

hello 隣接は、現在セッションに関わっている場合はアクティブと見なされ、関わっていない場合は非アクティブと見なされます。

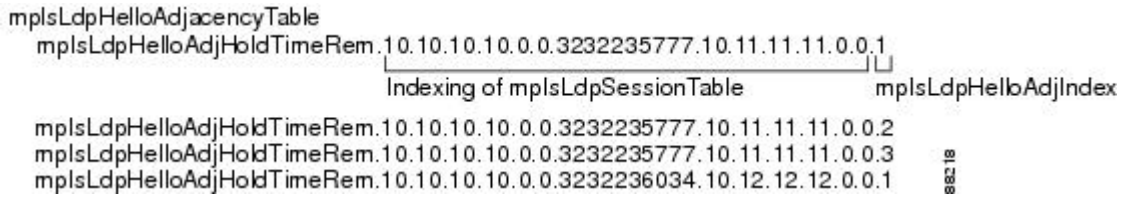
ターゲット hello 隣接は、そのピアに直接接続されず、ピアとの間のホップ カウントに制限がありません。リンク hello 隣接は、2 つのルータ間で直接接続されます。

次の図では、ルータ A には 2 つのリモート ピア（ルータ B および C）があります。ルータ A とルータ B の間にはプラットフォーム全体のセッションがあり、このセッションは 3 つのシリアル インターフェイスで構成されていて、そのうち 1 つのシリアル インターフェイスがアクティブです。また、ルータ C との間にもう 1 つのプラットフォーム全体の（ターゲット）セッションがあります。



次の図に、mplsLdpHelloAdjacencyTable のエントリを示します。上から下へ 4 つの mplsLdpHelloAdjHoldTime サンプル オブジェクトがあります。これらは 2 つのプラットフォーム全体のセッションと、上記の図に示されている 4 つのシリアル リンクを表します。

インデックスは mplsLdpSessionTable に基づいています。mplsLdpHelloAdjIndex によって 1 つのセッション内に異なるリンクが列挙される場合、アクティブ リンクは mplsLdpHelloAdjIndex = 1 です。



MPLS LDP MIB バージョン 8 アップグレードでの MPLS LDP MIB 通知生成イベント

snmp-server enable traps mpls ldp コマンドを発行して MPLS LDP MIB 通知機能を有効にする場合、ネットワーク内で特定のイベントが発生したことを伝えるため、通知メッセージが生成され、指定された NSM に送信されます。

LDP ステータス移行とイベント通知に関連する MPLS LDP MIB オブジェクトには、次のメッセージが含まれます。

- mplsLdpSessionUp：このメッセージは、LDP エンティティ（ローカル LSR）によって別の LDP エンティティ（ネットワーク内の隣接 LDP ピア）との LDP セッションが確立されると生成されます。
- mplsLdpSessionDown：このメッセージは、ローカル LSR とその隣接 LDP ピア間の LDP セッションが終了すると生成されます。
- mplsLdpPathVectorLimitMismatch：このメッセージは、ローカル LSR によって、その隣接ピアである LSR との LDP セッションが確立され、2 つの LSR でパス ベクトル制限が異なる場合に生成されます。

パス ベクトル制限の値の範囲は 0 ～ 255 です。値が 0 の場合、ループ検出はオフです。0 以外の 255 までの値の場合、ループ検出はオンで、さらにネットワーク内のループ状態が検知されるまでに LDP メッセージが通過できるホップの最大数が示されます。

ネットワーク内のすべての LDP 対応ルータに同じパス ベクトル制限を設定することを推奨します。mplsLdpPathVectorLimitMismatch オブジェクトが MPLS LDP MIB に存在するは、LDP 動作に関わっている 2 つのルータのパス ベクトル制限が異なる場合に NMS に警告メッセージを送信するためです。

（注） この通知が生成されるのは、配布方式がダウンストリームオンデマンドである場合だけです。

- mplsLdpFailedInitSessionThresholdExceeded：このメッセージは、ローカル LSR と隣接 LDP ピアが、それらの間に LDP セッションを確立しようとして失敗し、その試行回数が指定数を超えた場合に生成されます。デフォルトの試行回数は 8 回です。このデフォルト値は実装されており、変更できません。

デバイス間での非互換性が原因で、ローカル LSR と LDP ピア間の LDP セッションの確立に 8 回失敗しました。このため、この通知メッセージが生成されます。Cisco ルータは複数のプラットフォームで同じ機能をサポートします。

したがって、Cisco LSR 間で最も発生する可能性が高い非互換は、それぞれの ATM VPI/VCI ラベル範囲のミスマッチです。

たとえば、LSR に有効なラベルの範囲を指定し、その範囲が隣接 LDP ピアの範囲と重ならない場合、ルータは LDP ピアとの LDP セッションを 8 回確立しようとします。その後、mplsLdpFailedInitSessionThresholdExceeded 通知が生成され、情報メッセージとして NMS に送信されます。

ラベル範囲が重ならない LSR は、8 回のリトライしきい値を超えても、それらの LSR 間の LDP セッションを確立しようとし続けます。

そのような場合、LDP しきい値超過通知によって、ネットワーク内に注意すべき状態があることがネットワーク管理者に知らされます。

RFC 3036『*LDP Specification*』に、MPLS ネットワーク内の Cisco ルータやその他の LSR 間に存在する可能性がある非互換について詳しく記載されています。

このような非互換の例を次に示します。

- - LDP セッションのセットアップを試行する LSR の間で、ATM VPI/VCI 範囲（前述）が重複していない、またはフレーム リレー DLCI 範囲が重複していない。
  - ラベル配布方式がサポートされていない。
  - プロトコル データ ユニット（PDU）サイズが異なる。
  - LDP 機能のサポートのタイプが異なる。

MPLS LDP MIB バージョン 8 アップグレードの MIB テーブル

MPLS LDP MIB のバージョン 8 には、次のテーブルが含まれています。

- mplsLdpEntityTable：すべてのアクティブな LDP hello 隣接のエントリが格納されます。非アクティブな hello 隣接は、このテーブルではなく mplsLdpHelloAdjacencyTable に表示されます。このテーブルのインデックスは、インターフェイスのローカル LDP ID とピアのアクティブな hello 隣接の IP アドレスに基づいて作成されます

このテーブルでセッションの代わりにアクティブな hello 隣接を表示するメリットは、LDP セッションがアクティブではない場合（確立できない場合）でも、アクティブな hello 隣接が存在できる点です。IETF MPLS-LDP MIB の以前の実装では、このテーブルのエントリとしてセッションが使用されていました。これは適切な方法ではありませんでした。セッションがダウンすると、エージェントコードがセッションにアクセスできなくなるため、エンティティテーブル内のエントリが完全に失われるからです。その結果、MIB は失敗した LDP セッションに関する情報を提供できなくなります。

誘導隣接もこのテーブルに表示されます。ただし、誘導セッションが失敗すると隣接が失われるため、これらのエントリは、管理上（adminStatus）および動作上（operStatus）常にアップ状態になります。基礎となるインターフェイスが動作上ダウンした場合などは隣接が削除されるため、非誘導隣接が MIB から失われることがあります。

- mplsLdpEntityConfGenLRTable：グローバル ラベル スペース内にあるすべての LDP 対応インターフェイスのエントリが格納されます（シスコでは、これは LC-ATM を除くすべてのインターフェイスに適用されます。LC-ATM エンティティは代わりに mplsLdpEntityConfAtmLRTable に表示されます。）インデックスは、2 つのインデックス（mplsLdpEntityConfGenLRMin と mplsLdpEntityConfGenLRMax）が追加されていることを除き、mplsLdpEntityTable の場合と同じです。これらの追加インデックスを使用すると、複数のラベル範囲を定義できます。ただし、現在のシスコ実装では、グローバル ラベル範囲は 1 つしか許可されません。
- mplsLdpEntityAtmParmsTable：すべての LDP 対応 LC-ATM インターフェイスのエントリが格納されます。このテーブルのインデックスは mplsLdpEntityTable の場合と同じですが、LC-ATM インターフェイスだけが表示されます。
- mplsLdpEntityConfAtmLRTable：すべての LDP 対応 LC-ATM インターフェイスのエントリが格納されます。インデックスは、2 つのインデックス（mplsLdpEntityConfAtmLRMinVpi と mplsLdpEntityConfAtmLRMinVci）が追加されていることを除き、mplsLdpEntityTable の場合と同じです。これらの追加インデックスを使用すると、複数のラベル範囲を定義できます。ただし、現在のシスコ実装では、LC-ATM インターフェイスごとに 1 つのラベル範囲だけが許可されています。
- mplsLdpEntityStatsTable：mplsLdpEntityTable を拡張し、GET 操作と GETNEXT 操作の実行でまったく同一のインデックスを共有します。このテーブルには、エンティティの追加統計情報が表示されます。
- mplsLdpPeerTable：すべてのピア セッションのエントリが格納されます。このテーブルのインデックスは、セッションのローカル LDP ID、ピアのアクティブな hello 隣接の IP アドレス、およびピアの LDP ID に基づいて作成されます。

- mplsLdpHelloAdjacencyTable：すべての hello 隣接のエントリが格納されます。このテーブルのインデックスは、関連付けられたセッションのローカル LDP ID、ピアのアクティブな hello 隣接の IP アドレス、ピアの LDP ID、および隣接のリスト位置に設定された任意のインデックスに基づいて作成されます
- mplsLdpSessionTable：mplsLdpPeerTable を拡張し、GET 操作と GETNEXT 操作の実行で同じインデックスを共有します。このテーブルにはすべてのセッションが表示されます。
- mplsLdpAtmSesTable：LC-ATM セッションのエントリが格納されます。インデックスは、2 つのインデックス（mplsLdpSesAtmLRLowerBoundVpi と mplsLdpSesAtmLRLowerBoundVci）が追加されていることを除き、mplsLdpPeerTable の場合と同じです。これらの追加インデックスを使用すると、複数のラベル範囲を定義できます。ただし、現在のシスコ実装では、LC-ATM インターフェイスごとに 1 つのラベル範囲だけが許可されています。
- mplsLdpSesStatsTable：mplsLdpPeerTable を拡張し、GET 操作と GETNEXT 操作の実行でまったく同じインデックスを共有します。このテーブルには、セッションの追加統計情報が表示されます。

mplsLdpEntityTable

次の表に、mplsLdpEntityTable のオブジェクトとその説明を示します。

表 *28*：*mplsLdpEntityTable* のオブジェクトと説明

オブジェクト	説明
mplsLdpEntityEntry	2 つのピア間の潜在的なセッションである LDP エンティティを表します。
mplsLdpEntityLdpId	LDP ID（アクセス不能）は、ローカル LSR ID（4 オクテット）とラベル スペース ID（2 オクテット）で構成されます。
mplsLdpEntityIndex	この行を一意に識別するセカンダリ インデックス。ピアのアクティブな hello 隣接の IP アドレス（LSR に割り当てられた IP アドレスの 32 ビット表現）で構成されます（アクセス不能）。
mplsLdpEntityProtocolVersion	セッション初期化メッセージで使用される LDP プロトコルのバージョン番号。
mplsLdpEntityAdminStatus	この LDP エンティティの管理ステータスは常にアップです。hello 隣接が失敗した場合、このエンティティは mplsLdpEntityTable から失われます。

オブジェクト	説明
mplsLdpEntityOperStatus	この LDP エンティティの動作ステータス。値は、unknown(0)、enabled(1)、disabled(2) のいずれかです。
mplsLdpEntityTcpDscPort	LDP または TDP の TCP ディスカバリ ポート。デフォルト値は 646（LDP）です。
mplsLdpEntityUdpDscPort	LDP または TDP の UDP ディスカバリ ポート。デフォルト値は 646（LDP）です。
mplsLdpEntityMaxPduLength	初期化メッセージの共通セッションパラメータで送信される最大 PDU 長。
mplsLdpEntityKeepAliveHoldTimer	この LDP エンティティについて提示されるキープアライブ保留時間である 2 オクテットの値。
mplsLdpEntityHelloHoldTimer	この LDP エンティティについて提示される hello 保留時間である 2 オクテットの値。
mplsLdpEntityInitSesThreshold	このエンティティとそのピアが初期化メッセージの無限シーケンスに関わっている場合の通知のしきい値。 デフォルト値は 8 で、SNMP でも CLI でも変更できません。
mplsLdpEntityLabelDistMethod	特定の LDP セッションに指定されたラベル配布方法。値は、downstreamOnDemand(1) と downstreamUnsolicited(2) です。
mplsLdpEntityLabelRetentionMode	LC-ATM に conservative(1) を使用するか、またはその他のすべてのインターフェイスに liberal(2) を使用するように設定できます。
mplsLdpEntityPVLMisTrapEnable	mplsLdpPVLMismatch トラップが生成されるかどうかを示します。 値が enabled(1) の場合、トラップは生成されます。値が disabled(2) の場合、トラップは生成されません。デフォルトは disabled(2) です。 （注） mplsLdpPVLMismatch トラップが生成されるのは、mplsLdpEntityLabelDistMethod が downstreamOnDemand(1) の場合だけです。

オブジェクト	説明
mplsLdpEntityPVL	このオブジェクトの値が 0 の場合、パス ベクトルのループ検出はディセーブルになっています。値がゼロ（0）より大きい場合、パス ベクトルのループ検出はイネーブルになっており、その値がパス ベクトル制限になります。 （注） mplsLdpEntityPVL オブジェクトがゼロ（0）以外になるのは、mplsLdpEntityLabelDistMethod が downstreamOnDemand(1) の場合だけです。
mplsLdpEntityHopCountLimit	このオブジェクトの値が 0 の場合、ホップ カウンタを使用したループ検出はディセーブルになっています。 値がゼロ（0）より大きい場合、ホップ カウンタを使用したループ検出はイネーブルになっており、このオブジェクトによって、このエンティティのホップ カウントの最大許容値が指定されます。 （注） mplsLdpEntityHopCountLimit オブジェクトがゼロ（0）以外になるのは、mplsLdpEntityLabelDistMethod が downstreamOnDemand(1) の場合だけです。
mplsLdpEntityTargPeer	この LDP エンティティでターゲット隣接が使用されている場合、このオブジェクトは true(1) に設定されます。デフォルト値は false(2) です。
mplsLdpEntityTargPeerAddrType	拡張ディスカバリに使用されるインターネットワーク層アドレスのタイプ。このオブジェクトは、mplsLdpEntityTargPeerAddr の値の解釈方法を示します。
mplsLdpEntityTargPeerAddr	ターゲット隣接に使用されるインターネットワーク層アドレスの値。

オブジェクト	説明
mplsLdpEntityOptionalParameters	LDP 初期化メッセージのオプション パラメータを指定します。値が generic(1) の場合、このエンティティに関連付けられた LDP 初期化メッセージでオプション パラメータは送信されません。 LC-ATM では、atmParameters(2) を使用して、mplsLdpEntityAtmParmsTable の行がこのエントリに対応することを指定します。 （注） フレーム リレー パラメータはサポートされません。
mplsLdpEntityDiscontinuityTime	このエンティティのカウンタが 1 つ以上中断した場合、最後に中断したときの sysUpTime の値。関連するカウンタは、このエンティティに関連付けられた、mplsLdpEntityStatsTable に含まれている Counter32 または Counter64 オブジェクトの特定のインスタンスです。ローカル管理サブシステムを最後に再初期化してから中断が発生しなかった場合、このオブジェクトには値 0 が格納されます。
mplsLdpEntityStorType	このエントリのストレージタイプは、常に揮発の、読み取りのみの実装です。
mplsLdpEntityRowStatus	このオブジェクトは読み取り専用の実装であり、常に active です。

mplsLdpEntityConfGenLRTable

次の表に、mplsLdpEntityConfGenLRTable のオブジェクトとその説明を示します。

表 *29*：*mplsLdpEntityConfGenLRTable* のオブジェクトと説明

オブジェクト	説明
mplsLdpEntityConfGenLREntry	LDP Entity Configurable Generic Label Range Table 内の行。このテーブルの 1 つのエントリには、ラベルの 1 つの範囲に関する情報が格納されます。範囲は、上限（VPI/VCI ペア）と下限（VPI/VCI ペア）で定義されます。 現在の実装では、エンティティごとに 1 つのラベル範囲がサポートされています。

オブジェクト	説明
mplsLdpEntityConfGenLRMin	この範囲に設定されている最小ラベル（アクセス不能）。
mplsLdpEntityConfGenLRMax	この範囲に設定されている最大ラベル（アクセス不能）。
mplsLdpEntityConfGenIfIndxOrZero	この値は、プラットフォーム全体のエンティティの SNMP IF-MIB インデックスを表します。アクティブな hello 隣接がターゲットの場合、この値は 0 です。
mplsLdpEntityConfGenLRStorType	このエントリのストレージタイプは、常に揮発の、読み取りのみの実装です。
mplsLdpEntityConfGenLRRowStatus	このオブジェクトは読み取り専用の実装であり、常に active です。

mplsLdpEntityAtmParmsTable

次の表に、mplsLdpEntityAtmParmsTable のオブジェクトとその説明を示します。

表 *30* : ***mplsLdpEntityAtmParmsTable*** のオブジェクトと説明

オブジェクト	説明
mplsLdpEntityAtmParmsEntry	この LDP エンティティの ATM パラメータと ATM 情報を表します。
mplsLdpEntityAtmIfIndxOrZero	この値は、インターフェイス固有の LC-ATM エンティティの SNMP IF-MIB インデックスを表します。
mplsLdpEntityAtmMergeCap	このエンティティのマージ機能を表します。
mplsLdpEntityAtmLRComponents	初期化メッセージのラベル範囲コンポーネントの数。このエントリに対応する mplsLdpEntityConfAtmLRTable 内のエントリ数も表します。

オブジェクト	説明
mplsLdpEntityAtmVcDirectionality	このオブジェクトの値が bidirectional(0) の場合、特定の VPI 内の特定の VCI が両方向のラベルとして相互に独立して使用されます。 このオブジェクトの値が unidirectional(1) の場合、VPI 内の特定の VCI は一方向を指定します。
mplsLdpEntityAtmLsrConnectivity	ATM VP によってピア LSR を間接的に接続できるため、VPI 値はエンドポイント上で異なる場合があります。そのため、ラベルを VCI フィールド内で完全に符号化する必要があります。 値は direct(1)（デフォルト）と indirect(2) です。
mplsLdpEntityDefaultControlVpi	非 MPLS 接続のデフォルトの VPI 値。
mplsLdpEntityDefaultControlVci	非 MPLS 接続のデフォルトの VCI 値。
mplsLdpEntityUnlabTrafVpi	ラベル付けされていないトラフィックをサポートする VCC の VPI 値。この非 MPLS 接続は、ラベル付けされていない（IP）パケットの伝送に使用されます。
mplsLdpEntityUnlabTrafVci	ラベル付けされていないトラフィックをサポートする VCC の VCI 値。この非 MPLS 接続は、ラベル付けされていない（IP）パケットの伝送に使用されます。
mplsLdpEntityAtmStorType	このエントリのストレージタイプは、常に揮発の、読み取りのみの実装です。
mplsLdpEntityAtmRowStatus	このオブジェクトは読み取り専用の実装であり、常に active です。

mplsLdpEntityConfAtmLRTable

次の表に、mplsLdpEntityConfAtmLRTable のオブジェクトとその説明を示します。

表 *31*：*mplsLdpEntityConfAtmLRTable* のオブジェクトと説明

オブジェクト	説明
mplsLdpEntityConfAtmLREntry	LDP Entity Configurable ATM Label Range Table 内の行。このテーブルの 1 つのエントリには、ラベルの 1 つの範囲に関する情報が格納されます。範囲は、上限（VPI/VCI ペア）と下限（VPI/VCI ペア）で定義されます。これは、初期化メッセージで使用されるのと同じデータです。このラベル範囲はピアのラベル範囲と重なっている必要があります。
mplsLdpEntityConfAtmLRMinVpi	この範囲に設定されている最小 VPI 番号（アクセス不能）。
mplsLdpEntityConfAtmLRMinVci	この範囲に設定されている最小 VCI 番号（アクセス不能）。
mplsLdpEntityConfAtmLRMaxVpi	この範囲に設定されている最大 VPI 番号（アクセス不能）。
mplsLdpEntityConfAtmLRMaxVci	この範囲に設定されている最大 VCI 番号（アクセス不能）。
mplsLdpEntityConfAtmLRStorType	このエントリのストレージタイプは、常に揮発の、読み取りのみの実装です。
mplsLdpEntityConfAtmLRRowStatus	このオブジェクトは読み取り専用の実装であり、常に active です。

mplsLdpEntityStatsTable

次の表に、mplsLdpEntityStatsTable のオブジェクトとその説明を示します。

表 *32*：*mplsLdpEntityStatsTable* のオブジェクトと説明

オブジェクト	説明
mplsLdpEntityStatsEntry	これらのエントリは、各エントリの追加情報を提供することによって mplsLdpEntityTable を拡張します。
mplsLdpAttemptedSessions	この機能ではサポートされていません。

オブジェクト	説明
mplsLdpSesRejectedNoHelloErrors	この LDP エンティティによって拒否されたセッション、または送受信された no hello エラー通知メッセージのカウント。
mplsLdpSesRejectedAdErrors	この LDP エンティティによって拒否されたセッション、または送受信されたパラメータ アドバタイズメント モード エラー通知メッセージのカウント。
mplsLdpSesRejectedMaxPduErrors	この LDP エンティティによって拒否されたセッション、または送受信されたパラメータ最大 PDU 長エラー通知メッセージのカウント。
mplsLdpSesRejectedLRErrors	この LDP エンティティによって拒否されたセッション、または送受信されたパラメータ ラベル範囲通知メッセージのカウント。
mplsLdpBadLdpIdentifierErrors	この LDP エンティティに関連付けられているセッションによって検出された不正 LDP ID 重大エラー数のカウント。
mplsLdpBadPduLengthErrors	この LDP エンティティに関連付けられているセッションによって検出された不正 PDU 長重大エラー数のカウント。
mplsLdpBadMessageLengthErrors	この LDP エンティティに関連付けられているセッションによって検出された不正メッセージ長重大エラー数のカウント。
mplsLdpBadTlvLengthErrors	この LDP エンティティに関連付けられているセッションによって検出された不正 Type-Length-Value（TLV）長重大エラー数のカウント。
mplsLdpMalformedTlvValueErrors	この LDP エンティティに関連付けられているセッションによって検出された不正 TLV 値重大エラー数のカウント。
mplsLdpKeepAliveTimerExpErrors	この LDP エンティティに関連付けられているセッションによって検出されたセッション キープアライブ タイマー期限切れエラー数のカウント。

オブジェクト	説明
mplsLdpShutdownNotifReceived	この LDP エンティティに関連付けられているセッションについて受信したシャットダウン通知数のカウント。
mplsLdpShutdownNotifSent	この LDP エンティティに関連付けられているセッションについて送信されたシャットダウン通知数のカウント。

mplsLdpPeerTable

次の表に、mplsLdpPeerTable のオブジェクトとその説明を示します。

表 *33* : ***mplsLdpPeerTable*** のオブジェクトと説明

オブジェクト	説明
mplsLdpPeerEntry	セッションに関連する単一ピアの情報（アクセス不能）。 （注） このテーブルは、mplsLdpSessionTable によって拡張されます。
mplsLdpPeerLdpId	この LDP ピアの LDP ID（アクセス不能）は、ピア LSR ID（4 オクテット）とピア ラベル スペース ID（2 オクテット）で構成されます。
mplsLdpPeerLabelDistMethod	特定の LDP セッションのラベル配布方法。値は、downstreamOnDemand(1) と downstreamUnsolicited(2) です。

オブジェクト	説明
mplsLdpPeerLoopDetectionForPV	パス ベクトルに基づくループ検出が、このピアでディセーブルまたはイネーブルのいずれになっているかを示します。 ダウンストリーム未承諾配布（mplsLdpPeerLabelDistMethod が downstreamUnsolicited(2)）の場合、このオブジェクトの値は常に disabled(0) になり、ループ検出はディセーブルになります。 ダウンストリームオンデマンド配布（mplsLdpPeerLabelDistMethod が downstreamOnDemand(1)）の場合、パス ベクトルに基づくループ検出がイネーブルになっていれば、このオブジェクトの値は enabled(1) になります。
mplsLdpPeerPVL	このエントリの mplsLdpPeerLoopDetectionForPV の値が enabled(1) の場合、このオブジェクトは、このピアのパス ベクトル制限を表します。 このエントリの mplsLdpPeerLoopDetectionForPV の値が disabled(0) の場合、この値は 0 になります。

mplsLdpHelloAdjacencyTable

次の表に、mplsLdpHelloAdjacencyTable のオブジェクトとその説明を示します。

表 *34*：*mplsLdpHelloAdjacencyTable* のオブジェクトと説明

オブジェクト	説明
mplsLdpHelloAdjacencyEntry	各行が 1 つの LDP hello 隣接を表します。LDP セッションは、1 つ以上の hello 隣接を持つことができます（アクセス不能）。
mplsLdpHelloAdjIndex	この特定の隣接の識別名（アクセス不能）。アクティブな hello 隣接の mplsLdpHelloAdjIndex は 1 になります。
mplsLdpHelloAdjHoldTimeRem	この hello 隣接の残り時間。この間隔は、この hello 隣接に対応する次の hello 隣接を受信すると変わります。

オブジェクト	説明
mplsLdpHelloAdjType	この隣接は、このオブジェクトの値が link(1) である場合のリンク hello の結果です。それ以外の場合、この隣接は、ターゲット hello の結果であり、値は targeted(2) になります。

mplsLdpSessionTable

次の表に、mplsLdpSessionTable のオブジェクトとその説明を示します。

表 *35* : ***mplsLdpSessionTable*** のオブジェクトと説明

オブジェクト	説明
mplsLdpSessionEntry	このテーブルのエントリは、LDP エンティティと LDP ピア間の単一セッションに関する情報を表します。行内の情報は読み取り専用です。このテーブルは、mplsLdpPeerTable を拡張します。
mplsLdpSesState	セッションの現在の状態。すべての状態は、セッション ネゴシエーション動作の LDP または TDP ステート マシンに基づいています。 状態は次のとおりです。 • nonexistent(1) • initialized(2) • openrec(3) • opensent(4) • operational(5)
mplsLdpSesProtocolVersion	このセッションが使用している LDP プロトコルのバージョン。これは、セッションの初期化中にネゴシエーションされた LDP プロトコルのバージョンです。
mplsLdpSesKeepAliveHoldTimeRem	このセッションの残りのキープアライブ保留時間。
mplsLdpSesMaxPduLen	このセッションの LDP PDU の最大許容長。この値は、セッションの初期化中にネゴシエーションされた可能性があります。

オブジェクト	説明
mplsLdpSesDiscontinuityTime	このセッションのカウンタが 1 つ以上中断した場合、最後に中断したときの sysUpTime の値。関連するカウンタは、このセッションに関連付けられた、mplsLdpSesStatsTable に含まれている Counter32 または Counter64 オブジェクトの特定のインスタンスです。 このオブジェクトの初期値は、このテーブルにエントリが作成されたときの sysUpTime の値です。

mplsLdpAtmSesTable

次の表に、mplsLdpAtmSesTable のオブジェクトとその説明を示します。

表 *36*：*mplsLdpAtmSesTable* のオブジェクトと説明

オブジェクト	説明
mplsLdpAtmSesEntry	このテーブルのエントリは、LDP エンティティと LDP ピア間の単一のラベル範囲共通部分に関する情報を現します（アクセス不能）。
mplsLdpAtmSesLRLowerBoundVpi	この範囲の最小 VPI 番号（アクセス不能）。
mplsLdpAtmSesLRLowerBoundVci	この範囲の最小 VCI 番号（アクセス不能）。
mplsLdpAtmSesLRUpperBoundVpi	この範囲の最大 VPI 番号（読み取り専用）。
mplsLdpAtmSesLRUpperBoundVci	この範囲の最大 VCI 番号（読み取り専用）。

mplsLdpSesStatsTable

次の表に、mplsLdpSesStatsTable のオブジェクトとその説明を示します。

表 *37* : ***mplsLdpSesStatsTable*** のオブジェクトと説明

オブジェクト	説明
mplsLdpSesStatsEntry	このテーブルのエントリは、LDP エンティティと LDP ピア間の単一セッションに関する統計情報を表します。このテーブルは、mplsLdpPeerTable を拡張します。
mplsLdpSesStatsUnkMesTypeErrors	このオブジェクトは、このセッション中に検出された不明なメッセージ タイプ エラー数のカウントです。
mplsLdpSesStatsUnkTlvErrors	このオブジェクトは、このセッション中に検出された不明な TLV エラー数のカウントです。

MPLS LDP MIB バージョン 8 アップグレードにおける VPN コンテキスト

MPLS ボーダー ゲートウェイ プロトコル（BGP）4 バーチャル プライベート ネットワーク（VPN）環境では、各 VPN に個別の LDP プロセスを作成できます。これらのプロセスとその関連データを LDP コンテキストと呼びます。各コンテキストは、他のすべてのコンテキストとは独立しており、そのコンテキスト固有のデータだけが含まれています。

この機能によって、異なる MPLS VPN に対する異なるコンテキストへのサポートが追加されます。MIB のユーザは、特定の MPLS VPN の MPLS LDP プロセスを表示できます。VPN 対応 LDP MIB 機能によって、IETF MPLS-LDP MIB の構文が変わることはありません。テーブル内のエントリの数とタイプが変更されます。

IETF MPLS-LDP MIB は、同時に 1 つのコンテキストだけの情報を表示できます。SMNP セキュリティ名を使用して、グローバル コンテキストまたは MPLS VPN コンテキストを指定できます。

次の項では、VPN 対応 LDP MIB 機能に関連する内容について説明します。

SNMP コンテキスト

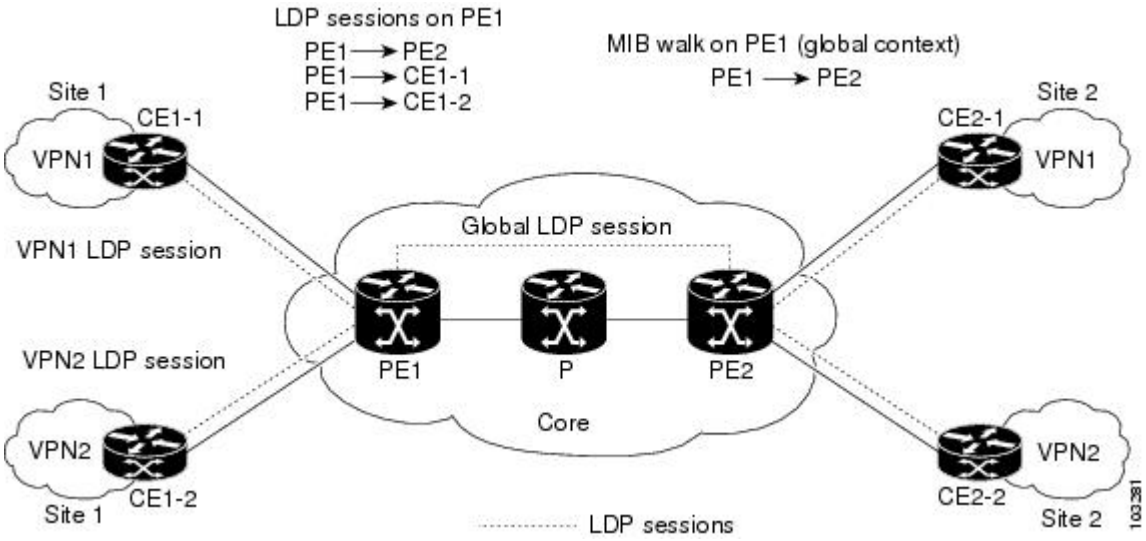
SNMP コンテキストによって、MIB データにアクセスする安全な方法が VPN ユーザに提供されます。VPN がコンテキストに関連付けられると、VPN 固有の MIB データがそのコンテキストに存在します。VPN をコンテキストに関連付けると、サービス プロバイダーが、複数 VPN でネットワークを管理できます。コンテキストを作成して VPN に関連付けることにより、サービス プロバイダーは、ある VPN のユーザが同じネットワーキング デバイス上で他の VPN のユーザに関する情報にアクセスするのを防ぐことができます。

VPN 対応 LDP MIB セッション

VPN 対応 LDP MIB 機能が実装される前は、MPLS LDP MIB に対する SNMP クエリーによって返されるのはグローバル セッションに関する情報だけでした。VPN コンテキストの LDP セッションに関する情報は返されませんでした。IETF MPLS LDP MIB はグローバル ルーティング テーブルから情報を取得しましたが、VPN ごとのルーティング データが格納された VPN ルーティングおよび転送（VRF）インスタンスから情報を取得しませんでした。MPLS LDP MIB はグローバルコンテキスト内の LDP プロセスだけを参照し、他のすべてのセッションを無視しました。VRF に対するクエリーによって情報は返されませんでした。VPN コンテキスト内の LDP プロセスは表示できます。

以下の図に、VPN 対応 LDP MIB 機能が実装される前の MPLS LDP セッションを含むサンプル MPLS VPN ネットワークを示します。

図 *17*：*VPN* 対応 *LDP MIB* 機能が実装される前の *MPLS LDP* セッションの設定



このソフトウェアよりも前の MIB ウォークでは、グローバルセッションの情報だけが表示されました。

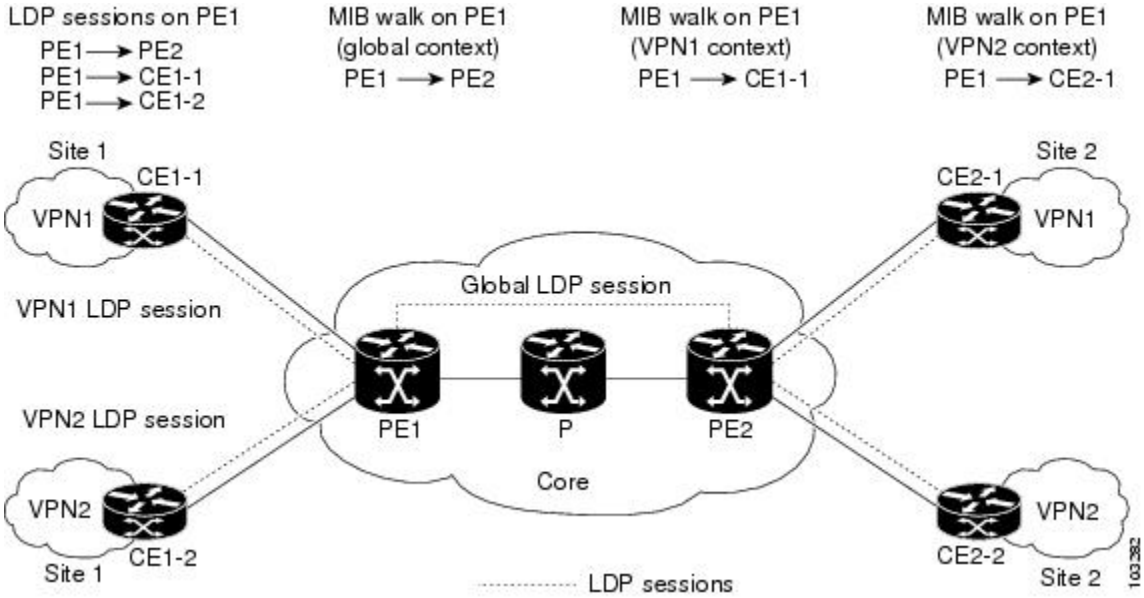
VPN 対応 LDP MIB 機能拡張により、IETF MPLS-LDP-MIB に対する SNMP クエリーでグローバルコンテキストと VPN コンテキストの両方がサポートされます。この機能を使用すると、VRF とコア（グローバル コンテキスト）に対する LDP クエリーを入力できます。クエリーは、異なる VPN からの LDP セッションを区別できます。VPN の LDP セッション情報は、その VPN のコンテキストに保存されます。したがって、1 つの VPN からの情報は、異なる VPN のユーザが使用することはできません。LDP MIB に対する VPN 対応アップデートによって、Carrier Supporting Carrier（CSC）ネットワークで動作している LDP プロセスを表示することもできます。

MPLS VPN では、サービス プロバイダー エッジ（PE）ルータに複数の VPN の VRF とグローバル ルーティング テーブルを含めることができます。同じデバイス上の VPN ごとに個別の LDP プロセスを設定するには、各 VPN に一意の securityName、contextName、および View-based Access

Control Model（VACM）ビューを設定する必要があります。VPN の securityName は、IETF MPLS LDP MIB に対して設定する必要があります。

以下の図に、VPN 対応 LDP MIB 機能を使用したサンプル MPLS VPN ネットワークの LDP セッションを示します。

図 *18*：*VPN* 対応 *LDP MIB* 機能を使用した *MPLS LDP* セッション



VPN 対応 LDP MIB 機能を使用すると、MPLS VPN LDP セッションまたはグローバル LDP セッションに対して MIB クエリーまたは MIB ウォークを実行できます。

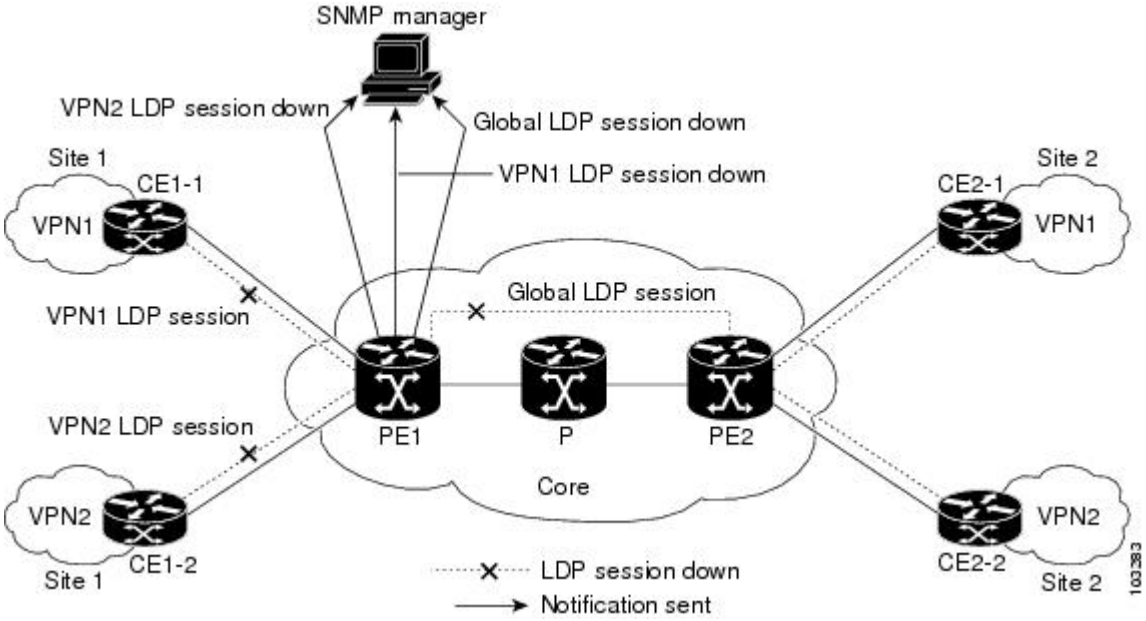
（注） 特定の VPN の LDP セッション情報を確認するには、**showmplsldpneighborvrf***vpn-name***detail** コマンドを使用します。

VPN 対応 LDP MIB の通知

VPN 対応 LDP MIB 機能が実装される前は、MPLS LDP セッションのすべての通知メッセージは、ネットワーク内の指定された同じネットワーク管理ステーション（NMS）に送信されました。通知をイネーブルにするには、**snmp-server enable traps mpls ldp** コマンドが使用されました。

以下の図に、VPN 対応 LDP MIB 機能が実装される前の LDP 通知の送信を示します。

図 *19*：*VPN* 対応 *LDP MIB* 機能が実装される前の *LDP* 通知の送信



VPN 対応 LDP MIB 機能では、VPN の複数の LDP コンテキストに対する LDP 通知がサポートされます。LDP 通知は、コア（グローバル コンテキスト）および異なる VPN に対して生成できます。LDP コンテキストごとに異なる NMS ホストに通知を送信できます。特定の VRF に関連付けられた LDP 通知は、その VRF に指定された NMS に送信されます。LDP グローバル通知は、グローバル トラップを受信するように設定された NMS に送信されます。

VPN 対応 LDP MIB 機能の LDP コンテキスト通知をイネーブルにするには、SNMP オブジェクト mplsLdpSessionsUpDownEnable（グローバル LDP コンテキストの場合だけ）または次の拡張グローバル コンフィギュレーション コマンドを使用します。

グローバル コンテキストの LDP 通知をイネーブルにするには、PE ルータで次のコマンドを使用します。

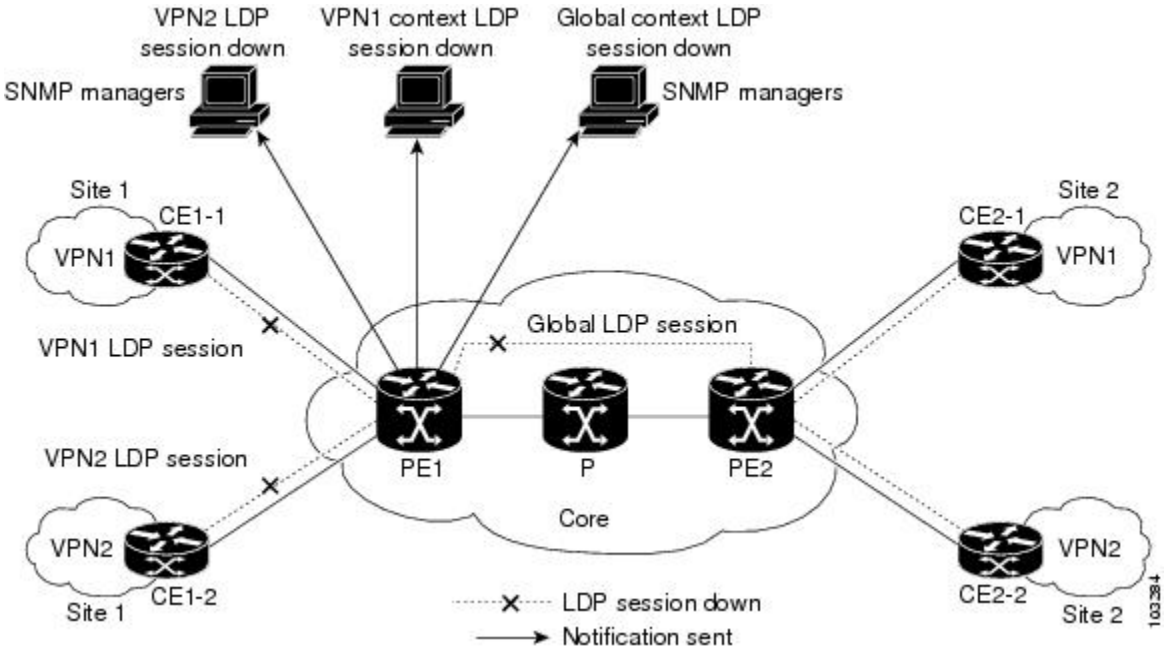
```
Router(config)# snmp-server host host-address traps community mpls-ldp
Router(config)# snmp-server enable traps mpls ldp
```

VPN コンテキストの LDP 通知をイネーブルにするには、PE ルータで次のコマンドを使用します。

```
Router(config)# snmp-server host host-address vrf vrf-name version {v1|v2c|v3}
community community-string udp-port upd-port mpls-ldp
Router(config)# snmp-server enable traps mpls ldp
```

以下の図に、VPN 対応 LDP MIB 機能を使用した LDP 通知を示します。

図 *20*：*VPN* 対応 *LDP MIB* 機能を使用した *LDP* 通知



MPLS LDP MIB バージョン 8 アップグレードの設定方法

SNMP エージェントのイネーブル化

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro***number*]
5. **end**
6. **write memory**
7. **show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 SNMP の情報が表示されない場合は、次のステップに進みます。 SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro***number*] 例： `Router(config)# snmp-server community public ro`	MPLS ラベル配布プロトコル（LDP）MIB に対して読み取り専用（ro）のコミュニティ ストリングを設定します。 • *string* 引数は、パスワードのように機能し、MPLS ネットワーク内のラベル スイッチング ルータ（LSR）上の SNMP 機能へのアクセスを許可します。 • オプションの **ro** キーワードでは、MPLS LDP MIB 内のオブジェクトへの読み取り専用（ro）アクセスを設定します。
ステップ 5	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**write memory** 例： `Router# write memory`	変更した SNMP 設定をルータの NVRAM に書き込み、SNMP 設定を永続的に保存します。
ステップ 7	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 snmp-server という文が表示される場合は、ルータで SNMP がイネーブルになっています。

	コマンドまたはアクション	目的
		SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。

分散型シスコ エクスプレス フォワーディングのイネーブル化

シスコ エクスプレス フォワーディングまたは分散型シスコ エクスプレス フォワーディングをイネーブルにするには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip cef distributed**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip cef distributed** 例： `Router(config)# ip cef distributed`	分散型シスコエクスプレスフォワーディングをイネーブルにします。
ステップ 4	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。

MPLS のグローバルなイネーブル化

MPLS をグローバルにイネーブルにするには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mpls ip**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mpls ip** 例： `Router(config)# mpls ip`	プラットフォーム用に通常ルーティングされるパスに沿って IPv4 パケットの MPLS 転送が行われるようにします。
ステップ 4	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。

LDP のグローバルなイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **mpls label protocol {ldp | tdp}**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mpls label protocol {ldp | tdp}** 例： `Router(config)# mpls label protocol ldp`	プラットフォームのデフォルトのラベル配布プロトコルを指定します。TDP は、すべてのソフトウェアリリースでサポートされなくなる可能性があります。
ステップ 4	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。

インターフェイス上の MPLS のイネーブル化

インターフェイス上の MPLS をイネーブルにするには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface***typeslot/subslot/port* [**.***subinterface-number*]
4. **mpls ip**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface***typeslot/subslot/port* [**.***subinterface-number*] 例： `Router(config)# interface FastEthernet 1/0/0`	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 ip** 例： `Router(config-if)# mpls ip`	特定のインターフェイス用に通常ルーティングされるパスに沿って IPv4 パケットの MPLS 転送が行われるようにします。
ステップ 5	**end** 例： `Router(config-if)# end`	特権 EXEC モードに戻ります。

インターフェイス上の LDP のイネーブル化

インターフェイス上の LDP をイネーブルにするには、次の作業を実行します。

手順の概要

1. **enable**
2. **configureterminal**
3. **interface***typeslot*/*subslot*/*port*[**.***subinterface-number*]
4. **mplslabelprotocolldp**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configureterminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface***typeslot*/*subslot*/*port*[**.***subinterface-number*] 例： `Router(config)# interface FastEthernet 1/0/0`	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	**mplslabelprotocolldp** 例： `Router(config-if)# mpls label protocol ldp`	指定したインターフェイスで使用するラベル配布プロトコルを指定します。
ステップ 5	**end** 例： `Router(config-if)# end`	特権 EXEC モードに戻ります。

VPN 対応 LDP MIB の設定

VPN に対する SNMP サポートの設定

バーチャル プライベート ネットワーク（VPN）またはリモート VPN の SNMP サポートを設定するには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server host***host-address* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [*notification-type*] [**vrf***vrf-name*]
4. **snmp-server engineID remote***ip-address* [**udp-port***udp-port-number*] [**vrf***vrf-name*] *engineid-string*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server host***host-address* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [*notification-type*] [**vrf***vrf-name*] 例： `Router(config)# snmp-server host example.com vrf trap-vrf`	SNMP 通知動作の受信者を指定し、SNMP 通知の送信に使用するバーチャル プライベート ネットワーク（VPN）ルーティング/転送（VRF）インスタンス テーブルを指定します。
ステップ 4	**snmp-server engineID remote***ip-address* [**udp-port***udp-port-number*] [**vrf***vrf-name*] *engineid-string*	ルータ上のリモート SNMP エンジンの名前を設定します。

	コマンドまたはアクション	目的
	例： `Router(config)# snmp-server engineID remote 172.16.20.3 vrf traps-vrf 80000009030000B064EFE100`	
ステップ 5	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。

VPN の SNMP コンテキストの設定

VPN の SNMP コンテキストを設定するには、次の作業を実行します。これにより、VPN の一意の SNMP コンテキストを設定して、VPN の LDP セッション情報にアクセスできるようになります。

SNMP コンテキスト

SNMP コンテキストによって、MIB データにアクセスする安全な方法が VPN ユーザに提供されます。VPN がコンテキストに関連付けられると、VPN 固有の MIB データがそのコンテキストに存在します。VPN をコンテキストに関連付けると、サービス プロバイダーが、複数 VPN でネットワークを管理できます。コンテキストを作成して VPN に関連付けることにより、サービス プロバイダーは、ある VPN のユーザが同じネットワーキング デバイス上で他の VPN のユーザに関する情報にアクセスするのを防ぐことができます。

VPN ルート識別子

ルート識別子（RD）によって、VPN のルーティングおよび転送テーブルが作成されます。Cisco ソフトウェアは、RD をカスタマーの IPv4 プレフィックスの先頭に追加して、それらのプレフィックスをグローバルに一意である VPN-IPv4 プレフィックスに変更します。

RD は、自律システム番号と任意の番号で構成される自律システム番号（ASN）相対 RD、または IP アドレスと任意の番号で構成される IP アドレス相対 RD のいずれかです。RD は、次のいずれかの形式で入力できます。

- 16 ビット ASN：101:3 などの 32 ビット数値
- 32 ビット IP アドレス：192.168.122.15:1 などの 16 ビット数値

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server context***context-name*
4. **ip vrf***vrf-name*
5. **rd***route-distinguisher*
6. **context***context-name*
7. **route-target** [**import** | **export** | **both**] *route-target-ext-community*
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server context***context-name* 例： `Router(config)# snmp-server context context1`	SNMP コンテキストを作成し、その名前を指定します。
ステップ 4	**ip vrf***vrf-name* 例： `Router(config)# ip vrf vrf1`	バーチャルプライベートネットワーク（VPN）ルーティングおよび転送（VRF）テーブルを設定し、VRF コンフィギュレーションモードを開始します。
ステップ 5	**rd***route-distinguisher* 例： `Router(config-vrf)# rd 100:120`	VPN ルート識別子を作成します。

	コマンドまたはアクション	目的
ステップ 6	**context***context-name* 例： `Router(config-vrf)# context context1`	SNMP コンテキストを特定の VRF に関連付けます。
ステップ 7	**route-target** [**import** | **export** | **both**] *route-target-ext-community* 例： `Router(config-vrf)# route-target export 100:1000`	（任意）VRF 用の route-target 拡張コミュニティを作成します。
ステップ 8	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。

SNMPv1 または SNMPv2 への SNMP VPN コンテキストの関連付け

SNMP VPN コンテキストを SNMPv1 または SNMPv2 に関連付けるには、次の作業を実行します。これにより、SNMPv1 または SNMPv2 を使用して VPN の LDP セッション情報にアクセスできます。

SNMPv1 または SNMPv2 のセキュリティ：SNMPv1 と SNMPv2 は、SNMPv3 ほど安全ではありません。SNMP バージョン 1 と 2 では、プレーン テキスト コミュニティを使用し、SNMP バージョン 3 で実行される認証やセキュリティ チェックを実行しません。

SNMP バージョン 1 または SNMP バージョン 2 を使用するときに VPN 対応 LDP MIB 機能を設定するには、コミュニティ名を VPN に関連付ける必要があります。関連付けると、SNMP は、特定のコミュニティ ストリングの着信要求を、設定されている VRF から受信した場合だけ処理します。着信パケットに含まれているコミュニティ ストリングに VRF が関連付けられていない場合は、VRF 以外のインターフェイス経由で着信した場合だけパケットを処理します。このプロセスによって、VPN 外のユーザがクリア テキスト コミュニティ ストリングを使用して VPN データを問い合わせるのを防ぐことができます。ただし、これは SNMPv3 を使用する場合ほど安全ではありません。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server user***usernamegroup-name* [**remote***host* [**udp-port***port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**}*auth-password*]} [**access***access-list*]
4. **snmp-servergroup***group-name* {**v1** | **v2c** | **v3**{**auth** | **noauth** | **priv**}} [**context***context-name*] [**read***readview*] [**write***writeview*] [**notify***notifyview*] [**access***access-list*]
5. **snmp-server view***view-nameoid-tree* {**included** | **excluded**}
6. **snmp-server enable traps** [*notification-type*]
7. **snmp-serverhost***host-address* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [**notification-type**] [**vrf***vrf-name*]
8. **snmpmibcommunity-map***community-name* [**context***context-name*] [**engineid***engine-id*] [**security-name***security-name*] **target-list***vpn-list-name*
9. **snmpmibtargetlist***vpn-list-name* {**vrf***vrf-name* | **host***ip-address*}
10. **no snmp-server trap authentication vrf**
11. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server user***usernamegroup-name* [**remote***host* [**udp-port***port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**}*auth-password*]} [**access***access-list*] 例： `Router(config)# snmp-server user customer1 group1 v1`	SNMP グループに新しいユーザを設定します。

	コマンドまたはアクション	目的
ステップ 4	**snmp-servergroup***group-name* {**v1** | **v2c** | **v3**{**auth** | **noauth** | **priv**}} [**context***context-name*] [**read***readview*] [**write***writeview*] [**notify***notifyview*] [**access***access-list*] 例： `Router(config)# snmp-server group group1 v1 context context1 read view1 write view1 notify view1`	新規 SNMP グループ、または SNMP ユーザを SNMP ビューにマッピングするテーブルを設定します。 • **context***context-name* キーワードと引数を使用して、指定した SNMP グループを設定されている SNMP コンテキストに関連付けます。
ステップ 5	**snmp-server view***view-nameoid-tree* {**included** | **excluded**} 例： `Router(config)# snmp-server view view1 ipForward included`	ビュー エントリを作成または更新します。
ステップ 6	**snmp-server enable traps** [*notification-type*] 例： `Router(config)# snmp-server enable traps`	システムで使用できるすべての SNMP 通知（トラップまたは応答要求）をイネーブルにします。
ステップ 7	**snmp-serverhost***host-address* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [**notification-type**] [**vrf***vrf-name*] 例： `Router(config)# snmp-server host 10.0.0.1 vrf customer1 public udp-port 7002`	SNMP 通知操作の受信者を指定します。
ステップ 8	**snmpmibcommunity-map***community-name* [**context***context-name*] [**engineid***engine-id*] [**security-name***security-name*] **target-list***vpn-list-name* 例： `Router(config)# snmp mib community-maps community1 context context1 target-list commAVpn`	SNMP コミュニティを SNMP コンテキスト、エンジン ID、またはセキュリティ名にマッピングします。
ステップ 9	**snmpmibtargetlist***vpn-list-name* {**vrf***vrf-name* | **host***ip-address*} 例： `Router(config)# snmp mib target list commAVpn vrf vrf1`	SNMP コミュニティに関連付けるターゲット VRF とホストのリストを作成します。

	コマンドまたはアクション	目的
ステップ 10	**no snmp-server trap authentication vrf** 例： `Router(config)# no snmp-server trap authentication vrf`	（任意）VRF インターフェイスで受信したパケットについて生成されるすべての SNMP 認証通知（トラップまたは応答要求）をディセーブルにします。 • このコマンドを使用して、関連付けられているコミュニティが正しくない VRF インターフェイス上のパケットに対する認証トラップだけをディセーブルにします。
ステップ 11	**exit** 例： `Router(config) exit`	特権 EXEC モードに戻ります。

MPLS LDP MIB バージョン 8 アップグレードの確認

MPLS LDP MIB バージョン 8 アップグレードが機能していることを確認するには、SNMP 管理ツールを使用して MIB ウォークを実行します。

MPLS LDP MIB バージョン 8 アップグレードの設定例

MPLS LDP MIB バージョン 8 アップグレードの例

次に、ホスト NMS 上で SNMP エージェントをイネーブルにする例を示します。

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# snmp-server community
```

次に、ホスト NMS 上の SNMPv1 と SNMPv2C をイネーブルにする例を示します。設定では、コミュニティ ストリング public を使用して、SNMP エージェントが読み取り専用アクセス権を持つすべての MPLS LDP MIB オブジェクトにアクセスすることを許可しています。

```
Router(config)# snmp-server community public
```

次に、comaccess コミュニティ ストリングを指定するアクセス リスト 4 のメンバに、すべての MPLS LDP MIB オブジェクトへの読み取り専用アクセスを許可する例を示します。その他の SNMP エージェントは MPLS LDP MIB オブジェクトにアクセスできません。

```
Router(config)# snmp-server community comaccess ro 4
```

次に、LDP をグローバルにイネーブルにしてからインターフェイスに対してイネーブルにする例を示します。

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# mpls label protocol ldp
Router(config)# interface FastEthernet1/0/0
Router(config-if)# mpls label protocol ldp
Router(config-if)# end
```

SNMPv1 または SNMPv2 の VPN 対応 SNMP コンテキストの設定：例

次に、SNMPv1 または SNMPv2 を使用して MPLS LDP MIB バージョン 8 の VPN 対応 SNMP コンテキストを設定する例を示します。

```
snmp-server context A
snmp-server context B
ip vrf CustomerA
 rd 100:110
 context A
 route-target export 100:1000
 route-target import 100:1000
!
ip vrf CustomerB
 rd 100:120
 context B
 route-target export 100:2000
 route-target import 100:2000
!
interface FastEthernet0/3/1
 description Belongs to VPN A
 ip vrf forwarding CustomerA
 ip address 10.0.0.0 255.255.0.0

interface FastEthernet0/3/2
 description Belongs to VPN B
 ip vrf forwarding CustomerB
 ip address 10.0.0.1 255.255.0.0
snmp-server user commA grp1A v1
snmp-server user commA grp2A v2c
snmp-server user commB grp1B v1
snmp-server user commB grp2B v2c
snmp-server group grp1A v1 context A read viewA write viewA notify viewA
snmp-server group grp1B v1 context B read viewB write viewB notify viewB
snmp-server view viewA ipForward included
snmp-server view viewA ciscoPingMIB included
snmp-server view viewB ipForward included
snmp-server view viewB ciscoPingMIB included
snmp-server enable traps
snmp-server host 10.0.0.3 vrf CustomerA commA udp-port 7002
snmp-server host 10.0.0.4 vrf CustomerB commB udp-port 7002
snmp mib community-map  commA context A target-list commAvpn
! Configures source address validation
snmp mib community-map  commB context B target-list commBvpn
! Configures source address validation
snmp mib target list commAvpn vrf CustomerA
! Configures a list of VRFs or from which community commA is valid
```

```
snmp mib target list commBvpn vrf CustomerB
! Configures a list of VRFs or from which community commB is valid
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
MPLS LDP の設定作業	MPLS ラベル配布プロトコル（LDP）
MPLS トラフィック エンジニアリング MIB（MPLS TE MIB）に対する SNMP エージェント サポートの説明	MPLS トラフィック エンジニアリング（TE）MIB
MPLS ネットワークでの MPLS で差別化されたサービス タイプの説明	MPLS QoS
SNMP コマンド	『*Network Management Command Reference*』
SNMP コンフィギュレーション VPN のための SNMP サポート	『Configuring SNMP Support』

標準

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

MIB

MIB	MIB のリンク
• MPLS ラベル配布プロトコル MIB（draft-ietf-mpls-ldp-mib-08.txt） • SNMP-VACM-MIB、SNMP 用 View-based Access Control Model（ACM）MIB	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャ セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
RFC 2233 MPLS LDP MIB をサポートする LDP 実装は、RFC 2026 第 10 項の規定に完全に準拠しています。その規定では、宛先ベースのルーティングプロトコルで決定された、通常ルーティングされるパスに沿った MPLS 転送を実行するネットワーク デバイスに LDP の実装を推奨しています。	『*Interfaces MIB*』

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 LDP MIB バージョン 8 アップグレードの機能情報

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

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

表 *38* : ***MPLS LDP MIB*** バージョン ***8*** アップグレードの機能情報

機能名	リリース	機能情報
MPLS LDP MIB バージョン 8 アップグレード	12.0(11)ST 12.2(2)T 12.0(21)ST 12.0(22)S 12.0(24)S 12.2(18)S 12.2(33)SRB 12.2(33)SB Cisco IOS XE Release 2.1	

機能名	リリース	機能情報
		MPLS ラベル配布プロトコル（LDP）MIB バージョン 8 アップグレード機能により、Internet Engineering Task Force（IETF）ドラフトバージョン 8 をサポートするよう LDP MIB が拡張されます。 Cisco IOS Release 12.0(11)ST で、Cisco 7200、Cisco 7500、および Cisco 12000 シリーズ ルータで MPLS LDP MIB の SNMP エージェントをサポートするために、この機能が導入されました。 Cisco IOS Release 12.2(2)T で、Cisco 7200 および Cisco 7500 シリーズ ルータで MPLS LDP MIB の SNMP エージェントをサポートするために、この機能が追加されました。 Cisco IOS Release 12.0(21)T で、Cisco 7200、Cisco 7500、および Cisco 12000 シリーズ インターネット ルータで MPLS LDP MIB の SNMP エージェントおよび LDP 通知をサポートするために、この機能が追加されました。 Release 12.0(22)S で、バージョン 1 が Cisco IOS Release 12.0(22)S に統合されました。 Cisco IOS Release 12.0(24)S で、この機能がバージョン 8 にアップグレードされました。 この機能は、Cisco IOS Release 12.2(18)S に統合されました。 Cisco IOS Release 12.2(33)SRB で、この MIB が廃止され、MPLS-LDP-STD-MIB（RVC 3815）に置き換えられました。

機能名	リリース	機能情報
		Cisco IOS Release 12.2(33)SB で、この MIB が廃止され、MPLS-LDP-STD-MIB（RVC 3815）に置き換えられました。 この機能が Cisco IOS XE Release 2.1 に統合され、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに実装されました。
		次のコマンドが導入または変更されました。**context**、**show mpls ldp neighbor**、**snmp mib community-map**、**snmp mib target list**、**snmp-server community**、**snmp-server context**、**snmp-server enable traps**（MPLS）、**snmp-server group**、**snmp-server host**、**snmp-server trap authentication vrf**。
MPLS VPN-VPN 対応 LDP MIB	12.0(27)S 12.2(28)SB 12.2(33)SRA 12.2(33)SXH 12.4(20)T	MPLS VPN-VPN 対応 LDP MIB では、任意の VRF とコア（グローバル コンテキスト）に対する LDP クエリを入力できます。 Cisco IOS Release 12.0(27)S で、MPLS VPN-VPN 対応 LDP MIB 機能のサポートが追加されました。 この機能は、Cisco IOS Release 12.2(28)SB で導入されました。 この機能は、Cisco IOS Release 12.2(33)SRA で統合されました。 Cisco IOS Release 12.2(33)SXH では、この機能が統合されました。 この機能は、Cisco IOS Release 12.4(20)T で統合されました。

用語集

ATM：Asynchronous Transfer Mode（非同期転送モード）。セルリレーの国際規格です。複数のサービスタイプ（音声、ビデオ、データなど）が固定長（53バイト）のセルで転送されます。固定長セルの場合は、ハードウェアでセルを処理できるため、伝送遅延が短縮されます。ATM は、E3、SONET、T3 などの高速送信メディアを利用するように設計されています。

ダウンストリームオンデマンド配布：ラベル配布方式。ダウンストリーム ラベル スイッチ ルータ（LSR）は、アップストリーム LSR によって要求された場合だけバインディングをアップストリームに送信します。

ダウンストリーム未承諾配布：ラベル配布方式。ダウンストリーム ラベル スイッチルータ（LSR）がネイバーアップストリーム LSR との新しいバインディングを確立する必要がある場合に、ラベルが分散されます。たとえば、エッジ LSR は、別のサブネットとの新しいインターフェイスをイネーブルにする場合があります。その場合、LSR は、このネットワークに到達するためのバインディングをアップストリーム ルータにアナウンスします。

応答要求：従来のトラップ通知メッセージよりも信頼性が高い通知メッセージのタイプ。信頼性が高いのは、応答要求メッセージ通知には確認応答が必要ですが、トラップ通知には必要ないためです。

ラベル：スイッチング ノードに対してデータの転送方法（パケットまたはセル）を指示する短い固定長のデータ ID。

ラベル配布：通常ルーティングされるパスに沿ったホップバイホップ転送をサポートするために、ラベルスイッチルータ（LSR）によって、ラベルバインディング情報の交換に使用される技術およびプロセス。

LDP：Label Distribution Protocol（ラベル配布プロトコル）。マルチプロトコル ラベル スイッチング（MPLS）ホップバイホップ転送およびラベルとネットワーク プレフィックス間のバインディングの配布をサポートするプロトコル。

LSP：Label Switched Path（ラベル スイッチド パス）。ラベル スイッチング技術がパケット転送に使用される、2 つのラベル スイッチング ルータ（LSR）間の設定済み接続。このパスは、マルチプロトコル ラベル スイッチング（MPLS）ネットワークを介した特定のパスでもあります。

LSR：Label Switch Router（ラベル スイッチ ルータ）。ネイティブなレイヤ 3 パケットを転送できるマルチプロトコル ラベル スイッチング（MPLS）ノード。LSR は、パケットに付加されたラベルの値に基づいてパケットを転送します。

MIB：Management Information Base（管理情報ベース）。簡易ネットワーク管理プロトコル（SNMP）などの、ネットワーク管理プロトコルが使用および維持するネットワーク管理情報のデータベース。MIB オブジェクトは、SNMP コマンドを使用して、通常はネットワーク管理システムを通じて変更または取得できます。MIB オブジェクトはツリー構造であり、ツリーにはパブリック（標準）ブランチとプライベート（独自）ブランチを含みます。

MPLS：Multiprotocol Label Switching（マルチプロトコル ラベル スイッチング）。ラベルを使用して IP トラフィックを転送するスイッチング方式。このラベルによって、ネットワーク内のルータ

およびスイッチが、事前に確立された IP ルーティング情報に基づくパケットの転送先を指示されます。

MPLS ラベル配布：ラベル スイッチド パス（LSP）トンネルのルーティングを行うための、コンストレイントベース ルーティング アルゴリズム。

NMS：Network Management Sstation（ネットワーク管理ステーション）。ネットワーク管理者がネットワーク上の他のデバイスと通信するために使用する、高性能なコンピュータ（通常は、エンジニアリング ワークステーション）。NMS は、通常、ネットワーク リソースの管理、統計情報の収集、およびさまざまなネットワーク管理および設定タスクの実行に使用されます。Simple Network Management Protocol（SNMP）のコンテキストでは、NMS は、情報を取得または修正するために管理対象デバイスの SNMP エージェントに対する SNMP クエリを実行するデバイスです。

通知：シンプル ネットワーク管理プロトコル（SNMP）エージェントがネットワーク管理ステーション、コンソール、または端末に送信する、重要なネットワーク イベントが発生したことを示すメッセージ。「トラップ」も参照してください。

RSVP：Resource Reservation Protocol。IP ネットワーク上でリソースの予約をサポートするためのプロトコル。IP エンド システムで実行されているアプリケーションは、RSVP を使用して、帯域幅、ジッタ、最大バーストなどの項目を指定することにより、受信するパケット ストリームの特性を他のノードに示すことができます。

RTR：Response Time Reporter。応答時間とアベイラビリティを測定することによってネットワーク パフォーマンス、ネットワーク リソース、およびアプリケーションを監視できるツール。

SNMP：Simple Network Management Protocol（シンプル ネットワーク管理プロトコル）。TCP/IP ネットワークで、ほとんど排他的に使用されているネットワーク管理プロトコル。SNMP を使用して、ユーザは、ネットワーク デバイスの監視と制御、設定の管理、統計情報の収集、パフォーマンスの監視、およびネットワーク セキュリティの確認ができます。

SNMP コミュニティ：インテリジェントなネットワーク デバイスによる SNMP 要求の検証を可能にする認証方式。

SNMPv2c：Simple Network Management Protocol バージョン 2c。SNMPv2c は、集中型と分散型の両方のネットワーク管理戦略をサポートし、Structure of Management Information（SMI）、プロトコル操作、管理アーキテクチャ、およびセキュリティが改善されています。

SNMPv3：Simple Network Management Protocol のバージョン 3。相互運用可能な標準ベースのネットワーク管理プロトコルです。SNMPv3 は、ネットワーク経由のパケットの認証と暗号化を組み合わせることによって、デバイスへのセキュア アクセスを実現します。

TLV：Type Length Value（タイプ、長さ、値）。さまざまな属性を伝送するために、いくつかのルーティング プロトコルで使用されるメカニズム。TLV を使用するプロトコルの例として、Cisco Discovery Protocol（CDP）、Label Discovery Protocol（LDP）、ボーダー ゲートウェイ プロトコル（BGP）などがあります。BGP は、TLV を使用して、ネットワーク層到達可能性情報（NLRI）、Multiple Exit Discriminator（MED）、ローカル プリファレンスなどの属性を伝送します。

トラップ：Simple Network Management Protocol（SNMP）エージェントがネットワーク管理ステーション、コンソール、または端末に送信する、重要なネットワーク イベントが発生したことを示すメッセージ。トラップ（通知）は応答要求よりも信頼性が低くなります。トラップの受信者が

受信の確認応答を送信しないので、トラップが受信されたかどうかをトラップの送信者が判断できないためです。「通知」も参照してください。

VCC：Virtual Channel Connection（仮想チャネル接続）。ATM ネットワーク内の 2 つのエンドポイント間でデータを伝送する、仮想チャネル リンク（VCL）で構成された論理回線。仮想回線接続と呼ばれることもあります。

VCI：Virtual Channel Identifier（仮想チャネル識別子）。ATM セルのヘッダーにある 16 ビットのフィールド。VCI は、仮想パス識別子（VPI）とともに、セルがその最終的な宛先に到達するまで一連の ATM スイッチを通過するときの次のネットワーク仮想チャネル リンク（VCL）を識別するために使用されます。

VCL：Virtual Channel Link（仮想チャネル リンク）。ATM ネットワーク内の 2 つの隣接スイッチ間に存在する論理接続。

VPI：Virtual Path Identifier（仮想パス識別子）。ATM セルのヘッダーにある 8 ビットのフィールド。VPI は、仮想チャネル識別子（VCI）とともに、セルがその最終的な宛先に到達するまで一連の ATM スイッチを通過するときの次のネットワーク仮想チャネル リンク（VCL）を識別するために使用されます。

VPN：Virtual Private Network（バーチャル プライベート ネットワーク）。トンネリングを使用することにより、パブリック TCP/IP ネットワーク経由でも IP トラフィックをセキュアに転送できるネットワーク。

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

第 8 章

MPLS VPN--MIB サポート

このマニュアルでは、ドラフトの *SMIv2* を使用した *MPLS/BGP* バーチャル プライベート ネットワーク管理情報ベース（*draft-ietf-ppvpn-mpls-vpn-mib-05.txt*）で実装されたマルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）管理のための Cisco ソフトウェアにおける簡易ネットワーク管理プロトコル（SNMP）エージェント サポートについて説明します。独自の MIB CISCO-IETF-PPVNP-MPLS-VPN-MIB の一部として実装された cMplsNumVrfRouteMaxThreshCleared 通知についても説明します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS VPN-MIB サポートの前提条件

- SNMP が、ラベル スイッチング ルータにインストールされてイネーブルになっている。
- MPLS が、ラベル スイッチング ルータでイネーブルになっている。
- マルチプロトコル ボーダー ゲートウェイ プロトコル（MP-BGP）が、ラベル スイッチング ルータでイネーブルになっている。
- シスコ エクスプレス フォワーディングが、ラベル スイッチング ルータでイネーブルになっている。

MPLS VPN-MIB サポートの制約事項

- mplsVpnNotificationEnable や mplsVpnVrfSecIllegalLabelRcvThresh などのトラップ関連のオブジェクトを除き、**snmpset** コマンドを使用した MIB の設定はサポートされません。
- mplsVpnVrfBgpNbrPrefixTable はサポートされません。

MPLS VPN--MIB サポートに関する情報

MPLS VPN の概要

MPLS VPN テクノロジーにより、サービス プロバイダーは、顧客のリモート オフィスからパブリック ネットワークに、プライベート ネットワークで提供されるのと同じセキュリティおよびサービス レベルで直接接続する、イントラネットおよびエクストラネット VPN サービスを提供することができます。各 VPN は、1 つ以上の VPN ルーティングおよび転送（VRF）インスタンスに関連付けられています。ルータに定義されている VPN ごとに VRF が作成されます。VRF には、MPLS VPN を管理および監視するのに必要な情報のほとんどが含まれています。具体的には、IP ルーティング テーブル、取得されたシスコ エクスプレス フォワーディング テーブル、転送テーブルを使用する一連のインターフェイス、およびルーティング テーブルに格納されている情報を制御するための一連のルールおよびルーティング プロトコル パラメータです。

MPLS VPN MIB の概要

プロバイダー プロビジョニング VPN（PPVPN）-MPLS-VPN MIB を使用すると、MPLS VRF 情報以外に、VRF に含まれているインターフェイス、および他の設定とモニタリング情報にアクセスできます。

PPVPN-MPLS-VPN MIB には次の利点があります。

- 標準ベースの SNMP インターフェイスにより、重要な MPLS VPN イベントに関する情報を取得できる。
- MPLS VPN の管理および監視に VRF 情報を利用できる。
- インターフェイス MIB とともに、VRF に割り当てられたインターフェイスに関する情報も利用できる。
- ルータのすべての VRF に関するパフォーマンス統計情報を利用できる。
- MPLS VPN 対応インターフェイスの動作ステータスが大きく変更された場合、注意を喚起する通知を生成し、キューに入れることができる。ネットワーク管理者が評価し、対策を講じることができるように、指定のネットワーク管理システム（NMS）に通知メッセージを転送できる。
- VPN ルーティング テーブルが容量の限度に近づくか、または超えたとき、詳細な警告を通知できる。
- VRF 対応インターフェイスで不正なラベルを受信した場合に、警告を通知できる。このようなラベルを受信した場合は、設定に誤りがあるか、またはセキュリティ違反が試みられた可能性があります。

また、このマニュアルでは、cMplsNumVrfRouteMaxThreshCleared 通知が含まれてている CISCO-IETF-PPVPN-MPLS-VPN-MIB についても説明します。

MPLS VPN MIB および IETF

PPVPN-MPLS-VPN MIB では SNMP エージェント コードが動作するため、標準化された SNMP ベースの方法を使用して、Cisco ソフトウェアで MPLS VPN を管理できます。

PPVPN-MPLS-VPN MIB は、Internet Engineering Task Force ドラフト MIB 仕様 *draft-ietf-ppvpn-mpls-vpn-mib-05.txt* に基づいています。この仕様には、MPLS VPN イベントをサポートする機能を説明したオブジェクトが規定されています。この IETF ドラフト MIB は随時改訂され、今後、標準規格となります。PPVPN-MPLS-VPN MIB のシスコ実装は、IETF ドラフト MIB の発展に追随し、それに伴い変更される可能性があります。

IETF ドラフト MIB と Cisco ソフトウェアに実装している MPLS VPN にはわずかに異なる部分があるため、PPVPN-MPLS-VPN MIB と Cisco ソフトウェアの内部データ構造との間でいくつかの軽微な変換が必要となります。このような変換は、SNMP エージェント コードによって実行されます。また、SNMP エージェントは優先度が低いプロセスとして動作し、Cisco ソフトウェアへの管

理インターフェイスとなります。SNMP が動作しても、デバイスの通常の機能にはほとんどオーバーヘッドがかかりません。

PPVPN-MPLS-VPN MIB に定義された SNMP オブジェクトは、標準の SNMP ユーティリティで表示できます。ネットワーク管理者は、SNMP v1、v2、および v3 の標準の SNMP get 操作および getnext 操作を使用して、PPVPN-MPLS-VPN MIB 内の情報を取得できます。

すべての PPVPN-MPLS-VPN MIB オブジェクトが、IETF ドラフト MIB に基づいています。このため、Cisco 固有の SNMP アプリケーションを使用することなく、PPVPN-MPLS-VPN MIB に関する機能および操作をサポートできます。

PPVPN-MPLS-VPN MIB でサポートされている機能

PPVPN-MPLS-VPN MIB では、次の操作を実行できます。

- ルータで MPLS VPN のルーティング情報および転送情報を収集する。
- VRF ルーティング テーブル内の情報を公開する。
- VPN および VRF インターフェイスに関する BGP 設定の情報と統計情報を収集する。
- 重要な MPLS VPN イベントが発生したときに、変更を伝える通知メッセージを発行する。
- 既存の SNMP コマンドライン インターフェイス（CLI）コマンドの拡張機能を使用して、MPLS VPN イベントに関する通知メッセージをイネーブル、ディセーブル、および設定する。
- 通知メッセージの送信先となる稼働環境の NMS の IP アドレスを指定する。
- 通知設定を不揮発性メモリに書き込む。

PPVPN-MPLS-VPN MIB の機能構造

PPVPN-MPLS-VPN MIB をサポートする SNMP エージェント コードは、Cisco ソフトウェア内のこのようなコードの既存モデルに準じます。また、その一部は、MIB ソース コードに基づいて Cisco ソフトウェア ツール セットにより生成されます。

SNMP エージェント コードは、Cisco ソフトウェアの MIB サポート コードに共通の階層構造となっており、次の 4 つのレイヤで構成されています。

- プラットフォームに依存しないレイヤ：このレイヤは、主に MIB 開発 Cisco ソフトウェア ツールセットによって生成され、プラットフォームや実装に依存しない機能を統合します。この Cisco MIB 開発ツール セットにより、MIB に関連付けられる標準のファイル セットが作成されます。
- アプリケーション インターフェイス レイヤ：このレイヤに属する MIB オブジェクトの機能、名前、およびテンプレート コードも、MIB 開発 Cisco ソフトウェア ツール セットによって生成されます。

- アプリケーション固有のレイヤ：このレイヤは、アプリケーション インターフェイス レイヤと次に説明する API/データ構造レイヤとをつなぐインターフェイスであり、Cisco ソフトウェアから必須情報を取得するのに必要な作業（データ構造内の検索など）を実行します。
- API/データ構造レイヤ：このレイヤには、SNMP 管理情報を設定または取得するために取得または呼び出される Cisco ソフトウェア内のデータ構造または API が含まれています。

PPVPN-MPLS-VPN MIB でサポートされているオブジェクト

PPVPN-MPLS-VPN MIB には、Cisco IOS ソフトウェアで読み取り専用の SNMP を使用して MPLS VPN 機能を管理するためのテーブルとオブジェクト定義が数多く含まれています。PPVPN-MPLS-VPN MIB は、抽象構文記法 1（ASN.1）に準拠し、これにより、理想的な MPLS VPN データベースが反映されています。

標準の SNMP ネットワーク管理アプリケーションを使用すると、GET 操作で PPVPN-MPLS-VPN MIB から情報を取得して表示できます。また、GETNEXT 操作で MIB データベース内の情報を走査して表示することもできます。

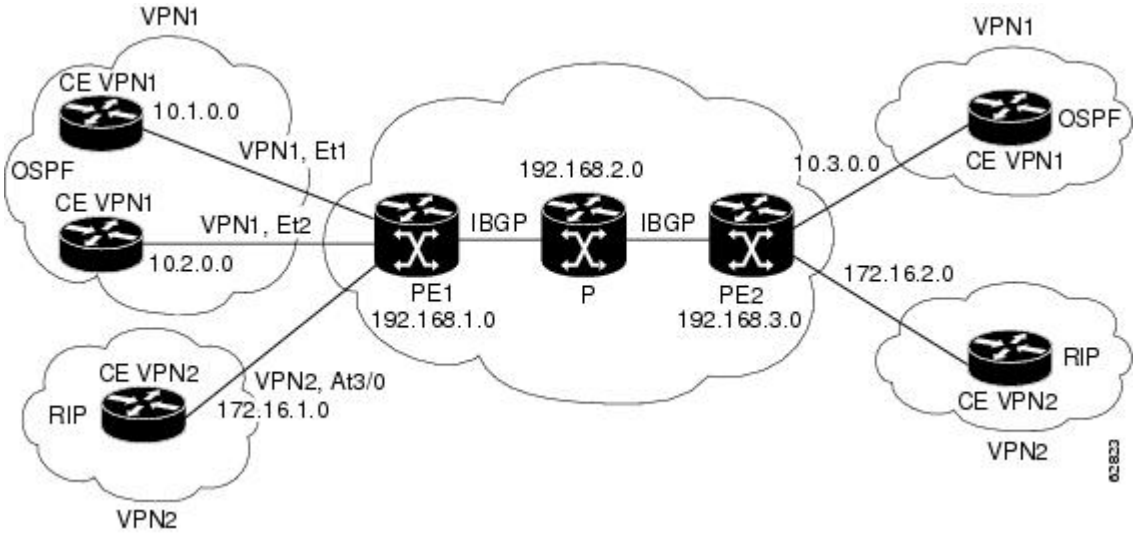
PPVPN-MPLS-VPN MIB テーブルおよびオブジェクトについては、以降の項で簡単に説明します。

以下の図に、単純な MPLS VPN の設定を示します。この設定には、VPN1 と VPN2 のラベルが付いたカスタマー MPLS VPN が 2 つ含まれています。また、PE1 と PE2 のラベルが付いたプロバイダー エッジ（PE）ルータ 2 台と、P のラベルが付いたプロバイダー コア ルータ 1 台で構成されている簡単なプロバイダー ネットワークも含まれています。以下の図に示されている設定例は次のとおりです。

- VRF 名：VPN1 および VPN2
- VRF に関連付けられたインターフェイス：Et1、Et2、および At3/0
- ルーティング プロトコル：Open Shortest Path First リンクステート（OSPF）、Routing Information Protocol（RIP）、および内部ボーダー ゲートウェイ プロトコル（IBGP）
- VPN1 に関連付けられたルート：10.1.0.0、10.2.0.0、および 10.3.0.0
- VPN2 に関連付けられたルート：172.16.1.0 および 172.16.2.0
- プロバイダー ネットワークに関連付けられたルート：192.168.1.0、192.168.2.0、および 192.168.3.0

このマニュアルでは、この設定に基づいて、PPVPN-MPLS-VPN MIB が監視および管理する MPLS VPN イベントについて説明しています。

図 *21*：*MPLS VPN* の設定例



スカラー オブジェクト

以下の表に、サポートされている PPVPN-MPLS-VPN MIB スカラー オブジェクトを示します。

表 *39*：*PPVPN-MPLS-VPN MIB* スカラー オブジェクト

MIB オブジェクト	機能
mplsVpnConfiguredVrfs	ルータに設定された VRF の数。最近削除されたものも含まれています。
mplsVpnActiveVrfs	ルータでアクティブな VRF の数。アップ状態である 1 つ以上のインターフェイスにアクティブな VRF が割り当てられます。
mplsVpnConnectedInterfaces	任意の VRF に割り当てられたインターフェイスの総数。

MIB オブジェクト	機能
mplsVpnNotificationEnable	すべての PPVPN-MPLS-VPN MIB 通知がイネーブルになっているかどうかを示す値。 • このオブジェクトを true に設定すると、PPVPN-MPLS-VPN MIB に定義されているすべての通知がイネーブルになります。 • このオブジェクトを false に設定すると、MIB に定義されているすべての通知がディセーブルになります。 これは、書き込み可能な数少ないオブジェクトの 1 つです。
mplsVpnVrfConfMaxPossibleRoutes	このルータが格納できる経路の数を示す値。システムで使用可能なメモリ容量に基づくため、この値は特定できません。このため、このオブジェクトはゼロ（0）に設定されます。

MIB テーブル

PPVPN-MPLS-VPN MIB 実装は、次のテーブルをサポートしています。ここでは、これらのテーブルについて説明します。

mplsVpnVrfTable

各 VRF は、それぞれの VRF 名（mplsVpnVrfName）で参照されます。以下の表に、このテーブルの MIB オブジェクトとその機能を示します。

表 *40*：*mplsVpnVrfTable* の *PPVPN-MPLS-VPN MIB* オブジェクト

MIB オブジェクト	機能
mplsVpnVrfName	この VRF に関連付けられた名前。このオブジェクトがテーブルのインデックスとして使用される場合、最初のオクテットは文字列の長さで、後続のオクテットは各文字の ASCII コードとなります。たとえば、「vpn1」は 4.118.112.110.49 と表されます。

mplsVpnVrfDescription	VRF の説明。これを指定するには、次のコンフィギュレーション コマンドを使用します。
	`Router(config)# ip vrf vrf-name`
	`Router(config-vrf)# description vrf-description`

MIB オブジェクト	機能

mplsVpnVrfRouteDistinguisher	この VRF のルート識別子。これを指定するには、次のコンフィギュレーション コマンドを使用します。
	
	`Router(config)# ip vrf vrf-name`
	`Router(config-vrf)# rd route-distinguisher`

mplsVpnVrfCreationTime	この VRF エントリが作成されたときの sysUpTime の値。
mplsVpnVrfOperStatus	この VRF の動作ステータス。VRF に関連付けられた少なくとも 1 つのインターフェイスがアップであると、VRF はアップ（1）となります。次の場合には、VRF はダウン（2）となります。 • ifOperStatus がアップ（1）となっているインターフェイスがない。 • この VRF に関連付けられたインターフェイスがない。
mplsVpnVrfActiveInterfaces	この VRF に割り当てられたインターフェイスのうち、アップしているインターフェイスの数。
mplsVpnVrfAssociatedInterfaces	動作ステータスとは関係なく、この VRF に割り当てられたインターフェイスの数。

mplsVpnVrfConfMidRouteThreshold	中間ルートしきい値。VRF ルートの数がこのしきい値を超えた場合、mplsNumVrfRouteMidThreshExceeded 通知が送信されます（通知がイネーブルで、かつ設定されている場合）。この値は、コンフィギュレーション モードで次のように **maximum routes***limit* {*warn-threshold* | **warn-only**} コマンドを使用して、最大値に占める割合として設定できます。
	
	`Router(config)# ip vrf vpn1`
	`Router(config-vrf)# maximum routes 1000 50`
	VRF vpn1 に対する中間しきい値または警告しきい値が、最大ルートしきい値の 50% として設定されます。
	次のコマンドでは、1000 ルートの中間しきい値を設定しています。このしきい値を超えると、mplsNumVrfRouteMidThreshExceeded 通知が送信されます。ただし、このコマンドでは最大ルートしきい値を設定していないため、引き続きルートを追加できます。
	
	`Router(config-vrf)# maximum routes 1000 warn-only`

MIB オブジェクト	機能

mplsVpnVrfConfHighRouteThreshold	最大ルートしきい値。VRF ルートの数がこのしきい値を超えた場合、mplsNumVrfRouteMaxThreshExceeded 通知が送信されます（通知がイネーブルで、かつ設定されている場合）。この値は、コンフィギュレーション モードで次のように **maximum routes***limit* {*warn-threshold* | **warn-only**} コマンドを使用して設定できます。
	
	`Router(config)# ip vrf vpn2`
	`Router(config-vrf)# maximum routes 1000 75`
	VRF vpn2 に対して最大ルートしきい値が 1000 ルートに設定されるとともに、中間しきい値または警告しきい値が最大ルートしきい値の 75% に設定されます。

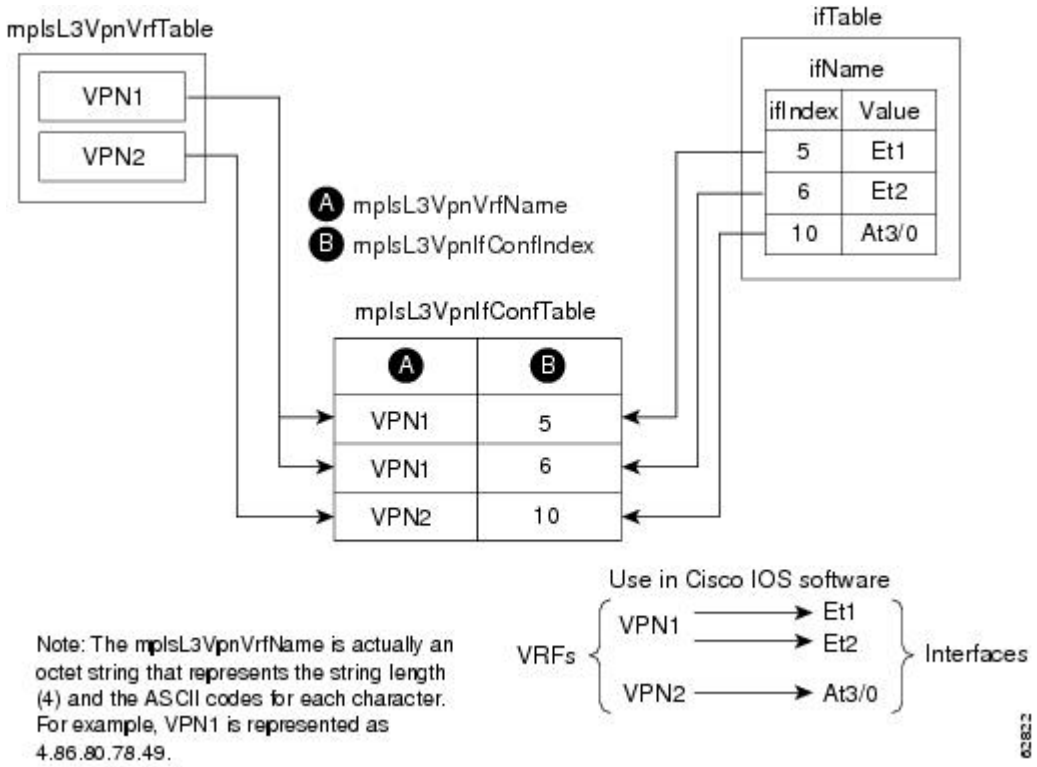
mplsVpnVrfConfMaxRoutes	この値は、mplsVpnVrfConfHighRouteThreshold と同じです。
mplsVpnVrfConfLastChanged	VRF の設定が変更されたとき、または VRF からインターフェイスが割り当てられたか、割り当てられなかったときの sysUpTime の値。 （注） このオブジェクトは、このテーブルの値が変更されたときにだけ更新されます。
mplsVpnVrfConfRowStatus	読み取り専用の実装。このオブジェクトは通常「active(1)」になりますが、VRF が最近削除された場合には「notInService(2)」になることもあります。
mplsVpnVrfConfStorageType	読み取り専用の実装。このオブジェクトは常に「volatile(2)」となります。

mplsVpnInterfaceConfTable

Cisco ソフトウェアでは、VRF は 1 つの MPLS VPN に関連付けられます。1 つの VRF に 0 個以上のインターフェイスを関連付けることができます。VRF は、MIB II（IFMIB）のインターフェイス グループの ifTable に定義されているインターフェイスを使用します。IFMIB には、インターフェイスを管理するためのオブジェクトが定義されています。この MIB の ifTable には、ネットワーク内の各インターフェイスに関する情報が登録されています。mplsVpnInterfaceConfTable は、

mplsVpnVrfTable の VRF を ifTable の転送インターフェイスに関連付けます。次の図に、ifTable および mplsVpnInterfaceConfTable に定義されている VRF とインターフェイスとの関係を示します。

図 *22*：***VRF***、インターフェイス ***MIB***、および ***mplsVpnInterfaceConfTable***



VPN インターフェイス コンフィギュレーション テーブル（mplsVpnInterfaceConfTable）のエントリは、各 VRF に割り当てられるインターフェイスを示します。このテーブルの情報は、**show ip vrf** コマンドを使用したときにも表示されます。

mplsVpnInterfaceConfTable には、インターフェイスをどのように VRF に割り当てるかが記載されています。ラベル スイッチング ルータ（LSR）が、MPLS VPN に対応できるインターフェイスごとに、このテーブルにエントリを作成します。

mplsVpnInterfaceConfTable のインデックスが次のように作成されます。

- mplsVpnVrfName：VRF 名
- mplsVpnInterfaceConfIndex：VRF に割り当てられたインターフェイスのインターフェイス MIB からの ifIndex と同じ識別子

以下の表に、このテーブルの MIB オブジェクトとその機能を示します。

表 *41*： ***mplsVpnInterfaceConfTable*** の ***PPVPN-MPLS-VPN MIB*** オブジェクト

MIB オブジェクト	機能
mplsVpnInterfaceConfIndex	VRF に割り当てられたこのインターフェイスのインターフェイス MIB ifIndex を提供します。
mplsVpnInterfaceLabelEdgeType	インターフェイスがプロバイダー エッジ インターフェイス（1）であるか、カスタマー エッジ インターフェイス（2）であるかを示します。 この値は常に providerEdge（1）になります。Cisco IOS では、customerEdge インターフェイスは VRF に割り当てられず、このテーブルに表示されないためです。
mplsVpnInterfaceVpnClassification	このインターフェイスが提供している VPN のタイプを指定します。Carrier Supporting Carrier（CsC）（1）、企業（2）、InterProvider（3）のいずれかになります。 このインターフェイスで MPLS がディセーブルである場合には企業（2）に設定され、MPLS がイネーブルになっている場合には Carrier Supporting Carrier（1）に設定されます。
mplsVpnInterfaceVpnRouteDistProtocol	このインターフェイスで BGP によるルートの再配布に使用されているルート配布プロトコルを示します。BGP（2）、OSPF（3）、RIP（4）のいずれかになります。 Cisco ソフトウェアでは、インターフェイス単位ではなく VRF 単位でルータ プロセスが定義され、再配布されます。このため、同じ VRF に割り当てられたすべてのインターフェイスで、このオブジェクトの値が同じものになります。
mplsVpnInterfaceConfStorageType	読み取り専用の実装。このオブジェクトは常に「volatile(2)」となります。
mplsVpnInterfaceConfRowStatus	読み取り専用の実装。このオブジェクトは通常「active(1)」になりますが、VRF が最近削除された場合には「notInService(2)」になることもあります。

mplsVpnVrfRouteTargetTable

ルート ターゲット テーブル（mplsVpnVrfRouteTargetTable）には、特定の VRF に対して定義されているルート ターゲット コミュニティが登録されます。MPLS VPN インスタンスをサポートする VRF 用に設定されたターゲットごとに、LSR がこのテーブルにエントリを作成します。

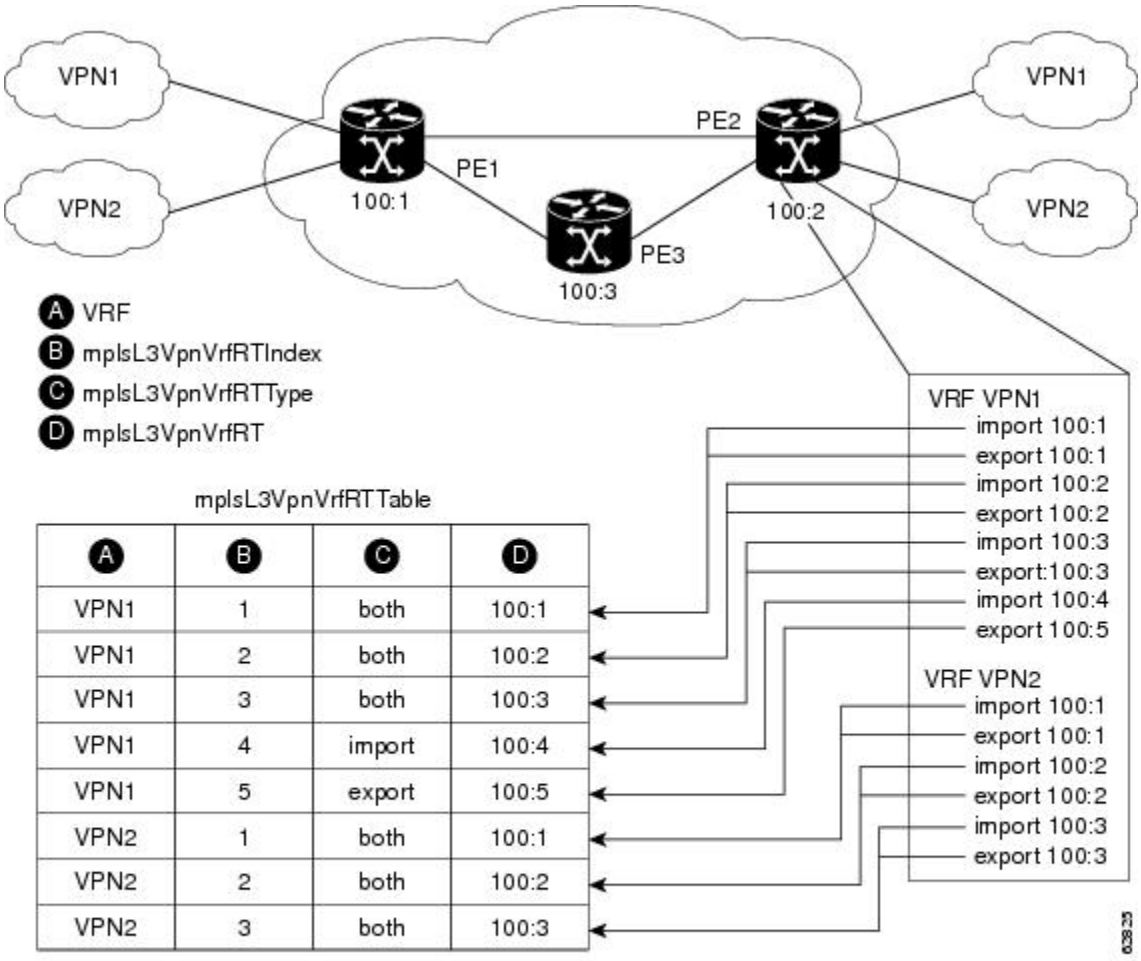
VPN ルーティング情報の配布は、BGP 拡張コミュニティによって実装される VPN ルート ターゲット コミュニティを使用して制御されます。VPN ルーティング情報の配布は、次のように機能します。

- カスタマーエッジ（CE）ルータから学習した VPN ルートが BGP に注入されると、VPN ルート ターゲット拡張コミュニティ属性のリストが、そのルートに関連付けられます。通常、ルート ターゲット コミュニティ値のリストは、ルートの学習元の VRF に関連付けられているルート ターゲットのエクスポート リストから設定されます。
- ルート ターゲット拡張コミュニティのインポート リストは、各 VRF に関連付けられています。インポート リストには、ルートが VRF にインポートされるために、ルートに設定されている必要のある、ルート ターゲット拡張コミュニティ属性が定義されています。たとえば、ある特定の VRF のインポート リストにルート ターゲット コミュニティ A、B、および C が含まれている場合、これらのルート ターゲット拡張コミュニティ（A、B、または C）のいずれかを伝送するすべての VPN ルートが VRF にインポートされます。

以下の図に、設定例および mplsVpnVrfRouteTargetTable との関係を示します。各 PE ルータにルート ターゲット テーブルが存在します。設定例には、ルート識別子（RD）が 100:1、100:2、および

100:3 のルータが示されています。図には RD が 100:4 および 100:5 のルータは示されていませんが、PE2 のルート ターゲットおよび mplsVpnVrfRouteTargetTable には含まれています。

図 *23*：設定例および ***mplsVpnVrfRouteTargetTable***



mplsL3VpnVrfRTTable

A	B	C	D
VPN1	1	both	100:1
VPN1	2	both	100:2
VPN1	3	both	100:3
VPN1	4	import	100:4
VPN1	5	export	100:5
VPN2	1	both	100:1
VPN2	2	both	100:2
VPN2	3	both	100:3

Note: The mplsL3VpnVrfName is actually an octet string that represents the string length (4) and the ASCII codes for each character. For example, VPN1 is represented as 4.86.80.78.49.

mplsVpnVrfRouteTargetTable には、各 VRF のインポート ルート ターゲットおよびエクスポート ルート ターゲットが登録されています。次の機能によって、テーブルにインデックスが作成されます。

- mplsVpnVrfName：VRF 名
- mplsVpnVrfRouteTargetIndex：ルート ターゲット エントリ識別子
- mplsVpnVrfRouteTargetType：エントリがインポート ルート ターゲット、エクスポート ルート ターゲット、またはその両方に定義されているかを指定する値。

以下の表に、このテーブルの MIB オブジェクトとその機能を示します。

表 *42*：*mplsVpnVrfRouteTargetTable* の *PPVPN-MPLS-VPN MIB* オブジェクト

MIB オブジェクト	機能
mplsVpnVrfRouteTargetIndex	テーブル内での各ルートターゲットの位置を定義する値。
mplsVpnVrfRouteTargetType	エントリが表すルートターゲットのタイプを決定します。インポート（1）、エクスポート（2）、両方（3）のいずれかになります。
mplsVpnVrfRouteTarget	このターゲットのルート識別子を決定します。
mplsVpnVrfRouteTargetDescr	ルート ターゲットの説明。このオブジェクトはサポートされません。このため、mplsVpnVrfRouteTarget と同じものになります。
mplsVpnVrfRouteTargetRowStatus	読み取り専用の実装。このオブジェクトは通常「active(1)」になりますが、VRF が最近削除された場合には「notInService(2)」になることもあります。

mplsVpnVrfBgpNbrAddrTable

BGP ネイバー アドレス テーブル（mplsVpnVrfBgpNbrAddrTable）は、特定の VRF に対して定義されている MPLS 外部ボーダー ゲートウェイ プロトコル（eBGP）ネイバーです。LSR が、VRF のアドレス ファミリに定義されている BGP ネイバーごとにエントリを作成します。

mplsVpnVrfBgpNbrAddrTable のインデックスが次のように作成されます。

- mplsVpnVrfName：VRF 名
- mplsVpnInterfaceConfIndex：VRF に割り当てられたインターフェイスのインターフェイス MIB からの ifIndex と同じ識別子
- mplsVpnVrfBgpNbrIndex：ネイバーの IP アドレス

以下の表に、このテーブルの MIB オブジェクトとその機能を示します。

表 *43*：*mplsVpnVrfBgpNbrAddrTable* の *PPVPN-MPLS-VPN MIB* オブジェクト

MIB オブジェクト	機能
mplsVpnVrfBgpNbrIndex	eBGP ネイバーの IPv4 アドレス。

MIB オブジェクト	機能
mplsVpnVrfBgpNbrRole	この eBGP ネイバーのロール。カスタマー エッジ（1）またはプロバイダーエッジ（2）になります。オブジェクト mplsVpnInterfaceVpnClassification が CSC である場合、この値はプロバイダーエッジ（2）になり、それ以外の場合はカスタマー エッジ（1）になります。
mplsVpnVrfBgpNbrType	この eBGP ネイバーのアドレスタイプ。MIB は、IPv4（1）だけをサポートします。このため、このオブジェクトは「ipv4 (1)」を返します。
mplsVpnVrfBgpNbrAddr	eBGP ネイバーの IP アドレス。
mplsVpnVrfBgpNbrRowStatus	読み取り専用の実装。このオブジェクトは通常「active(1)」になりますが、VRF が最近削除された場合には「notInService(2)」になることもあります。
mplsVpnVrfBgpNbrStorageType	読み取り専用の実装。このオブジェクトは常に「volatile(2)」となります。

mplsVpnVrfSecTable

VRF セキュリティ テーブル（mplsVpnVrfSecTable）には、VRF ごとにセキュリティに関する説明が登録されています。LSR が、MPLS VPN に対応できる VRF ごとに、このテーブルにエントリを作成します。

mplsVpnVrfSecTable は mplsVpnVrfTable を強化したもので、作成されるインデックスは同じです。

以下の表に、このテーブルの MIB オブジェクトとその機能を示します。

表 *44*：*mplsVpnVrfSecTable* の *PPVPN-MPLS-VPN MIB* オブジェクト

MIB オブジェクト	機能
mplsVpnVrfSecIllegalLabelViolations	VRF インターフェイスで不正に受信したラベルの数。このオブジェクトでは、不正なラベルだけがカウントされます。このため、このオブジェクトは、MPLS 対応（CSC 状況）の VRF インターフェイスにだけ適用されます。 ラベルが有効なラベル範囲を上回るか下回るか、ラベルがグローバル ラベル転送テーブルにないか、または誤った VRF（受信インターフェイスのテーブル ID と適切な VRF ラベル転送テーブルにあるテーブル ID とが一致しない）でラベルが受信されるたびに、このカウンタが増分されます。

MIB オブジェクト	機能
mplsVpnVrfSecIllegalLabelRcvThresh	この VRF で受信した不正なラベルの通知しきい値。このインターフェイスで受信した不正なラベルの数がこのしきい値を超えると、mplsNumVrfSecIllegalLabelThreshExceeded 通知が送信されます（通知がイネーブルになっており、かつ設定されている場合）。 このオブジェクトは、SNMP SET 操作をサポートするこの MIB エージェントの数少ないオブジェクトの 1 つです。この操作を使用すると、この値を変更できます。

mplsVpnVrfPerfTable

VRF パフォーマンス テーブル（mplsVpnVrfPerfTable）には、各 VRF の統計パフォーマンス情報が登録されています。LSR が、MPLS VPN に対応できる VRF ごとに、このテーブルにエントリを作成します。

mplsVpnVrfPerfTable は mplsVpnVrfTable を強化したもので、同じインデックスが作成されています。

以下の表に、このテーブルの MIB オブジェクトとその機能を示します。

表 *45*：*mplsVpnVrfPerfTable* の *PPVPN-MPLS-VPN MIB* オブジェクト

MIB オブジェクト	機能
mplsVpnVrfPerfRoutesAdded	ライフタイムの期間中にこの VRF に追加されたルートの数。
mplsVpnVrfPerfRoutesDeleted	この VRF から削除されたルートの数。
mplsVpnVrfPerfCurrNumRoutes	この VRF 内に現在定義されているルートの数。

mplsVpnVrfRouteTable

VRF ルーティング テーブル（mplsVpnVrfRouteTable）には、各 VRF の IP ルーティング テーブル情報が登録されています。このテーブルの情報には、**show ip route vrf***vrf-name* コマンドでアクセスすることもできます。上記の図に示されている PE1 を例に説明します。

- **show ip route vrf vpn1** コマンドを実行すると、次のような結果が得られます。

```
Router# show ip route vrf vpn1
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

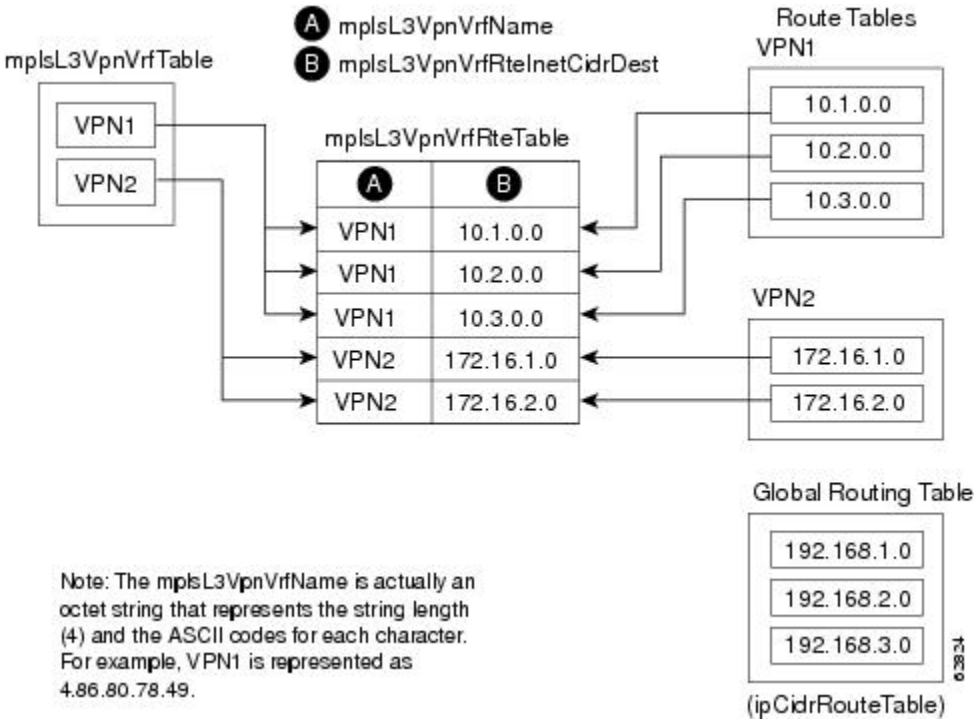
```
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
!
Gateway of last resort is not set
!
     10.0.0.0/32 is subnetted, 3 subnets
B       10.3.0.0 [200/0] via 192.168.2.1, 04:36:33
C       10.1.0.0/16 is directly connected, FastEthernet1
C       10.2.0.0/16 [200/0] directly connected FastEthernet2, 04:36:33
```

- **show ip route vrf vpn2** コマンドを実行すると、次のような結果が得られます。

```
Router# show ip route vrf vpn2
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
!
Gateway of last resort is not set
!
     172.16.0.0/32 is subnetted, 2 subnets
B       172.16.2.0 [200/0] via 192.168.2.1, 04:36:33
C       172.16.1.0 is directly connected, ATM 3/0
```

以下の図に、ルーティング テーブル、VRF、および mplsVpnVrfRouteTable の関係を示します。**show ip route vrf***vrf-name* コマンドを使用して、VPN1 および VPN2 ルート テーブルに関する情報を表示できます。グローバル ルート テーブルは、IP-FORWARD-MIB の ipCidrRouteTable と同じです。**show ip route** コマンドを使用して、グローバル ルート テーブルに関する情報を表示できます。

図 *24*：ルート テーブル、*VRF*、および *mplsVpnVrfRouteTable*



LSR が、設定されるルートごとにこのテーブルにエントリを作成します。MPLS VPN に対応できる特定の VRF のコンテキスト内で、ダイナミックまたはスタティックに作成します。

mplsVpnVrfRouteTable のインデックスが次のように作成されます。

- mplsVpnVrfName：VRF ルーティング コンテキストを提供する VRF 名
- mplsVpnVrfRouteDest：IP 宛先アドレス
- mplsVpnVrfRouteMask：IP 宛先マスク
- mplsVpnVrfRouteTos：IP ヘッダー ToS ビット
- mplsVpnVrfRouteNextHop：ルート エントリごとのネクスト ホップの IP アドレス

（注） ToS ビットはサポートされません。このため、常に 0 になります。

以下の表に、mplsVpnVrfRouteTable の MIB オブジェクトとその機能を示します。このテーブルは、VRF 固有のルートを表します。グローバル ルーティング テーブルは、IP-FORWARD-MIB の ipCidrRouteTable です。

表 46：*mplsVpnVrfRouteTable* の *PPVPN-MPLS-VPN MIB* オブジェクト

MIB オブジェクト	機能
mplsVpnVrfRouteDest	このルートに対して定義されている宛先 IP アドレス。
mplsVpnVrfRouteDestAddrType	IP 宛先アドレスのアドレス タイプ（mplsVpnVrfRouteDest）。この MIB 実装は、IPv4（1）だけをサポートします。このため、このオブジェクトの値は「ipv4(1)」となります。
mplsVpnVrfRouteMask	このルートに対して定義されている宛先 IP アドレス マスク。
mplsVpnVrfRouteMaskAddrType	宛先 IP アドレス マスクのアドレス タイプ。この MIB 実装は、IPv4（1）だけをサポートします。このため、このオブジェクトの値は「ipv4(1)」となります。
mplsVpnVrfRouteTos	このルートの IP ヘッダーから取得された ToS ビット。Cisco ソフトウェアが ToS ビットとしてサポートする値はゼロ（0）だけです。このため、このオブジェクトは常に 0 となります。
mplsVpnVrfRouteNextHop	このルートに対して定義されているネクスト ホップ IP アドレス。

MIB オブジェクト	機能
mplsVpnVrfRouteNextHopAddrType	ネクスト ホップ IP アドレスのアドレス タイプ。この MIB 実装は、IPv4（1）だけをサポートします。このため、このオブジェクトの値は「ipv4(1)」となります。
mplsVpnVrfRouteIfIndex	このルートの転送に使用されるインターフェイスのインターフェイス MIB ifIndex。ルートにインターフェイスが定義されていない場合には、オブジェクトは 0 になります。
mplsVpnVrfRouteType	このルートがローカルに定義されたルートであるか、リモートに定義されたルートであるかを定義します。
mplsVpnVrfRouteProto	このルートを VRF に追加したルーティング プロトコル。
mplsVpnVrfRouteAge	このルートが最後に更新されてから経過した秒数。
mplsVpnVrfRouteInfo	他の MIB から詳細な情報へのポインタ。このオブジェクトはサポートされず、常に「nullOID(0.0)」を返します。
mplsVpnVrfRouteNextHopAS	このルートのネクスト ホップの自律システム番号。このオブジェクトはサポートされず、常に 0 になります。
mplsVpnVrfRouteMetric1	このルートに使用されるプライマリ ルーティング メトリック。
mplsVpnVrfRouteMetric2 mplsVpnVrfRouteMetric3 mplsVpnVrfRouteMetric4 mplsVpnVrfRouteMetric5	このルートに使用される代替ルーティング メトリック。これらのオブジェクトは、Cisco Interior Gateway Routing Protocol（IGRP）および Cisco Enhanced Interior Gateway Routing Protocol（EIGRP）の場合にだけサポートされます。これらのオブジェクトによって、ルートに使用される帯域幅メトリックが表示されます。それ以外の場合、これらの値は -1 に設定されます。
mplsVpnVrfRouteRowStatus	読み取り専用の実装。このオブジェクトは通常「active(1)」になりますが、VRF が最近削除された場合には「notInService(2)」になることもあります。
mplsVpnVrfRouteStorageType	読み取り専用の実装。このオブジェクトは常に「volatile(2)」となります。

PPVPN-MPLS-VPN MIB 通知

ここでは、サポートされている PPVPN-MPLS-VPN MIB 通知について説明します。次の項目を取り上げます。

PPVPN-MPLS-VPN MIB 通知イベント

PPVPN-MPLS-VPN MIB では次の通知がサポートされます。

- mplsVrfIfUp：インターフェイスが確立され、VRF インスタンスがそのインターフェイスに割り当てられると、この通知が NMS に送信されます。
- mplsVrfIfDown：VRF がインターフェイスから削除されるか、またはインターフェイスの動作が「アップ」状態から「ダウン」状態に遷移すると、この通知が生成されて NMS に送信されます。
- mplsNumVrfRouteMidThreshExceeded：中間（警告）しきい値を超えると、この通知が生成されて送信されます。このしきい値は、次のコマンドを使用して CLI で設定できます。

```
Router(config)# ip vrf vrf-name
Router(config-vrf)#  maximum routes limit warn-threshold (% of max)
```

warn-threshold 引数は、*limit* 引数に指定した最大ルートの割合となります。また、次のコマンドで中間しきい値を設定することもできます。*limit* 引数には、警告しきい値を指定します。

```
Router(config-vrf)# maximum routes limit warn-threshold (% of max)
```

この通知は、しきい値を超えたときにだけ NMS に送信されます。（警告しきい値と最大しきい値の比較については、以下の図を参照してください）。ルートの数がこのしきい値を下回り、再度しきい値を超えるたびに、通知が NMS に送信されます。

- MplsNumVrfRouteMaxThreshExceeded：**maximum routes** コマンドの *limit* 引数で定義した最大数のルートがすでに含まれている VRF でルートを作成しようとすると、この通知が生成されて送信されます。

```
Router(config)# ip vrf vrf-name
Router(config-vrfmaximum routes limit warn-threshold (% of max)
```

最大しきい値を超えようとすると、トラップ通知が NMS に送信されます。ルートの数が最大しきい値を下回ると、再度最大しきい値に達するまでは、さらに mplsNumVrfRouteMaxThreshExceeded 通知が送信されることはありません。（この通知のしくみを示す例、および最大しきい値と警告しきい値の比較については、以下の図を参照してください。）

（注） VRF のルート数は、**maximum routes** コマンドによって設定されます。VRF に、**maximum routes***limitwarn-threshold* コマンドで設定したルート数を超えるルートを含めることはできません。PPVPN-MPLS-VPN MIB が実装される前は、このしきい値（または警告しきい値）に達しても、通知が送信されませんでした。

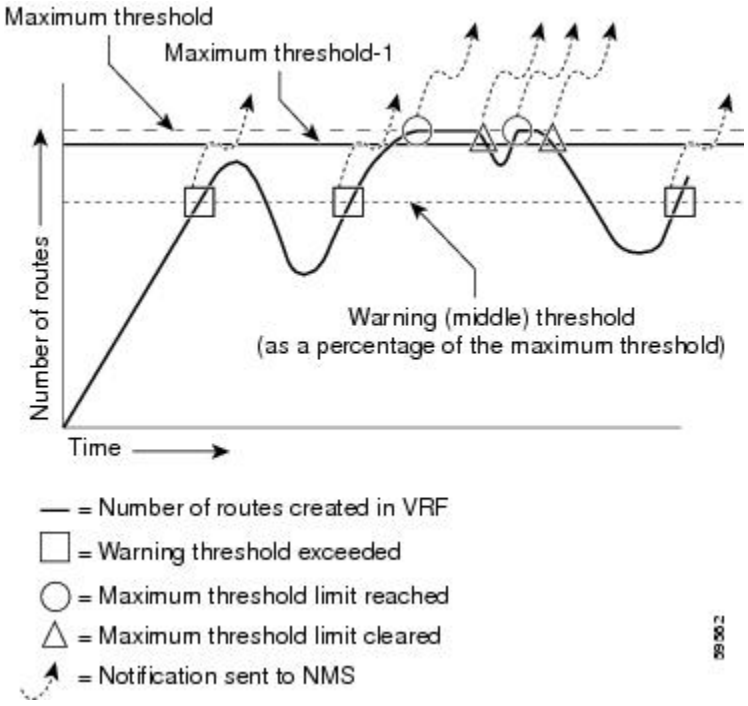
- mplsNumVrfSecIllegalLabelThreshExceeded：VRF インターフェイスで受信した不正なラベルの数がしきい値 mplsVpnVrfSecIllegalLabelRcvThresh を超えると、この通知が生成されて送信されます。このしきい値は、値 0 で定義されています。このため、VRF で初めて不正なラベルを受信すると、通知が送信されます。ラベルが有効なラベル範囲内にない場合、ラベルがラベル転送情報ベース（LFIB）に登録されていない場合、またはメッセージのテーブル ID が LFIB にあるラベルのテーブル ID に一致しない場合、ラベルは不正であると見なされます。

CISCO-IETF-PPVPN-MPLS-VPN MIB 通知イベント

次に示す CISCO-IETF-PPVPN-MPLS-VPN MIB 通知は Cisco ソフトウェアでサポートされています。

- cMplsNumVrfRouteMaxThreshCleared：VRF でのルート数が最大ルート数を超えようとしたものの、その後最大ルート数を下回ると、生成されて送信されます。最大ルート数がすでに含まれている VRF でルートを作成しようとすると、mplsNumVrfRouteMaxThreshExceeded 通知が送信されます（有効な場合）。ルート数が制限値を下回るように、VRF からルートを削除すると、cMplsNumVrfRouteMaxThreshCleared 通知が送信されます。**clear ip route vrf** コマンドを使用して、VRF からすべてのルートをクリアできます（cMplsNumVrfRouteMaxThreshCleared 通知がいつ送信されるかについては、次の図を参照してください）。

図 *25*：警告しきい値と最大しきい値の比較



通知仕様

SNMPv1 通知では、各 VPN 通知に汎用タイプの ID と、通知タイプを識別するための企業固有タイプ ID が含まれます。

- すべての VPN 通知に対する汎用タイプは、SNMP に対して定義されている汎用通知タイプの 1 つではないため、「企業固有」です。
- 企業固有タイプは次のように識別されます。
  - *mplsVrfIfUp* の場合は 1
  - *mplsVrfIfDown* の場合は 2
  - *mplsNumVrfRouteMidThreshExceeded* の場合は 3
  - *mplsNumVrfRouteMaxThreshExceeded* の場合は 4
  - *mplsNumVrfSecIllegalLabelThreshExceeded* の場合は 5
  - cMplsNumVrfRouteMaxThreshCleared の場合は 6

SNMPv2 では、通知タイプは通知メッセージ内に含まれている SnmpTrapOID Varbind（オブジェクト ID（OID）タイプと値で構成されている Variable Binding）で識別されます。

また、各通知には、PPVPN-MPLS-VPN MIB から取得された 2 つの追加的なオブジェクトも含まれています。この2つのオブジェクトから、イベントに関する次のような追加情報が得られます。

- VRF インターフェイスのアップ/ダウン通知には、*mplsVpnInterfaceConfIndex* 変数と *mplsVpnVrfName* 変数が含まれています。これらの変数はそれぞれ、SNMP インターフェイスインデックスと VRF 名を示します。
- 中間しきい値通知および最大しきい値通知には、*mplsVpnVrfName* 変数（VRF 名）と、VRF 内の現在のルート数を示す *mplsVpnVrfPerfCurrNumRoutes* 変数が含まれています。
- 不正なラベル通知には、*mplsVpnVrfName* 変数（VRF 名）と、VPN で現在の不正なラベル数を保持する *mplsVpnVrfSecIllegalLabelViolations* 変数が含まれています。

PPVPN-MPLS-VPN MIB 通知の監視

PPVPN-MPLS-VPN MIB 通知が有効になっている場合（『Cisco IOS Multiprotocol Label Switching Command Reference』で **snmp-server enable traps mpls vpn** コマンドを参照）、Cisco ソフトウェア内の特定の MPLS VPN イベントに関連する通知メッセージが生成され、ネットワーク内の指定された NMS に送信されます。SNMPv1 または SNMPv2 通知をサポートするユーティリティはいずれも、通知メッセージを受信できます。

PPVPN-MPLS-VPN MIB 通知メッセージを監視するには、SNMP 通知を表示するユーティリティをサポートしている NMS にログインし、表示ユーティリティを起動します。

PPVPN-MPLS-VPN MIB でサポートされていないオブジェクト

mplsVpnVrfBgpPathAttrTable の次のオブジェクトは、Cisco ソフトウェアにおける SNMP を使用した MPLS VPN 機能の管理ではサポートされません。

- mplsVpnVrfBgpPathAttrPeer
- mplsVpnVrfBgpPathAttrIpAddrPrefixLen
- mplsVpnVrfBgpPathAttrIpAddrPrefix
- mplsVpnVrfBgpPathAttrOrigin
- mplsVpnVrfBgpPathAttrASPathSegment
- mplsVpnVrfBgpPathAttrNextHop
- mplsVpnVrfBgpPathAttrMultiExitDisc
- mplsVpnVrfBgpPathAttrLocalPref
- mplsVpnVrfBgpPathAttrAtomicAggregate
- mplsVpnVrfBgpPathAttrAggregatorAS
- mplsVpnVrfBgpPathAttrAggregatorAddr
- mplsVpnVrfBgpPathAttrCalcLocalPref
- mplsVpnVrfBgpPathAttrBest
- mplsVpnVrfBgpPathAttrUnknown

MPLS VPN--MIB サポートの設定方法

SNMP コミュニティの設定

SNMP コミュニティ ストリングでは、SNMP マネージャと SNMP エージェントとの関係を定義します。コミュニティ ストリングは、ルータ上のエージェントへのアクセスを制御するパスワードのように機能します。

SNMP コミュニティを設定するには、次の作業を実行します。

手順の概要

1. **enable**
2. **show running-config** [*options*]
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [*acl-number*]
5. **do copy running-config startup-config**
6. **exit**
7. **show running-config** [*options*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** [*options*] 例： `Router# show running-config`	SNMP エージェントがすでに実行中かどうかが判別される実行コンフィギュレーションが表示されます。 • SNMP の情報が表示されない場合は、次のステップに進みます。SNMP 情報が表示される場合、必要に応じて、情報を変更できます。
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [*acl-number*] 例： `Router(config)# snmp-server community comaccess ro`	SNMP プロトコルへのアクセスを許可するようにコミュニティ アクセス ストリングを設定します。 • *string* 引数はパスワードのように機能し、SNMP プロトコルへのアクセスを許可します。 • **view***view-nameview-name* キーワードと引数のペアには、以前に定義されたビューの名前を指定します。ビューには、コミュニティで使用できるオブジェクトが定義されています。 • **ro** キーワードは、読み取り専用アクセスを指定します。MIB オブジェクトを取得できるのは、許可された管理ステーションだけです。

	コマンドまたはアクション	目的
		• **rw** キーワードは、読み取りと書き込みアクセスを指定します。MIB オブジェクトの取得と変更の両方を実行できるのは、許可された管理ステーションです。 • *acl-number* 引数は、1 ～ 99 の整数で、コミュニティ ストリングを使用した SNMP エージェントへのアクセスが許可される IP アドレスのアクセス リストを指定します。
ステップ 5	**do copy running-config startup-config** 例： `Router(config)# do copy running-config startup-config`	変更された設定をスタートアップ コンフィギュレーション ファイルとして NVRAM に保存します。 • **do** コマンドを使用すると、コンフィギュレーション モードで EXEC レベルのコマンドを実行できます。
ステップ 6	**exit** 例： `Router(config)# exit`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** [*options*] 例： `Router# show-running config | include smnp-server`	（任意）ルータ上の現在の設定情報、特定のインターフェイスの情報、またはマップクラス情報を表示します。 • **show running-config** コマンドを使用すると、**snmp-server** のステートメントが出力に表示されることをチェックできます。

ルータによる SNMP トラップ送信の設定

SNMP トラップをホストに送信するようにルータを設定するには、この作業を実行します。

snmp-server host コマンドを使用して、トラップを受信するホストを指定します。**snmp-server enable traps** コマンドでは、指定したトラップのトラップ生成メカニズムをグローバルにイネーブルにします。

ホストでトラップを受信するには、そのホストに **snmp-server host** コマンドを設定する必要があります。また通常は、**snmp-server enable traps** コマンドでトラップをグローバルにイネーブルにされていることも必要です。

（注） **snmp-server host** コマンド自体を使用して *community-string* 引数を設定できますが、**snmp-server host** コマンドを使用してこのストリングを定義してから **snmp-server community** コマンドを使用することを推奨します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server host***host-addr* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [*notification-type*] [**vrf***vrf-name*]
4. **snmp-server enable traps mpls vpn** [**illegal-label**] [**max-thresh-cleared**] [**max-threshold**] [**mid-threshold**] [**vrf-down**] [**vrf-up**]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server host***host-addr* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port***port*] [*notification-type*] [**vrf***vrf-name*] 例： `Router(config)# snmp-server host 172.20.2.160 traps comaccess mpls-vpn`	SNMP 通知操作の受信者を指定します。 • *host-addr* 引数には、ホスト（ターゲット受信者）の名前またはインターネット アドレスを指定します。 • **traps** キーワードを指定すると、このホストに SNMP トラップが送信されます。これはデフォルトです。 • **informs** キーワードを指定すると、このホストに SNMP 応答要求が送信されます。 • **version** キーワードは、トラップの送信に使用する SNMP のバージョンを指定します。最も安全なモデルはバージョン 3 です。このバージョンでは、**priv** キーワードを使用してパケットを暗号化できるためです。**version** キーワードを使用する場合は、次のいずれかを指定する必要があります。

	コマンドまたはアクション	目的
		• **1**：SNMPv1。情報の場合は、このオプションを使用できません。 • **2c**：SNMPv2C。 • **3**：SNMPv3。**version3** キーワードのあとに 3 つのオプション キーワード（**auth**、**noauth**、**priv**）を指定できます。 • *community-string* 引数は、通知操作で送信される、パスワードに似たコミュニティ文字列です。 • **udp-port***port* キーワードと引数のペアには、使用するホストの User Datagram Protocol（UDP）ポートを指定します。デフォルトは 162 です。 • *notification-type* 引数には、ホストに送信する通知のタイプを指定します。タイプが指定されていない場合、すべての通知が送信されます。 • **vrf***vrf-name* キーワードと引数のペアには、SNMP 通知の送信に使用する VRF テーブルを指定します。
ステップ 4	**snmp-server enable traps mpls vpn** [**illegal-label**] [**max-thresh-cleared**] [**max-threshold**] [**mid-threshold**] [**vrf-down**] [**vrf-up**] 例： `Router(config)# snmp-server enable traps mpls vpn vrf-down vrf-up`	ルータが MPLS VPN 固有の SNMP 通知（トラップと応答要求）を送信できるようにします。 • **illegal-label** キーワードを指定すると、VRF インターフェイスで不正なラベルを受信した場合に通知が送信されようになります。ラベルが有効な範囲内にないか、LFIB に登録されていないか、または LFIB にあるラベルのテーブル ID に一致しない場合、ラベルは不正であると見なされます。 • **max-thresh-cleared** キーワードを指定すると、ルート数が最大ルート数を超えようとしたあとで制限値を下回った場合に、通知が送信されるようになります。 • **max-threshold** キーワードを指定すると、ルートを作成しようとしたものの、最大ルート数に達したために作成できなかった場合に、通知が送信されるようになります。ルートの数が最大しきい値を下回ると、再度最大しきい値に達するまでは、さらに mplsNumVrfRouteMaxThreshExceeded 通知が送信されることはありません。最大しきい値は、VRF コンフィギュレーション モードの **maximum routes** コマンドによって決まります。 • **mid-threshold** キーワードを指定すると、作成したルートの数が警告しきい値を超えた場合に、警告の通知が送信されるようになります。この警告は、警告しきい値を超えたときにだけ送信されます。 • **vrf-down** キーワードを指定すると、VRF がインターフェイスから削除されるか、またはインターフェイスがダウン状態になった場合に、通知が送信されるようになります。

	コマンドまたはアクション	目的
		• **vrf-up** キーワードを指定すると、動作しているインターフェイスにVRFが割り当てられるか、またはVRFインターフェイスがアップ状態になった場合に、通知が送信されるようになります。
ステップ 5	**end** 例： `Router(config)# end`	（任意）終了して、特権 EXEC モードに戻ります。

MPLS VPN--SNMP 通知のしきい値の設定

次の作業を実行して、MPLS VPN--SNMP 通知にしきい値を設定します。

- 中間（警告）しきい値を超えると、mplsNumVrfRouteMidThreshExceeded 通知イベントが生成されて送信されます。CLI でこのしきい値を設定するには、VRF コンフィギュレーションモードで **maximum routes** コマンドを使用します。この通知は、しきい値を超えたときにだけ NMS に送信されます。ルートの数がこのしきい値を下回り、再度しきい値を超えるたびに、通知が NMS に送信されます。
- VRF コンフィギュレーション モードの **maximum routes** コマンドで定義された最大ルート数がすでに含まれている VRF でルートを作成しようとすると、mplsNumVrfRouteMaxThreshExceeded 通知イベントが生成されて送信されます。最大しきい値を超えようとすると、トラップ通知が NMS に送信されます。ルートの数が最大しきい値を下回ると、再度最大しきい値に達するまでは、さらに mplsNumVrfRouteMaxThreshExceeded 通知が送信されることはありません。

この通知のしくみを示す例、および最大しきい値と警告しきい値の比較については、上記の図を参照してください。

（注） VRF のルート数は、**maximum routes** コマンドによって設定されます。VRF に、**maximum routes***limit warn-threshold* コマンドで設定したルート数を超えるルートを含めることはできません。PPVPN-MPLS-VPN MIB が実装される前は、このしきい値（または警告しきい値）に達しても、通知が送信されませんでした。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip vrf***vrf-name*
4. **maximum routes***limit* {*warn-threshold* | **warn-only**}
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip vrf***vrf-name* 例： `Router(config)# ip vrf vpn1`	VRF ルーティング テーブルを設定し、VRF コンフィギュレーション モードを開始します。 • *vrf-name* 引数には、VRF に割り当てられている名前を指定します。
ステップ 4	**maximum routes***limit* {*warn-threshold*	**warn-only**} 例： `Router(config-vrf)# maximum routes 10000 80`	PEルータにインポートされるルートの数が多くなりすぎないように、VRF の最大ルート数を制限します。 • *limit* 引数には、VRF で許可されるルートの最大数を指定します。範囲は 1 ～ 4,294,967,295 です。 • *warn-threshold* 引数は、*warn-threshold* 引数に設定されたルートの数に達すると警告を生成し、*limit* 引数に設定された最大数を超えるルートを拒否します。警告しきい値は、*limit* 引数に指定された最大ルート数に占める割合を 1 ～ 100 で示した値となります。 • **warn-only** キーワードを指定すると、VRF に許可された最大ルート数が制限しきい値を超えたときに、システム ロギング エラー メッセージが発行されるようになります。ただし、引き続きルートを追加できます。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Router(config-vrf)# end`	（任意）終了して、特権 EXEC モードに戻ります。

MPLS VPN--SNMP サポートの設定例

例：SNMP コミュニティの設定

次に、簡単な SNMP コミュニティ グループをイネーブルにする例を示します。この設定では、SNMP クライアントがコミュニティ ストリング comaccess を使用して読み取り専用アクセス権ですべての PPVPN-MPLS-VPN MIB オブジェクトにアクセスすることを許可しています。

```
Router# configure terminal
Router(config)# snmp-server community comaccess ro
```

MPLS VPN--MIB サポート機能に対して SNMP マスター エージェントがイネーブルになっていることを確認します。

```
Router# show running-config | include snmp-server
Building configuration...
.
snmp-server community comaccess RO
```

（注） 「snmp-server」のステートメントが表示されない場合は、ルータで SNMP がイネーブルにされていません。

例：ルータによる SNMP トラップ送信の設定

次に、VRF がアップ状態またはダウン状態に移行した場合に、ルータがコミュニティ ストリング comaccess を使用して MPLS VPN 通知をホスト 172.20.2.160 に送信する例を示します。

```
Router# configure terminal
Router(config)# snmp-server host 172.20.2.160 traps comaccess mpls-vpn
Router(config)# snmp-server enable traps mpls vpn vrf-down vrf-up
```

例：MPLS VPN--SNMP 通知のしきい値の設定

次に、ルータ上の vpn1 という VRF に最大しきい値として 10,000 ルート、警告しきい値として最大しきい値の 80 % をそれぞれ設定する例を示します。

```
Router(config)# ip vrf vpn1
Router(config-vrf)# maximum routes 10000 80
```

次に、ルータ上の vpn2 という VRF に警告しきい値として 10,000 ルートを設定する例を示します。エラー メッセージが表示されますが、このコマンドでは最大ルートしきい値を設定していないため、引き続きルートを追加できます。

```
Router(config)# ip vrf vpn2
Router(config-vrf)# maximum routes 10000 warn-only
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
MPLS および MPLS アプリケーションに関連するコマンドの説明	『*Cisco IOS Multiprotocol Label Switching Command Reference*』
MPLS VPN の設定作業	『Configuring MPLS Layer 3 VPNs』
MPLS トラフィック エンジニアリング MIB（MPLS TE MIB）に対する Cisco ソフトウェアでの SNMP エージェント サポートの説明	MPLS トラフィック エンジニアリング（TE）MIB
MPLSディストリビューションプロトコルの概念と設定作業	MPLS ラベル配布プロトコル

標準

規格	タイトル
draft-ietf-ppvpn-mpls-vpn-mib-05	『*MPLS/BGP Virtual Private Network Management Information Base Using SMIv2*』

MIB

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

RFC

RFC	タイトル
RFC 2233	『*The Interfaces Group MIB using SMIv2*』
RFC 2547	『*BGP/MPLS VPNs*』

シスコのテクニカル サポート

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

MPLS VPN--MIB サポートの機能情報

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

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

表 *47*：***MPLS VPN--MIB*** サポートの機能情報

機能名	リリース	機能情報
MPLS VPN--MIB サポート	Cisco IOS XE Release 2.1	次のコマンドが導入または変更されました。**snmp-server enable traps mpls vpn**

用語集

6VPE ルータ：IPv4 ベースの MPLS コアを介して、BGP-MPLS IPv6 VPN サービスを提供するプロバイダー エッジ ルータ。コア方向のインターフェイスで 6PE 概念を実装する IPv6 VPN PE のデュアル スタック ルータです。

自律システム：同じルーティング プロトコルを共有し、同じシステム管理者の管理下にあるネットワークの集合。

ASN.1：Abstract Syntax Notation One（抽象構文記法 1）。特定のコンピュータ構造および表現技法に依存しないデータ タイプ。ISO International Standard 8824 に記述されています。

BGP：Border Gateway Protocol。別々の自律システムに属するルータ間でのルーティング情報交換に使用する外部ボーダー ゲートウェイ プロトコル。BGP は TCP を使用します。TCP は信頼性の高いプロトコルであるため、BGP ではデータ パケットのドロップまたはフラグメント化の問題が発生しません。

BGP プレフィックス：BGP を使用したルート アナウンス。プレフィックスは、パケットが通過する必要があるネットワークを示す自律システム番号のパスと、ルーティングされる IP ブロックで構成されます。BGP プレフィックスは、701 1239 42 206.24.14.0/24 のようになります（/24 の部分は、CIDR マスクと呼ばれます）。/24 は、このブロックのネットマスクに左側から 1 が 24 個存在することを示します。/24 は、ナチュラル マスク 255.255.255.0 に対応します。

CE ルータ：カスタマー エッジ ルータ。VPN プロバイダーと VPN カスタマーの境界にあり、カスタマーに属するルータ。

CIDR：Classless Interdomain Routing（クラスレス ドメイン間ルーティング）。BGP4 でサポートされ、ルート集約に基づく技術。CIDR を使用すると、ルータはルートをグループ化して、コアルータによって伝送されるルーティング情報の量を減らすことができます。グループ外のネットワークからは、複数の IP ネットワークが 1 つの大きなエンティティに見えます。CIDR では、IP アドレスとそのサブネット マスクは、ピリオドで区切った 4 オクテットとして記述され、そのあとにスラッシュとサブネット マスクを表す 2 桁の数字が続きます。

Cisco Express Forwarding：高度なレイヤ 3 IP スイッチング テクノロジー。シスコ エクスプレス フォワーディングによって、大規模でダイナミックなトラフィック パターンを持つネットワークのパフォーマンスおよびスケーラビリティが最適化されます。

コミュニティ：SNMP における、同じ管理ドメイン内の管理対象デバイスと NMS の論理グループ。

コミュニティ名：「コミュニティ ストリング」を参照してください。

コミュニティ ストリング：パスワードとして機能し、管理対象ステーションと SNMP エージェントを含むルータとの間で送信されるメッセージの認証に使用されるテキスト文字列。コミュニティ ストリングは、マネージャとクライアント間のすべてのパケットで送信されます。コミュニティ名とも呼ばれます。

IETF：Internet Engineering Task Force。インターネットの規格を策定している 80 を超えるワーキング グループで構成される委員会。IETF は ISOC の下部組織です。「ISOC」も参照してください。

応答要求：従来のトラップ通知メッセージよりも信頼性が高い通知メッセージのタイプ。信頼性が高いのは、応答要求メッセージ通知には確認応答が必要ですが、トラップ通知には必要ないためです。

ISOC：インターネット ソサエティ。1992 年に設立された国際的な非営利団体。インターネットの発展と利用の整備を行っています。さらに、ISOC は、IAB などの他のインターネット関連団体に権限を委任しています。ISOC の本部は米国バージニア州レストンにあります。

ラベル：スイッチング ノードに対してデータの転送方法（パケットまたはセル）を指示する短い固定長のデータ構造。

LDP：Label Distribution Protocol（ラベル配布プロトコル）。パケットの転送に使用されるラベル（アドレス）のネゴシエーションで使用される MPLS 対応ルータ間の標準プロトコル。

LFIB：Label Forwarding Information Base（ラベル転送情報ベース）。シスコのラベル スイッチング システムにおける、着信タグと送信タグ（ラベル）、およびラベル付けに適した関連する対応パケットに関する情報を保存するためのデータ構造。

LSR：Label Switch Router（ラベル スイッチ ルータ）。各パケット内にカプセル化されている固定長ラベルの値に基づいて MPLS パケットを転送するデバイス。

MIB：Management Information Base（管理情報ベース）。SNMP や CMIP などのネットワーク管理プロトコルにより使用および管理されるネットワーク管理情報のデータベース。MIB オブジェクトの値は、SNMP コマンドまたは CMIP コマンドを使用して変更および取得できます。これらのコマンドは通常、GUI のネットワーク管理システムから実行します。MIB オブジェクトはツリー構造であり、ツリーにはパブリック（標準）ブランチとプライベート（独自）ブランチを含みます。

MPLS：Multiprotocol Label Switching（マルチプロトコル ラベル スイッチング）。ネットワークを介してパケット（フレーム）を転送する方式。ネットワークのエッジにあるルータがラベルをパケット（フレーム）に適用できるようにします。ネットワーク コア内の ATM スイッチまたは既存のルータは、最小限のルックアップ オーバーヘッドでラベルに従ってパケットを切り替えることができます。

MPLS インターフェイス：MPLS トラフィックが有効になっているインターフェイス。

MPLS VPN：Multiprotocol Label Switching（マルチプロトコル ラベル スイッチング）Virtual Private Network（バーチャル プライベート ネットワーク）。レイヤ 3 バックボーンを使用して、パブリック インフラストラクチャを介してプライベート ネットワーク サービスを提供する、IP ネットワーク インフラストラクチャ。Cisco IOS ネットワークで MPLS VPN を使用すると、スケーラブルなレイヤ 3 VPN バックボーン サービス（アプリケーション、データ ホスティング ネットワーク コマース、テレフォニー サービスなど）を展開および管理する機能をビジネス上のカスタマーに提供できます。

MPLS VPN ソリューションでは、MPLS VPN は、共通の「バックボーン」ネットワークによって接続された一連のプロバイダー エッジ ルータであり、2 つ以上のカスタマー サイト間のプライベート IP 相互接続を特定のカスタマーに提供します。各 VPN には一連のプロビジョニング テンプレートとポリシーがあり、VPN は複数のプロバイダー管理ドメイン（PAD）にまたがることができます。

NMS：Network Management System（ネットワーク管理システム）。ネットワーク管理者がネットワーク上の他のデバイスと通信するために使用する、高性能なコンピュータ（通常は、エンジニアリング ワークステーション）。NMS は、通常、ネットワーク リソースの管理、統計情報の収集、およびさまざまなネットワーク管理および設定タスクの実行に使用されます。

通知：SNMP エージェントによってネットワーク管理ステーション、コンソール、または端末に送信されるメッセージ。これにより、Cisco IOS ソフトウェア内で重大なイベントが発生したことが示されます。「トラップ」も参照してください。

PE ルータ：プロバイダー エッジ ルータ。VPN プロバイダーと VPN カスタマーの境界にあり、プロバイダーに属するルータ。

QoS：Quality of Service。転送システムのパフォーマンスの尺度の 1 つであり、転送品質とサービスのアベイラビリティを反映したものです。

RIB：Routing Information Base（ルーティング情報ベース）。ルーティング テーブルとも呼ばれます。

RT：Route Target（ルート ターゲット）。ルータのグループ、およびそのグループの各ルータにある転送テーブルのサブセットを識別する拡張コミュニティ属性。転送テーブルは、ルータによって保持され、その拡張コミュニティ属性を伝送する BGP ルートを格納できます。RT は 64 ビット値で、Cisco IOS ソフトウェアは VRF でのルート更新のためにこの値を使用してルートを区別します。

SNMP：Simple Network Management Protocol（シンプル ネットワーク管理プロトコル）。TCP/IP ネットワークでほぼ独占的に使用されているネットワーク管理プロトコル。SNMP を使用すると、ネットワーク デバイスのモニタリングと制御、および設定、統計情報収集、パフォーマンス、セキュリティの管理が可能になります。「SNMP2」も参照してください。

SNMP2：SNMP バージョン 2。一般的なネットワーク管理プロトコルのバージョン 2。SNMP2 では、集中型および分散型のネットワーク管理方式がサポートされ、管理情報構造（SMI）、プロトコル動作、管理アーキテクチャ、およびセキュリティが改善されています。「SNMP」も参照してください。

トラップ：SNMP エージェントによってネットワーク管理ステーション、コンソール、または端末に送信されるメッセージ。これにより、重大なイベントが発生したことが示されます。受信者はトラップの受信時に確認応答を送信しないため、トラップ（通知）は応答要求よりも信頼性が低くなります。送信側は、トラップが受信されたかどうかを判断できません。「通知」も参照してください。

VPN：Virtual Private Network（バーチャル プライベート ネットワーク）。管理ポリシー セットにより、共有バックボーン ネットワーク上で相互に通信可能なサイトのグループ。VPN は、1 つまたは複数の物理ネットワークでリソースを共有するセキュアな IP ベースのネットワークです。VPN には、共有のバックボーンで安全に通信できる地理的に分散したサイトが含まれます。「MPLS VPN」も参照してください。

VPN ID：RFC 2685 に基づいて VPN を識別するメカニズム。VPN ID は、組織固有識別子（OUI）、IEEE Registration Authority によって割り当てられた 3 オクテットの 16 進数、および会社内で VPN を識別する 4 オクテットの 16 進数である VPN インデックスで構成されます。

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

第 9 章

Pseudowire Emulation Edge-to-Edge MIB

イーサネットサービス、フレームリレーサービス、およびATMサービス用Pseudowire Emulation Edge-to-Edge MIB機能は、パケット交換網（PSN）上のイーサネットサービス、フレームリレーサービス、およびATMサービスをエミュレートするAny Transport over Multiprotocol Label Switching（AToM）インフラストラクチャ内で簡易ネットワーク管理プロトコル（SNMP）をサポートします。Pseudowire Emulation Edge-to-Edge（PWE3）MIBには、次のものがあります。

- CISCO-IETF-PW-MIB（PW-MIB）
- CISCO-IETF-PW-MPLS-MIB（PW-MPLS-MIB）
- CISCO-IETF-PW-ENET-MIB（PW-ENET-MIB）
- CISCO-IETF-PW-FR-MIB（PW-FR-MIB）
- CISCO-IETF-PW-ATM-MIB（PW-ATM-MIB）

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

Pseudowire Emulation Edge-to-Edge MIB の前提条件

- SNMP が、ラベル スイッチング ルータ（LSR）でイネーブルになっている必要があります。
- MPLS が LSR でイネーブルになっている必要があります。
- 疑似回線が、イーサネット、フレームリレー、または ATM アクセス回線を使用して設定されている必要があります。詳細については、『Any Transport over MPLS』モジュールを参照してください。

Pseudowire Emulation Edge-to-Edge MIB の制約事項

PWE3 MIB では、MIB オブジェクトに対するアクセス権が読み取り専用（RO）に制限されます。ただし、cpwVcUp と cpwVcDown の通知がイネーブルなオブジェクトである cpwVcUpDownNotifEnable は除きます。このオブジェクトは、SNMP エージェントによって書き込み可能に拡張されています。

- PW-MIB の次のテーブルは、サポートされていません。
  - cpwVcPerfCurrentTable
  - cpwVcPerfIntervalTable
- PW-MPLS-MIB の次のオブジェクトは、サポートされていません。
  - cpwVcMplsOutboundIndexNext
  - cpwVcMplsInboundIndexNext
- PW-ENET-MIB の次のテーブルは、サポートされていません。
  - cpwVcEnetMplsPriMappingTable
  - cpwVcEnetStatsTable

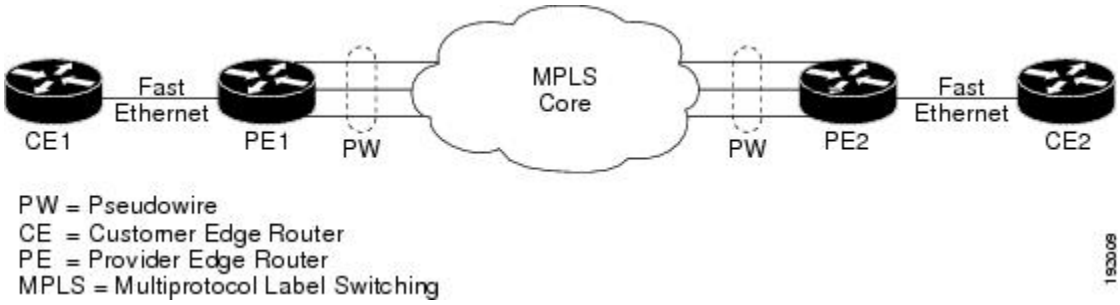
- PW-FR-MIB の次のテーブルは、サポートされていません。
  - cpwVcFrPMTable
- PW-ATM-MIB では、仮想パス（VP）ごとの大容量セル カウンタ、またはポートごとのセルの大容量セル カウンタはサポートされていません。
- ポート モードのセル リレーでの PW-ATM-MIB 仮想パス識別子（VPI）/仮想チャネル識別子（VCI）の値は 0 です。
- PW-ATM-MIB VP セルリレー VCI 値は 0 です。
- PW-ATM-MIB VP では、ATM アダプテーション層 5（AAL5）はサポートされていません。このため、すべてのパケット カウンタは無効です。

（注） この機能は、すべてのリリースのイーサネット、フレームリレー、および ATM でサポートされているわけではありません。詳細については、「イーサネット サービス、フレームリレー サービス、および ATM サービス用 Pseudowire Emulation Edge-to-Edge MIB の機能情報」を参照してください。

Pseudowire Emulation Edge-to-Edge MIB について

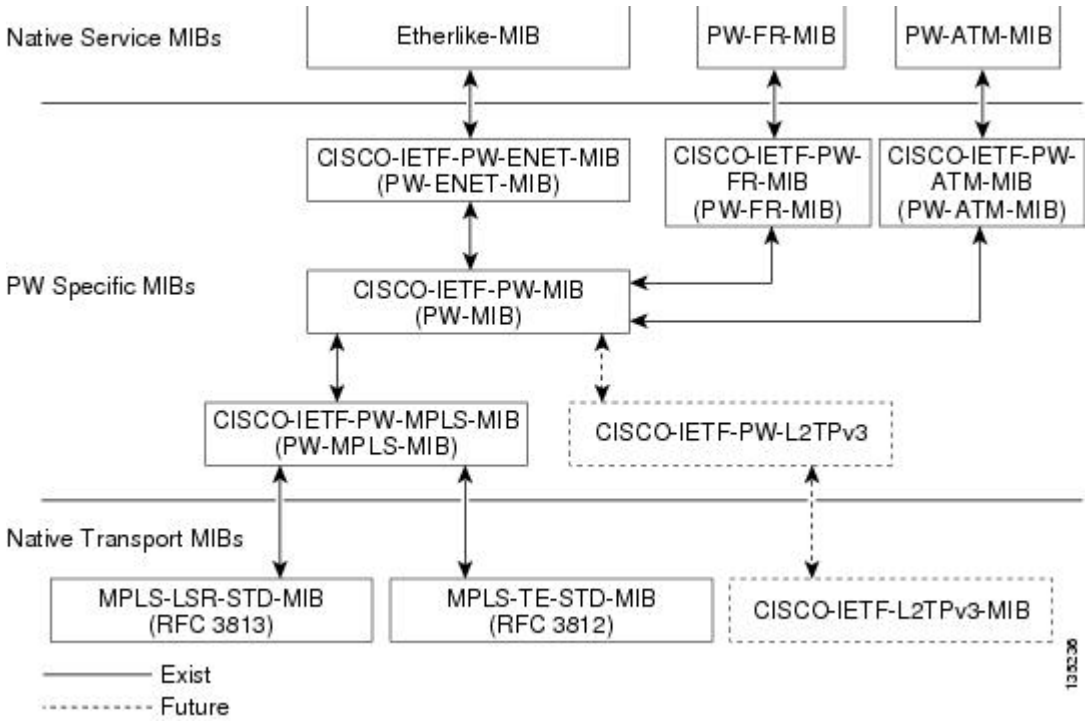
PWE3 MIB の擬似回線の機能

疑似回線は、以下の図に示すように、プロバイダーエッジ（PE）ルータのペアの間におけるポイントツーポイント接続です。その主な機能は、共通 MPLS 形式にカプセル化することによって、基礎となるコア MPLS ネットワーク経由でイーサネットなどのサービスをエミュレートすることです。共通 MPLS 形式へのサービスのカプセル化によって、疑似回線では、通信事業者は MPLS ネットワークにサービスを統合できます。



PWE3 MIB アーキテクチャ

以下の図に示している PWE3 MIB アーキテクチャは、3 つの MIB グループに分類されています。これらを一緒に使用すると、完全にエミュレートされたサービス、サービスをコア ネットワーク上で伝送するネイティブ転送、および 2 者間の関係が提供されます。



このアーキテクチャは、モジュラ式となっています。つまり、展開したあとには、新しくエミュレートされたサービス MIB モジュールまたは追加の転送 MIB モジュールを既存のインフラストラクチャに「プラグイン」するか、または既存のインフラストラクチャを拡張します。新しい固有のインフラストラクチャは必要ありません。したがって、完全に異なる管理戦略を導入する必要がある新しいサービスについて懸念することなく、管理アプリケーションを作成できます。このアーキテクチャは、既存のサービスモジュールと転送MIBモジュール間の汎用的なアソシエーション メカニズムであるため、ネイティブ MIB モジュールは、関連付けられた PWE3 固有の MIB がなくても機能します。このことには、サービスまたは転送を疑似回線に関連付ける Cisco ソフトウェアで、PWE3 固有の MIB がまだ展開されていない場合も、これらの MIB モジュールを照会できるというメリットがあります。ただし、疑似回線とのアソシエーションが存在しないという唯一の短所があります。

PWE3 MIB のコンポーネントおよび機能

PWE3 MIB には、次のコンポーネントおよび機能があります。

- PW-MIB（疑似回線 MIB）

この MIB では、PW-MPLS-MIB および PW-ENET-MIB を一緒にバインドし、疑似回線エミュレーションのステータス、および統計情報と設定情報を提供します。PW-MIB では、疑似回線の障害およびイベントの監視に関する通知についても定義します。

- PW-MPLS-MIB（疑似回線 MPLS-MIB）

この MIB には、ネットワーク マネージャが、疑似回線エミュレーション MPLS サービス（MPLS-トラフィック エンジニアリング（TE）-PSN および MPLS-non-TE-PSN など）の監視に使用できる管理対象オブジェクトが含まれています。

この MIB には、次の内容が表示されます。

-
  - ラベル配布プロトコル（LDP）シグナリングが行われ、MPLS TE トンネルに設定されていない優先パスを持つ仮想回線（VC）用の Cross-Connect（XC）インデックス。
  - TE トンネルに設定された優先パス、および TE トンネルである出力インターフェイスを持つ VC 用のトンネル インデックス。

- PW-ENET-MIB（疑似回線イーサネット サービス MIB）

この MIB には、ネットワーク マネージャが、疑似回線エミュレーション イーサネット サービスの監視に使用できる管理対象オブジェクトが含まれています。

- PW-FR-MIB（疑似回線フレームリレー サービス MIB）

この MIB には、ネットワーク マネージャが、疑似回線エミュレーション フレームリレー サービスの監視に使用できる管理対象オブジェクトが含まれています。

この MIB は、2 つのセグメント（フレーム リレー セグメントと疑似回線セグメント）で構成されている Frame Relay over Pseudowire（ATMoPW）接続を使用します。PW-FR-MIB によって、これらのセグメントにフックが提供されます。PW MIB には疑似回線セグメントに関する情報が含まれ、PW-FR-MIB にはフレームリレー セグメントに関する情報が含まれています。

PW-FR-MIB は、疑似回線サービスエミュレーション レイヤで定義され、上記の図に示すように、汎用 PW-MIB の最上位に位置しています。したがって、PW-FR-MIB は、PW-MIB の存在および PW-MIB によって提供されるサービスに大きく依存します。また、既存の PW-FR 接続エントリは、PW-MIB の既存の VC エントリに関連付けられています。

PW-FR-MIB および汎用 PW-MIB は、PW VC インデックス（PW-MIB をサポートするために定義されている内部インデックス）によって論理的に結合されています。各 PW VC インデックスは、PW-MIB および PW-FR-MIB の既存の VC エントリに一意にマッピングされます。

- PW-ATM-MIB（疑似回線 ATM サービス MIB）

この MIB には、ネットワーク マネージャが、疑似回線エミュレーション ATM サービスの監視に使用できる管理対象オブジェクトが含まれています。

この MIB では、2 つのセグメント（ATM セグメントと疑似回線セグメント）で構成されている ATM over Pseudowire（ATMoPW）接続を使用します。PW-ATM-MIB によって、これらのセグメントにフックが提供されます。PW MIB には疑似回線セグメントに関する情報が含まれ、PW-ATM-MIB には接続回線と呼ばれる ATM セグメントに関する情報が含まれています。

PW-ATM-MIB は、疑似回線サービス エミュレーション レイヤで定義され、上記の図に示すように、汎用 PW-MIB の最上位に位置しています。したがって、PW-ATM-MIB は、PW-MIB の存在および PW-MIB によって提供されるサービスに大きく依存します。また、既存の PW-ATM 接続エントリは、PW-MIB の既存の VC エントリに関連付けられています。

PW-ATM-MIB および汎用 PW-MIB は、PW VC インデックス（PW-MIB をサポートするために定義されている内部インデックス）によって論理的に結合されています。各 PW VC インデックスは、PW-MIB および PW-ATM-MIB の既存の VC エントリに一意にマッピングされます。

PW-MIB のテーブル

PW-MIB は、次のテーブルで構成されています。

- cpwVcTable：VC の作成に関する高レベルの汎用パラメータが含まれています。このテーブルは読み取り専用として実装され、特定の 1 つの接続を一意に識別する cpwVcIndex でインデックスが設定されます。このテーブル内の行は、エミュレートされた仮想接続を表します。このテーブルは、すべての VC タイプに使用されます。
- cpwVcPerfTotalTable：VC 開始時刻からの VC 単位のパフォーマンス情報を提供します。このテーブルは、cpwVcIndex でインデックスが設定されます。
- cpwVcIdMappingTable：VC タイプおよび VC ID 順序に基づいて、既存の VC の逆マッピングを提供します。このテーブルは、一般的に、既存の VC の要素管理ソフトウェア（EMS）順序照会に役立ちます。このテーブルは、cpwVcIdMappingVcType、cpwVcIdMappingVcID、cpwVcIdMappingPeerAddrType、および cpwVcIdMappingPeerAddr でインデックスが設定されます。このテーブルは、読み取り専用として実装されます。
- cpwVcPeerMappingTable：VC タイプおよび VC ID 順序に基づいて、既存の VC の逆マッピングを提供します。このテーブルは、一般的に、既存の VC の EMS 順序照会に役立ちます。このテーブルは、cpwVcPeerMappingPeerAddrType、cpwVcPeerMappingPeerAddr、cpwVcPeerMappingVcType、および cpwVcPeerMappingVcID でインデックスが設定されます。このテーブルは、読み取り専用として実装されます。

cpwVcTable

次の表に、cpwVcTable オブジェクトおよびその説明を示します。

表 *48*：*cpwVcTable* オブジェクトおよび説明

オブジェクト	説明
cpwVcType	VC 上を伝送されるサービスを示します。これは、回線タイプの情報です。

オブジェクト	説明
cpwVcOwner	オペレータによって設定され、この VC を確立するプロトコルを示します。値は次のとおりです。 • manual(1)：VC ラベルを含む VC テーブルのエントリの設定など、VC の設定にメンテナンスプロトコル（PW シグナリング）が必要ない場合に使用されます。 • maintenanceProtocol(2)：特定の PSN の VC の標準シグナリングに使用されます。たとえば、draft-martini-l2circuit-trans-mpls で指定された MPLS PSN の LDP、またはレイヤ 2 トンネリング プロトコル（L2TP）です。 • other(3)：その他のすべてのシグナリングタイプに使用されます。
cpwVcPsnType	オペレータによって設定され、VC が伝送される PSN タイプを示します。このオブジェクトに基づいて、関連する PSN テーブル エントリが PSN 固有の MIB モジュールに作成されます。たとえば、mpls(1) が定義されている場合、エージェントによってエントリが cpwVcMplsTable に作成され、これにより、MPLS PSN 設定がさらに定義されます。
cpwVcSetUpPriority	VC の相対的セットアップ プライオリティを最低から最高までの方式で定義します。0 が最高のプライオリティです。この値は、VC とこの機能をサポートする実装との間に競合リソースが存在する場合に、重要となります。これは、AToM では実装されないため、値 0 が使用されます。
cpwVcHoldingPriority	VC の相対的保留プライオリティを最低から最高までの方式で定義します。0 が最高のプライオリティです。この値は、VC とこの機能をサポートする実装との間に競合リソースが存在する場合に、重要となります。これは、AToM では実装されないため、値 0 が使用されます。

オブジェクト	説明
cpwVcInboundMode	プラットフォーム単位の VC ラベル スペースを使用する実装のセキュリティを高めることができます。モードは次のとおりです。 • strict(1) • loose(2) 厳格モードにおいて、PSN から着信するパケットは、MPLS では、着信トンネルテーブルを介して同じ VC に関連付けられているトンネルからの場合にだけ受け入れられ、L2TP または IP PSN では、送信元 IP アドレスによって識別されるトンネルからの場合にだけ受け入れられます。着信トンネル テーブル内のエントリは、VC セットアップに使用されるメンテナンス プロトコルによって明示的に設定されているか、または暗黙的に認識されます。 このようなアソシエーションが、認識されないか、設定されていないか、または想定されない場合は、緩和モードを設定する必要があります。また、ノードは、VC の伝送に使用される外部トンネルに関係なく、VC ラベルにだけ基づいてパケットを受け入れる必要があります。
cpwVcPeerAddrType	PW メンテナンス プロトコルが VC の作成に使用される場合に、ピア ノード メンテナンス プロトコル（シグナリング）アドレスのアドレスタイプを示します。PW メンテナンス プロトコルが使用されない場合、たとえば、cpwVcOwner が manual に設定されている場合は、unknown に設定する必要があります。
cpwVcPeerAddr	PW メンテナンスプロトコルエンティティのピア ノード アドレスの値が含まれています。このオブジェクトでは、無関係である場合（VC の手動設定）は値 0 が含まれます。
cpwVcID	L2TP の PW ID 属性値（AV）ペア、または LDP シグナリングの VC Forward Equivalence Class（FEC）要素内の出力 VC ID フィールドで使用します。

オブジェクト	説明
cpwVcLocalGroupID	VC セットアップのメンテナンス プロトコル内のピア PW に送信されるグループ ID フィールドで使用されます。使用されない場合は 0 となります。
cpwVcControlWord	ローカル ノードによって各パケットとともに制御ワードを送信するかどうかを定義します。
cpwVcLocalIfMtu	0 でない場合、メンテナンス プロトコルの省略可能な IfMtu オブジェクトは、値とともに送信され、VC に関連付けられているインターフェイス（または仮想インターフェイス）でのローカルでサポートされる最大伝送単位（MTU）サイズを表します。
cpwVcLocalIfString	各 VC は、サービス設定の一部として、ノードの ifTable 内のインターフェイス（または仮想インターフェイス）に関連付けられています。このオブジェクトでは、メンテナンスプロトコルが、メンテナンスプロトコルの一部として名前オブジェクトの ifTable に表示されるとおりにインターフェイスの名前を送信するかどうかを定義します。このオブジェクトが false に設定されている場合、オプションの要素は送信されません。
cpwVcRemoteGroupID	VC セットアップに使用したメンテナンス プロトコルを介して受信されたグループ ID フィールドから取得されます。使用されない場合は 0 となります。このオブジェクトが VC メンテナンス プロトコルで定義されていない場合、値 0xFFFF が使用されます。
cpwVcRemoteControlWord	VC の確立にメンテナンス プロトコルが使用される場合、このパラメータは、コントロール ワードを使用した受信のステータス、つまりパケットがコントロール ワードと共に受信されたかどうかを示します。メンテナンスプロトコルがリモート ノードからこのステータスをまだ受信していない間は、値 notYetKnown が使用されます。VC の手動設定では、このパラメータはローカル ノードに対し、受信パケットでカプセル化が予期されていることを示します。

オブジェクト	説明
cpwVcRemoteIfMtu	リモート ノードからメンテナンス プロトコルを介して受信したリモート インターフェイス MTU。このオブジェクトは、パラメータが使用できない場合、または使用されていない場合は、0 になります。
cpwVcRemoteIfString	メンテナンスプロトコルによって受信された、インターフェイスの説明ストリングを示します。適用できない場合、またはまだ認識されていない場合は、ヌル ストリングになります。
cpwVcOutboundVcLabel	PSN への発信方向で使用される VC ラベル。このオブジェクトは、オーナーが manual の場合は手動でセットアップされます。それ以外の場合は自動になります。例：MPLS PSN の場合、このラベルは VC タグの 20 ビットを表し、L2TP の場合、セッション ID の 32 ビットを表します。ラベルが認識されない場合（シグナリングが進行中）、このオブジェクトは 0xFFFF を返します。
cpwVcInboundVcLabel	PSN から受信するパケットについて、着信方向で使用される VC ラベル。このオブジェクトは、オーナーが manual の場合は手動でセットアップされます。それ以外の場合は自動になります。例：MPLS PSN の場合、このラベルは VC タグの 20 ビットを表し、L2TP の場合、このラベルはセッション ID の 32 ビットを表します。ラベルが認識されない場合（シグナリングが進行中）、このオブジェクトは 0xFFFF を返します。
cpwVcName	VC に割り当てられている標準名。
cpwVcDescr	VC に関する情報を含むテキスト文字列。説明がない場合、このオブジェクトには 0 長文字列が含まれます。
cpwVcCreateTime	VC が作成されたときのシステム時刻。
cpwVcUpTime	VC が両方の方向でアップ状態を継続している連続ティックの数（cpwVcOperStatus でアップ状態が読み取られます）。

オブジェクト	説明
cpwVcAdminStatus	VC の想定される動作ステータス。
cpwVcOperStatus	実際に組み合された VC の動作ステータスを示します。このオブジェクトは、cpwVcInboundOperStatus と cpwVcOutboundOperStatus の両方がアップ状態である場合に up になります。その他のすべての値については、両方向の VC が同じ値である場合、このオブジェクトにはその値が反映されます。そうでない場合は、2 つのうち、より重大なステータスに設定されます。重大度を最大から最小の順に並べると、unknown、notPresent、down、lowerLayerDown、dormant、testing、および up となります。オペレータは、障害分離を行うために、OperStatus の方向を調査できます。
cpwVcInboundOperStatus	着信方向での VC の実際の動作ステータスを示します。値は次のとおりです。 • up：VC は確立されており、パケットを渡す準備ができています。 • down：PW シグナリングがまだ終了していません。または、サービス レベルでは、VC がパケットを渡していないことを示します。 • testing：VC レベルでの AdminStatus が test に設定されています。 • dormant：VC は、プライオリティの高い VC によって必要なリソースが占有されているため、使用できません。 • notPresent：VC のセットアップに必要な一部のコンポーネントが欠落しています。 • lowerLayerDown：基礎となる PSN で、OperStatus が up になっていません。

オブジェクト	説明
cpwVcOutboundOperStatus	発信方向での VC の実際の動作ステータスを示します。値は次のとおりです。 • up：VC は確立されており、パケットを渡す準備ができています。 • down：PW シグナリングがまだ終了していません。または、サービス レベルでは、VC がパケットを渡していないことを示します。 • testing：VC レベルでの AdminStatus が test に設定されています。 • dormant：VC は、プライオリティの高い VC によって必要なリソースが占有されているため、使用できません。 • notPresent：VC のセットアップに必要な一部のコンポーネントが欠落しています。 • lowerLayerDown：基礎となる PSN で、OperStatus が up になっていません。
cpwVcTimeElapsed	未完了の秒数（現在の測定期間開始時点からの経過秒数）を含む秒数。システムの時刻クロックの調整などのような何らかの理由で、現在の間隔が最大値を超えると、エージェントは最大値を返します。cpwVcPerfIntervalTable は実装されていないため、0 となります。
cpwVcValidIntervals	すでに経過した、データが収集された 15 分間の間隔の数。PW 機能を備えたエージェントでは、*x* 以上の間隔をサポートできる必要があります。*x* の最小値は 4 です。*x* のデフォルトは 32 で、*x* の最大値は 96 です。この値は *x* ですが、測定が直前の *x* X 15 分以内に（再）開始された場合を除きます。この場合、値は完了した 15 分間隔の数になります。たとえば、エージェントがプロキシである場合は、一部の間隔を使用できないことがあります。この状況では、この間隔が、データが使用可能な最大間隔値になります。間隔が 0 に設定されます。
cpwVcRowStatus	常に active(1) である読み取り専用の実装。作成、変更、および削除に使用されます。

オブジェクト	説明
cpwVcStorageType	このオブジェクトのストレージタイプは、常に volatile(2) である読み取り専用の実装となります。

cpwVcPerfTotalTable

以下の表に、cpwVcPerfTotalTable オブジェクトおよびその説明を示します。

表 *49*：***cpwVcPerfTotalTable*** オブジェクトおよび説明

オブジェクト	説明
cpwVcPerfTotalInHCPackets	PSN から VC が受信するパケット数用の大容量カウンタ。
cpwVcPerfTotalInHCBytes	PSN から VC が受信するバイト数用の大容量カウンタ。
cpwVcPerfTotalOutHCPackets	VC が PSN に転送するパケット数用の大容量カウンタ。
cpwVcPerfTotalOutHCBytes	VC が PSN に転送するバイト数用の大容量カウンタ。
cpwVcPerfTotalDiscontinuityTime	このオブジェクトのカウンタが 1 つ以上中断した場合、最後に中断したときの sysUpTime の値。該当するカウンタは、Counter32 または Counter64 の特定のインスタンスです。ローカル管理サブシステムを最後に再初期化してから中断が発生しなかった場合、このオブジェクトには値 0 が格納されます。

cpwVcIdMappingTable

以下の表に、cpwVcIdMappingTable オブジェクトおよびその説明を示します。

表 *50*：***cpwVcIdMappingTable*** オブジェクトおよび説明

オブジェクト	説明
cpwVcIdMappingVcType	VC の VC タイプ（サービスを示す）。

オブジェクト	説明
cpwVcIdMappingVcID	VC の VC ID。VC が手動で設定されている場合は 0 になります。
cpwVcIdMappingPeerAddrType	ピア ノードの IP アドレス タイプ。
cpwVcIdMappingPeerAddr	ピア ノードの IP アドレス。
cpwVcIdMappingVcIndex	cpwVcTable の VC を表す値。

cpwVcPeerMappingTable

以下の表に、cpwVcPeerMappingTable オブジェクトおよびその説明を示します。

表 *51*：*cpwVcPeerMappingTable* オブジェクトおよび説明

オブジェクト	説明
cpwVcPeerMappingPeerAddrType	ピア ノードの IP アドレス タイプ。
cpwVcPeerMappingPeerAddr	ピア ノードの IP アドレス。
cpwVcPeerMappingVcType	VC の VC タイプ（サービスを示す）。
cpwVcPeerMappingVcID	VC の VC ID。VC が手動で設定されている場合は 0 になります。
cpwVcPeerMappingVcIndex	cpwVcTable の VC を表す値。

PW-MPLS-MIB のテーブル

PW-MPLS-MIB は、次のテーブルで構成されます。

- cpwVcMplsTable：MPLS PSN を介して伝送される VC の情報を指定します。このテーブルは、cpwVcIndex でインデックスが作成されます。
- cpwVcMplsOutboundTable：MPLS PSN を使用する VC を、PSN への発信 MPLS トンネルに関連付けます。また、VC だけの場合は、物理インターフェイスに関連付けます。このテーブルの行は、MPLS トンネルおよび PSN への MPLS トンネルを必要とする PW VC 間のリンクを表します。このテーブルは、cpwVcIndex およびサポートされていない追加のインデックスによってインデックスが作成されます。この結果、その値は 1 になります。オペレータは、MPLS PSN を必要とする各 PW VC 用に少なくとも 1 つのエントリをこのテーブルに作成します。VC だけの場合と cpwVcMplsOutboundIndex は、サポートされていません。

- cpwVcMplsInboundTable：MPLS PSN を使用する VC を、PSN から着信するパケットの着信 MPLS トンネルに関連付けます（このようなアソシエーションが主にセキュリティ上の理由から必要な場合）。このテーブルの行は、MPLS トンネルを必要とする PW VC と、PSN から到着するパケットの MPLS トンネルとの間のリンクを表します。このテーブルは、VC の識別に使用される一連のインデックス、cpwVcIndex、およびサポートされていない追加のインデックスによってインデックスが作成されます。この結果、その値は 1 になります。エントリは、ローカル エージェントによって自動的に、または厳格モードが必要な場合はオペレータによって手動でテーブルに作成されます。このテーブルは、適切な MPLS MIB をポイントします。MPLS-TE の場合、MPLS TE トンネルのインデックス作成に関連する 4 つの変数が設定されます。VC だけの場合と cpwVcMplsInboundIndex は、サポートされていません。
- cpwVcMplsNonTeMappingTable：着信または発信トンネルを非 TE アプリケーションの VC にマッピングします。このテーブルの行は、PW VC と MPLS-TE 以外の外部トンネルとの間のアソシエーションを表します。アプリケーションは、このテーブルを使用して、特定の非 TE MPLS 外部トンネルを介して伝送される PW を迅速に取得できます。このテーブルは、MPLS 非 TE トンネルの xconnect インデックス、および特定のエントリの VC の方向によってインデックスが作成されます。同じテーブルが、着信方向と発信方向の両方に使用されますが、それぞれの方向に使用される行は異なります。着信アソシエーションが認識されない場合、そのアソシエーションに関する行は存在しません。すべてのアソシエーションデータを表示できる場合は、行がローカル エージェントによって作成されます。
- cpwVcMplsTeMappingTable：着信または発信トンネルを MPLS-TE アプリケーションの VC にマッピングします。このテーブルの行は、PW VC と、その MPLS-TE 外部トンネルとの間のアソシエーションを表します。アプリケーションは、このテーブルを使用して、特定の TE MPLS 外部トンネルを介して伝送される PW を迅速に取得できます。このテーブルは、TE トンネル、VC 固有エントリの着信方向と発信方向、および VcIndex の 4 つによってインデックスが設定されます。同じテーブルが、着信方向と発信方向の両方に使用されますが、それぞれの方向に使用される行は異なります。着信アソシエーションが認識されない場合、そのアソシエーションに関する行は存在しません。すべてのアソシエーションデータを表示できる場合は、行がローカルエージェントによって作成されます。このテーブルは、疑似回線間のマッピングと、TE 以外の外部トンネルの xconnect インデックスまたはインデックスを示します。

cpwVcMplsTable

以下の表に、cpwVcMplsTable オブジェクトおよびその説明を示します。

表 *52*：***cpwVcMplsTable*** オブジェクトおよび説明

オブジェクト	説明
cpwVcMplsMplsType	オペレータによって設定され、外部トンネルタイプを示します（存在する場合）。値は次のとおりです。 • mplsTe(0)：外部トンネルが MPLS-TE によってセットアップされる場合に使用されます。 • mplsNonTe(1)：外部トンネルが LDP によって、または手動でセットアップされる場合に使用されます。
cpwVcMplsExpBitsMode	オペレータによって設定され、VC shim ラベル EXP ビットを決定する方法を示します。値は、次のとおりです。 • outerTunnel(1)：外部トンネルが存在し、cpwVcMplsMplsType が mplsTe または mplsNonTe である場合に使用されます。
cpwVcMplsExpBits	オペレータによって設定され、cpwVcMplsExpBitsMode が指定されている場合、VC shim ラベルで使用される MPLS EXP ビットを示します。値は 0 になります。
cpwVcMplsTtl	オペレータによって設定され、VC shim ラベルで使用される VC 存続可能時間（TTL）ビットを示します。値は 0 になります。
cpwVcMplsLocalLdpID	VC をローカル ノードに作成する LDP エンティティのローカル LDP ID。VC ラベルは、プラットフォームごとのラベル スペースから常に設定されるため、LDP ID の最後の 2 つのオクテットは、0 になります。
cpwVcMplsLocalLdpEntityID	ローカル ノード上の VC に使用される LDP エンティティのローカル LDP エンティティ インデックス。このオブジェクトが使用されない場合、すべて 0 に設定する必要があります。

オブジェクト	説明
cpwVcMplsPeerLdpID	LDP セッションによって識別されるピア LDP ID。これは、関係ない場合、またはまだ認識されていない場合はゼロになります。
cpwVcMplsStorageType	このオブジェクトのストレージタイプは、常に volatile(2) である読み取り専用の実装となります。

cpwVcMplsOutboundTable

以下の表に、cpwVcMplsOutboundTable オブジェクトおよびその説明を示します。

表 *53*：***cpwVcMplsOutboundTable*** オブジェクトおよび説明

オブジェクト	説明
cpwVcMplsOutboundIndex	テーブル内の VC ごとに複数の行を使用できるようにするための任意のインデックス。使用可能な次の空きインデックスは、cpwVcMplsOutboundIndexNext を使用して取得できます。このオブジェクトはサポートされていないため、値は 1 になります。
cpwVcMplsOutboundLsrXcIndex	オペレータによって設定されます。外部ラベルが MPL-LSR-MIB で定義されている場合、つまり、LDP によってまたは手動で設定されている場合、このオブジェクトは外部トンネルの xconnect インデックスをポイントします。それ以外の場合、このオブジェクトは 0 に設定されます。
cpwVcMplsOutboundTunnelIndex	発信トンネル（特に MPLS-TE 外部トンネル）用の一連のインデックスの一部。それ以外の場合、このオブジェクトは 0 に設定されます。
cpwVcMplsOutboundTunnelInstance	発信トンネル（特に MPLS-TE 外部トンネル）用の一連のインデックスの一部。それ以外の場合、このオブジェクトは 0 に設定されます。
cpwVcMplsOutboundTunnelLclLSR	発信トンネル（特に MPLS-TE 外部トンネル）用の一連のインデックスの一部。それ以外の場合、このオブジェクトはヌルに設定されます。

オブジェクト	説明
cpwVcMplsOutboundTunnelPeerLSR	発信トンネル（特に MPLS-TE 外部トンネル）用の一連のインデックスの一部。それ以外の場合、このオブジェクトはヌルに設定されます。
cpwVcMplsOutboundIfIndex	VC だけ（外部トンネルなし）の場合、このオブジェクトは、発信ポートの ifIndex を保有します。値は 0 です。
cpwVcMplsOutboundRowStatus	常に active(1) である読み取り専用の実装。作成、変更、および削除に使用されます。
cpwVcMplsOutboundStorageType	このオブジェクトのストレージタイプは、常に volatile(2) である読み取り専用の実装となります。

cpwVcMplsInboundTable

以下の表に、cpwVcMplsInboundTable オブジェクトおよびその説明を示します。

表 *54*：***cpwVcMplsInboundTable*** オブジェクトおよび説明

オブジェクト	説明
cpwVcMplsInboundIndex	テーブル内の VC ごとに複数の行を使用できるようにするための任意のインデックス。使用可能な次の空きインデックスは、cpwVcMplsInboundIndexNext を使用して取得できます。このオブジェクトはサポートされていないため、値は 1 になります。
cpwVcMplsInboundLsrXcIndex	外部ラベルが MPL-LSR-MIB で定義されている場合、つまり、LDP によってまたは手動で設定されている場合、このオブジェクトは外部トンネルの xconnect インデックスをポイントします。xconnect インデックスは、このオブジェクトの情報が認識されていない場合に 0 を取得する、着信方向の疑似回線を表します。
cpwVcMplsInboundTunnelIndex	着信トンネル、特に MPLS-TE 外部トンネルの一連のインデックスの一部。値は 0 になります。このオブジェクトは、入力ルータでの TE トンネルはサポートしません。

オブジェクト	説明
cpwVcMplsInboundTunnelInstance	着信トンネル、特に MPLS-TE 外部トンネルの一連のインデックスの一部。値は 0 になります。このオブジェクトは、入力ルータでの TE トンネルはサポートしません。
cpwVcMplsInboundTunnelLclLSR	着信トンネル（特に MPLS-TE 外部トンネル）用の一連のインデックスの一部。それ以外の場合は、ヌルに設定されます。このオブジェクトは、入力ルータでの TE トンネルはサポートしません。
cpwVcMplsInboundTunnelPeerLSR	着信トンネル（特に MPLS-TE 外部トンネル）用の一連のインデックスの一部。それ以外の場合、このオブジェクトはヌルに設定されます。このオブジェクトは、入力ルータでの TE トンネルはサポートしません。
cpwVcMplsInboundIfIndex	VC だけ（外部トンネルなし）の場合、このオブジェクトは、着信ポートの ifIndex を保有します。値は 0 です。
cpwVcMplsInboundRowStatus	常に active(1) である読み取り専用の実装。作成、変更、および削除に使用されます。
cpwVcMplsInboundStorageType	このオブジェクトのストレージタイプは、常に volatile(2) である読み取り専用の実装となります。

cpwVcMplsNonTeMappingTable

以下の表に、cpwVcMplsNonTeMappingTable オブジェクトおよびその説明を示します。

表 *55*：***cpwVcMplsNonTeMappingTable*** オブジェクトおよび説明

オブジェクト	説明
cpwVcMplsNonTeMappingTunnelDirection	行が、発信または着信マッピングを表すかどうかを識別します。
cpwVcMplsNonTeMappingXcTunnelIndex	疑似回線 LDP 生成 XC エントリの MPLS-LSR-MIB での XC インデックス。

オブジェクト	説明
cpwVcMplsNonTeMappingIfIndex	VC のためだけに VC が伝送されるポートを識別します。値は 0 です。
cpwVcMplsNonTeMappingVcIndex	cpwVcTable の VC を表します。

cpwVcMplsTeMappingTable

以下の表に、cpwVcMplsTeMappingTable オブジェクトおよびその説明を示します。

表 *56*：*cpwVcMplsTeMappingTable* オブジェクトおよび説明

オブジェクト	説明
cpwVcMplsTeMappingTunnelDirection	行が、発信マッピングを表すかどうかを識別します。
cpwVcMplsTeMappingTunnelIndex	MPLS-TE トンネルを識別する、概念行のインデックス。
cpwVcMplsTeMappingTunnelInstance	MPLS-TE トンネルのインスタンスを識別します。
cpwVcMplsTeMappingTunnelPeerLsrID	外部トンネルがMPLS-TEベースである場合に、ピア LSR を識別します。
cpwVcMplsTeMappingTunnelLocalLsrID	ローカル LSR を識別します。
cpwVcMplsTeMappingVcIndex	cpwVcTable の VC を表します。

PW-ENET-MIB のテーブル

PW-ENET-MIB は、次のテーブルで構成されます。

- cpwVcEnetTable：イーサネットがエミュレートされた各仮想接続のイーサネット ポートマッピングおよび VLAN 設定を提供します。このテーブルは、単一接続を一意に識別する cpwVcIndex でインデックスが作成されます。このテーブルの 2 番めのレベルのインデックスは、VC での VLAN を示す cpwVcEnetPwVlan です。このテーブルは、イーサネット VC タイプ（イーサネット VLAN、イーサネット、またはイーサネット仮想プライベート LAN サービス（VPLS））にのみ使用され、読み取り専用として実装されます。

cpwVcEnetTable

以下の表に、cpwVcEnetTable オブジェクトおよびその説明を示します。

表 *57*：*cpwVcEnetTable* オブジェクトおよび説明

オブジェクト	説明
cpwVcEnetPwVlan	VC でのフレームの VLAN 値。これは、テーブルのインデックスの 1 つであるため、複数の VLAN 値を PW VC に設定できます。この値は、非タグ付きフレームを示す場合、つまり、cpwVcEnetVlanMode 値が removeVlan の場合は 4096 です。この値は、cpwVcEnetVlanMode 値が noChange の場合は、アクセス回線の VLAN 値です。値 4097 は、このオブジェクトが適用できない、たとえば、イーサネットポートからのすべてのパケットを VC にマッピングする場合に使用されます。
cpwVcEnetVlanMode	アクセス回線と PW VC 間で VLAN フィールドが処理される方法を示します。このフィールドの有効値は、次のとおりです。 • noChange：cpwVcEnetPortVlan で指定しているように、VC に元のユーザ VLAN が含まれていることを示します。 • changeVlan：VC 上の VLAN フィールドが、ユーザのポート上の VLAN フィールドと異なる可能性があることを示します。 • removeVlan：VC でのカプセル化に、元の VLAN フィールドが含まれていないことを示します。

オブジェクト	説明
cpwVcEnetPortVlan	VC と、物理ポートまたは仮想ポートとの間の VLAN 値を変更する必要がある場合に、物理ポート（または VPLS 仮想ポート）での VLAN 値を定義します。cpwVcEnetVlanMode 値が noChange である場合は、cpwVcEnetPwVlan と同じになります。値 4096 は、VC に関連付けられている VLAN がないこと、つまり、非タグ付きフレームへのデフォルト VLAN の割り当てを示します。VC からのすべてのトラフィックがこのポートに転送される場合、この値は 4097 になり、関連がないことを示します。
cpwVcEnetPortIfIndex	ポイントツーポイント イーサネット サービス用 PW VC に関連付けられているイーサネットポートの ifIndex 値。VPLS の場合、この値は、VPLS インスタンスの仮想インターフェイスの ifIndex 値です。
cpwVcEnetVcIfIndex	ifTable での仮想インターフェイスとして VC をモデル化します。この値は、仮想インターフェイスが作成されていないことを示すために常に 0 になります。
cpwVcEnetRowStatus	常に active(1) である読み取り専用の実装。作成、変更、および削除に使用されます。
cpwVcEnetStorageType	このオブジェクトのストレージタイプは、常に volatile(2) である読み取り専用の実装となります。

PW-FR-MIB のテーブル

PW-FR-MIB は、次のテーブルで構成されます。

- cpwVcFrTable：フレームリレー VC と単一方向疑似回線ペア間で 1 対 1 の対応関係がある、1 対 1 のマッピング モードで動作する FRoPW 接続を表すエントリが含まれています。

cpwVcFrTable

以下の表に、cpwVcFrTable オブジェクトおよびその説明を示します。

表 *58*：*cpwVcFrTable* オブジェクトおよび説明

オブジェクト	説明
cpwVcFrIfIndex	FRoPW 接続のフレームリレー（FR）セグメントのインターフェイス ifIndex を返します。
cpwVcFrDlci	FRoPW接続のフレームリレーセグメントのデータリンク接続識別子（DLCI）を返します。
cpwVcFrAdminStatus	FRoPW 接続の管理ステータスを返します。
cpwVcFrOperStatus	FRoPW 接続の組み合された動作ステータスを返します。
cpwVcFrPw2FrOperStatus	FRoPW 接続の PW から FR への方向の動作ステータスを返します。
cpwVcFrRowStatus	常に active(1) である読み取り専用の実装。作成、変更、および削除に使用されます。
cpwVcFrStorageType	このオブジェクトのストレージタイプは、常に volatile(2) である読み取り専用の実装となります。

PW-ATM-MIB のテーブル

PW-ATM-MIB は、次のテーブルで構成されます。

- cpwVcAtmTable：PSN で伝送される ATM VC の情報を指定します。
- cpwVcAtmPerfTable：ATM VC のパフォーマンス関連の属性を指定します。

cpwVcAtmTable

以下の表に、cpwVcAtmTable オブジェクトおよびその説明を示します。

表 *59*：*cpwVcAtmTable* オブジェクトおよび説明

オブジェクト	説明
cpwAtmIf	ATM ネットワークとの間でセルを送受信する ATM インターフェイスを指定します。
cpwAtmVpi	ATM VC の VPI 値を指定します。

オブジェクト	説明
cpwAtmVci	ATM VC の VCI 値を指定します。
cpwAtmClpQosMapping	カプセル化プロトコルの Quality of Service（QoS）フィールド内の値を決定する、セル損失率優先度（CLP）ビットの存在を示します。この値は、発信トラフィック、つまり PSN に発信されるトラフィックにだけ使用できます。
cpwAtmRowStatus	常に active(1) である読み取り専用の実装。作成、変更、および削除に使用されます。
cpwAtmOamCellSupported	操作、管理、メンテナンス（OAM）セルが VC で転送されるかどうかを示します。
cpwAtmQosScalingFactor	PSN ドメインの QoS レートの計算時に、ATM QoS レートに適用される倍率を表します。
cpwAtmCellPacking	セル パッキングを実行するように VC が設定されているかどうかを識別します。
cpwAtmMncp	パッキングする必要があるセルの数を識別します。
cpwAtmEncap	MPLS またはレイヤ 2 トンネリング プロトコル バージョン 3（L2TPv3）が、転送として使用されるかどうかに関する情報を提供します。
cpwAtmPeerMncp	ピア インターフェイス用として 1 つのパケットにパッキングできる最大のセル数を表します。
cpwAtmMcptTimeout	使用される最大セル パッキング タイムアウト（MCPT）値を表します。

cpwVcAtmPerfTable

以下の表に、cpwVcAtmPerfTable オブジェクトおよびその説明を示します。

表 *60*：***cpwVcAtmPerfTable*** オブジェクトおよび説明

オブジェクト	説明
cpwAtmCellsReceived	PSN との間で送受信されたセルの数に関する情報を取得します。

オブジェクト	説明
cpwAtmCellsSent	ATM ネットワークに送信されたセルの数に関する情報を提供します。
cpwAtmCellsRejected	ポリシングが原因で、この VC によって拒否されたセルの数を示します。
cpwAtmCellsTagged	タグ付けされたセルの数を示します。
cpwAtmHCCellsReceived	VC によって受信されたセルの数用の大容量カウンタを提供します。
cpwAtmHCCellsRejected	VC によって拒否されたセルの数用の大容量カウンタを提供します。
cpwAtmHCCellsTagged	タグ付けされたセルの数用の大容量カウンタを提供します。
cpwAtmAvgCellsPacked	パッキングされた平均セル数を提供します。
cpwAtmPktsReceived	PW 上で AAL5 を実行するように VC が設定されている場合、ATM ネットワークに実際に送信された ATM AAL5 パケットの数を示します。
cpwAtmPktsSent	セルから再構築されたパケットの数を取得し、VC ラベルを割り当て、それらのパケットを PSN に送信します。
cpwAtmPktsRejected	ポリシングが原因で拒否されたパケットの数を示します。

PWE3 MIB のオブジェクト

PWE3 MIB が表す ASN.1 表記には、疑似回線サービスの具体的なコンポーネントが反映されます。これらの MIB によって、SNMP を使用するネットワーク管理アプリケーションは、この情報を取得して表示できます。これらの MIB では、標準の GETNEXT および GETBULK 機能がサポートされていますが、現在の実装での（SET による）設定機能はサポートしていません。

PWE3 MIB のスカラー オブジェクト

PWE3 MIB には、サポートされている次のスカラ オブジェクトが含まれています。

- cpwVcUpDownNotifEnable：このオブジェクトは、cpwVcUp 通知および cpwVcDown 通知の設定を反映します。いずれかの通知が、コマンドライン インターフェイス（CLI）を介して設定されている場合、このオブジェクトは、値 true(1) を持ちます。このオブジェクトが SNMP によって true(1) に設定されている場合、cpwVcUp 通知と cpwVcDown 通知の両方を発行できます。このオブジェクトが SNMP によって false(2) に設定されている場合、これらの通知は発行されません。

（注） cpwVcUpDownNotifEnable を設定できるのは、RW が **snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [**ipv6***nacl*] [*access-list-number*] コマンド用に設定されている場合のみです。

PWE3 MIB には、次のサポートされていないスカラー オブジェクトが含まれています。

- cpwVcIndexNext：行を cpwVcTable に追加した場合、次の cpwVcIndex 値が使用されることを示します。
- cpwVcNotifRate：デバイスから cpwVcUp/Down 通知を発行可能なレートを示します。
- cpwVcMplsOutboundIndexNext：cpwVcMplsOutboundTable にエントリを作成した場合、cpwVcMplsOutboundIndex に使用される適切な値を格納します。値 0 は、割り当てられていないエントリがいずれも使用できないことを示します。新しいエントリ用の cpwVcMplsOutboundIndex 値を取得するために、マネージャは、管理プロトコル取得操作を発行して、このオブジェクトの現在の値を取得します。各取得後に、ソフトウェア エージェントは、割り当てられていない次のインデックスに値を変更する必要があります。ただし、ソフトウェア エージェントは、作成された各行に対してこのような取得が行われることを想定していません。
- cpwVcMplsInboundIndexNextt：cpwVcMplsInboundTable にエントリを作成した場合、cpwVcMplsOutboundIndex に使用される適切な値を格納します。値 0 は、割り当てられていないエントリがいずれも使用できないことを示します。新しいエントリ用の cpwVcMplsInboundIndex 値を取得するために、マネージャは、管理プロトコル取得操作を発行して、このオブジェクトの現在の値を取得します。各取得後に、ソフトウェア エージェントは、割り当てられていない次のインデックスに値を変更する必要があります。ただし、ソフトウェア エージェントは、作成された各行に対してこのような取得が行われることを想定していません。

PWE3 MIB での通知

PW-MIB の cpwVcUp 通知と cpwVcDown 通知では、PW VC の範囲の operStatus 値の状態がいつ変更されたかを示します。

PW-MIB のこれらのオブジェクトの定義では、同じタイプ（アップまたはダウンのいずれか）のイベントは範囲に相互関連付けできる必要があることが示されています。これらの通知の実装では、この関連付けは実行されません。通知がイネーブルになっている場合、動作状態が変更された個々の VC 用に通知が生成されます。通知は、この MIB で説明しているように、VC のグループの動作状態変更をシグナリングしません。

PWE3 MIB の利点

PWE3 MIB は、イーサネット、フレームリレー、または ATM アクセス回線を監視するためのサービスおよびメカニズムに関する MPLS 関連情報を提供することで、Pseudowire Emulation Edge-to-Edge を管理する機能を提供します。

Pseudowire Emulation Edge-to-Edge MIB の設定方法

PWE3 MIB の SNMP エージェントのイネーブル化

手順の概要

1. **enable**
2. **show running-config** [**interface** | **map-class**]
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [**ipv6***nacl*] [*access-list-number*]
5. **end**
6. **write memory**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** [**interface** | **map-class**] 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 SNMP の情報が表示されない場合は、次のステップに進みます。 SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。 • オプションの **interface** キーワードを使用すると、インターフェイス固有の設定情報が表示されます。 • オプションの **map-class** キーワードを使用すると、ダイヤラまたはフレームリレーのマップクラス情報が表示されます。

	コマンドまたはアクション	目的
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [**ipv6***nacl*] [*access-list-number*] 例： `Router(config)# snmp-server community public ro`	MIB の SNMP へのアクセスを許可するようにコミュニティ アクセス ストリングを設定します。 • *string* 引数は、1 ～ 32 の英数字で構成されており、パスワードのように機能し、SNMP へのアクセスを許可します。コミュニティ ストリングに空白は使用できません。 • オプションの **view***view-name* キーワードと引数の組み合わせで、以前に定義されたビューを指定します。ビューは、SNMP コミュニティで使用できるオブジェクトを定義します。 • オプションの **ro** キーワードを使用すると、MIB のオブジェクトへの読み取り専用（ro）アクセスが設定されます。 • オプションの **rw** キーワードは、読み取り/書き込みアクセスであることを指定します。MIB オブジェクトの取得と変更の両方を実行できるのは、許可された管理ステーションです。 • オプションの **ipv6***nacl* キーワードと引数の組み合わせは、IPv6 名前付きアクセス リストを指定します。 • オプションの *access-list-number* 引数は、IP アドレスの標準アクセス リストを指定する 1 ～ 99 の整数、または SNMP エージェントへのアクセスを許可されている IP アドレスの標準アクセス リストの名前を示す文字列（64 文字以内）です。または、1300 ～ 1999 の整数で、コミュニティ ストリングを使用した SNMP エージェントへのアクセスが許可される、標準アクセス リスト番号の拡張範囲内の IP アドレスのリストを指定します。
ステップ 5	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**write memory** 例： `Router# write memory`	変更した SNMP 設定をルータの NVRAM に書き込み、SNMP 設定を永続的に保存します。

疑似回線クラスの設定

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

（注） 簡易設定では、この作業は任意です。**xconnect** コマンドの一部としてトンネリング方式を指定する場合は、疑似回線クラスを指定する必要はありません。

疑似回線クラスの設定グループでは、次のトンネリング メカニズム特性を指定します。

- カプセル化のタイプ
- 制御プロトコル
- ペイロード固有のオプション

AToM VC を正しく動作させるには、疑似回線クラスまたは **xconnect** コマンドの一部として **encapsulation mpls** コマンドを指定する必要があります。**xconnect** コマンドの一部として **encapsulation mpls** コマンドが指定されていないと、次のエラーが表示されます。

```
% Incomplete command.
```

いったん指定された **encapsulation mpls** コマンドは、**no encapsulation mpls** コマンドを使用して削除できません。また、**encapsulation l2tpv3** コマンドを使用しても、コマンドの設定は変更できません。このような方式では次のようなエラー メッセージが表示されます。

```
Encapsulation changes are not allowed on an existing pw-class.
```

このコマンドを削除するには、**no pseudowire-class** コマンドを使用して疑似回線を削除する必要があります。カプセル化のタイプを変更するには、**no pseudowire-class** コマンドで疑似回線を削除してから、疑似回線を再作成して新しいカプセル化タイプを指定します。

（注） 設定できるオプションは、多数あります。詳細については、『Any Transport over MPLS』モジュールを参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **pseudowire-class***name*
4. **encapsulation mpls**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**pseudowire-class***name* 例： `Router(config)# pseudowire-class atom`	指定した名前の疑似回線クラスを確立して、疑似回線クラス コンフィギュレーション モードに入ります。
ステップ 4	**encapsulation mpls** 例： `Router(config-pw)# encapsulation mpls`	トンネリング カプセル化を指定します。AToM の場合、カプセル化タイプは mpls です。

次の作業

cpwVcMIB、cpwVcMplsMIB、cpwVcEnetMIB、cpwVcFrMIB、および cpwVcAtmMIB で SNMP 管理ツールを使用して MIB ウォークを実行し、PW-MIB、PW-MPLS-MIB、PW-ENET-MIB、PW-FR-MIB、および PW-ATM-MIB の各オブジェクトにそれぞれ正しく入力されていることを確認します。

（注） SNMPv1 および SNMPv2c がサポートされています。

Pseudowire Emulation Edge-to-Edge MIB の設定例

PWE3 MIB：例

次の設定例では、SNMP マネージャはコミュニティ ストリング *public* を使用して、読み取り専用権限のあるすべてのオブジェクトにアクセスできます。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# snmp-server community public ro
```

（注） PWE3 MIB を設定する明示的な方法はありません。ただし、AToM の設定作業および例については、『Any Transport over MPLS』モジュールに記載されています。

PWE3 MIB に固有の通知があります。これらの設定に使用するコマンドの詳細については、「その他の参考資料」を参照してください。

その他の参考資料

関連資料

関連項目	マニュアル タイトル
MPLS および MPLS アプリケーションに関連するコマンドの説明	『*Multiprotocol Label Switching Command Reference*』
AToM および MPLS	「Any Transport over MPLS」モジュール

関連項目	マニュアル タイトル
疑似回線関連のインターネット ドラフト	• 『*An Architecture for Multi-Segment Pseudo Wire Emulation Edge-to-Edge*』、インターネット ドラフト、2007 年 12 月（draft-ietf-pwe3-ms-arch-03.txt） • 『Definitions for Textual Conventions and OBJECT-IDENTITIES for Pseudo-Wires Management』、インターネット ドラフト、2007 年 8 月 10 日（draft-ietf-pwe3-pw-tc-mib-09.txt） • 『Ethernet Pseudo Wire (PW) Management Information Base』、インターネット ドラフト、2007 年 8 月 30 日（draft-pwe3-enet-mib-10.txt） • 『*Managed Objects for ATM over Packet Switched Network (PSN)*』、インターネット ドラフト、2007 年 8 月 8 日（draft-ietf-pwe3-pw-atm-mib-02.txt） • 『Pseudo Wire (PW) Management Information Base』、インターネット ドラフト、2007 年 5 月 31 日（draft-ietf-pwe3-pw-mib-11.txt） • 『Pseudo Wire (PW) over MPLS PSN Management Information Base』、インターネット ドラフト、2007 年 8 月 11 日（draft-ietf-pwe3-pw-mpls-mib-11.txt） （注） SNMP MIB 機能の使用方法の詳細については、ご使用のネットワーク管理システムの適切なマニュアルを参照してください。

標準

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

MIB

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

RFC

RFC	タイトル
RFC 1156	『*Management Information Base for Network Management of TCP/IP-based Internets*』
RFC 1157	『*A Simple Network Management Protocol (SNMP)*』
RFC 1213	『*Management Information Base for Network Management of TCP/IP-based Internets: MIB-II*』
RFC 1315	『*Management Information Base for Frame Relay DTEs*』
RFC 3815	『*Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP)*』
RFC 3916	『*Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)*』
RFC 4619	『*Encapsulation Methods for Transport of Frame Relay over Multiprotocol Label Switching (MPLS) Networks*』

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

Pseudowire Emulation Edge-to-Edge MIB の機能情報

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

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

表 *61*：イーサネット サービス、フレーム リレー サービス、および *ATM* サービス用 ***Pseudowire Emulation Edge-to-Edge MIB*** の機能情報

機能名	リリース	機能情報
イーサネットサービス、フレームリレー サービス、および ATM サービス用 Pseudowire Emulation Edge-to-Edge MIB	Cisco IOS Release XE 2.3	イーサネットサービス、フレームリレー サービス、および ATM サービス用 Pseudowire Emulation Edge-to-Edge MIB 機能は、パケット交換網（PSN）上のイーサネット サービス、フレームリレー サービス、および ATM サービスをエミュレートする Any Transport over Multiprotocol Label Switching（AToM）インフラストラクチャ内で簡易ネットワーク管理プロトコル（SNMP）をサポートします。 Cisco IOS Release XE 2.3 で、この機能が Pseudowire Emulation Edge-to-Edge（PWE3）MIB として統合され、パケットスイッチドネットワーク（PSN）でのイーサネット、フレーム リレー、および ATM サービスをエミュレートする Any Transport over Multiprotocol Label Switching（AToM）インフラストラクチャ内で SNMP サポートを提供するようになりました。

用語集

AAL：ATM Adaptation Layer（ATM アダプテーション層）。AAL では、セルへのユーザ情報の変換を定義します。AAL1 および AAL2 では、音声やビデオなどのアイソクロナス トラフィックを処理します。AAL3/4 および AAL5 は、パケットのセグメンテーション リアセンブリによるデータ通信に関連します。

ATM：Asynchronous Transfer Mode（非同期転送モード）。セル ベースのデータ転送技術。チャネルの要求によってパケットの割り当てが決定されます。セルリ レーの国際標準であり、このモードでは、複数のサービスタイプ（音声、ビデオ、データなど）が、固定長（53バイト）のセルで

伝送されます。固定長セルの場合は、ハードウェアでセルを処理できるため、伝送遅延が短縮されます。ATM は、E3、SONET、T3 などの高速送信メディアを活用する設計になっています。

CE ルータ：カスタマー エッジ ルータ。カスタマー ネットワークに属し、プロバイダー エッジ（PE）ルータとのインターフェイスとなるルータ。

DLCI：Data-Link Connection Identifier（データリンク接続識別子）。フレームリレー ネットワーク内のPVCエンドポイントに割り当てられた固有の番号。フレームリレーネットワーク内のアクセス チャネルに含まれる特定のPVCエンドポイントを識別し、そのチャネルにのみ局所的に作用します。

カプセル化：特定のプロトコル ヘッダーにデータをラップすること。たとえば、イーサネット データは、ネットワークで送信される前に、特定のイーサネット ヘッダーでラップされます。また、異種ネットワークをブリッジングする場合は、一方のネットワークからのフレーム全体が、もう一方のネットワークのデータ リンク層プロトコルで使用されるヘッダーに単純に配置されます。

EoMPLS：Ethernet over Multiprotocol Label Switching（MPLS）。サービス プロバイダーが、カスタマー レイヤ 2 トラフィックをレイヤ 3 MPLS ネットワークを介してトンネリングできるようにするトンネリング メカニズムです。EoMPLS は、ポイントツーポイント ソリューションだけを提供します。EoMPLS は、レイヤ 2 トンネリングとも呼ばれています。

フレームリレー：業界標準のスイッチドデータ リンク層プロトコル。接続されたデバイス間で、高レベル データ リンク制御（HDLC）カプセル化を使用して複数の仮想回線を処理します。フレーム リレーは、一般的に置き代替可能と考えられているプロトコルである X.25 より効率的です。

IETF：Internet Engineering Task Force。インターネットおよび IP プロトコル スイートの標準を開発している、80 を超えるワーキング グループで構成される委員会です。

LDP：Label Distribution Protocol（ラベル配布プロトコル）。MPLS のホップバイホップ転送およびラベルとネットワーク プレフィックス間のバインディングの配布をサポートするプロトコル。このプロトコルのシスコ独自のバージョンは、タグ配布プロトコル（TDP）です。

LSP：Label Switched Path（ラベル スイッチド パス）。ラベル スイッチング技術がパケット転送に使用される、2 つのラベル スイッチング ルータ（LSR）間の設定済み接続。このパスは、MPLS ネットワークを介した特定のパスでもあります。

LSR：Label Switch Router（ラベル スイッチ ルータ）。ネイティブなレイヤ 3 パケットを転送できるマルチプロトコル ラベル スイッチング（MPLS）ノード。LSR は、パケットに付加されたラベルの値に基づいてパケットを転送します。

MIB：Management Information Base（管理情報ベース）。簡易ネットワーク管理プロトコル（SNMP）などのネットワーク管理プロトコルにより使用および管理される、ネットワーク管理情報のデータベース。MIB オブジェクトの値を変更または検索するには、通常はネットワーク管理システムを介して、SNMP コマンドを使用します。MIB オブジェクトはツリー構造であり、ツリーにはパブリック（標準）ブランチとプライベート（独自）ブランチを含みます。

MPLS：Multiprotocol Label Switching（マルチプロトコル ラベル スイッチング）。ラベルを使用して IP トラフィックを転送するスイッチング方式。このラベルによって、ネットワーク内のルータおよびスイッチが、事前に確立された IP ルーティング情報に基づくパケットの転送先を指示されます。

MTU：Maximum Transmission Unit（最大伝送ユニット）。特定のインターフェイスで処理できる最大パケット サイズ（バイト単位）。

NMS：Network Management System（ネットワーク管理システム）。ネットワークの少なくとも一部分の管理に責任を負うシステム。NMSは、一般的に適度にパワーのある装備の整ったコンピュータで、エンジニアリング ワークステーションなどです。NMSはエージェントと通信して、ネットワーク統計情報やリソースを追跡し続けるのに役立ちます。

通知：シンプル ネットワーク管理プロトコル（SNMP）エージェントがネットワーク管理ステーション、コンソール、または端末に送信する、重要なネットワーク イベントが発生したことを示すメッセージ。「トラップ」も参照してください。

OSPF：Open Shortest Path First。IS-IS プロトコルから派生した、リンクステート階層型の内部ゲートウェイ プロトコル ルーティング アルゴリズム。OSPF 機能には、最小コストによるルーティング、マルチパスのルーティング、およびロード バランシングが含まれます。

PE ルータ：プロバイダー エッジ ルータ。サービス プロバイダー ネットワーク内にあり、カスタマー エッジ（CE）ルータに接続されたルータ。

プライマリ トンネル：障害の発生時に、ラベル スイッチド パス（LSP）が高速リルートされるトンネル。バックアップ トンネルをプライマリ トンネルにすることはできません。

疑似回線：PW。パケット スイッチド ネットワーク（PSN）経由で 1 つのプロバイダー エッジから 1 つまたは複数の PE へ、エミュレートされたサービスの要素を伝送するメカニズム。

SNMP：Simple Network Management Protocol（シンプル ネットワーク管理プロトコル）。TCP/IP ネットワークでほぼ独占的に使用されている管理プロトコル。SNMP によって、ネットワーク デバイスを監視および制御し、設定、統計情報収集、パフォーマンス、およびセキュリティを管理する手段が提供されます。

トラップ：SNMP エージェントによってネットワーク管理ステーション、コンソール、または端末に送信されるメッセージ。これにより、重大なイベントが発生したことが示されます。トラップは応答要求より信頼性が低くなります。これは、トラップの受信時に、受信者が確認応答を送信しないためです。送信側は、トラップが受信されたかどうかを判断できません。

トンネル：2 つのピア間（ルータ間など）のセキュアな通信パス。

VC：Virtual Circuit（仮想回線）。2 つのネットワーク デバイス間に信頼性の高い通信を保証するために作成される論理回線。仮想回線は、相手先固定接続（PVC）または相手先選択接続（SVC）のいずれかになります。

第 10 章

MPLS トラフィック エンジニアリング - 高速リルート MIB

MPLS トラフィック エンジニアリング - 高速リルート MIB によって、Cisco ソフトウェアのマルチプロトコルラベルスイッチング（MPLS）高速リルート（FRR）機能の簡易ネットワーク管理プロトコル（SNMP）ベースのネットワーク管理を行うことができます。

高速リルート MIB には、次の機能があります。

- 通知を作成し、キューに入れることができる。
- コマンドラインインターフェイス（CLI）のコマンドを使用して、通知をイネーブルにし、通知の送信先 IP アドレスを指定できる。
- 通知の設定を不揮発性メモリに書き込むことができる。

MIB には、MPLS FRR 内の機能を説明するオブジェクトが含まれています。また、次のテーブルが含まれています。

- cmplsFrrConstTable
- cmplsFrrLogTable
- cmplsFrrFacRouteDBTable

また、MIB にはスカラ オブジェクト（つまり、テーブルに存在しないオブジェクト）も含まれています。詳細については、FRR MIB スカラー オブジェクト，（320 ページ）を参照してください。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS トラフィック エンジニアリング - 高速リルート MIB の前提条件

- ネットワークで、Intermediate System-to-Intermediate System（IS-IS）プロトコルまたは Open Shortest Path First（OSPF）プロトコルがサポートされている必要があります。
- SNMP が、ラベル スイッチ ルータ（LSR）にインストールされてイネーブルになっています。
- 各 LSR で MPLS がグローバルにイネーブルになっています。
- LSR でシスコ エクスプレス フォワーディングがイネーブルになっています。
- トラフィック エンジニアリング（TE）トンネルがイネーブルになっています。
- TE トンネルのいずれかで MPLS FRR がイネーブルになっています。
- リソース予約プロトコル（RSVP）がイネーブルになっています。

MPLS トラフィック エンジニアリング - 高速リルート MIB の制約事項

- FRR MIB の実装は、MIB オブジェクトの読み取り専用（RO）権限を持たないと行うことができません。

- 次のテーブルは実装されていません。
  - mplsFrrOne2OnePlrTable
  - mplsFrrDetourTable

MPLS トラフィック エンジニアリング - 高速リルート MIB に関する情報

MPLS トラフィック エンジニアリング - 高速リルート MIB の機能設計

FRR MIB を使用すると、Cisco ソフトウェアで FRR の標準的な SNMP ベースのネットワーク管理が可能になります。このためには、ネットワーク内の指定されたネットワーク管理ステーション（NMS）で SNMP エージェント コードが実行される必要があります。NMS は、MIB 内のネットワーク管理オブジェクトとユーザの対話の媒体となります。

FRR MIB は、インターネット技術特別調査委員会（IETF）ドラフト MIB 仕様 *draft-ietf-mpls-fastreroute-mib-02.txt* に基づいています。IETF ドラフト MIB は定期的な改訂を重ね、発展を遂げて標準になりつつあります。FRR MIB のシスコ実装は、IETF ドラフト MIB の発展に追随し、それに伴い変更される可能性があります。

IETF ドラフト MIB と Cisco ソフトウェア内の FRR はわずかに異なるため、FRR MIB オブジェクトと Cisco ソフトウェアの内部データ構造の間でいくつかの軽微な変換が必要となります。これらの変換は SNMP エージェントにより実行されます。SNMP エージェントは、NMS ワークステーション上で、ロー プライオリティのプロセスとしてバックグラウンドで実行され、Cisco ソフトウェアへの管理インターフェイスを提供します。

SNMP エージェントを使用すると、標準的な SNMP GET 操作を使用して FRR MIB オブジェクトにアクセスできます。FRR MIB 内のすべてのオブジェクトは、IETF ドラフト MIB で定義されている規定に準じます。

MPLS トラフィック エンジニアリング - 高速リルート MIB の機能構造

FRR MIB をサポートする SNMP エージェント コードは、Cisco ソフトウェア内のこのようなコードの既存モデルに準じます。また、その一部は、MIB ソース コードに基づいて Cisco ツール セットにより生成されます。生成されるコードの基礎となるのは、シスコ バージョンの FRR MIB CISCO-ietf-frr-mib です。

SNMP エージェント コードは、Cisco ソフトウェアの MIB サポート コードに共通の階層構造となっており、次のレイヤで構成されます。

- プラットフォームに依存しないレイヤ：このレイヤは、主に MIB 開発 Cisco ツール セットによって生成され、プラットフォームや実装に依存しない機能を統合します。これらの機能は、特定の MIB のコンテキストで、SNMP 標準機能を処理します。このレイヤは、GET、

GET-NEXT、および SET SNMP 操作に対してインデックスと範囲または列挙値のチェックを処理します。SNMP テーブルごと、またはオブジェクトのグループごとに、1 つの機能が生成されます。このレイヤが次のレイヤを呼び出します。

- アプリケーション インターフェイス レイヤ：Cisco ツール セットにより、MIB オブジェクトの機能名とテンプレート コードが生成されます。
- アプリケーション固有のレイヤ：このレイヤは、管理対象アプリケーション層から関連データを取得するためのメカニズムを提供します。このレイヤには、テーブルごとに 1 つのエントリ ポイント機能が含まれます。この機能は、他の 2 つの機能を呼び出します。インデックスに従って RSVP が関連データに対して保持する TE トンネル データベースを検索する機能と、データを構造に埋め込む機能です。
- 管理対象アプリケーション層：このレイヤにはすべての構造とメカニズムが含まれます。このレイヤは、MIB によって管理されます。

SNMP プロトコル要求および応答メッセージのシステム フロー

SNMP プロトコル要求および応答メッセージはいずれも、最終的には SNMP マスター エージェントによって処理されます。ルータ上でこのようなメッセージが受信されると、マスター エージェントは要求を解析し、要求の参照先の MIB を識別します。次にマスターエージェントは、GET、GET-NEXT、または SET 要求を使用して、MIB を担当するサブエージェントを照会します。FRR MIB サブエージェントは適切なデータを取得して、マスターエージェントにそれを返します。次にそのマスター エージェントが、NMS に SNMP 応答を返す役割を担います。すべての照会は IP SNMP Cisco ソフトウェア プロセス内で行われ、ロー プライオリティのタスクとして実行されます。

FRR MIB スカラー オブジェクト

スカラオブジェクトとは、テーブル内に存在しないオブジェクトのことです。各スカラオブジェクトは 1 つのインスタンス（つまり、1 つのオカレンス）を持ちます。

以下の表で、FRR MIB スカラー オブジェクトについて説明します。

表 *62*：スカラー オブジェクト

MIB オブジェクト	機能
cmplsFrrDetourIncoming	デバイスに入る迂回リンクステート パケット（LSP）の数。cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているため、このオブジェクトは 0 を返します。
cmplsFrrDetourOutgoing	デバイスを出る迂回 LSP の数。cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているため、このオブジェクトは 0 を返します。

MIB オブジェクト	機能
cmplsFrrDetourOriginating	デバイスから発信された迂回 LSP の数。cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているため、このオブジェクトは 0 を返します。
cmplsFrrSwitchover	cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているためにバックアップされるトンネルの数。
cmplsFrrNumOfConfIfs	FRR 保護が設定された MPLS インターフェイスの数。0 は、いずれのインターフェイスを通過する LSP も保護できることを示します。
cmplsFrrActProtectedIfs	cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているために FRR で保護されているインターフェイスの数。
cmplsFrrConfProtectingTuns	cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているために高速リルートで保護されるバックアップトンネルの数。
cmplsFrrActProtectedTuns	高速リルート機能により保護されるトンネルの数。cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されているため、このオブジェクトは 0 を返します。
cmplsFrrActProtectedLSPs	FRR で保護されている LSP の数。cmplsFrrConstProtectionMethod が facilityBackup(1) に設定されている場合、このオブジェクトは 0 を返します。
cmplsFrrConstProtectionMethod	Cisco ソフトウェアはファシリティ バックアップ保護方式だけをサポートしているため、このオブジェクトは常に facilityBackup(1) を返します。
cmplsFrrNotifsEnabled	この MIB 内に定義されている FRR 通知がイネーブルかディセーブルのどちらに設定されているかを示す値。イネーブルの場合、このオブジェクトは True(1) を返し、ディセーブルの場合は False(2) を返します。デフォルトでは、通知はディセーブルになっています。
cmplsFrrLogTableMaxEntries	FRR ログ テーブル内で許可される最大エントリ数。
cmplsFrrLogTableCurrEntries	FRR ログ テーブル内の現在のエントリ数。このオブジェクトは常に 0 を返します。
cmplsFrrNotifMaxRate	FRR MIB 通知の最大間隔レート。このオブジェクトは常に 0 を返します。

FRR MIB 通知の生成イベント

特定の FRR イベントが発生したあとは、通知が発行されます。

snmp-server enable traps mpls fast-reroute コマンドを発行して FRR MIB 通知機能をイネーブルにした場合、FRR イベントによって通知メッセージが生成され、ネットワーク内の指定された NMS に送信されてから、Cisco ソフトウェア内で特定のイベントが発生したことがシグナリングされます。

FRR ステータス移行とイベント通知に関連する FRR MIB オブジェクトには、cmplsFrrProtected が含まれます。大きな TE トンネル変更（つまり、TE トンネルの高速リルート）がある場合は、このメッセージが NMS に送信されます。

FRR MIB 通知の仕様

特定の FRR イベントが発生したあとは、通知が発行されます。

各 FRR 通知には、汎用タイプの ID と、通知タイプを識別するための企業固有タイプ ID が含まれます。すべての FRR 通知に対する汎用タイプは、SNMP に対して定義されている汎用通知タイプの 1 つではないため、「企業固有」です。cmplsFrrProtected の場合、企業固有タイプは 1 です。

各通知には、FRR トンネルを容易に識別できるように、FRR MIB からの次のオブジェクトが含まれています。

- cmplsFrrConstNumProtectingTunOnIf
- cmplsFrrConstNumProtectedTunOnIf
- cmplsFrrConstBandwidth

呼び出された時点では、既存の FRR コードによって、適切な FRR インターフェイス インデックスがすでに取得されています。その後、FRR インターフェイスを使用して、通知に含められる 3 つのオブジェクトのデータが入力されます。

FRR MIB 通知の監視

特定の FRR イベントが発生したあとは、通知が発行されます。

FRR MIB 通知が有効になっている場合（**snmp-server enable traps** コマンドを参照）、Cisco ソフトウェア内の特定の FRR イベントに関連する通知メッセージが生成されて、ネットワーク内の指定された NMS に送信されます。SNMPv1 または SNPv2 通知をサポートするユーティリティはいずれも、通知メッセージを受信できます。

FRR MIB 通知を監視するには、SNMP 通知を表示するユーティリティをサポートしている NMS にログインし、表示ユーティリティを起動します。

MPLS トラフィック エンジニアリング - 高速リルート MIB の MIB テーブル

FRR MIB は、次のテーブルで構成されます。

これらのテーブルは、さまざまなデータ構造にアクセスして、迂回、FRR データベース、およびロギングに関連する情報を取得します。

cmplsFrrConstTable

cmplsFrrConstTable は、FRR 対応のトンネルの設定と、それに付随するバックアップ トンネルの特性を表示します。保護対象のトンネルごとに、複数のバックアップ トンネルが存在することがあります。

次の機能によって、テーブルにインデックスが作成されます。

- cmplsFrrConstIfIndex
- cmplsFrrConstTunnelIndex
- cmplsFrrConstTunnelInstance

以下の表で、cmplsFrrConstTable の MIB オブジェクトについて説明します。

表 *63* : ***cmplsFrrConstTable*** オブジェクト

MIB オブジェクト	機能
cmplsFrrConstIfIndex	FRR が設定されているインターフェイスを一意に識別します。インデックスに 0 の値がある場合、FRR 機能を使用できるデバイス上のすべてのインターフェイスに設定が適用されます。
cmplsFrrConstTunnelIndex	FRR が要求されているトンネル。
cmplsFrrConstTunnelInstance	FRR が要求されているトンネル。トンネル ヘッドだけが表示され、トンネル ヘッドのインスタンスの値は常に 0 であるため、値は常に 0 となります。
cmplsFrrConstSetupPrio	バックアップ トンネルのセットアップ プライオリティ。
cmplsFrrConstHoldingPrio	バックアップ トンネルの保留プライオリティ。
cmplsFrrConstInclAnyAffinity	トンネルがリンクを通過するために設定する必要のある属性ビット。
cmplsFrrConstInclAllAffinity	トンネルがリンクを通過するために設定から除外する必要のある属性ビット。

MIB オブジェクト	機能
cmplsFrrConstExclAllAffinity	制約に指定されている管理グループのいずれも含まれていないリンクだけが、exclude-all 制約を満たします。
cmplsFrrConstHopLimit	バックアップ トンネルが通過できる最大ホップカウント。
cmplsFrrConstBandwidth	このトンネルのバックアップ トンネルの帯域幅。単位はキロ ビット/秒（kbps）です。
cmplsFrrConstRowStatus	このテーブル内の行を作成、変更、および削除します。

cmplsFrrLogTable

cmplsFrrLogTable は、オブジェクト cmplsFrrLogIndex によってインデックスが作成されます。インデックスは、FRR 機能の **showmplstraffic-engfast-reroutelogreroutes** コマンド内のログ エントリに対応しています。その **show** コマンドでは、一度に最大 32 個のエントリを保存できます。エントリが追加されると、一番古いエントリが新しいログ情報で上書きされます。

cmplsFrrLogTable は一度に最大 32 個のエントリを保存できます。新しいエントリが追加されると、古いエントリが上書きされます。インデックス cmplsFrrLogIndex は増加していくため、MIB のログテーブルエントリそれぞれに一意のインデックス値が付きます。したがって、エントリが 32 個しか表示されていなくても、32 個より多くのインデックスが存在している可能性があります。

以下の表で、cmplsFrrLogTable の MIB オブジェクトについて説明します。

表 *64*：***cmplsFrrLogTable*** オブジェクト

MIB オブジェクト	機能
cmplsFrrLogIndex	FRR イベントの番号。
cmplsFrrLogEventTime	ブートストラップ時刻からイベント発生時刻までの経過時間（ミリ秒）。
cmplsFrrLogInterface	この FRR イベントによる影響を受けたインターフェイスを識別します。mplsFrrConstProtectionMethod が oneToOneBackup(0) に設定されている場合、この値は 0 に設定できます。
cmplsFrrLogEventType	発生した FRR イベントのタイプ。このオブジェクトは、Protected または Other を返します。
cmplsFrrLogEventDuration	イベントの持続時間（ミリ秒）。

MIB オブジェクト	機能
cmplsFrrLogEventReasonString	実装固有のイベント説明。このオブジェクトは、interface down イベントまたは interface other イベントを返します。

cmplsFrrFacRouteDBTable

次のインデックスは、FRR 機能によって保護されるインターフェイスおよびトンネルを指定します。

- cmplsFrrFacRouteProtectedIfIndex
- cmplsFrrFacRouteProtectedTunIndex

次のインデックスは、保護対象のトンネルに保護を提供するバックアップ トンネルを指定します。

- cmplsFrrFacRouteProtectedIfIndex
- cmplsFrrFacRouteProtectingTunIndex
- cmplsFrrFacRouteProtectedTunIndex
- cmplsFrrFacRouteProtectedTunInstance
- cmplsFrrFacRouteProtectedTunIngressLSRId
- cmplsFrrFacRouteProtectedTunEgressLSRId

このバージョンの MIB は、MPLS TE MIB に対してすでに実行された作業を利用しようとします。これには TE トンネルに対する類似の参照機能が含まれているためです。

以下の表で、cmplsFrrFacRouteDBTable の MIB オブジェクトについて説明します。

表 *65*：***cmplsFrrFacRouteDBTable*** オブジェクト

MIB オブジェクト	機能
cmplsFrrFacRouteProtectedIfIndex	FRR 保護のために設定されたインターフェイス。
cmplsFrrFacRouteProtectingTunIndex	保護する側の（バックアップ）トンネルのトンネル番号。
cmplsFrrFacRouteProtectedTunIndex	mplsFrrFacRouteIfProtIdx で指定されているインターフェイス（およびこのインターフェイスを使用するすべてのトンネル）を保護するように指定されているトンネル ヘッド インターフェイスの mplsTunnelEntry プライマリ インデックス。
cmplsFrrFacRouteProtectedTunInstance	FRR で保護されている mplsTunnelEntry。トンネルを一意に識別するインスタンス。

MIB オブジェクト	機能
cmplsFrrFacRouteProtectedTunIngressLSRId	バックアップ LSR のインバウンド ラベル。
cmplsFrrFacRouteProtectedTunEgressLSRId	バックアップ LSR のアウトバウンド ラベル。
cmplsFrrFacRouteProtectedTunStatus	保護対象のトンネルのステート。有効な値は次のとおりです。 • active：トンネル ラベルはラベル転送情報ベース（LFIB）にすでに挿入されており、着信パケットに適用する準備ができています。 • ready：トンネルのラベルエントリはすでに作成されていますが、LFIB 内にありません。 • partial：トンネルのラベルエントリの作成が完了していません。
cmplsFrrFacRouteProtectingTunResvBw	バックアップ トンネルによって予約されている帯域幅の大きさ（秒当たりのメガバイト数）。
cmplsFrrFacRouteProtectingTunProtectionType	保護のタイプ：0 はリンク保護を指定し、1 はノード保護を指定します。

MPLS トラフィック エンジニアリング - 高速リルート MIB の設定方法

FRR MIB 通知に対する SNMP エージェントのイネーブル化

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro**] [*access-list-number*]
5. **snmp-server enable traps mpls fast-reroute protected**
6. **end**
7. **write memory**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 SNMP の情報が表示されない場合は、次のステップに進みます。 SNMP 情報が表示された場合は、情報を修正または変更できます。
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro**] [*access-list-number*] 例： `Router(config)# snmp-server community public ro`	FRR MIB に対して読み取り専用（RO）の SNMP コミュニティ ストリングを設定します。
ステップ 5	**snmp-server enable traps mpls fast-reroute protected** 例： `Router(config)# snmp-server enable traps mpls fast-reroute protected`	高速リルート トラップをイネーブルにします。
ステップ 6	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**write memory** 例： `Router# write memory`	変更した SNMP 設定をルータの NVRAM に書き込み、SNMP 設定を永続的に保存します。

シスコ エクスプレス フォワーディングのイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **ip cef distributed**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip cef distributed** 例： `Router(config)# ip cef distributed`	分散型シスコエクスプレスフォワーディングをイネーブルにします。
ステップ 4	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。

TE トンネルのイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **ip cef**
4. **mpls traffic-eng tunnels**
5. **interface***typeslot*/*subslot*/*port*[**.***subinterface*]
6. **mpls traffic-eng tunnels**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip cef** 例： `Router(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： `Router(config)# mpls traffic-eng tunnels`	デバイス上で MPLS TE トンネル機能をイネーブルにします。
ステップ 5	**interface***typeslot*/*subslot*/*port*[**.***subinterface*] 例： `Router(config)# interface POS1/0/0`	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。

	コマンドまたはアクション	目的
ステップ 6	**mpls traffic-eng tunnels** 例： `Router(config-if)# mpls traffic-eng tunnels`	インターフェイス上で MPLS TE トンネル機能をイネーブルにします。
ステップ 7	**end** 例： `Router(config-if)# end`	特権 EXEC モードに戻ります。

各 TE トンネルでの MPLS FRR のイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **interface***typeslot*/*subslot*/*port*[**.***subinterface*]
4. **tunnel mode mpls traffic-eng**
5. **tunnel mpls traffic-eng fast-reroute**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**interface***typeslot*/*subslot*/*port*[**.***subinterface*] 例： `Router(config)# interface POS1/0/0`	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	**tunnel mode mpls traffic-eng** 例： `Router(config-if)# tunnel mode mpls traffic-eng`	トンネルのモードを、トラフィック エンジニアリングの MPLS に設定します。
ステップ 5	**tunnel mpls traffic-eng fast-reroute** 例： `Router(config-if)# tunnel mpls traffic-eng fast-reroute`	保護対象の TE トンネル上で高速リルートをイネーブルにします。
ステップ 6	**end** 例： `Router(config-if)# end`	特権 EXEC モードに戻ります。

インターフェイスでのバックアップ トンネルのイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **interface***typeslot*/*subslot*/*port*[**.***subinterface*]
4. **mpls traffic-eng backup-path tunnel***interface*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • 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： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface***typeslot*/*subslot*/*port*[**.***subinterface*] 例： `Router(config)# interface POS1/0/0`	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	**mpls traffic-eng backup-path tunnel***interface* 例： `Router(config-if)# mpls traffic-eng backup-path tunnel1`	指定したインターフェイス上でバックアップ トンネルをイネーブルにします。
ステップ 5	**end** 例： `Router(config-if)# end`	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 - 高速リルート MIB の設定例

例：ホスト NMS での SNMP エージェントのイネーブル化

```
enable
 show running-config
 configure terminal
 snmp-server community public ro
 snmp-server enable traps mpls fast-reroute protected
```

```
 end
 write memory
```

例：シスコ エクスプレス フォワーディングのイネーブル化

```
enable
 configure terminal
 ip cef
 end
```

例：TE トンネルのイネーブル化

```
enable
 configure terminal
 ip cef
 mpls traffic-eng tunnels
 interface FastEthernet1/0/0
 mpls traffic-eng tunnels
 end
```

例：各 TE トンネルでの MPLS FRR のイネーブル化

```
enable
 configure terminal
 interface POS1/0/0
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng fast-reroute
 end
```

例：インターフェイスでのバックアップ トンネルのイネーブル化

```
enable
 configure terminal
 interface POS1/0/0
 mpls traffic-eng backup-path tunnel1
 end
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
MPLS および MPLS アプリケーションに関連するコマンドの説明	『*Multiprotocol Label Switching Command Reference*』

関連項目	マニュアル タイトル
MPLS トラフィック エンジニアリング MIB に対する SNMP エージェントのサポート	MPLS トラフィック エンジニアリング MIB
高速再ルーティング	『MPLS Traffic Engineering: Fast Reroute Link and Node Protection』

標準

規格	タイトル
MPLS-FRR-MIB	*draft-ietf-mpls-fastreroute-mib-02.txt*

MIB

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

RFC

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

シスコのテクニカル サポート

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

MPLS トラフィック エンジニアリング - 高速リルート MIB の機能情報

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

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

表 *66*：*MPLS* トラフィック エンジニアリング - 高速リルート *MIB* の機能情報

機能名	リリース	機能情報
MPLS Embedded Management--MPLS Fast Reroute MIB（IETF ドラフト v01）	Cisco IOS XE Release 2.3	高速リルート MIB は、マルチプロトコル ラベル スイッチング（MPLS）高速リルート（FRR）機能の SNMP ベースのネットワーク管理を提供します。

用語集

FEC：Forward Equivalence Class。転送のために同等に処理できるパケットのセット。したがって、1 つのラベルへのバインディングに適しています。たとえば、1 つのアドレス プレフィックス宛てのパケットのセットや任意のフローなどがあります。

フロー：一般に、一組のホスト間、または一組のホスト上にある一組のトランスポート プロトコル ポート間で転送されるパケットのセット。たとえば、同じ送信元アドレス、送信元ポート、宛先アドレス、および宛先ポートを持つパケットは、フローと見なされることがあります。

フローは、ネットワークの 2 つのエンドポイント間で（たとえば、ある LAN ステーションから別の LAN ステーションへ）転送されるデータのストリームでもあります。単一の回線上で複数のフローを転送できます。

フラグメンテーション：元のパケット サイズをサポートできないネットワーク メディアを介してパケットを送信するときに、パケットを小さい単位に分割するプロセス。

ICMP：Internet Control Message Protocol。エラーを報告し、IP パケット処理に関連するその他の情報を提供するネットワーク層インターネット プロトコル。RFC 792 に記載されています。

LFIB：Label Forwarding Information Base（ラベル転送情報ベース）。宛先および着信ラベルが発信インターフェイスおよびラベルに関連付けられている転送を管理するデータ構造および手段。

localhost：デバイスのホスト名を表す名前。localhost は、予約済みのループバック IP アドレス（127.0.0.1）を使用します。

LSP：Label Switched Path（ラベル スイッチド パス）。MPLS を使用してパケットを転送する 2 つのデバイス間の接続。

LSPV：Label Switched Path Verification。LSP ping サブプロセスであり、MPLS エコー要求とエコー応答を符号化および復号化し、MPLS エコー要求とエコー応答を送受信するために IP、MPLS、および AToM スイッチングとやり取りします。MPLS エコー要求発信元デバイスでは、対応するエコー応答が受信されていない未処理のエコー要求が格納されているデータベースを維持します。

MPLS ルータ アラート ラベル：MPLS ラベル 1。ルータ アラート ラベルを含む MPLS パケットは、処理のためにデバイスによって Route ルート プロセッサ（PR）の処理レベルにリダイレクトされます。これにより、これらのパケットはハードウェア ルーティング テーブルにおけるフォワーディング エラーを回避できます。

MRU：Maximum Receive Unit（最大受信ユニット）。LSP を介して転送できる、ラベル付きパケットの最大サイズ（バイト単位）。

MTU：Maximum Transmission Unit（最大伝送ユニット）。特定のインターフェイスで処理できる最大パケット サイズ（バイト単位）。

パント：ルータ アラートを含むパケットを処理のためにラインカードまたはインターフェイスからルート プロセッサ（RP）のレベル処理にリダイレクトします。

PW：pseudowire（疑似回線）。パケット スイッチド ネットワークを介して、エミュレートされた回線の重要な要素を、あるプロバイダー エッジ（PE）デバイスから別の PE デバイスに伝送するメカニズム。

RP：ルート プロセッサ。Cisco 7000 シリーズ ルータのプロセッサ モジュールで、CPU、システム ソフトウェア、およびデバイスで使用されるメモリ コンポーネントの大半が含まれます。スーパーバイザリ プロセッサと呼ばれることもあります。

RSVP：Resource Reservation Protocol。IP ネットワーク上でリソースの予約をサポートするためのプロトコル。IP エンド システム上で動作しているアプリケーションは、RSVP を使用して、受信するパケット ストリームの特性（帯域幅、ジッタ、最大バーストなど）を他のノードに示すことができます。RSVP は IPv6 に依存します。リソース予約設定プロトコルとも呼ばれます。

UDP：User Datagram Protocol。TCP/IP プロトコル スタックのコネクションレス型トランスポート層プロトコルです。UDP は、確認応答や配信保証なしでデータグラムを交換する単純なプロトコルです。エラー処理と再送信は、他のプロトコルで処理する必要があります。UDP は RFC 768 で定義されています。

第 11 章

MPLS トラフィック エンジニアリング MIB

MPLS トラフィック エンジニアリング MIB を使用すると、Cisco ソフトウェアで簡易ネットワーク管理プロトコル（SNMP）エージェントをサポートして、MPLS トラフィック エンジニアリング MIB（MPLS TE MIB）に実装されているとおりに、マルチプロトコル ラベル スイッチング（MPLS）トラフィック エンジニアリング（TE）を管理できます。SNMP エージェント コードが MPLS TE MIB と動作するため、標準化された SNMP ベースの方法を使用して、Cisco ソフトウェアで MPLS TE 機能を管理できます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS トラフィック エンジニアリング MIB の制約事項

- MIB オブジェクトに対して読み取り専用（RO）権限をサポートする。
- （書き込み可能となった）mplsTunnelTrapEnable オブジェクトを除き、SET 機能による設定がサポートされない。そのため、MPLS TE MIB ではインターフェイス MIB に対するインデックスの作成がサポートされている。
- （SET 機能による書き込みが可能となった）mplsTunnelTrapEnable オブジェクトに対する場合を除き、SNMP GET、GETNEXT、GETBULK の取得機能だけをサポートする。
- 保証帯域幅トラフィック エンジニアリング（GBTE）機能および自動帯域幅機能がサポートされない。

MPLS トラフィック エンジニアリング MIB に関する情報

MPLS トラフィック エンジニアリング MIB のシスコの実装

MPLS TE MIB は、インターネット技術特別調査委員会（IETF）が *draft-ietf-mpls-te-mib-05.txt* にまとめたドラフト MIB に基づいています。このドラフト MIB には、MPLS TE をサポートする機能について説明したオブジェクトが登録されています。

IETF ドラフト MIB と Cisco ソフトウェア内の TE 機能の実装はわずかに異なるため、MPLS TE MIB と Cisco ソフトウェアの内部データ構造との間でいくつかの軽微な変換が必要となります。これらの変換は、ネットワークのさまざまなホストにインストールされ、動作している SNMP エージェント コードによって行われます。低い優先順位でバックグラウンドで実行されているこの SNMP エージェント コードにより、Cisco ソフトウェアに対する管理インターフェイスが提供されます。

MPLS TE MIB に定義された SNMP オブジェクトは、標準の SNMP ユーティリティで表示できます。すべての MPLS TE MIB オブジェクトが、IETF ドラフト MIB に基づいています。このため、シスコ固有の SNMP アプリケーションを使用することなく、MPLS TE MIB に関する機能および操作をサポートできます。

MPLS トラフィック エンジニアリングの概要

Cisco ソフトウェアで MPLS TE 機能を使用すると、MPLS バックボーンによってレイヤ 2 ATM およびフレーム リレー ネットワークの TE 機能を複製および拡張できます。

サービス プロバイダーとインターネット サービス プロバイダー（ISP）のバックボーンを効率よく管理するには、TE機能が不可欠です。このようなバックボーンでは高伝送容量をサポートする必要があり、バックボーンがあるネットワークではリンクまたはノードの障害に対して高い復元力を確保する必要があります。

MPLS TE 機能が Cisco ソフトウェアに組み込まれているため、一般に WAN を経由する大量のトラフィックを豊富な機能で統合的に管理できます。MPLS TE 機能がレイヤ 3 ネットワーク サービスに統合されているため、既存のバックボーン伝送容量およびネットワーク トポロジによって制限が課せられるものの、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 MIB でサポートされている機能

- MPLS TE トンネルの動作ステータスの変更を知らせる通知メッセージを生成し、キューに入れる。
- MPLS TE トンネルの通知メッセージをイネーブル、ディセーブル、および設定するための既存の SNMP コマンドを拡張する。
- 通知メッセージが送信される動作環境で、ネットワーク管理ステーション（NMS）の名前または IP アドレスを指定する機能。
- 非揮発性メモリに通知設定を書き込む機能。

通知生成イベント

MPLS TE 通知がイネーブルになっている場合（**snmp-server enable traps mpls** コマンドを参照）、Cisco ソフトウェア内の特定のイベントに関連する通知メッセージが生成されて、ネットワーク内の指定された NMS に送信されます。

たとえば、MPLS TE トンネルが設定されている場合に、そのトンネルが「ダウン」状態から「アップ」状態に遷移すると、mplsTunnelUp 通知が NMS に送信されます。

逆に、MPLS TE トンネルが「アップ」状態から「ダウン」状態に遷移すると、mplsTunnelDown 通知が生成されて NMS に送信されます。

次の条件が満たされると、mplstunnelRerouted 通知が NMS に送信されます。

- 何らかの理由で既存の MPLS TE トンネルのシグナリング パスが失敗し、新規パス オプションがシグナリングされて有効になる（つまり、トンネルがリルートされる）。
- 既存の MPLS TE トンネルのシグナリング パスが問題なく動作しているが、さらに優れたパス オプションをシグナリングして有効にすることができる（つまり、トンネルを再最適化できる）。この再最適化は、次の方法でトリガーできます。
  - タイマー
  - **mpls traffic-eng reoptimize** コマンドの発行
  - トンネルの再シグナリングを必要とする設定変更

MPLS トラフィック エンジニアリング トンネルが再最適化されるときには、mplsTunnelReoptimized 通知は生成されません。ただし、mplsTunnelReroute 通知が生成されます。このため、NMS ではトンネル再最適化イベントとトンネル リルート イベントとを区別できません。

パス オプションは、新しいトンネルパスを確立する際の優先順位を指定するために設定できるパラメータです。たとえば、トンネル ヘッド設定を作成し、多数のパス オプションに 1 ～ n の番号を定義できます。「1」が最もプライオリティの高いオプションで、「n」が数に制限のないプライオリティの低いパス オプションです。このため、このように指定できるパス オプションの数に制限はありません。

通知の実装

MPLS TE トンネル インターフェイス（または FastEthernet や Packet over SONET（POS）インターフェイスなど他のデバイス インターフェイス）がアップ状態とダウン状態との間で遷移すると、インターフェイス MIB（ifMIB）リンク通知が生成されます。MPLS TE MIB 環境でこのような通知が発生すると、通知が MPLS TE トンネルに関連付けられているかどうかを確認するため、ソフトウェアによってインターフェイスがチェックされます。その場合、トンネル インターフェイスで発生している動作イベントに関する通知を NMS に提供するために、インターフェイスの MIB リンク通知が該当する mplsTunnelUp または mplsTunnelDown 通知と連結されます。このため、MPLS トラフィック エンジニアリング トンネル インターフェイスに関するインターフェイス MIB リンク通知が生成されると、指定の NMS に適切な mplsTunnelUp 通知または mplsTunnelDown 通知が送信されます。

MPLS TE トンネルのシグナリング パスが変更されるたびに、mplsTunnelRerouted 通知が生成されます。ただし、MPLS TE MIB でソフトウェア インテリジェンスが機能するため、トンネルの管理ステータスまたは動作ステータスのチェック時に TE トンネルがアップ状態またはダウン状態に遷移しても、リルート通知が NMS に送信されません。この場合、アップかダウン通知、またはリルート通知のどちらか一方を送信できます。両方は送信できません。そのため、ネットワークで不要なトラフィックが発生しません。

MPLS トラフィック エンジニアリング MIB の利点

- 標準ベースの SNMP インターフェイスにより、MPLS TE に関する情報を取得できる。
- MPLS TE トンネルでのトラフィック フローに関する情報を取得できる。
- 設定済みルートも含めた MPLS TE トンネル ルート、Interior Gateway Protocol（IGP）が計算したルート、および実際に通過したルートを示すことができる。
- リンクで障害が発生した場合にトンネルがどのようにリルートされたかをインターフェイス MIB とともに示すことができる。
- MPLS TE トンネルに使用される設定済みのリソースに関する情報を取得できる。
- MPLS TE トンネルの動作ステータスが大きく変更された場合、注意を喚起する通知を生成し、キューに入れることができる。
- ネットワーク管理者が評価または対処するために、指定された NMS に通知メッセージを転送する。

MPLS トラフィック エンジニアリング MIB のレイヤ構造

MPLS TE MIB をサポートする SNMP エージェント コードは、Cisco ソフトウェア内のこのようなコードの既存モデルに準じます。また、その一部は、MIB ソース コードに基づいて Cisco ツールセットにより生成されます。

SNMP エージェント コードは、Cisco ソフトウェアの MIB サポート コードとほぼ同じ階層構造となっており、次の 4 つのレイヤで構成されています。

- プラットフォームに依存しないレイヤ：このレイヤは、主に Cisco MIB 開発ツール セットによって生成され、プラットフォームや実装に依存しない機能を統合します。この Cisco MIB 開発ツール セットにより、MIB に関連付けられる標準のファイル セットが作成されます。
- アプリケーション インターフェイス レイヤ：このレイヤに属する MIB オブジェクトの機能、名前、およびテンプレート コードも、Cisco MIB 開発ツール セットによって生成されます。
- アプリケーション固有のレイヤ：このレイヤは、アプリケーション インターフェイス レイヤとアプリケーション プログラム インターフェイス（API）/データ構造レイヤとをつなぐインターフェイスであり、Cisco ソフトウェアから必須情報を取得するのに必要な作業（データ構造内の検索など）を実行します。
- API/データ構造レイヤ：このレイヤには、SNMP 管理情報を設定または取得するために取得または呼び出される Cisco ソフトウェア内のデータ構造または API が含まれています。

MPLS トラフィック エンジニアリング MIB に関連する機能およびテクノロジー

MPLS TE MIB 機能は、次の機能およびテクノロジーとともに使用されます。

- 標準ベースの SNMP ネットワーク管理アプリケーション
- MPLS
- MPLS TE

MPLS トラフィック エンジニアリング MIB でサポートされているオブジェクト

MPLS TE MIB には、Cisco ソフトウェアで読み取り専用の SNMP を使用して MPLS TE 機能を管理できるテーブルとオブジェクト定義が数多く含まれています。MPLS TE MIB は、抽象構文記法 1（ASN.1）に準拠し、これにより、理想的な MPLS TE データベースが反映されています。

標準の SNMP ネットワーク管理アプリケーションを使用すると、GET 操作で MPLS TE MIB から情報を取得して表示できます。また、GETNEXT 操作で MIB データベース内の情報を走査して表示することもできます。

Cisco ソフトウェアでサポートされている MPLS TE MIB テーブルおよびオブジェクトは次のとおりです。重要な MIB テーブル（太字タイプでハイライト表示）については、不随のテキストで簡単に説明されています。

- mplsTunnelConfigured：このノードに定義されているトンネル設定の総数。
- mplsTunnelActive：このノードに定義されているラベル スイッチド パス（LSP）の総数。
- mplsTunnelTEDistProto：使用中の IGP 配布プロトコル。
- mplsTunnelMaxHops：特定のトンネルが使用できるホップの最大数。
- mplsTunnelIndexNext：サポート対象外。0 に設定されます。
- mplsTunnelTable：このテーブルのエントリのうち、インスタンスが 0 で送信元アドレスが 0 のエントリは、トンネルヘッドの設定となります。このテーブルのそれ以外のエントリは、LSP のインスタンスとなり、シグナリングとスタンバイの両方があります。トンネルインスタンスがシグナルされると、その動作ステータス（operStatus）は「アップ」（1）に設定され、そのインスタンスはアクティブな LSP となります。

トンネル設定は、トンネル インターフェイスが定義されているトンネル ヘッドにだけ存在します。LSP はネットワークを通り、トンネル ヘッド、トンネル ミッドポイント、およびトンネル テールを必要とします。

トンネルテーブルのポインタは、他の MIB テーブルの対応するエントリを参照します。このようなポインタを使用すると、mplsTunnelTable でエントリを見つけ、他のテーブルへのポインタをたどってさらに情報を入手できます。ポインタには、mplsTunnelResourcePointer、mplsTunnelHopTableIndex、mplsTunnelARHopTableIndex、および mplsTunnelCHopTableIndex があります。

トンネル テーブルは、トンネル ID、トンネル インスタンス、トンネル送信元アドレス、およびトンネル宛先アドレスでインデックスが作成されます。各エントリの記述には適宜、エントリの適用性を示すアルファベットのサフィクスとして、トンネルヘッド設定専用の場合は（a）、LSP 専用の場合は（b）、トンネルヘッド設定と LSP に両方に使用できる場合は（c）が付与されています。

次に、各エントリのリストおよび説明を示します。

- - mplsTunnelIndex：トンネル ID と同じです（c）。
  - mplsTunnelInstance：LSP のトンネル インスタンス。ヘッド設定の場合は 0 になります（b）。
  - mplsTunnelIngressLSRId：LSP の送信元 IP アドレス。ヘッド設定の場合は 0 になります（b）。
  - mplsTunnelEgressLSRId：トンネルの宛先 IP アドレス（c）。
  - mplsTunnelName：トンネル インターフェイスのコマンド名（a）。
  - mplsTunnelDescr：トンネル設定および LSP を説明する名前（c）。
  - mplsTunnelIsIf：エントリがインターフェイスであるかどうかを示すインジケータ（c）。

- mplsTunnelIfIndex：ifMIB 内のトンネル インターフェイスのインデックス（a）。
- mplsTunnelXCPointer：（ミッドポイントだけでテールにはなし）MPLS LSR MIB の mplsXCTable 内の LSP へのポインタ（b）。
- mplsTunnelSignallingProto：トンネルで使用されているシグナリング プロトコル（c）。
- mplsTunnelSetupPrio：トンネルのセットアップ プライオリティ（c）。
- mplsTunnelHoldingPrio：トンネルの保留プライオリティ（c）。
- mplsTunnelSessionAttributes：セッション属性（c）。
- mplsTunnelOwner：トンネル オーナー（c）。
- mplsTunnelLocalProtectInUse：実装されていません（c）。
- mplsTunnelResourcePointer：リソース テーブルへのポインタ（b）。
- mplsTunnelInstancePriority：実装されていません（b）。
- mplsTunnelHopTableIndex：ホップ テーブルのインデックス（a）。
- mplsTunnelARHopTableIndex：AR ホップ テーブルのインデックス（b）。
- mplsTunnelCHopTableIndex：C ホップ テーブルのインデックス（b）。
- mplsTunnelPrimaryTimeUp：現在のパスがアップになっている秒数（a）。
- mplsTunnelPathChanges：トンネルが再シグナルされた回数（a）。
- mplsTunnelLastPathChange：最後のパスの再シグナリングが発生してからの秒数（a）。
- mplsTunnelCreationTime：トンネルが作成されたタイム スタンプ（a）。
- mplsTunnelStateTransitions：トンネルの状態が変更された回数（a）。
- mplsTunnelIncludeAnyAffinity：実装されていません（b）。
- mplsTunnelIncludeAllAffinity：トンネルがリンクを通過するために設定する必要のある属性ビット（a）。
- mplsTunnelExcludeAllAffinity：トンネルがリンクを通過するために設定してはならない属性ビット（a）。
- mplsTunnelPathInUse：トンネルのパスに使用されるパス オプション番号。アクティブなパス オプションがない場合、このオブジェクトは 0 になります（a）。
- mplsTunnelRole：ルータでのトンネルのロール。つまり、ヘッド、ミッドポイント、またはテール（c）。
- mplsTunneltotalUptime：トンネルがアップ状態になっている秒数（a）。
- mplsTunnelInstanceUptime：実装されていません（b）。
- mplsTunnelAdminStatus：トンネルの管理ステータス（c）。
- mplsTunnelOperStatus：トンネルの実際の動作ステータス（c）。

- mplsTunnelRowStatus：このオブジェクトは、新規トンネルの設定に使用されます。このオブジェクトは常に「アクティブ」であると見なされます（a）。
- mplsTunnelStorageType：トンネル エントリのストレージ タイプ（c）。

- mplsTunnelHopListIndexNext：次に mplsTunnelHopTable のインデックスとして使用できる有効なインデックス。
- **mplsTunnelHopTable**：このテーブルのエントリは、トンネル設定の場合にのみ存在し、トンネルに対して定義されたパス オプションに対応します。パス オプションには、明示的と動的の 2 つのタイプがあります。このテーブルには、明示パス オプションに記載されているすべてのホップが示されますが、動的なパス オプションについては宛先ホップだけとなります。トンネル ホップ テーブルは、トンネル ID、パス オプション、およびホップ カウントでインデックスが作成されます。

次に、各テーブル エントリのリストおよび説明を示します。

-
  - mplsTunnelHopListIndex：このテーブルのプライマリ インデックス。
  - mplsTunnelHopIndex：このテーブルのセカンダリ インデックス。
  - mplsTunnelHopAddrType：このホップのアドレスのタイプが IPv4 であるか IPv6 であるかを示します。
  - mplsTunnelHopIpv4Addr：このホップの IPv4 アドレス。
  - mplsTunnelHopIpv4PrefixLen：IPv4 アドレスのプレフィックス長。
  - mplsTunnelHopIpv6Addr：このホップの IPv6 アドレス。
  - mplsTunnelHopIpv6PrefixLen：IPv6 アドレスのプレフィックス長。
  - mplsTunnelHopAsNumber：このオブジェクトには、mplsTunnelHopAddrType の値に応じて、0 またはホップの自律システム番号が含まれます。
  - mplsTunnelHopLspId：このオブジェクトには、mplsTunnelHopAddrType の値に応じて、0 またはトンネルの LSP ID が含まれます。
  - mplsTunnelHopType：このトンネル ホップがストリクトまたはルーズのいずれでルーティングされるかを示します。
  - mplsTunnelHopRowStatus：このオブジェクトは、テーブルの新規行の設定に使用されます。
  - mplsTunnelHopStorageType：この MIB オブジェクトのストレージ タイプ。

- mplsTunnelResourceIndexNext：このオブジェクトには、mplsTunnelResourceTable にエントリを作成するときに、次に mplsTunnelResourceIndex に使用できる適切な値が含まれています。
- **mplsTunnelResourceTable**：このテーブルのエントリは、**show mpls traff9c-eng tunnels** コマンドを実行すると表示される「Tspec」情報に対応しています。これらのエントリは、LSP のためにだけ存在します。

トンネル リソーステーブルは、アドレスおよびホップ カウントでインデックスが作成されます。トンネル テーブルの mplsTunnelResourcePointer ポインタをたどるのが、このテーブルから情報を取得する最善の方法です。

次に、各テーブル エントリのリストおよび説明を示します。

- - mplsTunnelResourceIndex：このテーブルのプライマリ インデックス。
  - mplsTunnelResourceMaxRate：このトンネルでサポートされている最大レート（ビット/秒）。
  - mplsTunnelResourceMeanRate：このトンネルでサポートされている平均レート（ビット/秒）。
  - mplsTunnelResourceMaxBurstSize：このトンネルで許容されている最大バースト サイズ（バイト）。
  - mplsTunnelResourceRowStatus：このオブジェクトは、テーブルの新規行の設定に使用されます。
  - mplsTunnelResourceStorageType：この MIB オブジェクトのストレージ タイプ。
- **mplsTunnelARHopTable**：このテーブルのエントリは、トンネルが実際にたどり、ネットワークが正常にシグナルしたルートに対応しています。このテーブルに示されたホップは、Resource Reservation Protocol（RSVP）の Record Route Object（RRO）に示されたホップに対応しています。このテーブルの情報は、**show mpls traff9c-eng tunnels** コマンドを実行して表示することもできます。

実際のルート ホップ テーブルは、アドレスおよびホップ カウントでインデックスが作成されます。トンネル テーブルの mplsTunnelARHopTableIndex ポインタをたどるのが、このテーブルから情報を取得する最善の方法です。

次に、各テーブル エントリのリストおよび説明を示します。

- - mplsTunnelARHopListIndex：このテーブルのプライマリ インデックス。
  - mplsTunnelARHopIndex：このテーブルのセカンダリ インデックス。
  - mplsTunnelARHopIpv4Addr：このホップの IPv4 アドレス。
  - mplsTunnelARHopIpv4PrefixLen：IPv4 アドレスのプレフィックス長。
  - mplsTunnelARHopIpv6Addr：このホップの IPv6 アドレス。
  - mplsTunnelARHopIpv6PrefixLen：IPv6 アドレスのプレフィックス長。
  - mplsTunnelARHopAsNumber：このオブジェクトには、mplsTunnelARHopAddrType の値に応じて、0 またはホップの AS 番号が含まれます。
  - mplsTunnelARHopAddrType：この MIB エントリのアドレスのタイプ。IPv4 または IPv6 のいずれかになります。
  - mplsTunnelARHopType：このトンネル ホップがストリクトまたはルーズのいずれでルーティングされるかを示します。

- **mplsTunnelCHopTable**：このテーブルのエントリは、LSP のシグナリングに使用される RSVP の明示ルート オブジェクト（ERO）に対応しています。このテーブルのホップ リストには、Constraint-based Shortest Path First（CSPF）アルゴリズムで計算されるホップが登録されます。トンネルに「ルーズ」ホップが指定されている場合、このテーブルにはパスを完結するためにルーズ ホップ間に「埋められる」ホップが含まれます。完全明示パスを指定した場合、算出されたホップ テーブルは指定のパスに一致します。

算出されたホップテーブルは、アドレスおよびホップカウントでインデックスが作成されます。トンネル テーブルの Following the mplsTunnelCHopTableIndex ポインタをたどるのが、このテーブルから情報を取得する最善の方法です。

次に、各テーブル エントリのリストおよび説明を示します。

-
  - mplsTunnelCHopListIndex：このテーブルのプライマリ インデックス。
  - mplsTunnelCHopIndex：このテーブルのセカンダリ インデックス。
  - mplsTunnelCHopAddrType：このホップのアドレスのタイプが IPv4 であるか IPv6 であるかを示します。
  - mplsTunnelCHopIpv4Addr：このホップの IPv4 アドレス。
  - mplsTunnelCHopIpv4PrefixLen：IPv4 アドレスのプレフィックス長。
  - mplsTunnelCHopIpv6Addr：このホップの IPv6 アドレス。
  - mplsTunnelCHopIpv6PrefixLen：IPv6 アドレスのプレフィックス長。
  - mplsTunnelCHopAsNumber：このオブジェクトには、mplsTunnelHopAddrType の値に応じて、0 またはホップの自律システム番号が含まれます。
  - mplsTunnelCHopType：このトンネル ホップがストリクトまたはルーズのいずれでルーティングされるかを示します。

- **mplsTunnelPerfTable**：**mplsTunnelTable** を補強するトンネル パフォーマンス テーブルで、トンネルごとにパケット カウンタおよびバイト カウンタが用意されます。このテーブルには、次のパケット カウンタおよびバイト カウンタが含まれています。
  - mplsTunnelPerfPackets：このパケット カウンタは、トンネル ヘッドに対してだけ機能します。
  - mplsTunnelPerfHCPackets：このパケット カウンタは、トンネル ヘッドに対してだけ機能します。
  - mplsTunnelPerfErrors：このパケット カウンタは、トンネル ヘッドに対してだけ機能します。
  - mplsTunnelPerfBytes：このバイト カウンタは、トンネル ヘッドおよびトンネル ミッドポイントに対しては機能しますが、トンネル テールに対しては機能しません。
  - mplsTunnelPerfHCBytes：このバイト カウンタは、トンネル ヘッドおよびトンネル ミッドポイントに対しては機能しますが、トンネル テールに対しては機能しません。

- mplsTunnelTrapEnable：オブジェクト タイプ *mplsTunnelTrapEnable* は書き込み可能になりました。そのため、このオブジェクトタイプを「TRUE」に設定した場合、後続の通知がイネーブルになり、MPLS TE トンネルの動作ステータスに対する変更を監視できるようになります。
  - mplsTunnelUp
  - mplsTunnelDown
  - mplsTunnelRerouted

mplsTunnelTrapEnable オブジェクトを「FALSE」に設定した場合、このような動作ステータス通知は生成されません。このような通知機能は、*draft-ietf-mpls-te-mib-05.txt* にまとめられた IETF ドラフト マニュアルでの定義（mplsTeNotifications）に基づいています。

CLI から MPLS トラフィック エンジニアリング MIB 情報へのアクセス

以下の図に、MPLS TE MIB 内の特定のテーブルから情報を取得するために使用できるコマンドを示します。この図に示すように、MPLS TE MIB 内の情報の中にはコマンドで取得できないものもあります。

図 *26*：*MPLS TE MIB* 情報を取得するためのコマンド

	show mpls traffic-eng tunnels	show mpls traffic-eng tunnels summary	show ip explicit-paths	show interfaces	Not available in command
mplsTunnelTable	x				x
mplsTunnelHopTable	x		x		
mplsTunnelResourceTable	x				
mplsTunnelARHopTable	x				
mplsTunnelCHopTable	x				
mplsTunnelPerfTable	x			x	
Scalars	x	x			x

MPLS トラフィック エンジニアリング MIB からの情報の取得

ここでは、TE トンネルに関する情報を効率よく取得する方法について説明します。このような情報は、TE トンネルが数多く含まれている大規模ネットワークで便利です。

mplsTunnelTable を走査するときは、mplsTunnelName など単一の列を対象とします。このようにすると、トンネル設定ごとにインデックスを作成し、ホスト ルータに必要な LSP を用意できます。このようなインデックスを使用すると、GET 操作を実行して、mplsTunnelTable の任意の列および行から情報を取得できます。

mplsTunnelTable は、トンネルごとに他のテーブルへのポインタを提供します。たとえば、mplsTunnelResourcePointer 列は、mplsTunnelResourceTable 内のリソース割り当て情報にアクセスするのに使用できるオブジェクト ID（OID）となります。mplsTunnelHopTableIndex、mplsTunnelARHopTableIndex、mplsTunnelCHopTableIndex の各列はそれぞれ、mplsTunnelHopTable、mplsTunnelARHopTable、および mplsTunnelCHopTable のプライマリ インデックスとなります。ホップ テーブルの列とプライマリ インデックスを使用してこのように MPLS TE MIB を走査すると、そのトンネル設定のホップに関する情報を取得できます。

トンネルがインターフェイスとして処理されるため、トンネル テーブル列（mplsTunnelIfIndex）がインターフェイス MIB のインデックスとなり、そのインデックスを使用してトンネルに関するインターフェイス固有の情報を取得できます。

MPLS トラフィック エンジニアリング MIB の設定方法

ローカル ルータ上での各種 MPLS TE トンネル特性を管理するための SNMP エージェントのイネーブル化

MPLS TE MIB の SNMP エージェントは、デフォルトではディセーブルになっています。MPLS TE MIB に対して SNMP エージェントを有効にするには、次の作業を実行します。

手順の概要

1. **telnet***host*
2. **enable**
3. **show running-config**
4. **configure terminal**
5. **snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [**ipv6***nacl*] [*access-list-number*]
6. **snmp-server enable traps** [*identification-type*] [*notification-option*]
7. **exit**
8. **write memory**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**telnet***host* 例： `Router> telnet 192.172.172.172`	（*xxx.xxx.xxx.xxx* で表される）指定の IP アドレスで特定したルータに対して Telnet を実行します。
ステップ 2	**enable** 例： `Router# enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 3	**show running-config** 例： `Router# show running-config`	SNMP エージェントがすでに実行中かどうかが判別される実行コンフィギュレーションが表示されます。 • SNMP 情報が表示されない場合、手順 4 に進みます。SNMP 情報が表示される場合、必要に応じて、情報を変更できます。
ステップ 4	**configure terminal** 例： `Router# configure terminal`	グローバルコンフィギュレーションモードを開始します。
ステップ 5	**snmp-server community***string* [**view***view-name*] [**ro** | **rw**] [**ipv6***nacl*] [*access-list-number*] 例： `Router(config)# snmp-server community comaccess ro 4`	読み取り専用（RO）コミュニティストリングをイネーブルにします。
ステップ 6	**snmp-server enable traps** [*identification-type*] [*notification-option*] 例： `Router(config)# snmp-server enable traps`	SNMP 通知または SNMP 応答要求を SNMP ホストに送信するように LSR をイネーブルにします。 （注） このコマンドはオプションです。SNMP がイネーブルになると、ユーザが（TE MIB だけでなく）すべての MIB に対して照会できるようになります。
ステップ 7	**exit** 例： `Router(config)# exit`	グローバルコンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	**write memory** 例： `Router# write memory`	変更された設定を NVRAM に書き込み、設定を永続的に保存します。

SNMP エージェントのステータスの確認

ホスト ネットワーク デバイス上で SNMP エージェントがイネーブルにされたことを確認するには、次の手順を実行します。

手順の概要

1. **telnet***host*
2. **enable**
3. **show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**telnet***host* 例： `Router# telnet 192.172.172.172`	（*xxx.xxx.xxx.xxx* で表される）指定の IP アドレスで特定したターゲット デバイスに対して Telnet を実行します。
ステップ 2	**enable** 例： `Router# enable`	ターゲット デバイスで SNMP をイネーブルにします。
ステップ 3	**show running-config** 例： `Router# show running-config`	ターゲットデバイス上で実行コンフィギュレーションが表示され、表示された SNMP 情報の出力を調べるために、使用されます。

例

次に、ターゲット デバイスおよびその SNMP 情報の実行コンフィギュレーションが表示されます。

```
Router# show running-config
.
.
.
snmp-server community public ro
snmp-server community private ro
```

snmp-server ステートメントが上記の形で出力に表示されている場合、デバイス上で SNMP がイネーブルにされていることになります。

MPLS トラフィック エンジニアリング MIB の設定例

SNMPエージェントを利用して、ローカルルータ上のトンネルのMPLS TE 特性を管理する例

次に、ホスト ネットワーク デバイスで SNMP エージェントをイネーブルにする例を示します。

```
Router# configure terminal
Router(config)# snmp-server community private
```

次に、SNMPv1 および SNMPv2C をイネーブルにする例を示します。設定では、コミュニティ ストリング public を使用して、SNMP エージェントが読み取り専用アクセス権ですべての MPLS TE MIB オブジェクトにアクセスすることを許可しています。

```
Router(config)# snmp-server community public
```

次に、comaccess コミュニティ ストリングを指定するアクセス リスト 4 のメンバに、すべての MPLS TE MIB オブジェクトへの読み取り専用アクセスを許可する例を示します。その他の SNMP エージェントは MPLS TE MIB オブジェクトにアクセスできません。

```
Router(config)# snmp-server community comaccess ro 4
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』

関連項目	マニュアル タイトル
MPLS TE および拡張機能に関する情報	MPLS トラフィック エンジニアリングおよび拡張機能
MPLS TE コマンド	『*Multiprotocol Label Switching Command Reference*』
SNMP コマンド	『*Network Management Command Reference*』
SNMP コンフィギュレーション	『*Network Management Configuration Guide*』の「*Configuring SNMP Support*」

標準

規格	タイトル
draft-ietf-mpls-te-mib-05	『*MPLS Traffic Engineering Management Information Base Using SMIv2*』

MIB

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

RFC

RFC	タイトル
RFC 2026	「*The Internet Standards Process*」

シスコのテクニカル サポート

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

MPLS トラフィック エンジニアリング MIB の機能情報

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

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

表 *67* : *MPLS* トラフィック エンジニアリング *MIB* の機能情報

機能名	リリース	機能情報
MPLS トラフィック エンジニアリング MIB	Cisco IOS XE Release 2.3	MPLS トラフィック エンジニアリング MIB 機能を使用すると、Cisco ソフトウェアで SNMP エージェントをサポートして、MPLS TE MIB に実装されているとおりに、MPLS TE を管理できます。 Cisco IOS XE Release 2.3 では、Cisco ASR 1000 シリーズ Aggregation Services Router にこの機能が実装されました。 次のコマンドが導入または変更されました。**snmp-server community**、**snmp-server enable traps**、**snmp-server hpst**。

用語集

アフィニティ ビット：MPLS トラフィック エンジニアリング トンネルが通過するリンクの属性に関するトンネルの要件。トンネルのアフィニティ ビットとアフィニティ マスクは、トンネルを保持するさまざまなリンクの属性と一致する必要があります。

コール アドミッション プレシデンス：優先度が高い MPLS トラフィック エンジニアリング トンネルは、必要に応じて、優先度が低い MPLS トラフィック エンジニアリング トンネルをプリエンプション処理します。その用途としては、プライオリティが低いトンネルは別のパスを見つけることができるという前提の下で、ルーティングの難易度が高いトンネルに高いプライオリティを設定し、ルーティングの難易度が低いトンネルをプリエンプトすることなどが考えられます。

コンストレイントベース ルーティング：単に最短パスを使用するのではなく、リソースの要件とアベイラビリティを考慮して、バックボーンでのルートを決定するのに使用される手順およびプロトコル。

フロー：Point of Presence（POP）と呼ばれるポイントでバックボーンに入り、別のポイントから出ていくトラフィック負荷。バックボーン全体でトラフィック エンジニアリングする必要があります。トラフィック負荷は、入口 POP から出口 POP までの 1 つまたは複数の LSP トンネルを介して伝送されます。

ヘッドエンド：トンネルの始まりとなる LSR。トンネルの「ヘッド」、つまりトンネル インターフェイスもこの LSR に存在します。

応答確認：従来のトラップ通知メッセージよりも信頼性の高い通知メッセージのタイプ。信頼性が高いのは、応答確認メッセージでは確認応答が必要になるためです。

ラベル：スイッチング ノードに対してデータの転送方法（パケットまたはセル）を指示する短い固定長のデータ構造。

ラベル スイッチド パス（LSP）トンネル：パケットの伝送にラベル スイッチングが使用される、2 台のルータ間に設定された接続。

LSP：Label Switched Path（ラベル スイッチド パス）。ラベル付きパケットが複数のホップを介して通過するパス。このパスは、入力 LSR から開始し、出力 LSR で終了します。

LSR：Label Switch Router（ラベル スイッチ ルータ）。パケット内のラベル カプセル化の値に基づいて、パケットを転送するレイヤ 3 ルータ。

MIB：Management Information Base（管理情報ベース）。SNMP などのネットワーク管理プロトコルにより使用および管理される（MIB オブジェクトで構成される）ネットワーク管理情報のデータベース。MIB オブジェクトの値は、SNMP コマンドを使用して変更および取得できます。これらのコマンドは通常、GUI ベースのネットワーク管理システムから実行します。MIB オブジェクトはツリー構造であり、ツリーにはパブリック（標準）ブランチとプライベート（独自）ブランチを含みます。

MPLS：Multiprotocol Label Switching（マルチプロトコル ラベル スイッチング）。ラベルを使用して IP トラフィックを転送するスイッチング方式。このラベルによって、ネットワーク内のルータ

およびスイッチが、事前に確立された IP ルーティング情報に基づくパケットの転送先を指示されます。

NMS：Network Management Station（ネットワーク管理ステーション）。ネットワーク管理者がネットワーク内の他のデバイスとの通信に使用する、十分に装備された強力なコンピュータ（通常はエンジニアリング ワークステーション）。NMS は、通常、ネットワーク リソースの管理、統計情報の収集、およびさまざまなネットワーク管理および設定タスクの実行に使用されます。

通知：SNMP エージェントによってネットワーク管理ステーション、コンソール、または端末に送信されるメッセージ。これにより、Cisco IOS ソフトウェア内で重大なイベントが発生したことが示されます（トラップを参照）。

OSPF：Open Shortest Path First。IP のルーティングに使用されるリンクステート ルーティング プロトコル。

RSVP：Resource Reservation Protocol。ネットワーク リソースを予約するためのプロトコル。これにより、アプリケーション フローに対して Quality of Service（QoS）が保証されます。

SNMP：Simple Network Management Protocol（シンプル ネットワーク管理プロトコル）。TCP/IP ネットワークで、ほとんど排他的に使用されているネットワーク管理プロトコル。SNMP は、ネットワーク デバイスの監視と制御、設定の管理、統計の収集、パフォーマンスの監視、およびネットワーク セキュリティの確保を行う手段を提供します。

テールエンド：トンネルのダウンストリーム受信エンド。

トラフィック エンジニアリング：標準のルーティング方式を使用した場合に選択されるパスとは異なるパスで、ルーティングされたトラフィックがネットワークを通過できるようにする手法および処理。

トラップ：SNMP エージェントによってネットワーク管理ステーション、コンソール、または端末に送信されるメッセージ。これにより、Cisco IOS ソフトウェア内で重大なイベントが発生したことが示されます。トラップ（通知）は応答要求よりも信頼性が低くなります。トラップの受信者が受信の確認応答を送信しないので、トラップが受信されたかどうかをトラップの送信者が判断できないためです（「通知」を参照）。

VCC：Virtual Channel Connection（仮想チャネル接続）。ATM ネットワーク内の 2 つのエンドポイント間でデータを伝送する、VCL で構成された論理回線。仮想回線接続と呼ばれることもあります。

VCI：Virtual Channel Identifier（仮想チャネル識別子）。ATM セルのヘッダーにある 16 ビットのフィールド。VCI は、VPI とともに、次のネットワーク VCL が一連の ATM スイッチを経由して最終的な宛先に到達するセル パスであると識別するために使用されます。

VCL：Virtual Channel Link（仮想チャネル リンク）。VCL は ATM ネットワーク内の 2 つの隣接スイッチ間に存在する論理接続です。

VPI：Virtual Path Identifier（仮想パス識別子）。ATM セルのヘッダーにある 8 ビットのフィールド。VPI は、VCI とともに、次のネットワーク VCL が一連の ATM スイッチを経由して最終的な宛先に到達するセル パスであると識別するために使用されます。

第 12 章

MPLS-TP MIB

マルチプロトコル ラベル スイッチング トランスポート プロファイル（MPLS-TP）を使用すれば、トランスポート要件に対応できます。これは、トランスポート要件が Synchronous Optical Networking（SONET）および同期デジタル階層（SDH）時分割多重（TDM）テクノロジーから MPLS およびイーサネットテクノロジーへと発展したためです。現在、急速な標準規格の開発と市場需要の増加という両方の点から、MPLS-TP が強力に推進されています。最近、複数のサービス プロバイダーが、コア ネットワークは MPLS のまま、主に集約ネットワークとアクセス ネットワークのパケット トランスポート用に MPLS-TP テクノロジーを求めるようになっています（MPLS-TP は一部のプロバイダーからコア トランスポート用とも見なされています）。サービス プロバイダーの狙いは、イーサネット サービス、モバイル バックホール、非同期転送モード（ATM）集約交換、ビデオ トランスポート、長距離トランスポートなどの導入シナリオに対応するため MPLS-TP を使用することです。

MPLS TP MIB により、Simple Network Management Protocol（SNMP）を介して MPLS-TP 設定ノードをポーリングし、MPLS-TP ネットワークを監視および管理することができます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、Bug Search Tool およびご使用の

プラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MPLS-TP MIB の前提条件

- SNMP に関する一般知識
- SNMP を介して Cisco デバイスをクエリするために使用するソフトウェア

MPLS-TP MIB の制約事項

- MPLS-TP MIB は TP のトラップを指定しないため、トラップはサポートされません。
- MPLS-TP MIB モジュールは、ポイントツーポイント同時ルーティング双方向トンネルをサポートします。

MPLS-TP MIB に関する情報

MPLS-TP MIB の概要

MPLS-TP MIB は、SNMP プロセスの一部です。MIB は MPLS-TP 機能と相互作用して、オブジェクトとインデックスに必要なデータを取得します。

次の MIB が実装されています。

- CISCO-MPLS-TC-EXT-STD-MIB
- CISCO-MPLS-ID-STD-MIB
- CISCO-MPLS-LSR-EXT-STD-MIB
- CISCO-MPLS-TE-EXT-STD-MIB

CISCO-MPLS-TC-EXT-STD-MIB

この MIB モジュールには、MPLS ベースのトランスポート ネットワークのテキスト表記法が含まれています。

テキストの表記法	説明（IETF ドラフトより）
MplsGlobalId	このオブジェクトには、オペレータの固有識別子（Global_ID）のテキスト表記法が含まれています。Global_ID には、オペレータの自律システム番号（ASN）の 2 オクテットまたは 4 オクテットの値を含めることができます。Global_ID が 2 オクテット AS 番号から取得される場合、この 4 オクテット識別子の上位 2 オクテットをゼロに設定する必要があります。ASN 0 は予約済みです。Global_ID が 0 の場合、Global_ID は存在しません。
MplsNodeId	Node_ID は Global_ID の範囲内で割り当てられます。値 0（ドット付き 10 進表記では 0.0.0.0）は予約済みであり、使用できません。IPv4 アドレスが使用中の場合、このオブジェクトの値は LSR / 32 IPv4 ループバック アドレスから取得できます。
MplsLocalId	このテキストの表記法は、mplsTunnelTable のインデックス作成のために、小さなサイズの LSR 識別子で大きなサイズの Global_Node_ID と ICC に対応するときに使用されます。有効な IP アドレス範囲が 16777216（01.00.00.00）から開始するため、ローカル識別子は 1 ～ 16777215 の範囲で設定されます。この範囲は、mplsTunnelTable の Ingress/Egress を識別するために選択されます。 LSR-id は、IP アドレスまたはローカル識別子です。設定されている範囲が IP アドレスではない場合、管理者が完全な情報（Global_Node_ID）を mplsNodeConfigTable から取得すると想定されます。これにより、MPLS ベースのトランスポート ネットワークの双方向トンネル拡張に、既存の mplsTunnelTable が再利用されます。

CISCO-MPLS-ID-EXT-STD-MIB

この MIB モジュールには、トランスポート ネットワークでの MPLS トラフィック エンジニアリングのための汎用オブジェクト定義が含まれています。

オブジェクト	説明（IETF ドラフトより）
mplsGlobalId	このオブジェクトにより、管理者は MPLS-TP Global_ID とも呼ばれる一意のオペレータ ID を割り当てることができます。
mplsNodeId	オペレータまたはサービスプロバイダーはこのオブジェクトを使用して、MPLS-TP Node_ID を割り当てることができます。Node_ID は Global_ID の範囲内で割り当てられます。

MPLS LSR STD MIB

既存のラベル スイッチ ルータ（LSR）の MIB 機能は、以下の表に示す値を取得するために使用されます。TP では、FPI_IF4 の FPI タイプが IPv4 で使用されます。このリリースでは、IPv4 のみがサポートされています。

- **エンドポイント**：トンネルごとに、発信セグメント ラベルを表示する mplsOutSegmentTable（RFC 3813）の 1 つのエントリ、および動作 LSP の mplsInSegmentTable（RFC 3813）の 1 つのエントリが存在します。同様に、エントリは保護 LSP に表示されます。動作 LSP と保護 LSP の両方が設定されていることを前提としています。1 つの動作 LSP および 1 つの保護 LSP のみが設定されている場合は、エントリがそれに応じて表示されます。動作 LSP の mplsXCTable（RFC 3813）のトンネルごとに 2 つのエントリが存在し、保護 LSP も同様です。

- **ミッドポイント**：相互にルーティングされた双方向トンネルでは、ミッドポイントにはフォワードおよびリバース LSP が設定されています。したがって、フォワード LSP のための mplsInSegmentTable と mplsOutSegmentTable エントリのペア、およびリバース LSP のための mplsInSegmentTable と mplsOutSegmentTable のエントリが存在します。動作および保護 LSP が設定されている場合、上記にリストされているエントリは動作および保護 LSP の両方のために表示されます。mplsXCTable には 2 つのエントリが存在し、1 つはフォワード LSP 用、もう 1 つはリバース LSP 用です。動作および保護 LSP が設定されている場合、上記にリストされている mplsXCTable のエントリは動作および保護 LSP の両方のために表示されます。

- **mplsOutSegmentTable、mplsInSegmentTable、および mplsXCTable のインデックス**：mplsXCTable は mplsXCIndex（RFC3813）、（mplsXCInSegmentIndex RFC3813）、および mplsXCOutSegmentIndex（RFC3813）によってインデックスされます。mplsXCInSegmentIndex は、mplsInSegmentIndex と同じくローカル ラベルを含む 4 バイト オクテット文字列です。TP の mplsXCIndex は、オクテットの文字列形式で表されます。FPI_IF4 の FPI 値は lsd_common_issu_sensitive.enum ファイルから取得します。FPI 値 3 は TP に使用します。

  - エンドポイントでは、mplsXCIndex は、fpi_type、トンネル インデックス、および LSP 識別子を含むオクテット文字列として表されます。LSP 識別子は LSP が動作か保護かを指定します。LSP 識別子は、次の 2 つのタイプいずれかになります。

CFC_MPLS_CP_LSP_TYPE_WORKING：動作 LSP（整数値 2）、または CFC_MPLS_CP_LSP_TYPE_PROTECT：保護 LSP（整数値 3）。

```
|----|          |----||----||----||----|         |----|
FPI = 3         Tunnel-id                     LSP_ident
```

（注） 内部では、トンネル識別子を使用して if_number（発信インターフェイス）を取得し、if_number を使用して MFI をポーリングします

```
|----|
```

（1 バイト）。

- エンドポイントでは、mplsXCIndex は、fpi_type およびインラベルを含むオクテット文字列として表されます。ラベルの Fpi_type 値は 0 です。

```
|----|           |----||----||----||----|
FPI = 0          Label
```

mplsXCOutSegmentIndex は、mplsXCIndex と moi_index を加えたものと同じである mplsOutSegmentIndex と同じです。mplsOutSegmentIndex の最後の 2 バイトには、MOI リストインデックスが含まれています。

新しい cfc_mpls_cp_lsrmib_rfc_get_tp_label_id の MIB 機能は、MIB のチームが TP 関連のデータを取得するために作成されます。

オブジェクト	値およびこの値を取得するための関数
mplsOutSegmentTable	
mplsOutSegmentIndex	上記に説明されているように、このオブジェクトにはoutsegmentインデックスが含まれています。cfc_mpls_cp_lsrmib_rfc_get_outseg_entry 機能がこの値を取得するのに使用されます。
mplsOutSegmentInterface	このオブジェクトには、IDB から受信する outsegment インターフェイスが含まれます。cfc_mpls_cp_lsrmib_rfc_get_outseg_entry 機能がこの値を取得するのに使用されます。
mplsOutSegmentPushTopLabel	このオブジェクトは D_mplsOutSegmentPushTopLabel_true に設定されます。
mplsOutSegmentTopLabel	lsrmib_get_top_label 機能がこの値を取得するのに使用されます。
mplsOutSegmentTopLabelPtr	0.0 に設定します。
mplsOutSegmentNextHopAddrType	このオブジェクトの値には、mfi_out_info.nh.type の値が使用されます。

mplsOutSegmentNextHopAddr	このオブジェクトの値には、mfi_out_info.nh.ip_addr の値が使用されます。
mplsOutSegmentXCIndex	このオブジェクトには mplsXCTable の mplsXCIndex が含まれます。cfc_mpls_cp_lsrmib_rfc_get_xc_search_indices 機能が、この値を取得するのに使用されます。
mplsOutSegmentOwner	新規のマクロ（LSRMIB_MPLS_FPI_IF4）を追加します。これは D_mplsOutSegmentOwner_tp にマッピングされます。
mplsOutSegmentTrafficParamPtr	必ず 0.0 に設定します。
mplsOutSegmentRowStatus	D_mplsOutSegmentRowStatus_active
mplsOutSegmentStorageType	D_mplsInSegmentStorageType_volatile
mplsOutSegmentPerfTable	
mplsOutSegmentPerfOctets	mfi_out_info.bytes
mplsOutSegmentPerfPackets	mfi_out_info.packets
mplsOutSegmentPerfErrors	mfi_out_info.errors
mplsOutSegmentPerfDiscards	mfi_out_info.discards
mplsOutSegmentPerfHCOctets	MFI から入手してください。
mplsOutSegmentPerfDiscontinuityTime	lsrmib_get_discontinuity_time()
mplsInSegmentTable	
mplsInSegmentIndex	上記に説明されているように、このオブジェクトには insegmen インデックスが含まれています。lsrmib_get_in_label_id 機能が、この値を取得するのに使用されます。
mplsInSegmentInterface	これは 0 に設定されます。
mplsInSegmentLabel	lsrmib_get_in_label_id 機能が使用されます。
mplsInSegmentLabelPtr	必ず 0.0 に設定します。
mplsInSegmentNPop	デフォルト値の 1 に設定します。

mplsInSegmentAddrFamily	D_mplsInSegmentAddrFamily_ipV4 に設定します。
mplsInSegmentXCIndex	このオブジェクトにはmplsXCIndexが含まれます。cfc_mpls_cp_lsrmib_rfc_mfi_info_to_xc 機能が、この値を取得するのに使用されます。
mplsInSegmentOwner	D_mplsInSegmentOwner_other
mplsInSegmentTrafficParamPtr	0.0
mplsInSegmentRowStatus	D_mplsInSegmentRowStatus_active
mplsInSegmentStorageType	D_mplsInSegmentStorageType_volatile
mplsInSegmentPerfTable	
mplsInSegmentPerfOctets	mfi_out_info.bytes
mplsInSegmentPerfPackets	mfi_out_info.packets
mplsInSegmentPerfErrors	mfi_out_info.errors
mplsInSegmentPerfDiscards	mfi_out_info.discards
mplsInSegmentPerfHCOctets	MFI から入手してください。
mplsInSegmentPerfDiscontinuityTime	lsrmib_get_discontinuity_time()
mplsXCTable	
mplsXCIndex	cfc_mpls_cp_lsrmib_rfc_get_xc_search_indices 機能が、この値を取得するのに使用されます。
mplsXCInSegmentIndex	cfc_mpls_cp_lsrmib_rfc_get_xc_search_indices 機能が、この値を取得するのに使用されます。
mplsXCOutSegmentIndex	cfc_mpls_cp_lsrmib_rfc_get_xc_search_indices 機能が、この値を取得するのに使用されます。
mplsXCLSPId	cfc_mpls_cp_lsrmib_rfc_get_xc_search_indices が、この値を取得するのに使用されます。
mplsXCLabelStackIndex	このオブジェクトにはオクテット文字列 0.0 が含まれており、これはどのラベルも上位ラベルの下にスタックされないことを示します。

mplsXCOwner	RFC LSR MIB から、特定の値が TP に提供されることはありません。つまり、D_mplsXCOwner_other がこの値を取得するのに使用されます。
mplsXCRowStatus	D_mplsXCRowStatus_active に設定します。
mplsXCStorageType	D_mplsXCStorageType_volatile に設定します。
mplsXCAdminStatus	D_mplsXCAdminStatus_up に設定します。
mplsXCOperStatus	D_mplsXCOperStatus_up に設定します。

CISCO-MPLS-LSR-EXT-STD-MIB

mplsXCExtEntry：このテーブルのエントリは、mplsXCTable のエントリにより表される相互接続情報を、疎拡張によって拡張します。このテーブルのインデックスは mplsXCIndex、mplsXCInSegmentIndex、および mplsXCOutSegmentIndex です。

- **ミッドポイント**：ミッドポイントでは 2 つのエントリ（フォワード LSP のエントリとリバース LSP のエントリ）があります。保護 LSP と動作 LSP の両方が設定されている場合、LSP ごとに 2 つのエントリが作成されます。

- **エンドポイント**：エンドポイントには、mplsXCExtTunnelPointer に 2 つのエントリがあります。保護 LSP と動作 LSP の両方が設定されている場合、LSP ごとに 2 つのエントリが作成されます。

オブジェクト	説明	値およびこの値を取得するための関数

mplsXCExtTunnelPointer	このオブジェクトは、トンネル エントリ セグメントを指し示すバック ポインタを示します。mplsXCTable の該当するエントリの mplsXCRowStatus が active(1) の場合、このオブジェクトは変更できません。	（トンネルごとの）両方のエントリは、同じトンネル エントリを指し示します。TP からこの情報を取得するための新しい関数が作成される予定です。 エンドポイントでは、MIB コードがトンネル番号と LSP ID（動作/保護）を提供し、TP からその他の 2 つのトンネル インデックス（このトンネルの送信元のローカル ID と宛先のローカル ID）が戻されることが予期されます。 ミッドポイントでは、MIB コードが着信ラベルを提供し、TP から、トンネル インデックス、LSP インスタンス、source-local-id、および destination-local-id を提供する一意のトンネル エントリが戻されることが予期されます。
mplsXCOppositeDirXCPtr	このオブジェクトは、逆方向の XC エントリを指し示すポインタを示します。mplsXCTable の該当するエントリの mplsXCRowStatus が active(1) の場合、このオブジェクトは変更できません。	エンドポイントでは、このオブジェクトのエントリが 2 つあります。エンドポイントでは、出力セグメントを示すエントリに、逆方向の着信ラベルに対応する mplsXCLspId エントリが含まれています。着信ラベルに対応するエントリには、出力セグメントを表す mplsXCLspId が含まれています（つまり、TP トンネルの FPI タイプ 3 のインデックスが含まれています）。 ミッドポイントでは、このオブジェクトのエントリが 2 つあります。各エントリには、逆方向の着信ラベルを表す mplsXCLspId が含まれています。

MPLS-TE-STD-MIB および MPLS ドラフト TE MIB

MPLS-TE-STD-MIB の mplsTunnelTable は、TP トンネル エントリを示します。オブジェクトの詳細な説明については、RFC 3812 を参照してください。動作 LSP ごとに保護 LSP が設定されていることを前提とします。

TP 設定では、部分的な設定が可能です。LSP が部分的に設定されており、宛先ノード ID およびグローバル ID が指定されていない場合、ローカル ID は 0 に設定されます。

- **エンドポイント**：mplsTunnelTable には LSP ごとに 1 つのエントリがあります。
- **ミッドポイント**：動作 LSP の場合、mplsTunnelTable には、フォワード LSP に 1 つ、リバース LSP に 1 つのエントリがあります。同様に、保護 LSP が設定されている場合は、保護 LSP のエントリが表示されます。

オブジェクト	値およびこの値を取得するための関数
mplsTunnelIndex	エンドポイントでは、mplsTunnelIndex には送信元トンネル番号が格納されています。 ミッドポイントでは、mplsTunnelTable にはフォワード LSP の送信元トンネル番号と、リバース LSP の宛先トンネル番号が格納されています。
mplsTunnelInstance	LSP 番号が格納されています。この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelIngressLSRId	エンドポイントでは、トンネルの送信元の mplsNodeConfigLocalId の値が格納されています。 ミッドポイントでは、フォワード LSP のトンネルの送信元の mplsNodeConfigLocalId と、リバース LSP の宛先の mplsNodeConfigLocalId が格納されています。 値の範囲は 1 ～ 16777215 です。この値を取得するには tp_get_tunnel_detail 関数を使用します。

mplsTunnelEgressLSRId	エンドポイントでは、トンネルの宛先ノードの mplsNodeConfigLocalId の値が格納されています。 ミッドポイントでは、フォワード LSP のトンネルの宛先の mplsNodeConfigLocalId と、リバース LSP のトンネルの送信元の mplsNodeConfigLocalId が格納されています。 値の範囲は 1 ～ 16777215 です。この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelName	エンドポイントとミッドポイントの両方に対して適切なトンネル名が格納されています。この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelDescr	トンネルの説明が格納されています。この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelIsIf	TP トンネルは常にインターフェイスであるため、これは常に true です。
mplsTunnelIfIndex	トンネル ifindex が格納されています。tp_get_tunnel_detail 関数で IF 番号を取得できます。インターフェイスインデックスを取得するには、インターフェイス番号を使用できます。
mplsTunnelOwner	これは D_mplsTunnelOwner_other に設定されます。
mplsTunnelRole	この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelXCPointer	cfc_mpls_cp_lsrmib_rfc_make_XC_pointer 関数が使用されます。
mplsTunnelSignallingProto	None(1)。Cisco IOS の MPLS TP 実装にはコントロール プレーンがないため、シグナリング プロトコルがありません。
mplsTunnelSetupPrio	0。デフォルトでは、MPLS-TP LSP の優先度は 0 です。
mplsTunnelHoldingPrio	0。デフォルトでは、MPLS-TP LSP の優先度は 0 です。

mplsTunnelSessionAttributes	該当なし。0。
mplsTunnelLocalProtectInUse	このオブジェクトは、保護LSPを使用しているかどうかを示します。この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelResourcePointer	0.0。サポートされていません。
mplsTunnelPrimaryInstance	動作 LSP の LSP 番号を示すために使用されます。動作LSPが設定されていない場合、デフォルト値 0 が示されます。
mplsTunnelInstancePriority	該当なし。0。
mplsTunnelHopTableIndex	該当なし。0。
mplsTunnelPathInUse	該当なし。0。
mplsTunnelARHopTableIndex	該当なし。0。
mplsTunnelCHopTableIndex	該当なし。0。
mplsTunnelIncludeAnyAffinity	該当なし。0。
mplsTunnelIncludeAllAffinity	該当なし。0。
mplsTunnelTotalUpTime	この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelInstanceUpTime	この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelPrimaryUpTime	この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelPathChanges	該当なし。0。
mplsTunnelLastPathChange	該当なし。
mplsTunnelCreationTime	この値を取得するには tp_get_tunnel_detail 関数を使用します。
mplsTunnelStateTransitions	該当なし。0。

mplsTunnelAdminStatus	エンドポイントでは、この値を取得するには tp_get_tunnel_detail 関数を使用します。 TP はミッドポイントでの admin ステータスを維持しないため、ミッドポイントでは「testing(3)」に設定されます。
mplsTunnelOperStatus	エンドポイントでは、この値を取得するには tp_get_tunnel_detail 関数を使用します。 TP はミッドポイントでの oper ステータスを維持しないため、ミッドポイントでは「testing(3)」に設定されます。
mplsTunnelRowStatus	D_mplsTunnelRowStatus_active
mplsTunnelStorageType	D_mplsTunnelStorageType_readOnly
mplsTunnelPerfTable：このカウンタはサポートされていません。	

CISCO-MPLS-TE-EXT-STD-MIB

この MIB モジュールには、トランスポート ネットワークでの MPLS トラフィック エンジニアリングのための汎用オブジェクト定義が含まれています。

オブジェクト	説明（IETF ドラフトの定義）	値およびこの値を取得するための関数
mplsNodeConfigTable		
mplsNodeConfigLocalId	管理者はこのオブジェクトを使用して、Global_Node_IDをマップする一意のローカル識別子を割り当てることができます。	このテーブルは、TP ネットワーク内のノードを表すために使用されます。このオブジェクトは、ノードの一意のローカル値を提供します。このオブジェクトの値は 1 ～ 16777215 です。 TP は new tp_get_node_detail 関数を提供します。これは、このオブジェクトの値を取得するために使用されます。

mplsNodeConfigGlobalId	このオブジェクトは、グローバル オペレータ識別子を示します。	これは TP データ構造の mpls_tp_global_id_t global_id フィールドにマップされます。 このオブジェクトの値を取得するには、tp_get_node_detail を使用します。
mplsNodeConfigNodeId	このオブジェクトは、オペレータ内の Node_ID を示します。mplsNodeConfigIccId が NULL 以外の値で設定されている場合、このオブジェクト値はゼロです。	このオブジェクトは、TP データ構造の mpls_tp_node_id_t node_id フィールドにマップされます。 このオブジェクトの値を取得するには、tp_get_node_detail 関数を使用します。
mplsNodeConfigIccId	このオブジェクトにより、オペレータまたはサービス プロバイダーは、Ingress ID または Egress ID のいずれかに一意の MPLS-TP ITU-T Carrier Code（ICC）を設定できます。mplsNodeConfigGlobalId と mplsNodeConfigNodeId にゼロ以外の値が割り当てられている場合、このオブジェクトの値はゼロである必要があります。	このオブジェクトは 0 に設定されます。シスコ IOS 実装では、IP 互換の実装だけがサポートされます。
mplsNodeConfigRowStatus	このオブジェクトにより、管理者はこのテーブルの行を作成、変更、または削除できます。	これは「active」に設定されます。
mplsNodeConfigStorageType	この変数は、このオブジェクトのストレージ タイプを示します。値が「permanent」の概念行では、その行のすべての columnar オブジェクトへの書き込みアクセス権限を許可する必要はありません。	オブジェクトへの書き込みアクセス権限は一切許可されていないため、これは「readonly」に設定されます。
mplsNodeIpMapTable：このテーブルでは、**mplsNodeIpMapNodeId** および **mplsNodeIpMapLocalId** がインデックスとして使用されます。		
mplsNodeIpMapGlobalId	このオブジェクトは Global_ID を示します。	このオブジェクトの値を取得するには、tp_get_node_detail 関数を使用します。

mplsNodeIpMapNodeId	このオブジェクトは、オペレータ内の Node_ID を示します。	このオブジェクトの値を取得するには、tp_get_node_detail 関数を使用します。
mplsNodeIpMapLocalId	このオブジェクトには、mplsNodeConfigTable で定義されている IP 互換ローカル識別子が含まれています。	このオブジェクトの値を取得するには、tp_get_node_detail 関数を使用します。
mplsTunnelExtTable：このテーブルのインデックスは、**mplsTunnelTable** と同じです（**RFC 3812**）。		
mplsTunnelOppositeDirPtr	このオブジェクトは、フォワード LSP とリバース LSP が同じトンネルまたは異なるトンネルにある双方向トンネルにだけ適用できます。このオブジェクトは、mplsTunnelTable で 2 つのトンネル エントリを設定して双方向トンネルがセットアップされている場合に、逆方向のトンネルエントリを保持します。 値 zeroDotZero は、1 つのトンネル エントリが双方向トンネル セットアップに使用されていることを示します。	mplsTunnelTable の LSP あたりのトンネルごとに 1 つのエントリだけが表示されるため、このオブジェクトには値 0.0 が含まれます。
mplsTunnelReversePerfTable：このカウンタはサポートされていません。		
mplsNodeIccMapTable：TP の IP 互換実装だけがサポートされているため、このテーブルはサポートされていません。		

MPLS-TP MIB の設定方法

MPLS-TP MIB の設定

汎用 SNMP 設定では MPLS-TP MIB が自動的に有効になります。ただし、MPLS TP 機能を設定する必要があります。詳細については、『MPLS Transport Profile』を参照してください。

次に示す汎用 SNMP 設定タスクを実行する必要があります。

- SNMP エージェントの有効化（必須）

- SNMP エージェントのステータスの確認（任意）

SNMP エージェントのイネーブル化

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **snmp-server community***string* [**view***view-name*] [**ro**| **rw**][*number*]
5. **end**
6. **write memory**
7. **show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 SNMP の情報が表示されない場合は、次のステップに進みます。 SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。
ステップ 3	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**snmp-server community***string* [**view***view-name*] [**ro**| **rw**][*number*] 例： `Router(config)# snmp-server community public ro`	MPLS-TP MIB に対して読み取り専用（ro）のコミュニティ文字列を設定します。 • *string* 引数は、パスワードのように機能し、MPLS ネットワーク内のラベル スイッチング ルータ（LSR）上の SNMP 機能へのアクセスを許可します。 • オプションの **ro** キーワードでは、MPLS-TP MIB 内のオブジェクトへの読み取り専用（ro）アクセスを設定します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Router(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**write memory** 例： `Router# write memory`	変更した SNMP 設定をルータの NVRAM に書き込み、SNMP 設定を永続的に保存します。
ステップ 7	**show running-config** 例： `Router# show running-config`	ルータの実行コンフィギュレーションを表示して、デバイス上で SNMP エージェントがすでに実行中かどうかを判断します。 snmp-server という文が表示される場合は、ルータで SNMP がイネーブルになっています。 SNMP 情報が表示された場合は、必要に応じて情報を修正したり変更したりできます。

SNMP エージェントのステータスの確認

ホスト ネットワーク デバイス上で SNMP エージェントがイネーブルにされたことを確認するには、次の表に示す手順を実行します。

手順の概要

1. **enable**
2. **show running-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**show running-config** 例： `Device# show running-config`	ターゲット デバイスの実行コンフィギュレーションを表示します。

MPLS-TP MIB の設定例

例：SNMP エージェントのイネーブル化

次に、ホスト ネットワーク デバイスで SNMP エージェントをイネーブルにする例を示します。

```
Device# config terminal
Device(config)# snmp-server community
```

次に、SNMPv1 および SNMPv2C をイネーブルにする例を示します。設定では、コミュニティ ストリング *public* を使用して、SNMP エージェントが読み取り専用アクセス権ですべての MPLS TP MIB オブジェクトにアクセスすることを許可しています。

```
Device(config)# snmp-server community public
```

次に、*comaccess* コミュニティ ストリングを指定するアクセス リスト 4 のメンバに、すべての MPLS TP MIB オブジェクトへの読み取り専用アクセスを許可する例を示します。その他の SNMP エージェントは MPLS TP MIB オブジェクトにアクセスできません。

```
Device(config)# snmp-server community comaccess ro 4
```

例：SNMP エージェントのステータスの確認

次に、SNMP エージェントのステータスを確認する例を示します。

```
Device# show running-config
...

...

snmp-server community public RO

snmp-server community private RO
```

snmp-server ステートメントが上記の形で出力に表示されている場合、デバイス上で SNMP がイネーブルにされていることになります。

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
MPLS コマンド	『Cisco IOS Multiprotocol Label Switching Command Reference』
MPLS 転送プロファイル コンフィギュレーション資料	MPLS トランスポート プロファイル

標準および RFC

標準/RFC	タイトル
draft-ietf-mpls-tp-te-mib-02.txt	MPLS-TP トラフィック エンジニアリング（TE）Management Information Base (MIB)

シスコのテクニカル サポート

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

MPLS-TP MIB の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを

示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *68*：*MPLS-TP MIB* の機能情報

機能名	リリース	機能情報
MPLS-TP MIB	15.3(1)S XE 3S	同期光ファイバ ネットワーク（SONET）および同期デジタル ハイアラーキ（SDH）時分割多重（TDM）から MPLS テクノロジーおよびイーサネット テクノロジーへの要件の進化に応じて、転送要件を満たすことができます。

索 引