MPLS ラベル配布プロトコルの実装

MPLS（マルチ プロトコル ラベル スイッチング）は、ラベル スイッチングに基づいた転送メカニズムです。MPLS ネットワークでは、データ パケットにラベルが割り当てられ、ラベルの内容に基づいてパケット転送の決定が行われます。ラベル付きパケットを MPLS ネットワーク上で切り替えるために、さまざまな送信元と宛先のペアに対して所定のパスが確立されます。これらの所定のパスは、ラベル スイッチド パス（LSP）と呼ばれます。LSP を確立するために、MPLS シグナリング プロトコルが使用されます。Label Distribution Protocol（LDP）は、LSP を確立するために使用される MPLS シグナリング プロトコルです。このモジュールでは、MPLS LDP の設定方法について説明します。

- MPLS Label Distribution Protocol の実装に関する前提条件 （1 ページ）
- MPLS LDP の制約事項 （2 ページ）
- ラベル配布プロトコルの概要 （2 ページ）
- Label Distribution Protocol の設定 （2 ページ）
- MPLS Label Distribution Protocol：詳細 （10 ページ）

MPLS Label Distribution Protocol の実装に関する前提条件

次に、MPLS LDP を実装するための前提条件を示します。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- Cisco IOS XR ソフトウェアを実行している必要があります。
- 複合ミニイメージおよび MPLS パッケージをインストールする必要があります。
- IGP をアクティブにする必要があります。
- セッション ダウンがネイバーで隣接のダウンよりも前に発生するように、ネイバーなど、セッション保持時間の短い帯域幅を使用することを推奨します。次に、hello タイムのデフォルト値を示します。
  - 保持時間は 15 秒です。

- 間隔は 5 秒です。

たとえば、**holdtime** コマンドを使用して LDP セッション保持時間を 30 秒に設定できます。

MPLS LDP の制約事項

- LDP 統計情報は、**show mpls forwarding command** の出力に表示されません。

ラベル配布プロトコルの概要

IP 転送では、パケットがルータに到達すると、そのルータは IP ヘッダー内の宛先アドレスを確認し、ルート検索を実行してパケットをネクストホップに転送します。MPLS は、パケットがラベルに基づいて転送される転送メカニズムです。Label Distribution Protocol は、MPLS 環境でラベルを割り当て、配布、およびインストールします。これは、ネットワークとレイヤ間のルーティング情報をデータリンク レイヤのスイッチド パスに直接マッピングすることで、ラベル スイッチドルータ（LSR）がネットワークを通じて LSP を確立する一連の手順とメッセージです。これらの LSP には直接接続されたネイバーにエンドポイントを持たせたり（IP のホップバイホップ転送に対応）、ネットワーク出力ノードにエンドポイントを設定し、すべての中間ノードを介してスイッチングを可能にしたりできます。

LSP は、RSVP トラフィック エンジニアリング（TE）または LDP によりスタティックに作成できます。LDP により作成される LSP は、エンドツーエンド パスではなく、ホップバイホップ セットアップを実行します。LDP を使用すると、LSR で潜在的ピア ルータを検出し、これらのピアとの LDP セッションを確立して、ラベル バインディング情報を交換できます。ラベル バインディングを学習すると、LDP は MPLS フォワーディング プレーンを設定できるようになります。

LSP の設定方法については、MPLS Label Distribution Protocol：詳細 （10 ページ） を参照してください。

Label Distribution Protocol の設定

要件に応じて、LDP では、次のトピックで説明するいくつかの基本設定作業が必要です。

Label Distribution Protocol の設定

この項では、基本的な LDP 設定について説明します。ルータを LDP ピアとなる可能性があるルータに接続しているすべてのインターフェイス上で LDP をイネーブルにする必要があります。mpls ldp コンフィギュレーション モードでインターフェイスを指定することで、インターフェイス上で LDP を有効にできます。

設定例

次の例では、インターフェイス経由で LDP をイネーブルにする方法を示します。

```
RP/0/RP0/CPU0:Router(config)#  mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)#  router-id 192.168.70.1
RP/0/RP0/CPU0:Router(config-ldp)#  interface HundredGigE 0/9/0/0
RP/0/RP0/CPU0:Router(config-ldp-if)#  commit
```

Label Distribution Protocol 検出パラメータの設定

LDP を実行している LSR は、すべての LDP 対応インターフェイスで hello メッセージを送信して、互いを検出します。したがって、インターフェイス上で LDP hello メッセージを受信する LSR は、そのインターフェイス上の LDP ルータの存在を認識しています。LDP hello メッセージがインターフェイス上で送受信される場合、LDP を実行している 2 つの LSR 間のリンクには LDP 隣接関係があります。デフォルトでは、hello メッセージは 5 秒ごとに送信され、15 秒の保留時間があります。保留時間が切れる前に LSR がピアから検出 hello を受信しない場合、LSR は検出された LDP ネイバー リストからピア LSR を削除します。LDP 検出パラメータは、デフォルト パラメータを変更するように設定できます。

直接接続されていない LSR 間の LDP セッションは、ターゲット LDP セッションと呼ばれます。ターゲット LDP セッションの場合、LDP はターゲット hello メッセージを使用して拡張ネイバーを検出します。デフォルトでは、ターゲット hello メッセージは 10 秒ごとに送信され、90 秒の保留時間があります。

設定例

次の例に、以下の LDP 検出パラメータを設定する方法を示します。

- hello hold time
- hello interval
- targeted hello hold time
- targeted hello interval

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# router-id 192.168.70.1
RP/0/RP0/CPU0:Router(config-ldp)# discovery hello holdtime 30
RP/0/RP0/CPU0:Router(config-ldp)# discovery hello interval 10
RP/0/RP0/CPU0:Router(config-ldp)# discovery targeted-hello holdtime 120
RP/0/RP0/CPU0:Router(config-ldp)# discovery targeted-hello interval 15
RP/0/RP0/CPU0:Router(config-ldp)# commit
```

確認

このセクションでは、MPLS LDP 検出パラメータの設定を確認します。

```
RP/0/RP0/CPU0:Router# show mpls ldp parameters
LDP Parameters:
Role: Active
```

```
Protocol Version: 1
Router ID: 192.168.70.1
Discovery:
Link Hellos:     Holdtime:30 sec, Interval:10 sec
Targeted Hellos: Holdtime:120 sec, Interval:15 sec
Quick-start: Enabled (by default)
Transport address:       IPv4: 192.168.70.1
```

Label Distribution Protocol の targeted hellos の検出

直接接続されていない LSR 間の LDP セッションは、ターゲット LDP セッションと呼ばれます。直接接続されていない LDP ネイバーの場合は、両方のルータで LDP ネイバーシップを手動で設定する必要があります。

設定例

次に、直接接続されていないルータ Router1 および Router 2 に LDP を設定する例を示します。

```
RP/0/RP0/CPU0:Router1(config)#  mpls ldp
RP/0/RP0/CPU0:Router1(config-ldp)# router-id 192.168.70.1
RP/0/RP0/CPU0:Router1(config-ldp)# neighbor 172.20.10.10 targeted
RP/0/RP0/CPU0:Router1(config-ldp)# interface HundredGigE 0/9/0/0
RP/0/RP0/CPU0:Router1(config-ldp-if)# commit
```

```
RP/0/RP0/CPU0:Router2(config)#  mpls ldp
RP/0/RP0/CPU0:Router2(config-ldp)# router-id 172.20.10.10
RP/0/RP0/CPU0:Router2(config-ldp)# neighbor 192.168.70.1 targeted
RP/0/RP0/CPU0:Router2(config-ldp)# address-family ipv4
RP/0/RP0/CPU0:Router2(config-ldp-af)#discoverey targeted-hello accept
RP/0/RP0/CPU0:Router2(config-ldp-af)# commit
```

ラベル アドバタイズメント コントロール

LDP では、ラベルのアドバタイジングや受信を制御できます。ラベル アドバタイズメント コントロール（アウトバウンドフィルタリング）またはラベル受け入れコントロール（インバウンド フィルタリング）を使用して、ラベル バインディング情報の交換を制御できます。

ラベル アドバタイズメント コントロール（アウトバウンド フィルタリング）

Label Distribution Protocol はすべてのネイバーのすべてのプレフィックスのラベルをアドバタイズします。（拡張性やセキュリティが理由で）これが望ましくない場合、1 つ以上のピアに対する 1 つ以上のプレフィックスでローカル ラベル アドバタイズメントのアウトバウンド フィルタリングを実行するように LDP を設定できます。この機能は、LDP アウトバウンド ラベル フィルタリングまたはローカル ラベル アドバタイズメント コントロールと呼ばれています。**mpls ldp label advertise** コマンドを使用すると、ラベル バインディング情報の交換を制御できます。オプション キーワードを使用すると、選択プレフィックスをすべてのネイバーにアドバタイズ、選択プレフィックスを定義済みネイバーにアドバタイズ、またはすべてのプレフィックスのすべてのピアへのラベル アドバタイズメントをディセーブルにできます。選択してアドバタイズされるプレフィックスおよびピアは、アクセス リストで定義されます。

設定例：ラベル アドバタイズメント コントロール

次に、アウトバウンド ラベル アドバタイズメント コントロールを設定する例を示します。この例では、ネイバーはラベルアドバタイズメントをアドバタイズし受信するように指定されています。また、ラベル アドバタイズメントのためのインターフェイスも指定されています。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# address-family ipv4
RP/0/RP0/CPU0:Router(config-ldp-af)# label local advertise to 1.1.1.1:0 for pfx_ac11
RP/0/RP0/CPU0:Router(config-ldp-af)# label local advertise interface TenGigE 0/0/0/5

RP/0/RP0/CPU0:Router(config-ldp-af)# commit
```

ラベル受け入れコントロール（インバウンド フィルタリング）

LDP は、すべてのピアからのすべてのプレフィックスのラベルを（リモート バインディングとして）受け入れます。LDPは、リベラル ラベル保持モードで機能します。これは、LDPに、特定のプレフィックスのすべてのピアからのリモートバインディングを保持するように指示します。セキュリティ上の理由から、またはメモリを節約するため、特定のピアからのプレフィックスのセットのラベル バインディング受け入れを設定することで、この動作を上書きできます。プレフィックスの定義セットのリモート バインディングをフィルタリングする機能は、LDPインバウンドラベル フィルタリングまたはラベル受け入れコントロールとも呼ばれます。

設定例：ラベル受け入れコントロール（インバウンド フィルタリング）

次に、ラベル受け入れコントロールを設定する例を示します。この例では、LSRは、アクセス リストで定義されたプレフィックスについてネイバーからのラベル バインディングを受け入れて保持するように設定されています。

```
RP/0/RP0/CPU0:Router(config)#mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)#address-family ipv4
RP/0/RP0/CPU0:Router(config-ldp-af)#label remote accept from 192.168.1.1:0 for acl_1
RP/0/RP0/CPU0:Router(config-ldp-af)#label remote accept from 192.168.2.2:0 for acl_2
RP/0/RP0/CPU0:Router(config-ldp-af)#commit
```

ローカル ラベル割り当てコントロールの設定

LDP は、すべての IGP プレフィックスのラベル バインディングを作成し、そのすべてのピアからすべての IGP プレフィックスのラベル バインディングを受信します。LSR が多数の IGP プレフィックス用に複数のピアからラベル バインディングを受信する場合、かなりのメモリと CPU が消費されます。いくつかのシナリオでは、LDP ラベル バインディングの大半はアプリケーションにとって有用ではない場合があり、場合によってはローカル ラベルの割り当てを制限する必要があります。これは、アクセス リストを使用して、ローカル ラベルの割り当てをプレフィックスのセットに制限できる場合、LDP ローカル ラベル割り当てコントロールを使用して実行されます。ローカル ラベル割り当てを制限すると、メモリ使用要件の軽減、ローカル フォワーディングやネットワークおよびピアのアップデートの軽減など、いくつかのメリットがあります。

設定例

次の例に、IP アクセス リストを使用してローカル ラベル割り当てを設定し、ローカル ラベルが割り当ておよびアドバタイズできるプレフィックスのセットを指定する方法を示します。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# address-family ipv4
RP/0/RP0/CPU0:Router(config-ldp-af)# label local allocate for pfx_acl_1
RP/0/RP0/CPU0:Router(config-ldp-af)# commit
```

ダウンストリーム オン デマンドの設定

デフォルトでは、LDP は、すべてのルートのラベル アドバタイズメントがすべての LDP ピアから受信されるダウンストリーム未承諾モードを使用します。ダウンストリーム オン デマンド機能は、ダウンストリーム オンデマンド モードのサポートを強化します。このモードでは、ピアが明示的に要求しない限り、ラベルはそのピアにアドバタイズされません。同時に、ピアは自動的にラベルをアドバタイズしないため、ネクスト ホップが、リモート ラベルが割り当てられていないピアを示す場合、ラベル要求が必ず送信されます。

ダウンストリーム オンデマンド設定では、ACL を使用して、ダウンストリーム オンデマンド モードにピアのセットを指定します。ダウンストリーム オンデマンドを有効にするには、セッションの両方のピアで設定する必要があります。セッションの一方のピアだけでダウンストリーム オンデマンド機能が設定されている場合、そのセッションでは、ダウンストリーム オンデマンド モードを使用できません。

設定例

次に、LDP ダウンストリーム オン デマンドを設定する例を示します。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# session downstream-on-demand with ACL1
RP/0/RP0/CPU0:Router(config-ldp)# commit
```

明示的ヌル ラベルの設定

Cisco MPLS LDP は、暗黙的または明示的なヌル ラベルを、特定の LSR で終端するルートまたはプレフィックスのローカル ラベルとして使用します。これらのルートには、ローカルで接続またはアタッチされたすべてのネットワークが含まれます。デフォルトでは、ヌル ラベルは、LDP コントロール プレーンによる Penultimate Hop Popping（PHOP）メカニズムの実装を可能にする **implicit-null** です。これが望ましくない場合、LDP コントロール プレーンによる Ultimate Hop Popping（UHOP）メカニズムの実装を可能にする **explicit-null** ラベルを設定できます。明示的ヌル機能は、最終ホップ LSR で設定できます。アクセスリストを使用して、PHP を必要とする IP プレフィックスを指定することができます。

デフォルトで非ヌル ラベルを割り当てる必要がある場合でも、**implicit-null-override** コマンドを使用することで、特定のプレフィックスに暗黙的ヌル ローカル ラベルを適用できます。たとえば、デフォルトでは、LSR は、IGP ルートの非ヌル ラベルを割り当て、アドバタイズします。LSR の最後から 2 番目のホップでこのルートの LSP を終端する場合、**implicit-null-override**

コマンドを使用して、このプレフィックスに暗黙的ヌルラベルの割り当ておよびアドバタイズメントを適用できます。

設定例：明示的ヌル

次に、明示的ヌル ラベルを設定する例を示します。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# address-family ipv4
RP/0/RP0/CPU0:Router(config-ldp-af)# label local advertise explict-null
RP/0/RP0/CPU0:Router(config-ldp-af)# commit
```

設定例：暗黙的ヌルのオーバーライド

次に、プレフィックスのセットに対して暗黙的なヌルのオーバーライドを設定する例を示します。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# address-family ipv4
RP/0/RP0/CPU0:Router(config-ldp-af)# label local advertise implicit-null-override for
acl-1
RP/0/RP0/CPU0:Router(config-ldp-af)# commit
```

Label Distribution Protocol の自動設定

LDP 自動設定では、IGP プロトコルが有効になっているすべてのインターフェイスで自動的に LDP を設定できます。通常、LDP は、IGP ルートのラベルを割り当て、アドバタイズします。LDP は、IGP によりすべてのアクティブ インターフェイスでイネーブルにする必要があります。LDP の手動構成時には、LDP 下でインターフェイスのセットを定義する必要があります。この作業には時間がかかります。LDP 自動設定により、LDP 下で同じインターフェイスのリストを指定する必要がなくなり、設定作業が簡単になります。

設定例：OSPF に LDP 自動設定を有効にする

次に、指定した OSPF インスタンスに対して LDP 自動設定を有効にする例を示します。

```
RP/0/RP0/CPU0:Router(config)# router ospf 190
RP/0/RP0/CPU0:Router(config-ospf)# mpls ldp auto-config
RP/0/RP0/CPU0:Router(config-ospf)# area 8
RP/0/RP0/CPU0:Router(config-ospf-ar)# interface HundredGigE 0/9/0/0
RP/0/RP0/CPU0:Router(config-ospf-ar-if)# commit
```

セッション保護の設定

新しいリンクまたはノードがリンク障害後に起動すると、IP は、MPLS LDP よりも前の段階で速く収束し、MPLS 収束までに MPLS トラフィックが損失する可能性があります。リンクがフラップすると、リンク ディスカバリの損失のために LDP セッションもフラップします。LDP セッション保護により、トラフィックの損失が最小限に抑えられ、収束が迅速化され、既存の

LDP（リンク）セッションが保護されます。ピアに対してセッション保護が有効になっている場合、LDP は基本的な検出リンク hello に加えて、ターゲット hello（転送検出）の送信を開始します。ダイレクトリンクがダウンすると、ターゲット hello は、代替パスが存在していれば、そのパスを介してピア LSR に引き続き転送されます。したがって、LDP セッションは、リンクがダウンした後でも維持されます。

LDP セッション保護を設定して、すべてのピアまたはピアの特定のセット（peer-acl で指定）でセッションを自動的に保護することができます。LDP は、設定されると、プライマリ リンク隣接がすでに存在するネイバーのバックアップ targeted hello を自動的に開始します。これらのバックアップ targeted hello は、プライマリ リンク隣接がダウンしても、LDP セッションを保持します。

設定例

次の例では、アクセス コントロール リスト peer-acl-1 で指定されたピアの LDP セッション保護を、最大期間 60 秒に設定する方法を示します。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# session protection for peer-acl-1 duration 60
RP/0/RP0/CPU0:Router(config-ldp)# commit
```

Label Distribution Protocol 内部ゲートウェイ プロトコル（IGP）同期の設定

LDP と内部ゲートウェイ プロトコル（IGP）間の同期が失われると、MPLS トラフィックが失われます。たとえば、リンク アップ時、IGP は LDP 収束が発生する前にリンクをアドバタイズして使用できます。または、LDP セッションがダウンした後でも IGP でリンクを使用し続けることができます。

LDP IGP の同期化では、MPLS LDP がそのリンクで収束される場合にのみ、IGP が通常のメトリックでリンクをアドバタイズできるように LDP と IGP が調整されます。LDP では、LDP が適切なラベル バインディングを送信し、ピアから少なくとも 1 つのラベル バインディングを受信するリンクで、少なくとも 1 つの LDP セッションがアップで実行中の場合だけリンクが収束されると見なされます。LDP は、リンク アップまたはセッション ダウン イベント時にこの情報を IGP に通信し、IGP は、同期ステートに応じて機能します。

LDP-IGP 同期は、OSPF および ISIS プロトコルの両方でサポートされ、対応する IGP プロトコル コンフィギュレーション モードで設定されます。状況によっては、設定可能なインターバルで、再同期化の宣言を遅延する必要があります。LDP は、同期化の宣言を最大 60 秒遅延できる設定オプションを提供します。LDP は、リンク アップまたはセッション ダウン イベント時にこの情報を IGP に通信します。

LDP IGP 同期の設定：Open Shortest Path First（OSPF）の例

次に、OSPF インスタンスに LDP-IGP 同期を設定する例を示します。同期遅延は 30 秒に設定されています。

```
RP/0/RP0/CPU0:Router(config)# router ospf 100
RP/0/RP0/CPU0:Router(config-ospf)# mpls ldp sync
RP/0/RP0/CPU0:Router(config-ospf)# mpls ldp igp sync delay 30
RP/0/RP0/CPU0:Router(config-ospf)# commit
```

LDP IGP 同期の設定：Intermediate System to Intermediate System（IS-IS）

次に、IS-IS に LDP-IGP 同期を設定する例を示します。

```
RP/0/RP0/CPU0:Router(config)# router isis 100
RP/0/RP0/CPU0:Router(config-isis)#  interface HundredGigE 0/9/0/0
RP/0/RP0/CPU0:Router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:Router(config-isis-if-af)# mpls ldp sync
RP/0/RP0/CPU0:Router(config-isis-if-af)# commit
```

Label Distribution Protocol のグレースフル リスタートの設定

LDP グレースフル リスタートでは、LDP セッションが停止したときに LDP ピアが MPLS フォワーディング ステートを保持するメカニズムが提供されます。LDP グレースフル リスタートを使用しない場合、確立されたセッションで障害が発生すると、対応するフォワーディング ステートが、リスタートおよびピア ノードからすぐに消去されます。この場合、LDP フォワーディングは、最初から再起動する必要があるので、データおよび接続が失われる可能性があります。LDP グレースフル リスタートが設定されている場合、LDP セッションが再起動しても、トラフィックは中断することなく転送され続けます。LDP グレースフル リスタート機能は、セッション初期化中に 2 つのピア間でネゴシエーションされます。セッションの初期化中に、ルータはグレースフル リスタートの Typed Length Value（TLV）を送信することにより、LDP グレースフル リスタートを実行する機能をアドバタイズします。この TLV には、再接続時間と回復時間が含まれています。再接続時間と回復時間の値は、ルータでサポートされているグレースフル リスタート機能を示します。再接続時間は、再起動ルータが接続を確立するまでピア ルータが待機する時間です。ルータは、隣接ルータが再起動していることを検出すると、再接続を試みる前に回復時間の終了まで待機します。回復時間は、隣接ルータが再起動ルータに関する情報を維持する時間です。

設定例

次に、LDP グレースフル リスタートを設定する例を示します。この例では、隣接ルータがグレースフル リスタートするルータについてフォワーディング ステートを維持する時間を 180 に設定します。再接続時間は 169 秒に設定されています。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)#  interface HundredGigE 0/9/0/0
RP/0/RP0/CPU0:Router(config-ldp-if)# exit
RP/0/RP0/CPU0:Router(config-ldp)# graceful-restart
RP/0/RP0/CPU0:Router(config-ldp)# graceful-restart forwarding-state-holdtime 180
RP/0/RP0/CPU0:Router(config-ldp)# graceful-restart reconnect-timeout 169
RP/0/RP0/CPU0:Router(config-ldp)# commit
```

Label Distribution Protocol ノンストップ ルーティングの設定

LDP ノンストップ ルーティング（NSR）機能は、ルート プロセッサ（RP）または分散型ルート プロセッサ（DRP）のフェールオーバーなどの障害をルーティング ピアに見えないようにして、収束パフォーマンスへの負荷を最小限に抑えたり、回避したりします。デフォルトでは、NSR は、AToM 以外、すべての LDP セッションでグローバルにイネーブルにされています。

サービスの中断では、次のイベントが発生している場合があります。

- ルート プロセッサ（RP）または分散ルート プロセッサ（DRP）フェールオーバー
- LDP プロセスの再開
- Minimum Disruption Restart（MDR）

（注） グレースフル リスタート機能とは異なり、LDP NSR では、プロトコル拡張機能は必要なく、ネットワークの他のルータでのソフトウェア アップグレードの必要もありません。また、LDP NSR によりピア ルータで NSR をサポートする必要もありません。L2VPN 設定は、NSR ではサポートされていません。アクティブ LDP のプロセス障害によりセッションが損失します。その結果、RP スイッチオーバーがリカバリ アクションとして設定されるまで、NSR は提供できません。

設定例

次に、LDP ノンストップ ルーティングを設定する例を示します。

```
RP/0/RP0/CPU0:Router(config)# mpls ldp
RP/0/RP0/CPU0:Router(config-ldp)# nsr
RP/0/RP0/CPU0:Router(config-ldp)# commit
```

確認

```
RP/0/RP0/CPU0:Router# show mpls ldp nsr summary
Mon Dec  7 04:02:16.259 UTC
Sessions:
Total: 1, NSR-eligible: 1, Sync-ed: 0
(1 Ready)
```

MPLS Label Distribution Protocol：詳細

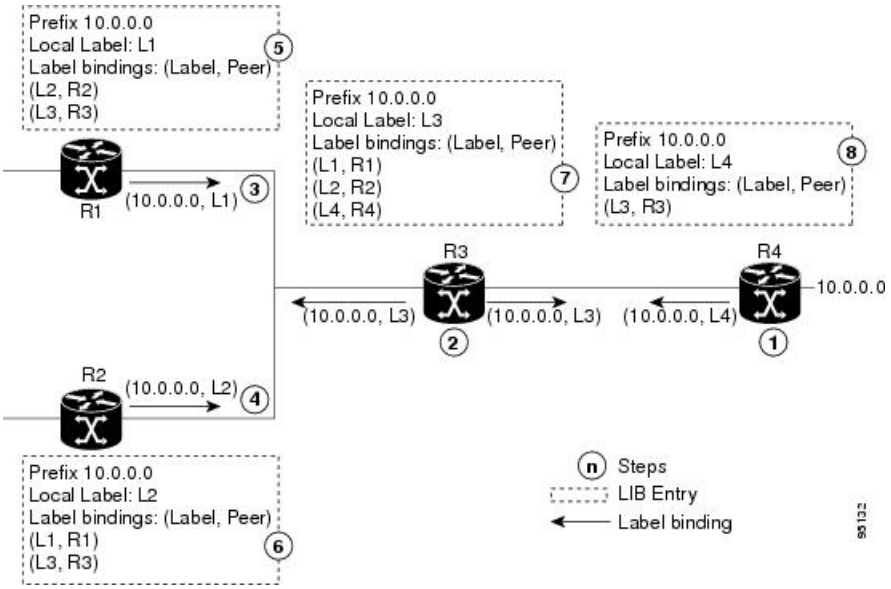
この項では、LSP の設定、LDP のグレースフル リスタート、および LDP セッション保護に関する詳細な概念情報について説明します。

ラベル スイッチド パスのセットアップ

MPLS パケットは、ラベル スイッチド パス（LSP）を使用して MPLS ネットワーク上のノード間で転送されます。LSP は、静的に、または LDP のような Label Distribution Protocol を使用して作成できます。LDP により作成されたラベル スイッチド パスはエンドツーエンド パスではなく、ホップバイホップ パスのセットアップを実行します。LDP により、ラベル スイッチ ルータ（LSR）は、潜在的なピア ルータを検出し、これらのピアとの LDP セッションを確立して、ラベル バインディング情報を交換します。

次の図は、LSP セットアップのためのラベル バインディングの交換プロセスを示します。

図 *1*: ラベル スイッチド パスのセットアップ



ネットワーク（10.0.0.0）では、ホップバイホップ LSP が各隣接ルータ（またはノード）間でセットアップされます。各ノードは、ローカル ラベルを割り当て、これをそのネイバーにバインディングとして渡します。

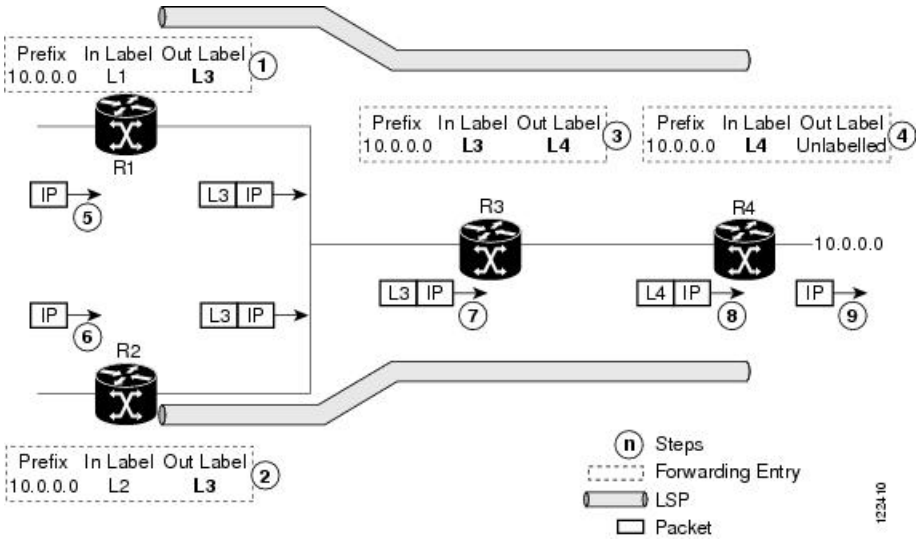
1. R4 は、ローカル ラベル L4 をプレフィックス 10.0.0.0 に割り当て、これをそのネイバー（R3）にアドバタイズします。
2. R3 は、ローカル ラベル L3 をプレフィックス 10.0.0.0 に割り当て、これをそのネイバー（R1、R2、R4）にアドバタイズします。
3. R1 は、ローカル ラベル L1 をプレフィックス 10.0.0.0 に割り当て、これをそのネイバー（R2、R3）にアドバタイズします。
4. R2 は、ローカル ラベル L2 をプレフィックス 10.0.0.0 に割り当て、これをそのネイバー（R1、R3）にアドバタイズします。
5. R1 のラベル情報ベース（LIB）は、ネイバーからのローカルおよびリモート ラベル バインディングを保持します。

6. R2 の LIB は、ネイバーからのローカルおよびリモート ラベル バインディングを保持します。
7. R3 の LIB は、ネイバーからのローカルおよびリモート ラベル バインディングを保持します。
8. R4 の LIB は、ネイバーからのローカルおよびリモート ラベル バインディングを保持します。

MPLS 転送

ラベル バインディングが学習されると、MPLS フォワーディング プレーンがセットアップされ、パケットは次の図に示すように転送されます。

図 *2 : MPLS* 転送



1. R3 は、FIB で通知されるように、10.0.0.0 のネクスト ホップなので、R1 は、ラベル バインディングを R3 から選択して、フォワーディング エントリ（レイヤ 1、レイヤ 3）をインストールします。
2. R3 は、10.0.0.0 のネクスト ホップなので（FIB で通知）、R2 は、ラベル バインディングを R3 から選択して、フォワーディング エントリ（レイヤ 2、レイヤ 3）をインストールします。
3. R4 は、10.0.0.0 のネクスト ホップなので（FIB で通知）、R3 は、ラベル バインディングを R4 から選択して、フォワーディング エントリ（レイヤ 3、レイヤ 4）をインストールします。
4. 10.0.0.0 のネクスト ホップは R4 外なので（FIB で通知）、R4 は、NO-LABEL をアウトバウンドとして使用して、フォワーディング エントリ（レイヤ 4）をインストールします。アウトバウンド パケットは IP のみで転送されます。

5. 入力 LSR R1 の着信 IP トラフィックは、ラベル インポーズされ、ラベル L3 の MPLS パケットとして転送されます。
6. 入力 LSR R2 の着信 IP トラフィックは、ラベル インポーズされ、ラベル L3 の MPLS パケットとして転送されます。
7. R3 は、ラベル L3 の MPLS パケットを受信し、MPLS ラベル フォワーディング テーブルで検索して、このパケットをラベル L4 の MPLS パケットとしてスイッチします。
8. R4 は、ラベル L4 の MPLS パケットを受信し、MPLS ラベル フォワーディング テーブルで検索して、ラベルを削除する必要があると判断します。次に、トップ ラベルをポップして、これを IP フォワーディング プレーンに渡します。
9. IP フォワーディングは、パケットを継承して、転送します。

Label Distribution Protocol のグレースフル リスタートの詳細

LDP（ラベル配布プロトコル）グレースフル リスタートは、コントロール プレーン メカニズムを提供して、ハイ アベイラビリティを保証し、ノンストップ フォワーディング（NSF）サービス中に障害を検出しリカバリできるようにします。グレースフル リスタートは、フォワーディングに影響を与えずに、シグナリングおよびコントロールプレーンの障害から回復する方法です。

LDP グレースフル リスタートを使用しない場合、確立されたセッションで障害が発生すると、対応するフォワーディング ステートが、リスタートおよびピア ノードからすぐに消去されます。この場合、LDP フォワーディングは、最初から再起動するので、データおよび接続が失われる可能性があります。

LDP グレースフル リスタート機能は、セッション初期化中に FT SESSION TLV で 2 つのピア間でネゴシエーションされます。この Typed Length Value（TLV）では、各ピアは、次の情報をピアにアドバタイズします。

再接続時間

この LSR がコントロール チャネル障害後に再接続するまで他のピアが待機する最大時間をアドバタイズします。

回復時間

他のピアがこの LSR を復元またはリフレッシュする最大時間をアドバタイズします。この時間は、先行のセッション障害後のセッション再確立中のみに使用されます。

FT フラグ

再起動により、このフラグの保存（ローカル）ノードのステートを復元できるかどうかを指定します。

グレースフル リスタート セッション パラメータが伝達され、セッションが起動し動作していると、グレースフル リスタート手順がアクティブになります。

マルチ リンク、または同じネイバーの targeted LDP hello 隣接、あるいはこれら両方のネットワークで LDP グレースフル リスタート プロセスを設定する場合、ネイバー コントロール プ

レーン障害時に任意の hello 隣接がタイムアウトになる前に、グレースフル リスタートがセッションでアクティブになっていることを確認します。これをアクティブにするには、たとえば、セッション タイムアウトが hello 隣接タイムアウトの前に発生するように、ネイバー間のセッション保持時間を低く設定します。LDPセッション保持時間は、次の式を使用して設定することを推奨します。

```
Session Holdtime <= (Hello holdtime - Hello interval) * 3
```

たとえば、リンク hello の保持時間およびインターバルがそれぞれデフォルト値の 15 秒および 5 秒である場合、セッション保持時間は、30 秒以下に設定します。

グレースフル リスタートのフレーズ

グレースフル リスタート メカニズムは、次のフェーズに分かれます。

制御通信障害の検出

システムが次のいずれかの状況を検出したときに、制御通信障害が検出されます。

- LDP hello ディスカバリ メッセージの欠落
- LDP キープアライブ プロトコル メッセージの欠落
- ピアとの Transmission Control Protocol（TCP）切断の検出

障害時のフォワーディング ステート メンテナンス

各 LSR での永続的フォワーディング ステートは、LDP コントロール プレーンにより、永続的ストレージ（チェックポイント）を介してアーカイブされます。コントロール プレーンのリカバリ中、フォワーディング プレーンは、フォワーディング ステートを保持しますが、ステイル マークを付けます。同様に、ピア コントロール プレーンも（ステイル マークを付けて）再起動中のノードに関連付けられているインストール済みフォワーディングを保持します。ローカル ノード フォワーディングとリモート ノード フォワーディングのプレーン ステートを組み合わせることで、NSF を保証し、トラフィックの損失を防ぎます。

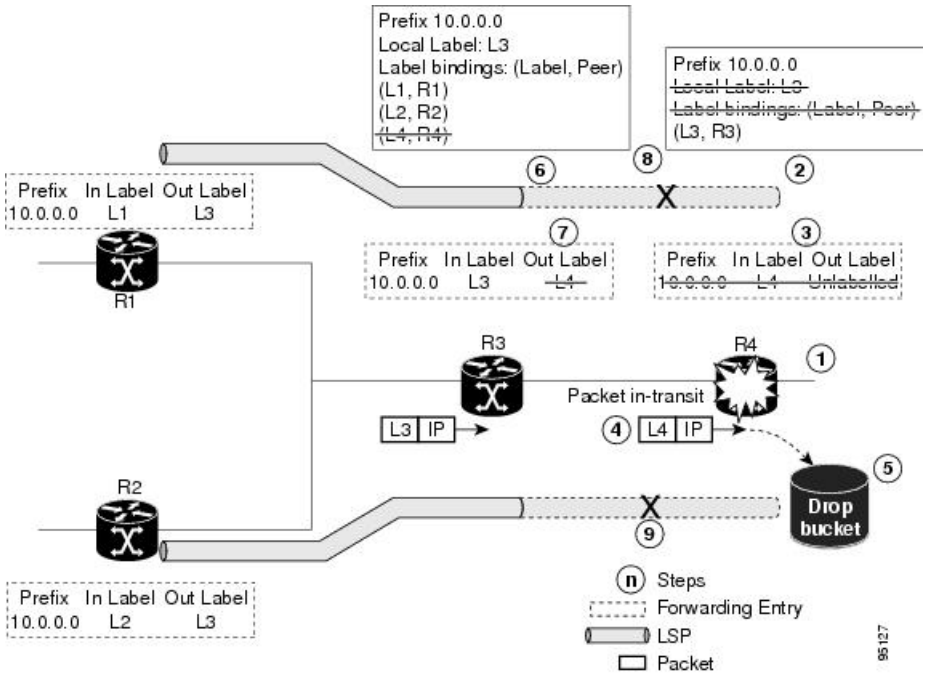
制御ステートのリカバリ

リカバリは、セッションが再確立され、ラベル バインディングが再び交換されるときに発生します。このプロセスにより、ピア ノードは、ステイル フォワーディング ステートを同期化およびリフレッシュできます。

コントロール プレーンの障害

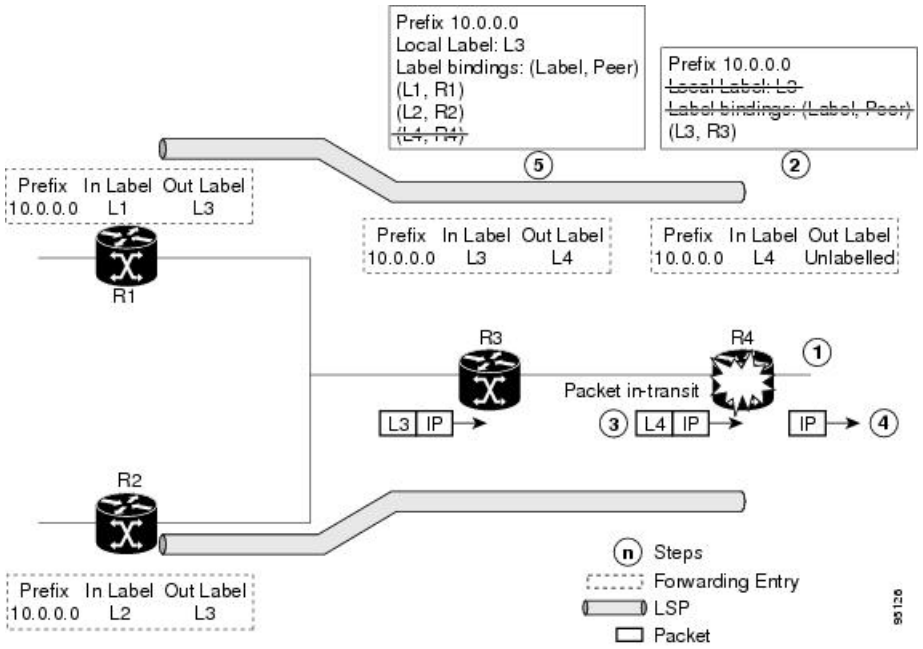
コントロール プレーン障害は、接続に影響します。ルータ コントロール プレーンによりインストールされたフォワーディング ステートが失われ、転送中パケットがドロップされ、NSF が損失する可能性があります。次の図に、コントロールプレーン障害とグレースフルリスタートによるリカバリを示し、接続の損失につながるコントロールプレーン障害の処理と結果、およびグレースフル リスタートを使用したリカバリについて説明します。

図 *3:* コントロール プレーンの障害



グレースフル リスタートによるリカバリ

図 *4:* グレースフル リスタートでのリカバリ



1. R4 LSR コントロール プレーンが再起動します。
2. コントロール プレーンが再起動すると、LIB が失われます。

3. R4 LDP コントロール プレーンによりインストールされたフォワーディング ステートがすぐに削除されます。
4. R3 から R4（ラベルはまだ L4）への転送中の任意のパケットが R4 に到着します。
5. R4 の MPLS フォワーディング プレーンが、ローカル ラベル L4 でルックアップを実行しますが失敗します。これにより、パケットがドロップされ、NSF が満たされなくなります。
6. R3 LDP ピアが、コントロール プレーン チャネルの障害を検出して、そのラベル バインディングを R4 から削除します。
7. R3 コントロール プレーンは、R4 からの出ラベルの使用を停止し、対応するフォワーディング ステート（リライト）を削除します。これにより、フォワーディングが失敗します。
8. R4 に接続されている確立済み LSP は、R3 で終端し、R1 から R4 へのエンドツーエンド LSP が終了します。
9. R4 に接続されている確立済み LSP は、R3 で終端し、R2 から R4 へのエンドツーエンド LSP が終了します。

LDP コントロール プレーンがリカバリすると、リスタート LSR は、そのフォワーディング ステートの保持タイマーを開始し、フォワーディング ステートをチェックポイント データから復元します。これにより、フォワーディング ステートおよびエントリが復元され、オールドマークが付けられます。

リスタート LSR は、正常に復元されたかどうかに関係なく、FT セッション TLV に示されているピアに再接続します。ステートが復元できた場合、バインディングは再び同期化されます。

リスタート ピアが接続し、ネイバー リカバリ タイマーを開始すると、ピア LSR は、（リスタート LSR により開始された）ネイバー再接続タイマーを停止します。ピア LSR は、リスタート ピアがそのステートを正常に復元できた場合、FT セッション TLV をチェックします。次に、対応するフォワーディング ステート エントリを復元し、リスタート ピアからバインディングを受信します。リカバリ タイマーが失効すると、任意のフォワーディング ステート（この段階ではステイル マークが付いています）が削除されます。

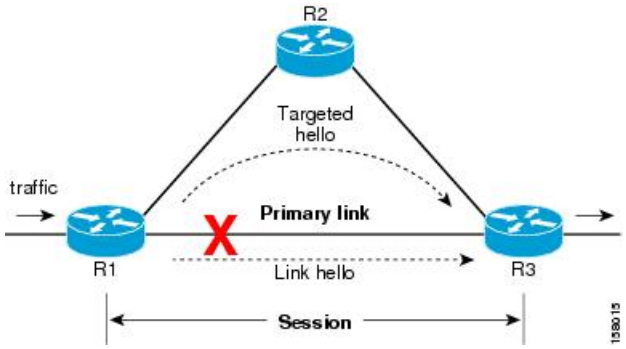
リスタート LSR が復元（再起動）に失敗した場合、リスタート LSR フォワーディング ステートおよびエントリは、タイムアウトになり削除されます。ネイバー関連のフォワーディング ステートまたはエントリは、再接続またはリカバリ タイマーが失効すると、ピア LSR により削除されます。

セッション保護の詳細

LDP セッション保護により、すべてのピアまたはピアの特定のセット（peer-acl で指定）でセッションを自動的に保護するように LDP を設定できます。LDP は、設定されると、プライマリ リンク隣接がすでに存在するネイバーのバックアップ targeted hello を自動的に開始します。これらのバックアップ targeted hello は、プライマリ リンク隣接がダウンしても、LDP セッションを保持します。

セッション保護の図は、ネイバー R1 と R3 の間の LDP セッション保護を示します。R1 および R3 間でのプライマリ リンク隣接は、リンクとバックアップが直接接続されます。ターゲット隣接は、R1 と R3 間で保守されます。ダイレクト リンクが失敗すると、LDP リンク隣接が破棄されますが、セッションは、targeted hello 隣接を使用してアップのまま実行します（R2 を介します）。ダイレクトリンクが再びアップになっても、LDP セッション ステートは変わらず、LDP は、すばやく収束し、MPLS トラフィックの転送を開始します。

図 *5*：セッション保護



（注） LDP セッション保護が（リンク障害時に）アクティブの場合、保護は無制限で保守されます。