

目次

[概要](#)

[背景説明](#)

[LSP ツリートレース-それがどのようにはたらくか](#)

[LSP ツリートレース-詳細な例](#)

[Cisco サポート コミュニティ - 特集対話](#)

概要

MPLS LSP PING は入力と出力間のラベル スイッチド パス (LSP) の健全性を検証するのに使用される基本的なツールです。この資料は LSP ツリートレースの発信側と応答側間の多重通路情報の相互対話を説明することを向けます。このツールのために利用可能な詳しいオプションに関しては[この資料](#)を参照することは役立ちます。

背景説明

MPLS EM のこの実装が。MPLS LSP 多重通路ツリートレース 機能はマルチプロトコル ラベル スイッチド (MPLS) データ平面障害を検出する RFC 4379 に基づいています。

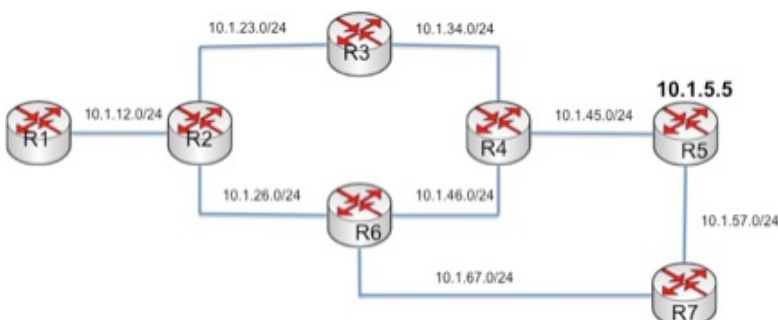
ループバックアドレス (127.x.x.x) としてプローブ パケットの IP 宛先アドレスの設定によって、LSP ツリートレースが得ます IP をルーティングされてパケットの回避によって LSP の失敗を検知するのに使用することができます。従ってエンドツーエンド接続問題がある時はいつでも、LSP 障害を除去するのに第一歩として LSP PING を使用することは有用です。

多重通路シナリオの場合には、LSP PING はすべての LSP 障害の識別を常に助けないかもしれません。それが注目されるかもしれませんので送信された複数の出力 インターフェイスである場合もあるラベル付きパケットの受信のどの Label Switch Router (LSR) でもハッシュアルゴリズムにパケットからある特定のキーおよび入力を出力 インターフェイスを決定するのに使用します。ベンダー、ハードウェア、等によっては、下記のオプションのうちのどれかがハッシュのために検討されるかもしれません:

1. だけ入ラベル スタック。
2. (ペイロードが IP なら) 入ラベル スタックおよび IP ヘッダー 詳細。
3. 入ラベル スタック、IP ヘッダーおよび転送する ヘッダ 詳細。

通常、Cisco ルータはスタックがサイズ 3 と等しいかまたはそれ以下である場合ラベルスタックおよび IP ヘッダーの組み合わせを考慮します (ペイロードとして IP と) 。

トポロジーの後の Assume。



R1-R7 はルータです。上記のトポロジーでは、下記にとして R1 から R5 へ 3 つの等価マルチパス (ECMP) ルーティングがあります、

PATH1: R1-R2-R3-R4-R5

PATH2: R1-R2-R6-R4-R5

PATH3: R1-R2-R6-R7-R5

R6 と R7 により間に問題が (壊れたラベル配布プロトコル (LDP) のように廃棄します仮定して下さいまたはミスプログラミング、等を) 分類しなさい PATH3 によって R1 から R5 にトラフィックはあることを。 R1 からの LSP PING が PATH1 か PATH2 を奪取する場合、R1 と R5 間のパスがうまくあることの仮定を終了することができます。

LSP は 127.0.0.0/8 範囲から誰としてでも IP宛先アドレスを設定する割り当てを ping します。1つの簡単なオプションが別の宛先アドレスの複数の ping パケットを送信することを手動で試みている間、すべての可能性のある ECMP パスが検証されるという保証がありません。送信元および宛先間のすべての可能性のあるパスを問い合わせ、検証する方法を必要とします。LSP 多重通路ツリートレースは活用しますか。多重通路情報符号化が。RFC4379 のセクション 3.3.1 で定義されてすべての ECMP パスを検証するのを助け。

LSP ツリートレース-それがどのようにはたらくか

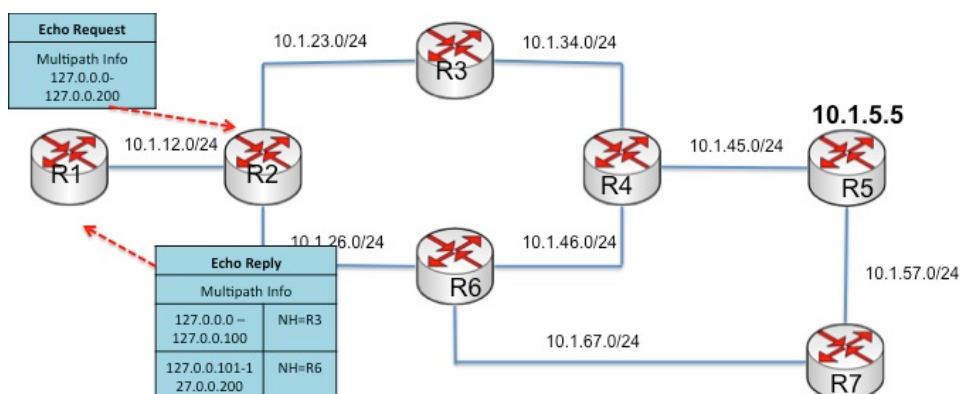
規則的な MPLS PING か traceroute は失敗が LSP ツリートレースがすべてのパスは実際にはたらいっていること検証するためによりよい方式を提供するどんなに中継ルータが ECMP 上のパケットをどのようにによって負荷分担するかないことを示すかもしれません。

LSP ツリートレースでは、発信側ルータは各ホップにインクレメンタル方法の上ラベルの TTL の設定によって MPLS エコー要求を送信します (1) から始まって。エコー要求は IP アドレス (127.0.0.0/8 範囲の内で) またはエントロピーラベル範囲の範囲を運ぶ多重通路情報 TLV を運びます。現在 Cisco デバイスは IP 目的地 オプションをサポートし、従って例は IP アドレス範囲と詳述されます。

要求パケットの受信の各中継 LSR はすべての ECMP 発信インターフェイスと答え、各インターフェイスのための要求からの IP アドレス (またはエントロピーラベル) の範囲を関連付けます。

LSP ツリートレース-詳細な例

トポロジーの後の Assume たとえば下記に。



わかりやすくするために、この例は 127.0.0.0-127.0.0.200 のアドレス範囲を使用します。LSP

ツリー トレースのステップの詳細はここにあります。

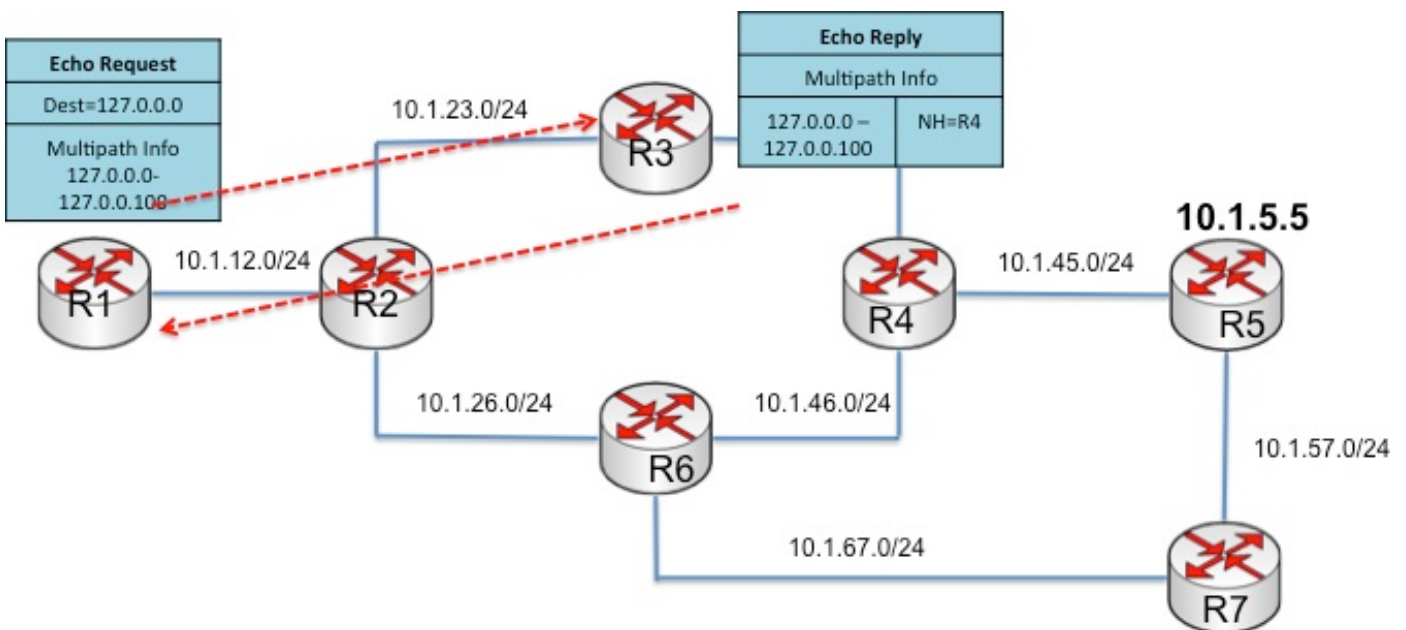
1) 発信側 (R1) は下記の詳細とのエコー要求を送信 します:

- 127.0.0.0 として IP目的地
- 127.0.0.200 に 127.0.0.0 としてアドレス範囲を運ぶ多重通路情報 TLV。
- 上ラベルの TTL は 1.に設定 されます。

2) 同じの受信の R2 は各出力 インターフェイスのための多重通路情報と応答を返します。 この例では、それは下記にとして答えます:

- IP目的地が 127.0.0.100 に 127.0.0.0 の内にある場合、パケットは R3 に送信 されます。
- IP目的地が 127.0.0.200 に 127.0.0.101 の内にある場合、パケットは R6 に送信 されます。

3) R1 は 2 つの可能性のある ECMP パスが認識し、ある従って 2.に設定 される TTL で 2 エコー要求を送信 する必要がありますことを。さまざまなテストから、発信側が 1 パスと次にことをへ行く前に常に終わることが観察されました。(しかしこれは特定の実装にあてはまるかもしれません)。

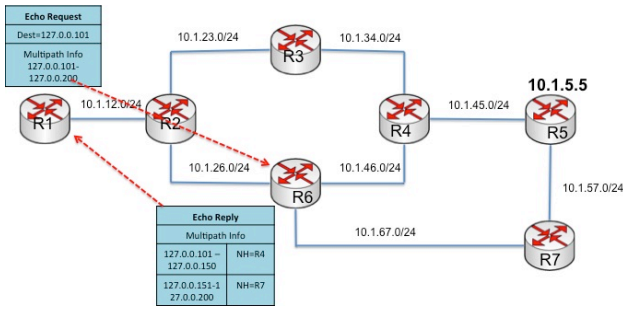


4) R1 は今下記の詳細とのエコー要求を送信 します:

- 127.0.0.0 として IP目的地
- 127.0.0.100 に 127.0.0.0 としてアドレス範囲を運ぶ多重通路情報 TLV。
- 上ラベルの TTL は 2.に設定 されます。

5) R2 は R3 に (宛先アドレスが 127.0.0.0) するようにパケットを転送します。 同じの受信の R3 は同じ多重通路情報と出力 インターフェイスが 1 つだけあるので、応答を返します。

同じ R5 に達するまで当てはまります。



6) PATH1 トレースが完了した (出力からの応答を受け取った後)、発信側は今 PATH2 を問い合わせます。これは下記の詳細とのエコー要求の送信によって実行された:

- 127.0.0.101 として IP 目的地
- 127.0.0.200 に 127.0.0.101 としてアドレス範囲を運ぶ多重通路情報 TLV
- 2. に設定される上ラベルの TTL。

7) R2 は R6 に (宛先アドレスが 127.0.0.101) するようにパケットを転送します。同じの受信の R6 は下記にとして多重通路情報と応答を返します:

- IP 目的地が 127.0.0.150 に 127.0.0.101 の内にある場合、パケットは R4 に送信されます。
- IP 目的地が 127.0.0.200 に 127.0.0.151 の内にある場合、パケットは R7 に送信されます。

8) R1 は 3. R1 が下記の詳細との次のエコー要求の送信によって PATH2 を問い合わせ続けるので総可能性のあるパスをする 1 より多くの ECMP パスがあることを認識します:

- 127.0.0.101 として IP 目的地
- 127.0.0.150 に 127.0.0.101 としてアドレス範囲を運ぶ多重通路情報 TLV
- 3. に設定される上ラベルの TTL。

9) R2 は R6 に (宛先が 127.0.0.101) するように (宛先が 127.0.0.101) ある R6 が R4 にそれを転送するようにパケットを転送し。R4 doesn か。t に ECMP パスがあり、従って同じ多重通路情報と応答を返します。次のパケットは出力 R5 に到着します。

10) PATH2 トレースが完了したので、R1 は PATH3 のためのクエリを続けます。これは下記の詳細とのエコー要求の送信によって実行された:

- 127.0.0.151 として IP 目的地
- 127.0.0.200 に 127.0.0.151 としてアドレス範囲を運ぶ多重通路情報 TLV
- 3. に設定される上ラベルの TTL。

11) R2 は R7 にそれからそれを転送する R6 にパケットを転送します。R7 は同じ多重通路情報 TLV と応答を返します。次のパケットは出力 ルータ R5 に到着します。

これらのステップが完了する後、R1 は詳細の下で持っています:

Multipath Information		
	Address Range	Path
PATH1	127.0.0.0 to 127.0.0.100	R1-R2-R3-R4-R5
PATH2	127.0.0.101 to 127.0.0.150	R1-R2-R6-R4-R5
PATH3	127.0.0.151 to 127.0.0.200	R1-R2-R6-R7-R8

127.0.0.0 および 127.0.0.100 内の宛先アドレスの使用によって、パケット転送は PATH1 に他の範囲からのアドレスを使用しながらそれぞれパス上のパケットを転送することに影響を及ぼす影響を及ぼされます。

12) この場合発信側は 255 にすべてのパスが検証されたエンドツーエンドであるように設定された TTL で 3 つのエコー要求パケットを送信し、各範囲から『address』を選択します。

ECMP トレースに使用するべきコマンドは *traceroute MPLS 多重通路 ipv4 <prefix> <mask>* です。以下は出力例です。

```
R1#traceroute mpls multipath ipv4 10.1.5.5 255.255.255.255
```

```
Starting LSP Multipath Traceroute for 10.1.5.5/32
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,
'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
```

```
LLL!
Path 0 found,
output interface Et0/0.12 nexthop 10.1.12.2
source 10.1.12.1 destination 127.0.0.4
```

```
LL!
Path 1 found,
output interface Et0/0.12 nexthop 10.1.12.2
source 10.1.12.1 destination 127.0.0.2
```

```
L!
Path 2 found,
output interface Et0/0.12 nexthop 10.1.12.2
source 10.1.12.1 destination 127.0.0.0
```

```
Paths (found/broken/unexplored) (3/0/0)
```

```
Echo Request (sent/fail) (9/0)
Echo Reply (received/timeout) (9/0)
Total Time Elapsed 27 ms
```

3 つのパスがあり、すべてのパスがうまく働いていることを出力の上でそれに示します注意して下さい。コマンドの上の冗長なノブを使用する下記にとしてすべてのホップをリストします:

```
R1#traceroute mpls multipath ipv4 10.1.5.5 255.255.255.255 verbose
```

```
Starting LSP Multipath Traceroute for 10.1.5.5/32
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,
'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
LLL!
Path 0 found,
output interface Et0/0.12 nexthop 10.1.12.2
source 10.1.12.1 destination 127.0.0.4
0 10.1.12.1 10.1.12.2 MRU 1500 [Labels: 22 Exp: 0] multipaths 0
L 1 10.1.12.2 10.1.23.3 MRU 1500 [Labels: 23 Exp: 0] ret code 8 multipaths 2
L 2 10.1.23.3 10.1.34.4 MRU 1500 [Labels: 22 Exp: 0] ret code 8 multipaths 1
L 3 10.1.34.4 10.1.45.5 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8 multipaths 1
```

```
! 4 10.1.45.5, ret code 3 multipaths 0
LL!
Path 1 found,
output interface Et0/0.12 nexthop 10.1.12.2
source 10.1.12.1 destination 127.0.0.2
0 10.1.12.1 10.1.12.2 MRU 1500 [Labels: 22 Exp: 0] multipaths 0
L 1 10.1.12.2 10.1.26.6 MRU 1500 [Labels: 16 Exp: 0] ret code 8 multipaths 2
L 2 10.1.26.6 10.1.46.4 MRU 1500 [Labels: 22 Exp: 0] ret code 8 multipaths 2
L 3 10.1.46.4 10.1.45.5 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8 multipaths 1
! 4 10.1.45.5, ret code 3 multipaths 0
L!
Path 2 found,
output interface Et0/0.12 nexthop 10.1.12.2
source 10.1.12.1 destination 127.0.0.0
0 10.1.12.1 10.1.12.2 MRU 1500 [Labels: 22 Exp: 0] multipaths 0
L 1 10.1.12.2 10.1.26.6 MRU 1500 [Labels: 16 Exp: 0] ret code 8 multipaths 2
L 2 10.1.26.6 10.1.67.7 MRU 1500 [Labels: 17 Exp: 0] ret code 8 multipaths 2
L 3 10.1.67.7 10.1.57.5 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8 multipaths 1
! 4 10.1.57.5, ret code 3 multipaths 0
Paths (found/broken/unexplored) (3/0/0)
Echo Request (sent/fail) (9/0)
Echo Reply (received/timeout) (9/0)
Total Time Elapsed 29 ms
```