Der Traceroute-Befehl in MPLS

Inhalt

Einführung

In diesem Dokument wird die Funktionsweise des **traceroute-**Befehls in einer MPLS-Umgebung (Multiprotocol Label Switching) erläutert.

Voraussetzungen

Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

- Grundlegendes MPLS-Wissen

Weitere Informationen finden Sie in den Häufig gestellten Fragen zu MPLS für Anfänger.

Verwendete Komponenten

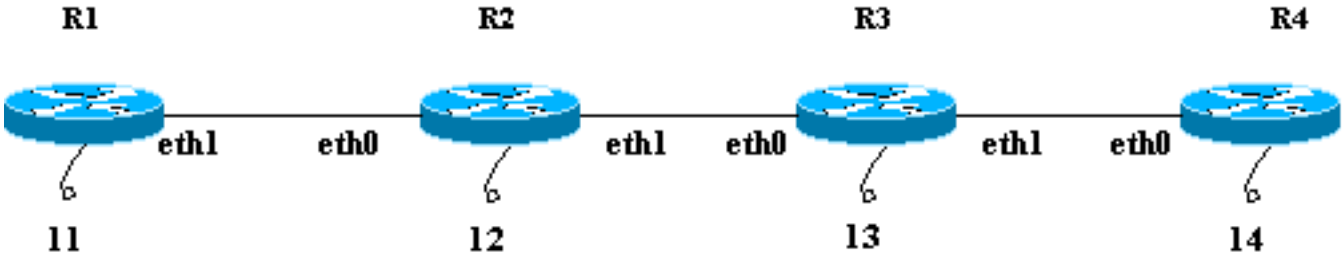
Dieses Dokument ist nicht auf bestimmte Software- und Hardwareversionen beschränkt.

Konventionen

Weitere Informationen zu Dokumentkonventionen finden Sie unter Cisco Technical Tips Conventions (Technische Tipps zu Konventionen von Cisco).

Normal traceroute-Befehl

In diesem Abschnitt wird beschrieben, wie ein herkömmlicher **Traceroute-**Befehl funktioniert. Dieses Diagramm zeigt eine Dienstanbieter-Konfiguration, bei der Router 1 (R1) und Router 4 (R4) Provider Edge-Router (PE) und Router 2 (R2) und Router 3 (R3) Provider-Router (P) sind.



In diesem Beispiel wird ein **Traceroute** zum R4-Loopback 14 von R1 geleitet. R1 verwendet ein User Datagram Protocol (UDP)-Datagramm mit einem beliebigen Zielport-Wert größer als 32000. Wenn Sie einen solchen hohen Wert für die Portnummer auswählen, wird sichergestellt, dass dieser Port für den beabsichtigten Empfänger nicht vorhanden ist. Dieses Datagramm wird in ein IP-Paket eingefügt.

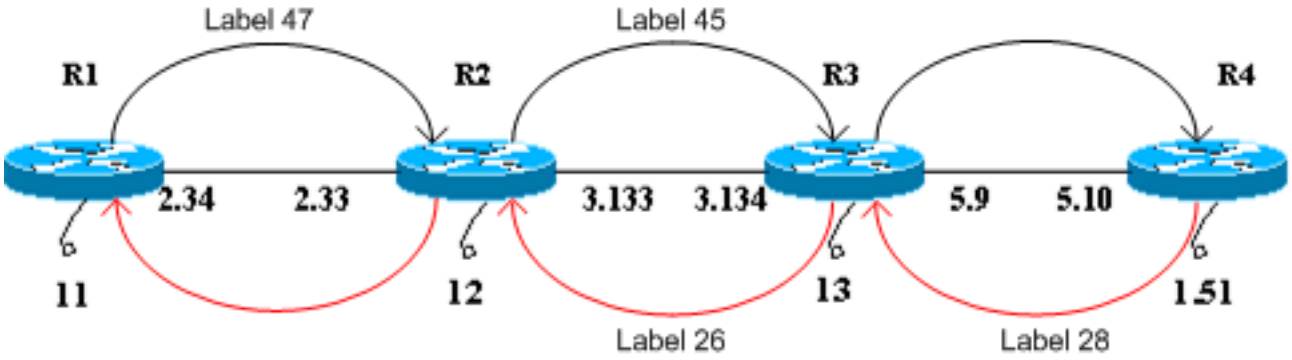
Hinweis: Wenn in diesem Dokument ein IP-Paket erwähnt wird, ist es ein IP-Paket, das das UDP-Datagramm enthält.

Dies ist eine Folge von Ereignissen für einen normalen **Traceroute-**Befehl:

1. R1 sendet das IP-Paket mit der Zieladresse 14 und einer Time to Live (TTL) von 1 über seine eth1-Schnittstelle.
2. R2 empfängt das Paket und stellt fest, dass es nicht der beabsichtigte Empfänger ist und die TTL des Pakets 1 ist. Es verwirft das Paket und sendet eine ICMP-Meldung (Internet Control Message Protocol), die abgelaufen ist. Die Quelladresse dieser ICMP-Nachricht ist die IP-Adresse des R2 eth0 (die Adresse der Schnittstelle, die das ursprüngliche Paket empfangen hat).
3. Nach Erhalt der ICMP-Nachricht sendet R1 ein anderes IP-Paket, das für 14 bestimmt ist, mit einer TTL von 2 über seine eth1-Schnittstelle.
4. R2 empfängt das Paket und stellt fest, dass es nicht der beabsichtigte Empfänger ist und der beabsichtigte Empfänger über R3 erreicht werden kann. Er senkt die TTL (von 2 auf 1) und leitet das Paket an R3 weiter. R3 empfängt das Paket und stellt fest, dass es nicht der beabsichtigte Empfänger ist. Die TTL ist 1. Es verwirft das Paket und sendet eine abgelaufene TTL-ICMP-Nachricht an R1 mit seiner eth0-Adresse als Quelladresse.
5. R1 empfängt die ICMP-Nachricht und sendet ein weiteres IP-Paket über seine eth1-Schnittstelle mit einem TTL-Wert von 3 an 14. R2 und R3 verringern unterwegs die TTL und geben sie an R4 weiter. R4 ruft das Paket ab, erkennt, dass es der beabsichtigte Empfänger ist, und versucht, eine Verbindung zum Port-Wert im UDP-Datagramm herzustellen. R4 stellt fest, dass dieser Port nicht vorhanden ist, und sendet eine Fehlermeldung `für den` ICMP-`Port`, der nicht erreichbar ist, an R1.Wie zuvor lautet die Quelladresse dieser ICMP-Meldung eth0 von R4. Das **Traceroute-**Programm verfügt jetzt über alle ICMP-Fehlermeldungen mit den entsprechenden Quelladressen und hat die vollständige Route zum Ziel.

MPLS-Traceroute-Befehl

Betrachten Sie das gleiche Szenario, das im Abschnitt Normal Traceroute Command beschrieben wird, mit Ausnahme aller Router, R1 bis R4, wird jetzt Label Switching statt IP-Forwarding betrieben. Die Testumgebung wird in diesem Diagramm dargestellt. Alle in der Testumgebung gezeigten Schnittstellen befinden sich im Netzwerk 10.13.0.0.



Für die Zwecke dieses Dokuments gehen wir davon aus, dass

- R1 verwendet ein Label von 47, um R4 zu erreichen, und leitet Pakete an R2 weiter.
- R2 verwendet ein Label von 45, um R4 zu erreichen, und leitet Pakete an R3 weiter.
- R3 öffnet das Label und leitet das Paket an R4 weiter.
- R4 verwendet ein Label von 28, um R1 zu erreichen, und leitet Pakete an R3 weiter.
- R3 verwendet ein Label von 26, um R1 zu erreichen und Pakete an R2 weiterzuleiten.
- R2 öffnet das Label und leitet das Paket an R1 weiter.

Diese Schritte zeigen die Ereignissequenz für die Durchführung einer **Traceroute** von R1 zum R4-Loopback 10.13.1.51.

1. R1 sendet ein Label-Switched Packet mit einem Label von 47 und einem TTL von 1 bis R2. Das TTL-Feld des IP-Pakets wird in das TTL-Feld des Label-Headers kopiert.
2. R2 erkennt, dass es sich nicht um den beabsichtigten Empfänger handelt und dass die TTL 1 ist. Es verwirft das Paket und erstellt eine abgelaufene ICMP-Nachricht (TTL abgelaufen), wie es bei einem normalen IP-Paket der Fall wäre. In diesem Fall wird das ICMP-Nachrichtenpaket pro ICMP-Erweiterungen für MPLS generiert.
3. R2 fügt das Label 47 (das eingehende Label, das abgelaufen ist) an die ICMP-Nachricht an. Das Paket wird nicht direkt an R1 gesendet. Stattdessen konsultiert sie ihre Label Forwarding Information Base (LFIB) und stellt fest, dass sie für Pakete mit einem Label von 47 ein Label von 45 verwenden sollte. Das Paket wird mit dem Label "45" gekennzeichnet und die abgelaufene TTL-ICMP-Nachricht an R3 gesendet.
4. R3 öffnet das Label und sendet es an R4. R4 erkennt, dass das Ziel R1 ist, gibt ein Label von 28 an die Nachricht und sendet es über R3 und R2 an R1.
5. Die ICMP-Fehlermeldung läuft bis zum anderen Ende, bevor sie an R1 zurückgesendet wird. In diesem Beispiel wird veranschaulicht:

Die gesniffenen Pakete an der Ethernet-Schnittstelle an R4 bestätigen die Schritte 1-5. In der Sniffer-Ausgabe ist **`Frame 1` das eingehende Paket und `Frame 2`** das ausgehende Paket von R4. Die Ausgabe wird entsprechend dieser Diskussion formatiert, und die zu beachtenden

Punkte sind fett dargestellt.

Frame 1 (182 on wire, 182 captured)
Ethernet II
Destination: 00:04:4e:7a:74:00 (Cisco_7a:74:00)
Source: 00:03:fd:1c:86:84 (Cisco_1c:86:84)
Type: IP (0x0800)
Internet Protocol
Version: 4
Header length: 20 bytes
Time to live: 254
Protocol: ICMP (0x01)
Header checksum: 0x1b8e (correct)
Source: 10.13.2.33 (10.13.2.33)
Destination: 10.13.2.34 (10.13.2.34)
Internet Control Message Protocol
Type: 11 (Time-to-live exceeded)
Code: 0 (TTL equals 0 during transit)
Checksum: 0x0c88 (correct)
Data (140 bytes)
04500 001c 9e19 0000 0111 044a 0a0d 0222E.........J..."
100a0d 0133 989d 829a 0008 cd37 0000 0000...3.......7....
200000 0000 0000 0000 0000 0000 0000 0000................
300000 0000 0000 0000 0000 0000 0000 0000................
400000 0000 0000 0000 0000 0000 0000 0000................
500000 0000 0000 0000 0000 0000 0000 0000................
600000 0000 0000 0000 0000 0000 0000 0000................
700000 0000 0000 0000 0000 0000 0000 0000................
802000 edf2 0008 0101 00**02 f**101..........

Frame 2 (186 on wire, 186 captured)
Ethernet II
Destination: 00:03:fd:1c:86:84 (Cisco_1c:86:84)
Source: 00:04:4e:7a:74:00 (Cisco_7a:74:00)
Type: MPLS label switched packet (0x8847)
MultiProtocol Label Switching Header
MPLS Label: Unknown (28)
MPLS Experimental Bits: 6
MPLS Bottom Of Label Stack: 1
MPLS TTL: 253
Internet Protocol
Version: 4
Header length: 20 bytes
Time to live: 253
Protocol: ICMP (0x01)
Header checksum: 0x1c8e (correct)
Source: 10.13.2.33 (10.13.2.33)
Destination: 10.13.2.34 (10.13.2.34)
Internet Control Message Protocol
Type: 11 (Time-to-live exceeded)
Code: 0 (TTL equals 0 during transit)
Checksum: 0x0c88 (correct)
Data (140 bytes)
04500 001c 9e19 0000 0111 044a 0a0d 0222E.........J..."
100a0d 0133 989d 829a 0008 cd37 0000 0000...3.......7....
200000 0000 0000 0000 0000 0000 0000 0000................
300000 0000 0000 0000 0000 0000 0000 0000................
400000 0000 0000 0000 0000 0000 0000 0000................
500000 0000 0000 0000 0000 0000 0000 0000................
600000 0000 0000 0000 0000 0000 0000 0000................
700000 0000 0000 0000 0000 0000 0000 0000................
802000 edf2 0008 0101 0002 f101..........

In **Frame 1** der Ausgabe ist das erste von R4 empfangene Paket die abgelaufene TTL-ICMP-Nachricht von R2 (10.13.2.33, die Schnittstelle, auf der das ursprüngliche Paket empfangen wurde) an R1 (10.13.2.34). Im Datenbereich der ICMP-Nachricht ist das MPLS-Label (20 Byte) bei 0x89 Byte und der erste Eintrag von 0x8A abgelaufen und der Wert ist 0x02F oder 47. Dies ist das eingehende Label des Pakets mit einer TTL von 1. R2 fügt dieses Label in die ICMP-Fehlermeldung an.In **Frame 2** der Ausgabe wird der `Typ` als `MPLS-Label-Switched-Paket` angezeigt, d. h., es handelt sich um ein MPLS-Paket. R4 setzt das Label 28 an Frame 1 und leitet es über den Label-Switching-Pfad an R1 weiter. Der MPLS-Header im Frame ist fett formatiert. Wenn Sie sich auch auf den TTL-Teil des Pakets beziehen, ist in Frame 1 der Wert 254 und in Frame 2 der Wert 253. R4 hat sie um 1 verringert.

6. R1 empfängt die ICMP-Nachricht und sendet ein weiteres Paket mit dem Label 47 und einem TTL von 2 an R2. R2 tauscht Etiketten aus, verringert TTL (von 2 auf 1) und leitet sie an R3 weiter. Wie in Schritt 2 sendet R3 eine abgelaufene TTL-ICMP-Nachricht, die an das eingehende Label angehängt ist, das an R4 abgelaufen ist, und R4 sendet diese dann zurück an R1.Die hier gezeigte Sniffer-Ausgabe bei R4 bestätigt Schritt 6:

```
Frame 3 (182 on wire, 182 captured)
Ethernet II
Destination: 00:04:4e:7a:74:00 (Cisco_7a:74:00)
Source: 00:03:fd:1c:86:84 (Cisco_1c:86:84)
Type: IP (0x0800)
Internet Protocol
Version: 4
Header length: 20 bytes
Time to live: 255
Protocol: ICMP (0x01)
Header checksum: 0x146f (correct)
Source: 10.13.3.134 (10.13.3.134)
Destination: 10.13.2.34 (10.13.2.34)
Internet Control Message Protocol
Type: 11 (Time-to-live exceeded)
Code: 0 (TTL equals 0 during transit)
Checksum: 0x0c88 (correct)
Data (140 bytes)
04500 001c 9e1b 0000 0211 0348 0a0d 0222E.........H..."
100a0d 0133 9292 829b 0008 d341 0000 0000...3.......A....
200000 0000 0000 0000 0000 0000 0000 0000................
300000 0000 0000 0000 0000 0000 0000 0000................
400000 0000 0000 0000 0000 0000 0000 0000................
500000 0000 0000 0000 0000 0000 0000 0000................
600000 0000 0000 0000 0000 0000 0000 0000................
700000 0000 0000 0000 0000 0000 0000 0000................
802000 0df3 0008 0101 0002 d101..........

Frame 4 (186 on wire, 186 captured)
Ethernet II
Destination: 00:03:fd:1c:86:84 (Cisco_1c:86:84)
Source: 00:04:4e:7a:74:00 (Cisco_7a:74:00)
Type: MPLS label switched packet (0x8847)
MultiProtocol Label Switching Header
MPLS Label: Unknown (28)
MPLS Experimental Bits: 6
MPLS Bottom Of Label Stack: 1
MPLS TTL: 254
Internet Protocol
Version: 4
Header length: 20 bytes
Time to live: 254
Protocol: ICMP (0x01)
```

```
Header checksum: 0x156f (correct)
Source: 10.13.3.134 (10.13.3.134)
Destination: 10.13.2.34 (10.13.2.34)
Internet Control Message Protocol
Type: 11 (Time-to-live exceeded)
Code: 0 (TTL equals 0 during transit)
Checksum: 0x0c88 (correct)
Data (140 bytes)
04500 001c 9e1b 0000 0211 0348 0a0d 0222E.........H..."
100a0d 0133 9292 829b 0008 d341 0000 0000...3.......A....
200000 0000 0000 0000 0000 0000 0000 0000................
300000 0000 0000 0000 0000 0000 0000 0000................
400000 0000 0000 0000 0000 0000 0000 0000................
500000 0000 0000 0000 0000 0000 0000 0000................
600000 0000 0000 0000 0000 0000 0000 0000................
700000 0000 0000 0000 0000 0000 0000 0000................
802000 0df3 0008 0101 0002 d101..........
```

Aus der **Ausgabe** des **`Frames 3` können Sie bestimmen, dass Frame 3 das ICMP-Paket von R3 nach R1 ist.** Die Quelladresse (10.13.3.134) ist die Adresse, an die das ursprüngliche Paket empfangen wird. Die ICMP-Fehlermeldung enthält die abgelaufenen Label-Informationen am Ende der Datenkomponente. Der Wert ist 0 x 02 d, d. h. 45. **`Frame 4`** ist das MPLS-Paket, das von R4 an R1 gesendet wird.

7. Nach Erhalt der ICMP-Nachricht sendet R1 ein weiteres Paket mit dem Label 47 und einem TTL 3. R2 und R3 verringern unterwegs die TTL und leiten das Paket an R4 weiter. R4 weist darauf hin, dass es sich um den beabsichtigten Empfänger handelt und dass der UDP-Datagrammport nicht erreichbar ist. Es sendet eine ICMP-`Port-`Nachricht, `die nicht erreichbar` ist, an R1 bis R3 und R2.In dieser Sniffer-Ausgabe sind wichtige Punkte in Fettschrift zu erkennen:

```
Frame 5 (60 on wire, 60 captured)
Ethernet II
Destination: 00:04:4e:7a:74:00 (Cisco_7a:74:00)
Source: 00:03:fd:1c:86:84 (Cisco_1c:86:84)
Type: IP (0x0800)
Trailer: 000000000000000000000000000000000...
Internet Protocol
Version: 4
Header length: 20 bytes
Time to live: 1
Protocol: UDP (0x11)
Header checksum: 0x0446 (correct)
Source: 10.13.2.34 (10.13.2.34)
Destination: 10.13.1.51 (10.13.1.51)
User Datagram Protocol
Source port: 37647 (37647)
Destination port: 33436 (33436)
Length: 8
Checksum: 0xd2c3 (correct)

Frame 6 (74 on wire, 74 captured)
Ethernet II
Destination: 00:03:fd:1c:86:84 (Cisco_1c:86:84)
Source: 00:04:4e:7a:74:00 (Cisco_7a:74:00)
Type: MPLS label switched packet (0x8847)
MultiProtocol Label Switching Header
MPLS Label: Unknown (28)
MPLS Experimental Bits: 6
MPLS Bottom Of Label Stack: 1
MPLS TTL: 255
```

```
Internet Protocol
Version: 4
Header length: 20 bytes
Time to live: 255
Protocol: ICMP (0x01)
Header checksum: 0x5694 (correct)
Source: 10.13.5.10 (10.13.5.10)
Destination: 10.13.2.34 (10.13.2.34)
Internet Control Message Protocol
Type: 3 (Destination unreachable)
Code: 3 (Port unreachable)
Checksum: 0x1485 (correct)
Data (28 bytes)
04500 001c 9e1d 0000 0111 0446 0a0d 0222E.........F..."
100a0d 0133 930f 829c 0008 d2c3...3........
```

Bild 5 zeigt, dass das UDP-Datagramm von R1 an R4 gesendet wird. Der Zielport-Wert im UDP-Datagramm ist 33436 (größer als 32000), wie im Abschnitt Normal Traceroute Command beschrieben.Im **Frame 6 sendet R4 einen** nicht erreichbaren ICMP-Typ und einen Port-Code ohne Erreichbarkeit an R1. Bei allen früheren ICMP-Nachrichten von R2 und R3 wurde das Typfeld als Time-to-Live-Überschreitung festgelegt. Die Ausgabe des Befehls **traceroute** wird hier angezeigt:

```
R1#traceroute
Protocol [ip]:
Target IP address: 10.13.1.51
Source address: 10.13.2.34
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]: 1
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 10.13.1.51
1 10.13.2.33 [MPLS: Label 47 Exp 0] 0 msec
2 10.13.3.134 [MPLS: Label 45 Exp 0] 0 msec
3 10.13.5.10 4 msec
R1#
```

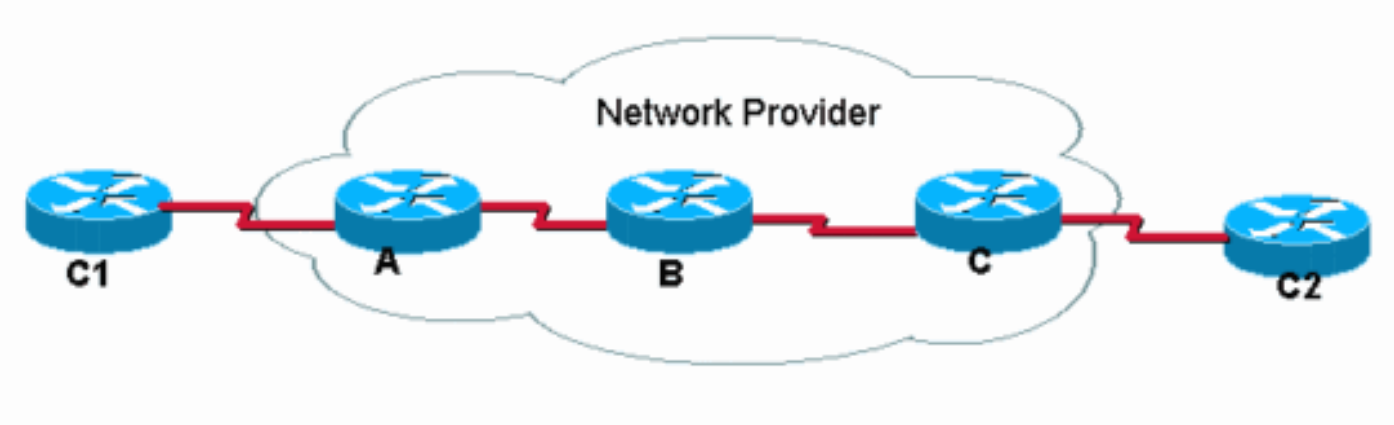
Der Befehl **traceroute** verwendet standardmäßig drei Probes für jeden TTL-Wert. Es sendet drei Pakete mit einer TTL von 1, drei Pakete mit einer TTL von 2 usw. Dieser **Traceroute**-Befehl wird mit einer einzigen Sonde ausgegeben, sodass es einfach zu verfolgen und zu debuggen. Wie in der Ausgabe zu sehen ist, zeigt der Befehl **traceroute** auch den Wert des abgelaufenen Labels an.

no mpls ip propagate-ttl-Befehl

Wenn Sie MPLS konfigurieren, wird ein Label vom Label Switch Router (LSR) auferlegt, wenn ein IP-Paket an die MPLS-Domäne weitergeleitet wird. Diese Bezeichnung muss einen Wert im TTL-Feld enthalten. Standardmäßig liest LSR das TTL-Feld im IP-Header des eingehenden Pakets, setzt es um 1 herab und kopiert das übrige Feld in das TTL-Feld des MPLS-Headers. Die Core-LSRs betrachten nur das oberste Label. Wenn der TTL-Wert nicht 0 erreicht, wird das Paket weitergeleitet. Der Egress-Edge-LSR, der das Label aufhebt, kopiert die im Label-TTL-Feld verbleibenden Elemente in das TTL-Feld des IP-Headers und leitet das IP-Paket dann außerhalb der MPLS-Domäne weiter.

Dieses Verhalten kann mit dem Konfigurationsbefehl **no mpls ip propagate-ttl** geändert werden. Der Eingangs-Edge-LSR verwendet den Wert 255 als TTL-Wert im Label, wenn er angewendet wird. Der Egress-Edge-LSR kopiert den TTL-Wert des Labels beim Beenden des Labels nicht in den IP-Header. Das Ergebnis ist, dass der IP-Header-TTL die Hops im MPLS-Core nicht widerspiegelt. Wenn Kunden also eine **Traceroute** von einer Seite ihres Netzwerks zur anderen durchführen, werden die Router im MPLS-Core-Netzwerk nicht in den **Traceroute-**Informationen angezeigt. Es ist wichtig, die TTL-Propagierung in den Eingangs- und Ausgangs-Edge-LSRs zu deaktivieren. Andernfalls hat der IP-Header möglicherweise einen höheren Wert, wenn er die MPLS-Domäne verlässt, als er es bei seiner Eingabe hatte.

Hier ein Beispiel:



C1 führt eine **Traceroute** zu C2 aus. Bei der standardmäßigen IP-TTL-Weiterleitungsoperation sieht der **Traceroute** in C1 wie folgt aus:

```
C1#traceroute C2.cust.com

Tracing the route to C2.cust.com
  1 A.provider.net       44 msec  36 msec  32 msec
  2 B provider.net      164 msec  132 msec  128 msec
  3 C.provider.net148 msec  156 msec  152 msec
  4 C2.cust.com         180 msec  *  181 msec
```

Diese Ausgabe veranschaulicht das typische **Traceroute-**Verhalten in einem MPLS-Netzwerk. Da der Label-Header eines bezeichneten Pakets den TTL-Wert des ursprünglichen IP-Pakets überträgt, werden die Routen in den Pfad-Drop-Paketen, für die die TTL überschritten wird, angegeben. Daher zeigt **Traceroute** alle Router im Pfad an. Das Verhalten ist:

1. Das erste Paket ist ein IP-Paket mit TTL gleich 1. Router A senkt die TTL und verwirft das Paket, weil es 0 erreicht. Eine ICMP-TTL-Überschreitungsmeldung wird an die Quelle gesendet.
2. Das zweite gesendete Paket ist ein IP-Paket mit TTL gleich 2. Router A senkt die TTL, kennzeichnet das Paket und leitet es an Router B weiter.
3. Router B senkt den TTL-Wert im MPLS-Header, verwirft das Paket und sendet eine ICMP TTL-Überschreitung-Nachricht an die Quelle. Da es sich um ein MPLS-Paket handelte, das verworfen wurde, muss die Rücksendeadresse für die ICMP-Nachricht von der Quelladresse im IP-Header im MPLS-Paket abgeleitet werden. Diese IP-Adresse ist Router B jedoch möglicherweise nicht bekannt. Router B leitet die ICMP-Nachrichten über denselben Label Switched Path (LSP) weiter, den das verworfene Paket weiterleitet (in Richtung Router C). Am Ende des LSP wird das Label entfernt und die ICMP-Nachrichten werden entsprechend der Zieladresse im IP-Header (in Richtung Router C1) weitergeleitet.

4. Das dritte Paket (TTL ist 3) verarbeitet ähnliche Pakete wie die vorherigen Pakete. Allerdings verwirft Router C nun das Paket, basierend auf der TTL im IP-Header. Router B wurde aufgrund von vorletzten Hop-Popping zuvor aus dem Label entfernt, und die TTL wurde in den IP-Header kopiert.
5. Das vierte Paket (TTL = 4) erreicht das endgültige Ziel, an dem die TTL des IP-Headers überprüft wird.

Wenn die IP-TTL-Propagierung im globalen Konfigurationsmodus mit dem Befehl **no mpls ip propagate-ttl** deaktiviert ist, wird der TTL-Wert nicht in den IP-Header kopiert, und die **Traceroute** in C1 bis C2 sieht wie folgt aus:

```
C1#traceroute C2.cust.com

Tracing the route to C2.cust.com
  1 A.provider.net   44 msec  36 msec  32 msec
  2 C2.cust.com     180 msec  *  181 msec
```

Wenn der Befehl **traceroute** in dieser Situation verwendet wird, werden die ICMP-Antworten nur von Routern empfangen, die echte TTL im IP-Header sehen. In dieser Situation führt Router C1 einen **Traceroute-**Befehl aus (wie abgebildet), aber die Core-Router kopieren die TTL nicht in das und vom Label. Es führt zu diesem Verhalten:

1. Das erste Paket ist ein IP-Paket mit TTL gleich 1. Router A senkt die TTL, verwirft das Paket und sendet eine ICMP TTL-Überschreitung-Nachricht an die Quelle.
2. Das zweite Paket ist ein IP-Paket mit TTL gleich 2. Router A senkt die TTL, kennzeichnet das Paket und legt die TTL im MPLS-Header auf 255 fest.
3. Router B senkt die TTL im MPLS-Header auf 254, entfernt das MPLS-Label und kopiert den TTL-Wert im MPLS-Header in das TTL-Feld des IP-Headers.
4. Router C senkt die IP-TTL und sendet das Paket an den nächsten Hop-Router C2. Das Paket hat das endgültige Ziel erreicht.

Zugehörige Informationen

- Erläuterungen zu Ping- und Traceroute-Befehlen
- mpls ip propagate-ttl-Befehl
- Support-Seite für MPLS-Technologie
- Technischer Support - Cisco Systems