El comando traceroute en MPLS

Contenido

Introducción

Este documento explica cómo funciona el comando traceroute en un entorno MPLS (Multiprotocol Label Switching).

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Conocimiento básico de MPLS

Consulte Preguntas Frecuentes sobre MPLS para Principiantes para obtener más información.

Componentes Utilizados

Este documento no tiene restricciones específicas en cuanto a versiones de software y de hardware.

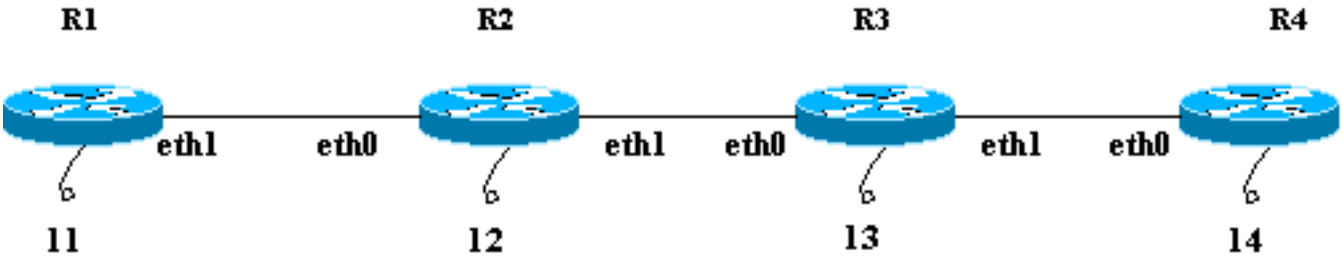
Convenciones

Consulte Convenciones de Consejos TécnicosCisco para obtener más información sobre las convenciones del documento.

Comando Normal Traceroute

Esta sección describe cómo funciona un comando traceroute tradicional. Este diagrama muestra una configuración de proveedor de servicios en la que el router 1 (R1) y el router 4 (R4) son

routers periféricos del proveedor (PE) y el router 2 (R2) y el router 3 (R3) son routers del proveedor (P).



Este ejemplo dirige un comando traceroute al loopback 14 del R4 desde el R1. R1 utiliza un datagrama UDP (protocolo de datagramas de usuario) con un valor de puerto de destino arbitrario mayor que 32000. Si selecciona un valor tan alto para el número de puerto, se asegura de que dicho puerto no exista en el destinatario deseado. Coloca este datagrama en un paquete IP.

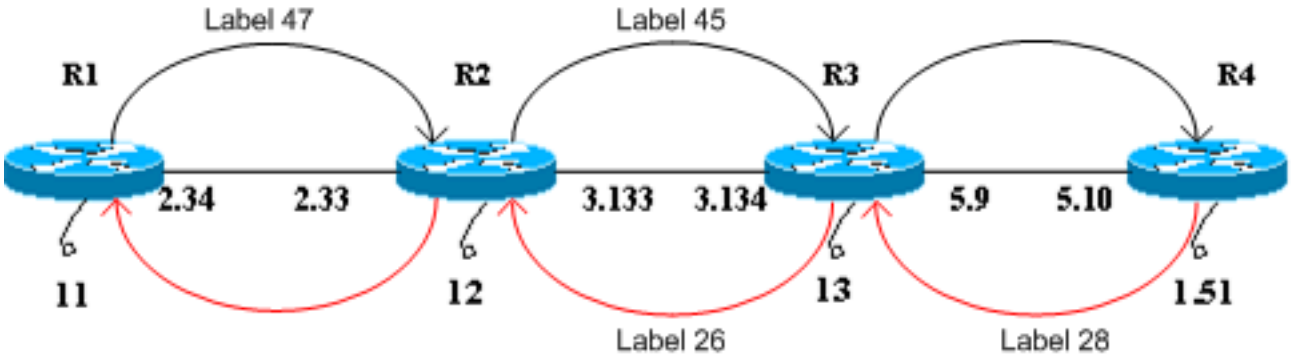
Nota: A lo largo de este documento, cada vez que se menciona un paquete IP, es un paquete IP que contiene el datagrama UDP.

Esta es una secuencia de eventos para un comando normal **traceroute**:

1. R1 envía el paquete IP con una dirección de destino de 14 y un Tiempo de vida (TTL) de 1 a través de su interfaz eth1.
2. R2 recibe el paquete y observa que no es el destinatario deseado y que el TTL del paquete es 1. Descarta el paquete y envía un mensaje de protocolo de mensajes de control de Internet (ICMP) caducado por TTL a R1. La dirección de origen de este mensaje ICMP es la dirección IP del R2 eth0 (la dirección de la interfaz que recibió el paquete original).
3. Al recibir el mensaje ICMP, R1 envía otro paquete IP destinado a 14 con un TTL de 2 a través de su interfaz eth1.
4. R2 recibe el paquete y observa que no es el destinatario deseado y que se puede alcanzar al destinatario deseado a través de R3. Reduce el TTL (de 2 a 1) y reenvía el paquete a R3. R3 recibe el paquete y ve que no es el destinatario deseado. El TTL es 1. Descarta el paquete y envía un mensaje ICMP caducado TTL a R1 con su dirección eth0 como dirección de origen.
5. R1 recibe el mensaje ICMP y envía otro paquete IP a 14 a través de su interfaz eth1 con un valor TTL de 3. En el camino, R2 y R3 disminuyen el TTL y lo pasan a R4. R4 recibe el paquete, constata que es el destinatario previsto e intenta conectarse con el valor de puerto en el datagrama UDP. R4 encuentra que este puerto no existe y envía un mensaje de error `puerto` ICMP inalcanzable a R1.Como antes, la dirección de origen de este mensaje ICMP es eth0 de R4. El programa **traceroute** ahora tiene todos los mensajes de error ICMP con las direcciones de origen correspondientes y tiene la ruta completa al destino.

Comando MPLS traceroute

Considere este mismo escenario detallado en la sección Comando normal traceroute, excepto que todos los routers, R1 a R4, ahora realizan switching de etiquetas en lugar de reenvío de IP. Se muestra la configuración del banco de pruebas en este diagrama. Todas las interfaces mostradas en el banco de pruebas se encuentran en la red 10.13.0.0.



A los efectos del presente documento, suponga que:

- R1 utiliza una etiqueta de 47 para alcanzar R4 y reenvía los paquetes a R2.
- R2 utiliza una etiqueta de 45 para alcanzar R4 y reenvía los paquetes a R3.
- R3 salta la etiqueta y reenvía el paquete a R4.
- R4 utiliza una etiqueta de 28 para alcanzar R1 y reenvía paquetes a R3.
- R3 utiliza una etiqueta de 26 para alcanzar R1 y reenvía los paquetes a R2.
- R2 muestra la etiqueta y reenvía el paquete a R1.

Estos pasos muestran la secuencia de eventos para realizar un **traceroute** desde R1 al loopback R4 10.13.1.51.

1. R1 envía un paquete conmutado etiqueta con una etiqueta de 47 y un TTL de 1 al R2. El campo TTL del paquete IP se copia en el campo TTL del encabezado de etiqueta.
2. R2 ve que no es el destinatario deseado y que el TTL es 1. Descarta el paquete y crea un mensaje ICMP caducado por TTL, como lo haría para un paquete IP normal. En este caso, el paquete de mensaje ICMP se genera por extensiones ICMP para MPLS.
3. R2 agrega la etiqueta 47 (la etiqueta entrante que ha caducado) al mensaje ICMP. No envía el paquete directamente a R1. En su lugar, consulta su base de información de reenvío de etiquetas (LFIB) y encuentra que debe utilizar una etiqueta de 45 para los paquetes recibidos con una etiqueta de 47. Pone una etiqueta de 45 en el paquete y envía el mensaje ICMP caducado TTL a R3.
4. R3 salta la etiqueta y la envía a R4. R4 ve que el destino es R1, le da una etiqueta de 28 al mensaje y lo envía a través de R3 y R2 hacia R1.
5. El mensaje de error de ICMP viaja hasta el otro extremo antes de ser enviado de regreso a R1. Este ejemplo proporciona una ilustración:

Los paquetes rastreados en la interfaz Ethernet en R4 confirman los pasos 1-5. En la salida del sniffer, `Frame 1` es el paquete entrante y `Frame 2` es el paquete saliente de R4. La salida está formateada para reflejar esta discusión y los puntos destacados se encuentran en negrita.

```
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 0002 f101...........

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...........
```

En el **Frame 1** de la salida, el primer paquete recibido por R4 es el mensaje ICMP caducado

por TTL de R2 (10.13.2.33, la interfaz en la que se recibió el paquete original) a R1 (10.13.2.34). En la parte de datos del mensaje ICMP, en bytes 0x89 y el primer bloque de 0x8A, la etiqueta MPLS (20 bytes) caduca y su valor es 0x02F, o 47. Esta es la etiqueta entrante del paquete con un TTL de 1. R2 agrega esta etiqueta en el mensaje de error ICMP.En **el Frame 2** de la salida, el `tipo` se muestra como `paquete conmutado de etiquetas MPLS`, lo que significa que es un paquete MPLS. R4 coloca una etiqueta de 28 a Frame 1 y la reenvía a R1 a través de la trayectoria de switching de etiquetas. El encabezado MPLS en la trama está en negrita. Además, si se refiere a la parte TTL del paquete, en el Frame 1 su valor es 254 y en el Frame 2 es 253. R4 lo ha reducido en 1.

6. R1 recibe el mensaje ICMP y envía otro paquete con una etiqueta de 47 y un TTL de 2 a R2. R2 cambia etiquetas, disminuye TTL (de 2 a 1) y reenvía a R3. Al igual que en el Paso 2, R3 envía un mensaje ICMP caducado por TTL agregado con la etiqueta entrante que caducó a R4, y R4 luego lo envía de vuelta a R1.La salida del sabueso en R4, que se muestra aquí, confirma el Paso 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..........
```

Desde la salida de **Frame 3**, puede determinar que Frame 3 es el paquete ICMP de R3 a R1. La dirección de origen (10.13.3.134) es la dirección en la que se recibe el paquete original. El mensaje de error ICMP contiene la información de etiqueta vencida al final de la porción de datos. Su valor es 0x02d, que es 45. **La trama 4** es el paquete MPLS que se envía de R4 a R1.

7. Al recibir el mensaje ICMP, R1 envía otro paquete con una etiqueta de 47 y un TTL de 3. En su camino, R2 y R3 disminuyen el TTL y reenvían el paquete a R4. El R4 observa que es el receptor deseado y encuentra al puerto de datagrama UDP inalcanzable. Envía un mensaje de puerto ICMP inalcanzable a R1 a R3 y R2.En este resultado del sabueso, los puntos importantes que se deben tener en cuenta son los siguientes:

```
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: 00000000000000000000000000000000...
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........
```

La trama 5 muestra que el datagrama UDP es enviado por R1 a R4. El valor del puerto de destino en el datagrama UDP es 33436 (mayor que 32000), como se describe en la sección Comando de traceroute normal.En **Frame 6**, R4 envía un `tipo ICMP de destino inalcanzable` y un código `puerto inalcanzable` a R1. Todos los mensajes ICMP anteriores de R2 y R3 tenían el campo de tipo configurado como `tiempo de vida excedido`. A continuación, se muestra el resultado del comando extended traceroute:

```
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#
```

De forma predeterminada, el comando **traceroute** utiliza tres sondas para cada valor TTL. Envía tres paquetes con un TTL de 1, tres paquetes con un TTL de 2, y así sucesivamente. Este comando **traceroute** se ejecuta con una sola sonda, por lo que es fácil rastrear y depurar. Como se ve en el resultado, el comando **traceroute** muestra también el valor de etiqueta caducado.

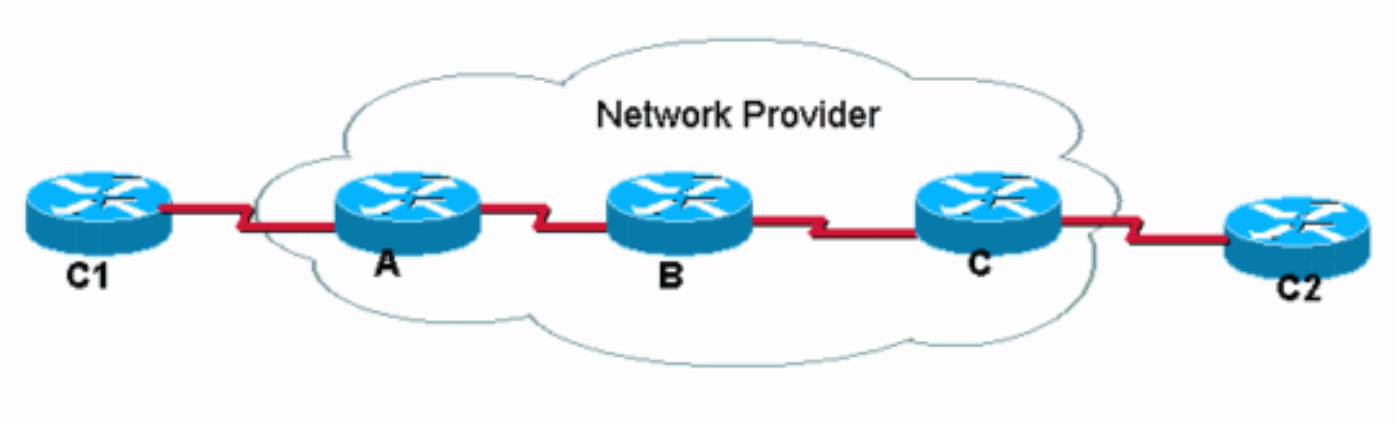
Comando no mpls ip propagate-ttl

Cuando configura MPLS, el router de switch de etiquetas (LSR) impone una etiqueta cuando se reenvía un paquete IP al dominio MPLS. Esta etiqueta debe tener un valor en el campo TTL. De forma predeterminada, el LSR lee el campo TTL en el encabezado IP del paquete entrante, lo disminuye en 1 y copia lo que queda en el campo TTL del encabezado MPLS. Los LSR de núcleo sólo miran la etiqueta superior. Si el valor TTL no alcanza 0, se reenvía el paquete. El LSR de borde de egreso que elimina las etiquetas copia lo que quedó en el campo TTL de etiquetas en el campo TTL del encabezado IP y luego, reenvía el paquete IP fuera del dominio MPLS.

Este comportamiento se puede cambiar con el comando de configuración no mpls ip propagate-ttl. El borde de ingreso LSR usa el valor 255 como el valor TTL en la etiqueta al imponerlo. El LSR

de borde de egreso no copia el valor TTL de la etiqueta en un encabezado IP cuando dispara la etiqueta.B El resultado neto es que el encabezado IP TTL no refleja los saltos tomados a través del núcleo MPLS; por lo tanto, cuando los clientes realizan un **traceroute** de un lado de su red a otro, los routers en la red de núcleo MPLS no aparecen en la **información traceroute**. Es importante inhabilitar la propagación TTL tanto en los LSR del borde de entrada como de salida. De lo contrario, el encabezado IP puede tener un valor más alto cuando deja el dominio MPLS que cuando lo ingresó.

Aquí tiene un ejemplo:



C1 realiza un **traceroute** a C2. Con la operación de propagación TTL de IP predeterminada, el **traceroute** en C1 tiene el siguiente aspecto:

```
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
```

Este resultado ilustra el comportamiento típico de la ruta de rastreo en una red MPLS. Como el encabezado de etiqueta de un paquete etiquetado transporta el valor TTL del paquete IP original, las rutas en los paquetes de descarte de trayectoria para los cuales se excede el TTL. En consecuencia, la ruta de seguimiento muestra todos los routers en el trayecto. El comportamiento es el siguiente:

1. El primer paquete es un paquete IP con TTL igual a 1. El Router A disminuye el TTL y descarta el paquete porque alcanza 0. Se envía un mensaje ICMP TTL excedido al origen.
2. El segundo paquete enviado es un paquete IP con TTL igual a 2. El Router A disminuye el TTL, etiqueta el paquete y lo reenvía al Router B.
3. El router B disminuye el valor de TTL en el encabezado de MPLS, abandona el paquete y envía un mensaje ICMP TTL-exceeded a la fuente. Ya que lo que se eliminó era un paquete MPLS, la dirección de retorno para el mensaje ICMP debe obtenerse desde la dirección de origen en el encabezado IP dentro del paquete MPLS. Pero es posible que el router B no conozca esa dirección IP, por lo que el router B reenvía los mensajes ICMP a lo largo de la misma ruta conmutada de etiquetas (LSP) que el paquete descartado estaba viajando (en la dirección hacia el router C). Al final del LSP, se elimina la etiqueta y los mensajes ICMP se reenvían según la dirección de destino en el encabezado IP (hacia el router C1).
4. El tercer paquete (TTL es 3) experimenta un procesamiento similar al de los paquetes anteriores, excepto que el router C ahora es el que coloca el paquete, basado en el TTL del

encabezado IP. El Router B, debido a la explosión del penúltimo salto, ha eliminado previamente la etiqueta y el TTL se copió en el encabezado IP.

5. El cuarto paquete (TTL es 4) alcanza el destino final donde se evalúa el TTL del encabezado de IP.

Si la propagación TTL IP se inhabilita con el comando no mpls ip propagate-ttl en el modo de configuración global, el valor TTL no se copia en el encabezado IP y el **traceroute** en C1 a C2 se muestra de la siguiente manera:

```
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
```

Cuando se utiliza el comando **traceroute** en esta situación, las respuestas ICMP sólo se reciben de los routers que ven TTL real almacenado en el encabezado IP. En esta situación, el Router C1 está ejecutando un comando **traceroute** (como se muestra), pero los routers de núcleo no copian el TTL a y desde la etiqueta. Esto da como resultado este comportamiento:

1. El primer paquete es un paquete IP con TTL igual a 1. El Router A disminuye el TTL, descarta el paquete y envía un mensaje de ICMP TTL excedido al origen.
2. El segundo paquete es un paquete IP con TTL igual a 2. El Router A disminuye el TTL, etiqueta el paquete y establece el TTL en el encabezado MPLS en 255.
3. El Router B disminuye el TTL en el encabezado MPLS a 254, quita la etiqueta MPLS y copia el valor TTL en el encabezado MPLS en el campo TTL del encabezado IP.
4. El Router C disminuye el TTL IP y envía el paquete al Router C2 de salto siguiente. El paquete ha llegado al destino final.

Información Relacionada

- Comprensión de los comandos ping y traceroute
- Comando mpls ip propagate-ttl
- Página de Soporte de Tecnología MPLS
- Soporte Técnico - Cisco Systems