

# De opdracht Traceroute in MPLS

## Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Conventies](#)

[Normale traceroute-opdracht](#)

[MPLS-routeropdracht](#)

[geen MPLS ip propagate-tl opdracht](#)

[Gerelateerde informatie](#)

## Inleiding

Dit document legt de manier uit waarop de opdracht **traceroute** werkt in een MPLS-omgeving (Multiprotocol Label Switching).

## Voorwaarden

### Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Basiskennis van MPLS

Raadpleeg [MPLS FAQ for Beginners](#) voor meer informatie.

### Gebruikte componenten

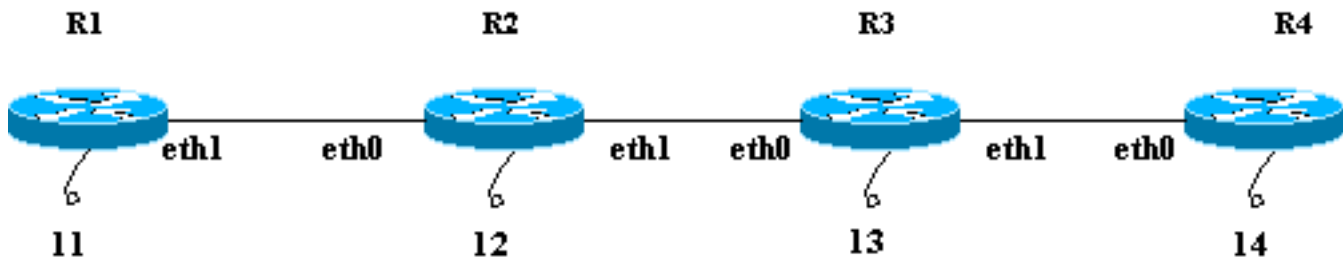
Dit document is niet beperkt tot specifieke software- en hardware-versies.

### Conventies

Raadpleeg [Cisco Technical Tips Conventions \(Conventies voor technische tips van Cisco\) voor meer informatie over documentconventies.](#)

## Normale traceroute-opdracht

Deze sectie beschrijft hoe een traditioneel tracerouteopdracht werkt. Dit diagram toont een instelling van een serviceprovider waar router 1 (R1) en router 4 (R4) routers van providers zijn en router 2 (R2) en router 3 (R3) de routers van providers (P).



Dit voorbeeld voert een **traceroute** naar de R4 loopback 14 uit R1. R1 gebruikt een User Datagram Protocol (UDP) datagram met een willekeurige waarde van de doelpoort groter dan 32000. Als u zo'n hoge waarde voor het havenaantal selecteert, waarborgt dit dat een dergelijke haven niet op de geplande ontvanger bestaat. Dit datagram wordt in een IP-pakket gezet.

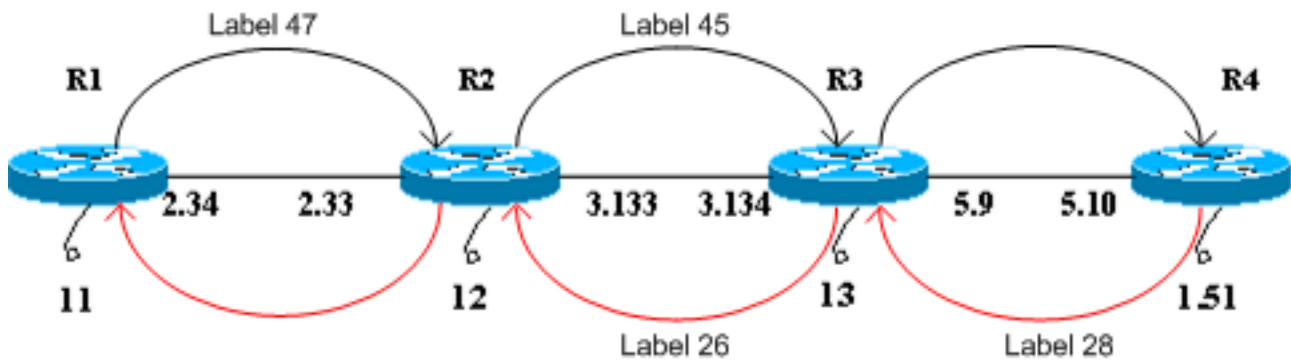
**N.B.:** In dit document is wanneer een IP-pakket wordt vermeld, het een IP-pakket dat het UDP-datagram bevat.

Dit is een opeenvolging van gebeurtenissen voor een normaal bevel **van** tracoute:

1. R1 stuurt het IP-pakket met een doeladres van 14 en een Time to Live (TTL) van 1 door middel van de eth1-interface.
2. R2 ontvangt het pakket en merkt op dat het niet de bedoelde ontvanger is en dat de TTL van het pakket 1 is. Het daalt het pakket en stuurt een TTL-verlopen bericht van de Internet Control Message Protocol (ICMP) naar R1. Het bronadres van dit ICMP-bericht is het IP-adres van de R2 eth0 (het adres van de interface die het oorspronkelijke pakket heeft ontvangen).
3. Na ontvangst van het ICMP-bericht, verstuurt R1 een ander IP-pakket dat voor 14 met een TTL van 2 is bestemd via zijn eth1-interface.
4. R2 ontvangt de verpakking en merkt op dat deze niet de beoogde ontvanger is en dat de bedoelde ontvanger via R3 kan worden bereikt. De TTL (van 2 tot 1) wordt verlaagd en de verpakking naar R3 wordt doorgestuurd. R3 ontvangt de verpakking en ziet erop toe dat deze niet de bedoelde ontvanger is. Het TTL is 1. Het druppelt het pakket en stuurt een terugverlopen ICMP bericht naar R1 met zijn eth0 adres als bronadres.
5. R1 ontvangt het ICMP-bericht en stuurt een ander IP-pakket naar 14 via zijn eth1-interface met een TTL-waarde van 3. Onderweg neemt R2- en R3-decrement de TTL af en geeft deze door aan R4. R4 krijgt het pakket, ziet dat het de bedoelde ontvanger is en probeert verbinding te maken met de poortwaarde in het UDP-datagram. R4 vindt deze poort niet en verstuurt een ICMP `poort onbereikbare` foutmelding naar R1. Zoals tevoren is het bronadres van dit ICMP-bericht eth0 of R4. Het tracerouteprogramma heeft nu alle ICMP-foutmeldingen met de corresponderende bronadressen en heeft de volledige route naar de bestemming.

## [MPLS-routeropdracht](#)

Denk aan dit zelfde scenario gedetailleerd in de sectie van de [Normal Tracker](#) het Gezicht, behalve alle routers, R1 door R4, nu etiket omschakeling in plaats van IP door te sturen. De instellingen van de testbank worden in dit schema weergegeven. Alle interfaces die in de testbank worden getoond, zijn in het 10.13.0.0 netwerk.



Voor de toepassing van dit document wordt ervan uitgegaan dat:

- R1 gebruikt een etiket van 47 om R4 te bereiken en brengt pakketten naar R2 door.
- R2 gebruikt een etiket van 45 om R4 te bereiken en brengt pakketten naar R3 door.
- R3 popt het label in en stuurt het pakket naar R4.
- R4 gebruikt een etiket van 28 om R1 te bereiken en brengt pakketten naar R3 door.
- R3 gebruikt een etiket van 26 om R1 te bereiken en brengt pakketten naar R2 door.
- R2 doet het label op en stuurt het pakje naar R1.

Deze stappen tonen de opeenvolging van gebeurtenissen om een **traceroute** van R1 naar de R4 loopback 10.13.1.51 uit te voeren.

1. R1 verstuurt een label-switched pakket met een label van 47 en een TTL-kwaliteit van 1 tot R2. Het TTL-veld van het IP-pakket wordt gekopieerd naar het TTL-veld van de label-header.
2. R2 ziet dat het niet de bedoelde ontvanger is en TTL 1 is. Het druppelt het pakket af en maakt een TTL-verlopen ICMP bericht, zoals het voor een regelmatig IP-pakket zou doen. In dit geval wordt het ICMP-berichtpakket per ICMP-uitbreidingen voor MPLS gegenereerd.
3. R2 voegt het label 47 (het inkomende etiket dat is verlopen) toe aan het ICMP-bericht. Het wordt niet rechtstreeks naar R1 verzonden. In plaats daarvan raadpleegt zij haar etiketteringsinformatiebasis (LFIB) en vindt zij dat zij een label van 45 moet gebruiken voor pakketten die worden ontvangen met een label van 47. Het brengt een etiket van 45 op de verpakking en stuurt het TTL-verlopen ICMP-bericht naar R3.
4. R3 brengt het etiket op en stuurt het naar R4. R4 ziet dat de bestemming R1 is, geeft een etiket van 28 aan het bericht en stuurt het door R3 en R2 naar R1.
5. De ICMP-foutmelding reist helemaal naar het andere uiteinde voordat ze wordt teruggestuurd naar R1. Dit voorbeeld geeft een illustratie:



De ingekapselde pakketten op de Ethernet interface bij R4 bevestigen stappen 1-5. In de uitvoer van het snuifvat is **Frame 1** het inkomende pakket en **Frame 2** is het uitgaande pakket van R4. De uitvoer is geformatteerd om deze discussie te weerspiegelen, en de punten om op te merken zijn in vet.

**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.....

In **Frame 1** van de uitvoer is het eerste pakket dat door R4 wordt ontvangen het TTL-

verlopen ICMP-bericht van R2 (10.13.2.33, de interface waarop het oorspronkelijke pakket werd ontvangen) naar R1 (10.13.2.34). In het gegevensgedeelte van het ICMP-bericht, bij bytes 0x89 en de eerste kiestoon van 0x8A, is het MPLS-label (20 bytes) verlopen en de waarde is 0x02F of 47. Dit is het inkomende label van het pakket met een TTL van 1. R2 voegt dit label toe aan de ICMP-foutmelding. In **Frame 2** van de uitvoer wordt het type weergegeven als MPLS-label met elkaar geschakeld, wat betekent dat het een MPLS-pakket is. R4 zet een etiket van 28 aan Frame 1 en zendt het naar R1 door het label-switching pad. De MPLS-kop in het frame is vet. Ook, als u naar het TTL gedeelte van het pakket verwijst, is in Frame 1 zijn waarde 254, en in Frame 2 is het 253. R4 heeft het met 1 vermindert.

6. R1 ontvangt het ICMP-bericht en verstuurt een ander pakket met een label van 47 en een TTL van 2 tot R2. R2-swaps, stappen TTL (van 2 tot 1) en doorsturen naar R3. Net zoals in Stap 2 verstuurt R3 een TTL-verlopen ICMP-bericht dat is toegevoegd aan R4 en R4 stuurt het dan terug naar R1. De uitvoer van de sluipschutter bij R4, die hier wordt getoond, bevestigt Stap 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.....

```

Van de uitvoer van **Frame 3** kunt u bepalen dat **Frame 3** het ICMP-pakket is van R3 naar R1. Het bronadres (10.13.3.134) is het adres waarop het oorspronkelijke pakket is ontvangen. De ICMP-foutmelding bevat de verlopen labelinformatie aan het einde van het gegevensgedeelte. Zijn waarde is 0x02d, wat 45 is. **Frame 4** is het MPLS-pakket dat van R4 naar R1 wordt verzonden.

7. Na ontvangst van het ICMP-bericht stuurt R1 een andere verpakking met een label van 47 en een TTL van 3. Onderweg neemt R2 en R3 de TTL af en stuurt het pakket naar R4. R4 merkt op dat het de bedoelde ontvanger is en vindt de UDP datagram poort onbereikbaar. Het stuurt een **onbereikbaar** bericht van ICMP naar R1 door R3 en R2. In deze sector zijn belangrijke punten die moeten worden opgetekend met een donkere blik:

```

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.....

```

**Frame 5** toont dat het UDP-datagram door R1 naar R4 wordt verstuurd. De waarde van de doelpoort in het UDP-datagram is 33436 (groter dan 32000), zoals besproken in de sectie [Normal traceroute](#). In **Frame 6**, verstuurt R4 een bestemming onbereikbaar ICMP type en een poort onbereikbare code naar R1. Alle eerdere ICMP berichten van R2 en R3 hadden het type veld ingesteld als tijd-aan-leven overschreden. De output van het uitgebreide tracerouteopdracht wordt hier getoond:

```

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#

```

Standaard gebruikt de opdracht **traceroute** drie sondes voor elke TTL-waarde. Ze sturen drie pakketten met een TTL van 1, drie pakketten met een TTL van 2, enzovoort. Dit tracoute bevel wordt uitgegeven met één enkele sonde, dus is het gemakkelijk te traceren en te debug. Zoals in de uitvoer wordt gezien, toont de opdracht **Tracoute** ook de verlopen waarde van het etiket.

## [geen MPLS ip propagate-tl opdracht](#)

Wanneer u MPLS vormt, wordt een label opgelegd door de labelrouter switch (LSR) wanneer een IP-pakket naar het MPLS-domein wordt doorgestuurd. Dit label moet een waarde in het TTL-veld hebben. Standaard leest LSR het TTL-veld in de IP-header van het inkomende pakket, wordt deze met 1 verlaagd en kopieert de tekst die achterblijft in het TTL-veld van de MPLS-header. De kern-LSR's kijken alleen naar het bovenste label. Als de TTL-waarde niet 0 wordt bereikt, wordt het pakket verzonden. De rand LSR die het label opslaat kopieert wat in het veld label TTL blijft, naar het TTL-veld van de IP-header en stuurt het IP-pakket naar buiten het MPLS-domein.

Dit gedrag kan worden gewijzigd met de configuratieopdracht [no mpls ip propagate-tl](#). De inbraakrand LSR gebruikt de waarde van 255 als de TTL-waarde in het label bij het instellen van

de waarde. De rand LSR kopieert de waarde van het label niet naar de IP-kop wanneer u het label opneemt. Het nettoresultaat is dat de IP-header TTL niet de hop weergeeft die over de MPLS-kern wordt genomen; dus wanneer klanten een **traceroute** van één kant van hun netwerk aan een andere kant uitvoeren, verschijnen de routers in het MPLS kernnetwerk niet in de informatie van **Tracoute**. Het is belangrijk om TTL-propagatie uit te schakelen in zowel de inloop- als bovenrand LSR's. Anders heeft de IP-kop mogelijk een hogere waarde wanneer het het MPLS-domein verlaat dan wanneer de header ingevoerd wordt.

Dit is een voorbeeld:



C1 voert een **traceroute** aan C2 uit. Met de standaard IP TTL-propagatiehandeling lijkt de **traceroute** in C1 op dit:

```
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
```

Deze output illustreert typisch traceroutedgedrag in een netwerk MPLS. Aangezien de labelkop van een geëtiketteerd pakket de TTL-waarde van het oorspronkelijke IP-pakket draagt, vallen de routes in de pad-pakketten waarvoor de TTL wordt overschreden. Daarom toont Tracoute alle routers in het pad. Het gedrag is:

1. Het eerste pakket is een IP-pakket met TTL gelijk aan 1. Router A daalt de TTL en daalt het pakket omdat het 0 bereikt. Een ICMP TTL-overschreden bericht wordt naar de bron verzonden.
2. Het tweede verzonden pakket is een IP-pakket met TTL gelijk aan 2. De router A vermindert de TTL, etiketteert het pakket en zendt het pakket naar router B.
3. Router B verlaagt de TTL-waarde in de MPLS-header, druppelt het pakket en stuurt een TTL-overschreden ICMP-bericht naar de bron. Aangezien het een MPLS-pakket was dat is gevallen, moet het adres voor het ICMP-bericht worden afgeleid van het bronadres in de IP-header in het MPLS-pakket. Maar dat IP-adres is in feite misschien niet bekend bij Router B, dus stuurt router B de ICMP-berichten door langs hetzelfde label switched pad (LSP) dat het gedropte pakket onderweg was (in de richting van router C). Aan het eind van het LSP wordt het etiket verwijderd en de ICMP-berichten worden doorgestuurd volgens het doeladres in de IP-kop (naar router C1).
4. Het derde pakket (TTL is 3) ervaart gelijke verwerking zoals de vorige pakketten, behalve dat



Router C nu het pakket druppelt, gebaseerd op de TTL in de IP header. Router B, vanwege voorlaatste hoppoppen, verwijderde eerder het label en de TTL werd gekopieerd in de IP header.

5. Het vierde pakket (TTL is 4) bereikt de eindbestemming waar het TTL van de IP-header wordt onderzocht.

Als IP TTL-propagatie uitgeschakeld is met het opdracht [no mpls ip propagate-ttl in de mondiale configuratiemodus](#), wordt de TTL-waarde niet gekopieerd in de IP-header en **traceroute** in C1 tot C2 ziet er zo uit:

```
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
```

Wanneer het bevel **traceroute** in deze situatie wordt gebruikt, worden de antwoorden van ICMP slechts ontvangen van die routers die echte TTL zien opgeslagen in de IP header. In deze situatie, voert de router C1 een bevel van **het tracoute** uit (zoals getoond), maar de kernrouters kopiëren niet de TTL naar en van het etiket. Het resultaat in dit gedrag:

1. Het eerste pakje is een IP-pakket met TTL gelijk aan 1. Router A is in stappen van de TTL, daalt het pakket en stuurt een TTL-overschreden ICMP-bericht naar de bron.
2. Het tweede pakje is een IP-pakket met TTL gelijk aan 2. Router A neemt het TTL-pakket af, labelt het pakket en stelt de TTL in de MPLS-header in op 255.
3. Router B verlaagt de TTL in de MPLS-header tot 254, verwijdert het MPLS-label en kopieert de TTL-waarde in de MPLS-header naar het TTL-veld van de IP-header.
4. De router C beslist de IP TTL en stuurt het pakket naar de volgende hoprouter C2. Het pakket heeft de eindbestemming bereikt.

## [Gerelateerde informatie](#)

- [De opdrachten Ping en Traceroute begrijpen](#)
- [MPLS IP-propagatie voor RTL-opdracht](#)
- [MPLS-ondersteuningspagina voor technologie](#)
- [Technische ondersteuning - Cisco-systemen](#)