Informazioni sui comandi ping e traceroute

Sommario

Introduzione

In questo documento viene descritto l'uso dei comandi ping e traceroute sui Cisco Router.

Prerequisiti

Requisiti

Nessun requisito specifico previsto per questo documento.

Componenti usati

Il documento può essere consultato per tutte le versioni software o hardware.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

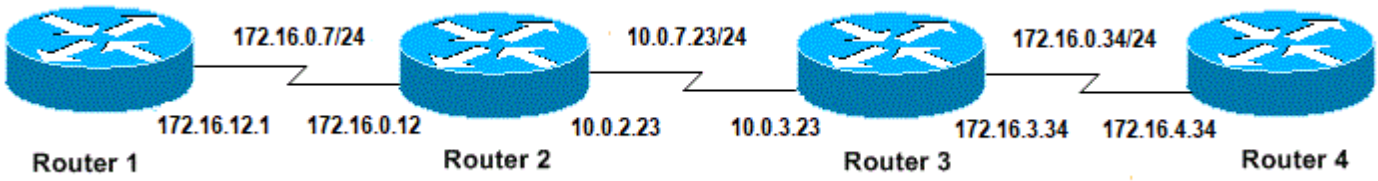
Convenzioni

Per ulteriori informazioni sulle convenzioni usate, consultare il documento Cisco sulle convenzioni nei suggerimenti tecnici.

Premesse

Nota: qualsiasi comando di debug su un router di produzione può causare problemi gravi. Prima di impartire un comando di debug leggere la sezione Uso del comando di debug.

In questo documento gli esempi si basano su questa configurazione di base:



Configurazione di base di IP e router

Il comando ping

Il comando ping viene usato spesso per risolvere i problemi di accessibilità dei dispositivi. A tal fine, usa una serie di messaggi Echo ICMP (Internet Control Message Protocol) per stabilire:

- Se un host remoto è attivo o non attivo.
- Il ritardo di andata e ritorno usato per comunicare con l'host.
- La perdita di pacchetti.

Il comando ping invia un pacchetto di richiesta echo a un indirizzo, quindi attende una risposta. Il ping ha esito positivo solo se:

- la richiesta echo raggiunge la destinazione e
- la destinazione è in grado di ottenere una risposta echo alla sorgente entro un tempo predeterminato chiamato timeout. Sui router Cisco il valore predefinito di questo timeout è due secondi.

Il valore TTL di un pacchetto ping non può essere modificato.

Il prossimo esempio di codice mostra il comando ping dopo aver abilitato il comando debug ip packet detail.

Avviso: quando il comando debug ip packet detail viene usato su un router di produzione può aumentare l'utilizzo della CPU e causare un grave calo delle prestazioni o

un'interruzione della rete.

<#root>

```
Router1#

debug ip packet detail

IP packet debugging is on (detailed)

Router1#

ping 172.16.0.12


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/6/8 ms

Router1#
Jan 20 15:54:47.487: IP: s=172.16.12.1 (local), d=172.16.0.12 (Serial0), len 100,
  sending
Jan 20 15:54:47.491:

ICMP type=8

, code=0

!--- This is the ICMP packet 172.16.12.1 sent to 172.16.0.12.
!--- ICMP type=8 corresponds to the echo message.

Jan 20 15:54:47.523: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 100,
  rcvd 3
Jan 20 15:54:47.527:

ICMP type=0

, code=0

!--- This is the answer we get from 172.16.0.12. !--- ICMP type=0 corresponds to the echo reply message
!--- By default, the repeat count is five times, so there will be five
!--- echo requests, and five echo replies.
```

Possibili valori di tipo ICMP

Tipo ICMP	Valore letterale
0	echo-reply
3	destination unreachable code 0 = net unreachable 1 = host unreachable 2 = protocol unreachable 3 = port unreachable 4 = fragmentation needed, and DF set 5 = source route failed
4	source-quench
5	redirect code 0 = redirect datagrams for the network 1 = redirect datagrams for the host 2 = redirect datagrams for the type of service and network 3 = redirect datagrams for the

	type of service and host
6	alternate-address
8	echo
9	router-advertisement
10	router-solicitation
11	time-exceeded code 0 = time to live exceeded in transit 1 = fragment reassembly time exceeded
12	parameter-problem
13	timestamp-request
14	timestamp-reply
15	information-request
16	information-reply
17	mask-request
18	mask-reply
31	conversion-error
32	mobile-redirect

Possibili caratteri di output della funzione ping

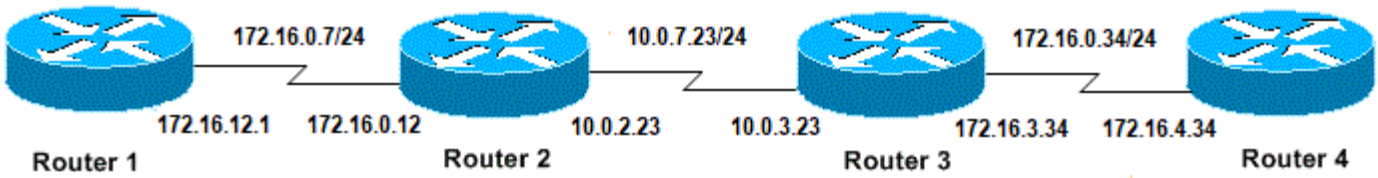
Carattere	Descrizione
!	Il punto esclamativo indica che è stata ricevuta una risposta.
.	Il punto indica che il tempo di attesa di una risposta da parte del server di rete è scaduto.
U	È stata ricevuta una PDU di errore per "destinazione non raggiungibile".
Q	Richiesta di rallentamento dell'origine (destinazione occupata).
M	Impossibile frammentare.
?	Tipo di pacchetto sconosciuto.
&	Durata del pacchetto superata.

Impossibile eseguire il ping

Se non è possibile eseguire correttamente il ping su un indirizzo IP, considerare le cause elencate in questa sezione.

Problema del router

Di seguito sono riportati alcuni esempi di tentativi di ping non riusciti che possono causare il problema, e le operazioni da eseguire per risolverlo. Questo esempio è accompagnato dal diagramma della topologia della rete:



Problemi del router

<#root>

Router1#

```
!
interface Serial0
ip address 172.16.12.1 255.255.255.0
no fair-queue
clockrate 64000
!
```

Router2#

```
!
interface Serial0
ip address 10.0.2.23 255.255.255.0
no fair-queue
clockrate 64000
!
interface Serial1
ip address 172.16.0.12 255.255.255.0
!
```

Router3#

```
!
interface Serial0
ip address 172.16.3.34 255.255.255.0
no fair-queue
!
interface Serial1
ip address 10.0.3.23 255.255.255.0
!
```

Router4#

```
!
interface Serial0
ip address 172.16.4.34 255.255.255.0
no fair-queue
clockrate 64000
!
```

Provare a eseguire il ping del Router4 dal Router1:

<#root>

```
Router1#
```

ping 172.16.4.34

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:
.....
```

Success rate is 0 percent (0/5)

Risultati:

<#root>

```
Router1#
```

debug ip packet

```
IP packet debugging is on
```

Avviso: quando il comando debug ip packet viene usato su un router di produzione può aumentare l'utilizzo della CPU e causare un grave calo delle prestazioni o un'interruzione della rete.

<#root>

```
Router1#
```

ping 172.16.4.34

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:

Jan 20 16:00:25.603: IP: s=172.16.12.1 (local), d=172.16.4.34, len 100, unroutable.
Jan 20 16:00:27.599: IP: s=172.16.12.1 (local), d=172.16.4.34, len 100, unroutable.
Jan 20 16:00:29.599: IP: s=172.16.12.1 (local), d=172.16.4.34, len 100, unroutable.
Jan 20 16:00:31.599: IP: s=172.16.12.1 (local), d=172.16.4.34, len 100, unroutable.
Jan 20 16:00:33.599: IP: s=172.16.12.1 (local), d=172.16.4.34, len 100, unroutable.
Success rate is 0 percent (0/5)
```

Dal momento che sul Router1 non vengono eseguiti protocolli di routing, il router non può sapere dove inviare il pacchetto e genera il messaggio "unroutable" (non indirizzabile).

Aggiungere un routing statico al Router1:

<#root>

```
Router1#

configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#

ip route 0.0.0.0 0.0.0.0 Serial0
```

Risultati:

<#root>

```
Router1#

debug ip packet detail

IP packet debugging is on (detailed)

Router1#

ping 172.16.4.34

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)

Jan 20 16:05:30.659: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
  sending
Jan 20 16:05:30.663:     ICMP type=8, code=0
Jan 20 16:05:30.691: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 56,
  rcvd 3
Jan 20 16:05:30.695:     ICMP type=3, code=1
Jan 20 16:05:30.699: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
  sending
Jan 20 16:05:30.703:     ICMP type=8, code=0
Jan 20 16:05:32.699: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
  sending
Jan 20 16:05:32.703:     ICMP type=8, code=0
Jan 20 16:05:32.731: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 56,
  rcvd 3
Jan 20 16:05:32.735:     ICMP type=3, code=1
Jan 20 16:05:32.739: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
  sending
Jan 20 16:05:32.743:     ICMP type=8, code=0
```

Esaminare il problema del Router2:

<#root>

```
Router2#

debug ip packet detail

IP packet debugging is on (detailed)

Router2#
Jan 20 16:10:41.907: IP: s=172.16.12.1 (Serial1), d=172.16.4.34, len 100, unroutable
Jan 20 16:10:41.911:     ICMP type=8, code=0
Jan 20 16:10:41.915: IP: s=172.16.0.12 (local), d=172.16.12.1 (Serial1), len 56, sending
Jan 20 16:10:41.919:

ICMP type=3, code=1

Jan 20 16:10:41.947: IP: s=172.16.12.1 (Serial1), d=172.16.4.34, len 100, unroutable
Jan 20 16:10:41.951:     ICMP type=8, code=0
Jan 20 16:10:43.943: IP: s=172.16.12.1 (Serial1), d=172.16.4.34, len 100, unroutable
Jan 20 16:10:43.947:     ICMP type=8, code=0
Jan 20 16:10:43.951: IP: s=172.16.0.12 (local), d=172.16.12.1 (Serial1), len 56, sending
Jan 20 16:10:43.955:     ICMP type=3, code=1
Jan 20 16:10:43.983: IP: s=172.16.12.1 (Serial1), d=172.16.4.34, len 100, unroutable
Jan 20 16:10:43.987:     ICMP type=8, code=0
Jan 20 16:10:45.979: IP: s=172.16.12.1 (Serial1), d=172.16.4.34, len 100, unroutable
Jan 20 16:10:45.983:     ICMP type=8, code=0
Jan 20 16:10:45.987: IP: s=172.16.0.12 (local), d=172.16.12.1 (Serial1), len 56, sending
Jan 20 16:10:45.991:     ICMP type=3, code=1
```

Il Router1 ha inviato correttamente i suoi pacchetti al Router2, ma il Router2 non sa come accedere all'indirizzo 172.16.4.34. Il Router2 invia un messaggio "unreachable ICMP" (ICMP non raggiungibile) al Router1.

Abilitare il protocollo RIP (Routing Information Protocol) sul Router2 e sul Router3:

```
Router2#
router rip
 network 172.16.0.7
 network 10.0.7.23
Router3#
router rip
 network 10.0.7.23
 network 172.16.0.34
```

Risultati:

<#root>

```
Router1#

debug ip packet

IP packet debugging is on

Router1#

ping 172.16.4.34
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:

Jan 20 16:16:13.367: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending.
Jan 20 16:16:15.363: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending.
Jan 20 16:16:17.363: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending.
Jan 20 16:16:19.363: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending.
Jan 20 16:16:21.363: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending.
Success rate is 0 percent (0/5)
```

Il Router1 invia i pacchetti al Router4, ma il Router4 non risponde.

Possibile problema sul Router4:

<#root>

```
Router4#
```

debug ip packet

```
IP packet debugging is on

Router4#
Jan 20 16:18:45.903: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
  rcvd 3
Jan 20 16:18:45.911: IP: s=172.16.4.34 (local), d=172.16.12.1, len 100,
```

unroutable

```
Jan 20 16:18:47.903: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
  rcvd 3
Jan 20 16:18:47.907: IP: s=172.16.4.34 (local), d=172.16.12.1, len 100, unroutable
Jan 20 16:18:49.903: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
  rcvd 3
Jan 20 16:18:49.907: IP: s=172.16.4.34 (local), d=172.16.12.1, len 100, unroutable
Jan 20 16:18:51.903: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
  rcvd 3
Jan 20 16:18:51.907: IP: s=172.16.4.34 (local), d=172.16.12.1, len 100, unroutable
Jan 20 16:18:53.903: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
  rcvd 3
Jan 20 16:18:53.907: IP: s=172.16.4.34 (local), d=172.16.12.1, len 100, unroutable
```

Il Router4 riceve i pacchetti ICMP e prova a rispondere all'indirizzo 172.16.12.1, ma poiché non ha un routing verso questa rete, il tentativo non riesce.

Aggiungere un routing statico al Router4:

<#root>

```
Router4(config)#
ip route 0.0.0.0 0.0.0.0 Serial0
```

Ora entrambi i dispositivi possono comunicare fra loro:

```
<#root>
Router1#
ping 172.16.4.34

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/35/36 ms
```

Interfaccia non attiva

In questa situazione l'interfaccia smette di funzionare. Il prossimo esempio mostra un tentativo di eseguire il ping del Router4 dal Router1:

```
<#root>
Router1#
ping 172.16.4.34

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
```

Poiché il routing è corretto, eseguire una procedura di risoluzione del problema progressiva. Provare a eseguire il ping del Router2:

```
<#root>
Router1#
ping 172.16.0.12

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
```

Secondo l'esempio precedente, il problema si trova tra il Router2 e il Router3. Probabilmente l'interfaccia seriale sul Router3 è stata arrestata:

<#root>

```
Router3#
```

show ip interface brief

```
Serial0   172.16.3.34    YES manual up                      up
Serial1   10.0.3.23    YES manual administratively down   down
```

Questo problema è semplice da risolvere:

<#root>

```
Router3#
```

configure terminal

```
Enter configuration commands, one per line.  End with CNTL/Z.
Router3(config)#
```

interface serial1

```
Router3(config-if)#
```

no shutdown

```
Router3(config-if)#
Jan 20 16:20:53.900: %LINK-3-UPDOWN: Interface Serial1, changed state to up
Jan 20 16:20:53.910: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1,
  changed state to up
```

Comando access-list

In questo scenario viene autorizzato solo il traffico telnet in ingresso sul Router4 dall'interfaccia Serial0.

<#root>

```
Router4(config)#
```

access-list 100 permit tcp any any eq telnet

```
Router4(config)#
```

interface serial0

```
Router4(config-if)#
```

```
ip access-group 100 in


Router1#
configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#
access-list 100 permit ip host 172.16.12.1 host 172.16.4.34
Router1(config)#
access-list 100 permit ip host 172.16.4.34 host 172.16.12.1
Router1(config)#
end
Router1#
debug ip packet 100


IP packet debugging is on
Router1#
debug ip icmp
ICMP packet debugging is on
```

Provare a eseguire il ping del Router4:

<#root>

```
Router1#
ping 172.16.4.34


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.34, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)

Jan 20 16:34:49.207: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending
Jan 20 16:34:49.287: IP: s=172.16.4.34 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:34:49.291: ICMP: dst (172.16.12.1)
administratively prohibited unreachable
 rcv from 172.16.4.34
Jan 20 16:34:49.295: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending
Jan 20 16:34:51.295: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending
Jan 20 16:34:51.367: IP: s=172.16.4.34 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
```

```
Jan 20 16:34:51.371: ICMP: dst (172.16.12.1) administratively prohibited unreachable
 rcv from 172.16.4.34
Jan 20 16:34:51.379: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100,
 sending
```

Alla fine di un comando access-list c'è sempre un comando deny all (rifiuta tutto) implicito. Ciò significa che i pacchetti ICMP che entrano nell'interfaccia 0 seriale sul router4 vengono rifiutati e il router 4 invia un messaggio ICMP "administrative ban unreachable" (proibito dall'amministratore come non raggiungibile) all'origine del pacchetto originale, come mostrato nel messaggio di debug. La soluzione consiste nell'aggiungere questa riga al comando access-list:

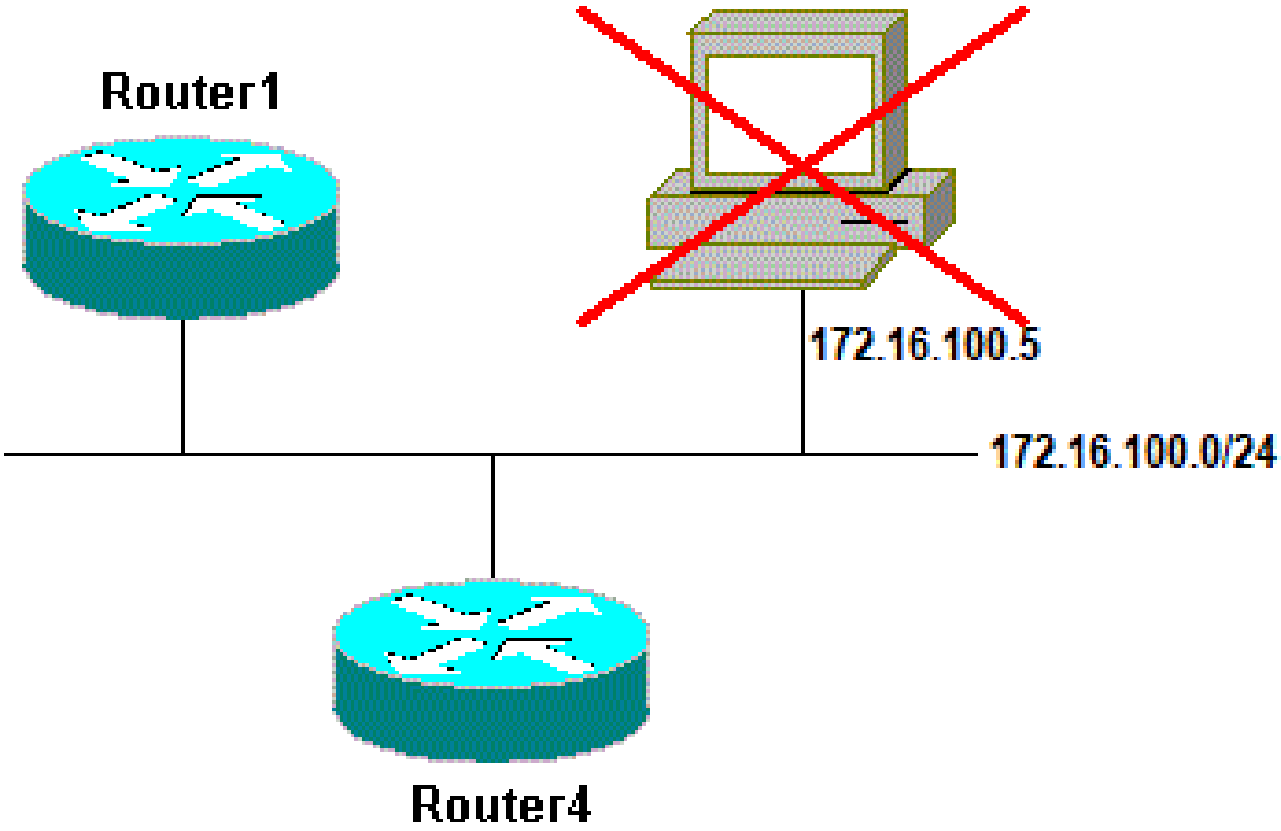
<#root>

```
Router4(config)#
```

access-list 100 permit icmp any any

Problema relativo al protocollo ARP (Address Resolution Protocol)

In questo scenario, questa è la connessione Ethernet:



Problema relativo al protocollo ARP (Address Resolution Protocol)

<#root>

```
Router4#
```

ping 172.16.100.5

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.100.5, timeout is 2 seconds:

Jan 20 17:04:05.167: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  sending
Jan 20 17:04:05.171: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
```

encapsulation failed

```
.
Jan 20 17:04:07.167: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  sending
Jan 20 17:04:07.171: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  encapsulation failed.
Jan 20 17:04:09.175: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  sending
Jan 20 17:04:09.183: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  encapsulation failed.
Jan 20 17:04:11.175: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  sending
Jan 20 17:04:11.179: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  encapsulation failed.
Jan 20 17:04:13.175: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  sending
Jan 20 17:04:13.179: IP: s=172.16.100.4 (local), d=172.16.100.5 (Ethernet0), len 100,
  encapsulation failed.
Success rate is 0 percent (0/5)
Router4#
```

In questo esempio, il ping non funziona e la causa segnalata nel messaggio è "encapsulation failed" (errore di incapsulamento). Questo significa che il router sa su quale interfaccia deve inviare il pacchetto ma non sa come farlo. In questo caso, è necessario capire come funziona il protocollo ARP (Address Resolution Protocol).

Il protocollo ARP viene utilizzato per mappare l'indirizzo di layer 2 (indirizzo MAC) su un indirizzo di layer 3 (indirizzo IP). Il comando show arp consente di effettuare un controllo in tal senso:

<#root>

```
Router4#
```

show arp

```
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  172.16.100.4               -   0000.0c5d.7a0d  ARPA   Ethernet0
Internet  172.16.100.7              10   0060.5cf4.a955  ARPA   Ethernet0
```

Tornare al problema che ha generato l'errore di incapsulamento, ma questa volta abilitare il

comando debug arp :

<#root>

```
Router4#

debug arp

ARP packet debugging is on

Router4#

ping 172.16.100.5



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.100.5, timeout is 2 seconds:

Jan 20 17:19:43.843: IP ARP: creating incomplete entry for IP address: 172.16.100.5
  interface Ethernet0
Jan 20 17:19:43.847: IP ARP: sent req src 172.16.100.4 0000.0c5d.7a0d,


dst 172.16.100.5 0000.0000.0000 Ethernet0.

Jan 20 17:19:45.843: IP ARP: sent req src 172.16.100.4 0000.0c5d.7a0d,
                 dst 172.16.100.5 0000.0000.0000 Ethernet0.
Jan 20 17:19:47.843: IP ARP: sent req src 172.16.100.4 0000.0c5d.7a0d,
                 dst 172.16.100.5 0000.0000.0000 Ethernet0.
Jan 20 17:19:49.843: IP ARP: sent req src 172.16.100.4 0000.0c5d.7a0d,
                 dst 172.16.100.5 0000.0000.0000 Ethernet0.
Jan 20 17:19:51.843: IP ARP: sent req src 172.16.100.4 0000.0c5d.7a0d,
                 dst 172.16.100.5 0000.0000.0000 Ethernet0.
Success rate is 0 percent (0/5)
```

L'output precedente mostra che il Router4 trasmette i pacchetti e li invia all'indirizzo di broadcast Ethernet FFFF.FFFF.FFFF. In questo caso, l'indirizzo 0000.0000.0000 indica che il Router4 cerca l'indirizzo MAC della destinazione 172.16.100.5. Poiché non conosce l'indirizzo MAC durante la richiesta ARP, in questo esempio usa l'indirizzo 0000.0000.000 come segnaposto nei frame di trasmissione inviati dall'interfaccia Ethernet 0 e chiede quale indirizzo MAC corrisponda all'indirizzo 172.16.100.5. In assenza di risposta, l'indirizzo MAC corrispondente all'indirizzo IP nell'output show arp viene contrassegnato come incompleto:

<#root>

```
Router4#

show arp

Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  172.16.100.4               -   0000.0c5d.7a0d  ARPA   Ethernet0

Internet  172.16.100.5               0   Incomplete      ARPA

Internet  172.16.100.7               2   0060.5cf4.a955  ARPA   Ethernet0
```

Dopo un periodo predeterminato, questa voce incompleta viene eliminata dalla tabella ARP. Finché l'indirizzo MAC non viene incluso nella tabella ARP, il ping non riesce a causa dell'errore di incapsulamento.

Ritardo

Per impostazione predefinita, se non si riceve una risposta dal dispositivo remoto entro due secondi, il ping ha esito negativo:

<#root>

```
Router1#

ping 172.16.0.12



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12,

timeout is 2 seconds:

.....
Success rate is 0 percent (0/5)
```

Nelle reti con collegamento lento o un lungo ritardo, due secondi non sono sufficienti. È possibile modificare questa impostazione predefinita con un ping esteso:

<#root>

```
Router1#

ping

Protocol [ip]:
Target IP address:  172.16.0.12
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:

30

Extended commands [n]:
Sweep range of sizes [n]:

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 30 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1458/2390/6066 ms
```

Per ulteriori informazioni sul comando ping esteso, vedere Informazioni sui comandi ping e

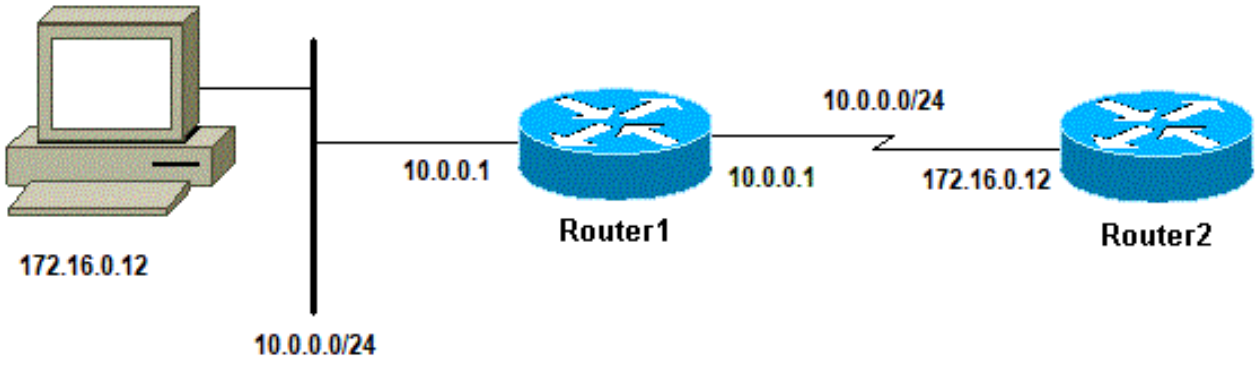
traceroute estesi.

Nell'esempio precedente, aumentando il timeout il ping è riuscito.

Nota: il tempo di andata e ritorno medio supera i due secondi.

Indirizzo di origine corretto

Questo esempio rappresenta uno scenario comune:



Indirizzo di origine corretto

Aggiungere un'interfaccia LAN sul Router1:

<#root>

```
Router1(config)#
interface ethernet0
Router1(config-if)#
ip address 10.0.0.1 255.255.255.0
```

Da una postazione sulla LAN, è possibile eseguire il ping del Router1. Dal Router1 è possibile eseguire il ping del Router2. Tuttavia, da una postazione sulla LAN, non è possibile eseguire il ping del Router2.

Dal Router1 è possibile eseguire il ping del Router2 perché, per impostazione predefinita, si utilizza l'indirizzo IP dell'interfaccia in uscita come indirizzo di origine nel pacchetto ICMP. Il Router2 non ha informazioni su questa nuova LAN. Se deve rispondere a un pacchetto proveniente da questa rete, non sa come gestirlo.

<#root>

```
Router1#
debug ip packet
```

```
IP packet debugging is on
```

Avviso: quando il comando debug ip packet viene usato su un router di produzione può aumentare l'utilizzo della CPU e causare un grave calo delle prestazioni o un'interruzione della rete.

<#root>

```
Router1#
```

ping 172.16.0.12

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/7/9 ms
Router1#

Jan 20 16:35:54.227: IP: s=172.16.12.1 (local), d=172.16.0.12 (Serial0), len 100, sending
Jan 20 16:35:54.259: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 100, rcvd 3
```

L'output dell'esempio precedente funziona perché l'indirizzo di origine del pacchetto inviato è 172.16.12.1. Per simulare un pacchetto proveniente dalla LAN, occorre usare un ping esteso:

<#root>

```
Router1#
```

ping

```
Protocol [ip]:
Target IP address: 172.16.0.12
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
```

10.0.0.1

```
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:

Jan 20 16:40:18.303: IP: s=10.0.0.1 (local), d=172.16.0.12 (Serial0), len 100,
 sending.
```

```
Jan 20 16:40:20.303: IP: s=10.0.0.1 (local), d=172.16.0.12 (Serial0), len 100,
 sending.
Jan 20 16:40:22.303: IP: s=10.0.0.1 (local), d=172.16.0.12 (Serial0), len 100,
 sending.
Jan 20 16:40:24.303: IP: s=10.0.0.1 (local), d=172.16.0.12 (Serial0), len 100,
 sending
Jan 20 16:40:26.303: IP: s=10.0.0.1 (local), d=172.16.0.12 (Serial0), len 100,
 sending.
Success rate is 0 percent (0/5)
```

Questa volta, l'indirizzo di origine è 10.0.0.1 e non funziona. I pacchetti vengono inviati, ma non si riceve alcuna risposta. Per risolvere questo problema, aggiungere un routing all'indirizzo 10.0.0.0 nel Router2. La regola di base è che il dispositivo che riceve il ping deve anche sapere come rispondere al dispositivo che lo ha inviato.

Numero elevato di pacchetti eliminati nella coda

Quando un pacchetto entra nel router, il router tenta di inoltrarlo a livello di interrupt. Se non riesce a trovare una corrispondenza in una tabella della cache appropriata, il pacchetto verrà inserito nella coda di input dell'interfaccia in ingresso ed elaborato. Alcuni pacchetti vengono sempre elaborati, ma con la configurazione appropriata e in reti stabili, la velocità di elaborazione dei pacchetti non deve mai congestionare la coda di input. Se la coda di input è piena, il pacchetto viene eliminato.

Questo avviene anche se l'interfaccia è attiva, ma un numero elevato di pacchetti eliminati nella coda di input impedisce il ping del dispositivo. È possibile controllare i pacchetti di input eliminati con il comando show interface.

<#root>

```
Router1#
```

show interface Serial0/0/0

```
Serial0/0/0 is up, line protocol is up

  MTU 1500 bytes, BW 1984 Kbit, DLY 20000 usec,
     reliability 255/255, txload 69/255, rxload 43/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:02, output 00:00:00, output hang never
  Last clearing of "show interface" counters 01:28:49
```

Input queue: 76/75/5553/0

```
 (size/max/drops/flushes);
     Total output drops: 1760
  Queueing strategy: Class-based queueing
  Output queue: 29/1000/64/1760 (size/max total/threshold/drops)
     Conversations  7/129/256 (active/max active/max total)
     Reserved Conversations 4/4 (allocated/max allocated)
     Available Bandwidth 1289 kilobits/sec
```

```
!--- Output supressed
```

Come si evince dall'output, il numero di pacchetti di input eliminati nella coda è elevato. Per la risoluzione dei problemi relativi ai pacchetti di input e di output eliminati nella coda, fare riferimento alla sezione Risoluzione dei problemi relativi ai pacchetti di input e di output eliminati nella coda.

Il comando traceroute

Il comando traceroute viene usato per rilevare i percorsi dei pacchetti quando viaggiano verso la destinazione. Il dispositivo (ad esempio, un router o un PC) invia una sequenza di datagrammi UDP (User Datagram Protocol) a un indirizzo di porta non valido sull'host remoto.

Vengono inviati tre datagrammi, ciascuno con il valore del campo Time-To-Live (TTL) impostato su uno. Un valore TTL di 1 provoca il timeout del datagramma non appena raggiunge il primo router del percorso, che a sua volta risponde con un messaggio ICMP di tipo TEM (Time Exceeded Message) per segnalare che il datagramma è scaduto.

Altri tre messaggi UDP vengono ora inviati, ciascuno con il valore TTL impostato su 2; il secondo router restituisce il messaggio ICMP di tipo TEM. Questo processo continua finché i pacchetti non raggiungono effettivamente l'altra destinazione. Poiché questi datagrammi provano ad accedere a una porta non valida sull'host di destinazione, vengono restituiti messaggi ICMP di tipo "porta non raggiungibile"; questo evento segnala che il programma Traceroute è terminato.

Lo scopo di questo comando è registrare l'origine dei messaggi ICMP di tipo "tempo scaduto" per fornire una traccia del percorso del pacchetto verso l'indirizzo di destinazione.

<#root>

```
Router1#
```

traceroute 172.16.4.34

```
Type escape sequence to abort.
Tracing the route to 172.16.4.34

  1 172.16.0.12 4 msec 4 msec 4 msec
  2 10.0.3.23 20 msec 16 msec 16 msec
  3 172.16.4.34 16 msec *  16 msec

Jan 20 16:42:48.611: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.615:     UDP src=39911, dst=
```

33434

```
Jan 20 16:42:48.635: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.639:
```

ICMP type=11, code=0

```
!--- ICMP Time Exceeded Message from Router2.

Jan 20 16:42:48.643: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.647:     UDP src=34237, dst=33435
Jan 20 16:42:48.667: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.671:     ICMP type=11, code=0
Jan 20 16:42:48.675: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.679:     UDP src=33420, dst=33436
Jan 20 16:42:48.699: IP: s=172.16.0.12 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.703:     ICMP type=11, code=0
```

Questa è la prima sequenza di pacchetti inviati con TTL=1. Il primo router, in questo caso Router2 (172.16.0.12), elimina il pacchetto e restituisce all'origine (172.16.12.1) un messaggio ICMP type=11, ossia un messaggio TEM "tempo scaduto".

<#root>

```
Jan 20 16:42:48.707: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.711:     UDP src=35734, dst=33437
Jan 20 16:42:48.743: IP: s=
```

10.0.3.23

```
 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.747:
```

ICMP type=11, code=0

```
!--- ICMP Time Exceeded Message from Router3.

Jan 20 16:42:48.751: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.755:     UDP src=36753, dst=33438
Jan 20 16:42:48.787: IP: s=10.0.3.23 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.791:     ICMP type=11, code=0
Jan 20 16:42:48.795: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.799:     UDP src=36561, dst=33439
Jan 20 16:42:48.827: IP: s=10.0.3.23 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.831:     ICMP type=11, code=0
```

Lo stesso processo si verifica sul Router3 (10.0.3.23) con TTL=2:

<#root>

```
Jan 20 16:42:48.839: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.843:     UDP src=34327, dst=33440
Jan 20 16:42:48.887: IP: s=
```

172.16.4.34

```
 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:48.891:
```

ICMP type=3, code=3

!--- Port Unreachable message from Router4.

```
Jan 20 16:42:48.895: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:48.899:     UDP src=37534, dst=33441
Jan 20 16:42:51.895: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 28,
 sending
Jan 20 16:42:51.899:     UDP src=37181, dst=33442
Jan 20 16:42:51.943: IP: s=172.16.4.34 (Serial0), d=172.16.12.1 (Serial0), len 56,
 rcvd 3
Jan 20 16:42:51.947:     ICMP type=3, code=3
```

Impostando TTL=3 è finalmente possibile raggiungere il Router4. Questa volta, poiché la porta non è valida, il Router4 reinvia al Router1 un messaggio ICMP type=3, destinazione non raggiungibile, e code=3, a indicare che la porta non è raggiungibile.

Nella tabella seguente vengono elencati i caratteri visualizzati nell'output del comando traceroute.

Caratteri di testo IP traceroute

Carattere	Descrizione
nn msec	Per ciascun nodo, il tempo di ritorno in millisecondi del numero di sonde specificato
*	Sonda scaduta
A	Non consentito a livello amministrativo (ad esempio, access-list)
Q	Richiesta di rallentamento dell'origine (destinazione occupata)
I	Test utente interrotto
U	Porta non raggiungibile
H	Host non raggiungibile
N	Rete non raggiungibile
P	Protocollo non raggiungibile
T	Timeout
?	Tipo di pacchetto sconosciuto

Prestazioni

Usando i comandi ping e traceroute è possibile ricavare il tempo di andata e ritorno (RTT, Round-Trip Time), ovvero il tempo necessario per inviare un pacchetto echo e ottenere una risposta. Può dare un'idea approssimativa del ritardo sul collegamento. Tuttavia, questi dati non sono abbastanza precisi per essere utilizzati per valutare le prestazioni.

Quando il router stesso è la destinazione del pacchetto, il pacchetto deve essere gestito dalla CPU del router. Il processore deve gestire le informazioni del pacchetto e inviare una risposta. Questo non è l'obiettivo principale di un router. Per definizione, un router è progettato per indirizzare i pacchetti e risponde al ping solo in un'ottica di best-effort.

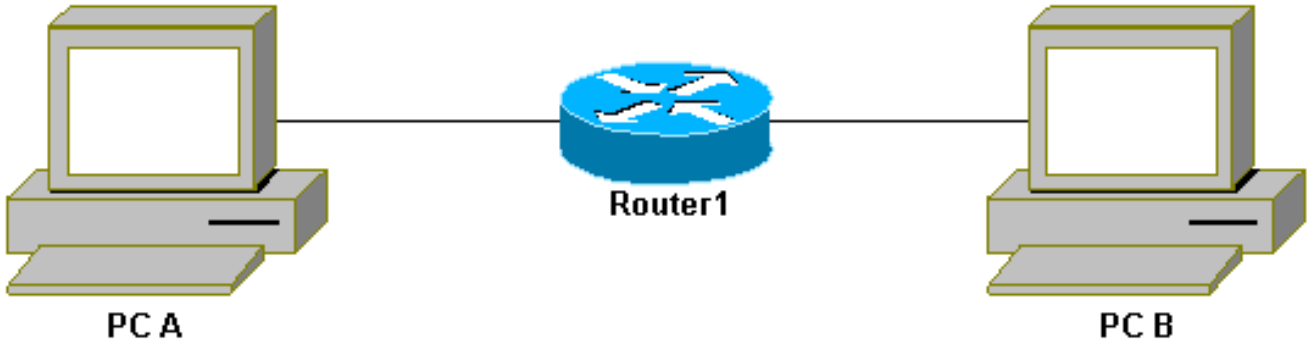
Per illustrare questa situazione, riportiamo un esempio di ping tra il Router1 e il Router2:

```
<#root>

Router1#

ping 172.16.0.12



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
```

Il tempo RTT è di circa quattro millisecondi. Dopo aver abilitato alcune funzioni ad alta intensità di processo sul Router2, provare a eseguire il ping tra il Router2 e il Router1.

```
<#root>

Router1#

ping 172.16.0.12



Type

escape sequence

 to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/25/28 ms
```

Qui il tempo RTT è aumentato notevolmente. Il Router2 è occupato e la sua priorità non è rispondere al ping. Per valutare le prestazioni del router, è preferibile prendere in considerazione il traffico che lo attraversa.



Traffico attraverso il router

Il traffico viene commutato rapidamente e gestito dal router con la massima priorità. La rete di base illustra questo processo:

Rete di base con 3 router

Eseguire il ping del Router3 dal Router1:

<#root>

```
Router1#
```

ping 10.0.3.23

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.3.23, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/32 ms
```

Il traffico passa attraverso il Router2 e viene commutato rapidamente. Abilitare la funzione ad alta intensità di processo sul Router2:

<#root>

```
Router1#
```

ping 10.0.3.23

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.3.23, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/36 ms
```

Non si nota praticamente alcuna differenza. Questo perché, sul Router2, i pacchetti vengono ora gestiti a livello di interrupt.

Uso del comando di debug

Prima di usare i comandi di debug, consultare la sezione Informazioni importanti sui comandi di debug.

I diversi comandi di debug trattati in questo documento mostrano cosa accade quando si utilizza un comando ping o traceroute. Questi comandi possono aiutare a risolvere vari problemi. Tuttavia, in un ambiente di produzione, i comandi di debug richiedono cautela. Se la CPU non è potente o deve gestire molti pacchetti, è facile che il dispositivo si arresti. Per ridurre al minimo l'effetto del comando di debug sul router esistono due metodi. Un metodo consiste nell'utilizzare gli elenchi degli accessi per limitare il traffico specifico che si desidera monitorare.

Di seguito è riportato un esempio:

<#root>

```
Router4#

debug ip packet ?

  <1-199>      Access list
  <1300-2699>  Access list (expanded range)
  detail       Print more debugging detail

Router4#

configure terminal

Router4(config)#

access-list 150 permit ip host 172.16.12.1 host 172.16.4.34

Router4(config)#^

Z


Router4#

debug ip packet 150

IP packet debugging is on for access list 150

Router4#

show debug
```

```
Generic IP:
  IP packet debugging is on for access list 150

Router4#
```

show access-list

```
Extended IP access list 150
    permit ip host 172.16.12.1 host 172.16.4.34 (5 matches)
```

Con questa configurazione, il Router4 stampa solo il messaggio di debug corrispondente all'access-list 150. Un ping proveniente dal Router1 genera questo messaggio:

```
Router4#
Jan 20 16:51:16.911: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:51:17.003: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:51:17.095: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:51:17.187: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:51:17.279: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
```

La risposta al problema non proviene dal Router4, perché questi pacchetti non corrispondono all'elenco degli accessi. Per visualizzarli, aggiungere:

<#root>

```
Router4(config)#
```

access-list 150 permit ip host 172.16.12.1 host 172.16.4.34

```
Router4(config)#
```

access-list 150 permit ip host 172.16.4.34 host 172.16.12.1

Risultati:

```
Jan 20 16:53:16.527: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:53:16.531: IP: s=172.16.4.34 (local), d=172.16.12.1 (Serial0), len 100,
 sending
Jan 20 16:53:16.627: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:53:16.635: IP: s=172.16.4.34 (local), d=172.16.12.1 (Serial0), len 100,
 sending
Jan 20 16:53:16.727: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
```

```
 rcvd 3
Jan 20 16:53:16.731: IP: s=172.16.4.34 (local), d=172.16.12.1 (Serial0), len 100,
 sending
Jan 20 16:53:16.823: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:53:16.827: IP: s=172.16.4.34 (local), d=172.16.12.1 (Serial0), len 100,
 sending
Jan 20 16:53:16.919: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
Jan 20 16:53:16.923: IP: s=172.16.4.34 (local), d=172.16.12.1 (Serial0), len 100,
 sending
```

Un altro modo per ridurre le conseguenze del comando di debug consiste nell'inserire i messaggi di debug in un buffer e visualizzarli con il comando show log dopo aver disattivato il debug:

<#root>

```
Router4#

configure terminal

Router4(config)#

no logging console

Router4(config)#

logging buffered 5000

Router4(config)#^

Z



Router4#

debug ip packet


IP packet debugging is on
Router4#

ping 172.16.12.1



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/36/37 ms

Router4#

undebug all

All possible debugging has been turned off

Router4#

show log
```

```
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: disabled
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 61 messages logged
    Trap logging: level informational, 59 message lines logged

Log Buffer (5000 bytes):

Jan 20 16:55:46.587: IP: s=172.16.4.34 (local), d=172.16.12.1 (Serial0), len 100,
 sending
Jan 20 16:55:46.679: IP: s=172.16.12.1 (Serial0), d=172.16.4.34 (Serial0), len 100,
 rcvd 3
```

I comandi ping e traceroute sono utili per risolvere i problemi di accesso della rete. Inoltre, sono molto facili da usare. I tecnici di rete li utilizzano ampiamente.

Informazioni correlate

- Informazioni sui comandi ping e traceroute estesi
- Supporto tecnico – Cisco Systems

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).