

Ping 및 Traceroute 명령 이해

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[표기 규칙](#)

[배경 정보](#)

[ping 명령](#)

[Ping할 수 없음](#)

[라우터 문제](#)

[인터페이스 중단](#)

[access-list 명령](#)

[ARP\(Address Resolution Protocol\) 문제](#)

[Delay](#)

[올바른 소스 주소](#)

[높은 입력 대기열 삭제](#)

[traceroute 명령](#)

[Performance](#)

[debug 명령 사용](#)

[관련 정보](#)

소개

이 문서에서는 Cisco 라우터에서 ping 및 **traceroute** 명령을 사용하는 방법을 설명합니다.

사전 요구 사항

요구 사항

이 문서에 대한 특정 요건이 없습니다.

사용되는 구성 요소

이 문서는 특정 소프트웨어 및 하드웨어 버전으로 한정되지 않습니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

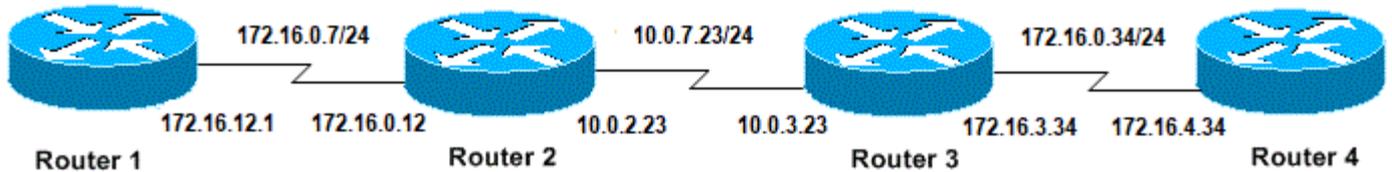
표기 규칙

문서 규칙에 대한 자세한 내용은 Cisco 기술 팁 표기 규칙을 참조하십시오.

배경 정보

참고: 프로덕션 라우터에서 사용되는 모든 디버그 명령은 심각한 문제를 일으킬 수 있습니다. 디버그 명령을 [실행하기 전에 Use the Debug Command](#) 섹션을 읽어 보십시오.

이 문서에서는 이 기본 컨피그레이션을 이 문서의 예에 사용합니다.



IP 및 라우터의 기본 구성

ping 명령

ping 명령은 디바이스의 액세스 가능성을 트러블슈팅하는 데 사용되는 매우 일반적인 방법입니다. 일련의 ICMP(Internet Control Message Protocol) 에코 메시지를 사용하여 다음을 확인합니다.

- 원격 호스트의 활성 또는 비활성 여부.
- 호스트와 통신하는 데 사용되는 왕복 지연 시간입니다.
- 패킷 손실.

ping 명령은 먼저 어떤 주소에 에코 요청 패킷을 보낸 다음 회신이 올 때까지 기다립니다. 다음과 같은 경우에만 ping이 성공합니다.

- 에코 요청이 대상에 도달하는 경우
- 대상이 시간 초과라는 미리 결정된 시간 내에 소스에 대한 에코 응답을 다시 가져올 수 있는 경우 이 시간 제한의 기본값은 Cisco 라우터에서 2초입니다.

ping 패킷의 TTL 값은 변경할 수 없습니다.

다음 코드 예제에서는 debug ip packet detail 명령이 활성화된 후 ping 명령을 보여 줍니다.

경고: debug ip packet detail 명령을 운영 라우터에서 사용하면 CPU 사용률이 높아질 수 있습니다. 따라서 심각한 성능 저하나 네트워크 중단이 발생할 수 있습니다.

```
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.
```

가능한 ICMP 유형 값

ICMP 유형

리터럴

0	echo-reply
3	목적지 도달 불가 코드 0 = 네트 도달 불가 1 = 호스트 도달 불가 2 = 프로토콜 도달 불가 3 = 포트 불가 4 = 단편화 필요, DF 세트 5 = 소스 경로 실패
4	source-quench
5	redirect code 0 = redirect datagrams for the network 1 = redirect datagrams for the host 2 = redi datagrams for the type of service and network 3 = redirect datagrams for the type of service and
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

Ping 기능에서 가능한 출력 문자

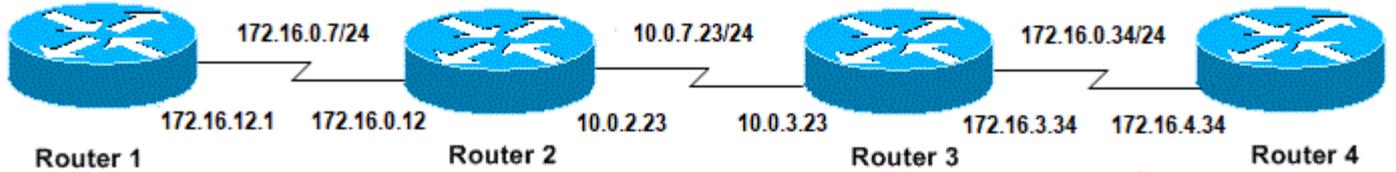
문자	설명
!	각 느낌표는 회신 수신을 나타냅니다.
.	각 기호는 네트워크 서버가 응답을 대기하는 동안 시간이 초과되었음을 나타냅니다.
U	대상에 연결할 수 없는 오류 PDU가 수신되었습니다.
Q	소스 억제(대상이 너무 많이 사용 중임)
M	프래그먼트 불가.
?	알 수 없는 패킷 유형.
&	패킷 수명 초과.

Ping할 수 없음

IP 주소에 성공적으로 ping할 수 없는 경우 이 섹션에 나열된 원인을 고려하십시오.

라우터 문제

다음은 ping 시도에 실패한 예이며, 문제를 확인할 수 있고, 문제를 해결하기 위해 수행할 작업입니다. 이 예는 다음 네트워크 토폴로지 다이어그램과 함께 표시됩니다.



라우터 문제

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
!
  
```

Router1에서 Router4를 ping해 보십시오.

```

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)
  
```

결과:

```

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

경고: debug ip packet 명령이 프로덕션 라우터에서 사용될 경우 CPU 사용률이 높아질 수 있습니다. 따라서 심각한 성능 저하나 네트워크 중단이 발생할 수 있습니다.

```
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)
```

Router1에서 실행되는 라우팅 프로토콜이 없으므로 패킷을 어디로 보낼지 모르며 "라우팅할 수 없음" 메시지가 발생합니다.

Router1에 고정 경로를 추가합니다.

```
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
```

결과:

```
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
```

Router2에서 무엇이 잘못되었는지 확인합니다.

```
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
```

Router1이 패킷을 Router2에 올바르게 보냈지만 Router2가 주소 172.16.4.34에 액세스하는 방법을 모릅니다. Router2는 Router1에 "연결할 수 없는 ICMP" 메시지를 다시 보냅니다.

Router2 및 Router3에서 RIP(Routing Information Protocol)를 활성화합니다.

```
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
```

결과:

```
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)

Router1은 패킷을 Router4로 전송하지만 Router4는 응답을 다시 전송하지 않습니다.

Router4에 가능한 문제:

```
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
```

라우터 4는 ICMP 패킷을 수신하고 172.16.12.1에 응답하려고 시도하지만 이 네트워크에 대한 경로가 없으므로 실패합니다.

Router4에 고정 경로 추가:

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

이제 양쪽이 서로 액세스할 수 있습니다.

```
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
```

인터페이스 중단

인터페이스가 더 이상 작동하지 않는 상황입니다. 다음 예에서는 Router1에서 Router4를 ping하려고 합니다.

```
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)
```

라우팅이 올바르게 단계별 문제 해결을 수행합니다. Router2에 ping을 시도합니다.

```
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
```

앞의 예에서 문제는 Router2와 Router3 간의 문제입니다. Router3의 직렬 인터페이스가 종료되었

을 수 있습니다.

```
Router3#show ip interface brief
Serial0 172.16.3.34 YES manual up up
Serial1 10.0.3.23 YES manual administratively down down
```

이는 간단하게 해결할 수 있습니다.

```
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
```

access-list 명령

이 시나리오에서는 텔넷 트래픽만 인터페이스 Serial0을 통해 Router4에 들어갈 수 있습니다.

```
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
```

Router4를 ping해 보십시오.

```
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
```

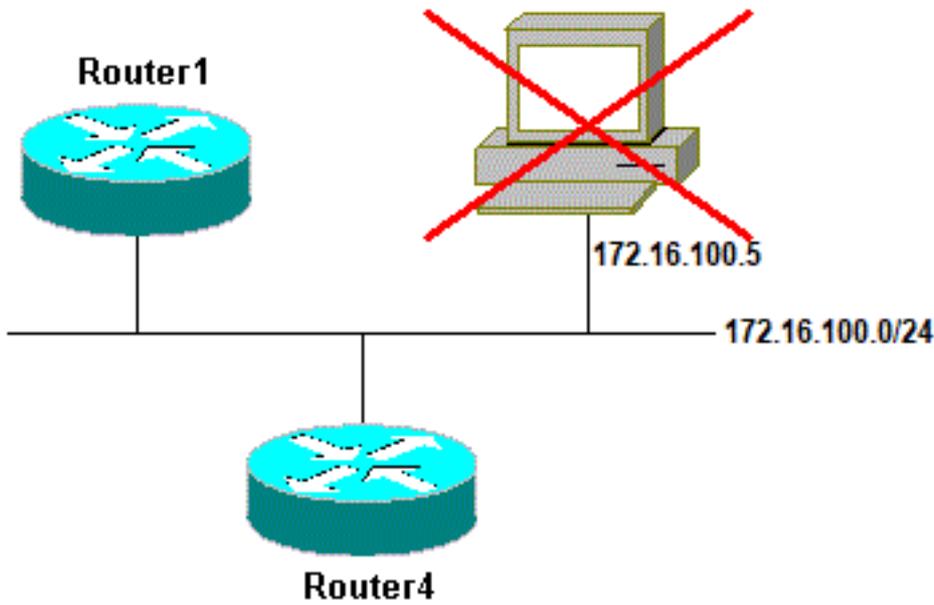
```
Jan 20 16:34:51.379: IP: s=172.16.12.1 (local), d=172.16.4.34 (Serial0), len 100, sending
```

access-list 명령의 끝에 항상 암시적으로 모두 거부 있습니다. 즉, Router4의 Serial 0 인터페이스로 들어오는 ICMP 패킷이 거부되고 Router 4가 디버그 메시지에 표시된 것처럼 원래 패킷의 소스에 ICMP "관리적으로 금지되는 도달 불가" 메시지를 보냅니다. 해결 방법은 access-list 명령에 이행을 추가하는 것입니다.

```
Router4(config)#access-list 100 permit icmp any any
```

ARP(Address Resolution Protocol) 문제

이 시나리오에서 이더넷 연결은 다음과 같습니다.



주소 확인 프로토콜 문제

```
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#
```

이 예에서는 "캡슐화 실패" 메시지로 인해 ping이 작동하지 않습니다. 즉, 라우터는 어떤 인터페이스에서 패킷을 전송해야 하는지 알고 있지만 어떻게 해야 하는지 알지 못합니다. 이 경우 ARP(Address Resolution Protocol)의 작동 방식을 이해해야 합니다.

ARP는 레이어 2 주소(MAC 주소)를 레이어 3 주소(IP 주소)에 매핑하는 데 사용되는 프로토콜입니다. **show arp** 명령을 사용하여 이를 확인할 수 있습니다.

```
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
```

"캡슐화 실패" 문제로 돌아가지만 이번에는 **debug arp** 명령을 활성화합니다.

```
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)
```

이전 출력은 Router4가 패킷을 브로드캐스트하여 이더넷 브로드캐스트 주소 FFFF.FFFF.FFFF로 전송하는 것을 보여줍니다. 여기서 0000.0000.0000은 Router4가 대상 172.16.100.5의 MAC 주소를 찾는다는 의미입니다. 이 예에서 ARP가 요청되는 동안에는 MAC 주소를 모르므로, 0000.000.000을 인터페이스 이더넷 0에서 전송된 브로드캐스트 프레임의 자리 표시자로 사용하고 172.16.100.5에 해당하는 MAC 주소를 묻습니다. 응답이 없으면 **show arp** 출력에 IP 주소에 해당하는 MAC 주소가 불완전한 것으로 표시됩니다.

```
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
```

사전에 정해진 시간이 지난 후에는 불완전한 항목이 ARP 테이블에서 제거됩니다. MAC 주소가 ARP 테이블에 없으면 "캡슐화 실패"로 인해 ping이 실패합니다.

Delay

기본적으로 2초 내에 원격 측에서 응답을 받지 못하면 ping이 실패합니다.

```
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)
```

링크 속도가 느리거나 지연 시간이 긴 네트워크에서는 2초가 충분하지 않습니다. 확장 ping을 사용하여 이 기본값을 변경할 수 있습니다.

```
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
```

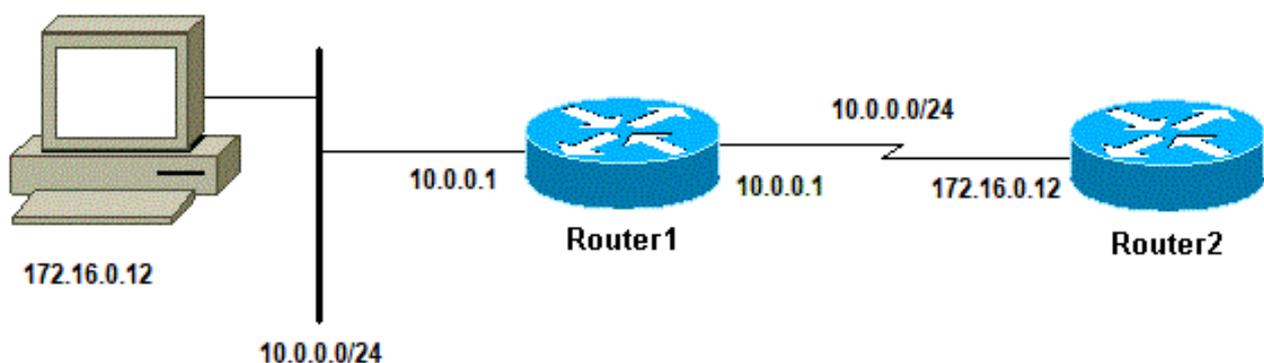
확장 ping 명령에 대한 자세한 내용은 확장 Ping [및 확장 Traceroute 명령 이해를 참조하십시오](#).

이전 예에서 시간 초과가 증가하면 ping에 성공했습니다.

참고: 평균 왕복 시간이 2초 이상이다.

올바른 소스 주소

이 예는 일반적인 시나리오입니다.



올바른 소스 주소

Router1에 LAN 인터페이스를 추가합니다.

```
Router1(config)#interface ethernet0
```

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

LAN의 스테이션에서 Router1로 ping을 수행할 수 있습니다. Router1에서 Router2로 ping을 수행할 수 있습니다. 하지만 LAN의 스테이션에서 Router2로 ping을 수행할 수 없습니다.

Router1에서 Router2로 ping을 수행할 수 있는데, 기본적으로 발신 인터페이스의 IP 주소를 ICMP 패킷의 소스 주소로 사용하기 때문입니다. 라우터2에 이 새 LAN에 대한 정보가 없습니다. 이 네트워크에서 패킷에 응답해야 하는 경우, 처리 방법을 모릅니다.

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

경고: debug ip packet 명령을 운영 라우터에서 사용하면 CPU 사용률이 높아질 수 있습니다. 따라서 심각한 성능 저하나 네트워크 중단이 발생할 수 있습니다.

```
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
```

이전 출력 예는 전송된 패킷의 소스 주소가 172.16.12.1이므로 작동합니다. LAN에서 패킷을 시뮬레이션하려면 확장 ping을 사용해야 합니다.

```
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)
```

이번에는 소스 주소가 10.0.0.1인데 작동하지 않습니다. 패킷은 전송되지만 응답은 수신되지 않습니다. 이 문제를 해결하려면 Router2의 10.0.0.0에 경로를 추가합니다. 기본 규칙은 ping된 디바이스가 ping의 소스에 응답을 보내는 방법도 알아야 합니다.

높은 입력 대기열 삭제

패킷이 라우터에 진입하면 라우터는 인터럽트 레벨에서 패킷을 전달하려고 시도합니다. 해당 캐시 테이블에서 일치하는 항목을 찾을 수 없는 경우, 패킷은 수신 인터페이스의 입력 대기열에 대기하며 처리되기를 기다립니다. 일부 패킷은 항상 처리됩니다. 그러나 처리되는 패킷 비율 때문에 입력 대기열이 혼잡해지는 일이 없도록 적절한 컨피그레이션과 안정된 네트워크를 사용하십시오. 입력 대기열이 꽉 차면 패킷이 삭제됩니다.

인터페이스가 가동 중이고 입력 대기열이 많이 삭제되어 디바이스를 ping할 수 없습니다. **show interface** 명령을 사용하여 입력 삭제를 확인할 수 있습니다.

```
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 suppressed

출력에서 표시된 것과 같이 입력 대기열 삭제가 높습니다. 입출력 대기열 [삭제 문제를 해결하려면 Troubleshoot Input Queue Drops and Output Queue Drops](#)(입력 대기열 삭제 및 출력 대기열 삭제 문제 해결)를 참조하십시오.

traceroute 명령

traceroute 명령은 패킷이 목적지로 이동할 때 실제로 사용하는 경로를 검색하는 데 사용됩니다. 디바이스(예: 라우터 또는 PC)가 UDP(User Datagram Protocol) 데이터그램 시퀀스를 원격 호스트의 잘못된 포트 주소로 전송합니다.

3개의 데이터그램을 보내는데, 각각 TTL(Time-to-Live) 필드 값이 1로 설정되어 있습니다. TTL 값이 1이면, 데이터그램이 경로의 첫 번째 라우터에 도달하는 즉시 "시간 초과"됩니다; 그런 다음 이 라우터는 데이터그램이 만료되었음을 나타내는 ICMP TEM(Time Exceeded Message)으로 응답합니다.

TTL 값이 각각 2로 설정된 또 다른 3개의 UDP 메시지가 전송되어 두 번째 라우터가 ICMP TEM을 반환합니다. 이 프로세스는 패킷이 실제로 다른 대상에 도달할 때까지 계속됩니다. 이러한 데이터그램은 대상 호스트에서 유효하지 않은 포트에 액세스하려고 시도하므로 ICMP Port Unreachable Messages가 반환되고, 이는 연결할 수 없는 포트를 나타냅니다. 이 이벤트는 Traceroute 프로그램의 종료를 나타냅니다.

그 목적은 패킷이 대상에 도달한 경로를 추적하기 위해 각 ICMP Time Exceeded Message의 소스를 기록하는 것입니다.

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
```

TTL=1로 전송되는 패킷의 첫 번째 시퀀스입니다. 첫 번째 라우터(이 경우 Router2(172.16.0.12)는 패킷을 삭제하고 소스(172.16.12.1)로 유형=11 ICMP 메시지를 다시 전송합니다. 이는 TEM(Time Exceeded Message)에 해당합니다.

```
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
```

TTL=2인 Router3(10.0.3.23)에도 동일한 프로세스가 발생합니다.

```
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
```

TTL=3이면 Router4에 마침내 연결됩니다. 이번에는 포트가 유효하지 않기 때문에, Router4는 유형=3, 목적지 도달 불가 메시지, 코드=3으로 포트 도달 불가 ICMP 메시지를 Router1로 다시 전송합니다.

다음 표에는 traceroute 명령 출력에 나타날 수 있는 문자가 나열되어 있습니다.

IP Traceroute 텍스트 문자

문자	설명
nn msec	각 노드에서 지정된 수의 프로브가 왕복하는 데 걸린 시간(밀리초)입니다.
*	프로브 시간 초과
A	관리상 금지됨(예: access-list)
Q	소스 억제(대상이 너무 많이 사용 중임)
I	사용자가 테스트를 중단
U	포트 연결 불가
H	호스트 연결 불가
네트워킹	네트워크 연결 불가
P	프로토콜 연결 불가
T	Timeout(시간 초과)
?	알 수 없는 패킷 유형

Performance

ping 및 traceroute 명령으로 RTT(왕복 시간)를 얻을 수 있습니다. 에코 패킷을 보내고 응답을 받는 데 필요한 시간입니다. 이는 링크에 대한 지연을 대략적으로 파악할 수 있습니다. 하지만 이러한 수치는 성능 평가에 사용할 정도로 정확하지 않습니다.

패킷 대상이 라우터 자체인 경우 이 패킷은 프로세스 전환되어야 합니다. 프로세서는 이 패킷의 정보를 처리하고 응답을 다시 보내야 합니다. 이는 라우터의 주요 목표가 아닙니다. 정의상 라우터는 패킷을 라우팅하기 위해 구축됩니다. 답변된 ping은 최선형 서비스로 제공됩니다.

이를 설명하기 위해 Router1에서 Router2로의 ping의 예입니다.

```
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
```

RTT는 약 4밀리초입니다. Router2에서 일부 프로세스 집약적 기능을 활성화한 후 Router1에서 Router2로 ping을 수행합니다.

```
Router1#ping 172.16.0.12
```

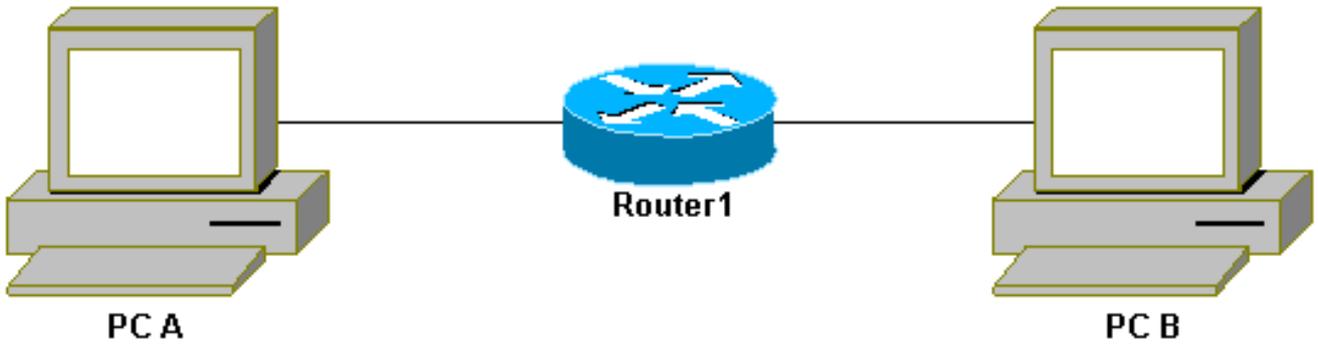
```
Type escape sequence to abort.
```

```
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
```

이 부분에서 RTT가 크게 증가했습니다. 라우터2는 상당히 사용 중이므로 ping에 응답하지 않는 것이 우선입니다. 라우터를 통과하는 트래픽을 사용하면 라우터 성능을 더 효과적으로 테스트할 수 있습니다.



라우터를 통한 트래픽

그러면 트래픽이 빠르게 전환되고 우선순위가 가장 높은 라우터에 의해 처리됩니다. 기본 네트워크에서는 이를 설명합니다.



크 3 라우터

기본 네트워크

Router1에서 Router3을 ping합니다.

```
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
```

트래픽은 Router2를 거쳐 이제 빠르게 전환됩니다. 라우터2에서 프로세스 집약적 기능을 활성화합니다.

```
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
```

차이는 거의 없습니다. Router2에서 패킷이 이제 인터럽트 레벨에서 처리되기 때문입니다.

debug 명령 사용

debug 명령을 사용하기 전에 [Debug 명령에 대한 중요 정보를 참조하십시오.](#)

이 문서에서 사용되는 다양한 debug 명령은 ping 또는 traceroute 명령을 사용했을 때 어떤 일이 발생하는지 보여줍니다. 이 명령은 문제 해결에 도움이 될 수 있습니다. 그러나 프로덕션 환경에서는 디버깅을 신중하게 사용해야 합니다. CPU가 강력하지 않거나 프로세스 전환 패킷이 많은 경우에는 디바이스가 쉽게 중단될 수 있습니다. debug 명령이 라우터에 미치는 영향을 최소화하는 몇 가지

방법이 있습니다. 한 가지 방법은 액세스 목록을 사용하여 모니터링할 특정 트래픽을 좁히는 것입니다.

예를 들면 다음과 같습니다.

```
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)
```

이 컨피그레이션에서는 Router4가 access-list 150과 일치하는 디버그 메시지만 인쇄합니다. Router1에서 ping하면 이 메시지가 표시됩니다.

```
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
```

이 패킷은 access-list와 일치하지 않으므로 이 문제에 대한 해답은 Router4에서 나오지 않습니다. 해당 항목을 보려면 다음을 추가합니다.

```
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
```

결과:

```
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
```

debug 명령의 영향을 낮추는 또 다른 방법은 debug가 꺼진 후 debug 메시지를 버퍼링하고 show log 명령으로 표시하는 것입니다.

```
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
```

ping 및 traceroute 명령은 네트워크 액세스 문제를 해결하는 데 사용할 수 있는 유용한 유틸리티입니다. 또한 사용하기 매우 쉽습니다. 이 두 명령은 네트워크 엔지니어가 널리 사용하는 명령입니다.

관련 정보

- [Extended ping 및 Extended Traceroute 명령 이해](#)
- [Technical Support - Cisco Systems](#)

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.