PROTOCOL ANALYSIS INSTITUTE

Comparing ICMP and UDP Traceroute Methods

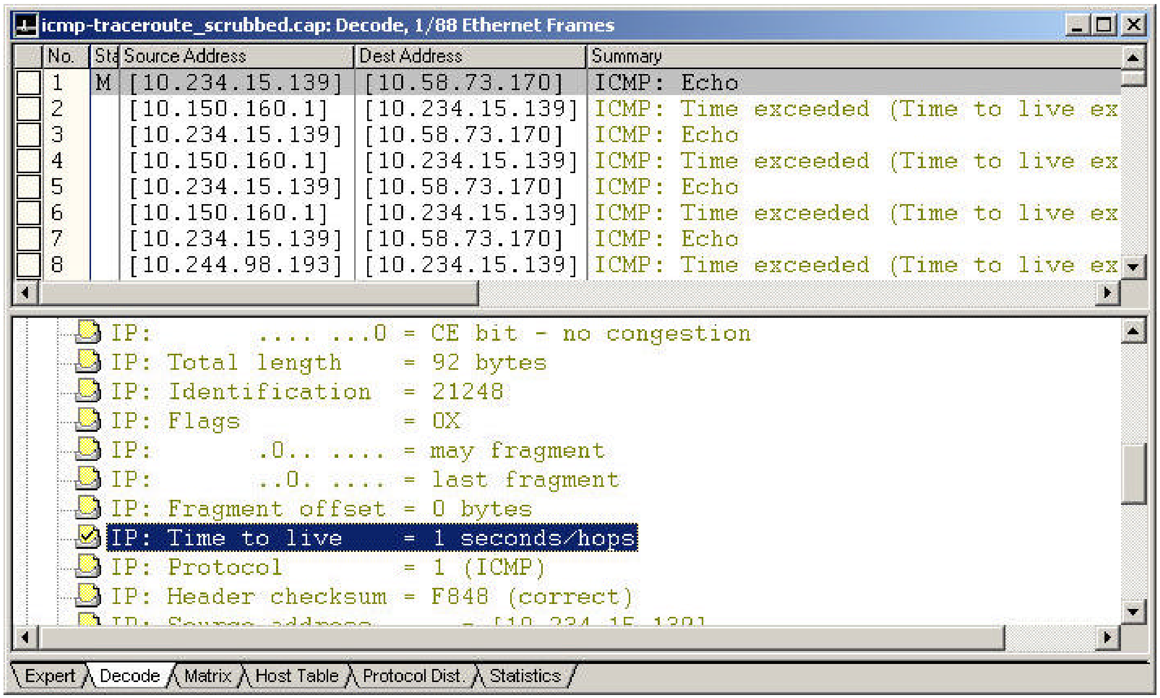
Laura Chappell, Sr. Protocol Analyst
Protocol Analysis Institute
www.packet-level.com; www.podbooks.com

===

Traceroute is used to trace back along a path to a target in order to learn the routers that lie between the originator and target. Traceroute is often used to identify a sluggish link when troubleshooting a slow communication.

When most people run the traceroute utility from the command prompt (`tracert`), they expect their system to send a series of Internet Control Message Protocol (ICMP) packets crossing the wire – the ICMP Echo Request, ICMP Time Exceeded and ICMP Echo Reply packets). This is the default functionality of traceroute.

Figure 1 shows a portion of an ICMP based Traceroute operation performed using the MS Windows traceroute utility.



No.	Sta	Source Address	Dest Address	Summary
1	M	[10.234.15.139]	[10.58.73.170]	ICMP: Echo
2		[10.150.160.1]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex
3		[10.234.15.139]	[10.58.73.170]	ICMP: Echo
4		[10.150.160.1]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex
5		[10.234.15.139]	[10.58.73.170]	ICMP: Echo
6		[10.150.160.1]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex
7		[10.234.15.139]	[10.58.73.170]	ICMP: Echo
8		[10.244.98.193]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex

```
IP:        .... ...0 = CE bit - no congestion
IP: Total length     = 92 bytes
IP: Identification   = 21248
IP: Flags            = 0X
IP:        .0.. .... = may fragment
IP:        ..0. .... = last fragment
IP: Fragment offset  = 0 bytes
IP: Time to live     = 1 seconds/hops
IP: Protocol         = 1 (ICMP)
IP: Header checksum  = F848 (correct)
IP: Source address   = [10.234.15.139]
```

Figure 1: The standard traceroute utility generates numerous ICMP Echo Request and ICMP Time Exceeded messages.

Following an ICMP Traceroute Communication

Figure 1 shows a client (10.234.15.139) sending a standard ICMP Echo Request packet across the wire with an IP Time to Live (TTL) field value of 1 (second/hop) to the target (10.58.73.170). When this packet is received by the first router along the path it must be dropped because it has no remaining lifetime – the receiving router (10.150.160.1) cannot decrement the TTL field to zero and forward it on.

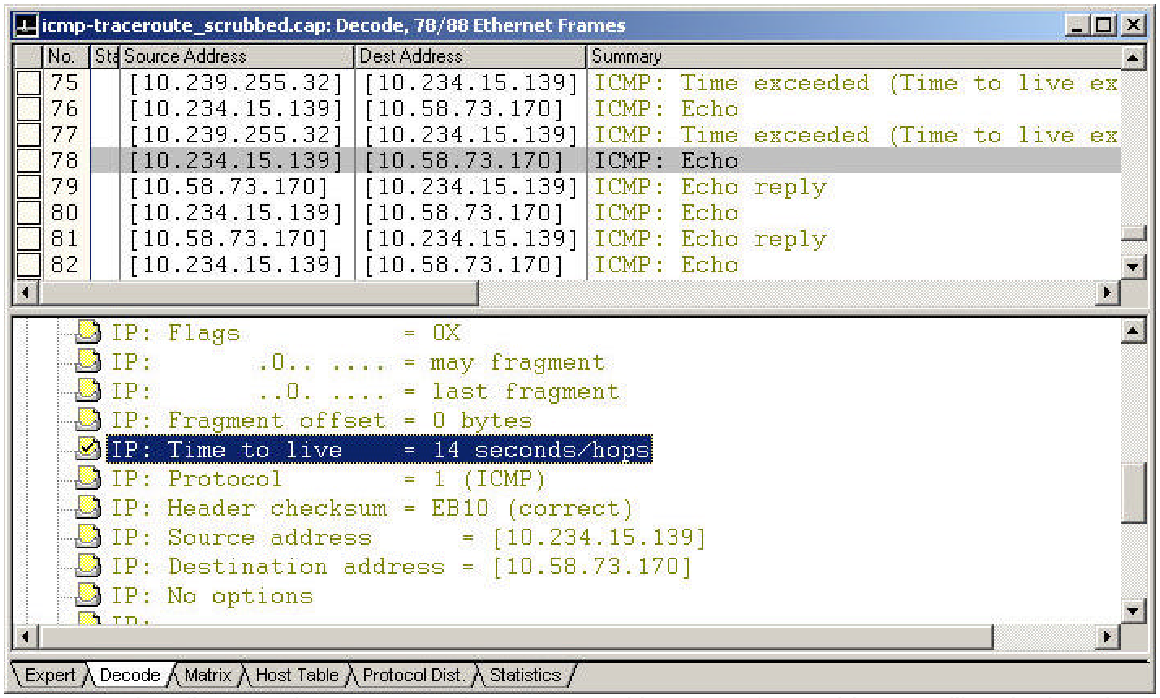
The router 10.150.160.1 sends an ICMP Time Exceeded/Time to Live Expired in Transit packet back to the client at 10.234.15.139. When the client receives this Time Exceeded/Time to Live Expired packet, the client examines the source IP address in the to note the first router along the path.

When you examine the trace file **ICMP-traceroute** available online at www.packet-level.com > Library > Traces (available in .cap/.dmp/.pkt formats), you'll notice that the traceroute utility used in the development of this article sends each TTL set three times out to the wire. This is common practice.

Once the first router has been discovered, it's time to learn the next router along the path. The client increments the TTL value and sends another set of ICMP Echo Request packets with a TTL of two (seconds/hops). These packets can be forwarded by the first router along the path (which decrements the TTL value to one) to the next router along the path.

The second router along the path receives the ICMP packets which now have a TTL of one. That router (10.244.98.193) respond with an ICMP Time Exceeded/Time to Live Expired in Transit message back to the source. The Client learns the second router along the path.

Standard traceroute utilities continue to increment the TTL field until they receive an ICMP Echo reply from the target. In Figure 2 we can see the completion of a traceroute operation using ICMP. In Packet 79 we see a response directly from our desired target, 10.58.73.170. Using ICMP's echo request process we have identified routers along a path and we have determined there is an active system at the target IP address.



icmp-traceroute_scrubbed.cap: Decode, 78/88 Ethernet Frames

	No.	Sta	Source Address	Dest Address	Summary
□	75		[10.239.255.32]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex
□	76		[10.234.15.139]	[10.58.73.170]	ICMP: Echo
□	77		[10.239.255.32]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex
□	78		[10.234.15.139]	[10.58.73.170]	ICMP: Echo
□	79		[10.58.73.170]	[10.234.15.139]	ICMP: Echo reply
□	80		[10.234.15.139]	[10.58.73.170]	ICMP: Echo
□	81		[10.58.73.170]	[10.234.15.139]	ICMP: Echo reply
□	82		[10.234.15.139]	[10.58.73.170]	ICMP: Echo

```
IP: Flags            = 0X
IP:         .0.. .... = may fragment
IP:         ..0. .... = last fragment
IP: Fragment offset  = 0 bytes
IP: Time to live     = 14 seconds/hops
IP: Protocol         = 1 (ICMP)
IP: Header checksum  = EB10 (correct)
IP: Source address       = [10.234.15.139]
IP: Destination address  = [10.58.73.170]
IP: No options
IP:
```

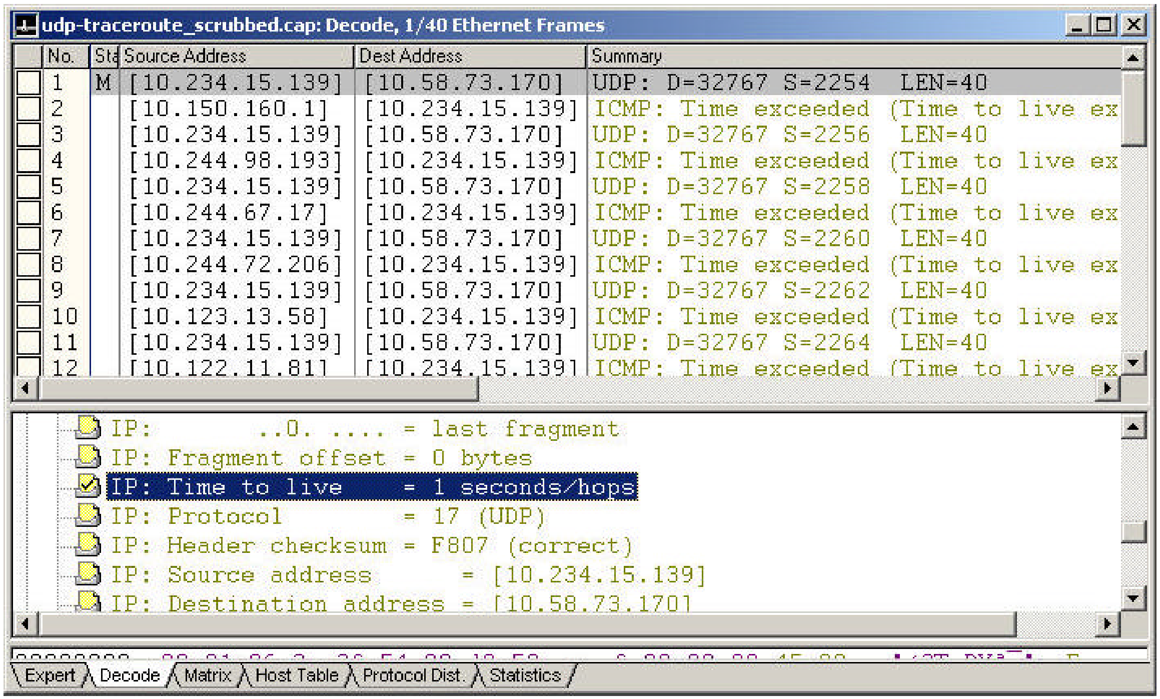
Figure 2: The client sends an ICMP Echo Request with a TTL of 14 – this one will make it to the target and trigger an ICMP Echo Reply packet.

This form of traceroute may work most of the time, but what if a network administrator has turned off the ICMP Echo Reply function on the target device or they filter out all incoming ICMP Echo Requests? In that case, they'll need to find an alternate way to discover the path. This is where UDP traceroute comes in.

Following a UDP Traceroute Communication

UDP traceroute is similar to ICMP traceroute in the fact that it plays with the TTL field in the IP header. In a UDP traceroute, the client transmits a simple UDP packet to an invalid destination port value.

Figure 4 shows a partial decode of this UDP packet. In the first packet, the client (10.234.15.139) sets the TTL field to one (second/hop) and the UDP destination port number is 32767 (D=32767 in the upper window). This is an unregistered UDP port number; we don't actually expect the destination to accept communications on that port number. In fact, we hope the destination will not accept this communication and send back an ICMP Destination Unreachable/Port Unreachable packet.



udp-traceroute_scrubbed.cap: Decode, 1/40 Ethernet Frames

```
No. Sta Source Address     Dest Address       Summary
1   M   [10.234.15.139]    [10.58.73.170]     UDP: D=32767 S=2254  LEN=40
2       [10.150.160.1]     [10.234.15.139]    ICMP: Time exceeded (Time to live ex
3       [10.234.15.139]    [10.58.73.170]     UDP: D=32767 S=2256  LEN=40
4       [10.244.98.193]    [10.234.15.139]    ICMP: Time exceeded (Time to live ex
5       [10.234.15.139]    [10.58.73.170]     UDP: D=32767 S=2258  LEN=40
6       [10.244.67.17]     [10.234.15.139]    ICMP: Time exceeded (Time to live ex
7       [10.234.15.139]    [10.58.73.170]     UDP: D=32767 S=2260  LEN=40
8       [10.244.72.206]    [10.234.15.139]    ICMP: Time exceeded (Time to live ex
9       [10.234.15.139]    [10.58.73.170]     UDP: D=32767 S=2262  LEN=40
10      [10.123.13.58]     [10.234.15.139]    ICMP: Time exceeded (Time to live ex
11      [10.234.15.139]    [10.58.73.170]     UDP: D=32767 S=2264  LEN=40
12      [10.122.11.81]     [10.234.15.139]    ICMP: Time exceeded (Time to live ex
```

```
IP:        ..0. .... = last fragment
IP: Fragment offset = 0 bytes
IP: Time to live     = 1 seconds/hops
IP: Protocol         = 17 (UDP)
IP: Header checksum = F807 (correct)
IP: Source address      = [10.234.15.139]
IP: Destination address = [10.58.73.170]
```

Figure 3: The UDP packet contains a TTL of 1 and an unregistered destination port number in hopes of triggering an ICMP Destination Unreachable/Port Unreachable packet from the target.

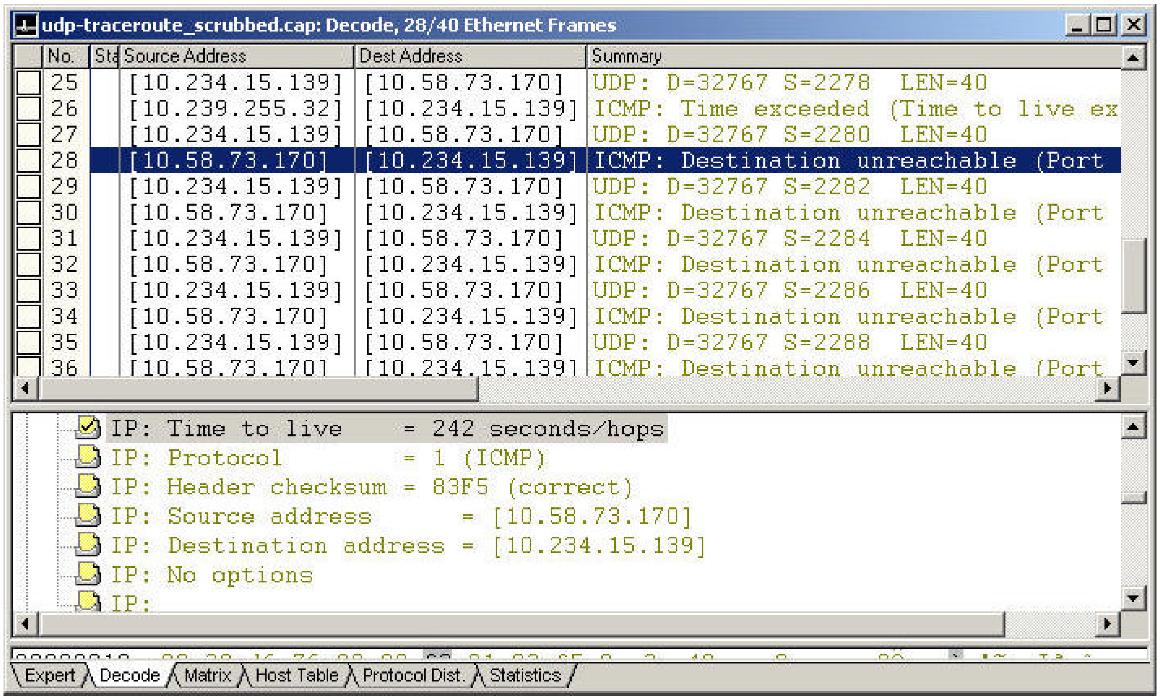
The first router along the path (10.150.160.1) responds with an ICMP Time Exceeded/Time to Live Expiring in Transit packet.

Next, 10.234.15.139 sends another UDP packet to the same invalid port number (packet 3). This time the second router along the path (10.244.98.193) responds with an ICMP Time Exceeded/Time to Live Exceeded in Transit message (packet 4).

***Note**: Unlike the ICMP traceroute utility which sent sets of three TTL packets in a set, the UDP traceroute utility shown in Figure 3 and 4 only sends one TTL set before incrementing to the next value.*

This process continues until the TTL value is high enough to reach the target. At that point the target responds with an ICMP Destination Unreachable/Port Unreachable message. This is the target’s way of indicating that it does not support any service at port number 32767.

In Figure 4 the target has finally been reached. This UDP traceroute utility repeats the final UDP traceroute query multiple times to the target.



No.	Sta	Source Address	Dest Address	Summary
25		[10.234.15.139]	[10.58.73.170]	UDP: D=32767 S=2278 LEN=40
26		[10.239.255.32]	[10.234.15.139]	ICMP: Time exceeded (Time to live ex
27		[10.234.15.139]	[10.58.73.170]	UDP: D=32767 S=2280 LEN=40
28		[10.58.73.170]	[10.234.15.139]	ICMP: Destination unreachable (Port
29		[10.234.15.139]	[10.58.73.170]	UDP: D=32767 S=2282 LEN=40
30		[10.58.73.170]	[10.234.15.139]	ICMP: Destination unreachable (Port
31		[10.234.15.139]	[10.58.73.170]	UDP: D=32767 S=2284 LEN=40
32		[10.58.73.170]	[10.234.15.139]	ICMP: Destination unreachable (Port
33		[10.234.15.139]	[10.58.73.170]	UDP: D=32767 S=2286 LEN=40
34		[10.58.73.170]	[10.234.15.139]	ICMP: Destination unreachable (Port
35		[10.234.15.139]	[10.58.73.170]	UDP: D=32767 S=2288 LEN=40
36		[10.58.73.170]	[10.234.15.139]	ICMP: Destination unreachable (Port

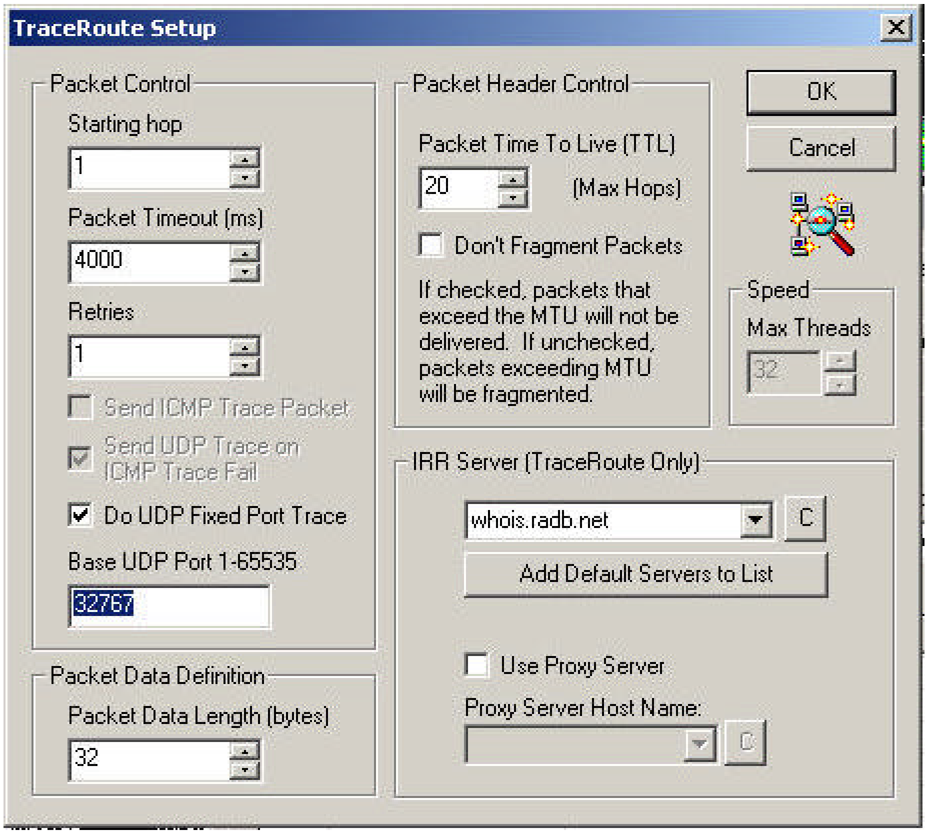
```
IP: Time to live     = 242 seconds/hops
IP: Protocol         = 1 (ICMP)
IP: Header checksum = 83F5 (correct)
IP: Source address       = [10.58.73.170]
IP: Destination address = [10.234.15.139]
IP: No options
IP:
```

Figure 4: The target responds with an ICMP Destination Unreachable/Port Unreachable.

An unusually high number of ICMP packets on your network should alert you to something unusual happening on the wire.

The UDP traceroute utility used in the creation of this article is part of NetScanTools Pro (an excellent product available online at www.netscantools.com) . Figure 5 displays the setup screen for the traceroute utility in NetScanTools Pro. As you can see, the base UDP port number was defined as 32767 and there are three options for performing traceroute:

- Send ICMP trace packet (ICMP traceroute only)
- Send UDP trace on ICMP trace fail (ICMP, then UDP traceroute)
- Do UDP fixed port trace (UDP traceroute only)



Packet Control

Starting hop: 1

Packet Timeout (ms): 4000

Retries: 1

□ Send ICMP Trace Packet

☑ Send UDP Trace on ICMP Trace Fail

☑ Do UDP Fixed Port Trace

Base UDP Port 1-65535: 32767

Packet Data Definition

Packet Data Length (bytes): 32

Packet Header Control

Packet Time To Live (TTL): 20 (Max Hops)

□ Don't Fragment Packets

If checked, packets that exceed the MTU will not be delivered. If unchecked, packets exceeding MTU will be fragmented.

Speed

Max Threads: 32

IRR Server (TraceRoute Only): whois.radb.net

□ Use Proxy Server

Proxy Server Host Name:

Figure 5: NetScanTools Pro includes a traceroute utility that can perform ICMP traceroutes or UDP traceroutes.

Keep in mind that if the target blocks outbound ICMP Destination Unreachable packets, then even UDP traceroute won't work.

Although ICMP traceroute is the 'king' of traceroute functions, UDP traceroute offers a simple alternate method to discover the path to a target.

Trace files at www.packet-level.com > Library > Trace Files:

- ICMP-traceroute (.cap/.dmp/.pkt)
- UDP-traceroute (.cap/.dmp/.pkt)