Traceroute Torn Apart By Ankit Fadia ankit@bol.net.in

The Traceroute program is a very useful debugging tool, which can be used to find a number of useful things about the host and client, the routers, and systems that data passes through on its way from the source to the destination and a whole of lot related information.

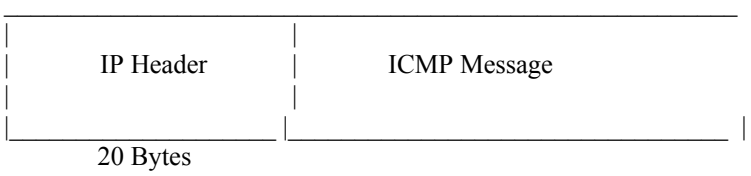
However, Traceroute is most commonly used to find out the path taken by an IP datagram from the source to the destination. For Example, one could use Traceroute to find out the IP Addresses or hostnames of all the systems that data from your system has to pass through to reach your favorite site.

But, there is one thing that one has to keep in mind, while implementing Traceroute. You see, there are simply no guarantees that two consecutive IP datagrams from the same source to the same destination have to take the exact same route. However, again, most of the time they do take the same route.

Traceroute is a part of protocol called the Internet Control Message Protocol or the ICMP, which is popularly used for debugging purposes or Network Diagnosis.

The ICMP protocol communicates error messages and other malfunctions or problems that might have occurred while the data transactions between two systems were taking place. So, it can also be called the 'Network Problem Diagnosis' protocol. It is basically the protocol, which reports any error that might have occurred while the data transfer was still taking place.

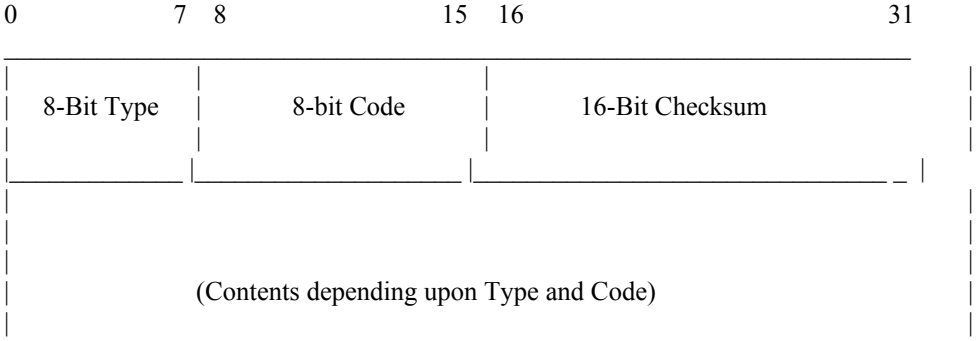
All ICMP messages are transmitted as IP datagrams. A Typical ICMP message encapsulated within an IP datagram would be as follows:



The first 4 bytes have the same format and specification for all the messages, however, the remaining part of the datagram differs from message to message depending upon the kind or message i.e. the type of error message or the type of message carried by the datagram.

Although we will not be discussing the entire ICMP protocol in this manual, let me just give a quick description of the various parts of an ICMP message. For a more detailed version of the description refer to RFC 792.

The format of a typical ICMP message is as follows:



The ‘type’ field can have any of the 15 different values, which determine or represent a particular ICMP error message. For Example, a value of 3 in the type field specifies the ‘Destination unreachable’ error message. Like this there are 15 different values, which the ‘type’ field can have, with each value representing a particular specific error type.

Now, an error message like the above was a very general or rather a very broad error message. It does not tell you the exact cause of the error or where exactly or what exactly caused the error to occur. In order to ensure that the user or application running on either the client side or the server side, ICMP has the provision whereby, each main error specified by the ‘type’ value has a number of sub errors, which give a more specific reason or cause of the error.

Such sub-errors, which are more specific and more helpful, are specified in ICMP in the ‘code’ field. For Example, a ‘Type’ Value of 3 and a ‘code’ value of 0 specifies that the error caused is: ‘Network Unreachable’. While a ‘Type’ Value of 3 and a ‘code’ value of 1 specifies that the error caused is: ‘Host Unreachable’. In order to get a detailed description of ICMP codes and types and its working, wait for my manual on ICMP.

Now, that we know a bit about ICMP let us come back to the actual subject of this manual: Traceroute.

Traceroute uses the ICMP protocol and the TTL or Time to Live field (which is a part of the IP protocol). If you remember the diagram of an ICMP message given earlier in this manual, you would know that the first 20 bytes consisted of the IP Header. It is in this header that you will find a number of fields, which actually are a part of the Internet Protocol (IP). Now, the TTL field is a part of the IP protocol and is stored here.

So what is the TTL field for? The Time to Live field or TTL is an 8-bit field that sets the upper limit (maximum limit) of the number of routers through which a datagram can pass. It basically determines how long a datagram will be alive. It contains the value after which the datagram is discarded. It is initiated and set at the sender’s end to a predefined value (normally 32 or 64) and this value is then decremented by one by every router that handles it. When this value is finally decremented to 0, then the datagram is thrown away and ICMP is called upon to report the error to the sender. This prevents datagrams from infinitely looping through routers.

TTL basically would be something like determining the age of the datagram. As if saying: “Ok, Mr. Datagram, you can pass through only x number of routers, after that you will have to leave the wired world.” The following section will make it absolutely clear as to what exactly TTL is.

****
HACKING TRUTHS NOTE: The recommended initial value of TTL is currently 64. (According to RFC of Assigned Numbers.) However, older systems use 15 or 32 as the initial values.
****

Each router that handles an IP datagram is required to decrement the datagram’s TTL value by one or by the number of seconds it holds it. As almost all routers do not hold a datagram longer that 1 second, the TTL value has essentially become a hop counter, giving the number of routers through which the datagram has passed.

This TTL feature was essential to prevent datagrams from ending up in infinite loops, which may occur due to unfinished transactions or when the client or server shuts disconnects from the network without closing the open connections.

Now, when a router receives a datagram whose TTL field value is set to either 0 or 1, then it will not forward the datagram. Instead the router discards all datagrams received with TTL set to 0. Once the datagram has been discarded, the router sends an ICMP message saying: 'Time Exceeded.' (Back to the sender.)

This ICMP message that the sender has received from the router, is actually an IP datagram whose source address is same as that of the router's. So, in effect the sender to get the IP Address of the Router can use this IP datagram.

Now, what Traceroute does is as follows:

Traceroute sends an IP Datagram with a TTL value of 1 to the destination system. Now, the first router to handle the datagram decrements the TTL value by 1 and then discards the datagram. After that, it sends an ICMP error message with its own IP Address as the source address of the message to your system. This way your system can record the IP of the first router on the path to the destination system. Then Traceroute sends a new datagram, this time with a TTL value of 2 to the destination host. In this case, the second router on the path decrements the TTL value to 0 and discards it and its IP address is recorded. This process continues until the datagram reaches the destination system.

One thing to note here is that the IP datagram received by the destination system will have a TTL value of 1 (without making the decrement). Even then it will not discard it, as the destination has been reached. So this means that the destination system does not discard the IP datagram or in other words does not create any Time Exceeded ICMP error message. So, this in turn should mean that your system should have no method of deducing whether the destination system has been reached or not. Right? Well yes. Just to solve this problem, traceroute uses another mechanism.

Traceroute sends UDP datagrams to the destination system at extremely high UDP port numbers. It chooses high ports such that no application is likely to be running at that port. This value is normally higher than 30000. These UDP datagrams received at a high UDP port, causes the UDP module running on the destination system to generate an ICMP error message of 'Port Unreachable'. All, traceroute now has to do is to differentiate between the ICMP error messages of 'Time Exceeded' and 'Port Unreachable' in order to deduce when the destination system has been reached.

****
HACKING TRUTH: An ICMP error Message of the 'Time Exceeded' error has a 'type' value of 11 and a 'code' value of 0. While an ICMP message of the ' Port Unreachable' error has a 'type' value of 3 and a 'code' value of 3.
****
So working of traceroute can be summarized as follows: Send UDP datagrams to the destination host starting with a TTL value of 1 and increasing the TTL by 1 to locate each router in the path. Each router in the path returns an ICMP time exceeded. The destination system instead returns a port unreachable. This is used to differentiate when the destination system has been reached.

Let us take an example of Traceroute to see how it works and what all it returns:

```
host2 # traceroute xyz.com
traceroute to xyz.com  (202.xx.12.34), 30 hops max, 40 byte packets
   1      isp.net    (202.xy.34.12)    20ms     10ms     10ms
   2      xyz.com (202.xx.12.34)    130ms    130ms    130ms
```

The first unnumbered line above gives the hostname and IP address of the destination system and tells the user that the TTL value will not increase beyond 30. It also indicates that the datagram size will be 40 bytes, which allows for the 20-byte IP Header, the 8-byte UDP header, and 12 bytes of user data.

The next two lines start with the TTL value, followed by the router name (with its IP Address), which returned the Time Exceeded on that instance. For each value of TTL, three different datagrams are sent.

And for each returned ICMP message the round-trip time is calculated and printed. If no response is received within 5 seconds for any of the three datagrams, an asterisk is printed instead and the next datagram is sent. In the above case, the first three datagrams had a TTL value of 1 and were returned by the host is.net whose IP is 202.xy.34.12 in 20, 10 and 10 ms. The next three datagrams had a TTL value of 2 and were returned by host xyz.com whose IP is 202.xx.12.34 (this is also the destination system) with a gap of 130ms of each other. This second host also sent a Port Unreachable ICMP error, thus traceroute knew that the destination host has been reached.

Implementing Traceroute Functionality

CtraceRoute is a freeware MFC class to implement trace route functionality. You can download the header files, the Documentation and help files from here: Click Here To Download

That is all for know, hope you liked this manual. Till the next update, this is Ankit Fadia saying goodbye!!!

_____

Have you sent me an email, which I haven't replied yet? Well, this kindly read following:

I apologize for not being able to get back to you. But, I assure you I will reply to you as soon as possible, please bear with me. In order to release the congestion of emails, you can now contact me via Instant Messaging Software. I use MSN Messenger and the email address which I use is: ankit_fadia@hotmail.com (Do Not send mail to this address. I do not check this account.) . Simply download MSN Messenger or a Multi Messenger Platforms software and search for the above email address and add it to your contact list. Then the next time I am online you will be informed and you can post your question ot me. However, that doesn't mean I will not be answering my emails. I try and answer all my emails except questions like How to Hack Hotmail etc. However, most of the times my replies come real slow. Sorry. ☺

_____

Ankit Fadia
ankit@bol.net.in
http://hackingtruths.box.sk

To receive tutorials written by Ankit Fadia on Everything you ever dreamt of in your Inbox, join his mailing list by sending a blank email to: programmingforhackers-subscribe@egroups.com