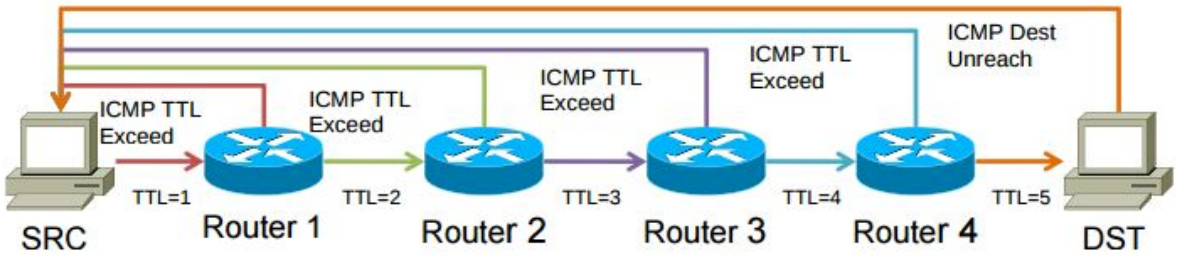
Assignment-5

Implementing Traceroute using raw sockets

Introduction:

Traceroute is a utility that records the route (the specific gateway computers at each hop) through the Internet between your computer and a specified destination computer. It also calculates and displays the amount of time each hop took. Traceroute is a handy tool both for understanding where problems are in the Internet network and for getting a detailed sense of the Internet itself.



Traceroute Algorithm:

1. Traceroute launches a probe packet towards the final destination, with an initial TTL value of 1.
2. Each router that handles the packet along the way decrements the TTL by 1, until the TTL reaches 0.
3. When the TTL value reaches 0, the router which discarded the packet sends an ICMP TTL EXCEED message back to the original sender.
4. The Traceroute utility receives this ICMP TTL EXCEED packet, and uses the time difference between the original probe packet and the returned ICMP packet to calculate the round-trip latency for this router “hop”.
5. Repeat this process again from step 1, with a new initial TTL value of N+1
6. Eventually, the final destination receives the Traceroute probe packet, and sends back a reply packet other than an ICMP TTL EXCEED. The Traceroute utility uses this to know that the Traceroute is now complete, and ends the process.

Sample:

> traceroute www.google.com

traceroute to www.google.com (216.58.196.68), 30 hops max, 60 byte packets
1 10.228.193.25 (10.228.193.25) 93.667 ms 100.668 ms 100.747 ms
2 10.228.213.18 (10.228.213.18) 143.902 ms 143.902 ms 143.972 ms
3 116.202.227.17 (116.202.227.17) 100.831 ms 100.821 ms 100.933 ms
4 10.228.5.152 (10.228.5.152) 190.506 ms 190.570 ms 190.689 ms
5 72.14.205.145 (72.14.205.145) 162.073 ms 162.139 ms 162.783 ms
6 209.85.242.219 (209.85.242.219) 162.323 ms 72.14.235.69 (72.14.235.69) 161.982 ms
209.85.242.219 (209.85.242.219) 149.637 ms
7 216.239.48.215 (216.239.48.215) 186.733 ms 186.850 ms 72.14.238.178
(72.14.238.178) 194.569 ms
8 209.85.242.233 (209.85.242.233) 202.620 ms 212.680 ms 212.750 ms
9 216.239.41.49 (216.239.41.49) 219.254 ms 226.155 ms 226.163 ms
10 kul01s09-in-f4.1e100.net (216.58.196.68) 226.316 ms 232.653 ms 232.730 ms

The first line of the output describes what the command is doing. It lists the destination system (www.google.com), destination IP address (216.58.196.68), and the maximum number of hops that will be used in the traceroute (30).
The remainder of the output shows information on each hop, which is typically a router, in the path between the sender and the final destination. Each line has format,

hop_number host_name (IP_address) packet_round_trip_times

Here is what each field means:

- **hop_number**: A sequential count of the number of degrees of separation the host is from your computer. Traffic from hosts with higher numbers have to go through more computers to get routed.
- **host_name**: This field contains the result of a reverse DNS lookup on the host's IP address, if available. If no information is returned from the reverse DNS query, the IP address itself is given.
- **IP_address**: This field contains the IP address for this network hop.
- **packet_round_trip_times**: The remainder of the line gives the round-trip times for a packet to the host and back again. By default, three packets are sent to each host and each attempt is appended to the end of the line.

Objective:

Raw sockets were introduced in Assignment 3, utilize this knowledge to implement ***traceroute*** which gives the similar result as above.

NOTES:

Setting the TTL Value

1. Need to control the IP TTL value
2. Raw socket with ICMP does not let us write IP header values
3. Use setsockopt() to set TTL value

```
setsockopt(raw, IPPROTO_IP, IP_TTL, (char *) &ttl, sizeof(ttl))
        Or
int on = 1; setsockopt(raw, IPPROTO_IP, IP_HDRINCL, &on, sizeof(on))
```