

COL 672 Assignment-1

Kashish jain-2021JCS2240

February 4,2021

1.Networking Tools:

1.1 ipconfig: This command gives TCP/IP configuration details of the machine.

1.2 nslookup: This command is used to ask the IP address of some domain from DNS server.

1.3 ping: This command is used to check whether a given domain is reachable or not.

1.4 tracert: This command is used to trace the route of packet to reach the destination server.

A.

Ipconfig: is a fast way to determine computer's IP address and other information, such as address of its default gateway, subnet mask.

- I. The IP address of my system when connected to Touchnet internet connection: 192.168.0.101
- II. The IP address of my system when connected to Reliance jio fiber: 192.168.29.202
- III. Because of the change in Network Infrastructure, IP addresses changes, with the change in ISP.

B.

Nslookup: nslookup stands for: "name server lookup".

It is a network administration command line tool used for querying the domain name system to obtain domain name or IP address mapping or other DNS records.

- I. I have used nslookup command to query internet domain servers.

- The IP address of Google.com, with default DNS server(using reliance jio fiber then using Touchnet connection):

```
> www.google.com
Server:  reliance.reliance
Address:  192.168.29.1

Non-authoritative answer:
Name:      www.google.com
Addresses:  2404:6800:4009:804::2004
            216.58.203.4

> exit

C:\Users\PRANAV>nslookup
Default Server:  dns.google
Address:  8.8.8.8

> www.google.com
Server:  dns.google
Address:  8.8.8.8

Non-authoritative answer:
Name:      www.google.com
Addresses:  2404:6800:4002:82f::2004
            216.58.200.196
```

- The IP address of Google.com, with open DNS servers: ns1.google.com,1.1.1.1(cloudfare):

```
C:\Users\PRANAV>nslookup www.google.com ns1.google.com
Server:  ns1.google.com
Address:  216.239.32.10

Name:      www.google.com
Addresses:  2404:6800:4002:825::2004
            142.250.194.228

C:\Users\PRANAV>nslookup www.google.com 1.1.1.1
Server:  one.one.one.one
Address:  1.1.1.1

Non-authoritative answer:
Name:      www.google.com
Addresses:  2404:6800:4002:810::2004
            216.58.196.100
```

- The IP addresses of facebook.com with default DNS server (using Touchnet connection, then reliance jio fiber):

```
C:\Users\PRANAV>nslookup
Default Server:  dns.google
Address:  8.8.8.8

> www.facebook.com
Server:  dns.google
Address:  8.8.8.8

Non-authoritative answer:
Name:     star-mini.c10r.facebook.com
Addresses:  2a03:2880:f144:82:face:b00c:0:25de
            157.240.198.35
Aliases:  www.facebook.com
```

```
C:\Users\PRANAV>nslookup
Default Server:  reliance.reliance
Address:  192.168.29.1

> www.facebook.com
Server:  reliance.reliance
Address:  192.168.29.1

Non-authoritative answer:
Name:     star-mini.c10r.facebook.com
Addresses:  2a03:2880:f16e:81:face:b00c:0:25de
            157.240.242.35
Aliases:  www.facebook.com
```

- The IP addresses of facebook.com along with it's canonical name:

```
> set q=ns
> www.facebook.com
Server:  reliance.reliance
Address:  192.168.29.1

Non-authoritative answer:
www.facebook.com      canonical name = star-mini.c10r.facebook.com
star-mini.c10r.facebook.com
      primary name server = a.ns.c10r.facebook.com
      responsible mail addr = dns.facebook.com
      serial = 561956141
      refresh = 300 (5 mins)
      retry = 600 (10 mins)
      expire = 600 (10 mins)
      default TTL = 300 (5 mins)
```

- The IP addresses of facebook.com with open servers:

```

C:\Users\PRANAV>nslookup www.facebook.com ns1.google.com
Server: ns1.google.com
Address: 216.239.32.10

*** ns1.google.com can't find www.facebook.com: Query refused

C:\Users\PRANAV>nslookup www.facebook.com 1.1.1.1
Server: one.one.one.one
Address: 1.1.1.1

Non-authoritative answer:
Name: star-mini.c10r.facebook.com
Addresses: 2a03:2880:f12f:83:face:b00c:0:25de
157.240.16.35
Aliases: www.facebook.com

C:\Users\PRANAV>nslookup www.facebook.com 8.8.8.8
Server: dns.google
Address: 8.8.8.8

Non-authoritative answer:
Name: star-mini.c10r.facebook.com
Addresses: 2a03:2880:f144:82:face:b00c:0:25de
157.240.198.35
Aliases: www.facebook.com

C:\Users\PRANAV>nslookup www.facebook.com d.ns.facebook.com
Server: d.ns.facebook.com
Address: 185.89.219.12

Name: www.facebook.com

```

Domain\DNS	1.1.1.1(Cloud fare)	8.8.8.8(googl e public DNS)	ns1.google.c om	d.ns.facebook. com
www.google.co m	216.58.196.1 00	142.250.194. 196	142.250.193.3 6	refused
www.facebook. com	157.240.16.35	157.240.198.35	refused	157.240.16.20

- If we mention the exact DNS server for the domain, we will not get the non-authoritative label.
- Non authoritative label indicates, that the result of DNS query is coming indirectly from authoritative DNS servers.
- On changing the DNS servers, IP addresses are not changing much, because these domains have multiple host servers which have similar but different addresses.

C.

Ping (packet internet groper): It is used to check the network connectivity between host and server. By using ICMP echo and reply messages it gives information.

Different options available:

1. Ping -t: used to ping a particular target continuously.
2. Ping -a: used to resolve the IP address of hostname.
3. Ping -n: used to control the number of echo request to send.
4. Ping -l: used to set the size of buffer.
5. Ping -i: used to set the TTL value.

I used ping command for three different hosts using reliance jio fiber:

HOST	Min TTL to reach destination	Maximum packet size(bytes)
www.iitd.ac.in	21	35512
www.google.com	14	1472
www.facebook.com	14	1472

1. www.iitd.ac.in:

- The maximum size of ping packets that I could send is 35512 bytes. On adding 8 bytes of ICMP header, it becomes 35520 bytes.

- The minimum TTL which succeeded is:21. This shows that packet made 21 hops to reach iitd server from my host.

```
C:\Users\PRANAV>ping -i 21 www.iitd.ac.in

Pinging www.iitd.ac.in [103.27.9.24] with 32 bytes of data:
Reply from 103.27.9.24: bytes=32 time=8ms TTL=47
Reply from 103.27.9.24: bytes=32 time=8ms TTL=47
Reply from 103.27.9.24: bytes=32 time=9ms TTL=47
Reply from 103.27.9.24: bytes=32 time=9ms TTL=47

Ping statistics for 103.27.9.24:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 9ms, Average = 8ms

C:\Users\PRANAV>ping -i 20 www.iitd.ac.in

Pinging www.iitd.ac.in [103.27.9.24] with 32 bytes of data:
Reply from 103.27.9.24: TTL expired in transit.
Reply from 103.27.9.24: TTL expired in transit.
Reply from 103.27.9.24: TTL expired in transit.
Reply from 103.27.9.24: TTL expired in transit.

Ping statistics for 103.27.9.24:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Request timed out.

Ping statistics for 103.27.9.24:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
    Request timed out.

C:\Users\PRANAV>ping -l 35512 www.iitd.ac.in

Pinging www.iitd.ac.in [103.27.9.24] with 35512 bytes of data:
Reply from 103.27.9.24: bytes=35512 time=60ms TTL=46
Reply from 103.27.9.24: bytes=35512 time=58ms TTL=46
Reply from 103.27.9.24: bytes=35512 time=50ms TTL=46
Reply from 103.27.9.24: bytes=35512 time=51ms TTL=46

Ping statistics for 103.27.9.24:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 50ms, Maximum = 60ms, Average = 56ms

C:\Users\PRANAV>ping -l 35513 www.iitd.ac.in

Pinging www.iitd.ac.in [103.27.9.24] with 35513 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 103.27.9.24:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
    Request timed out.

C:\Users\PRANAV>
```

2.www.google.com:

- The maximum packet size that I could send is 1472 bytes. On adding 8 bytes of ICMP header, it becomes 1480 bytes.
- The minimum TTL which succeeded is 14.

```

C:\Users\PRANAV>ping -i 13 www.google.com

Pinging www.google.com [172.217.167.196] with 32 bytes of data:
Reply from 209.85.252.71: TTL expired in transit.
Reply from 209.85.252.71: TTL expired in transit.
Reply from 209.85.252.71: TTL expired in transit.
Reply from 209.85.252.71: TTL expired in transit.

Ping statistics for 172.217.167.196:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Users\PRANAV>ping -i 14 www.google.com

Pinging www.google.com [172.217.167.196] with 32 bytes of data:
Reply from 172.217.167.196: bytes=32 time=9ms TTL=112
Reply from 172.217.167.196: bytes=32 time=8ms TTL=112
Reply from 172.217.167.196: bytes=32 time=8ms TTL=112
Reply from 172.217.167.196: bytes=32 time=9ms TTL=112

Ping statistics for 172.217.167.196:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 8ms, Maximum = 9ms, Average = 8ms

```

```

C:\Users\PRANAV>ping -l 1472 www.google.com

Pinging www.google.com [216.58.203.4] with 1472 bytes of data:
Reply from 216.58.203.4: bytes=68 (sent 1472) time=34ms TTL=111
Reply from 216.58.203.4: bytes=68 (sent 1472) time=42ms TTL=111
Reply from 216.58.203.4: bytes=68 (sent 1472) time=34ms TTL=111
Reply from 216.58.203.4: bytes=68 (sent 1472) time=34ms TTL=111

Ping statistics for 216.58.203.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 34ms, Maximum = 42ms, Average = 36ms

C:\Users\PRANAV>ping -l 1473 www.google.com

Pinging www.google.com [216.58.203.4] with 1473 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 216.58.203.4:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

```

3.www.facebook.com:

- The maximum packet size that I could send is: 1472 bytes. On adding 8 bytes of ICMP header, it becomes 1480 bytes.
- The minimum TTL which succeeded is 14.

```
C:\Users\PRANAV>ping -l 1472 www.facebook.com

Pinging star-mini.c10r.facebook.com [31.13.79.35] with 1472 bytes of data:
Reply from 31.13.79.35: bytes=1472 time=35ms TTL=50
Reply from 31.13.79.35: bytes=1472 time=36ms TTL=50
Reply from 31.13.79.35: bytes=1472 time=40ms TTL=50
Reply from 31.13.79.35: bytes=1472 time=35ms TTL=50

Ping statistics for 31.13.79.35:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 35ms, Maximum = 40ms, Average = 36ms
```

```
C:\Users\PRANAV>ping -l 1473 www.facebook.com

Pinging star-mini.c10r.facebook.com [31.13.79.35] with 1473 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 31.13.79.35:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

```
C:\Users\PRANAV>ping -i 13 www.facebook.com

Pinging star-mini.c10r.facebook.com [157.240.242.35] with 32 bytes of data:
Reply from 157.240.39.1: TTL expired in transit.
Reply from 157.240.39.1: TTL expired in transit.
Reply from 157.240.39.1: TTL expired in transit.
Reply from 157.240.39.1: TTL expired in transit.

Ping statistics for 157.240.242.35:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
C:\Users\PRANAV>ping -i 14 www.facebook.com

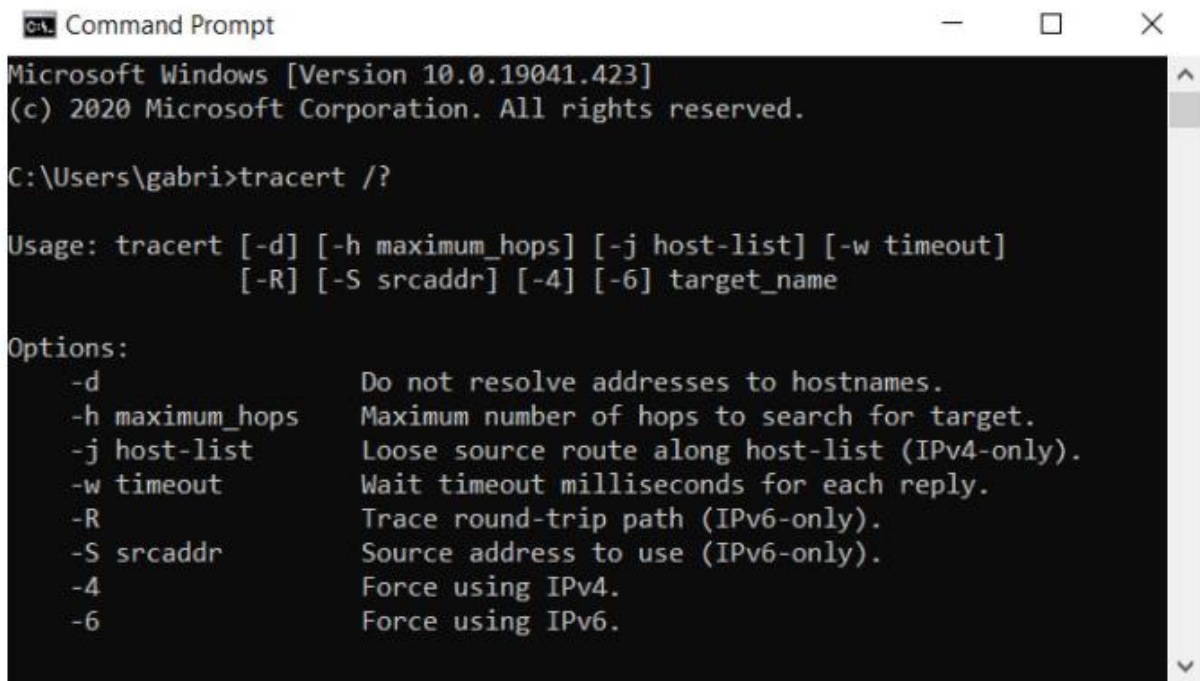
Pinging star-mini.c10r.facebook.com [157.240.242.35] with 32 bytes of data:
Reply from 157.240.242.35: bytes=32 time=31ms TTL=50
Reply from 157.240.242.35: bytes=32 time=32ms TTL=50
Reply from 157.240.242.35: bytes=32 time=33ms TTL=50
Reply from 157.240.242.35: bytes=32 time=37ms TTL=50

Ping statistics for 157.240.242.35:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 31ms, Maximum = 37ms, Average = 33ms
```


D.

Tracert command: It is used to trace the path ,which packet follows to reach the destination. It helps us diagnosing the source of many problems.

Different options available:



```
Command Prompt
Microsoft Windows [Version 10.0.19041.423]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\gabri>tracert /?

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
              [-R] [-S srcaddr] [-4] [-6] target_name

Options:
    -d                Do not resolve addresses to hostnames.
    -h maximum_hops   Maximum number of hops to search for target.
    -j host-list       Loose source route along host-list (IPv4-only).
    -w timeout         Wait timeout milliseconds for each reply.
    -R                Trace round-trip path (IPv6-only).
    -S srcaddr         Source address to use (IPv6-only).
    -4                Force using IPv4.
    -6                Force using IPv6.
```

- Traceroute utilizes the “time to live” field and ICMP echo packets, to track the route of packet to destination.
- IP addresses returned from tracert on google.com, facebook.com were in IPV4 only.
- In below outputs, some of initial IP addresses 10.*.*.*,192.* are private IP addresses.
- By using option -4, we can force tracert to use IPV4 ie: tracert -4 <domain_name> , for eg: tracert -4 www.google.com.
- In both the results given below, all gateways use IPV4. Also tracert uses IPV4 by default.

```

C:\Users\PRANAV>tracert www.google.com

Tracing route to www.google.com [216.58.200.196]
over a maximum of 30 hops:

  1     3 ms     1 ms     1 ms  192.168.0.1
  2     4 ms     1 ms     1 ms  192.168.5.1
  3     3 ms     3 ms     3 ms  10.0.0.1
  4    13 ms    13 ms    12 ms  103.74.244.1
  5    10 ms     9 ms     8 ms  45.120.248.31
  6    16 ms    14 ms    13 ms  74.125.243.97
  7    12 ms    12 ms    10 ms  172.253.51.5
  8      *     16 ms     8 ms  del11s07-in-f4.1e100.net [216.58.200.196]

Trace complete.

C:\Users\PRANAV>tracert www.facebook.com

Tracing route to star-mini.c10r.facebook.com [157.240.198.35]
over a maximum of 30 hops:

  1     4 ms     1 ms     1 ms  192.168.0.1
  2     4 ms     2 ms     2 ms  192.168.5.1
  3     4 ms     2 ms     2 ms  10.0.0.1
  4    119 ms    2 ms     2 ms  103.74.244.1
  5    16 ms    12 ms    17 ms  as32934.del.extreme-ix.net [45.120.248.45]
  6     9 ms     9 ms     8 ms  po104.psw04.del1.tfbnw.net [157.240.50.231]
  7    16 ms    15 ms    12 ms  157.240.39.1
  8    12 ms     8 ms     8 ms  edge-star-mini-shv-01-del1.facebook.com [157.240.198.35]

Trace complete.

C:\Users\PRANAV>tracert -4 www.facebook.com

Tracing route to star-mini.c10r.facebook.com [157.240.198.35]
over a maximum of 30 hops:

  1     2 ms     1 ms     1 ms  192.168.0.1
  2     3 ms    28 ms     2 ms  192.168.5.1
  3     7 ms     2 ms     3 ms  10.0.0.1
  4    10 ms     2 ms     2 ms  103.74.244.1
  5    29 ms    23 ms    31 ms  as32934.del.extreme-ix.net [45.120.248.45]
  6    16 ms    14 ms    16 ms  po104.psw04.del1.tfbnw.net [157.240.50.231]
  7     8 ms     8 ms     8 ms  157.240.39.1
  8    13 ms    12 ms     9 ms  edge-star-mini-shv-01-del1.facebook.com [157.240.198.35]

Trace complete.

```

E)iitd.ac.in:

- I observed that routers 6,13,14,15,16,17,18 did not acknowledge the packets in the given time period.

- ‘* * *’ at any hop represents that the particular gateway either send ICMP “time exceeded messages” with a too small TTL to reach us or doesn’t send them at all.
- Other reasons could be, blocking of request by some firewall, or due to security reasons router doesn’t allowed, or due to presence of lot of traffic.
- One can use `tracert -T <domainname>` or `tracert -I <domainname>`, to avoid this problem. “T” stands for TCP and “I” stands for ICMP. Because Traceroute uses UDP by default which is unreliable, thus using this flag will make traceroute use ICMP ECHO messages.
- Or we can increase the number of probes per ttl which is generally 3 by default.
- We can see that Last 2 addresses are same .This is possible when there is a firewall at the destination node and that firewall is answering the traceroute requests with the IP address of that node or destination NAT is configured on the firewall at the destination IP or this is possible when the traffic is passing through the firewall which is having a global NAT rule at that point. Every returned packet from that point and beyond will have the NAT address, not the real IP of the hop that decremented the TTL and sent an icmp echo reply.

```
C:\Users\PRANAV>tracert www.iitd.ac.in

Tracing route to www.iitd.ac.in [103.27.9.24]
over a maximum of 30 hops:

  0  1 ms    1 ms    1 ms   192.168.0.1
  1  3 ms    2 ms    1 ms   192.168.5.1
  2  3 ms    5 ms    2 ms   10.0.0.1
  3 23 ms    5 ms    6 ms   103.74.244.1
  4  8 ms   10 ms    5 ms   static-197.198.99.14-tataidc.co.in [14.99.198.197]
  5  *        *        *      Request timed out.
  6  9 ms    5 ms    8 ms   10.43.147.42
  7  6 ms    5 ms    6 ms   14.141.116.253.static-Delhi.vsnl.net.in [14.141.116.253]
  8  9 ms    5 ms    5 ms   172.31.169.86
  9  9 ms    7 ms    7 ms   172.23.185.30
 10 11 ms   16 ms    8 ms   115.110.210.38.static-Delhi.vsnl.net.in [115.110.210.38]
 11  9 ms   10 ms   11 ms   136.232.148.254.static.jio.com [136.232.148.254]
 12  *        *        *      Request timed out.
 13  *        *        *      Request timed out.
 14  *        *        *      Request timed out.
 15  *        *        *      Request timed out.
 16  *        *        *      Request timed out.
 17  *        *        *      Request timed out.
 18  *        *        *      Request timed out.
 19 19 ms   17 ms   11 ms   103.27.9.24
 20 12 ms   11 ms   11 ms   103.27.9.24
 21 14 ms   11 ms   12 ms   103.27.9.24
```

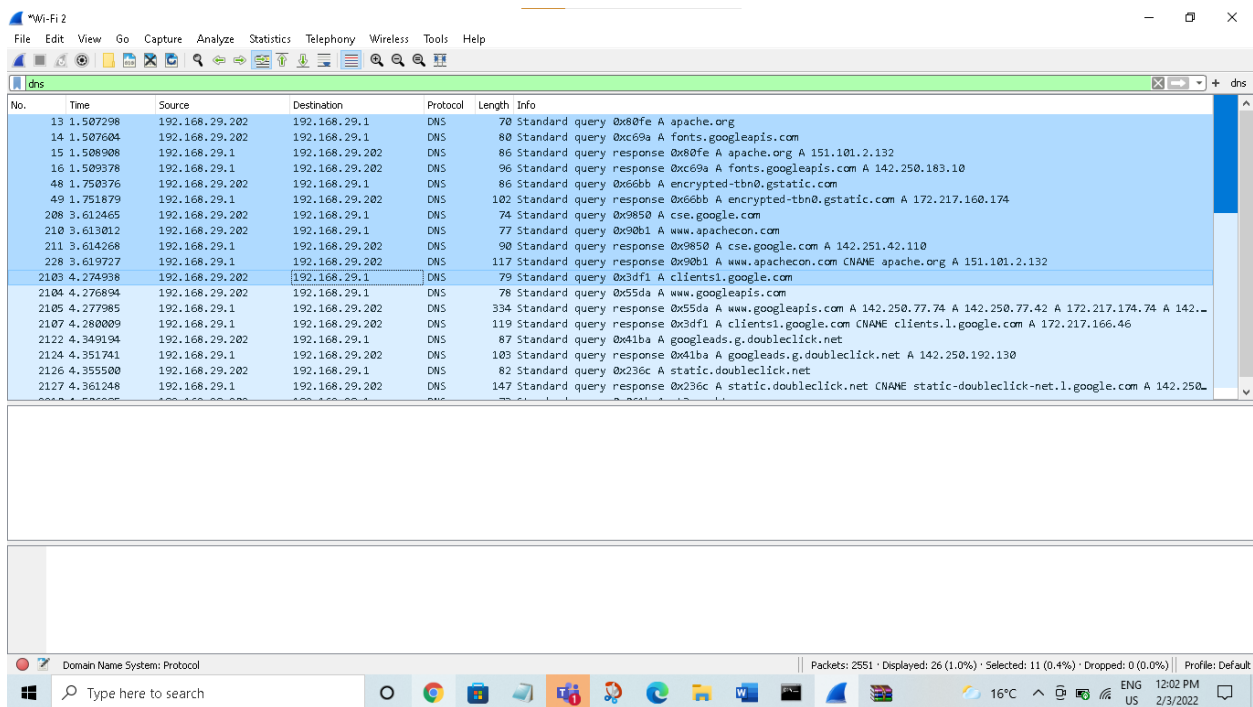
2.Packet Analysis:

a. Apache DNS:

Start time:1.507298(seconds since beginning of capture)

End time:1.508908

Total time taken to get IP address of apache.org:0.1610 sec



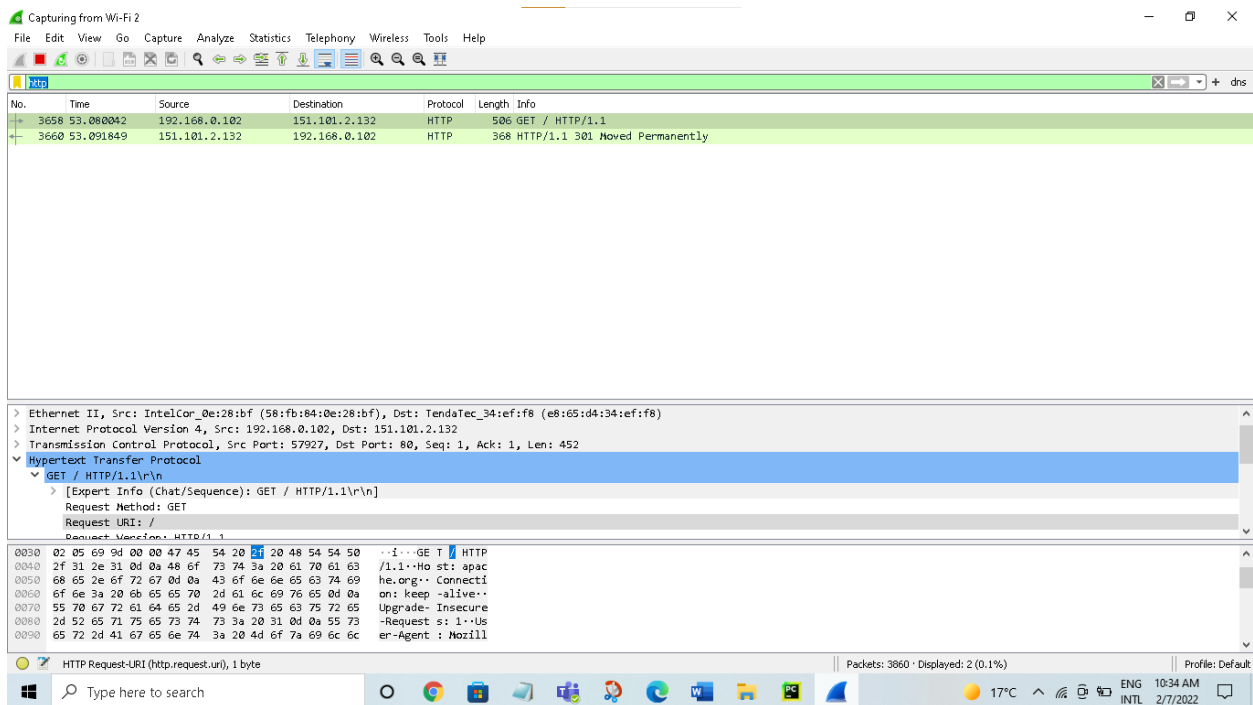
No.	Time	Source	Destination	Protocol	Length	Info
13	1.507298	192.168.29.202	192.168.29.1	DNS	70	Standard query 0x80fe A apache.org
14	1.507604	192.168.29.202	192.168.29.1	DNS	80	Standard query 0xc69a A fonts.googleapis.com
15	1.508908	192.168.29.1	192.168.29.202	DNS	86	Standard query response 0x80fe A apache.org A 151.101.2.132
16	1.509378	192.168.29.1	192.168.29.202	DNS	96	Standard query response 0xc69a A fonts.googleapis.com A 142.250.183.10
48	1.750376	192.168.29.202	192.168.29.1	DNS	86	Standard query 0x66bb A encrypted-tbn0.gstatic.com
49	1.751879	192.168.29.1	192.168.29.202	DNS	102	Standard query response 0x66bb A encrypted-tbn0.gstatic.com A 172.217.160.174
208	3.612465	192.168.29.202	192.168.29.1	DNS	74	Standard query 0x9850 A cse.google.com
210	3.613012	192.168.29.202	192.168.29.1	DNS	77	Standard query 0x98b1 A www.apachecon.com
211	3.614268	192.168.29.1	192.168.29.202	DNS	90	Standard query response 0x9850 A cse.google.com A 142.251.42.110
228	3.619727	192.168.29.1	192.168.29.202	DNS	117	Standard query response 0x98b1 A www.apachecon.com CNAME apache.org A 151.101.2.132
2103	4.274938	192.168.29.202	192.168.29.1	DNS	79	Standard query 0x3df1 A clients1.google.com
2104	4.276894	192.168.29.202	192.168.29.1	DNS	78	Standard query 0x55da A www.googleapis.com
2105	4.277985	192.168.29.1	192.168.29.202	DNS	334	Standard query response 0x55da A www.googleapis.com A 142.250.77.74 A 142.250.77.42 A 172.217.174.74 A 142...
2107	4.280009	192.168.29.1	192.168.29.202	DNS	119	Standard query response 0x3df1 A clients1.google.com CNAME clients1.google.com A 172.217.166.46
2122	4.349194	192.168.29.202	192.168.29.1	DNS	87	Standard query 0x41ba A googleads.g.doubleclick.net
2124	4.351741	192.168.29.1	192.168.29.202	DNS	103	Standard query response 0x41ba A googleads.g.doubleclick.net A 142.250.192.130
2126	4.355500	192.168.29.202	192.168.29.1	DNS	82	Standard query 0x236c A static.doubleclick.net
2127	4.361248	192.168.29.1	192.168.29.202	DNS	147	Standard query response 0x236c A static.doubleclick.net CNAME static-doubleclick-net.l.google.com A 142.250...

b. Apache HTTP:

This is output that I got on my system. After applying HTTP filter on <http://apache.org>, I just found one HTTP GET request and in response 301 Moved Permanently, which is used for permanent redirecting. For upgrading users from HTTP to HTTPS, the 301 redirects are considered the best practice. Then I clicked on response packet, and found a message in http text file where it was showing

redirection to <https://apache.org> (http changed to https). So because apache.org website uses HTTPS, I got this result .

HTTPS is very secure than HTTP, because it uses TLS(SSL) to perform encryption of http requests and responses.



c. Total Time:

- Time taken to download the whole webpage is calculated by subtracting the time at which the last content object was received from the time at which the first DNS request was received.
- I obtained last content ie: application data from apache.org at 10:1695 sec.
- So on subtracting 10:1695 from 1.507298 , we get webpage download time.

99	10.149656	192.168.0.102	8.8.8.8	DNS	74 Standard query 0xe2fc A cse.google.com
100	10.154901	151.101.2.132	192.168.0.102	TCP	66 443 → 50994 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1440 SACK_PERM=1 WS=512
101	10.155011	192.168.0.102	151.101.2.132	TCP	54 50994 → 443 [ACK] Seq=1 Ack=1 Win=132352 Len=0
102	10.155254	192.168.0.102	151.101.2.132	TLSv1.3	571 Client Hello
103	10.159421	192.168.0.102	216.58.196.106	QUIC	115 0-RTT, DCID=d8bdfaec1bb37210
104	10.160163	192.168.0.102	172.217.166.3	QUIC	120 0-RTT, DCID=cd744efcaa3035d1
105	10.161242	151.101.2.132	192.168.0.102	TCP	54 443 → 50994 [ACK] Seq=1 Ack=518 Win=147456 Len=0
106	10.166383	151.101.2.132	192.168.0.102	TLSv1.3	1494 Server Hello, Change Cipher Spec, Application Data
107	10.166818	151.101.2.132	192.168.0.102	TCP	1494 443 → 50994 [ACK] Seq=1441 Ack=518 Win=147456 Len=1440 [TCP segment of a reassembled PDU]
108	10.166875	192.168.0.102	151.101.2.132	TCP	54 50994 → 443 [ACK] Seq=518 Ack=2881 Win=132352 Len=0
109	10.168019	151.101.2.132	192.168.0.102	TCP	1494 443 → 50994 [ACK] Seq=2881 Ack=518 Win=147456 Len=1440 [TCP segment of a reassembled PDU]
110	10.169399	151.101.2.132	192.168.0.102	TLSv1.3	986 Application Data, Application Data, Application Data
111	10.169512	192.168.0.102	151.101.2.132	TCP	54 50994 → 443 [ACK] Seq=518 Ack=5253 Win=132352 Len=0
112	10.179389	216.58.196.106	192.168.0.102	QUIC	1292 Initial, SCID=d8bdfaec1bb37210, PKN: 1, ACK, PADDING
113	10.181242	172.217.166.3	192.168.0.102	QUIC	1292 Initial, SCID=cd744efcaa3035d1, PKN: 1, ACK, PADDING
114	10.189319	192.168.0.102	4.2.2.2	DNS	74 Standard query 0xe2fc A cse.google.com
115	10.220830	172.217.166.3	192.168.0.102	QUIC	1292 Protected Payload (KP0)

d. cse.iitd.ac.in HTTP:

After applying HTTP filter on cse.iitd.ac.in, I just found one HTTP GET request and in response 301 Moved Permanently, which is used for permanent redirecting. For upgrading users from HTTP to HTTPS, the 301 redirects are considered the best practice. So I did not find the http traffic same as that could be found on any http://.... website. Then I clicked on response packet, and found a message in http text file where it was showing redirection to <https://iitd.ac.in> (http changed to https). So because cse. iitd website uses HTTPS, I got this result .

HTTPS is very secure than HTTP, because it uses TLS(SSL) to perform encryption of http requests and responses.

The screenshot shows the Wireshark interface with the 'http' filter applied. The packet list pane displays two packets: a GET request (No. 534) and a 301 Moved Permanently response (No. 538). The selected packet (No. 538) is expanded in the packet details pane, showing the 'Hypertext Transfer Protocol' section with the status '301 Moved Permanently' and the 'Location' header pointing to 'https://iitd.ac.in'. The packet bytes pane shows the raw data of the response, including the status line 'HTTP/1.1 301 Moved Permanently' and the 'Location' header.

3.Traceroute using ping:

Code file is given in traceroute.py file (in python).

As traceroute uses “time to live” field to make every gateway which is present on the path to destination, respond with TIME_EXCEEDED message. I utilized the same approach as used by traceroute.

How code works:

- 1.User is asked to type the domain name, for eg: www.google.com.
- 2.Then, ICMP requests are sent to the destination and every received ICMP failed response gives the IP address of the router at which TTL expired. Because if at some router TTL becomes 0, and that router is not a destination then ICMP “time exceeded message” response is sent back to the source. So result screen shows the IP address of every such router.

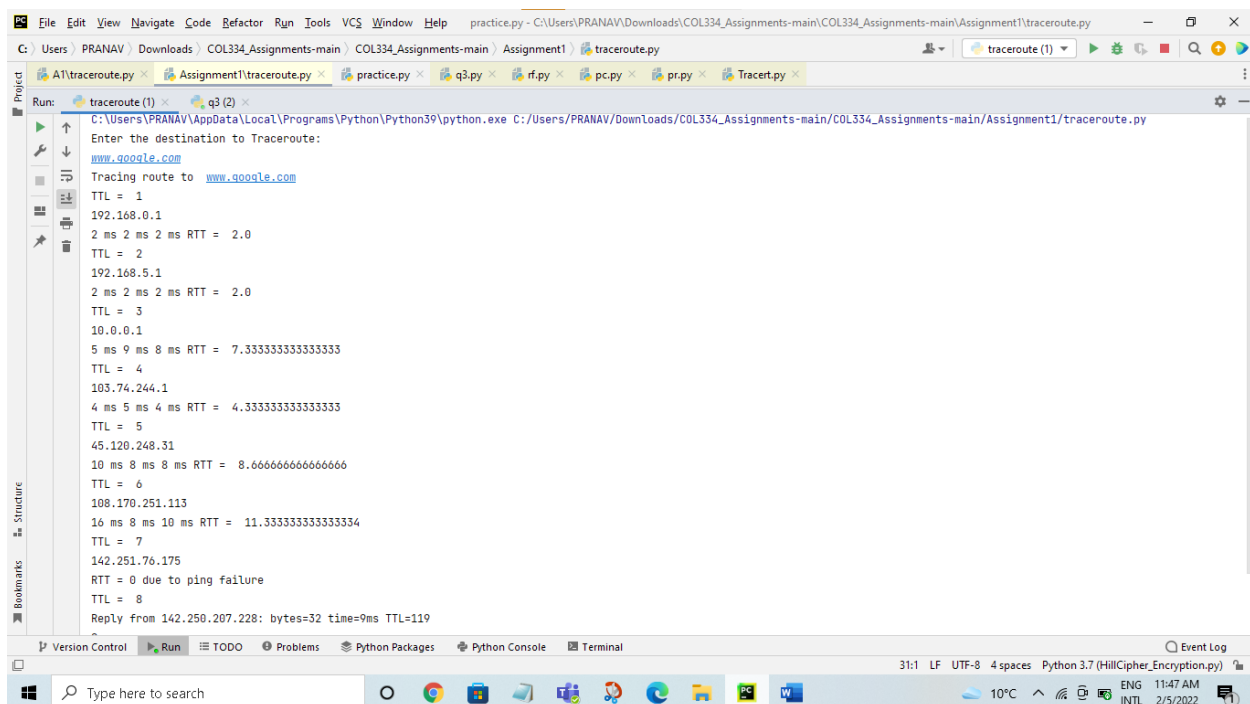
3. Now as our main aim is to get RTT to reach every router that is present on route to destination, so after getting the IP address of the router at which TTL expired, that IP address is pinged to get the corresponding RTT.

4. Also there is one assumption, that during the entire run of traceroute, path followed by the ping requests remains the same.

Observations:

1. With number of hops, RTT value is also increasing.

2. We get destination address in IPV6, if we ping using IPV6.



The screenshot shows a Python script named `traceroute.py` being executed in an IDE. The script performs a traceroute to `www.google.com`. The output shows the path taken by the packets, including the IP addresses of the routers and the Round Trip Time (RTT) for each hop. The traceroute shows 8 hops, with the final hop being the destination IP `142.250.207.228`. The RTT values for each hop are: 2.0, 2.0, 7.333333333333333, 4.333333333333333, 8.666666666666666, 11.333333333333334, and 0 (due to ping failure). The final hop shows a successful ping with 32 bytes and a 9ms RTT.

```
Run: C:\Users\PRANAV\AppData\Local\Programs\Python\Python39\python.exe C:/Users/PRANAV/Downloads/COL334_Assignments-main/COL334_Assignments-main/Assignment1/traceroute.py
Enter the destination to Traceroute:
www.google.com
Tracing route to www.google.com
TTL = 1
192.168.0.1
2 ms 2 ms 2 ms RTT = 2.0
TTL = 2
192.168.5.1
2 ms 2 ms 2 ms RTT = 2.0
TTL = 3
10.0.0.1
5 ms 9 ms 8 ms RTT = 7.333333333333333
TTL = 4
103.74.244.1
4 ms 5 ms 4 ms RTT = 4.333333333333333
TTL = 5
45.120.248.31
10 ms 8 ms 8 ms RTT = 8.666666666666666
TTL = 6
108.170.251.113
16 ms 8 ms 10 ms RTT = 11.333333333333334
TTL = 7
142.251.76.175
RTT = 0 due to ping failure
TTL = 8
Reply from 142.250.207.228: bytes=32 time=9ms TTL=119
```

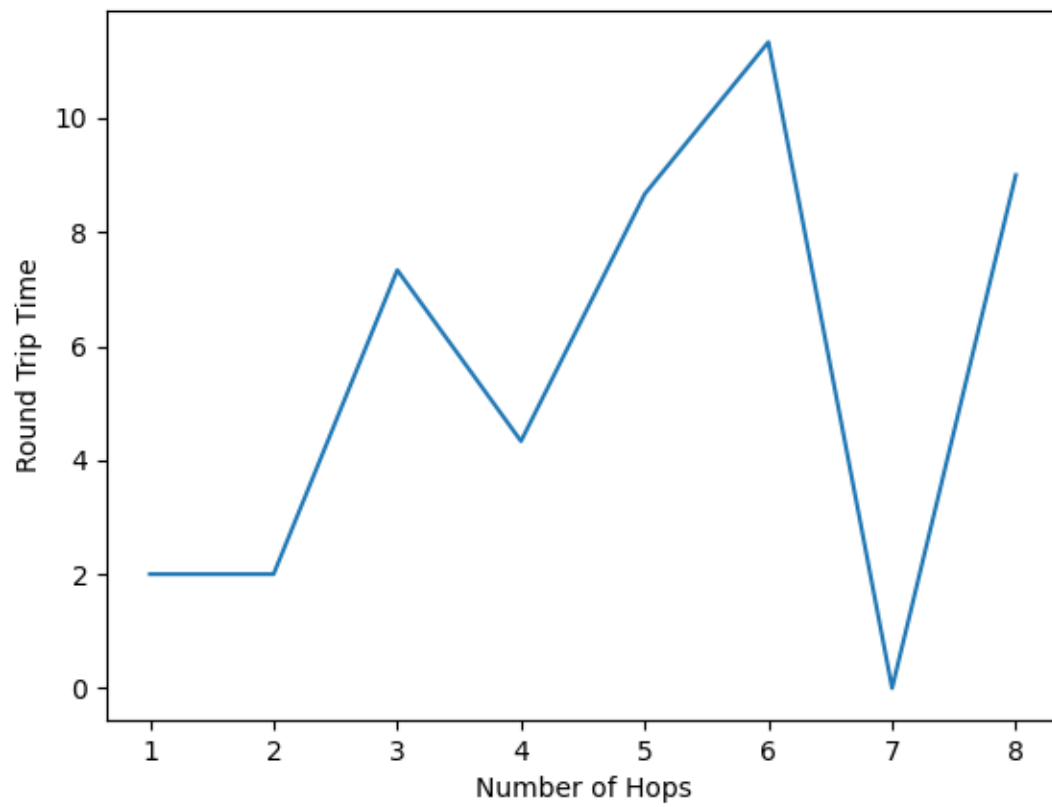



FIG: RTT vs HOP plot for www.google.com