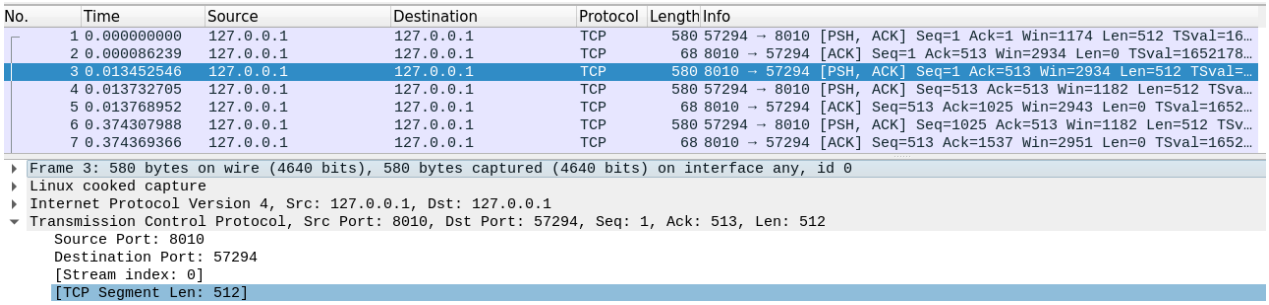
CN Assignment-2
Hardik Garg, 2019040

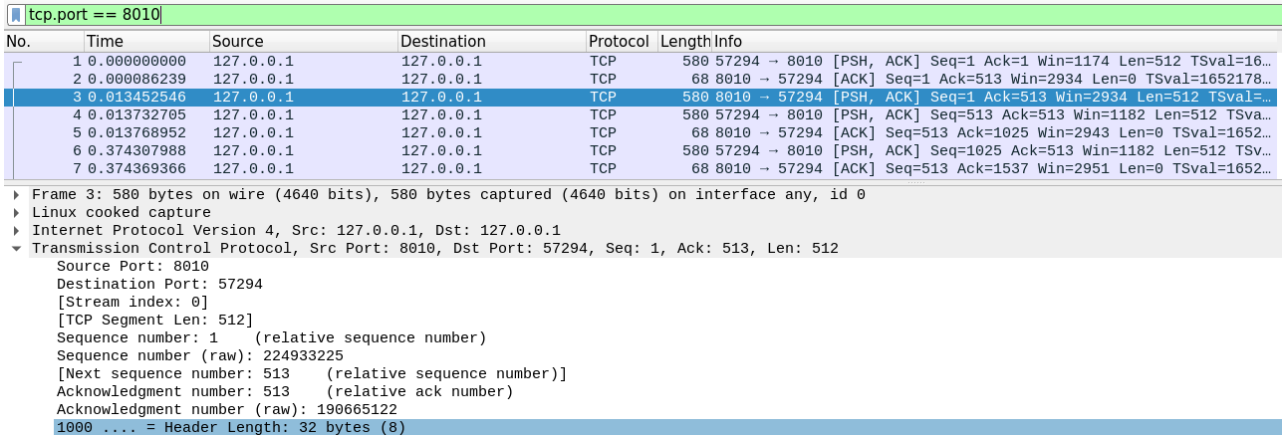
Q1

Screenshot

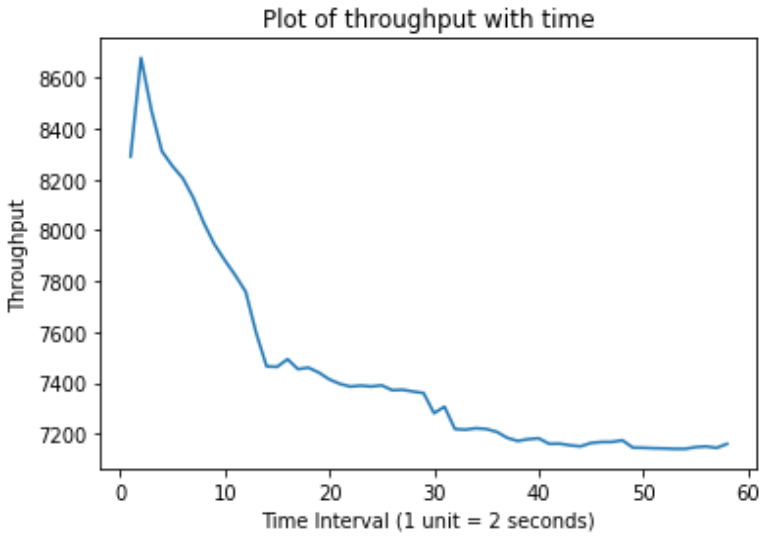


Above screenshot shows packet capture (packet trace) of “any” network interfaces via wireshark.

Screenshot



Above screenshot shows applying a filter to the “any” network interface packet trace to get only the packets corresponding to the TCP socket program written for A-1. This is done by querying the port number for the server (8010 in this case). A number of such packets were sent manually for 120 seconds. A graph was plotted for the throughput every 2 seconds. The final graph is as shown -

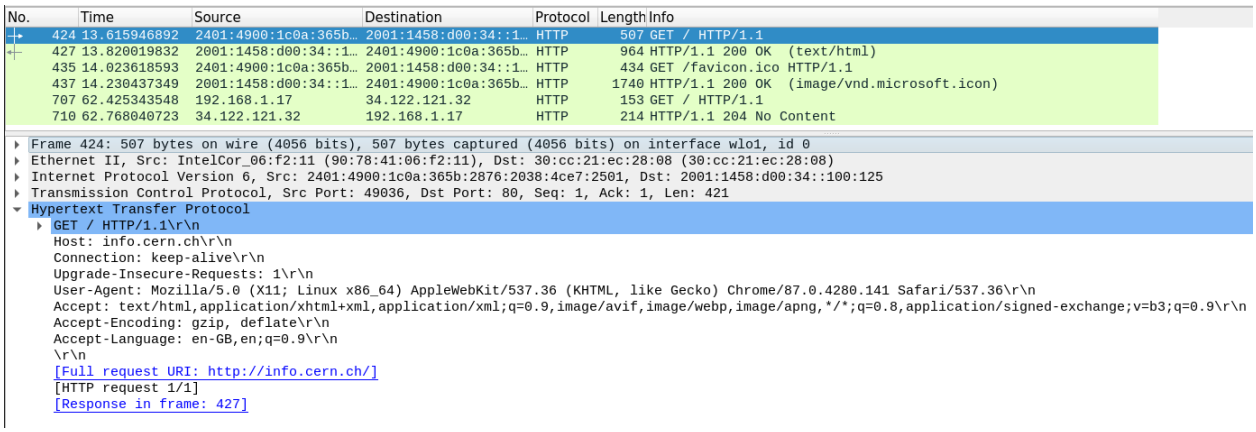


As we observe the throughput decreases with each 2 second time quanta. This happens because with time, we send more packets which leads to network congestion and traffic. This also increases the probability of packet miss which leads to more time per packet to reach its destination. Also note the sharp (but brief increase in the beginning is justified as there is no congestion in the beginning)

Q2

The details of the 5 packets captured are -

1. Screenshot Packet-1

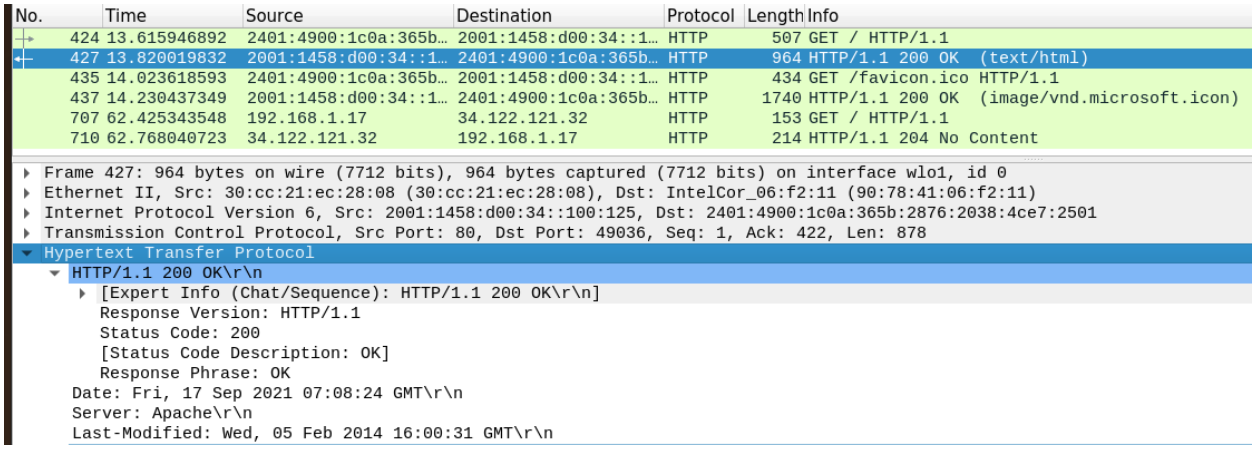


No.	Time	Source	Destination	Protocol	Length	Info
424	13.615946892	2401:4900:1c0a:365b…	2001:1458:d00:34::1…	HTTP	507	GET / HTTP/1.1
427	13.820019832	2001:1458:d00:34::1…	2401:4900:1c0a:365b…	HTTP	964	HTTP/1.1 200 OK (text/html)
435	14.023618593	2401:4900:1c0a:365b…	2001:1458:d00:34::1…	HTTP	434	GET /favicon.ico HTTP/1.1
437	14.230437349	2001:1458:d00:34::1…	2401:4900:1c0a:365b…	HTTP	1740	HTTP/1.1 200 OK (image/vnd.microsoft.icon)
707	62.425343548	192.168.1.17	34.122.121.32	HTTP	153	GET / HTTP/1.1
710	62.768040723	34.122.121.32	192.168.1.17	HTTP	214	HTTP/1.1 204 No Content

```
Frame 424: 507 bytes on wire (4056 bits), 507 bytes captured (4056 bits) on interface wlo1, id 0
Ethernet II, Src: IntelCor_06:f2:11 (90:78:41:06:f2:11), Dst: 30:cc:21:ec:28:08 (30:cc:21:ec:28:08)
Internet Protocol Version 6, Src: 2401:4900:1c0a:365b:2876:2038:4ce7:2501, Dst: 2001:1458:d00:34::100:125
Transmission Control Protocol, Src Port: 49036, Dst Port: 80, Seq: 1, Ack: 1, Len: 421
Hypertext Transfer Protocol
  GET / HTTP/1.1\r\n
  Host: info.cern.ch\r\n
  Connection: keep-alive\r\n
  Upgrade-Insecure-Requests: 1\r\n
  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36\r\n
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
  Accept-Encoding: gzip, deflate\r\n
  Accept-Language: en-GB,en;q=0.9\r\n
  \r\n
  [Full request URI: http://info.cern.ch/]
  [HTTP request 1/1]
  [Response in frame: 427]
```

   - HTTP Request Packet
   - GET type request
   - User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36\r\n (chrome web browser type user agent)
   - Request URL - [http://info.cern.ch/]
   - Browser - Chrome (version - 87.0.4280.141)

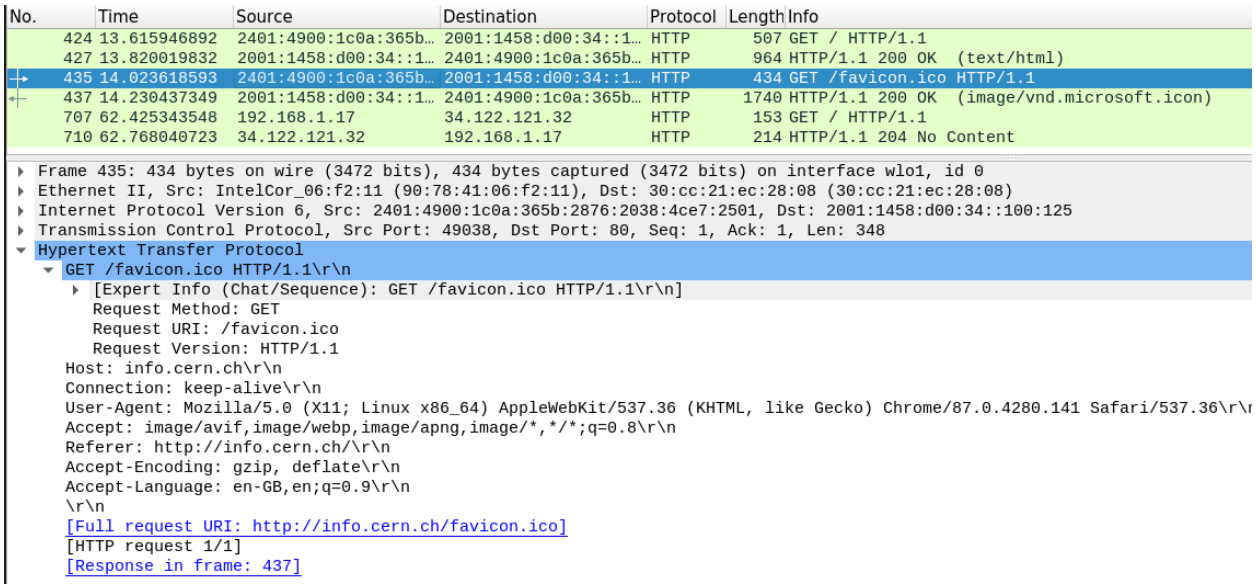
2. Screenshot Packet-2



```
No.  Time          Source                Destination           Protocol Length Info
 424 13.615946892  2401:4900:1c0a:365b… 2001:1458:d00:34::1… HTTP      507 GET / HTTP/1.1
 427 13.820019832  2001:1458:d00:34::1… 2401:4900:1c0a:365b… HTTP      964 HTTP/1.1 200 OK  (text/html)
 435 14.023618593  2401:4900:1c0a:365b… 2001:1458:d00:34::1… HTTP      434 GET /favicon.ico HTTP/1.1
 437 14.230437349  2001:1458:d00:34::1… 2401:4900:1c0a:365b… HTTP     1740 HTTP/1.1 200 OK  (image/vnd.microsoft.icon)
 707 62.425343548  192.168.1.17          34.122.121.32         HTTP      153 GET / HTTP/1.1
 710 62.768040723  34.122.121.32         192.168.1.17          HTTP      214 HTTP/1.1 204 No Content

▸ Frame 427: 964 bytes on wire (7712 bits), 964 bytes captured (7712 bits) on interface wlo1, id 0
▸ Ethernet II, Src: 30:cc:21:ec:28:08 (30:cc:21:ec:28:08), Dst: IntelCor_06:f2:11 (90:78:41:06:f2:11)
▸ Internet Protocol Version 6, Src: 2001:1458:d00:34::100:125, Dst: 2401:4900:1c0a:365b:2876:2038:4ce7:2501
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 49036, Seq: 1, Ack: 422, Len: 878
▾ Hypertext Transfer Protocol
  ▾ HTTP/1.1 200 OK\r\n
    ▸ [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
      Response Version: HTTP/1.1
      Status Code: 200
      [Status Code Description: OK]
      Response Phrase: OK
    Date: Fri, 17 Sep 2021 07:08:24 GMT\r\n
    Server: Apache\r\n
    Last-Modified: Wed, 05 Feb 2014 16:00:31 GMT\r\n
```

- HTTP Response Packet
- HTTP response code - 200
- HTTP Response Description - OK (success)

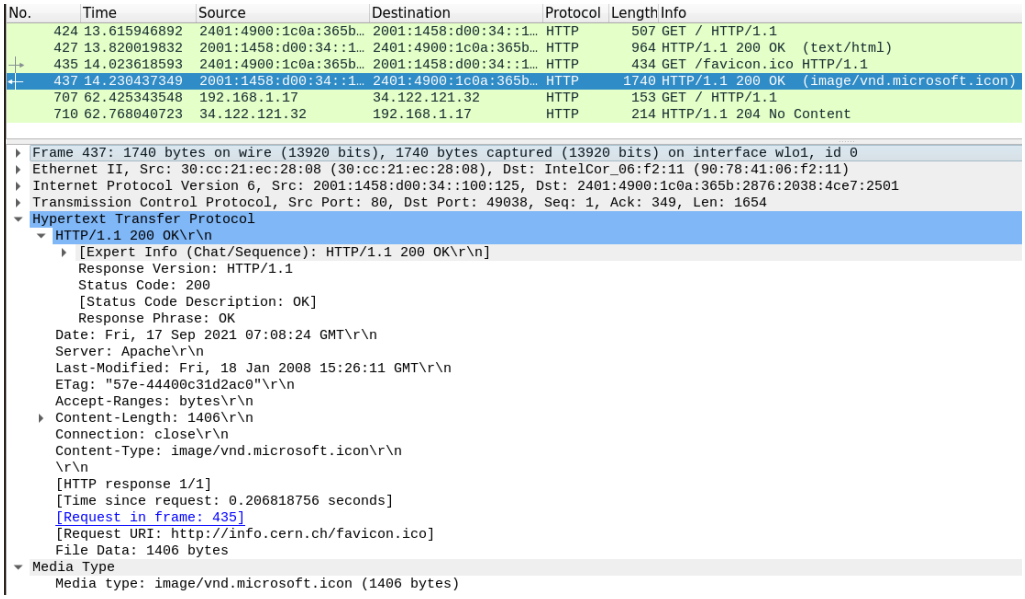
3. Screenshot Packet-3



```
No.  Time          Source                Destination           Protocol Length Info
 424 13.615946892  2401:4900:1c0a:365b… 2001:1458:d00:34::1… HTTP      507 GET / HTTP/1.1
 427 13.820019832  2001:1458:d00:34::1… 2401:4900:1c0a:365b… HTTP      964 HTTP/1.1 200 OK  (text/html)
 435 14.023618593  2401:4900:1c0a:365b… 2001:1458:d00:34::1… HTTP      434 GET /favicon.ico HTTP/1.1
 437 14.230437349  2001:1458:d00:34::1… 2401:4900:1c0a:365b… HTTP     1740 HTTP/1.1 200 OK  (image/vnd.microsoft.icon)
 707 62.425343548  192.168.1.17          34.122.121.32         HTTP      153 GET / HTTP/1.1
 710 62.768040723  34.122.121.32         192.168.1.17          HTTP      214 HTTP/1.1 204 No Content

▸ Frame 435: 434 bytes on wire (3472 bits), 434 bytes captured (3472 bits) on interface wlo1, id 0
▸ Ethernet II, Src: IntelCor_06:f2:11 (90:78:41:06:f2:11), Dst: 30:cc:21:ec:28:08 (30:cc:21:ec:28:08)
▸ Internet Protocol Version 6, Src: 2401:4900:1c0a:365b:2876:2038:4ce7:2501, Dst: 2001:1458:d00:34::100:125
▸ Transmission Control Protocol, Src Port: 49038, Dst Port: 80, Seq: 1, Ack: 1, Len: 348
▾ Hypertext Transfer Protocol
  ▾ GET /favicon.ico HTTP/1.1\r\n
    ▸ [Expert Info (Chat/Sequence): GET /favicon.ico HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /favicon.ico
      Request Version: HTTP/1.1
    Host: info.cern.ch\r\n
    Connection: keep-alive\r\n
    User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36\r\n
    Accept: image/avif,image/webp,image/apng,image/*,*/*;q=0.8\r\n
    Referer: http://info.cern.ch/\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-GB,en;q=0.9\r\n
    \r\n
    [Full request URI: http://info.cern.ch/favicon.ico]
    [HTTP request 1/1]
    [Response in frame: 437]
```

- HTTP Request Packet
- GET type request
- User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36\r\n (chrome web browser type user agent)
- Request URL - [http://info.cern.ch/favicon.ico]
- Browser - Chrome (version - 87.0.4280.141)

4. Screenshot Packet-4



- HTTP Response Packet
- HTTP response code - 200
- HTTP Response Description - OK (success)

Q3

(a)

```
hardeekh@hardeekh-Inspiron:~$ ifconfig wlo1
wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.17  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::408c:26b3:4875:a1f1  prefixlen 64  scopeid 0x20<link>
        inet6 2401:4900:1c0a:365b:65d3:3180:74bc:e32d  prefixlen 64  scopeid 0x0<global>
        inet6 2401:4900:1c0a:365b:1fba:a099:66c1:bf61  prefixlen 64  scopeid 0x0<global>
        ether 90:78:41:06:f2:11  txqueuelen 1000  (Ethernet)
        RX packets 14454193  bytes 14714987149 (14.7 GB)
        RX errors 0  dropped 2775  overruns 0  frame 0
        TX packets 7337282  bytes 3736186598 (3.7 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

IP address of my network interface is **192.168.1.17**

(b)

IP address using whatsmyip website is **122.161.223.97**

As we observe, both of them are different. Ifconfig shows the local IP address whereas whatsmyip shows the IP address provided by the ISP. The address provided by the ISP is different because we are connected via many routers to the ISP (not directly), therefore, the IP address changes and is masked. Other factors such as location and use of proxies also play a role. Also, if two devices go through the same series of routers, then their local IPs will be different but the ip obtained using whatsmyip will be the same.

Q4
(a)

The command used is -
i. **sudo ifconfig wlo1 mtu 3000** (set mtu for the interface to 3000)
ii. **ping www.iiitd.ac.in -s 3000 -c 1** (ping 1 packet of size 3000 bytes)

Output

```
hardeekh@hardeekh-Inspiron:~$ sudo ifconfig wlo1 mtu 3000
SIOCSIFMTU: Invalid argument
hardeekh@hardeekh-Inspiron:~$ ping www.iiitd.ac.in -s 3000 -c 1
PING iiitd.ac.in (103.25.231.30) 3000(3028) bytes of data.

--- iiitd.ac.in ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

Explanation - MTU stands for Maximum Transmission Unit and tells the maximum size of a packet which can be transmitted over the network. We get SIOCSIFMTU error here which arises when MTU value is set out of range. The max possible value I could set in my machine is 2304 which is less than 3000 (the default value was 1500)

```
hardeekh@hardeekh-Inspiron:~$ sudo ifconfig wlo1 mtu 2304
hardeekh@hardeekh-Inspiron:~$ sudo ifconfig wlo1
wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 2304
```

(b)
command - **sudo netstat -vatp**
Explanation of Flags - v stand for verbose (displays additional information as needed), a stands for all (prints all active connections), t stands for TCP (specify the protocol), p stands for pid (to display process id)

Output

```
hardeekh@hardeekh-Inspiron:~$ sudo netstat -vatp
[sudo] password for hardeekh:
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 localhost:33060         0.0.0.0:*               LISTEN      895/mysqld
tcp        0      0 localhost:mysql         0.0.0.0:*               LISTEN      895/mysqld
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN      619/systemd-resolve
tcp        0      0 localhost:ipp           0.0.0.0:*               LISTEN      55721/cupsd
tcp        0      0 hardeekh-Inspiron:49986 bom12s12-in-f10.1:https TIME_WAIT   -
tcp        0      1 hardeekh-Inspiron:46272 151.101.12.193:https    FIN_WAIT1   -
tcp        0      0 hardeekh-Inspiron:34178 bom12s21-in-f13.1:https TIME_WAIT   -
tcp        0      0 hardeekh-Inspiron:46930 192.168.1.87:8009       ESTABLISHED 17316/chrome --type
tcp        0      1 hardeekh-Inspiron:36588 151.101.129.69:https    FIN_WAIT1   -
tcp        0      0 hardeekh-Inspiron:51844 192.168.1.87:8008       ESTABLISHED 17316/chrome --type
tcp6       0      0 [::]:1716               [::]:*                  LISTEN      2020/kdeconnectd
tcp6       0      0 ip6-localhost:ipp       [::]:*                  LISTEN      55721/cupsd
tcp6       0      0 hardeekh-Inspiron:48860 del03s13-in-x0a.1:https TIME_WAIT   -
tcp6       0      1 2401:4900:1c0a:36:42198 g2600-1417-002c-0:https FIN_WAIT1   -
tcp6       0      0 hardeekh-Inspiron:48838 del03s13-in-x0a.1:https TIME_WAIT   -
tcp6       0      0 hardeekh-Inspiron:48836 del03s13-in-x0a.1:https TIME_WAIT   -
tcp6       0      0 hardeekh-Inspiron:48856 del03s13-in-x0a.1:https TIME_WAIT   -
tcp6       0      0 hardeekh-Inspiron:55738 del11s12-in-x03.1:https ESTABLISHED 17316/chrome --type
tcp6       0      1 2401:4900:1c0a:36:42196 g2600-1417-002c-0:https FIN_WAIT1   -
tcp6       0      0 hardeekh-Inspiron:44006 del12s11-in-x0e.1:https ESTABLISHED 17316/chrome --type
tcp6       0      0 hardeekh-Inspiron:48844 del03s13-in-x0a.1:https TIME_WAIT   -
tcp6       0      0 hardeekh-Inspiron:55718 del11s12-in-x03.1:https ESTABLISHED 17316/chrome --type
tcp6       0      0 hardeekh-Inspiron:48852 del03s13-in-x0a.1:https TIME_WAIT   -
```

Q5

(a)

Performing nslookup on youtube.com, the commands are -

i. **sudo nslookup -type=soa youtube.com** (perform search of authority type query on youtube.com to get result to be used for authoritative result)

ii. **sudo nslookup youtube.com ns1.google.com** (do a lookup to get authoritative result)

Output

```
hardeekh@hardeekh-Inspiron:~$ nslookup -type=soa youtube.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
youtube.com
        origin = ns1.google.com
        mail addr = dns-admin.google.com
        serial = 397037882
        refresh = 900
        retry = 900
        expire = 1800
        minimum = 60

Authoritative answers can be found from:

hardeekh@hardeekh-Inspiron:~$ nslookup youtube.com ns1.google.com
Server:         ns1.google.com
Address:        2001:4860:4802:32::a#53

Name:   youtube.com
Address: 142.250.194.110
Name:   youtube.com
Address: 2404:6800:4002:821::200e
```

Explanation - To get an authoritative result, we first perform a search of authority query on the starting URL (youtube.com in our case). As a result we get the origin URL through which we can perform the authoritative lookup.

(b)
Command - **nslookup -debug www.google.com**
Output

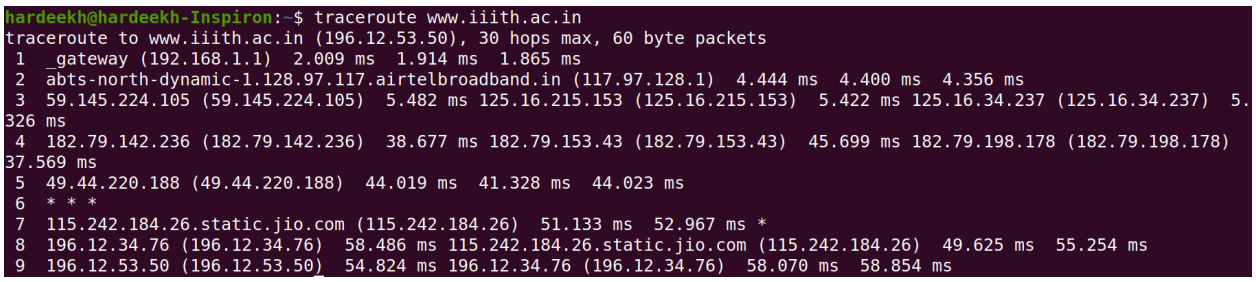
```
hardeekh@hardeekh-Inspiron:~$ nslookup -debug www.google.com
Server:         127.0.0.53
Address:        127.0.0.53#53

------------
    QUESTIONS:
        www.google.com, type = A, class = IN
    ANSWERS:
    ->  www.google.com
        internet address = 216.58.196.196
        ttl = 59
    AUTHORITY RECORDS:
    ADDITIONAL RECORDS:
------------
Non-authoritative answer:
Name:   www.google.com
Address: 216.58.196.196
------------
    QUESTIONS:
        www.google.com, type = AAAA, class = IN
    ANSWERS:
    ->  www.google.com
        has AAAA address 2404:6800:4009:82c::2004
        ttl = 87
    AUTHORITY RECORDS:
    ADDITIONAL RECORDS:
------------
Name:   www.google.com
Address: 2404:6800:4009:82c::2004
```

We get two outputs here - one corresponding to **type A which gives 59 seconds as the time to live and stands for the IPv4** type address denoted by internet address fields. The other is the **type AAAA which gives 87 seconds as the time to live and stands for the IPv6** address denoted by has AAAA address field. Packets corresponding to both of them are loaded and hence we have ttl values for both of them. Also note that these are non-authoritative results and packets stored in the cache for each of them will expire after their respective ttl values.

Q6

(a)

Screenshot



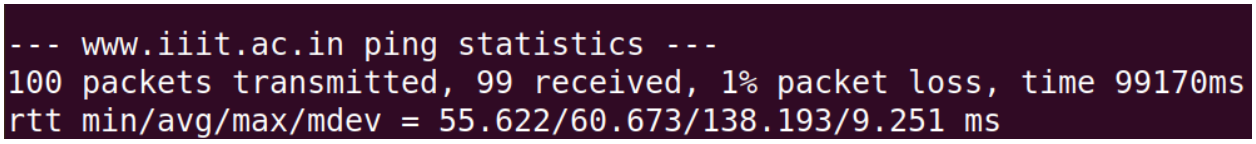
There are **9 intermediate hosts (out of which Host-6 is not visible)**. Average latency of each one of them is as follows (traceroute sends three packets to each host and returns the Round Trip Time or RTT for each of them which is also 2*latency, assuming symmetry in sending and receiving times, therefore, latency = RTT/2) -

1. (2.009+1.914+1.865)/(3*2) = 0.9645 ms
2. (4.444+4.400+4.356)/(3*2) = 2.200 ms
3. (5.482+5.422+5.326)/(3*2) = 2.705 ms
4. (38.677+45.699+37.569)/(3*2) = 20.324 ms
5. (44.019+41.328+44.023)/(3*2) = 21.662 ms
6. Host not Visible
7. (51.133+52.967)/(2*2) = 26.025 ms (packet loss in the third attempt at sending a packet)
8. (58.486+49.625+55.254)/(3*2) = 27.228 ms
9. (54.824+58.070+58.854)/(3*2) = 28.625 ms

(b)

Command - **sudo ping -c 100 www.iiith.ac.in**

Screenshot



Average Latency - 60.673/2 = 30.336 ms

(c)

Total latency of all hosts using traceroute = 1.929+4.400+5.410+40.648+43.123+54.455+57.249 = **207.214/2 = 103.607 ms**

Reasoning - the two latencies are **not the same. Latency using traceroute > Latency using ping.** This is because ping command by default works as a best-effort service, which means that packets sent through ping are simply forwarded across routers (hosts) whereas in case of traceroute, packets sent to each host also wait for a timeout response from the host before

proceeding. In short, for ping every host simply forwards packets but in traceroute every host sends back a response also (which makes traceroute a more reliable metric compared to ping)

(d)
Max Latency using traceroute = **28.625 ms** which is comparable to the average latency through ping (**30.336 ms).** The reason for this is - when we consider only one intermediate host, traceroute behaves similar to ping in the sense that there is only packet forwarding in both of them, the response is still sent in traceroute, however, it is pipelined with the forwarding time and we don't have to consider any time waiting for acknowledgements in terms of timeout response because we are considering just one host as the bottleneck host. Explanation for ping remains the same.

(e)

```
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 192.168.1.1
1.1.168.192.in-addr.arpa. 0     IN      PTR     192.168.1.1.
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 117.97.128.1
1.128.97.117.in-addr.arpa. 28800 IN     PTR     abts-north-dynamic-1.128.97.117.airtelbroadband.in.
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 122.185.42.189
189.42.185.122.in-addr.arpa. 86400 IN   PTR     nsg-corporate-189.42.185.122.airtel.in.
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 122.185.42.193
193.42.185.122.in-addr.arpa. 86388 IN   PTR     nsg-corporate-193.42.185.122.airtel.in.
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 116.119.61.121
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 182.79.142.232
;; connection timed out; no servers could be reached

hardeekh@hardeekh-Inspiron:~$ 182.79.141.180
182.79.141.180: command not found
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 182.79.141.180
;; connection timed out; no servers could be reached

hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 49.44.220.188
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 115.242.184.26
26.184.242.115.in-addr.arpa. 3600 IN    PTR     115.242.184.26.static.jio.com.
hardeekh@hardeekh-Inspiron:~$ sudo dig +noall +answer ptr,cname -x 196.12.34.76
```

To perform reverse DNS lookup (host name from IP), we use the dig command with some flags (+noall,+answer,-x) to display only the relevant details (PTR record gives us the hostname and CNAME record(s) give us the aliases, if any). Some hosts have 2-3 IPs associated with them. This is because 3 packets are sent to each host so they have similar IPs (observe first two values of the IPv4 address)

The host names and aliases for each of the hosts are shown in the screenshot above. Some host IPs took too long to respond, resulting in a connection timeout.

Q7
Commands -
i. **sudo ifconfig lo down** (shut down the lo interface)
ii. **ping 127.0.0.1** (send packets to 127.0.0.1 IP)

Explanation - through the first command, we shut down the lo (loopback interface) and then send packets to it which will lead to 100% packet loss as shown in below screenshot -

```
hardeekh@hardeekh-Inspiron:~$ sudo ifconfig lo down
hardeekh@hardeekh-Inspiron:~$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
^C
--- 127.0.0.1 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7150ms
```

In order to send packets successfully, we will have to turn on the loopback interface using [sudo ifconfig lo up]