COMPUTER NETWORKS

REPORT

Ans1.

a)

```
talhasee@talhasee:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.233.128  netmask 255.255.255.0  broadcast 192.168.233.255
        inet6 fe80::f9c0:b50f:1d7c:3089  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:2c:99:ec  txqueuelen 1000  (Ethernet)
        RX packets 91146  bytes 134230530 (134.2 MB)
        RX errors 680  dropped 746  overruns 0  frame 0
        TX packets 46302  bytes 2549469 (2.5 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  base 0x2000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 221  bytes 19737 (19.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 221  bytes 19737 (19.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

talhasee@talhasee:~$
```

My IP address – **192.168.233.128**

b)

IP address using ifconfig – **192.168.233.128**

IP address using webpage - **157.37.205.207**

Both are different because first one is **private IP** which identifies uniquely my computer in local network and second one is **public IP** which is provided by ISP to uniquely identified on wider internet.

Ans 2.

```
talhasee@talhasee:~$ nslookup -query=ns www.iiitd.ac.in
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
www.iiitd.ac.in canonical name = iiitd.ac.in.
iiitd.ac.in     nameserver = ns2.iiitd.ac.in.
iiitd.ac.in     nameserver = ns1.iiitd.ac.in.

Authoritative answers can be found from:
ns2.iiitd.ac.in internet address = 103.25.231.52

talhasee@talhasee:~$
```

We can get authoritative answer by checking NS(**nameserver** records instead of **A** records) by passing arguments `**-query = ns**`.

b)

```
talhasee@talhasee:~$ dig +ttlunits www.iiitd.ac.in

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> +ttlunits www.iiitd.ac.in
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6787
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.iiitd.ac.in.               IN      A

;; ANSWER SECTION:
www.iiitd.ac.in.        5s      IN      CNAME   iiitd.ac.in.
iiitd.ac.in.            5s      IN      A       103.25.231.30

;; Query time: 95 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Thu Sep 22 21:26:36 IST 2022
;; MSG SIZE  rcvd: 74

talhasee@talhasee:~$
```

TTL for www.iiitd.ac.in is **5s** means it will cache the record for 5s specified by TTL and if this query of caching nameserver got resolved before the TTL expired than it will rely on the cached record rather than retrieving the same information from authoritative nameservers again.

Ans 3.

a)

```
talhasee@talhasee:~$ traceroute google.in
traceroute to google.in (142.250.194.228), 30 hops max, 60 byte packets
 1  _gateway (192.168.43.220)  11.615 ms  11.439 ms  11.523 ms
 2  * * *
 3  56.14.87.137 (56.14.87.137)  57.589 ms 56.14.87.169 (56.14.87.169)  63.262 ms 56.14.87.145 (56.14.87.145)  63.532 ms
 4  192.168.44.232 (192.168.44.232)  63.194 ms 192.168.44.236 (192.168.44.236)  62.947 ms  63.202 ms
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  142.250.47.144 (142.250.47.144)  74.201 ms * 74.125.147.192 (74.125.147.192)  49.419 ms
12  209.85.148.118 (209.85.148.118)  49.185 ms  53.642 ms 72.14.195.34 (72.14.195.34)  40.173 ms
13  * * 142.251.52.228 (142.251.52.228)  54.509 ms
14  72.14.233.216 (72.14.233.216)  54.236 ms 142.250.236.54 (142.250.236.54)  54.024 ms 66.249.95.74 (66.249.95.74)  62.656 ms
15  74.125.244.196 (74.125.244.196)  62.704 ms del12s08-in-f4.1e100.net (142.250.194.228)  62.327 ms 142.251.52.217 (142.251.52.217)  62.475 ms
talhasee@talhasee:~$
```

Average latencies of each intermediate hosts

1- (11.615+11.439+11.523)/3 = 11.525

3- (57.589+63.262+63.532)/3 = 61.461

11- (74.201+0+49.419)/3 = 41.206

12- (49.185+53.642+40.173)/3 = 47.666

13- (0+0+54.509)/3 = 18.06

14- (54.236+54.024+62.656)/3 = 56.972

15- (62.704+62.327+62.475)/3 = 62.502

b)

```
--- google.in ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 99191ms
rtt min/avg/max/mdev = 26.267/51.775/230.233/26.941 ms
talhasee@talhasee:~$
```

Command – ping -c 100 google.in

Average latency – 51.775(two-way-latency)

Average latency – 25.8875ms (one-way-latency)

c)

```
--- columbia.edu ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 99170ms
rtt min/avg/max/mdev = 292.323/377.627/882.255/103.446 ms
talhasee@talhasee:~$
```

Command – ping -c 100 columbia.edu

Average latency – 377.627ms (two-way-latency)

Average latency – 188.8135ms (one-way-latency)

d)

By summing all the intermediate latencies and given one two way latencies in summary both are different because in ping command it just keep on sending packets but in traceroute it waits for response from each intermediate host.

e)

The highest ping latency among the intermediate hosts provided by traceroute command is different from average latency of provide by ping command because it's the TTL which is displaying. As in traceroute it sends a packet and moves on and if that packet doesn't responded before TTL expired then it shows the TTL in traceroute.

f)

14 hops in traceroute google.in and 26 hops in traceroute Columbia.edu because different hosts are on different networks and traceroute take different path to reach there.

Ans 4

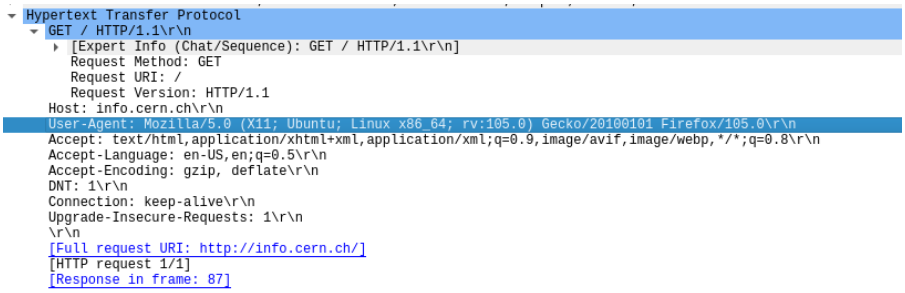
```
talhasee@talhasee:~$ ping -c 10 -s 1000000 127.0.0.1
ping: WARNING: probably, rcvbuf is not enough to hold preload
PING 127.0.0.1 (127.0.0.1) 1000000(1000028) bytes of data.

--- 127.0.0.1 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9194ms

talhasee@talhasee:~$
```

By increasing packet size more than the pipeline can handle we can make 100% packet loss.

Ans 5

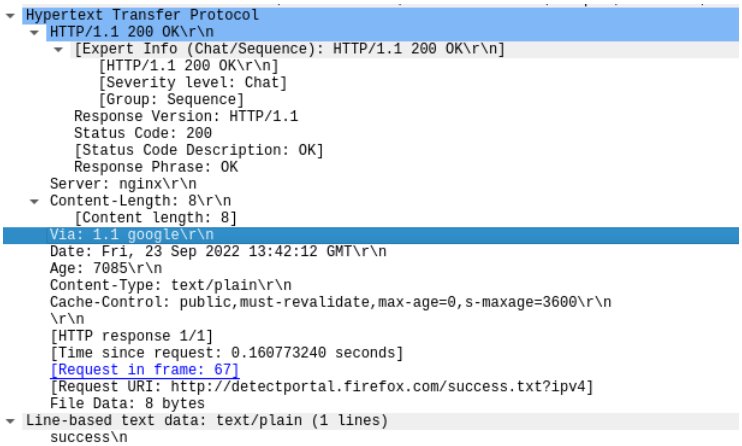
For HTTP request packets

- HTTP request type – '**GET**'
- User agent type - : Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:105.0) Gecko/20100101 Firefox/105.0\r\n
- HTTP request packet's URL – '**/**'



For HTTP response packets

- HTTP response code – '**200**'
- HTTP response description – '**OK**'
- Name and version of web server – 'nginx\r\n



Web Objects get downloaded

1 object (i.e., icon image) got downloaded and it is done by opening different tcp connections.

On the basis of above statement, it is opening multiple tcp connection for sending every new packet. Hence, it is **non-persistent.**

Ans 6

a) sudo netstat -tp

b)

```
talhasee@talhasee:~$ sudo netstat -tp
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 talhasee:51002          82.221.107.34.bc.g:http ESTABLISHED 2394/firefox
tcp        0      0 talhasee:42898          ec2-35-164-56-167:https ESTABLISHED 2394/firefox
tcp        0      0 talhasee:46270          server-18-66-63-1:https ESTABLISHED 2394/firefox
tcp        0      0 talhasee:48998          123.208.120.34.bc:https ESTABLISHED 2394/firefox
tcp        0      0 talhasee:51010          82.221.107.34.bc.g:http ESTABLISHED 2394/firefox
tcp        0      0 talhasee:36464          webafs706.cern.ch:http  ESTABLISHED 2394/firefox
talhasee@talhasee:~$
```