SIBA UNIVERSITY

Computer Science

COMPUTER NETWORKING

LAB No. 1

Name: Aamir Hussain

CMS ID: 023-22-0080

BSCS-V(E)

Submit Date:28/08/2024

Lab Activities

1: Know your communication interface
unplug the RJ 45 or ethernet cable or UTP cable
replug the connector and observe the connectivity process

Ans:

I. **First, we identify our network interface:** Before unplugging the Ethernet cable, it's good to know which network interface on your system is handling the connection.

Open “cmd” and enter command**: “ipconfig /all”**

then check network activity by this command.

II. **Unplug the RJ45 (Ethernet) Cable**: Physically disconnect the Ethernet cable from your laptop or desktop.

Observe the Change in Network Status: As soon as you unplug the cable, you will likely see a notification on your screen indicating that the network connection has been lost.

III. **Replug the Ethernet Cable:** Reconnect the Ethernet cable to your laptop or desktop.

Observe the Change in Network Status: After reconnecting, your system will attempt to re-establish a connection to the network.

2: Now find the hostname of PC/Laptop.

First, we open Command prompt (cmd).

Then enter command = “hostname”.

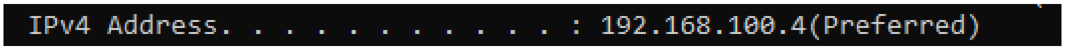
Observation:

3: Now find the interfaces (virtual and physical) PC/Laptop.

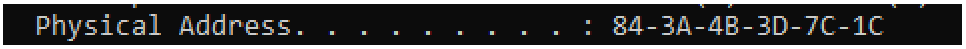
First, we open Command prompt (cmd).

Then enter command = “ipconfig /all”.

Observation:



```
IPv4 Address. . . . . . . . . . . : 192.168.100.4(Preferred)
```



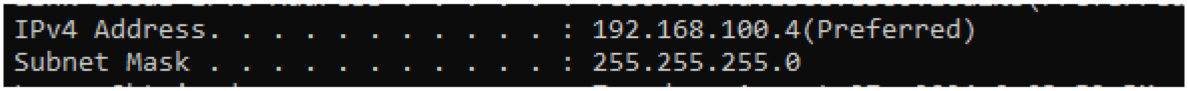
```
Physical Address. . . . . . . . . : 84-3A-4B-3D-7C-1C
```

4: Now find the IP address of your PC/Laptop.

First, we open Command prompt (cmd).

Then enter command = “ipconfig /all”.

Observation:



```
IPv4 Address. . . . . . . . . . . : 192.168.100.4(Preferred)
Subnet Mask . . . . . . . . . . . : 255.255.255.0
```

5: DNS server: how many?

which is primary?

which is public and which is private?

Number of DNS Servers:

1. Primary server: This server is main server that the quires first.

2. Secodary Server: If the primary server is unavailable the backup server which we say the secondary server device will query.

3. Additional DNS Server: some cases their setup of DNS server more than two server because used for redundancy.

Primary Server: The Primary DNS Server is the first in the list of DNS servers that the system will try to use.

Public Server: These are provided by external organizations and are available to anyone on the internet. Examples include:

- **Google DNS: 8.8.8.8 and 8.8.4.4**

Private DNS Servers: These are typically provided by your Internet Service Provider (ISP) or a local network administrator. These are usually used within a specific network and may not be accessible outside of it.

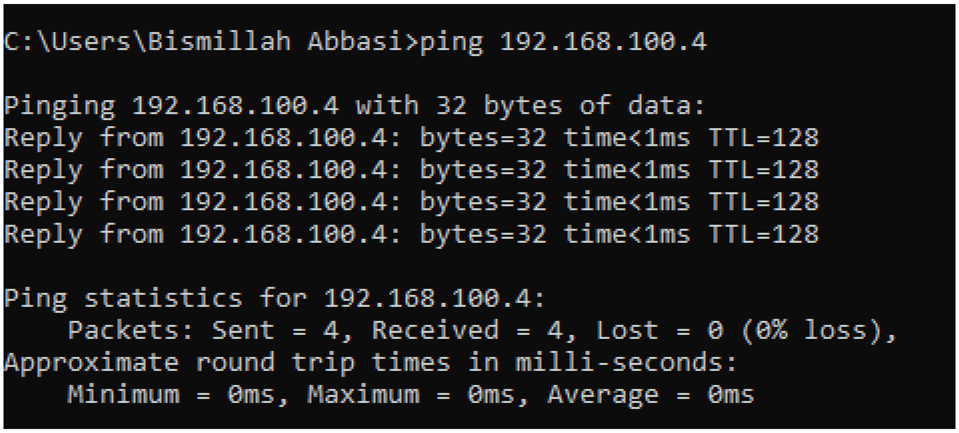
6: Now find the ping our neighbors PC/Laptop.

First, we open Command prompt (cmd).

Then enter command = “ping IP/hostname”.

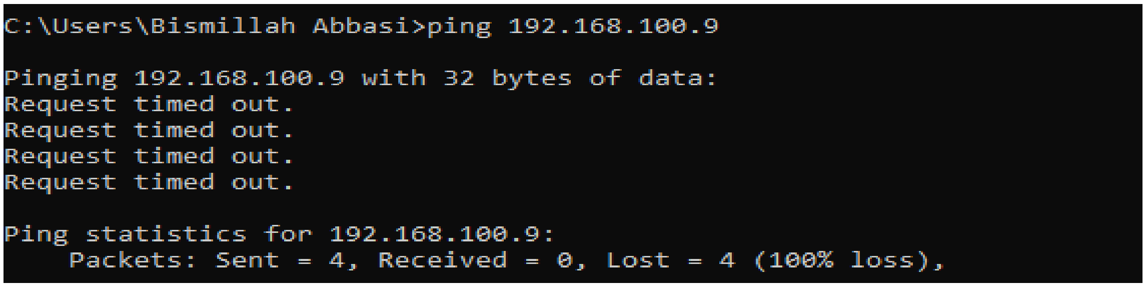
Observation:

Successful Scenario:



```
C:\Users\Bismillah Abbasi>ping 192.168.100.4

Pinging 192.168.100.4 with 32 bytes of data:
Reply from 192.168.100.4: bytes=32 time<1ms TTL=128
Reply from 192.168.100.4: bytes=32 time<1ms TTL=128
Reply from 192.168.100.4: bytes=32 time<1ms TTL=128
Reply from 192.168.100.4: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.100.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Unsuccessful Scenario:



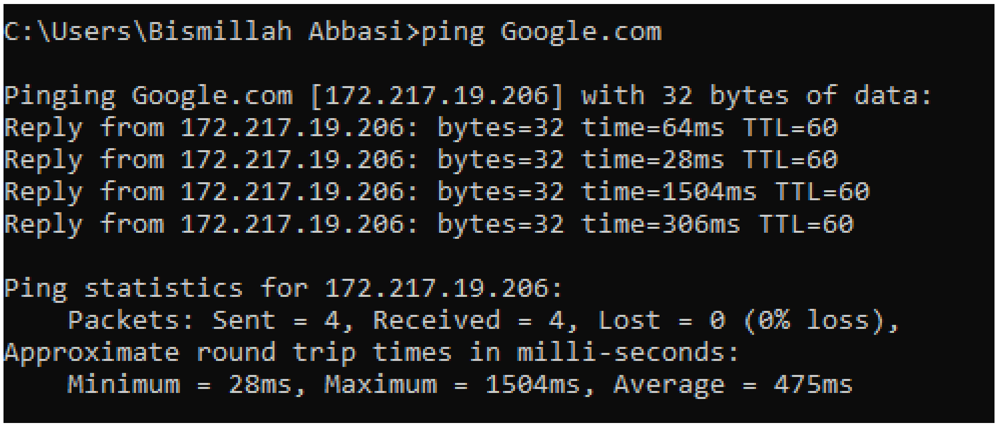
```
C:\Users\Bismillah Abbasi>ping 192.168.100.9

Pinging 192.168.100.9 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.100.9:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

7: Now find ping a hostname PC/Laptop.

First, we open Command prompt (cmd).

Then enter command = “ping IP/hostname”.

Observation:



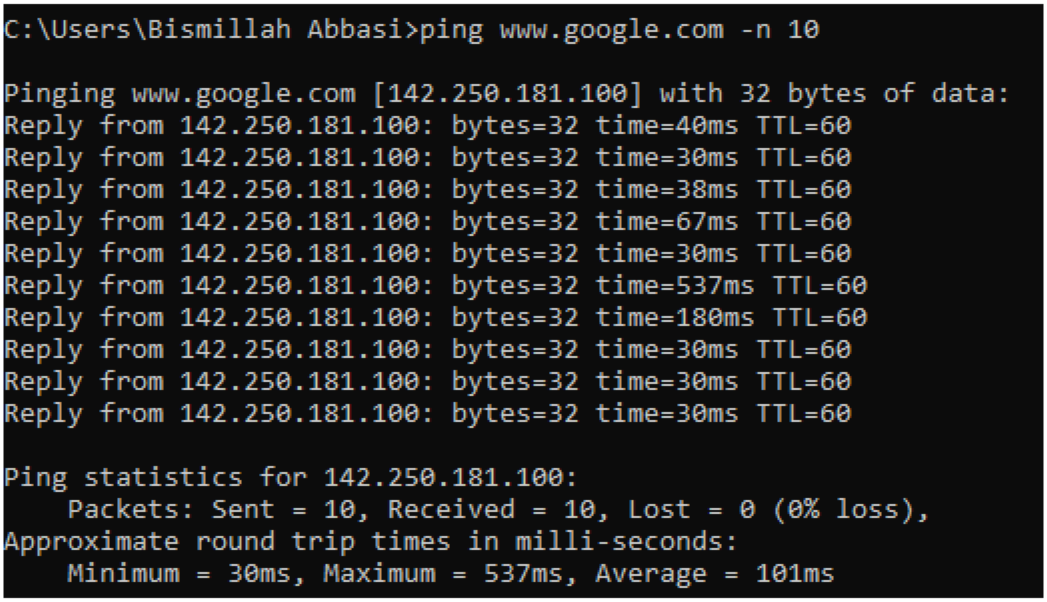
```
C:\Users\Bismillah Abbasi>ping Google.com

Pinging Google.com [172.217.19.206] with 32 bytes of data:
Reply from 172.217.19.206: bytes=32 time=64ms TTL=60
Reply from 172.217.19.206: bytes=32 time=28ms TTL=60
Reply from 172.217.19.206: bytes=32 time=1504ms TTL=60
Reply from 172.217.19.206: bytes=32 time=306ms TTL=60

Ping statistics for 172.217.19.206:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 28ms, Maximum = 1504ms, Average = 475ms
```

8: Now find ping with -n PC/Laptop.

First we open Command prompt (cmd).

Then enter command = “ping IP/hostname -n (any number)”.

Observation:



```
C:\Users\Bismillah Abbasi>ping www.google.com -n 10

Pinging www.google.com [142.250.181.100] with 32 bytes of data:
Reply from 142.250.181.100: bytes=32 time=40ms TTL=60
Reply from 142.250.181.100: bytes=32 time=30ms TTL=60
Reply from 142.250.181.100: bytes=32 time=38ms TTL=60
Reply from 142.250.181.100: bytes=32 time=67ms TTL=60
Reply from 142.250.181.100: bytes=32 time=30ms TTL=60
Reply from 142.250.181.100: bytes=32 time=537ms TTL=60
Reply from 142.250.181.100: bytes=32 time=180ms TTL=60
Reply from 142.250.181.100: bytes=32 time=30ms TTL=60
Reply from 142.250.181.100: bytes=32 time=30ms TTL=60
Reply from 142.250.181.100: bytes=32 time=30ms TTL=60

Ping statistics for 142.250.181.100:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 30ms, Maximum = 537ms, Average = 101ms
```

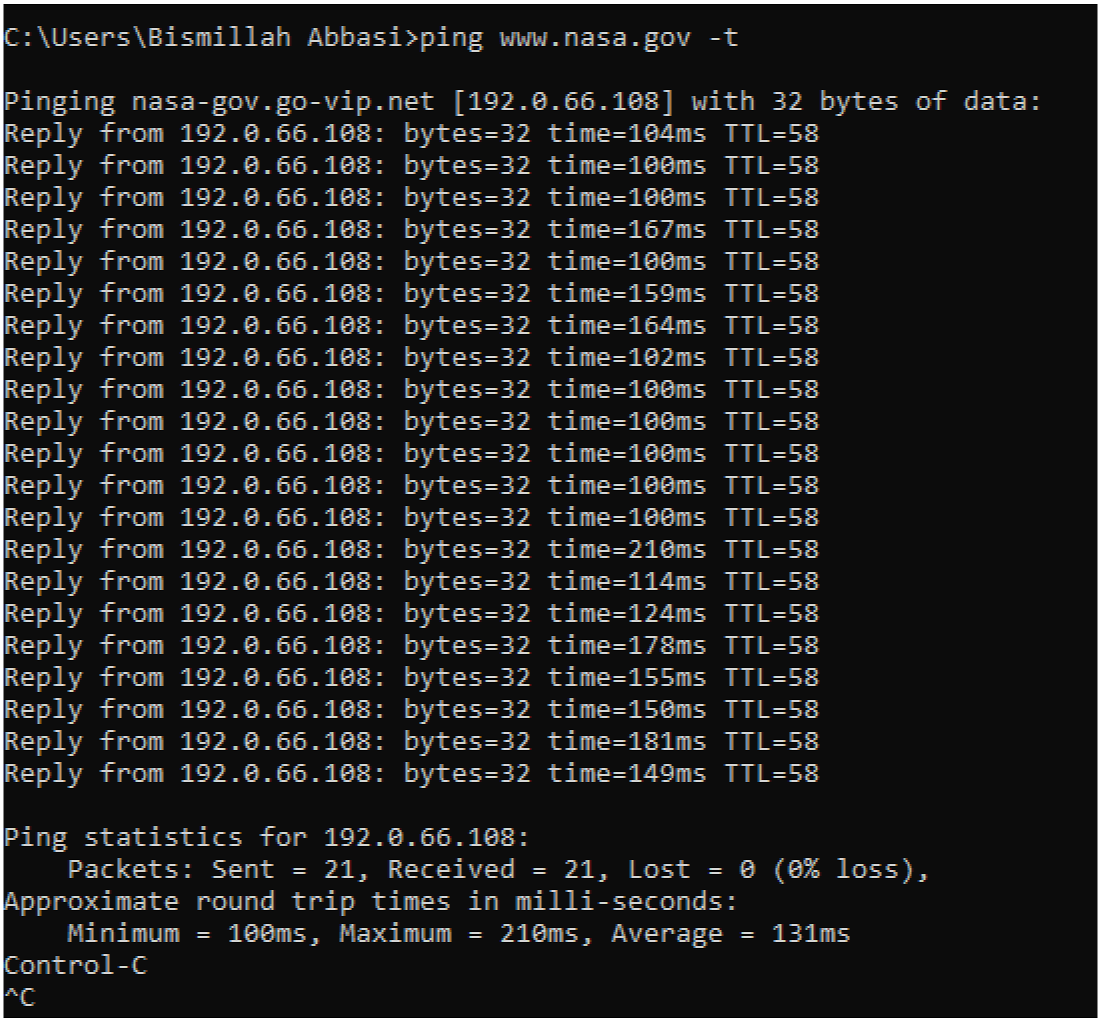
9: Now find ping with -t PC/Laptop.

First, we open Command prompt (cmd).

Then enter command = “ping IP/hostname -t”.

To stop this command by Ctrl+C.

Observation:



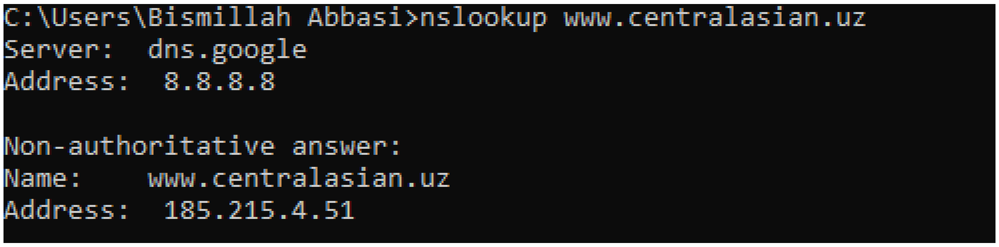
```
C:\Users\Bismillah Abbasi>ping www.nasa.gov -t

Pinging nasa-gov.go-vip.net [192.0.66.108] with 32 bytes of data:
Reply from 192.0.66.108: bytes=32 time=104ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=167ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=159ms TTL=58
Reply from 192.0.66.108: bytes=32 time=164ms TTL=58
Reply from 192.0.66.108: bytes=32 time=102ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=100ms TTL=58
Reply from 192.0.66.108: bytes=32 time=210ms TTL=58
Reply from 192.0.66.108: bytes=32 time=114ms TTL=58
Reply from 192.0.66.108: bytes=32 time=124ms TTL=58
Reply from 192.0.66.108: bytes=32 time=178ms TTL=58
Reply from 192.0.66.108: bytes=32 time=155ms TTL=58
Reply from 192.0.66.108: bytes=32 time=150ms TTL=58
Reply from 192.0.66.108: bytes=32 time=181ms TTL=58
Reply from 192.0.66.108: bytes=32 time=149ms TTL=58

Ping statistics for 192.0.66.108:
    Packets: Sent = 21, Received = 21, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 100ms, Maximum = 210ms, Average = 131ms
Control-C
^C
```

10: Now find use name server (dns) to know IP address of www.centralasian.uz.

First, we open Command prompt (cmd).

Then enter command = “nslookup www.centralasian.uz”.

Observation:



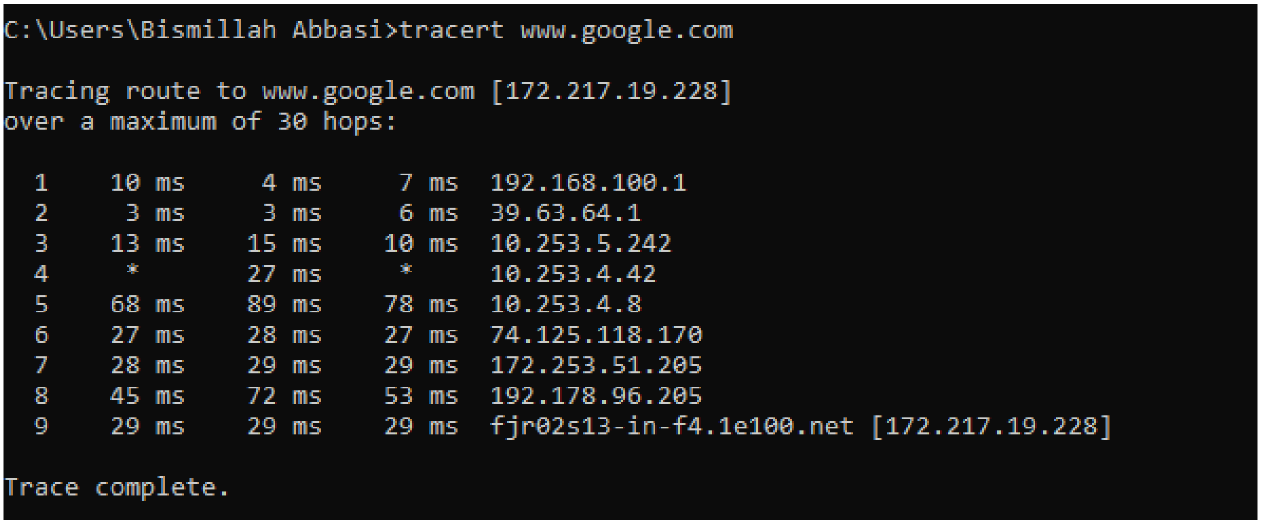
```
C:\Users\Bismillah Abbasi>nslookup www.centralasian.uz
Server:  dns.google
Address:  8.8.8.8

Non-authoritative answer:
Name:    www.centralasian.uz
Address:  185.215.4.51
```

11: Now trace www.google.com.

First, we open Command prompt (cmd).

Then enter command = "tracert www.google.com".

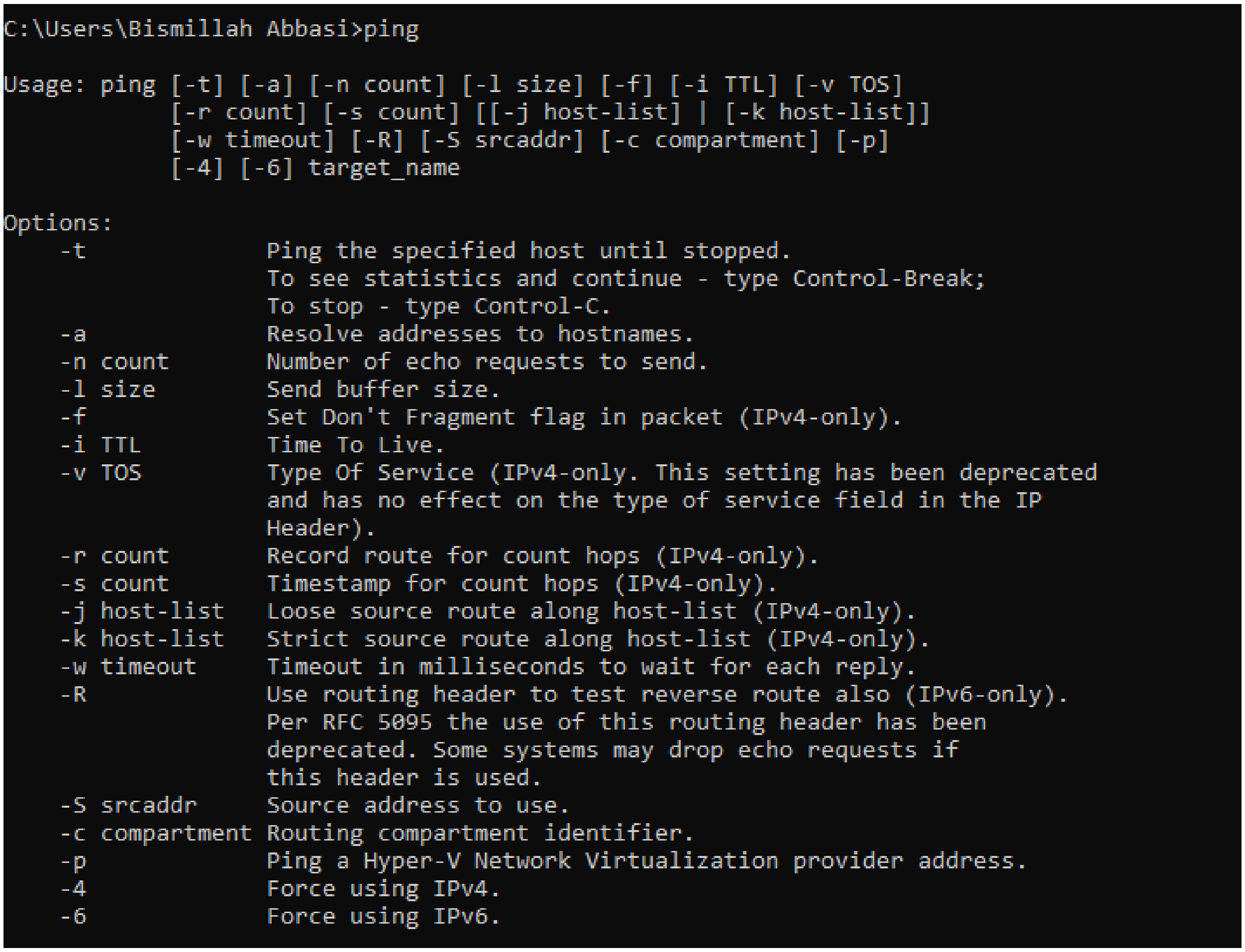
Observation:



```
C:\Users\Bismillah Abbasi>tracert www.google.com

Tracing route to www.google.com [172.217.19.228]
over a maximum of 30 hops:

  1    10 ms     4 ms     7 ms  192.168.100.1
  2     3 ms     3 ms     6 ms  39.63.64.1
  3    13 ms    15 ms    10 ms  10.253.5.242
  4     *       27 ms     *     10.253.4.42
  5    68 ms    89 ms    78 ms  10.253.4.8
  6    27 ms    28 ms    27 ms  74.125.118.170
  7    28 ms    29 ms    29 ms  172.253.51.205
  8    45 ms    72 ms    53 ms  192.178.96.205
  9    29 ms    29 ms    29 ms  fjr02s13-in-f4.1e100.net [172.217.19.228]

Trace complete.
```

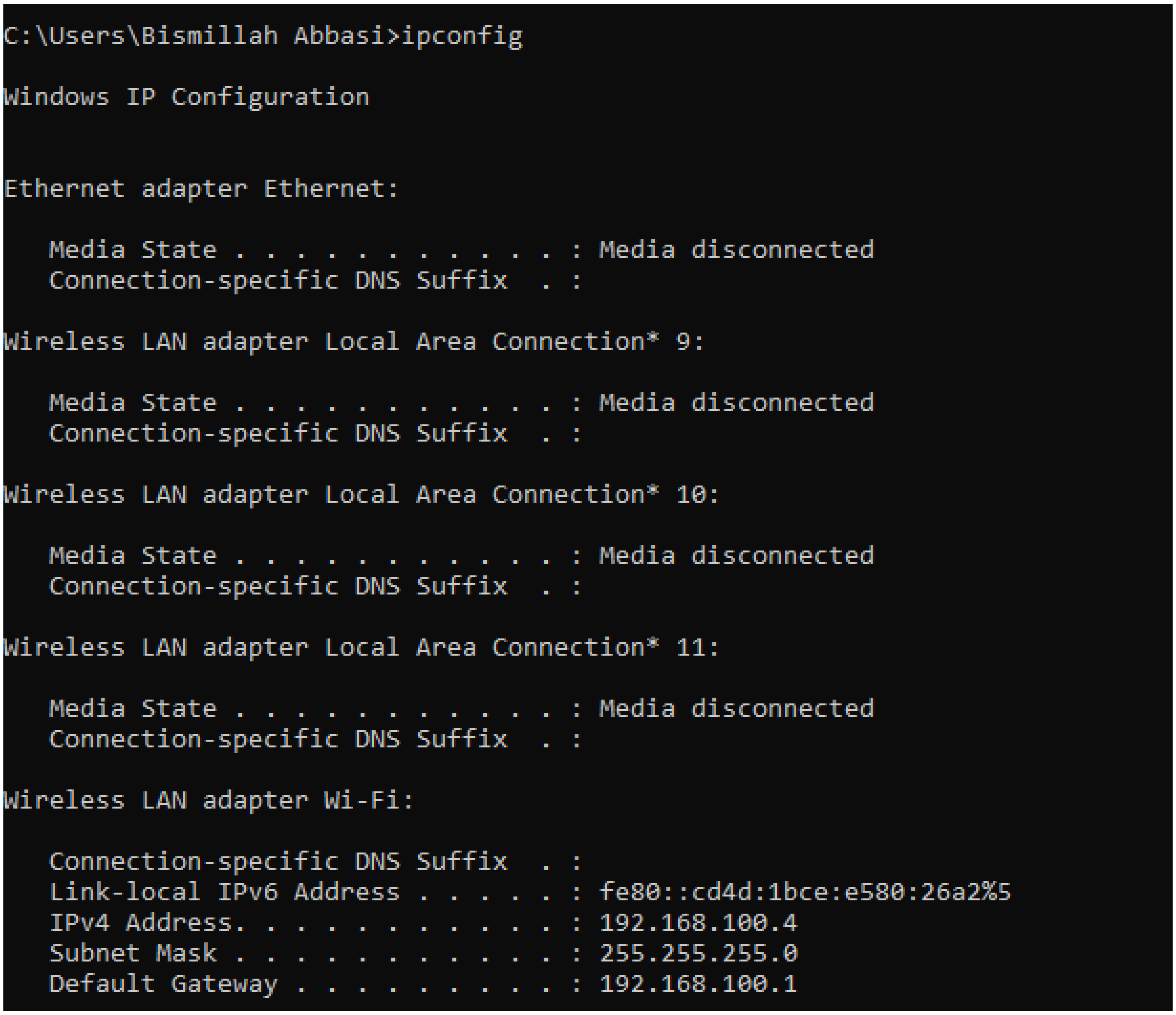
Practice commands which help full for above these activities

1. **Ping:** It is a command-line tool used to test network connectivity between two devices. It sends a series of data packets to a specified IP address or hostname and measures the time it takes for the packets to be received. This helps determine if a network connection is working properly.



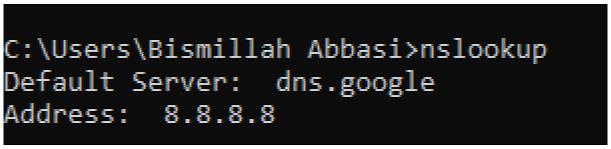
```
C:\Users\Bismillah Abbasi>ping

Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
            [-r count] [-s count] [[-j host-list] | [-k host-list]]
            [-w timeout] [-R] [-S srcaddr] [-c compartment] [-p]
            [-4] [-6] target_name

Options:
    -t             Ping the specified host until stopped.
                   To see statistics and continue - type Control-Break;
                   To stop - type Control-C.
    -a             Resolve addresses to hostnames.
    -n count       Number of echo requests to send.
    -l size        Send buffer size.
    -f             Set Don't Fragment flag in packet (IPv4-only).
    -i TTL         Time To Live.
    -v TOS         Type Of Service (IPv4-only. This setting has been deprecated
                   and has no effect on the type of service field in the IP
                   Header).
    -r count       Record route for count hops (IPv4-only).
    -s count       Timestamp for count hops (IPv4-only).
    -j host-list   Loose source route along host-list (IPv4-only).
    -k host-list   Strict source route along host-list (IPv4-only).
    -w timeout     Timeout in milliseconds to wait for each reply.
    -R             Use routing header to test reverse route also (IPv6-only).
                   Per RFC 5095 the use of this routing header has been
                   deprecated. Some systems may drop echo requests if
                   this header is used.
    -S srcaddr     Source address to use.
    -c compartment Routing compartment identifier.
    -p             Ping a Hyper-V Network Virtualization provider address.
    -4             Force using IPv4.
    -6             Force using IPv6.
```

2. **Ipconfig**: is a command-line tool used to display information about the network configuration of a computer. It shows details like IP addresses, subnet masks, default gateways, and DNS server settings.



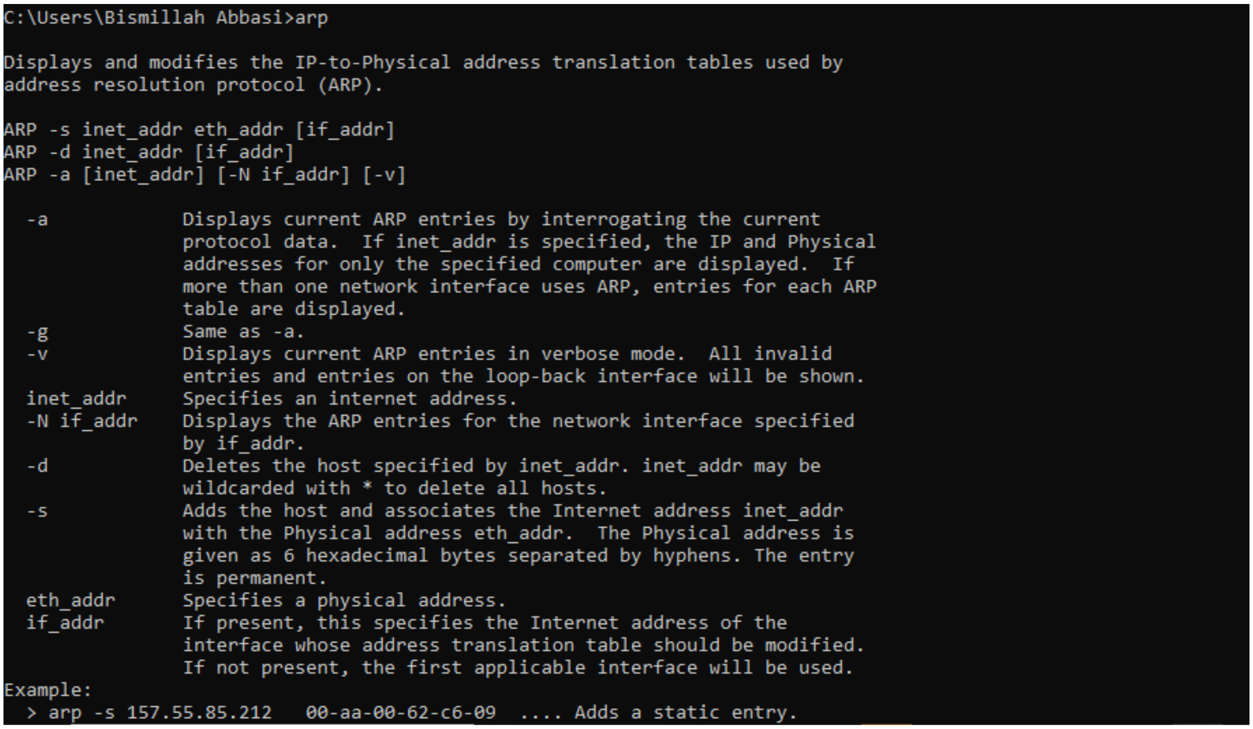
```
C:\Users\Bismillah Abbasi>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 9:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 10:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 11:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::cd4d:1bce:e580:26a2%5
   IPv4 Address. . . . . . . . . . . : 192.168.100.4
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.100.1
```

3. **ns lookup www.centralasian.uz:** Looks up the IP address for www.centralasian.uz using the `nslookup` command.



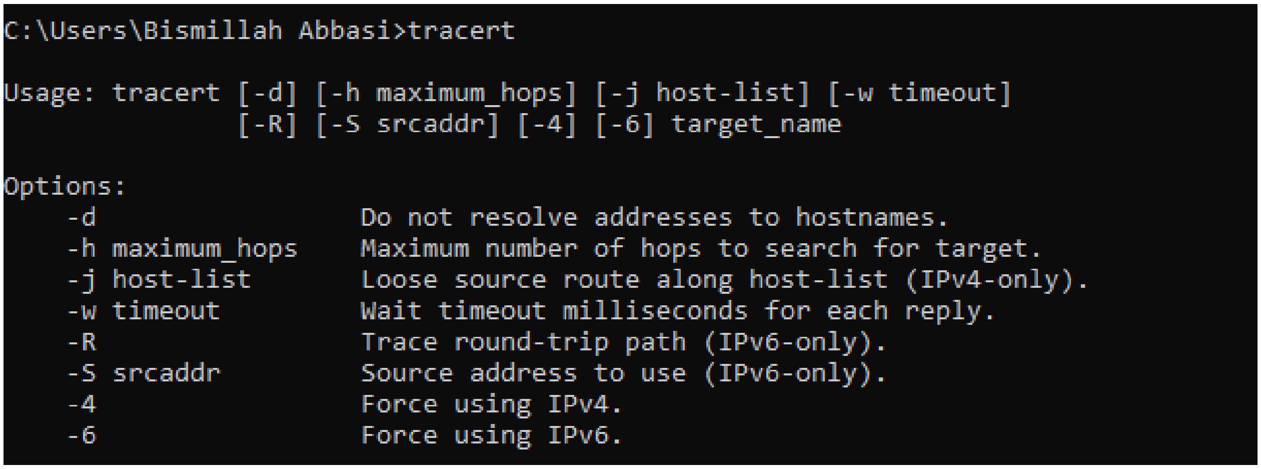
```
C:\Users\Bismillah Abbasi>nslookup
Default Server:  dns.google
Address:  8.8.8.8
```

4. **arp -a :** Displays the ARP cache entries on the system.



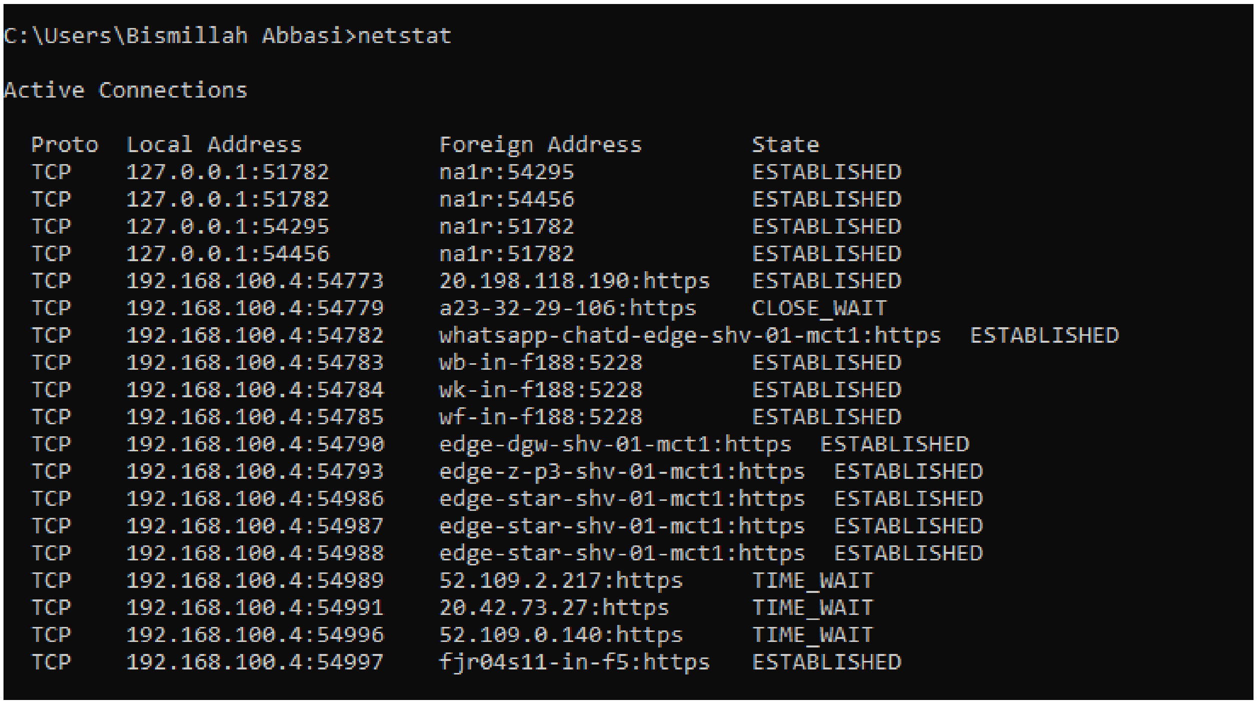
```
C:\Users\Bismillah Abbasi>arp

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.  All invalid
                entries and entries on the loop-back interface will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be modified.
                If not present, the first applicable interface will be used.
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
```

5. **tracert www.google.com:** Traces the route data packets take to reach Google (www.google.com) by showing network "hops".



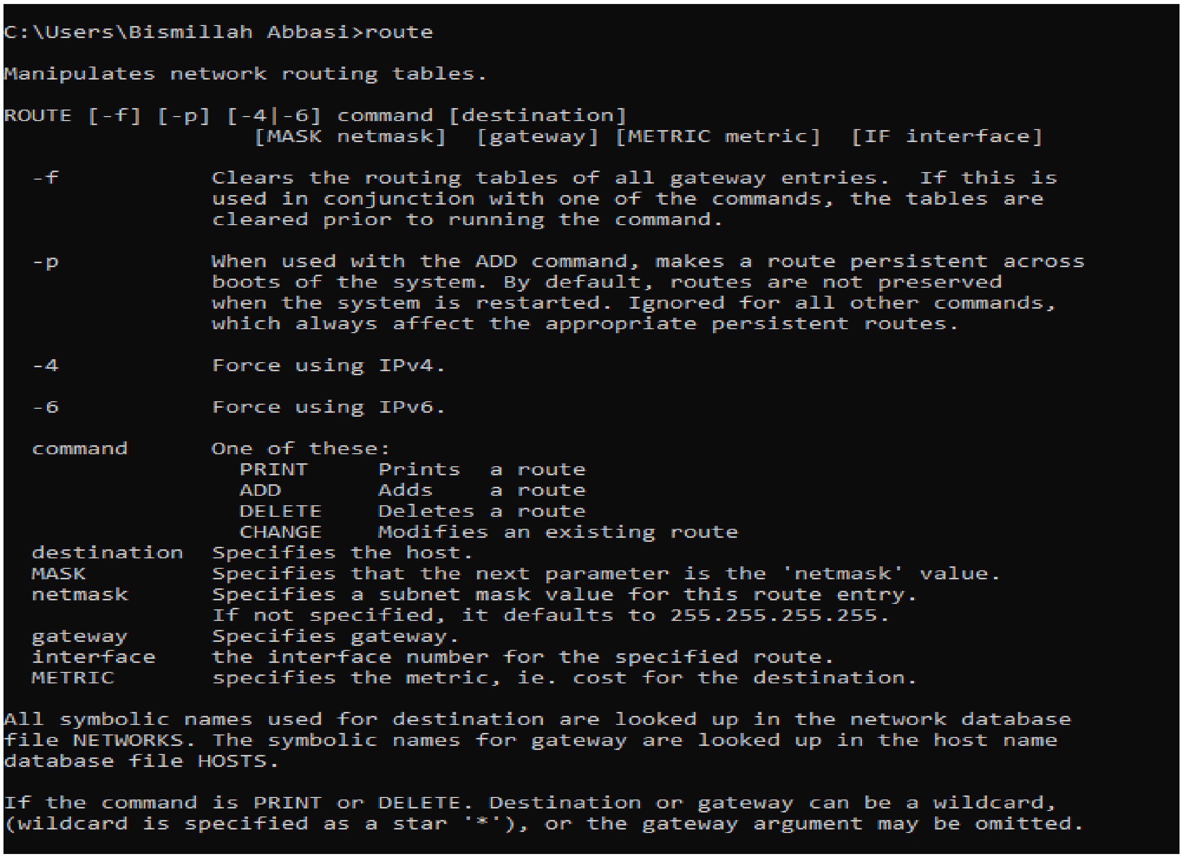
```
C:\Users\Bismillah Abbasi>tracert

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.
```

6. **netsta:** Netstat shows network connections, statistics, and routing tables.

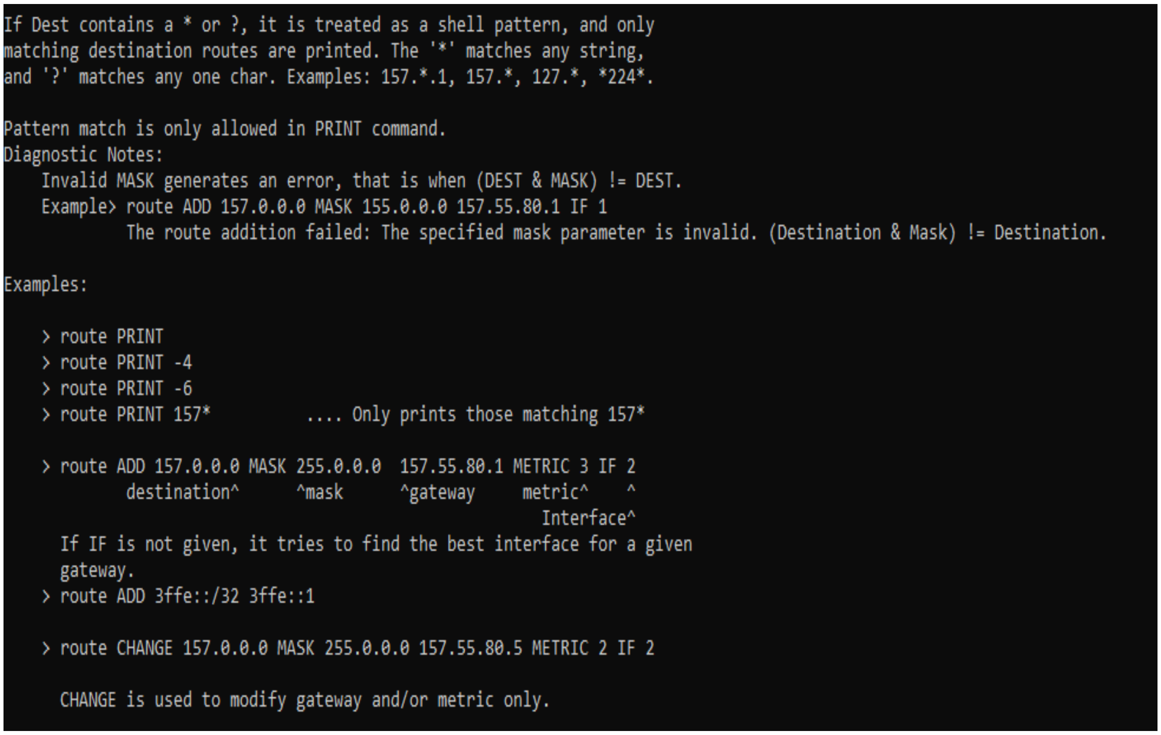


```
C:\Users\Bismillah Abbasi>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:51782        na1r:54295             ESTABLISHED
  TCP    127.0.0.1:51782        na1r:54456             ESTABLISHED
  TCP    127.0.0.1:54295        na1r:51782             ESTABLISHED
  TCP    127.0.0.1:54456        na1r:51782             ESTABLISHED
  TCP    192.168.100.4:54773    20.198.118.190:https   ESTABLISHED
  TCP    192.168.100.4:54779    a23-32-29-106:https    CLOSE_WAIT
  TCP    192.168.100.4:54782    whatsapp-chatd-edge-shv-01-mct1:https  ESTABLISHED
  TCP    192.168.100.4:54783    wb-in-f188:5228        ESTABLISHED
  TCP    192.168.100.4:54784    wk-in-f188:5228        ESTABLISHED
  TCP    192.168.100.4:54785    wf-in-f188:5228        ESTABLISHED
  TCP    192.168.100.4:54790    edge-dgw-shv-01-mct1:https  ESTABLISHED
  TCP    192.168.100.4:54793    edge-z-p3-shv-01-mct1:https  ESTABLISHED
  TCP    192.168.100.4:54986    edge-star-shv-01-mct1:https  ESTABLISHED
  TCP    192.168.100.4:54987    edge-star-shv-01-mct1:https  ESTABLISHED
  TCP    192.168.100.4:54988    edge-star-shv-01-mct1:https  ESTABLISHED
  TCP    192.168.100.4:54989    52.109.2.217:https     TIME_WAIT
  TCP    192.168.100.4:54991    20.42.73.27:https      TIME_WAIT
  TCP    192.168.100.4:54996    52.109.0.140:https     TIME_WAIT
  TCP    192.168.100.4:54997    fjr04s11-in-f5:https   ESTABLISHED
```

7. **Routing is the process of selecting paths in a network along which to send data packets.**



```
C:\Users\Bismillah Abbasi>route

Manipulates network routing tables.

ROUTE [-f] [-p] [-4|-6] command [destination]
                  [MASK netmask]  [gateway] [METRIC metric]  [IF interface]

  -f           Clears the routing tables of all gateway entries.  If this is
               used in conjunction with one of the commands, the tables are
               cleared prior to running the command.

  -p           When used with the ADD command, makes a route persistent across
               boots of the system. By default, routes are not preserved
               when the system is restarted. Ignored for all other commands,
               which always affect the appropriate persistent routes.

  -4           Force using IPv4.

  -6           Force using IPv6.

  command      One of these:
                 PRINT     Prints  a route
                 ADD       Adds    a route
                 DELETE    Deletes a route
                 CHANGE    Modifies an existing route
  destination  Specifies the host.
  MASK         Specifies that the next parameter is the 'netmask' value.
  netmask      Specifies a subnet mask value for this route entry.
               If not specified, it defaults to 255.255.255.255.
  gateway      Specifies gateway.
  interface    the interface number for the specified route.
  METRIC       specifies the metric, ie. cost for the destination.

All symbolic names used for destination are looked up in the network database
file NETWORKS. The symbolic names for gateway are looked up in the host name
database file HOSTS.

If the command is PRINT or DELETE. Destination or gateway can be a wildcard,
(wildcard is specified as a star '*'), or the gateway argument may be omitted.
```



```
If Dest contains a * or ?, it is treated as a shell pattern, and only
matching destination routes are printed. The '*' matches any string,
and '?' matches any one char. Examples: 157.*.1, 157.*, 127.*, *224*.

Pattern match is only allowed in PRINT command.
Diagnostic Notes:
    Invalid MASK generates an error, that is when (DEST & MASK) != DEST.
    Example> route ADD 157.0.0.0 MASK 155.0.0.0 157.55.80.1 IF 1
             The route addition failed: The specified mask parameter is invalid. (Destination & Mask) != Destination.

Examples:

    > route PRINT
    > route PRINT -4
    > route PRINT -6
    > route PRINT 157*          .... Only prints those matching 157*

    > route ADD 157.0.0.0 MASK 255.0.0.0  157.55.80.1 METRIC 3 IF 2
             destination^      ^mask      ^gateway     metric^    ^
                                                         Interface^
      If IF is not given, it tries to find the best interface for a given
      gateway.
    > route ADD 3ffe::/32 3ffe::1

    > route CHANGE 157.0.0.0 MASK 255.0.0.0 157.55.80.5 METRIC 2 IF 2

      CHANGE is used to modify gateway and/or metric only.
```