NAME : SANGAM BIRLA

ADMI. NO. : U22CS040

ASSIGNMENT : 01 (Computer Network)

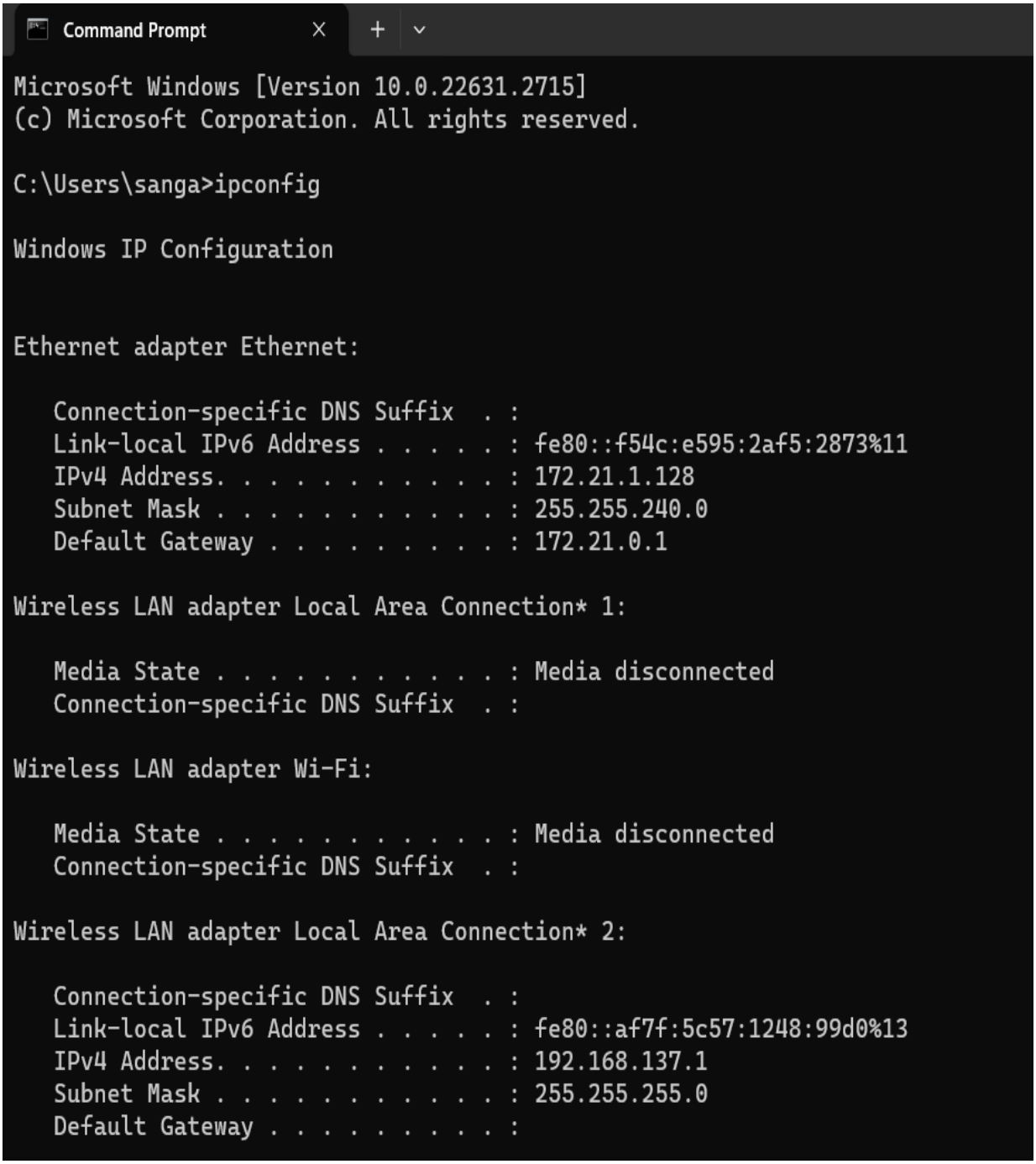
Study Networking/Ip Commands and Prepare a Report.

Introduction

Networking and IP commands are essential tools for managing and troubleshooting network configurations. This report provides an overview of commonly used commands across various operating systems such as Windows, Linux, and macOS. These commands facilitate tasks like checking network configurations, diagnosing connectivity issues, and managing routing tables.

1.ipconfig (Windows) / ifconfig (Linux/macOS)

The **ipconfig** command in Windows and **ifconfig** in Linux/macOS are used to display the configuration of network interfaces on a system. This includes information such as IP addresses, subnet masks, and gateway addresses. Running these commands helps administrators quickly assess the current network settings on a machine.



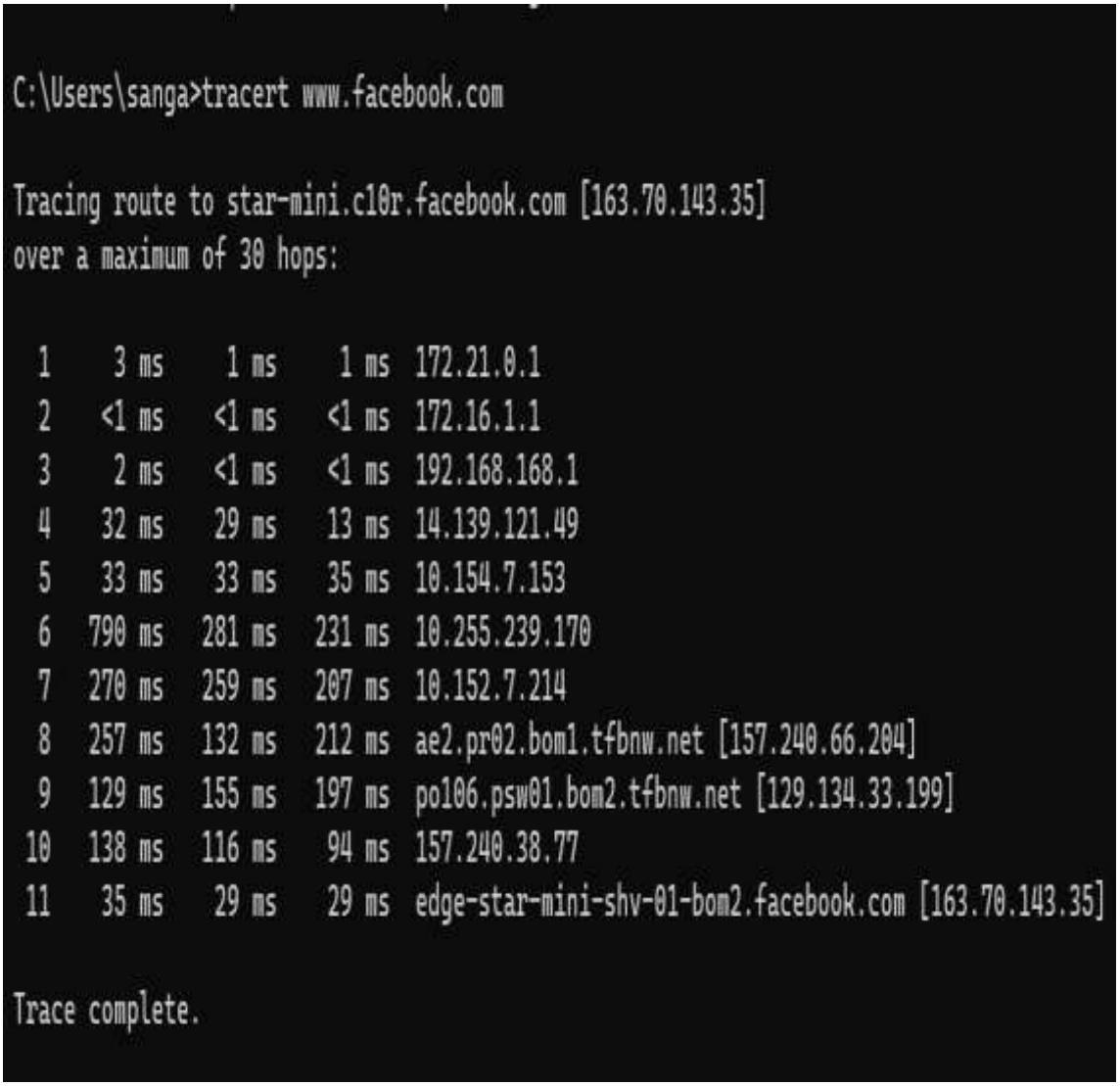
2.ping

The **ping** command is a universal tool for testing network connectivity. By providing the hostname or IP address of a destination, it measures the roundtrip time for packets to travel to and from that destination. The results help determine if a host is reachable and the quality of the connection.

```
C:\Users\sanga>ping www.google.com

Pinging www.google.com [216.58.203.36] with 32 bytes of data:
Reply from 216.58.203.36: bytes=32 time=70ms TTL=116
Reply from 216.58.203.36: bytes=32 time=8ms TTL=116
Reply from 216.58.203.36: bytes=32 time=47ms TTL=116
Reply from 216.58.203.36: bytes=32 time=161ms TTL=116

Ping statistics for 216.58.203.36:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 8ms, Maximum = 161ms, Average = 71ms
```

3.tracert (Windows) / traceroute (Linux/macOS)

tracert in Windows and **traceroute** in Linux/macOS allow administrators to trace the route packets take to reach a destination. This is particularly useful for diagnosing network latency issues and identifying the IP addresses of routers in the path.



4.netstat

The **netstat** command provides a range of network-related information. It can display active network connections, listening ports, routing tables, and interface statistics. This command is valuable for monitoring network activity and identifying potential issues.

```
C:\Users\sanga>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:49682        DESKTOP-9MUHOID:49683  ESTABLISHED
  TCP    127.0.0.1:49683        DESKTOP-9MUHOID:49682  ESTABLISHED
  TCP    127.0.0.1:49684        DESKTOP-9MUHOID:49685  ESTABLISHED
  TCP    127.0.0.1:49685        DESKTOP-9MUHOID:49684  ESTABLISHED
  TCP    127.0.0.1:49719        DESKTOP-9MUHOID:49720  ESTABLISHED
  TCP    127.0.0.1:49720        DESKTOP-9MUHOID:49719  ESTABLISHED
  TCP    127.0.0.1:49721        DESKTOP-9MUHOID:49722  ESTABLISHED
  TCP    127.0.0.1:49722        DESKTOP-9MUHOID:49721  ESTABLISHED
  TCP    127.0.0.1:49725        DESKTOP-9MUHOID:49726  ESTABLISHED
  TCP    127.0.0.1:49726        DESKTOP-9MUHOID:49725  ESTABLISHED
  TCP    172.21.1.128:49410     20.198.119.143:https   ESTABLISHED
  TCP    172.21.1.128:49977     server-18-239-111-90:https  ESTABLISHED
  TCP    172.21.1.128:51614     172.16.1.1:8090        ESTABLISHED
  TCP    172.21.1.128:51669     sg-in-f188:https       ESTABLISHED
  TCP    172.21.1.128:55831     dns:https              ESTABLISHED
  TCP    172.21.1.128:55835     ec2-3-6-211-252:https  ESTABLISHED
  TCP    172.21.1.128:55853     server-18-172-64-86:https  ESTABLISHED
  TCP    172.21.1.128:55857     199.232.254.137:https  ESTABLISHED
  TCP    172.21.1.128:55881     104.18.28.147:https    ESTABLISHED
  TCP    172.21.1.128:55903     104.17.1.41:https      ESTABLISHED
  TCP    172.21.1.128:55905     104.18.29.147:https    ESTABLISHED
  TCP    172.21.1.128:55908     sb-in-f84:https        TIME_WAIT
  TCP    172.21.1.128:55926     bom07s31-in-f1:https   TIME_WAIT
  TCP    172.21.1.128:55927     bom12s11-in-f3:https   TIME_WAIT
  TCP    172.21.1.128:55932     any-in-2215:https      TIME_WAIT
  TCP    172.21.1.128:55933     bom07s18-in-f10:https  TIME_WAIT
  TCP    172.21.1.128:55934     bom07s20-in-f10:https  TIME_WAIT
  TCP    172.21.1.128:55935     bom07s20-in-f10:https  TIME_WAIT
  TCP    172.21.1.128:55938     bom07s45-in-f14:https  TIME_WAIT
  TCP    172.21.1.128:55940     a23-63-111-186:https   ESTABLISHED
  TCP    172.21.1.128:55944     sc-in-f84:https        TIME_WAIT
  TCP    172.21.1.128:55946     bom07s32-in-f14:https  TIME_WAIT
  TCP    172.21.1.128:55948     bom12s19-in-f3:https   TIME_WAIT
  TCP    172.21.1.128:55952     bom07s25-in-f10:https  TIME_WAIT
  TCP    172.21.1.128:55954     any-in-2015:https      TIME_WAIT
  TCP    172.21.1.128:55958     li781-4:https          ESTABLISHED
  TCP    172.21.1.128:55959     li695-222:https        ESTABLISHED
  TCP    172.21.1.128:55971     a23-212-5-81:https     CLOSE_WAIT
  TCP    172.21.1.128:55975     kul01s09-in-f67:https  ESTABLISHED
  TCP    172.21.1.128:55976     bom07s16-in-f3:https   TIME_WAIT
  TCP    172.21.1.128:56019     sc-in-f84:https        ESTABLISHED
  TCP    172.21.1.128:56020     bom12s12-in-f14:https  ESTABLISHED
  TCP    172.21.1.128:56022     a104-120-74-51:http    TIME_WAIT
  TCP    172.21.1.128:56023     a104-120-81-153:http   TIME_WAIT
  TCP    172.21.1.128:56029     52.109.124.155:https   FIN_WAIT_1
  TCP    172.21.1.128:56033     a104-71-61-138:http    TIME_WAIT
  TCP    172.21.1.128:56038     52.111.252.15:https    TIME_WAIT
  TCP    172.21.1.128:56041     216.239.34.117:https   TIME_WAIT
  TCP    172.21.1.128:56042     216.239.34.117:https   ESTABLISHED
  TCP    172.21.1.128:56045     52.182.143.208:https   FIN_WAIT_1
  TCP    172.21.1.128:56051     152.195.38.76:http     ESTABLISHED
  TCP    172.21.1.128:56060     40.99.111.18:https     ESTABLISHED
  TCP    172.21.1.128:56061     e2a:https              ESTABLISHED
  TCP    172.21.1.128:56063     a104-120-74-51:http    TIME_WAIT
  TCP    172.21.1.128:56064     a23-58-95-162:http     TIME_WAIT
  TCP    172.21.1.128:56078     104.215.155.1:https    ESTABLISHED
  TCP    172.21.1.128:56079     ec2-35-165-220-198:https  ESTABLISHED
  TCP    172.21.1.128:56082     13.67.10.228:8883      ESTABLISHED
  TCP    172.21.1.128:56083     52.109.112.178:https   ESTABLISHED
  TCP    172.21.1.128:56084     40.126.17.133:https    ESTABLISHED
  TCP    172.21.1.128:64173     46:https               ESTABLISHED
```

5.nslookup

nslookup is a tool for querying DNS servers to obtain information about domain names or IP addresses. It assists administrators in verifying DNS configurations and troubleshooting domain resolution problems.

6.arp

The **arp** command is used to view and manipulate the ARP cache, which maps IP addresses to MAC addresses. Administrators can use this command to inspect and manage the local ARP table, helping in the resolution of layer 2 connectivity issues.

```
C:\Users\sanga>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.
  > arp -a                                    .... Displays the arp table.
```

7.route

The **route** command manages the IP routing table, which determines the path that network traffic takes. By using **route print** (Windows) or **route** (Linux), administrators can view and modify the routing table, influencing how packets are directed through the network.

```
Microsoft Windows [Version 10.0.22631.2715]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sanga>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.

    > route DELETE 157.0.0.0
    > route DELETE 3ffe::/32

C:\Users\sanga>print
No file to print

C:\Users\sanga>
```

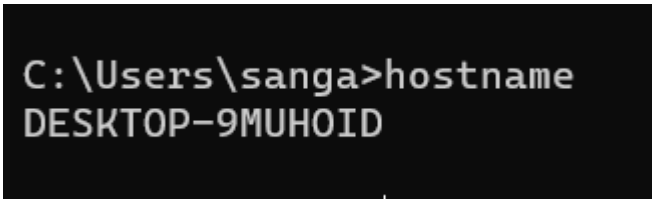
8.ifconfig (Linux/macOS) / ip (Linux)

The **ifconfig** command (Linux/macOS) and **ip** command (Linux) are essential for configuring and displaying information about network interfaces. Administrators can use these commands to assign IP addresses, set up interfaces, and troubleshoot connectivity problems.

```
Windows IP Configuration


Wireless LAN adapter Wi-Fi:

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

Wireless LAN adapter Local Area Connection* 1:

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

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::f54c:e595:2af5:2873%11
   IPv4 Address. . . . . . . . . . . : 172.21.1.128
   Subnet Mask . . . . . . . . . . . : 255.255.240.0
   Default Gateway . . . . . . . . . : 172.21.0.1

Wireless LAN adapter Local Area Connection* 2:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::af7f:5c57:1248:99d0%13
   IPv4 Address. . . . . . . . . . . : 192.168.137.1
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :

C:\Users\sanga>
```

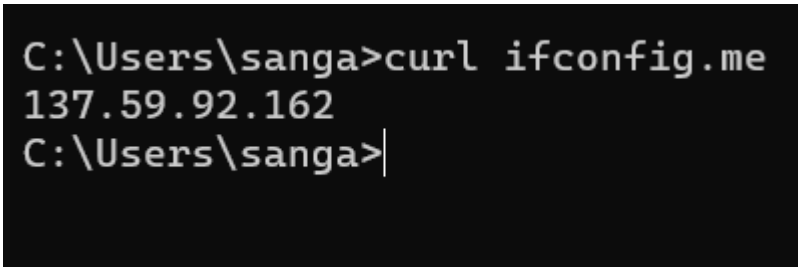
9.Hostname

Windows hostname is given by hostname command.



10.curl ifconfig.me

This command give ip address of my system.



Conclusion

In conclusion, these networking and IP commands form the foundation of network administration and troubleshooting. They empower administrators to diagnose connectivity issues, configure network interfaces, and maintain the overall health of the network. Understanding these commands is crucial for anyone involved in network management and support. Continued exploration and utilization of these commands will enhance the efficiency and effectiveness of network-related tasks.