Experiment No. 1

Aim: To study RJ45 and CAT6 Cabling and connection using crimping tool.

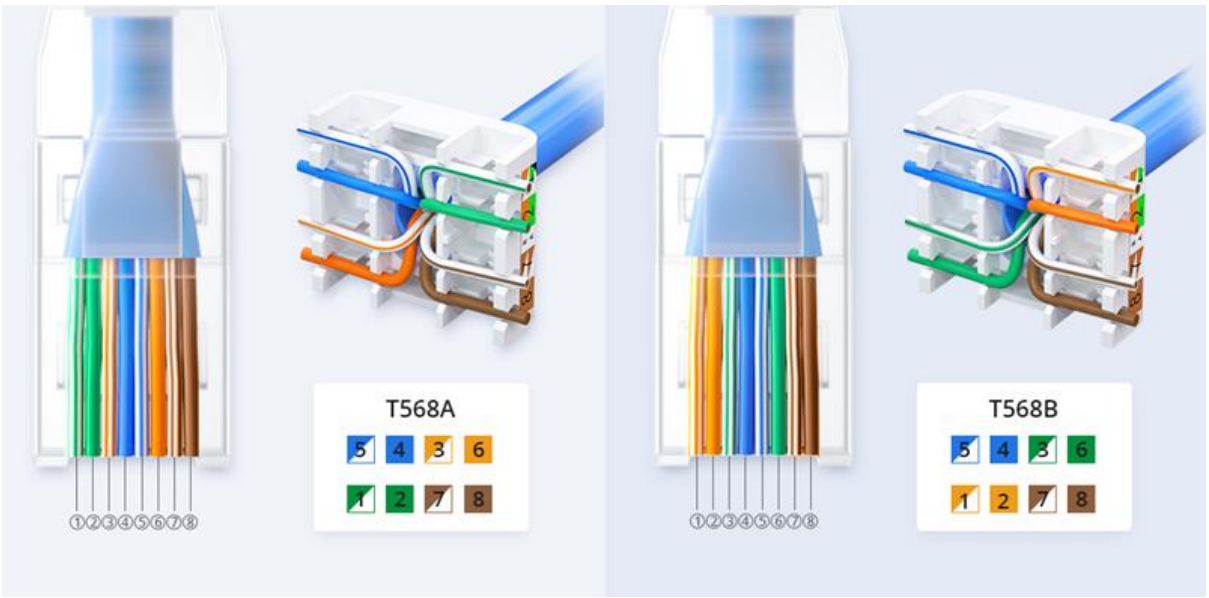
Requirements: RJ45 connector, CAT6 Cable and Crimping tool.

Theory:
RJ45 Interface/Connector: RJ45 interface is considered the most common twisted-pair connector for Ethernet cables and networks.

- "RJ" means "registered jack" — a standardized telecommunication network interface for connecting voice and data equipment to a service provided by a local exchange carrier or long-distance carrier.
- "45" is the number of the interface standard. Physically speaking, the connectors that registered jacks use are mainly the modular connector and 50-pin miniature ribbon connector types. RJ45 connector is an **8-position, 8-contact (8P8C) modular plug, and jack**, applied for Ethernet-based local area networks (LAN). RJ45 cable plug is usually made of a plastic piece with eight pins on the port. Four of the pins are used for sending and receiving data, and the other four are used for other technologies or power networking devices.
- RJ45 connectors can support 10Gbps over Ethernet

How to identify RJ45 interface: RJ45 is identified using color code scheme. T568A vs T568B are the two common wiring schemes, which are used to terminate the twisted-pair cable onto the connector interface. The two standards define how the RJ45 pinouts arrange the individual eight wires when linking the RJ45 connector to a cable. These wiring layouts have their own color convention to follow for electrical compatibility. The T-568B wiring scheme is considered to be the more commonly used one.

The differences between T568A vs T568B in color conventions are shown in the figure below.



With regard to the two standards, there are two different connectivity forms. If both ends of the patch cords are wired on the basis of one standard, it is a straight-through connection. If not, it is a crossover connection. Some networking applications require a crossover Ethernet cable, which has a T-568A connector on one end and a T-568B connector on the other. This type of

cable is typically used for direct computer-to-computer connections when there is no router, hub, or switch available.

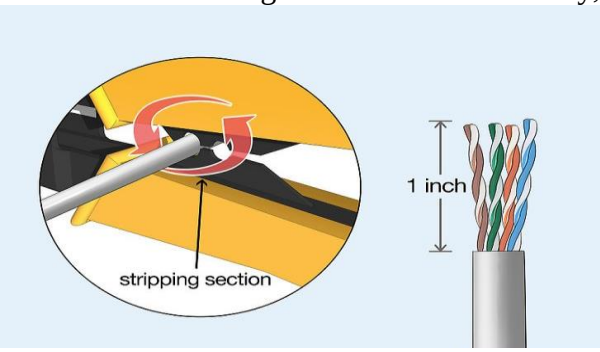
Cat6 Cable: Short for Category 6, Cat6 is an Ethernet cable standard defined by the Electronic Industries Association (EIA) and Telecommunications Industry Association (TIA). What is Cat6 cable used for? As the sixth generation of twisted pair Ethernet cabling, Cat6 cable consists of four twisted pairs and is either terminated by an RJ45 or terminated on a patch or a keystone jack. Theoretically, the maximum speed of the Cat6 network cable is 10Gbps.
A single run of Ethernet cable is designed to work at a maximum distance of 100 meters (328 ft). A length longer than that will result in issues such as dropped packets, reduced performance, and loss of signal when deploying Cat6 cable. The max length of a Cat6 cable usually consists of 90 meters (295 ft) of solid "horizontal" cabling between the patch panel and the wall jack, plus 5 meters (16 ft) of stranded patch cable between each jack and the attached device. For 10GBASE-T, an unshielded Cat6 cable should not exceed 55 meters.

Crimping tool: It is a tool that is used to connect RJ45 connector to Cat6 cable

Steps to crimp RJ45 with Cat6 using Crimping tool:

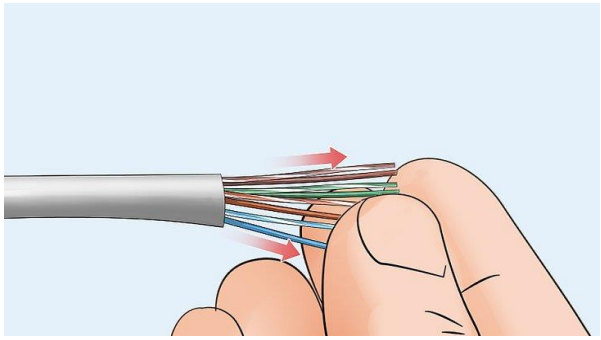
Step 1: **Strip the cable back 1 inch (25 mm) from the end**. Insert the cable into the stripper section of the tool and squeeze it tight. Then, rotate the crimping tool around the cable in a smooth and even motion to create a clean cut. Keep the tool clamped and pull away towards the end of the wire to remove the sheathing.

- The stripping section is a round hole near the handle of the tool.
- The sheathing should come off cleanly, leaving the wires exposed.



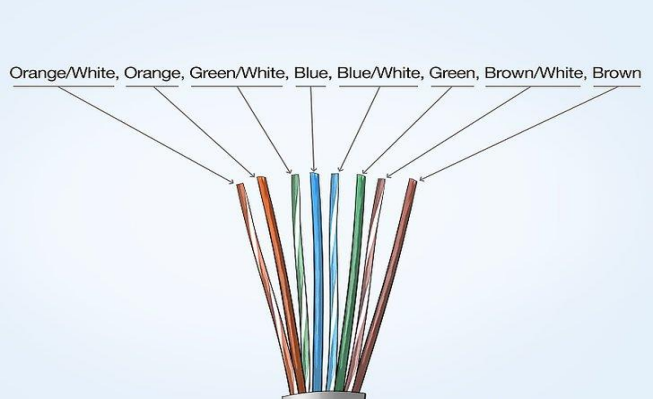
Step 2: **Untwist and straighten the wires inside of the cable.** Inside of the cable you'll see a bunch of smaller wires twisted together. Separate the twisted wires and straighten them out so they're easier to sort into the right order.

- Cut off the small plastic wire separator or core so it's out of the way.
- Don't cut off or remove any of the wires or you won't be able to crimp them into the connector.



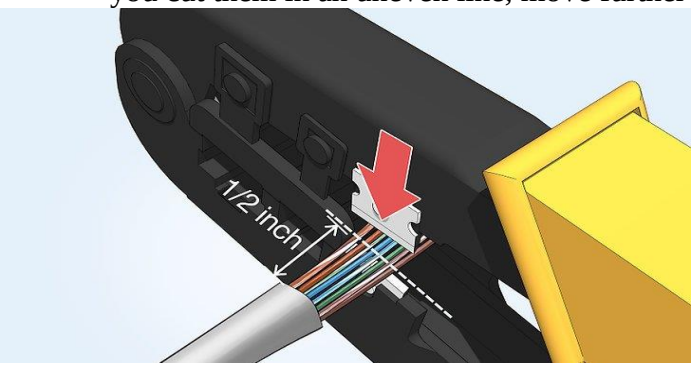
Step 3: **Arrange the wires into the right order.** Use your fingers to put the wires in the correct order so they can be properly crimped. The proper sequence is as follows from left to right: Orange/White, Orange, Green/White, Blue, Blue/White, Green, Brown/White, Brown.

- There are 8 wires in total that need to be arranged in the right sequence.
- Note that the wires labeled Orange/White or Brown/White indicate the small wires that have 2 colors.



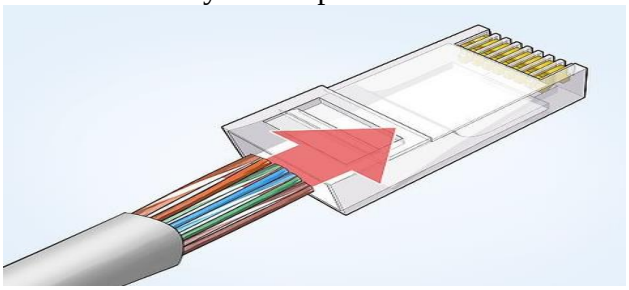
Step 4: **Cut the wires into an even line ½ inch (13 mm) from sheathing.** Hold the wires with your thumb and index finger to keep them in order. Then, use the cutting section of the crimping tool to cut them into an even line.

- The cutting section of the tool will resemble wire cutters.
- The wires must be in an even line to be crimped into the RJ-45 connector properly. If you cut them in an uneven line, move further down the wires and cut them again.



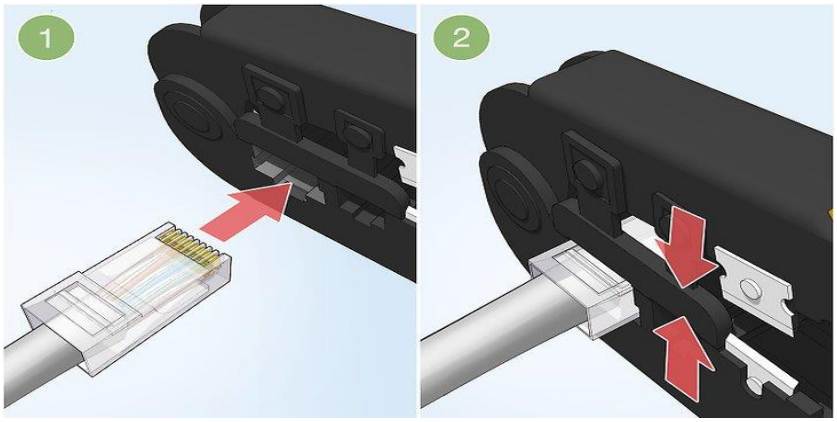
Step 5: **Insert the wires into the RJ-45 connector.** Hold the RJ-45 connector so the clip is on the underside and the small metal pins are facing up. Insert the cable into the connector so that each of the small wires fits into the small grooves in the connector.[5]

- The sheathing of the cable should fit just inside of the connector so it's past the base.
- If any of the small wires bend or don't fit into a groove correctly, take the cable out and straighten the wires with your fingers before trying again.
- The wires must be inserted in the correct order and each wire must fit into a groove before you crimp the connector.



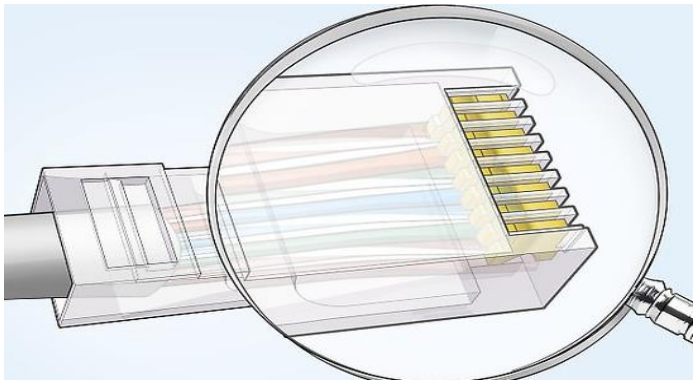
Step 6: **Stick the connector into the crimping part of the tool and squeeze twice.** Insert the connector in the crimping section of the tool until it can't fit any further. Squeeze the handles to crimp the connector and secure the wires. Release the handles, then squeeze the tool again to make sure all of the pins are pushed down.

- The crimping tool pushes small pins in the grooves down onto the wires to hold and connect them to the RJ-45 connector.



Step 7: **Remove the cable from the tool and check that all of the pins are down.** Take the connector out of the tool and look at the pins to see that they're all pushed down in an even line. Lightly tug at the connector to make sure it's attached to the cable.

- If any of the pins aren't pushed down, put the wire back into the crimping tool and crimp it again.



Conclusion: We have successfully learnt crimping/cabling of RJ45 connector to Cat6 Cable using crimping tool by carefully following the above given steps.

Experiment No. 2

Aim: To use basic networking commands in Linux (ping, tracert, nslookup, netstat, ARP, RARP, ip, ifconfig, dig, route)

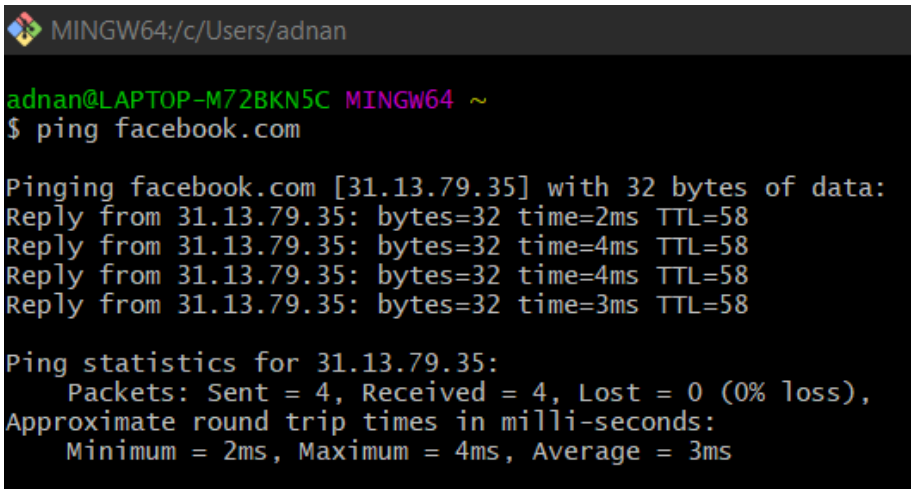
Requirements: Windows/Linux/MAC OS in PC/Laptop, compatible version of terminal in OS.

Theory:

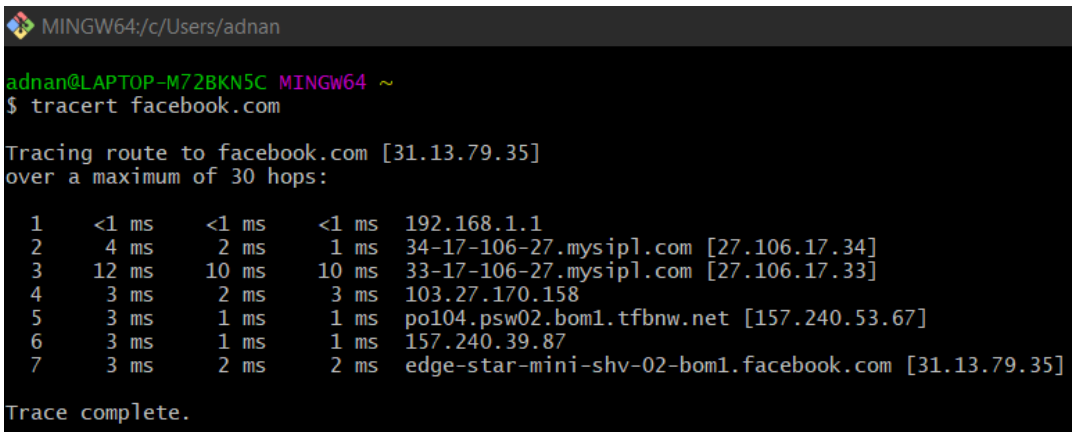
Networking commands	Operations
ping(packet internet groper)	This command is used to check the network connectivity between host and server/host.
tracert(trace route)	It's used to show the path from the source computer to the destination computer
nslookup**(**name server lookup**)**	It translates a domain name to an IP address and vice versa
netstat	Displays active TCP connections, ports on which the computer is listening, Ethernet statistics, the IP routing table, IPv4 statistics (for the IP, ICMP, TCP, and UDP protocols), and IPv6 statistics (for the IPv6, ICMPv6, TCP over IPv6, and UDP over IPv6 protocols).
arp(address resolution protocol)	Displays and modifies entries in the Address Resolution Protocol (ARP) cache.
rarp	RARP provides the opposite service to ARP in that it is used when only the ethernet address is known and the IP address is needed.
ip	This is used to assign an address to a network interface and/or configure network interface parameters on Linux operating systems
ipconfig/ifconfig	Displays all current TCP/IP network configuration values and refreshes Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings
dig(domain information groper)	The dig command, allows you to query information about various DNS records, including host addresses, mail exchanges, and name servers.
route	Displays and modifies the entries in the local IP routing table.

Commands and Output:

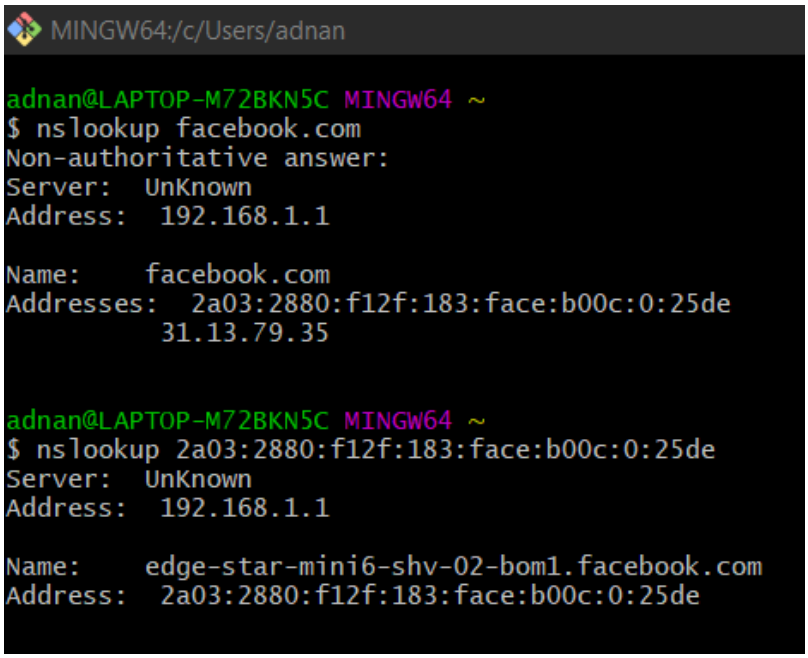
ping:



tracert:



nslookup:



netstat:

```
MINGW64:/c/Users/adnan

adnan@LAPTOP-M72BKN5C MINGW64 ~
$ netstat -ao

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            LAPTOP-M72BKN5C:0      LISTENING       1320
  TCP    0.0.0.0:445            LAPTOP-M72BKN5C:0      LISTENING       4
  TCP    0.0.0.0:3306           LAPTOP-M72BKN5C:0      LISTENING       6344
  TCP    0.0.0.0:5040           LAPTOP-M72BKN5C:0      LISTENING       10180
  TCP    0.0.0.0:5357           LAPTOP-M72BKN5C:0      LISTENING       4
  TCP    0.0.0.0:5432           LAPTOP-M72BKN5C:0      LISTENING       6540
  TCP    0.0.0.0:33060          LAPTOP-M72BKN5C:0      LISTENING       6344
  TCP    0.0.0.0:49664          LAPTOP-M72BKN5C:0      LISTENING       900
  TCP    0.0.0.0:49665          LAPTOP-M72BKN5C:0      LISTENING       1004
  TCP    0.0.0.0:49666          LAPTOP-M72BKN5C:0      LISTENING       1972
  TCP    0.0.0.0:49667          LAPTOP-M72BKN5C:0      LISTENING       2216
  TCP    0.0.0.0:49668          LAPTOP-M72BKN5C:0      LISTENING       4648
  TCP    0.0.0.0:49676          LAPTOP-M72BKN5C:0      LISTENING       932
  TCP    0.0.0.0:50128          LAPTOP-M72BKN5C:0      LISTENING       4
  TCP    127.0.0.1:3213         LAPTOP-M72BKN5C:0      LISTENING       5924
  TCP    127.0.0.1:5354         LAPTOP-M72BKN5C:0      LISTENING       5028
  TCP    127.0.0.1:5354         LAPTOP-M72BKN5C:49670  ESTABLISHED     5028
  TCP    127.0.0.1:5354         LAPTOP-M72BKN5C:49671  ESTABLISHED     5028
  TCP    127.0.0.1:5939         LAPTOP-M72BKN5C:0      LISTENING       5584
  TCP    127.0.0.1:27015        LAPTOP-M72BKN5C:0      LISTENING       5020
  TCP    127.0.0.1:49670        LAPTOP-M72BKN5C:5354   ESTABLISHED     5020
  TCP    127.0.0.1:49671        LAPTOP-M72BKN5C:5354   ESTABLISHED     5020
  TCP    127.0.0.1:49672        LAPTOP-M72BKN5C:49673  ESTABLISHED     6344
  TCP    127.0.0.1:49673        LAPTOP-M72BKN5C:49672  ESTABLISHED     6344
```

arp:

```
MINGW64:/c/Users/adnan

adnan@LAPTOP-M72BKN5C MINGW64 ~
$ arp -a

Interface: 192.168.56.1 --- 0x7
  Internet Address      Physical Address      Type
  192.168.56.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static

Interface: 192.168.1.108 --- 0xe
  Internet Address      Physical Address      Type
  192.168.1.1           38-6b-1c-be-27-71     dynamic
  192.168.1.105         34-ce-00-23-01-1d     dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

ip:

```
adnan@adnan-VirtualBox:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:17:b3:0f brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85211sec preferred_lft 85211sec
    inet6 fe80::a614:9179:6a9f:bf95/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
adnan@adnan-VirtualBox:~$ ip link ls up
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group default qlen 1000
    link/ether 08:00:27:17:b3:0f brd ff:ff:ff:ff:ff:ff
adnan@adnan-VirtualBox:~$
```

Ipconfig/ifconfig:

```
adnan@LAPTOP-M72BKN5C MINGW64 ~
$ ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

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

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::bde0:e9ac:a1e2:4330%7
   IPv4 Address. . . . . . . . . . . : 192.168.56.1
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :

Wireless LAN adapter Local Area Connection* 1:

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

Wireless LAN adapter Local Area Connection* 2:

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

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::f9b5:dc45:7c55:698a%14
   IPv4 Address. . . . . . . . . . . : 192.168.1.108
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1

Ethernet adapter Ethernet 3:

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

Ethernet adapter Bluetooth Network Connection:

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

dig:

```
adnan@adnan-VirtualBox:~$ dig linux.org

; <<>> DiG 9.16.6-Ubuntu <<>> linux.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52074
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;linux.org.                     IN      A

;; ANSWER SECTION:
linux.org.              300     IN      A       104.21.50.111
linux.org.              300     IN      A       172.67.161.161

;; Query time: 8 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Sun Aug 01 18:46:18 IST 2021
;; MSG SIZE  rcvd: 70
```

route:

```
adnan@LAPTOP-M72BKN5C MINGW64 ~
$ route print
===========================================================================
Interface List
  6...54 05 db 10 4a 64 ......Realtek PCIe GbE Family Controller
  7...0a 00 27 00 00 07 ......VirtualBox Host-Only Ethernet Adapter
 21...a4 b1 c1 17 71 a5 ......Microsoft Wi-Fi Direct Virtual Adapter
 11...a6 b1 c1 17 71 a4 ......Microsoft Wi-Fi Direct Virtual Adapter #2
 14...a4 b1 c1 17 71 a4 ......Intel(R) Wi-Fi 6 AX201 160MHz
  9...00 ff 6c d4 a8 15 ......TeamViewer VPN Adapter
 20...a4 b1 c1 17 71 a8 ......Bluetooth Device (Personal Area Network)
  1...........................Software Loopback Interface 1
===========================================================================

IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      192.168.1.1    192.168.1.108     35
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    331
      192.168.1.0    255.255.255.0         On-link     192.168.1.108    291
    192.168.1.108  255.255.255.255         On-link     192.168.1.108    291
    192.168.1.255  255.255.255.255         On-link     192.168.1.108    291
     192.168.56.0    255.255.255.0         On-link      192.168.56.1    281
     192.168.56.1  255.255.255.255         On-link      192.168.56.1    281
   192.168.56.255  255.255.255.255         On-link      192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link      192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link     192.168.1.108    291
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link      192.168.56.1    281
  255.255.255.255  255.255.255.255         On-link     192.168.1.108    291
===========================================================================
Persistent Routes:
  None

IPv6 Route Table
===========================================================================
Active Routes:
 If Metric Network Destination      Gateway
  1    331 ::1/128                  On-link
  7    281 fe80::/64                On-link
 14    291 fe80::/64                On-link
  7    281 fe80::bde0:e9ac:a1e2:4330/128
                                    On-link
 14    291 fe80::f9b5:dc45:7c55:698a/128
                                    On-link
  1    331 ff00::/8                 On-link
  7    281 ff00::/8                 On-link
 14    291 ff00::/8                 On-link
===========================================================================
Persistent Routes:
  None
```

Conclusion: We have successfully executed and got the output of basic networking commands (ping, tracert, nslookup, netstat, ARP,RARP, ip, ifconfig, dig, route) in Linux Shell.

Experiment No. 3

Aim: To build a simple network topology and configure it for static routing protocol using packet tracer.

Requirements: Windows OS in P.C and Stable version of CISCO packet tracer.

Theory:

The arrangement of wires, work stations (P.C.) and other peripherals in a network is known as network topology.

Some of the topologies widely known and in used are Mesh Topology, Star Topology, Bus Topology, Ring Topology and Hybrid Topology (combination of two or more topology).

We're using simple network topology analogous to Star topology for static routing protocol using CISCO packet tracer.

Network Configuration:

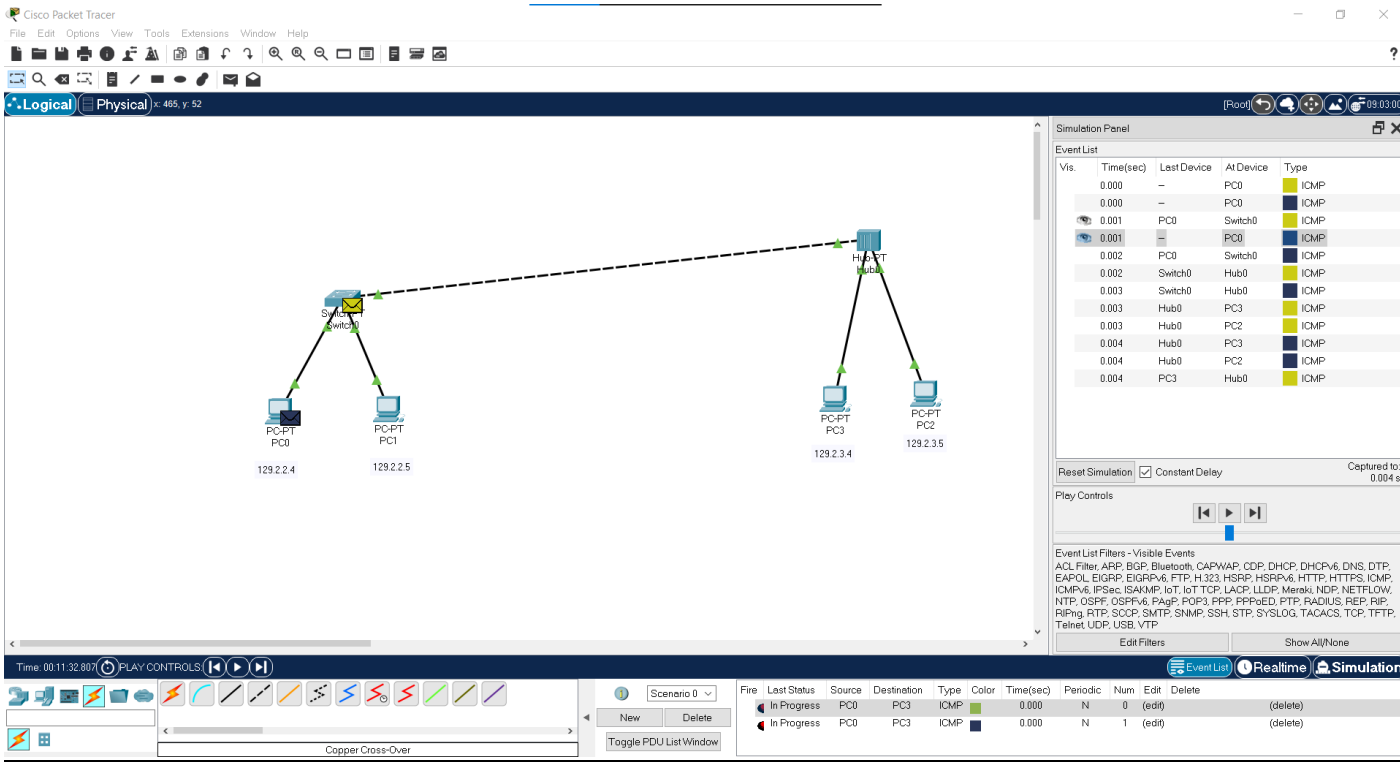
1) There are 2 different Gateways (having different address) connected through switch and hub using copper cross-over wire.

2) Switch in 1st gateway is connected to two P.C. in Star topology using copper straight-through wire each having same Gateway address but different I.P address.

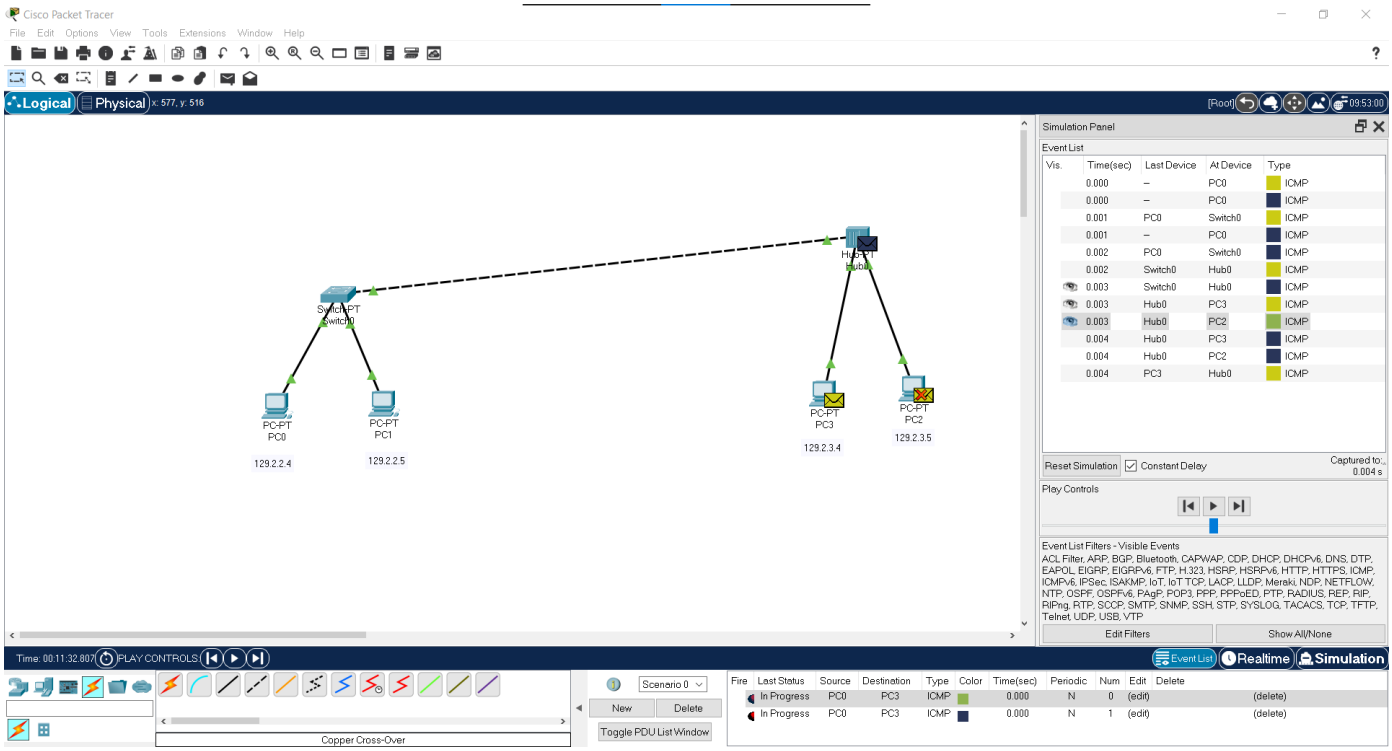
3) Hub in 2nd gateway is connected to two P.C. in Star topology using copper straight-through wire each having same Gateway address but different I.P address.

Demonstration of sending packet from PC0 (in 1st Gateway) to PC3 (in 2nd gateway):

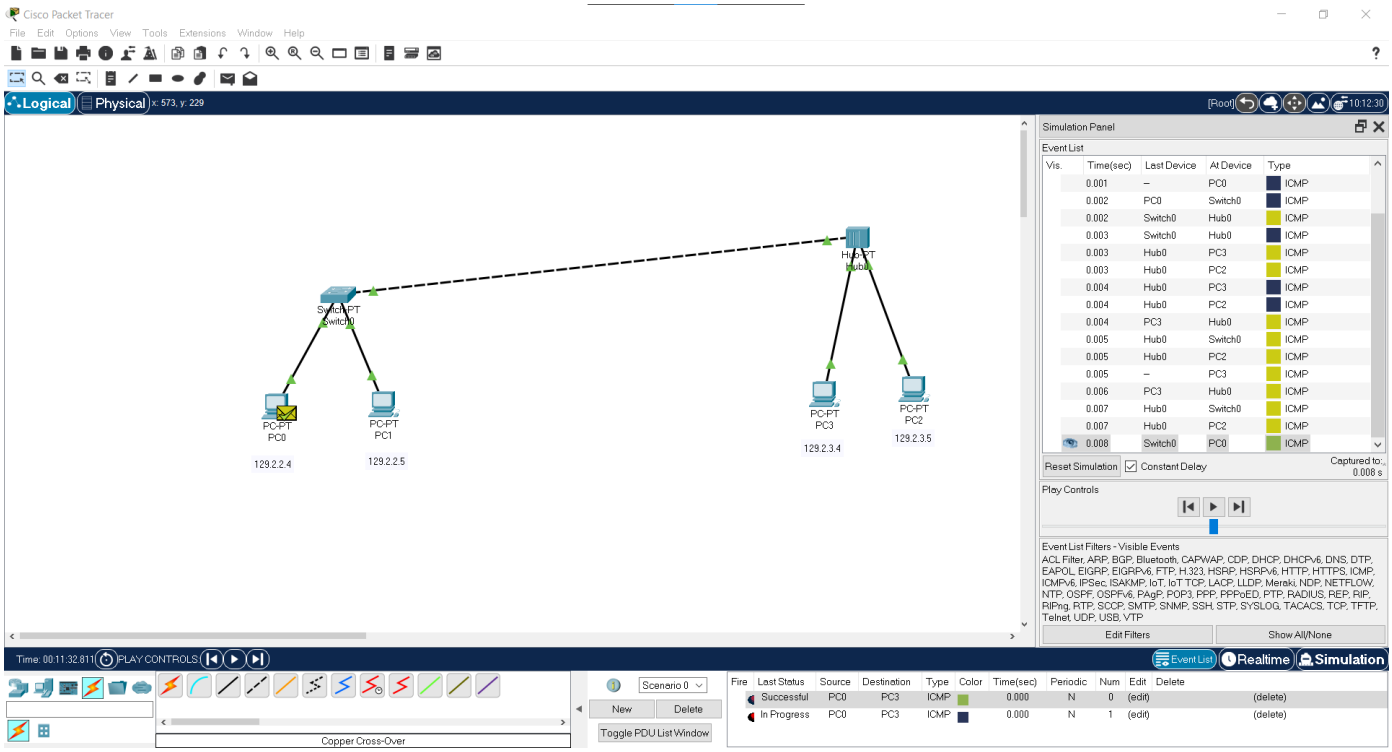
1) Packet send from PC0 to Switch:



2) Packet route to HUB from switch which in turn route to all PC in HUB network and PC3 accepted the package and other PC rejected it:



3) PC0 received acknowledgment packet has been delivered successfully:



Conclusion: We have successfully understand the concept of network topology and built a simple network topology and configure it for static routing protocol using CISCO packet tracer.

Experiment No. 5

Aim: To set up multiple IP addresses on a single LAN and using netstat and route commands viewing current routing table.

Requirement: Windows/Linux/MAC OS in PC/Laptop, compatible version of terminal in OS.

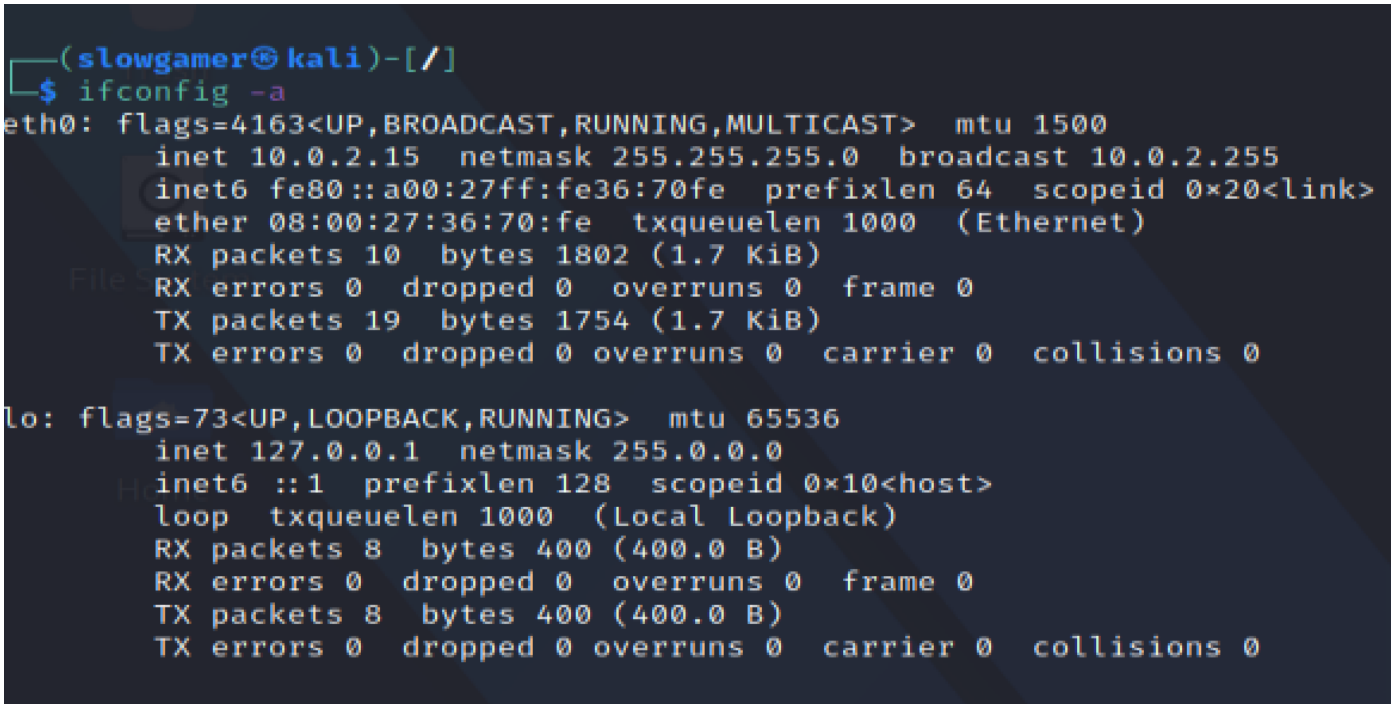
Theory:

The concept of creating or configuring multiple IP addresses on a single network interface is called IP aliasing. IP aliasing is very useful for setting up multiple virtual sites on Apache using one single network interface with different IP addresses on a single subnet network.

The main advantage of using this IP aliasing is, you don't need to have a physical adapter attached to each IP, but instead you can create multiple or many virtual interfaces (aliases) to a single physical card.

Below we create virtual interface and assign multiple IP Address in Kali Linux:

1) Before ipaliasing:



2) After ipaliasing:

```
┌──(slowgamer㉿kali)-[/]
└─$ sudo ifconfig eth0:0 10.0.1.15 up

┌──(slowgamer㉿kali)-[/]
└─$ sudo ifconfig eth0:1 10.0.1.16 up

┌──(slowgamer㉿kali)-[/]
└─$ sudo ifconfig eth0:2 10.0.1.17 up

┌──(slowgamer㉿kali)-[/]
└─$ ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe36:70fe  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:36:70:fe  txqueuelen 1000  (Ethernet)
        RX packets 10  bytes 1802 (1.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 19  bytes 1754 (1.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0:0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.1.15  netmask 255.0.0.0  broadcast 10.255.255.255
        ether 08:00:27:36:70:fe  txqueuelen 1000  (Ethernet)

eth0:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.1.16  netmask 255.0.0.0  broadcast 10.255.255.255
        ether 08:00:27:36:70:fe  txqueuelen 1000  (Ethernet)

eth0:2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.1.17  netmask 255.0.0.0  broadcast 10.255.255.255
        ether 08:00:27:36:70:fe  txqueuelen 1000  (Ethernet)

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 8  bytes 400 (400.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 400 (400.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

3) Netstat and route command:

```
┌──(slowgamer㉿kali)-[~]
└─$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         10.0.2.2        0.0.0.0         UG        0 0          0 eth0
10.0.0.0        0.0.0.0         255.0.0.0       U         0 0          0 eth0
10.0.1.15       0.0.0.0         255.255.255.255 UH        0 0          0 eth0
10.0.1.16       0.0.0.0         255.255.255.255 UH        0 0          0 eth0
10.0.1.17       0.0.0.0         255.255.255.255 UH        0 0          0 eth0
10.0.1.18       0.0.0.0         255.255.255.255 UH        0 0          0 eth0
10.0.2.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0

┌──(slowgamer㉿kali)-[~]
└─$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.0.2.2        0.0.0.0         UG    100    0        0 eth0
10.0.0.0        0.0.0.0         255.0.0.0       U     0      0        0 eth0
10.0.1.15       0.0.0.0         255.255.255.255 UH    0      0        0 eth0
10.0.1.16       0.0.0.0         255.255.255.255 UH    0      0        0 eth0
10.0.1.17       0.0.0.0         255.255.255.255 UH    0      0        0 eth0
10.0.1.18       0.0.0.0         255.255.255.255 UH    0      0        0 eth0
10.0.2.0        0.0.0.0         255.255.255.0   U     100    0        0 eth0
```

Conclusion: We have successfully added Multiple IP addresses to single NIC in Kali Linux OS and using netstat and route command we have displayed routing table.

Experiment No. 6

Aim: To perform remote login using Telnet server.

Requirement: Windows/Linux OS in P.C., CISCO Packet Tracer.

Theory:

Telnet:

Telnet is a network protocol used to virtually access a computer and to provide a two-way, collaborative and text-based communication channel between two machines. It follows a user command Transmission Control Protocol/Internet Protocol (TCP/IP) networking protocol for creating remote sessions. On the web, Hypertext Transfer Protocol (HTTP) and File Transfer Protocol (FTP) simply enable users to request specific files from remote computers, while, through Telnet, users can log on as a regular user with the privileges they are granted to the specific applications and data on that computer.

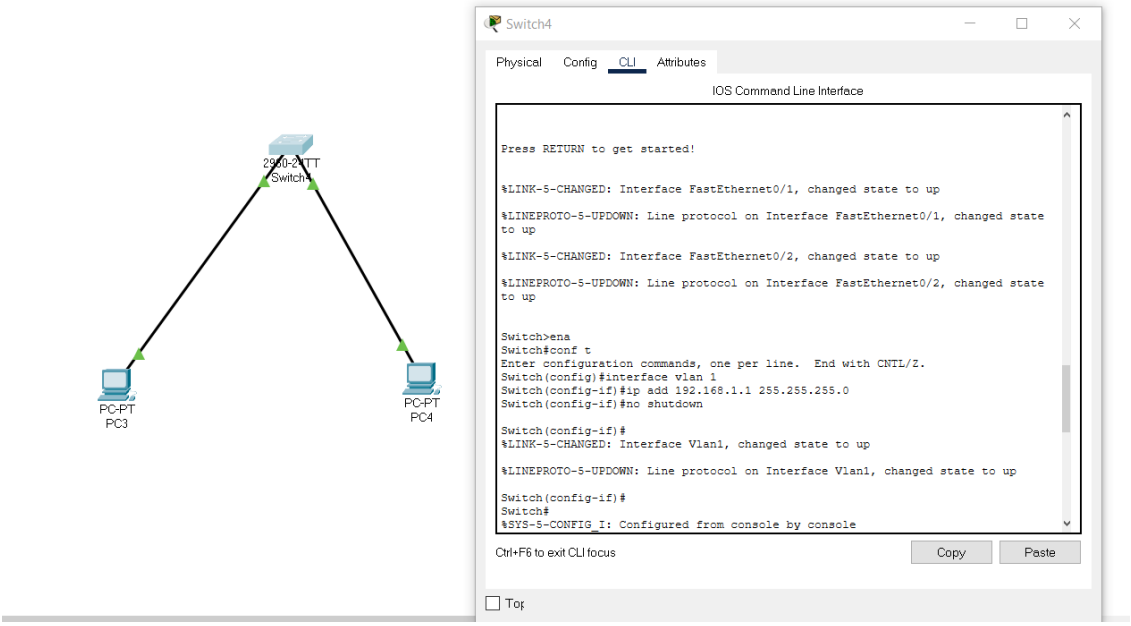
Telnet Working:

Telnet is a type of client-server protocol that can be used to open a command line on a remote computer, typically a server. Users can utilize this tool to ping a port and find out whether it is open. Telnet works with what is called a virtual terminal connection emulator, or an abstract instance of a connection to a computer, using standard protocols to act like a physical terminal connected to a machine.

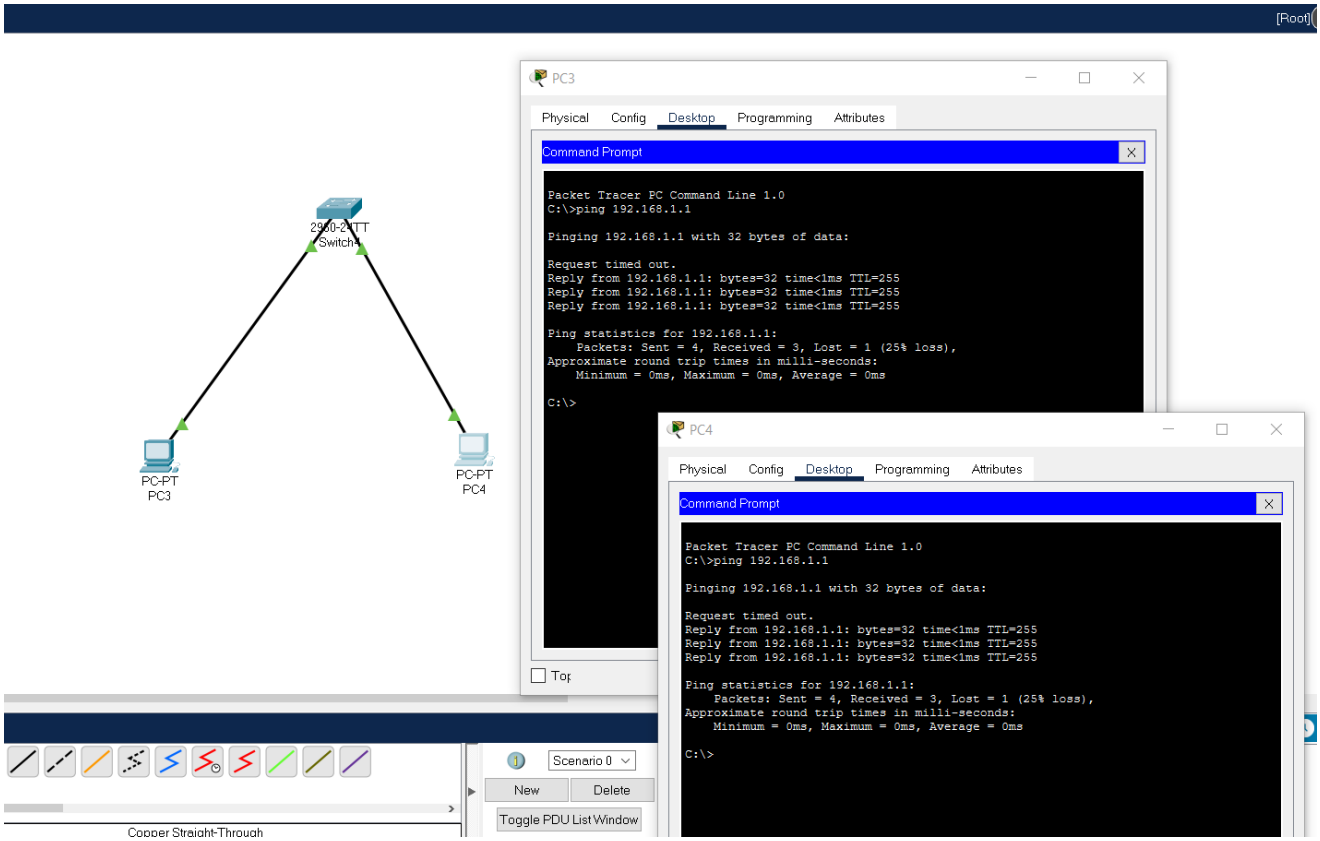
Users connect remotely to a machine using Telnet, sometimes referred to as Telnetting into the system. They are prompted to enter their username and password combination to access the remote computer, which enables the running of command lines as if logged in to the computer in person. Despite the physical location of users, their IP address will match the computer logged in to rather than the one physically used to connect.

Telnet using CISCO Packet Tracer:

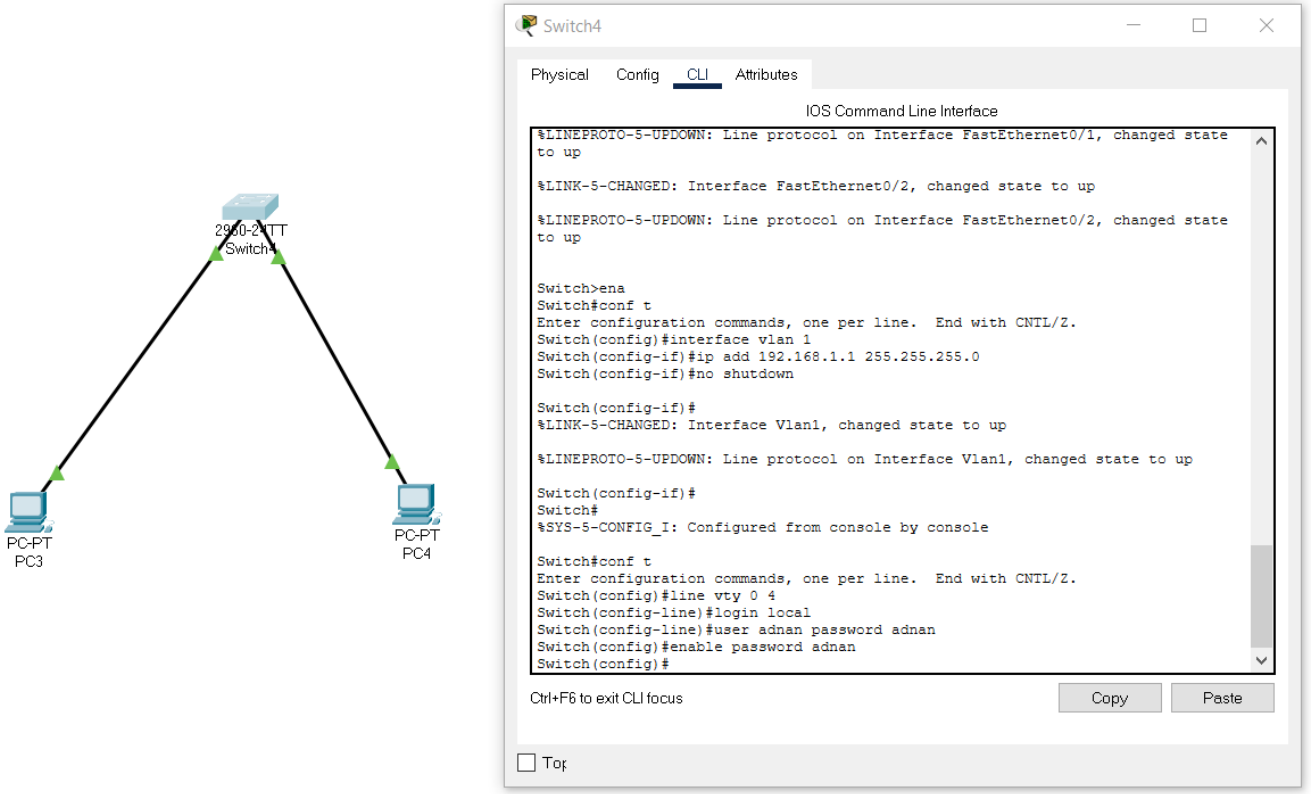
1) Configuring Switch:



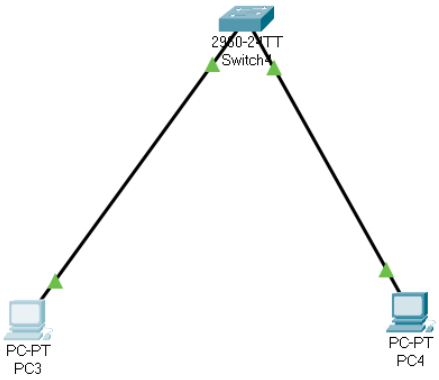
2) Pinging switch from PC3 and PC4:

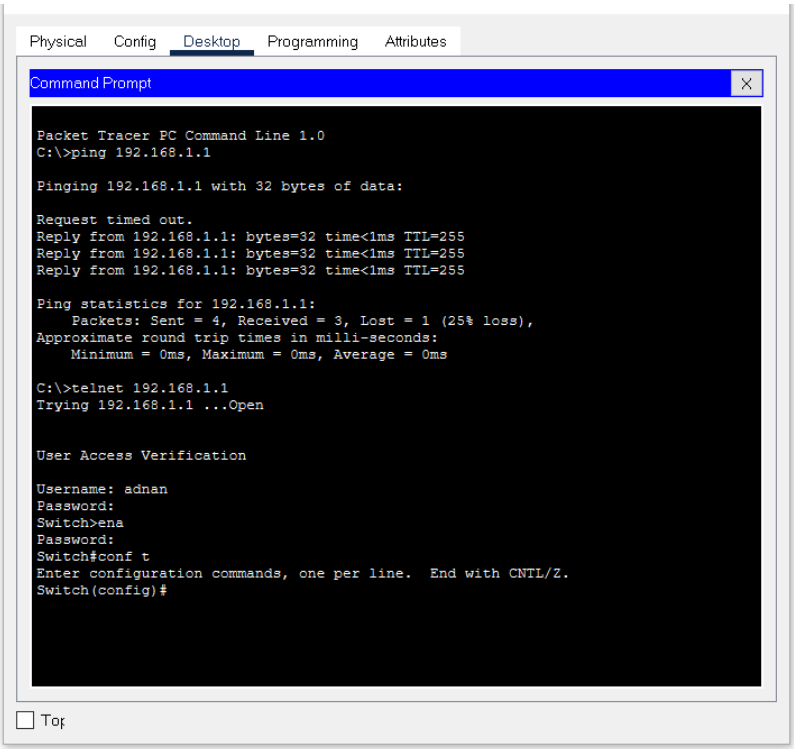


3) Configuring switch(Telnet Server) for remote login using Telnet:



4) Remote login from PC3 to Telnet Server using telnet command:





Conclusion: We have successfully remote login from P.C. to Telnet Server by performing proper configuration in CISCO Packet Tracer.