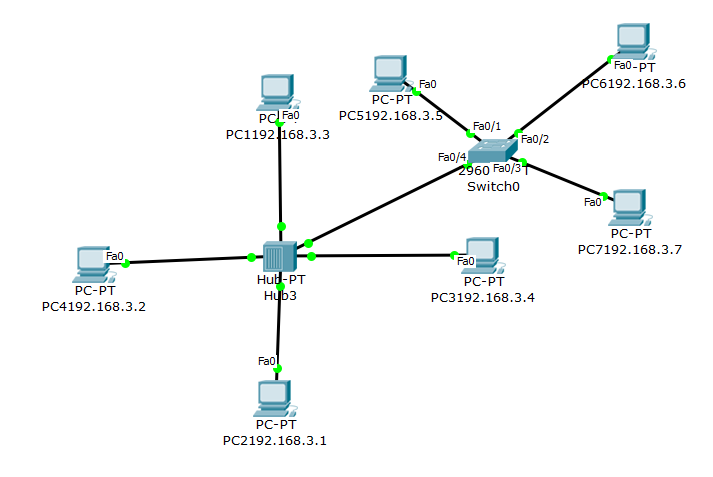
**Objective:** In hub any data packet coming from one port is sent to all other ports and receiver computers decide if the packet is for it or not. Instead of broadcasting the frames everywhere, a switch actually checks for the destination MAC address and forward it to the relevant port to reach that computer only.

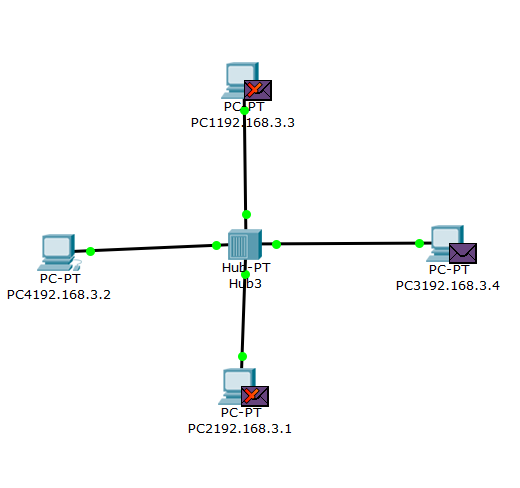
**Apparatus:** PC, simulator software (CISCO packet tracer).

**Network Diagram:**

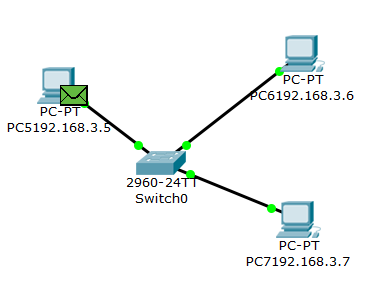


**Methodology:**

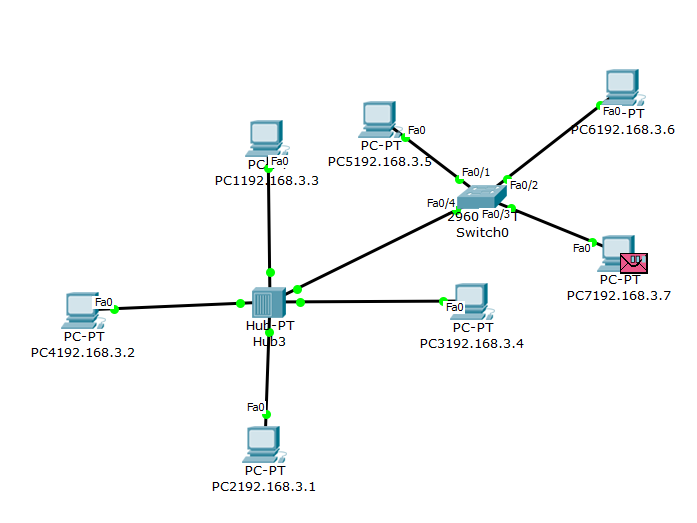
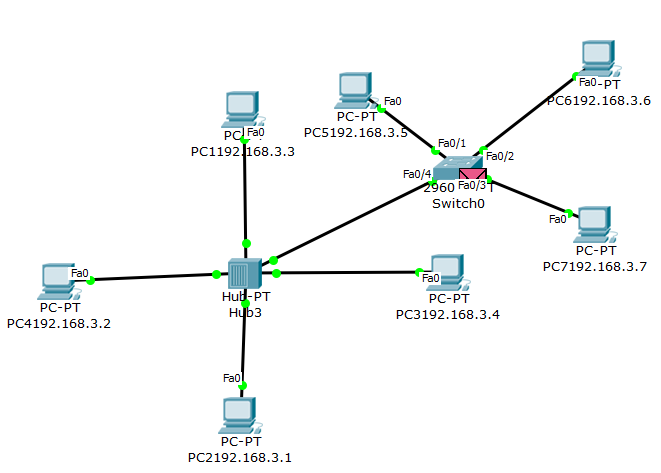
1. At first, 4 generic PCs and a generic hub have been taken and the PCs are provided with their IPs. Then PCs have been connected with the hub with copper straight through wire. A packet has been sent from IP 192.168.3.2 to IP 192.168.3.4 through hub.

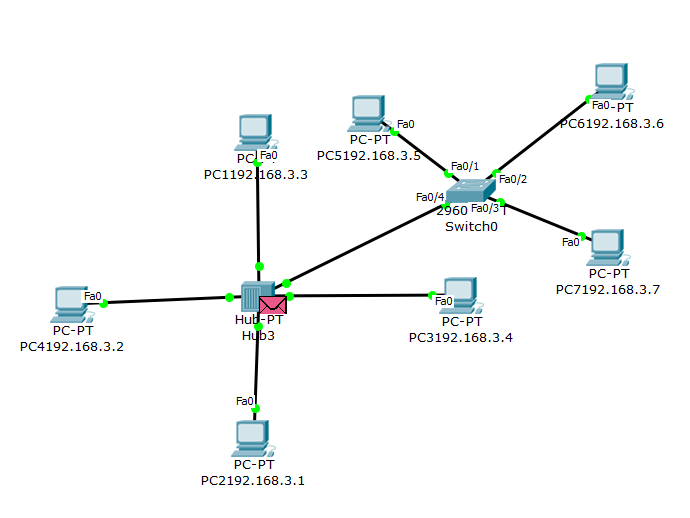
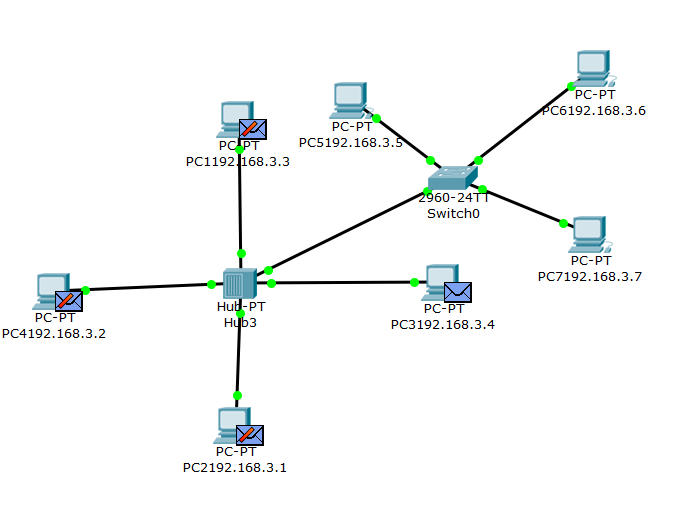


1. Then a 2960 switch has been connected with 3 generic PCs. The PCs are provided with their IP addresses. A data packet has been sent from IP 192.168.3.7 to IP 192.168.3.5 through the switch.

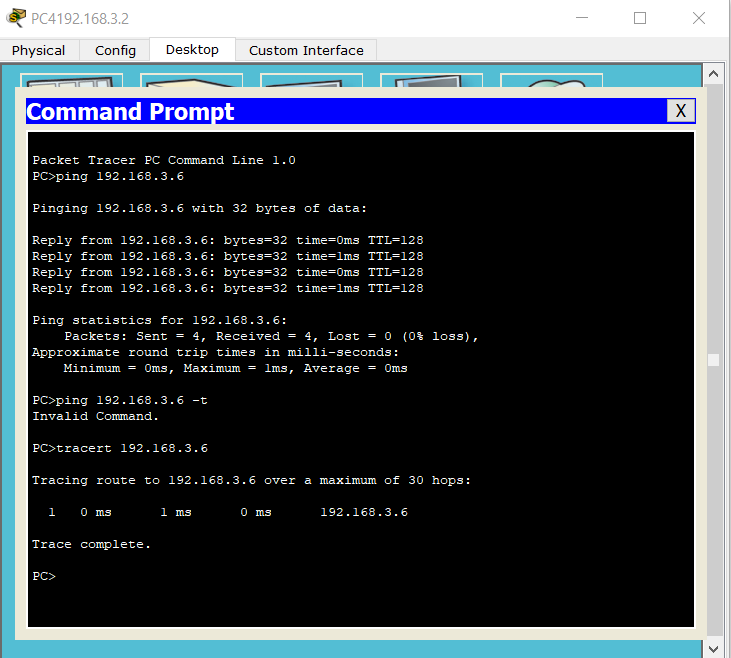


1. Now the switch and the hub have been connected and a packet has been sent from IP 192.168.3.7 to IP 192.168.3.4.





**Result:** Now to check data sending to a particular IP the following ping command can be used. Tracert command has been used to check which IPs have been roamed to reach the server.

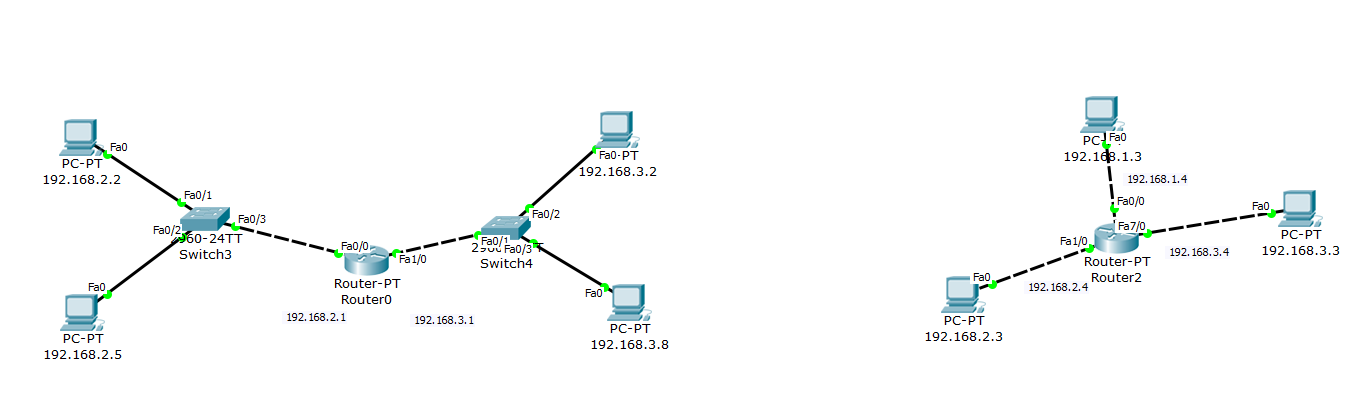


**Discussion:** In this experiment, we can see data transmission though hub and switch. Hub sends data to all the connected devices and switch sends data only to the destination device by checking MAC address.

**Objective:** A router is a device that connects two or more packet-switched networks or subnetworks.It takes data packets from devices and directs them to the right place using IP addresses. A switch actually checks for the destination MAC address and forward it to the relevant port to reach that computer only.

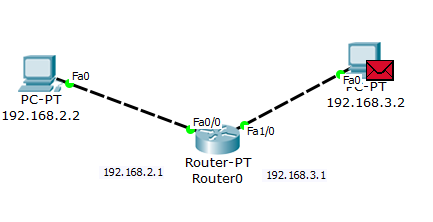
**Apparatus:** PC, simulator software (CISCO packet tracer).

**Network Diagram:**

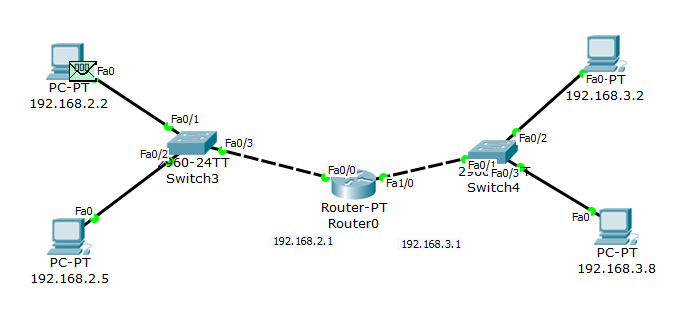
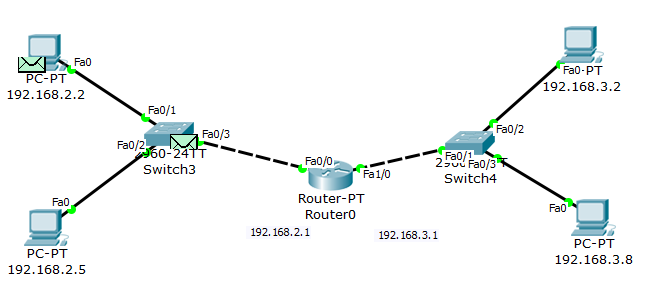
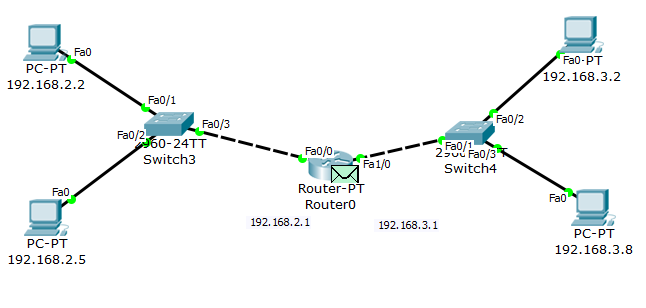
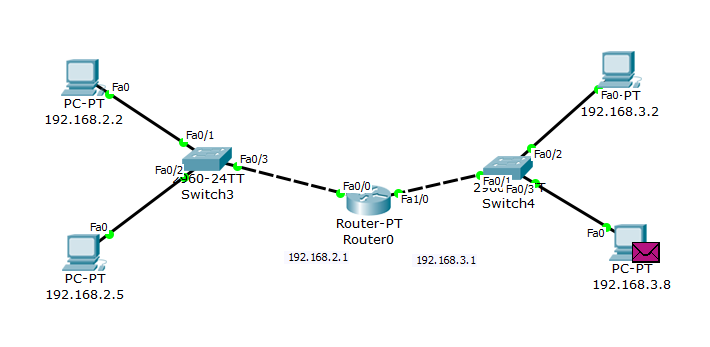


**Methodology:**

1. At first, 2 generic PCs and a generic router have been taken. PCs are provided with their IPs and the router is provided with fast Ethernet port address. They have been connected with copper cross-over wire. A data packet has been sent from IP 192.168.2.2 to IP 192.168.3.2 through the router. Also checking using ping command.

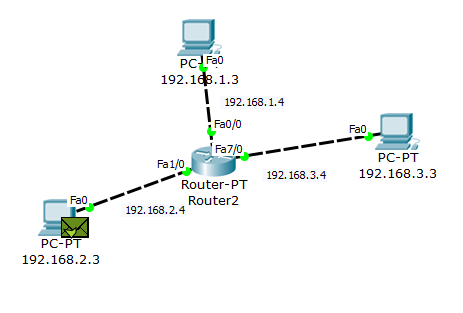


1. Next adding 2960 switch between router and PCs and sending a packet from IP 192.168.2.2 to IP 192.168.3.8. Copper straight-through wire has been used to connect PCs with switches.

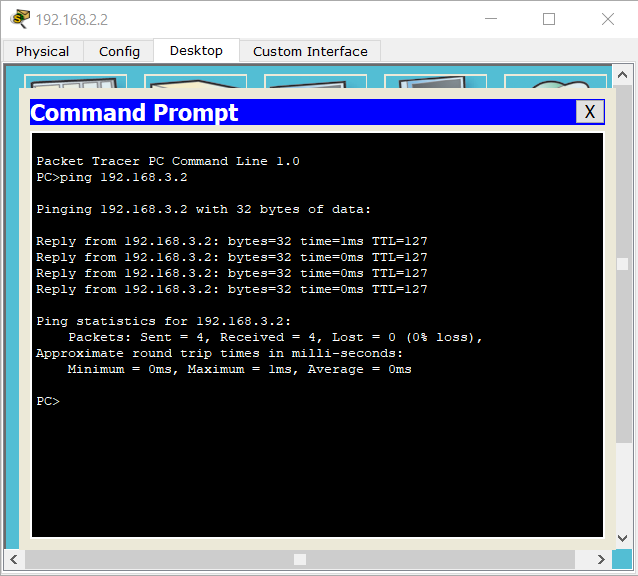


1. Now inserting 2 Ethernet cards to the router and sending packets through the new cards of the router.





**Result:**

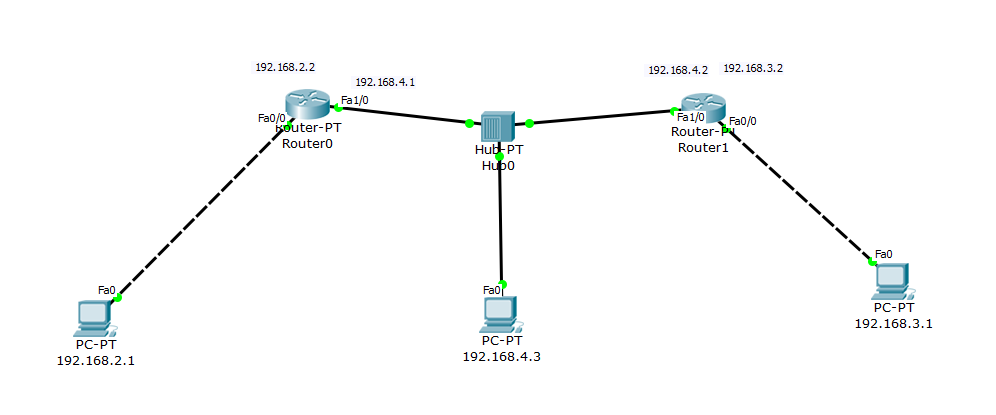


**Discussion:** In this experiment, we can observe data transmission using routers. Routers can connect multiple network and send data only to the destination network by checking IP addresses.

**Objective:** A router is a device that connects two or more packet-switched networks or subnetworks.It takes data packets from devices and directs them to the right place using IP addresses. In hub any data packet coming from one port is sent to all other ports and receiver computers decide if the packet is for it or not.

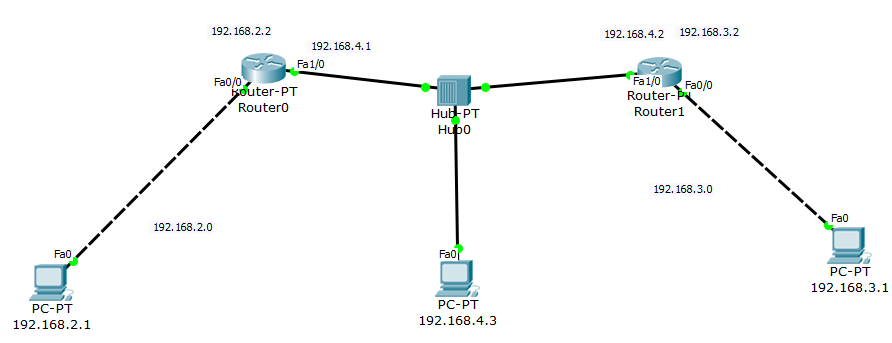
**Apparatus:** PC, simulator software (CISCO packet tracer).

**Network Diagram:**

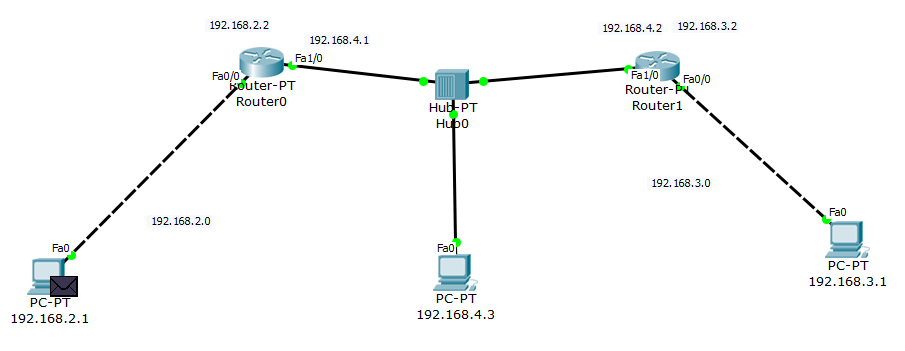
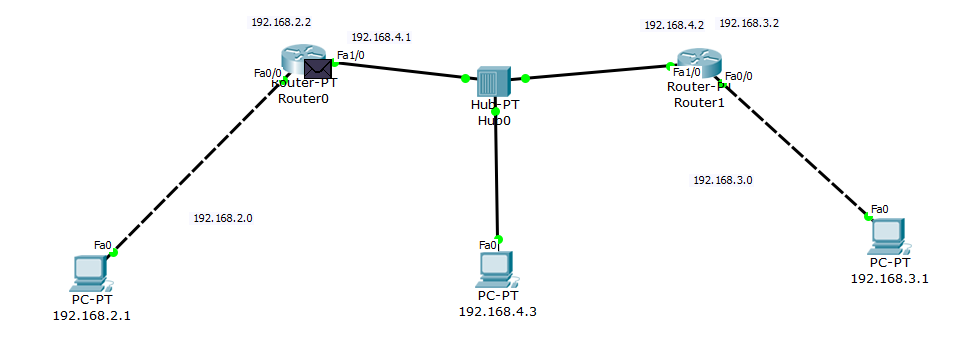
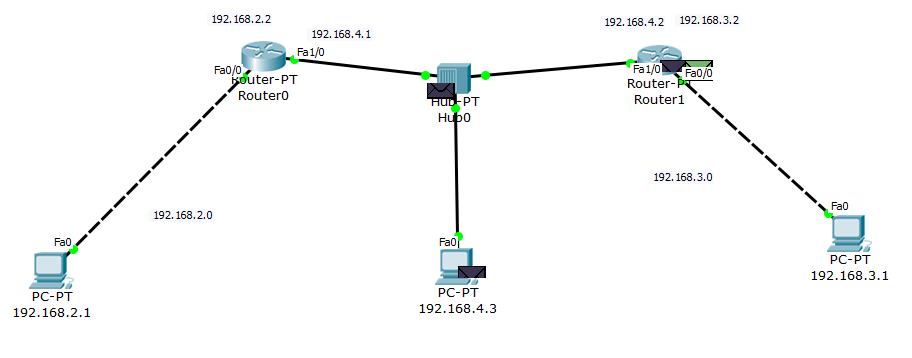


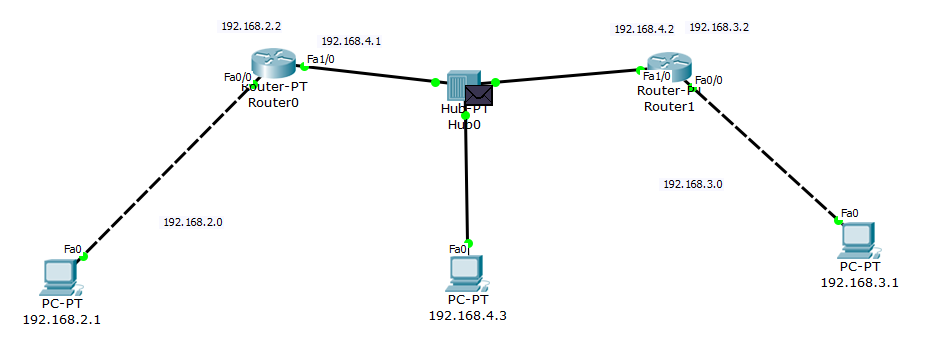
**Methodology:**

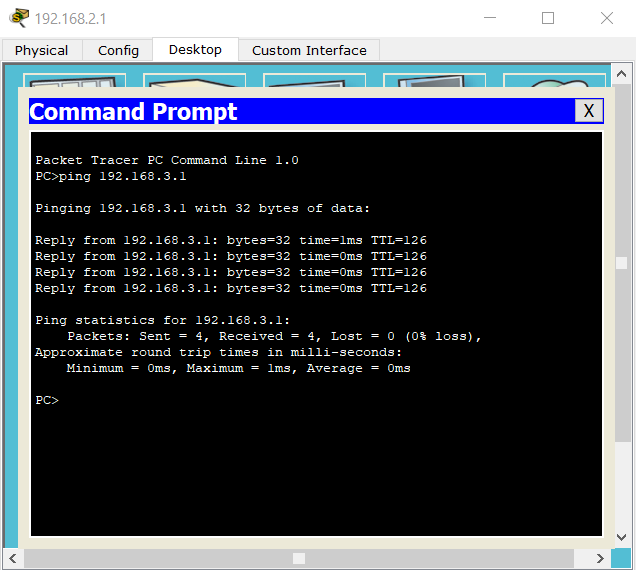
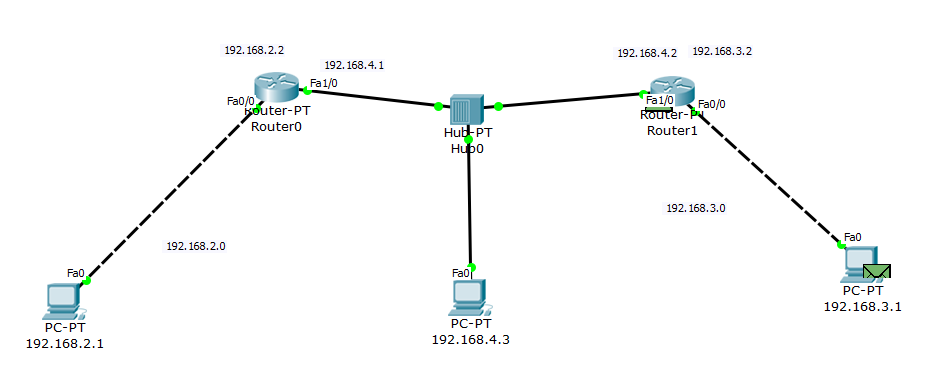
1. At first, 3 generic PCs, 2 generic routers and a generic hub have been taken. PCs are provided with their IPs along with default gateway and the routers are provided with fast Ethernet port address along with their static routes. Copper cross-over wire has been used to connect the PCs with routers and copper straight-through wire has been used to connect the hub with routers and PCs.



**Result:** An ICMP packet has been sent from IP 192.168.2.1 to IP 192.168.3.1. Also checking data transmission using ping command.





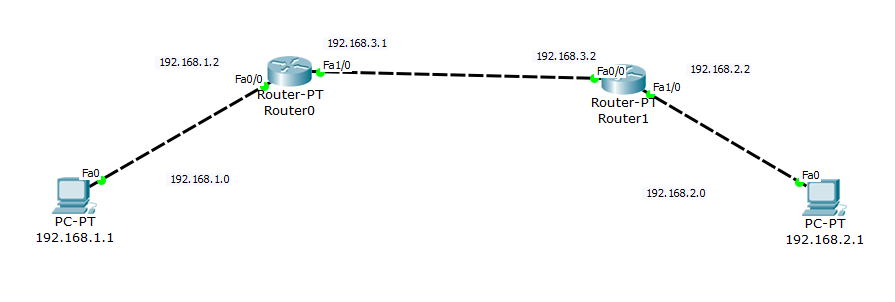


**Discussion:** In this experiment, we can get idea of data transmission using router and hub. Hub sends the data to each of the connected devices and routers are sending data only to the destination network.

**Objective:** A router is a device that connects two or more packet-switched networks or subnetworks.It takes data packets from devices and directs them to the right place using IP addresses.

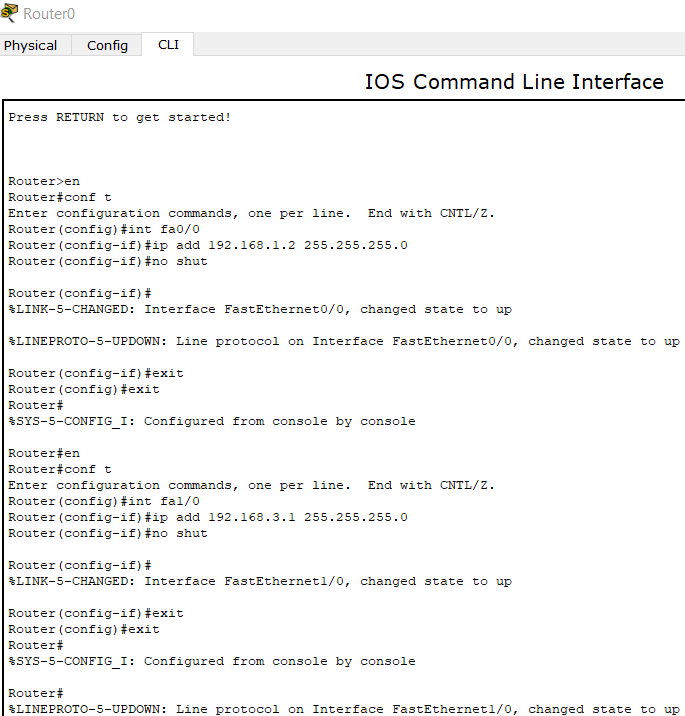
**Apparatus:** PC, simulator software (CISCO packet tracer).

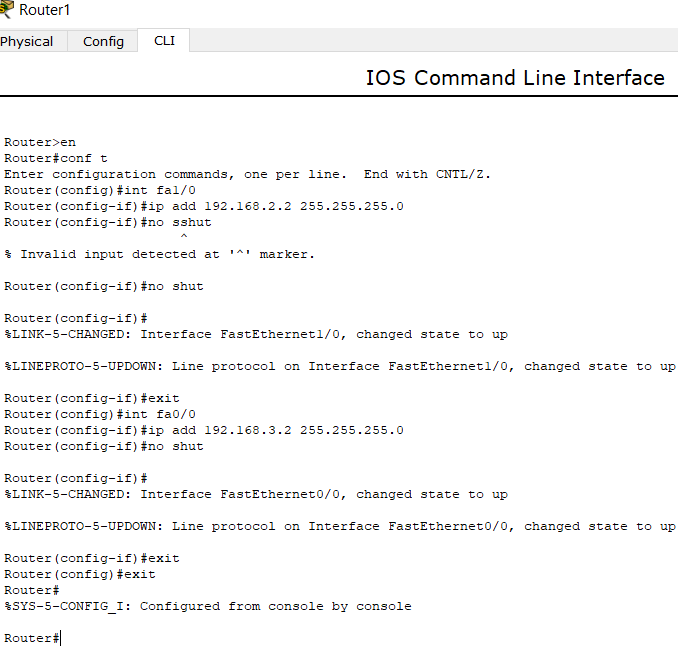
**Network Diagram:**



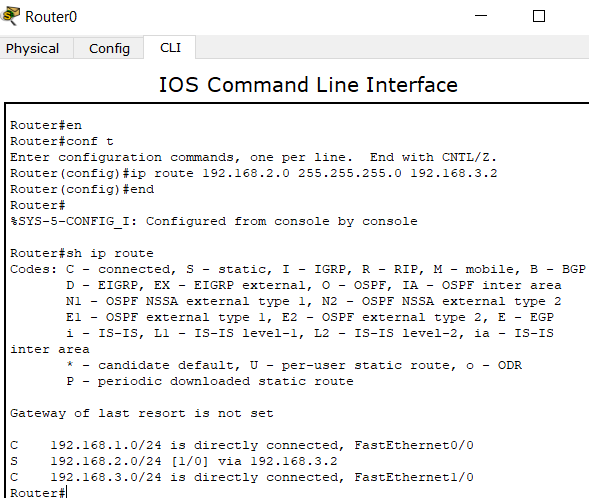
**Methodology:**

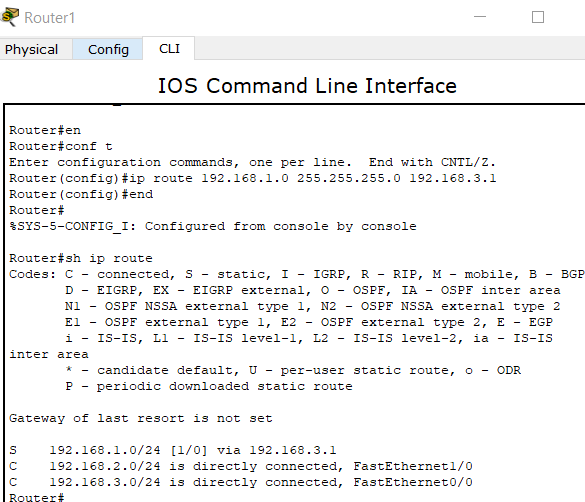
1. At first, 2 generic PCs and 2 generic routers have been taken. Copper cross-over wire has been used to connect the PCs with routers. PCs are provided with their IPs along with default gateway. The routers are provided with fast Ethernet port address through command line.



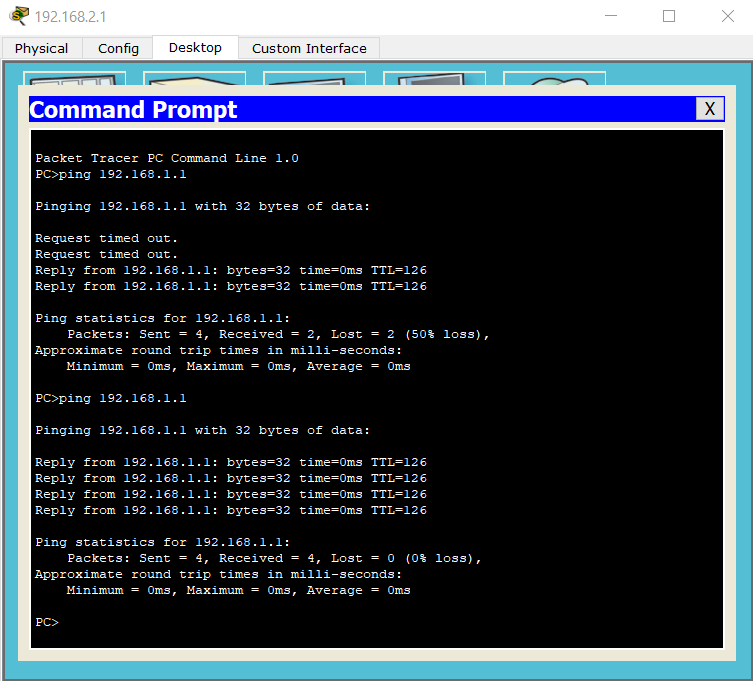


1. Now that all the connections are enabled, the static routes of both the routers are set using command line.





**Result:** An ICMP packet has been sent from IP 192.168.2.1 to IP 192.168.1.1 using ping command. At first there will be request timed out as the routers don’t know the way of another router. But next time, the router recognizes it’s way to another.

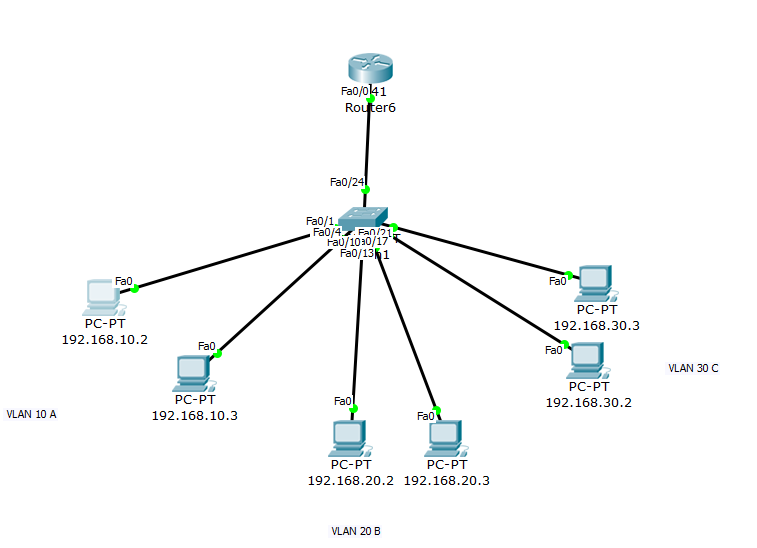


**Discussion:** In this experiment, we can observe data transmission using routers. Both the values of fast Ethernet ports and static routes has been set using command line interface instead of putting them manually. Routers can connect multiple network and send data only to the destination network by checking IP addresses.

**Objective:** A router is a device that connects two or more packet-switched networks or subnetworks.It takes data packets from devices and directs them to the right place using IP addresses. A switch actually checks for the destination MAC address and forward it to the relevant port to reach that computer only. Switch-2960 has 24 ports. VLANs are made for security issue so that data isn’t visible between one another.

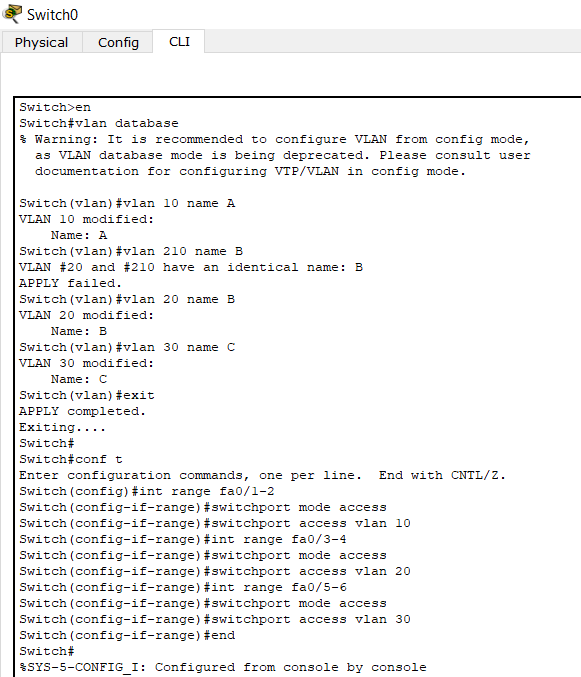
**Apparatus:** PC, simulator software (CISCO packet tracer).

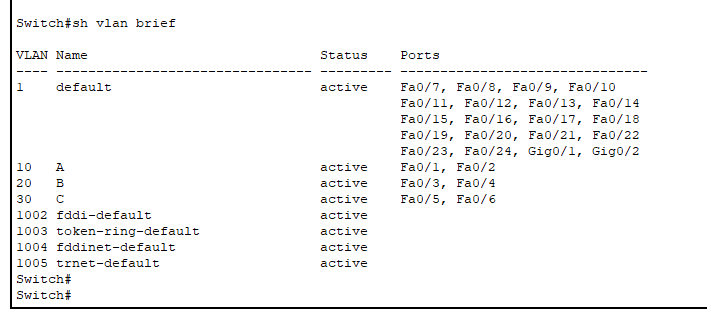
**Network Diagram:**



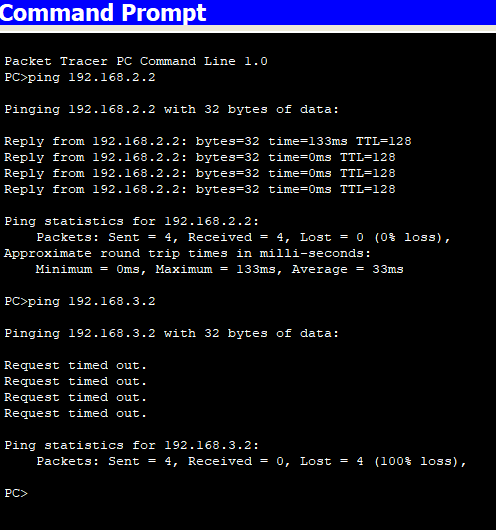
**Methodology:**

1. At first, 6 generic PCs and a 2960 switch have been taken. Copper straight-through wire has been used to connect the PCs with switch. PCs are provided with their IPs. Also the switch has been connected with their corresponding fast Ethernet port through command line interface.

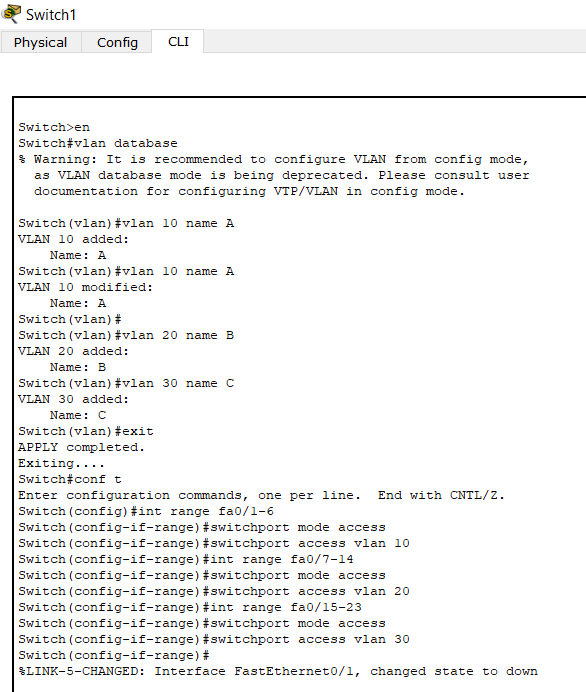


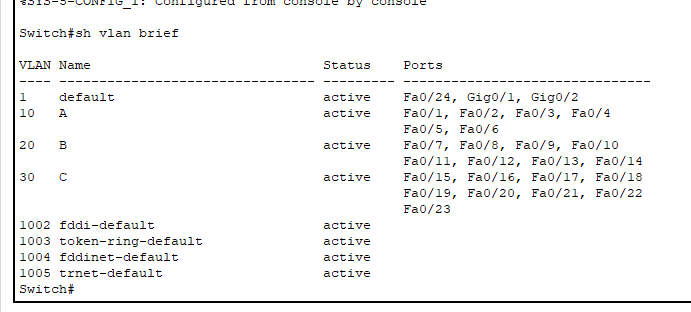


1. Next an ICMP packet has been sent from IP 192.168.2.1 to IP 192.168.2.2 using ping command. This transmission is successful. But when a packet has been added from the same IP to IP 192.168.3.2, which belongs to another VLAN, transmission is failed.

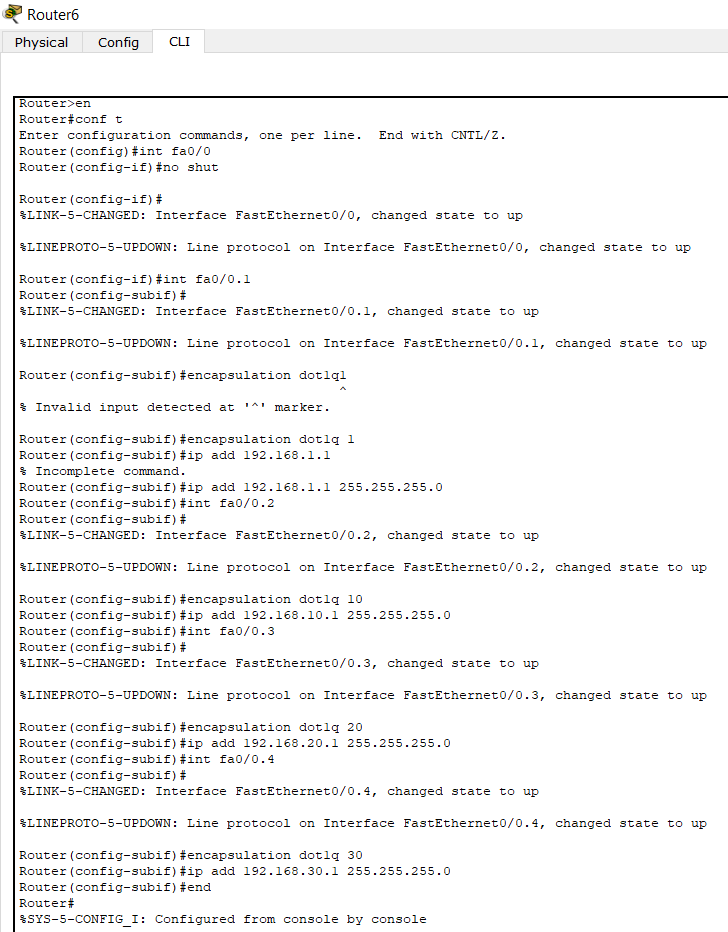


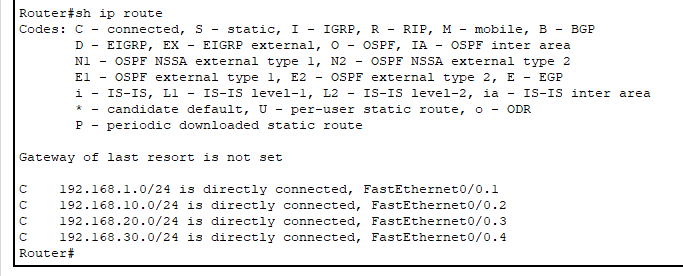
1. Now checking VLAN under sub-interface. 6 generic PCs, a 2960 switch and a router has been taken and connected using copper straight-through wire. They have been assigned with their corresponding IPs. Also the switch has been connected with their corresponding fast Ethernet port through command line interface.



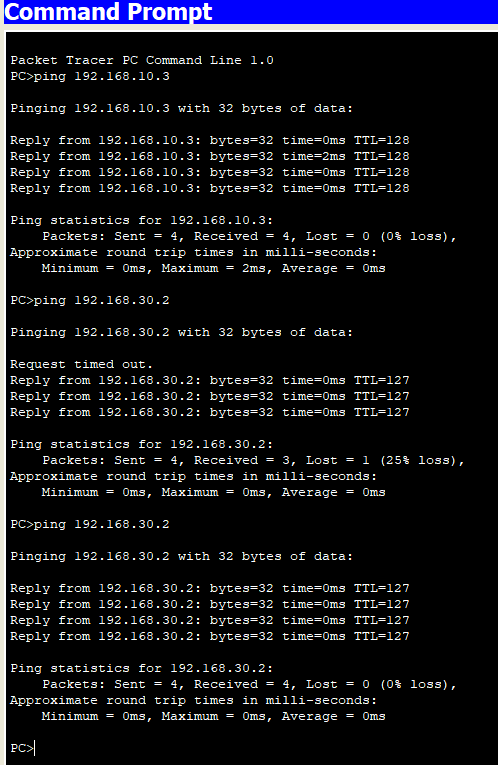


1. Then the router has been assigned fast Ethernet port along with sub interface port through command line interface.

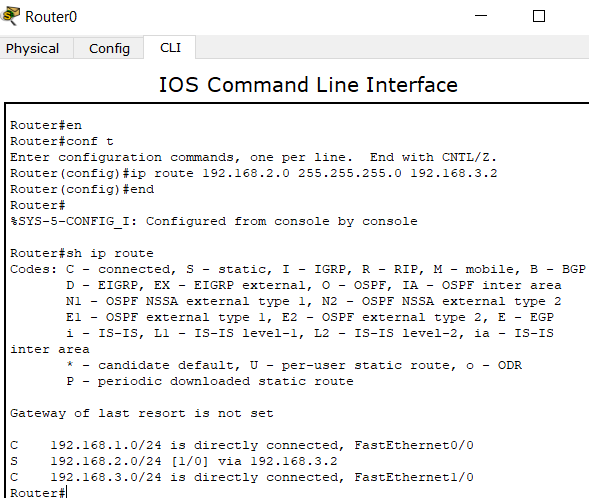


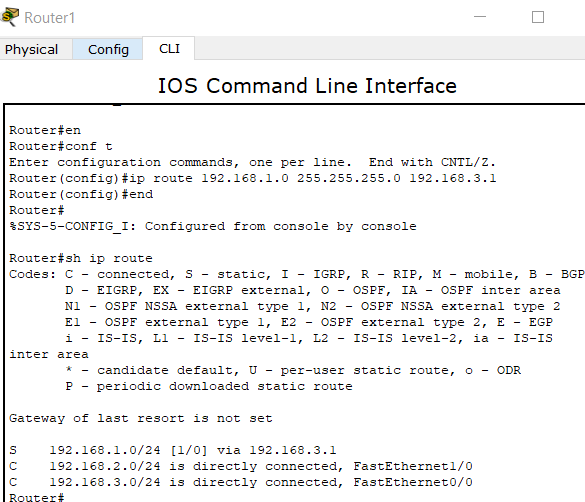


**Result:** An ICMP packet has been sent from IP 192.168.10.2 to IP 192.168.10.3 using ping command. The transmission is successful. But for IP 192.168.30.2 which exists in another VLAN from the previous IP, at first there will be request timed out as the routers don’t know the way. But next time, the router recognizes its way to IP thus making the transmission successful.

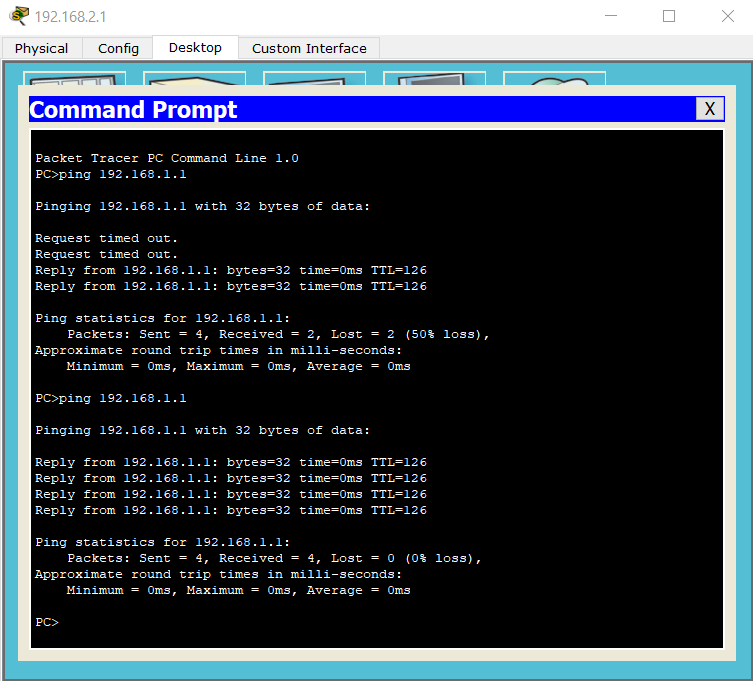


**Discussion:** In this experiment, we can observe VLAN configuration using switch and router. Both the values of fast Ethernet ports and static routes has been set using command line interface instead of putting them manually for switch and router. When only switch has been used, packet can only be transmitted within a particular VLAN, not with others. When router has been added, it checks the packet which is sent via the switch. Here interface is converted to sub-interface. Router acts as a VLAN itself. This way using VLAN data transmission can be done more securely.





**Result:** An ICMP packet has been sent from IP 192.168.2.1 to IP 192.168.1.1 using ping command. At first there will be request timed out as the routers don’t know the way of another router. But next time, the router recognizes it’s way to another.

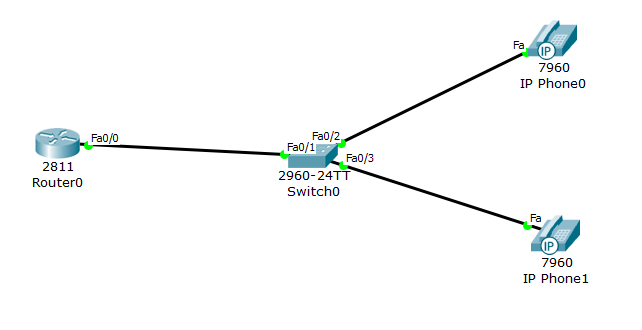


**Discussion:** In this experiment, we can observe data transmission using routers. Both the values of fast Ethernet ports and static routes has been set using command line interface instead of putting them manually. Routers can connect multiple network and send data only to the destination network by checking IP addresses.

**Objective:** Telephone has its unique IP address and number. Telephones can be connected using switches and routers can help to maintain the network.

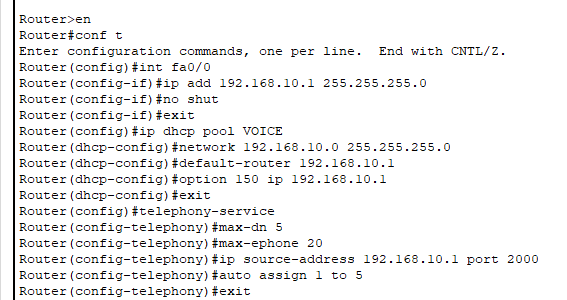
**Apparatus:** PC, simulator software (CISCO packet tracer).

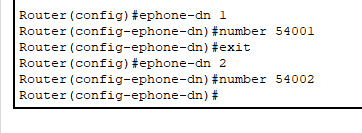
**Network Diagram:**

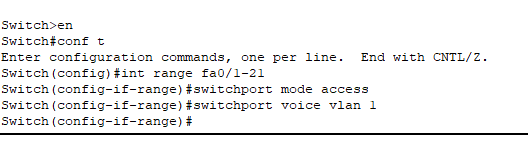


**Methodology:**

1. At first, 2 IP Phones, a 2960 switch and a 2811 router have been taken. Copper straight-through wire has been used to connect them together.
2. Next, both the router and switch have been configured using CLI.

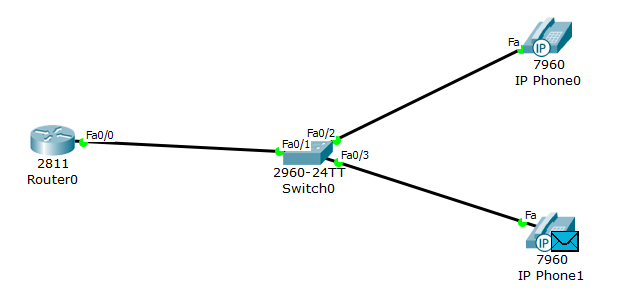
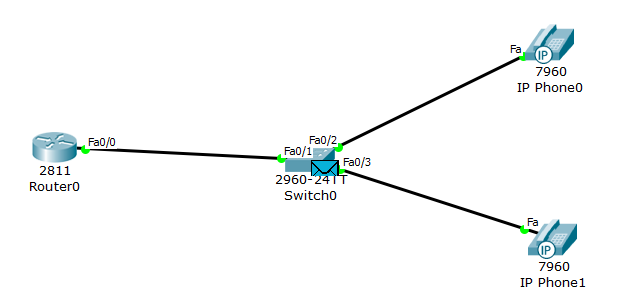


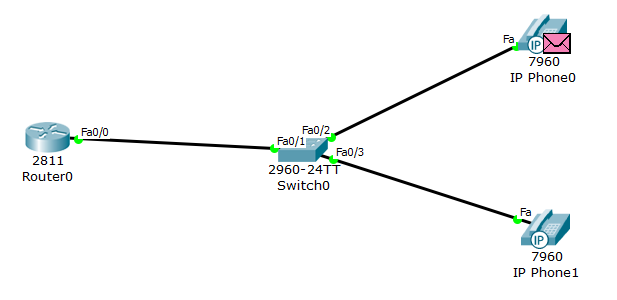




**Result:** Now verifying the connection by dialing and sending ICMP packets.





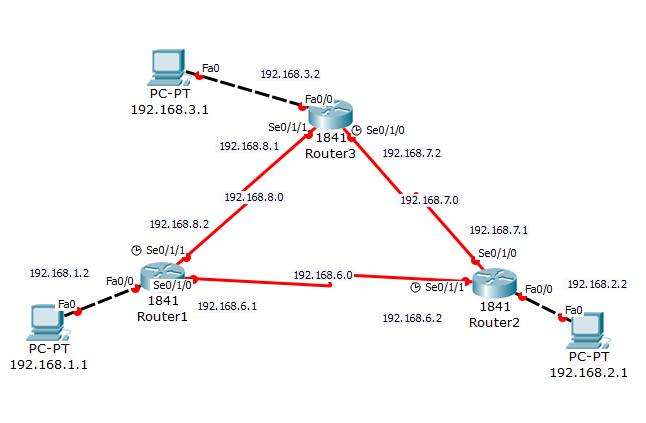


**Discussion:** In this experiment, we can observe telephone service using switch and router. Both the values of fast Ethernet ports and static routes has been set using command line interface instead of putting them manually for switch and router. Data can be successfully transferred from one IP Phone to another.

**Objective:** DTE (Data Terminal EQUIPMENT) and DCE (Data Communications EQUIPMENT) both are serial ports, where DCE, makes signal conversion, coding, and line clocking, i.e. controls the data rate like MODEM, and the other side (DTE) only receives or sends data at the clock rated provided by DCE like PC.Using this transmission packets find the shortest way from source to destination.

**Apparatus:** PC, simulator software (CISCO packet tracer).

**Network Diagram:**

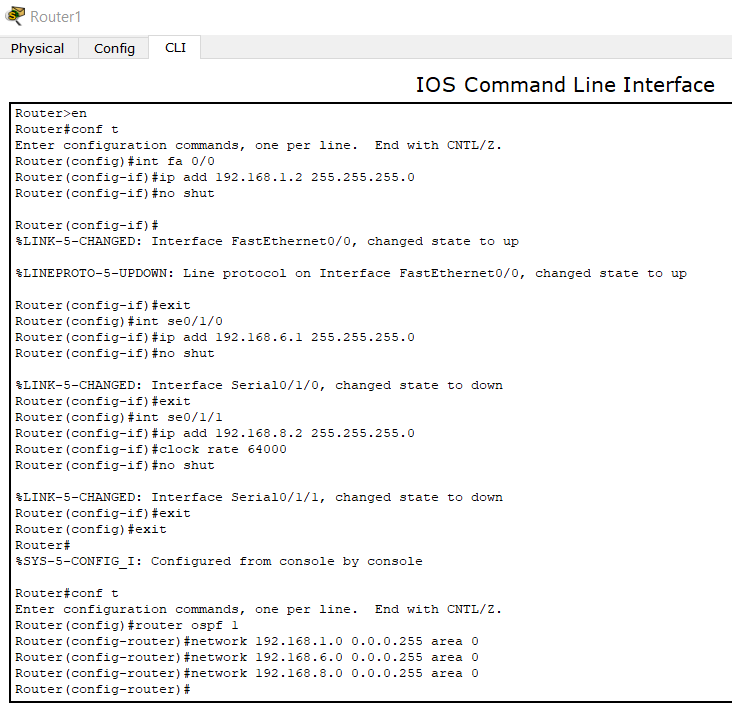
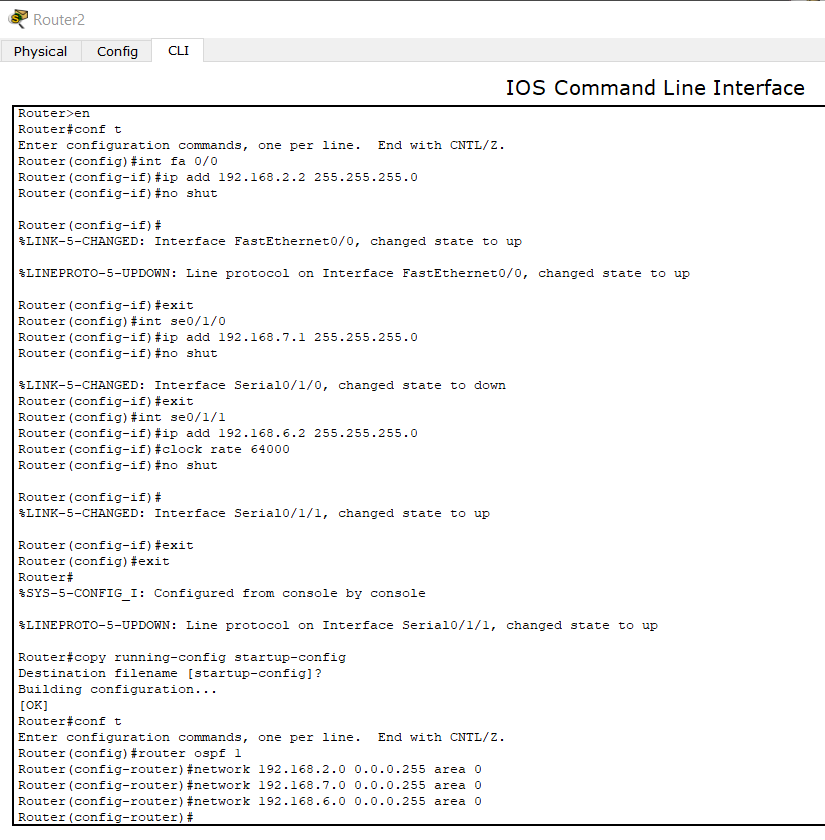


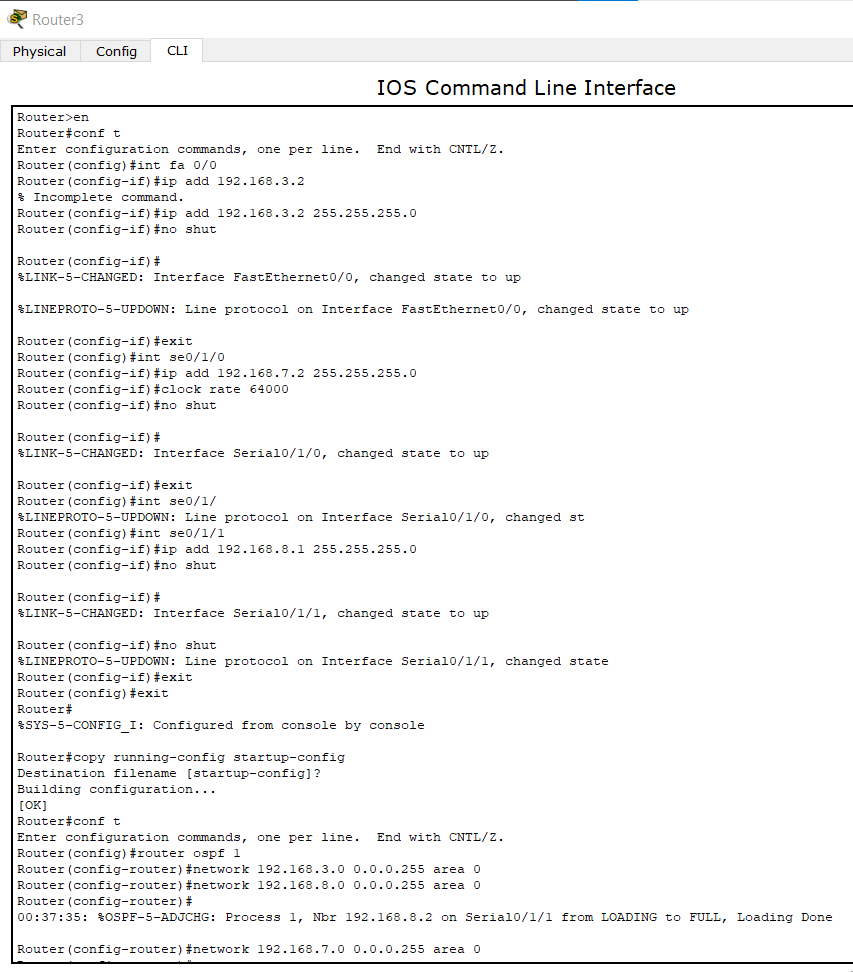
**Methodology:**

1. At first, 3 PCs and 3 1841 routers have been taken. Copper cross-over wire has been used to connect the PCs with routers. Routers have been configured with WIC-2T modules and connected using serial DCE wire.

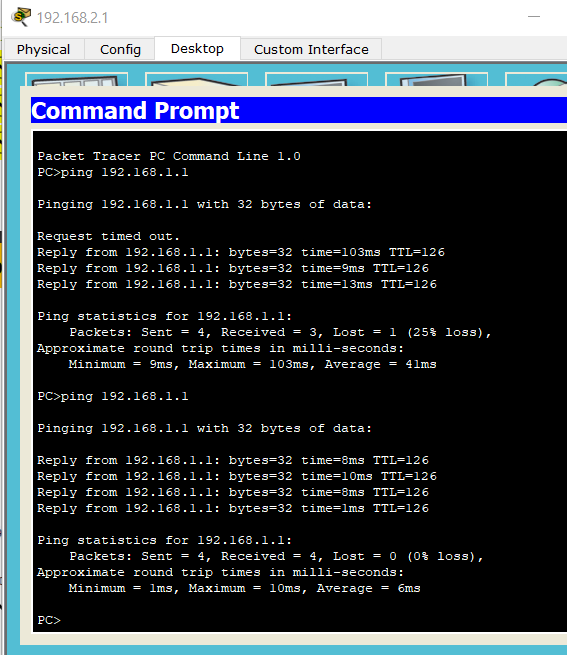


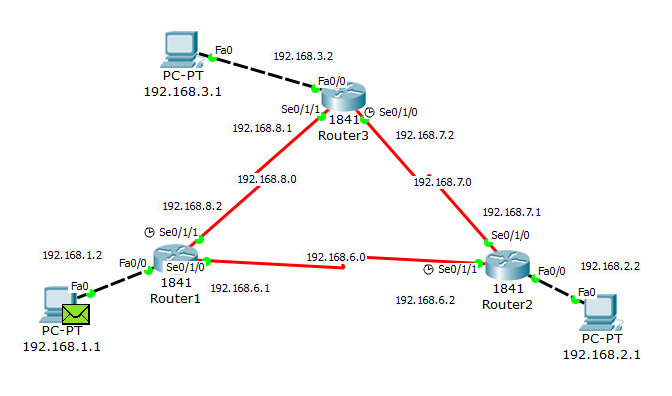
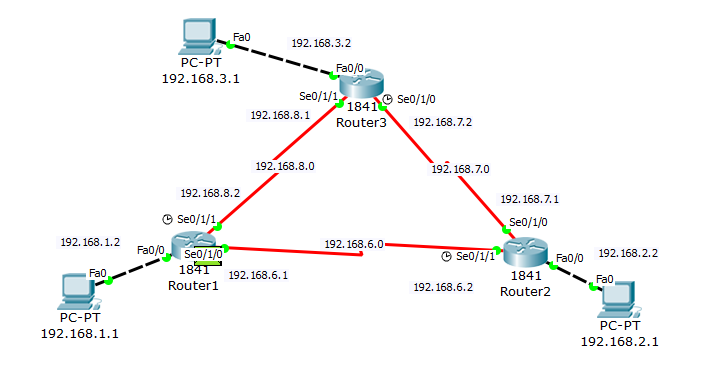
1. Next, all the PCs are provided with IPs and gateway and each of the router has been configured using CLI.

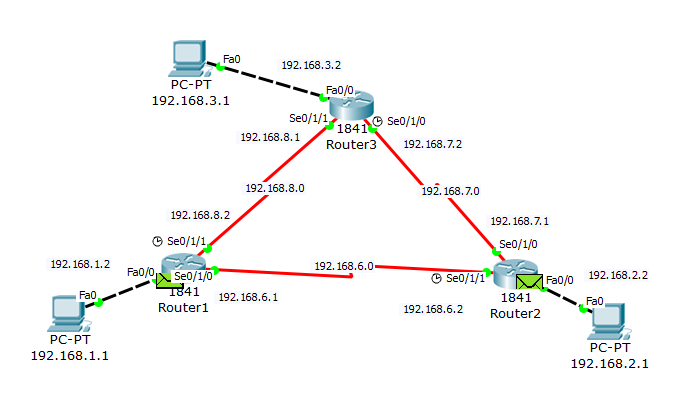
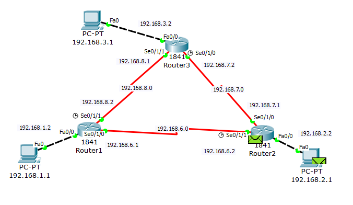




**Result:** The network has been verified using ping from IP 192.168.2.1 to IP 192.168.3.1. Also an ICMP packet has been sent from IP 192.168.1.1 to IP 192.168.2.1. The transmission is successful. At first there will be request timed out as the routers don’t know the way. But next time, the router recognizes its way to IP thus making the transmission successful.







**Discussion:** In this experiment, we can observe how OSPF algorithm works with routers. Using DCE with serial Ethernet, transmission packets find their shortest path to reach destination so that they don’t have to travel a long way through each of the alternative routers.

**Objective:** RSA algorithm is asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes that the Public Key is given to everyone and Private Key is kept private. A client sends its public key to the server and requests for some data. The server encrypts the data using client’s public key and sends the encrypted data. Client receives this data and decrypts it.

**Apparatus:** PC, programming platform (Mat Lab).

**Methodology:**

1. At first, RSA for text encryption and recovery are implemented.

**Code:**

e = 3;

n = 33;

d = 7; %RSA parameters

y='JAHANGIRNAGAR'; %input string

z = double(y); %ASCII values

S=z-60; %to reduce size of the integer

for i=1:length(z)

Encrypt(i)=mod(S(i)^e, n);

end

char(Encrypt) % encrypted string

Encrypt=double(Encrypt);

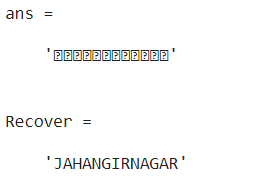
for j=1:length(z)

Decrypt(j)=mod(Encrypt(j)^d, n);

end

Recover=char(Decrypt+60)

**Output:**

****

1. RSA for image encryption and decryption for image has been implemented.

**Code:**

e = 3;

n = 33;

d = 7;

N=256;

I=imread('C:\Users\LAB 2\Desktop\snapshot2.png');

I=rgb2gray(I);

I=imresize(I,[N, N]);

subplot(2,2,1)

imshow(I)

title('Original Image')

I=double(I);

R=mod(I,16);

%Remainder of the image

for i=1:N

for j=1:N

Q(i,j)=uint8((I(i,j)/16)-0.5);

% Quiescent of the image

end

end

Q=double(Q);

for i =1:N

for j = 1:N

Qe(i, j)=mod(Q(i,j)^e, n);

Re(i, j)=mod(R(i,j)^e, n);

%Decryption of image

Qd(i, j)=mod((Qe(i,j))^d, n);

Rd(i, j)=mod((Re(i,j))^d, n);

end

end

Rec=Qd\*16+Rd;

subplot(2,2,2)

imshow(uint8(Qe)-200)

title('Encrypted Quiescent Image')

subplot(2,2,3)

imshow(uint8(Re)-200)

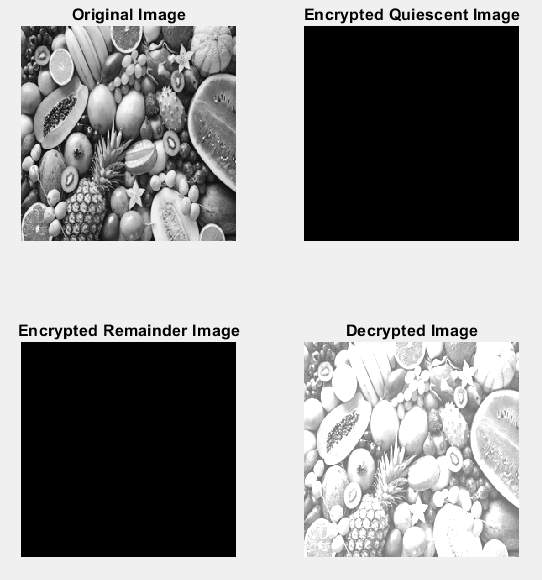
title('Encrypted Remainder Image')

subplot(2,2,4)

imshow(uint8(Rec)+100)

title('Decrypted Image')

**Output:**



1. Then RSA is applied for RGB image.

**Code:**

I=imread('F:\4-1\Computer Networks\Lab\fruits.jpg');

I1=I(:,:,1); %red plate

I2=I(:,:,2); %green plate

I3=I(:,:,3); %blue plate

subplot(2,2,1)

imshow(I1)

title('Red plate')

subplot(2,2,2)

imshow(I2)

title('Green plate')

subplot(2,2,3)

imshow(I3)

title('Blue plate')

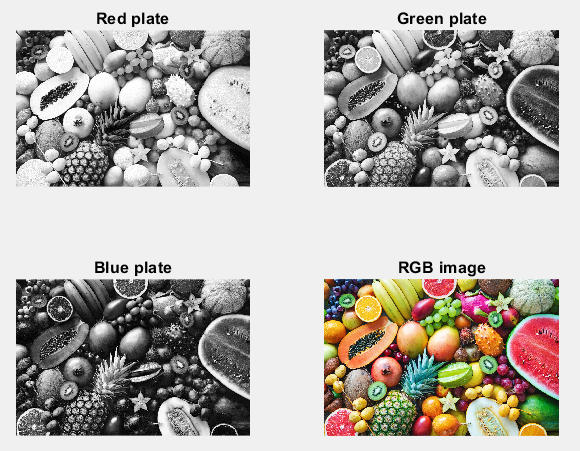
Y=cat(3,I1,I2,I3);

subplot(2,2,4)

imshow(Y)

title('RGB image')

**Output:**

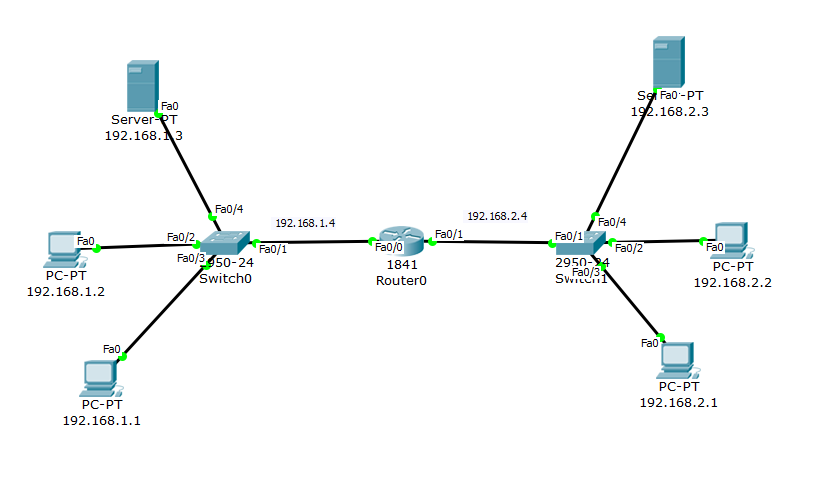


**Discussion:** In this experiment, we can observe how RSA algorithm works with image encryption and decryption. Using RSA colors of an image can also be categorized.

**Objective:** DNS servers translate requests for specific domains into IP addresses, controlling which server users with access when they enter the domain name into their browser.

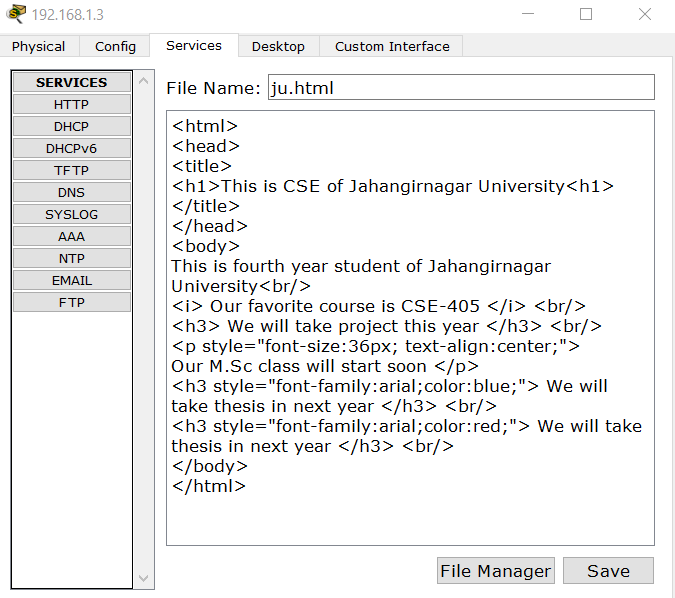
**Apparatus:** PC, simulator software (CISCO packet tracer).

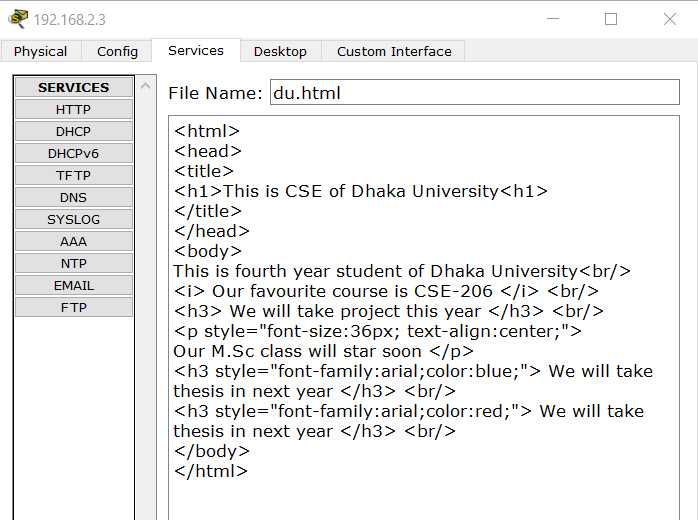
**Network Diagram:**



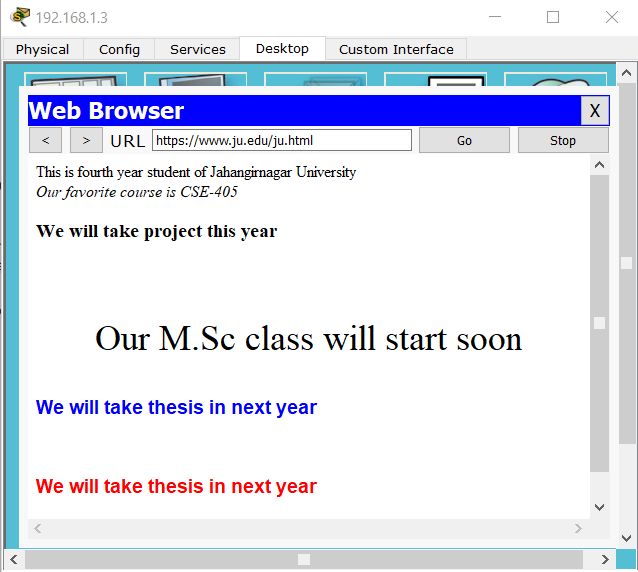
**Methodology:**

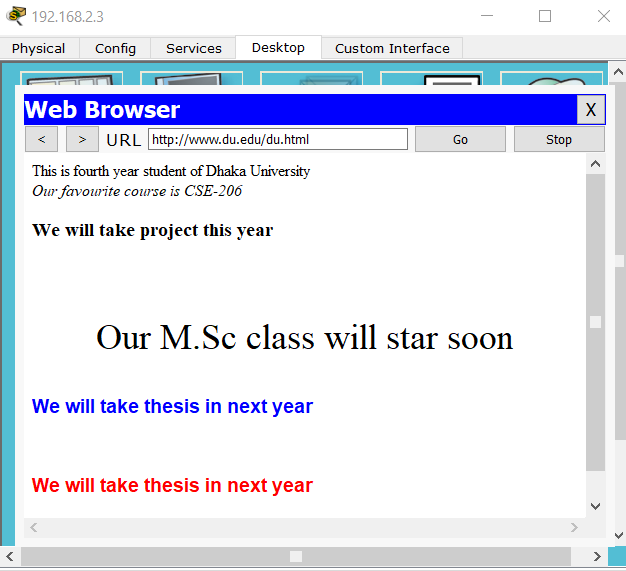
1. At first, 4 generic PCs, a 1841 router, 2 2950T switches and 2 servers have been taken. Copper straight-through wire has been used to connect them. PCs and servers are provided with their IPs, gateway along with DNS server.
2. Router has been provided with their MAC and IP address to Fast Ethernet ports. Also the servers have been loaded with HTMLs.

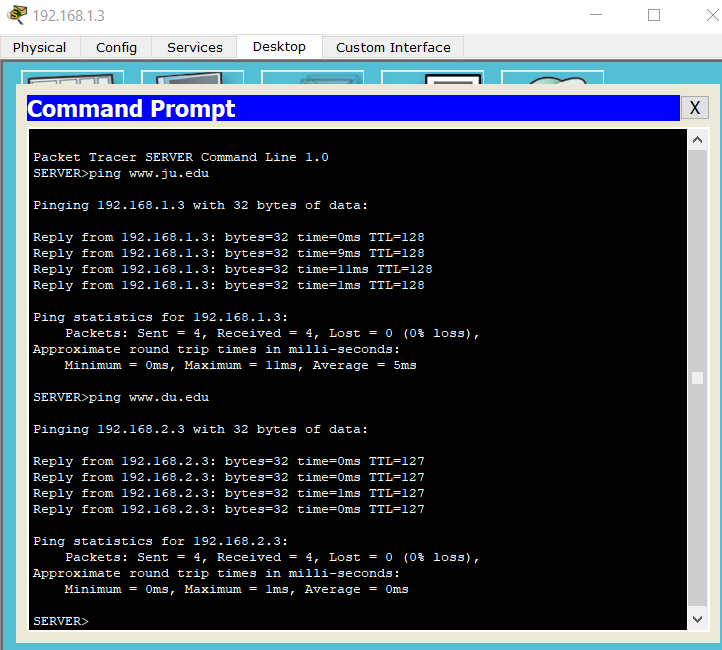




**Result:**

****

****

****

**Discussion:** In this experiment, we can observe how DNS server is configured. We have used simple HTML to create the webpages. We can add further database connection to the pages.

**Objective:** Sockets are commonly used for client and server interaction. Typical system configuration places the server on one machine, with the clients on other machines. The clients connect to the server, exchange information, and then disconnect.

**Apparatus:** PC, programming platform (Python IDLE).

**Methodology:**

1. At first, a socket object is created.

**Code:**

# An example script to connect to Google using socket

# programming in Python

import socket

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

print ("Socket successfully created")

# default port for socket

port=80

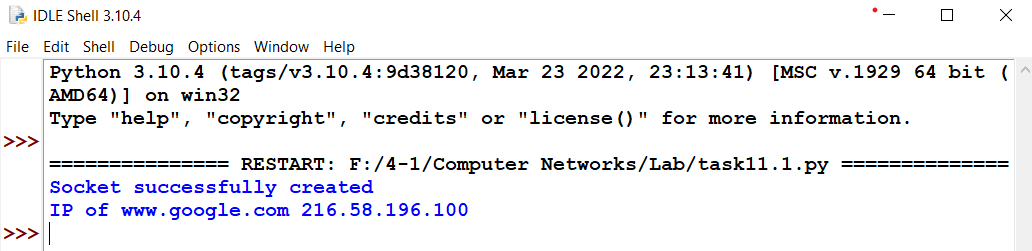
host\_ip = socket.gethostbyname('www.google.com')

# connecting to the server

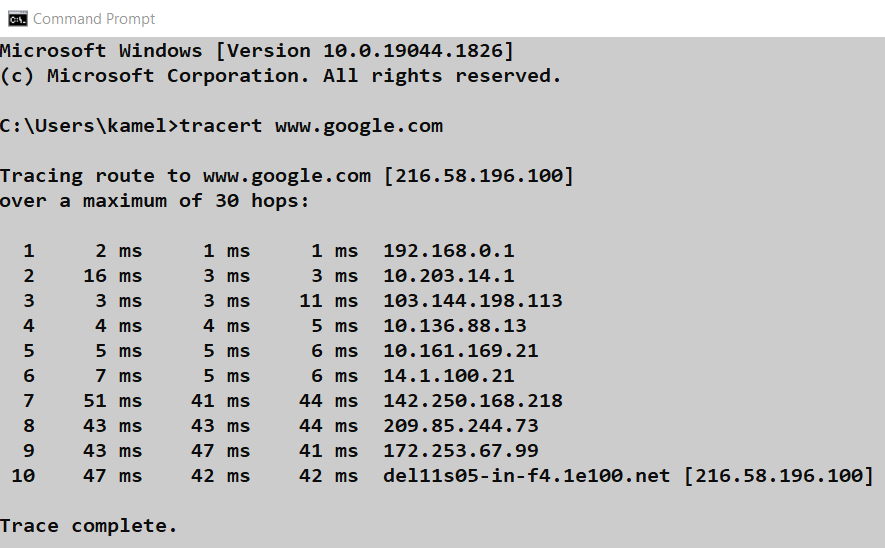
s.connect((host\_ip, port))

print ('IP of [www.google.com',host\_ip](http://www.google.com',host_ip))

**Output:**

****

1. Checking the same IP for tracert.



1. Now checking for another website.

**Code:**

# An example script to connect to Google using socket

# programming in Python

import socket

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

print ("Socket successfully created")

# default port for socket

port=80

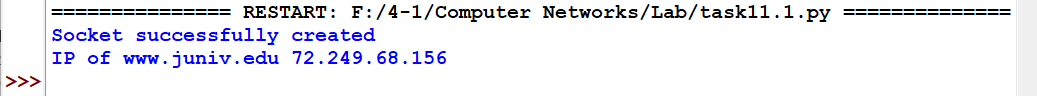
host\_ip = socket.gethostbyname('www.juniv.edu')

# connecting to the server

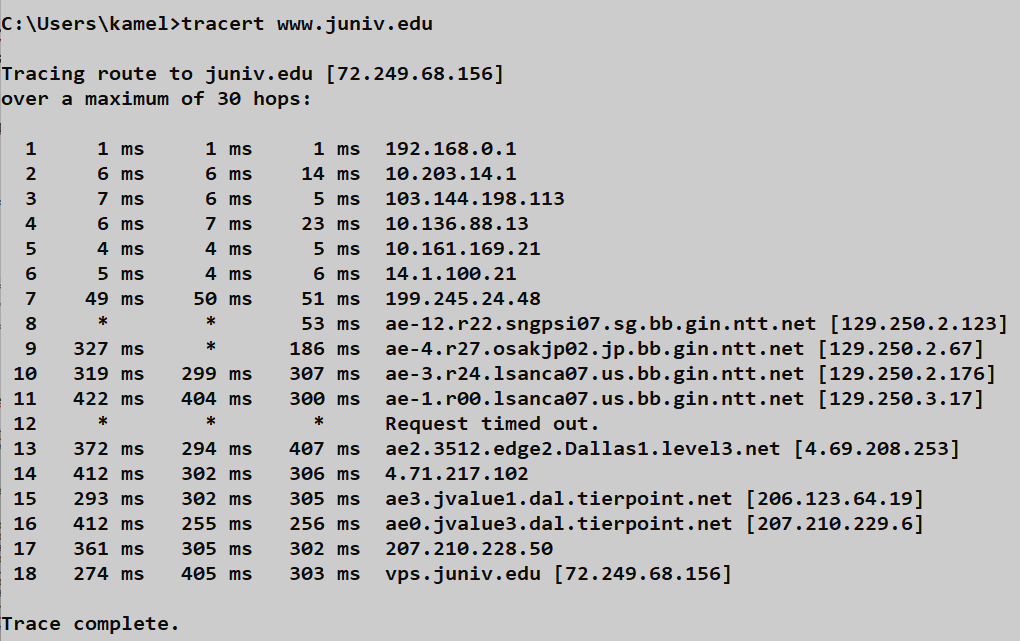
s.connect((host\_ip, port))

print ('IP of [www.juniv.edu',host\_ip](http://www.juniv.edu',host_ip))

**Output:**



**Tracert:**



1. Now establishing client-server communication. For this server side connection has been made first.

**Code:**

import socket

LOCALHOST = "127.0.0.1"

PORT = 8080

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind((LOCALHOST, PORT))

server.listen(1)

print("Server started")

print("Waiting for client request..")

clientConnection, clientAddress = server.accept()

print("Connected client :" , clientAddress)

msg = ''

while True:

in\_data = clientConnection.recv(1024)

msg = in\_data.decode()

if msg=='bye':

break

print("From Client :" , msg)

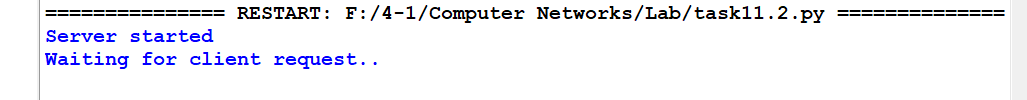
out\_data = input()

clientConnection.send(bytes(out\_data,'UTF-8'))

print("Client disconnected....")

clientConnection.close()

**Output:**



1. Then client side connection has been made and messages can be passed within server and client.

**Code:**

import socket

SERVER = "127.0.0.1"

PORT = 8080

client = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

client.connect((SERVER, PORT))

client.sendall(bytes("This is from Client",'UTF-8'))

while True:

in\_data = client.recv(1024)

print("From Server :" ,in\_data.decode())

out\_data = input()

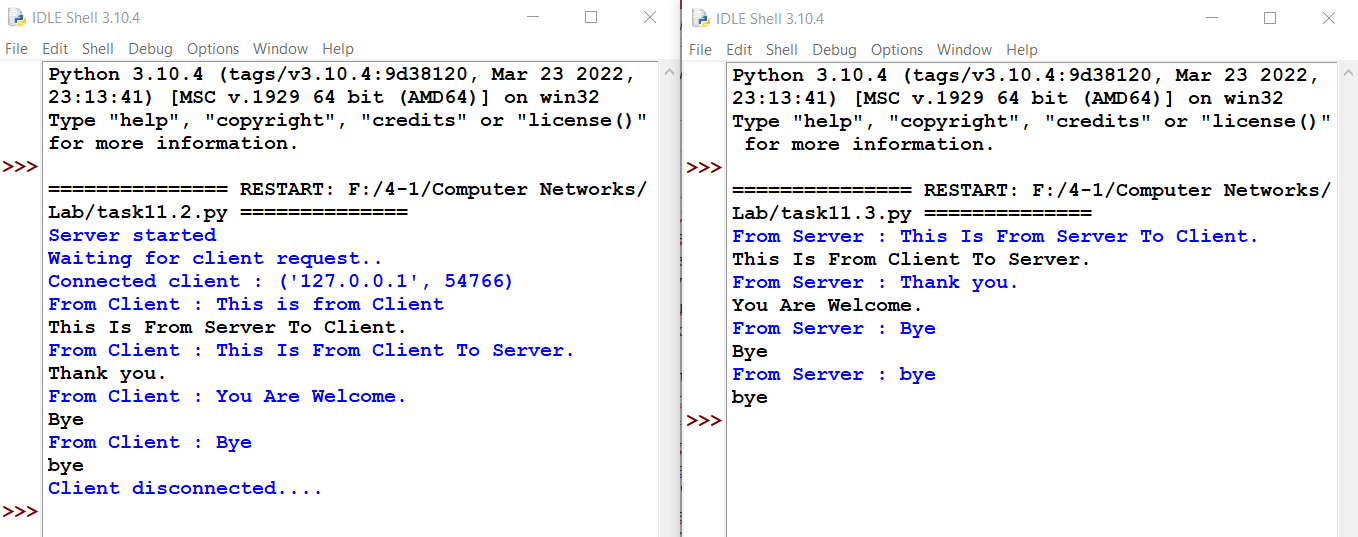
client.sendall(bytes(out\_data,'UTF-8'))

if out\_data=='bye':

break

client.close()

**Output:**

****

**Discussion:** Socket programming is a means of communicating data between two computers across a network. Connections can be made using either a connection-oriented protocol or a connectionless protocol. In our case, we use TCP/IP which is a connection-oriented protocol.