

EXPERIMENT-05

QUESTION: Configure The Telnet protocol using Cisco packet Tracer.

Step1] Draw a topology as shown and assign the IP addresses to all PC's.

Step2] Configure IP address to router.

```
router(config)# interface fastEthernet 0/0
```

```
router(config-if)# ip address 10.0.0.1 255.0.0.0
```

```
router(config-if)# no shutdown
```

```
router(config-if)# exit
```

Step3] To set privilege mode password.

Click on router and go to CLI tab & type below.

```
router(config)# enable password 1234
```

```
router(config)# exit
```

Step4] To configure Telnet

```
router# conf t
```

Enter configuration commands one per line. End with ctrl+z

```
router(config)# line vty 0 4
```

```
router(config)# password cisco
```

```
router(config-line)# login
```

```
router(config-line)# exit
```

Step 5] To check Telnet configuration

router # sh run.

Step 6] To access cisco router via telnet connection from any PC.

click on any PC

click on desktop

Select command prompt & type the following commands

PC > ping 10.0.0.1

PC > telnet 10.0.0.1

EXPERIMENT-06

QUESTION: Configure the static routing using Cisco Packet Tracer.

- Step1] Draw the topology as shown below and assign IP addresses to all PC's.
- Step2] Configure IP address to router 1.
For Fast Ethernet 0/0, and
For Fast Ethernet 0/1.
- Step3] Configure IP address to router2
For Fast Ethernet 0/0, and
For Fast Ethernet 0/1.
- Step4] To set up static routing
For Router 1:-
In CLI
router(config)#
router(config)# ip route 192.168.2.0 255.255.255.0 192.168.3.2
In config window:
click on static
Then add opposite network address 192.168.2.0 and
next hop address 192.168.3.1 along with subnet
mask address 255.255.255.0

For router 2:

In CLI:-

Router (config) #

Router (config) # ip route 192.168.1.0 255.255.255.0 192.168.3.1

In config window:

click on static . Add opposite Network address 192.168.3.0

of next hop address 192.168.3.1 along with subnet mask address 255.255.255.0 .

Steps] To check connectivity between the networks using static routing.

click on any PC > desktop > select command prompt
type the following.

PC > ping 192.168.2.1.

EXPERIMENT-07

Configure the RIP routing using
Cisco Packet Tracer.

- Step 1) Draw a topology as shown below and assign IP address to all PC's.
- Step 2) Configure IP address to router 1.
For Fast Ethernet 0/0, and
For Fast Ethernet 0/1
- Step 3) Configure IP address for router 2
For Fast Ethernet 0/0 and
for Fast Ethernet 0/1.
- Step 4) To set up Dynamic routing.
For Router 1:
In CLI:

```
router(config)# router rip
Router(config)# network 192.168.3.0
Router(config-router)# network 192.168.2.0.
```

 In config window:
 click on RIP
 Then add opposite Network address 192.168.2.0 and 192.168.3.0.

For router 2:

In CLI :-

Router (config) # Router rip

Router (config-router) # network 192.168.3.0

Router (config-router) # network 192.168.1.0

In config window:

click on RIP

Then add opposite networks 192.168.1.0 & 192.168.3.0

Steps] To check connectivity between two networks using RIP routing

click on any PC > Desktop > Select command prompt and type below commands.

PC > ping 192.168.2.1 .

EXPERIMENT-08

QUESTION : Configure the static NAT using Cisco Packet Tracer.

- NAT : It is a process in which one or more local IP address is translated into global IP or vice versa. It allows multiple devices to access internet through single public IP address.

Step 1] Draw a topology as shown below & assign IP addresses to all PC's.

Step 2] Configure IP address to router 1
 For Fast Ethernet 0/0
 For Serial 2/0

Step 3] Configure IP address to router 2
 For serial 2/0.

Step 4] To setup static NAT

Router# sh ip nat translation

Router# config t

Router(config)# ip nat inside source static 10.0.0.2 192.168.1.3

Provide interface for NAT cable

Router(config)# int fa0/0

Router(config-if)# ip nat inside
 exit

Router(config)# int serial 2/0

Router(config-if)# ip nat outside
exit

exit
Router# sh ip nat translation

Steps] To check connectivity between two network

Click on any PC > Desktop > Select command prompt

PC > ping 192.168.2.1.

EXPERIMENT-09

QUESTION: Configure Dynamic NAT using Cisco Packet Tracer.

- NAT : It is a process in which one or more local IP addresses is translated into global IP or vice versa.

It allows multiple devices to access internet through single IP address.

Step1] Draw a topology as shown below and assign IP addresses to all PC's.

Step2] Configure IP address for router 1.

For Fast Ethernet 0/0.

For Serial 2/0.

Step3] Configure IP address for router 2.

For Serial 2/0.

Step4] To setup Dynamic NAT

```
router(config)# access-list 1 permit 10.0.0.2 0.0.0.0
```

```
router(config)# access-list 1 permit 10.0.0.3 0.0.0.0
```

```
router(config)# ip nat pool nslab 192.168.2.3 192.168.2.4  
netmask 255.255.255.0
```

```
router(config)# ip nat inside source list 1 pool nslab
```

```
router(config)# int fa0/0  
router(config-if)# ip nat inside  
router(config-if)# exit  
router(config)# int serial 2/0  
router(config-if) ip nat outside  
router# sh ip nat translation
```

Steps) To check connectivity between 2 networks
click on any PC > click on desktop > command prompt
& type
PC > ping 192.168.2.1.

MODULE-3

SOCKET PROGRAMMING

EXPERIMENT-1

- QUESTION: To find website address using Socket program.

- CODE:

```

import java.net.*;
import java.util.*;
public class IPfindee {
    public static void main(String[] args) {
        String host;
        Scanner inp = new Scanner(System.in);
        System.out.print("\nEnter Host name:");
        host = input.nextLine();
        try {
            InetAddress add = InetAddress.getByName(host);
            System.out.println("IP : " + add.toString());
        } catch (UnknownHostException whEx) {
            System.out.println("Couldn't find :" + host);
        }
    }
}

```

EXPERIMENT-02

- QUESTION : To find local host IP address using Socket Programming.
- CODE :

```
import java.util.*;
import java.net.*;
public class LocalIP
{
    public static void main (String [] args)
    {
        try {
            // InetAddress add = InetAddress.getByName
            InetAddress add = InetAddress.getLocalHost();
            System.out.println (add);
        }
        catch (UnknownHostException uhEx)
        {
            System.out.println ("Couldn't find local host");
        }
    }
}
```

EXPERIMENT-03

— QUESTION: Write a program to communicate between client & server using UDP protocol.

— CODE:

- Client side:

```

import java.net.*;
import java.io.*;

class client
{
    public static DatagramSocket ds;
    public static byte buffer[] = new byte[1024];
    public static int clientport = 1789, serverport = 1790;
    public static void main(String args[]) throws Exception
    {
        byte buffer[] = new byte[1024];
        ds = new DatagramSocket(clientport);
        BufferedReader bread = new BufferedReader(new InputStreamReader(
            System.in));
        System.out.print("Address");
        String msg = bread.readLine();
        buffer = msg.getBytes();
        ds.send(new DatagramPacket(buffer, msg.length(),
            InetAddress.getLocalHost(), serverport));
        System.out.println("Client is waiting for your data");
        System.out.print("Press ctrl+c to come out");
    }
}

```

```

while(true)
{
    DatagramPacket dp = new DatagramPacket(buffer, buffer.length);
    ds.receive(dp);
    String pdata = new String(dp.getData(), 0, dp.getLength());
    if(pdata.equals("End"))
        break;
    System.out.println(pdata);
    String str = reader.readLine();
    buffer = str.getBytes();
    ds.send(new DatagramPacket(buffer, str.length(),
        InetAddress.getLocalHost, serverport));
}
}

```

→ Server Side:

```

import java.net.*;
import java.io.*;
class Server
{
    public static DatagramSocket ds;
    public static int clientport = 1789, serverport = 1790;
    public static void main(String[] args) throws Exception
    {
        byte buffer = new byte[1024];
        ds = new DatagramSocket(serverport);
        BufferedReader reader = new BufferedReader(
            new InputStreamReader(System.in)));

```

```

System.out.println("Waiting for connection");
DatagramPacket dp = new DatagramPacket(buffer, buffer.length);
ds.receive(dp);
String pdate = new String(dp.getData(), 0, dp.getLength());
System.out.println("Connected");
String st = "Hello";
buffer = st.getBytes();
ds.send(new DatagramPacket(buffer, st.length(), clientport));
break;
while(true)
{
    ds.receive(dp);
    String recv = new String(dp.getData(), 0, dp.getLength());
    System.out.println(recv);
    buffer = str.getBytes();
    if(str == null || str.equals("End"))
    {
        ds.send(new DatagramPacket(buffer, str.length(),
        InetAddress.getLocalHost(), clientport));
        break;
    }
    ds.send(new DatagramPacket(buffer, str.length(),
    InetAddress.getLocalHost(), clientport));
}
}
}

```

EXPERIMENT-04

- QUESTION:- Write a program to communicate between client & sever using TCP protocol.

- CODE:

→ Client side:

```

import java.net.*;
import java.io.*;

public class tcpclient
{
    public static void main(String [] args) throws IOException
    {
        Socket s = new Socket("localhost", 55);
        DataInputStream in = new DataInputStream(s.getInputStream());
        DataOutputStream out = new DataOutputStream(s.getOutputStream());
        DataInputStream sysin = new DataInputStream(System.in);
        while (true)
        {
            String str = in.readLine();
            System.out.println("Message from Server" + str);
            if (str.equals("end"))
                break;
            System.out.print("Enter reply: ");
            String line = sysin.readLine();
            out.writeBytes(line + "\n");
        }
        s.close();
    }
}

```

→ Server Side:

```

import java.io.*;
import java.net.*;

class TCPServer
{
    public static void main (String [] args) throws IOException
    {
        ServerSocket ss = new ServerSocket(55);
        Socket s = ss.accept();
        System.out.println("Connected");
        DataInputStream in = new DataInputStream(s.getInputStream());
        DataOutputStream out = new DataOutputStream(s.getOutputStream());
        DataInputStream sysin = new DataInputStream (System.in);
        while(true)
        {
            System.out.print("Enter a string:");
            String str = sysin.readLine();
            out.writeBytes(str + "\n");
            if(str.equals("end"))
                break;
            System.out.println("Message :" + in.readLine());
        }
        ss.close();
    }
}

```

EXPERIMENT-05

QUESTION: Program to check connectivity of given Hostname.

CODE:

```

import java.net.*;
import java.io.*;
public class ping{
    public static void main(String [] args) throws IOException {
        String host = "";
        Scanner inp = new Scanner(System.in);
        System.out.println("Enter host name:");
        host = inp.nextLine();
        try {
            InetAddress add = InetAddress.getByName(host);
            System.out.println("IP : " + add.toString());
            System.out.println("Sending ping request to" + host);
            if (add.isReachable(5000))
                System.out.println(host + " is reachable");
            else
                System.out.println(host + " is not reachable");
        }
        catch(UnknownHostException uhEx)
        {
            System.out.println("Couldn't find" + host);
        }
    }
}

```