PRACTICALS

For linux experiments : do only student login , not guest login

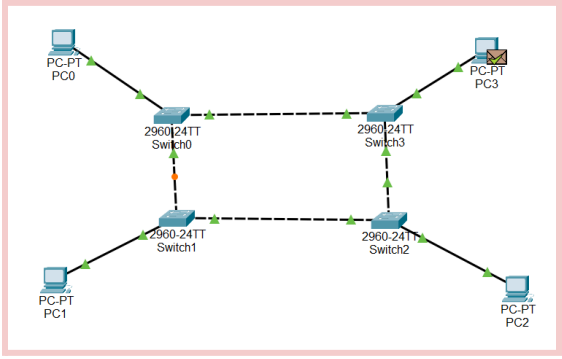
1. Basic linux commands

- **Ifconfig** : ifconfig
- **Ping** : first do ifconfig and then take the ip address from there and perform- ping 192.168.20.13 (dont type these numbers type the ip address written in the screen)
- **Tracepath :** tracepath.google.com
- **Traceroute :** traceroute google.com , traceroute rait.ac.in (any one of the two)
- **Host :** www.google.com
- **Nslookup :** nslookup google.com nslookup rait.ac.in (any one of the two)
- **Netstat :** netstat
- **IP :** ip
- **Dig** : dig google.com, dig rait.ac.in (any one of the two)

2. Network topology: cisco packet

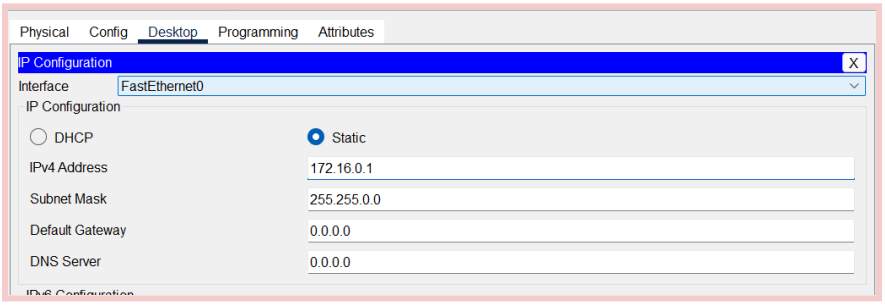
Ring topology:

- 4 PCs , 4 2960 switches
- Connection by automatic wire (it has a thunderbolt icon)

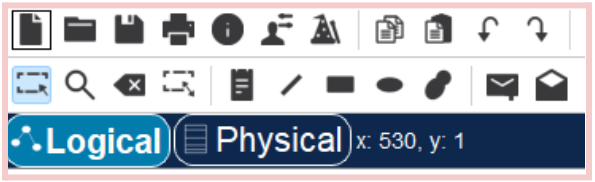


- Connect each PCs to one switch and connect all switches among themselves.
- Wait for arrows to turn green.

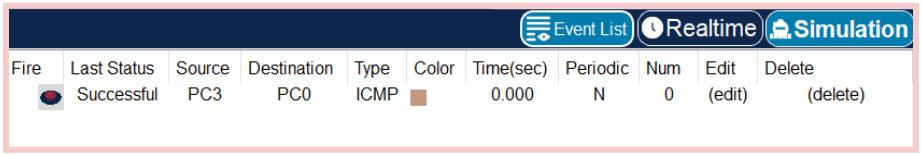
- Click on PC, select Desktop, Choose IP configuration, enter a random IP Address(172.16.0.1)in ipv4 Address section and the subnet mask gets automatically entered. Repeat the same for all 4PCs.



- Click on simulation in the right bottom corner. Click on show All None. click on edit filters and choose ICMP.
- Select the message icon(2nd row,2nd from right) on the left top and click on PC3 and PC0.

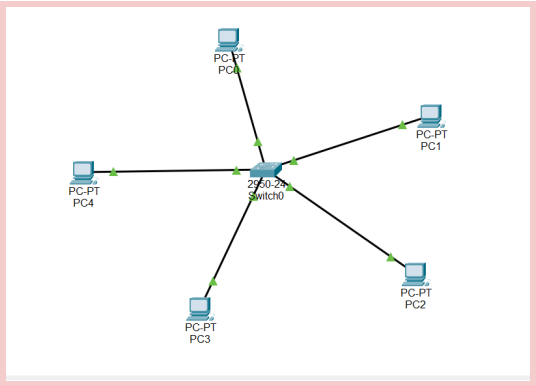


- click on the play button on the simulation window. Wait until the message is passed and shows successful status at right bottom



Star Topology:

- 5 PCs and 1 2950-24 switch



- Connect all PCs to switch using same wire option as above
- Wait for arrows to turn green
- Follow all the remaining steps same as above

3. Hamming code : c code

```
#include<stdio.h>

void main() {
   int data[10];
   int dataatrec[10],c,c1,c2,c3,i;

   printf("Enter 4 bits of data one by one\n");
   scanf("%d",&data[0]);
   scanf("%d",&data[1]);
   scanf("%d",&data[2]);
   scanf("%d",&data[4]);

   //Calculation of even parity
   data[6]=data[0]^data[2]^data[4];
data[5]=data[0]^data[1]^data[4];
data[3]=data[0]^data[1]^data[2];

printf("\nEncoded data is\n");
for(i=0;i<7;i++)
      printf("%d ",data[i]);

   printf("\n\nEnter received data bits one by one\n");
   for(i=0;i<7;i++)
      scanf("%d",&dataatrec[i]);

   c1=dataatrec[6]^dataatrec[4]^dataatrec[2]^dataatrec[0];
```

```
c2=dataatrec[5]^dataatrec[4]^dataatrec[1]^dataatrec[0];
c3=dataatrec[3]^dataatrec[2]^dataatrec[1]^dataatrec[0];
c=c3*4+c2*2+c1 ;

   if(c==0) {
printf("\nNo error while transmission of data\n");
   }
else {
printf("\nError on position %d",c);

printf("\nData sent : ");
       for(i=0;i<7;i++)
        printf("%d",data[i]);

printf("\nData received : ");
       for(i=0;i<7;i++)
        printf("%d",dataatrec[i]);
printf("\nCorrect message is\n");

//if errorneous bit is 0 we complement it else vice versa
if(dataatrec[7-c]==0)
dataatrec[7-c]=1;
       else
dataatrec[7-c]=0;
for (i=0;i<7;i++) {
printf("%d",dataatrec[i]);
}
 ,}
}
```

4. Packet filtering by IP forwarding: linux

- Add Routes

  sudo route add -net 162.50.1.1 gw 192.168.52.1 netmask 255.255.255.255
  password
  netstat -rn

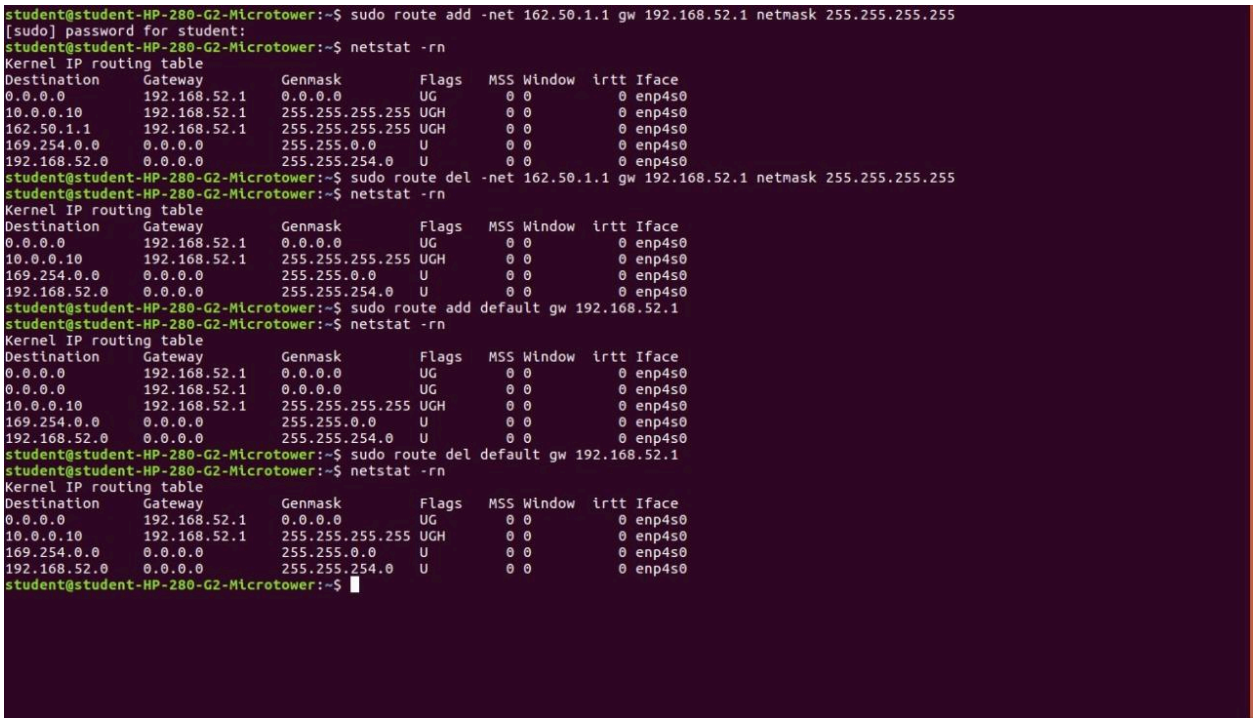
- Delete Routes

  sudo route del -net 162.50.1.1 gw 192.168.52.1 netmask 255.255.255.255
  netstat -rn

- Change default gateway

  sudo route add default gw 192.168.52.1
  netstat -rn

- Delete gateway added
  sudo route del default gw 192.168.52.1
  netstat -rn



5. Socket programming (exp 8) : linux

- Make 2 different file : one for server side and one for client side

Server side
sudo nano MyServer.java **(IN LINUX)**

```
import java.io.*; import java.net.*;
public class MyServer
public static void main(String[] args){
try { ServerSocket ss=new ServerSocket(6666); Socket s=ss.accept();//establishes connection
DataInputStream dis=new DataInputStream(s.getInputStream()); String
str=(String)dis.readUTF(); System.out.println("message= "+str); ss.close();
}
catch(Exception e){System.out.println(e);
}
```

```
}
}
```

After copy paste, press cltr+x , y , enter (this will save your file)

Client side server
sudo nano MyClient.java **(IN LINUX)**

```
import java.io.*; import java.net.*;
public class MyClient {
public static void main(String[] args) {
try { Socket s=new Socket("localhost",6666); DataOutputStream dout=new
DataOutputStream(s.getOutputStream()); dout.writeUTF("Hello Server"); dout.flush();
dout.close(); s.close(); }
catch(Exception e){System.out.println(e);
}
}
}
```

After copy paste, press cltr+x , y , enter (this will save your file)

To run the file in Linux:
If **javac not found** do this:
sudo apt install default-jdk

javac MyServer.java
java MyServer

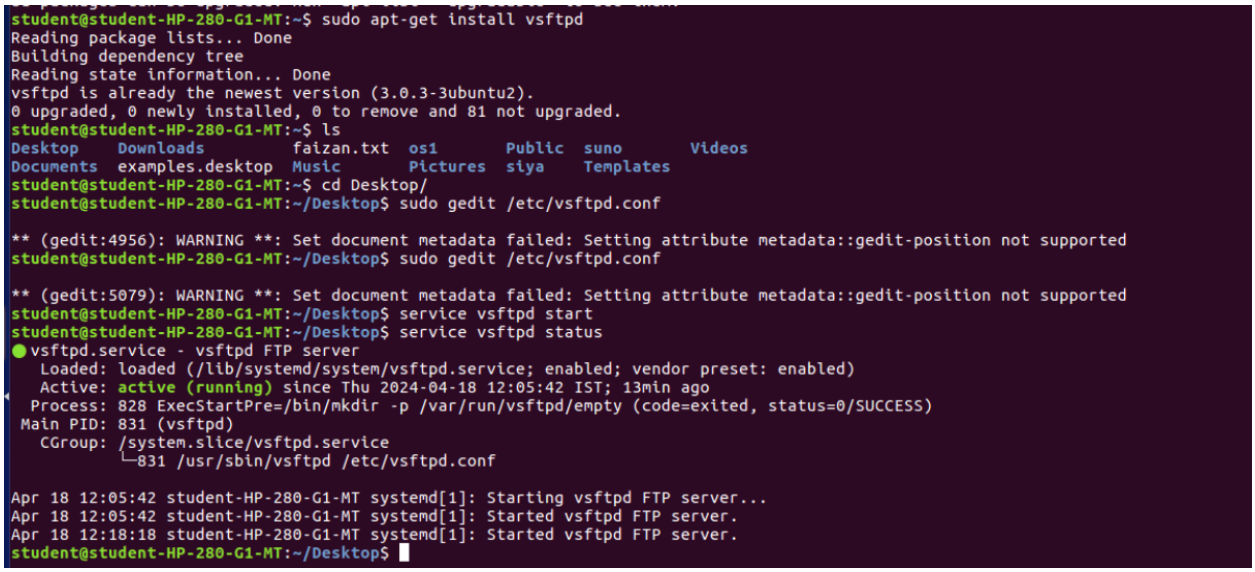
OPEN NEW TERMINAL KEEP OLD ONE OPEN

javac MyClient.java
java MyClient
After this you should find the message on MyServer terminal

6. Perform file transfer - linux

- sudo apt-get install vsftpd
- sudo nano /etc/vsftpd.conf
- sudo service vsftpd start
- service vsftpd status
- service vsftpd stop
- service vsftpd status

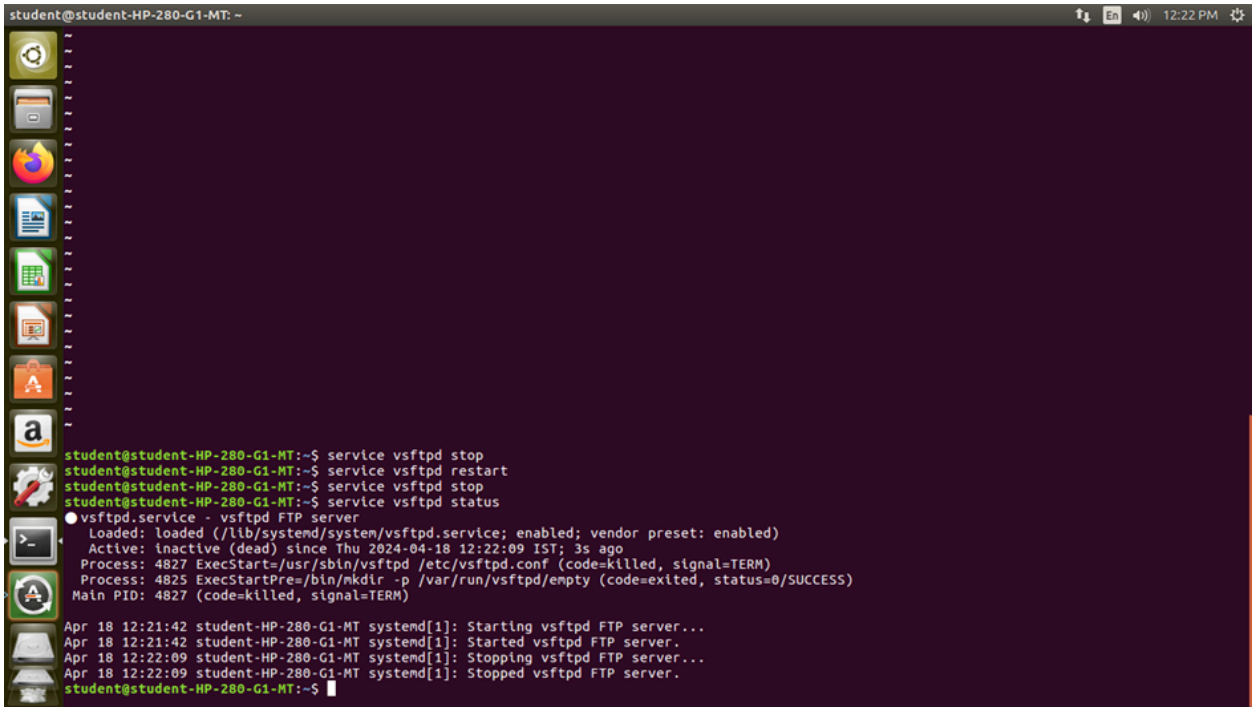
- sudo service vsftpd start
- Ifconfig
- ftp <your **IP-address>**
- <username ubuntu ka>
- <password>
- ls
- Lcd
- get <your local directory that lcd gives>
- put <your local directory that lcd gives>
- pwd
- put /home/student/Desktop/FTP logged in.png
- sudo get /home/student/Desktop/FTP logged in.png
- quit

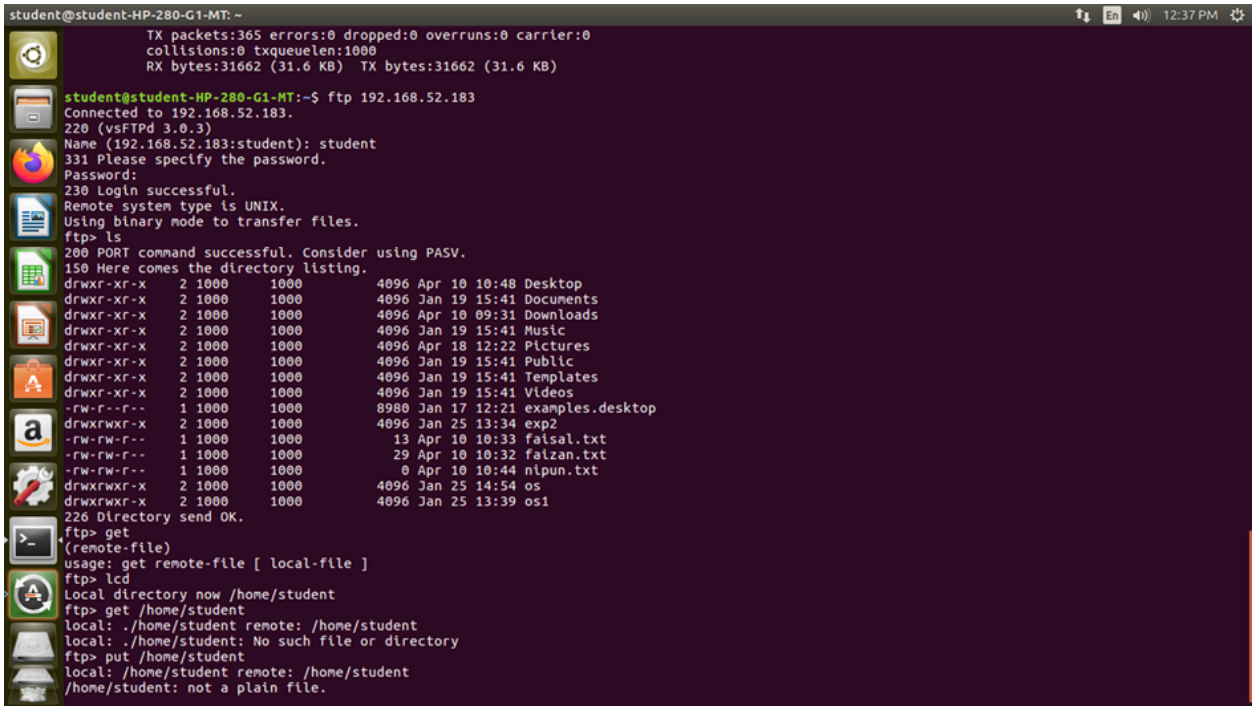


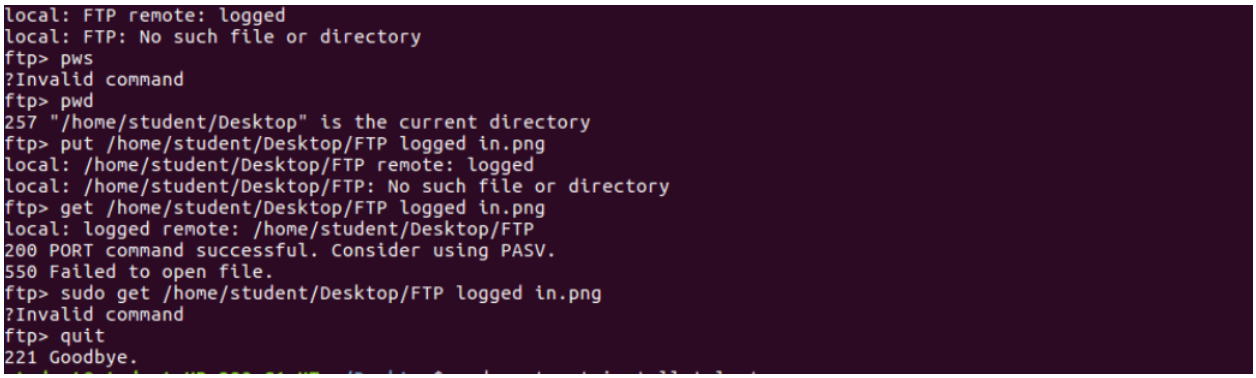
```
student@student-HP-280-G1-MT:~/Desktop$ ifconfig
enp3s0    Link encap:Ethernet  HWaddr 8c:dc:d4:45:60:b2
          inet addr:192.168.52.54  Bcast:192.168.53.255  Mask:255.255.254.0
          inet6 addr: fe80::995:2a6:a37e:43b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46924 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10198 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:27417194 (27.4 MB)  TX bytes:1889677 (1.8 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:1336 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1336 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140516 (140.5 KB)  TX bytes:140516 (140.5 KB)

student@student-HP-280-G1-MT:~/Desktop$ ^C
student@student-HP-280-G1-MT:~/Desktop$ service vsftpd restart
student@student-HP-280-G1-MT:~/Desktop$ ftp 192.168.52.54
Connected to 192.168.52.54.
220 (vsFTPd 3.0.3)
Name (192.168.52.54:student): student
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

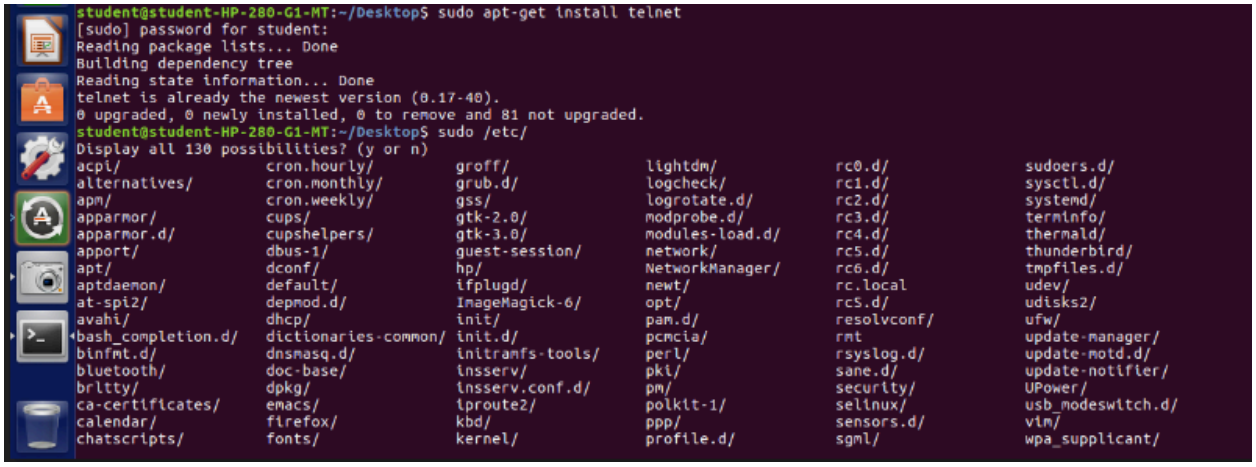


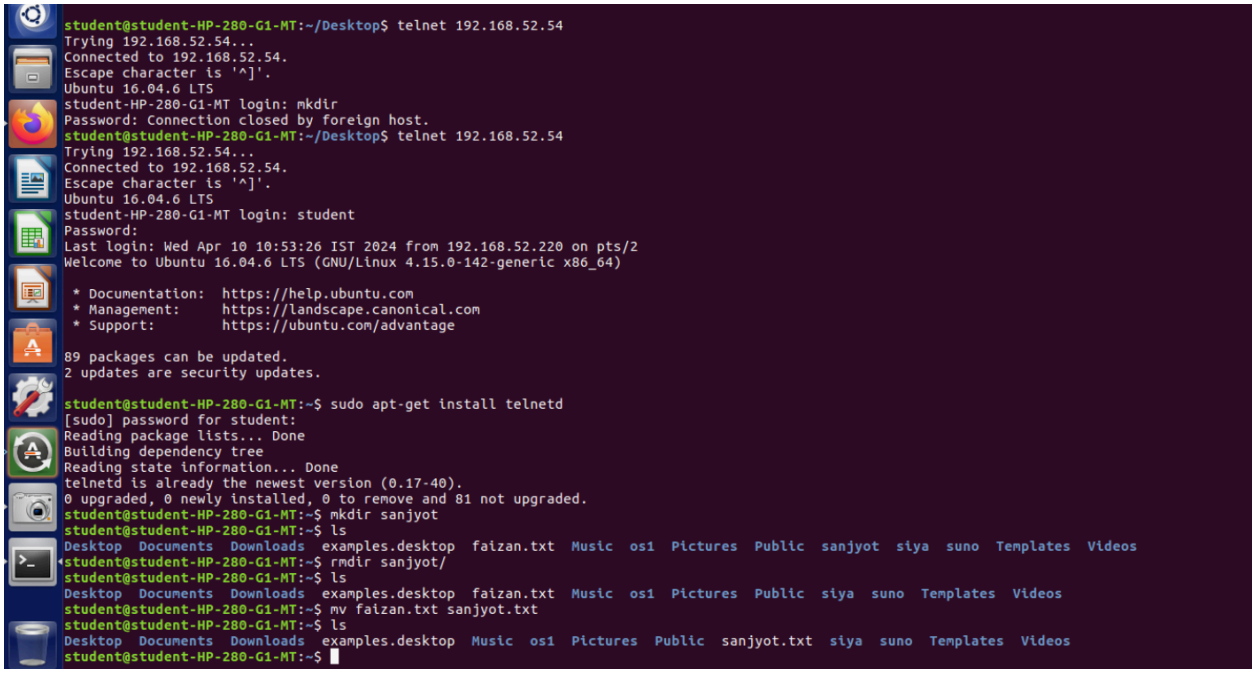




7.Perform remote login - linux

- sudo apt-get install telnet
- <password>
- sudo /etc/
- Ifconfig
- telnet <your ip address>
- sudo apt-get install telnetd
- mkdir <directory name>
- ls
- rmdir <directory name>
- ls
- mv <any .txt file> <any .txt file>
- ls





If you get error that states **telnet not found** do this:

- sudo nano /etc/xinetd.d/telnet
    - Enter the following code in the file:
      service telnet { disable = no flags = REUSE socket_type = stream wait = no user = root server = /usr/sbin/in.telnetd log_on_failure += USERID }
- sudo /etc/init.d/xinetd restart
- telnet localhost

8. Setup network - cisco packet

This is too tough to do and nobody knows how to complete it in 1hr (takes 3hrs)
God forbid that you should not get this experiment

If you get , leave lab and run for your life
Get minimum marks or get your experiment changed
Good luck, love for you <3<3