Computer Networks Assignment 3

SCS2205- Computer Network 1

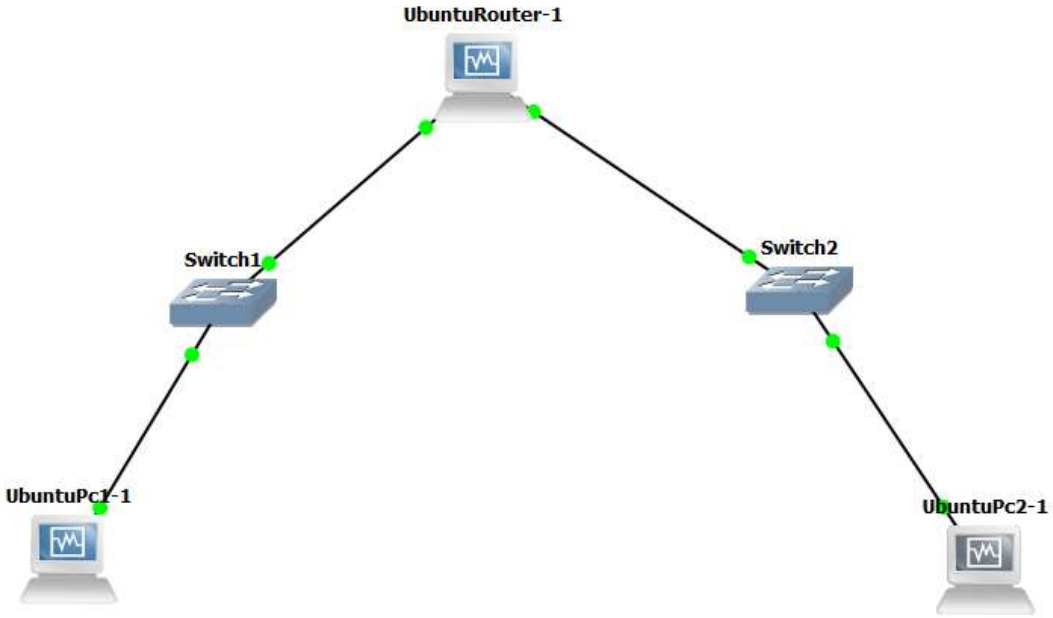
Practical Assignment –

2018cs015 – 18000152

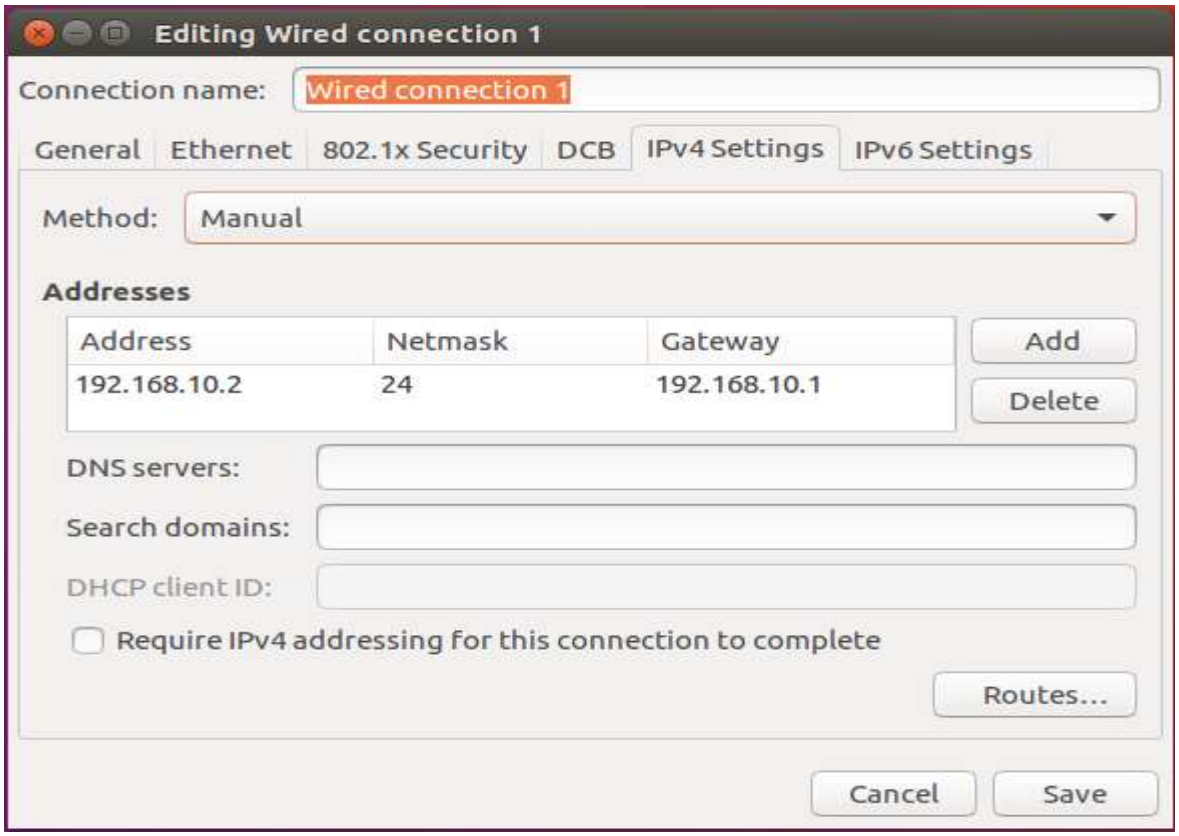
H.P.N ARAVINDA

Part 1

1. Device configurations



Configure the network by assigning IP addresses to relevant interfaces.



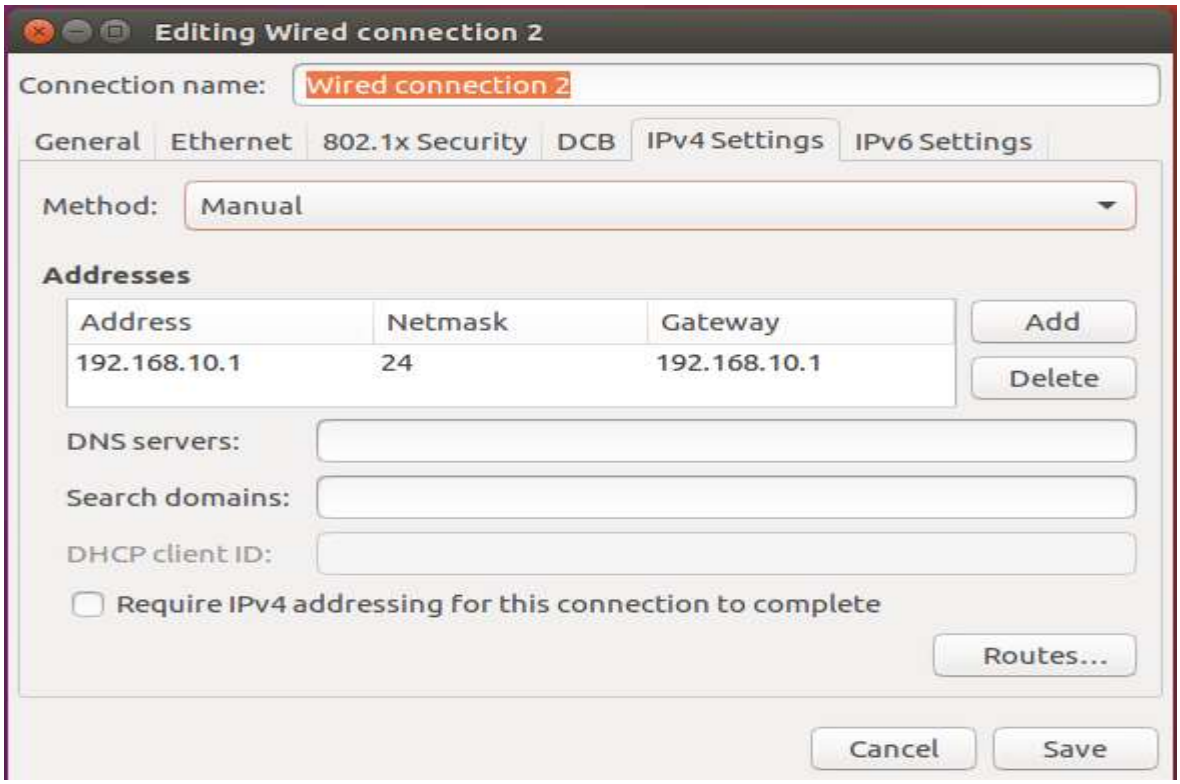

Editing Wired connection 1
Connection name:
Wired connection 1
General
Ethernet
802.1x Security
DCB
IPv4 Settings
IPv6 Settings
Device:
enp0s3 (08:00:27:ED:CC:2F)
Cloned MAC address:
MTU:
automatic
bytes
Wake on LAN:
Default
Phy
Unicast
Multicast
Ignore
Broadcast
Arp
Magic
Wake on LAN password:
Cancel
Save

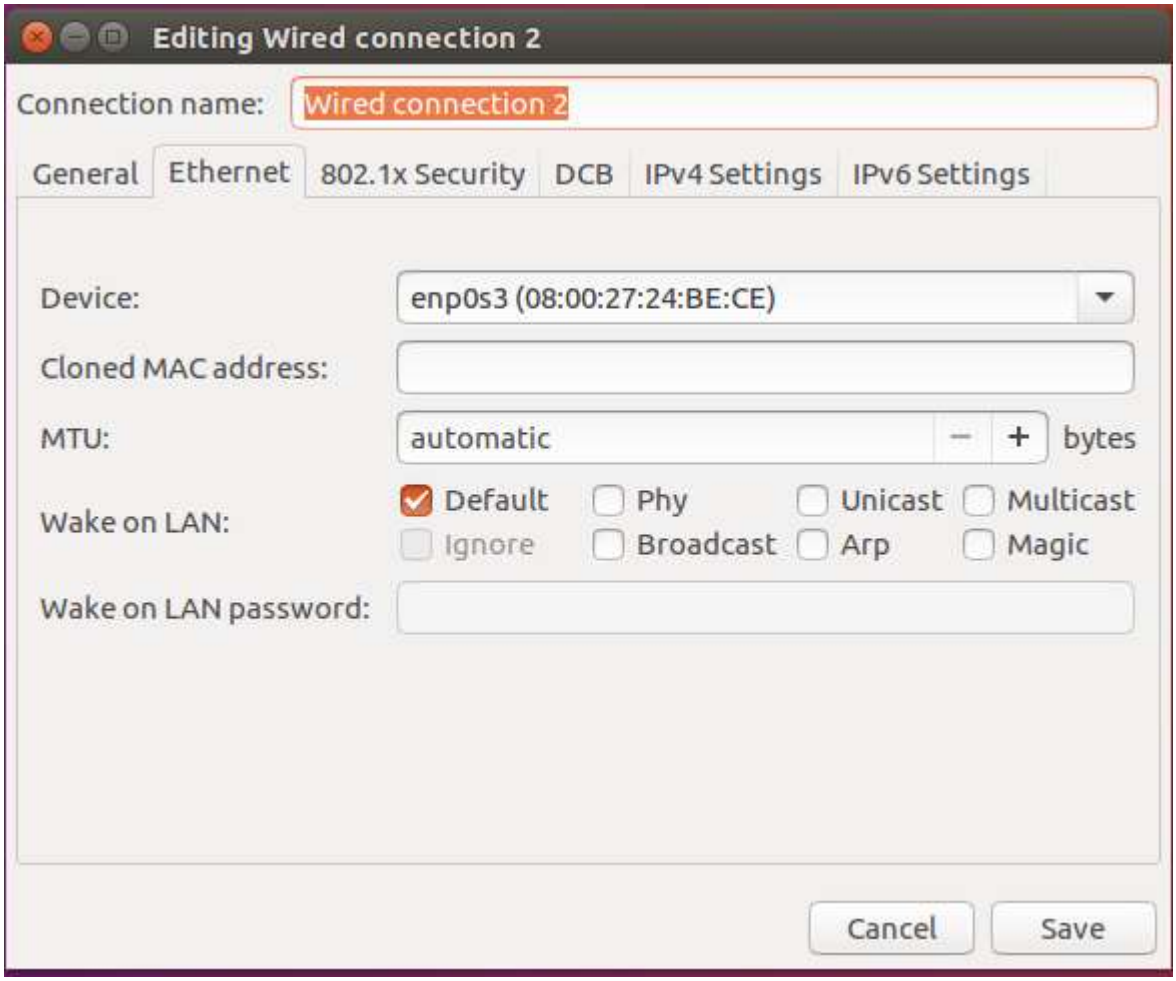

Editing Wired connection 1
Connection name:
Wired connection 1
General
Ethernet
802.1x Security
DCB
IPv4 Settings
IPv6 Settings
Method:
Manual
Addresses
Address
Netmask
Gateway
192.168.20.2
24
192.168.20.1
Add
Delete
DNS servers:
Search domains:
DHCP client ID:
Require IPv4 addressing for this connection to complete
Routes...
Cancel
Save


Editing Wired connection 1
Connection name:
Wired connection 1
General
Ethernet
802.1x Security
DCB
IPv4 Settings
IPv6 Settings
Device:
enp0s3 (08:00:27:9D:85:1D)
Cloned MAC address:
MTU:
automatic
bytes
Wake on LAN:
Default
Phy
Unicast
Multicast
Ignore
Broadcast
Arp
Magic
Wake on LAN password:
Cancel
Save

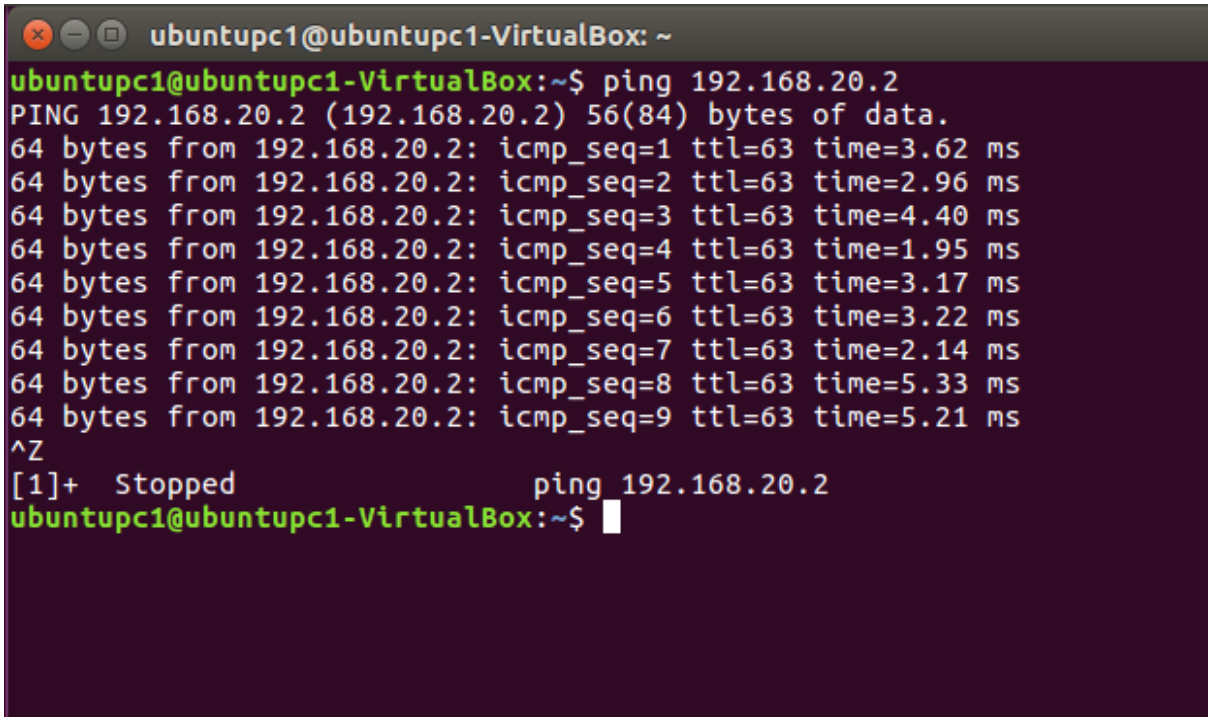

Editing Wired connection 1
Connection name:
Wired connection 1
General
Ethernet
802.1x Security
DCB
IPv4 Settings
IPv6 Settings
Method:
Manual
Addresses
Address
Netmask
Gateway
192.168.20.1
24
192.168.20.1
Add
Delete
DNS servers:
Search domains:
DHCP client ID:
Require IPv4 addressing for this connection to complete
Routes...
Cancel
Save

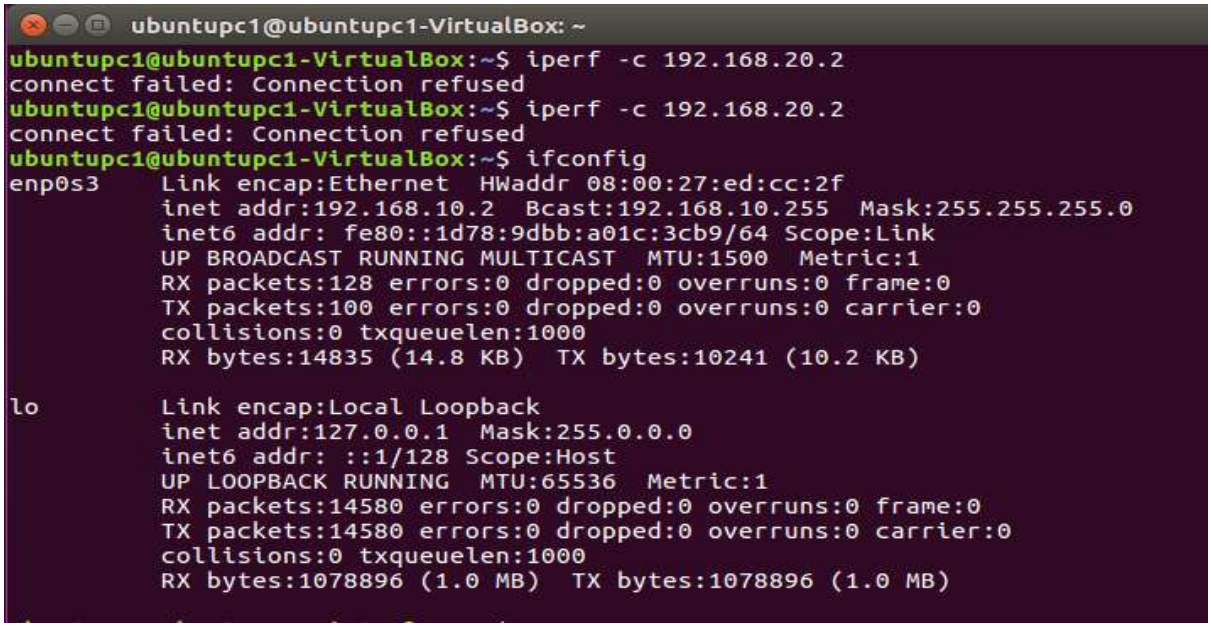

Editing Wired connection 1
Connection name:
Wired connection 1
General
Ethernet
802.1x Security
DCB
IPv4 Settings
IPv6 Settings
Device:
enp0s8 (08:00:27:5B:80:44)
Cloned MAC address:
MTU:
automatic
bytes
Wake on LAN:
Default
Phy
Unicast
Multicast
Ignore
Broadcast
Arp
Magic
Wake on LAN password:
Cancel
Save

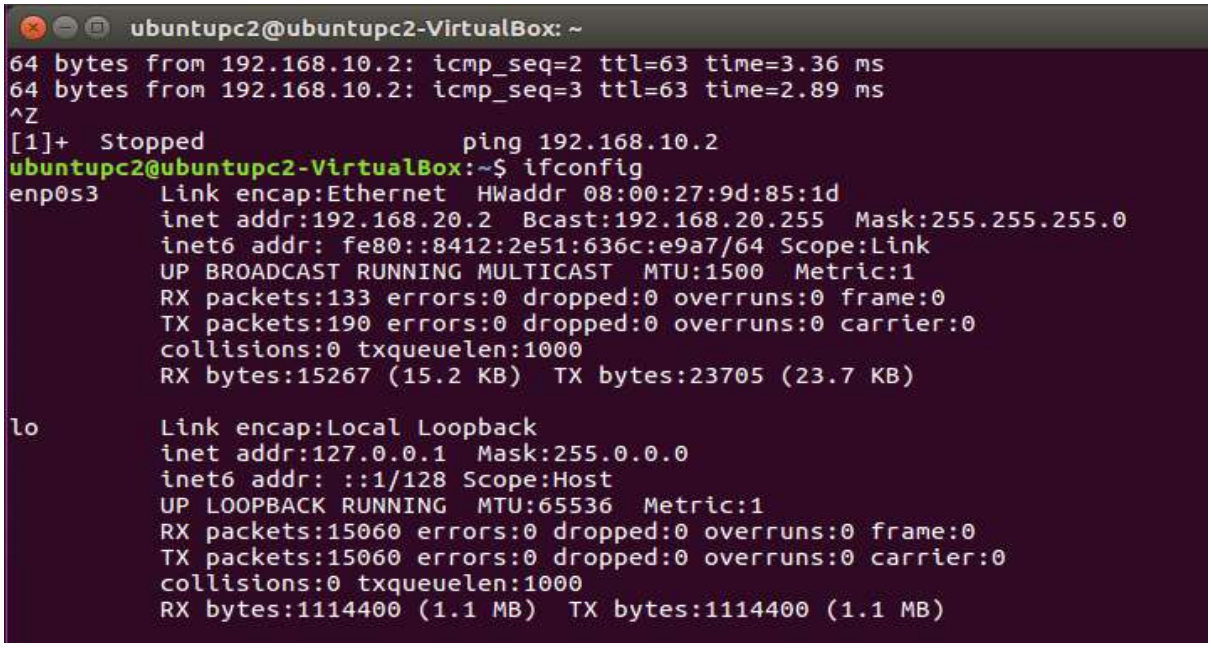

Editing Wired connection 2
Connection name:
Wired connection 2
General
Ethernet
802.1x Security
DCB
IPv4 Settings
IPv6 Settings
Method:
Manual
Addresses
Address
Netmask
Gateway
192.168.10.1
24
192.168.10.1
Add
Delete
DNS servers:
Search domains:
DHCP client ID:
Require IPv4 addressing for this connection to complete
Routes...
Cancel
Save

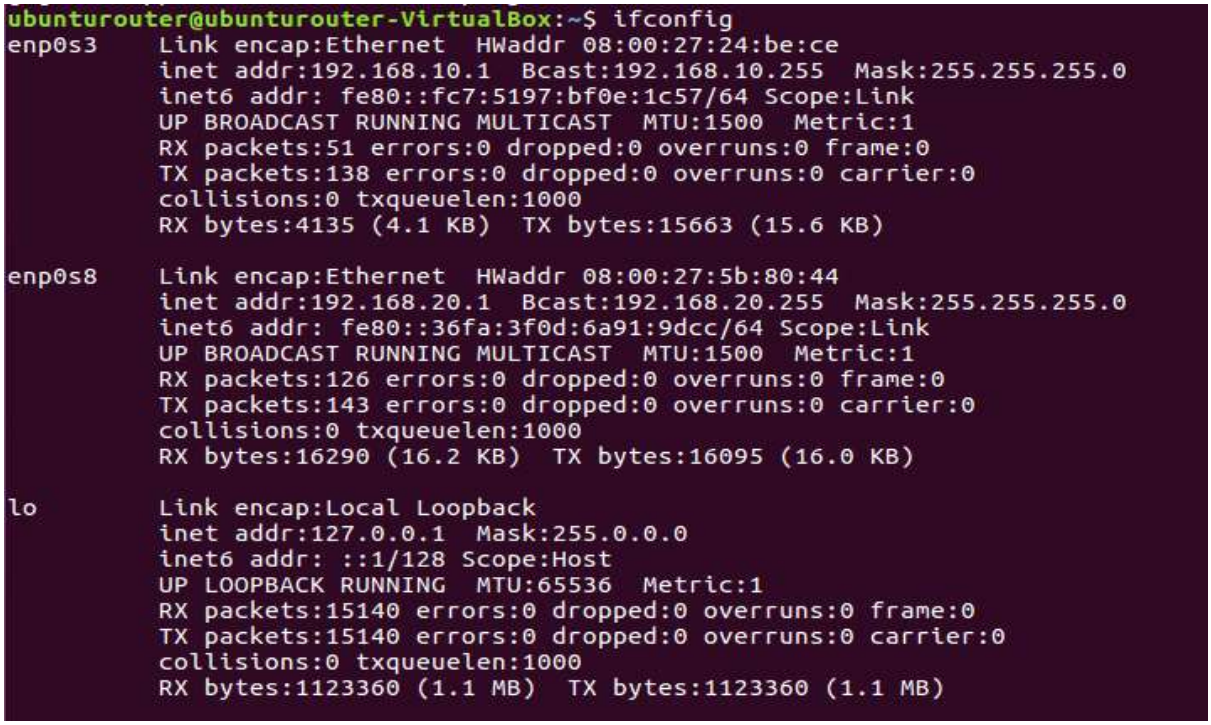


Check whether the Connection is established between UbuntuPc1 and UbuntuPc2.

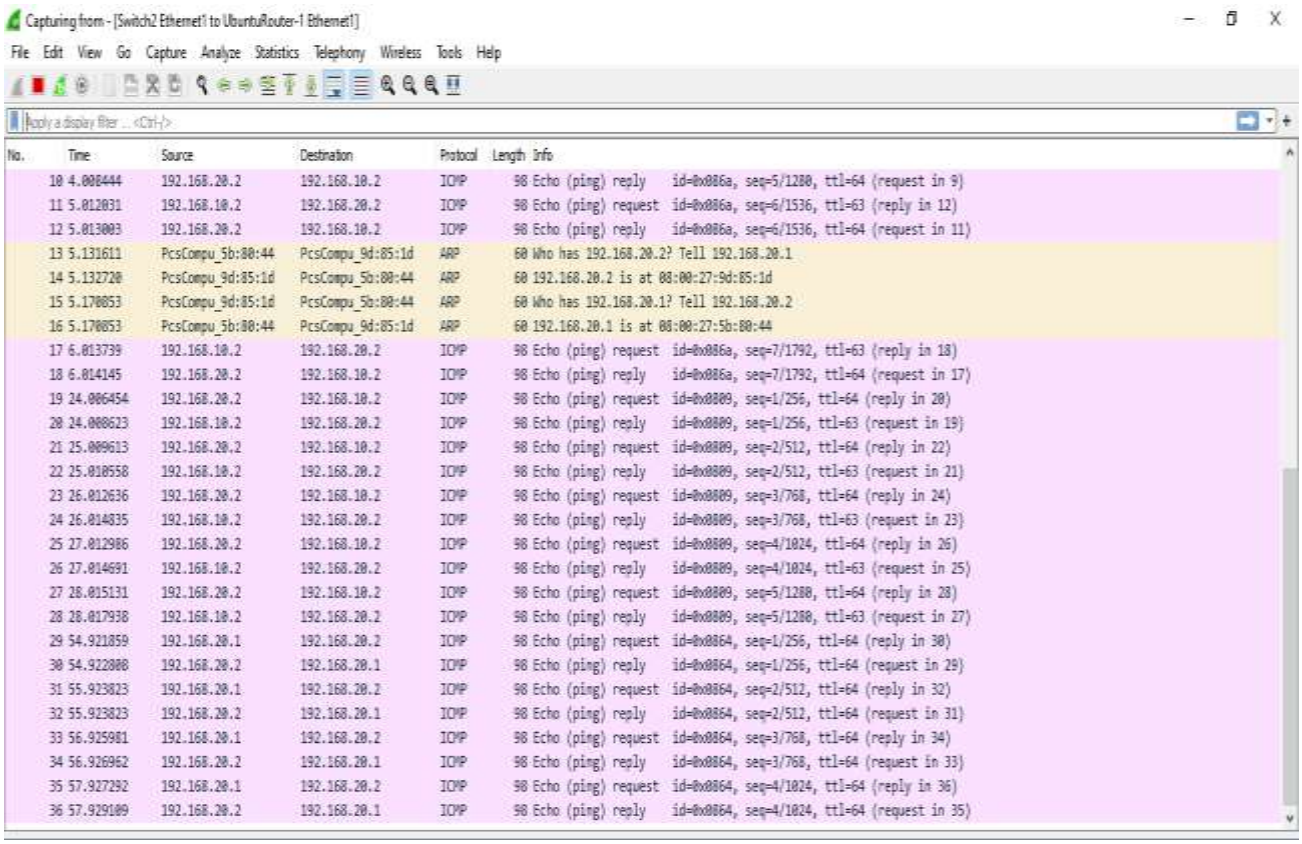

ubuntupc1@ubuntupc1-VirtualBox: ~
ubuntupc1@ubuntupc1-VirtualBox:~$ ping 192.168.20.2
PING 192.168.20.2 (192.168.20.2) 56(84) bytes of data.
64 bytes from 192.168.20.2: icmp_seq=1 ttl=63 time=3.62 ms
64 bytes from 192.168.20.2: icmp_seq=2 ttl=63 time=2.96 ms
64 bytes from 192.168.20.2: icmp_seq=3 ttl=63 time=4.40 ms
64 bytes from 192.168.20.2: icmp_seq=4 ttl=63 time=1.95 ms
64 bytes from 192.168.20.2: icmp_seq=5 ttl=63 time=3.17 ms
64 bytes from 192.168.20.2: icmp_seq=6 ttl=63 time=3.22 ms
64 bytes from 192.168.20.2: icmp_seq=7 ttl=63 time=2.14 ms
64 bytes from 192.168.20.2: icmp_seq=8 ttl=63 time=5.33 ms
64 bytes from 192.168.20.2: icmp_seq=9 ttl=63 time=5.21 ms
^Z
[1]+ Stopped ping 192.168.20.2
ubuntupc1@ubuntupc1-VirtualBox:~$


ubuntupc1@ubuntupc1-VirtualBox: ~
ubuntupc1@ubuntupc1-VirtualBox:~$ iperf -c 192.168.20.2
connect failed: Connection refused
ubuntupc1@ubuntupc1-VirtualBox:~$ iperf -c 192.168.20.2
connect failed: Connection refused
ubuntupc1@ubuntupc1-VirtualBox:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:ed:cc:2f
inet addr:192.168.10.2 Bcast:192.168.10.255 Mask:255.255.255.0
inet6 addr: fe80::1d78:9dbb:a01c:3cb9/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:128 errors:0 dropped:0 overruns:0 frame:0
TX packets:100 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:14835 (14.8 KB) TX bytes:10241 (10.2 KB)
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:14580 errors:0 dropped:0 overruns:0 frame:0
TX packets:14580 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1078896 (1.0 MB) TX bytes:1078896 (1.0 MB)

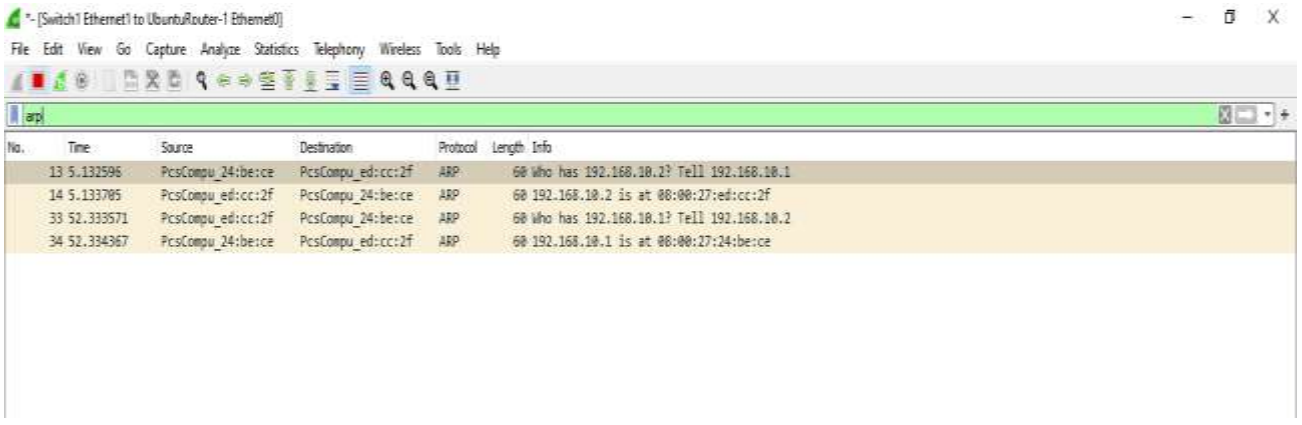

ubuntupc2@ubuntupc2-VirtualBox: ~
64 bytes from 192.168.10.2: icmp_seq=2 ttl=63 time=3.36 ms
64 bytes from 192.168.10.2: icmp_seq=3 ttl=63 time=2.89 ms
^Z
[1]+ Stopped ping 192.168.10.2
ubuntupc2@ubuntupc2-VirtualBox:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:9d:85:1d
inet addr:192.168.20.2 Bcast:192.168.20.255 Mask:255.255.255.0
inet6 addr: fe80::8412:2e51:636c:e9a7/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:133 errors:0 dropped:0 overruns:0 frame:0
TX packets:190 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:15267 (15.2 KB) TX bytes:23705 (23.7 KB)
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:15060 errors:0 dropped:0 overruns:0 frame:0
TX packets:15060 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1114400 (1.1 MB) TX bytes:1114400 (1.1 MB)

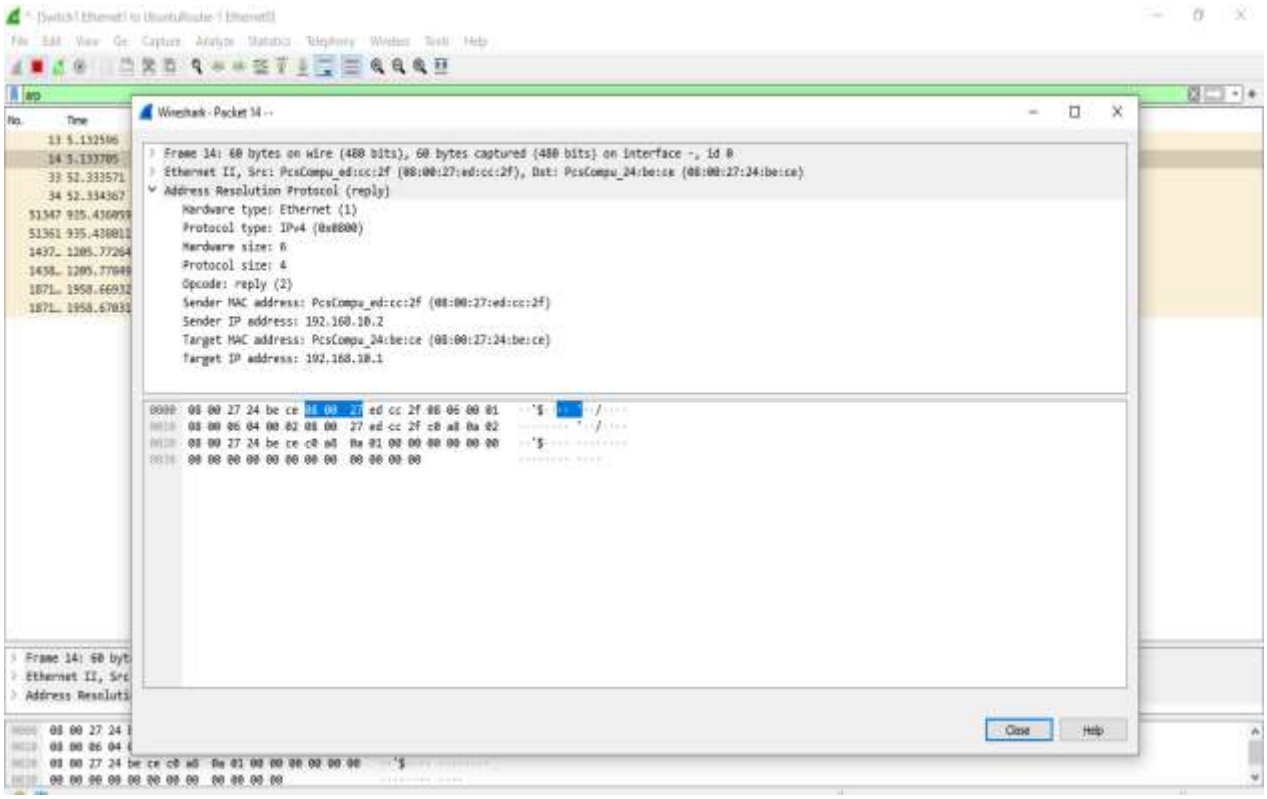

ubunturouter@ubunturouter-VirtualBox:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:24:be:ce
inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
inet6 addr: fe80::fc7:5197:bf0e:1c57/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:51 errors:0 dropped:0 overruns:0 frame:0
TX packets:138 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4135 (4.1 KB) TX bytes:15663 (15.6 KB)
enp0s8 Link encap:Ethernet HWaddr 08:00:27:5b:80:44
inet addr:192.168.20.1 Bcast:192.168.20.255 Mask:255.255.255.0
inet6 addr: fe80::36fa:3f0d:6a91:9dcc/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:126 errors:0 dropped:0 overruns:0 frame:0
TX packets:143 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:16290 (16.2 KB) TX bytes:16095 (16.0 KB)
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:15140 errors:0 dropped:0 overruns:0 frame:0
TX packets:15140 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1123360 (1.1 MB) TX bytes:1123360 (1.1 MB)

2. Capture the Network packets on ubunturouter while checking the connection between UbuntuPc1 UbuntuPc2.



3. Filter the ARP packets and identify the sender and destination, IP addresses and MAC addresses.





Sender MAC address: 08:00:27:ed:cc:2f

Sender IP address: 192.168.10.2

Receiver MAC address: 08:00:27:24:be:ce

Receiver IP address: 192.168.10.1

The ping is done between UbuntuPc1 and UbunduPc2. The packet capture is done between the UbuntuPc1 and UbuntuRouter.

So the sender IP is 192.168.10.2(UbuntuPc1) and the receiver IP address is 192.168.10.1(UbuntuRouter).

Part 2

1. Install iperf on UbuntuPC2.

```
ubuntupc2@ubuntupc2-VirtualBox:~$ iperf
Usage: iperf [-s|-c host] [options]
Try `iperf --help' for more information.
```

I also iperf install to the UbuntuPc1.

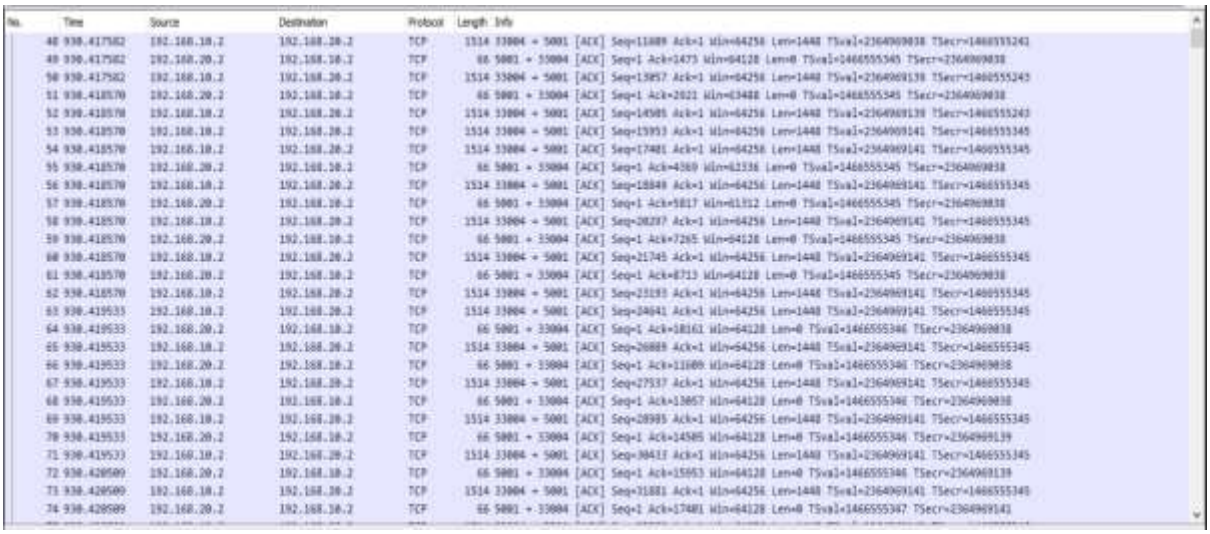
```
ubuntupc1@ubuntupc1-VirtualBox:~$ sudo apt install iperf
[sudo] password for ubuntupc1:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  iperf
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 50.5 kB of archives.
After this operation, 156 kB of additional disk space will be used.
Get:1 http://lk.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 iperf am
ubuntu0.1 [50.5 kB]
LibreOffice Writer
n 1s (31.1 kB/s)
Selecting previously unselected package iperf.
(Reading database ... 177094 files and directories currently installed.)
Preparing to unpack .../iperf_2.0.5+dfsg1-2ubuntu0.1_amd64.deb ...
Unpacking iperf (2.0.5+dfsg1-2ubuntu0.1) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for doc-base (0.10.7) ...
Processing 1 added doc-base file...
Setting up iperf (2.0.5+dfsg1-2ubuntu0.1) ...
ubuntupc1@ubuntupc1-VirtualBox:~$ sudo apt-get update
Hit:1 http://lk.archive.ubuntu.com/ubuntu xenial InRelease
Hit:2 http://security.ubuntu.com/ubuntu xenial-security InRelease
Hit:3 http://lk.archive.ubuntu.com/ubuntu xenial-updates InRelease
Hit:4 http://lk.archive.ubuntu.com/ubuntu xenial-backports InRelease
Reading package lists... Done
ubuntupc1@ubuntupc1-VirtualBox:~$
```

2. Run the iperf service and try to connect to it by using the UbuntuPC1.

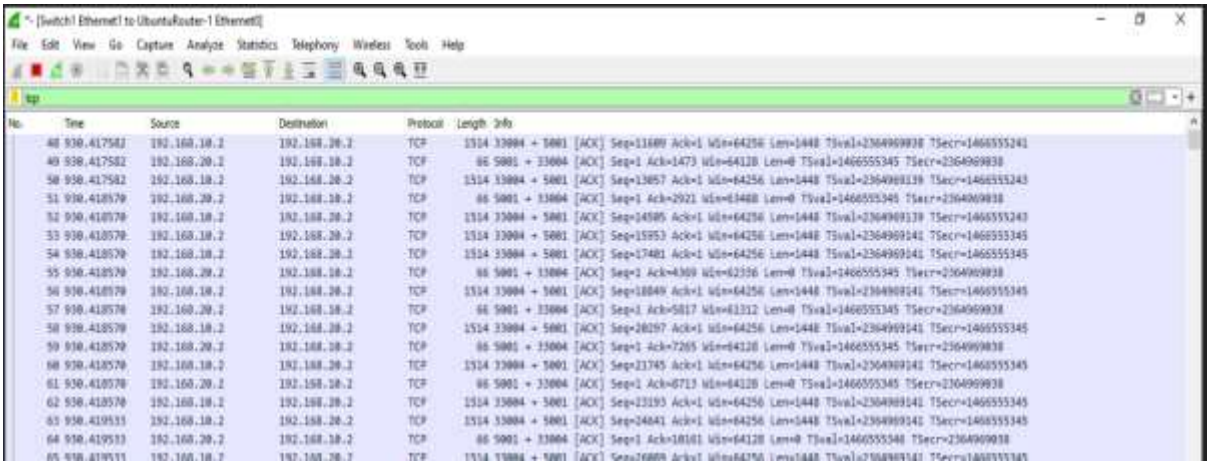
```
ubuntupc2@ubuntupc2-VirtualBox: ~
ubuntupc2@ubuntupc2-VirtualBox:~$ iperf -s
------------------------------------------------------------
Server listening on TCP port 5001
TCP window size:  128 KByte (default)
------------------------------------------------------------
```

```
ubuntupc1@ubuntupc1-VirtualBox: ~
64 bytes from 192.168.20.2: icmp_seq=6 ttl=63 time=4.79 ms
64 bytes from 192.168.20.2: icmp_seq=7 ttl=63 time=3.41 ms
^Z
[1]+  Stopped                 ping 192.168.20.2
ubuntupc1@ubuntupc1-VirtualBox:~$ iperf
Usage: iperf [-s|-c host] [options]
Try `iperf --help' for more information.
ubuntupc1@ubuntupc1-VirtualBox:~$ iperf -c 192.168.20.2
------------------------------------------------------------
Client connecting to 192.168.20.2, TCP port 5001
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 192.168.10.2 port 33004 connected with 192.168.20.2 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  93.0 MBytes  77.8 Mbits/sec
ubuntupc1@ubuntupc1-VirtualBox:~$ iperf -c 192.168.20.2
------------------------------------------------------------
Client connecting to 192.168.20.2, TCP port 5001
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 192.168.10.2 port 33006 connected with 192.168.20.2 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  79.1 MBytes  66.3 Mbits/sec
ubuntupc1@ubuntupc1-VirtualBox:~$
```

3. Capture network packets on UbuntuRouter.



4. Filter the packets and identify the TCP packets related to the iperf connection.



5. Mention the information you can extract from the packets captured from the UbuntuRouter.

   Frame number
   Sender's MAC address and receiver's MAC address
   IP addresses of source and destination
   Internet protocol
   TCP port numbers of the source and destination.
   Data inside the packets can also be analyzed through the TCP stream but it shows a stream of numbers because the iperf transfers random data to analyze the network performance.