

CPSC 456 Assignment 1

Q1:Please provide the names and roles of all team members. All team members should participate in answering all questions.

You will have to download and import some GNS3 appliances for this assignment:

- Open vSwitch: This is an open source switch operating system that supports VLANs and many other features. You can download it from [this link](#)

[Links to an external site.](#)

- Firefox VM: This is a lightweight graphical VM that has the Firefox browser installed.

You can download it from [this link](#)

[Links to an external site.](#)

To import an appliance, follow these steps (using Open vSwitch as an example):

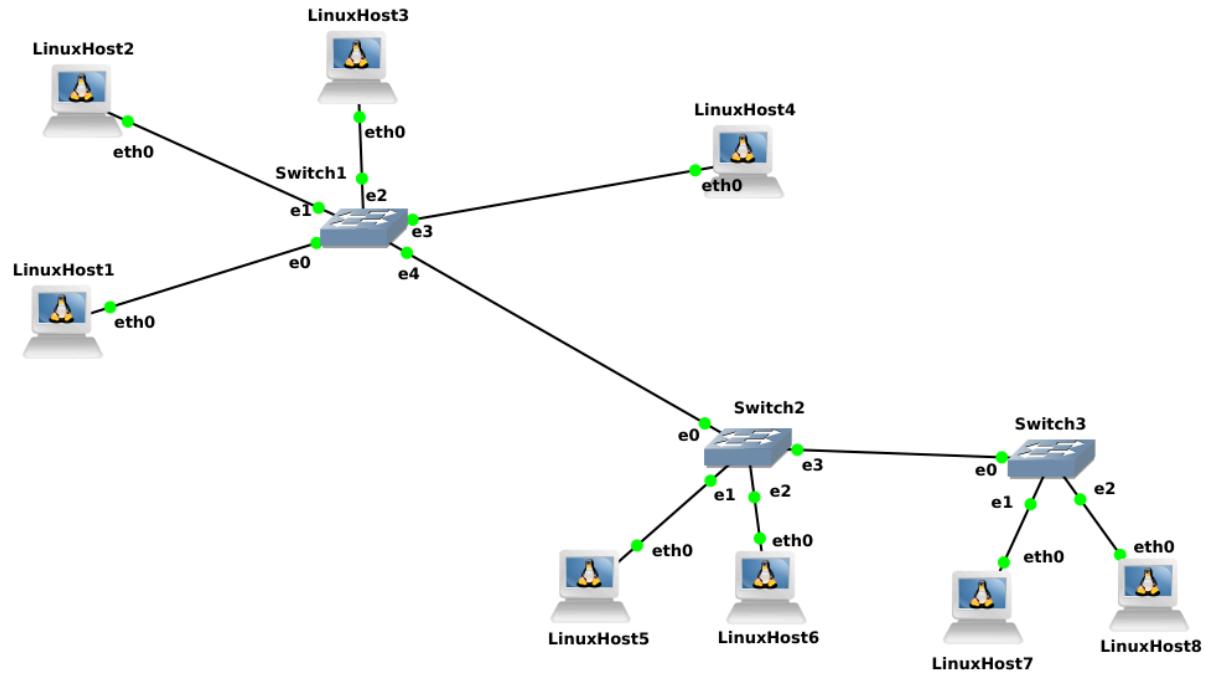
Download the appliance file from the link above.

1. Go to Edit->Import Appliance in GNS3 and select the file you downloaded.
2. Click next and accept all the prompts.
3. For more details on importing appliances, please check [this page](#)

[Links to an external site.](#)

Question 2

1. Build the following topology in GNS3 using the Ubuntu Docker Guest as a system and include the screenshot:



1. Use the ifconfig command on every system to obtain its MAC address of eth0 on every system. Please prove your findings with screenshots. For example:

```

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost4 console is now available... Press RETURN to get started.
root@LinuxHost4:~# ifconfig
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 fe80::94c3:1aff:fe86:ff64 prefixlen 64 scopeid 0x20<link>
        ether 96:c3:1a:86:ff:64 txqueuelen 1000 (Ethernet)
          RX packets 12 bytes 840 (840.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 12 bytes 936 (936.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

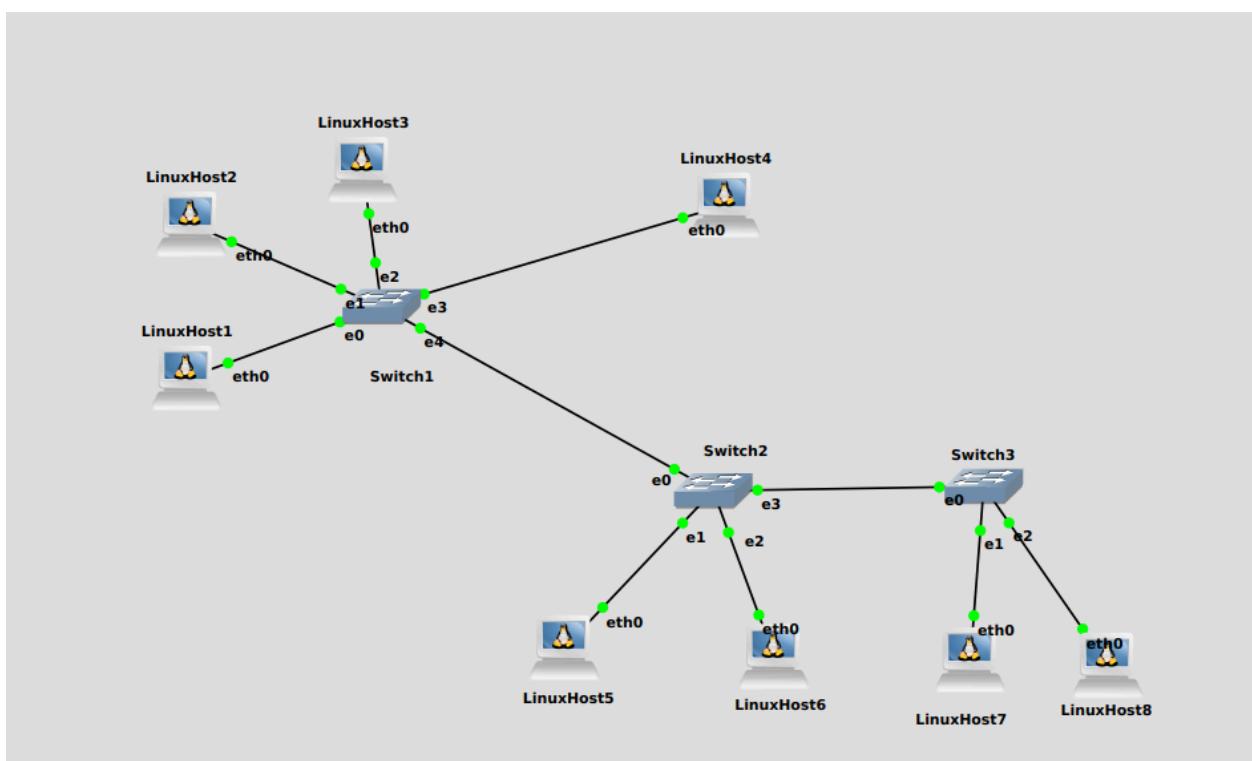
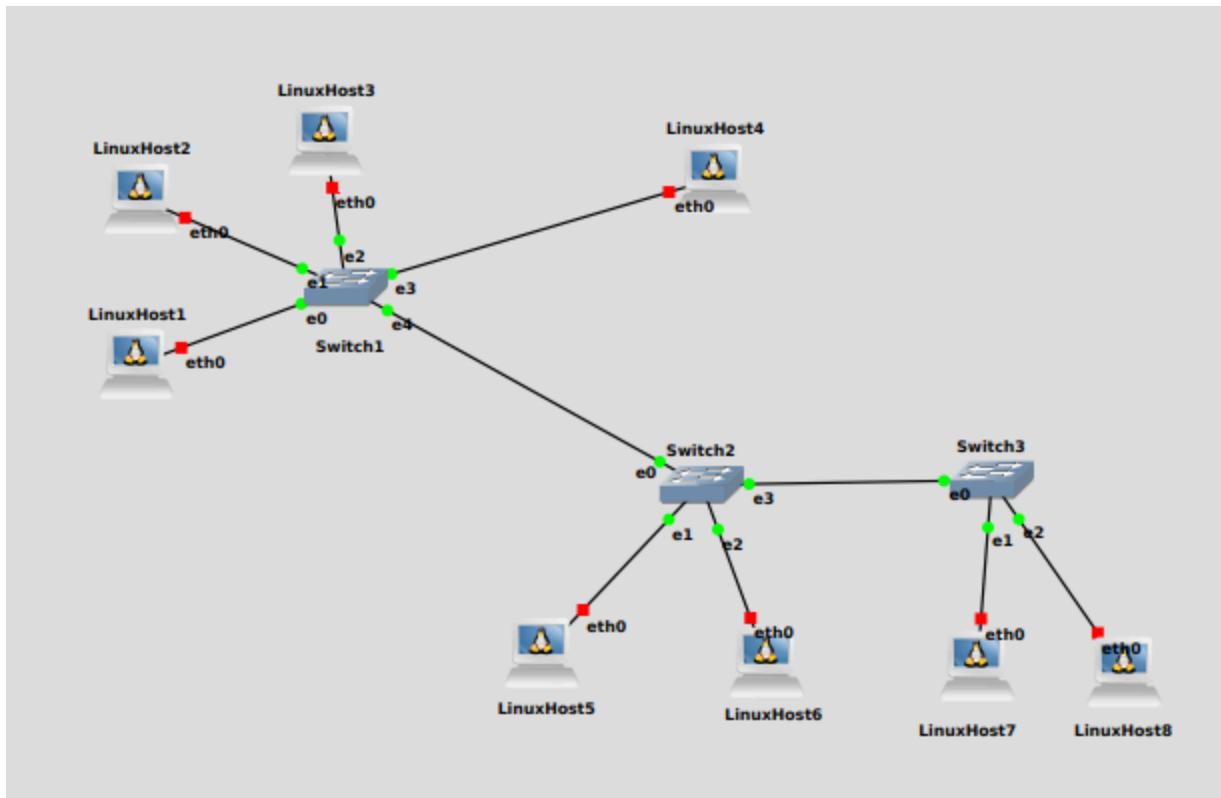
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
root@LinuxHost4:~# 
  
```

2. For each switch construct the Addressable Memory (CAM) table for each switch. Each table should have the form of:

| MAC/HW Address | Switch Port | VLAN |
|-------------------|-------------|------|
| 08:00:27:d3:74 | e4 | 1 |
| 00:50:79:66:68:00 | e0 | 1 |
| 00:50:79:66:68:01 | e1 | 1 |
| 00:50:79:66:68:02 | e2 | 1 |
| 00:50:79:66:68:03 | e3 | 1 |

Answers and Screenshots

Build the following topology in GNS3 using the Ubuntu Docker Guest as a system and include the screenshot:



Use the ifconfig command on every system to obtain its MAC address of eth0 on every system. Please prove your findings with screenshots. For example:

```
[+] root@LinuxHost1: ~ Q - x
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost1 console is now available... Press RETURN to get started.
root@LinuxHost1:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:39ff:fea3:c100 prefixlen 64 scopeid 0x20<link>
        ether 02:42:39:a3:c1:00 txqueuelen 1000 (Ethernet)
        RX packets 54 bytes 4452 (4.4 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 8 bytes 656 (656.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost1:~#
```

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost1 console is now available... Press RETURN to get started.
root@LinuxHost1:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:39ff:fea3:c100 prefixlen 64 scopeid 0x20<link>
        ether 02:42:39:a3:c1:00 txqueuelen 1000 (Ethernet)
        RX packets 54 bytes 4452 (4.4 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 8 bytes 656 (656.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost1:~#
```

```
[+] root@LinuxHost2: ~ Q - x
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost2 console is now available... Press RETURN to get started.
root@LinuxHost2:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:f7ff:fec5:3a00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:f7:c5:3a:00 txqueuelen 1000 (Ethernet)
        RX packets 68 bytes 5432 (5.4 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 10 bytes 796 (796.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost2:~#
```

```
[+] root@LinuxHost3: ~ Q - x
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost3 console is now available... Press RETURN to get started.
root@LinuxHost3:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:f2ff:fe0f:3a00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:f2:0f:3a:00 txqueuelen 1000 (Ethernet)
        RX packets 71 bytes 5642 (5.6 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 10 bytes 796 (796.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost3:~#
```

```
root@LinuxHost4: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost4 console is now available... Press RETURN to get started.
root@LinuxHost4:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:8bff:fe8c:ea00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:8b:8c:ea:00 txqueuelen 1000 (Ethernet)
        RX packets 69 bytes 5422 (5.4 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 11 bytes 866 (866.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost4:~#
```

```
root@LinuxHost5: ~
Connected to localhost.
Escape character is '^]'.
LinuxHost5 console is now available... Press RETURN to get started.
root@LinuxHost5:~# if
if      ifconfig
root@LinuxHost5:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:f7ff:fe0f:7a00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:f7:0f:7a:00 txqueuelen 1000 (Ethernet)
        RX packets 72 bytes 5612 (5.6 KB)
        RX errors 0 dropped 1 overruns 0 frame 0
        TX packets 11 bytes 866 (866.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost5:~#
```

```
[+] root@LinuxHost6: ~ Q - x
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost6 console is now available... Press RETURN to get started.
root@LinuxHost6:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:16ff:fe6f:c700 prefixlen 64 scopeid 0x20<link>
        ether 02:42:16:6f:c7:00 txqueuelen 1000 (Ethernet)
        RX packets 72 bytes 5612 (5.6 KB)
        RX errors 0 dropped 1 overruns 0 frame 0
        TX packets 11 bytes 866 (866.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost6:~#
```

```
[+] root@LinuxHost7:~ Q - x
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost7 console is now available... Press RETURN to get started.
root@LinuxHost7:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:8eff:fed2:6b00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:8e:d2:6b:00 txqueuelen 1000 (Ethernet)
        RX packets 61 bytes 4642 (4.6 KB)
        RX errors 0 dropped 6 overruns 0 frame 0
        TX packets 11 bytes 866 (866.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost7:~#
```

```
[+] root@LinuxHost8:~ Q - x
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost8 console is now available... Press RETURN to get started.
root@LinuxHost8:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:10ff:fedf:db00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:10:df:db:00 txqueuelen 1000 (Ethernet)
        RX packets 61 bytes 4622 (4.6 KB)
        RX errors 0 dropped 6 overruns 0 frame 0
        TX packets 11 bytes 866 (866.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost8:~#
```

For each switch construct the Addressable Memory (CAM) table for each switch. Each table should have the form of:

Switch 1

| MAC/HW Address | Switch Port | VLAN |
|-------------------|-------------|------|
| 02:42:39:a3:c1:00 | e0 | 1 |
| 02:42:f7:c5:3a:00 | e1 | 1 |
| 02:42:f2:0f:3a:00 | e2 | 1 |
| 02:42:8b:8c:ea:00 | e3 | 1 |
| 02:42:f7:0f:7a:00 | e4 | 1 |
| 02:42:16:6f:c7:00 | e4 | 1 |
| 02:42:8e:d2:6b:00 | e4 | 1 |
| 02:42:10:df:db:00 | e4 | 1 |

Switch 2

| MAC/HW Address | Switch Port | VLAN |
|-------------------|-------------|------|
| 02:42:39:a3:c1:00 | e0 | 1 |
| 02:42:f7:c5:3a:00 | e0 | 1 |
| 02:42:f2:0f:3a:00 | e0 | 1 |
| 02:42:8b:8c:ea:00 | e0 | 1 |
| 02:42:f7:0f:7a:00 | e1 | 1 |
| 02:42:16:6f:c7:00 | e2 | 1 |
| 02:42:8e:d2:6b:00 | e3 | 1 |
| 02:42:10:df:db:00 | e3 | 1 |

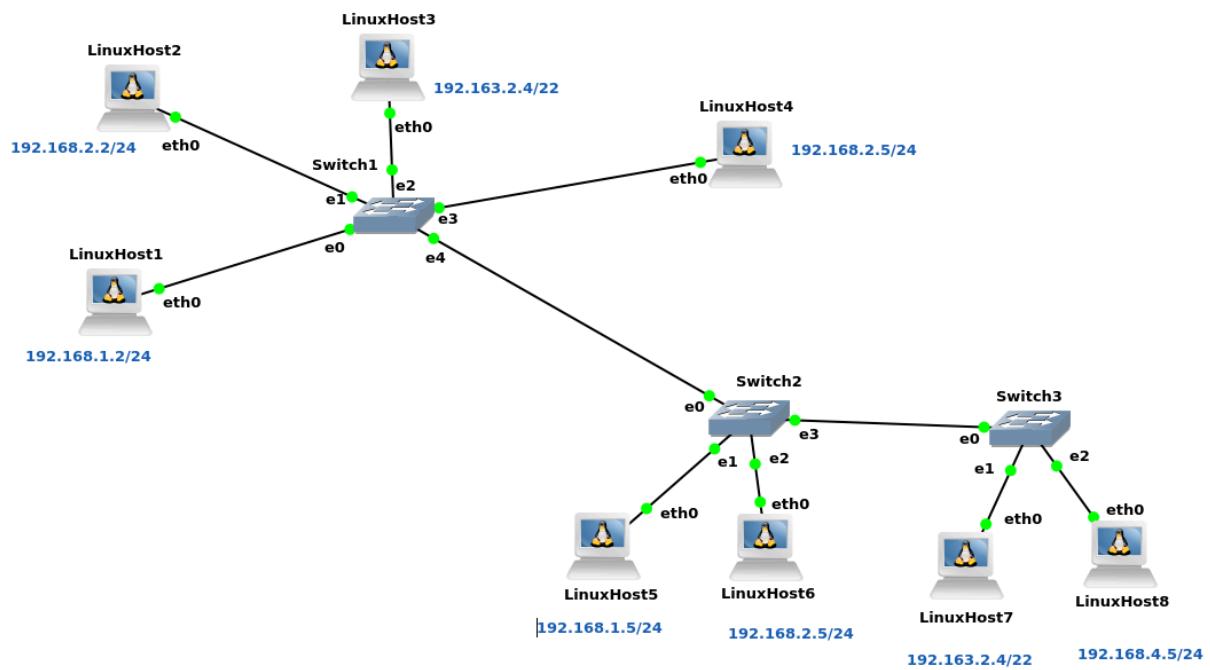
Switch 3

| MAC/HW Address | Switch Port | VLAN |
|-------------------|-------------|------|
| 02:42:39:a3:c1:00 | e0 | 1 |
| 02:42:f7:c5:3a:00 | e0 | 1 |
| 02:42:f2:0f:3a:00 | e0 | 1 |
| 02:42:8b:8c:ea:00 | e0 | 1 |
| 02:42:f7:0f:7a:00 | e0 | 1 |
| 02:42:16:6f:c7:00 | e0 | 1 |
| 02:42:8e:d2:6b:00 | e1 | 1 |
| 02:42:10:df:db:00 | e2 | 1 |

Question 3

Configure the static IP of each Linux Host with the IP and subnet as follows:

You will need to modify the /etc/network/interfaces file on each system that has network adapter configuration. After saving the file, restart the system and use ifconfig to verify the IP and the subnet mask of eth0. This will confirm that your changes were applied correctly



You can use the following commands:

```
root@LinuxHost:~# pico /etc/network/interfaces
```

Next, uncomment the following lines in the file, and replace the IP address and the subnet mask with the appropriate network mask and IP.

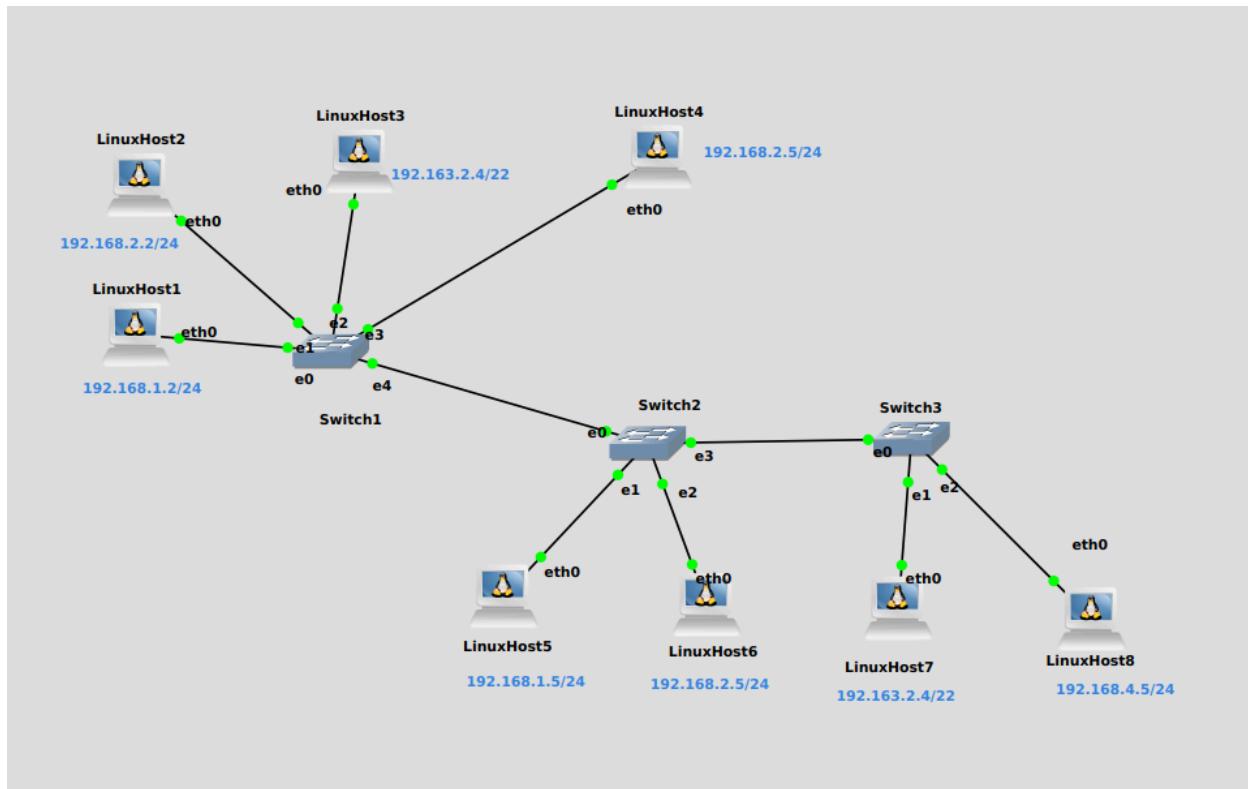
```
#auto eth0  
  
#iface eth0 inet static  
  
#    address 192.168.0.2  
  
#    netmask 255.255.255.0
```

Next, uncomment the following lines in the file, and replace the IP address and the subnet mask with the appropriate network mask and IP.

You can save files in the pico editor by pressing Ctrl+x.

- (1) Please provide screenshots for the ifconfig output to prove that your changes have taken effect.
- (2) What systems can each system communicate with? Please explain why a system can or cannot communicate with the system in question.

1. Screenshots for this question



```

root@LinuxHost1: ~
GNU nano 7.2          /etc/network/interfaces *

#
# This is a sample network config, please uncomment lines to configure the netw>
#

# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

# Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.0.2
#    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1

^G Help      ^O Write Out  ^W Where Is  ^K Cut      ^T Execute  ^C Location
^X Exit      ^R Read File  ^\ Replace   ^U Paste    ^J Justify  ^/ Go To Line
#iface eth0 inet static

```

```
root@LinuxHost1:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:b9ff:fe02:7900 prefixlen 64 scopeid 0x20<link>
        ether 02:42:b9:02:79:00 txqueuelen 1000 (Ethernet)
            RX packets 84 bytes 6552 (6.5 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 12 bytes 936 (936.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost1:~#
```

```
Escape character is '^].
LinuxHost2 console is now available... Press RETURN to get started.
root@LinuxHost2:~# pico /etc/network/interfaces
root@LinuxHost2:~# pico /etc/network/interfaces
root@LinuxHost2:~# pico /etc/network/interfaces
root@LinuxHost2:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:bfff:fe16:4a00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:bf:16:4a:00 txqueuelen 1000 (Ethernet)
            RX packets 89 bytes 6882 (6.8 KB)
            RX errors 0 dropped 1 overruns 0 frame 0
            TX packets 13 bytes 1006 (1.0 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost2:~#
```

```
root@LinuxHost4: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost4 console is now available... Press RETURN to get started.
root@LinuxHost4:~# pico /etc/network/interfaces
root@LinuxHost4:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:67ff:fe7e:5d00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:67:7e:5d:00 txqueuelen 1000 (Ethernet)
        RX packets 76 bytes 5688 (5.6 KB)
        RX errors 0 dropped 4 overruns 0 frame 0
        TX packets 13 bytes 1006 (1.0 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost4:~#
```

```
root@LinuxHost3: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
pico /etc/network/interfaces
root@LinuxHost3:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:dcff:fe14:5000 prefixlen 64 scopeid 0x20<link>
        ether 02:42:dc:14:50:00 txqueuelen 1000 (Ethernet)
        RX packets 95 bytes 7322 (7.3 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 13 bytes 1006 (1.0 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost3:~#
```

root@LinuxHost5: ~

```
GNU nano 7.2          /etc/network/interfaces *
```

```
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

## Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.1.5
    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost5: ~

```
Escape character is '^]'.
LinuxHost5 console is now available... Press RETURN to get started.
root@LinuxHost5:~# pico /etc/net
netconfig  network/  networks
root@LinuxHost5:~# pico /etc/network/interfaces
root@LinuxHost5:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::42:edff:fe02:2000  prefixlen 64  scopeid 0x20<link>
        ether 02:42:ed:02:20:00  txqueuelen 1000  (Ethernet)
        RX packets 90  bytes 6824 (6.8 KB)
        RX errors 0  dropped 5  overruns 0  frame 0
        TX packets 14  bytes 1076 (1.0 KB)
        TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

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 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

root@LinuxHost5:~#
```

```
root@LinuxHost6: ~
GNU nano 7.2          /etc/network/interfaces *

#
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

## Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.2.5
#    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute   ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify   ^/ Go To Line
```

```
root@LinuxHost6: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost6 console is now available... Press RETURN to get started.
root@LinuxHost6:~# pico /etc/network/interfaces
root@LinuxHost6:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 fe80::42:16ff:fec1:4700 prefixlen 64 scopeid 0x20<link>
        ether 02:42:16:c1:47:00 txqueuelen 1000 (Ethernet)
        RX packets 89 bytes 6734 (6.7 KB)
        RX errors 0 dropped 5 overruns 0 frame 0
        TX packets 14 bytes 1076 (1.0 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost6:~#
```

root@LinuxHost7: ~

```
GNU nano 7.2          /etc/network/interfaces *
```

```
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

## Static config for eth0
auto eth0
iface eth0 inet static
    address 192.163.2.4
    netmask 255.255.252.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost6: ~

```
Escape character is '^]'.
jj
bash: jj: command not found
root@LinuxHost6:~# if
if      ifconfig
root@LinuxHost6:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet6 fe80::42:16ff:fec1:4700  prefixlen 64  scopeid 0x20<link>
      ether 02:42:16:c1:47:00  txqueuelen 1000  (Ethernet)
      RX packets 89  bytes 6734 (6.7 KB)
      RX errors 0  dropped 5  overruns 0  frame 0
      TX packets 14  bytes 1076 (1.0 KB)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

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 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

root@LinuxHost6:~#

root@LinuxHost7: ~

```
GNU nano 7.2          /etc/network/interfaces
#
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*
#
## Static config for eth0
auto eth0
iface eth0 inet static
    address 192.163.2.4
    netmask 255.255.252.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1

[ Read 19 lines ]
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost7: ~

```
Connected to localhost.
Escape character is '^]'.
pico/etc/network/interfaces
bash: pico/etc/network/interfaces: No such file or directory
root@LinuxHost7:~# pico /etc/network/interfaces
root@LinuxHost7:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 fe80::42:2fff:fe54:6200 prefixlen 64 scopeid 0x20<link>
        ether 02:42:2f:54:62:00 txqueuelen 1000 (Ethernet)
        RX packets 91 bytes 6914 (6.9 KB)
        RX errors 0 dropped 4 overruns 0 frame 0
        TX packets 14 bytes 1076 (1.0 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost7:~#
```

GNU nano 7.2 /etc/network/interfaces *

```
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

## Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.4.5
    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost8: ~

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
pico /etc/network/interfaces
root@LinuxHost8:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::42:b2ff:fedd:7a00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:b2:dd:7a:00 txqueuelen 1000 (Ethernet)
        RX packets 82 bytes 6108 (6.1 KB)
        RX errors 0 dropped 5 overruns 0 frame 0
        TX packets 14 bytes 1076 (1.0 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost8:~#
```

```
root@LinuxHost2: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost2 console is now available... Press RETURN to get started.
root@LinuxHost2:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.2 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:ffff:fe16:4a00 prefixlen 64 scopeid 0x20<link>
            ether 02:42:bf:16:4a:00 txqueuelen 1000 (Ethernet)
            RX packets 61 bytes 4942 (4.9 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 9 bytes 726 (726.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost2:~#
```

```
root@LinuxHost2: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost2 console is now available... Press RETURN to get started.
root@LinuxHost2:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.2 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:ffff:fe16:4a00 prefixlen 64 scopeid 0x20<link>
            ether 02:42:bf:16:4a:00 txqueuelen 1000 (Ethernet)
            RX packets 61 bytes 4942 (4.9 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 9 bytes 726 (726.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost2:~# pico /etc/network/
```

```
root@LinuxHost3: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost3 console is now available... Press RETURN to get started.
root@LinuxHost3:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.163.2.4 netmask 255.255.252.0 broadcast 0.0.0.0
        inet6 fe80::42:dcff:fe14:5000 prefixlen 64 scopeid 0x20<link>
            ether 02:42:dc:14:50:00 txqueuelen 1000 (Ethernet)
            RX packets 70 bytes 5572 (5.5 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 10 bytes 796 (796.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost3:~#
```

```
root@LinuxHost4: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost4 console is now available... Press RETURN to get started.
root@LinuxHost4:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.5 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:67ff:fe7e:5d00 prefixlen 64 scopeid 0x20<link>
            ether 02:42:67:7e:5d:00 txqueuelen 1000 (Ethernet)
            RX packets 54 bytes 4152 (4.1 KB)
            RX errors 0 dropped 3 overruns 0 frame 0
            TX packets 10 bytes 796 (796.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost4:~#
```

```
root@LinuxHost5: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost5 console is now available... Press RETURN to get started.
root@LinuxHost5:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.5 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:edff:fe02:2000 prefixlen 64 scopeid 0x20<link>
            ether 02:42:ed:02:20:00 txqueuelen 1000 (Ethernet)
            RX packets 66 bytes 5160 (5.1 KB)
            RX errors 0 dropped 3 overruns 0 frame 0
            TX packets 10 bytes 796 (796.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost5:~#
```

```
root@LinuxHost6: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost6 console is now available... Press RETURN to get started.
root@LinuxHost6:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.5 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:16ff:fec1:4700 prefixlen 64 scopeid 0x20<link>
            ether 02:42:16:c1:47:00 txqueuelen 1000 (Ethernet)
            RX packets 67 bytes 5210 (5.2 KB)
            RX errors 0 dropped 3 overruns 0 frame 0
            TX packets 11 bytes 866 (866.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost6:~#
```

```
root@LinuxHost7: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost7 console is now available... Press RETURN to get started.
root@LinuxHost7:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.163.2.4 netmask 255.255.252.0 broadcast 0.0.0.0
        inet6 fe80::42:2fff:fe54:6200 prefixlen 64 scopeid 0x20<link>
            ether 02:42:2f:54:62:00 txqueuelen 1000 (Ethernet)
            RX packets 68 bytes 5260 (5.2 KB)
            RX errors 0 dropped 4 overruns 0 frame 0
            TX packets 11 bytes 866 (866.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost7:~#
```

```
root@LinuxHost8: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost8 console is now available... Press RETURN to get started.
root@LinuxHost8:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.4.5 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:b2ff:fedd:7a00 prefixlen 64 scopeid 0x20<link>
            ether 02:42:b2:dd:7a:00 txqueuelen 1000 (Ethernet)
            RX packets 62 bytes 4732 (4.7 KB)
            RX errors 0 dropped 2 overruns 0 frame 0
            TX packets 11 bytes 866 (866.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost8:~#
```

2. Each system

System 1:

IP Address/Subnet Mask: 192.168.1.2/24

Network Id: 11000000.10101000.00000001.00000010 ^
11111111.11111111.11111111.00000000 = 11000000.10101000.00000001.00000000 =
192.168.1.0

System 2:

IP Address/Subnet Mask: 192.168.2.2/24

Network Id: 11000000.10101000.00000010.00000010 ^
11111111.11111111.11111111.00000000 = 11000000.10101000.00000010.00000000 =
192.168.2.0

System 3:

IP Address/Subnet Mask: 192.163.2.4/22

Network Id: 11000000.10100011.00000010.00000100 ^
11111111.11111111.11111100.00000000 = 11000000.10100011.00000000.00000000 =
192.163.0.0

System 4:

IP Address/Subnet Mask: 192.168.2.5/24

Network Id: 11000000.10101000.00000010.00000101 ^
11111111.11111111.11111111.00000000 = 11000000.10101000.00000010.00000000 =
192.168.2.0

System 5:

IP Address/Subnet Mask: 192.168.1.5/24

Network Id: 11000000.10101000.00000001.00000101 ^
11111111.11111111.11111111.00000000 = 11000000.10101000.00000001.00000000 =
192.168.1.0

System 6:

IP Address/Subnet Mask: 192.168.2.5/24

Network Id: 11000000.10101000.00000010.00000101^
11111111.11111111.11111111.00000000 = 11000000.10101000.00000010.00000000 =
192.168.2.0

System 7:

IP Address/Subnet Mask: 192.163.2.4/22

Network Id: 11000000.10100011.00000010.00000100 ^
11111111.11111111.11111100.00000000 = 11000000.10100011.00000000.00000000 =
192.163.0.0

System 8:

IP Address/Subnet Mask: 192.168.4.5/24

Network Id: 11000000.10101000.00000100.00000101 ^
11111111.11111111.11111111.00000000 = 11000000.10101000.00000100.00000000 =
192.168.4.0

Based on all of the network ids of all the systems, systems 1 and 5 can communicate with each other since they both have matching network ids but they can't communicate with other systems since they don't have matching network ids. Systems 2, 4 and 6 can communicate with each other since they both have matching network ids but they can't communicate with other systems since they don't have matching network ids. Systems 3 and 7 can communicate with each other since they both have matching network ids but they can't communicate with other systems since they don't have matching network ids. System 8 can't communicate with any of the devices since they don't have any matching network ids with other systems.

| | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 |
|----|----|----|----|----|----|----|----|----|
| S1 | Y | N | N | N | Y | N | N | N |
| S2 | N | Y | N | Y | N | Y | N | N |
| S3 | N | N | Y | N | N | N | Y | N |
| S4 | N | Y | N | Y | N | Y | N | N |
| S5 | Y | N | N | N | Y | N | N | N |
| S6 | N | Y | N | Y | N | Y | N | N |
| S7 | N | N | Y | N | N | N | Y | N |
| S8 | N | N | N | N | N | N | N | N |

Question 4

In this question, we will configure a Cisco router to function as the DHCP server for the network in prior questions.

Toward this goal,

Next, change all systems to use DHCP. That is, edit the /etc/network/interface file on each system as follows:

Comment out these lines corresponding to static configuration

```
# Static config for eth0
```

```
#auto eth0
```

```
#iface eth0 inet static
```

```
#    address 192.168.0.2
```

```
#    netmask 255.255.255.0
```

```
#    gateway 192.168.0.1
```

```
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf  
  
# DHCP config for eth0  
  
##### Uncomment the lines below #####  
  
#auto eth0  
  
#iface eth0 inet dhcp  
  
#    hostname UbuntuDockerGuest-1
```

Add a 7200 Cisco router to your topology. Set the IP of its interface FastEthernet 0/0 to 192.168.1.1 with a /24 mask. Enable the interface to act as a DHCP server that assigns IP addresses in the 192.168.1.0/24 subnet and behaves as the gateway for the network, as explained in class.

Restart all systems and show the screenshots proving that all systems have received an IP address from the router

Screenshots for this question

```
R1
to administratively down
*Feb  1 03:29:11.927: %LINK-5-CHANGED: Interface FastEthernet3/1, changed state
to administratively down
*Feb  1 03:29:11.927: %LINK-5-CHANGED: Interface FastEthernet4/0, changed state
to administratively down
*Feb  1 03:29:11.927: %LINK-5-CHANGED: Interface FastEthernet4/1, changed state
to administratively down
*Feb  1 03:29:11.927: %LINK-5-CHANGED: Interface FastEthernet5/0, changed state
to administratively down
R1#
R1#configu
R1#configure ter
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#inter
R1(config)#interface Fast
R1(config)#interface FastEthernet 0
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip
R1(config-if)#ip ad
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#end
R1#
```

```
root@LinuxHost4: ~
GNU nano 7.2          /etc/network/interfaces
#
# This is a sample network config, please uncomment lines to configure the netw>
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*
#
# Static config for eth0
#auto eth0
#iface eth0 inet static
#    address 192.168.2.5
#    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf
#
# DHCP config for eth0
auto eth0
iface eth0 inet dhcp
    hostname LinuxHost4

[ Read 19 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut      ^T Execute   ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste    ^J Justify   ^/ Go To Line
```

LinuxHost5

```
GNU nano 7.2          /etc/network/interfaces
#
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*
#
# Static config for eth0
#auto eth0
#iface eth0 inet static
#    address 192.168.1.5
#    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf
#
# DHCP config for eth0
auto eth0
iface eth0 inet dhcp
    hostname LinuxHost5

[ Read 19 lines ]
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost6: ~

```
GNU nano 7.2          /etc/network/interfaces *
#
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*
#
# Static config for eth0
#auto eth0
#iface eth0 inet static
#    address 192.168.2.5
#    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf
#
# DHCP config for eth0
auto eth0
iface eth0 inet dhcp
    hostname LinuxHost6
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost6: ~

```
GNU nano 7.2          /etc/network/interfaces *
```

```
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

# Static config for eth0
#auto eth0
#iface eth0 inet static
#    address 192.168.2.5
#    netmask 255.255.255.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
auto eth0
iface eth0 inet dhcp
    hostname LinuxHost6
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^L Replace ^U Paste ^J Justify ^/ Go To Line

root@LinuxHost7: ~

```
GNU nano 7.2          /etc/network/interfaces *
```

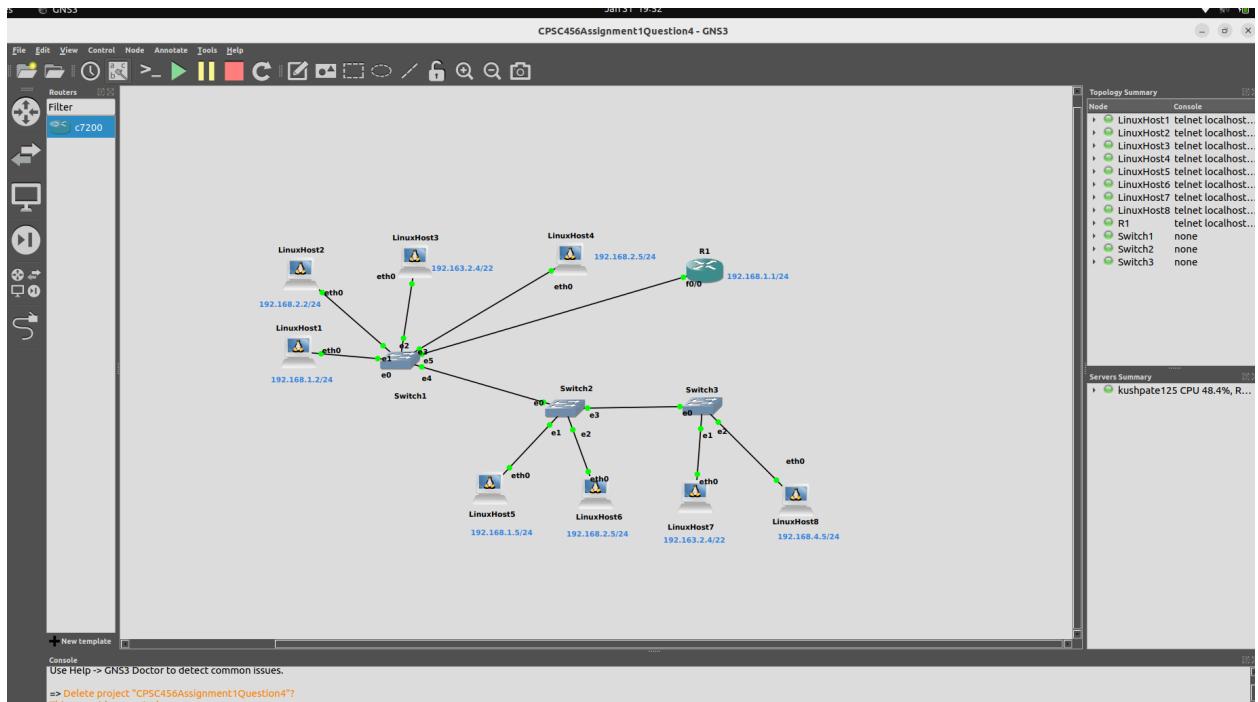
```
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

# Static config for eth0
#auto eth0
#iface eth0 inet static
#    address 192.163.2.4
#    netmask 255.255.252.0
#    gateway 192.168.0.1
#    up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
auto eth0
iface eth0 inet dhcp
    hostname LinuxHost7
```

Save modified buffer?

Y Yes N No Cancel



```
root@LinuxHost1: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost1 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending discover
udhcpc: sending select for 192.168.1.100
udhcpc: lease of 192.168.1.100 obtained, lease time 172800
root@LinuxHost1:~#
```

```
root@LinuxHost2: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost2 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending discover
udhcpc: sending select for 192.168.1.101
udhcpc: lease of 192.168.1.101 obtained, lease time 172800
root@LinuxHost2:~#
```

```
root@LinuxHost3: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost3 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.1.102
udhcpc: lease of 192.168.1.102 obtained, lease time 172800
root@LinuxHost3:~#
```

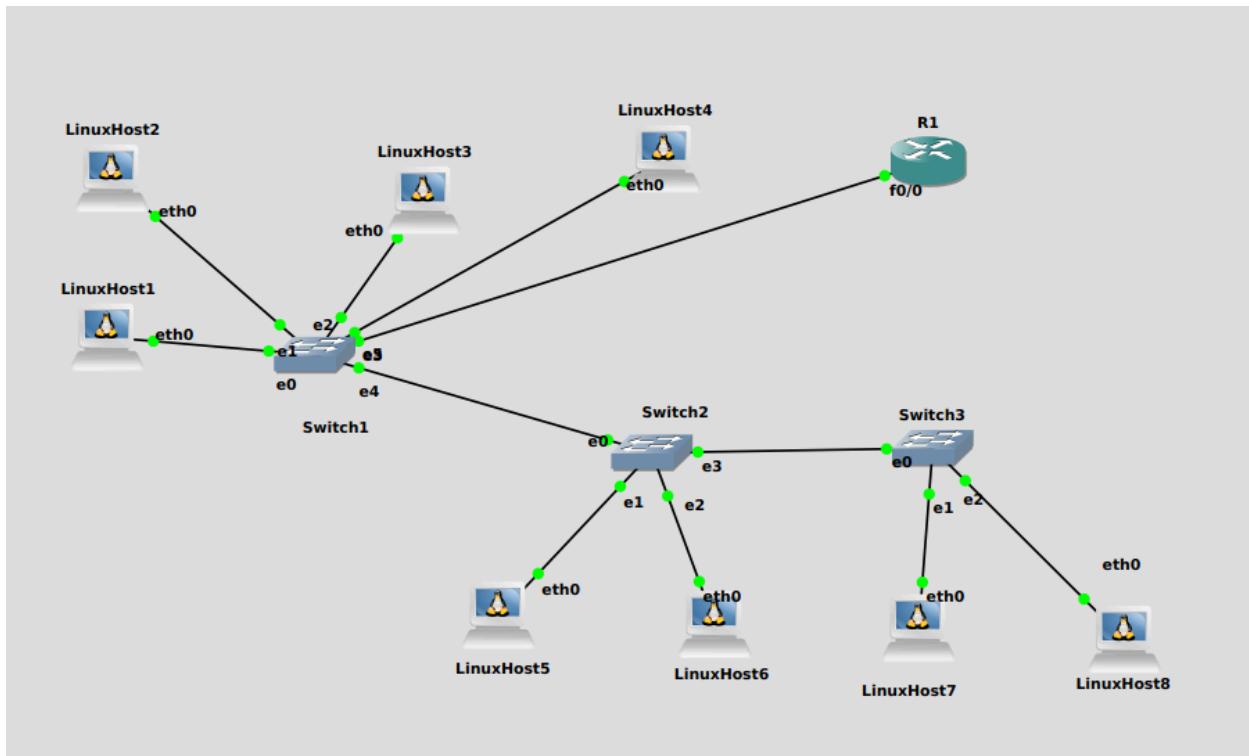
```
root@LinuxHost4: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost4 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.1.103
udhcpc: lease of 192.168.1.103 obtained, lease time 172800
root@LinuxHost4:~#
```

```
root@LinuxHost5: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost5 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.1.104
udhcpc: lease of 192.168.1.104 obtained, lease time 172800
root@LinuxHost5:~#
```

A screenshot of a terminal window titled "root@LinuxHost6: ~". The window has a dark purple background and a light purple header bar. The title bar includes standard window control icons (minimize, maximize, close) and a search icon. The terminal output is displayed in white text:

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost6 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.1.105
udhcpc: lease of 192.168.1.105 obtained, lease time 172800
root@LinuxHost6:~#
```

```
root@LinuxHost8: ~
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost8 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.1.106
udhcpc: lease of 192.168.1.106 obtained, lease time 172800
root@LinuxHost8:~#
```



```
*Feb  3 07:07:33.127: %LINK-5-CHANGED: Interface FastEthernet4/1, changed state  
to administratively down  
*Feb  3 07:07:33.127: %LINK-5-CHANGED: Interface FastEthernet5/0, changed state  
to administratively down  
R1#  
R1#configure  
R1#configure ter  
R1#configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#interface fast  
R1(config)#interface fastEthernet 0/0  
R1(config-if)#ip address  
R1(config-if)#ip address 192.168.1.1 255.255.255.0  
R1(config-if)#no shutdown  
R1(config-if)#end  
R1#  
*Feb  3 07:08:50.795: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t  
o up  
R1#  
*Feb  3 07:08:50.939: %SYS-5-CONFIG_I: Configured from console by console  
R1#  
*Feb  3 07:08:51.795: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEtherne  
t0/0, changed state to up  
R1#configure █
```

```
root@LinuxHost1:~  ×          R1  ×          R1  ×  ✓  
R1(config-if)#no shutdown  
R1(config-if)#end  
R1#  
*Feb  3 07:08:50.795: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t  
o up  
R1#  
*Feb  3 07:08:50.939: %SYS-5-CONFIG_I: Configured from console by console  
R1#  
*Feb  3 07:08:51.795: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEtherne  
t0/0, changed state to up  
R1#configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#service dhcp  
R1(config)#ip dhcp pool MYDHCPPOOL  
R1(dhcp-config)#netw  
R1(dhcp-config)#network 192  
R1(dhcp-config)#network 192.168.1.0 255.255.255.0  
R1(dhcp-config)#lease 2  
R1(dhcp-config)#default-  
R1(dhcp-config)#default-router 192.168.1.1  
R1(dhcp-config)#end  
R1#  
*Feb  3 07:11:23.563: %SYS-5-CONFIG_I: Configured from console by console  
R1#6cxtleorrmp256color █
```

```
root@LinuxHost1:~ x R1 x R1 x v
*Feb 3 07:07:33.123: %LINK-5-CHANGED: Interface FastEthernet2/1, changed state
to administratively down
*Feb 3 07:07:33.123: %LINK-5-CHANGED: Interface FastEthernet3/0, changed state
to administratively down
*Feb 3 07:07:33.123: %LINK-5-CHANGED: Interface FastEthernet3/1, changed state
to administratively down
*Feb 3 07:07:33.123: %LINK-5-CHANGED: Interface FastEthernet4/0, changed state
to administratively down
*Feb 3 07:07:33.127: %LINK-5-CHANGED: Interface FastEthernet4/1, changed state
to administratively down
*Feb 3 07:07:33.127: %LINK-5-CHANGED: Interface FastEthernet5/0, changed state
to administratively down
R1#
R1#configure
R1#configure ter
R1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#interface fast
R1(config)#interface fastEthernet 0/0
R1(config-if)#ip address
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#end
R1#
```

When I restarted the systems all the screenshots showed that all devices are connected to the DHCP server

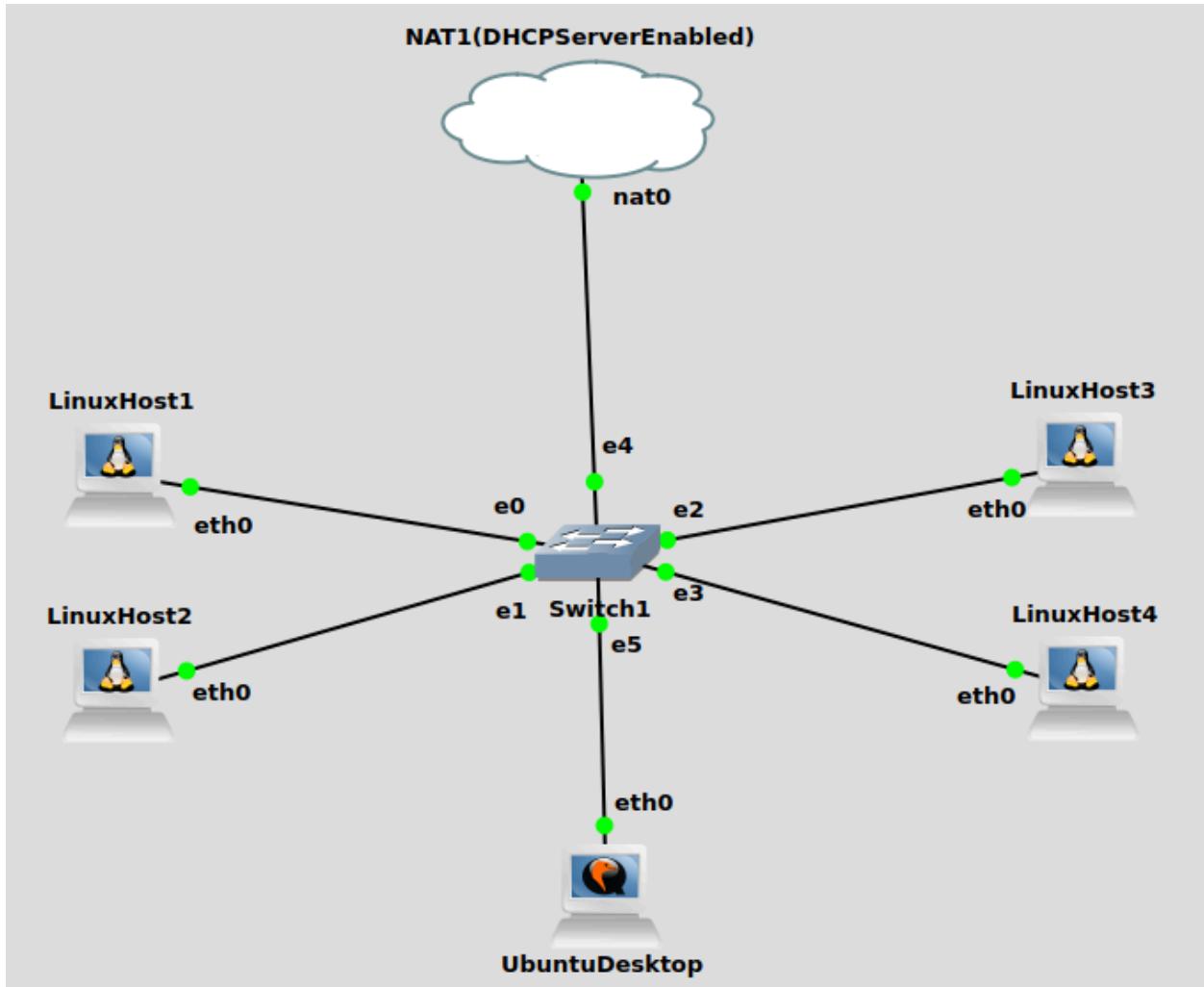
Question 5

Netcat is a network utility that lets us set up a server on one system and a client on another system. It is often used for creating a reverse shell on a compromised system in network and penetration testing.

Please follow these steps:

Build a topology like the one below. The NAT node is a GNS3 device that simulates a NAT router and a DHCP server (Don't forget to enable DHCP on all

systems. The UbuntuDesktop may take some time to boot. The password is osboxes.org).



On LinuxHost1 and LinuxHost2 run the following commands to install the netcat commands:

```
# apt update // This will update the package system in Linux to allow us to install applications from the Ubuntu repositories.
```

```
# apt install netcat-traditional // Install the netcat utility. Answer "Y" when prompted.
```

Create a TCP server listening on port 12345 of LinuxHost1:

```
# netcat -l -p 12345
```

Next, run the following command on LinuxHost2:

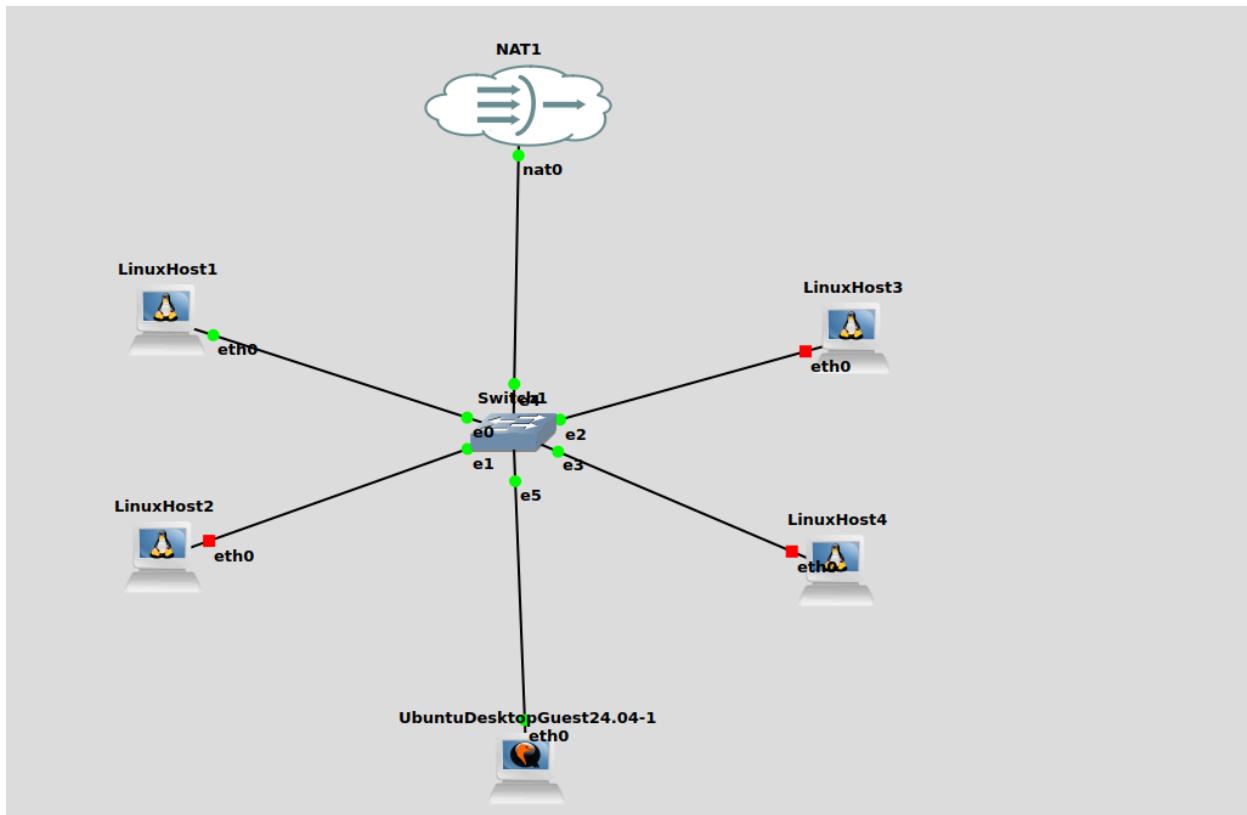
```
# netcat <IP ADDRESS OF HOST 1> 12345 // Connect to the Linux server at  
the provided IP and port
```

Next, type some text and press enter. You should see the next appear on the server prompt of LinuxHost2.

Submit the screenshots to prove what you have done.

Please save this topology -- you will need it for the nmap lab

Screenshots for this question



```

edit View Control Node Annotate Tools Help
root@LinuxHost1: ~
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [774 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages [19.3 MB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [15.5 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [784 kB]
Get:12 http://archive.ubuntu.com/ubuntu noble/main amd64 Packages [1808 kB]
Get:13 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1302 kB]
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [20.1 kB]
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1064 kB]
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [799 kB]
Get:17 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [15.1 kB]
Fetched 28.0 MB in 3s (8091 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
27 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@LinuxHost1:~#

```

```
edit View Control Node Annotate Tools Help
root@LinuxHost1: ~
Setting up libcurl4t64:amd64 (8.5.0-2ubuntu10.6) ...
Setting up libnl-route-3-200:amd64 (3.7.0-0.3build1.1) ...
Setting up curl (8.5.0-2ubuntu10.6) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
root@LinuxHost1:~# apt install netcat-traditional
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  netcat-traditional
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 61.3 kB of archives.
After this operation, 138 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu noble/universe amd64 netcat-traditional amd64 1.10-48 [61.3 kB]
Fetched 61.3 kB in 0s (136 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package netcat-traditional.
(Reading database ... 6117 files and directories currently installed.)
Preparing to unpack .../netcat-traditional_1.10-48_amd64.deb ...
Unpacking netcat-traditional (1.10-48) ...
Setting up netcat-traditional (1.10-48) ...
update-alternatives: using /bin/nc.traditional to provide /bin/nc (nc) in auto mode
```

```
edit View Control Node Annotate Tools Help
root@LinuxHost1: ~
LinuxHost1 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.170
udhcpc: lease of 192.168.122.170 obtained, lease time 3600
root@LinuxHost1:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.122.170  netmask 255.255.255.0  broadcast 192.168.122.255
        ether 02:42:d4:97:4f:00  txqueuelen 1000  (Ethernet)
          RX packets 4  bytes 801 (801.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 5  bytes 968 (968.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

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 0  bytes 0 (0.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 0  bytes 0 (0.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

root@LinuxHost1:~#
```

```
edit View Control Node Annotate Tools Help
root@LinuxHost2: ~
root@LinuxHost2:~# apt update
Get:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:3 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [774 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:5 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [784 kB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [15.5 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [1038 kB]
Get:8 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages [19.3 MB]
Get:10 http://archive.ubuntu.com/ubuntu noble/restricted amd64 Packages [117 kB]
Get:11 http://archive.ubuntu.com/ubuntu noble/main amd64 Packages [1808 kB]
Get:12 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [331 kB]
Get:13 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1064 kB]
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [20.1 kB]
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [799 kB]
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1
```

```
edit View Control Node Annotate Tools Help
root@LinuxHost2: ~
root@LinuxHost2:~# netcat-traditional
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 61.3 kB of archives.
After this operation, 138 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu noble/universe amd64 netcat-traditional a
md64 1.10-48 [61.3 kB]
Fetched 61.3 kB in 1s (52.6 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package netcat-traditional.
(Reading database ... 6117 files and directories currently installed.)
Preparing to unpack .../netcat-traditional_1.10-48_amd64.deb ...
Unpacking netcat-traditional (1.10-48) ...
Setting up netcat-traditional (1.10-48) ...
update-alternatives: using /bin/nc.traditional to provide /bin/nc (nc) in auto m
ode
update-alternatives: warning: skip creation of /usr/share/man/man1/nc.1.gz becau
se associated file /usr/share/man/man1/nc.traditional.1.gz (of link group nc) do
esn't exist
update-alternatives: warning: skip creation of /usr/share/man/man1/netcat.1.gz b
ecause associated file /usr/share/man/man1/nc.traditional.1.gz (of link group nc
) doesn't exist
root@LinuxHost2:~#
```

```
edit View Control Node Annotate Tools Help
root@LinuxHost1: ~
root@LinuxHost1: ~ x
root@LinuxHost2: ~ x v

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost1 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.170
udhcpc: lease of 192.168.122.170 obtained, lease time 3600
root@LinuxHost1:~# netcat -l -p 12345
```

```
edit View Control Node Annotate Tools Help
root@LinuxHost2: ~
root@LinuxHost1: ~ x
root@LinuxHost2: ~ x v

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
LinuxHost2 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.232
udhcpc: lease of 192.168.122.232 obtained, lease time 3600
root@LinuxHost2:~# netcat 192.168.122.170 12345
hello
```

The screenshot shows a terminal window with two tabs. The active tab is 'root@LinuxHost2: ~'. The other tab is 'root@LinuxHost1: ~'. The terminal background is dark.

```
root@LinuxHost1:~          x          root@LinuxHost2:~ x v
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost2 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.232
udhcpc: lease of 192.168.122.232 obtained, lease time 3600
root@LinuxHost2:~# netcat 192.168.122.170 12345
hello
```

The screenshot shows a terminal window with two tabs open. The left tab, titled 'root@LinuxHost1: ~', contains the following text:

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost1 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.170
udhcpc: lease of 192.168.122.170 obtained, lease time 3600
root@LinuxHost1:~# netcat -l -p 12345
hello
```

The right tab, titled 'root@LinuxHost2: ~', is currently inactive.

The screenshot shows a terminal window with two tabs. The active tab is 'root@LinuxHost2: ~'. The other tab is 'root@LinuxHost1: ~'. The terminal background is dark purple.

```
root@LinuxHost1:~          x          root@LinuxHost2:~          x          ^
```

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost2 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.232
udhcpc: lease of 192.168.122.232 obtained, lease time 3600
root@LinuxHost2:~# netcat 192.168.122.170 12345
hello
how are you
```

The screenshot shows a terminal window with two tabs open. The left tab, titled "root@LinuxHost1:~", displays the output of a "netcat" command on host 1. The right tab, titled "root@LinuxHost2:~", shows the interaction between the two hosts. Host 2 has sent a "hello" message to host 1.

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost2 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.232
udhcpc: lease of 192.168.122.232 obtained, lease time 3600
root@LinuxHost2:~# netcat 192.168.122.170 12345
hello
how are you
good
```

The screenshot shows a terminal window with two tabs open. The left tab, titled "root@LinuxHost1: ~", contains the following text:

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost1 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.170
udhcpc: lease of 192.168.122.170 obtained, lease time 3600
root@LinuxHost1:~# netcat -l -p 12345
hello
how are you
good
```

The right tab, titled "root@LinuxHost2: ~", is currently inactive.

The screenshot shows two terminal windows side-by-side. The left window, titled 'root@LinuxHost1: ~', has a command prompt and some system logs. The right window, titled 'root@LinuxHost2: ~', also has a command prompt and shows the output of the 'netcat' command. Both windows have a dark background and light-colored text.

```
root@LinuxHost1:~ Trying 127.0.0.1... Connected to localhost. Escape character is '^]'. LinuxHost2 console is now available... Press RETURN to get started. udhcpc: started, v1.30.1 udhcpc: sending discover udhcpc: sending select for 192.168.122.232 udhcpc: lease of 192.168.122.232 obtained, lease time 3600 root@LinuxHost2:~# netcat 192.168.122.170 12345 hello how are you good
```

Based on the screenshots it shows what ever I type in the command prompt in one terminal will show up in another terminal, they both send and receive the messages from each other via the same port number of 12345 but they have different IP Addresses, the computer command "netcat -i -p 12345" ,shows that the computer of 192.168.122.170 leading the communication but the computer with the command "netcat 192.168.122.170 12345" connects to the computer of ip address 192.168.122.170 and form a peer to peer communication architecture. There is communication happening between two different computers

Question 6

Next, we will experiment with the ARP poisoning attacks discussed in class. You will use the Ubuntu Desktop system to spy on the traffic between LinuxHost1 and LinuxHost2 from the previous problem.

To do this, view the console for the Ubuntu Desktop system. Please note: if the system runs way too slow, you can try shutting it down the VM, then right-clicking on the VM, choosing "Configure" and then in the "RAM" textbox increasing the amount of memory for the system and powering it up again.

Next, open the terminal in Ubuntu Desktop and run the following commands to install ettercap -- a tool for conducting ARP spoofing attacks and Wireshark, which is a packet sniffer so we can see the traffic:

```
# apt install wireshark ettercap-graphical
```

When prompted about whether non-superusers should be allowed to capture packets, choose "Yes".

1. What are the IP addresses of LinuxHost1 and LinuxHost2?
2. Next, we will use ettercap to conduct a MAC spoofing attack such that LinuxHost1 is deceived into believing that the MAC associated with the IP of LinuxHost1 is the MAC of Ubuntu Desktop and to deceive that LinuxHost2 that the IP associated with the MAC of LinuxHost1 is that of the Ubuntu Desktop.

To do this, we will run the following command:

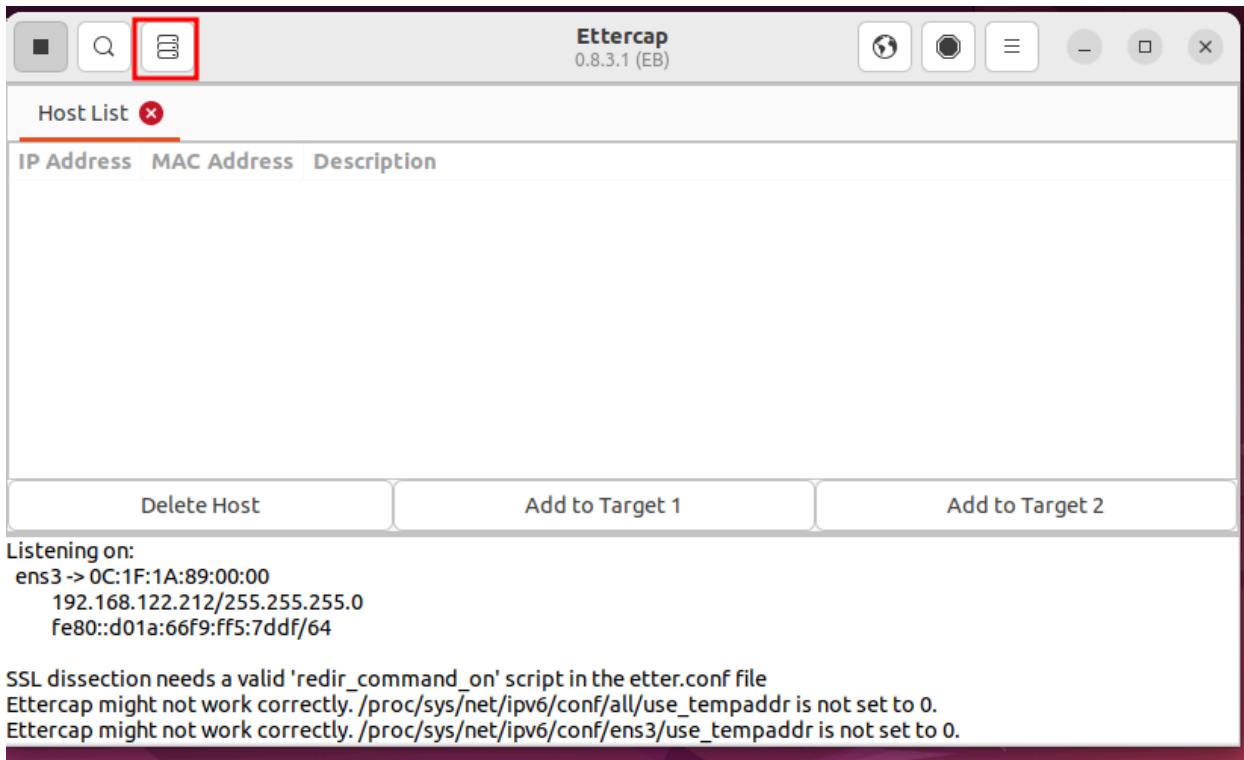
```
# sudo ettercap -G // Starts ettercap in the graphical mode.
```

A screen similar to the below should appear:

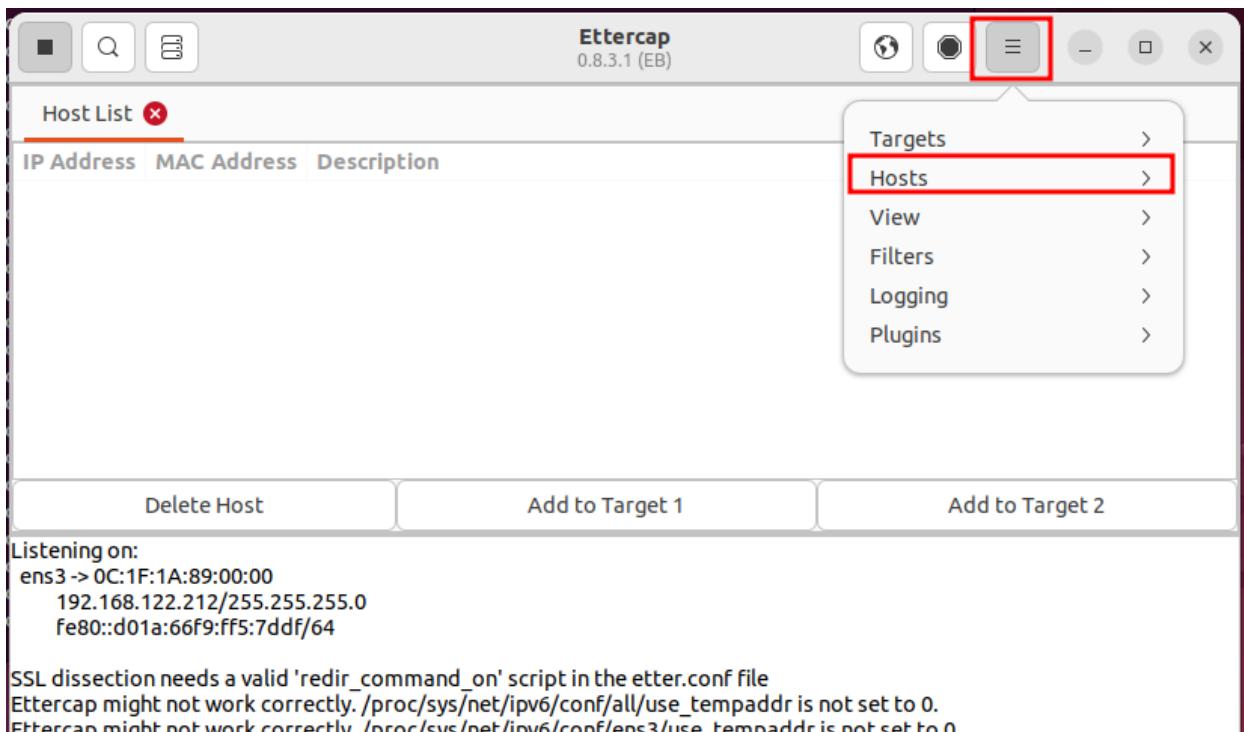


Confirm the default network interface on which to capture traffic (by clicking the checkmark).

Next, click on the "Hosts List" option:

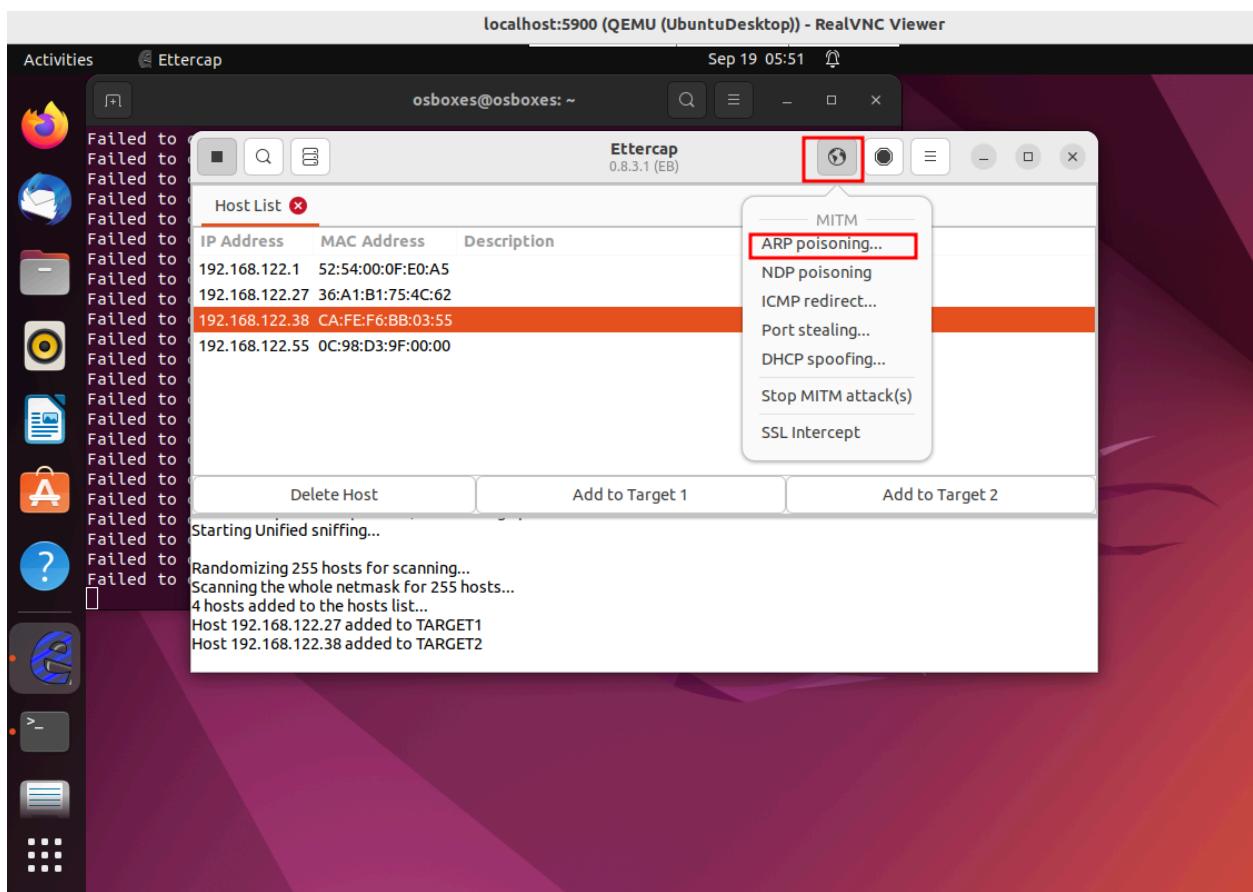


In the Ettercap menu, select "Hosts" --> "Scan for Hosts"



1. From the resulting list that appears, click on the IP and MAC of LinuxHost1 and click on "Add to Target 1" and then select LinuxHost2 and select "Add to Target 2". These will be the systems on which we will use ARP poisoning to achieve a Man-in-the-Middle attack. Please include screenshots of all of your screens.

2. From the MITM attack menu, select "ARP Poisoning":



In the menu that appears, click "Ok". This will start the attack.

3. Next, start Wireshark by running the command "sudo wireshark" in another terminal. Select the same interface that you told ettercap to sniff on (in the above example it was "ens3". We will be using Wireshark to see the data transiting between the two systems.

4. In the Wireshark filter window type "tcp" to filter out all packets except TCP.

5. Repeat the netcat experiment in the last question. You should see multiple captured packets in the Wireshark packet list. It may look similar to this:

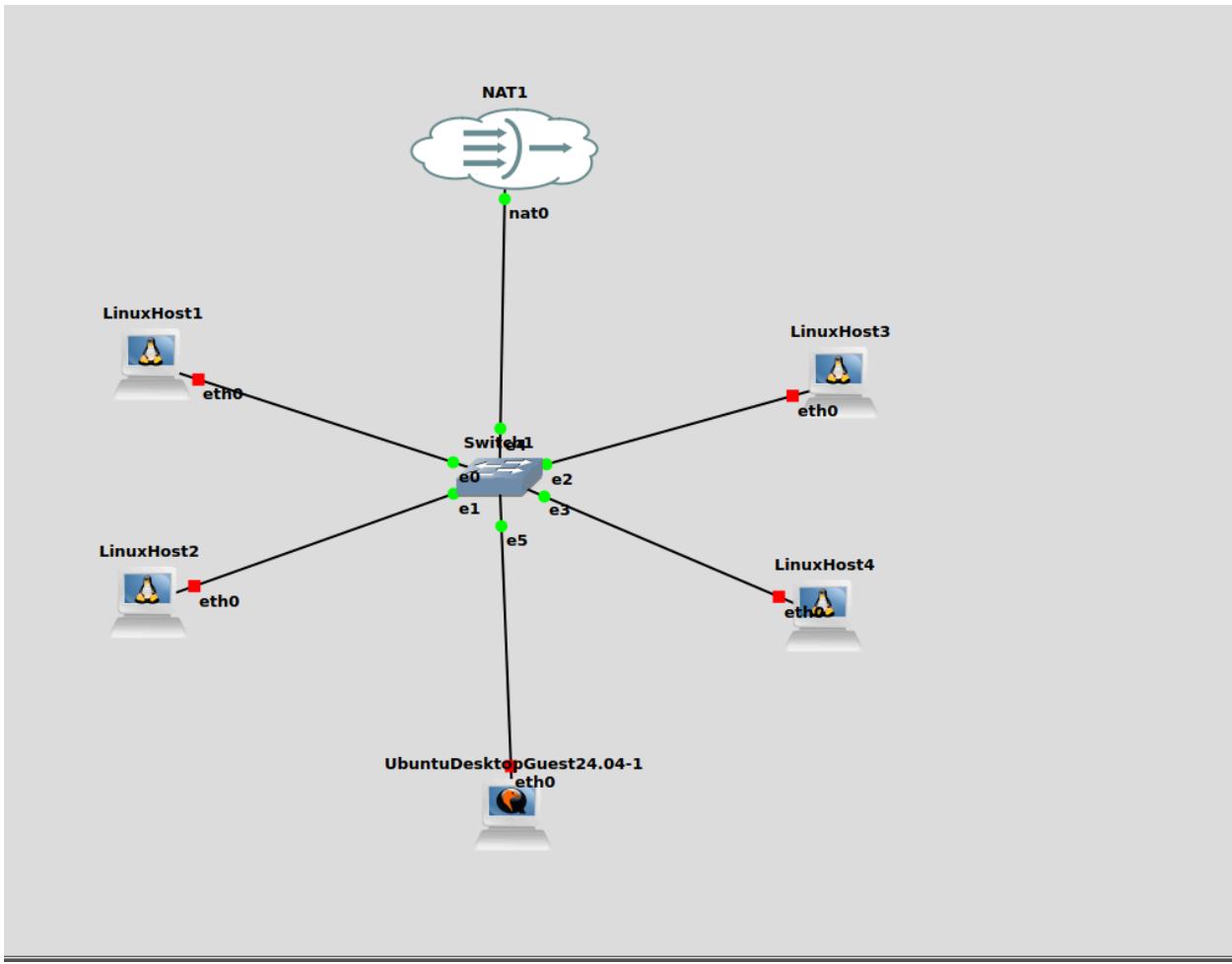
| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|---------------|-----------------|-----------------|----------|--------|--|
| 17 | 22.252423057 | 192.168.122.38 | 192.168.122.27 | TCP | 72 | 12345 → 40406 [PSH, ACK] Seq=1 Ack=1 Win=510 Len=6 T |
| 18 | 22.255457686 | 192.168.122.38 | 192.168.122.27 | TCP | 72 | [TCP Retransmission] 12345 → 40406 [PSH, ACK] Seq=1 |
| 19 | 22.256302451 | 192.168.122.27 | 192.168.122.38 | TCP | 66 | 40406 → 12345 [ACK] Seq=1 Ack=7 Win=502 Len=0 TSval= |
| 20 | 22.283298315 | 192.168.122.27 | 192.168.122.38 | TCP | 66 | [TCP Dup ACK 19#1] 40406 → 12345 [ACK] Seq=1 Ack=7 w |
| 111 | 133.927833093 | 192.168.122.212 | 34.122.121.32 | TCP | 74 | 55556 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK |
| 112 | 133.954259017 | 34.122.121.32 | 192.168.122.212 | TCP | 58 | 80 → 55556 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MS |
| 113 | 133.954347366 | 192.168.122.212 | 34.122.121.32 | TCP | 54 | 55556 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0 |
| 114 | 133.954966322 | 192.168.122.212 | 34.122.121.32 | HTTP | 141 | GET / HTTP/1.1 |

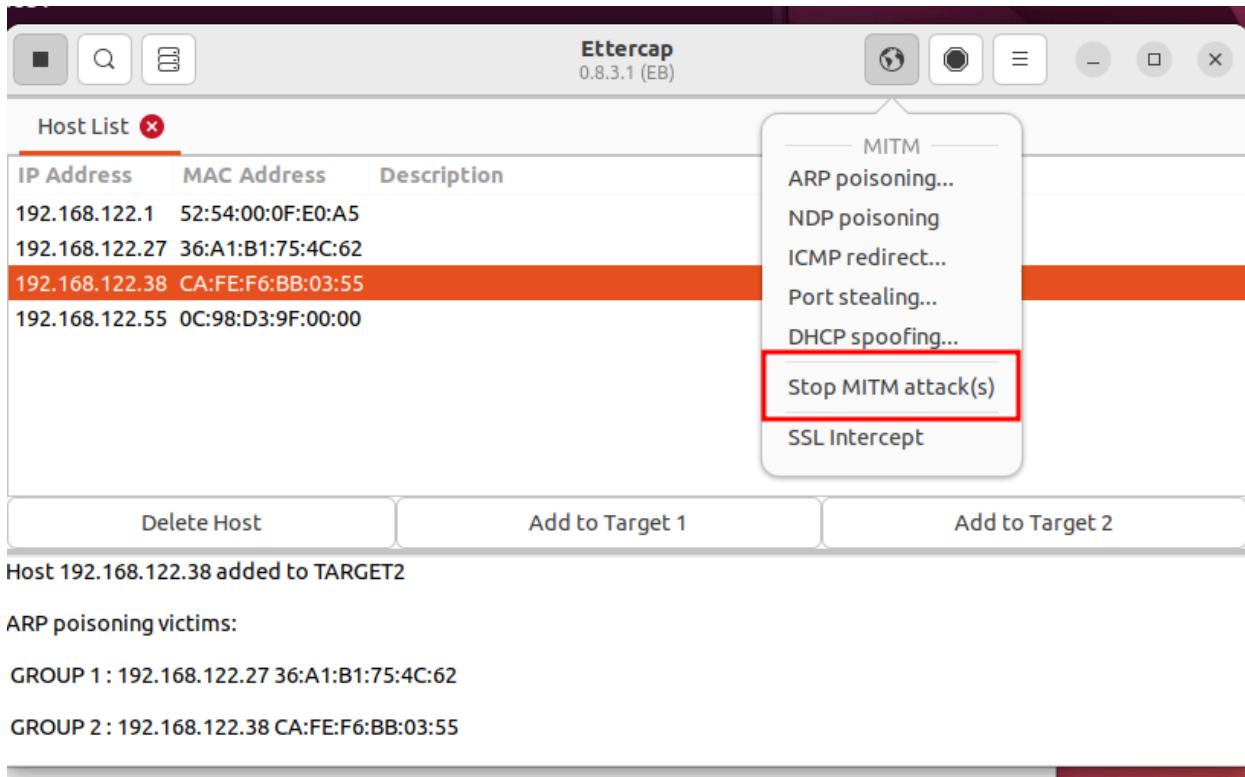
Look through the packets. In the field below, you will see the headers and the contents of the packets. Can you see the messages that are being sent between LinuxHost1 and LinuxHost2 over TCP? Present a screenshot of these packets and their contents.

5. Why does the above happen? In your explanation please include screenshots of arp tables from both victim systems (you can use the arp -n command to view the arp tables).

6. Stop the attack by choosing the "Stop MITM Attacks" from the "MITM menu":

Topology we used





1. The IP Address for LinuxHost1 is 192.168.122.170 and The IP Address for LinuxHost2 is 192.168.122.232

The screenshot shows a terminal window with two tabs open. The left tab, titled "root@LinuxHost1: ~", displays the following log output:

```
Trying 127.0.0.1...
Connected to root@LinuxHost1.
Escape character is '^].
LinuxHost1 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.170
udhcpc: lease of 192.168.122.170 obtained, lease time 3600
root@LinuxHost1:~# netcat -l -p 12345
```

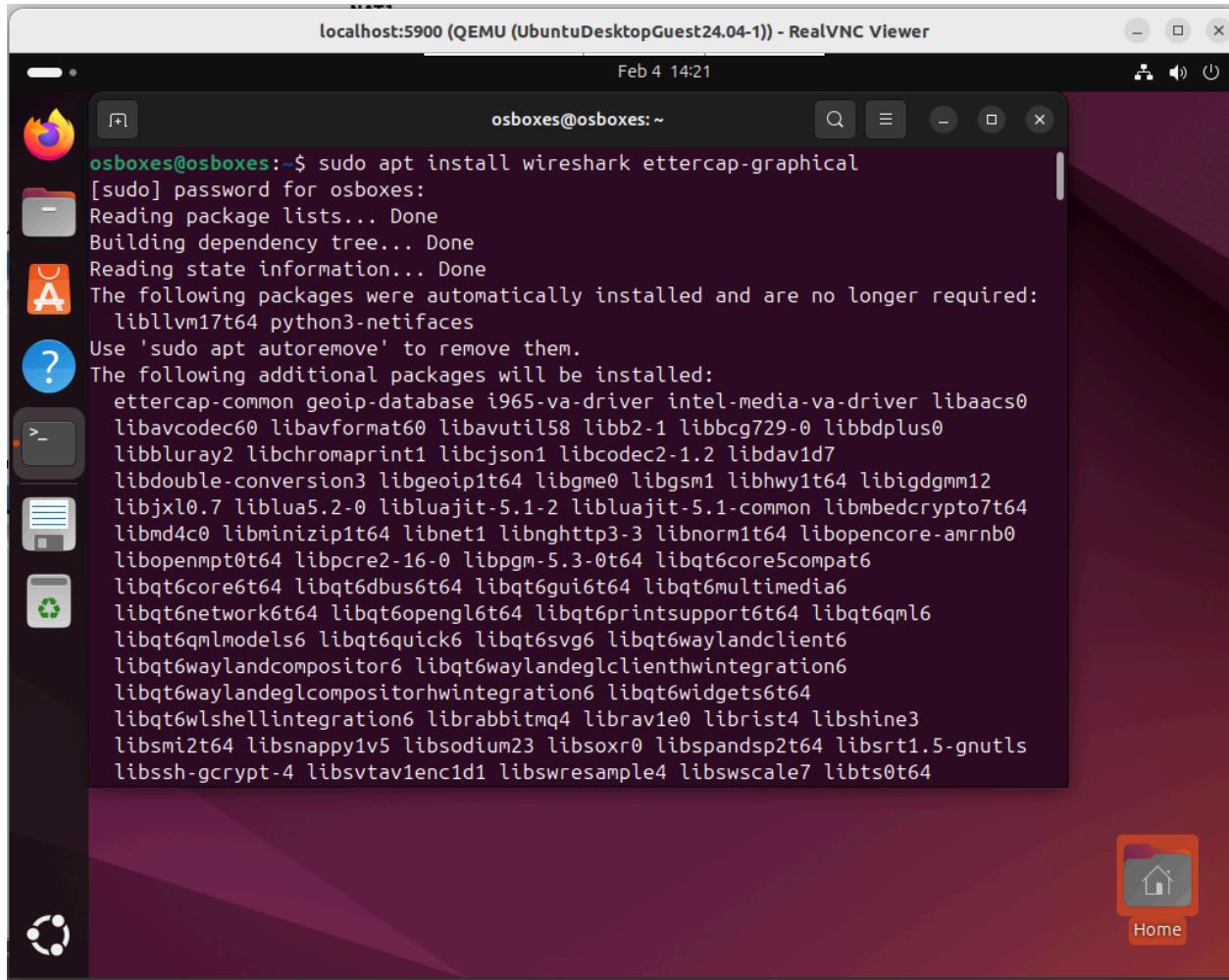
The right tab, titled "root@LinuxHost2: ~", is currently inactive.

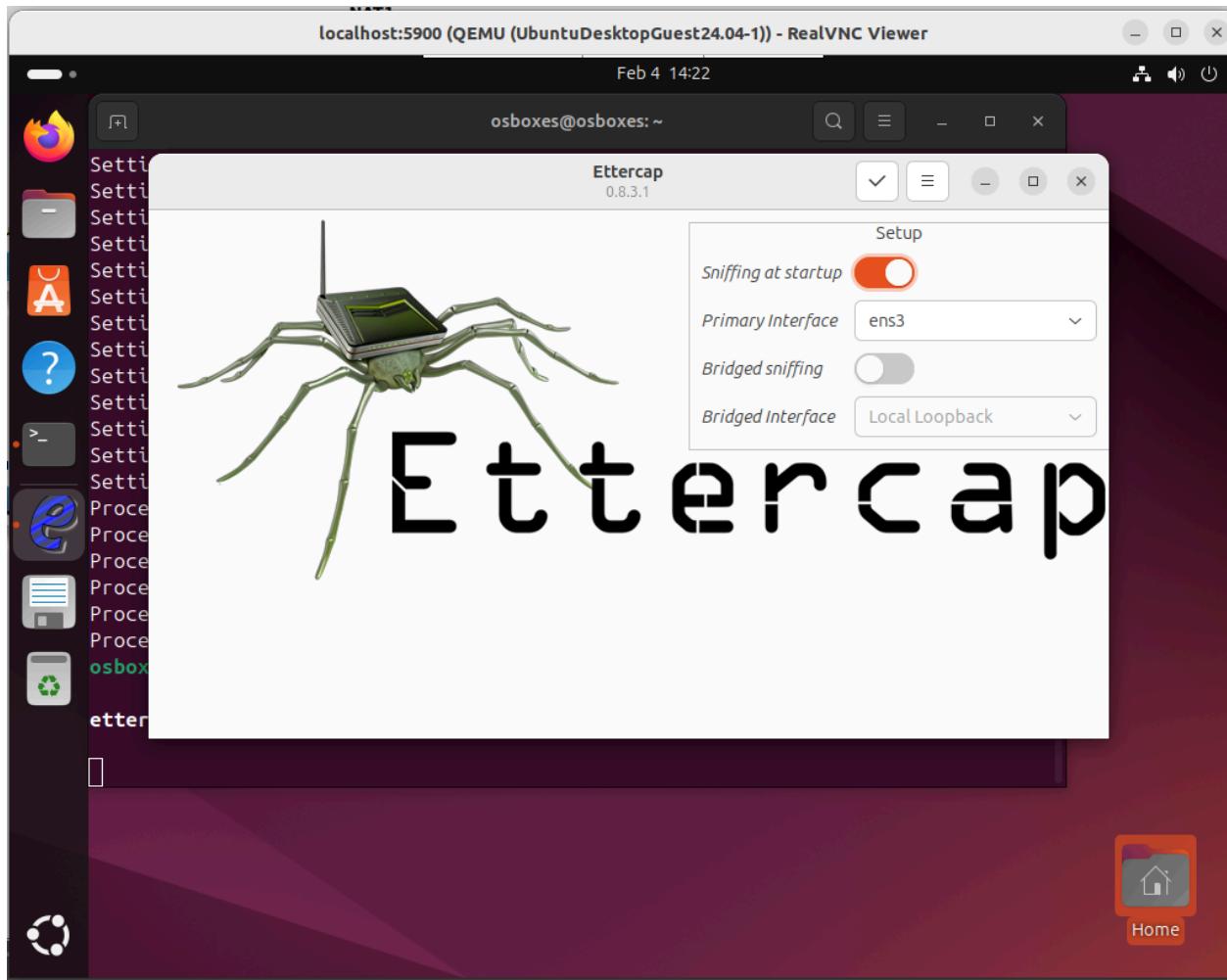
The screenshot shows a terminal window with two tabs open. The left tab is titled "root@LinuxHost1:~" and the right tab is titled "root@LinuxHost2:~". The "root@LinuxHost2:~" tab is active and displays the following text:

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^].
LinuxHost2 console is now available... Press RETURN to get started.
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.168.122.232
udhcpc: lease of 192.168.122.232 obtained, lease time 3600
root@LinuxHost2:~# netcat 192.168.122.170 12345
hello
```

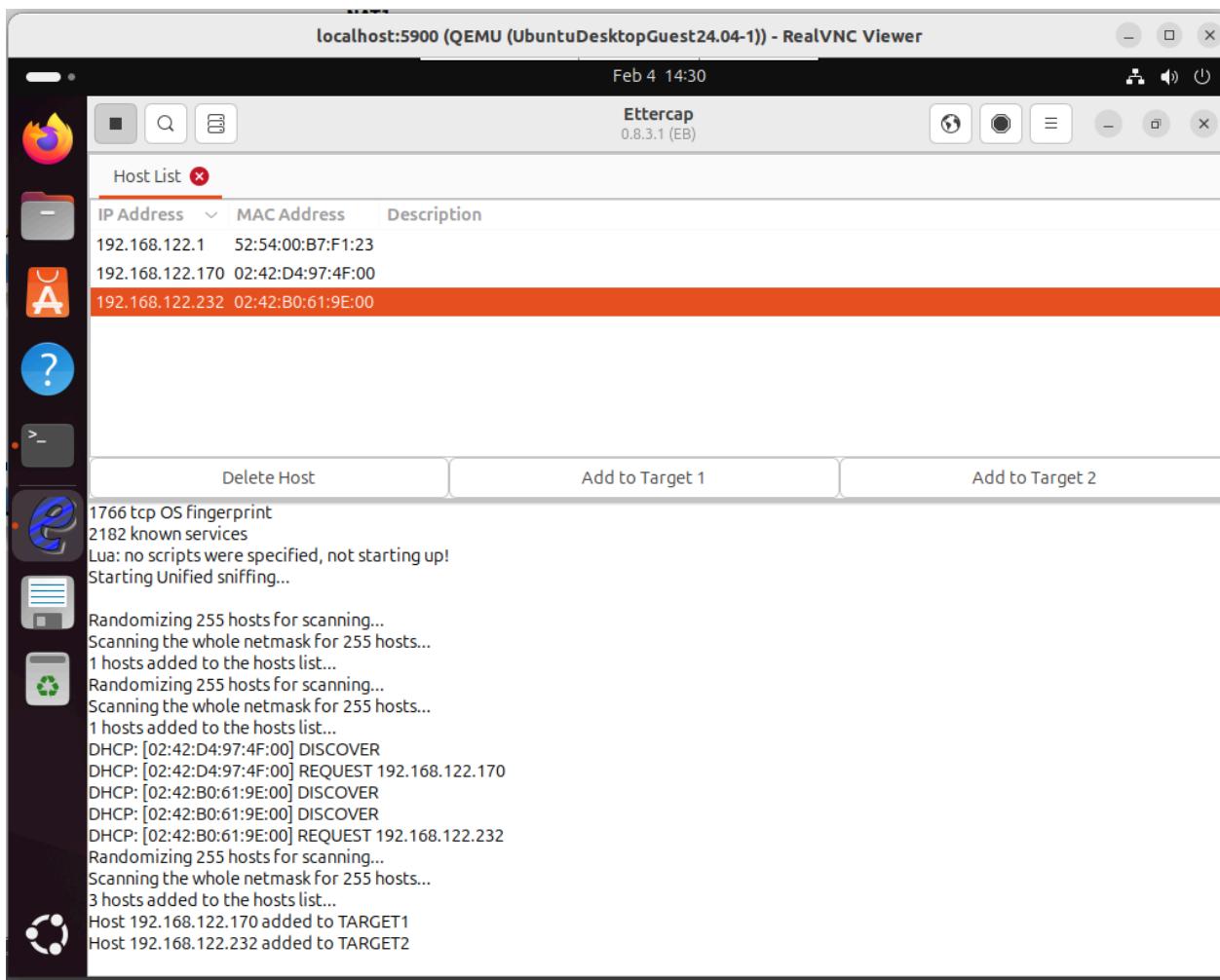
2. Screenshots for this question

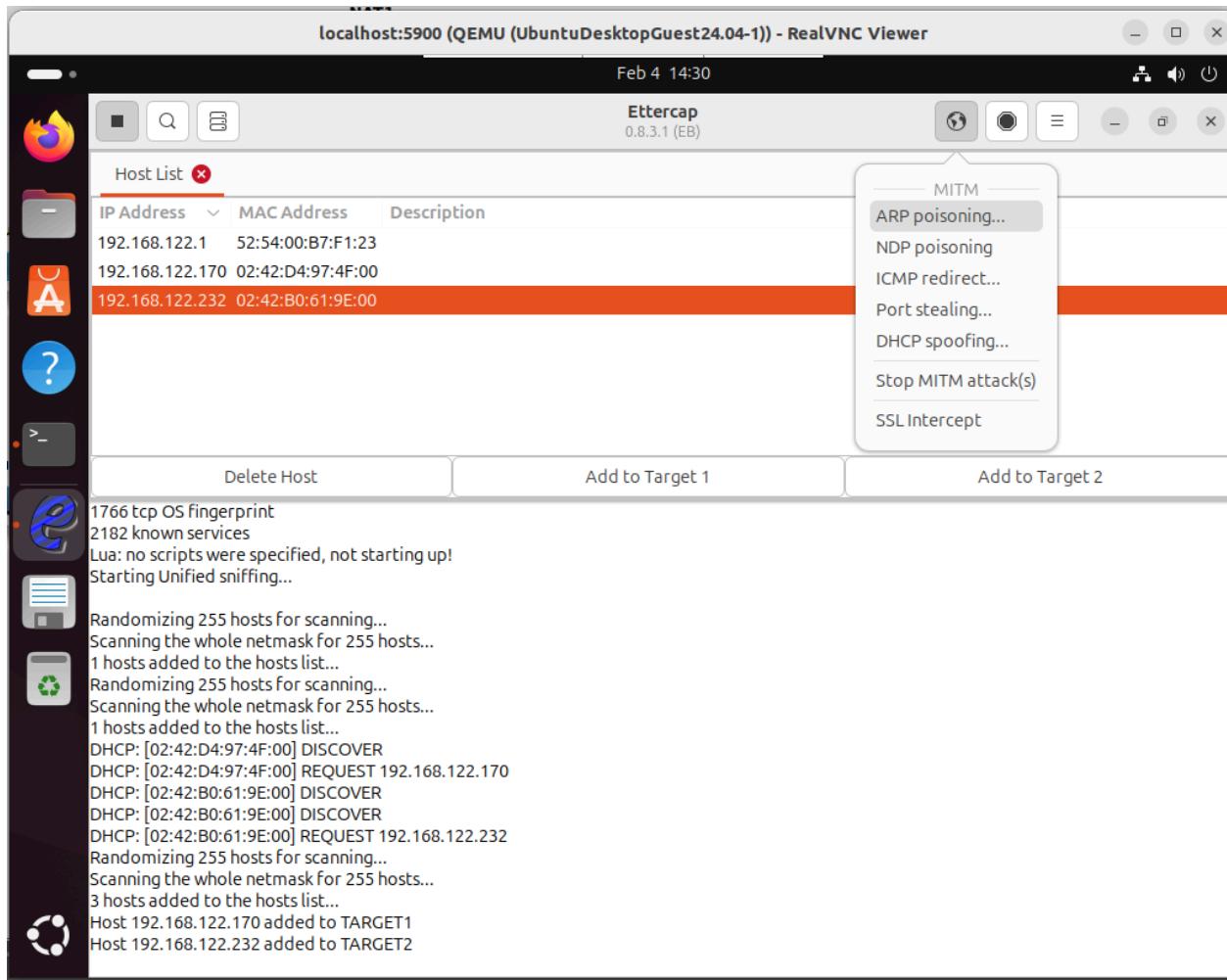
1. Setting up ethercat

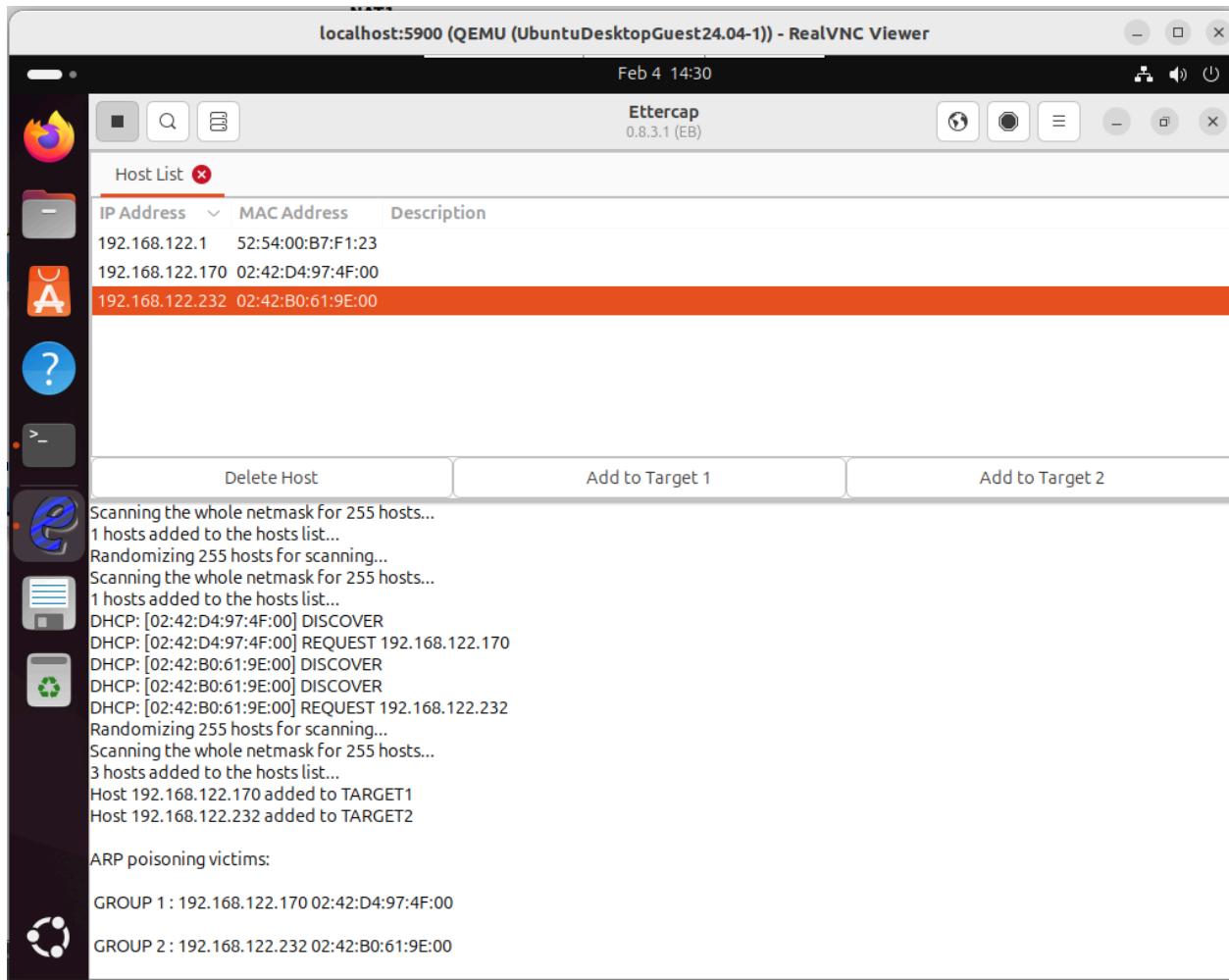




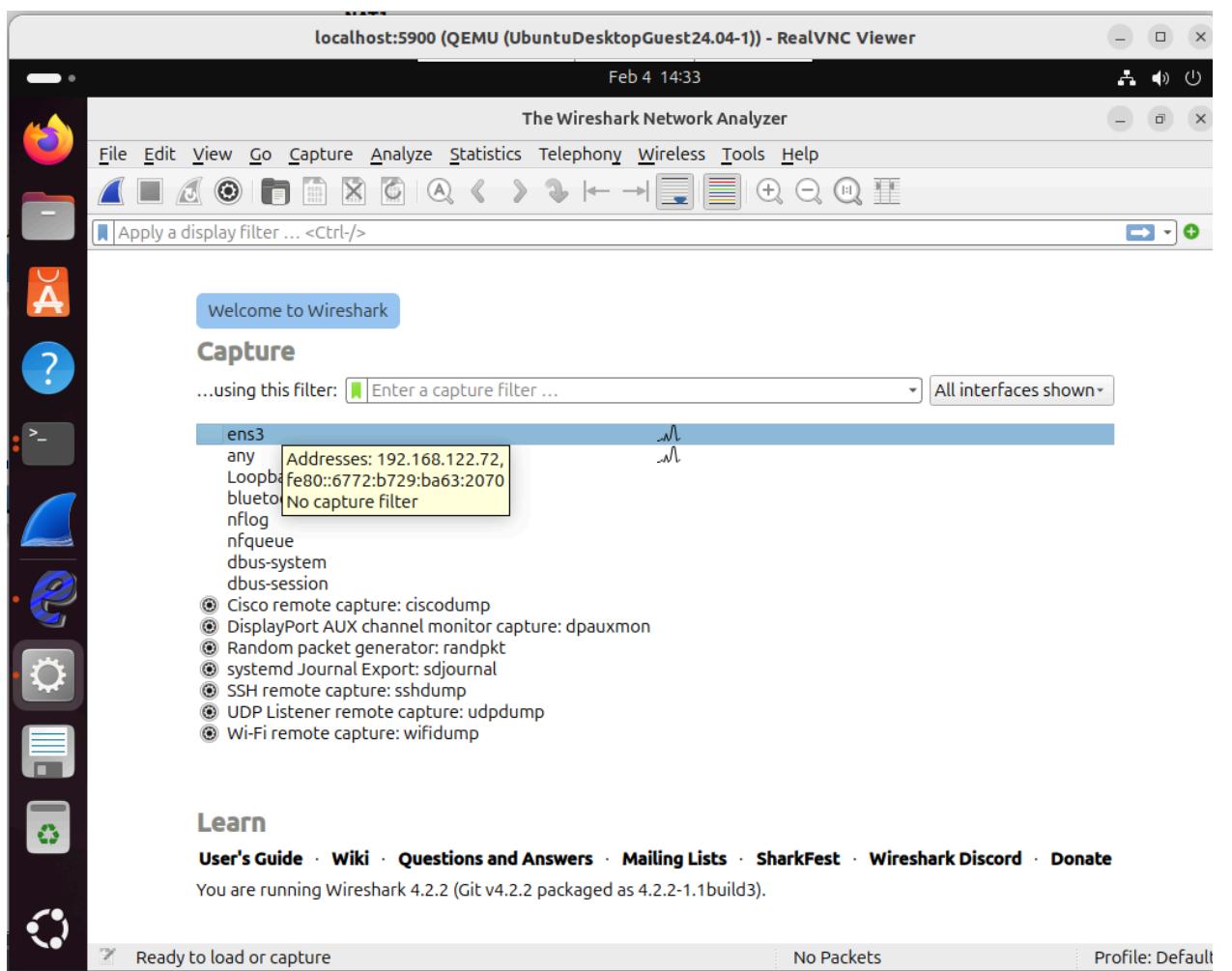
2. Selecting ARP Poisining

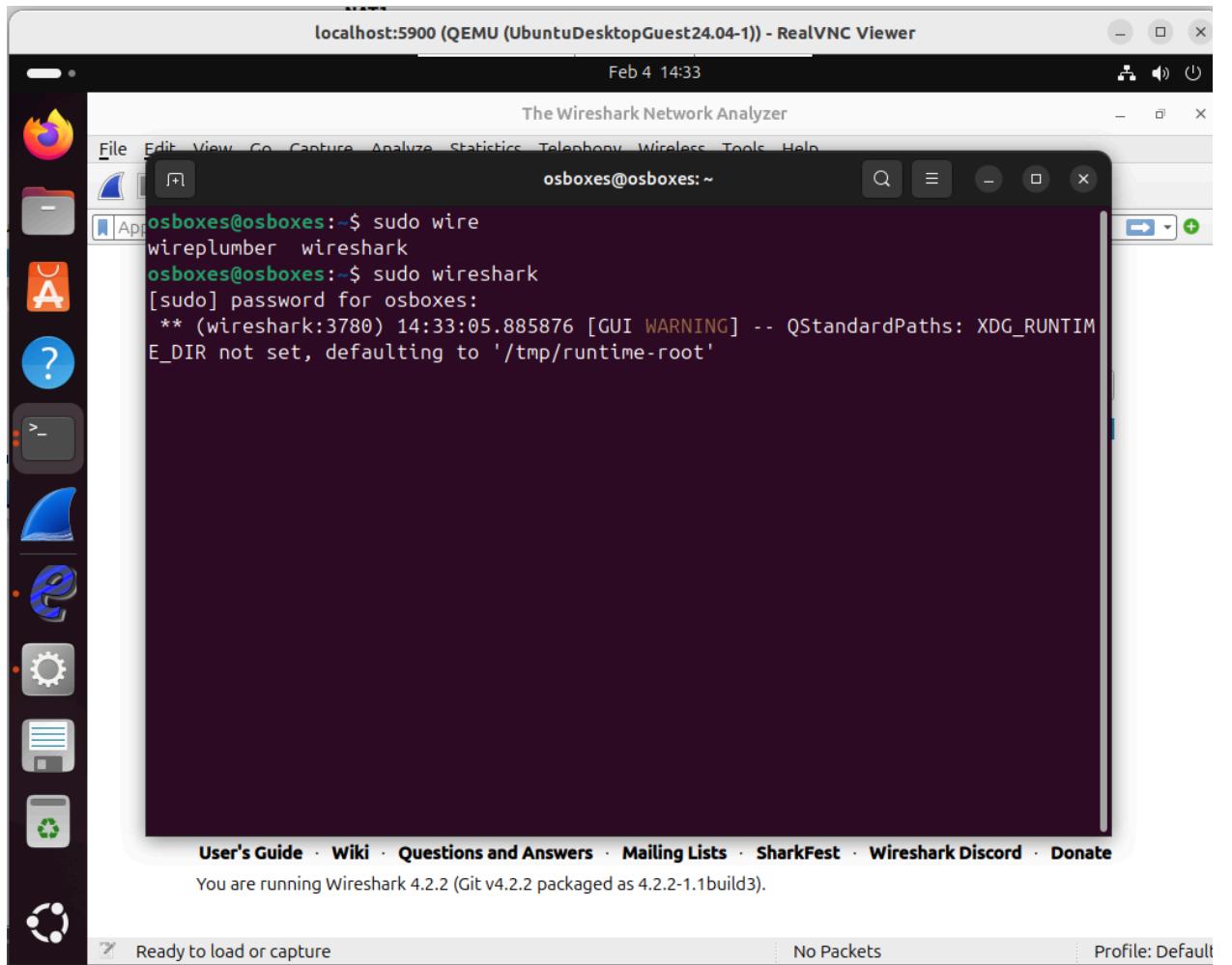






3. Running wireshark





localhost:5900 (QEMU (UbuntuDesktopGuest24.04-1)) - RealVNC Viewer

Feb 4 14:38

osboxes@osboxes: ~

ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Host List

| IP Address | MAC Address | Description |
|------------------------|--------------------------|-------------|
| 192.168.122.1 | 52:54:00:B7:F1:23 | |
| 192.168.122.170 | 02:42:D4:97:4F:00 | |
| 192.168.122.232 | 02:42:B0:61:9E:00 | |

Delete Host Add to Target 1 Add to Target 2

Starting unified sniffing...

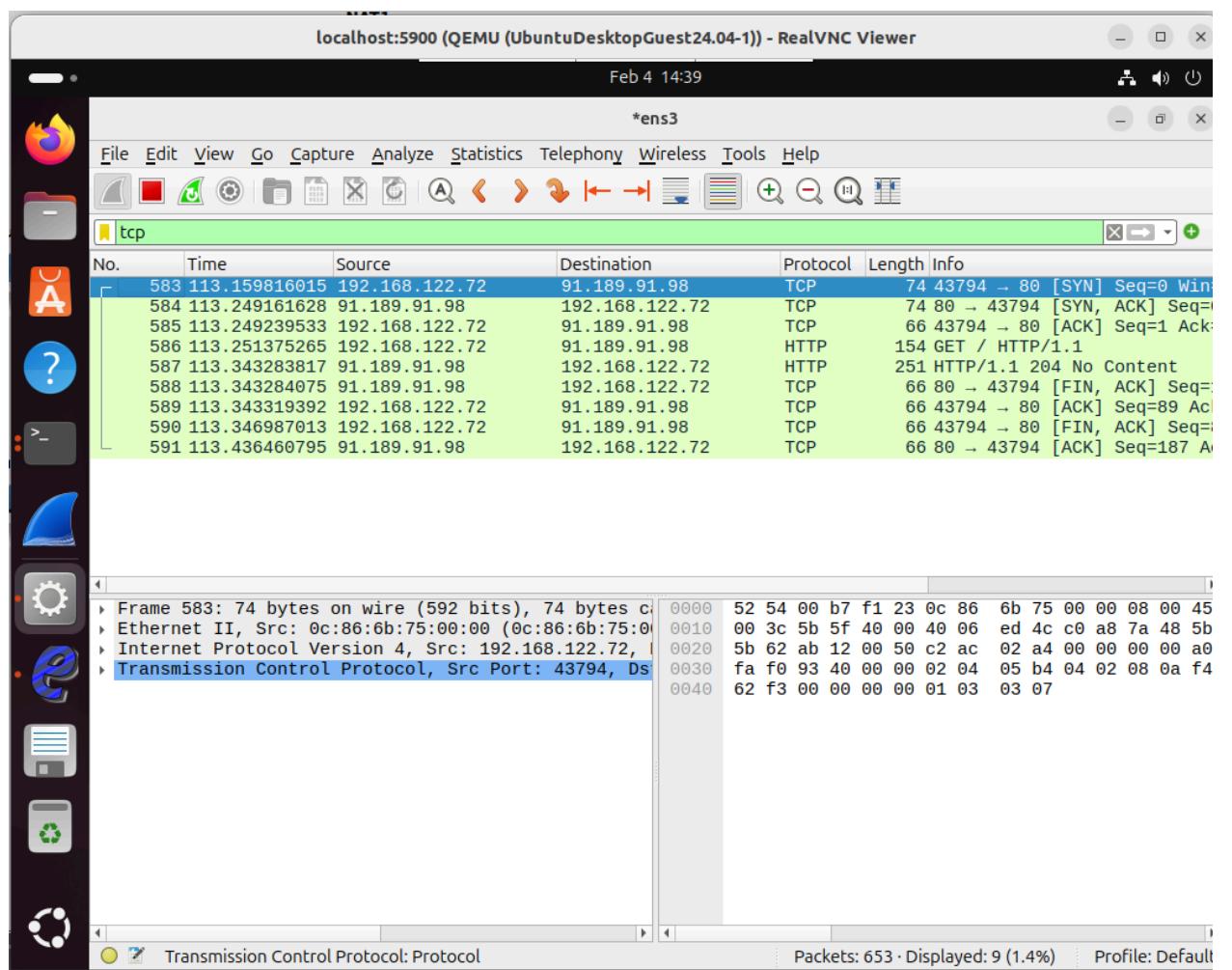
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
3 hosts added to the hosts list...
Host 192.168.122.170 added to TARGET1
Host 192.168.122.232 added to TARGET2

ens3: <live capture in progress>

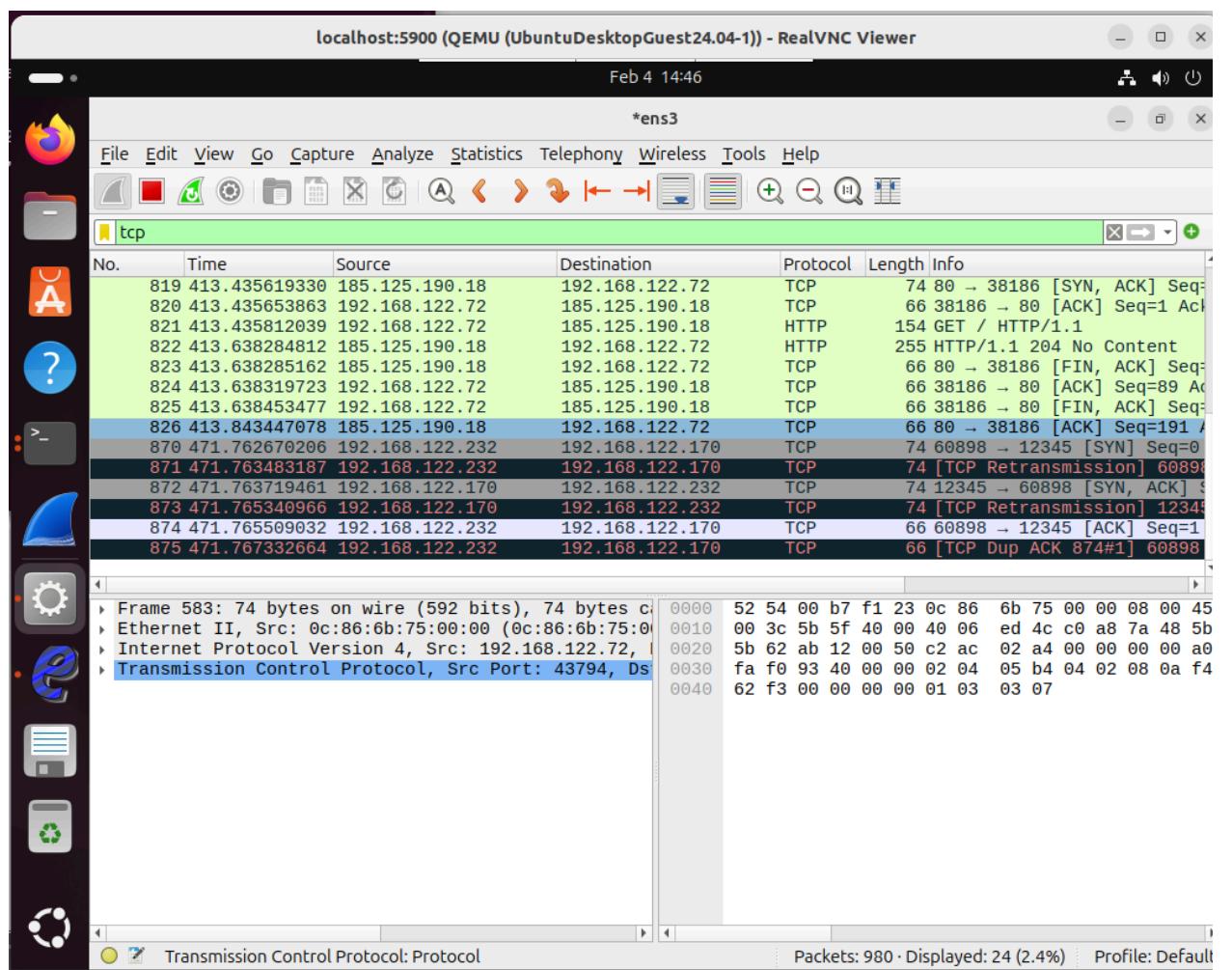
Packets: 597 · Displayed: 597 (100.0%) · Profile: Default

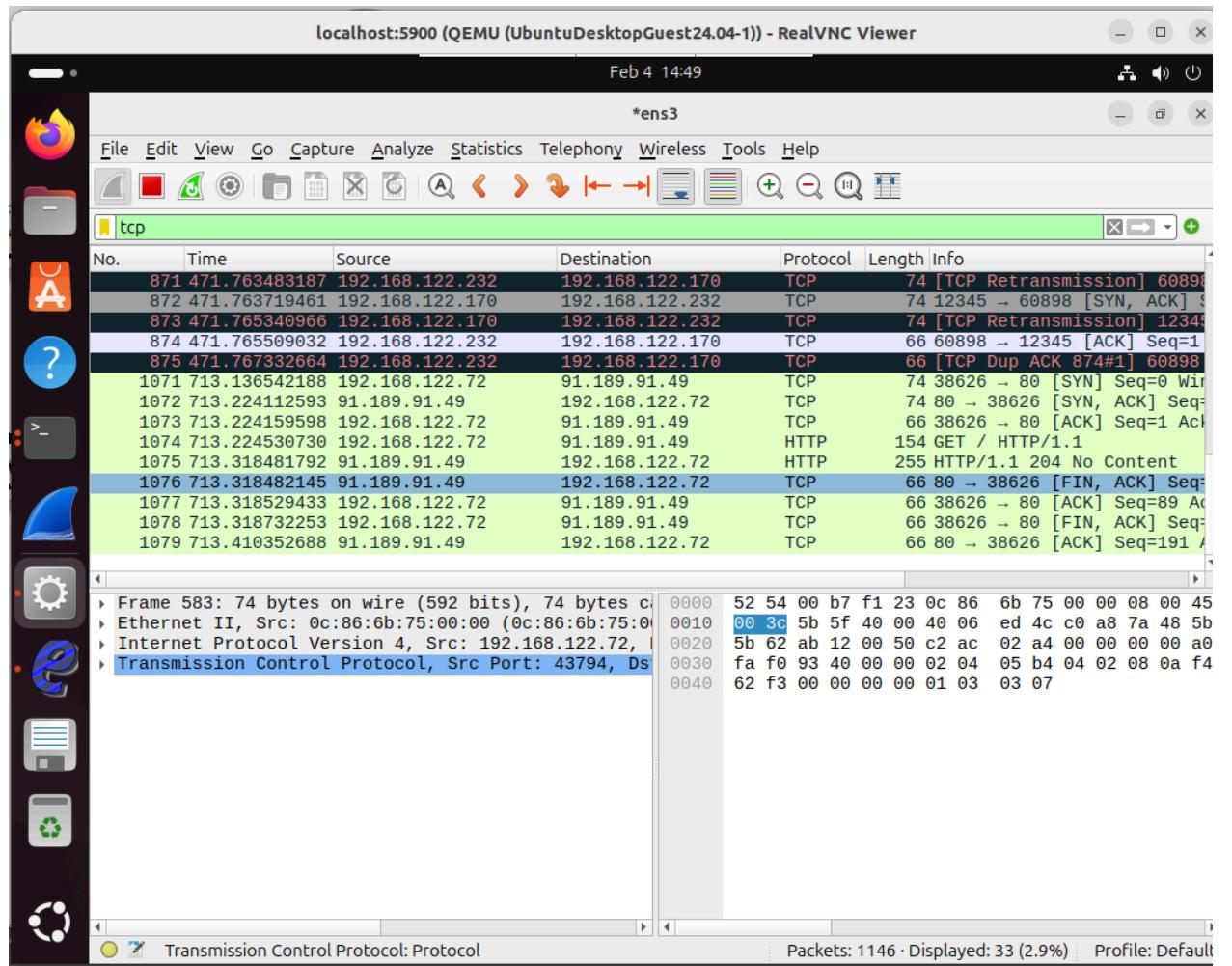
The screenshot shows the Ettercap 0.8.3.1 application running on a Linux desktop. The interface includes a host list table with three entries: 192.168.122.1, 192.168.122.170, and 192.168.122.232. The 192.168.122.232 entry is highlighted with an orange selection bar. Below the host list, status messages indicate the application is performing a scan of the subnet (Randomizing 255 hosts for scanning...) and adding hosts to targets (Scanning the whole netmask for 255 hosts..., 3 hosts added to the hosts list...). The bottom of the window shows network interface information (ens3: <live capture in progress>) and packet statistics (Packets: 597 · Displayed: 597 (100.0%) · Profile: Default).

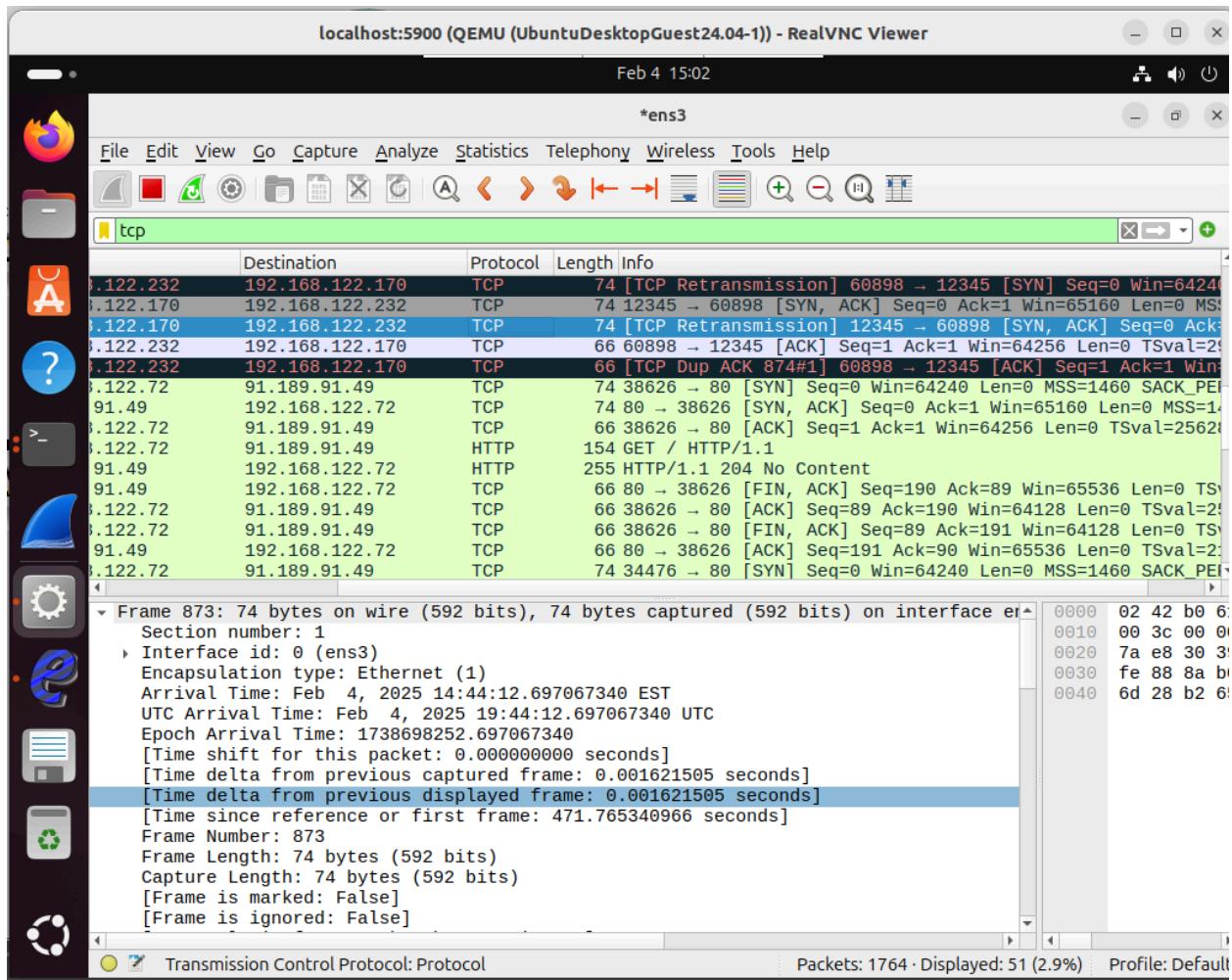
4. Tcp filter

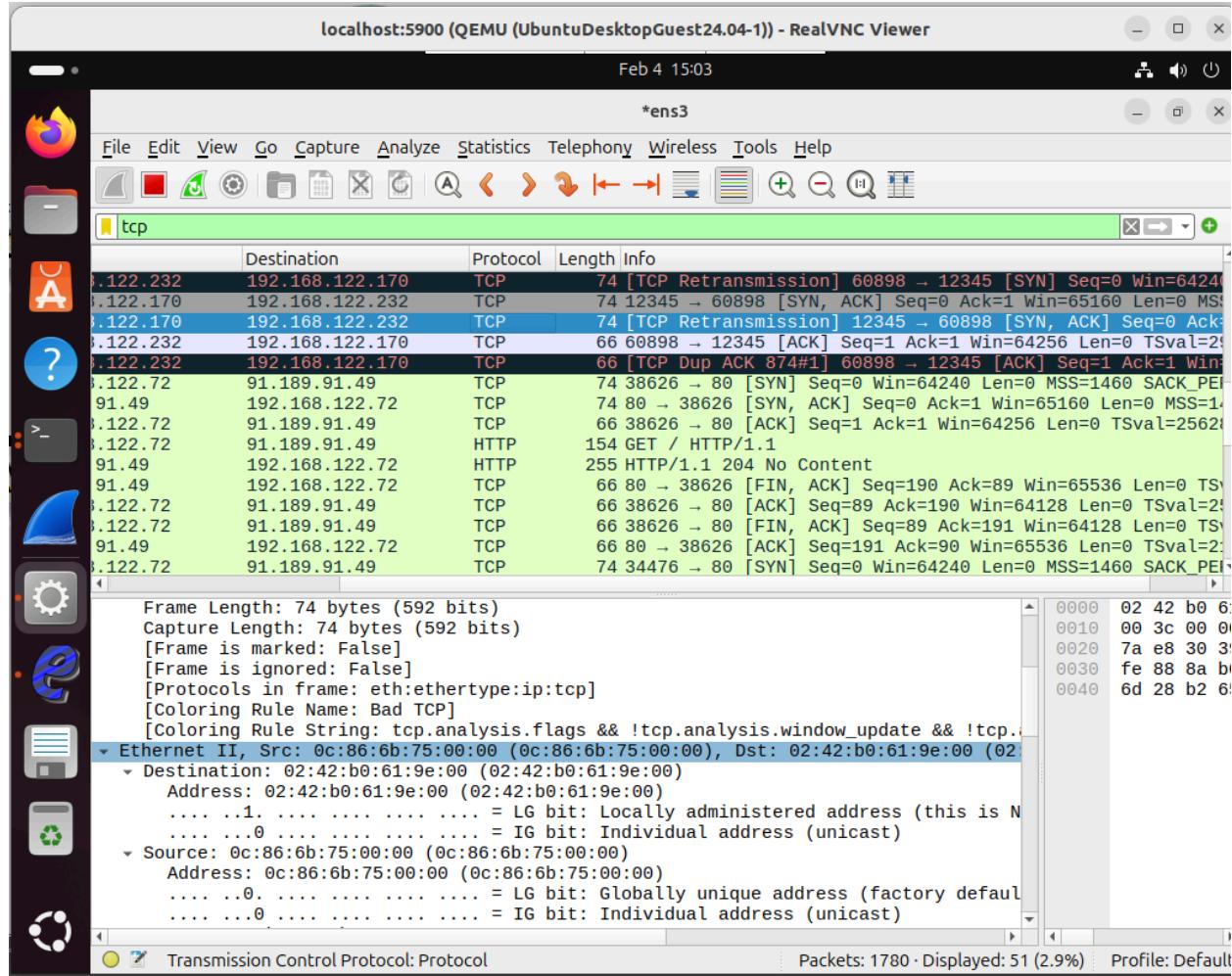


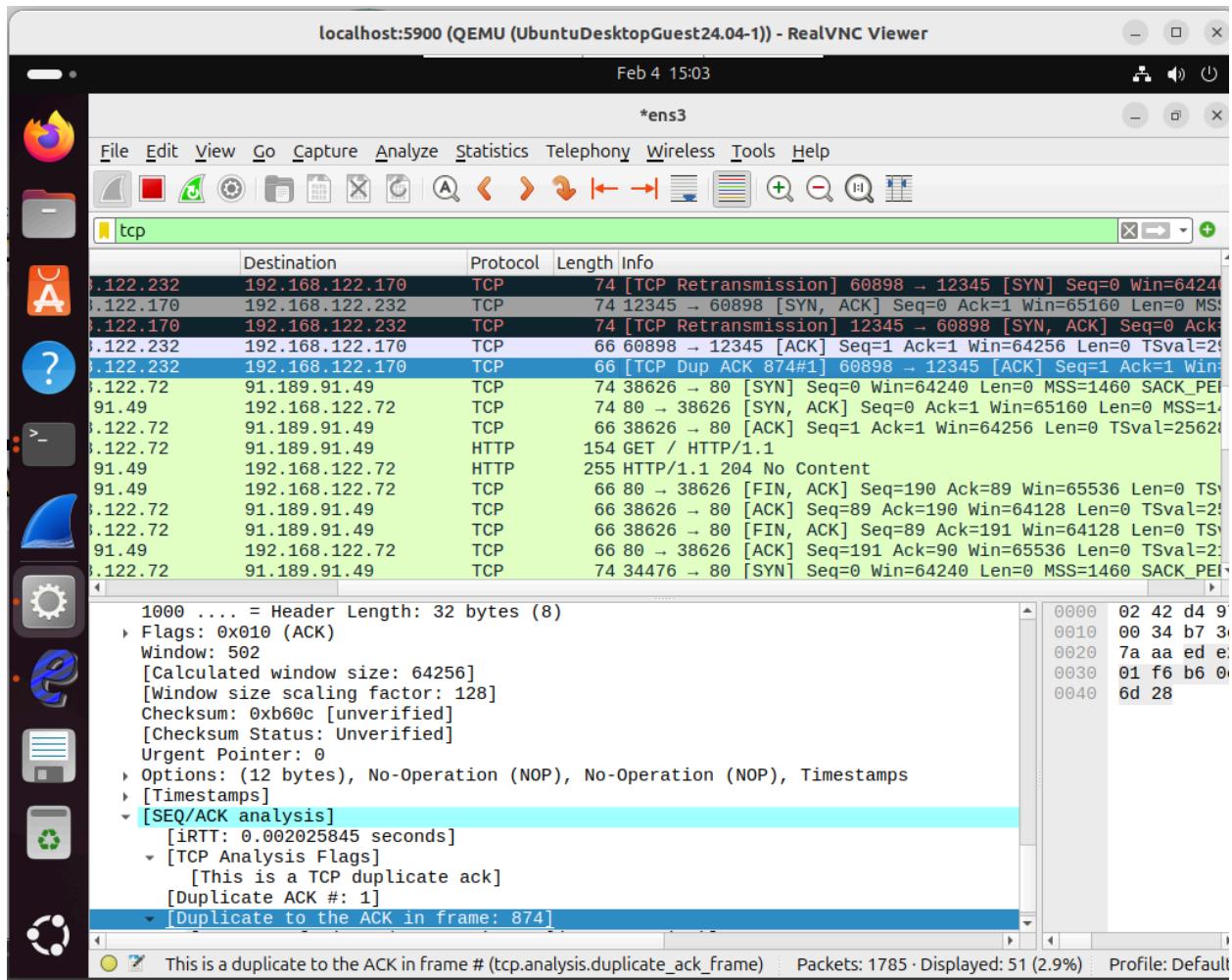
5. Repetition of netcat experiment

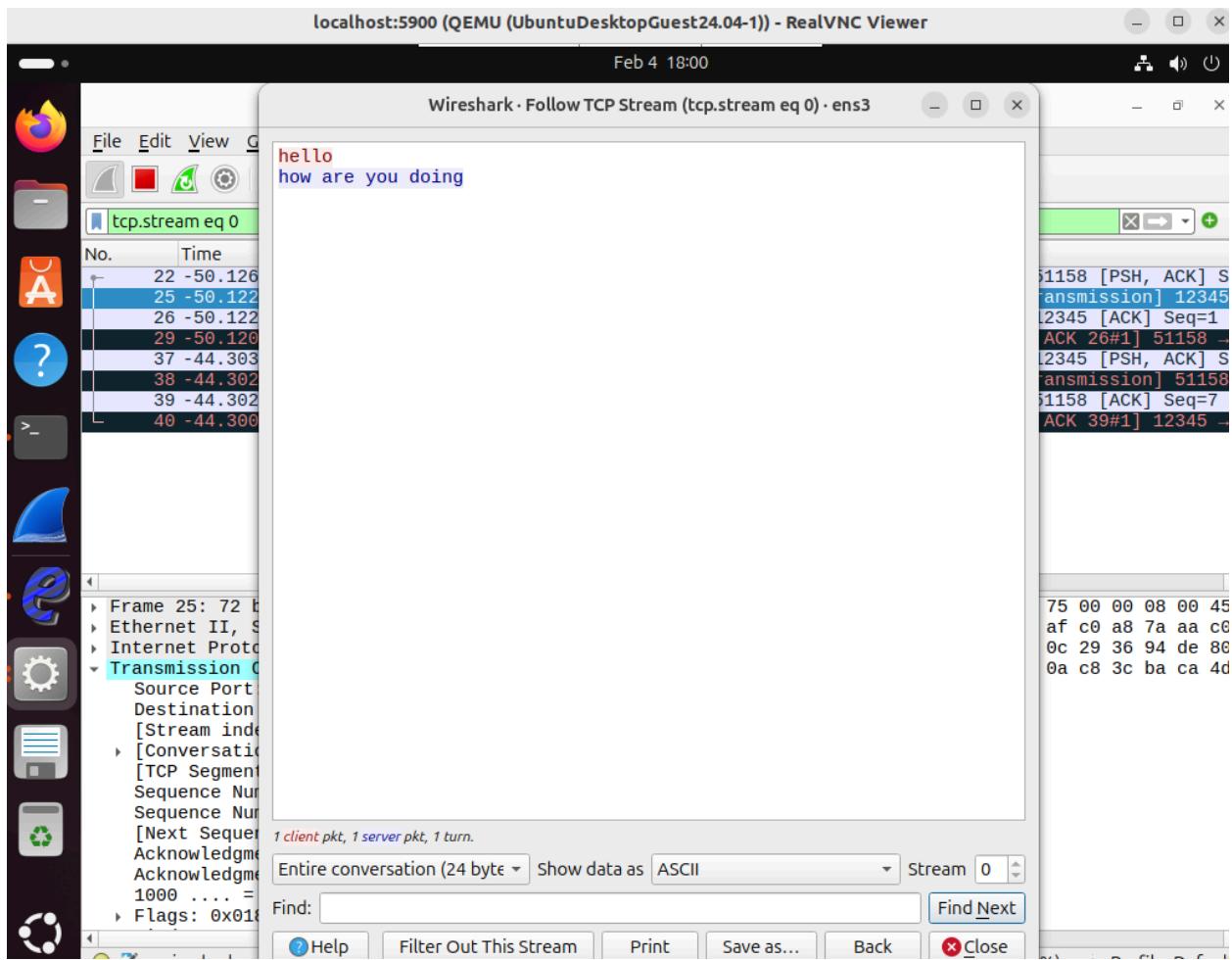


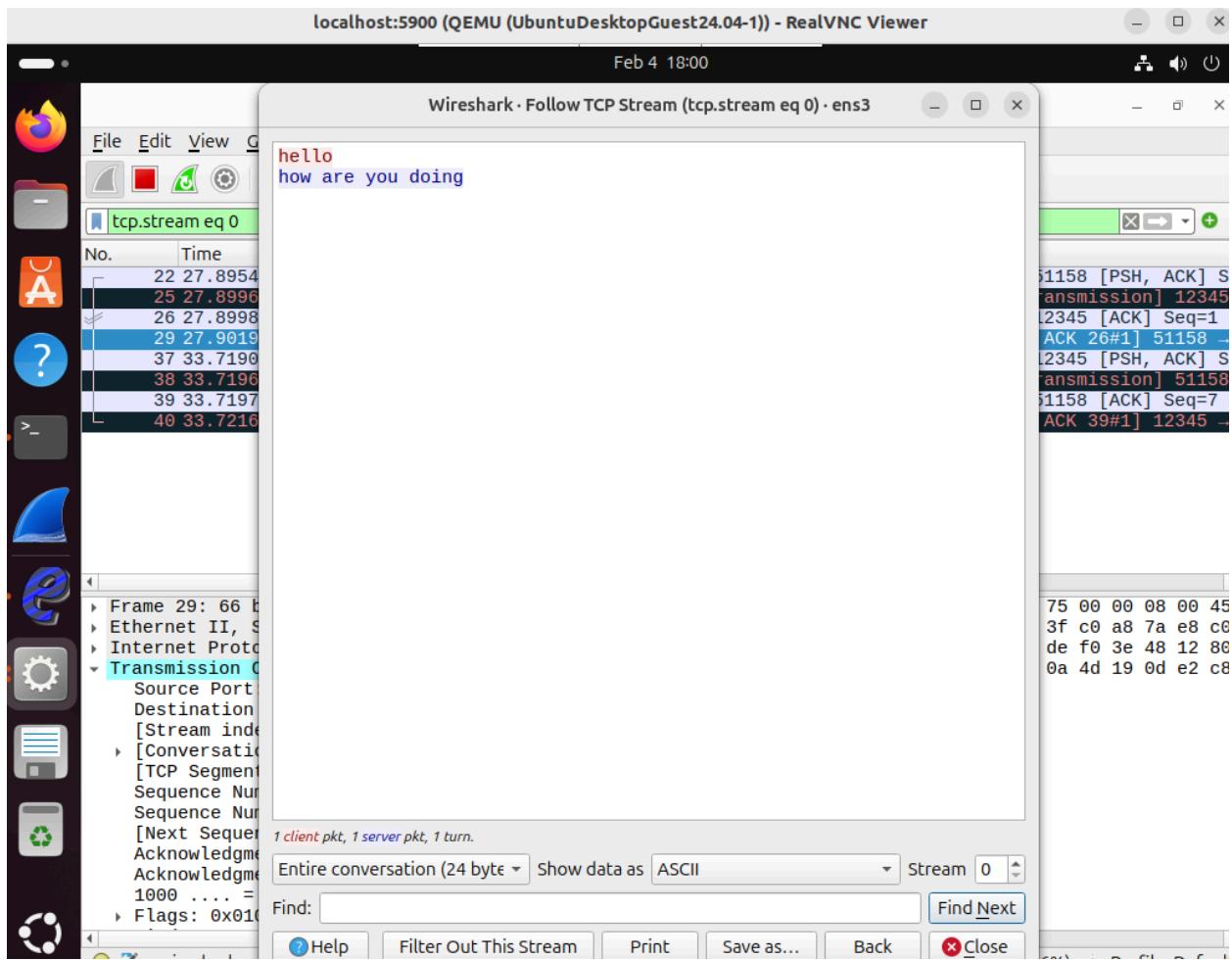


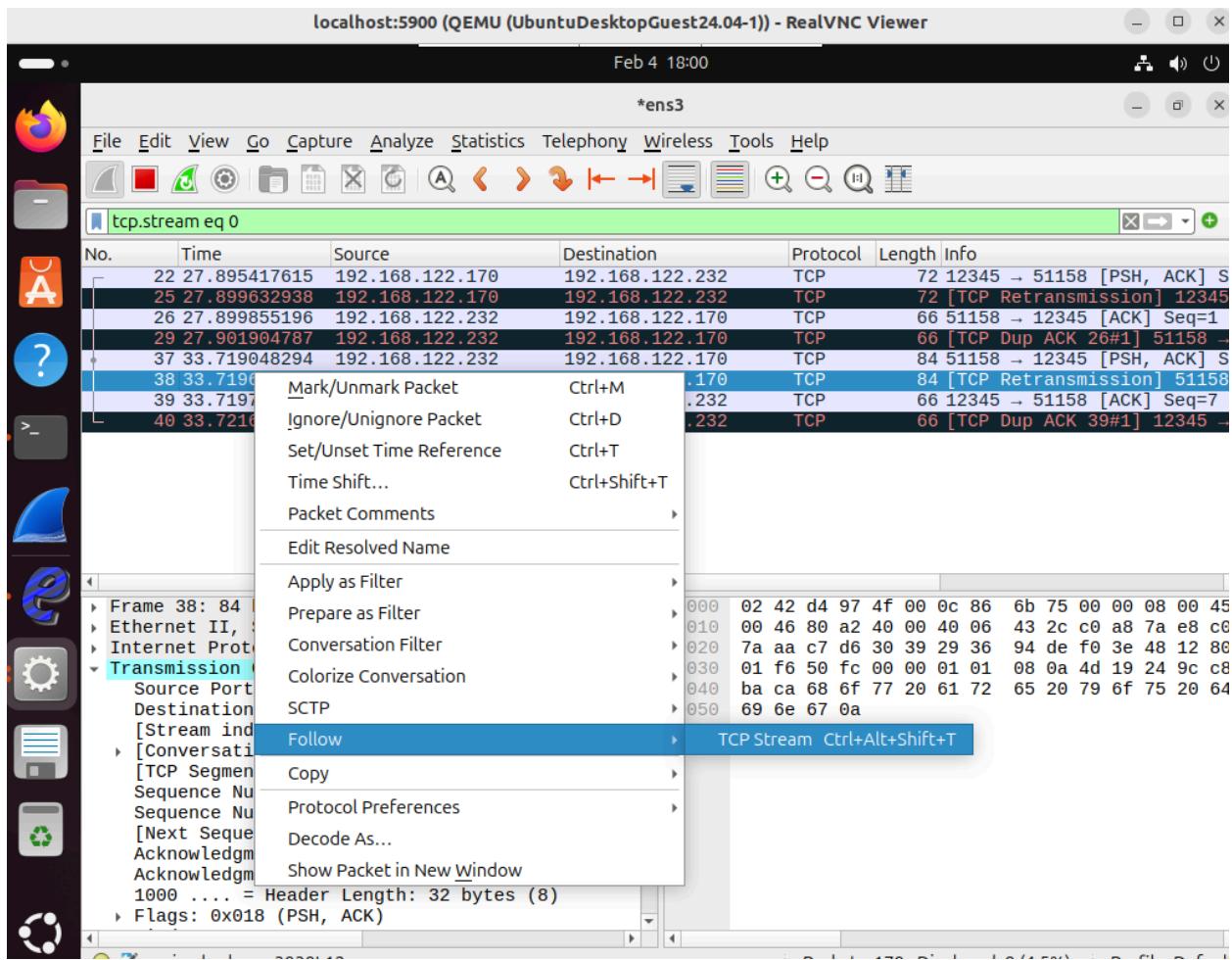


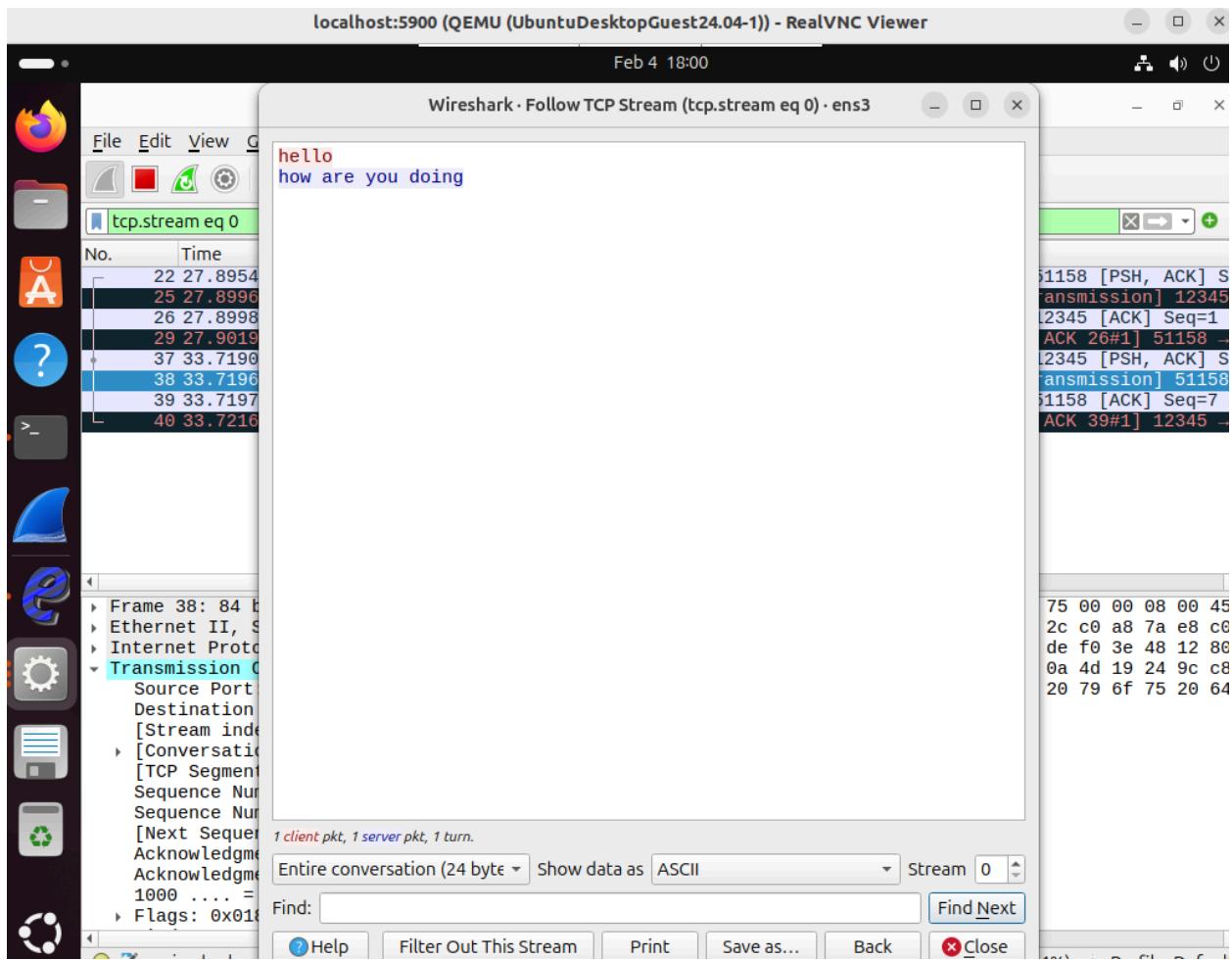


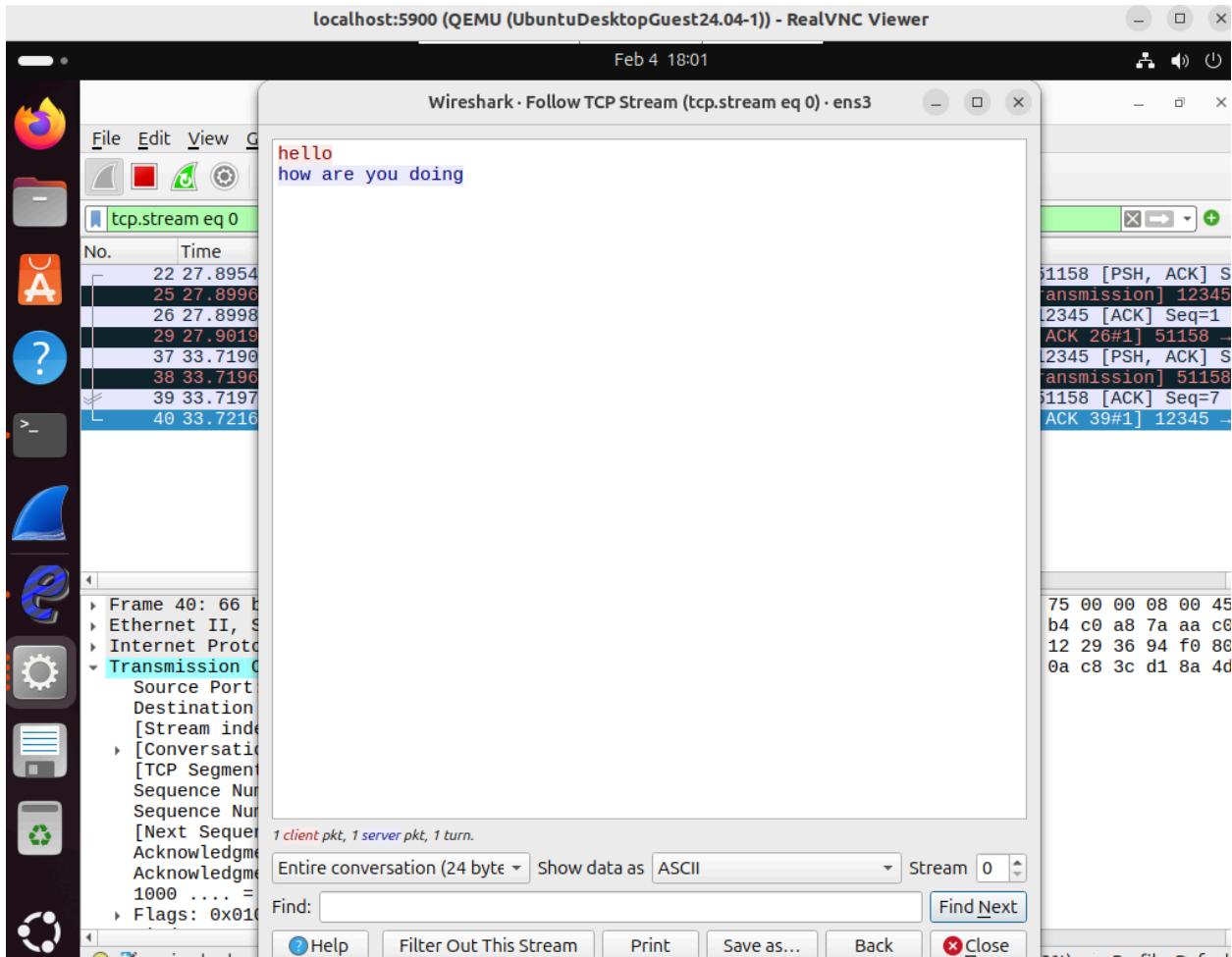






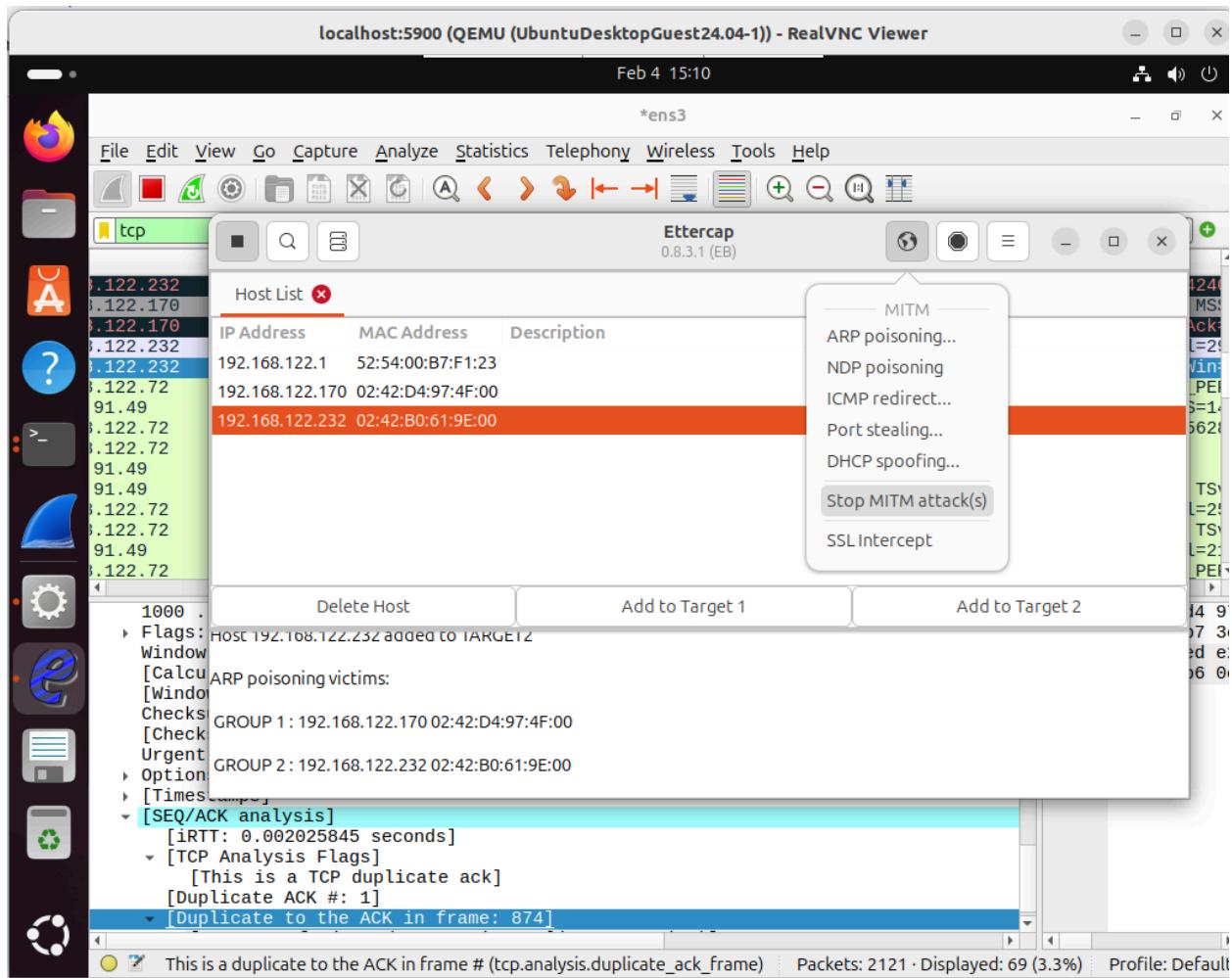


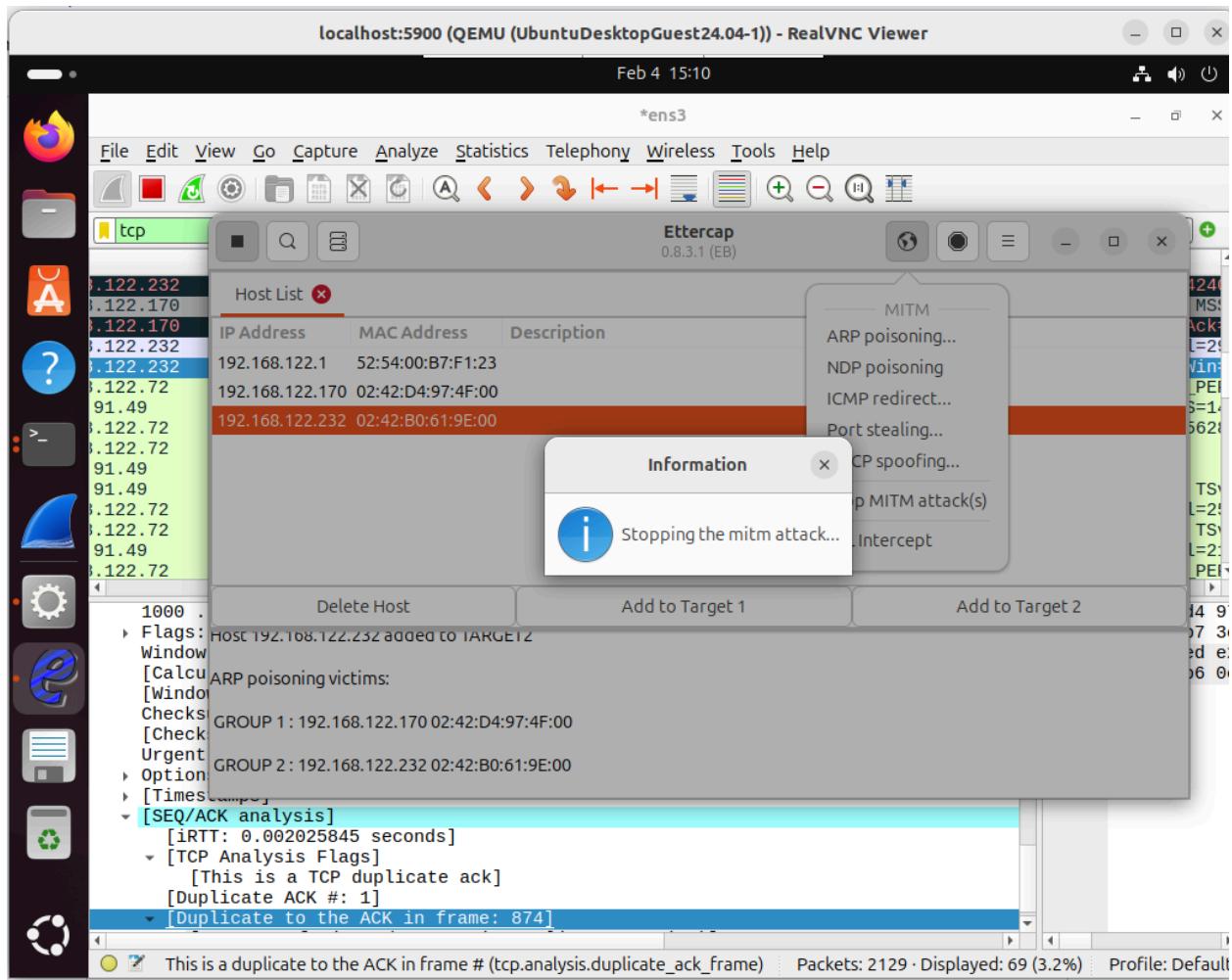


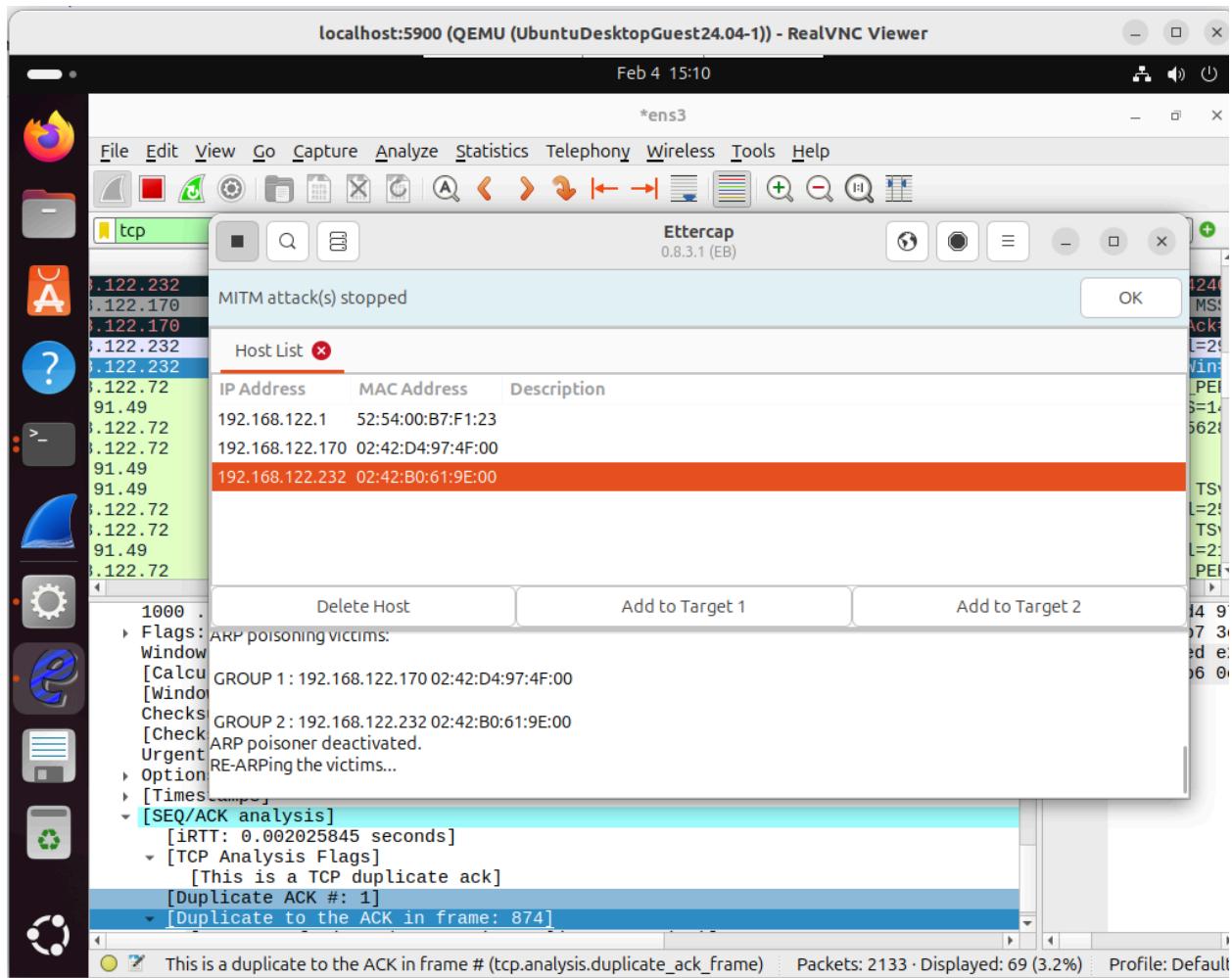


Answer to part 5, based on the screenshots it shows that the ubuntu desktop guest conducts a man in the middle attack while the two linux host(LinuxHost1 and LinuxHost2) are sending and communicating messages with each other and the ubuntu guest desktop uses ettercat to conduct the man in the middle attack and can modify the messages if they want it to for the man in the middle attack. Or passiviyel intercept the message in the man in the middle attack.

6. Stopping attack







Question 7

Next, add the Firefox VM to the topology in the previous question (connect it to the switch), turn the VM on, and open its console (it may take a while for the GUI to boot).

Next, in your Ubuntu VM conduct an ARP poisoning attack between the Firefox VM and the internet gateway (that role is being assumed by the NAT node whose IP is always "192.168.122.1". Please make sure the Wireshark is running, but can change your packet filter to "http".

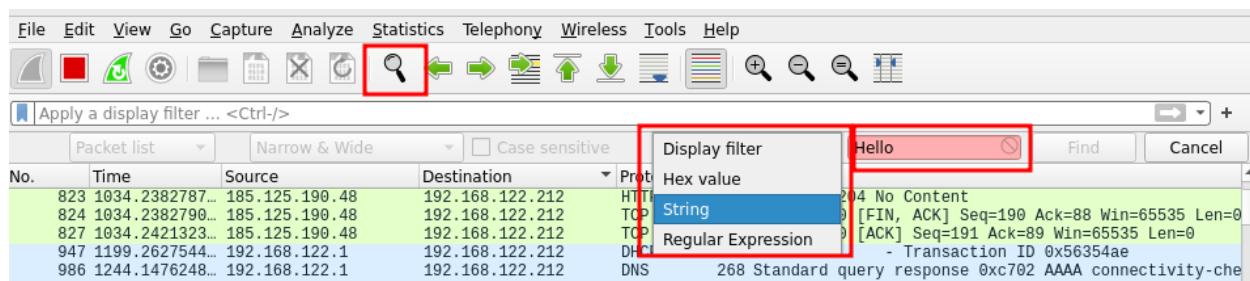
- Next, open the Firefox browser in the Firefox VM and go to:

[<http://testphp.vulnweb.com/login.php>]

- Fill in any user name and password (please use something unique and random) and click "Login".

- Can you see the contents of what you submitted in the Wireshark packet capture? To find it, you can do the following:

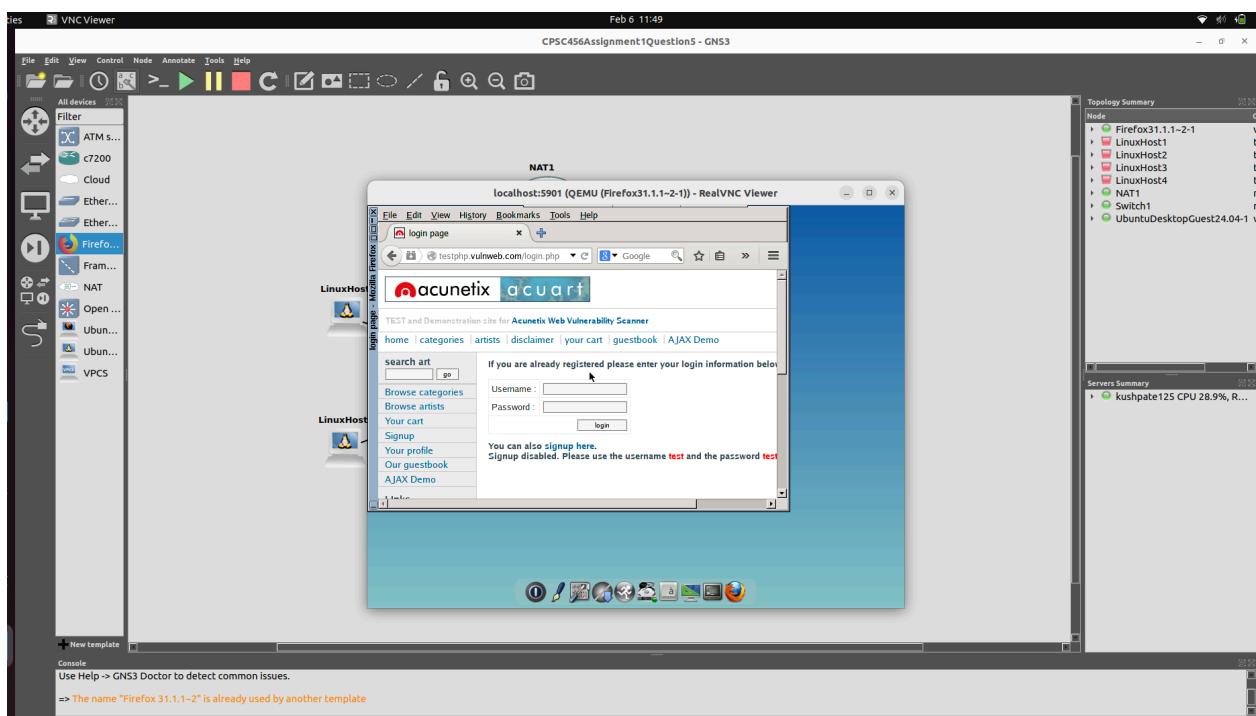
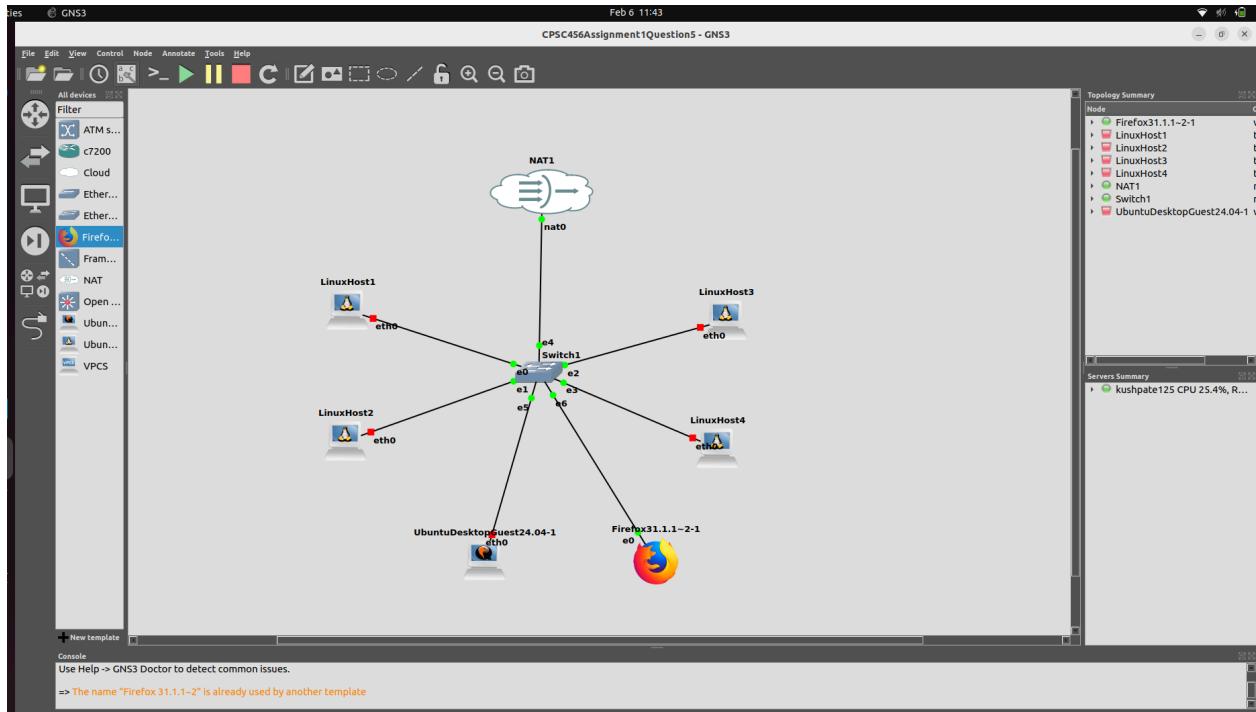
- Remove all filters (and press enter)
- Click on the magnifying glass icon
- Select "String" from the drop-down menu
- Type part of your user name or password in the search box

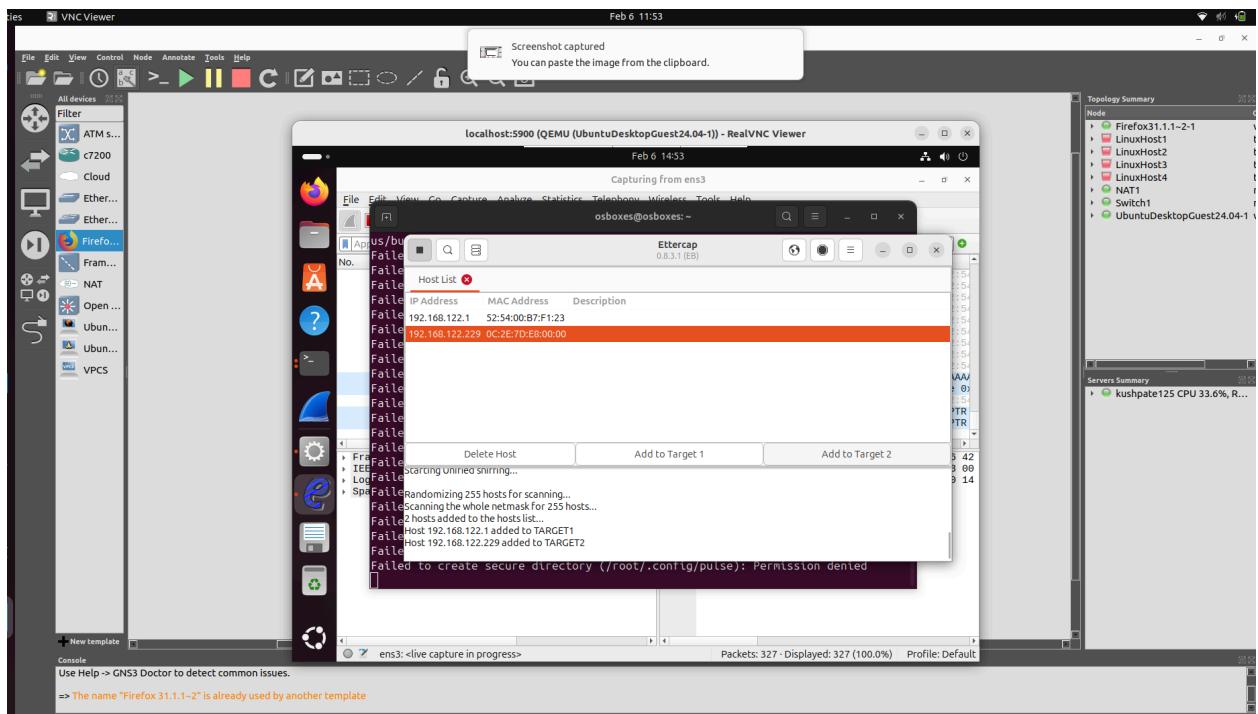
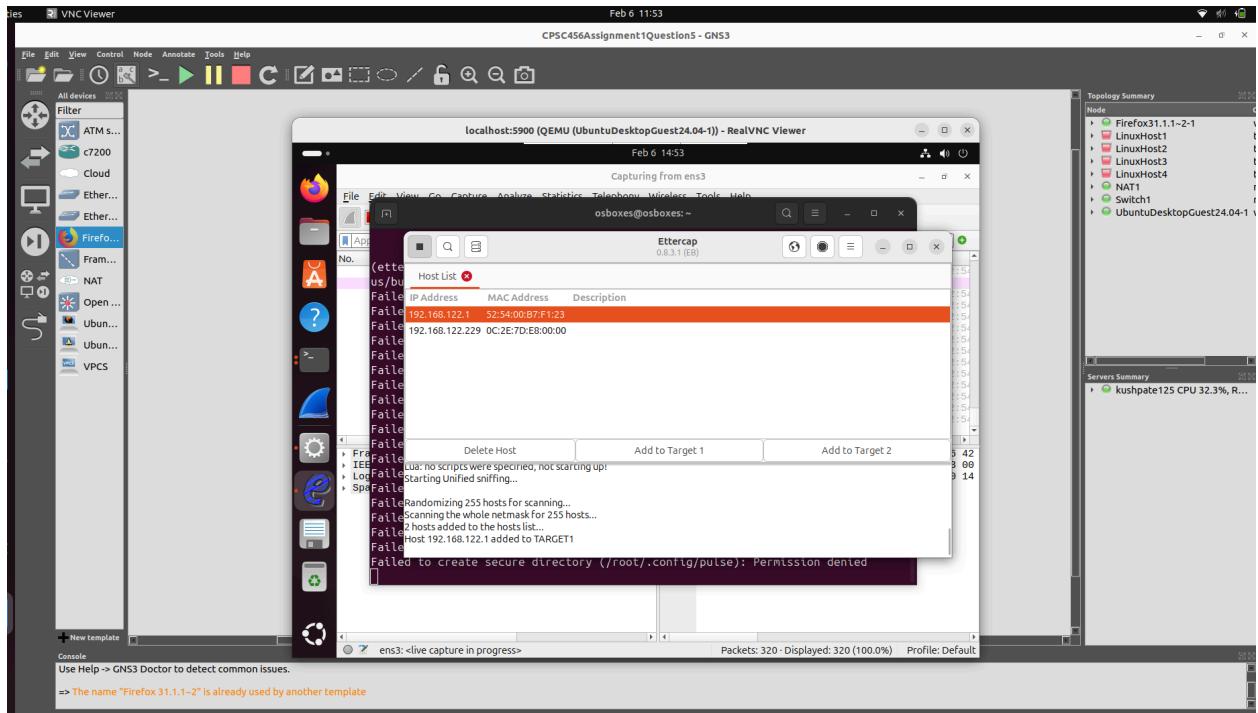


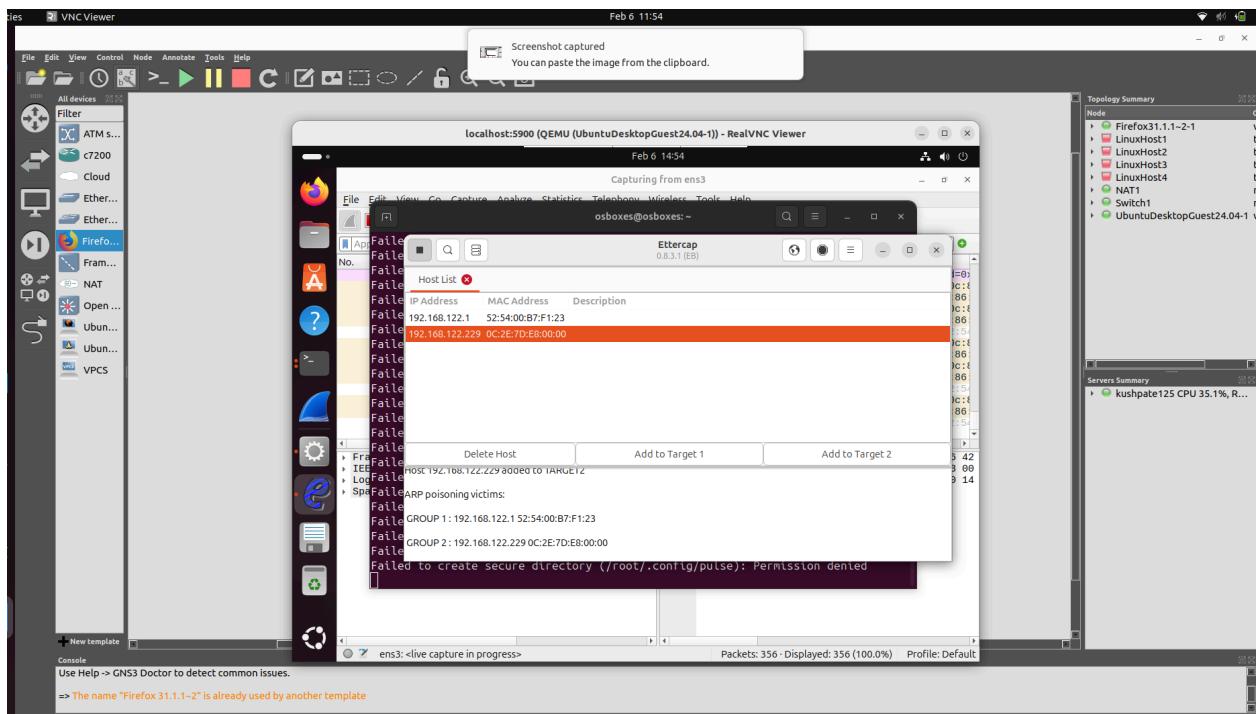
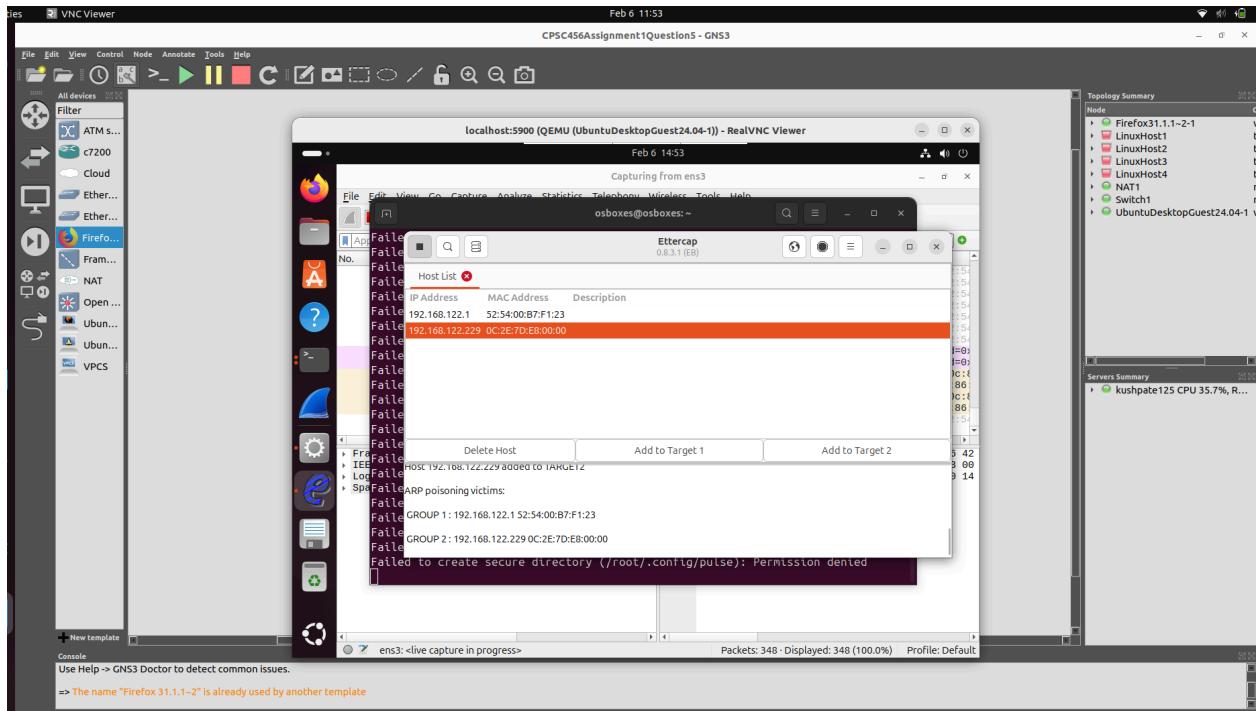
Please describe what is going on. Include screenshots of your observations (especially the packets with the data).

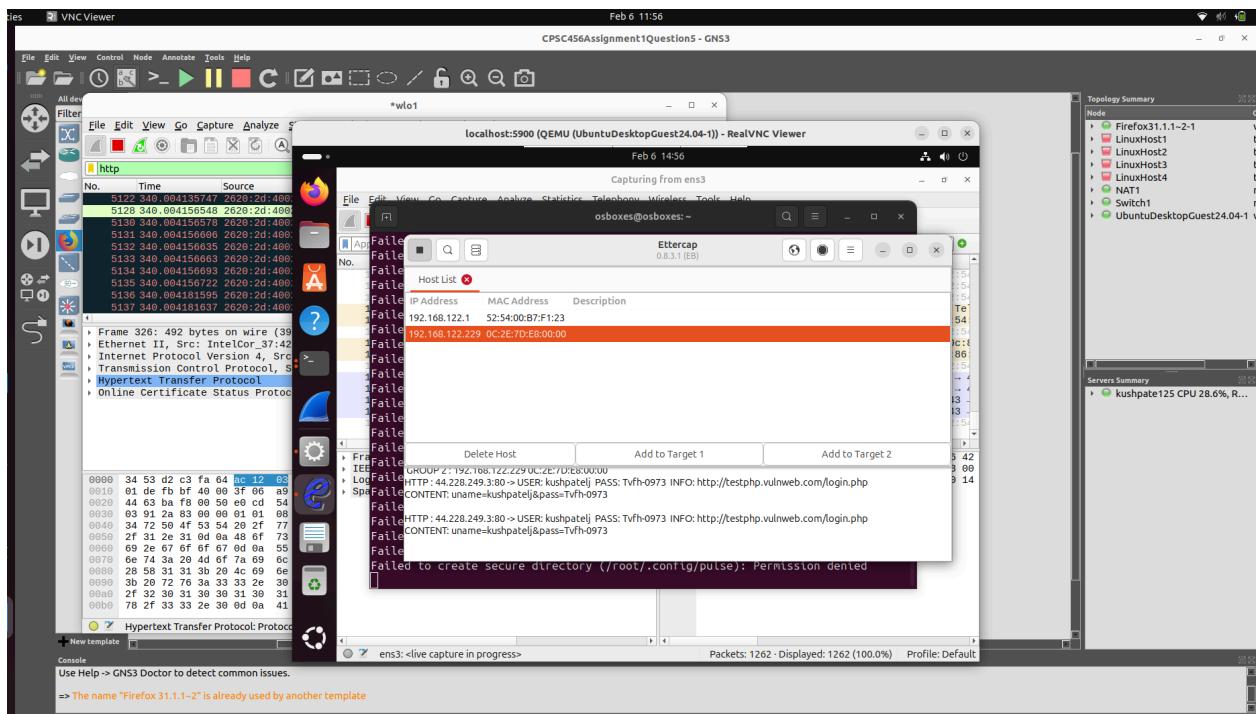
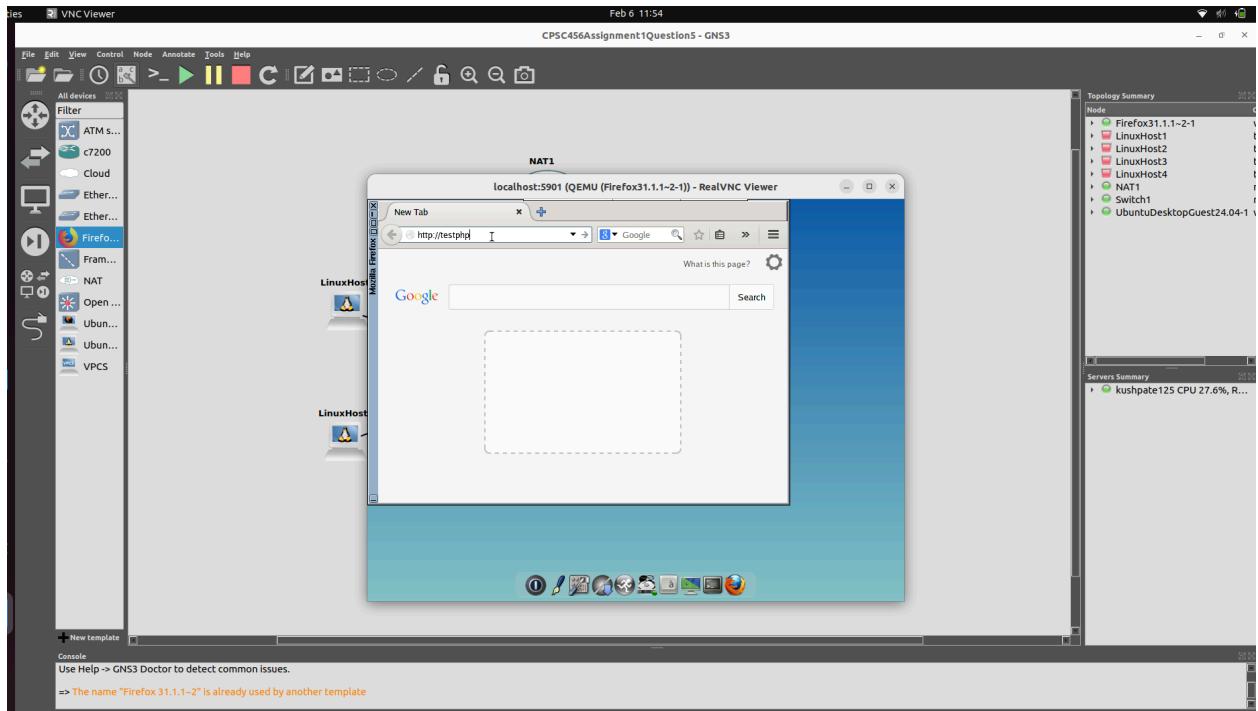
Does this pose risks in the enterprise setting?

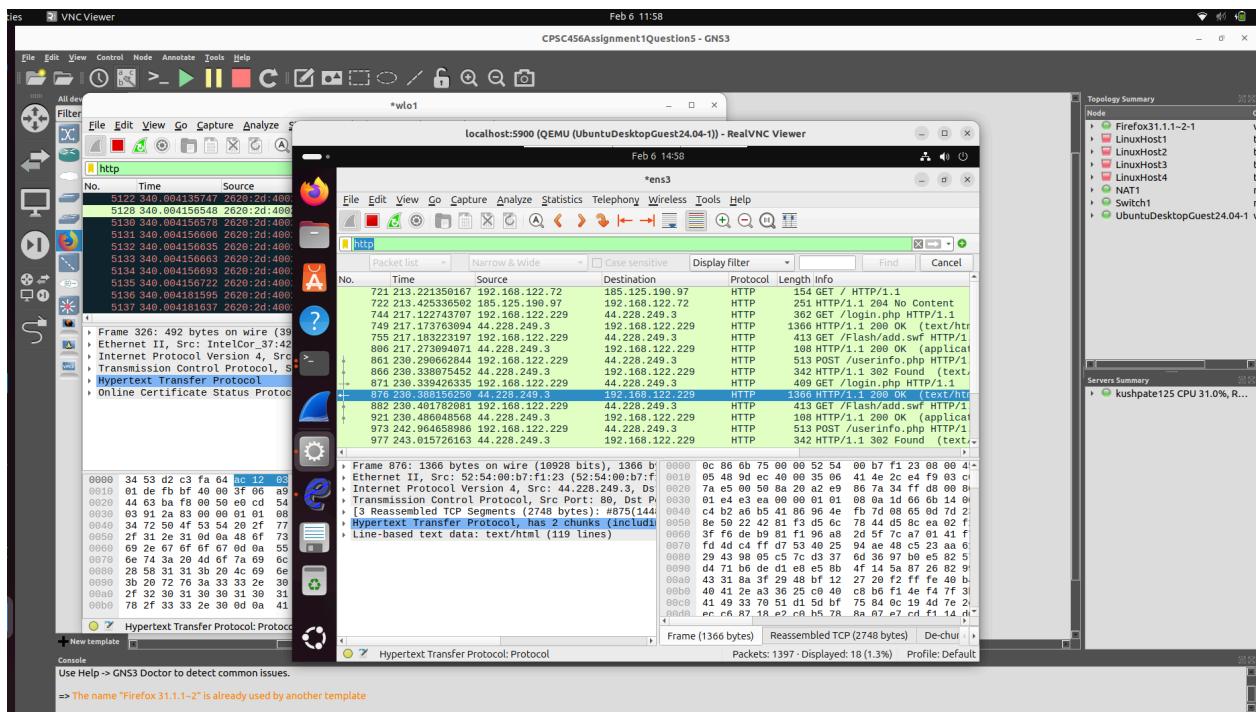
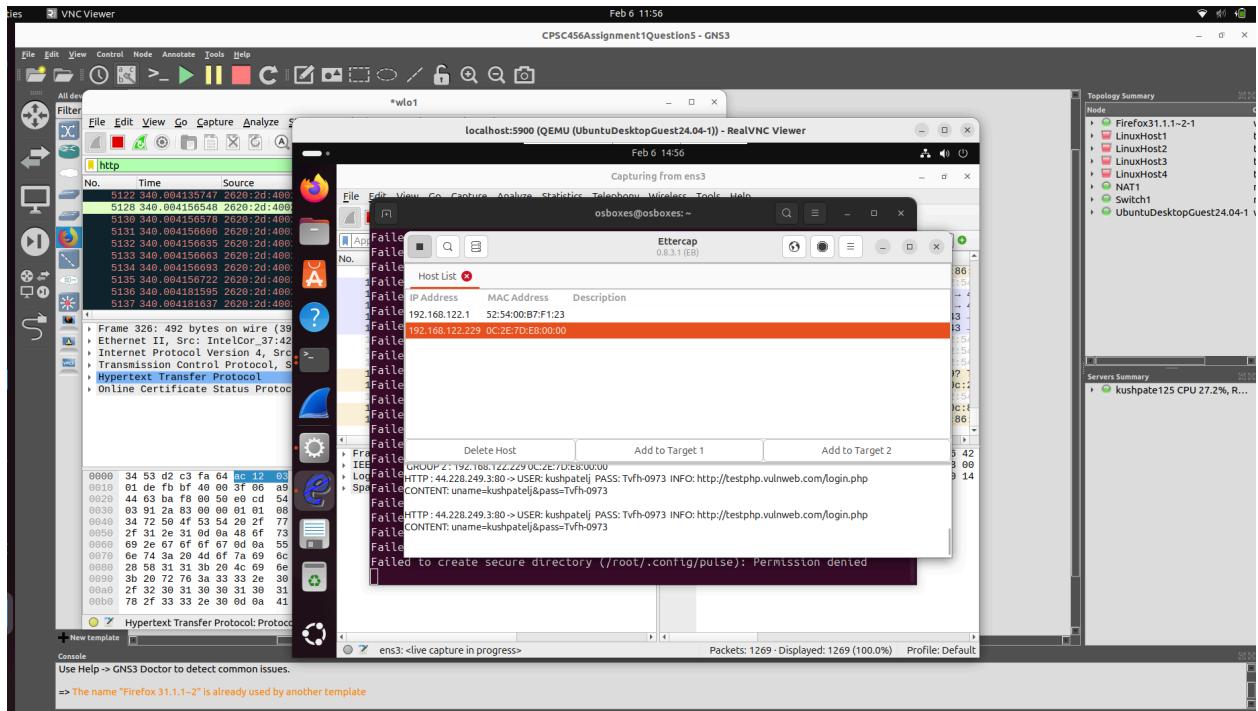
Screenshots to this question

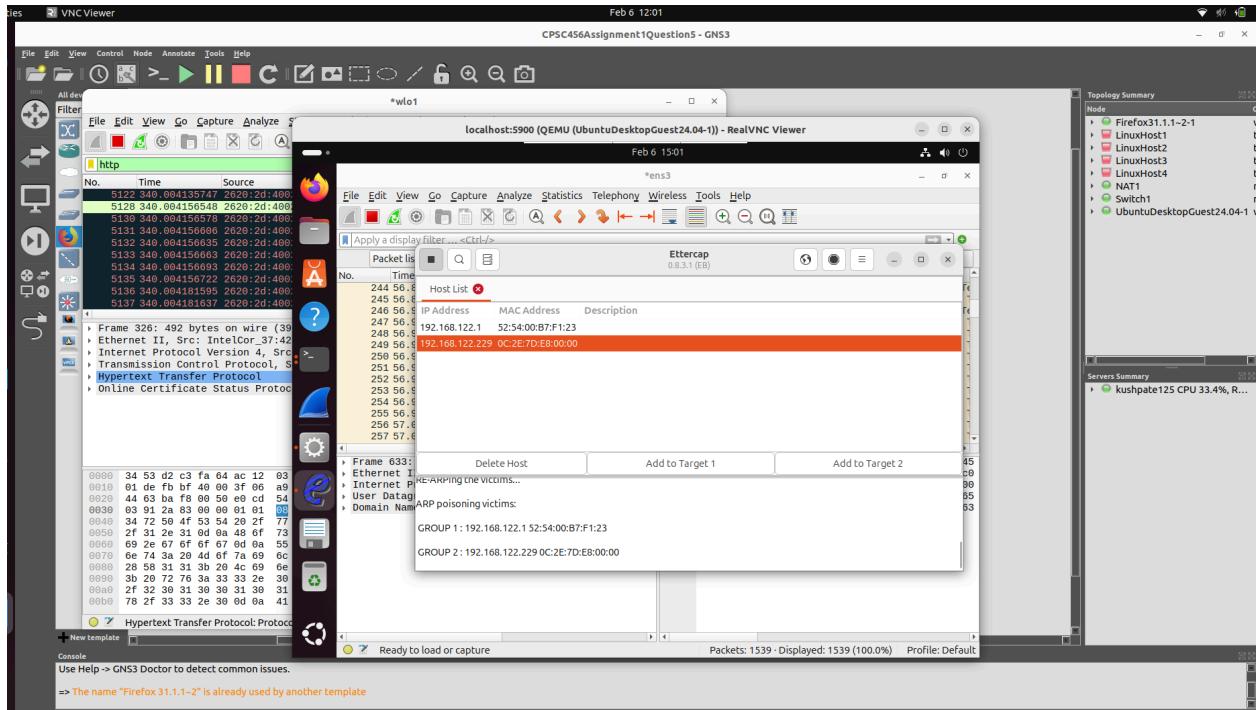
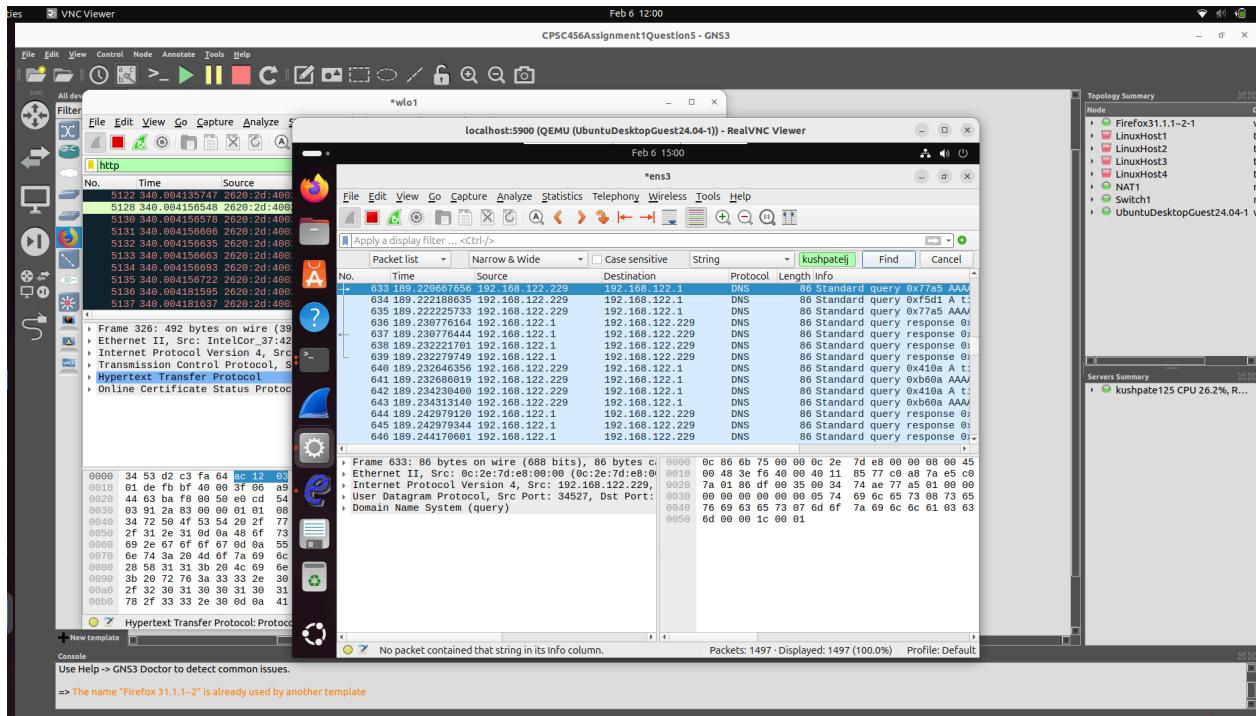


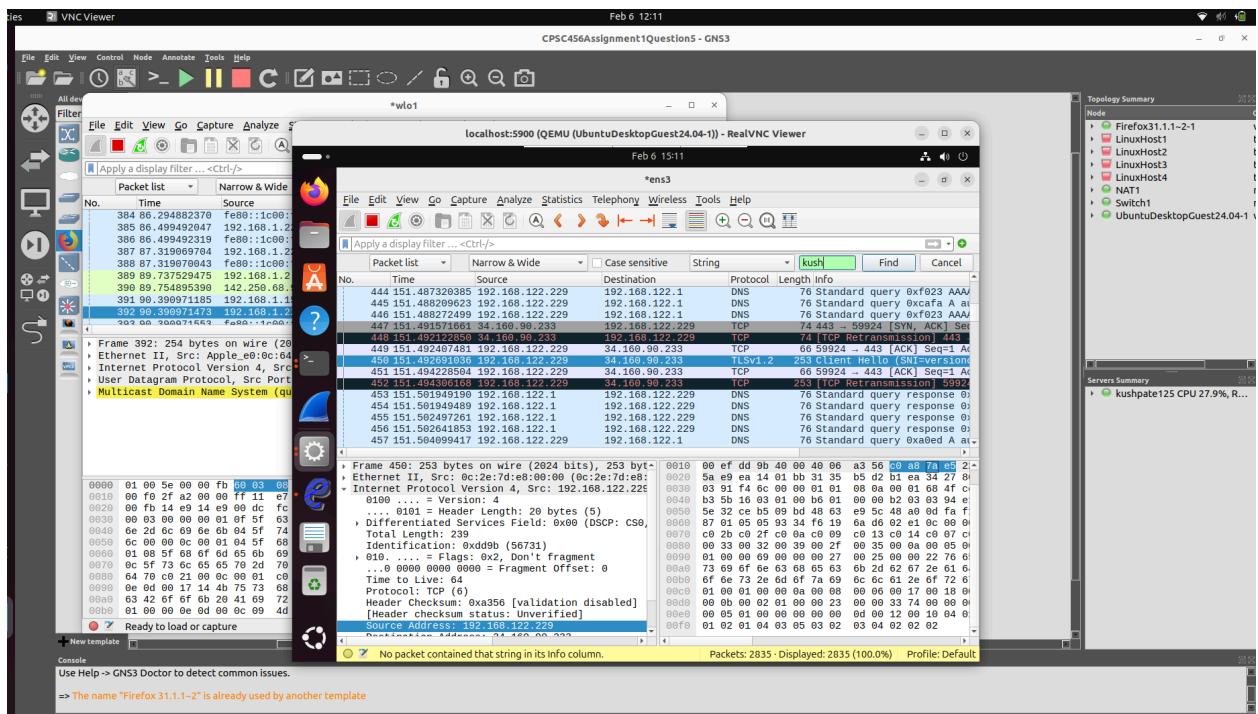
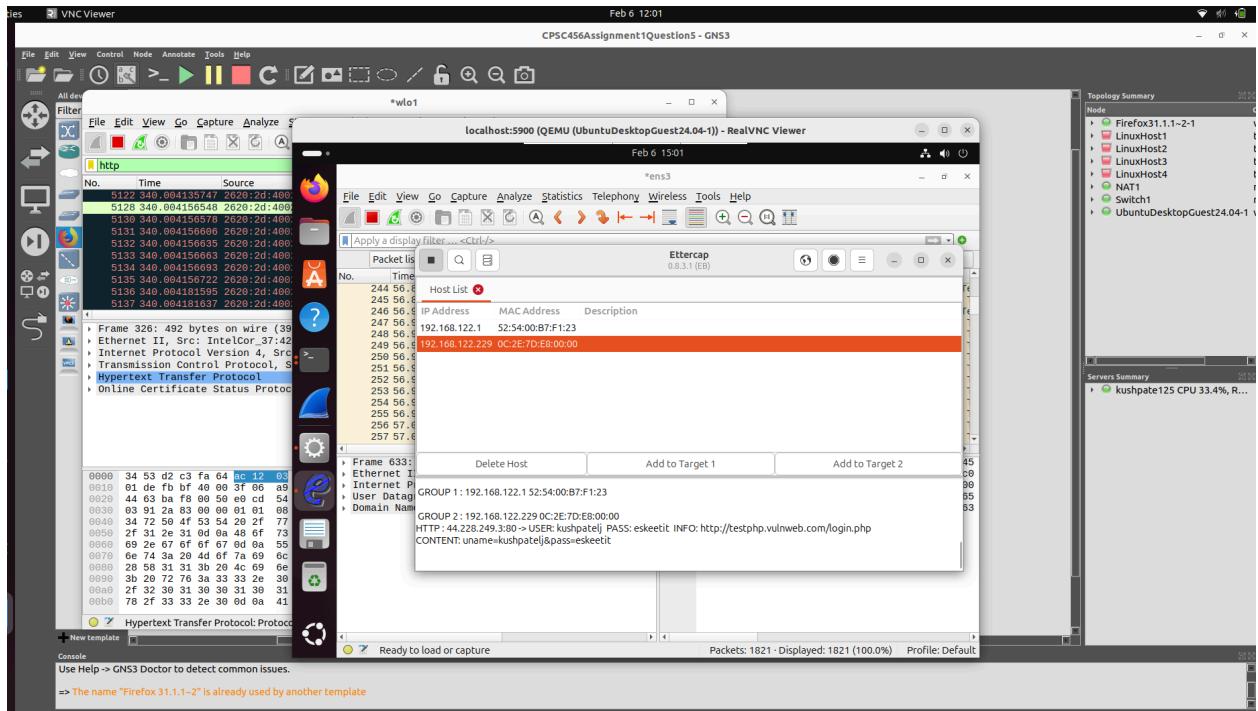


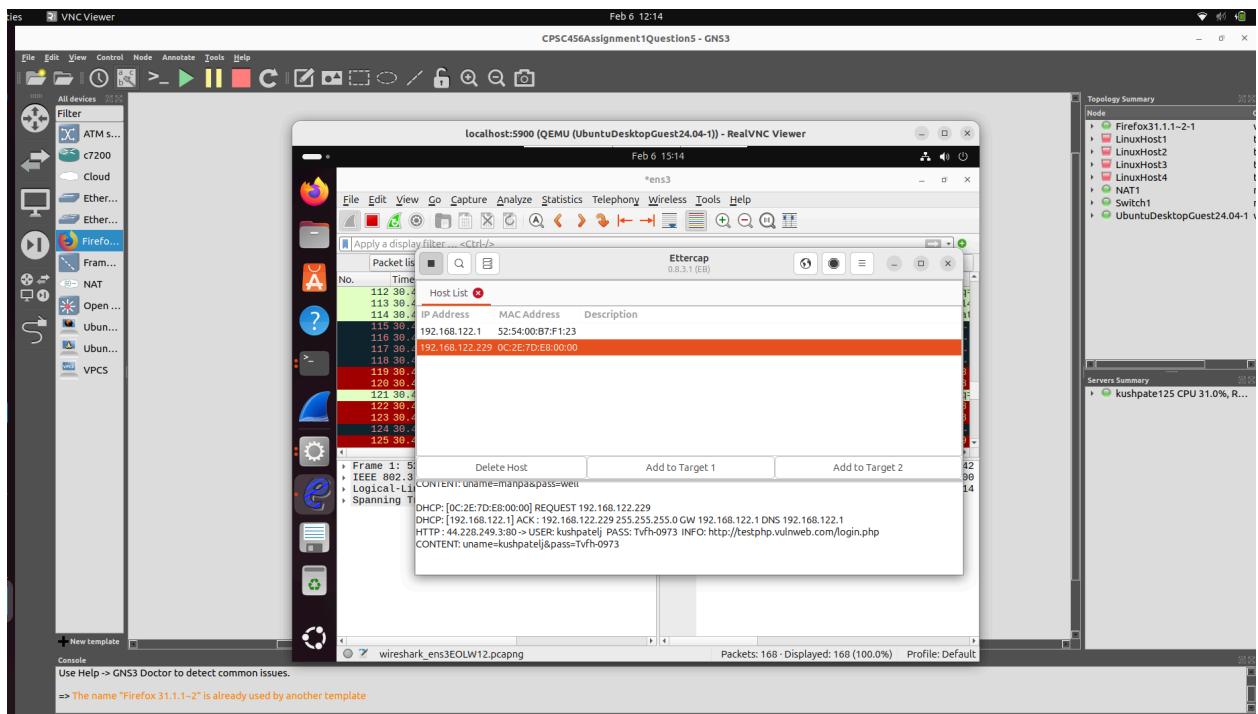
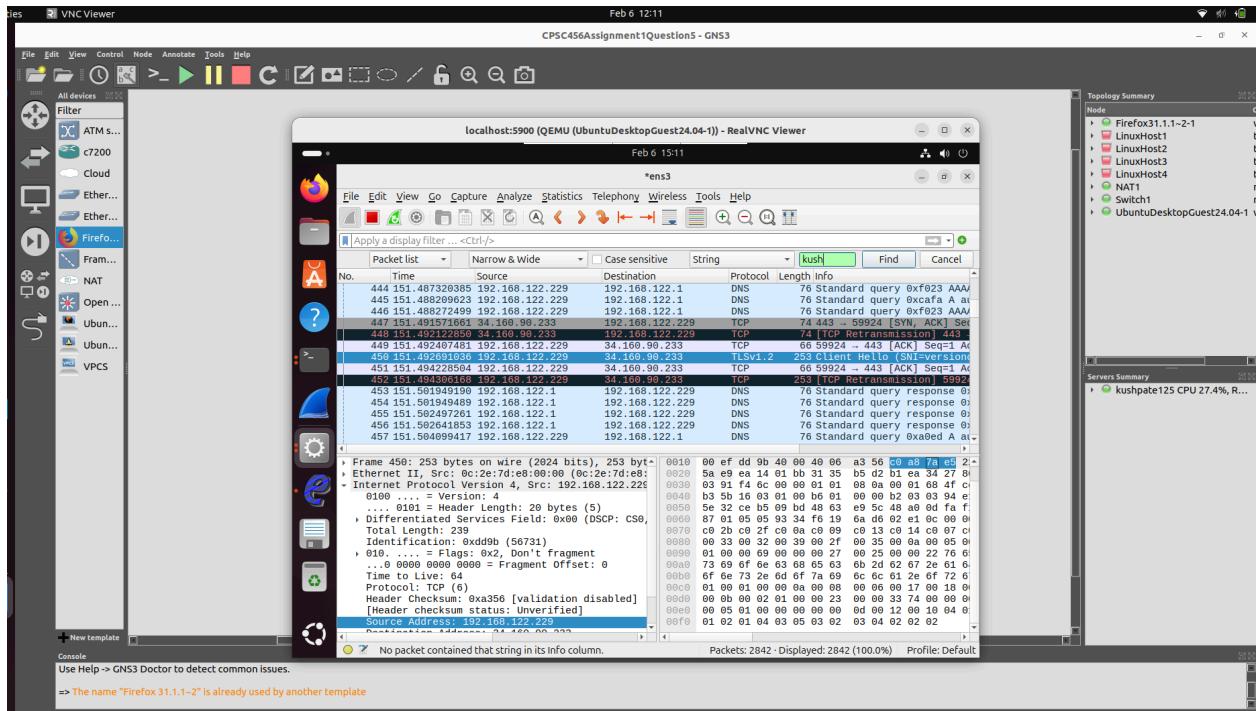


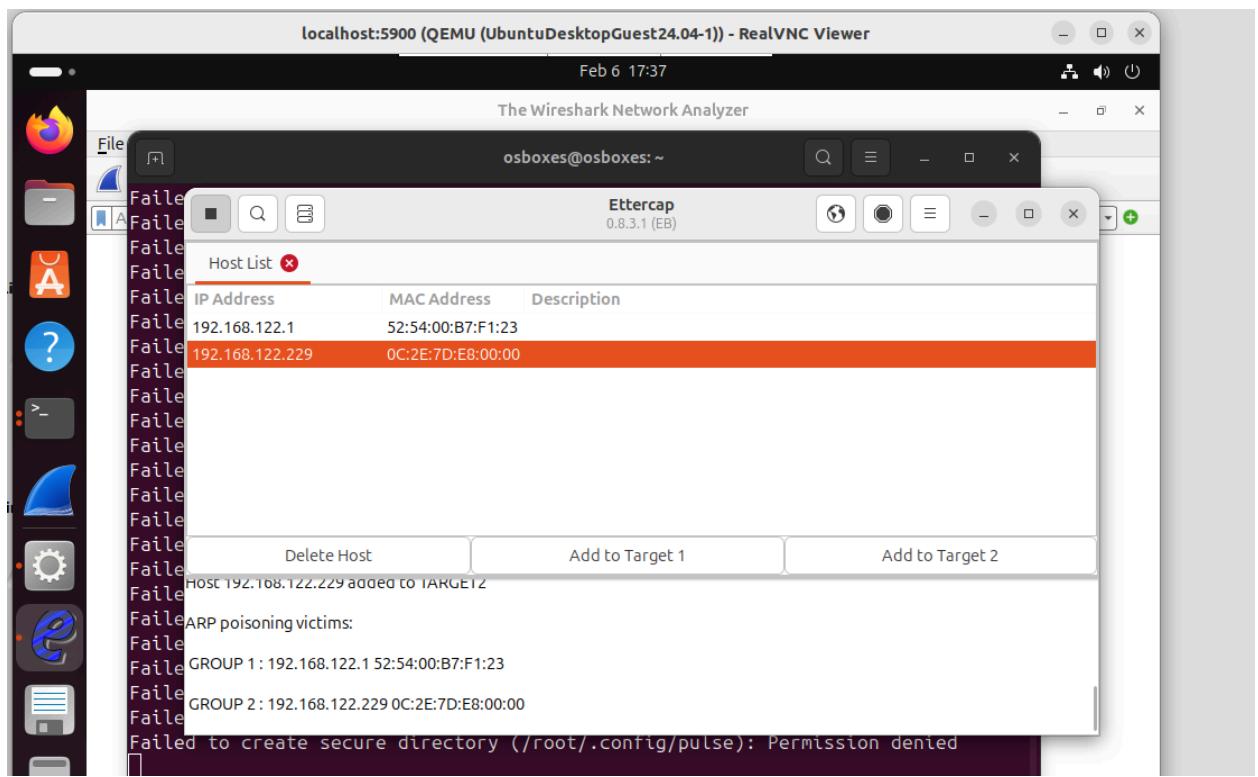
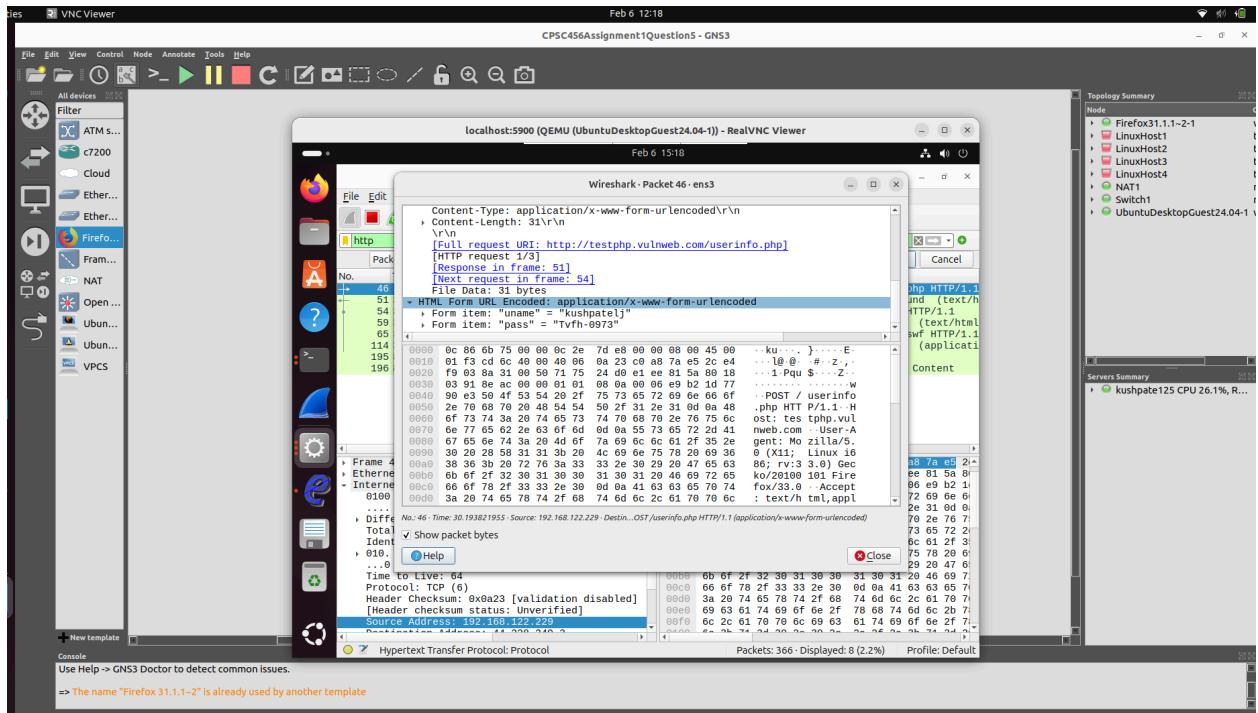


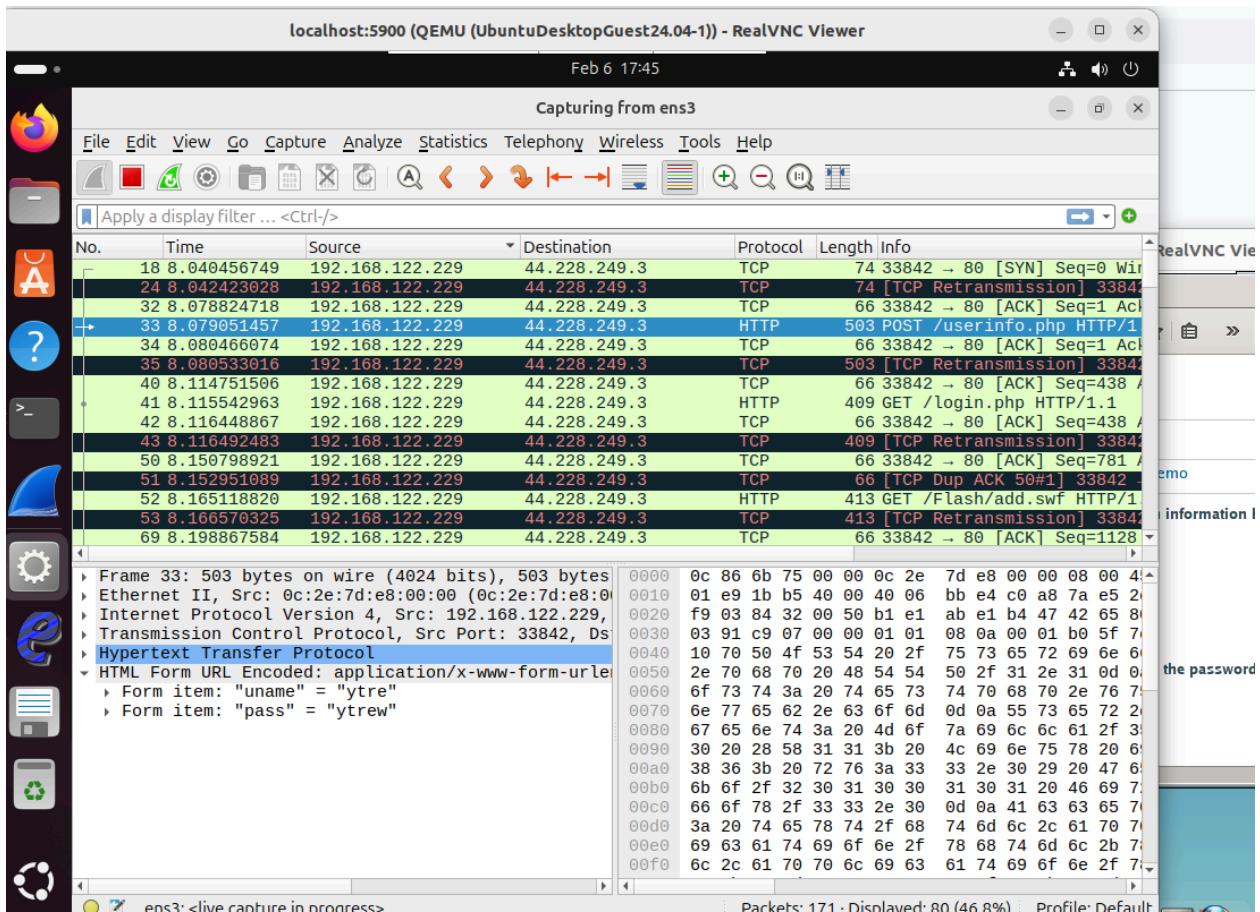


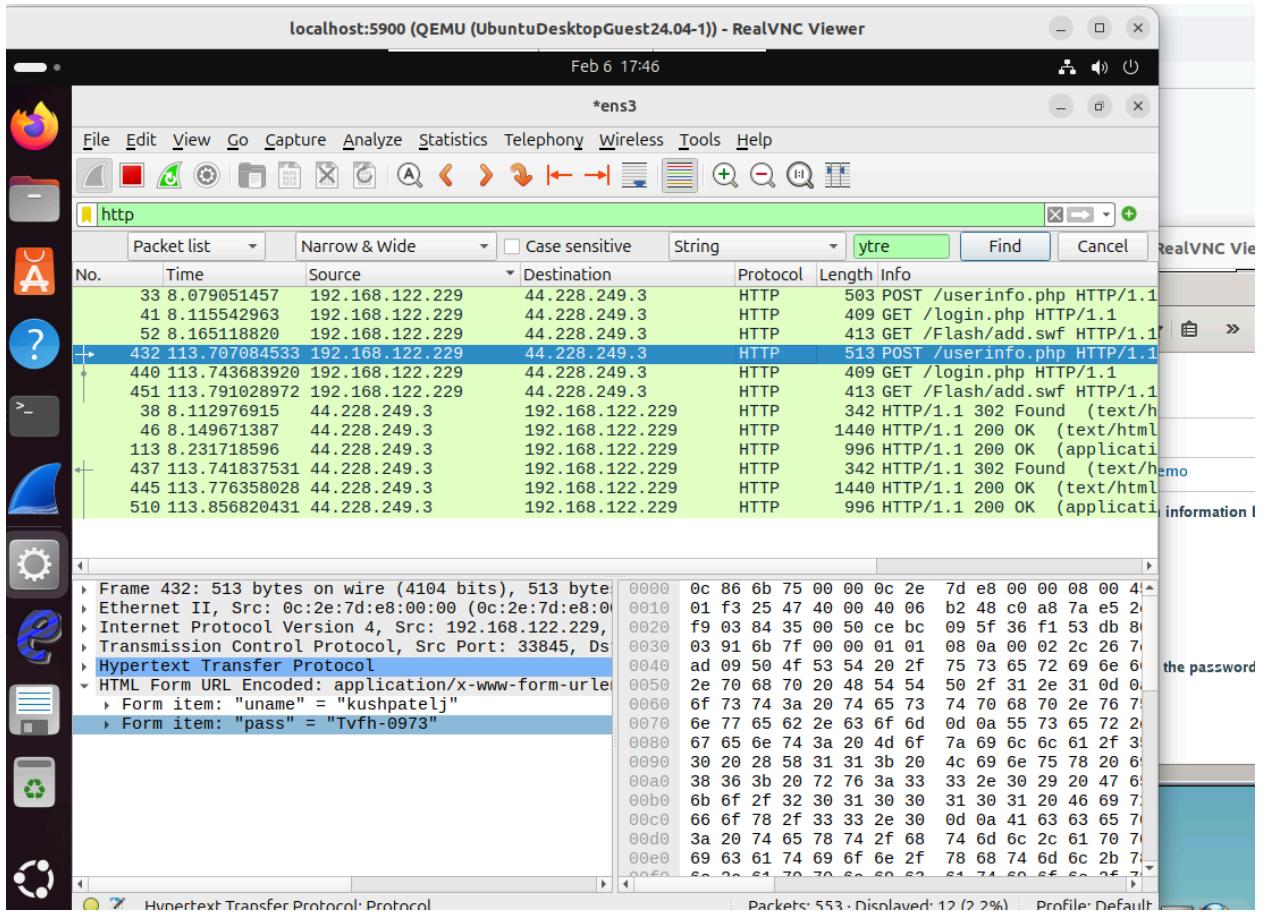


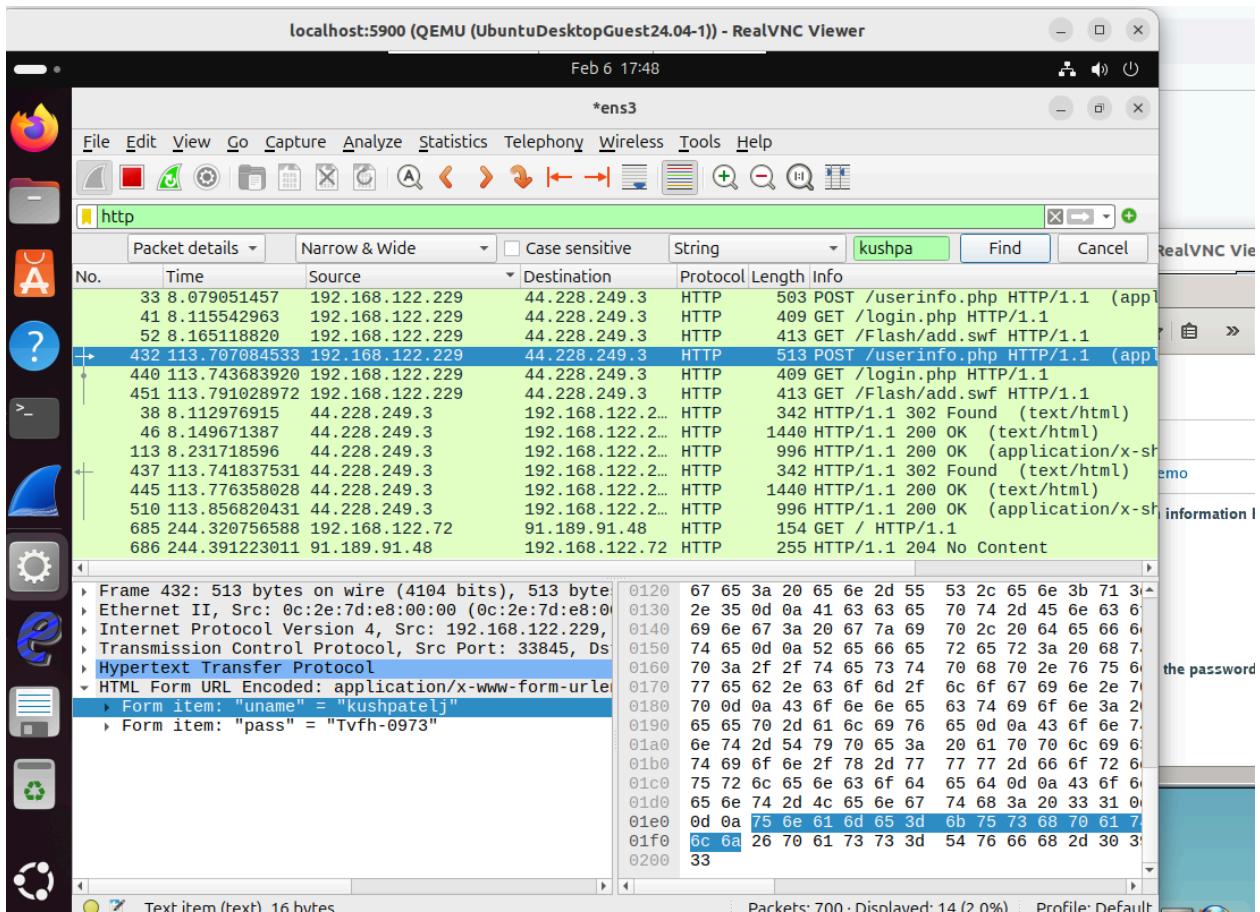


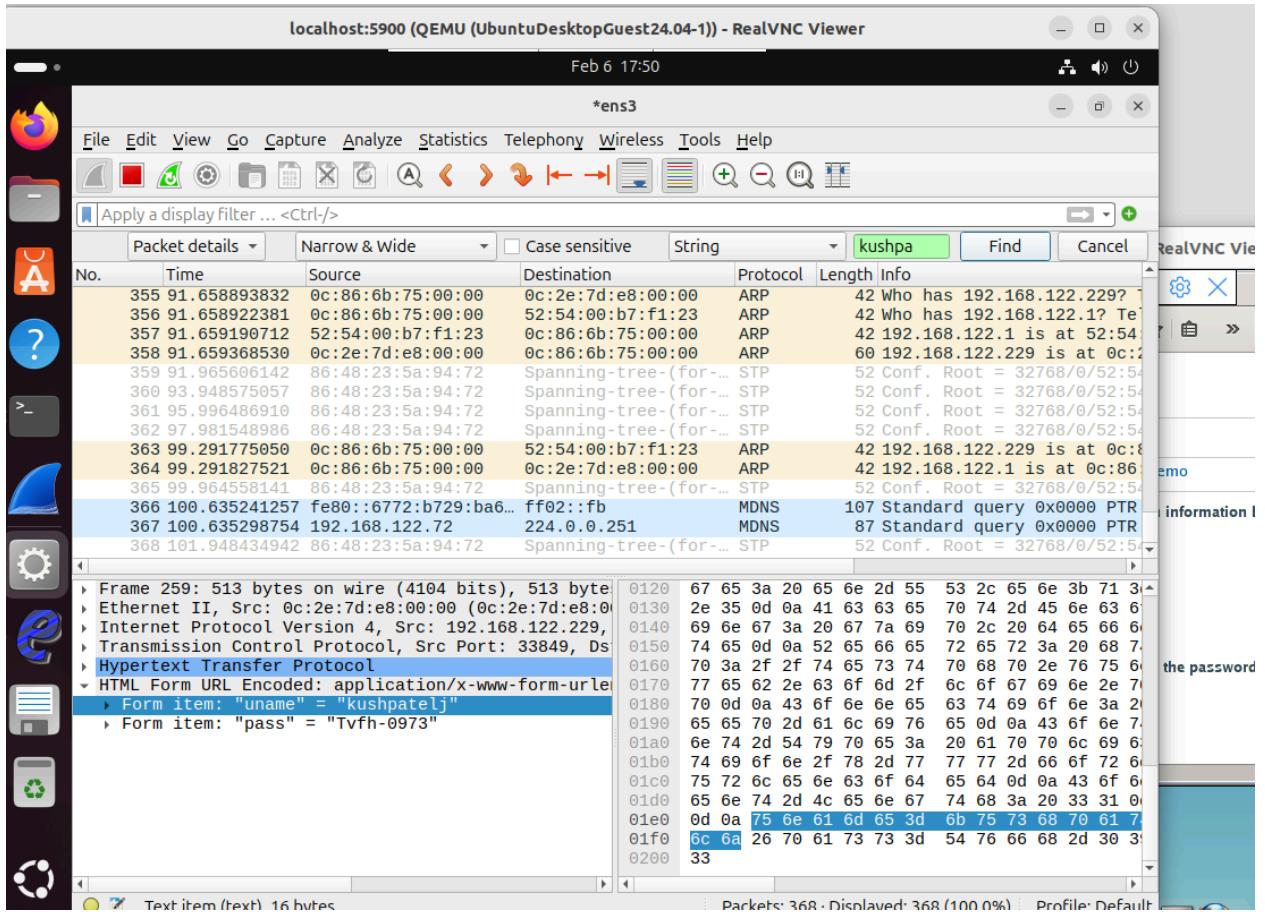










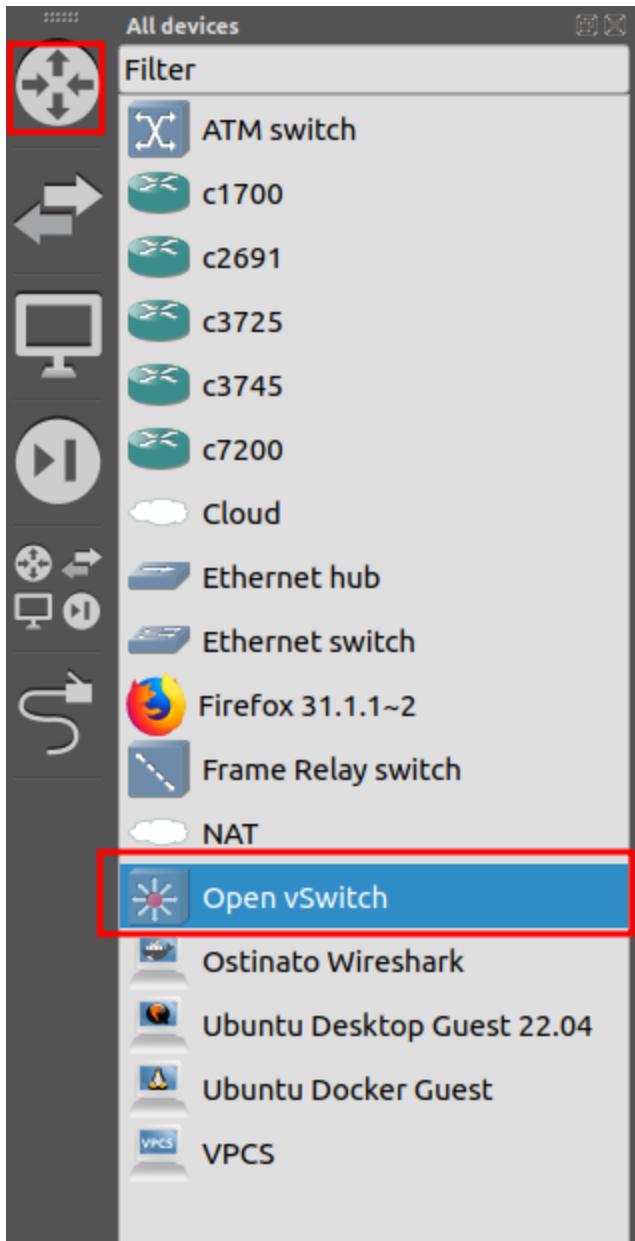


Based on the screenshots shown, when running the wireshark,I can see the username and password when I click the search then click on packet details and string and type part of my username in the find, I'm able to find the username and password I inputted. A passive man in the middle is happening based on the screenshots because i'm eavesdropping on the traffic of the network and intercepting the user login credentials. Since the passwords isn't salted and the http lacks the ssl certificate and encryption, the communication is in raw form, and the attackers will be able to see the password. This is considered very risky and unsafe in the enterprise setting because everyone in the enterprise setting could lose all their information and passwords.

Question 8

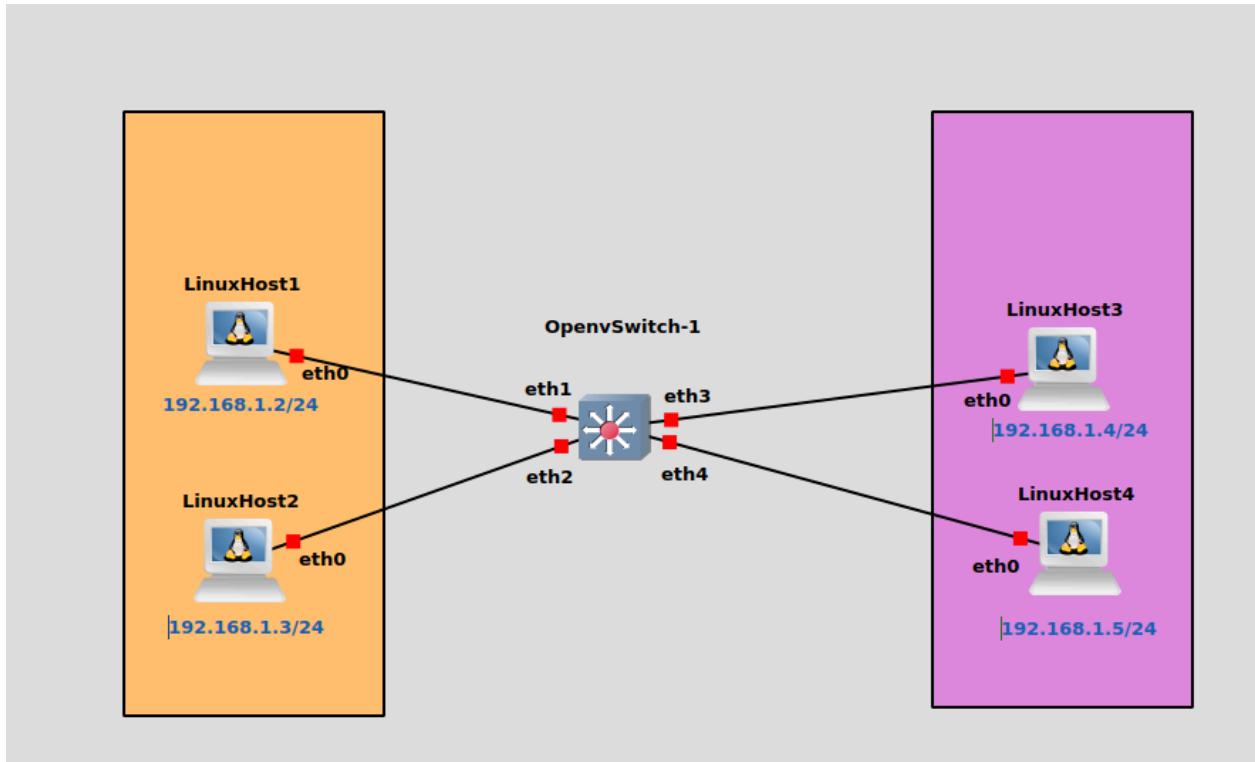
Open vSwitch is a real-world open source operating system used in switch devices. It supports a variety of features including VLANs. In this exercise, you will experiment with Open vSwitch to practice the separate two systems into different VLANs.

Drag the vSwitch into your topology:



Construct the following topology using Ubuntu Docker Guest VMs and configure the VMs with the IPs and subnet masks as follows (please make sure you connect the systems to the ports of the switch exactly as shown below):

- We will create two VLANs on the switch to isolate the systems. LinuxHost1 and LinuxHost2 will be on VLAN 10, and LinuxHost3 and LinuxHost4 will be on VLAN 20.



1. To assign the ports to the VLANs, open the switch console and enter these commands:

- `ovs-vsctl set port eth1 tag=10`
- `ovs-vsctl set port eth2 tag=10`
- `ovs-vsctl set port eth3 tag=20`
- `ovs-vsctl set port eth4 tag=20`
- To verify the configuration, enter this command and take a screenshot of the output of the command "ovs-vsctl show". You should see the VLAN tags under the corresponding ports, like this:

•
•

Port "eth2"

tag: 10

Interface "eth2"

Port "eth3"

tag: 20

Interface "eth3"

.

.

.

2. Next, try to ping LinuxHost2 from LinuxHost1 and take a screenshot of the result. This should work because they are on the same VLAN. Explain why.

3. Next, try to ping LinuxHost3 from LinuxHost4 and take a screenshot of the result. This should also work because they are on the same VLAN. Explain why.

4. Next, try to ping LinuxHost3 from LinuxHost1 and take a screenshot of the result. This should not work because they are on different VLANs. Explain why.

Screenshots

```
root@LinuxHost1:~# ifconfig
Connected to localhost.
Escape character is '^].
LinuxHost1 console is now available... Press RETURN to get started.
ip: RTNETLINK answers: Network is unreachable
root@LinuxHost1:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.2 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:18ff:fedc:e100 prefixlen 64 scopeid 0x20<link>
            ether 02:42:18:dc:e1:00 txqueuelen 1000 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LinuxHost1:~#
```

```
root@LinuxHost2:~# nano /etc/network/interfaces
GNU nano 7.2          /etc/network/interfaces *

#
# This is a sample network config, please uncomment lines to configure the network
#
# Uncomment this line to load custom interface files
# source /etc/network/interfaces.d/*

# Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.1.3
    netmask 255.255.255.0
    gateway 192.168.0.1
    # up echo nameserver 192.168.0.1 > /etc/resolv.conf

# DHCP config for eth0
#auto eth0
#iface eth0 inet dhcp
#    hostname UbuntuDockerGuest-1

[ Read 19 lines ]
^G Help      ^O Write Out ^W Where Is ^K Cut      ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste    ^J Justify   ^/ Go To Line
```

OpenvSwitch-1

```
ovs-docker      ovs-parse-backtrace  ovs-vswitchd
ovs-dpctl       ovs-pki
ovs-dpctl-top   ovs-test
OpenvSwitch-1:$ ovs-
ovs-appctl      ovs-l3ping          ovs-vlan-test
ovs-bugtool     ovs-ofctl           ovs-vsctl
ovs-docker      ovs-parse-backtrace  ovs-vswitchd
ovs-dpctl       ovs-pki
ovs-dpctl-top   ovs-test
OpenvSwitch-1:$ ovs-vs
ovs-vsctl      ovs-vswitchd
OpenvSwitch-1:$ ovs-vsctl set port eth1 tag=10
OpenvSwitch-1:$ ovs-vsctl set port eth2 tag=10
OpenvSwitch-1:$ ovs-vsctl set port eth3 tag=20
OpenvSwitch-1:$ ovs-vsctl set port eth4 tag=20
OpenvSwitch-1:$ ovs-
ovs-appctl      ovs-ofctl           ovs-vswitchd
ovs-bugtool     ovs-parse-backtrace  ovs_comp_helper
ovs-docker      ovs-pki
ovs-dpctl       ovs-test            ovsdb-client
ovs-dpctl-top   ovs-vlan-test       ovsdb-server
ovs-l3ping      ovs-vsctl
OpenvSwitch-1:$ ovs-
ovs-appctl      ovs-l3ping          ovs-vlan-test
ovs-bugtool     ovs-ofctl           ovs-vsctl
```

OpenvSwitch-1

```
ovs-bugtool      ovs-ofctl           ovs-vsctl
ovs-docker      ovs-parse-backtrace  ovs-vswitchd
ovs-dpctl       ovs-pki
ovs-dpctl-top   ovs-test
OpenvSwitch-1:$ ovs-vs
ovs-vsctl      ovs-vswitchd
OpenvSwitch-1:$ ovs-vsctl set port eth1 tag=10
OpenvSwitch-1:$ ovs-vsctl set port eth2 tag=10
OpenvSwitch-1:$ ovs-vsctl set port eth3 tag=20
OpenvSwitch-1:$ ovs-vsctl set port eth4 tag=20
OpenvSwitch-1:$ ovs-
ovs-appctl      ovs-ofctl           ovs-vswitchd
ovs-bugtool     ovs-parse-backtrace  ovs_comp_helper
ovs-docker      ovs-pki
ovs-dpctl       ovs-test            ovsdb-client
ovs-dpctl-top   ovs-vlan-test       ovsdb-server
ovs-l3ping      ovs-vsctl
OpenvSwitch-1:$ ovs-
ovs-appctl      ovs-l3ping          ovs-vlan-test
ovs-bugtool     ovs-ofctl           ovs-vsctl
ovs-docker      ovs-parse-backtrace  ovs-vswitchd
ovs-dpctl       ovs-pki
ovs-dpctl-top   ovs-test
OpenvSwitch-1:$ ovs-vsctl show
```

```
Port eth8
    Interface eth8
Port br0
    Interface br0
        type: internal
Port eth9
    Interface eth9
Port eth11
    Interface eth11
Port eth12
    Interface eth12
Port eth6
    Interface eth6
Port eth15
    Interface eth15
Port eth0
    Interface eth0
Port eth1
    tag: 10
    Interface eth1
Port eth4
    tag: 20
    Interface eth4
OpenvSwitch-1:$
```

```
Bridge br2
    datapath_type: netdev
    Port br2
        Interface br2
            type: internal
Bridge br1
    datapath_type: netdev
    Port br1
        Interface br1
            type: internal
Bridge br3
    datapath_type: netdev
    Port br3
        Interface br3
            type: internal
Bridge br0
    datapath_type: netdev
    Port eth13
        Interface eth13
    Port eth7
        Interface eth7
    Port eth10
        Interface eth10
    Port eth5
```

OpenvSwitch-1

```
ovs-dpctl      ovs-test      ovsdb-server
ovs-dpctl-top  ovs-vlan-test ovsdb-tool
ovs-l3ping     ovs-vsctl
OpenvSwitch-1:$ ovs-
ovs-appctl     ovs-l3ping   ovs-vlan-test
ovs-bugtool    ovs-ofctl    ovs-vsctl
ovs-docker     ovs-parse-backtrace ovs-vswitchd
ovs-dpctl     ovs-pki
ovs-dpctl-top  ovs-test
OpenvSwitch-1:$ ovs-vsctl show
39659ed5-24be-4ec9-8441-9f04f78d35f3
  Bridge br2
    datapath_type: netdev
    Port br2
      Interface br2
        type: internal
  Bridge br1
    datapath_type: netdev
    Port br1
      Interface br1
        type: internal
  Bridge br3
    datapath_type: netdev
    Port br3
```

OpenvSwitch-1

```
  Interface eth2
  Port eth8
    Interface eth8
  Port br0
    Interface br0
      type: internal
  Port eth9
    Interface eth9
  Port eth11
    Interface eth11
  Port eth12
    Interface eth12
  Port eth6
    Interface eth6
  Port eth15
    Interface eth15
  Port eth0
    Interface eth0
  Port eth1
    tag: 10
    Interface eth1
  Port eth4
    tag: 20
    Interface eth4
OpenvSwitch-1:$
```

```
root@LinuxHost1:~# ifconfig  
OpenvSwitch-1      inet6 fe80::42:18ff:fedc:e100  prefixlen 64  scopeid 0x20<link>  
                    ether 02:42:18:dc:e1:00  txqueuelen 1000  (Ethernet)  
                    RX packets 1  bytes 70 (70.0 B)  
                    RX errors 0  dropped 0  overruns 0  frame 0  
                    TX packets 7  bytes 586 (586.0 B)  
                    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0  
  
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 0  bytes 0 (0.0 B)  
        RX errors 0  dropped 0  overruns 0  frame 0  
        TX packets 0  bytes 0 (0.0 B)  
        TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0  
  
root@LinuxHost1:~# ping 192.168.1.3  
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.  
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=15.2 ms  
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=4.59 ms  
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=4.61 ms  
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=6.08 ms  
64 bytes from 192.168.1.3: icmp_seq=5 ttl=64 time=5.04 ms
```

```
root@LinuxHost1:~# ping 192.168.1.3  
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.  
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=15.2 ms  
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=4.59 ms  
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=4.61 ms  
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=6.08 ms  
64 bytes from 192.168.1.3: icmp_seq=5 ttl=64 time=5.04 ms  
64 bytes from 192.168.1.3: icmp_seq=6 ttl=64 time=3.65 ms  
64 bytes from 192.168.1.3: icmp_seq=7 ttl=64 time=6.05 ms  
64 bytes from 192.168.1.3: icmp_seq=8 ttl=64 time=6.15 ms  
64 bytes from 192.168.1.3: icmp_seq=9 ttl=64 time=5.80 ms  
64 bytes from 192.168.1.3: icmp_seq=10 ttl=64 time=3.04 ms  
64 bytes from 192.168.1.3: icmp_seq=11 ttl=64 time=5.36 ms  
64 bytes from 192.168.1.3: icmp_seq=12 ttl=64 time=4.74 ms  
64 bytes from 192.168.1.3: icmp_seq=13 ttl=64 time=6.07 ms  
64 bytes from 192.168.1.3: icmp_seq=14 ttl=64 time=3.37 ms  
64 bytes from 192.168.1.3: icmp_seq=15 ttl=64 time=5.79 ms  
64 bytes from 192.168.1.3: icmp_seq=16 ttl=64 time=5.03 ms  
64 bytes from 192.168.1.3: icmp_seq=17 ttl=64 time=5.05 ms  
64 bytes from 192.168.1.3: icmp_seq=18 ttl=64 time=5.09 ms  
64 bytes from 192.168.1.3: icmp_seq=19 ttl=64 time=3.88 ms  
64 bytes from 192.168.1.3: icmp_seq=20 ttl=64 time=4.30 ms
```

```
root@LinuxHost1: ~
OpenvSwitch-1
root@LinuxHost1: ~

64 bytes from 192.168.1.3: icmp_seq=32 ttl=64 time=5.67 ms
64 bytes from 192.168.1.3: icmp_seq=33 ttl=64 time=5.51 ms
64 bytes from 192.168.1.3: icmp_seq=34 ttl=64 time=3.31 ms
64 bytes from 192.168.1.3: icmp_seq=35 ttl=64 time=4.73 ms
64 bytes from 192.168.1.3: icmp_seq=36 ttl=64 time=5.59 ms
64 bytes from 192.168.1.3: icmp_seq=37 ttl=64 time=5.81 ms
64 bytes from 192.168.1.3: icmp_seq=38 ttl=64 time=5.87 ms
64 bytes from 192.168.1.3: icmp_seq=39 ttl=64 time=3.12 ms
64 bytes from 192.168.1.3: icmp_seq=40 ttl=64 time=5.88 ms
64 bytes from 192.168.1.3: icmp_seq=41 ttl=64 time=5.58 ms
64 bytes from 192.168.1.3: icmp_seq=42 ttl=64 time=5.59 ms
64 bytes from 192.168.1.3: icmp_seq=43 ttl=64 time=5.98 ms
64 bytes from 192.168.1.3: icmp_seq=44 ttl=64 time=2.99 ms
64 bytes from 192.168.1.3: icmp_seq=45 ttl=64 time=5.58 ms
64 bytes from 192.168.1.3: icmp_seq=46 ttl=64 time=5.30 ms
64 bytes from 192.168.1.3: icmp_seq=47 ttl=64 time=5.16 ms
64 bytes from 192.168.1.3: icmp_seq=48 ttl=64 time=5.13 ms
64 bytes from 192.168.1.3: icmp_seq=49 ttl=64 time=5.44 ms
64 bytes from 192.168.1.3: icmp_seq=50 ttl=64 time=4.97 ms
^C
--- 192.168.1.3 ping statistics ---
50 packets transmitted, 50 received, 0% packet loss, time 49071ms
rtt min/avg/max/mdev = 2.338/5.269/15.210/1.729 ms
root@LinuxHost1: #
```

```
root@LinuxHost4: ~
OpenvSwitch-1
root@LinuxHost4: ~

Connected to localhost.
Escape character is '^]'.
LinuxHost4 console is now available... Press RETURN to get started.
ip: RTNETLINK answers: Network is unreachable
root@LinuxHost4:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.5 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::42:30ff:fe5:8d00 prefixlen 64 scopeid 0x20<link>
            ether 02:42:30:f5:8d:00 txqueuelen 1000 (Ethernet)
            RX packets 11 bytes 866 (866.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 11 bytes 866 (866.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
root@LinuxHost4: #
```

```
root@LinuxHost4:~# ping 192.168.1.4
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.
64 bytes from 192.168.1.4: icmp_seq=1 ttl=64 time=3.93 ms
64 bytes from 192.168.1.4: icmp_seq=2 ttl=64 time=3.11 ms
64 bytes from 192.168.1.4: icmp_seq=3 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=4 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=5 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=6 ttl=64 time=5.60 ms
64 bytes from 192.168.1.4: icmp_seq=7 ttl=64 time=3.18 ms
64 bytes from 192.168.1.4: icmp_seq=8 ttl=64 time=3.22 ms
64 bytes from 192.168.1.4: icmp_seq=9 ttl=64 time=3.17 ms
64 bytes from 192.168.1.4: icmp_seq=10 ttl=64 time=2.86 ms
64 bytes from 192.168.1.4: icmp_seq=11 ttl=64 time=2.89 ms
64 bytes from 192.168.1.4: icmp_seq=12 ttl=64 time=2.69 ms
64 bytes from 192.168.1.4: icmp_seq=13 ttl=64 time=3.04 ms
64 bytes from 192.168.1.4: icmp_seq=14 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=15 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=16 ttl=64 time=2.86 ms
64 bytes from 192.168.1.4: icmp_seq=17 ttl=64 time=5.05 ms
64 bytes from 192.168.1.4: icmp_seq=18 ttl=64 time=5.33 ms
64 bytes from 192.168.1.4: icmp_seq=19 ttl=64 time=2.76 ms
```

```
64 bytes from 192.168.1.4: icmp_seq=13 ttl=64 time=3.04 ms
64 bytes from 192.168.1.4: icmp_seq=14 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=15 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=16 ttl=64 time=2.86 ms
64 bytes from 192.168.1.4: icmp_seq=17 ttl=64 time=5.05 ms
64 bytes from 192.168.1.4: icmp_seq=18 ttl=64 time=5.33 ms
64 bytes from 192.168.1.4: icmp_seq=19 ttl=64 time=2.76 ms
64 bytes from 192.168.1.4: icmp_seq=20 ttl=64 time=5.44 ms
64 bytes from 192.168.1.4: icmp_seq=21 ttl=64 time=2.96 ms
64 bytes from 192.168.1.4: icmp_seq=22 ttl=64 time=2.87 ms
64 bytes from 192.168.1.4: icmp_seq=23 ttl=64 time=3.66 ms
64 bytes from 192.168.1.4: icmp_seq=24 ttl=64 time=3.07 ms
64 bytes from 192.168.1.4: icmp_seq=25 ttl=64 time=3.05 ms
64 bytes from 192.168.1.4: icmp_seq=26 ttl=64 time=3.36 ms
64 bytes from 192.168.1.4: icmp_seq=27 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=28 ttl=64 time=3.40 ms
64 bytes from 192.168.1.4: icmp_seq=29 ttl=64 time=3.03 ms
64 bytes from 192.168.1.4: icmp_seq=30 ttl=64 time=3.03 ms
64 bytes from 192.168.1.4: icmp_seq=31 ttl=64 time=3.48 ms
^C
--- 192.168.1.4 ping statistics ---
31 packets transmitted, 31 received, 0% packet loss, time 30049ms
rtt min/avg/max/mdev = 2.693/3.411/5.600/0.790 ms
root@LinuxHost4:~#
```

```
root@LinuxHost1:~# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
      inet 192.168.1.2 netmask 255.255.255.0 broadcast 0.0.0.0  
      inet6 fe80::42:18ff:fedc:e100 prefixlen 64 scopeid 0x20<link>  
        ether 02:42:18:dc:e1:00 txqueuelen 1000 (Ethernet)  
          RX packets 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 8 bytes 656 (656.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
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 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 0 bytes 0 (0.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@LinuxHost1:~# ping 192.168.1.4  
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.  
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=2 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=3 Destination Host Unreachable
```

```
root@LinuxHost1:~# ifconfig  
inet6 ::1 prefixlen 128 scopeid 0x10<host>  
loop txqueuelen 1000 (Local Loopback)  
RX packets 0 bytes 0 (0.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 0 bytes 0 (0.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@LinuxHost1:~# ping 192.168.1.4  
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.  
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=2 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=3 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=4 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=5 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=6 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=8 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=9 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=10 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=11 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=12 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=13 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=14 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=15 Destination Host Unreachable
```

```
root@LinuxHost1:~# ping 192.168.1.4
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable
From 192.168.1.2 icmp_seq=2 Destination Host Unreachable
From 192.168.1.2 icmp_seq=3 Destination Host Unreachable
From 192.168.1.2 icmp_seq=4 Destination Host Unreachable
From 192.168.1.2 icmp_seq=5 Destination Host Unreachable
From 192.168.1.2 icmp_seq=6 Destination Host Unreachable
From 192.168.1.2 icmp_seq=8 Destination Host Unreachable
From 192.168.1.2 icmp_seq=9 Destination Host Unreachable
From 192.168.1.2 icmp_seq=10 Destination Host Unreachable
From 192.168.1.2 icmp_seq=11 Destination Host Unreachable
From 192.168.1.2 icmp_seq=12 Destination Host Unreachable
From 192.168.1.2 icmp_seq=13 Destination Host Unreachable
From 192.168.1.2 icmp_seq=14 Destination Host Unreachable
From 192.168.1.2 icmp_seq=15 Destination Host Unreachable
From 192.168.1.2 icmp_seq=16 Destination Host Unreachable
From 192.168.1.2 icmp_seq=17 Destination Host Unreachable
From 192.168.1.2 icmp_seq=18 Destination Host Unreachable
From 192.168.1.2 icmp_seq=19 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.1.2 icmp_seq=20 Destination Host Unreachable
From 192.168.1.2 icmp_seq=21 Destination Host Unreachable
```

```
From 192.168.1.2 icmp_seq=14 Destination Host Unreachable
From 192.168.1.2 icmp_seq=15 Destination Host Unreachable
From 192.168.1.2 icmp_seq=16 Destination Host Unreachable
From 192.168.1.2 icmp_seq=17 Destination Host Unreachable
From 192.168.1.2 icmp_seq=18 Destination Host Unreachable
From 192.168.1.2 icmp_seq=19 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.1.2 icmp_seq=20 Destination Host Unreachable
From 192.168.1.2 icmp_seq=21 Destination Host Unreachable
From 192.168.1.2 icmp_seq=23 Destination Host Unreachable
From 192.168.1.2 icmp_seq=24 Destination Host Unreachable
From 192.168.1.2 icmp_seq=25 Destination Host Unreachable
From 192.168.1.2 icmp_seq=26 Destination Host Unreachable
From 192.168.1.2 icmp_seq=27 Destination Host Unreachable
From 192.168.1.2 icmp_seq=28 Destination Host Unreachable
From 192.168.1.2 icmp_seq=29 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.1.2 icmp_seq=30 Destination Host Unreachable
From 192.168.1.2 icmp_seq=31 Destination Host Unreachable
^C
--- 192.168.1.4 ping statistics ---
34 packets transmitted, 0 received, +29 errors, 100% packet loss, time 33786ms
pipe 4
root@LinuxHost1:~#
```

```
root@LinuxHost1:~
```

```
OpenvSwitch-1      x      root@LinuxHost4:~ x      root@LinuxHost1:~ x
```

```
64 bytes from 192.168.1.3: icmp_seq=18 ttl=64 time=4.64 ms
64 bytes from 192.168.1.3: icmp_seq=19 ttl=64 time=4.54 ms
64 bytes from 192.168.1.3: icmp_seq=20 ttl=64 time=4.69 ms
64 bytes from 192.168.1.3: icmp_seq=21 ttl=64 time=4.54 ms
64 bytes from 192.168.1.3: icmp_seq=22 ttl=64 time=4.86 ms
64 bytes from 192.168.1.3: icmp_seq=23 ttl=64 time=4.60 ms
64 bytes from 192.168.1.3: icmp_seq=24 ttl=64 time=4.59 ms
64 bytes from 192.168.1.3: icmp_seq=25 ttl=64 time=4.59 ms
64 bytes from 192.168.1.3: icmp_seq=26 ttl=64 time=4.04 ms
64 bytes from 192.168.1.3: icmp_seq=27 ttl=64 time=4.82 ms
64 bytes from 192.168.1.3: icmp_seq=28 ttl=64 time=3.31 ms
64 bytes from 192.168.1.3: icmp_seq=29 ttl=64 time=4.63 ms
64 bytes from 192.168.1.3: icmp_seq=30 ttl=64 time=4.61 ms
64 bytes from 192.168.1.3: icmp_seq=31 ttl=64 time=4.61 ms
64 bytes from 192.168.1.3: icmp_seq=32 ttl=64 time=4.62 ms
64 bytes from 192.168.1.3: icmp_seq=33 ttl=64 time=4.59 ms
64 bytes from 192.168.1.3: icmp_seq=34 ttl=64 time=4.63 ms
64 bytes from 192.168.1.3: icmp_seq=35 ttl=64 time=4.87 ms
64 bytes from 192.168.1.3: icmp_seq=36 ttl=64 time=4.93 ms
64 bytes from 192.168.1.3: icmp_seq=37 ttl=64 time=4.94 ms
64 bytes from 192.168.1.3: icmp_seq=38 ttl=64 time=4.93 ms
64 bytes from 192.168.1.3: icmp_seq=39 ttl=64 time=4.87 ms
64 bytes from 192.168.1.3: icmp_seq=40 ttl=64 time=4.99 ms
```

```
root@LinuxHost1:~
```

```
OpenvSwitch-1      x      root@LinuxHost4:~ x      root@LinuxHost1:~ x
```

```
64 bytes from 192.168.1.3: icmp_seq=25 ttl=64 time=4.59 ms
64 bytes from 192.168.1.3: icmp_seq=26 ttl=64 time=4.04 ms
64 bytes from 192.168.1.3: icmp_seq=27 ttl=64 time=4.82 ms
64 bytes from 192.168.1.3: icmp_seq=28 ttl=64 time=3.31 ms
64 bytes from 192.168.1.3: icmp_seq=29 ttl=64 time=4.63 ms
64 bytes from 192.168.1.3: icmp_seq=30 ttl=64 time=4.61 ms
64 bytes from 192.168.1.3: icmp_seq=31 ttl=64 time=4.61 ms
64 bytes from 192.168.1.3: icmp_seq=32 ttl=64 time=4.62 ms
64 bytes from 192.168.1.3: icmp_seq=33 ttl=64 time=4.59 ms
64 bytes from 192.168.1.3: icmp_seq=34 ttl=64 time=4.63 ms
64 bytes from 192.168.1.3: icmp_seq=35 ttl=64 time=4.87 ms
64 bytes from 192.168.1.3: icmp_seq=36 ttl=64 time=4.93 ms
64 bytes from 192.168.1.3: icmp_seq=37 ttl=64 time=4.94 ms
64 bytes from 192.168.1.3: icmp_seq=38 ttl=64 time=4.93 ms
64 bytes from 192.168.1.3: icmp_seq=39 ttl=64 time=4.87 ms
64 bytes from 192.168.1.3: icmp_seq=40 ttl=64 time=4.99 ms
64 bytes from 192.168.1.3: icmp_seq=41 ttl=64 time=5.83 ms
64 bytes from 192.168.1.3: icmp_seq=42 ttl=64 time=5.47 ms
64 bytes from 192.168.1.3: icmp_seq=43 ttl=64 time=6.00 ms
^C
--- 192.168.1.3 ping statistics ---
43 packets transmitted, 43 received, 0% packet loss, time 42068ms
rtt min/avg/max/mdev = 1.841/5.014/20.205/2.441 ms
root@LinuxHost1:~#
```

1. Screenshots for part 1

The screenshot shows a terminal window titled "OpenvSwitch-1". The window contains a list of Open vSwitch tools and their descriptions, followed by several configuration commands. The configuration commands include setting VLAN tags for ports eth1 through eth4 and listing them again. Finally, the "ovs-vsctl show" command is run to display the current Open vSwitch configuration.

```
ovs-bugtool      ovs-ofctl      ovs-vsctl
ovs-docker       ovs-parse-backtrace  ovs-vswitchd
ovs-dpctl        ovs-pki
ovs-dpctl-top    ovs-test
OpenvSwitch-1:$ ovs-vs
ovs-vsctl       ovs-vswitchd
OpenvSwitch-1:$ ovs-vsctl set port eth1 tag=10
OpenvSwitch-1:$ ovs-vsctl set port eth2 tag=10
OpenvSwitch-1:$ ovs-vsctl set port eth3 tag=20
OpenvSwitch-1:$ ovs-vsctl set port eth4 tag=20
OpenvSwitch-1:$ ovs-
ovs-appctl       ovs-ofctl      ovs-vswitchd
ovs-bugtool       ovs-parse-backtrace  ovs_comp_helper
ovs-docker        ovs-pki
ovs-dpctl         ovs-test
ovs-dpctl-top    ovs-vlan-test   ovsdb-client
ovs-l3ping        ovs-vsctl
OpenvSwitch-1:$ ovs-
ovs-appctl       ovs-l3ping     ovs-vlan-test
ovs-bugtool       ovs-ofctl      ovs-vsctl
ovs-docker        ovs-parse-backtrace  ovs-vswitchd
ovs-dpctl         ovs-pki
ovs-dpctl-top    ovs-test
OpenvSwitch-1:$ ovs-vsctl show
```

```
Port eth8
    Interface eth8
Port br0
    Interface br0
        type: internal
Port eth9
    Interface eth9
Port eth11
    Interface eth11
Port eth12
    Interface eth12
Port eth6
    Interface eth6
Port eth15
    Interface eth15
Port eth0
    Interface eth0
Port eth1
    tag: 10
    Interface eth1
Port eth4
    tag: 20
    Interface eth4
OpenvSwitch-1:$
```

```
Bridge br2
    datapath_type: netdev
    Port br2
        Interface br2
            type: internal
Bridge br1
    datapath_type: netdev
    Port br1
        Interface br1
            type: internal
Bridge br3
    datapath_type: netdev
    Port br3
        Interface br3
            type: internal
Bridge br0
    datapath_type: netdev
    Port eth13
        Interface eth13
    Port eth7
        Interface eth7
    Port eth10
        Interface eth10
    Port eth5
```

OpenvSwitch-1

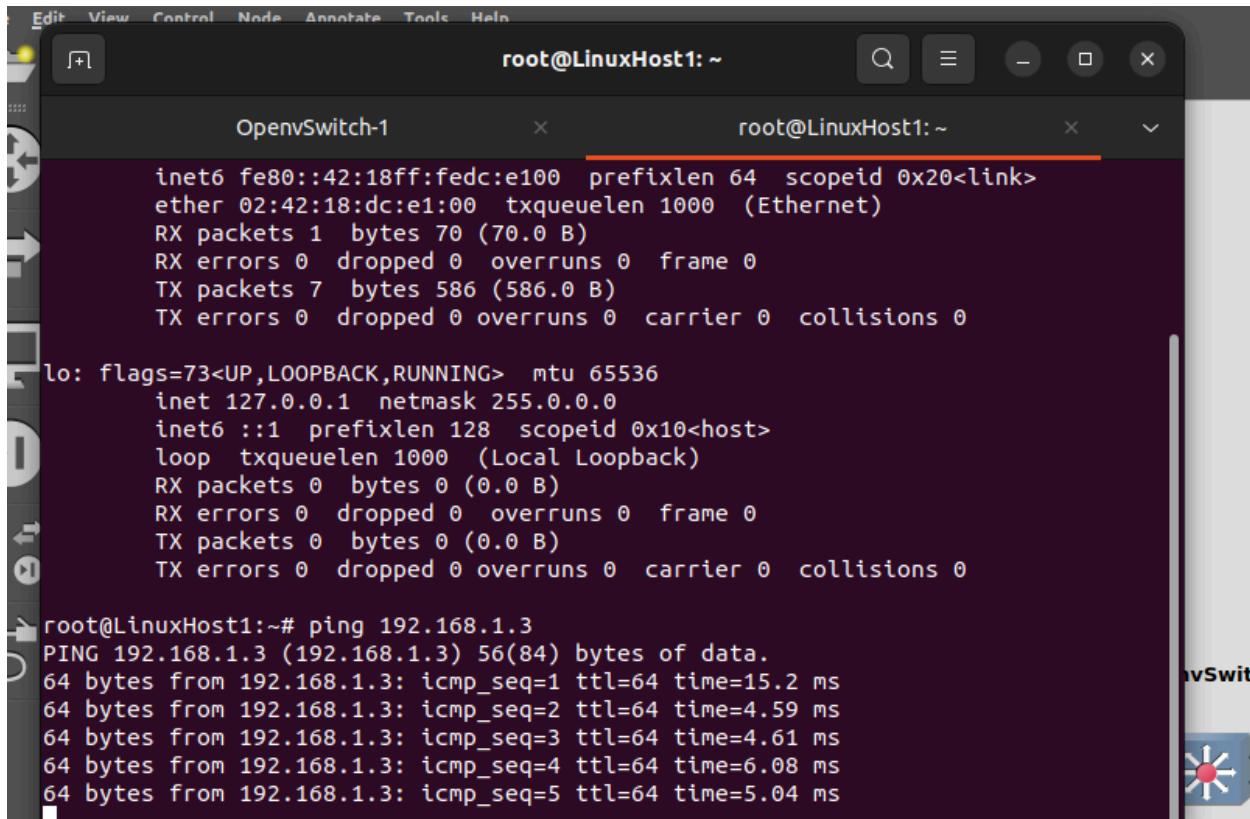
```
ovs-dpctl      ovs-test      ovsdb-server
ovs-dpctl-top  ovs-vlan-test ovsdb-tool
ovs-l3ping     ovs-vsctl
OpenvSwitch-1:$ ovs-
ovs-appctl     ovs-l3ping   ovs-vlan-test
ovs-bugtool    ovs-ofctl    ovs-vsctl
ovs-docker     ovs-parse-backtrace ovs-vswitchd
ovs-dpctl     ovs-pki
ovs-dpctl-top  ovs-test
OpenvSwitch-1:$ ovs-vsctl show
39659ed5-24be-4ec9-8441-9f04f78d35f3
  Bridge br2
    datapath_type: netdev
    Port br2
      Interface br2
        type: internal
  Bridge br1
    datapath_type: netdev
    Port br1
      Interface br1
        type: internal
  Bridge br3
    datapath_type: netdev
    Port br3
```

OpenvSwitch-1

```
  Interface eth2
  Port eth8
    Interface eth8
  Port br0
    Interface br0
      type: internal
  Port eth9
    Interface eth9
  Port eth11
    Interface eth11
  Port eth12
    Interface eth12
  Port eth6
    Interface eth6
  Port eth15
    Interface eth15
  Port eth0
    Interface eth0
  Port eth1
    tag: 10
    Interface eth1
  Port eth4
    tag: 20
    Interface eth4
OpenvSwitch-1:$
```

Based on the screenshots when I inputted the command, ovs-vsctl set port eth1 tag=10 ,ovs-vsctl set port eth2 tag=10, ovs-vsctl set port eth3 tag=20, ovs-vsctl set port eth4 tag=20, ,ovs-vsctl-show, it shows that the devices connected to eth1 and eth2 are in VLAN 10 based on tag=10 and eth3 and eth4 are in VLAN 20 based on tag=20

2.Screenshots for part 2



The screenshot shows a terminal window with two tabs. The left tab is titled 'OpenvSwitch-1' and the right tab is titled 'root@LinuxHost1:~'. The terminal displays the following output:

```
inet6 fe80::42:18ff:fedc:e100  prefixlen 64  scopeid 0x20<link>
ether 02:42:18:dc:e1:00  txqueuelen 1000  (Ethernet)
RX packets 1  bytes 70 (70.0 B)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 7  bytes 586 (586.0 B)
TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

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 0  bytes 0 (0.0 B)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 0  bytes 0 (0.0 B)
TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

root@LinuxHost1:~# ping 192.168.1.3
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=15.2 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=4.59 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=4.61 ms
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=6.08 ms
64 bytes from 192.168.1.3: icmp_seq=5 ttl=64 time=5.04 ms
```

```
root@LinuxHost1:~# ifconfig  
OpenvSwitch-1      inet6 fe80::42:18ff:fedc:e100  prefixlen 64  scopeid 0x20<link>  
                    ether 02:42:18:dc:e1:00  txqueuelen 1000  (Ethernet)  
                    RX packets 1  bytes 70 (70.0 B)  
                    RX errors 0  dropped 0  overruns 0  frame 0  
                    TX packets 7  bytes 586 (586.0 B)  
                    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0  
  
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 0  bytes 0 (0.0 B)  
        RX errors 0  dropped 0  overruns 0  frame 0  
        TX packets 0  bytes 0 (0.0 B)  
        TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0  
  
root@LinuxHost1:~# ping 192.168.1.3  
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.  
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=15.2 ms  
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=4.59 ms  
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=4.61 ms  
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=6.08 ms  
64 bytes from 192.168.1.3: icmp_seq=5 ttl=64 time=5.04 ms
```

```
root@LinuxHost1:~# ping 192.168.1.3  
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.  
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=15.2 ms  
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=4.59 ms  
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=4.61 ms  
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=6.08 ms  
64 bytes from 192.168.1.3: icmp_seq=5 ttl=64 time=5.04 ms  
64 bytes from 192.168.1.3: icmp_seq=6 ttl=64 time=3.65 ms  
64 bytes from 192.168.1.3: icmp_seq=7 ttl=64 time=6.05 ms  
64 bytes from 192.168.1.3: icmp_seq=8 ttl=64 time=6.15 ms  
64 bytes from 192.168.1.3: icmp_seq=9 ttl=64 time=5.80 ms  
64 bytes from 192.168.1.3: icmp_seq=10 ttl=64 time=3.04 ms  
64 bytes from 192.168.1.3: icmp_seq=11 ttl=64 time=5.36 ms  
64 bytes from 192.168.1.3: icmp_seq=12 ttl=64 time=4.74 ms  
64 bytes from 192.168.1.3: icmp_seq=13 ttl=64 time=6.07 ms  
64 bytes from 192.168.1.3: icmp_seq=14 ttl=64 time=3.37 ms  
64 bytes from 192.168.1.3: icmp_seq=15 ttl=64 time=5.79 ms  
64 bytes from 192.168.1.3: icmp_seq=16 ttl=64 time=5.03 ms  
64 bytes from 192.168.1.3: icmp_seq=17 ttl=64 time=5.05 ms  
64 bytes from 192.168.1.3: icmp_seq=18 ttl=64 time=5.09 ms  
64 bytes from 192.168.1.3: icmp_seq=19 ttl=64 time=3.88 ms  
64 bytes from 192.168.1.3: icmp_seq=20 ttl=64 time=4.30 ms
```

The screenshot shows a terminal window titled "root@LinuxHost1:~". The window has two tabs: "OpenvSwitch-1" and "root@LinuxHost1:~". The "root@LinuxHost1:~" tab is active and displays the following output:

```
64 bytes from 192.168.1.3: icmp_seq=32 ttl=64 time=5.67 ms
64 bytes from 192.168.1.3: icmp_seq=33 ttl=64 time=5.51 ms
64 bytes from 192.168.1.3: icmp_seq=34 ttl=64 time=3.31 ms
64 bytes from 192.168.1.3: icmp_seq=35 ttl=64 time=4.73 ms
64 bytes from 192.168.1.3: icmp_seq=36 ttl=64 time=5.59 ms
64 bytes from 192.168.1.3: icmp_seq=37 ttl=64 time=5.81 ms
64 bytes from 192.168.1.3: icmp_seq=38 ttl=64 time=5.87 ms
64 bytes from 192.168.1.3: icmp_seq=39 ttl=64 time=3.12 ms
64 bytes from 192.168.1.3: icmp_seq=40 ttl=64 time=5.88 ms
64 bytes from 192.168.1.3: icmp_seq=41 ttl=64 time=5.58 ms
64 bytes from 192.168.1.3: icmp_seq=42 ttl=64 time=5.59 ms
64 bytes from 192.168.1.3: icmp_seq=43 ttl=64 time=5.98 ms
64 bytes from 192.168.1.3: icmp_seq=44 ttl=64 time=2.99 ms
64 bytes from 192.168.1.3: icmp_seq=45 ttl=64 time=5.58 ms
64 bytes from 192.168.1.3: icmp_seq=46 ttl=64 time=5.30 ms
64 bytes from 192.168.1.3: icmp_seq=47 ttl=64 time=5.16 ms
64 bytes from 192.168.1.3: icmp_seq=48 ttl=64 time=5.13 ms
64 bytes from 192.168.1.3: icmp_seq=49 ttl=64 time=5.44 ms
64 bytes from 192.168.1.3: icmp_seq=50 ttl=64 time=4.97 ms
^C
--- 192.168.1.3 ping statistics ---
50 packets transmitted, 50 received, 0% packet loss, time 49071ms
rtt min/avg/max/mdev = 2.338/5.269/15.210/1.729 ms
root@LinuxHost1:~#
```

Based on the screenshot, LinuxHost1 pinged to LinuxHost2 and when it did it was able to scan all packets that were going on in LinuxHost2, because they are both part of the same VLAN(VLAN10) based on tag=10, LinuxHost1 can check the network activity of LinuxHost2, it sends a message to that specific address in the ping and after the ping ends it shows the ping stats of the ip address of the device it pinged and it shows that all the packets have been successfully received and there has been no packet loss. Since they were on the same VLAN no packets were lost. They should be able to connect with each other as they are both physically and logically connected.

3.Screenshots for part 3

```
root@LinuxHost4:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.5 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 fe80::42:30ff:fe5:8d00 prefixlen 64 scopeid 0x20<link>
        ether 02:42:30:f5:8d:00 txqueuelen 1000 (Ethernet)
            RX packets 11 bytes 866 (866.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 11 bytes 866 (866.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
root@LinuxHost4:~# ping 192.168.1.4
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.
64 bytes from 192.168.1.4: icmp_seq=1 ttl=64 time=3.93 ms
64 bytes from 192.168.1.4: icmp_seq=2 ttl=64 time=3.11 ms
64 bytes from 192.168.1.4: icmp_seq=3 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=4 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=5 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=6 ttl=64 time=5.60 ms
64 bytes from 192.168.1.4: icmp_seq=7 ttl=64 time=3.18 ms
64 bytes from 192.168.1.4: icmp_seq=8 ttl=64 time=3.22 ms
64 bytes from 192.168.1.4: icmp_seq=9 ttl=64 time=3.17 ms
64 bytes from 192.168.1.4: icmp_seq=10 ttl=64 time=2.86 ms
64 bytes from 192.168.1.4: icmp_seq=11 ttl=64 time=2.89 ms
64 bytes from 192.168.1.4: icmp_seq=12 ttl=64 time=2.69 ms
64 bytes from 192.168.1.4: icmp_seq=13 ttl=64 time=3.04 ms
64 bytes from 192.168.1.4: icmp_seq=14 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=15 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=16 ttl=64 time=2.86 ms
64 bytes from 192.168.1.4: icmp_seq=17 ttl=64 time=5.05 ms
64 bytes from 192.168.1.4: icmp_seq=18 ttl=64 time=5.33 ms
64 bytes from 192.168.1.4: icmp_seq=19 ttl=64 time=2.76 ms
```

The screenshot shows a terminal window titled "root@LinuxHost4: ~". The window has two tabs: "OpenvSwitch-1" and "root@LinuxHost4: ~". The "root@LinuxHost4: ~" tab is active and displays the following output:

```
64 bytes from 192.168.1.4: icmp_seq=13 ttl=64 time=3.04 ms
64 bytes from 192.168.1.4: icmp_seq=14 ttl=64 time=3.15 ms
64 bytes from 192.168.1.4: icmp_seq=15 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=16 ttl=64 time=2.86 ms
64 bytes from 192.168.1.4: icmp_seq=17 ttl=64 time=5.05 ms
64 bytes from 192.168.1.4: icmp_seq=18 ttl=64 time=5.33 ms
64 bytes from 192.168.1.4: icmp_seq=19 ttl=64 time=2.76 ms
64 bytes from 192.168.1.4: icmp_seq=20 ttl=64 time=5.44 ms
64 bytes from 192.168.1.4: icmp_seq=21 ttl=64 time=2.96 ms
64 bytes from 192.168.1.4: icmp_seq=22 ttl=64 time=2.87 ms
64 bytes from 192.168.1.4: icmp_seq=23 ttl=64 time=3.66 ms
64 bytes from 192.168.1.4: icmp_seq=24 ttl=64 time=3.07 ms
64 bytes from 192.168.1.4: icmp_seq=25 ttl=64 time=3.05 ms
64 bytes from 192.168.1.4: icmp_seq=26 ttl=64 time=3.36 ms
64 bytes from 192.168.1.4: icmp_seq=27 ttl=64 time=3.09 ms
64 bytes from 192.168.1.4: icmp_seq=28 ttl=64 time=3.40 ms
64 bytes from 192.168.1.4: icmp_seq=29 ttl=64 time=3.03 ms
64 bytes from 192.168.1.4: icmp_seq=30 ttl=64 time=3.03 ms
64 bytes from 192.168.1.4: icmp_seq=31 ttl=64 time=3.48 ms
^C
--- 192.168.1.4 ping statistics ---
31 packets transmitted, 31 received, 0% packet loss, time 30049ms
rtt min/avg/max/mdev = 2.693/3.411/5.600/0.790 ms
```

Based on the screenshot, LinuxHost4 pinged to LinuxHost3 and when it did it was able to scan all packets that were going on in LinuxHost3 because they are both part of the same VLAN(VLAN20) based on tag=20, LinuxHost4 can check the network activity of LinuxHost3, it sends a message to that specific address in the ping and after the ping ends it shows the ping stats of the ip address of the device it pinged and it shows that all the packets have been successfully received and there has been no packet loss. Since they were on the same VLAN no packets were lost. They are on the same VLAN and they can communicate with each other, they are both physically and logically connecting.

4.Screenshots for part 4

```
root@LinuxHost1:~# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
      inet 192.168.1.2 netmask 255.255.255.0 broadcast 0.0.0.0  
      inet6 fe80::42:18ff:fedc:e100 prefixlen 64 scopeid 0x20<link>  
        ether 02:42:18:dc:e1:00 txqueuelen 1000 (Ethernet)  
          RX packets 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 8 bytes 656 (656.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
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 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 0 bytes 0 (0.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@LinuxHost1:~# ping 192.168.1.4  
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.  
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=2 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=3 Destination Host Unreachable
```

```
root@LinuxHost1:~# ifconfig  
inet6 ::1 prefixlen 128 scopeid 0x10<host>  
loop txqueuelen 1000 (Local Loopback)  
RX packets 0 bytes 0 (0.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 0 bytes 0 (0.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@LinuxHost1:~# ping 192.168.1.4  
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.  
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=2 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=3 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=4 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=5 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=6 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=8 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=9 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=10 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=11 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=12 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=13 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=14 Destination Host Unreachable  
From 192.168.1.2 icmp_seq=15 Destination Host Unreachable
```

```
root@LinuxHost1:~# ping 192.168.1.4
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable
From 192.168.1.2 icmp_seq=2 Destination Host Unreachable
From 192.168.1.2 icmp_seq=3 Destination Host Unreachable
From 192.168.1.2 icmp_seq=4 Destination Host Unreachable
From 192.168.1.2 icmp_seq=5 Destination Host Unreachable
From 192.168.1.2 icmp_seq=6 Destination Host Unreachable
From 192.168.1.2 icmp_seq=8 Destination Host Unreachable
From 192.168.1.2 icmp_seq=9 Destination Host Unreachable
From 192.168.1.2 icmp_seq=10 Destination Host Unreachable
From 192.168.1.2 icmp_seq=11 Destination Host Unreachable
From 192.168.1.2 icmp_seq=12 Destination Host Unreachable
From 192.168.1.2 icmp_seq=13 Destination Host Unreachable
From 192.168.1.2 icmp_seq=14 Destination Host Unreachable
From 192.168.1.2 icmp_seq=15 Destination Host Unreachable
From 192.168.1.2 icmp_seq=16 Destination Host Unreachable
From 192.168.1.2 icmp_seq=17 Destination Host Unreachable
From 192.168.1.2 icmp_seq=18 Destination Host Unreachable
From 192.168.1.2 icmp_seq=19 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.1.2 icmp_seq=20 Destination Host Unreachable
From 192.168.1.2 icmp_seq=21 Destination Host Unreachable
```

```
From 192.168.1.2 icmp_seq=14 Destination Host Unreachable
From 192.168.1.2 icmp_seq=15 Destination Host Unreachable
From 192.168.1.2 icmp_seq=16 Destination Host Unreachable
From 192.168.1.2 icmp_seq=17 Destination Host Unreachable
From 192.168.1.2 icmp_seq=18 Destination Host Unreachable
From 192.168.1.2 icmp_seq=19 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.1.2 icmp_seq=20 Destination Host Unreachable
From 192.168.1.2 icmp_seq=21 Destination Host Unreachable
From 192.168.1.2 icmp_seq=23 Destination Host Unreachable
From 192.168.1.2 icmp_seq=24 Destination Host Unreachable
From 192.168.1.2 icmp_seq=25 Destination Host Unreachable
From 192.168.1.2 icmp_seq=26 Destination Host Unreachable
From 192.168.1.2 icmp_seq=27 Destination Host Unreachable
From 192.168.1.2 icmp_seq=28 Destination Host Unreachable
From 192.168.1.2 icmp_seq=29 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.1.2 icmp_seq=30 Destination Host Unreachable
From 192.168.1.2 icmp_seq=31 Destination Host Unreachable
^C
--- 192.168.1.4 ping statistics ---
34 packets transmitted, 0 received, +29 errors, 100% packet loss, time 33786ms
pipe 4
root@LinuxHost1:~#
```

Based on the screenshots when LinuxHost1 pinged to LinuxHost3, it wasn't able to scan and track the network happening in LinuxHost3, because the destination host was unreachable due to both LinuxHost1 which was VLAN 10 and LinuxHost3 which was VLAN 20 being on different VLANS and since they were on different VLANS the packets were lost everything, and it wasn't able to reach the destination since they were both on two different VLANs they are unable to check their data. Since the packets and traffic from LinuxHost3 were unable to reach LinuxHost1 . They are unable to communicate with each other, they may be physically connected but they are not due to different vlans due to VLAN trunking