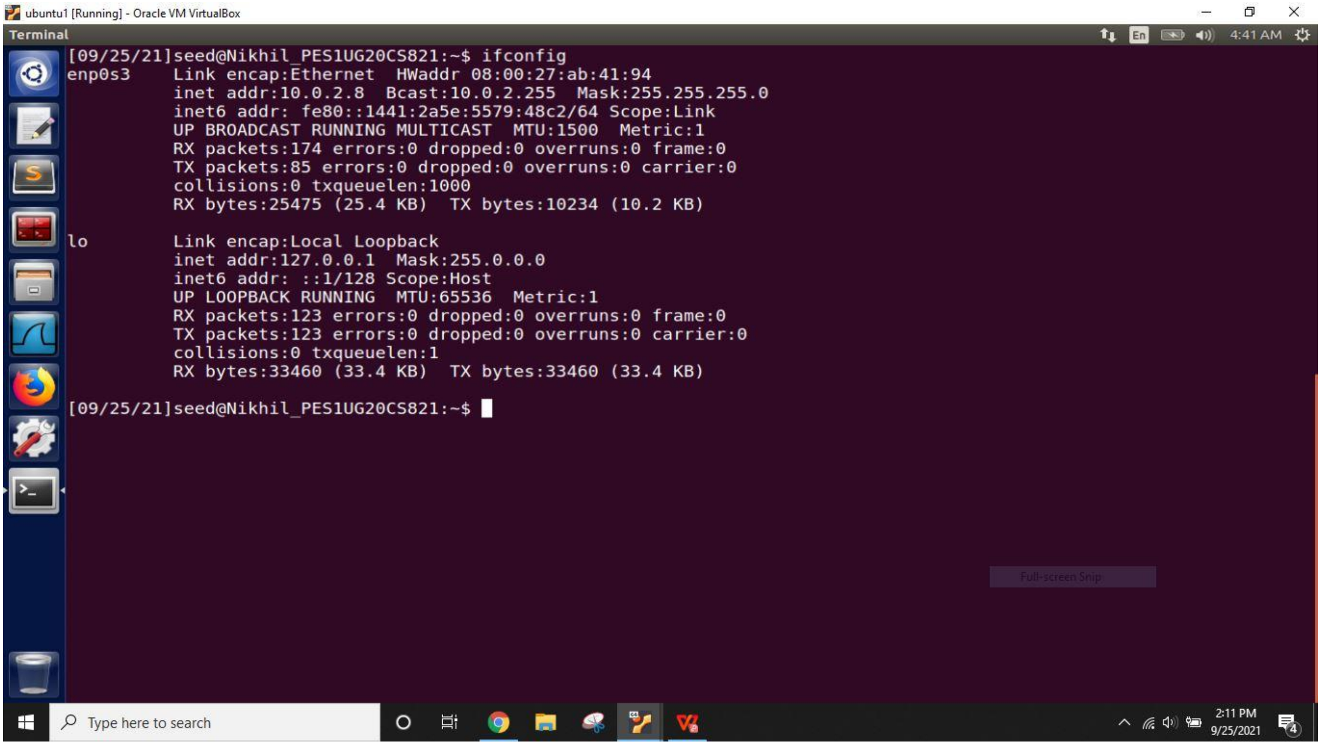
Assignment-2
Sniff and Spoof Packets using pcap (C programs)

Name: Nikhil T M
SRN: PES1UG20CS821
Subject:Computer Network Security
Code:UE19CS326

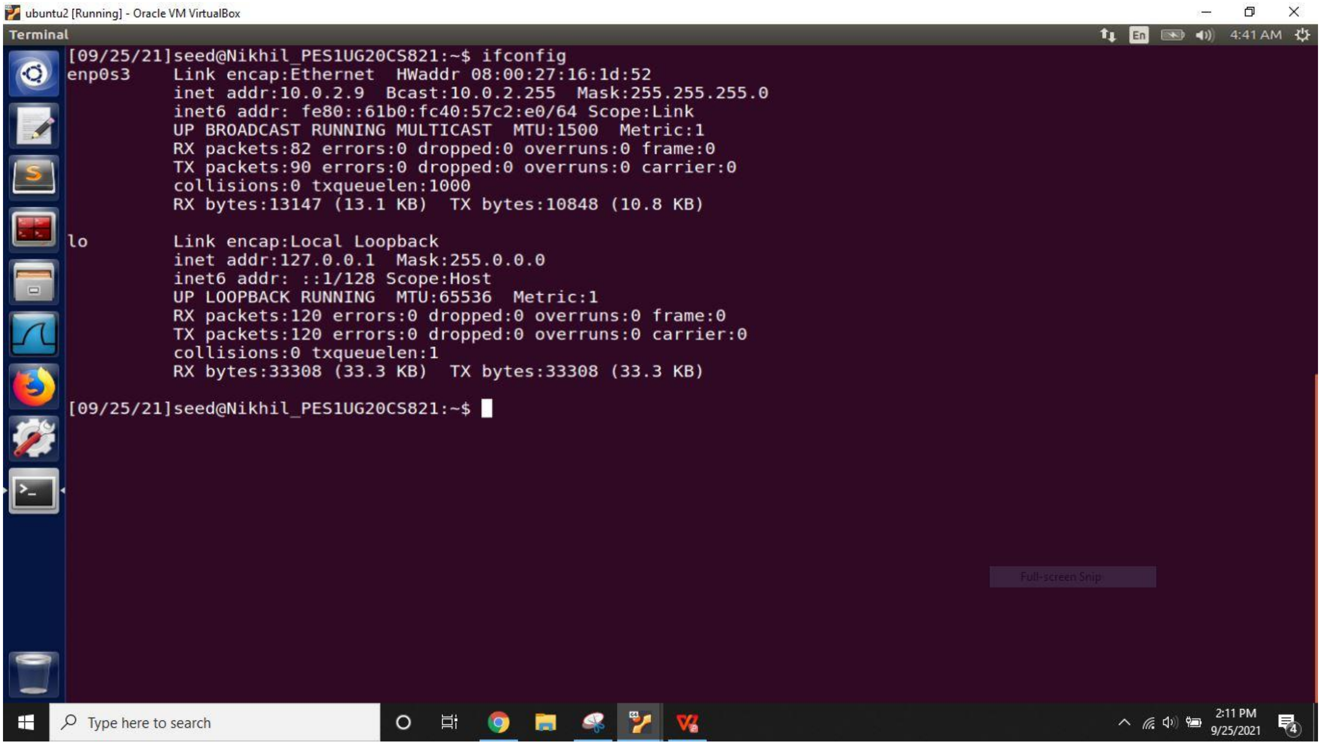
Task 1: Sniffing - Writing Packet Sniffing Program

Attacker machine: 10.0.2.8



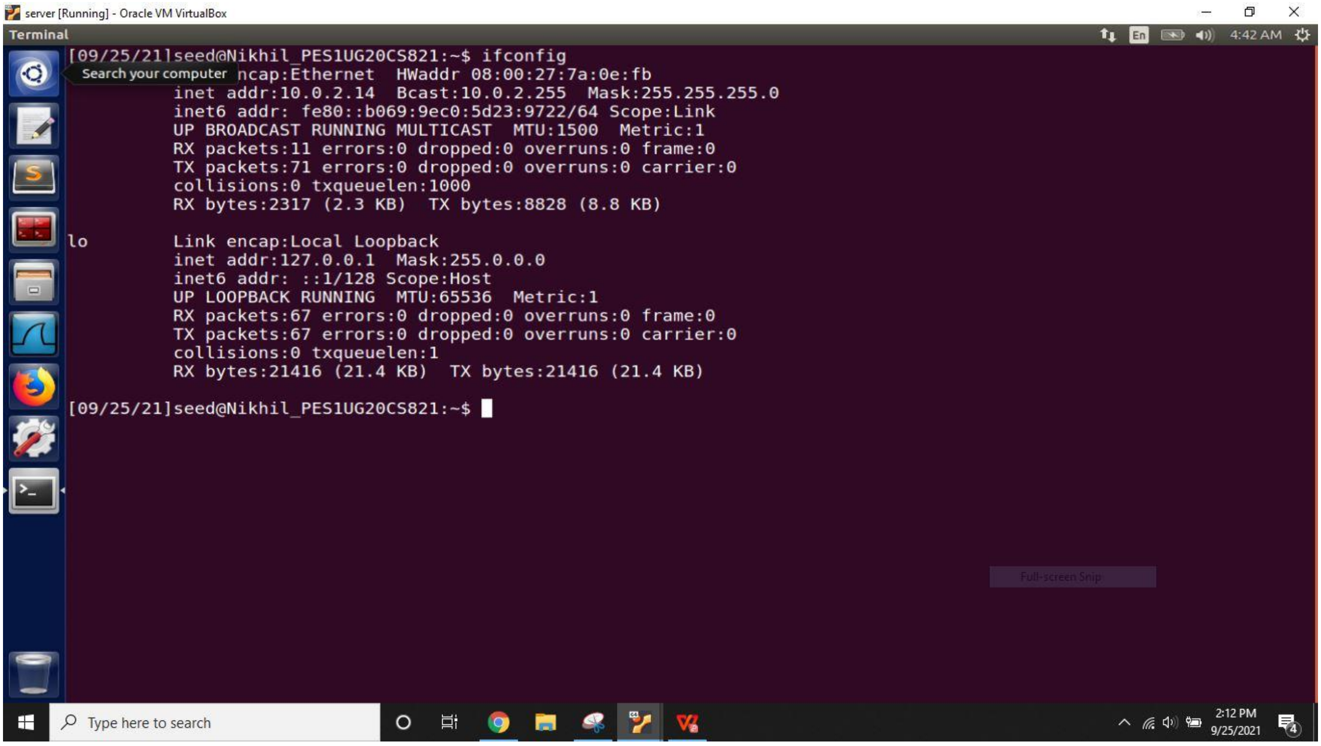
```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:ab:41:94
          inet addr:10.0.2.8  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::1441:2a5e:5579:48c2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:174 errors:0 dropped:0 overruns:0 frame:0
          TX packets:85 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:25475 (25.4 KB)  TX bytes:10234 (10.2 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:123 errors:0 dropped:0 overruns:0 frame:0
          TX packets:123 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:33460 (33.4 KB)  TX bytes:33460 (33.4 KB)

[09/25/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Victim machine: 10.0.2.9



```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:16:1d:52
          inet addr:10.0.2.9  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::61b0:fc40:57c2:e0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:82 errors:0 dropped:0 overruns:0 frame:0
          TX packets:90 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13147 (13.1 KB)  TX bytes:10848 (10.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:120 errors:0 dropped:0 overruns:0 frame:0
          TX packets:120 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:33308 (33.3 KB)  TX bytes:33308 (33.3 KB)

[09/25/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Server machine: 10.0.2.14

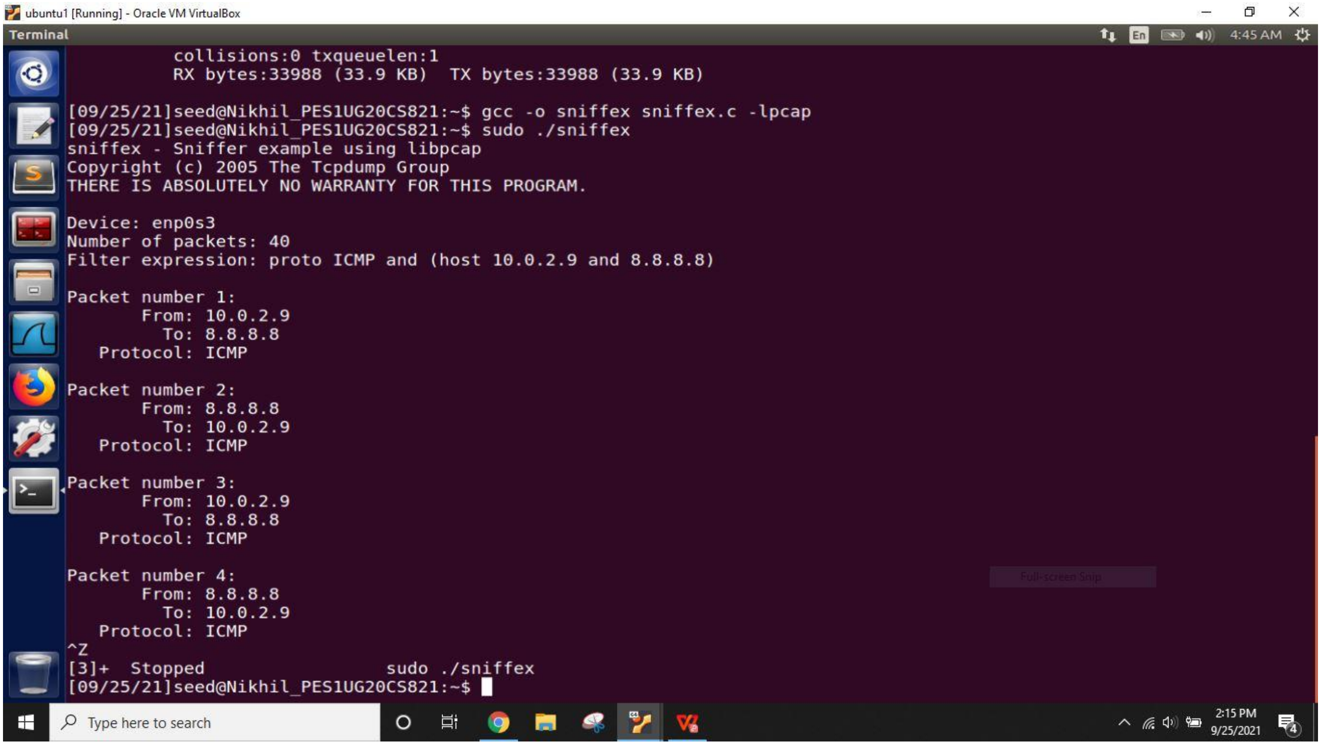


```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ifconfig
                 ncap:Ethernet  HWaddr 08:00:27:7a:0e:fb
          inet addr:10.0.2.14  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::b069:9ec0:5d23:9722/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:11 errors:0 dropped:0 overruns:0 frame:0
          TX packets:71 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2317 (2.3 KB)  TX bytes:8828 (8.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:67 errors:0 dropped:0 overruns:0 frame:0
          TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:21416 (21.4 KB)  TX bytes:21416 (21.4 KB)

[09/25/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Task 1.1: Understanding how a Sniffer Works

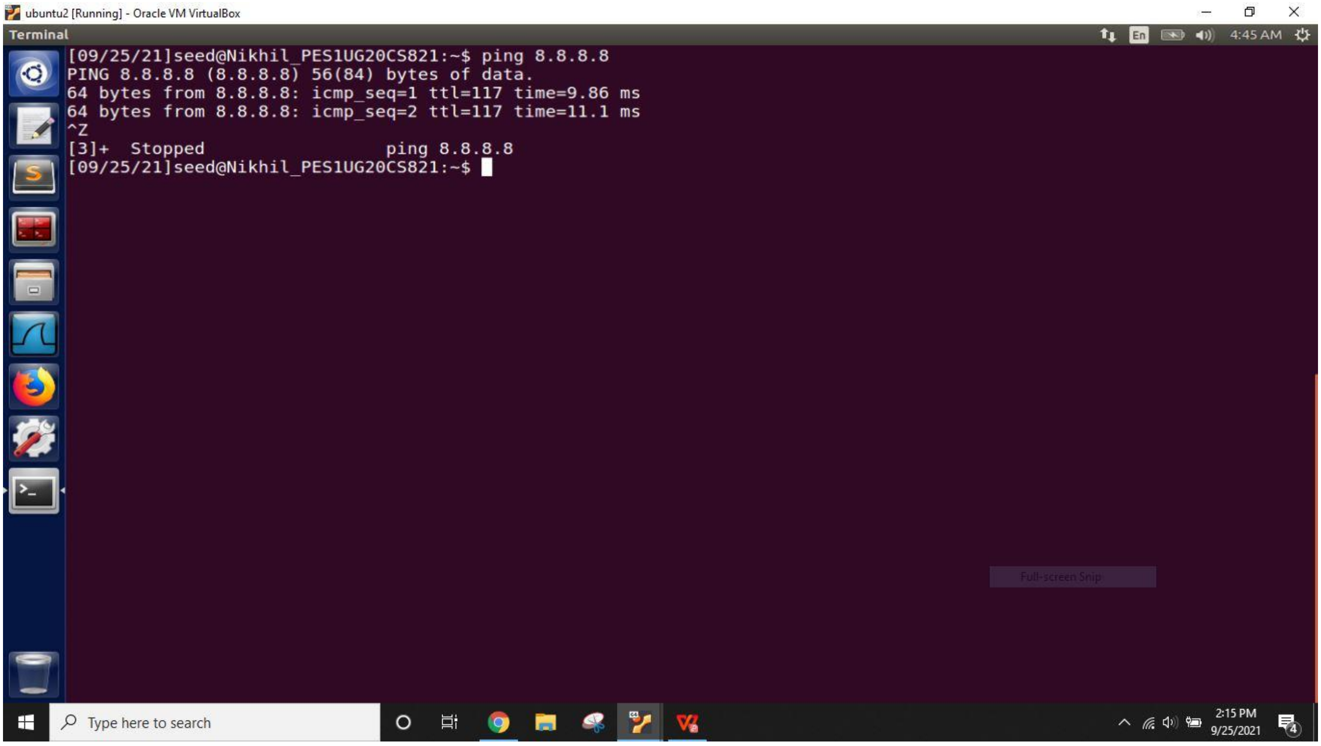
When promiscuous mode is ON

Observation on attacker



```
          collisions:0 txqueuelen:1
          RX bytes:33988 (33.9 KB)  TX bytes:33988 (33.9 KB)

[09/25/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o sniffex sniffex.c -lpcap
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./sniffex
sniffex - Sniffer example using libpcap
Copyright (c) 2005 The Tcpdump Group
THERE IS ABSOLUTELY NO WARRANTY FOR THIS PROGRAM.

Device: enp0s3
Number of packets: 40
Filter expression: proto ICMP and (host 10.0.2.9 and 8.8.8.8)

Packet number 1:
       From: 10.0.2.9
         To: 8.8.8.8
   Protocol: ICMP

Packet number 2:
       From: 8.8.8.8
         To: 10.0.2.9
   Protocol: ICMP

Packet number 3:
       From: 10.0.2.9
         To: 8.8.8.8
   Protocol: ICMP

Packet number 4:
       From: 8.8.8.8
         To: 10.0.2.9
   Protocol: ICMP
^Z
[3]+  Stopped                 sudo ./sniffex
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Observation on victim machine

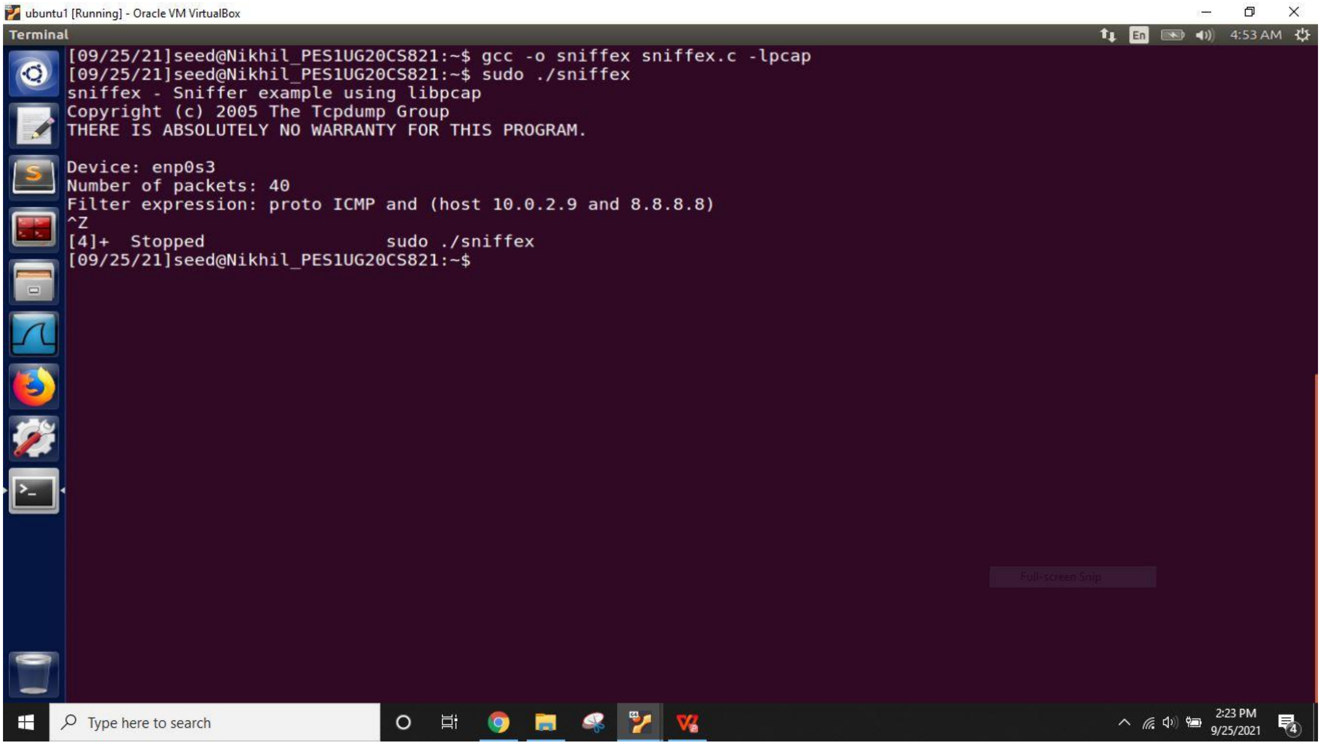


```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=9.86 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=11.1 ms
^Z
[3]+  Stopped                 ping 8.8.8.8
[09/25/21]seed@Nikhil_PES1UG20CS821:~$
```

Promiscuous Mode is a network card background that does not filter incoming packets by MAC.The Promiscuous Mode is turned ON in this method so the packets are not filtered so all the packets can be sniffed

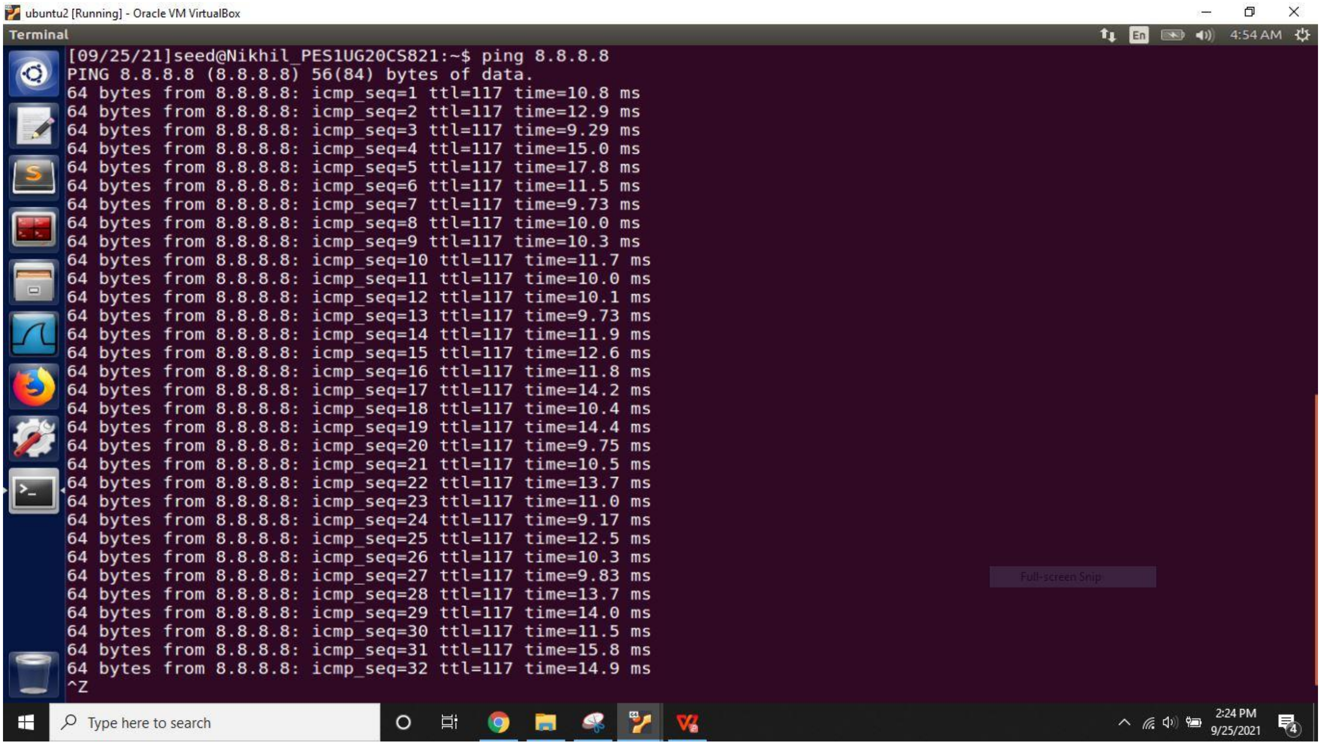
When promiscuous mode is OFF

Observation on attacker machine



```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o sniffex sniffex.c -lpcap
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./sniffex
sniffex - Sniffer example using libpcap
Copyright (c) 2005 The Tcpdump Group
THERE IS ABSOLUTELY NO WARRANTY FOR THIS PROGRAM.

Device: enp0s3
Number of packets: 40
Filter expression: proto ICMP and (host 10.0.2.9 and 8.8.8.8)
^Z
[4]+  Stopped                 sudo ./sniffex
[09/25/21]seed@Nikhil_PES1UG20CS821:~$
```

Observation on victim machine

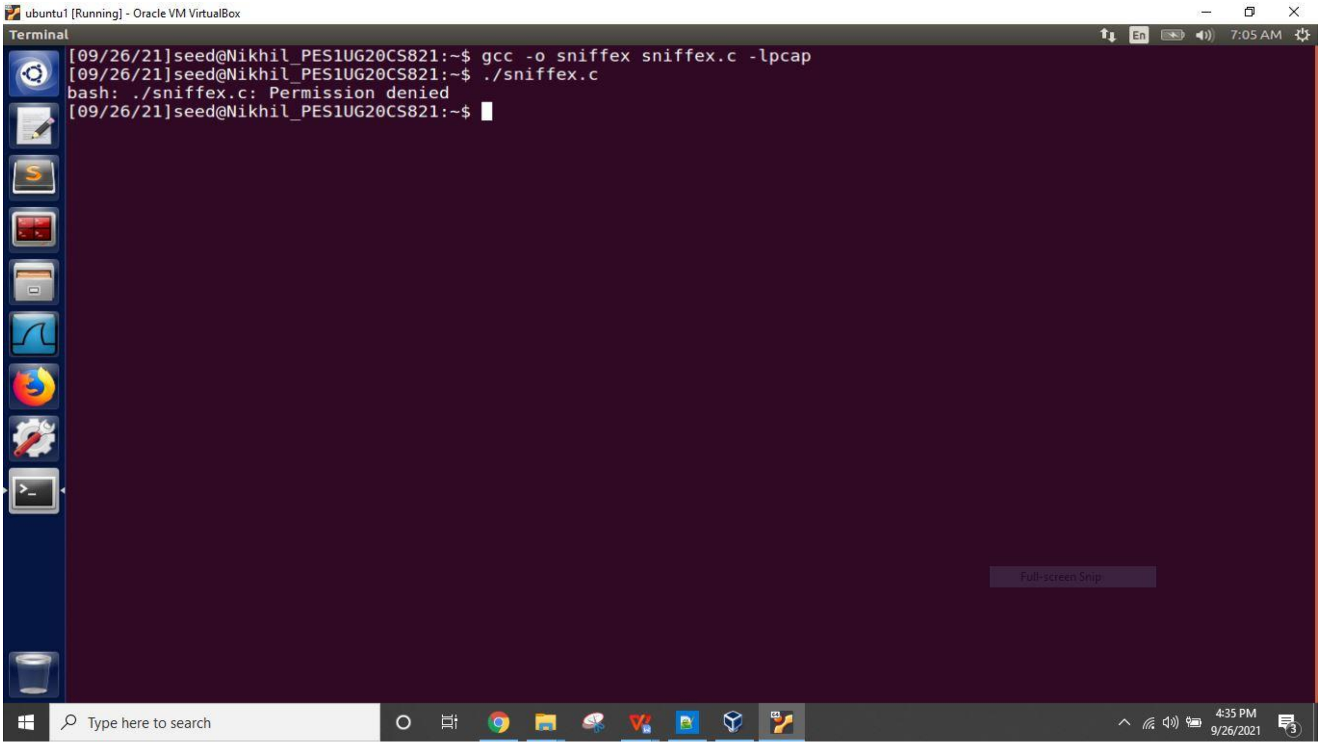


```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=10.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=12.9 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=117 time=9.29 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=117 time=15.0 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=117 time=17.8 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=117 time=11.5 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=117 time=9.73 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=117 time=10.0 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=117 time=10.3 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=117 time=11.7 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=117 time=10.0 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=117 time=10.1 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=117 time=9.73 ms
64 bytes from 8.8.8.8: icmp_seq=14 ttl=117 time=11.9 ms
64 bytes from 8.8.8.8: icmp_seq=15 ttl=117 time=12.6 ms
64 bytes from 8.8.8.8: icmp_seq=16 ttl=117 time=11.8 ms
64 bytes from 8.8.8.8: icmp_seq=17 ttl=117 time=14.2 ms
64 bytes from 8.8.8.8: icmp_seq=18 ttl=117 time=10.4 ms
64 bytes from 8.8.8.8: icmp_seq=19 ttl=117 time=14.4 ms
64 bytes from 8.8.8.8: icmp_seq=20 ttl=117 time=9.75 ms
64 bytes from 8.8.8.8: icmp_seq=21 ttl=117 time=10.5 ms
64 bytes from 8.8.8.8: icmp_seq=22 ttl=117 time=13.7 ms
64 bytes from 8.8.8.8: icmp_seq=23 ttl=117 time=11.0 ms
64 bytes from 8.8.8.8: icmp_seq=24 ttl=117 time=9.17 ms
64 bytes from 8.8.8.8: icmp_seq=25 ttl=117 time=12.5 ms
64 bytes from 8.8.8.8: icmp_seq=26 ttl=117 time=10.3 ms
64 bytes from 8.8.8.8: icmp_seq=27 ttl=117 time=9.83 ms
64 bytes from 8.8.8.8: icmp_seq=28 ttl=117 time=13.7 ms
64 bytes from 8.8.8.8: icmp_seq=29 ttl=117 time=14.0 ms
64 bytes from 8.8.8.8: icmp_seq=30 ttl=117 time=11.5 ms
64 bytes from 8.8.8.8: icmp_seq=31 ttl=117 time=15.8 ms
64 bytes from 8.8.8.8: icmp_seq=32 ttl=117 time=14.9 ms
^Z
```

The Promiscuous Mode is turned OFF in this method so the packets are filtered so all the packets cannot be sniffed so nothing can be viewed in the attacker machine even though the victim pings successfully.

i. The library packages used in the above program which is required for sniffing are package called pcap.Packet Capture or PCAP (also known as libpcap) is an application programming interface (API) that captures live network packet data from OSI model Layers 2-7. Network analyzers like Wireshark create .pcap files to collect and record packet data from a network. These PCAP files can be used to view TCP/IP and UDP network packets.these are used in the c program to perform sniffing

ii. We need sudo command to run the above program because promiscuous mode cannot be achieved with normal privileges if we run the above program without sudo command then the error message is displayed as “permission denied” which as shown below



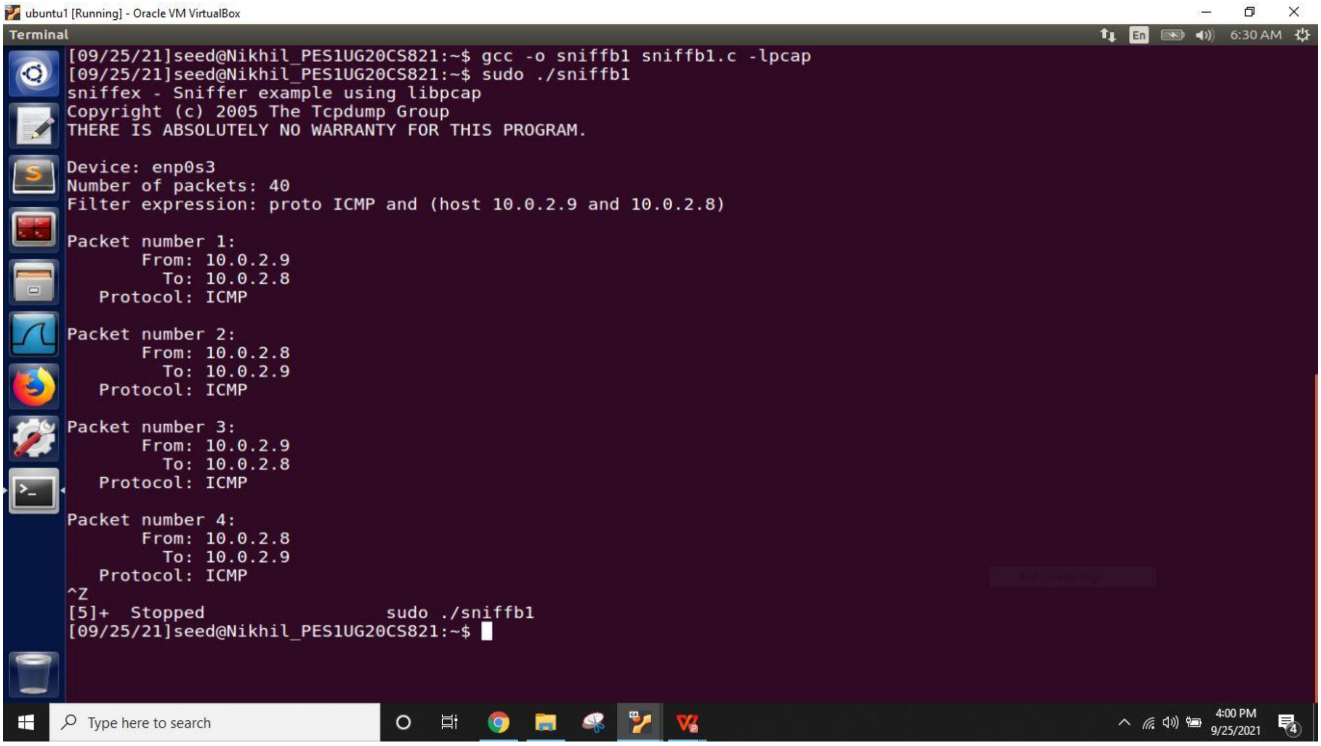
```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o sniffex sniffex.c -lpcap
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ ./sniffex.c
bash: ./sniffex.c: Permission denied
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
```

iii. We can turn on and turn off the promiscuous mode from the c program.The argument 1 in the below code represents the promiscuous mode is ON it can turned OFF by changing it to 0

“handle = pcap_open_live(dev, SNAP_LEN, 1, 1000, errbuf);

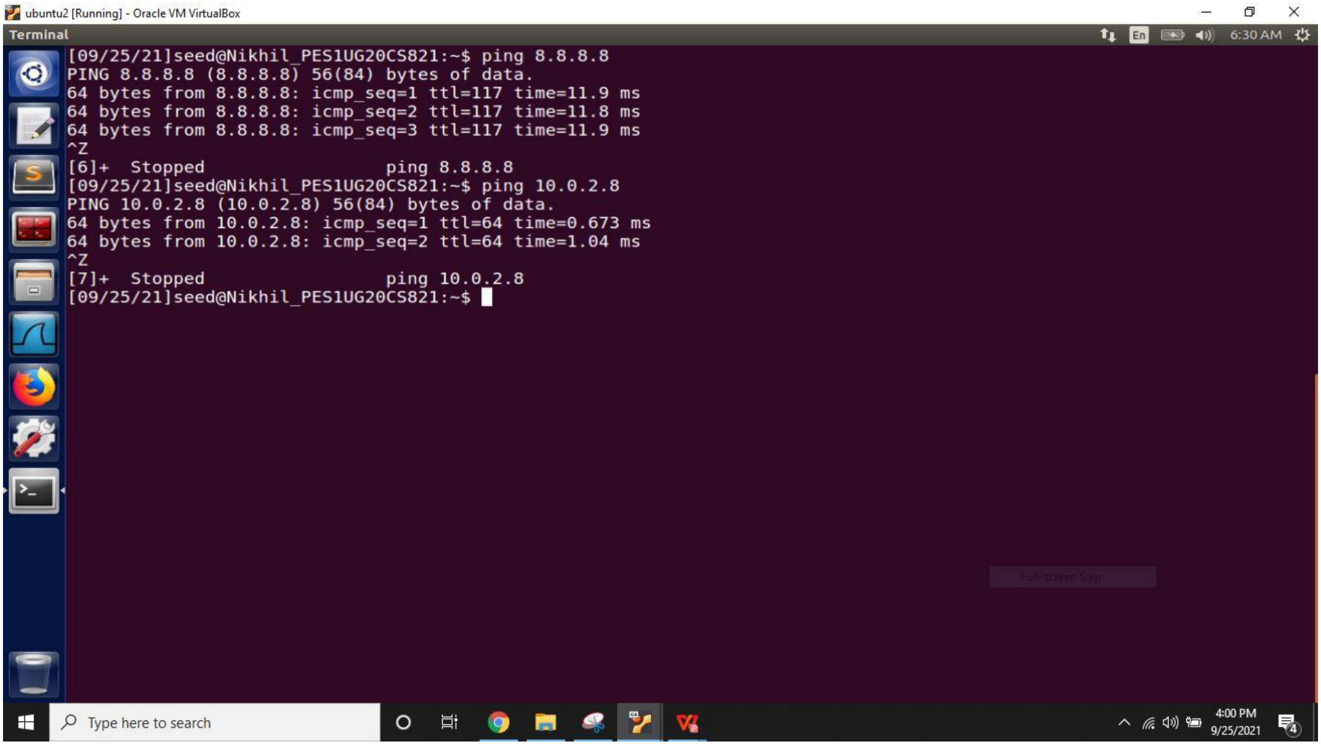
Task 1.2: Writing Filters

Observation on the attacker machine



```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o sniffb1 sniffb1.c -lpcap
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./sniffb1
sniffex - Sniffer example using libpcap
Copyright (c) 2005 The Tcpdump Group
THERE IS ABSOLUTELY NO WARRANTY FOR THIS PROGRAM.

Device: enp0s3
Number of packets: 40
Filter expression: proto ICMP and (host 10.0.2.9 and 10.0.2.8)

Packet number 1:
       From: 10.0.2.9
         To: 10.0.2.8
   Protocol: ICMP

Packet number 2:
       From: 10.0.2.8
         To: 10.0.2.9
   Protocol: ICMP

Packet number 3:
       From: 10.0.2.9
         To: 10.0.2.8
   Protocol: ICMP

Packet number 4:
       From: 10.0.2.8
         To: 10.0.2.9
   Protocol: ICMP
^Z
[5]+  Stopped                 sudo ./sniffb1
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ 
```

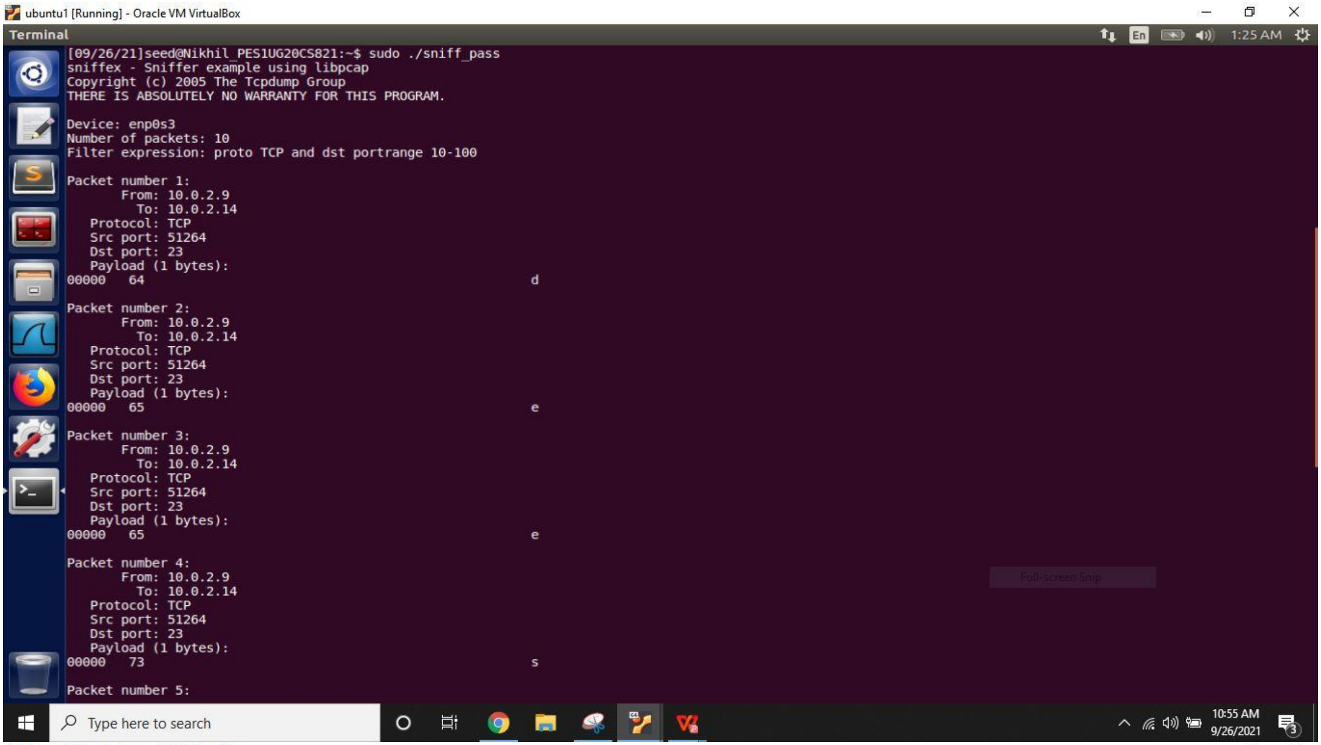
Observation on the victim machine



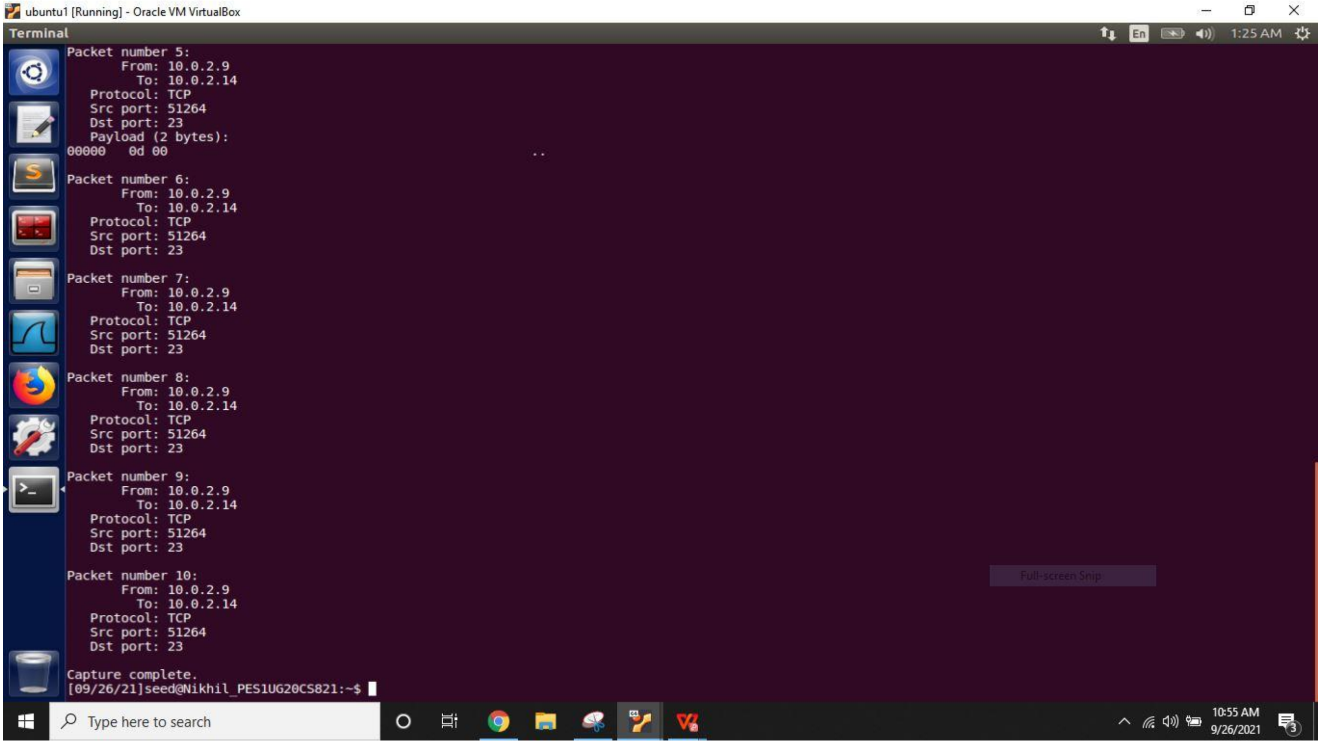
```
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=11.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=11.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=117 time=11.9 ms
^Z
[6]+  Stopped                 ping 8.8.8.8
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ ping 10.0.2.8
PING 10.0.2.8 (10.0.2.8) 56(84) bytes of data.
64 bytes from 10.0.2.8: icmp_seq=1 ttl=64 time=0.673 ms
64 bytes from 10.0.2.8: icmp_seq=2 ttl=64 time=1.04 ms
^Z
[7]+  Stopped                 ping 10.0.2.8
[09/25/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Task 1.3: Sniffing Passwords

Observation on the attacker machine

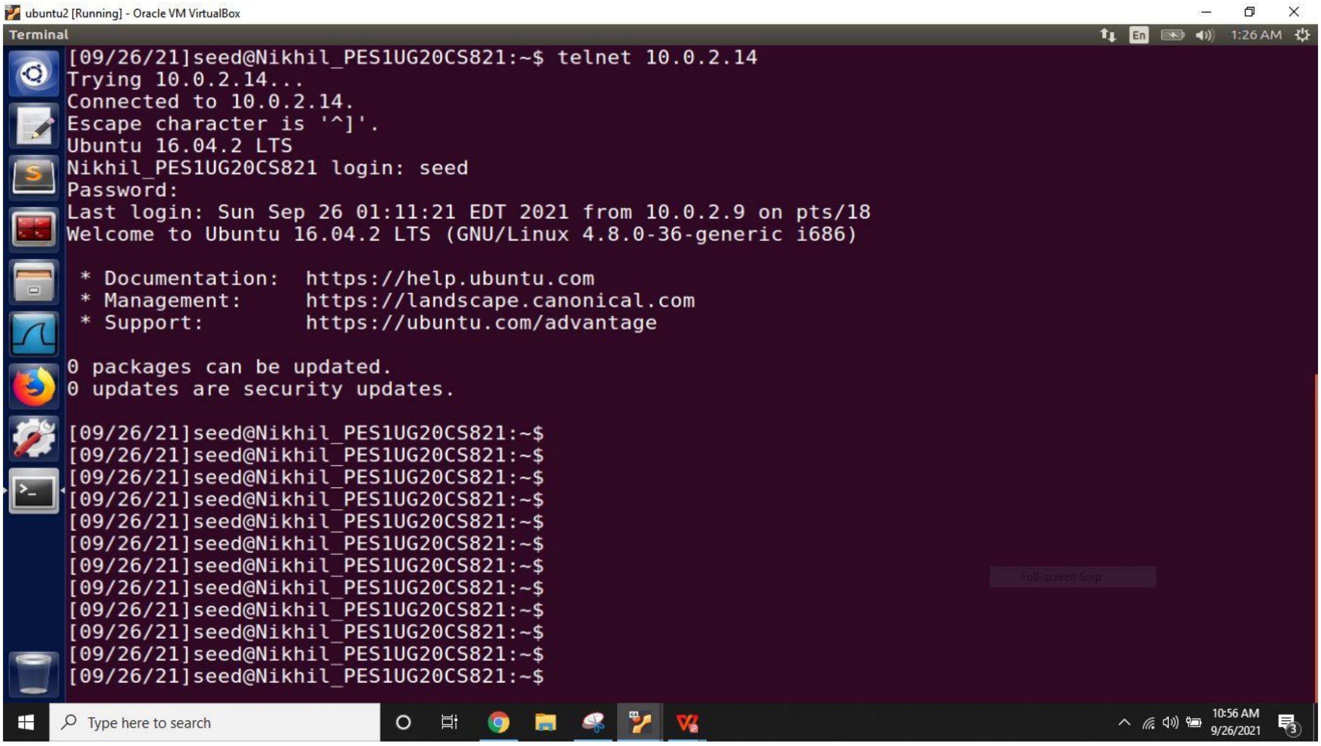


```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./sniff_pass
sniffex - Sniffer example using libpcap
Copyright (c) 2005 The Tcpdump Group
THERE IS ABSOLUTELY NO WARRANTY FOR THIS PROGRAM.

Device: enp0s3
Number of packets: 10
Filter expression: proto TCP and dst portrange 10-100

Packet number 1:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23
   Payload (1 bytes):
00000   64                                                d

Packet number 2:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23
   Payload (1 bytes):
00000   65                                                e

Packet number 3:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23
   Payload (1 bytes):
00000   65                                                e

Packet number 4:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23
   Payload (1 bytes):
00000   73                                                s

Packet number 5:
```



```
Packet number 5:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23
   Payload (2 bytes):
00000   0d 00                                             ..

Packet number 6:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23

Packet number 7:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23

Packet number 8:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23

Packet number 9:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23

Packet number 10:
       From: 10.0.2.9
         To: 10.0.2.14
   Protocol: TCP
   Src port: 51264
   Dst port: 23

Capture complete.
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Observation on the victim machine



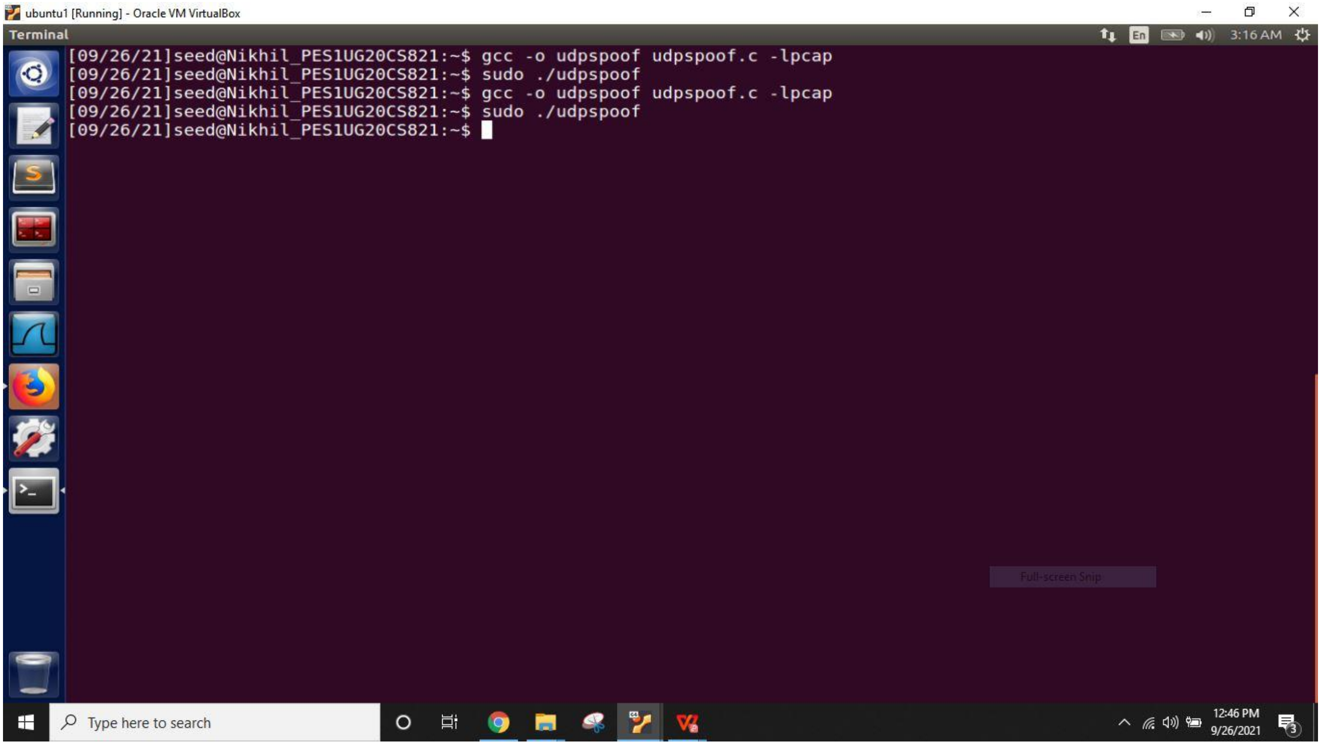
```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ telnet 10.0.2.14
Trying 10.0.2.14...
Connected to 10.0.2.14.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
Nikhil_PES1UG20CS821 login: seed
Password:
Last login: Sun Sep 26 01:11:21 EDT 2021 from 10.0.2.9 on pts/18
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
```

In this method the victim tries to connect to the server machine through telnet command during this time the attacker sniffs the password from the victim which is visible in the attacker machine the password is “dees” Which is successfully sniffed by the attacker.

Task 2: Spoofing

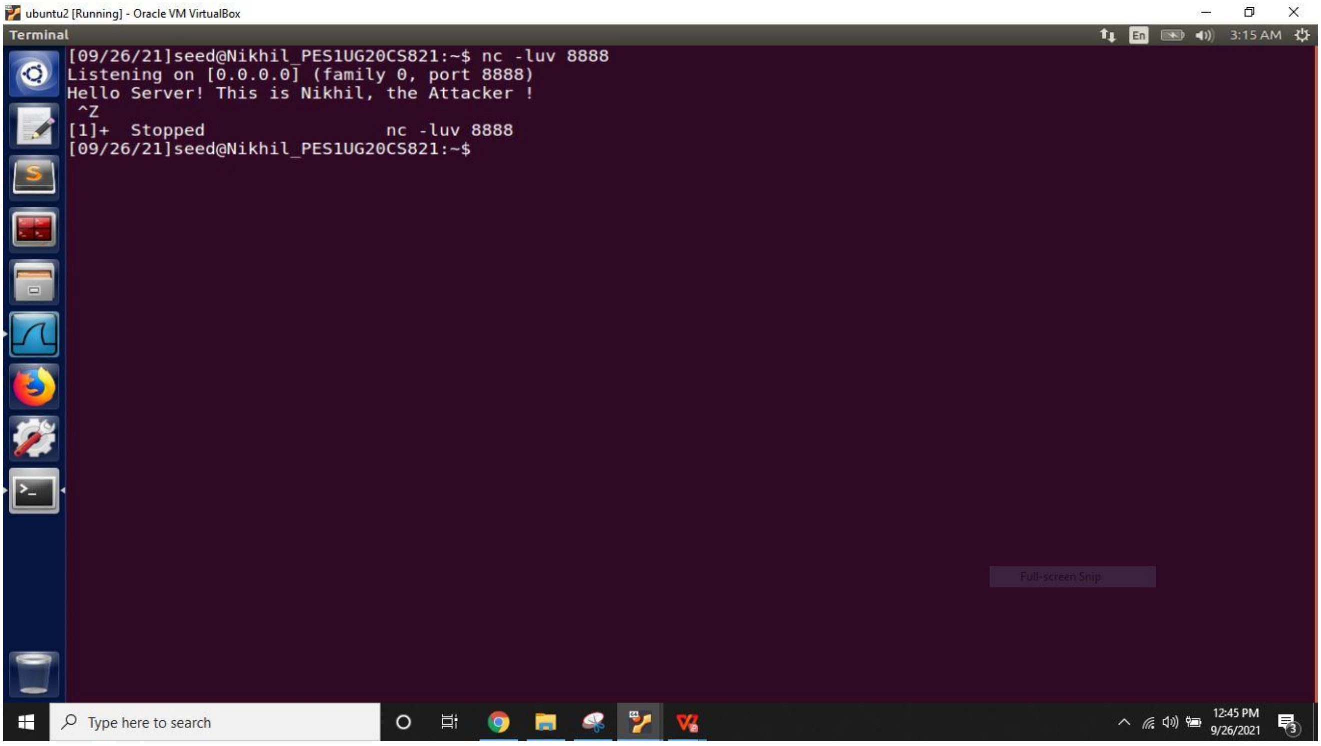
Task 2.1 - A Writing a spoofing program:

Observation on the attacker machine

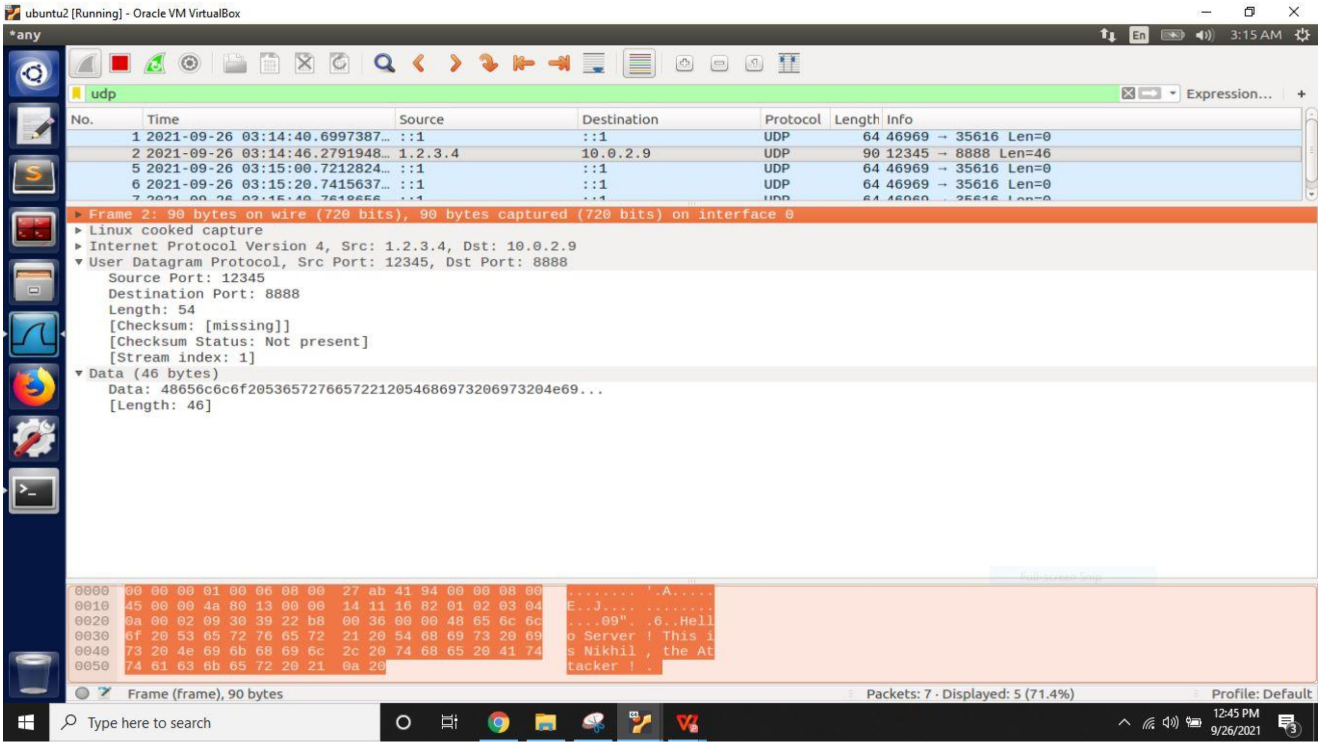


```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o udpspoof udpspoof.c -lpcap
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./udpspoof
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o udpspoof udpspoof.c -lpcap
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./udpspoof
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
```

Observation on the victim machine



```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ nc -luv 8888
Listening on [0.0.0.0] (family 0, port 8888)
Hello Server! This is Nikhil, the Attacker !
 ^Z
[1]+  Stopped                 nc -luv 8888
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
```

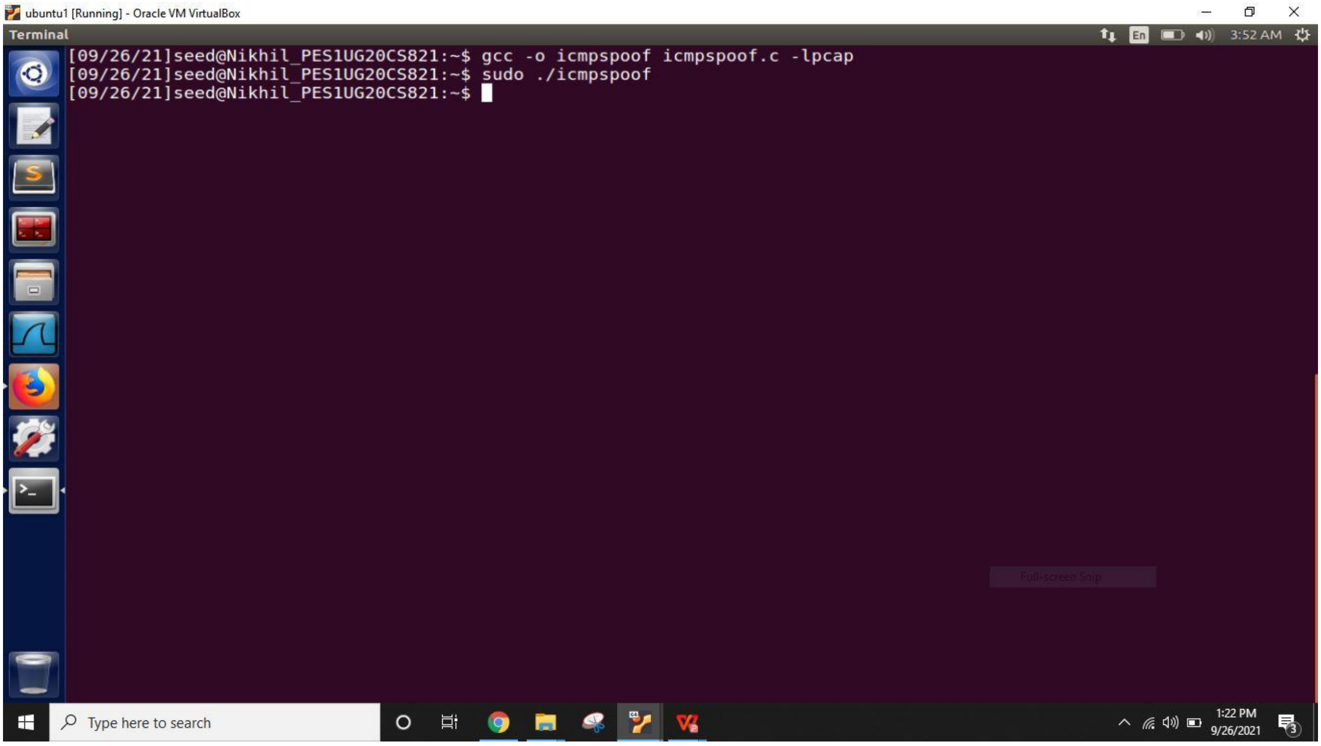


```
udp

No.  Time                           Source    Destination  Protocol  Length  Info
   1 2021-09-26 03:14:40.6997387…   ::1       ::1          UDP           64  46969 → 35616 Len=0
   2 2021-09-26 03:14:46.2791948…   1.2.3.4   10.0.2.9     UDP           90  12345 → 8888 Len=46
   5 2021-09-26 03:15:00.7212824…   ::1       ::1          UDP           64  46969 → 35616 Len=0
   6 2021-09-26 03:15:20.7415637…   ::1       ::1          UDP           64  46969 → 35616 Len=0

▸ Frame 2: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0
▸ Linux cooked capture
▸ Internet Protocol Version 4, Src: 1.2.3.4, Dst: 10.0.2.9
▾ User Datagram Protocol, Src Port: 12345, Dst Port: 8888
    Source Port: 12345
    Destination Port: 8888
    Length: 54
    [Checksum: [missing]]
    [Checksum Status: Not present]
    [Stream index: 1]
▾ Data (46 bytes)
    Data: 48656c6c6f20536572766572212054686973206973204e69...
    [Length: 46]

0000  00 00 00 01 00 06 08 00  27 ab 41 94 00 00 08 00   ........ '.A.....
0010  45 00 00 4a 80 13 00 00  14 11 16 82 01 02 03 04   E..J.... ........
0020  0a 00 02 09 30 39 22 b8  00 36 00 00 48 65 6c 6c   ....09". .6..Hell
0030  6f 20 53 65 72 76 65 72  21 20 54 68 69 73 20 69   o Server ! This i
0040  73 20 4e 69 6b 68 69 6c  2c 20 74 68 65 20 41 74   s Nikhil , the At
0050  74 61 63 6b 65 72 20 21  0a 20                     tacker ! .

Frame (frame), 90 bytes          Packets: 7 · Displayed: 5 (71.4%)          Profile: Default
```

In this task the attacker is going to perform UDP spoofing which is captured in the victim wireshark it shows the packets are from the IP address "1.2.3.4" which is not existed and the message from the attacker can be seen the victim machine when he listens to the port 8888 using the command nc(netcat).

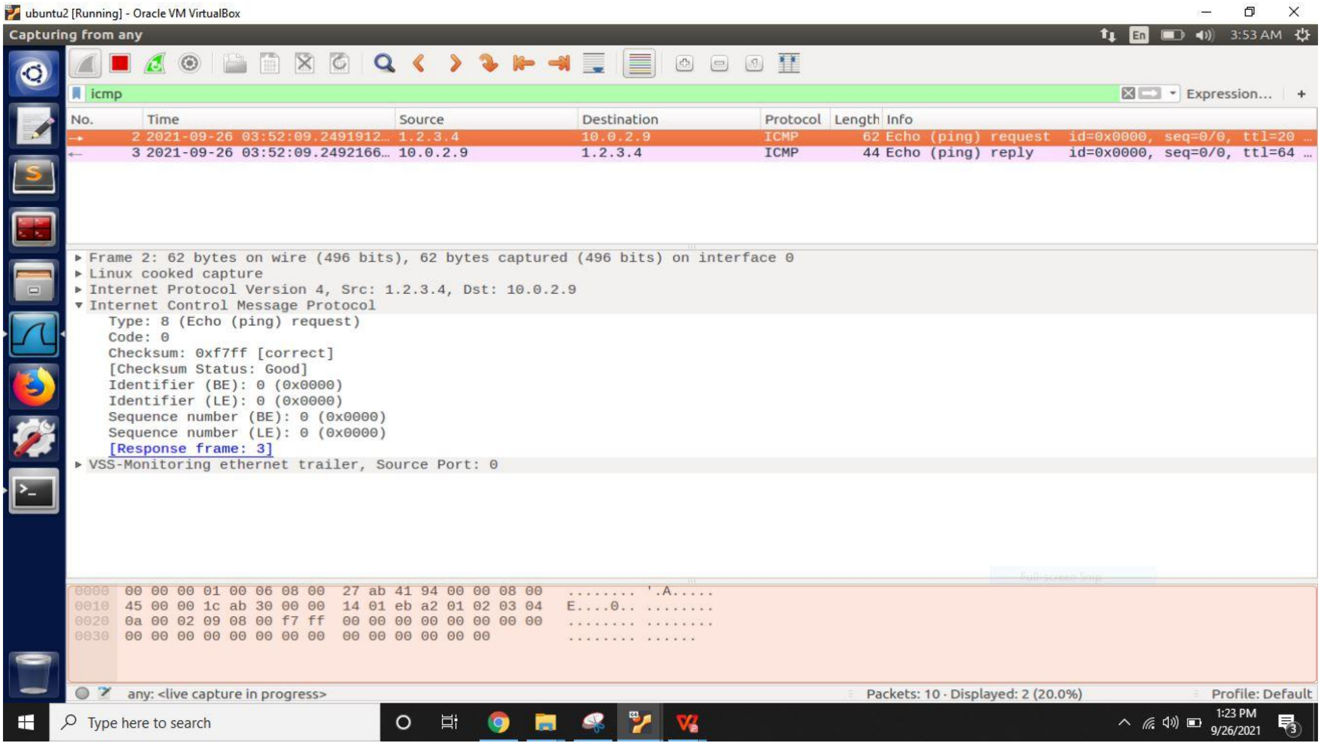
Task 2.2 – Spoof an ICMP Echo Request

Observation on the attacker machine



```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o icmpspoof icmpspoof.c -lpcap
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./icmpspoof
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Observation on the victim machine



```
No.  Time                          Source     Destination  Protocol  Length  Info
  2  2021-09-26 03:52:09.2491912…  1.2.3.4    10.0.2.9     ICMP          62  Echo (ping) request  id=0x0000, seq=0/0, ttl=20 …
  3  2021-09-26 03:52:09.2492166…  10.0.2.9   1.2.3.4      ICMP          44  Echo (ping) reply    id=0x0000, seq=0/0, ttl=64 …
```

```
▸ Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
▸ Linux cooked capture
▸ Internet Protocol Version 4, Src: 1.2.3.4, Dst: 10.0.2.9
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xf7ff [correct]
    [Checksum Status: Good]
    Identifier (BE): 0 (0x0000)
    Identifier (LE): 0 (0x0000)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
    [Response frame: 3]
▸ VSS-Monitoring ethernet trailer, Source Port: 0
```

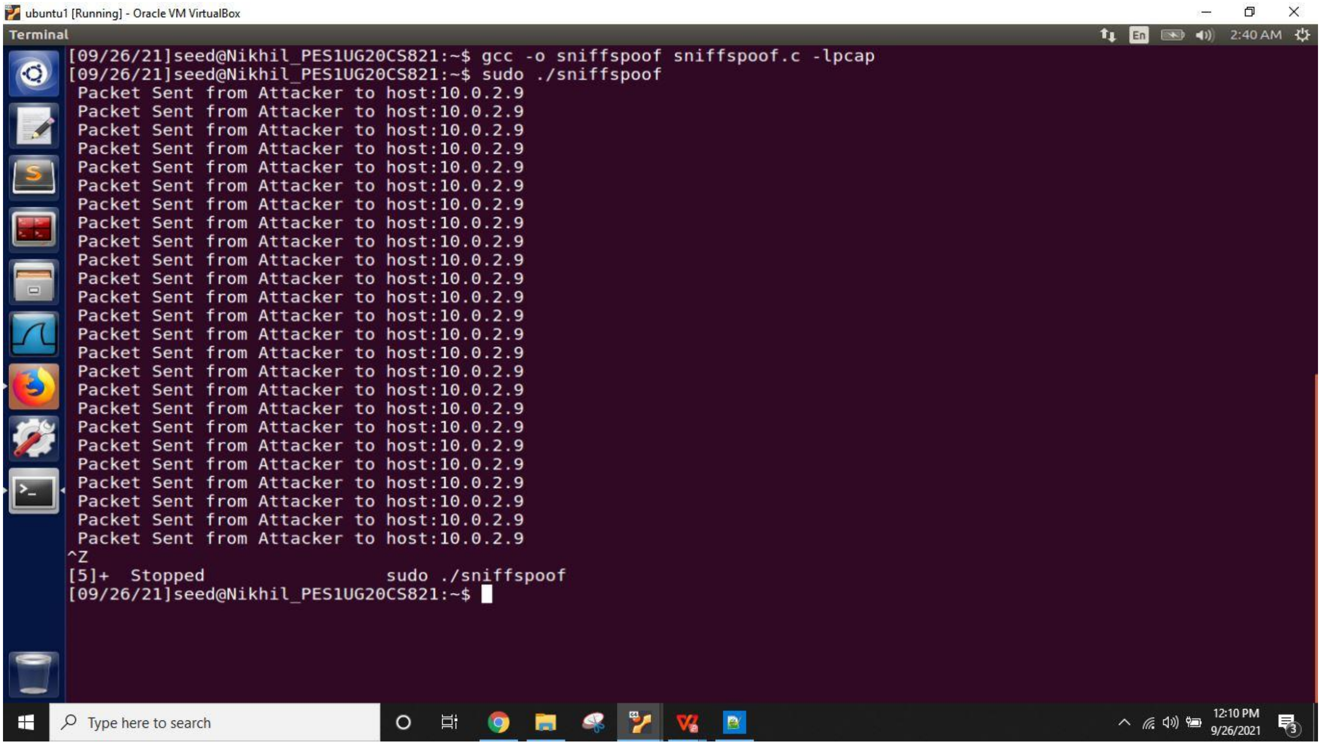
```
0000  00 00 00 01 00 06 08 00  27 ab 41 94 00 00 08 00   ........ '.A.....
0010  45 00 00 1c ab 30 00 00  14 01 eb a2 01 02 03 04   E....0.. ........
0020  0a 00 02 09 08 00 f7 ff  00 00 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00         ........ ......
```

```
any: <live capture in progress>     Packets: 10 · Displayed: 2 (20.0%)     Profile: Default
```

In this task the attacker is going to spoof an RCMP request which is replied by the victim machine. The attacker uses the non existing IP address 1.2.3.4 to perform ICMP request spoof but the victim machine replies to that request which can be seen the victim's wireshark.

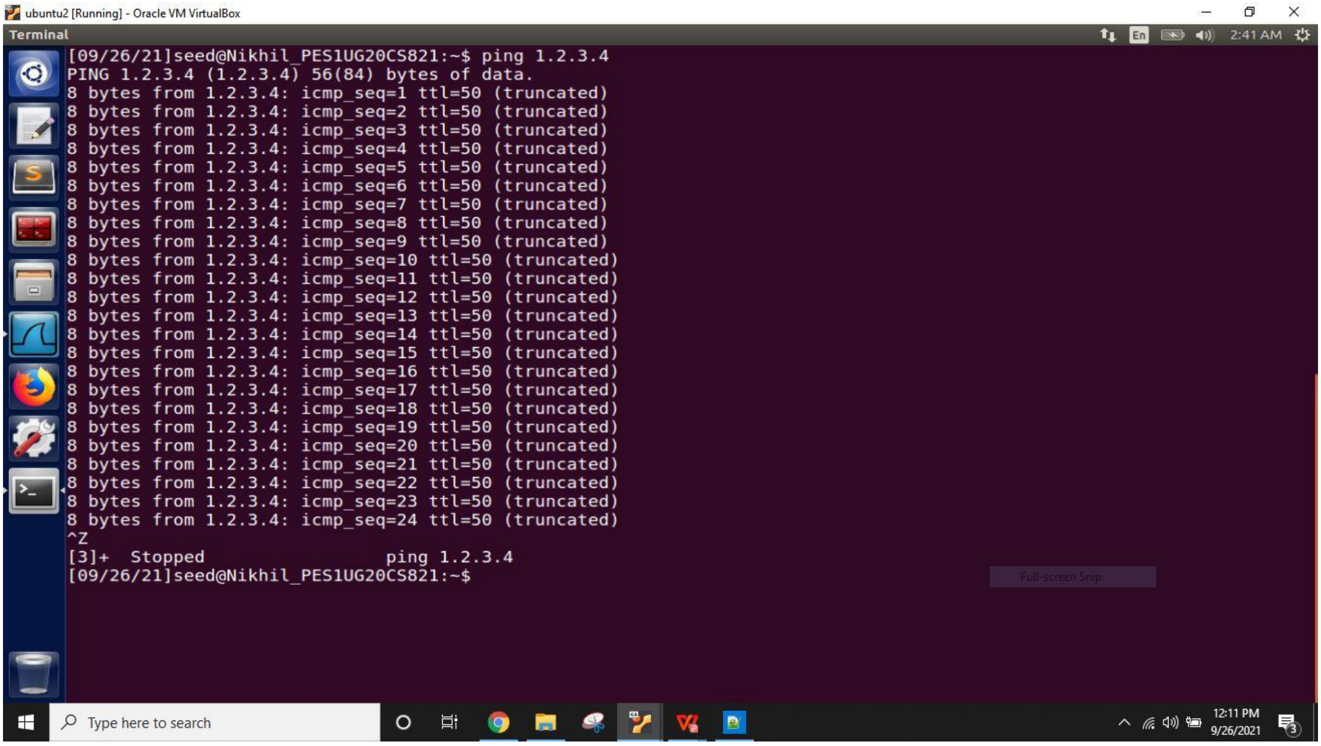
Task 2.3 – Sniff and then Spoof

Observation on the attacker machine

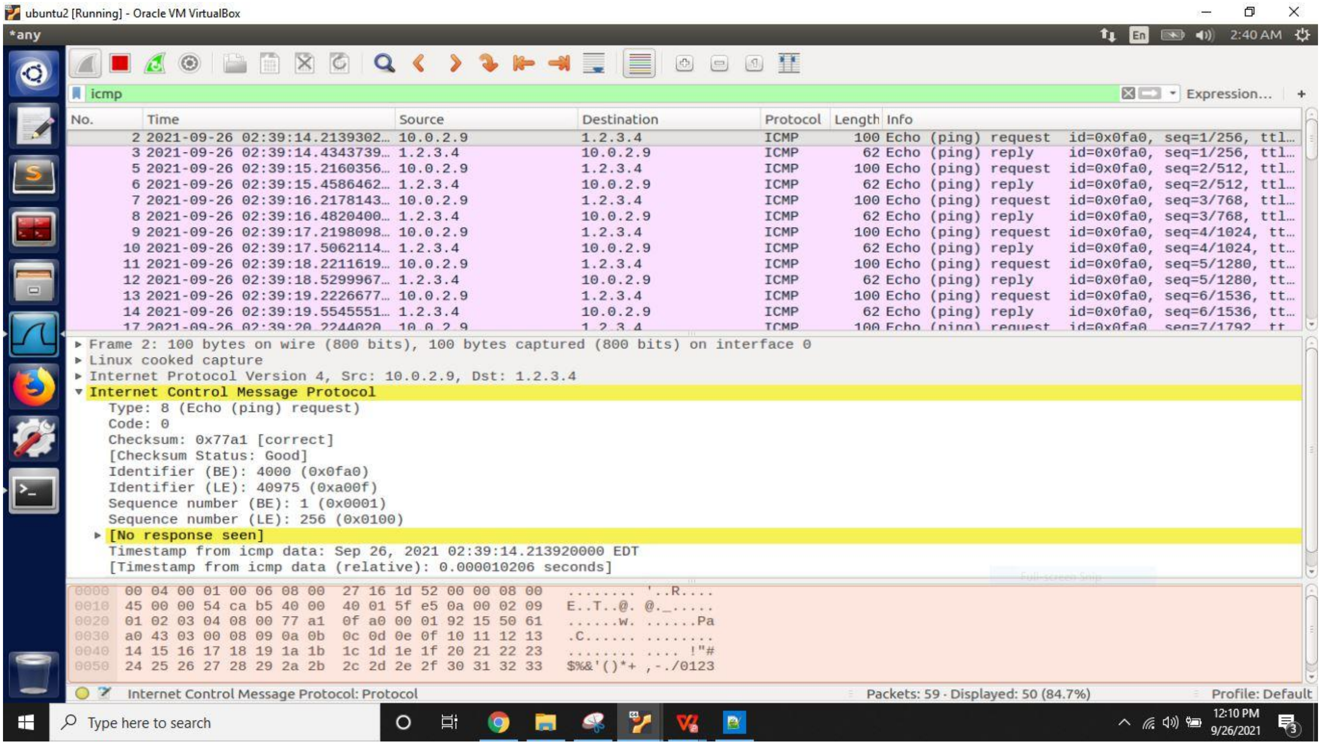


```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ gcc -o sniffspoof sniffspoof.c -lpcap
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ sudo ./sniffspoof
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
 Packet Sent from Attacker to host:10.0.2.9
^Z
[5]+  Stopped                 sudo ./sniffspoof
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ 
```

Observation on the victim machine



```
[09/26/21]seed@Nikhil_PES1UG20CS821:~$ ping 1.2.3.4
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
8 bytes from 1.2.3.4: icmp_seq=1 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=2 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=3 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=4 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=5 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=6 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=7 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=8 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=9 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=10 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=11 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=12 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=13 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=14 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=15 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=16 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=17 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=18 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=19 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=20 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=21 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=22 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=23 ttl=50 (truncated)
8 bytes from 1.2.3.4: icmp_seq=24 ttl=50 (truncated)
^Z
[3]+  Stopped                 ping 1.2.3.4
[09/26/21]seed@Nikhil_PES1UG20CS821:~$
```



```
No.  Time                          Source     Destination  Protocol Length Info
   2 2021-09-26 02:39:14.2139302… 10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0, seq=1/256, ttl…
   3 2021-09-26 02:39:14.4343739… 1.2.3.4    10.0.2.9     ICMP       62 Echo (ping) reply    id=0x0fa0, seq=1/256, ttl…
   5 2021-09-26 02:39:15.2160356… 10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0, seq=2/512, ttl…
   6 2021-09-26 02:39:15.4586462… 1.2.3.4    10.0.2.9     ICMP       62 Echo (ping) reply    id=0x0fa0, seq=2/512, ttl…
   7 2021-09-26 02:39:16.2178143… 10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0, seq=3/768, ttl…
   8 2021-09-26 02:39:16.4820400… 1.2.3.4    10.0.2.9     ICMP       62 Echo (ping) reply    id=0x0fa0, seq=3/768, ttl…
   9 2021-09-26 02:39:17.2198098… 10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0, seq=4/1024, tt…
  10 2021-09-26 02:39:17.5062114… 1.2.3.4    10.0.2.9     ICMP       62 Echo (ping) reply    id=0x0fa0, seq=4/1024, tt…
  11 2021-09-26 02:39:18.2211619… 10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0, seq=5/1280, tt…
  12 2021-09-26 02:39:18.5299967… 1.2.3.4    10.0.2.9     ICMP       62 Echo (ping) reply    id=0x0fa0, seq=5/1280, tt…
  13 2021-09-26 02:39:19.2226677… 10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0, seq=6/1536, tt…
  14 2021-09-26 02:39:19.5545551… 1.2.3.4    10.0.2.9     ICMP       62 Echo (ping) reply    id=0x0fa0, seq=6/1536, tt…
  17 2021-09-26 02:39:20.2244020  10.0.2.9   1.2.3.4      ICMP      100 Echo (ping) request  id=0x0fa0  seq=7/1792  tt
```

```
▸ Frame 2: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0
▸ Linux cooked capture
▸ Internet Protocol Version 4, Src: 10.0.2.9, Dst: 1.2.3.4
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x77a1 [correct]
    [Checksum Status: Good]
    Identifier (BE): 4000 (0x0fa0)
    Identifier (LE): 40975 (0xa00f)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
  ▸ [No response seen]
    Timestamp from icmp data: Sep 26, 2021 02:39:14.213920000 EDT
    [Timestamp from icmp data (relative): 0.000010206 seconds]
```

```
0000  00 04 00 01 00 06 08 00  27 16 1d 52 00 00 08 00   ........ '..R....
0010  45 00 00 54 ca b5 40 00  40 01 5f e5 0a 00 02 09   E..T..@. @._.....
0020  01 02 03 04 08 00 77 a1  0f a0 00 01 92 15 50 61   ......w. ......Pa
0030  a0 43 03 00 08 09 0a 0b  0c 0d 0e 0f 10 11 12 13   .C...... ........
0040  14 15 16 17 18 19 1a 1b  1c 1d 1e 1f 20 21 22 23   ........ .... !"#
0050  24 25 26 27 28 29 2a 2b  2c 2d 2e 2f 30 31 32 33   $%&'()*+ ,-./0123
```

Internet Control Message Protocol: Protocol   Packets: 59 · Displayed: 50 (84.7%)   Profile: Default

In this task both sniffing and spoofing are done by the attacker. The victim is going to ping the non existing IP address 1.2.3.4 but the victim will get the response this is done by the attacker.The attacker will sniff the packets and spoof it but the victim doesn't get to know that he is getting response from the attacker instead of machine having IP address 1.2.3.4