NETWORK SECURITY

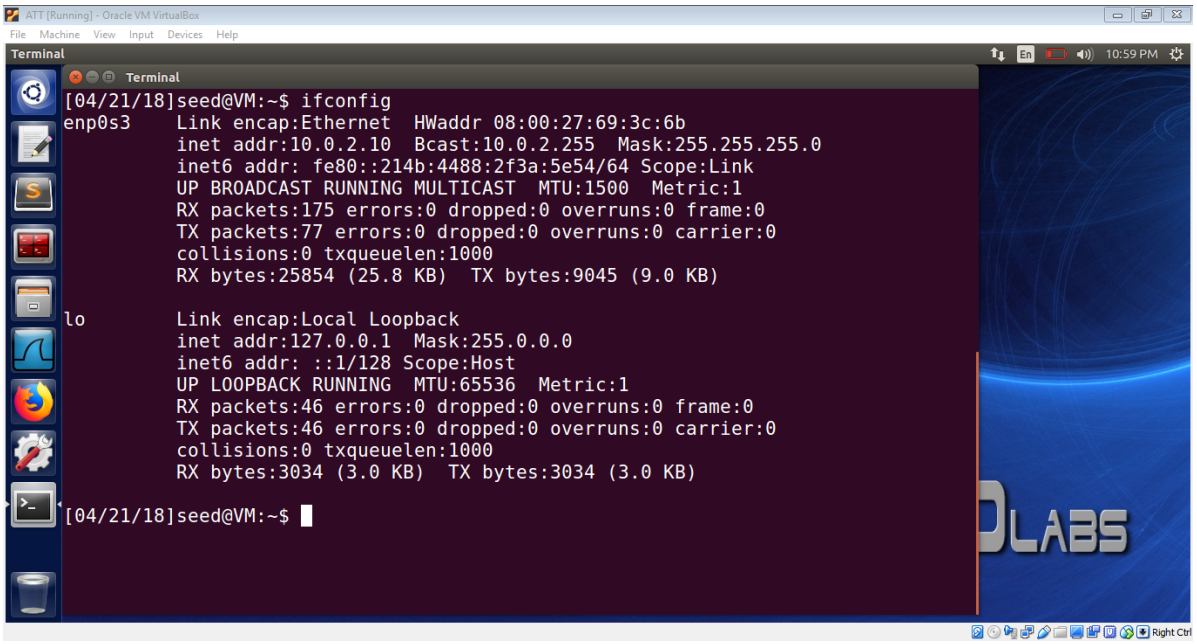
HOMEWORK-3

ANIMESH JAIN

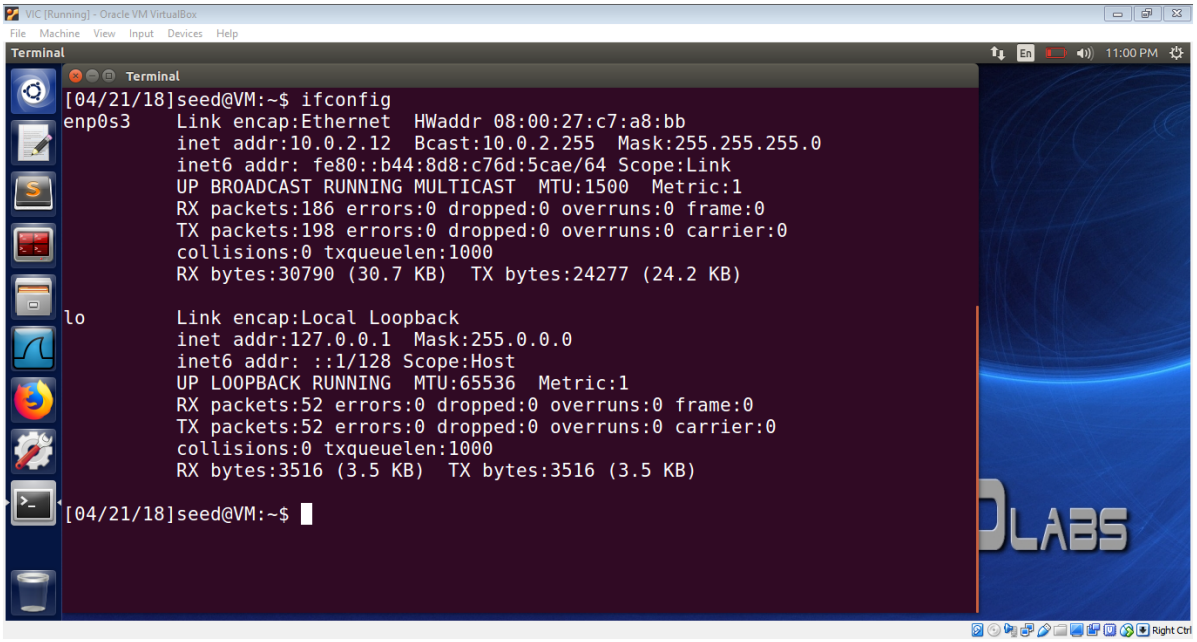
UIN – 669653208

As mentioned in the given Lab manual, we need at least 3 machines to conduct this lab. We will use one machine as the attacker and one as the victim machine. Remaining one will be used as the observer. Machines created are: -

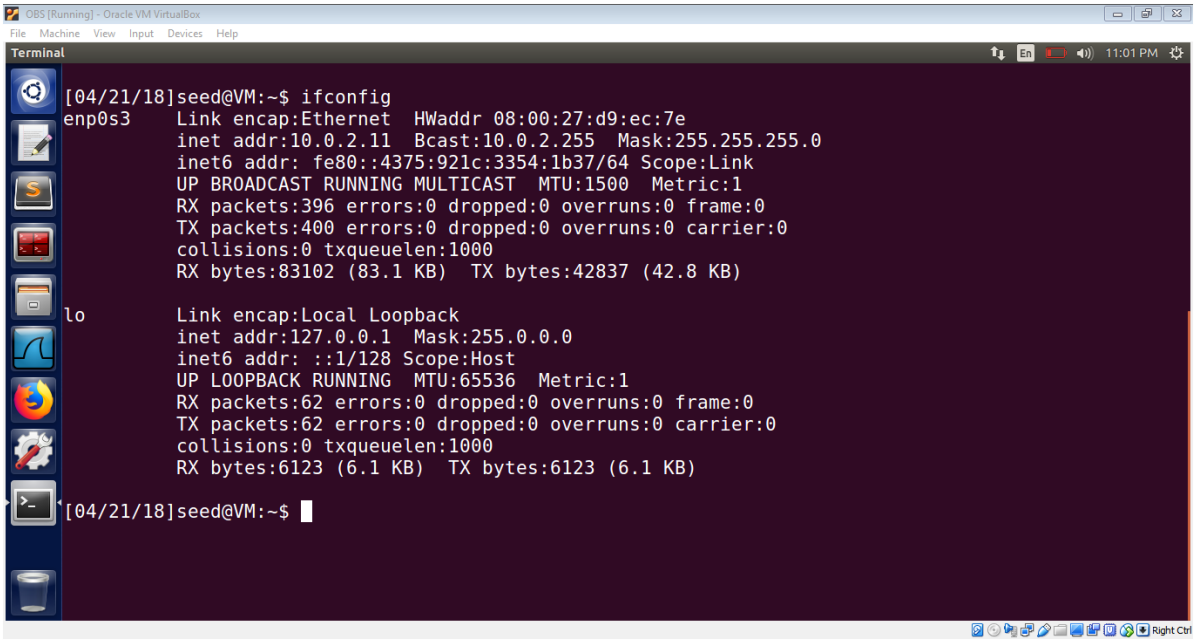
MACHINE "ATT" – Attacker (IP address – 10.0.2.10)



MACHINE "VIC" – Victim (IP address – 10.0.2.12)



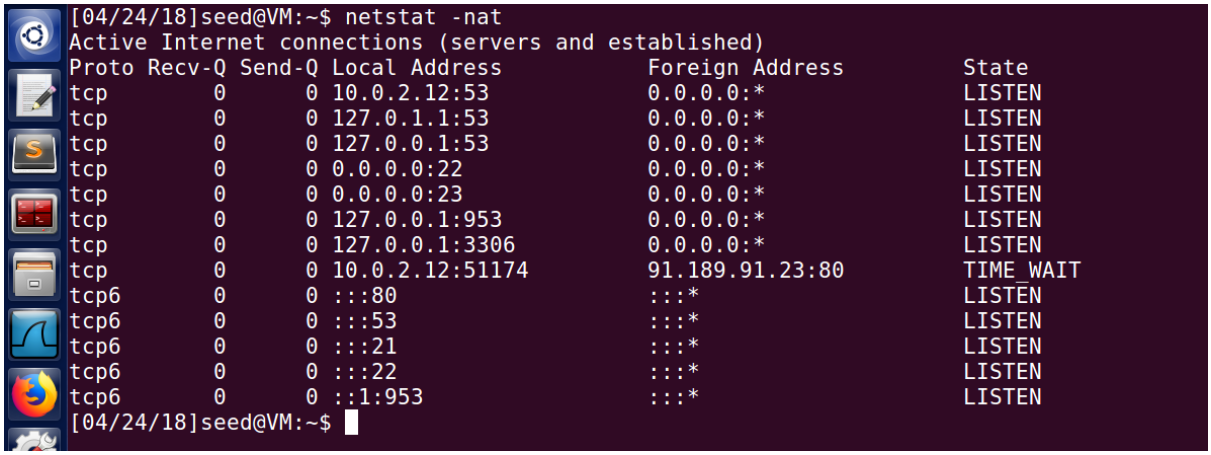
MACHINE "OBS" – Observer (IP address – 10.0.2.11)



We also need to install the **netwox tools** and the **scapy tools** in our machines as they will be used in the lab to launch different kinds of attacks. Once we are done with setting up the entire lab environment we will move forward to conduct the Lab tasks.

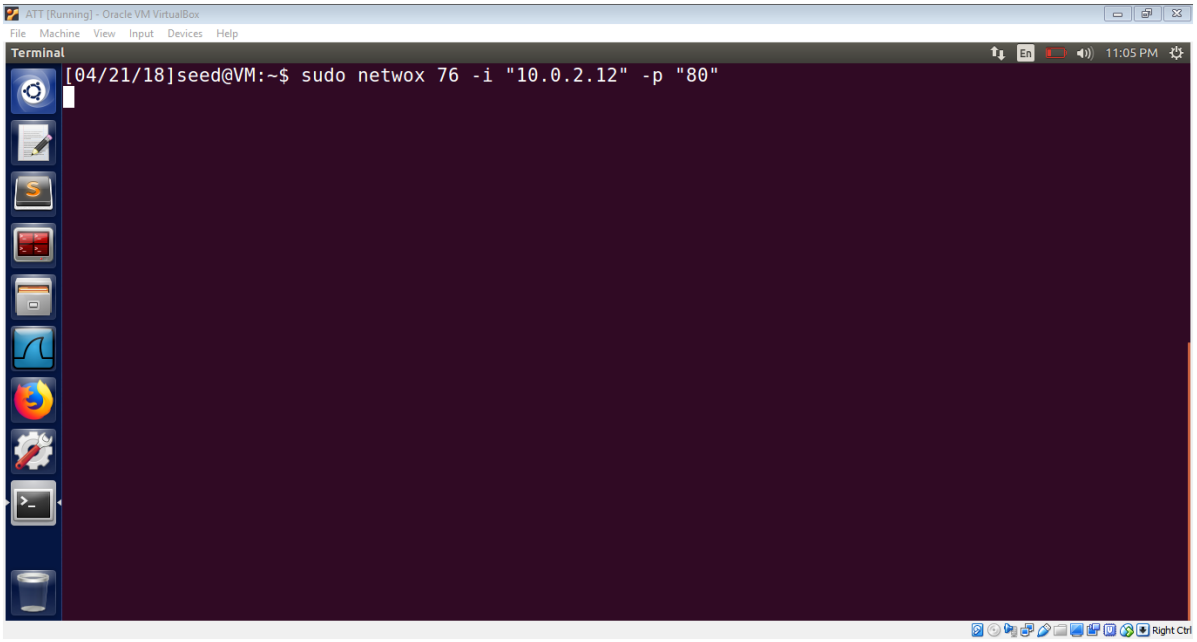
3.1 TASK 1: SYN FLOODING ATTACK

In this task the tool we require to launch the SYN flooding attack is the **netwox tool numbered 76**. The command we use to launch the attack is "**sudo netwox 76 -i ip -p port [-spoofip]**. Now since we want to attack the victim machine, that is machine "VIC", we will give the ip as the ip address of machine "VIC", that is "10.0.2.12" and we will choose port as 80 which is used for http.



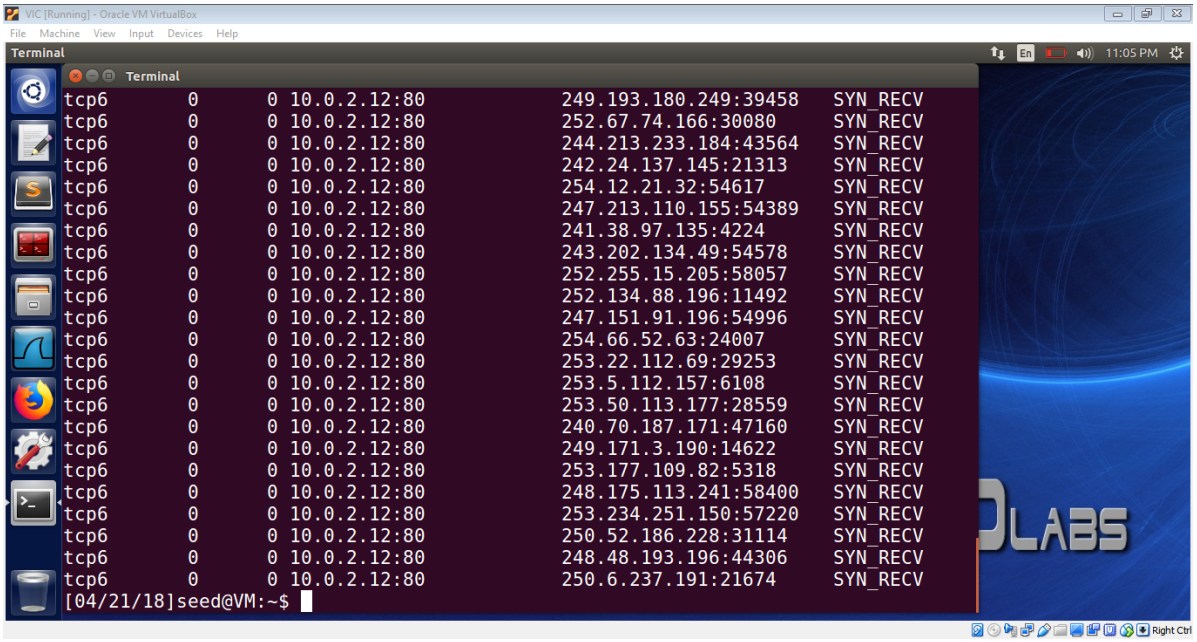
```
[04/24/18]seed@VM:~$ netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 10.0.2.12:53            0.0.0.0:*               LISTEN
tcp        0      0 127.0.1.1:53            0.0.0.0:*               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 10.0.2.12:51174         91.189.91.23:80         TIME_WAIT
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 :::53                   :::*                    LISTEN
tcp6       0      0 :::21                   :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:953                 :::*                    LISTEN
[04/24/18]seed@VM:~$
```

First, we will look at the state of the machine "VIC" when no attack has been launched from the attacker machine. So, to list the TCP queuing we will use command "**netstat -nat**". This command helps in checking the usage of the queue and to know the status of the connections. When there is no attack, we observe that on machine "VIC" the state of the connections is "LISTEN". This tells that the machine is listening to all ports who wants to request for the connection establishment. This the desired state when there is no attack.



```
[04/21/18]seed@VM:~$ sudo netwox 76 -i "10.0.2.12" -p "80"
```

Now, attacking machine "VIC' using the command which is shown in the screenshot. This will launch the SYN flooding attack. This way the attacker will send many SYN request to "VIC" without wanting to finish the 3-way handshake process.



Now we will make use of command “netstat-nat” on the victim machine to check the queuing of the TCP packets and to know the status of the connections. We observe that the state of the connections is “SYN_RECV”. This is because the attacker has flooded the victim with the request packets which makes the victim’s queue to be full and therefore the victim is not able to take requests for any more connections. This tells that the **attack was successful**.

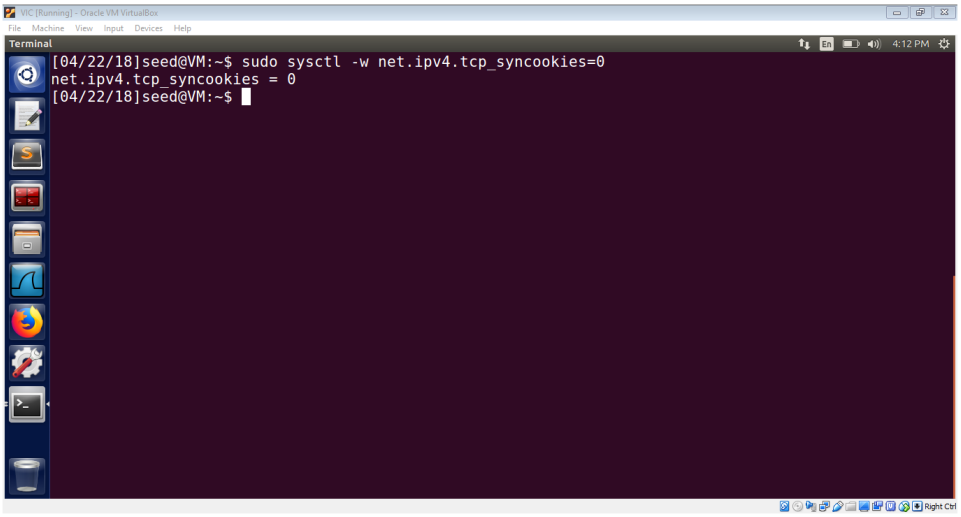
SYN COOKIE COUNTERMEASURE:

SYN cookie is used as a defense mechanism to stop the SYN flooding attack. **To check the cookie status, we can use “sudo sysctl -a | grep cookie” command”.**

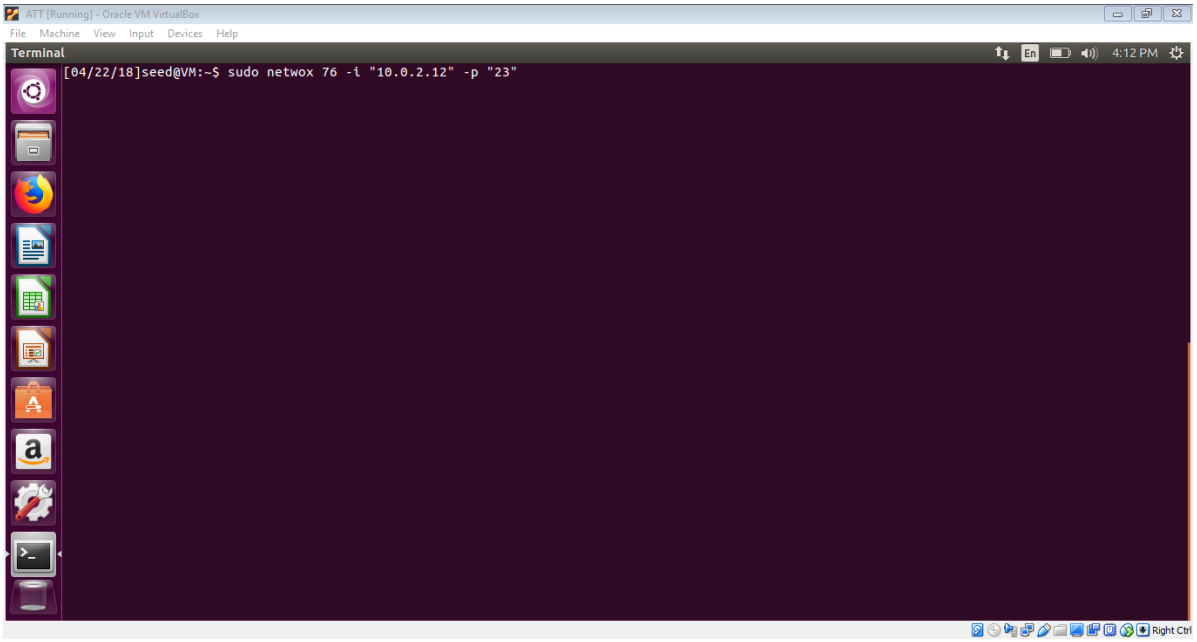
Working of Cookie: - When a connection request is sent to a server so instead of sending the SYN RECV packets to the sender, the server sends a cookie to the sender and allocates the resources to the sender only when the sender sends back the cookie along with the SYN request to the Server. This way the server authenticates the client and can stop the SYN flooding attack.

Turning off SYN cookie

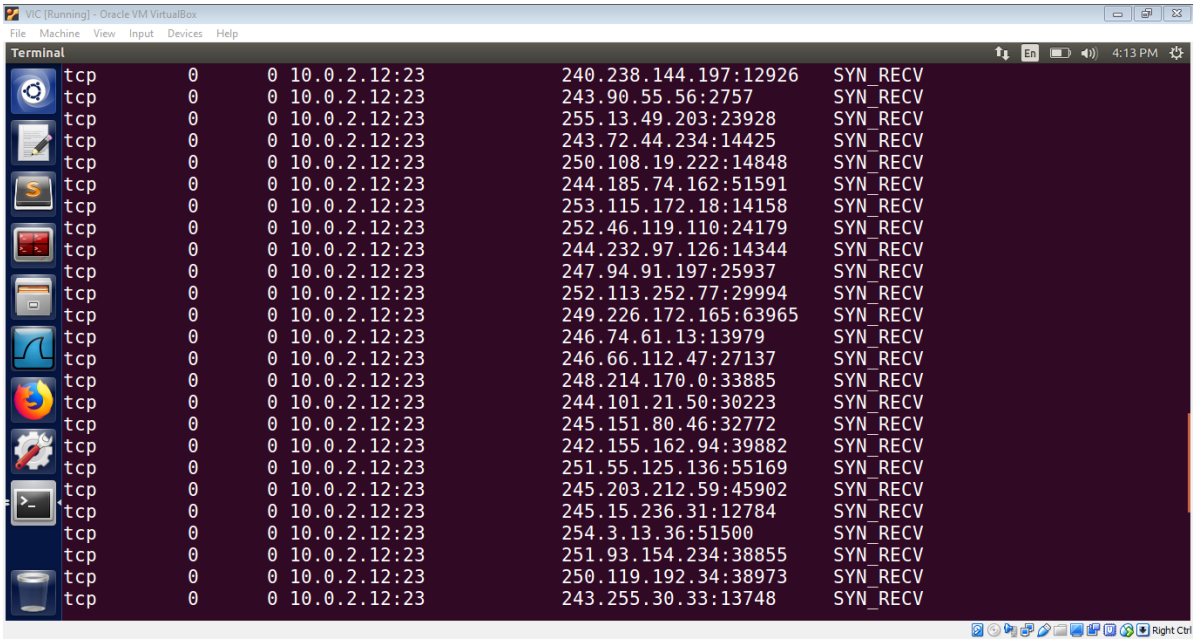
To **disable the SYN cookie** (if it is already enabled, that is set to 1) we use “**sudo sysctl -w net.ipv4.tcp_syncookies=0**” command.



Turning the **SYN cookies off** on the “VIC” machine. (see the above screenshot).



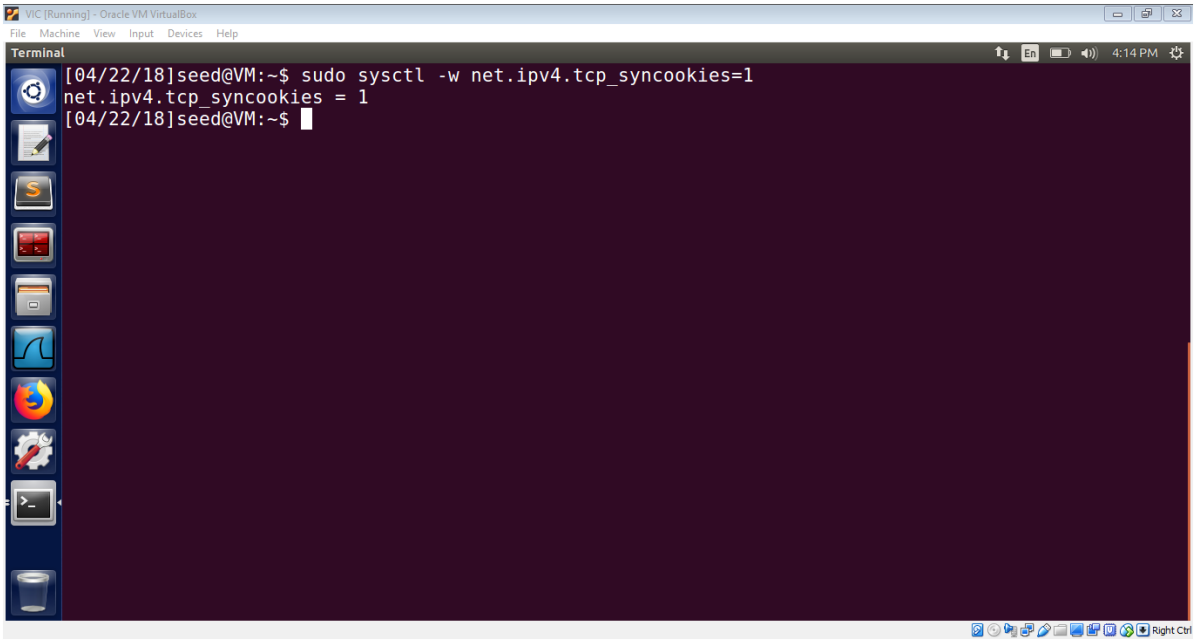
Now going on the “ATT machine” and launching the SYN flooding attack to “VIC” machine.

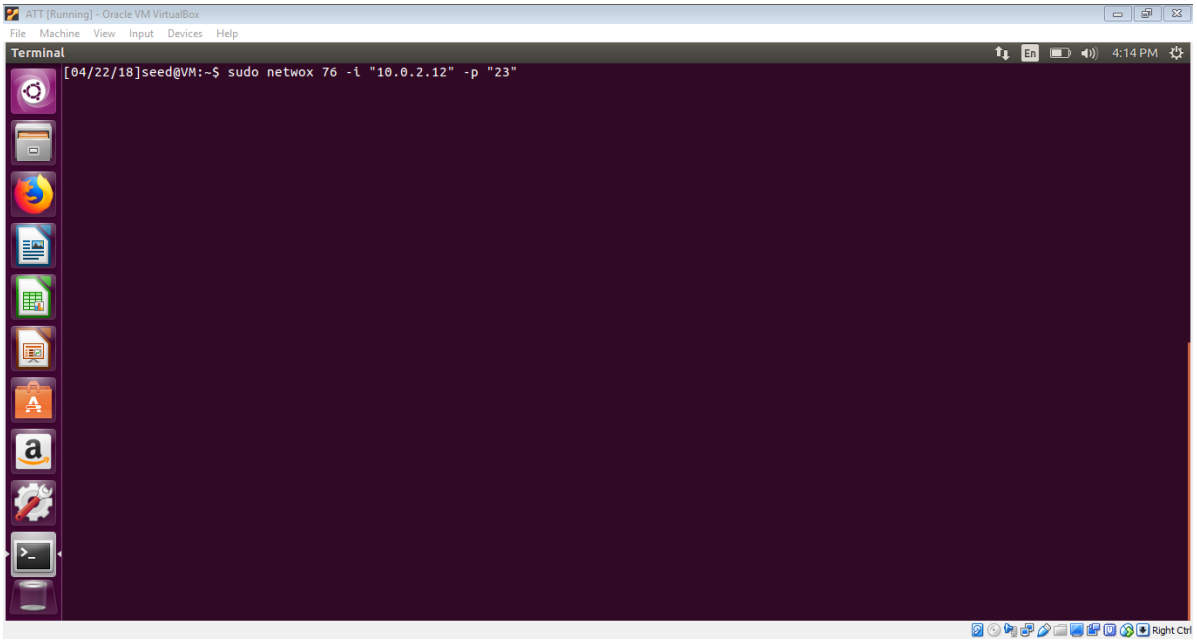


Again, we can see that after giving “netstat -nat” command that the status of the connections is “SYN_RECV”. This represents that when SYN cookies are disabled, **attack on machine “VIC” is successful** as “SYN_RECV” represents that “VIC” is not able to listen to any more connection requests.

Turning on SYN cookies

To **enable** the SYN cookie (if it is already enabled, that is set to 1) we use “**sudo sysctl -w net.ipv4.tcp_syncookies=0**” command.



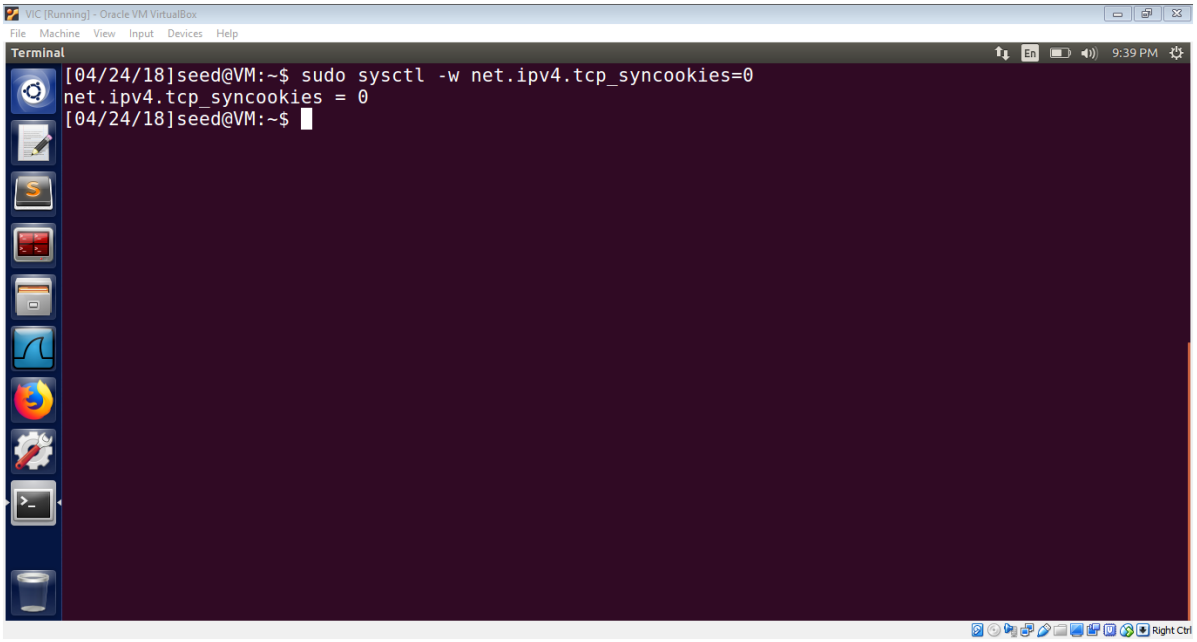


On “ATT” giving the command to launch the SYN flooding attack on machine “VIC”.

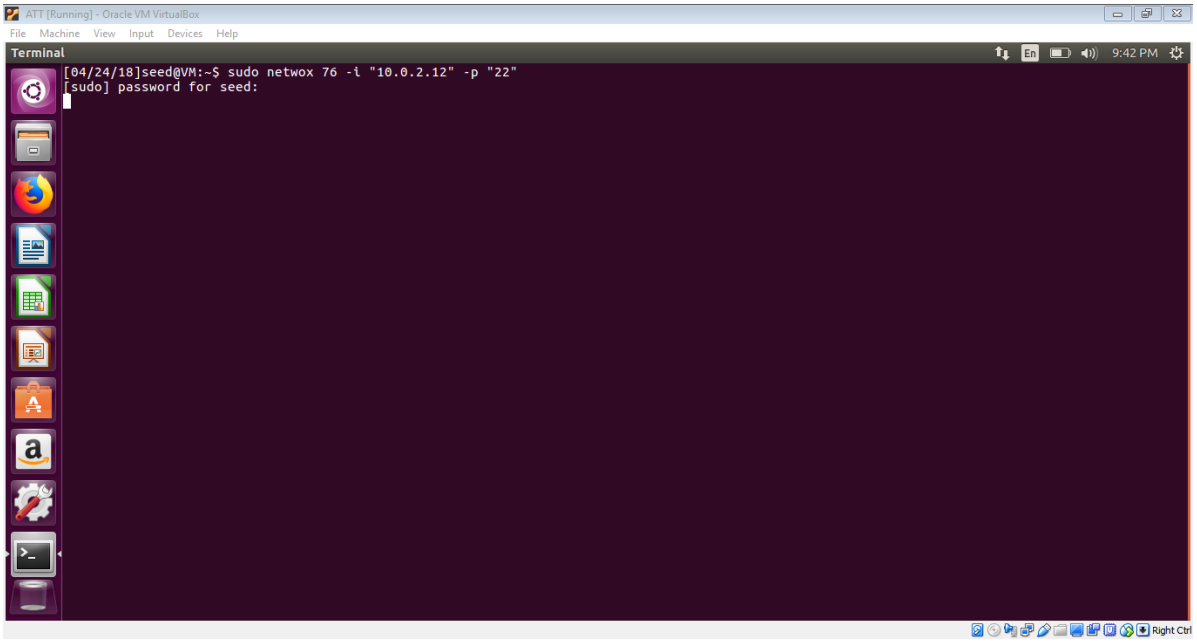
On giving the “netstat -nat” command on machine “VIC” we should get the state of the connections to be either “LISTEN” or “ESTABLISHED”. If these are the state, then it represents that the attack was not successful. This is what we desire for that is **unsuccessful attack**. The purpose was to showcase that when the cookies are enabled, then the SYN flooding attack can be stopped. But due to some issues we are getting the state to be “SYN_RECV”, which is incorrect.

To showcase the correct working of the SYN cookies I will establish a SSH connection between machine “VIC” and machine “OBV” and then launch the SYN flooding attack from machine “ATT”.

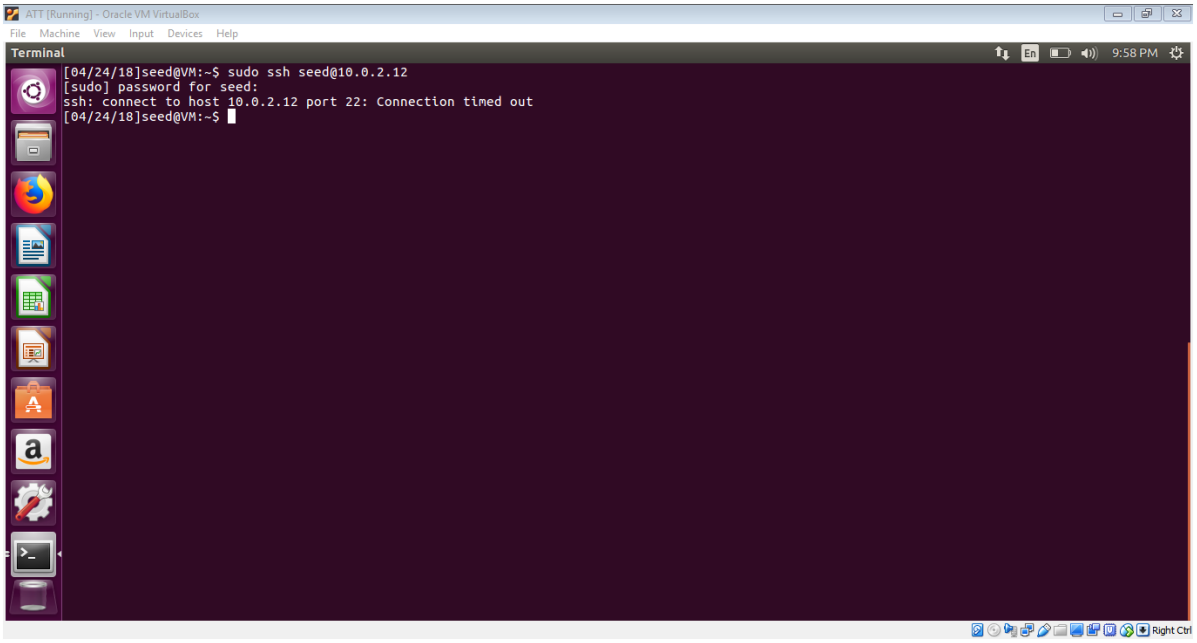
SYN COOKIES = 0



On machine “VIC” turning off the SYN cookies as shown in the above screenshot.

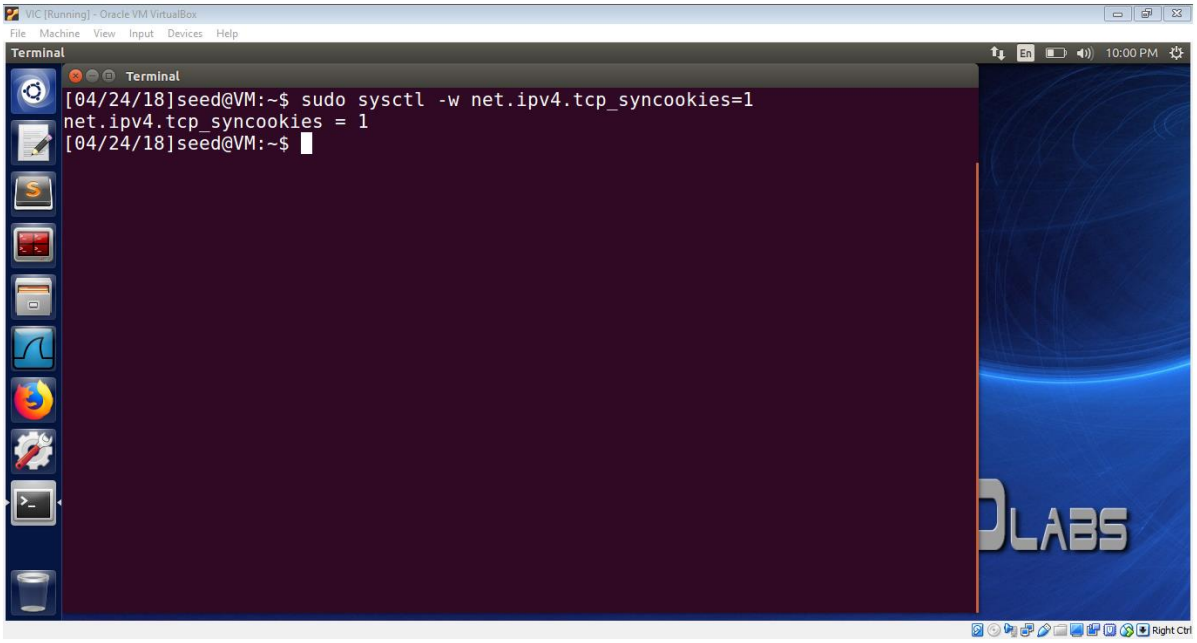


Launching the SYN flooding attack from “ATT” to “VIC”.

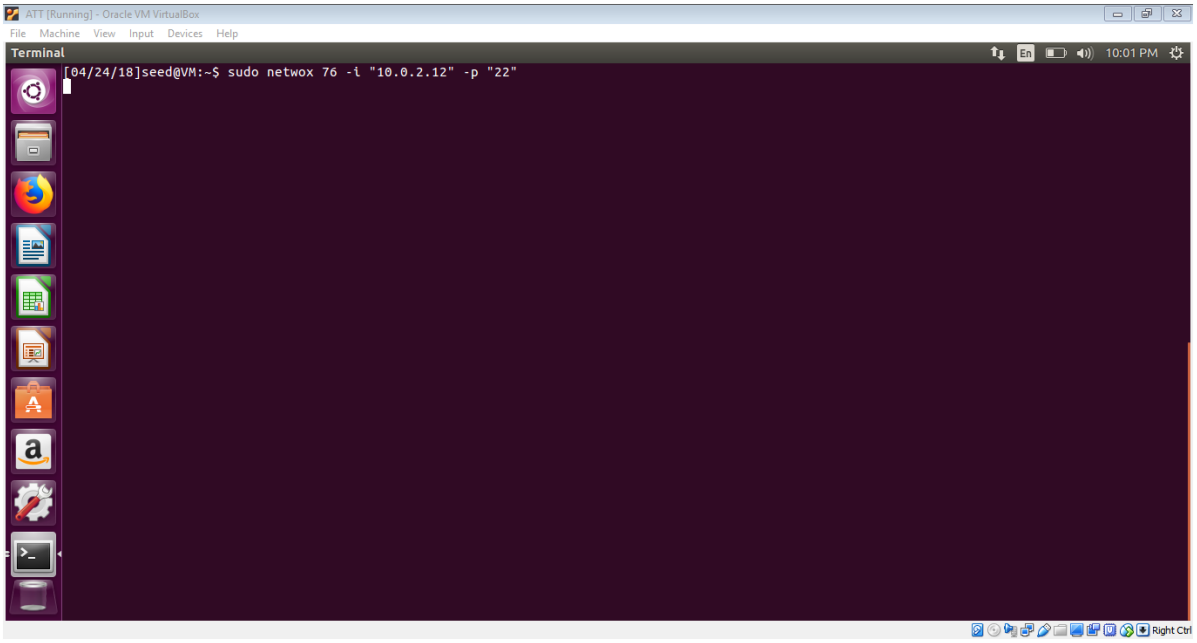


Now trying to establish connection between “ATT” and “VIC” by SSHing to “VIC” we observe that the connection was not established. This represents that the **attack was successful**.

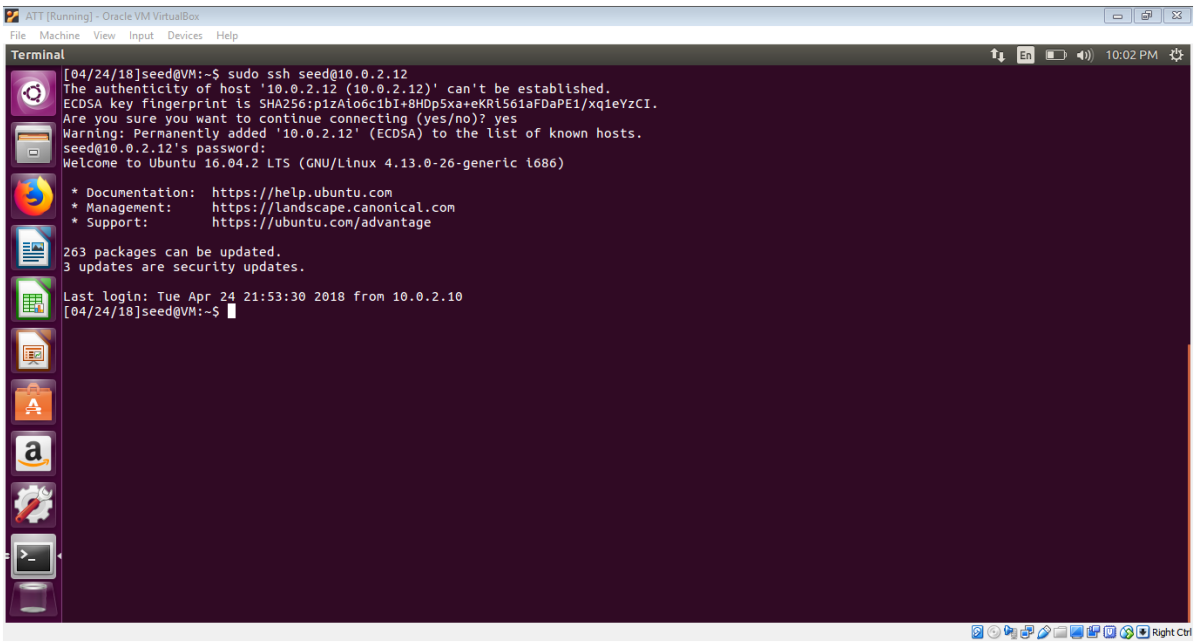
SYN COOKIES = 1



Turning the SYN cookies ON on machine “VIC”.



Launching the SYN flooding attack from “ATT” to “VIC”.



Now trying to establish connection between “ATT” and “VIC” by SSHing to “VIC” we observe that the connection was established. This represents that the **attack was unsuccessful**. **This tells that using SYN cookie we can defend the SYN flooding attacks.**

This is the correct working of SYN cookies.

3.2 TASK 2: TCP RST ATTACKS ON TELNET AND SSH CONNECTIONS

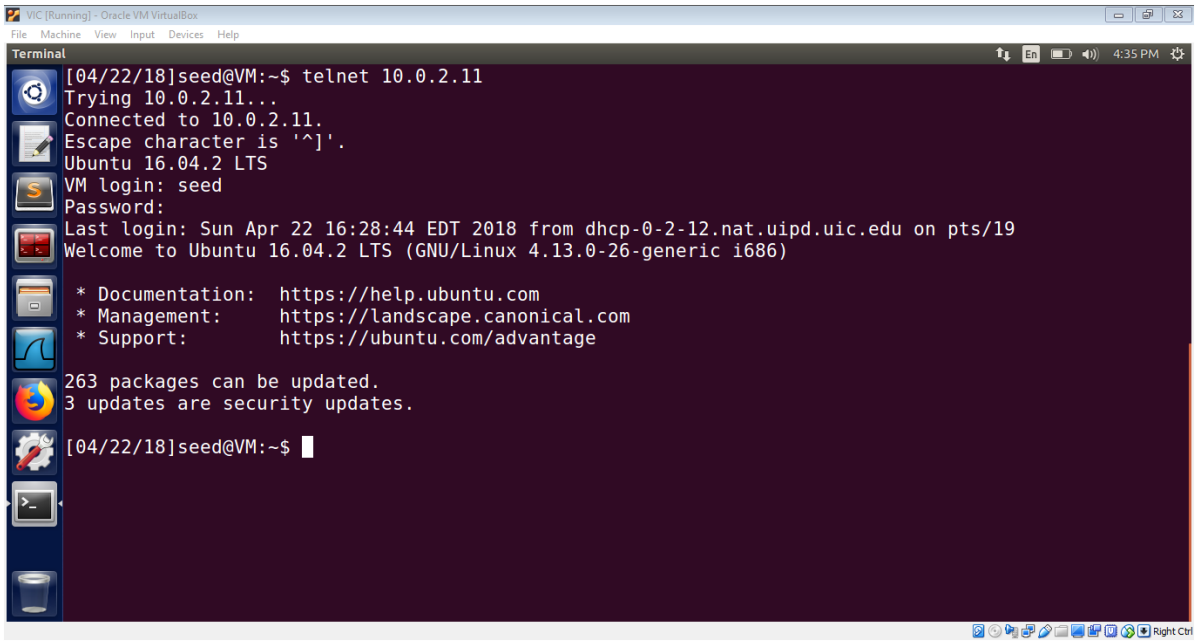
In this task we will establish a connection (SSH and TELNET) between machine “VIC” and machine “OBS”. Then we will launch an TCP RST attack to break the connection between the two machines.

So here we will assign machine “ATT” to be the attacker whereas “VIC” and “OBS” will be the victims.

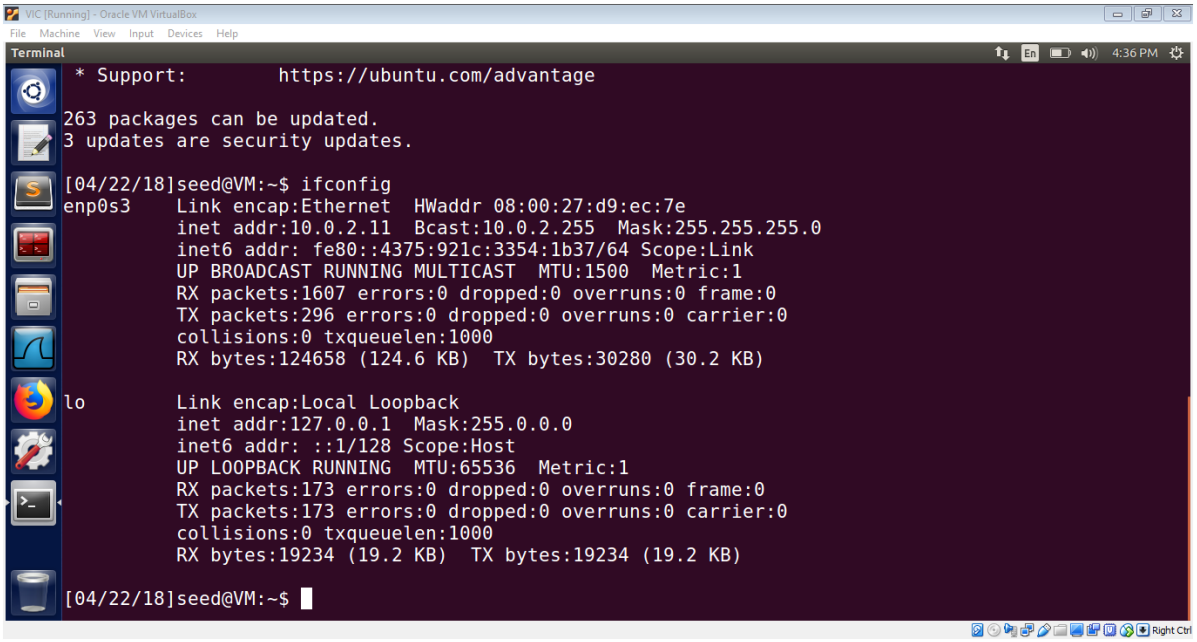
a) **CONNECTION ESTABLISHMENT BETWEEN “VIC” AND “OBS” THROUGH TELNET**.

To launch the attack, we will use the **netwox tool numbered 78**. On machine “ATT”, which is the machine we will use to launch the TCP RST attack we will give the following command: -

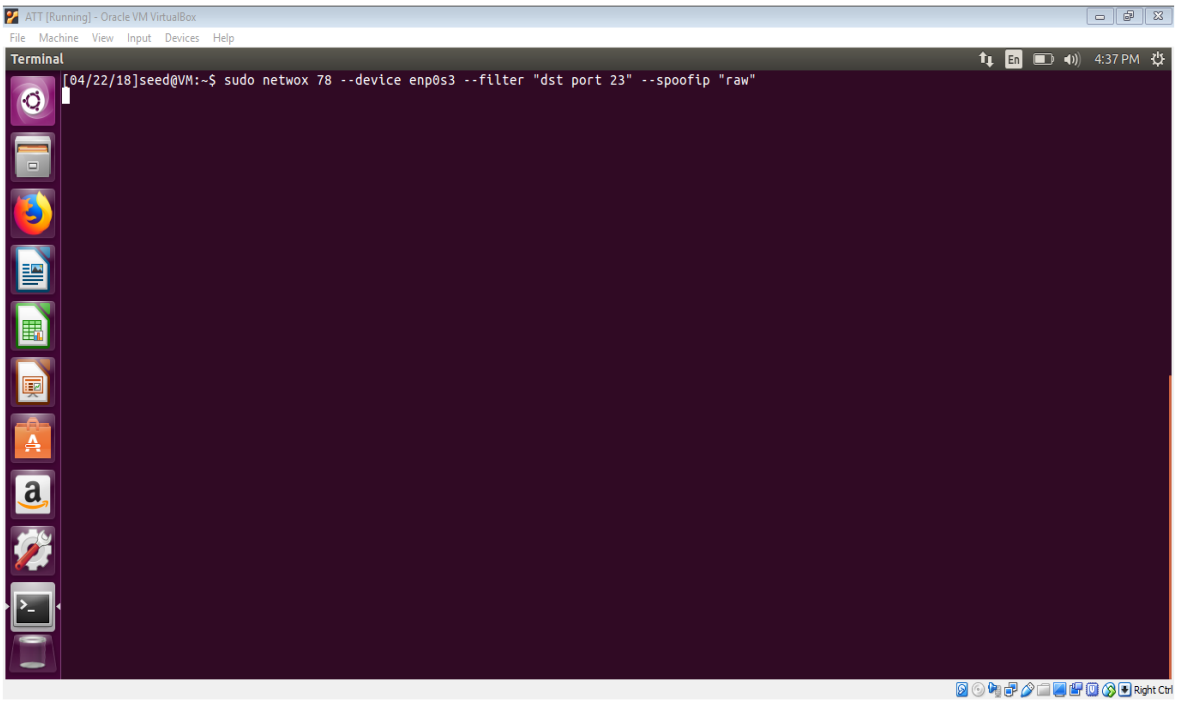
- **Sudo netwox 78 –device ens0p3 –filter “dst port 23” –spoofip “raw”**.



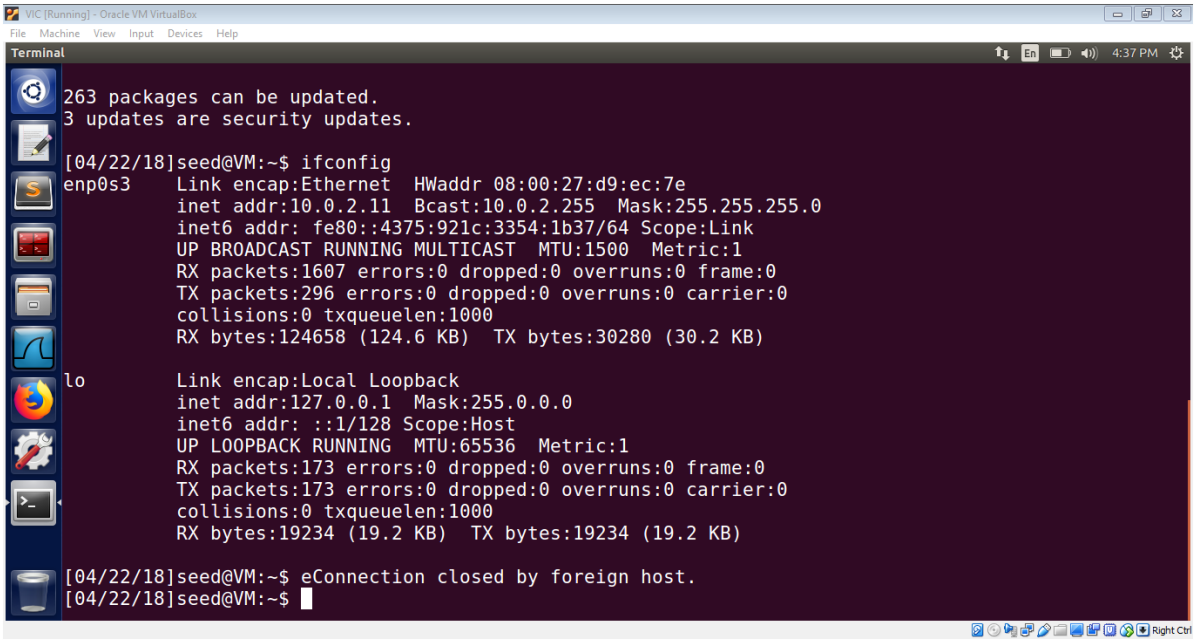
In the above screenshot we are establishing the connection between “VIC” and “OBS” using Telnet.



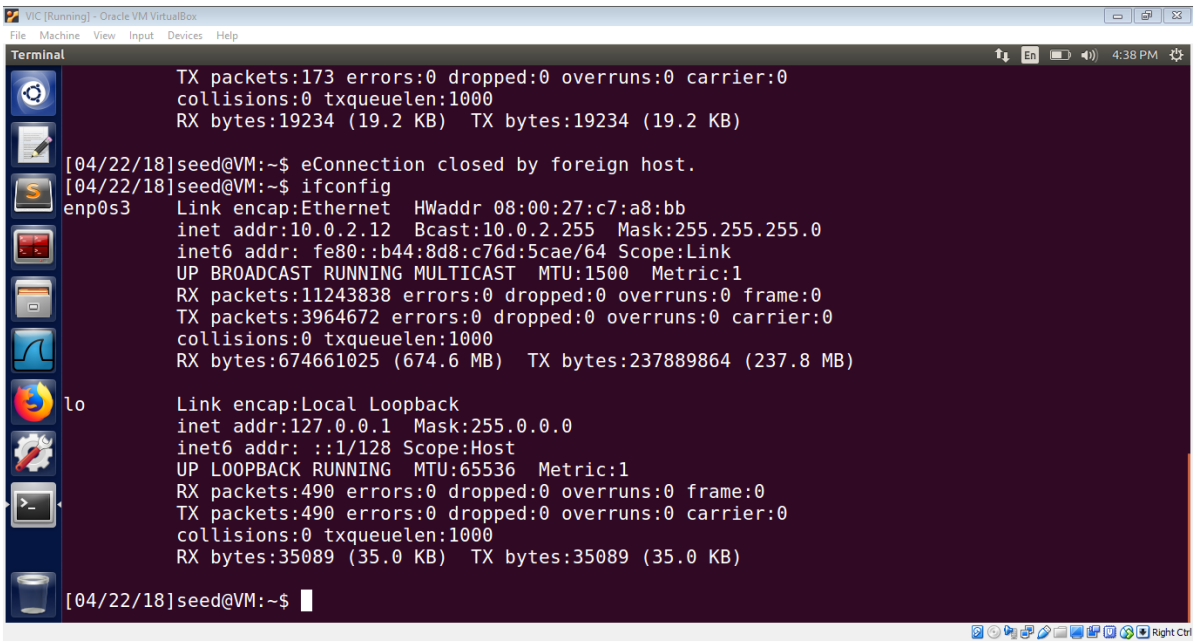
In machine “VIC” when we type the “ifconfig” command to check the machine IP address, we see that it is showing the IP address of machine “OBS” which represents the connection has been established.



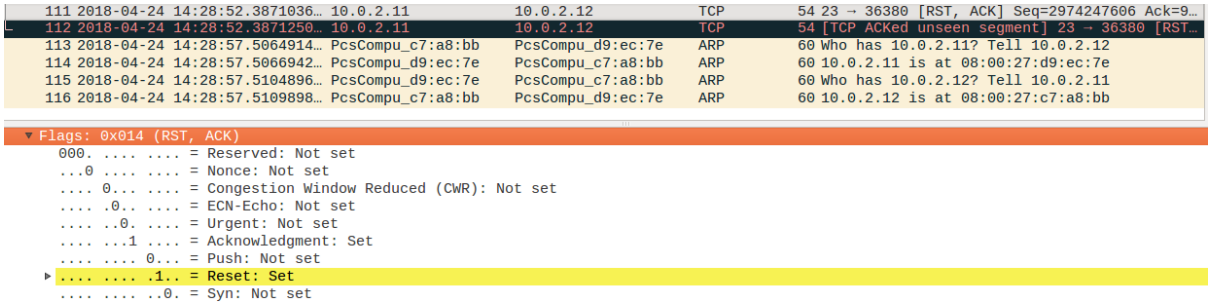
Here we are giving the command on machine “ATT” to launch the TCP RST attack to break the established telnet connection between the two victim machines.



After the attack is launched when we try to access machine “VIC” we get the above-mentioned status, i.e. “Connection closed by foreign host”. This signifies that the connection between the two victim machines is broken.



If we type the “ifconfig” command again on “VIC”, the IP address we receive is the IP address of machine “VIC” and not machine “OBS”.



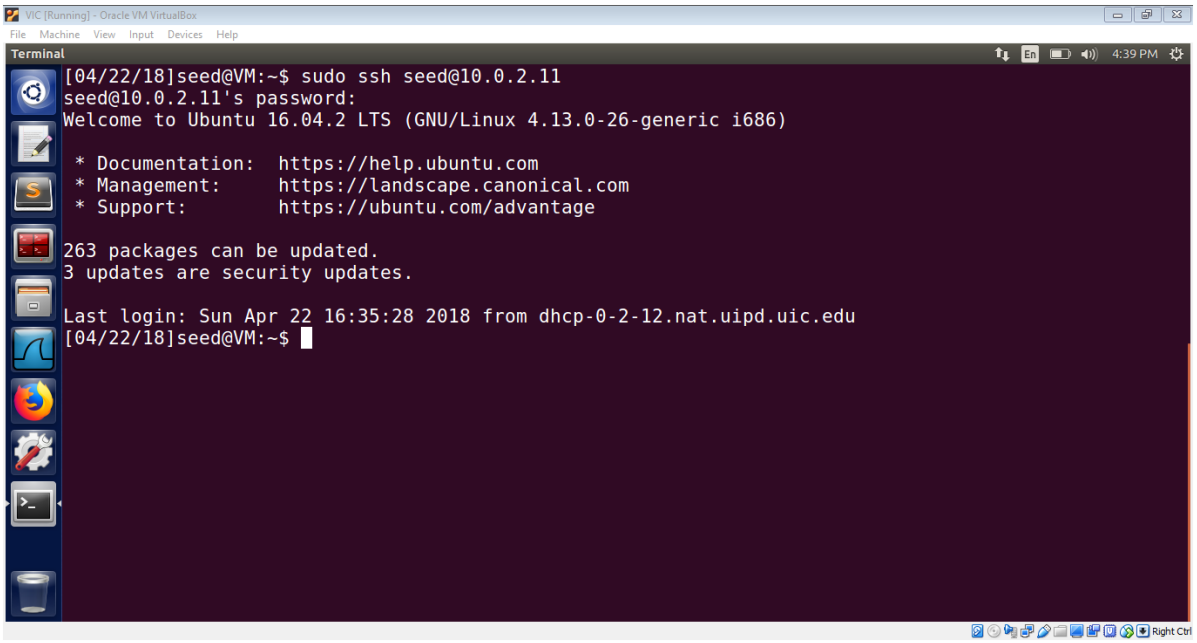
If we go through the packet we can clearly say that the **reset flag is set to 1**, which in turn will terminate the connection.

b) CONNECTION ESTABLISHMENT BETWEEN "VIC" AND "OBS" THROUGH SSH.

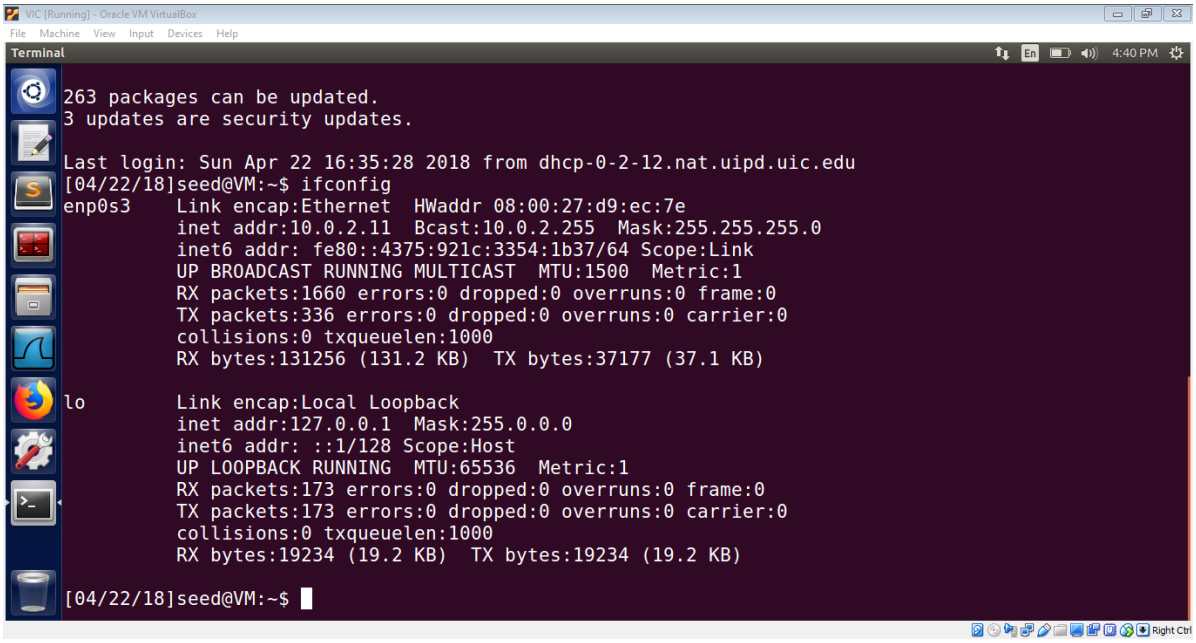
In this case also we will do the same things that we did in part (a). Only difference is that here the connection establishment between machine "VIC" and machine "OBS" is made using SSH. The command that we will give on machine "ATT" to attack will be almost same. Here instead of giving the destination port as 23 we must assign port 22 as the destination port.

Command to launch the TCP RST attack is: -

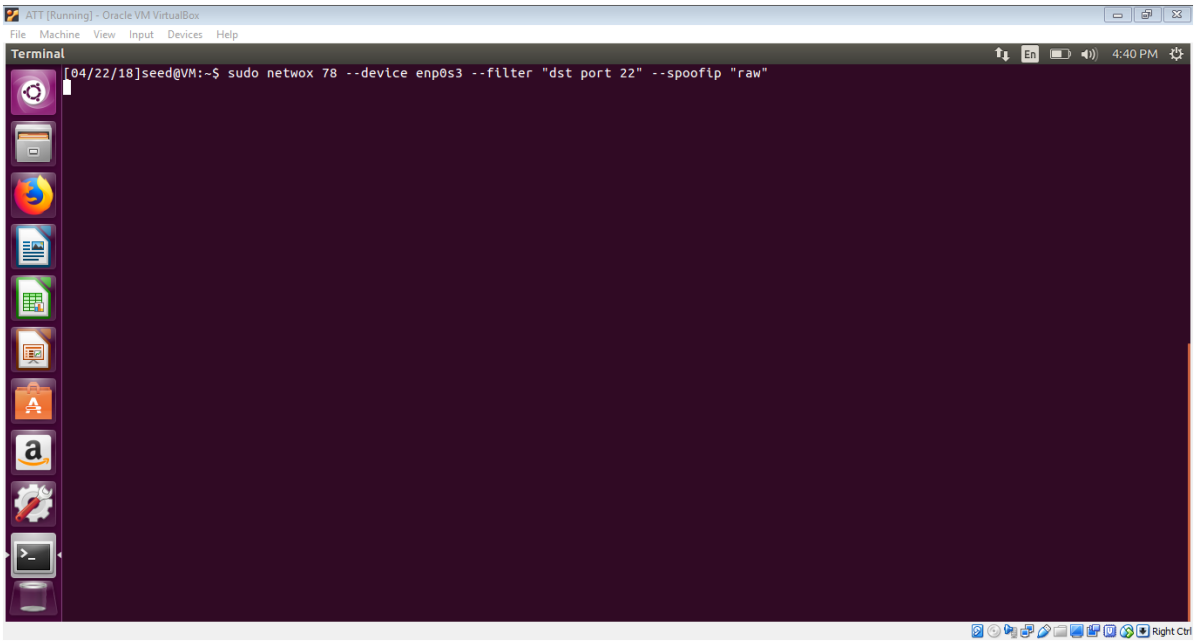
- **Sudo netwox 78 –device ens0p3 –filter "dst port 22" –spoofip "raw"**.



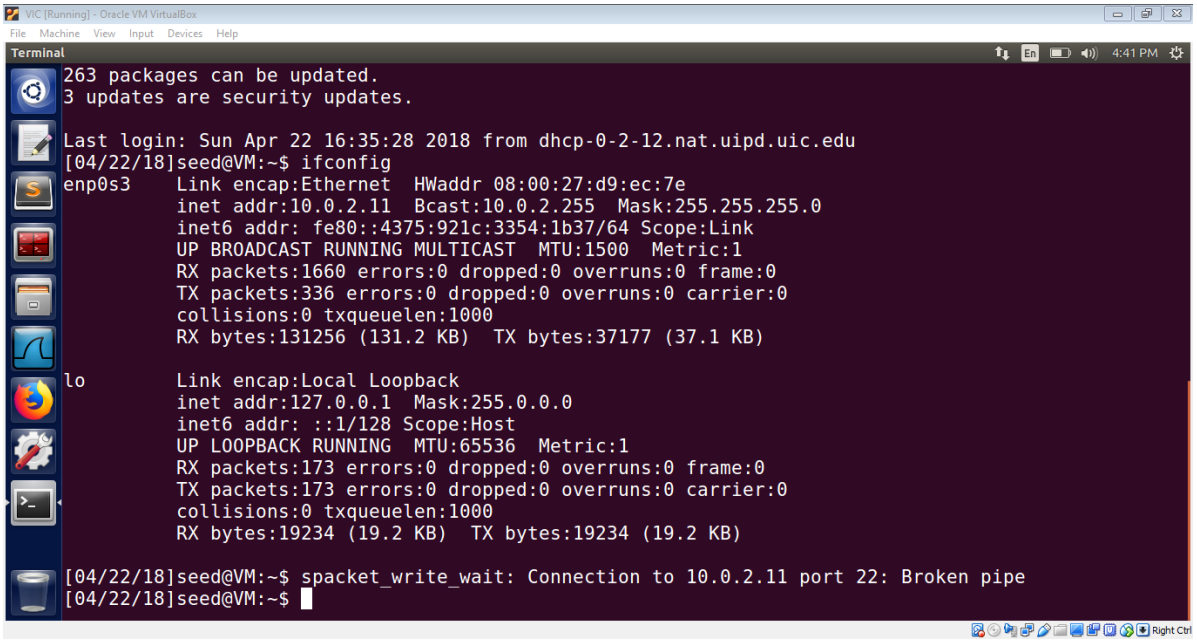
Establishing the connection between "VIC" and "OBS" using SSH.



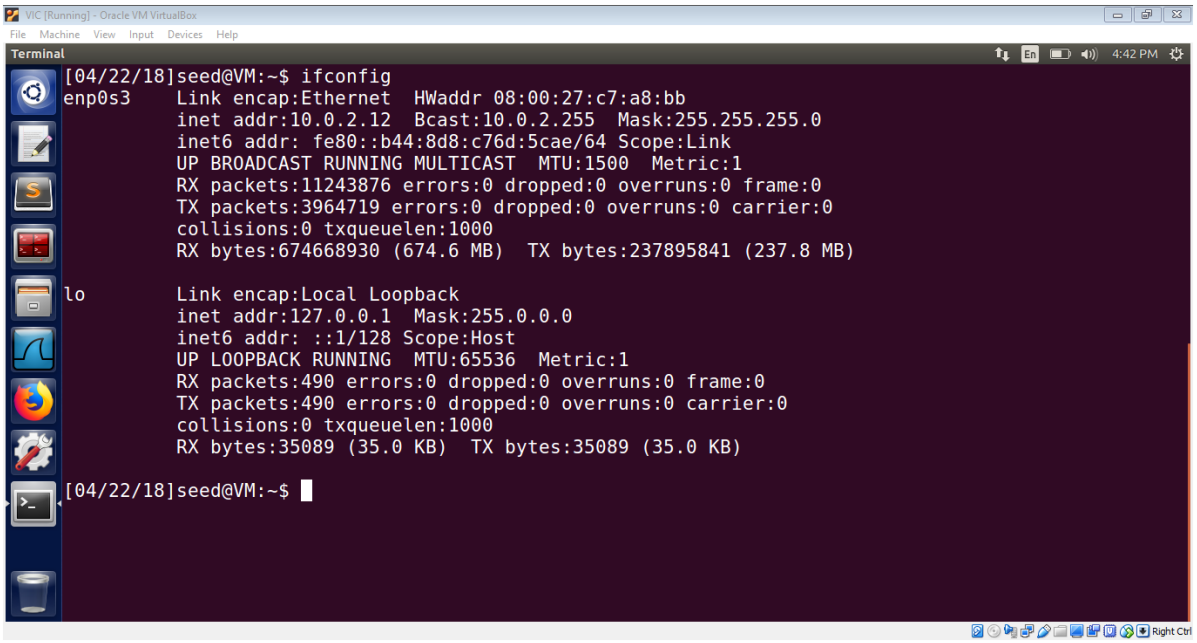
Using the “ifconfig” command we get confirmation that the connection establishment is done as we see the IP address of machine “OBS” on machine “VIC”.



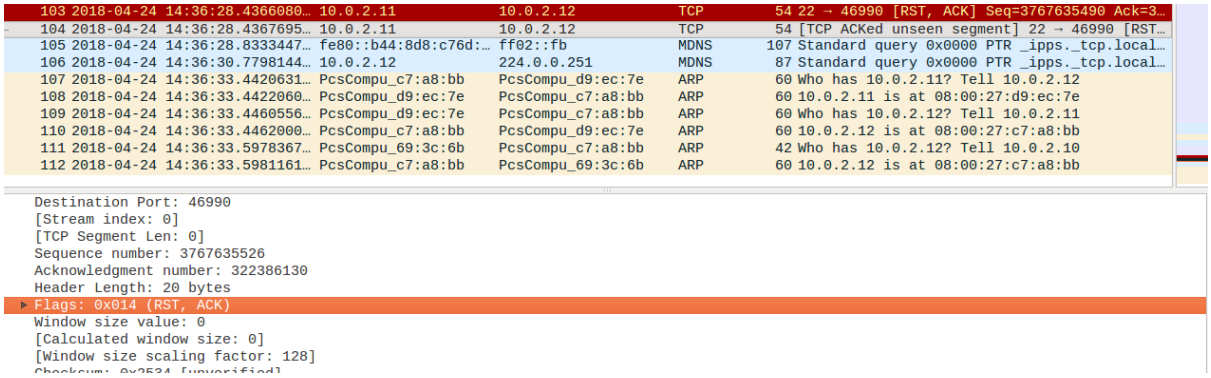
Giving the command on machine “ATT” to launch the TCP RST attack to break the established connection between the two victim machines.



Accessing the machine “VIC” after the attack is launched gives the command in the above screenshot which signifies that the connection is broken between the machines “VIC” and “OBS”.



On again using the “ifconfig” command on “VIC” we get to see the IP address of machine “VIC” and not of “OBS” which indicates connection between the two machines is no longer established.



103	2018-04-24 14:36:28.4366080…	10.0.2.11	10.0.2.12	TCP	54	22 → 46990 [RST, ACK] Seq=3767635490 Ack=3…
104	2018-04-24 14:36:28.4367695…	10.0.2.11	10.0.2.12	TCP	54	[TCP ACKed unseen segment] 22 → 46990 [RST…
105	2018-04-24 14:36:28.8333447…	fe80::b44:8d8:c76d:…	ff02::fb	MDNS	107	Standard query 0x0000 PTR _ipps._tcp.local…
106	2018-04-24 14:36:30.7798144…	10.0.2.12	224.0.0.251	MDNS	87	Standard query 0x0000 PTR _ipps._tcp.local…
107	2018-04-24 14:36:33.4420631…	PcsCompu_c7:a8:bb	PcsCompu_d9:ec:7e	ARP	60	Who has 10.0.2.11? Tell 10.0.2.12
108	2018-04-24 14:36:33.4422060…	PcsCompu_d9:ec:7e	PcsCompu_c7:a8:bb	ARP	60	10.0.2.11 is at 08:00:27:d9:ec:7e
109	2018-04-24 14:36:33.4460556…	PcsCompu_d9:ec:7e	PcsCompu_c7:a8:bb	ARP	60	Who has 10.0.2.12? Tell 10.0.2.11
110	2018-04-24 14:36:33.4462000…	PcsCompu_c7:a8:bb	PcsCompu_d9:ec:7e	ARP	60	10.0.2.12 is at 08:00:27:c7:a8:bb
111	2018-04-24 14:36:33.5978367…	PcsCompu_69:3c:6b	PcsCompu_c7:a8:bb	ARP	42	Who has 10.0.2.12? Tell 10.0.2.10
112	2018-04-24 14:36:33.5981161…	PcsCompu_c7:a8:bb	PcsCompu_69:3c:6b	ARP	60	10.0.2.12 is at 08:00:27:c7:a8:bb

```
Destination Port: 46990
[Stream index: 0]
[TCP Segment Len: 0]
Sequence number: 3767635526
Acknowledgment number: 322386130
Header Length: 20 bytes
▸ Flags: 0x014 (RST, ACK)
Window size value: 0
[Calculated window size: 0]
[Window size scaling factor: 128]
```

If we go through the packet we can clearly say that the **reset flag is set to 1**, which in turn will terminate the connection.

CONDUCTING THE TCP ATTACK USING SCAPY

Using Scapy also we can attack networks as Scapy is a python script that permits the user to sniff and forge the network packets. In this task, we will make use of one such script that is provided in the question manual itself. We will first establish the connection between "VIC" and "OBS" using TELNET and conduct the attack using Scapy. After this we will repeat the same process by establishing the connection between the two machines using SSH.

```
#!/usr/bin/python
from scapy.all import *

ip  = IP(src="@@@@", dst="@@@@")
tcp = TCP(sport=@@@@, dport=@@@@, flags="@@@@", seq=@@@@, ack=@@@@)
```

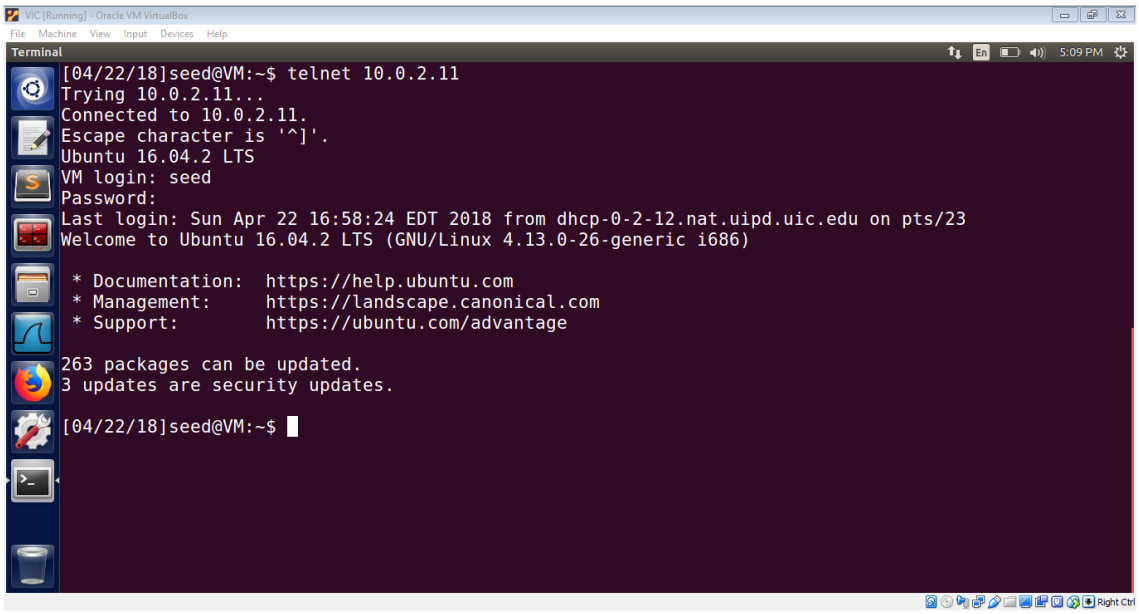
```
pkt = ip/tcp
ls(pkt)
send(pkt,verbose=0)
```

CONNECTION ESTABLISHMENT BETWEEN "VIC" AND "OBS" THROUGH TELNET

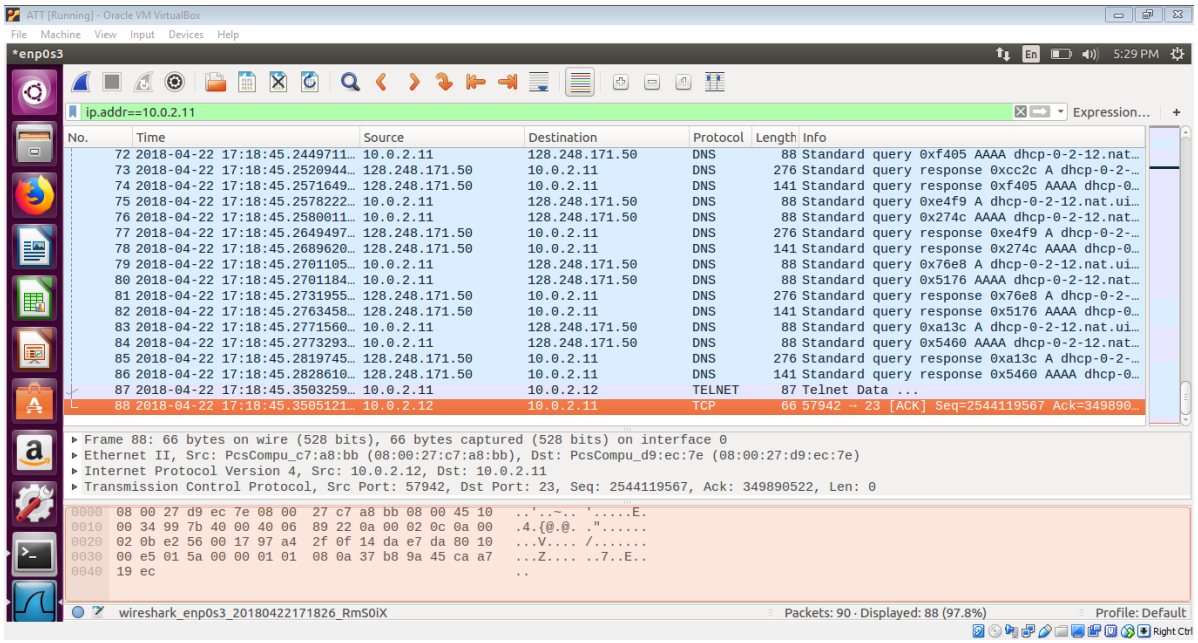
Procedure: -

- On attacker machine, that is machine "ATT", we will open the wireshark and will start capturing the packets.
- Then we will go to machine "VIC" from where we want to establish connection to "OBS" using Telnet.
- We will go back to the wireshark on machine "ATT" and will chose a TCP packet that we will be having at the last.
- Noting down the values in front of TCP parameters like :- Src Port, Dst port, Seq and Ack.
- Noting down the Source and Destination IP addresses as well.
- Installing Scapy.

- Entering “Sudo Scapy” to enter Scapy.
- Writing the python script.
- Accessing machine “VIC”.
- Stopping packet capture on Wireshark.



After starting the packet capture on the attacker machine, we will establish the connection between “VIC” and “OBS” using telnet.



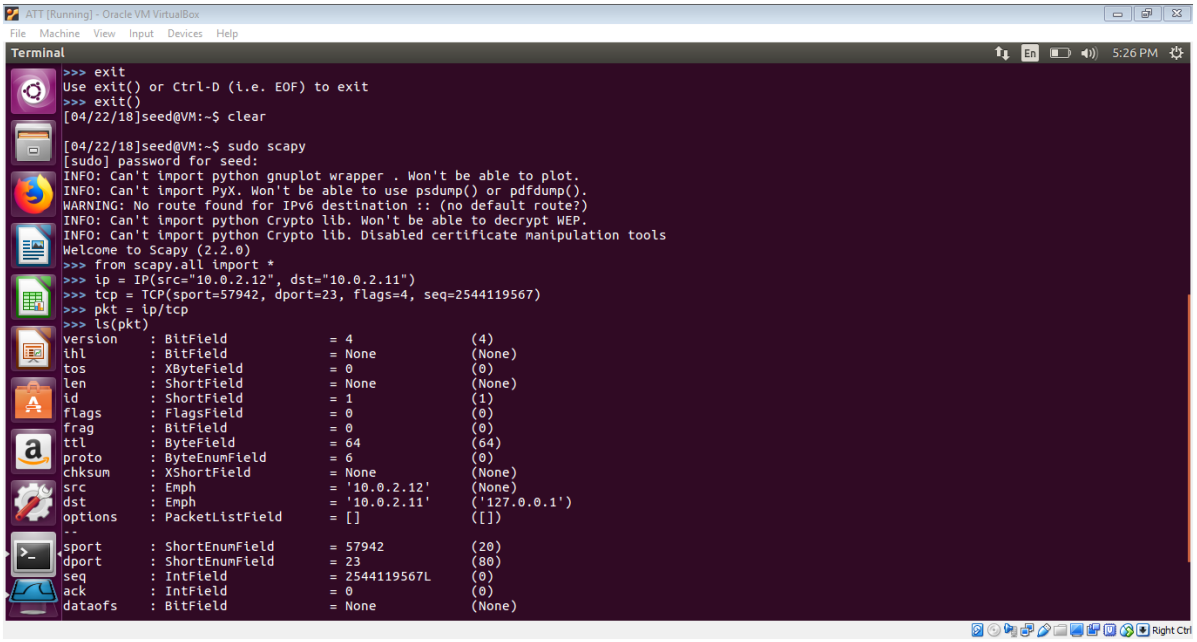
When we analyze a TCP packet we get following parameters: -

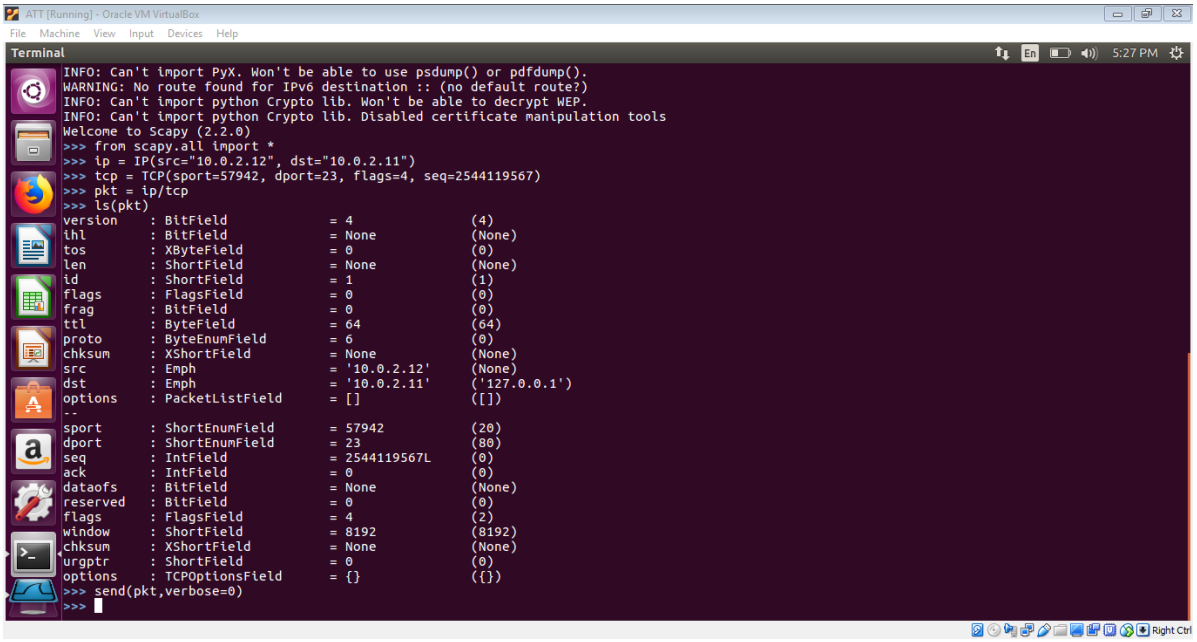
- Source IP = 10.0.2.12
- Destination IP = 10.0.2.11
- Src port = 57942
- Dst port = 23

- Seq = 254119567

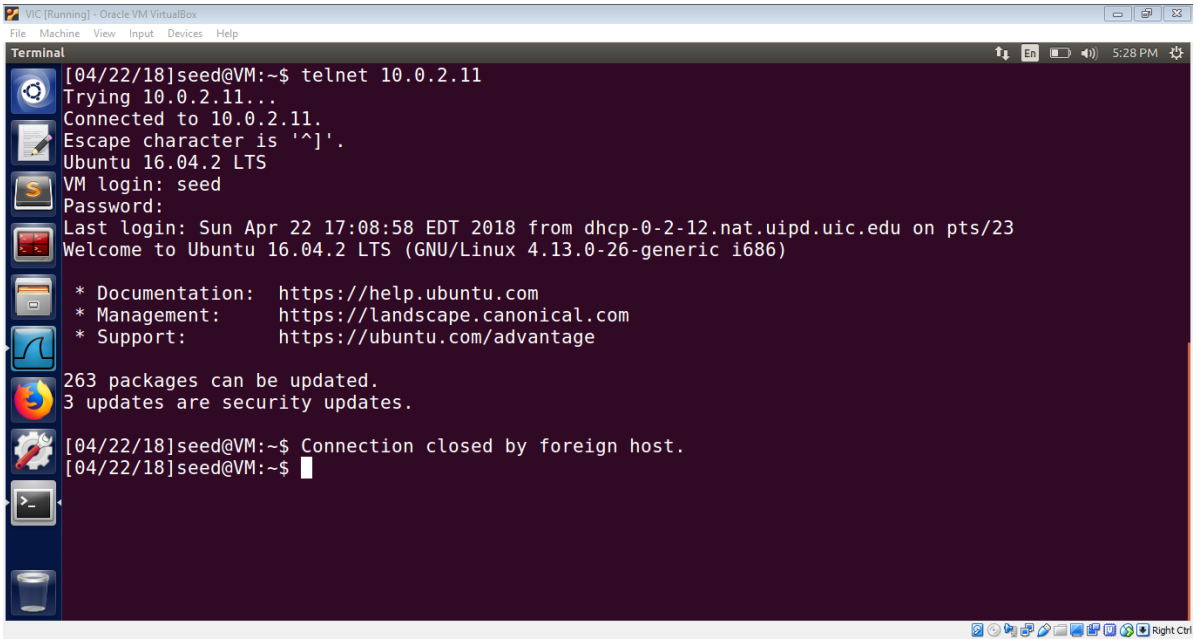
We will set the value of flags = 4

We will put these values in the script mentioned in the question manual and will run the script.





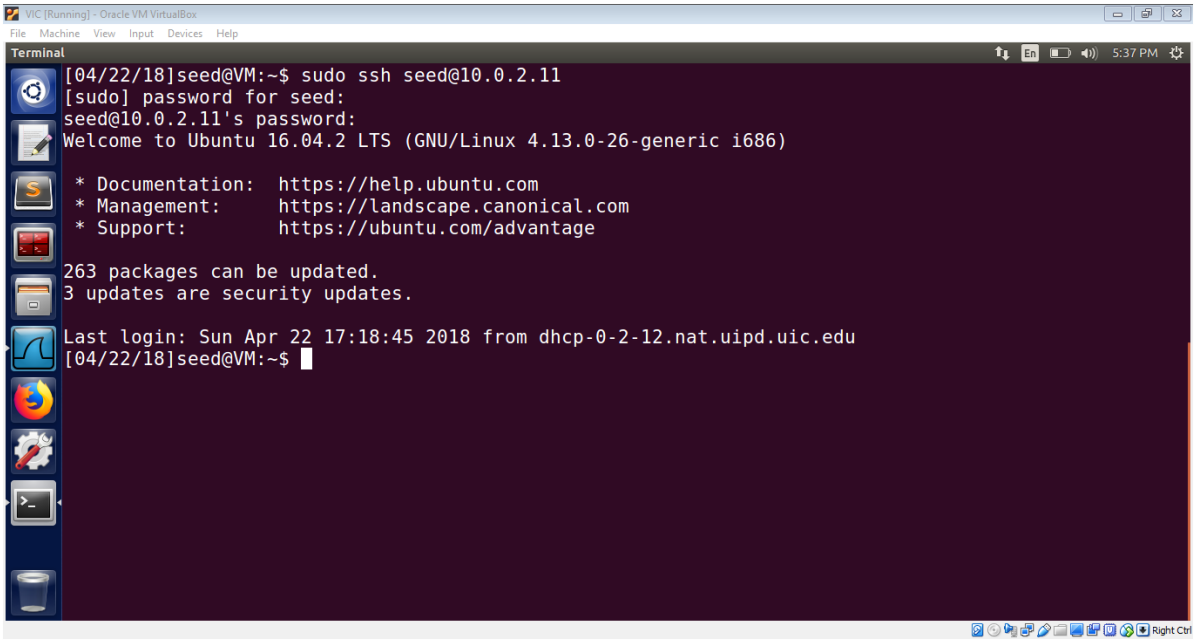
Running the script means we launched the TCP RST attack using scapy.



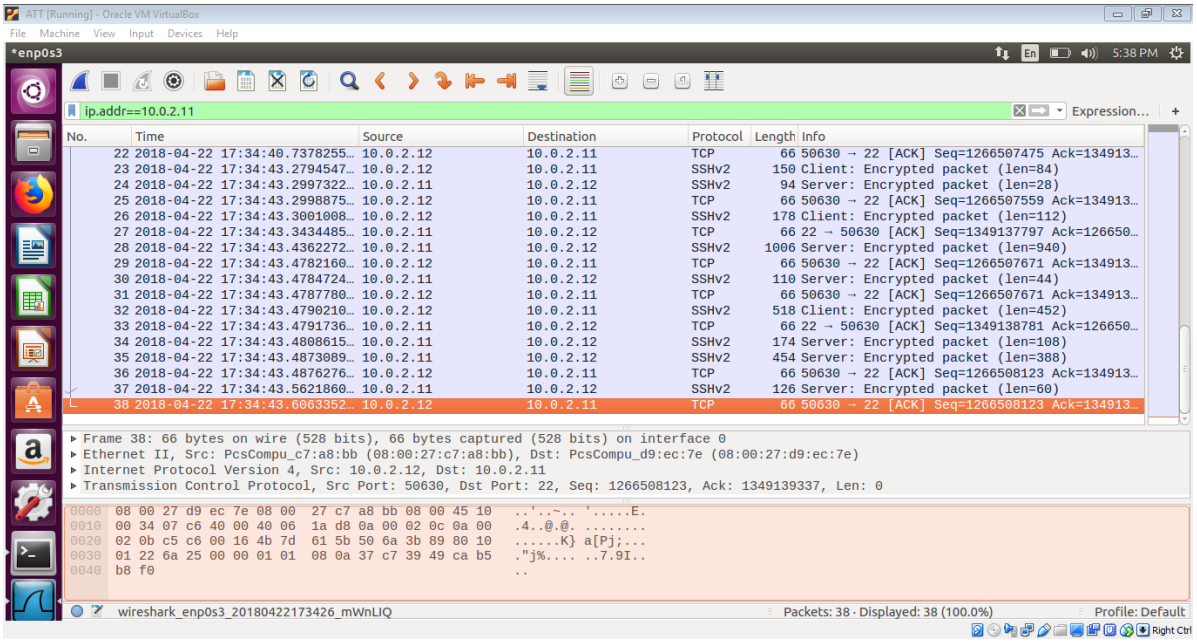
On trying to access the machine "VIC" we see that we get a command stating, "**Connection closed by foreign host**", which means the connection established using Telnet has now been broken. This is what we wanted to achieve.

CONNECTION ESTABLISHMENT BETWEEN "VIC" AND "OBS" THROUGH SSH

We will follow the exact same procedure as above. Just the connection between "VIC" and "OBS" will be done through SSH.



After starting the packet capture on the attacker machine, we will establish the connection between "VIC" and "OBS" using SSH.

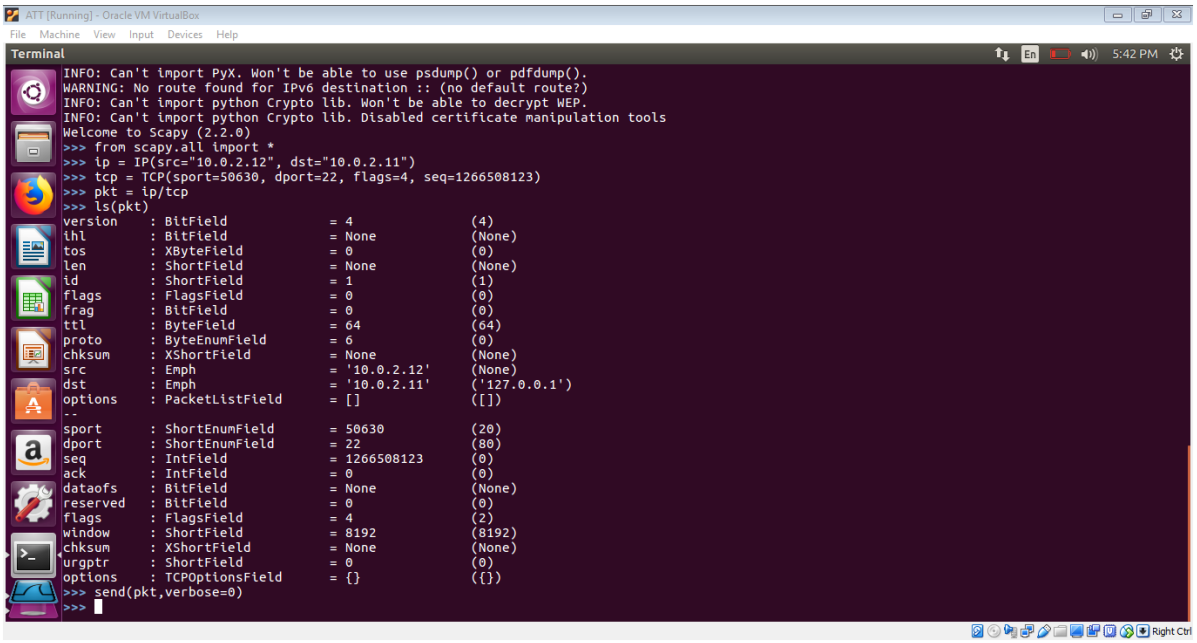


When we analyze a TCP packet we get following parameters: -

- Source IP = 10.0.2.12
- Destination IP = 10.0.2.11
- Src port = 50630
- Dst port = 22
- Seq = 1266508123

We will set the value of flags = 4

We will put these values in the script mentioned in the question manual and will run the script.

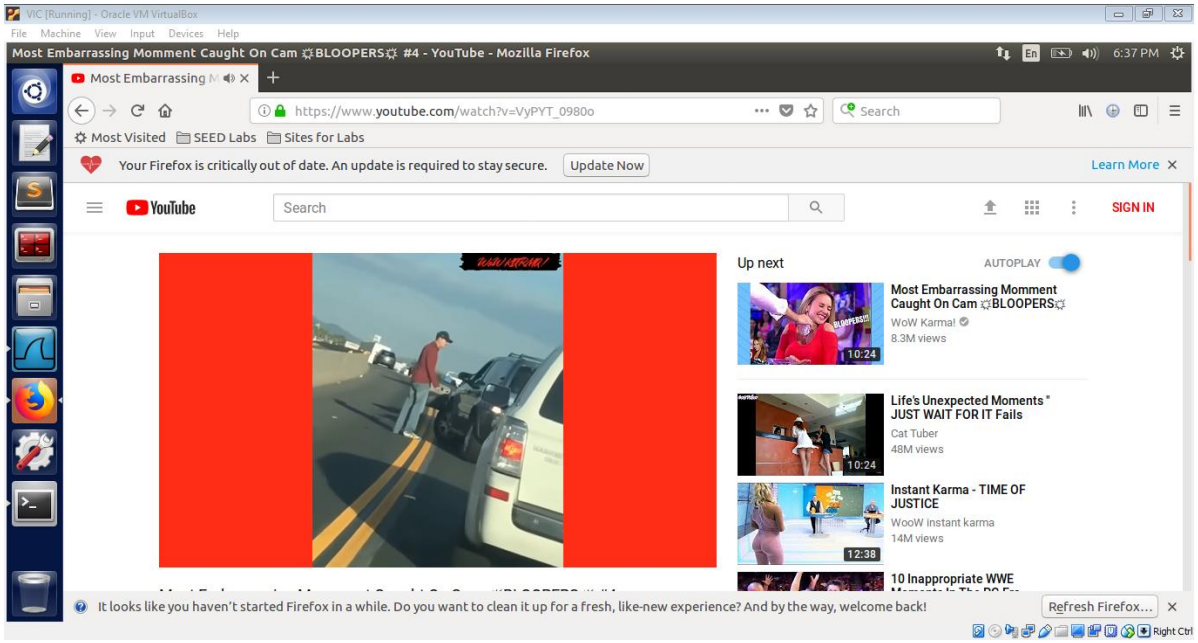


Running the script means we launched the TCP RST attack using scapy.

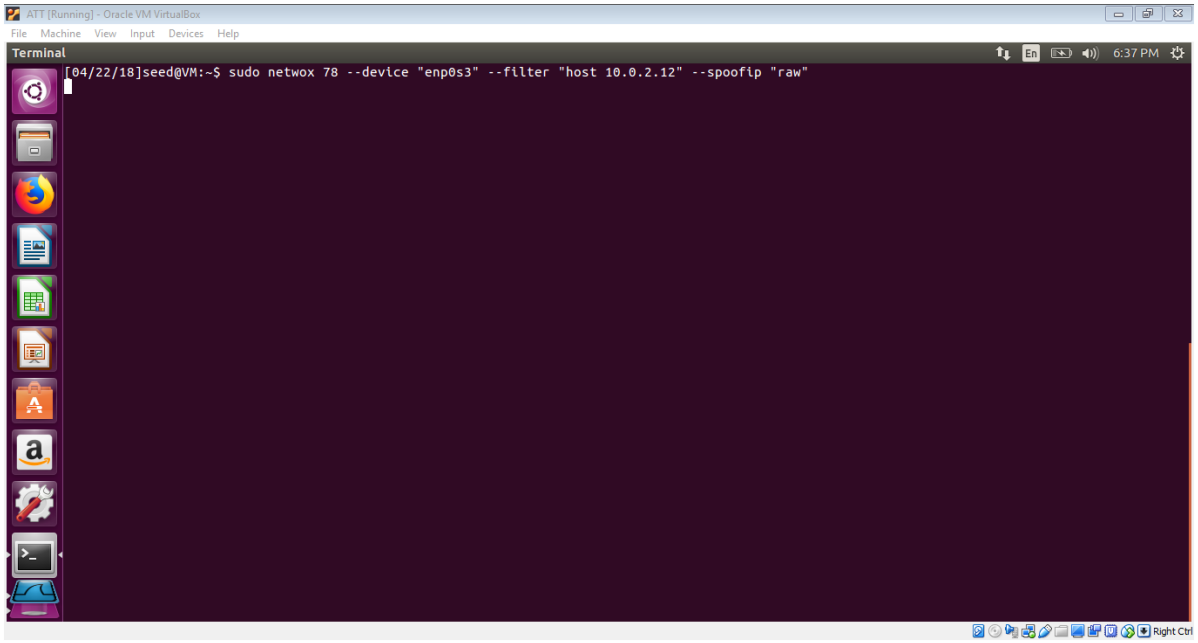
On trying to access the machine "VIC" we see that we get a command stating "**packet_write_wait: Connection to 10.0.2.11 port 22: broken pipe**", which means the connection established using SSH has now been broken. This is what we wanted to achieve.

3.3 TASK 3: TCP RST ATTACKS ON VIDEO STREAMING APPLICATIONS

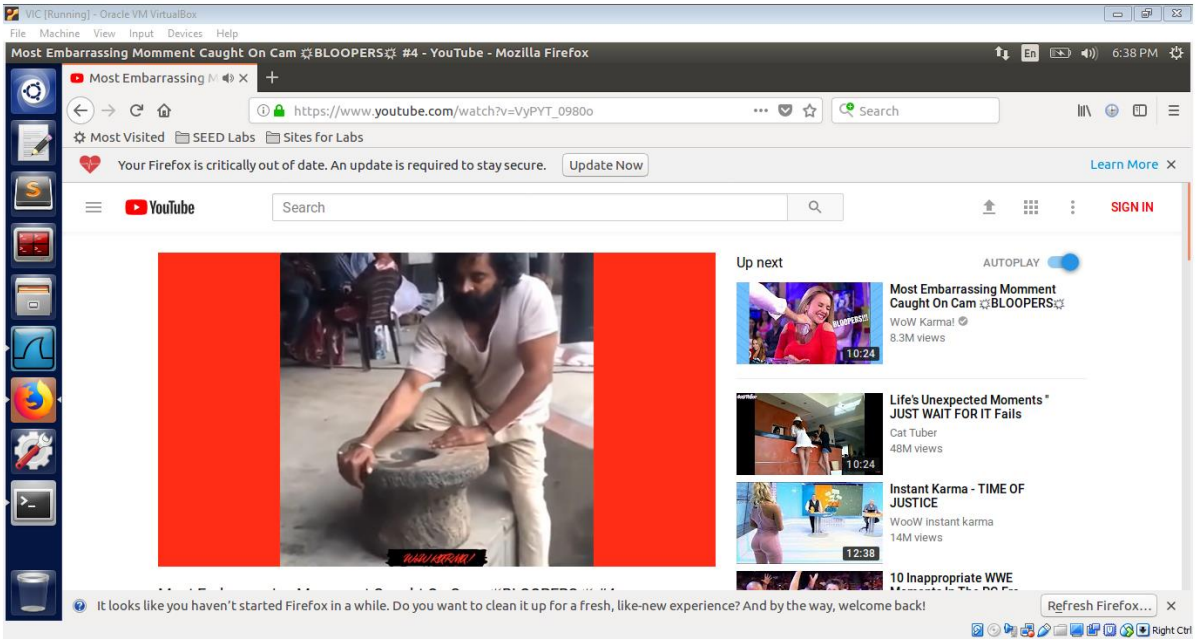
The video streaming web site chosen in this task is **YOUTUBE**. Task is to establish a TCP connection between "VIC" and video streaming web site. We need to launch an attack such that the connection between the machine and the video streaming website is disrupted.

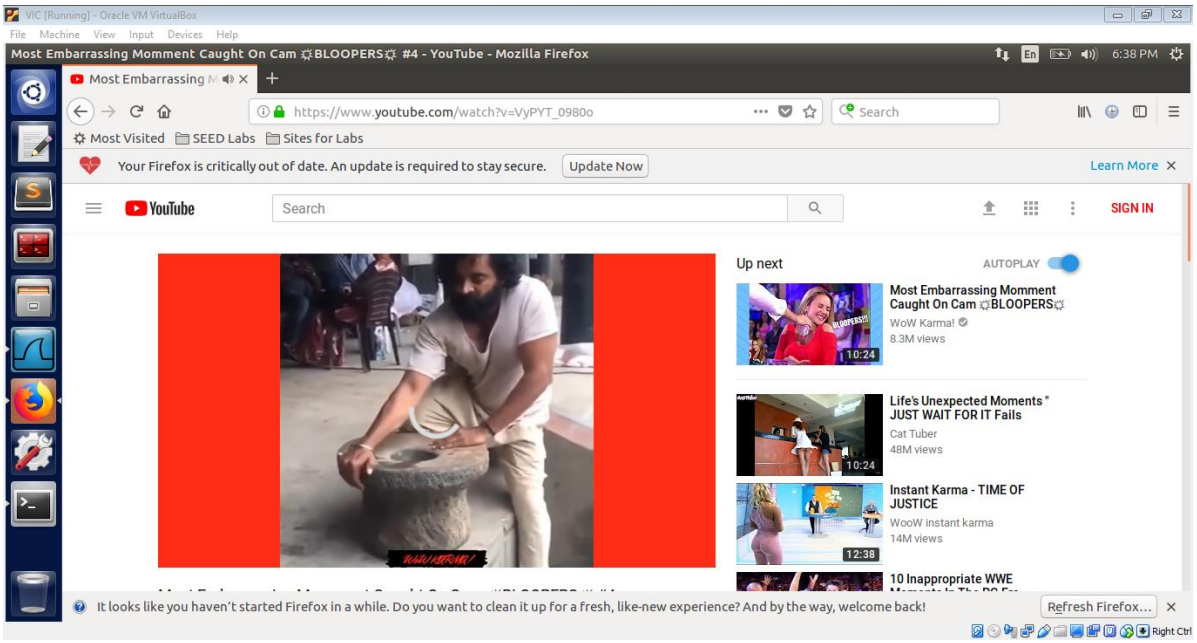


On machine "VIC" we will open the browser and we will open YOUTUBE, our video streaming platform for this task. We will start streaming a video.



While the video is being streamed we will move to machine "ATT", our attacker machine and give the command using netwox tool numbered 78 to launch the TCP RST attack.





After the attack is launched we go back to the machine where our video is streaming, and we will observe that the video will continue playing till the place it buffered and after that **the video will be stopped**. This is because of the attack launched by the attacker machine on the victim machine which leads to disruptions of established TCP connection and hence results in stoppage of video streaming.

If we want to continue streaming the video, then we must stop the attack from machine 'ATT" by using Ctrl+C on machine "ATT'. After doing this, the attack will be stopped, and the video will start streaming.