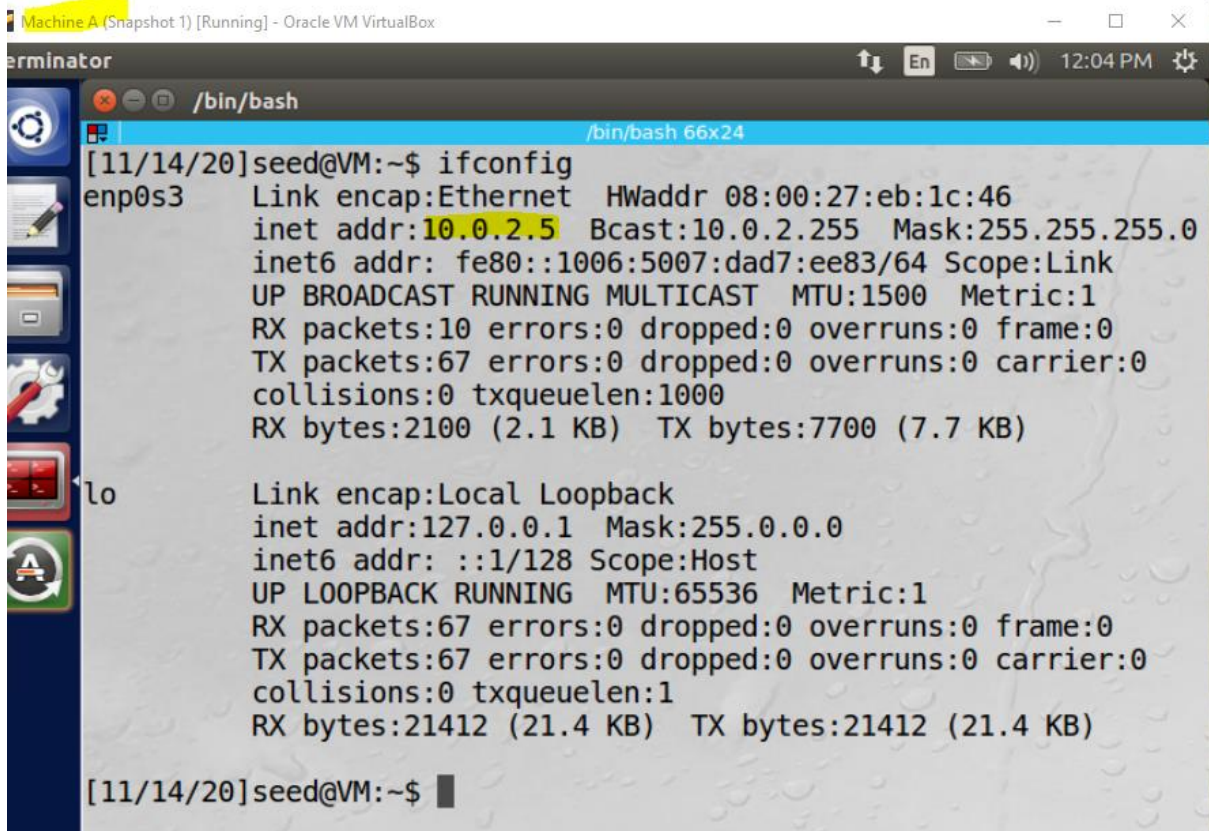


## Lab: Linux Firewall Exploration

### Task 1: Using Firewall

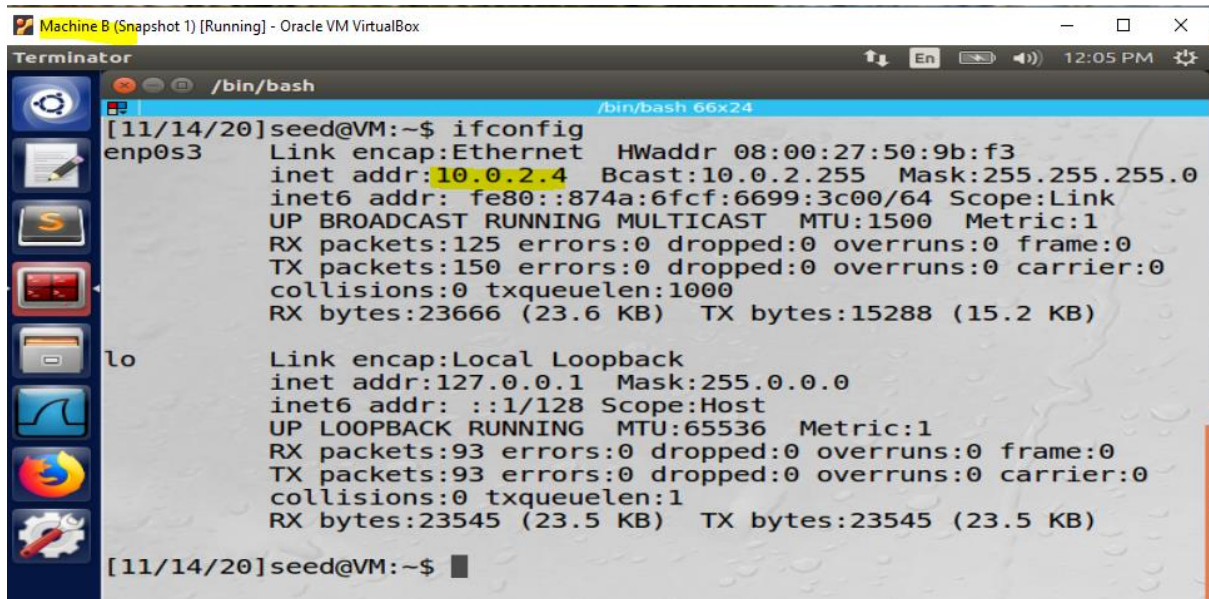
- With the help of ifconfig command, checked the IP's of both Machine A and Machine B



```
[11/14/20]seed@VM:~$ ifconfig
enp0s3  Link encap:Ethernet  HWaddr 08:00:27:eb:1c:46
        inet addr:10.0.2.5  Bcast:10.0.2.255  Mask:255.255.255.0
        inet6 addr: fe80::1006:5007:dad7:ee83/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:10 errors:0 dropped:0 overruns:0 frame:0
        TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:2100 (2.1 KB)  TX bytes:7700 (7.7 KB)

lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:65536  Metric:1
        RX packets:67 errors:0 dropped:0 overruns:0 frame:0
        TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1
        RX bytes:21412 (21.4 KB)  TX bytes:21412 (21.4 KB)

[11/14/20]seed@VM:~$
```



```
[11/14/20]seed@VM:~$ ifconfig
enp0s3  Link encap:Ethernet  HWaddr 08:00:27:50:9b:f3
        inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
        inet6 addr: fe80::874a:6fcf:6699:3c00/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:125 errors:0 dropped:0 overruns:0 frame:0
        TX packets:150 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:23666 (23.6 KB)  TX bytes:15288 (15.2 KB)

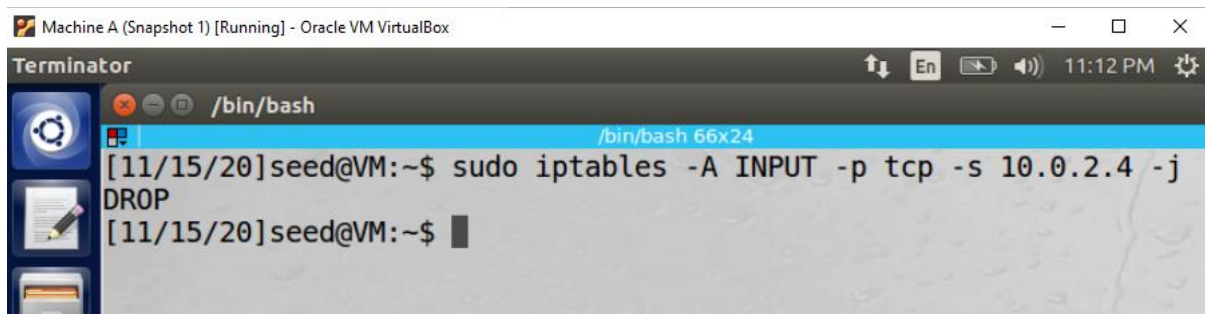
lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:65536  Metric:1
        RX packets:93 errors:0 dropped:0 overruns:0 frame:0
        TX packets:93 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1
        RX bytes:23545 (23.5 KB)  TX bytes:23545 (23.5 KB)

[11/14/20]seed@VM:~$
```

- These two machines can communicate amongst each other. Verified using telnet command.  
telnet 10.0.2.4

- Now to disable the communication between both machines, executed below command on Machine A

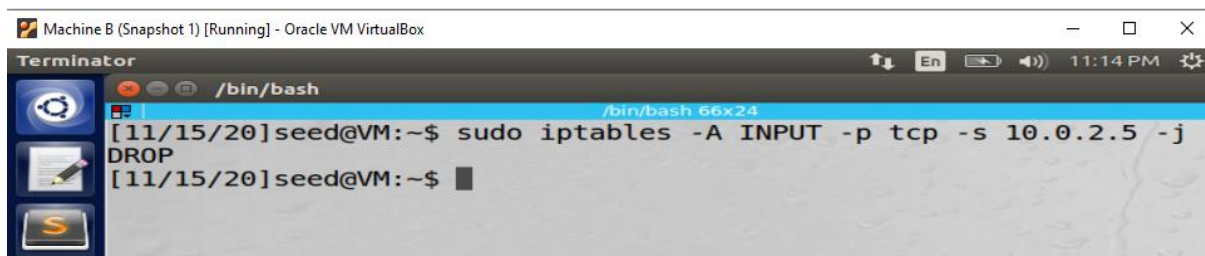
*sudo iptables -A INPUT -p tcp -s 10.0.2.4 -j DROP*



The screenshot shows a terminal window titled "Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox". The terminal is running a shell with the prompt "seed@VM:~\$". The command "sudo iptables -A INPUT -p tcp -s 10.0.2.4 -j DROP" has been entered and executed successfully. The output shows the command being run and the prompt returning.

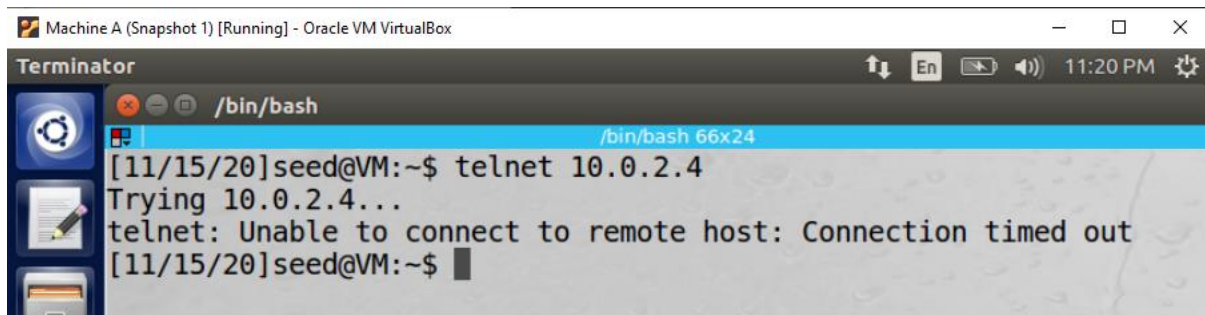
- Then execute iptables command on Machine B (10.0.2.4)

*sudo iptables -A INPUT -p tcp -s 10.0.2.5 -j DROP*



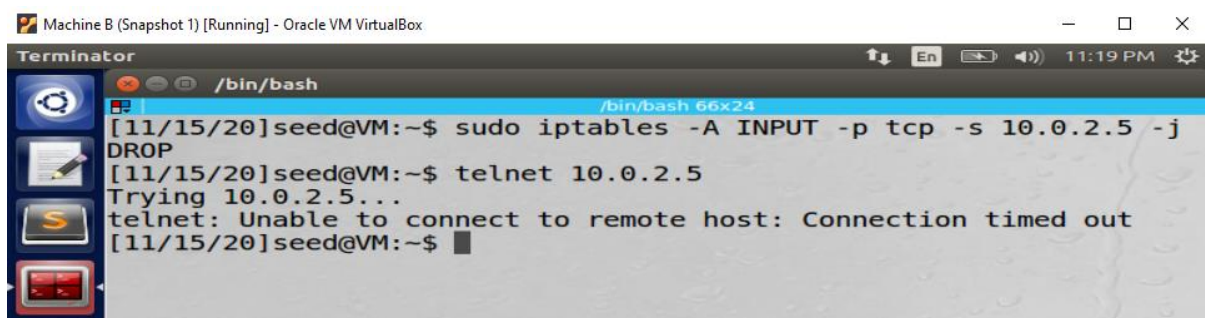
The screenshot shows a terminal window titled "Machine B (Snapshot 1) [Running] - Oracle VM VirtualBox". The terminal is running a shell with the prompt "seed@VM:~\$". The command "sudo iptables -A INPUT -p tcp -s 10.0.2.5 -j DROP" has been entered and executed successfully. The output shows the command being run and the prompt returning.

- Validated Machine A to Machine B connectivity using telnet. Hence, telnet cannot be done from Machine A to Machine B.



The screenshot shows a terminal window titled "Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox". The terminal is running a shell with the prompt "seed@VM:~\$". The command "telnet 10.0.2.4" has been entered. The output shows "Trying 10.0.2.4..." followed by "telnet: Unable to connect to remote host: Connection timed out".

- Similarly, validating connectivity from Machine B to Machine A using telnet. Hence, getting connection timed out.

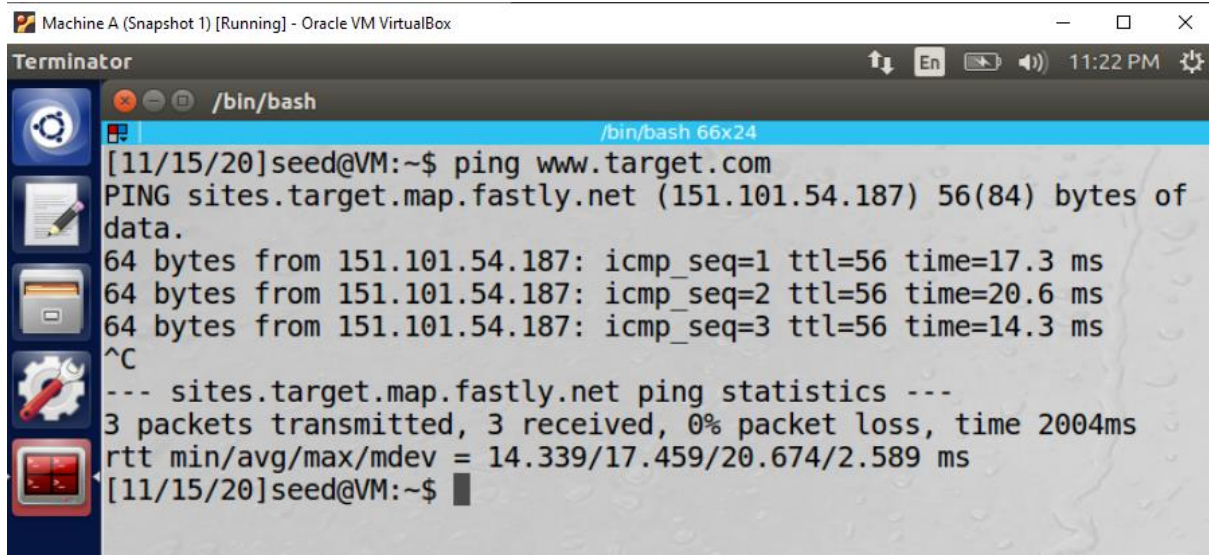


The screenshot shows a terminal window titled "Machine B (Snapshot 1) [Running] - Oracle VM VirtualBox". The terminal is running a shell with the prompt "seed@VM:~\$". The command "telnet 10.0.2.5" has been entered. The output shows "Trying 10.0.2.5..." followed by "telnet: Unable to connect to remote host: Connection timed out".

- Took [www.target.com](http://www.target.com) website to block in Machine A. Validated connectivity of this website using ping command on Machine A.

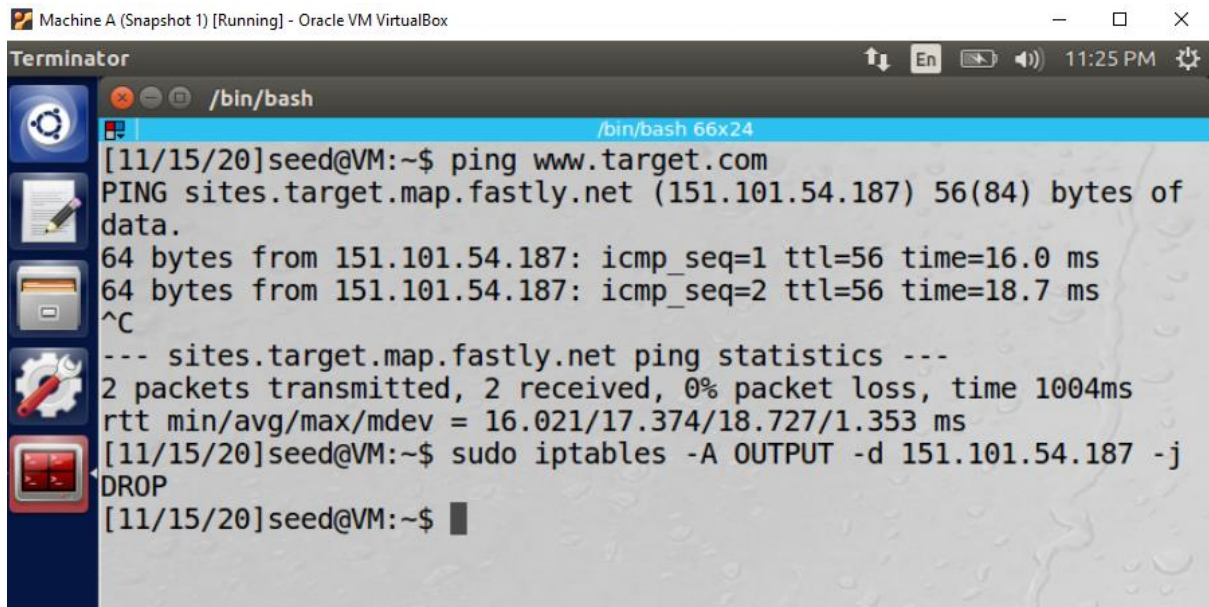
*Ping [www.target.com](http://www.target.com)*





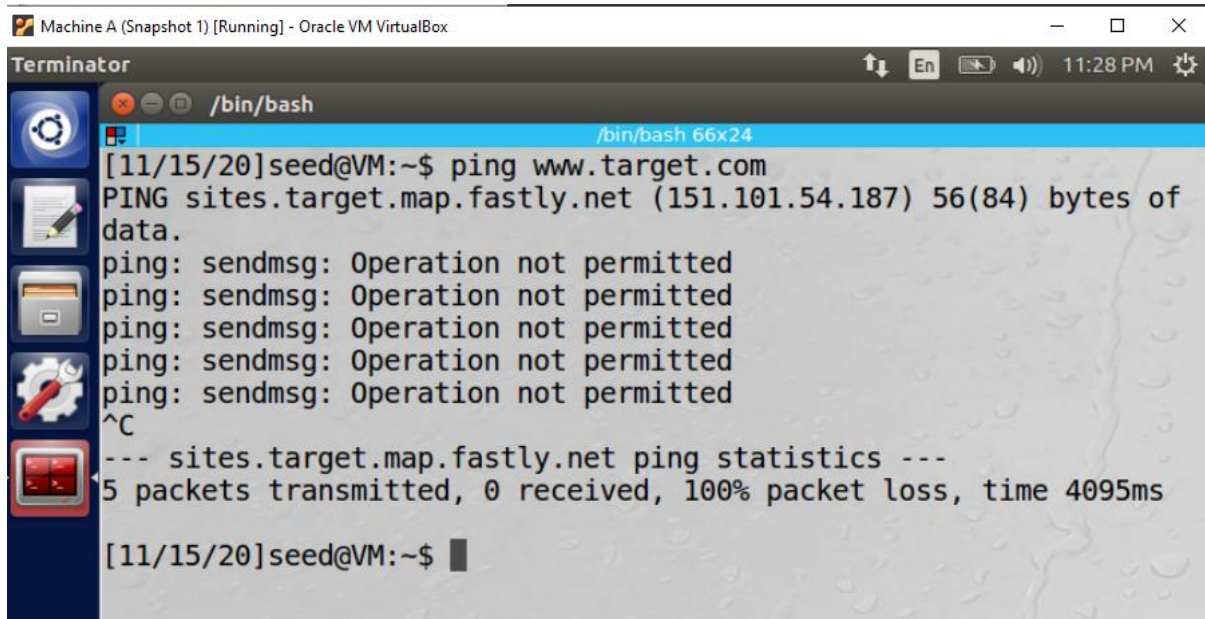
```
Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox
Terminator
/bin/bash
[11/15/20]seed@VM:~$ ping www.target.com
PING sites.target.map.fastly.net (151.101.54.187) 56(84) bytes of data.
64 bytes from 151.101.54.187: icmp_seq=1 ttl=56 time=17.3 ms
64 bytes from 151.101.54.187: icmp_seq=2 ttl=56 time=20.6 ms
64 bytes from 151.101.54.187: icmp_seq=3 ttl=56 time=14.3 ms
^C
--- sites.target.map.fastly.net ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 14.339/17.459/20.674/2.589 ms
[11/15/20]seed@VM:~$
```

- Dropping [www.target.com](http://www.target.com) incoming packets from Machine A using iptables command `sudo iptables -A OUTPUT -d 151.101.54.187 -j DROP`



```
Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox
Terminator
/bin/bash
[11/15/20]seed@VM:~$ ping www.target.com
PING sites.target.map.fastly.net (151.101.54.187) 56(84) bytes of data.
64 bytes from 151.101.54.187: icmp_seq=1 ttl=56 time=16.0 ms
64 bytes from 151.101.54.187: icmp_seq=2 ttl=56 time=18.7 ms
^C
--- sites.target.map.fastly.net ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1004ms
rtt min/avg/max/mdev = 16.021/17.374/18.727/1.353 ms
[11/15/20]seed@VM:~$ sudo iptables -A OUTPUT -d 151.101.54.187 -j DROP
[11/15/20]seed@VM:~$
```

- Again ping [www.target.com](http://www.target.com) from Machine A. Now, the website is blocked on Machine A and gives 100% packet loss, as shown in below snapshot:



## Task 2: Implementing a Simple Firewall

- Using LKM and Netfilter to implement the packet filtering module.

```
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/netfilter.h>
#include <linux/netfilter_ipv4.h>
#include <linux/ip.h>
#include <linux/tcp.h>
#include <linux/inet.h>

/* This is the structure we shall use to register our function */
static struct nf_hook_ops nfho;

/* This is the hook function itself */
unsigned int hook_func(void *priv, struct sk_buff *skb, const struct nf_hook_state *state) {
    struct iphdr *iph;
    struct tcphdr *tcph;

    iph = ip_hdr(skb);
    tcph = (void *)iph+iph->ihl*4;

    if(iph->saddr == in_aton("10.0.2.5") && iph->daddr == in_aton("10.0.2.6")) {
        printk("Dropping packet from %d.%d.%d.%d to %d.%d.%d.%d", ((unsigned char *)&iph->saddr)[0], ((unsigned char *)&iph->saddr)[1], ((unsigned char *)&iph->saddr)[2], ((unsigned char *)&iph->saddr)[3], ((unsigned char *)&iph->daddr)[0], ((unsigned char *)&iph->daddr)[1], ((unsigned char *)&iph->daddr)[2], ((unsigned char *)&iph->daddr)[3]);
        return NF_DROP;
    }
    else {
        return NF_ACCEPT;
    }
}

/* Initialization routine */
int init_module() {
    /* Fill in our hook structure */
    nfho.hook = hook_func; /* Handler function */
    nfho.hooknum = NF_INET_PRE_ROUTING; /* First hook for IPv4 */
    nfho.pf = PF_INET;
    nfho.priority = NF_IP_PRI_FIRST; /* Make our function first */
    nf_register_hook(&nfho);
    return 0;
}

/* Cleanup routine */
void cleanup_module() {
    nf_unregister_hook(&nfho);
}
```

- Make file for LKM and Netfilter

```
obj-m += filter.o
all:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules

clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

### Task 3: Evading Egress Filtering

- Validated connectivity of [www.syr.edu](http://www.syr.edu) from Machine-A and then executed command to block this website.  
`sudo iptables -A INPUT -s 128.230.18.200 -j DROP`
- Observed that the website [www.syr.edu](http://www.syr.edu) blocked from Machine A as shown in snapshot (100% packet loss).

Note: Since [www.facebook.com](http://www.facebook.com) has many IP's. Therefore, I have used [www.syr.edu](http://www.syr.edu).

The screenshot shows a terminal window titled "Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox". The terminal output is as follows:

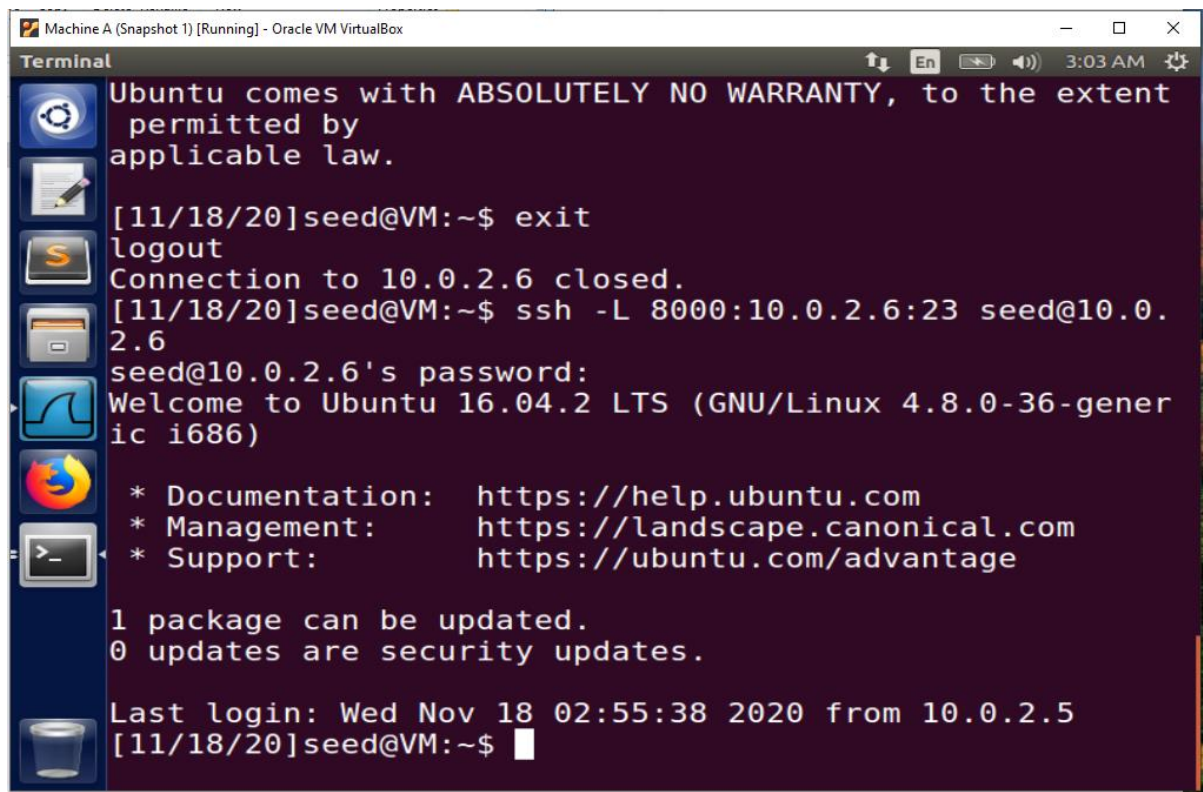
```
39 time=100 ms
64 bytes from syr.edu (128.230.18.200): icmp_seq=2 ttl=
39 time=93.9 ms
64 bytes from syr.edu (128.230.18.200): icmp_seq=3 ttl=
39 time=102 ms
64 bytes from syr.edu (128.230.18.200): icmp_seq=4 ttl=
39 time=105 ms
^C
--- syr.edu ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time
3006ms
rtt min/avg/max/mdev = 93.907/100.597/105.407/4.210 ms
[11/18/20]seed@VM:~$ sudo iptables -A INPUT -s 128.230.
18.200 -j DROP
[11/18/20]seed@VM:~$ ping www.syr.edu
PING syr.edu (128.230.18.200) 56(84) bytes of data.
^C
--- syr.edu ping statistics ---
11 packets transmitted, 0 received, 100% packet loss, t
ime 10221ms
[11/18/20]seed@VM:~$
```

### Task 3.a: Telnet to Machine B through the firewall

- Establishing SSH tunnel between the localhost (port 8000) and machine C (port 22) when packets come out of C's end, it will be forwarded to Machine C's port 23 (telnet port)  
`ssh -L 8000:10.0.2.6:23 seed@10.0.2.6`

Note: Since Machine B vm got crashed. So, I have taken Machine A and Machine C





Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox

Terminal

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

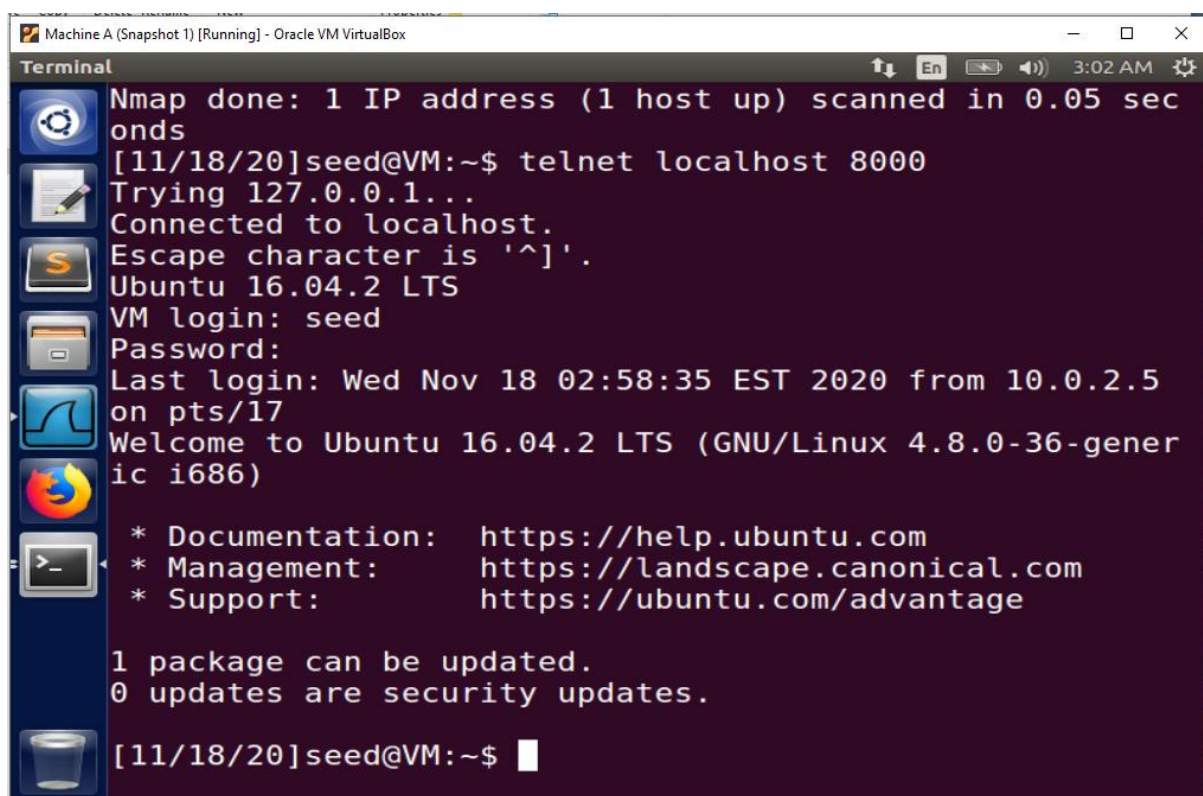
[11/18/20]seed@VM:~\$ exit  
logout  
Connection to 10.0.2.6 closed.  
[11/18/20]seed@VM:~\$ ssh -L 8000:10.0.2.6:23 seed@10.0.2.6  
seed@10.0.2.6's password:  
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

\* 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: Wed Nov 18 02:55:38 2020 from 10.0.2.5  
[11/18/20]seed@VM:~\$

- Now from Machine A, connecting to Machine C using tunnel  
*telnet localhost 8000*



Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox

Terminal

Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds

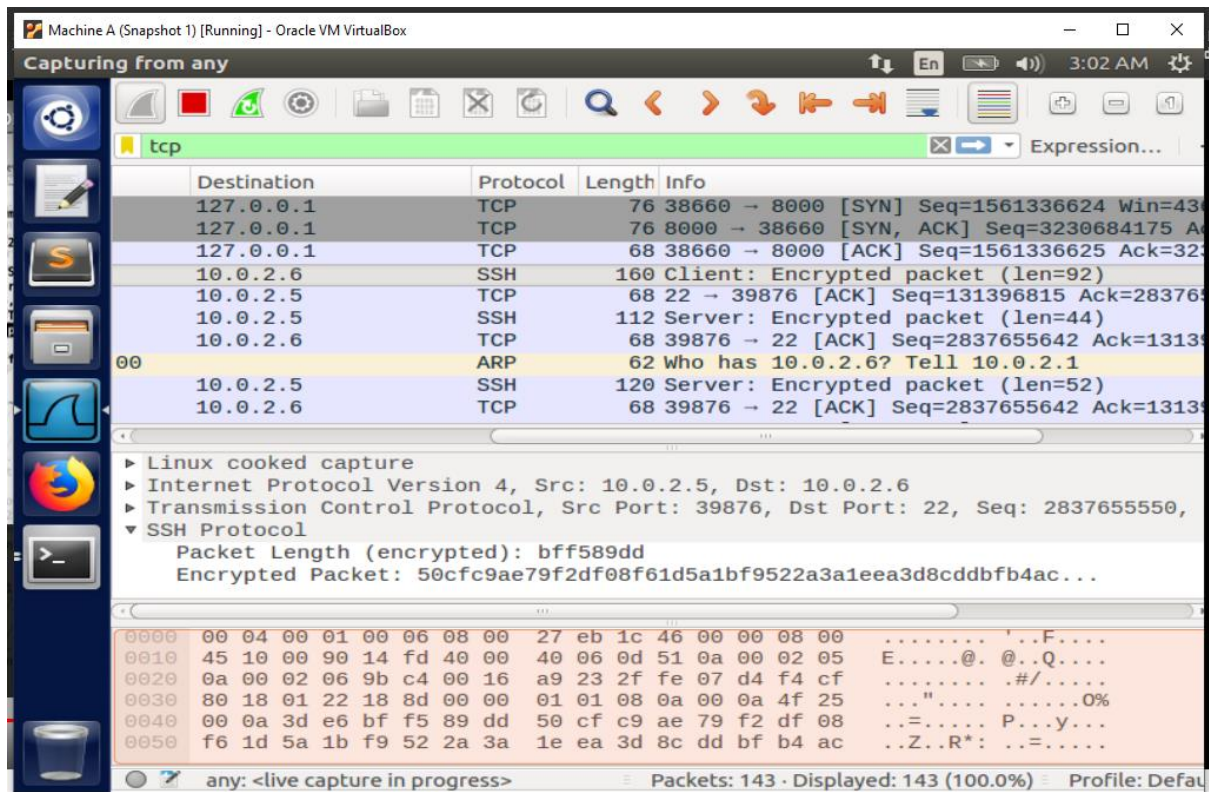
[11/18/20]seed@VM:~\$ telnet localhost 8000  
Trying 127.0.0.1...  
Connected to localhost.  
Escape character is '^]'.  
Ubuntu 16.04.2 LTS  
VM login: seed  
Password:  
Last login: Wed Nov 18 02:58:35 EST 2020 from 10.0.2.5 on pts/17  
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

\* 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.

[11/18/20]seed@VM:~\$

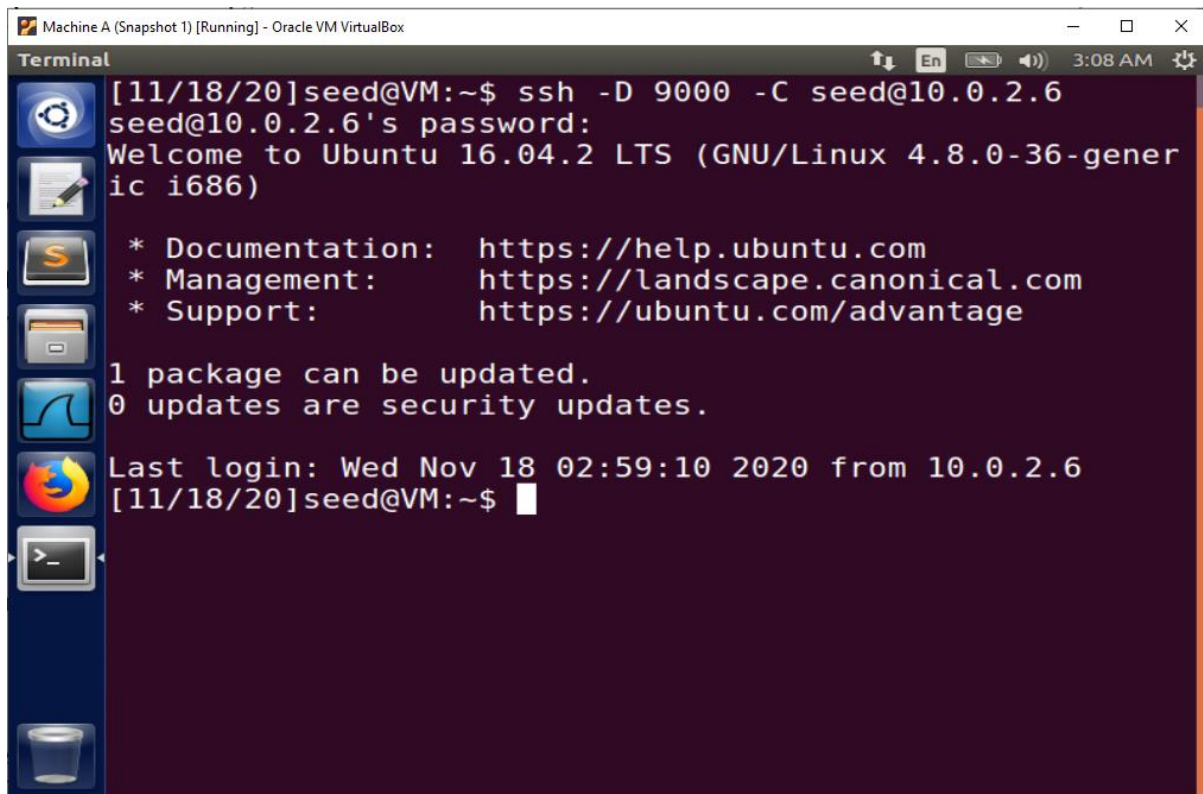
- In the wireshark of Machine A, observe SSH Encrypted Packet. Refer below snapshot:



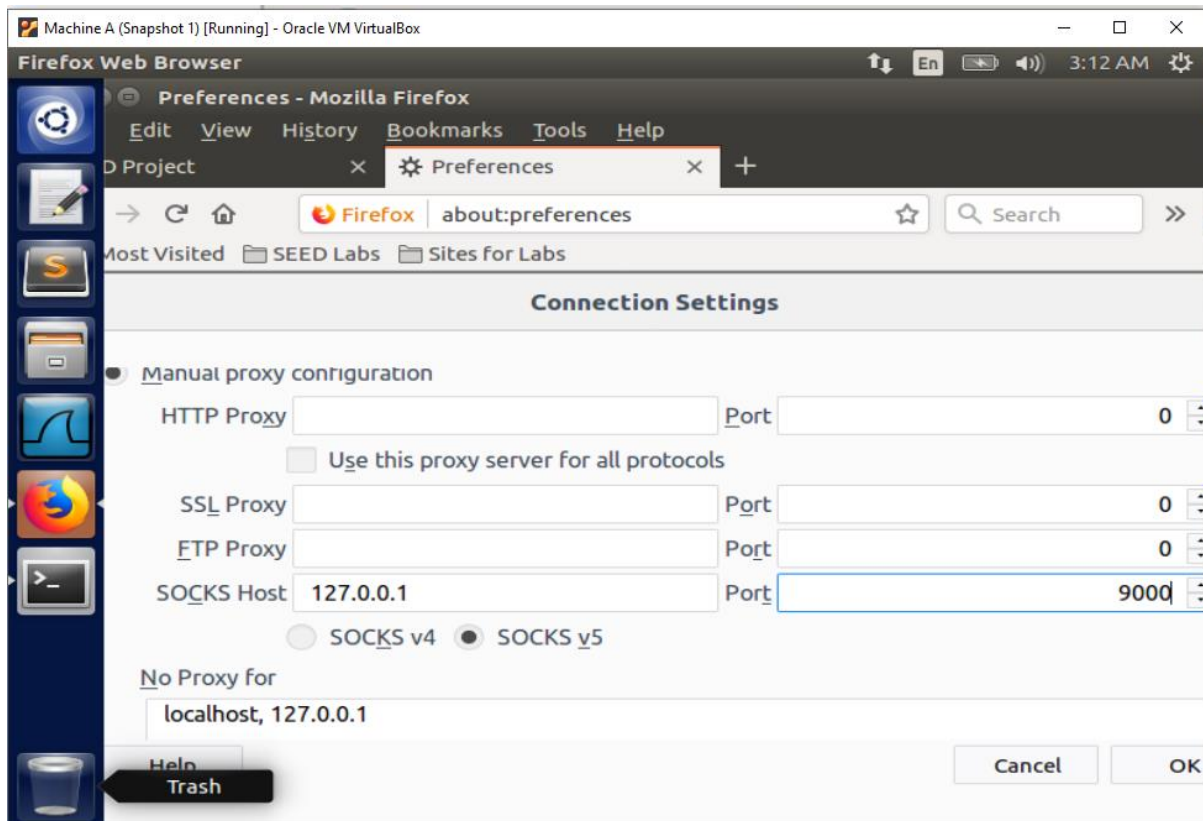
### Task 3.b: Connect to Facebook using SSH Tunnel

- Using -D in the command to dynamically forward the packet based on the destination information of the packet.

`ssh -D 9000 -C seed@10.0.2.6`



- To apply dynamic forwarding to port in Firefox, using SOCKS proxy which can be selected from connection setting



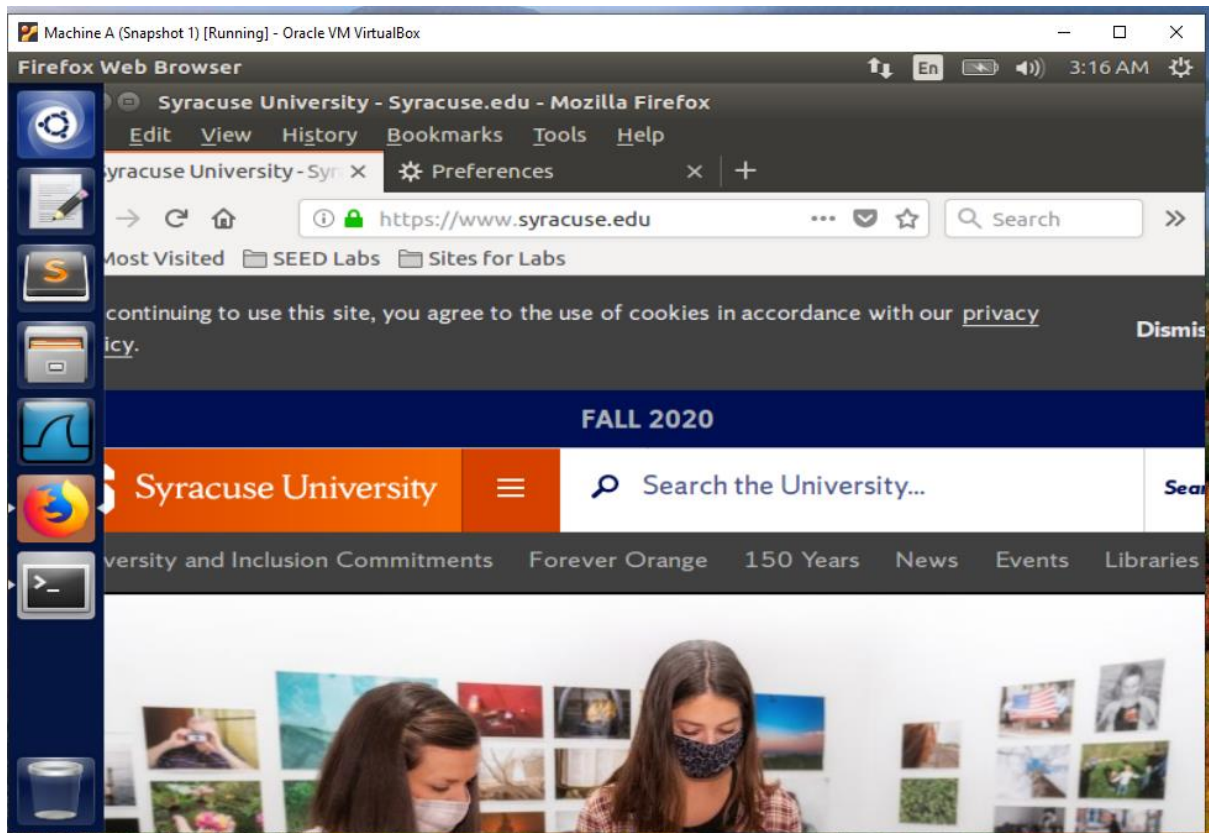
Note: Since Facebook may be using multiple Ips, I have done this task on [www.syr.edu](http://www.syr.edu). This is the same as recommended for Task 3a



- i. **Run Firefox and go visit the Facebook page. Can you see the Facebook page? Please describe your observation.**

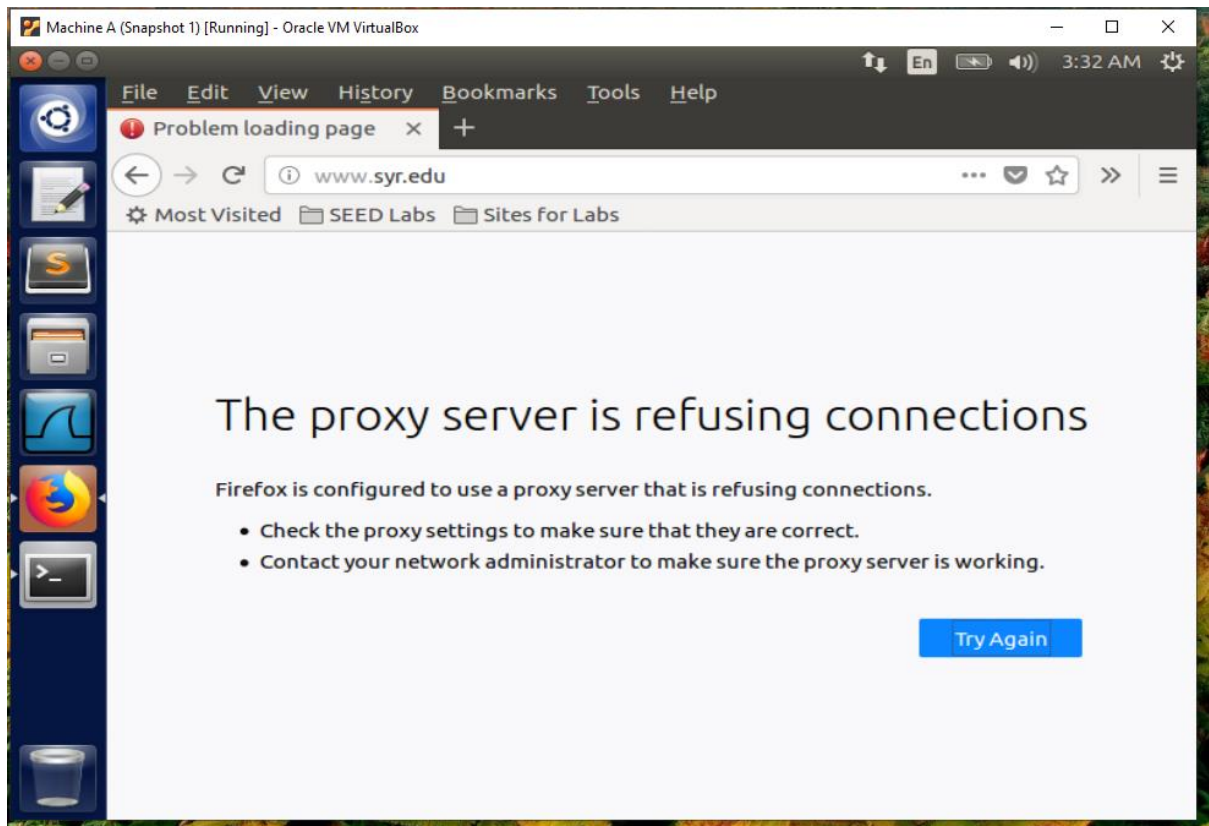
Since I am using [www.syr.edu](http://www.syr.edu) website. So, I am doing this task on this website.

Yes, I am able to see the [www.syr.edu](http://www.syr.edu) page loaded.



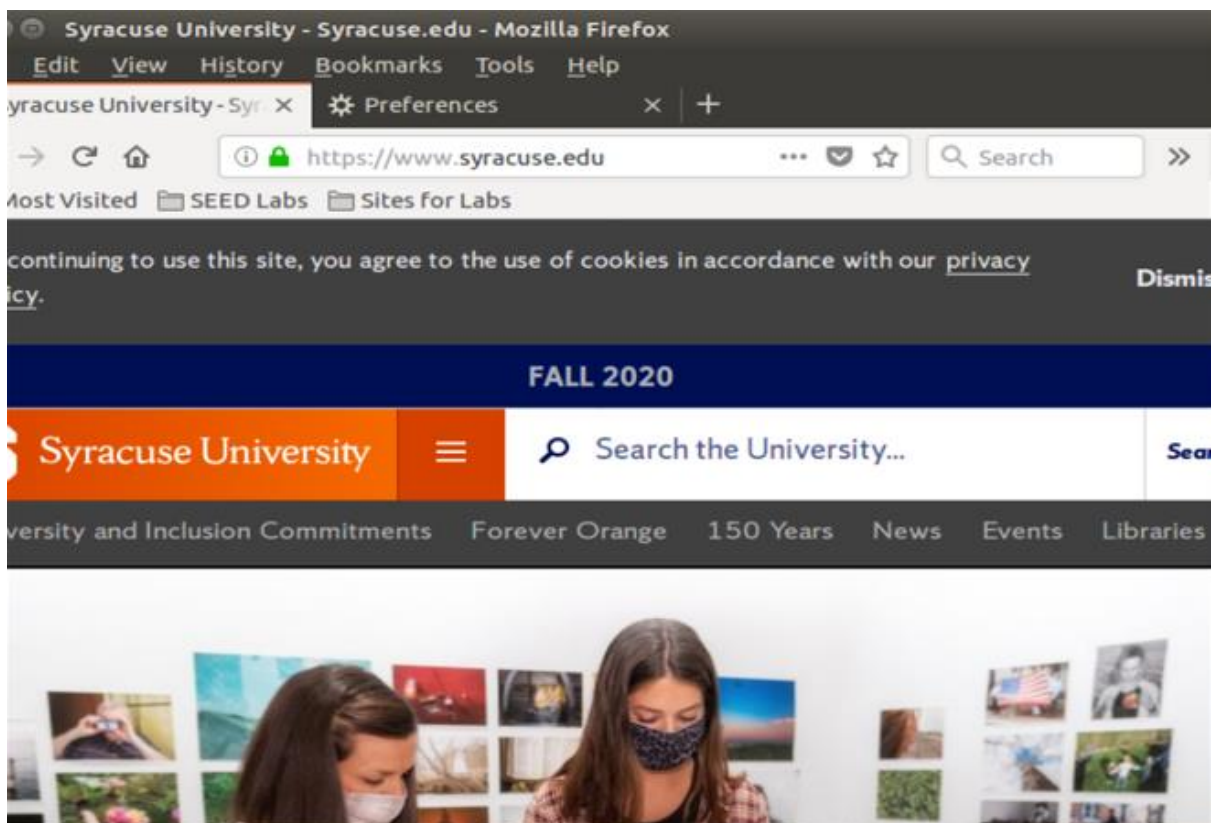
- ii **After you get the facebook page, break the SSH tunnel, clear the Firefox cache, and try the connection again. Please describe your observation.**

After breaking the SH tunnel ad clearing the firefox browser cache and history. I observed that [www.syr.edu](http://www.syr.edu) was not able to load. It gave connection refused. Refer below snaphot.



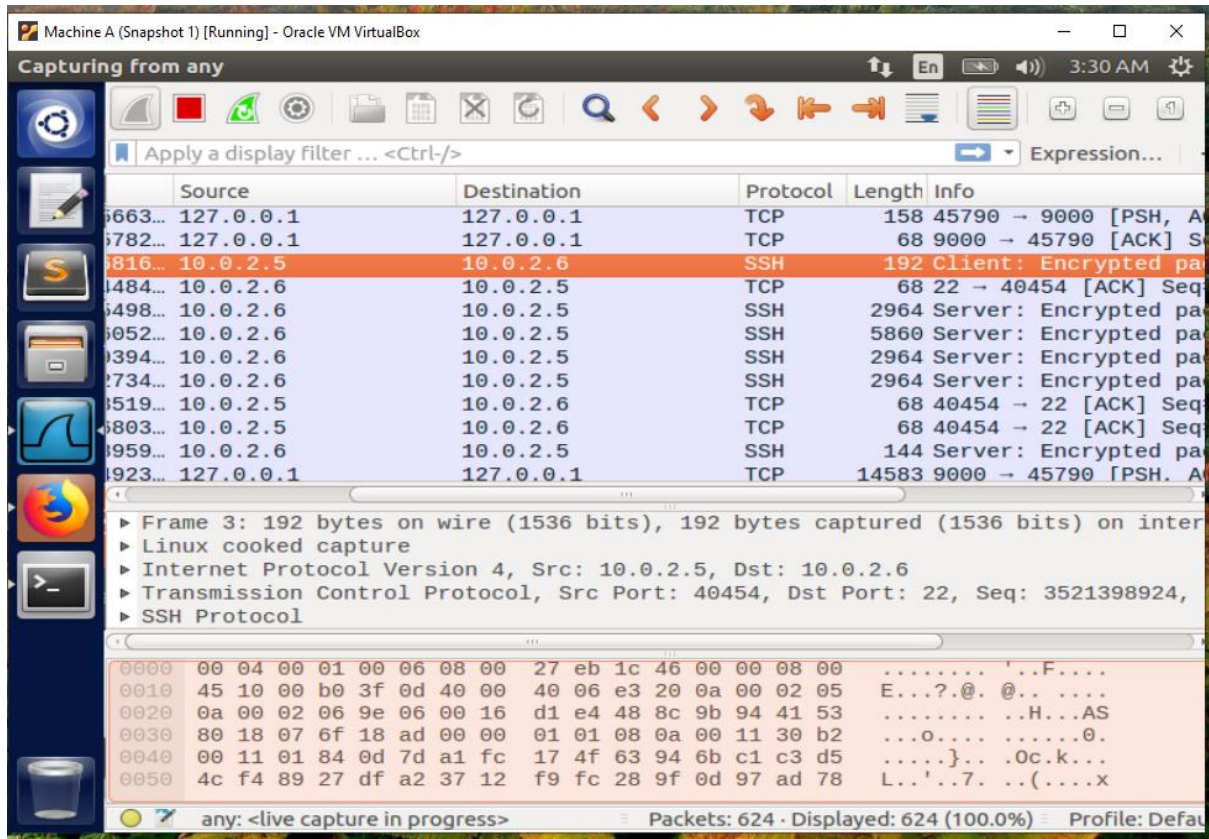
iii **Establish the SSH tunnel again and connect to Facebook. Describe your observation.**

Again, establishing the SSH tunnel connection to [www.syr.edu](http://www.syr.edu), I was able to load the page.



iv Please explain what you have observed, especially on why the SSH tunnel can help bypass the egress filtering. You should use Wireshark to see what exactly is happening on the wire. Please describe your observations and explain them using the packets that you have captured.

I observed whenever the SSH tunnel connection establish, then I was able to connect to [www.syr.edu](http://www.syr.edu) and when the SSH tunnel connection is lost, then I was not able to connect to this website. The packets sent from Machine A (source) to Machine B (destination) were encrypted as they were sent using SSH protocol. Refer below wireshark screenshot.

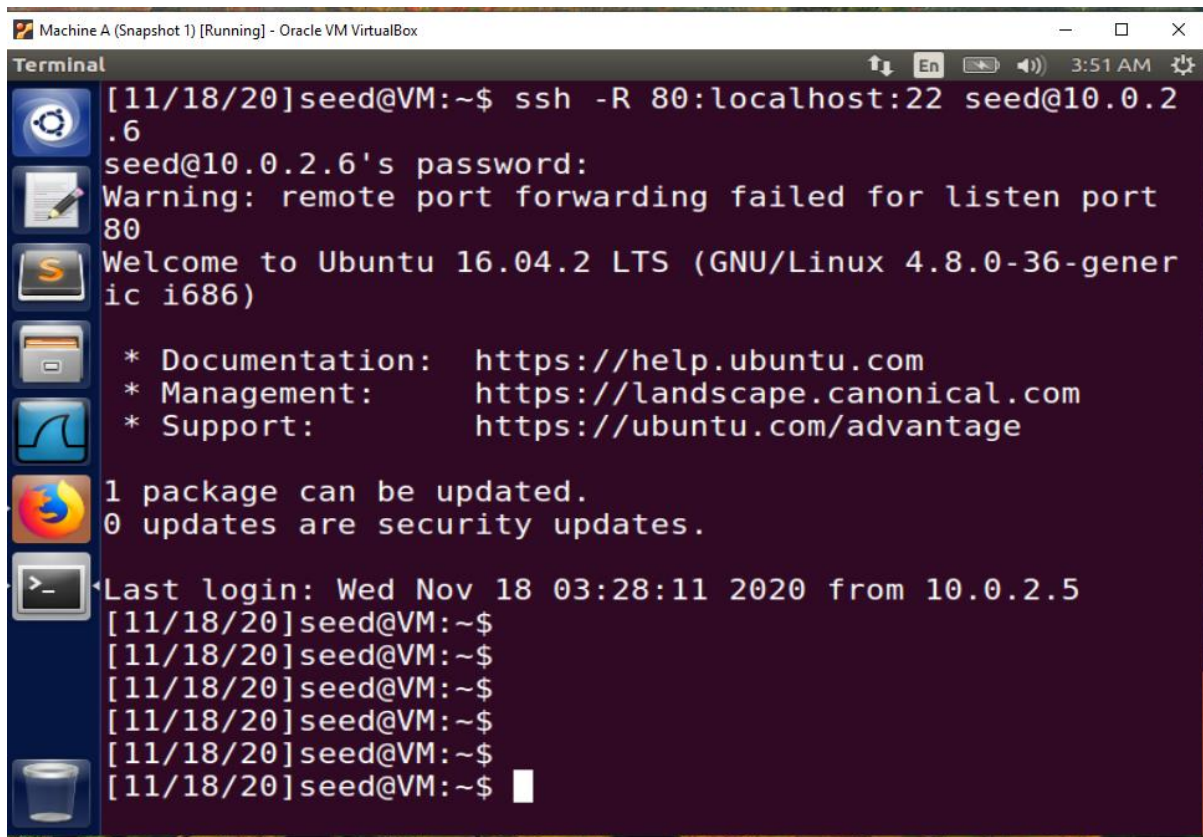


## Task 4: Evading Ingress Filtering

Applying reverse SSH on Machine A where blocking Machine B from accessing its port 80(web server) and 22 (SSH server). Executed below reverse SSH command on Machine A.

`ssh -R 80:localhost:22 seed@10.0.2.6`





```
Machine A (Snapshot 1) [Running] - Oracle VM VirtualBox
Terminal
[11/18/20]seed@VM:~$ ssh -R 80:localhost:22 seed@10.0.2.6
seed@10.0.2.6's password:
Warning: remote port forwarding failed for listen port 80
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.

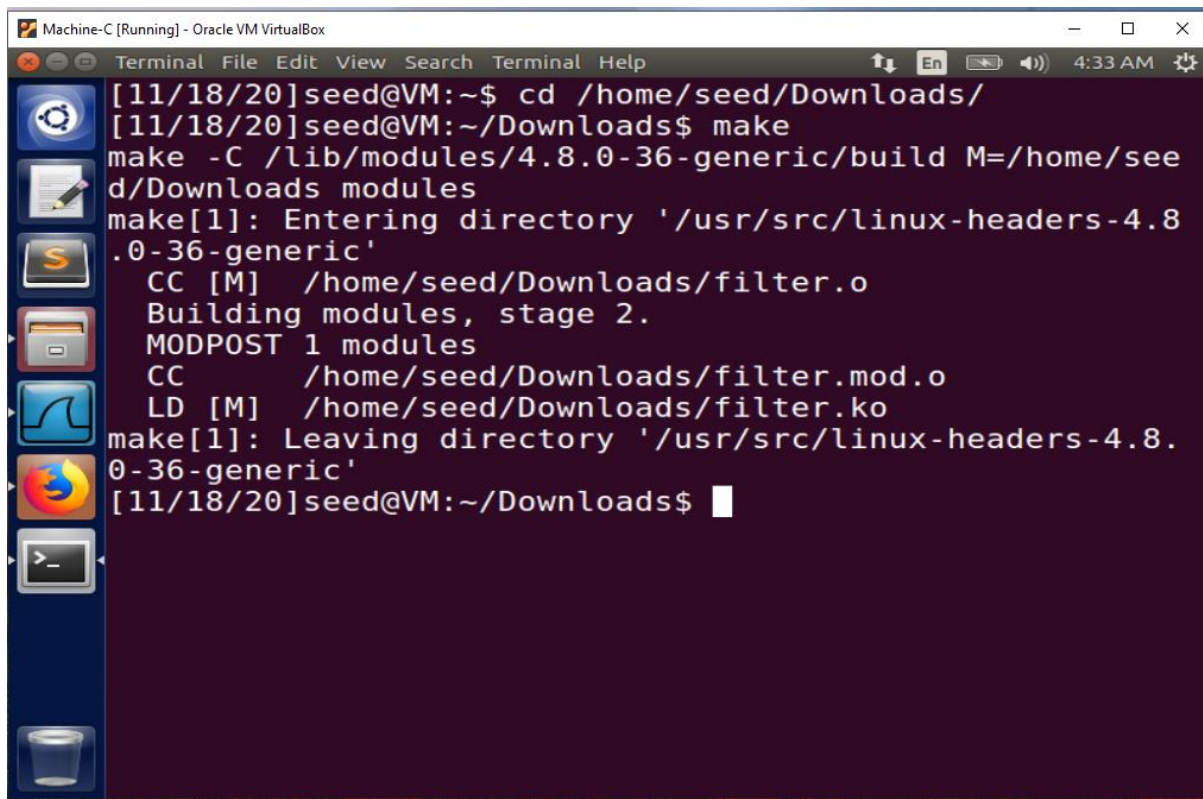
Last login: Wed Nov 18 03:28:11 2020 from 10.0.2.5
[11/18/20]seed@VM:~$
[11/18/20]seed@VM:~$
[11/18/20]seed@VM:~$
[11/18/20]seed@VM:~$
[11/18/20]seed@VM:~$
[11/18/20]seed@VM:~$
```

### 3.1 Loadable Kernel Module

- Loading make file as used in Task-2

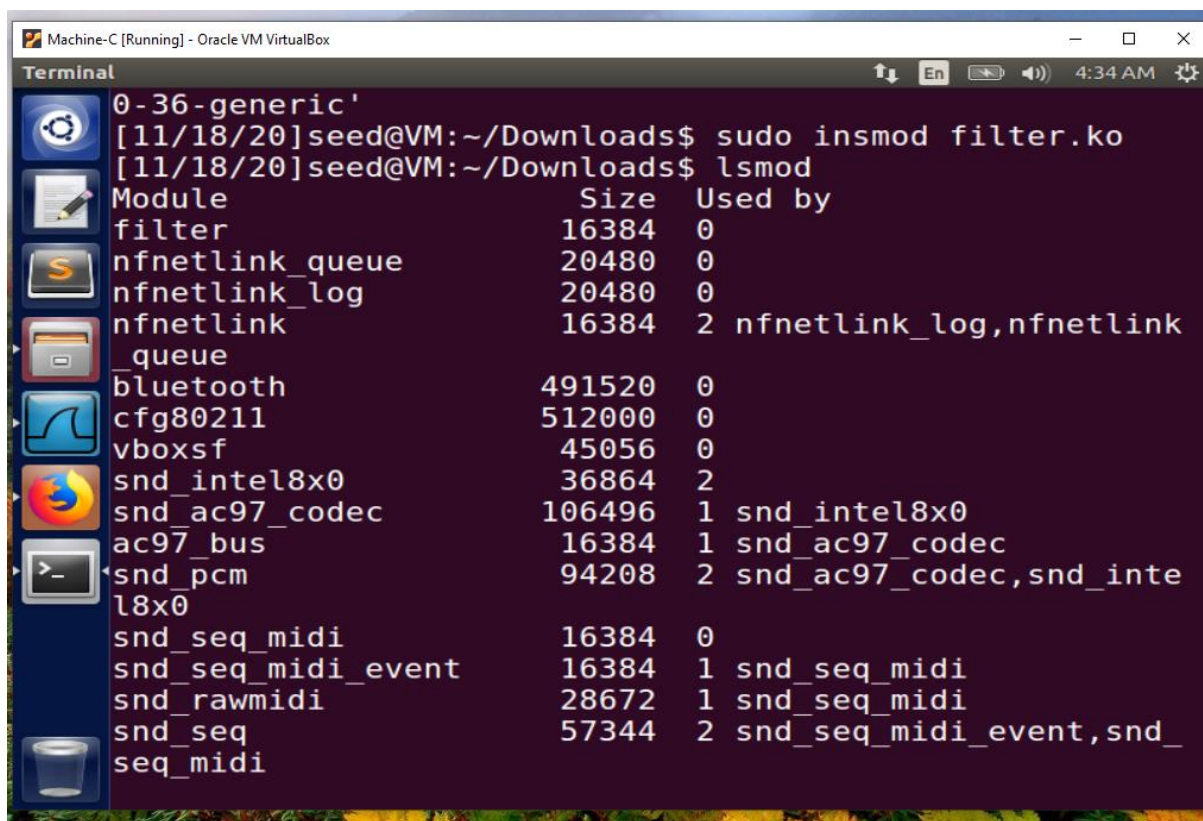
```
obj-m += filter.o
all:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules

clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```



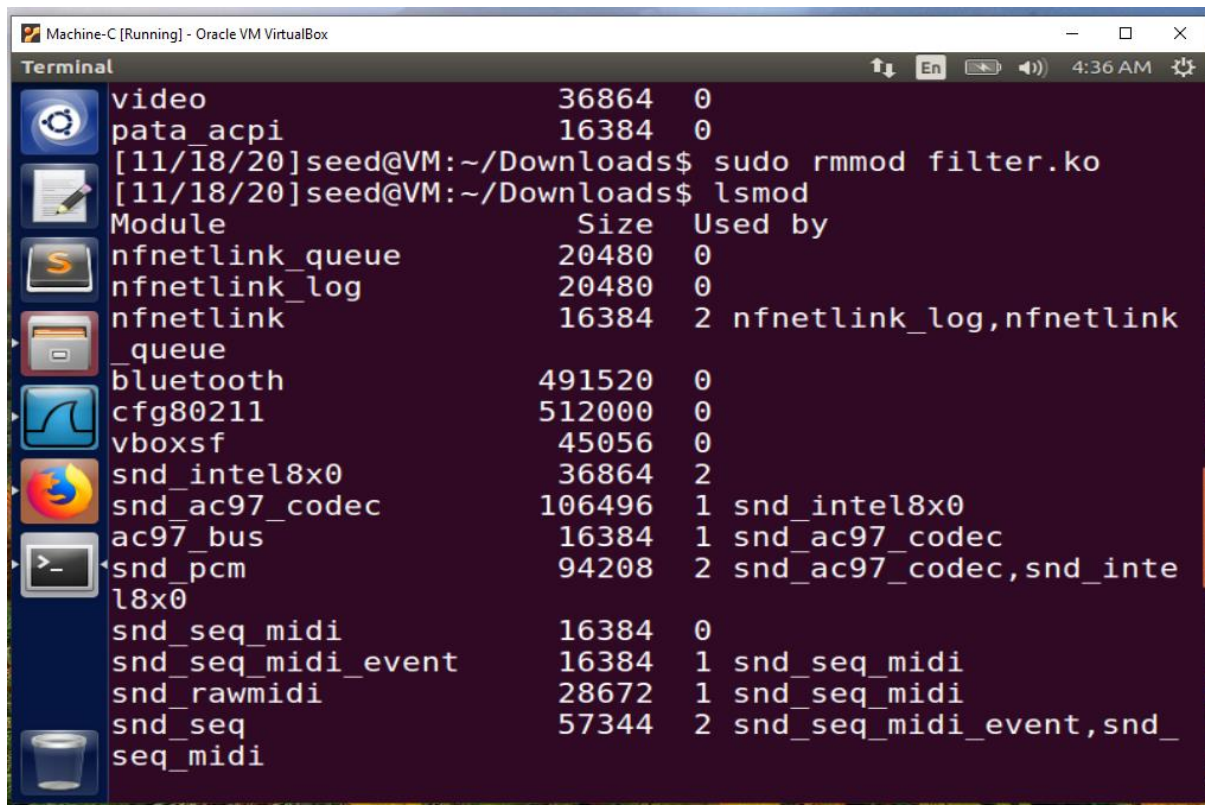
```
Machine-C [Running] - Oracle VM VirtualBox
Terminal File Edit View Search Terminal Help
[11/18/20]seed@VM:~$ cd /home/seed/Downloads/
[11/18/20]seed@VM:~/Downloads$ make
make -C /lib/modules/4.8.0-36-generic/build M=/home/seed/Downloads modules
make[1]: Entering directory '/usr/src/linux-headers-4.8.0-36-generic'
  CC [M]  /home/seed/Downloads/filter.o
Building modules, stage 2.
MODPOST 1 modules
  CC      /home/seed/Downloads/filter.mod.o
  LD [M]  /home/seed/Downloads/filter.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.8.0-36-generic'
[11/18/20]seed@VM:~/Downloads$
```

- After the module has been built using make file, load the module with the help of below command.  
[\*sudo insmod filter.ko\*](#)
- After listing the module, I was able to see filter using **lsmod** command.



```
Machine-C [Running] - Oracle VM VirtualBox
Terminal
0-36-generic'
[11/18/20]seed@VM:~/Downloads$ sudo insmod filter.ko
[11/18/20]seed@VM:~/Downloads$ lsmod
Module                               Size  Used by
filter                               16384  0
nfnetlink_queue                     20480  0
nfnetlink_log                        20480  0
nfnetlink                            16384  2 nfnetlink_log,nfnetlink_queue
bluetooth                           491520  0
cfg80211                             512000  0
vboxsf                               45056  0
snd_intel8x0                          36864  2
snd_ac97_codec                       106496  1 snd_intel8x0
ac97_bus                             16384  1 snd_ac97_codec
snd_pcm                              94208  2 snd_ac97_codec,snd_intel8x0
snd_seq_midi                          16384  0
snd_seq_midi_event                   16384  1 snd_seq_midi
snd_rawmidi                          28672  1 snd_seq_midi
snd_seq                              57344  2 snd_seq_midi_event,snd_seq_midi
```

- Remove filter.ko module using command `sudo rmmod filter.ko`



The screenshot shows a terminal window titled "Machine-C [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```

video                36864    0
pata_acpi             16384    0
[11/18/20]seed@VM:~/Downloads$ sudo rmmod filter.ko
[11/18/20]seed@VM:~/Downloads$ lsmod
Module                Size  Used by
nfnetlink_queue       20480    0
nfnetlink_log         20480    0
nfnetlink              16384    2 nfnetlink_log,nfnetlink_queue
bluetooth             491520   0
cfg80211              512000   0
vboxsf                45056    0
snd_intel8x0          36864    2
snd_ac97_codec        106496    1 snd_intel8x0
ac97_bus              16384    1 snd_ac97_codec
snd_pcm               94208    2 snd_ac97_codec,snd_intel8x0
snd_seq_midi          16384    0
snd_seq_midi_event    16384    1 snd_seq_midi
snd_rawmidi           28672    1 snd_seq_midi
snd_seq               57344    2 snd_seq_midi_event,snd_seq_midi
seq_midi

```

## 3.2 A Simple Program that Uses Netfilter

- With the help of program used in Task.2



```

#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/netfilter.h>
#include <linux/netfilter_ipv4.h>
#include <linux/ip.h>
#include <linux/tcp.h>
#include <linux/inet.h>

/* This is the structure we shall use to register our function */
static struct nf_hook_ops nfho;

/* This is the hook function itself */
unsigned int hook_func(void *priv, struct sk_buff *skb, const struct nf_hook_state *state) {
    struct iphdr *iph;
    struct tcphdr *tcph;

    iph = ip_hdr(skb);
    tcph = (void *)iph+iph->ihl*4;

    if(iph->saddr == in_aton("10.0.2.5") && iph->daddr == in_aton("10.0.2.6")) {
        printk("Dropping packet from %d.%d.%d.%d to %d.%d.%d.%d", ((unsigned char *)&iph->saddr)[0], ((unsigned char *)&iph->saddr)[1], ((unsigned char *)&iph->saddr)[2], ((unsigned char *)&iph->saddr)[3], ((unsigned char *)&iph->daddr)[0], ((unsigned char *)&iph->daddr)[1], ((unsigned char *)&iph->daddr)[2], ((unsigned char *)&iph->daddr)[3]);
        return NF_DROP;
    }
    else {
        return NF_ACCEPT;
    }
}

/* Initialization routine */
int init_module() {
    /* Fill in our hook structure */
    nfho.hook = hook_func; /* Handler function */
    nfho.hooknum = NF_INET_PRE_ROUTING; /* First hook for IPv4 */
    nfho.pf = PF_INET;
    nfho.priority = NF_IP_PRI_FIRST; /* Make our function first */
    nf_register_hook(&nfho);
    return 0;
}

/* Cleanup routine */
void cleanup_module() {
    nf_unregister_hook(&nfho);
}

```

- Observed that packets were dropped when sent from source (Machine C) to destination (Machine A)

```

Machine-C [Running] - Oracle VM VirtualBox
Terminal
[11/18/20]seed@VM:~$ exit
logout
Connection closed by foreign host.
[11/18/20]seed@VM:~/Downloads$ sudo rmmod filter.ko
[11/18/20]seed@VM:~/Downloads$ nano filter.c
[11/18/20]seed@VM:~/Downloads$ make
make -C /lib/modules/4.8.0-36-generic/build M=/home/seed/Downloads modules
make[1]: Entering directory '/usr/src/linux-headers-4.8.0-36-generic'
CC [M] /home/seed/Downloads/filter.o
Building modules, stage 2.
MODPOST 1 modules
CC /home/seed/Downloads/filter.mod.o
LD [M] /home/seed/Downloads/filter.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.8.0-36-generic'
[11/18/20]seed@VM:~/Downloads$ sudo insmod filter.ko
[11/18/20]seed@VM:~/Downloads$ telnet 10.0.2.5
Trying 10.0.2.5...

```

- [illegible]