



DEPARTMENT OF COMPUTER SCIENCE &  
ENGINEERING

Session: Jan 2021 – May 2021

**COMPUTER NETWORK SECURITY  
LAB – 3**

**NAME : AMRUTHA BS**

## Lab Setup

Task 1, 2

VM1 : 10.0.2.8

VM2 : 10.0.2.15

## Task 3

VM1 : 10.0.2.15

VM2 : 10.0.2.5

VM3 : 10.0.2.4

## Task 4

VM1 : 10.0.2.10

VM2 : 10.0.2.11

## **Task 1: Using Firewall**

In this task, we will use two VMs (VM1 and VM2). We first show that VM1 can telnet to VM2.

**Command:**

**telnet 10.0.2.15**

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ telnet 10.0.2.15
Trying 10.0.2.15...
Connected to 10.0.2.15.
Escape character is '^].
Ubuntu 16.04.2 LTS
PES2201800618_AmruthaBS_VM2 login: seed
Password:
Last login: Sun Feb  7 13:14:39 IST 2021 from 10.0.2.15 on pts/25
/usr/lib/update-notifier/update-motd-fsck-at-reboot[:59: integer expression expected: 0
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.

[02/21/21]seed@PES2201800618_AmruthaBS_VM2:~$
```

We need to first prevent VM1 from being able to telnet to VM2. For this we will configure the firewall using ufw. We will first enable the firewall on VM1.

### Commands:

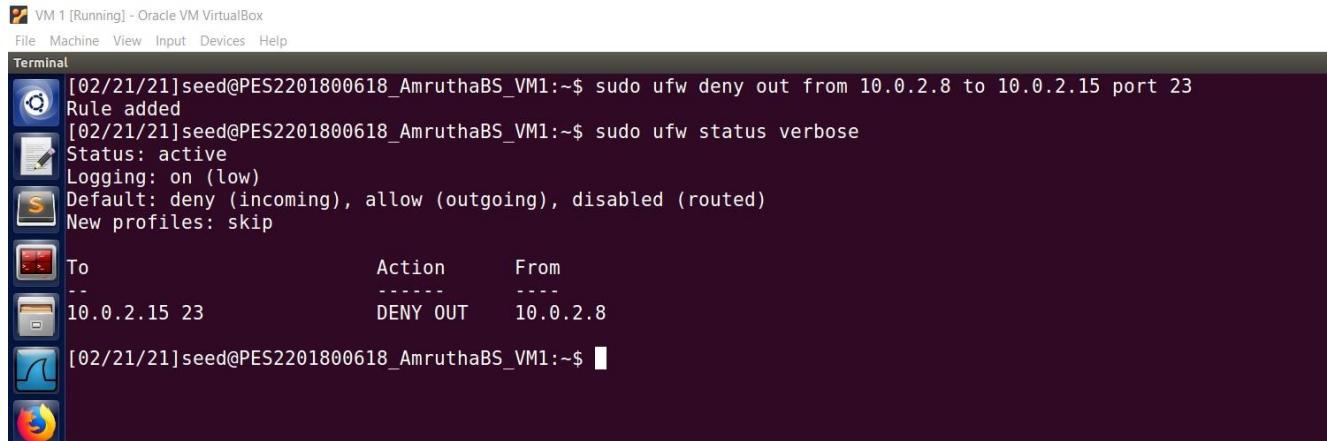
**sudo ufw enable**  
**sudo ufw status verbose**

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw enable
Firewall is active and enabled on system startup
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

We will next configure the firewall on VM1 to deny telnet (port 23) to VM2.

### Command:

**sudo ufw deny out from 10.0.2.8 to 10.0.2.15 port 23 sudo  
ufw status verbose**

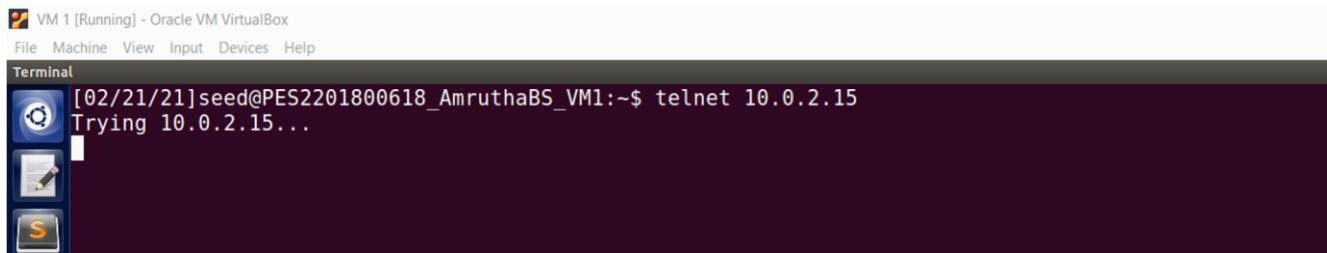


VM 1 [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
Terminal  
[02/21/21]seed@PES2201800618\_AmruthaBS\_VM1:~\$ sudo ufw deny out from 10.0.2.8 to 10.0.2.15 port 23  
Rule added  
[02/21/21]seed@PES2201800618\_AmruthaBS\_VM1:~\$ sudo ufw status verbose  
Status: active  
Logging: on (low)  
Default: deny (incoming), allow (outgoing), disabled (routed)  
New profiles: skip  
To Action From  
-- -----  
10.0.2.15 23 DENY OUT 10.0.2.8  
[02/21/21]seed@PES2201800618\_AmruthaBS\_VM1:~\$ █

We now try to telnet from VM1 to VM2. Due to the firewall rule, the telnet is denied.

**Command:**

**telnet 10.0.2.15**



VM 1 [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
Terminal  
[02/21/21]seed@PES2201800618\_AmruthaBS\_VM1:~\$ telnet 10.0.2.15  
Trying 10.0.2.15...  
█

We next need to deny telnet from VM2 to VM1. We will first test whether telnet works from VM2 to VM1.

**Command:**

**telnet 10.0.2.8**

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM2:~$ telnet 10.0.2.8
Trying 10.0.2.8...
```

We next delete the firewall rule in VM1. We add a rule to prevent VM2 from being able to do telnet to VM1.

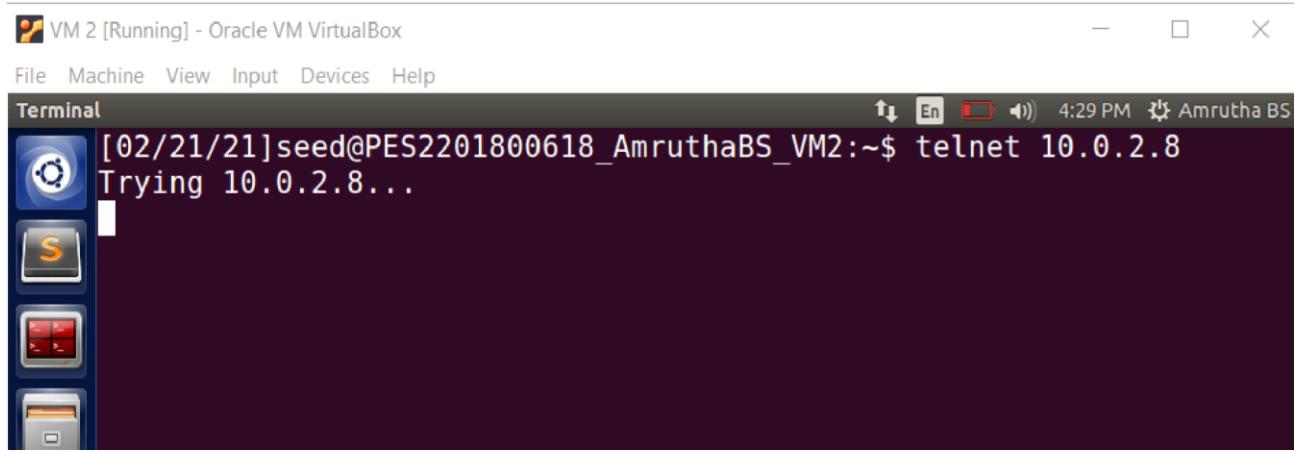
### Commands:

**sudo ufw delete 1**

**sudo ufw deny in from 10.0.2.10 to 10.0.2.9 port 23**

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw delete 1
Deleting:
deny out from 10.0.2.8 to 10.0.2.15 port 23
Proceed with operation (y|n)? y
Rule deleted
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw deny in from 10.0.2.15 to 10.0.2.8 port 23
Rule added
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

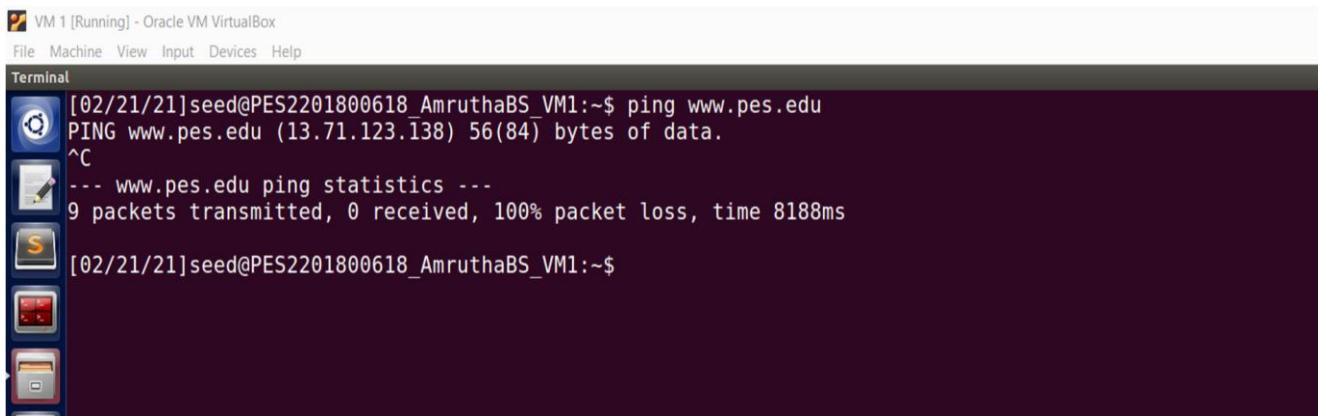
We can try to telnet from VM2 to VM1 again. The telnet is blocked and does not work.



```
[02/21/21]seed@PES2201800618_AmruthaBS_VM2:~$ telnet 10.0.2.8
Trying 10.0.2.8...
```

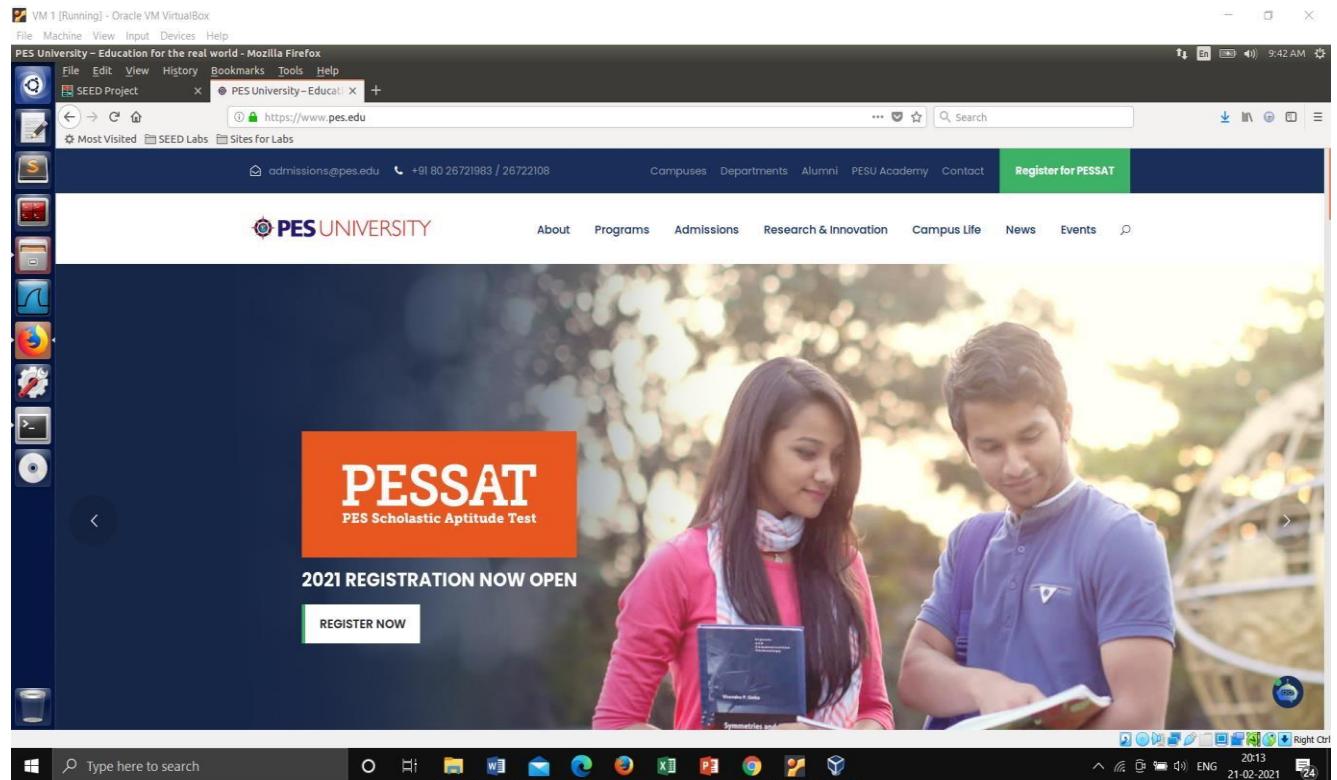
We next need to block VM1 from visiting a website. We use the www.pes.edu website. We first find its IP address.

**Command:** ping  
**www.pes.edu**



```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ ping www.pes.edu
PING www.pes.edu (13.71.123.138) 56(84) bytes of data.
^C
--- www.pes.edu ping statistics ---
9 packets transmitted, 0 received, 100% packet loss, time 8188ms
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

We also test whether the website is accessible to the browser.  
We are able to access the website.



We next add the firewall rule to prevent VM1 from accessing the IP address for www.pes.edu **Commands:**

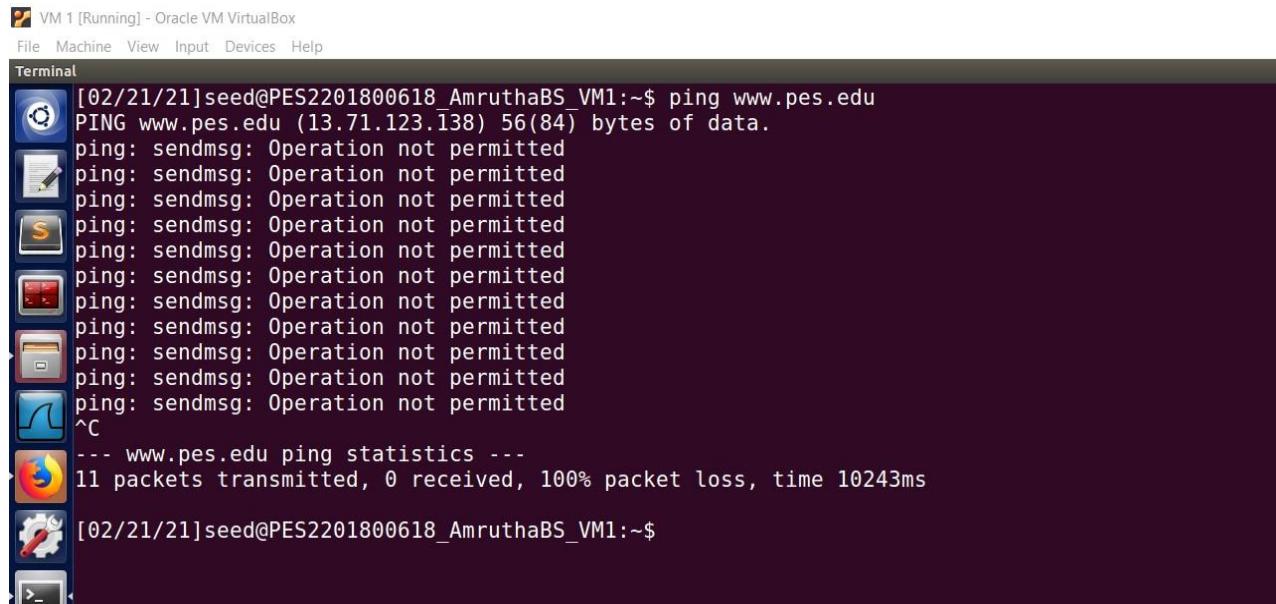
**sudo ufw delete 1**

**sudo ufw deny out to 13.71.123.138**

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw delete 1
Deleting:
deny from 10.0.2.15 to 10.0.2.8 port 23
Proceed with operation (y|n)? y
Rule deleted
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw deny out to 13.71.123.138
Rule added
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

With the firewall rule in place, we next try to ping www.pes.edu. We see the message “Operation not permitted” because the firewall has blocked it.

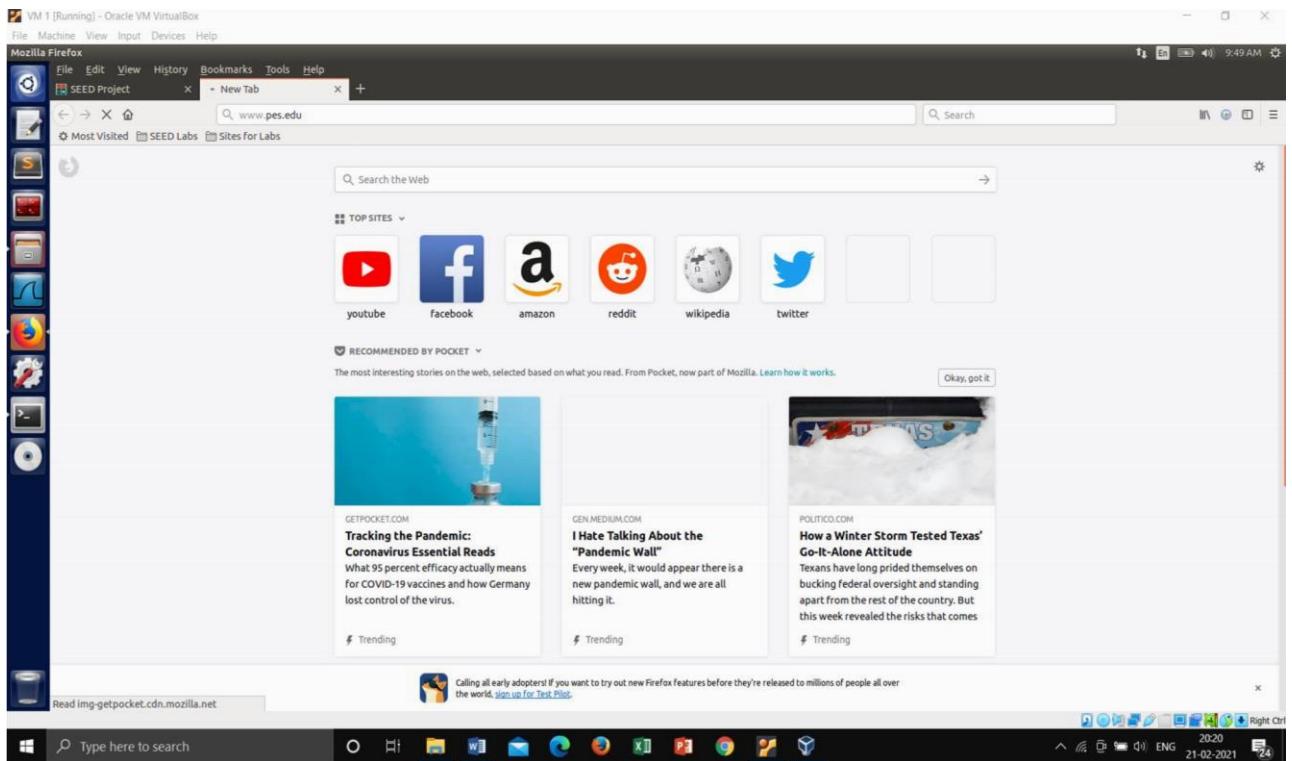
This can be verified by revisiting www.pes.edu in the Firefox browser. The page is not loaded.



The screenshot shows a terminal window within Oracle VM VirtualBox. The window title is "VM 1 [Running] - Oracle VM VirtualBox". The menu bar includes File, Machine, View, Input, Devices, and Help. The terminal window displays the following command and its output:

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$ ping www.pes.edu
PING www.pes.edu (13.71.123.138) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
^C
--- www.pes.edu ping statistics ---
11 packets transmitted, 0 received, 100% packet loss, time 10243ms
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

This can be verified by revisiting www.pes.edu in the Firefox browser. The page is not loaded.



## Task 2: How Firewall Works

In this task, we will develop a firewall using netfilter and LKM. We implement five rules in this firewall:

- Block telnet from VM1 to VM2
- Block telnet from VM2 to VM1
- Block external website access from VM1
- Block ssh from VM1 to VM2
- Block ssh from VM2 to VM1

VM 1 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

lkm.c (~/task2) - gedit

```
tcp= (struct tcphdr*)((__u32 *)ip + ip->ihl);
sou_port = tcp->source;
des_port = tcp->dest;

if(sou_ip == in_aton("10.0.2.8") && des_ip == in_aton("10.0.2.15") && ntohs(des_port) == 23){
    printk(KERN_INFO "blocking telnet:VM1 to VM2.\n");
    return NF_DROP;
}

if(sou_ip == in_aton("10.0.2.15") && des_ip == in_aton("10.0.2.8") && ntohs(des_port) == 23){
    printk(KERN_INFO "blocking telnet:VM2 to VM1.\n");
    return NF_DROP;
}

if(sou_ip == in_aton("10.0.2.8") && ntohs(des_port) == 80){
    printk(KERN_INFO "blocking external website access\n");
    return NF_DROP;
}

if(sou_ip == in_aton("10.0.2.8") && des_ip == in_aton("10.0.2.15") && ntohs(des_port) == 22){
    printk(KERN_INFO "blocking ssh: VM1 to VM2.\n");
    return NF_DROP;
}

if(sou_ip == in_aton("10.0.2.15") && des_ip == in_aton("10.0.2.8") && ntohs(des_port) == 22){
    printk(KERN_INFO "blocking ssh:VM2 to VM1.\n");
    return NF_DROP;
}

printk(KERN_INFO "allow packet.\n");
return NF_ACCEPT;
}

int init_module()
{
    printk(KERN_INFO "hello netfilter!\n");
    nfho_in.hook = hook_func;
    nfho_in.hooknum = NF_INET_PRE_ROUTING;
    nfho_in(pf = PF_INET;
    nfho_in.priority = NF_IP_PRI_FIRST;
    nf_register_hook(&nfho_in);
    nfho_out.hook = hook_func;
    nfho_out.hooknum = NF_INET_POST_ROUTING;
    nfho_out(pf = PF_INET;
    nfho_out.priority = NF_IP_PRI_FIRST;
    nf_register_hook(&nfho_out);
    return 0;
}
void cleanup_module()
```

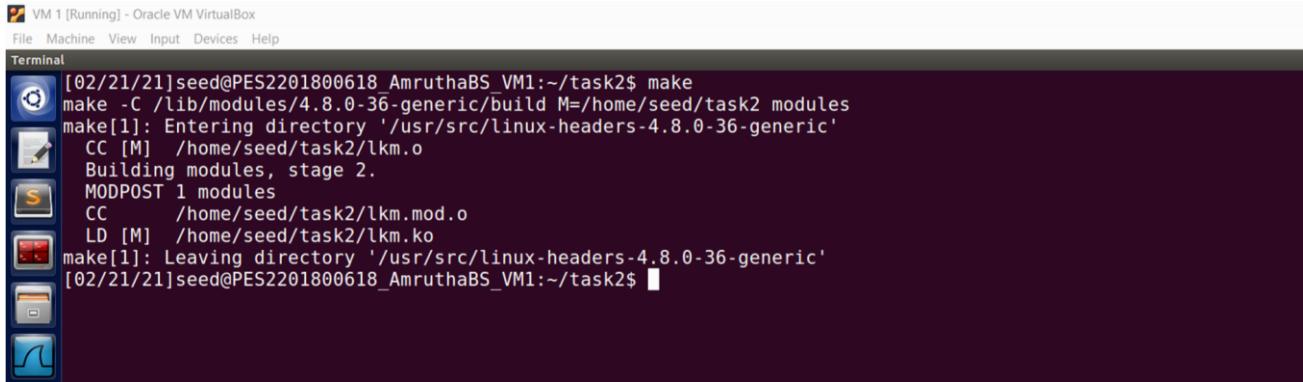
Type here to search

Windows Start File Explorer Mail Edge Excel Powerpoint Google Chrome

Below is the code for the firewall: lkm.c

Makefile:

The program can be compiled using the make command.



```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$ make
make -C /lib/modules/4.8.0-36-generic/build M=/home/seed/task2 modules
make[1]: Entering directory '/usr/src/linux-headers-4.8.0-36-generic'
  CC [M]  /home/seed/task2/lkm.o
Building modules, stage 2.
MODPOST 1 modules
  CC      /home/seed/task2/lkm.mod.o
  LD [M]  /home/seed/task2/lkm.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.8.0-36-generic'
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$
```

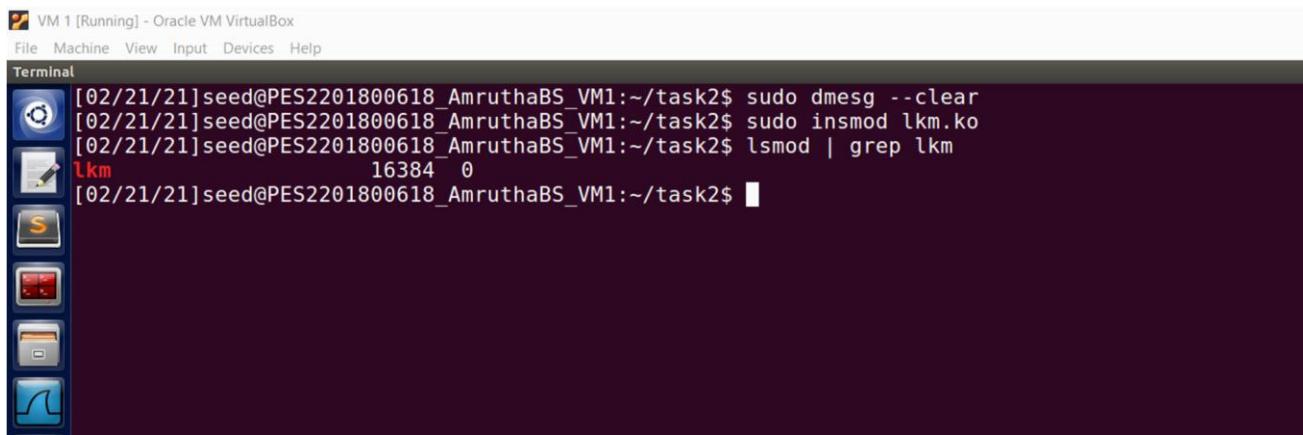
The compiled kernel module (lkm.ko) can be inserted using insmod:

**Commands:**

**sudo dmesg --clear**

**sudo insmod lkm.ko**

**lsmod | grep lkm**



```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$ sudo dmesg --clear
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$ sudo insmod lkm.ko
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$ lsmod | grep lkm
lkm               16384  0
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$
```

We first test whether telnet from VM1 to VM2 is blocked.

**Command:**

**telnet 10.0.2.15**

```
[02/21/21]seed@PES2201800618_AmruthaBS_VM1:~/task2$ telnet 10.0.2.15
Trying 10.0.2.15...
```

We next test whether telnet from VM2 to VM1 is blocked.

```
Terminal [02/21/21]seed@PES2201800618_AmruthaBS_VM2:~$ telnet 10.0.2.8
Trying 10.0.2.8...
telnet: Unable to connect to remote host: No route to host
[02/21/21]seed@PES2201800618_AmruthaBS_VM2:~$
```

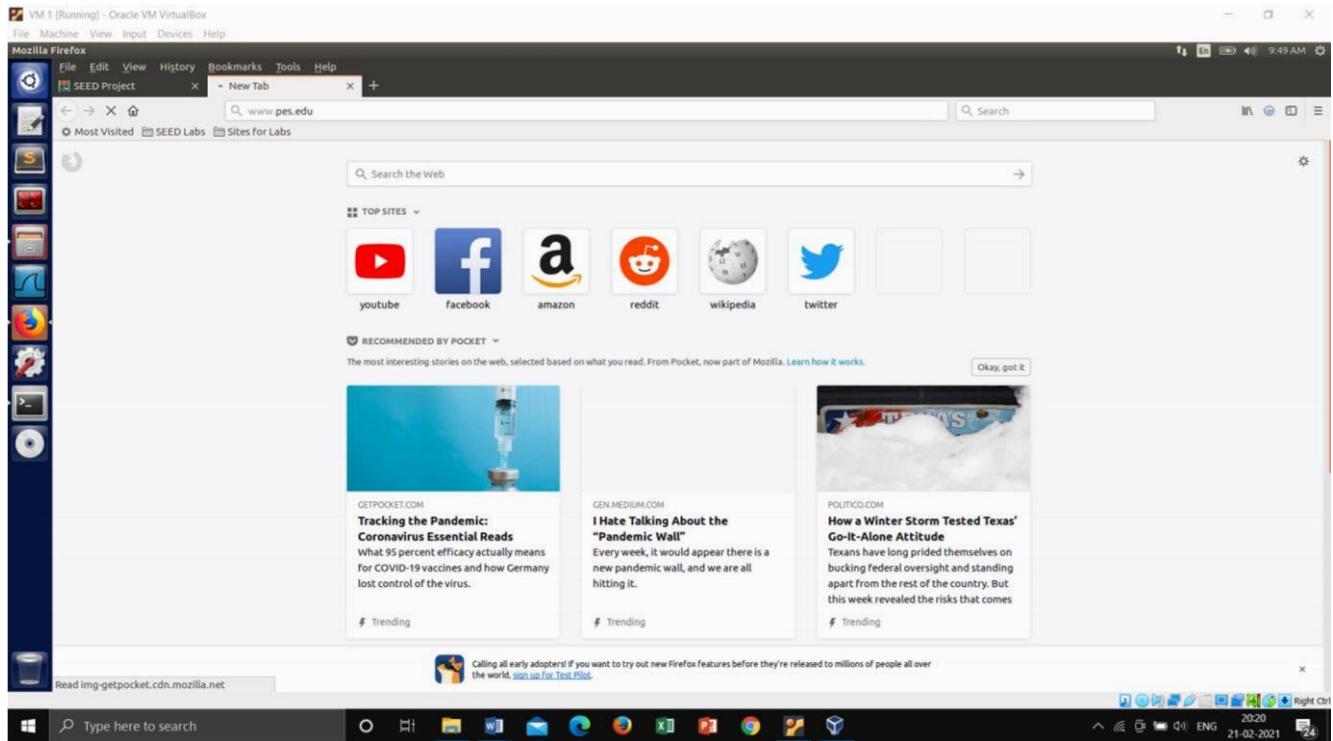
**Packets being dropped can be seen here:**

**Commands:**

**dmesg | tail -10 (This command will show u dropping packets)**

```
[ 292.065487] allow packet.
[ 324.001830] blocking telnet:VM2 to VM1.
[ 325.017469] blocking telnet:VM2 to VM1.
[ 327.033705] blocking telnet:VM2 to VM1.
[ 331.099894] blocking telnet:VM2 to VM1.
[ 337.048292] allow packet.
[ 337.048298] allow packet.
[ 337.048307] allow packet.
[ 337.048308] allow packet.
[ 339.295905] blocking telnet:VM2 to VM1.
```

We next test whether external website access (port 80) is allowed from VM1. We try to load the page www.pes.edu in the Firefox browser and the page does not load.



## Command:

`dmesg | tail -10`

```
[ 562.445424] allow packet.  
[ 562.445499] allow packet.  
[ 562.449530] allow packet.  
[ 562.449581] allow packet.  
[ 562.449585] allow packet.  
[ 562.450382] blocking external website access  
[ 563.193805] blocking external website access  
[ 563.481729] blocking external website access  
[ 565.209021] blocking external website access  
[ 565.496938] blocking external website access
```

ssh is blocked from VM1 to VM2.

```
[03/07/21]seed@PES2201800618_AmruthaBS_VM1:~$ ssh seed@10.0.2.15
```

ssh is blocked from VM2 to VM1.

```
[03/07/21]seed@PES2201800618_AmruthaBS_VM2:~$ ssh seed@10.0.2.8
```

We can see the packets being dropped

```
[ 615.743885] blocking ssh: VM1 to VM2.  
[ 617.758641] blocking ssh: VM1 to VM2.  
[ 621.980389] blocking ssh: VM1 to VM2.  
[ 630.168783] blocking ssh: VM1 to VM2.  
[ 638.016896] allow packet.  
[ 639.028488] allow packet.  
[ 648.379347] blocking ssh:VM2 to VM1.  
[ 649.402815] blocking ssh:VM2 to VM1.  
[ 651.420306] blocking ssh:VM2 to VM1.  
[ 655.614142] blocking ssh:VM2 to VM1.
```

### Task 3: Evading Egress Filtering

In this task, we will be using a ssh tunnel to evade egress filtering. Please delete all the firewall rules from the previous task using the ufw delete <rule\_number> command.

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

In this lab, we will use three VMs. VM1 will be blocked from being able to telnet to VM2. We will utilize a ssh tunnel to allow VM1 to telnet to VM3 via VM2. The diagram below depicts the tunnel (in the diagram, the home machine is VM1, the apollo machine is VM2 and the work machine is VM3).

We will first block VM1 from being able to telnet to any other machine.

**Command:**

**sudo ufw enable**

**ufw status verbose**

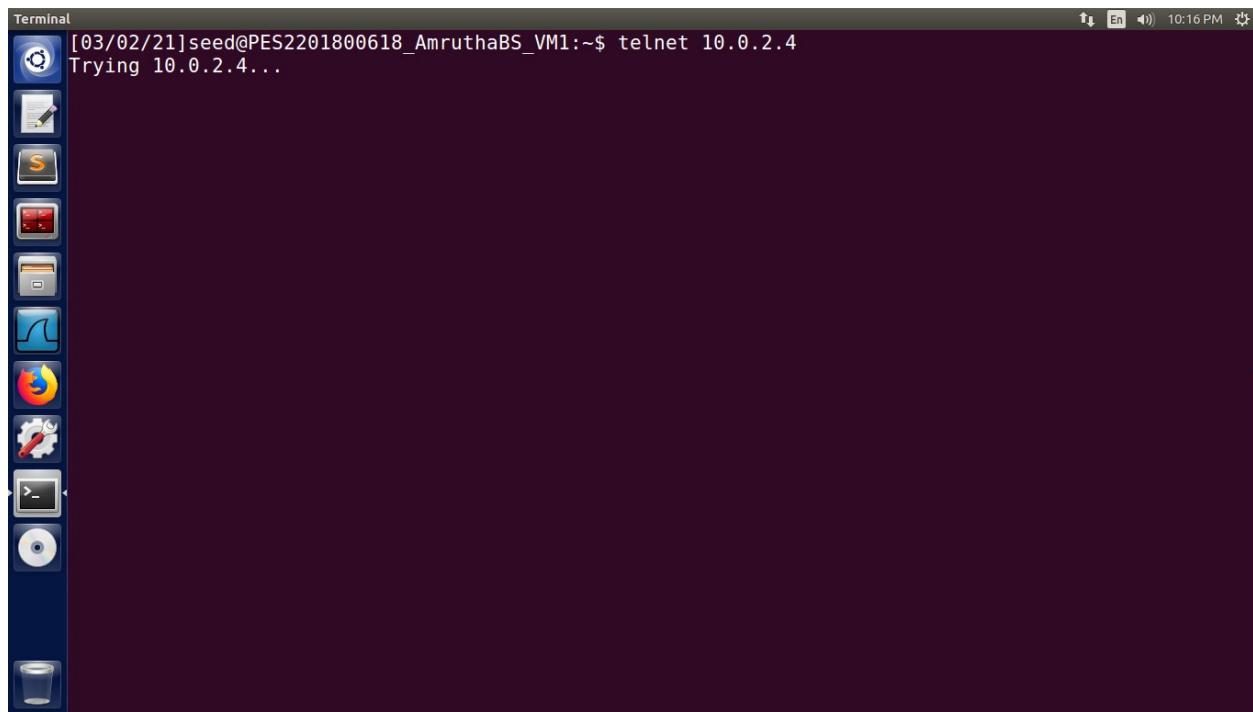
**sudo ufw deny out from 10.0.2.15 to any port 23**

**sudo ufw status verbose**

```
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw enable
Firewall is active and enabled on system startup
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw deny out from 10.0.2.15 to any port 23
Rule added
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
To           Action      From
--           -----      ---
23          DENY OUT   10.0.2.15
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

The effect of the previous firewall rule can be observed below. The telnet is blocked. **Command:**

**telnet 10.0.2.4**

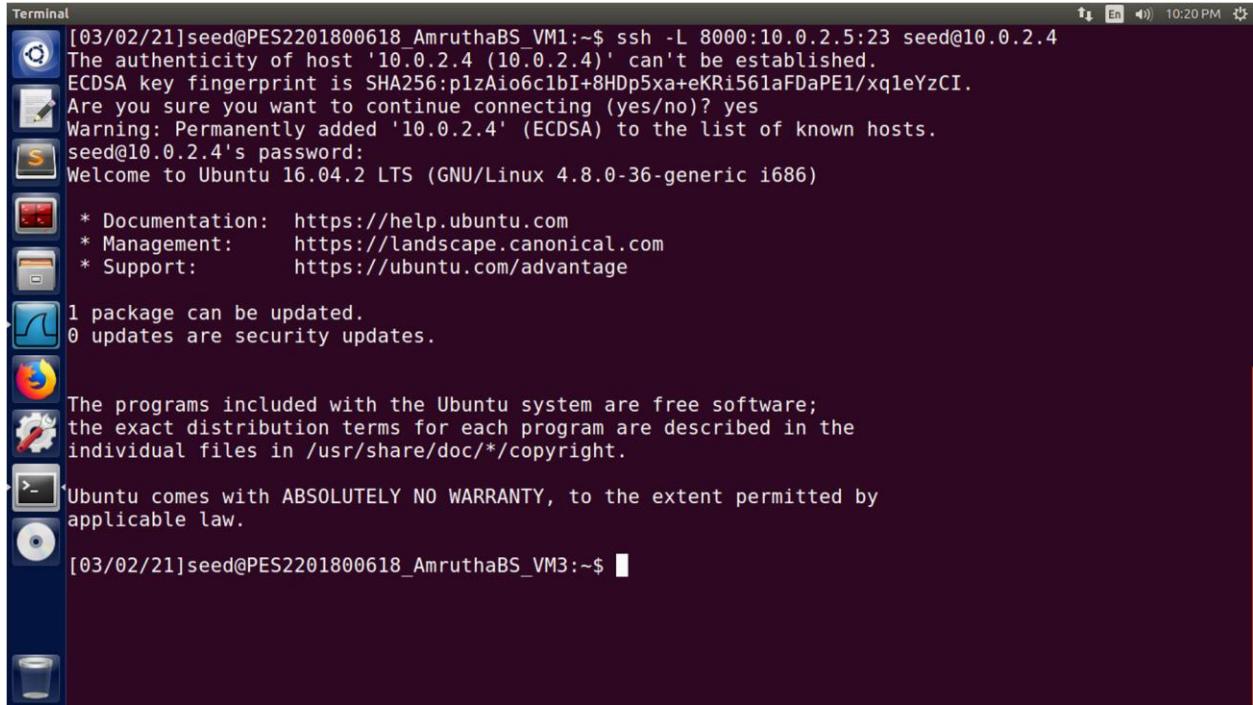


```
Terminal [03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ telnet 10.0.2.4
Trying 10.0.2.4...
```

We next setup a ssh tunnel between VM1 and VM2 to allow VM1 to telnet via VM2, evading the firewall on VM1. The ssh command below allows VM1 to use its local port 8000 to telnet to VM3 via VM2.

**Command:**

```
ssh -L 8000:10.0.2.5:23 seed@10.0.2.4
```

A screenshot of a Linux desktop environment, specifically Ubuntu 16.04 LTS, showing a terminal window. The terminal window has a dark background and displays the following text:

```
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ ssh -L 8000:10.0.2.5:23 seed@10.0.2.4
The authenticity of host '10.0.2.4 (10.0.2.4)' can't be established.
ECDSA key fingerprint is SHA256:plzAio6c1bI+8HDp5xa+eKRi561aFDaPE1/xqleYzCI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.4' (ECDSA) to the list of known hosts.
seed@10.0.2.4'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.

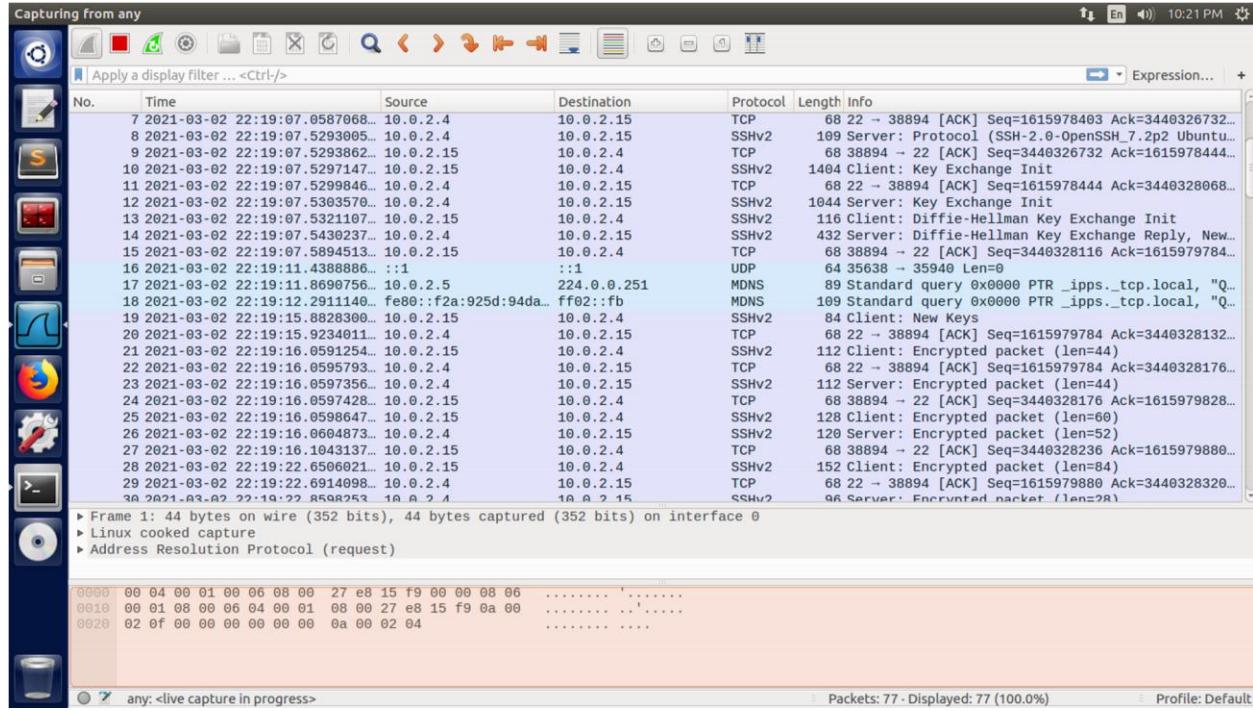
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

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

[03/02/21]seed@PES2201800618_AmruthaBS_VM3:~$
```

The desktop interface includes a dock with icons for Dash, Home, Applications, and the Dash search bar.

VM1 Wireshark capture.



VM2 Wireshark capture.

Capturing from any

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
3	2021-03-02 22:19:16.775831...	10.0.2.5	224.0.0.251	MDNS	89	Standard query 0x0000 PTR _ipps._tcp.local, "Q...
4	2021-03-02 22:19:17.1981255...	fe80::f2a:925d:94da...	ff02::fb	MDNS	109	Standard query 0x0000 PTR _ipps._tcp.local, "Q...
5	2021-03-02 22:19:27.677977...	::1	::1	UDP	64	38279 → 40289 Len=0
6	2021-03-02 22:19:36.8410395...	10.0.2.5	10.0.2.3	DHCP	344	DHCP Request - Transaction ID 0xbff517a7d
7	2021-03-02 22:19:37.4040391...	10.0.2.3	10.0.2.5	DHCP	592	DHCP ACK - Transaction ID 0xbff517a7d
8	2021-03-02 22:19:42.0437531...	PcsCompu_96:29:52	ARP	44	Who has 10.0.2.3? Tell 10.0.2.5	
9	2021-03-02 22:19:42.0442029...	PcsCompu_5b:3e:10	ARP	62	10.0.2.3 is at 08:00:27:5b:3e:10	
10	2021-03-02 22:19:47.6977856...	::1	::1	UDP	64	38279 → 40289 Len=0
11	2021-03-02 22:19:49.7898584...	10.0.2.4	224.0.0.251	MDNS	89	Standard query 0x0000 PTR _ipps._tcp.local, "Q...
12	2021-03-02 22:19:49.8752326...	fe80::e907:f1e1:940...	ff02::fb	MDNS	109	Standard query 0x0000 PTR _ipps._tcp.local, "Q...
13	2021-03-02 22:20:07.7078077...	::1	::1	UDP	64	38279 → 40289 Len=0
14	2021-03-02 22:20:27.7339418...	fe80::a2e5:b8eb:29a...	ff02::fb	MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM...
15	2021-03-02 22:20:22.7000142...	10.0.2.15	224.0.0.251	MDNS	162	Standard query 0x0000 PTR _ftp._tcp.local, "QM...
16	2021-03-02 22:20:27.7196378...	::1	::1	UDP	64	38279 → 40289 Len=0
17	2021-03-02 22:20:47.7339418...	::1	::1	UDP	64	38279 → 40289 Len=0
18	2021-03-02 22:21:07.7506903...	::1	::1	UDP	64	38279 → 40289 Len=0
19	2021-03-02 22:21:27.7660678...	::1	::1	UDP	64	38279 → 40289 Len=0
20	2021-03-02 22:21:47.7752217...	::1	::1	UDP	64	38279 → 40289 Len=0
21	2021-03-02 22:22:07.7914856...	::1	::1	UDP	64	38279 → 40289 Len=0
22	2021-03-02 22:22:27.8117072...	::1	::1	UDP	64	38279 → 40289 Len=0
23	2021-03-02 22:22:47.8354482...	::1	::1	UDP	64	38279 → 40289 Len=0
24	2021-03-02 22:23:08.8448820...	::1	::1	UDP	64	38279 → 40289 Len=0
25	2021-03-02 22:23:27.8543280...	::1	::1	UDP	64	38279 → 40289 Len=0

► Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0  
 ► Linux cooked capture  
 ► Internet Protocol Version 6, Src: ::1, Dst: ::1  
 ► User Datagram Protocol, Src Port: 38279, Dst Port: 40289

0000 00 00 03 04 00 06 00 00 00 00 00 00 00 00 86 dd .. ....  
 0010 60 07 8b 38 00 08 11 40 00 00 00 00 00 00 00 00 ..8...@ .. ....  
 0020 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 .. ....  
 0030 00 00 00 00 00 00 01 95 87 9d 61 00 08 00 1b .. .... .a....

any: <live capture in progress>

Packets: 25 · Displayed: 25 (100.0%) · Profile: Default

VM3 Wireshark capture.

Capturing from any

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
4	2021-03-02 22:19:09.5381865...	10.0.2.15	10.0.2.4	TCP	76	38894 → 22 [SYN] Seq=3440326690 Win=29200 Len=0
5	2021-03-02 22:19:09.5382239...	10.0.2.4	10.0.2.15	TCP	76	22 → 38894 [SYN, ACK] Seq=1615978402 Ack=3440326
6	2021-03-02 22:19:09.5391496...	10.0.2.15	10.0.2.4	TCP	68	38894 → 22 [ACK] Seq=3440326691 Ack=1615978403 W
7	2021-03-02 22:19:09.7051445...	10.0.2.15	10.0.2.4	SSHv2	109	Client: Protocol (SSH-2.0-OpenSSH_7.2p2 Ubuntu-4
8	2021-03-02 22:19:09.7051809...	10.0.2.4	10.0.2.15	TCP	68	22 → 38894 [ACK] Seq=1615978403 Ack=3440326732 W
9	2021-03-02 22:19:10.1759380...	10.0.2.4	10.0.2.15	SSHv2	109	Server: Protocol (SSH-2.0-OpenSSH_7.2p2 Ubuntu-4
10	2021-03-02 22:19:10.1764424...	10.0.2.15	10.0.2.4	TCP	68	38894 → 22 [ACK] Seq=3440326732 Ack=1615978444 W
11	2021-03-02 22:19:10.1767339...	10.0.2.15	10.0.2.4	SSHv2	1404	Client: Key Exchange Init
12	2021-03-02 22:19:10.1767382...	10.0.2.4	10.0.2.15	TCP	68	22 → 38894 [ACK] Seq=1615978444 Ack=3440328068 W
13	2021-03-02 22:19:10.1771159...	10.0.2.4	10.0.2.15	SSHv2	1044	Server: Key Exchange Init
14	2021-03-02 22:19:10.1791488...	10.0.2.15	10.0.2.4	SSHv2	116	Client: Diffie-Hellman Key Exchange Init
15	2021-03-02 22:19:10.1894948...	10.0.2.4	10.0.2.15	SSHv2	432	Server: Diffie-Hellman Key Exchange Reply, New K
16	2021-03-02 22:19:10.2365862...	10.0.2.15	10.0.2.4	TCP	68	38894 → 22 [ACK] Seq=3440328116 Ack=1615979784 W
17	2021-03-02 22:19:14.5164798...	10.0.2.5	224.0.0.251	MDNS	89	Standard query 0x0000 PTR _ipps._tcp.local, "QM"
18	2021-03-02 22:19:14.9379758...	fe80::f2a:925d:94da...	ff02::fb	MDNS	109	Standard query 0x0000 PTR _ipps._tcp.local, "QM"
19	2021-03-02 22:19:18.5299452...	10.0.2.15	10.0.2.4	SSHv2	84	Client: New Keys
20	2021-03-02 22:19:18.5700254...	10.0.2.4	10.0.2.15	TCP	68	22 → 38894 [ACK] Seq=1615979784 Ack=3440328132 W
21	2021-03-02 22:19:18.7061965...	10.0.2.15	10.0.2.4	SSHv2	112	Client: Encrypted packet (len=44)
22	2021-03-02 22:19:18.7062187...	10.0.2.4	10.0.2.15	TCP	68	22 → 38894 [ACK] Seq=1615979784 Ack=3440328176 W
23	2021-03-02 22:19:18.7064317...	10.0.2.4	10.0.2.15	SSHv2	112	Server: Encrypted packet (len=44)
24	2021-03-02 22:19:18.7067319...	10.0.2.15	10.0.2.4	TCP	68	38894 → 22 [ACK] Seq=3440328176 Ack=1615979828 W
25	2021-03-02 22:19:18.7068650...	10.0.2.15	10.0.2.4	SSHv2	120	Client: Encrypted packet (len=60)
26	2021-03-02 22:19:18.7072584...	10.0.2.4	10.0.2.15	SSHv2	120	Server: Encrypted packet (len=52)

► Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0  
 ► Linux cooked capture  
 ► Internet Protocol Version 6, Src: ::1, Dst: ::1  
 ► User Datagram Protocol, Src Port: 43059, Dst Port: 36158

0000 00 00 03 04 00 06 00 00 00 00 00 00 00 00 86 dd .. ....  
 0010 60 03 37 a6 00 08 11 40 00 00 00 00 00 00 00 00 ..7....@ .. ....  
 0020 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 .. ....  
 0030 00 00 00 00 00 00 00 01 a8 33 8d 3e 00 08 00 1b .. .... .3.>....

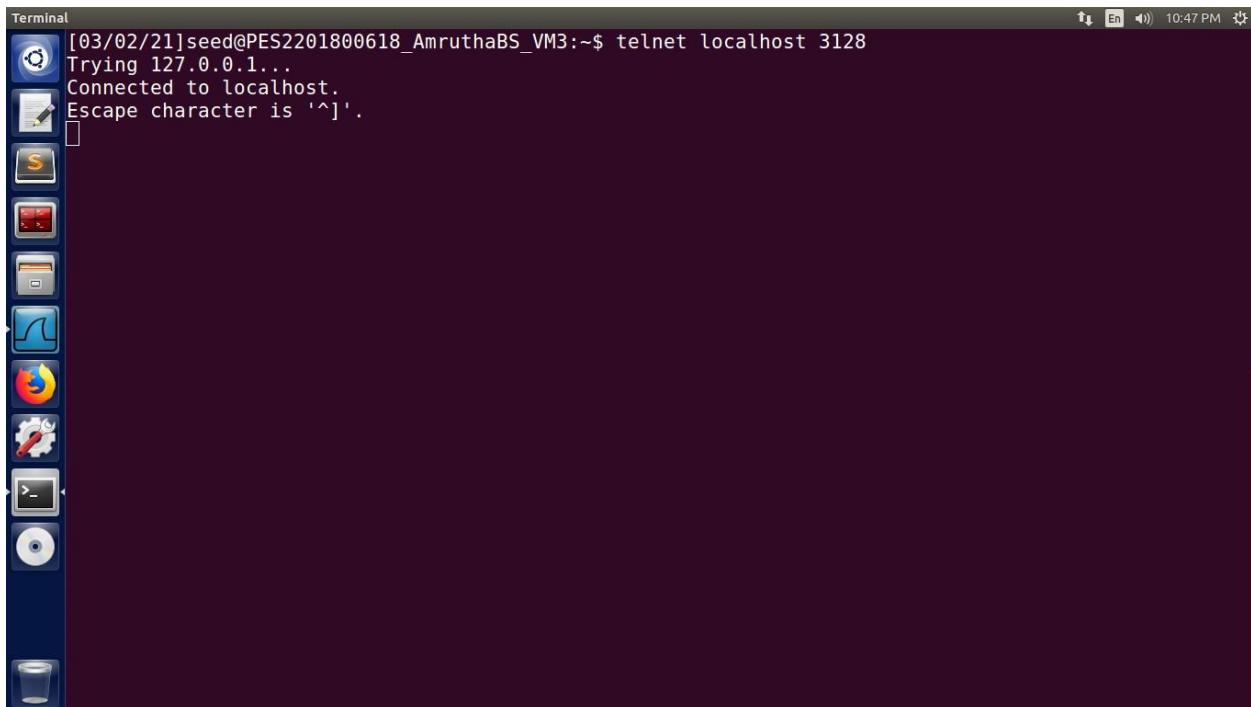
any: <live capture in progress>

Packets: 148 · Displayed: 148 (100.0%) · Profile: Default

With the ssh tunnel setup, we can now telnet from VM1 to VM3 even though the firewall policy on VM1 denies outgoing telnet.

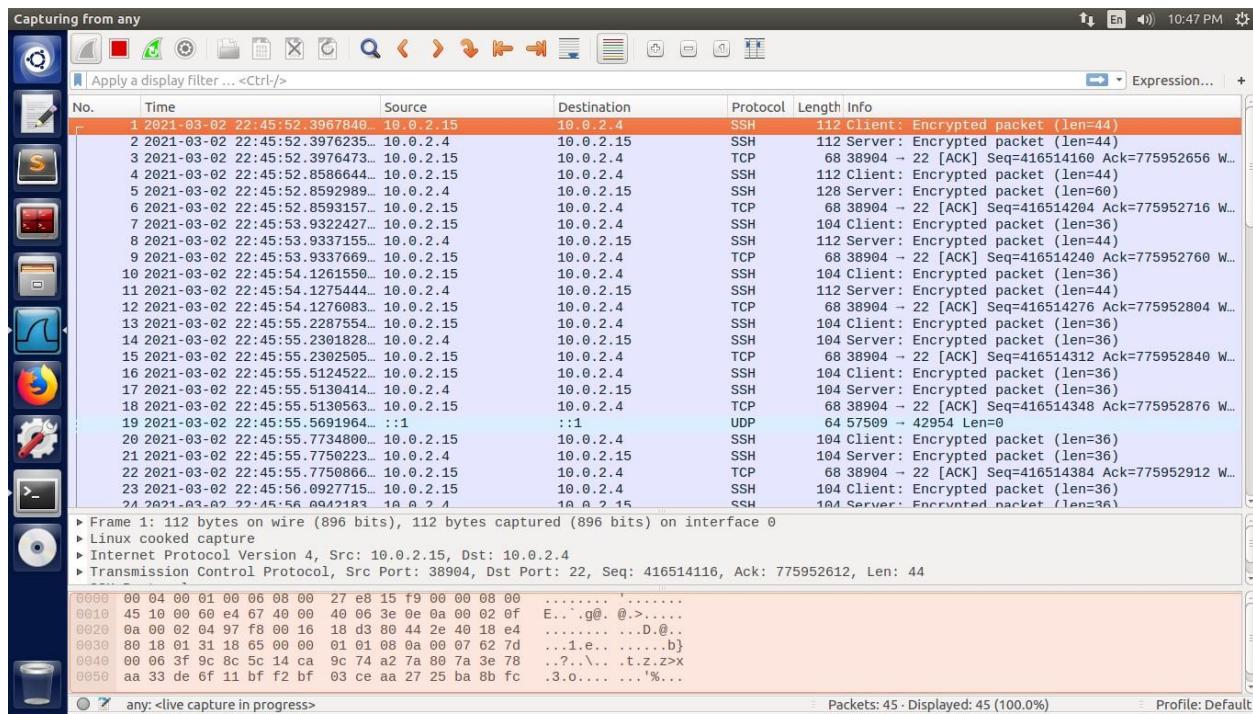
**Command:**

**telnet localhost 3128**

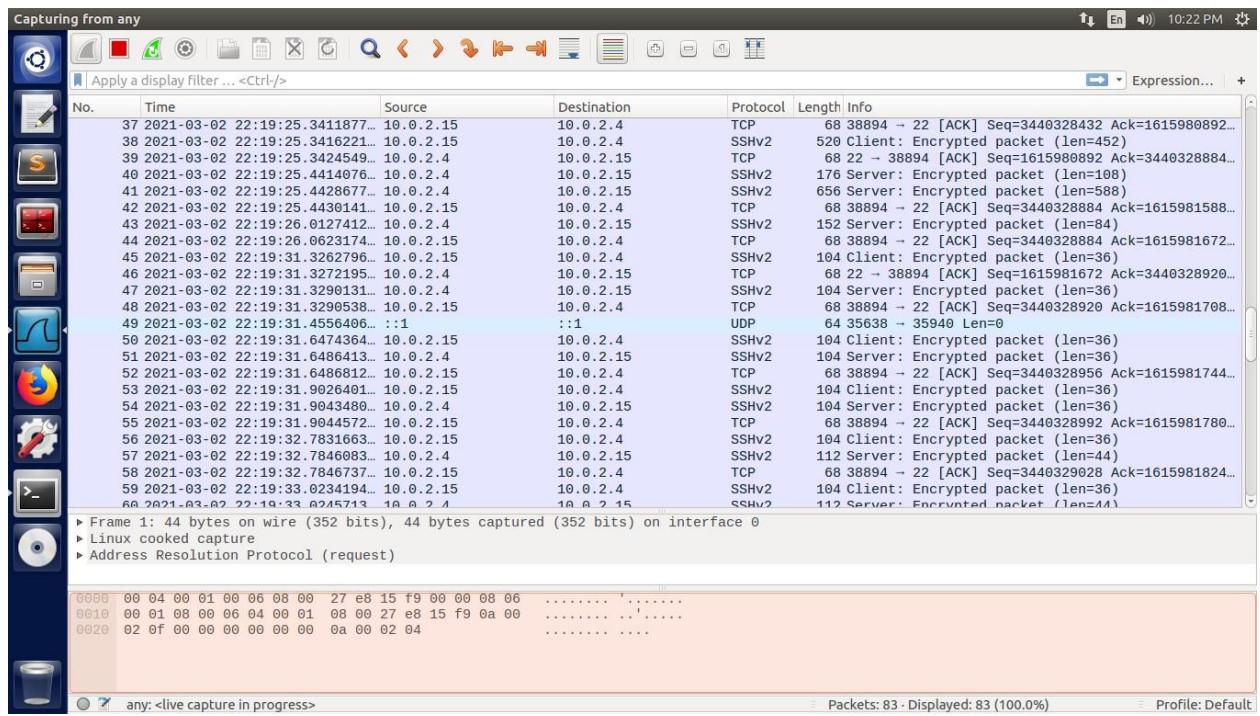


The screenshot shows a terminal window with a dark blue background. At the top, it displays the command: [03/02/21]seed@PES2201800618\_AmruthaBS\_VM3:~\$ telnet localhost 3128. Below this, the terminal shows the output of the command: Trying 127.0.0.1... Connected to localhost. Escape character is '^]'. To the left of the terminal window, there is a vertical dock containing icons for various applications: a terminal icon, a file manager icon, a browser icon, a settings gear icon, a terminal icon with a greater-than sign, and a disc icon.

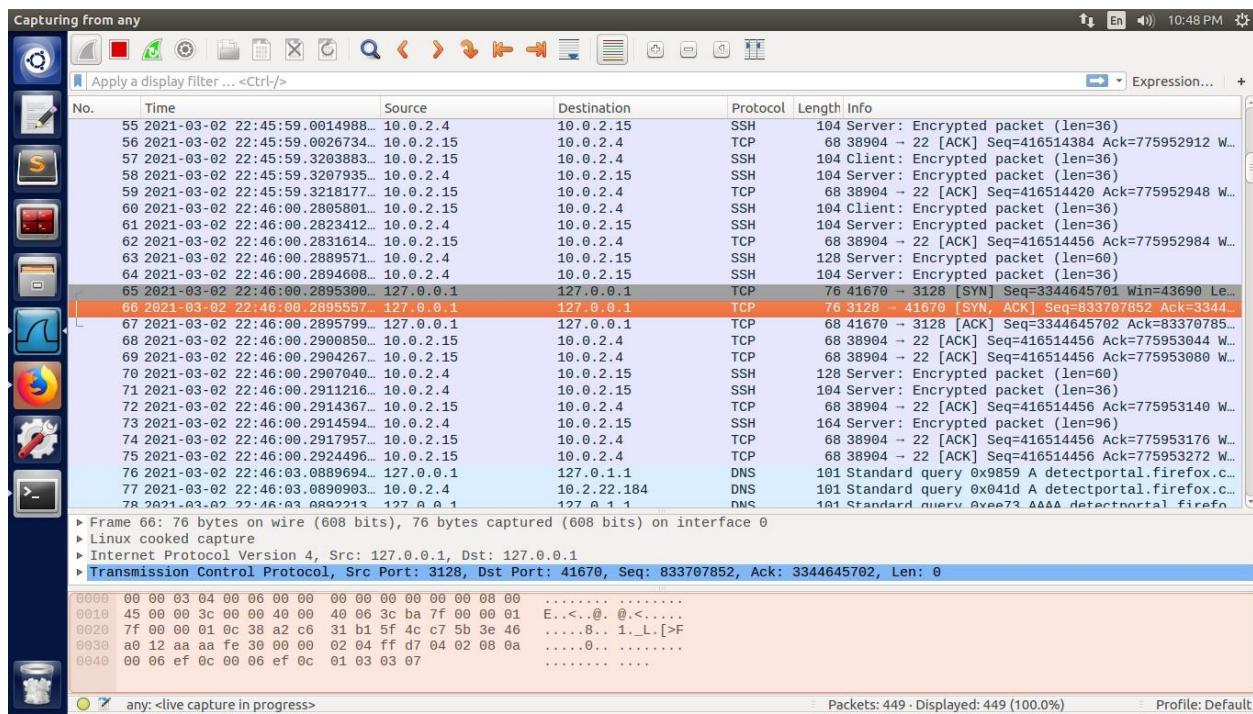
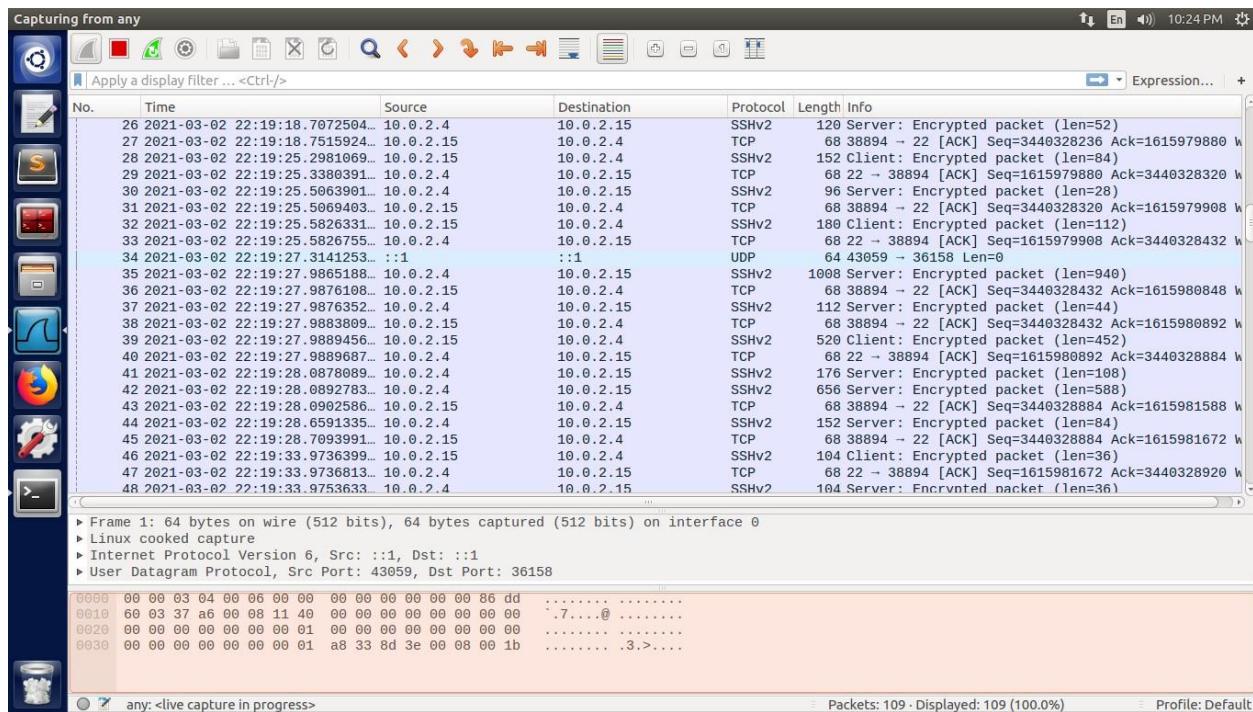
VM1 Wireshark capture.



VM2 Wireshark capture.



VM3 Wireshark capture.



### **Task 3.b: Connecting to Google using SSH tunnel**

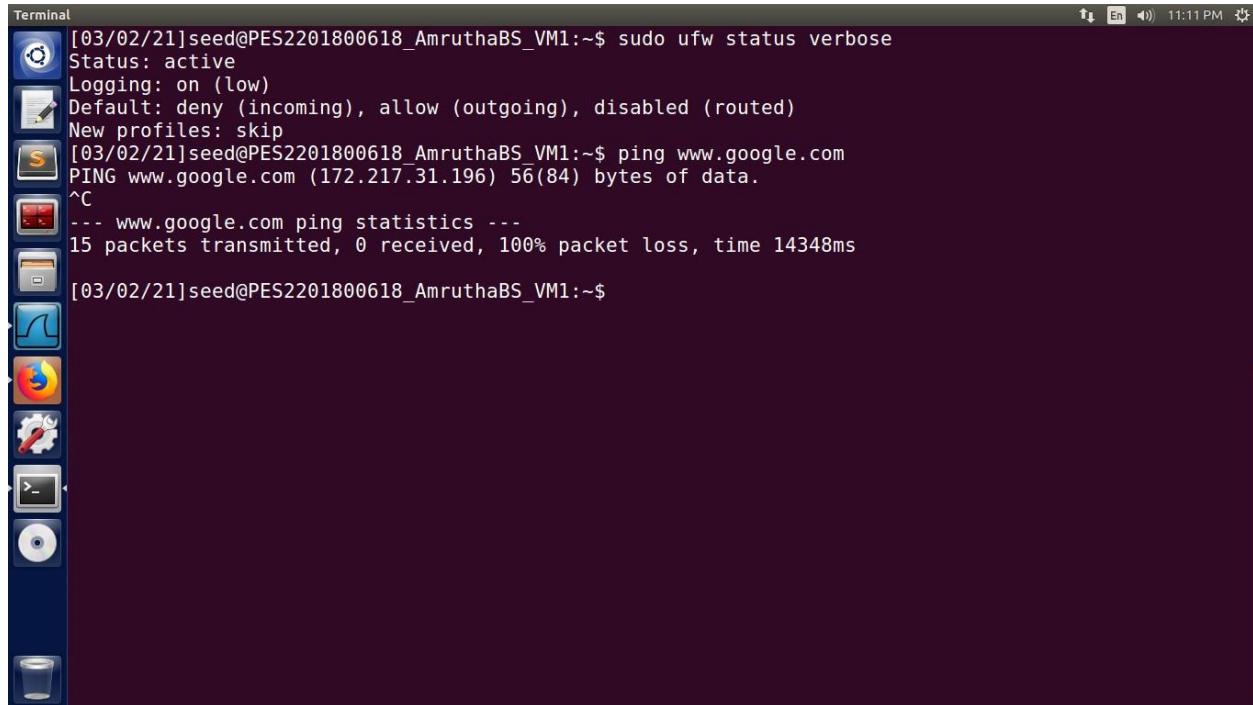
In this task, we will setup a firewall rule to block VM1 from visiting www.google.com but then leverage dynamic forwarding via ssh tunnel to visit www.google.com from VM1 via VM2.

We first find the IP address of www.google.com a shown below:

**Commands:**

**sudo ufw status verbose ping**

**www.google.com**



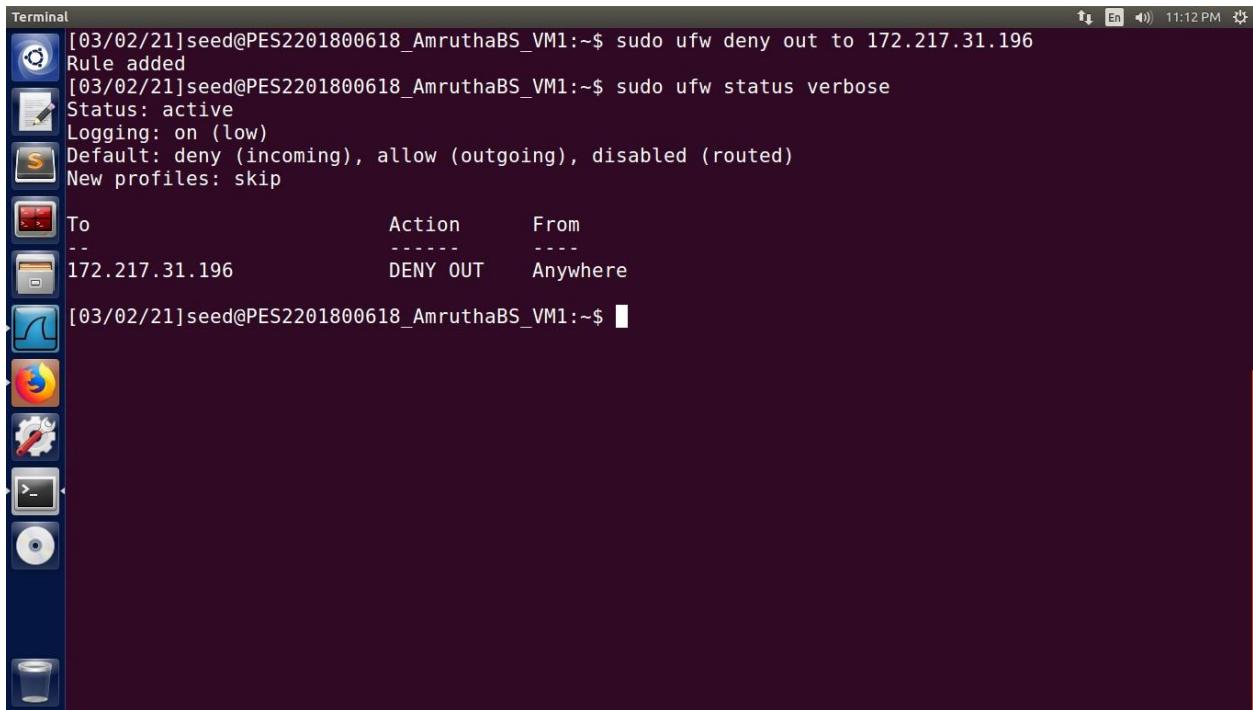
```
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ ping www.google.com
PING www.google.com (172.217.31.196) 56(84) bytes of data.
^C
--- www.google.com ping statistics ---
15 packets transmitted, 0 received, 100% packet loss, time 14348ms
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

Given the IP address, we can setup a firewall rule to block traffic to that IP.

**Command:**

**sudo ufw deny out to 172.217.31.196 sudo**

**ufw status verbose**

A screenshot of a Linux desktop environment, likely Ubuntu, showing a terminal window. The terminal window has a dark blue header bar with the word "Terminal" and a small gear icon. The main area of the terminal shows command-line output. The user has run the command "sudo ufw deny out to 172.217.31.196", which adds a new rule. Then, they run "sudo ufw status verbose" to see the current configuration, which shows the status as active, logging as on (low), and the default policy as deny (incoming), allow (outgoing), disabled (routed). A new profile named "skip" is listed. The output also includes a table of rules:

To	Action	From
--	-----	-----
172.217.31.196	DENY OUT	Anywhere

[03/02/21]seed@PES2201800618\_AmruthaBS\_VM1:~\$ █  
The desktop interface includes a dock on the left with icons for Dash, Home, Applications, and a terminal. A system tray at the top right shows battery level, signal strength, and the time (11:12 PM).

```
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw deny out to 172.217.31.196
Rule added
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

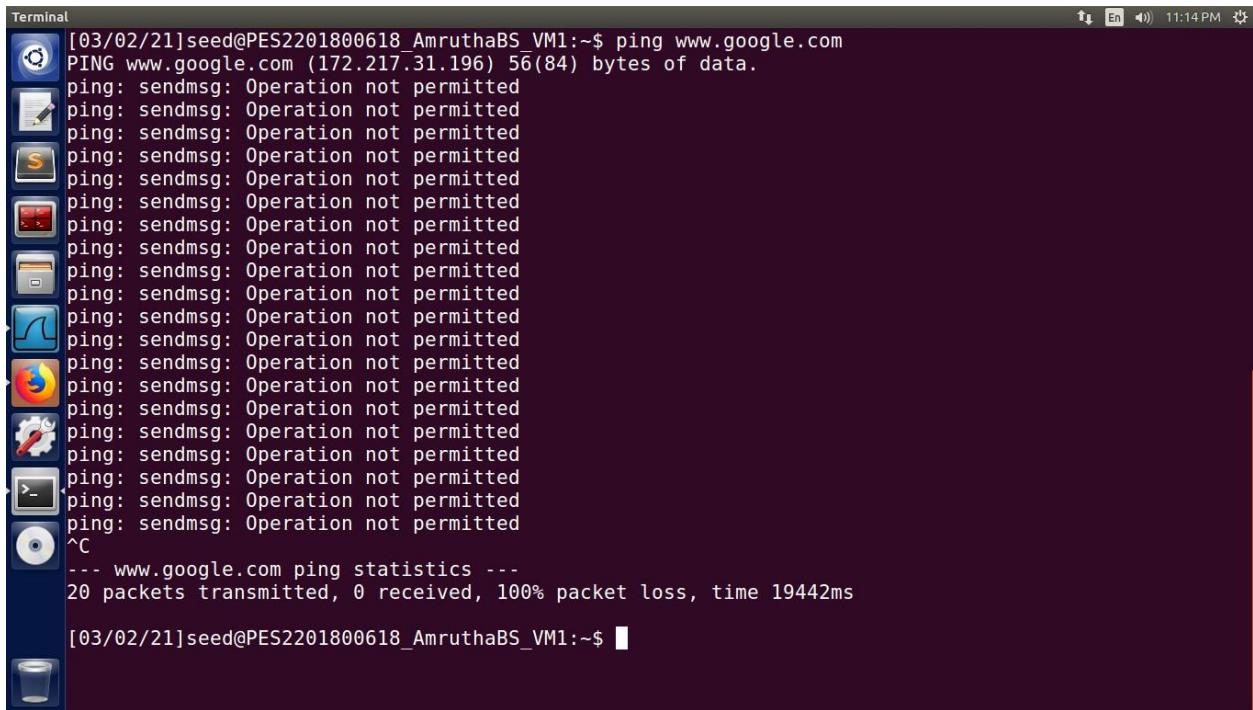
To           Action      From
--           -----      -----
172.217.31.196  DENY OUT  Anywhere

[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ █
```

With the firewall rule in place, we can try to ping www.google.com. The operation is not permitted because it is being blocked by the firewall.

**Command:**

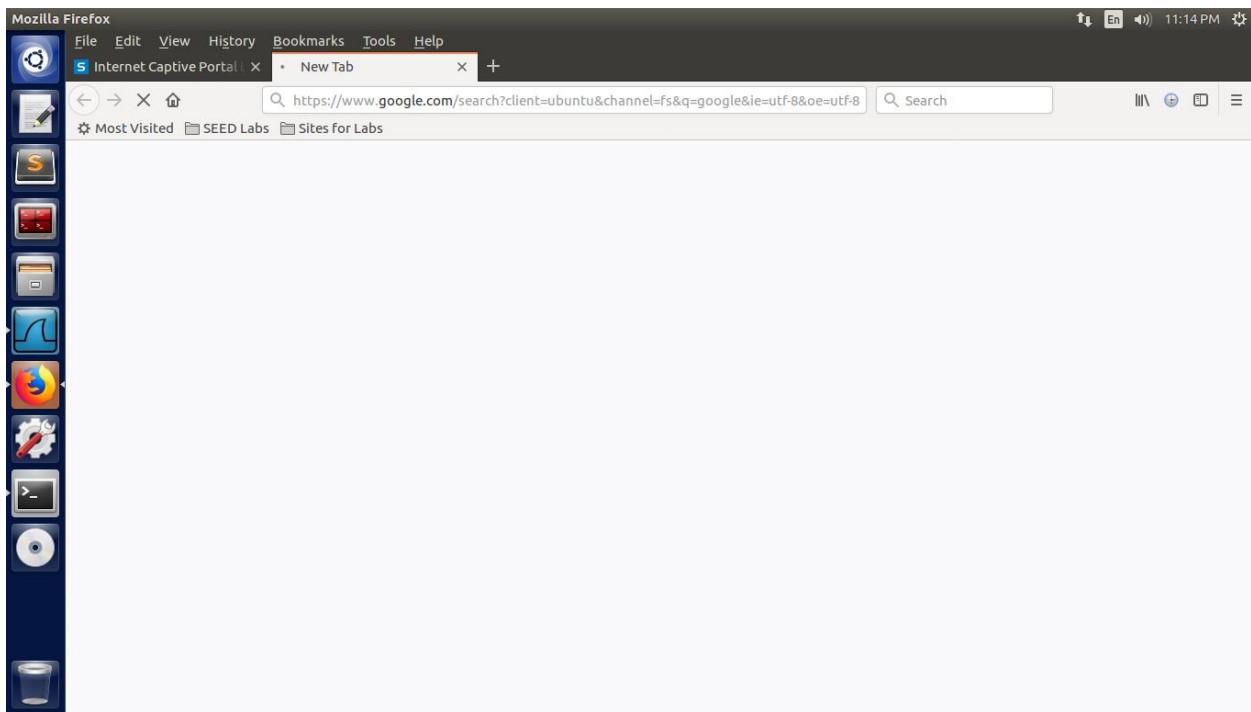
**ping www.google.com**

A screenshot of an Ubuntu desktop environment. In the top right corner, there is a system tray with icons for battery, signal strength, and time (11:14 PM). Below the tray, a terminal window is open with the following command and output:

```
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ ping www.google.com
PING www.google.com (172.217.31.196) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
^C
--- www.google.com ping statistics ---
20 packets transmitted, 0 received, 100% packet loss, time 19442ms
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

The terminal window has a dark background with light-colored text. To the left of the terminal, there is a vertical dock containing icons for various applications like a text editor, file manager, and browser.

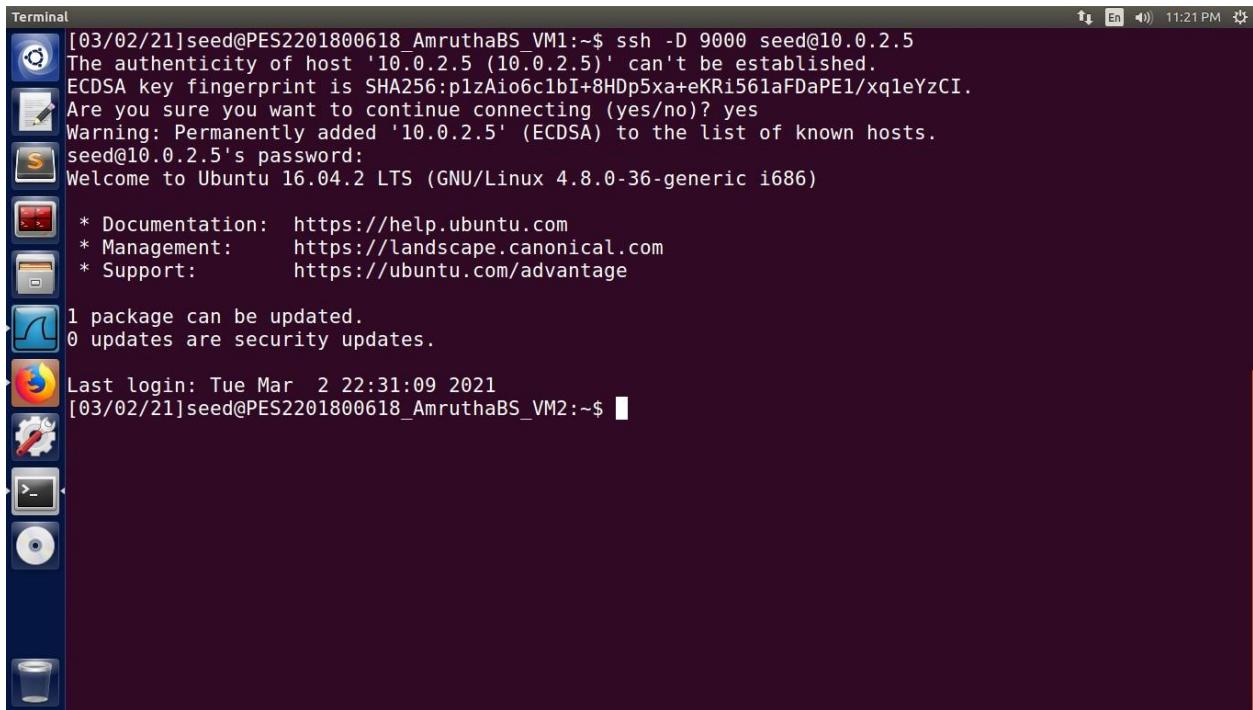
We can also try to visit [www.google.com](http://www.google.com) through the browser, where we are unable to load the page.



We now setup a ssh tunnel with dynamic port forwarding between VM1 and VM2. With this tunnel setup, VM1 will be able to use its local port 9000 to send a request to [www.google.com](http://www.google.com) via VM2.

**Command:**

```
ssh -D 9000 seed@10.0.2.11
```

A screenshot of an Ubuntu desktop environment. On the left, there's a vertical dock containing icons for various applications like Dash, Home, and System Settings. The main area is a terminal window titled "Terminal" with the following content:

```
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ ssh -D 9000 seed@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:plzAio6clbI+8HDp5xa+eKRi561aFDaPE1/xqleYzCI.
Are you sure you want to continue connecting (yes/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

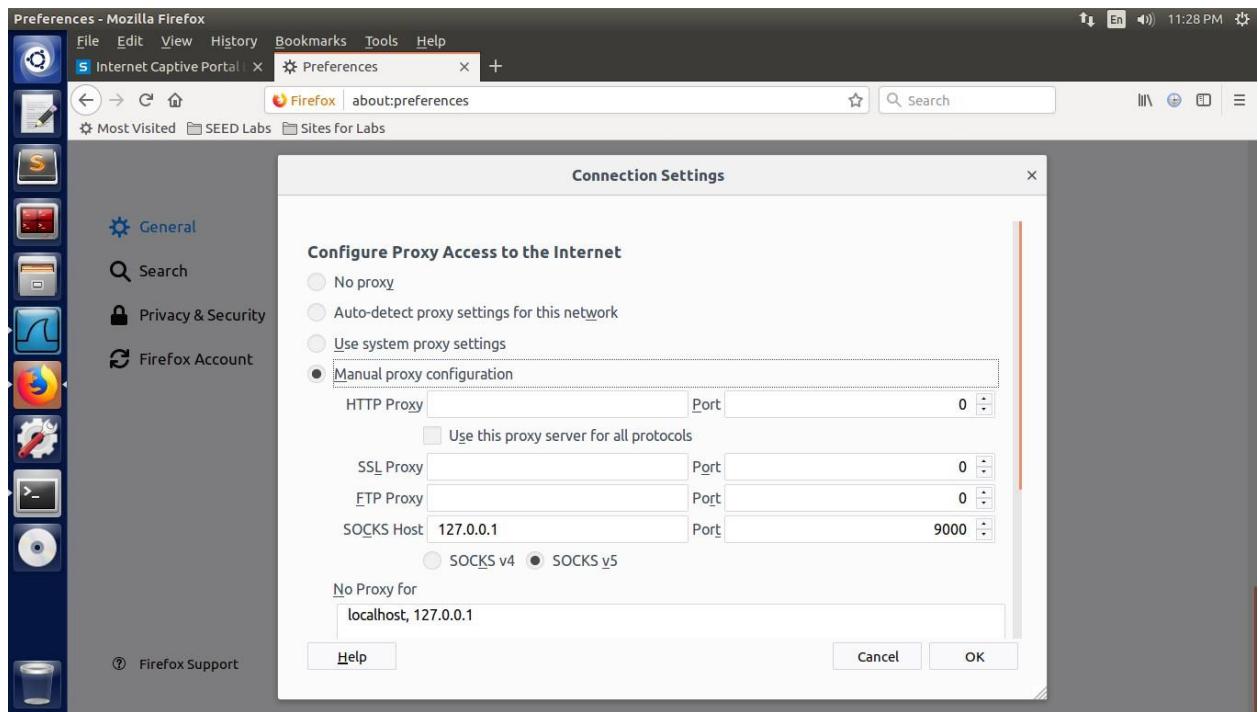
1 package can be updated.
0 updates are security updates.

Last login: Tue Mar  2 22:31:09 2021
[03/02/21]seed@PES2201800618_AmruthaBS_VM2:~$
```

To use the established tunnel, we need to set the proxy settings in the firefox browser as shown below. With the proxy settings in place, we are able to visit [www.google.com](http://www.google.com) from VM1 even though the firewall has a policy to block it. We next disable the tunnel to see its effect.

**Command:**

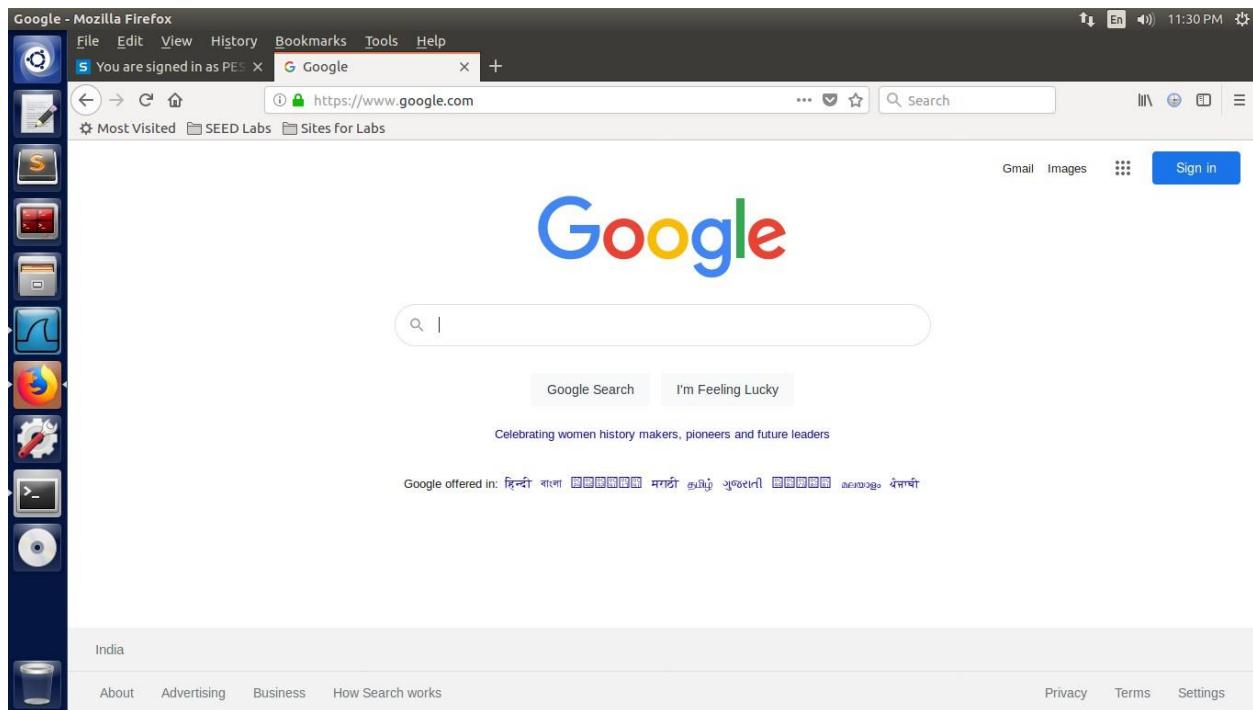
**ssh -D 9000 seed@10.0.2.11**



We also clear the browser cache to make sure firefox does not show the webpage it just loaded.

**Command:**

**ssh -D 9000 seed@10.0.2.11**

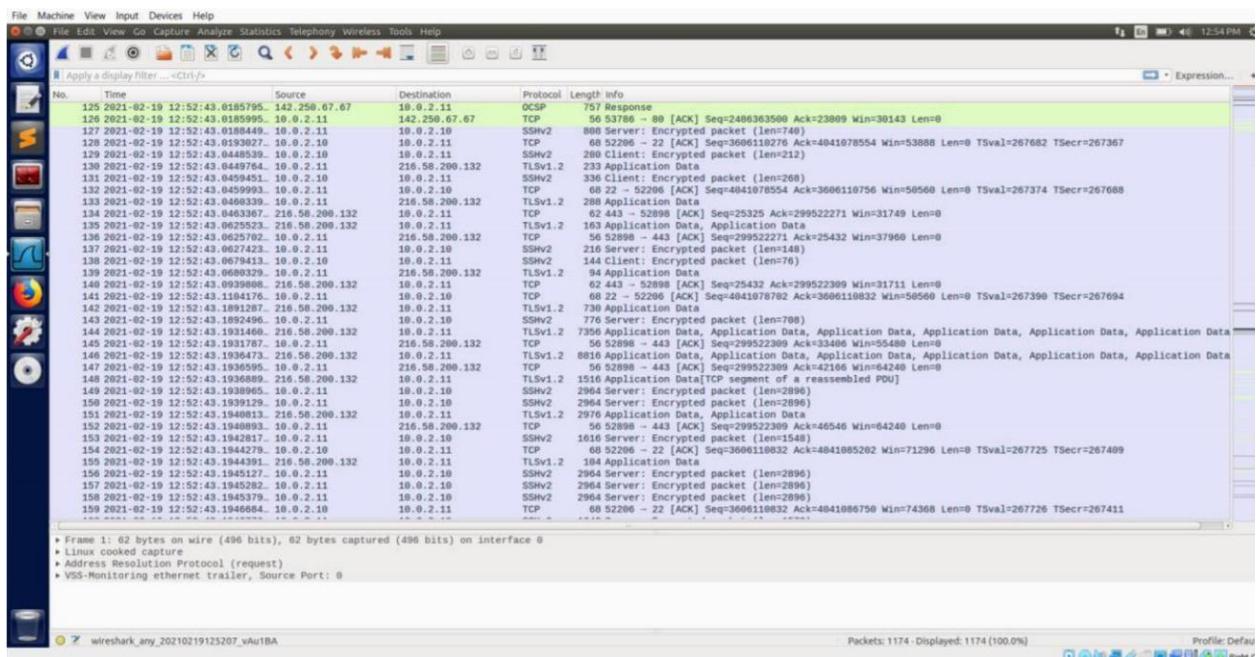


## VM1 Wireshark capture.

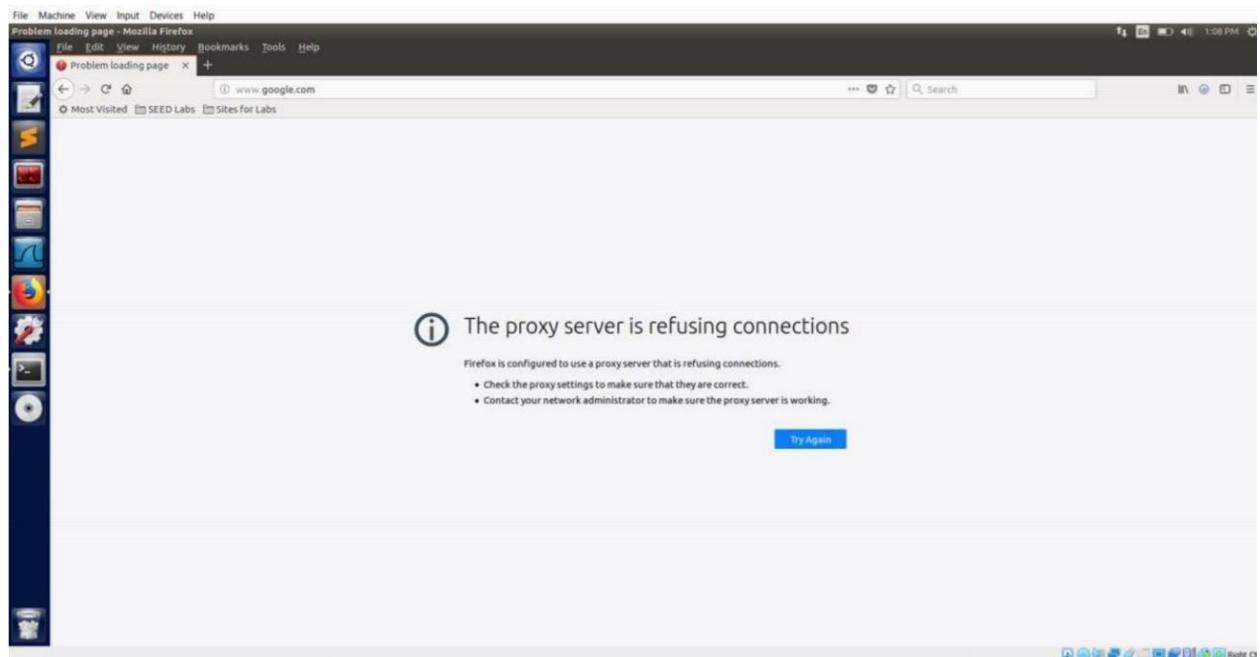
No.	Time	Source	Destination	Protocol	Length	Info
7	2021-02-19 12:52:16.6437468...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061095619 Ack=4041068809 Win=29312 Len=0 TSval=261032 TSecr=266717
8	2021-02-19 12:52:16.6441404...	10.0.2.10	10.0.2.11	SSHV2	100	Client: Protocol (SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.2)
9	2021-02-19 12:52:16.6444090...	10.0.2.11	10.0.2.10	TCP	60	22 - 52206 [ACK] Seq=40410688099 Ack=3606105660 Win=29056 Len=0 TSval=200717 TSecr=261032
10	2021-02-19 12:52:17.5252930...	10.0.2.11	10.0.2.10	SSHV2	100	Server: Protocol (SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.2)
11	2021-02-19 12:52:17.5252956...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061056609 Ack=4041068850 Win=29312 Len=0 TSval=261252 TSecr=266937
12	2021-02-19 12:52:17.5253055...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061056609 Ack=4041068850 Win=29312 Len=0 TSval=261252 TSecr=266937
13	2021-02-19 12:52:17.5253078...	10.0.2.11	10.0.2.10	TCP	60	52206 - 22 [ACK] Seq=36061056609 Ack=4041068850 Win=29312 Len=0 TSval=261252 TSecr=266937
14	2021-02-19 12:52:17.5470695...	10.0.2.11	10.0.2.10	SSHV2	1844	Server: Key Exchange Init
15	2021-02-19 12:52:17.5499513...	10.0.2.10	10.0.2.11	SSHV2	110	Client: Diffie-Hellman Key Exchange Init
16	2021-02-19 12:52:17.5508054...	10.0.2.11	10.0.2.10	SSHV2	432	Server: Diffie-Hellman Key Exchange Reply, New Keys, Encrypted packet (len=84)
17	2021-02-19 12:52:17.6023060...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061070949 Ack=4041070199 Win=33152 Len=0 TSval=261272 TSecr=266946
18	2021-02-19 12:52:19.2981386...	PcsComp...	5e:7a:0d	ARP	44	Who has 10.0.2.37 Tell 10.0.2.10
19	2021-02-19 12:52:19.8011829...	10.0.2.10	10.0.2.11	ARP	64	Request Who has 10.0.2.37 Tell 10.0.2.10
20	2021-02-19 12:52:19.8011829...	10.0.2.10	10.0.2.11	SSHV2	84	Client: New Keys
21	2021-02-19 12:52:19.8427116...	10.0.2.11	10.0.2.10	TCP	60	22 - 52206 [ACK] Seq=40410701999 Ack=36061070609 Win=31872 Len=0 TSval=261517 TSecr=261821
22	2021-02-19 12:52:20.5235955...	10.0.2.10	10.0.2.11	SSHV2	112	Client: Encrypted packet (len=44)
23	2021-02-19 12:52:20.5246591...	10.0.2.11	10.0.2.10	TCP	60	22 - 52206 [ACK] Seq=40410701919 Ack=36061071049 Win=31872 Len=0 TSval=261687 TSecr=262002
24	2021-02-19 12:52:20.5247199...	10.0.2.11	10.0.2.10	SSHV2	112	Server: Encrypted packet (len=44)
25	2021-02-19 12:52:20.5247200...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061071050 Ack=4041070234 Win=33152 Len=0 TSval=262002 TSecr=261687
26	2021-02-19 12:52:20.5242233...	10.0.2.10	10.0.2.11	SSHV2	120	Client: Encrypted packet (len=60)
27	2021-02-19 12:52:20.56462113...	10.0.2.11	10.0.2.10	TCP	60	22 - 52206 [ACK] Seq=4041070234 Ack=36061071649 Win=31872 Len=0 TSval=261698 TSecr=262002
28	2021-02-19 12:52:20.6447170...	10.0.2.11	10.0.2.10	SSHV2	120	Server: Encrypted packet (len=52)
29	2021-02-19 12:52:20.6859828...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061071699 Ack=40410702866 Win=33152 Len=0 TSval=262043 TSecr=261717
30	2021-02-19 12:52:22.9322657...	10.0.2.10	10.0.2.11	SSHV2	152	Client: Encrypted packet (len=84)
31	2021-02-19 12:52:22.9322658...	10.0.2.11	10.0.2.10	TCP	60	52206 - 22 [ACK] Seq=40410702866 Ack=36061071809 Win=31872 Len=0 TSval=2622684 TSecr=262664
32	2021-02-19 12:52:22.9945074...	10.0.2.11	10.0.2.10	SSHV2	98	Server: Encrypted packet (len=20)
33	2021-02-19 12:52:22.9945726...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061072489 Ack=4041070314 Win=33152 Len=0 TSval=2622620 TSecr=262305
34	2021-02-19 12:52:23.0188820...	10.0.2.10	10.0.2.11	SSHV2	100	Client: Encrypted packet (len=112)
35	2021-02-19 12:52:23.0113207...	10.0.2.11	10.0.2.10	TCP	60	22 - 52206 [ACK] Seq=4041070314 Ack=36061073669 Win=31872 Len=0 TSval=262309 TSecr=262624
36	2021-02-19 12:52:25.5452535...	:1	:1	UDP	64	33281 - 521287 Len=9
37	2021-02-19 12:52:26.5620283...	10.0.2.11	10.0.2.10	SSHV2	1800	Server: Encrypted packet (len=948)
38	2021-02-19 12:52:26.5620284...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=40410702866 Ack=36061071809 Win=31872 Len=0 TSval=263597 TSecr=263181
39	2021-02-19 12:52:26.5429844...	10.0.2.11	10.0.2.10	SSHV2	112	Server: Encrypted packet (len=44)
40	2021-02-19 12:52:26.5430546...	10.0.2.10	10.0.2.11	TCP	60	52206 - 22 [ACK] Seq=36061073669 Ack=4041071298 Win=35672 Len=0 TSval=263507 TSecr=263192
41	2021-02-19 12:52:26.5434882...	10.0.2.10	10.0.2.11	SSHV2	520	Client: Encrypted packet (len=452)

Frame 1: 344 bytes on wire (2752 bits), 344 bytes captured (2752 bits) on interface 0  
 ▾ Linux cooked capture  
 ▾ Internet Protocol Version 4, Src: 10.0.2.10, Dst: 10.0.2.3  
 ▾ User Datagram Protocol, Src Port: 60, Dst Port: 67  
 ▾ Bootstrap Protocol (Request)

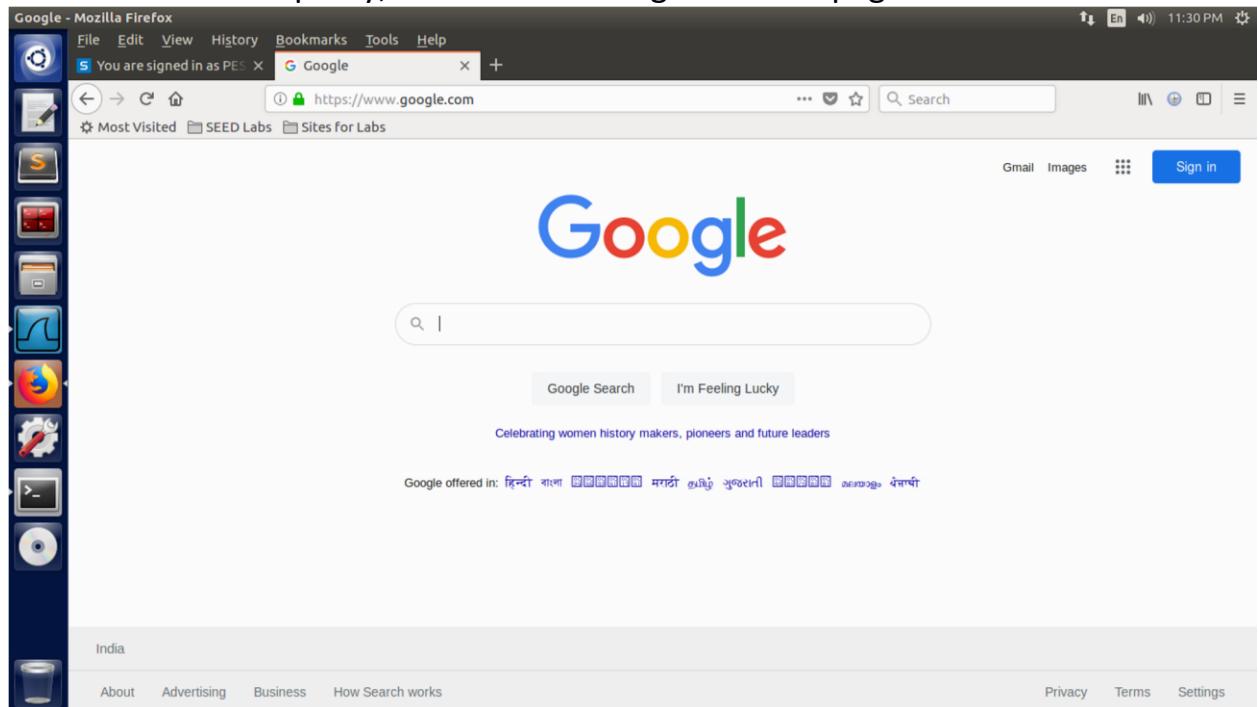
## VM2 Wireshark capture.



If we try to visit [www.google.com](http://www.google.com) again, we observe that the browser informs the proxy is refusing connections. The browser is still configured to use proxy, but with the tunnel not running any more, the browser cannot use local port 9000 to get the result.



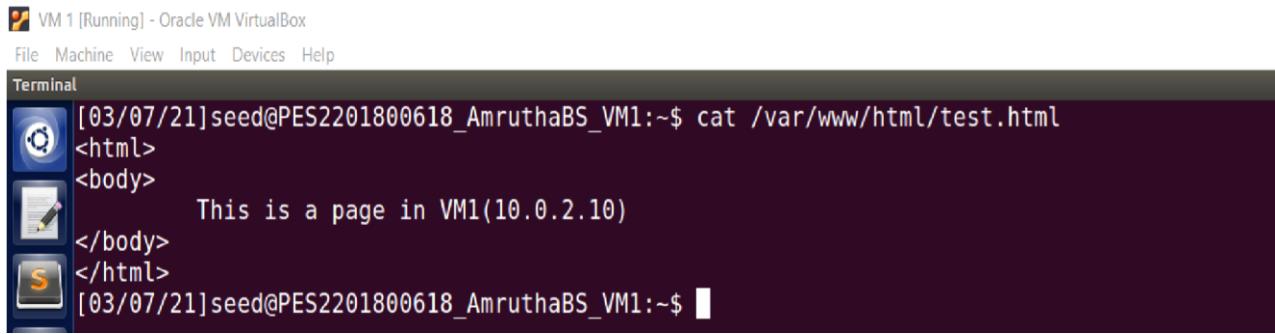
If we re-enable the proxy, the browser will get the webpage.



#### **Task 4: Evade Ingress Filtering**

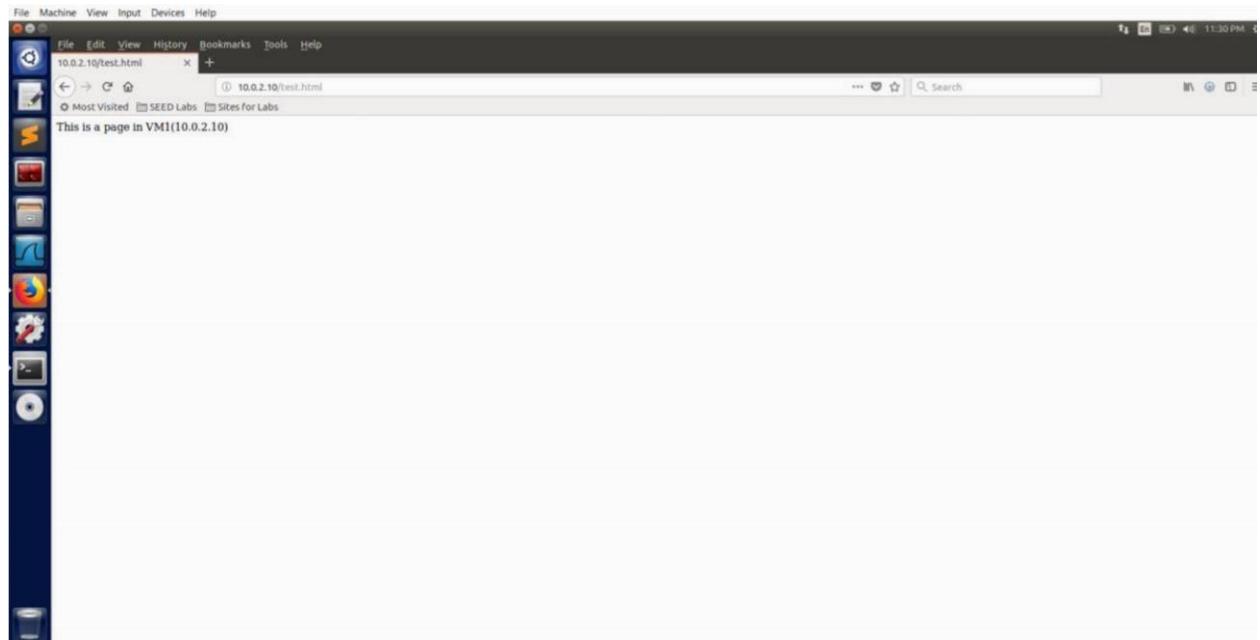
In this task, we will block incoming port 80 and port 22 on VM1, but still access a web page on the web server in VM1 from VM2 by using a reverse ssh tunnel.

Our goal is to access a secret page on VM1 (test.html) from VM2. The content of the page is shown below:



```
VM 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminal
[03/07/21]seed@PES2201800618_AmruthaBS_VM1:~$ cat /var/www/html/test.html
<html>
<body>
    This is a page in VM1(10.0.2.10)
</body>
</html>
[03/07/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

With no firewall rules setup on VM1, we try to access the page from VM2.



sudo ufw status verbose

```
Terminal [03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
[03/02/21]seed@PES2201800618_AmruthaBS_VM1:~$ █
```

A screenshot of a terminal window titled 'Terminal'. The command 'sudo ufw status verbose' is run, and the output shows the following configuration:

- Status: active
- Logging: on (low)
- Default: deny (incoming), allow (outgoing), disabled (routed)
- New profiles: skip

The terminal window has a dark background with white text and includes icons for file, terminal, and system status.

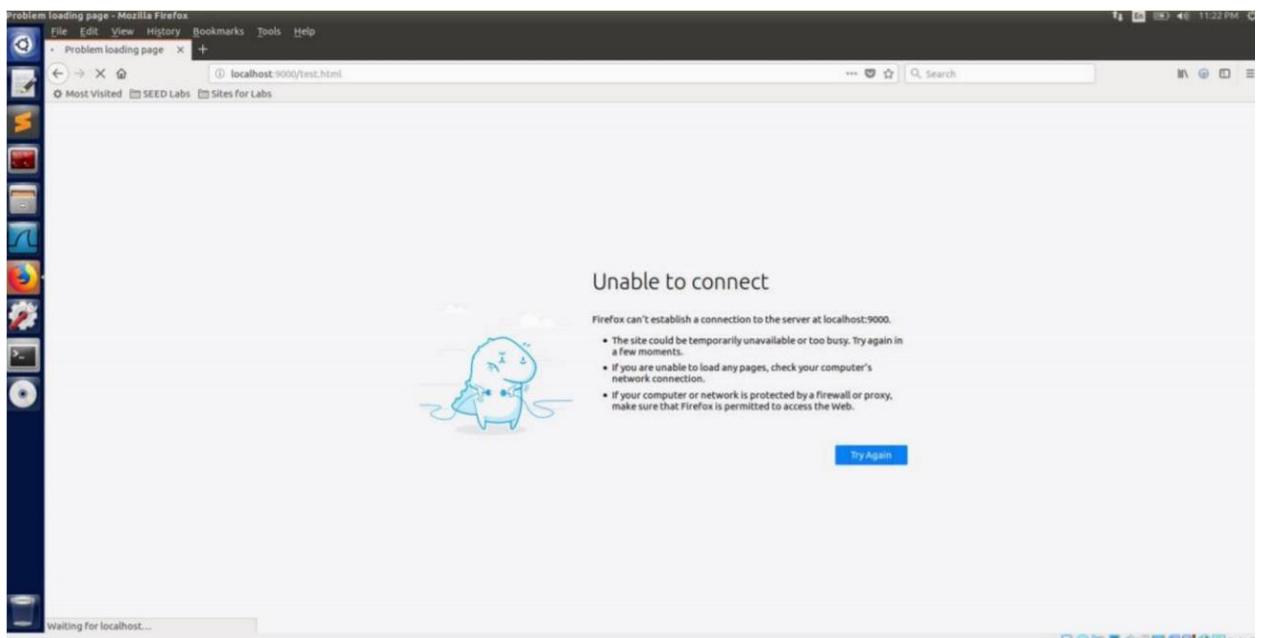
We block incoming requests on port 80 and port 22 on VM1

```
Terminal [03/03/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw deny in from any to 10.0.2.10 port 80
Rule added
[03/03/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw deny in from any to 10.0.2.10 port 22
Rule added
[03/03/21]seed@PES2201800618_AmruthaBS_VM1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

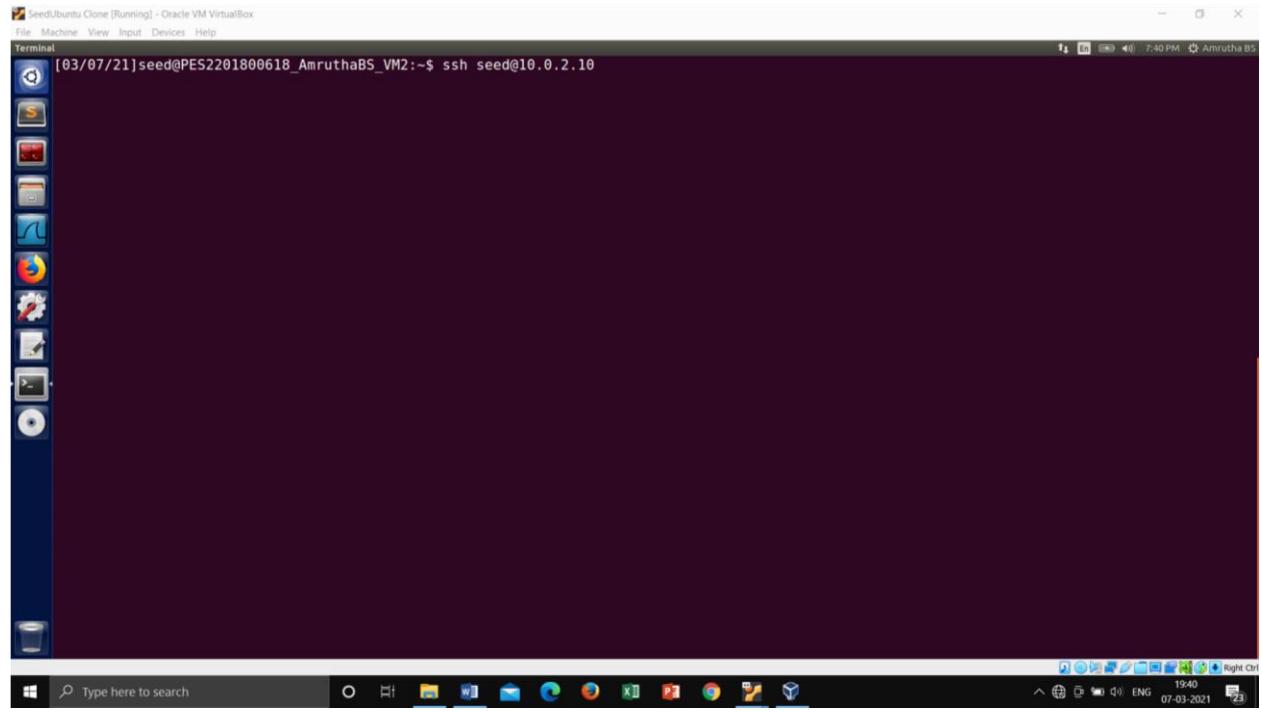
To           Action      From
--          ----       -----
10.0.2.10 80    DENY IN    Anywhere
10.0.2.10 22    DENY IN    Anywhere

[03/03/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

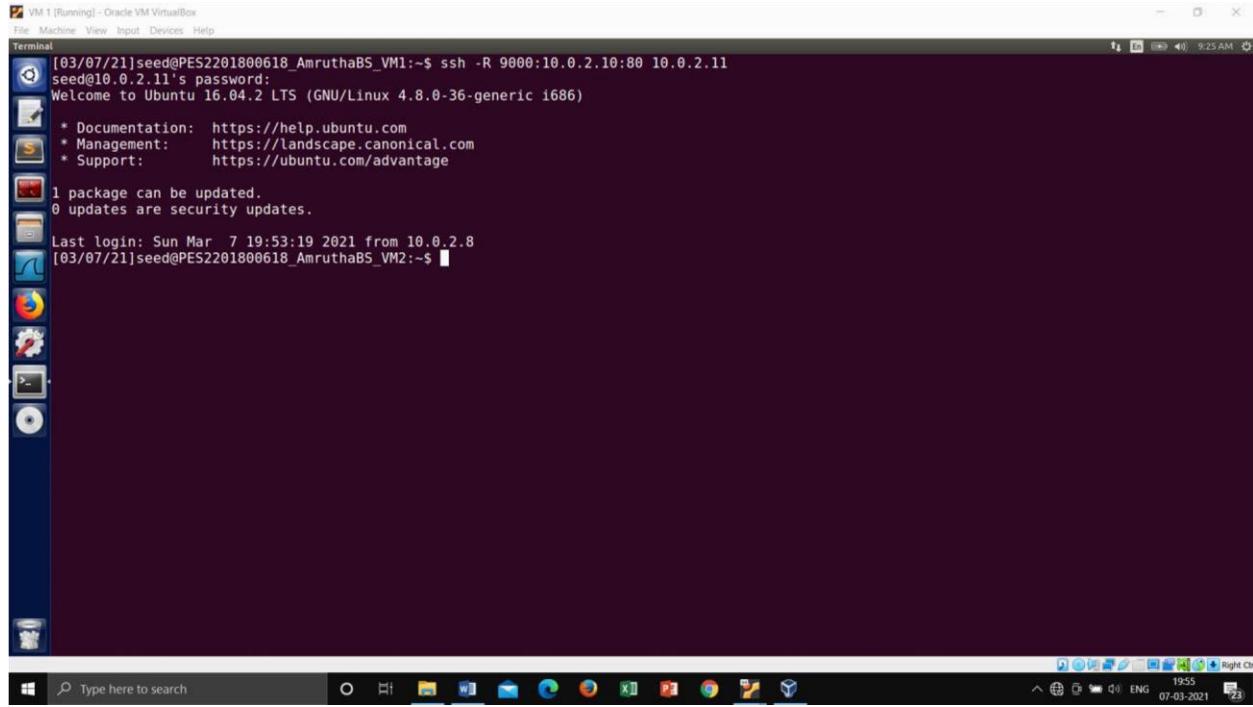
If we try to access the page again from VM2, the page is no longer accessible



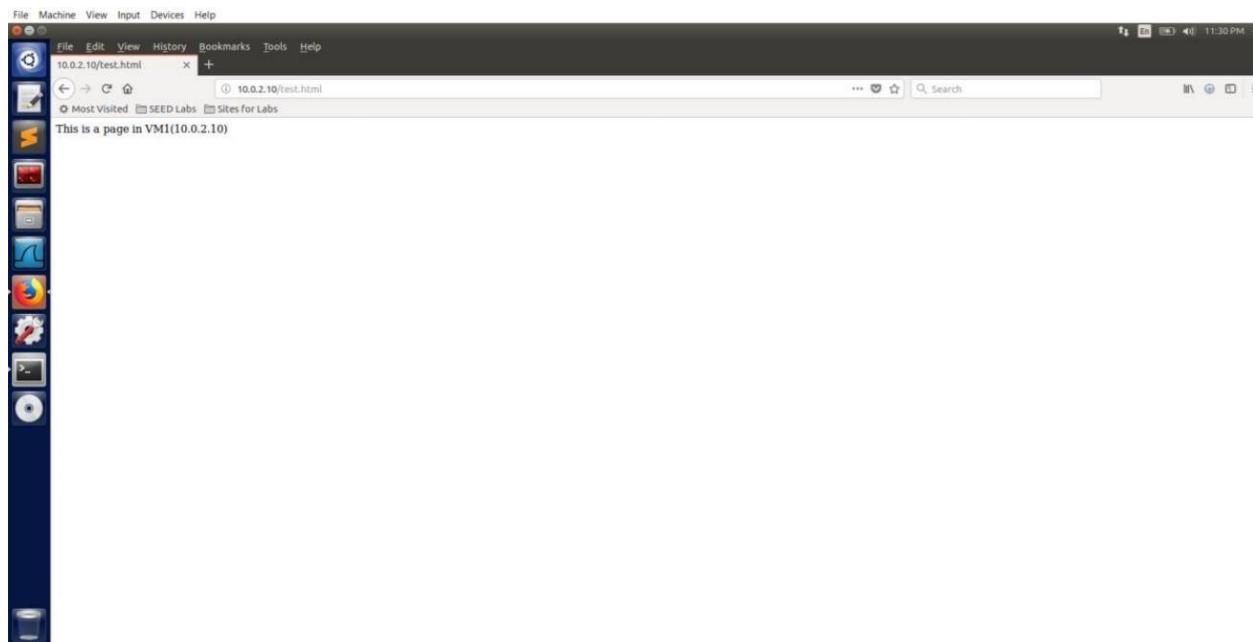
VM2 cannot ssh to VM1 because port 22 is blocked.



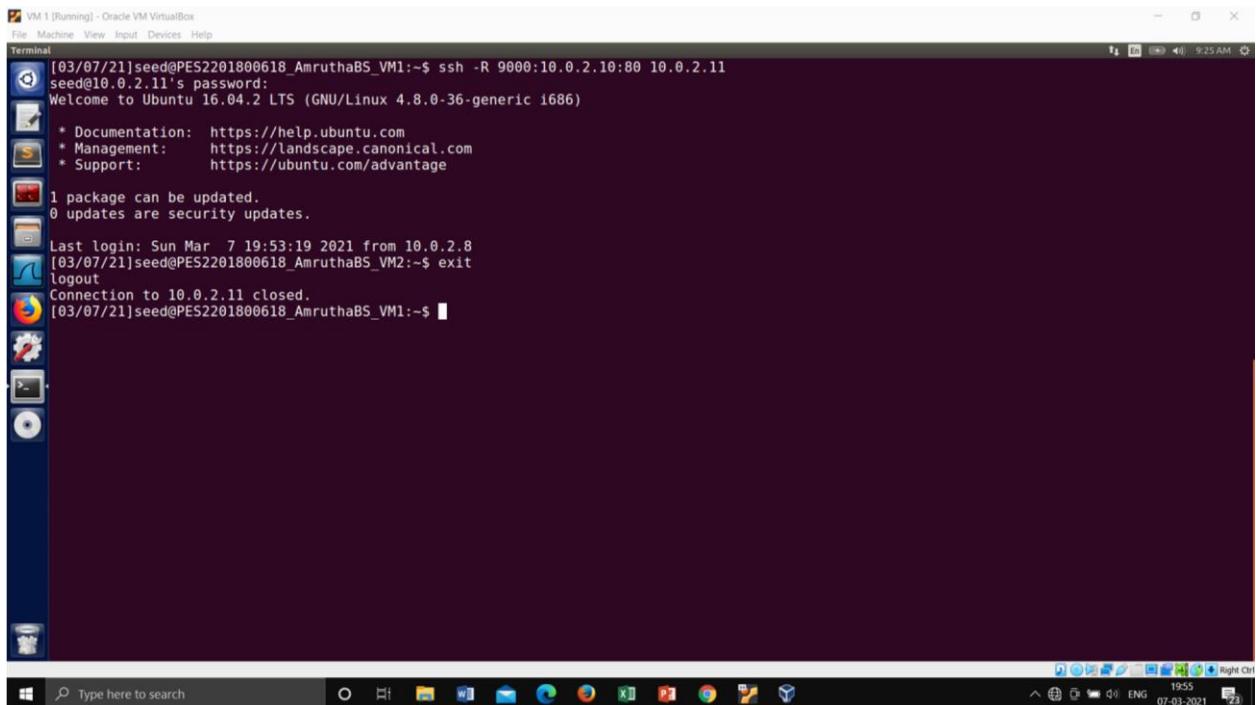
We set up a reverse tunnel. Using this VM2 can use its local port 9000 to access port 80 on VM1.



With the tunnel setup, we can access the webpage using port 9000 on VM2



## Breaking the tunnel by using exit command.



```
[03/07/21]seed@PES2201800618_AmruthaBS_VM1:~$ ssh -R 9000:10.0.2.10:80 10.0.2.11
seed@10.0.2.11'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: Sun Mar  7 19:53:19 2021 from 10.0.2.8
[03/07/21]seed@PES2201800618_AmruthaBS_VM2:~$ exit
logout
Connection to 10.0.2.11 closed.
[03/07/21]seed@PES2201800618_AmruthaBS_VM1:~$
```

With the tunnel broken, we can no longer access the page on VM1 from VM2

