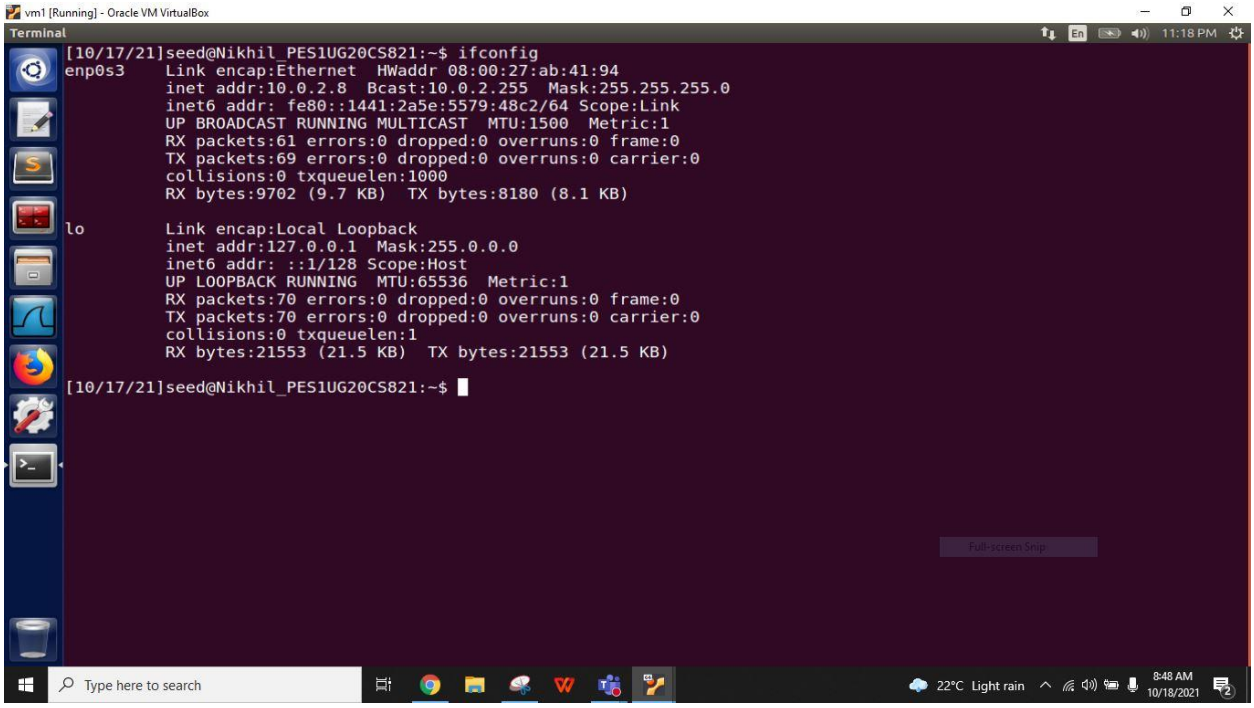
# Linux Firewall Exploration Lab

Name:Nikhil T M
SRN:PES1UG20CS821
Section:F

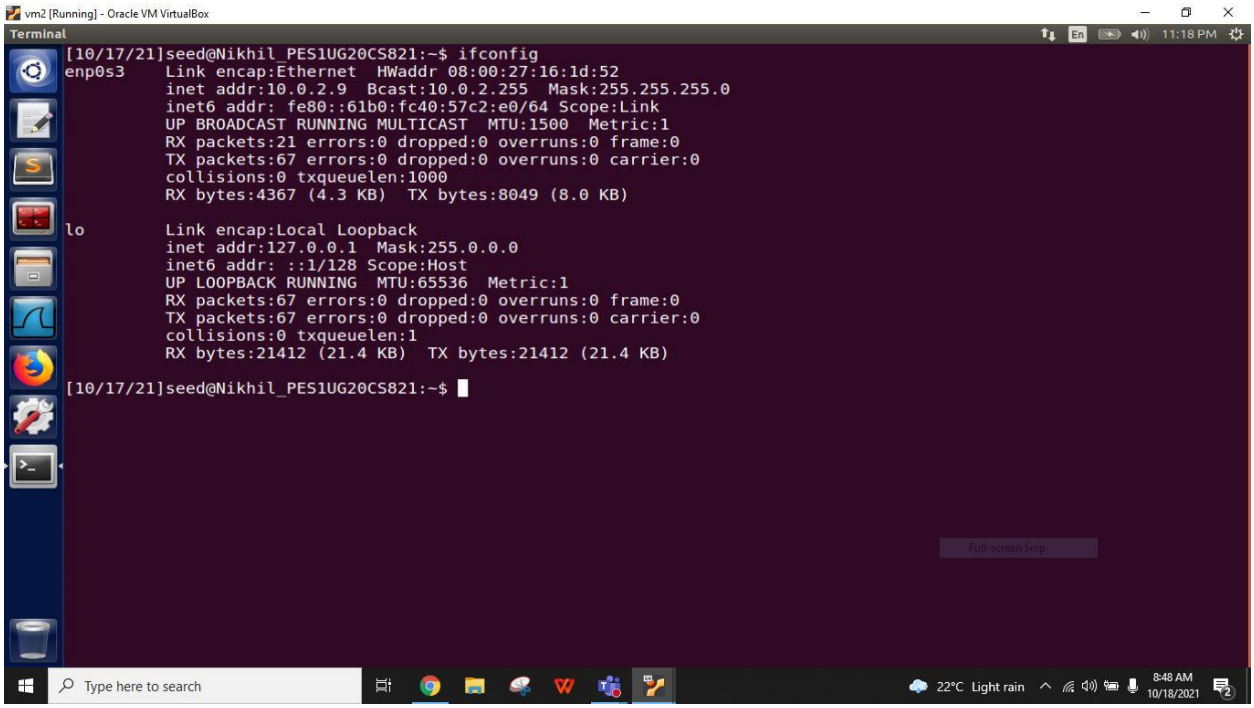
## Lab Setup

| | |
|---|---|
| Vm1 | 10.0.2.8 |
| Vm2 | 10.0.2.9 |
| Vm3 | 10.0.2.14 |

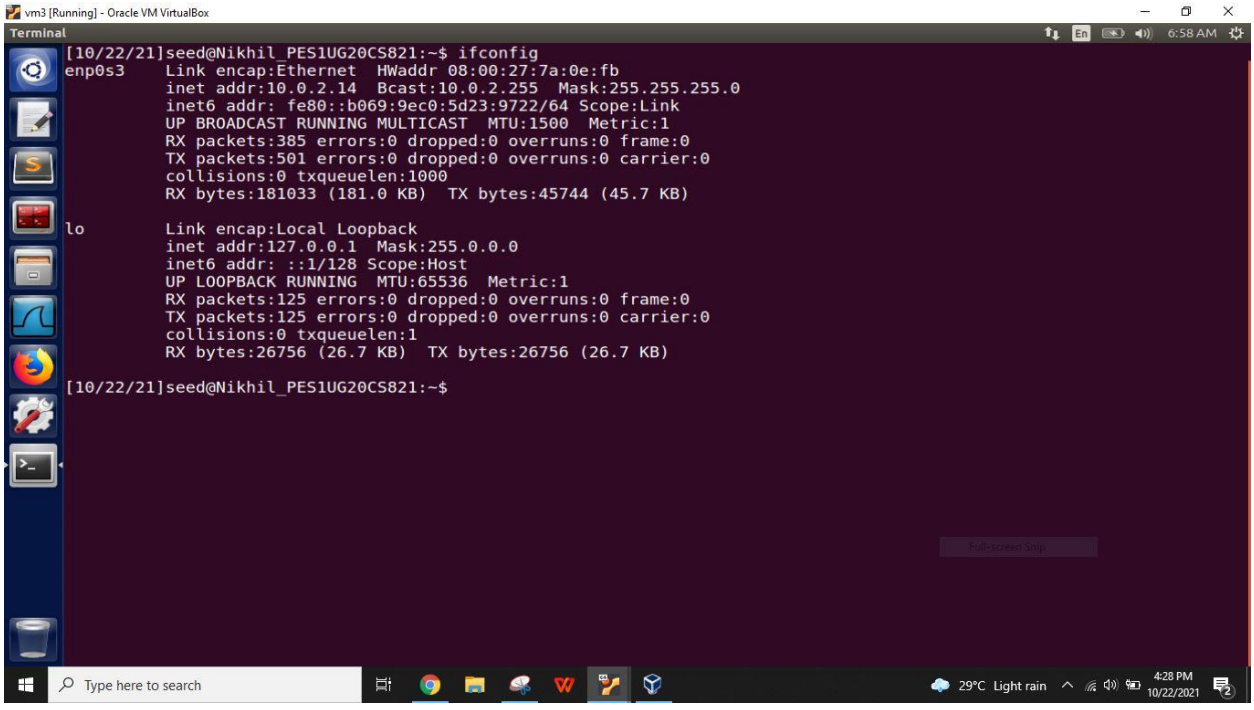
## Vm1



# Vm2

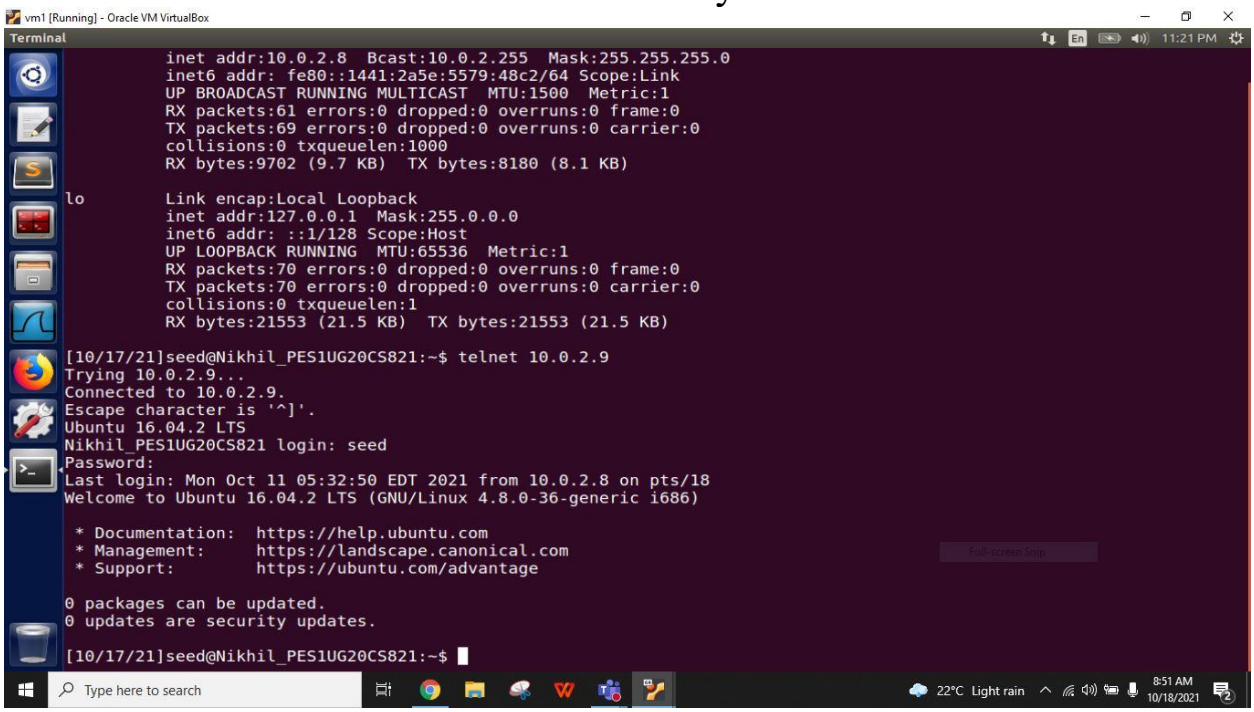

vm2 [Running] - Oracle VM VirtualBox
Terminal
11:18 PM
[10/17/21]seed@Nikhil_PES1UG20CS821:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:16:1d:52
inet addr:10.0.2.9 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::61b0:fc40:57c2:e0/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:21 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:4367 (4.3 KB) TX bytes:8049 (8.0 KB)
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets: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)
[10/17/21]seed@Nikhil_PES1UG20CS821:~$
Type here to search
22°C Light rain
8:48 AM
10/18/2021

# Vm3

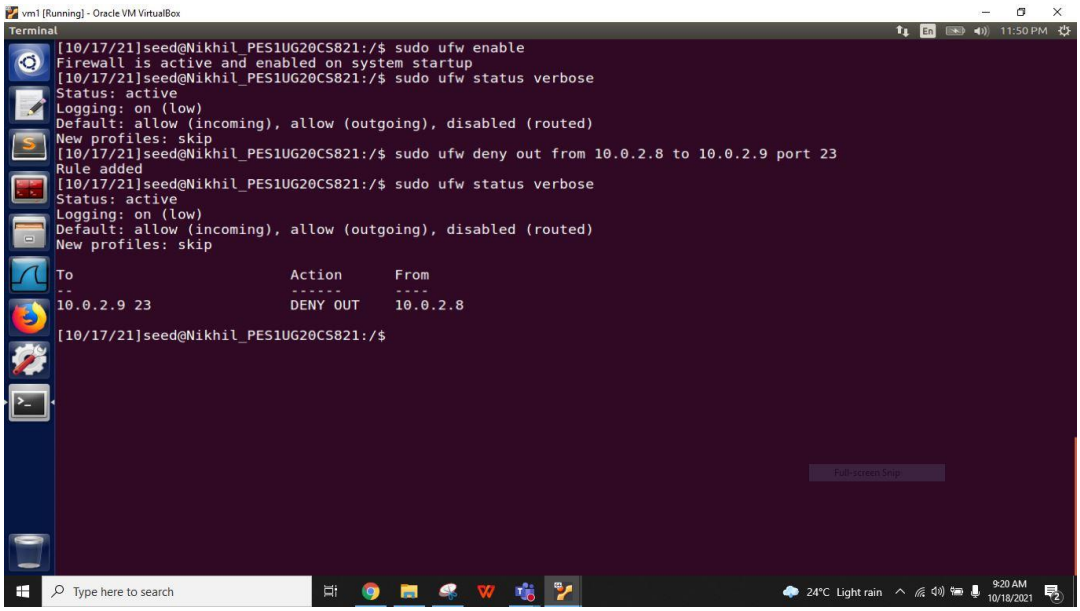

vm3 [Running] - Oracle VM VirtualBox
Terminal
6:58 AM
[10/22/21]seed@Nikhil_PES1UG20CS821:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:7a:0e:fb
inet addr:10.0.2.14 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::b069:9ec0:5d23:9722/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:385 errors:0 dropped:0 overruns:0 frame:0
TX packets:501 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:181033 (181.0 KB) TX bytes:45744 (45.7 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:125 errors:0 dropped:0 overruns:0 frame:0
TX packets:125 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:26756 (26.7 KB) TX bytes:26756 (26.7 KB)
[10/22/21]seed@Nikhil_PES1UG20CS821:~$
Type here to search
29°C Light rain
4:28 PM
10/22/2021

## Task 1: Using Firewall

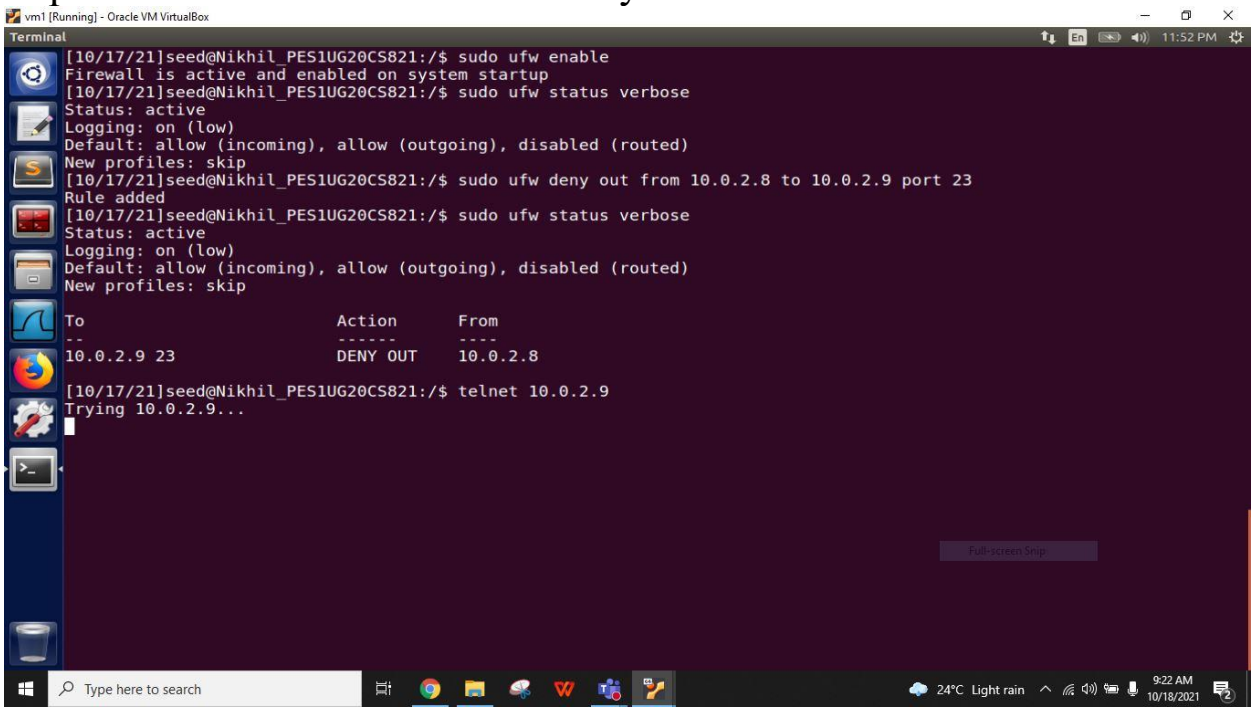
first we check whether we able to connect to vm2 from vm1 using telnet. We can see that we connected successfully.



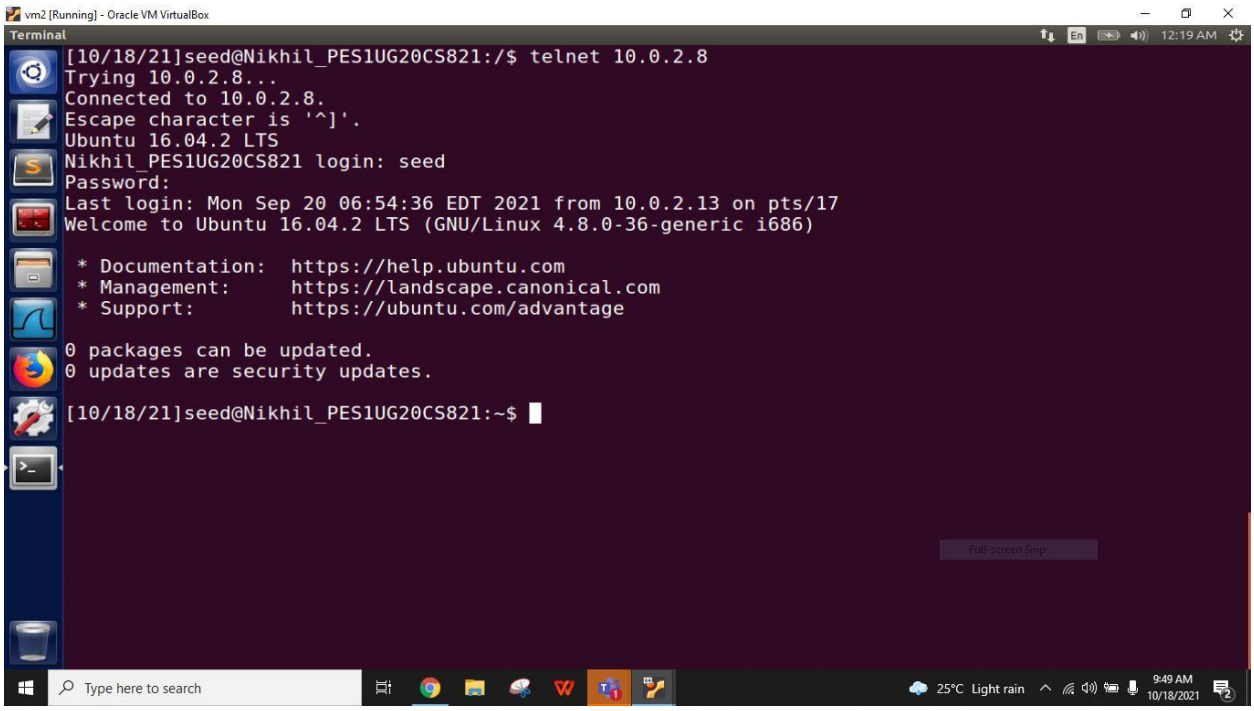

Now on vm1 we are going to enable the firewall and set an rule sucj that from port 23 that is telnet from vm1 to vm2 is going to block And we can see that the rules is successfully added by checking the status of the firewall



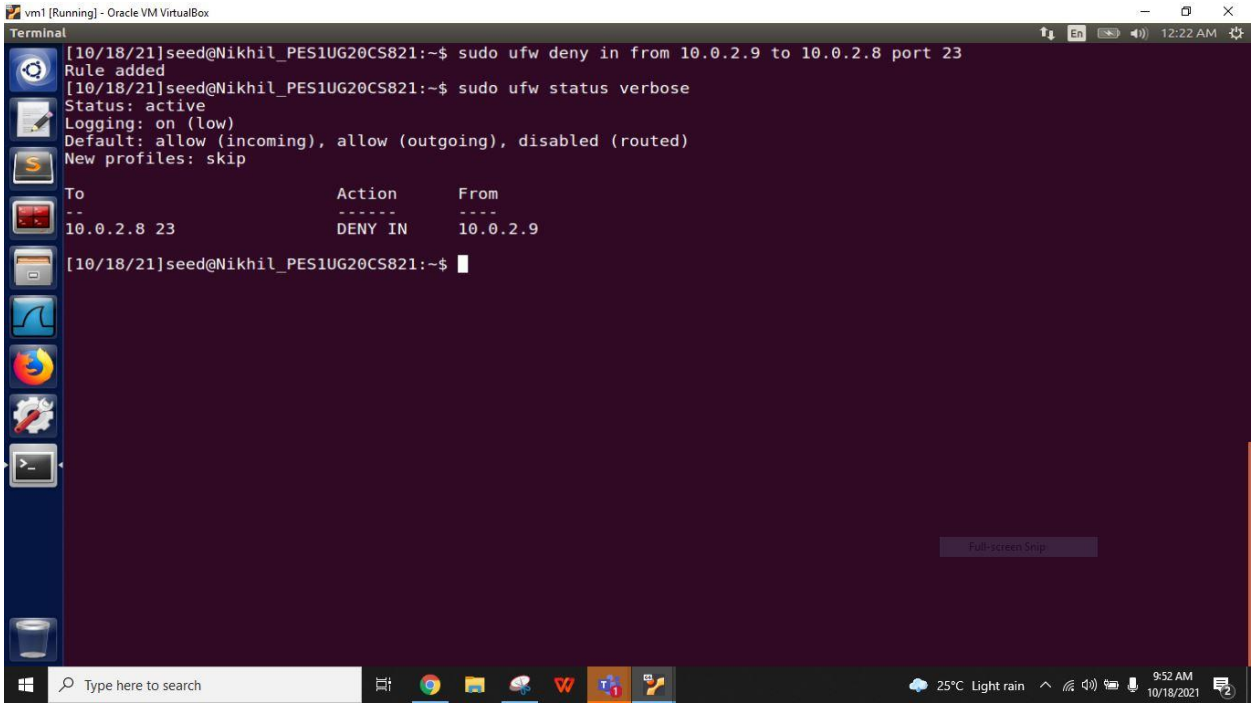
On we try to establish a connection using telnet from vm1 to vm2 which we blocked using firewall we can see that we are not able to connect that implies the rule is added successfully



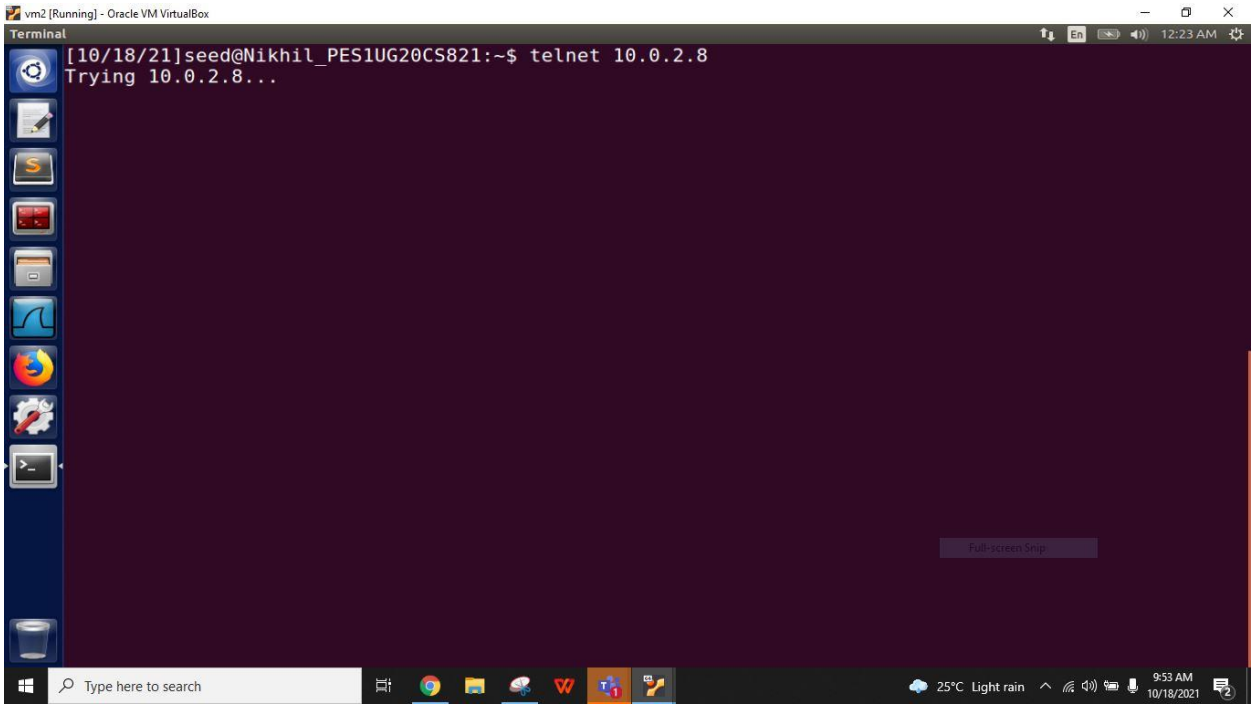

Now we go back to vm2 and establish a connection to vm1 using telnet and we can achieve this as rules only added from connecting vm1 to vm2 not vm2 to vm1



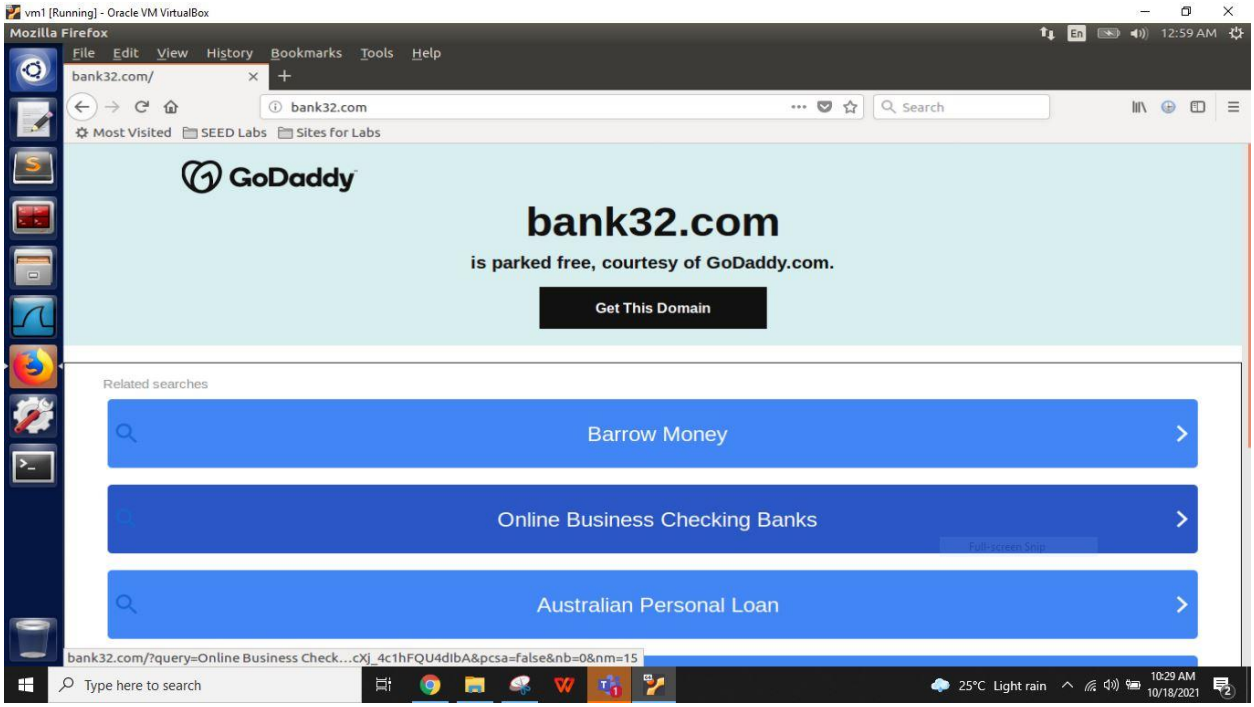
Now we add another rule in vm1 such that to block port 23 that is telnet from vm2 to vm1



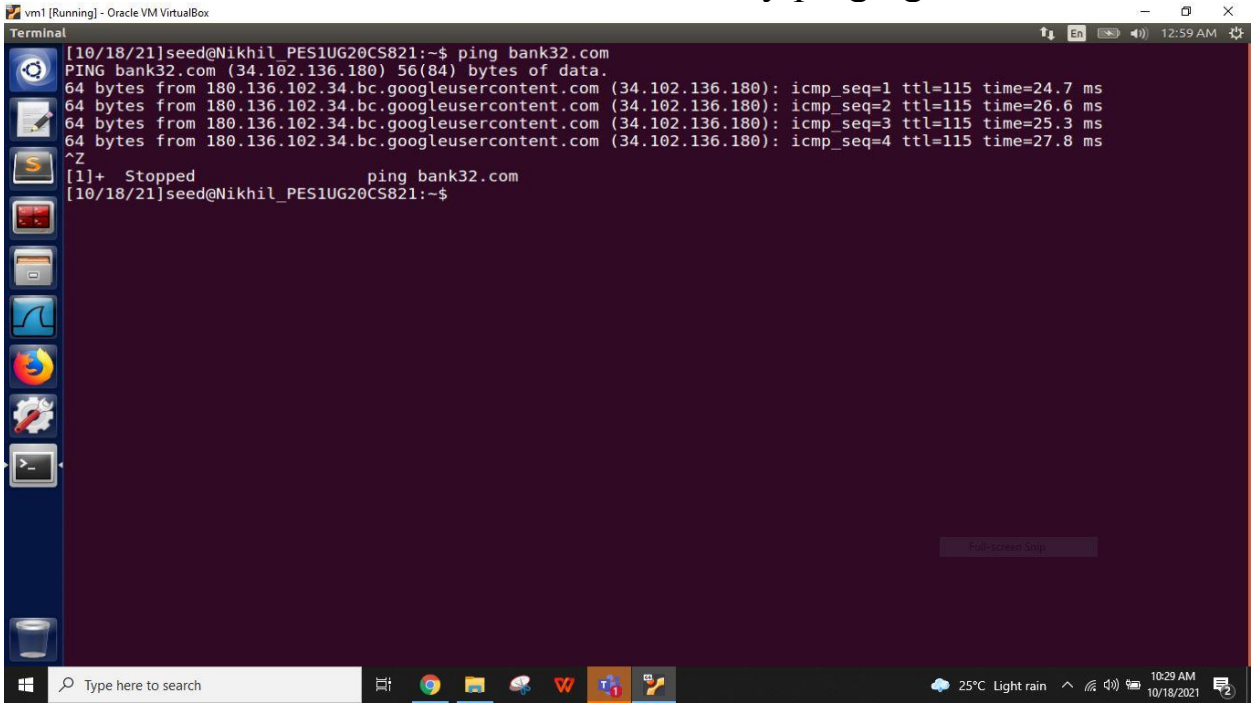

After adding the rule in vm1 now we go back to vm2 and try to connect to vm1 through telnet which is not possible as the telnet port 23 in vm1 in blocked



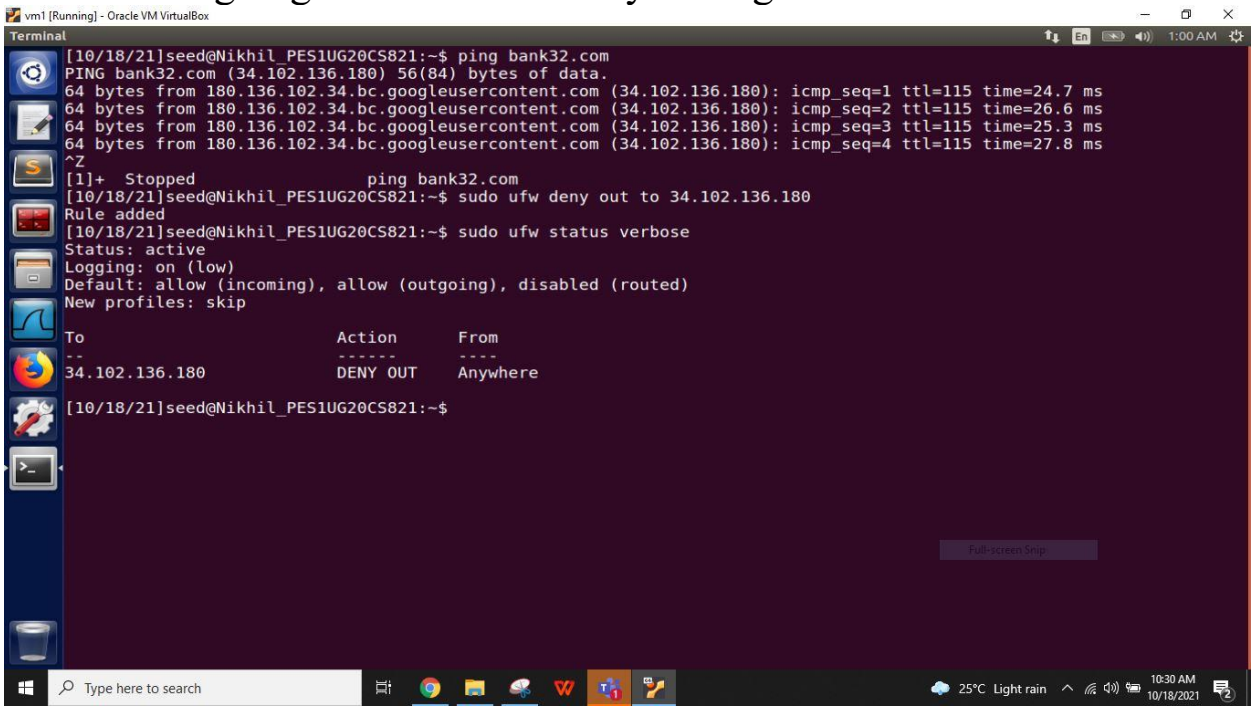
In this task first we are going to visit www.bank32.com in the vm1 browser and which is loaded in the browser of vm1



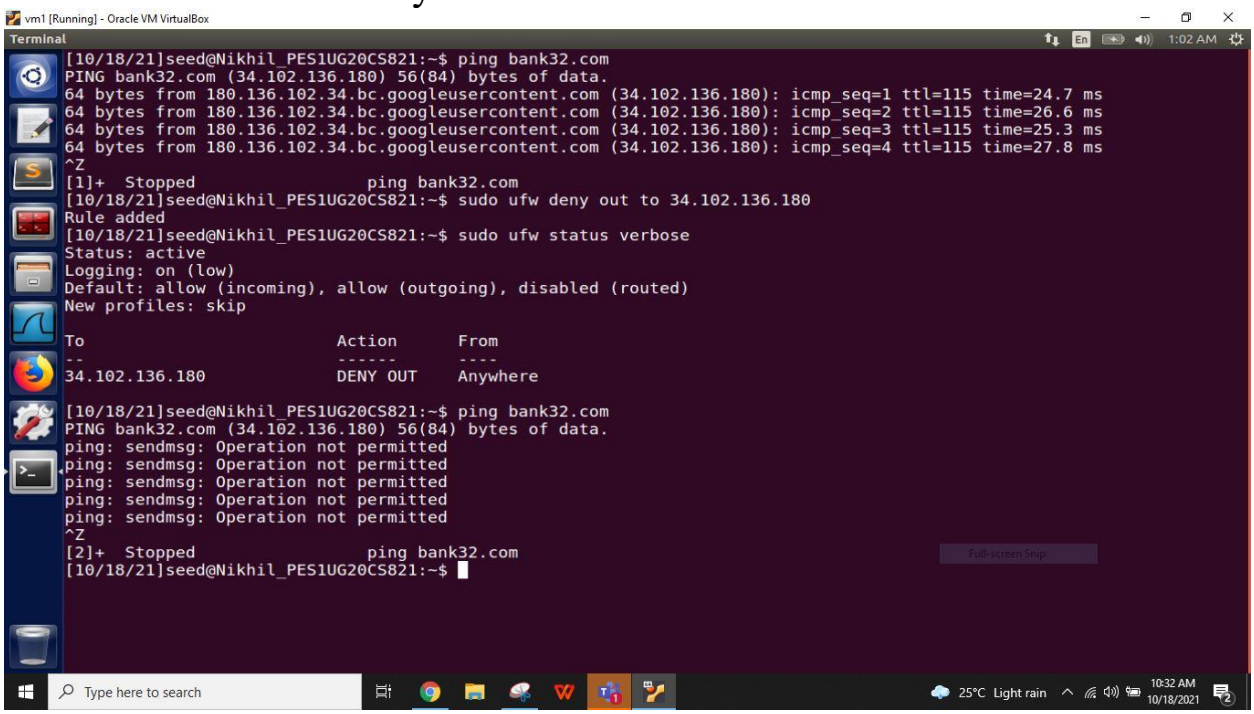

Now we obtain IP address of bank32.com by pinging it



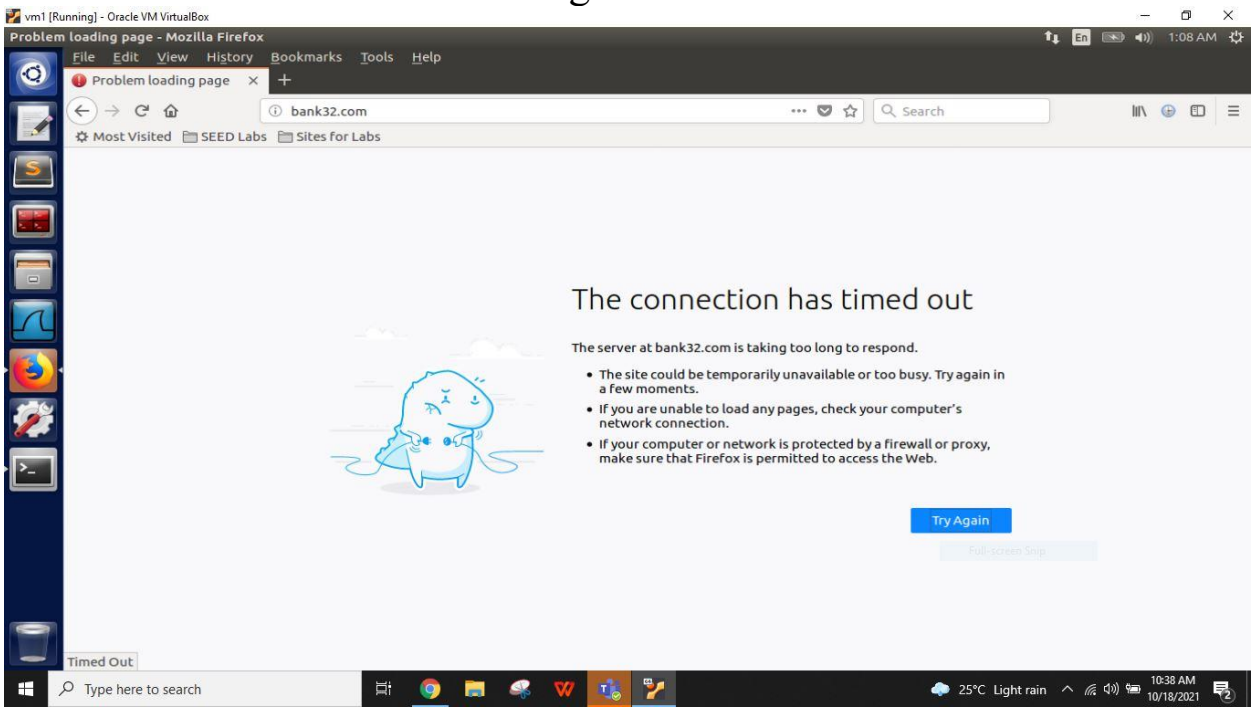
Now we block vm1 from visiting the site by adding the rule in the vm1.we are going to block the site by adding its IP address




After adding the rule and blocking it we are going to ping the same site and we can see that in terminal it shows operation not permitted this means the rule is successfully added and we blocked the site bank32.com successfully

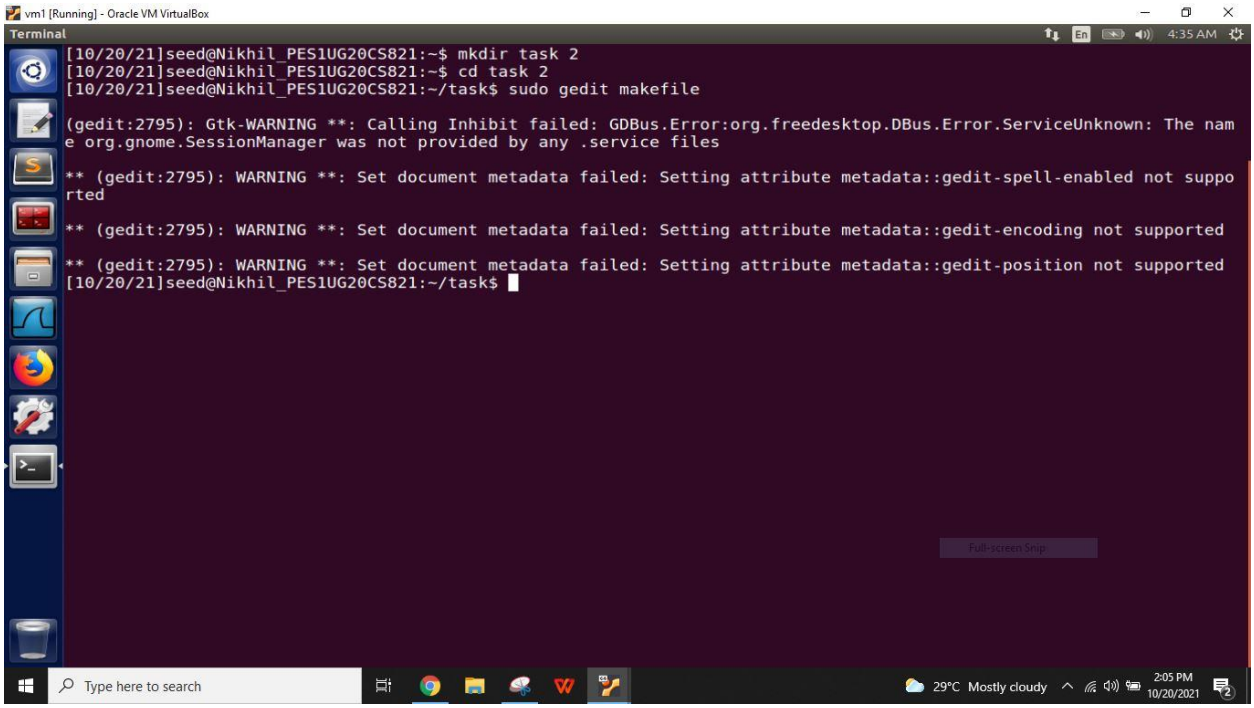


And we clear the cache and history in the web browser and going to access the site again in the browser and can see that it cant able to load it as that IP address is blocked using firewall

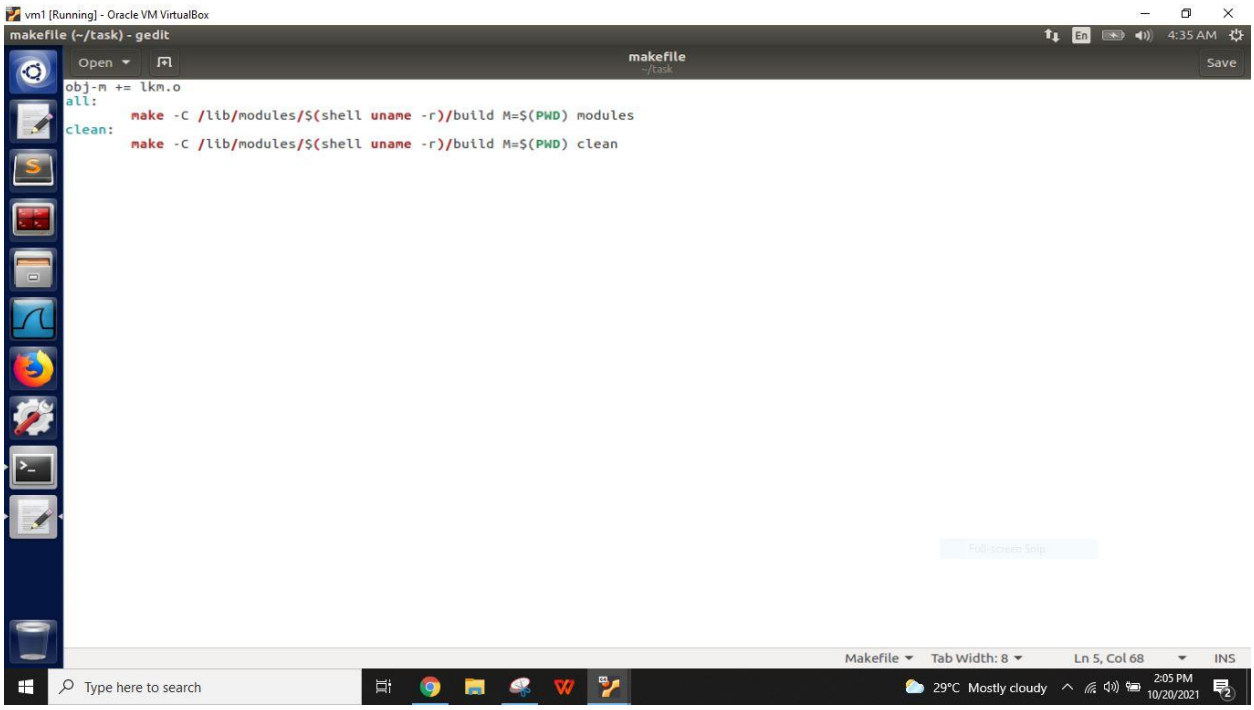


# Task 2: How Firewall Works

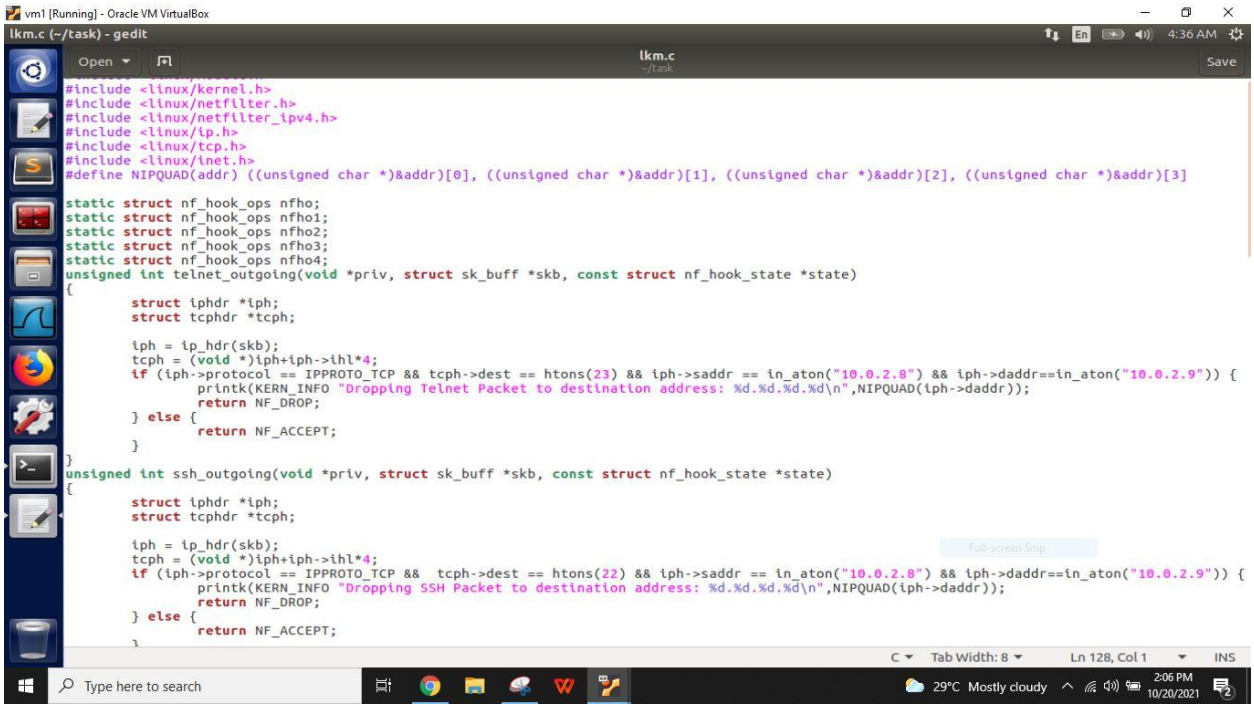
in task2 we create a now folder named task 2 and create a c file names as lkm.c and a Make file



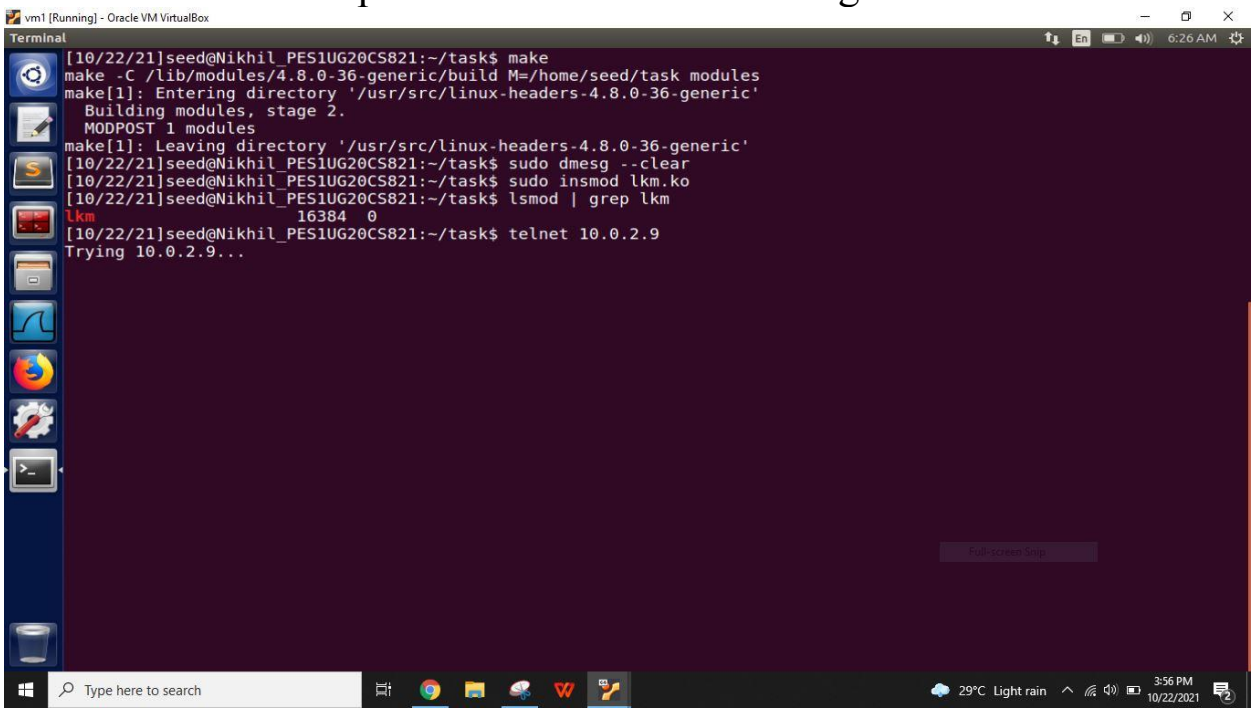

Code for the file Make



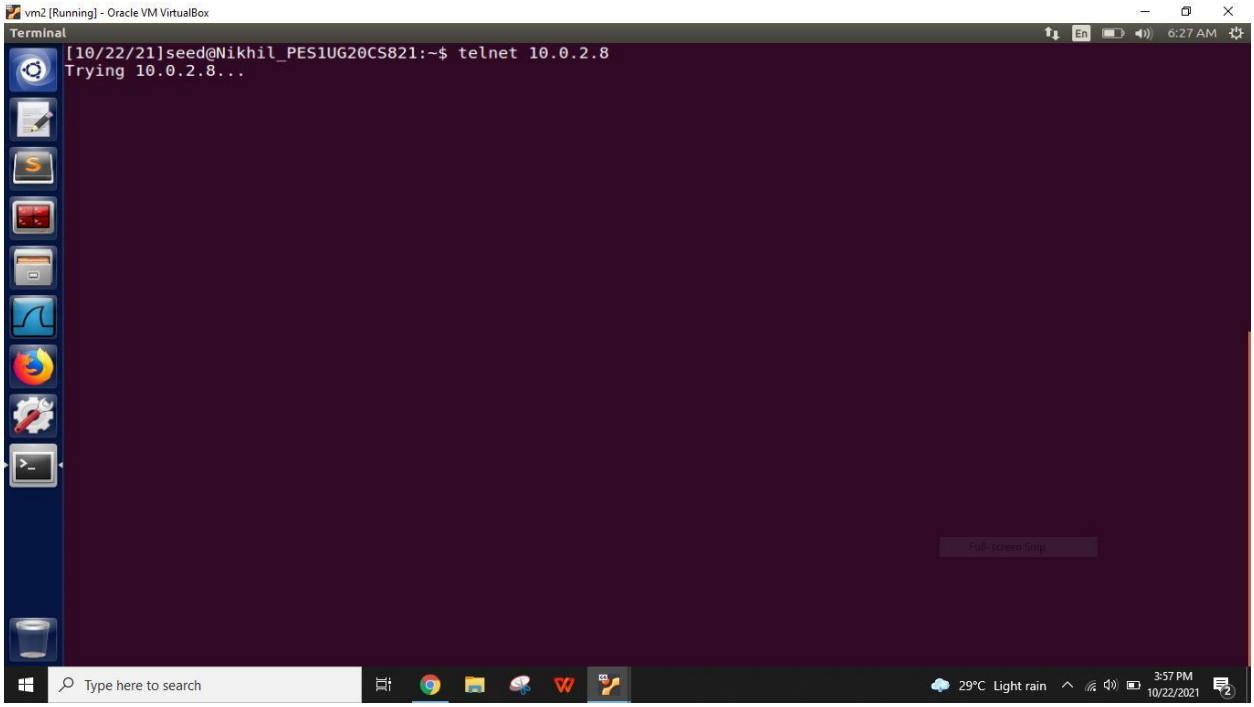
Code for the lk.c file



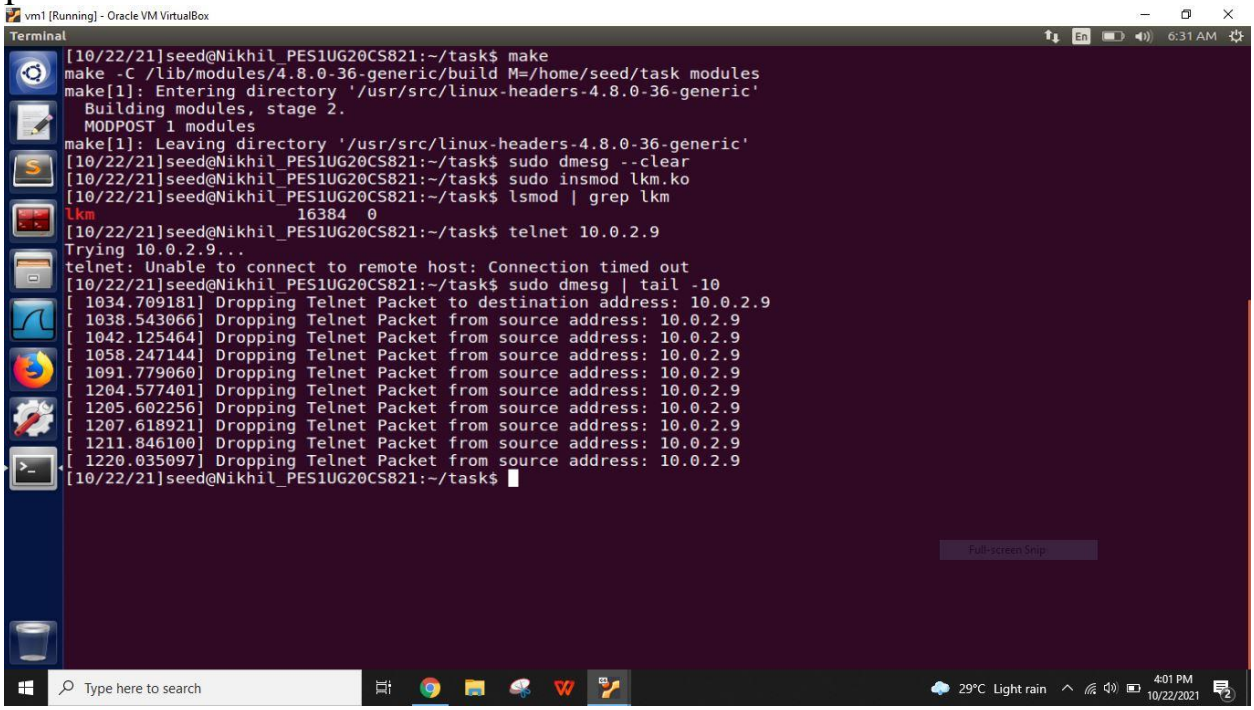

Now we build the make file by executing the command make in the same path where the lk.c and Make file present.now clear the dmesg section and The compiled lkm.ko is inserted using insmod



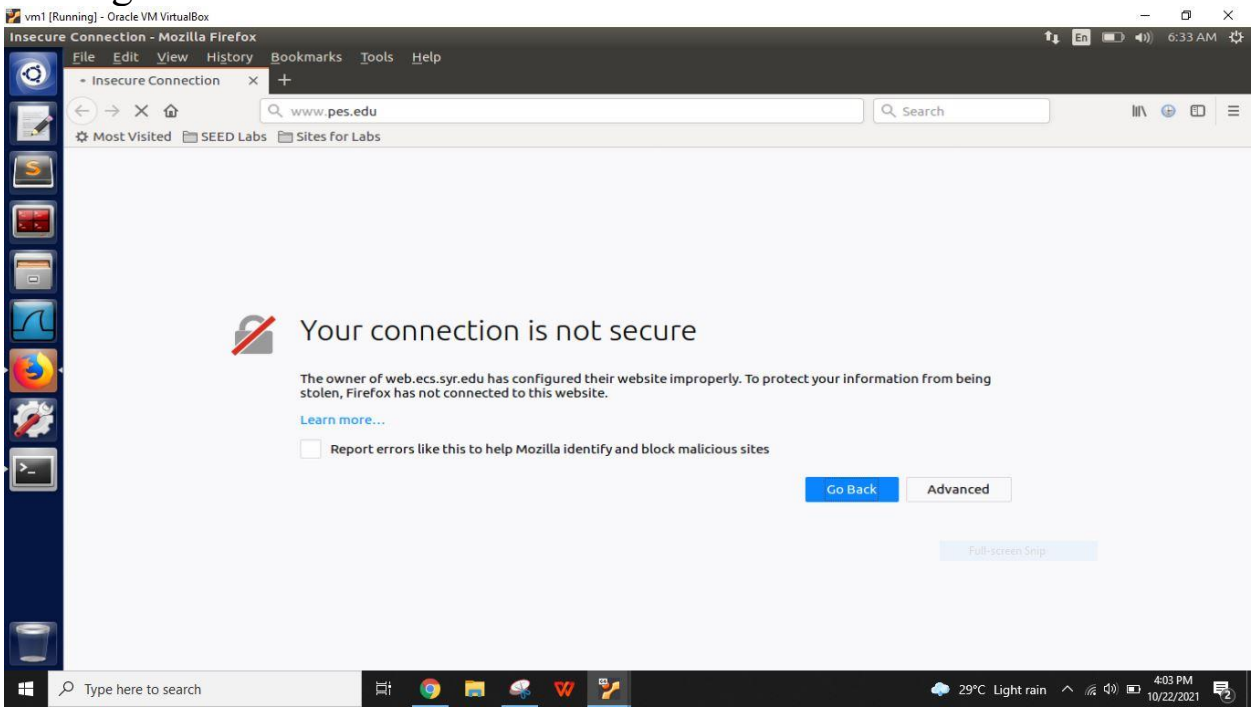
After building the make file we try to connect vm1 from vm2 which is blocked and can be seen in the terminal below



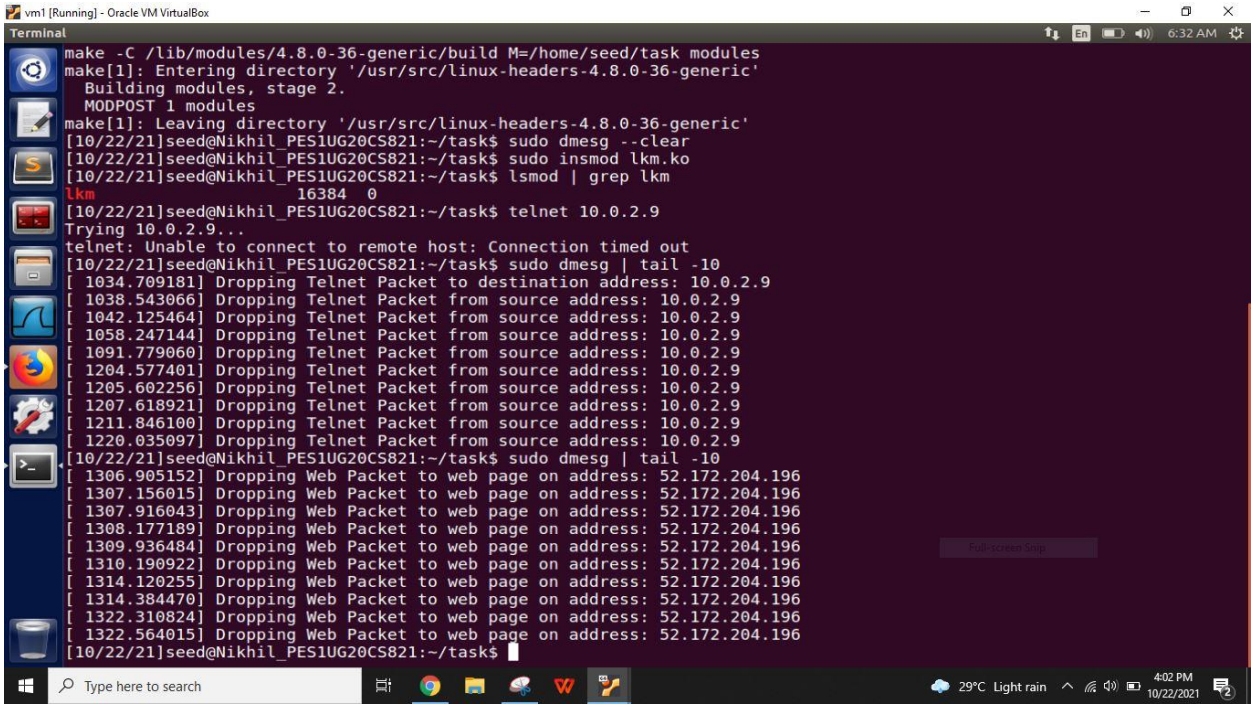

After executing the dmesg command we can see that the dropping telnet packets can be seen



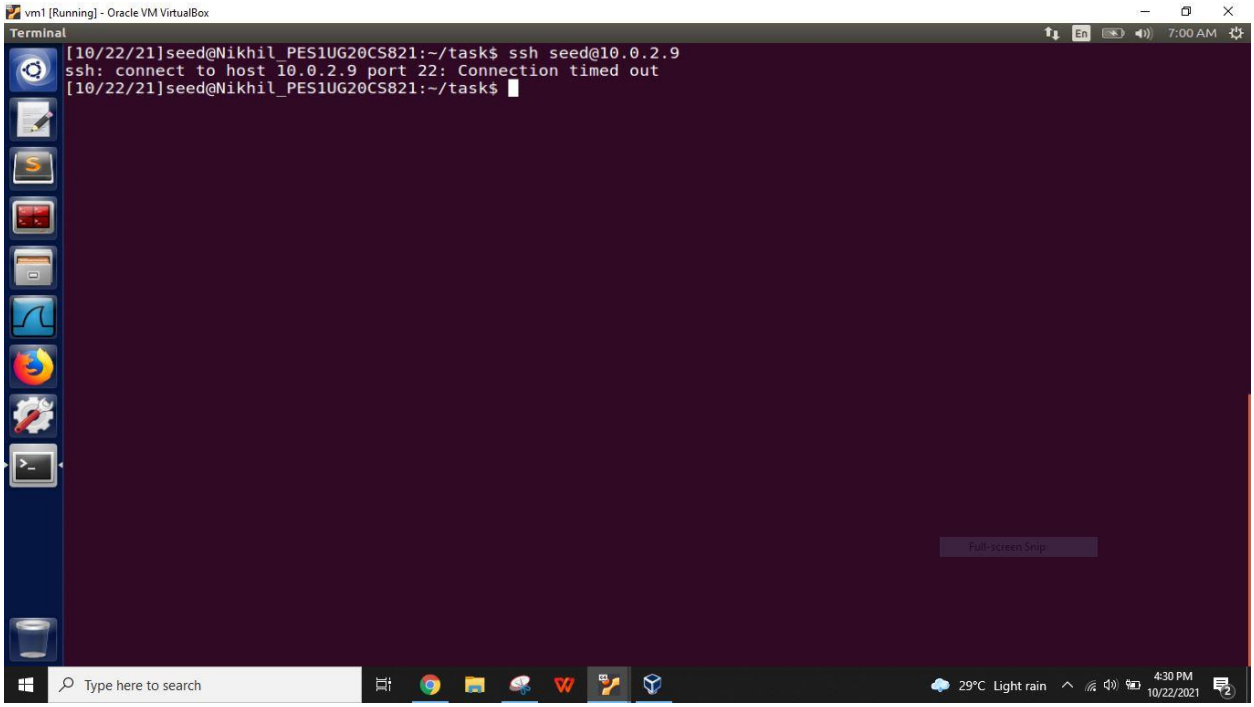
We also blocked the pes.edu site in the c file so now we go to the browser and browse the pes.edu site and can be seen that the site is not loading and it is blocked



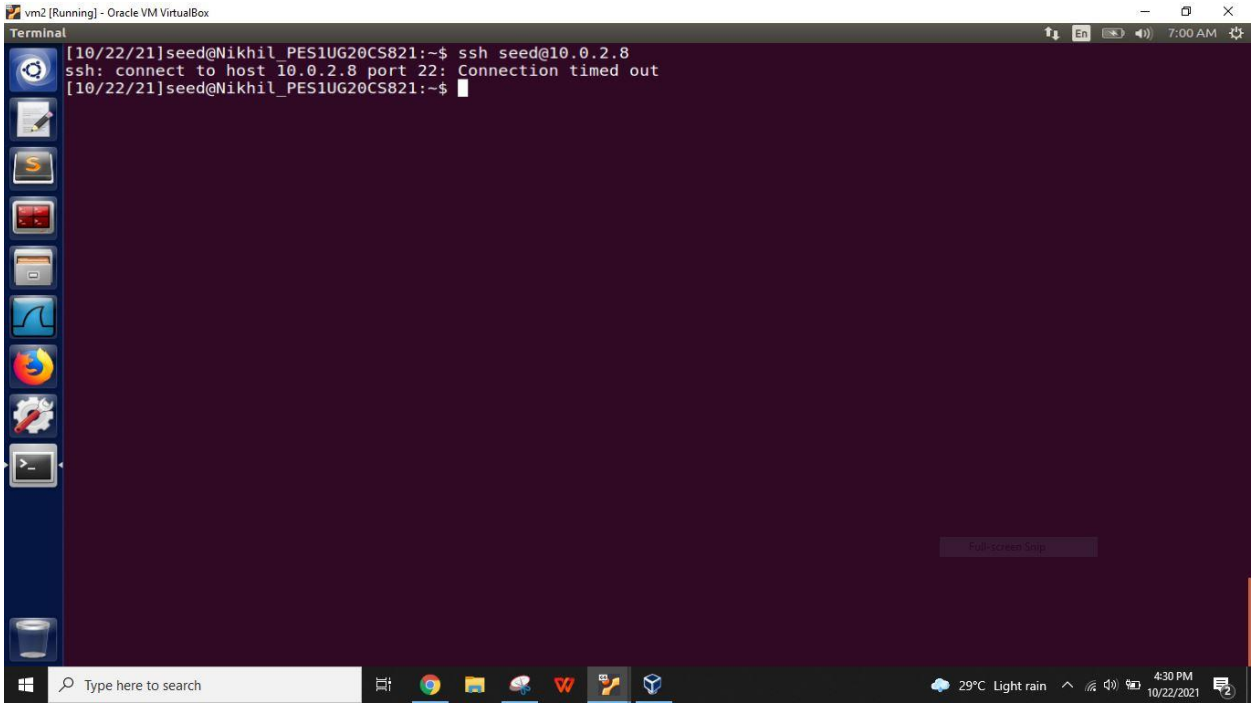

While the pes.edu page is surfing we open the another terminal in the same vm and can see the telnet packets dropping with the help of dmesg command



Now we test whether the ssh is blocked from vm1 to vm2 and we can see that it is blocked



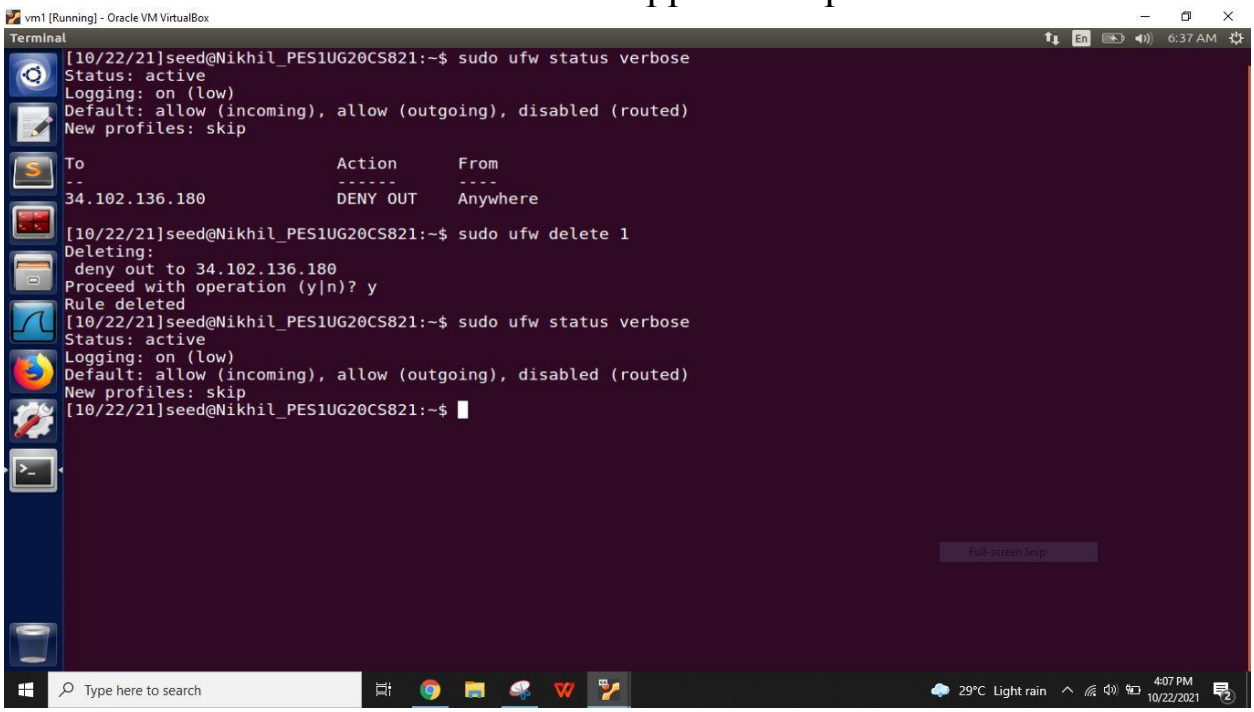

Now we test whether the ssh is blocked from vm2 to vm1 and we can see that it is blocked



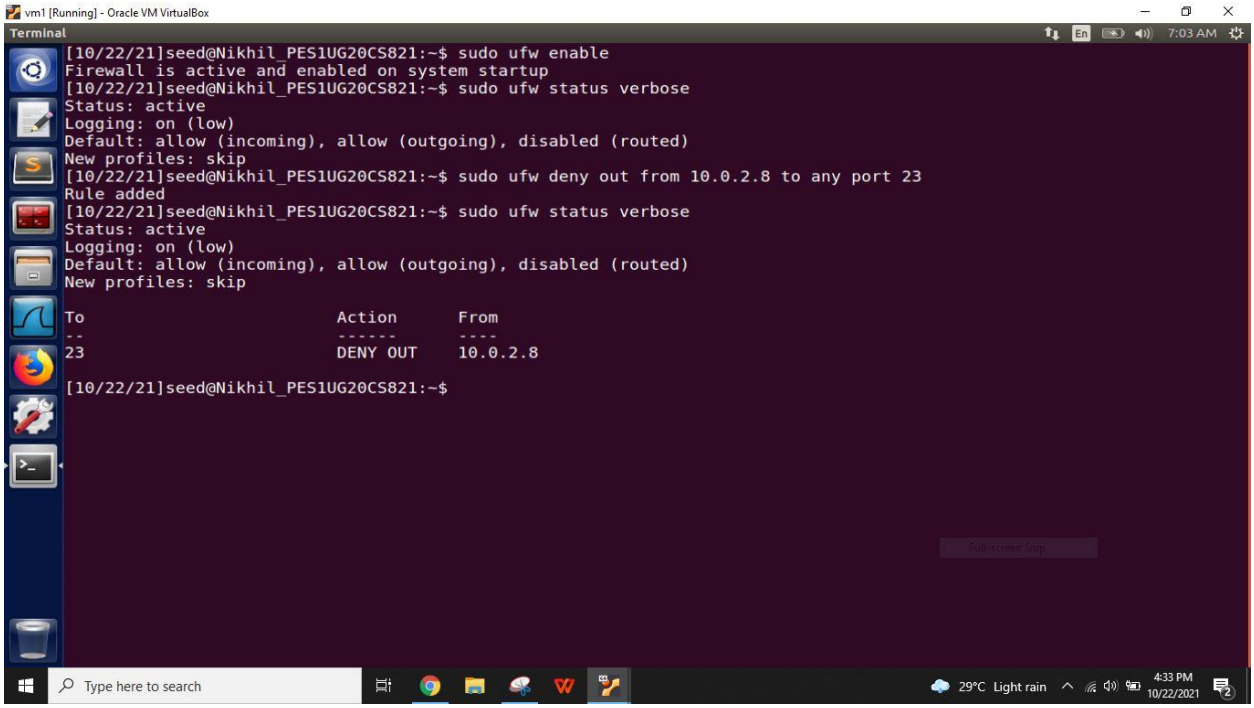
# Task 3: Evading Egress Filtering

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

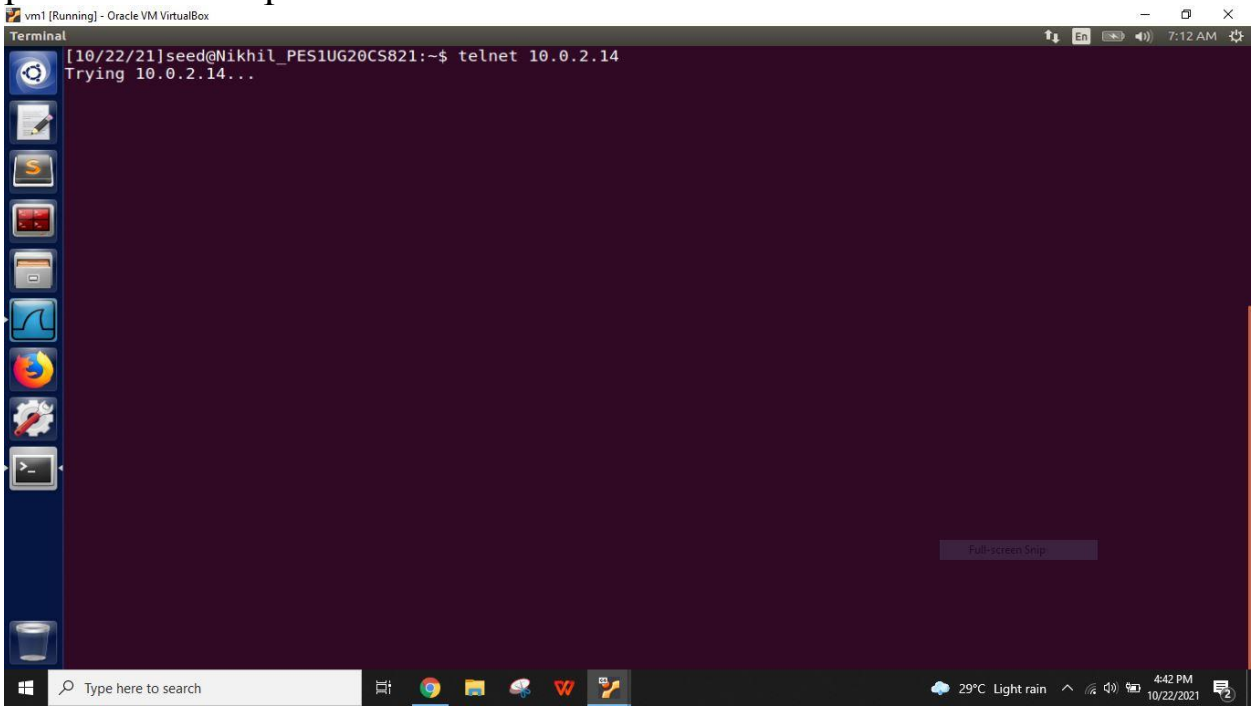
Now we delete the old rules that we applied for previous tasks



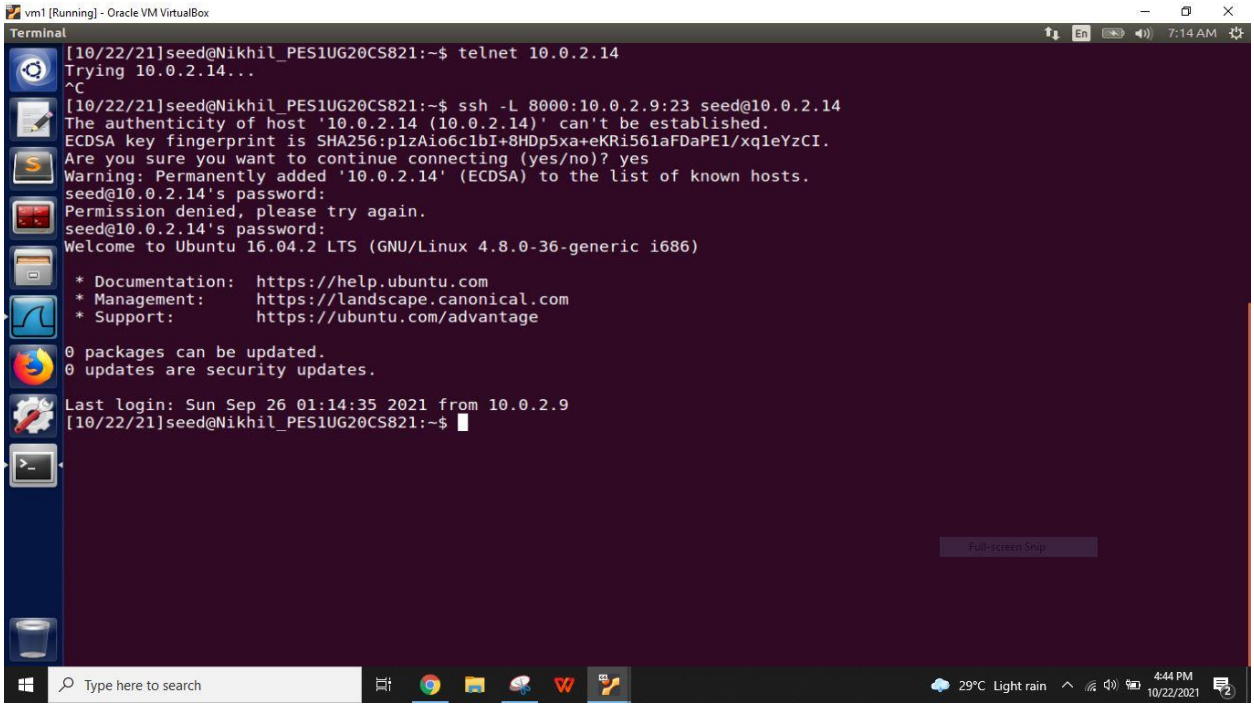

Now set the new rule on vm1 to block all the from port 23 from IP address 10.0.2.8



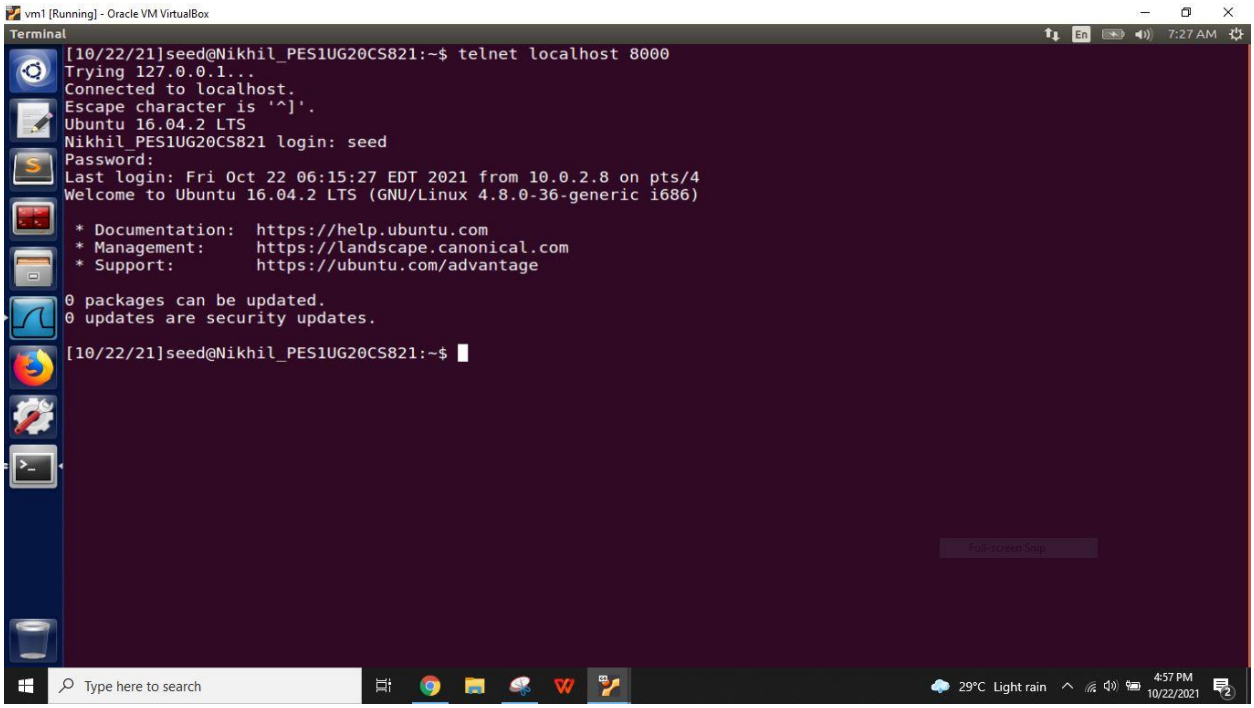
Now try to connect from vm1 to vm3 through telnet which is not possible as the port 23 is blocked from vm1



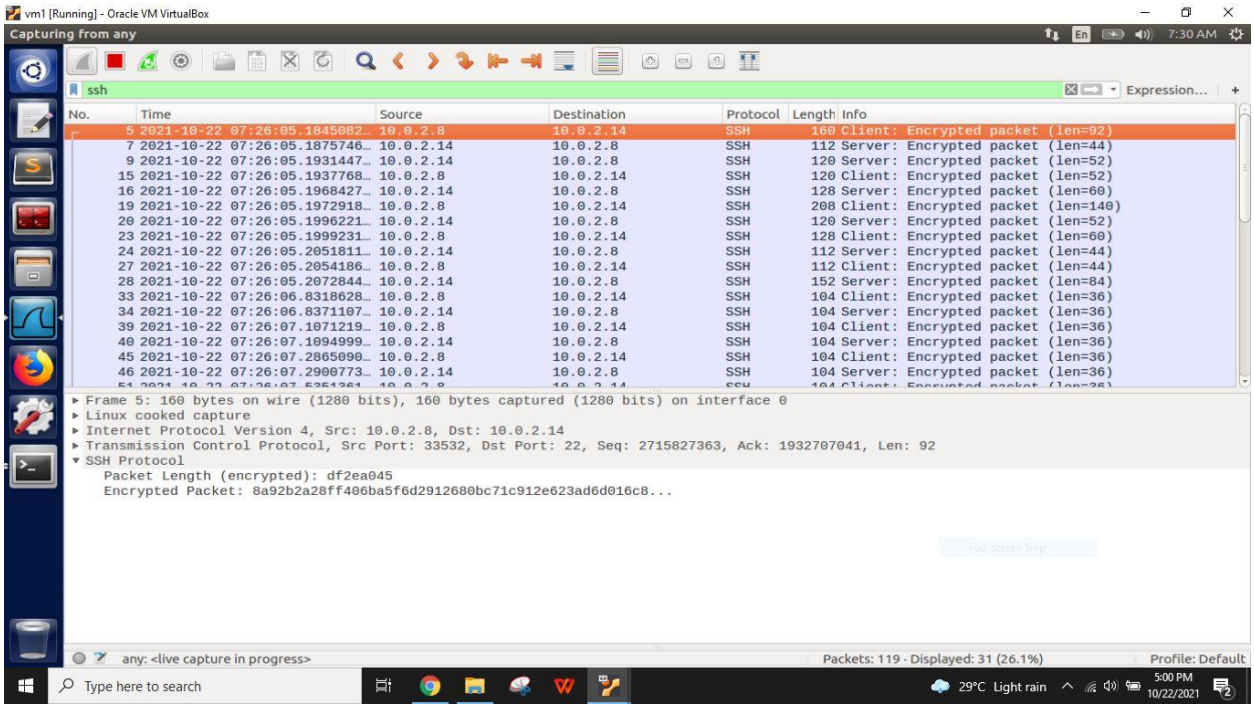

Since the port 23 is blocked we try to establish a connection using the ssh from vm1 to vm3



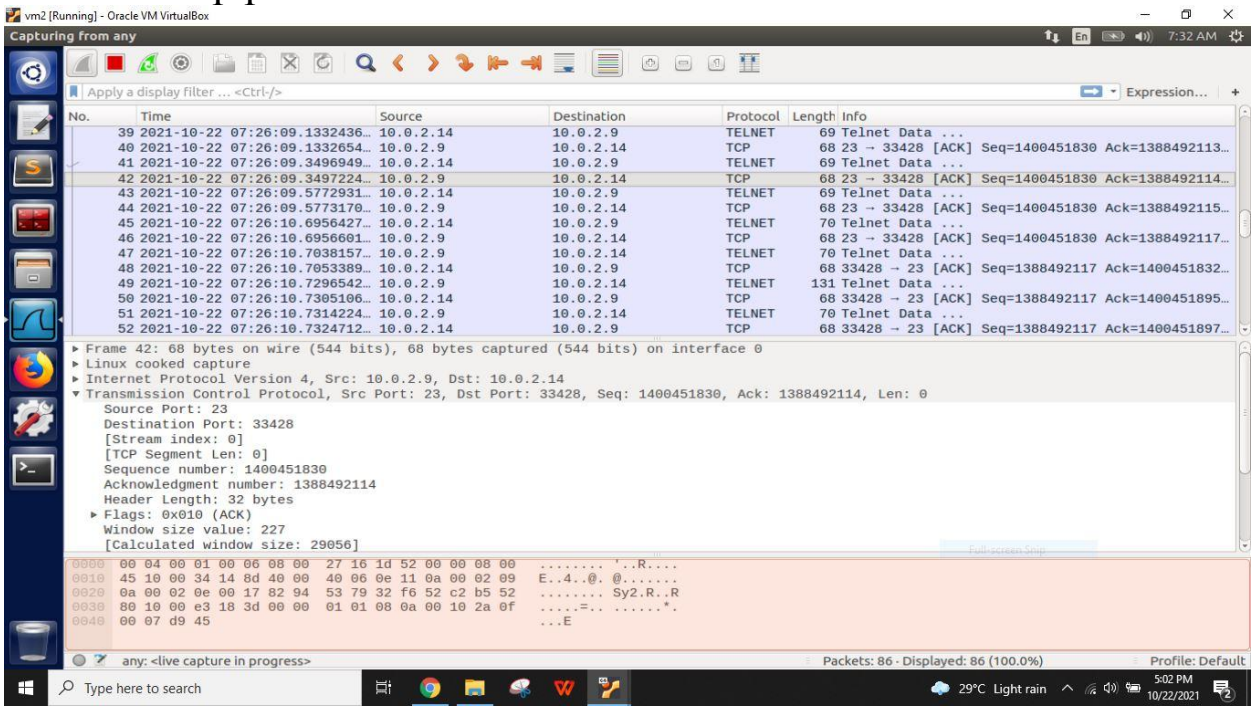
Through the ssh connection we can able to telnet the local host at port 8000



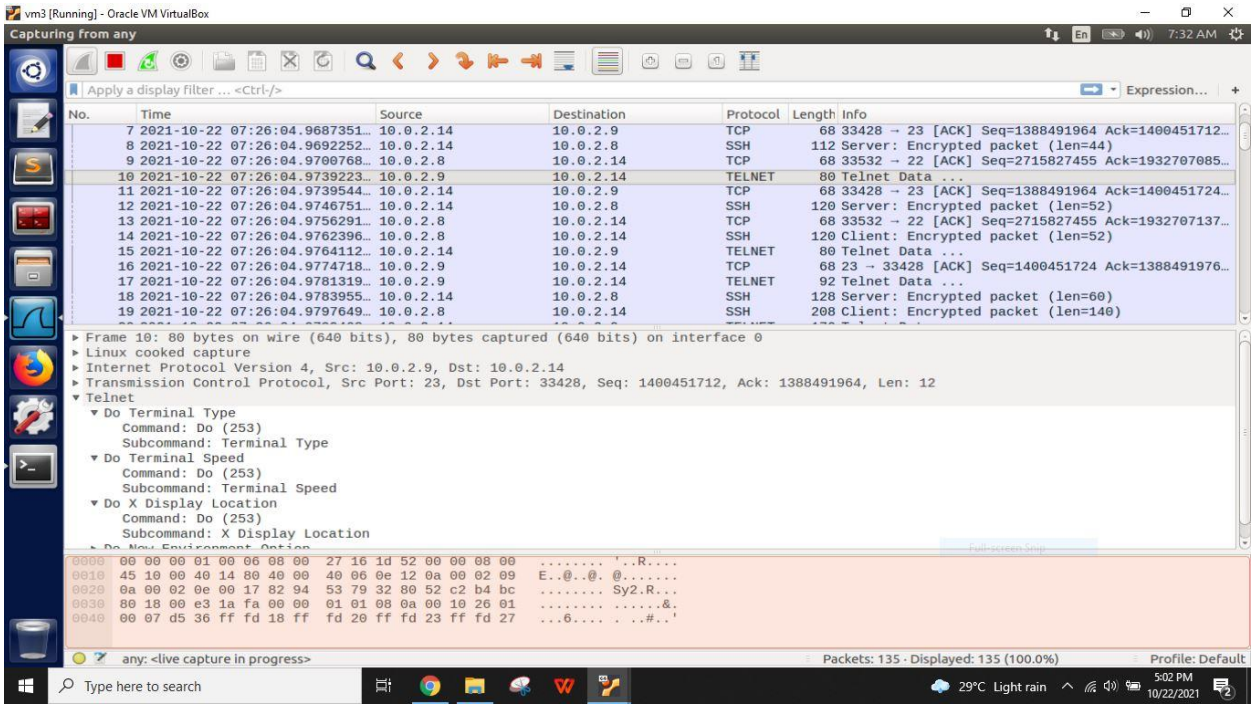

Meanwhile we capture the packets from wireshark in all three vm's
In vm1 wireshark capture we can see that ssh connection is established from vm1 to vm3



In vm2 packets capture through wireshark we can observe that both telnet and tcp packets flow from vm2 to vm3

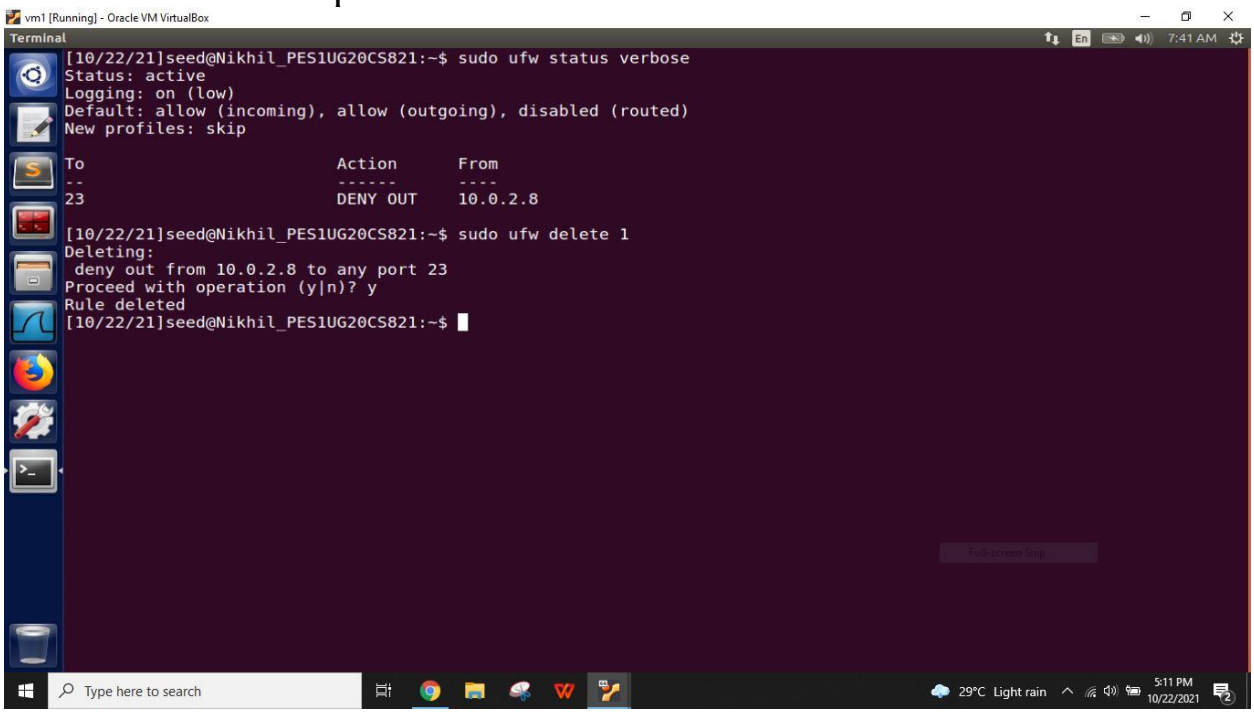



In vm3 wireshark obseravtion we can see telnet,tcp and ssh all three protocols are used.ssh is used by the vm1.tcp and telnet is used by the vm2

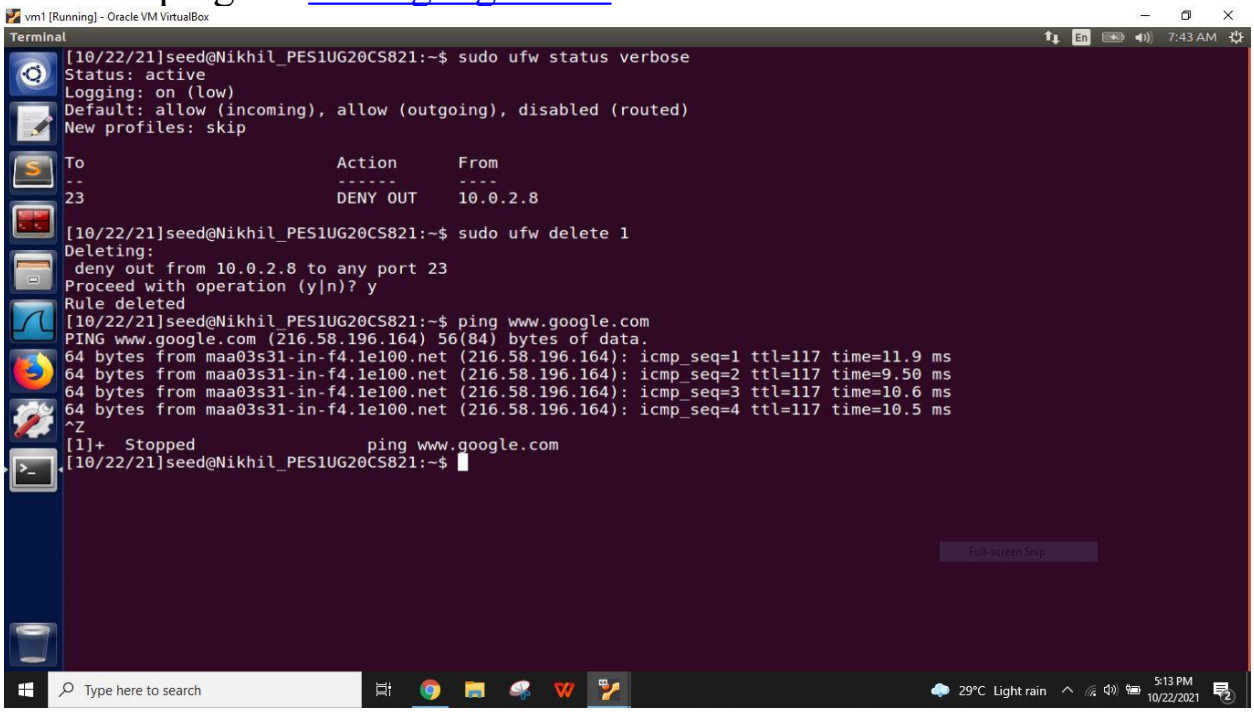


**Task 3.b: Connecting to Google using SSH tunnel**

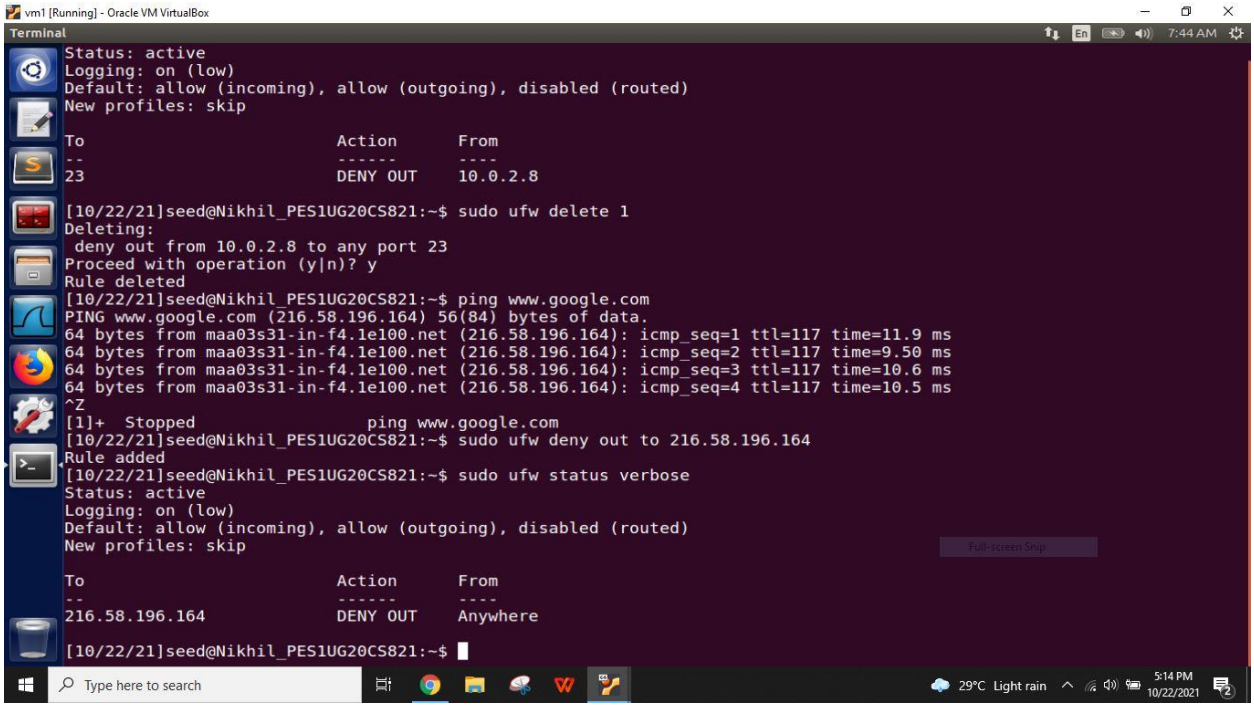
first we delete the previous rules we added



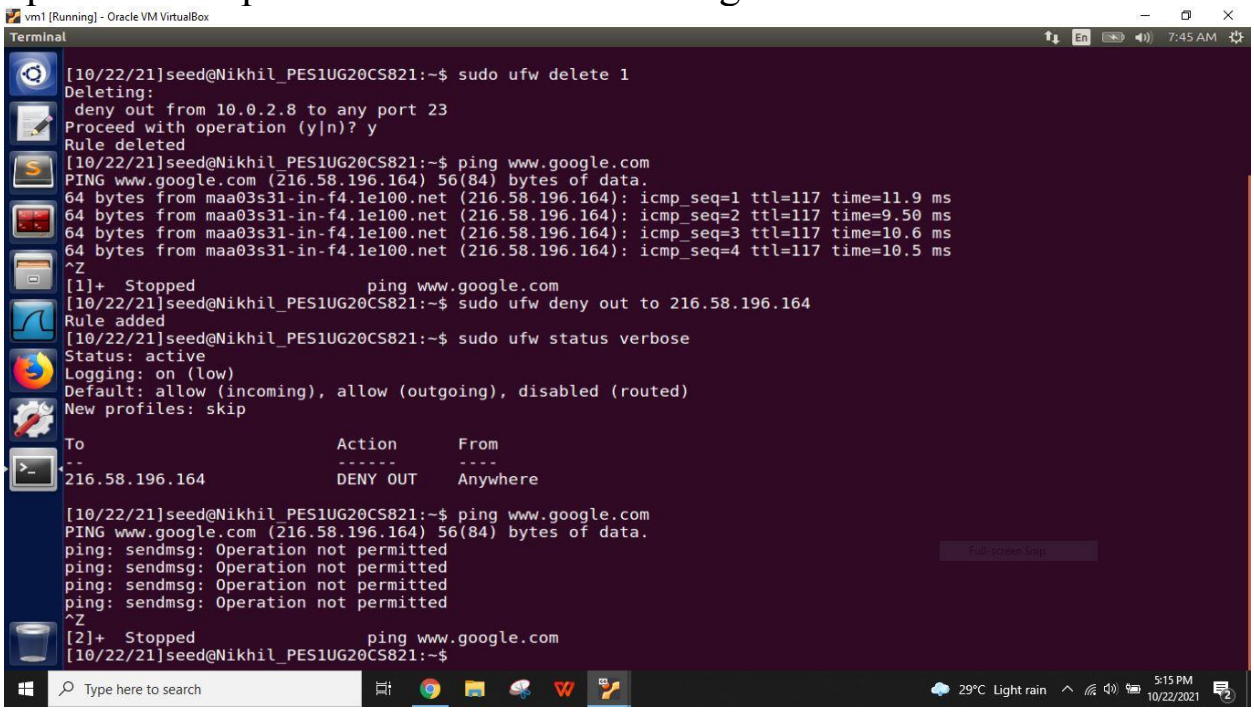

Now we ping the www.google.com to obtain the IP address of it



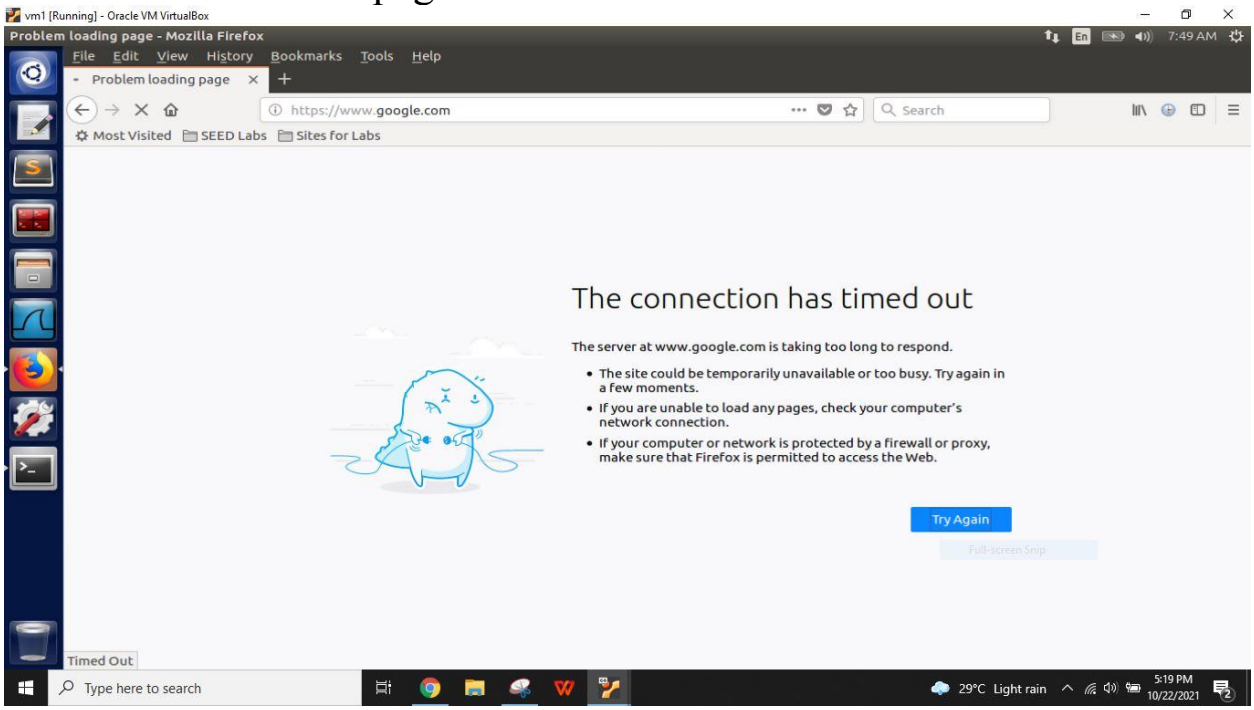

After obtaining the ip address by pinging use to add rule so that we can block to visit that site.



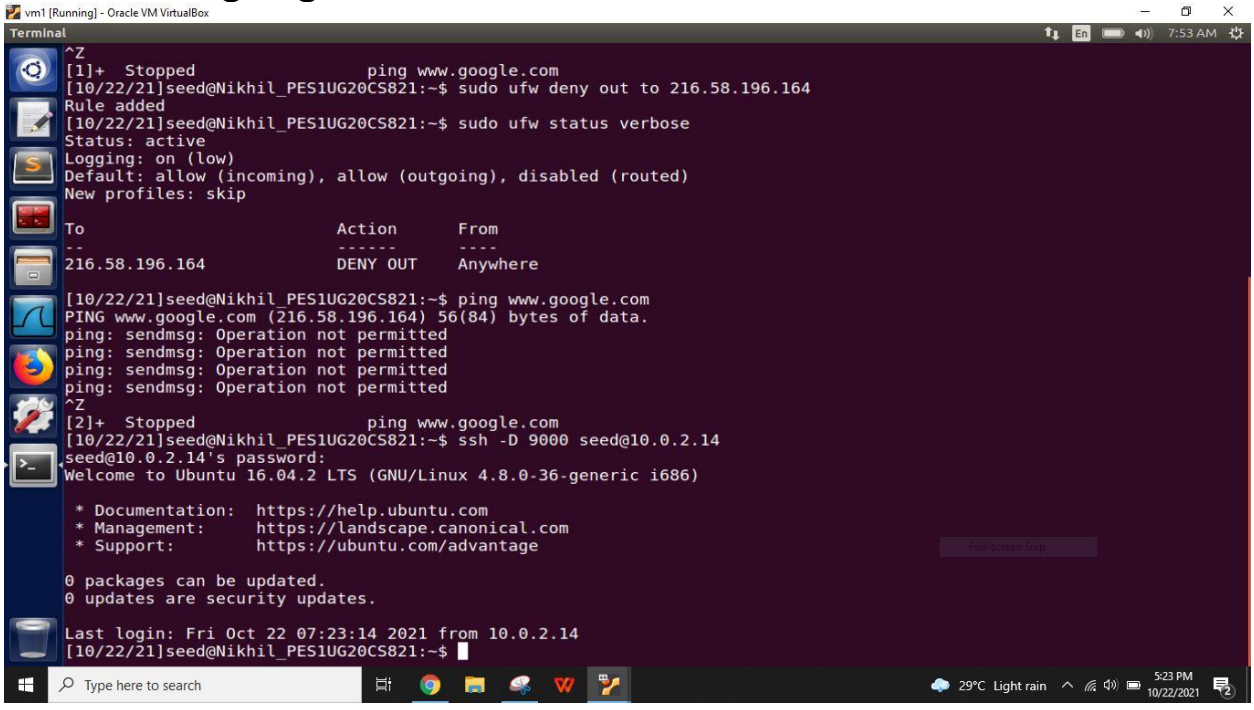

After adding the rule try to ping the same website and we will get operation not permitted as that site is being blocked



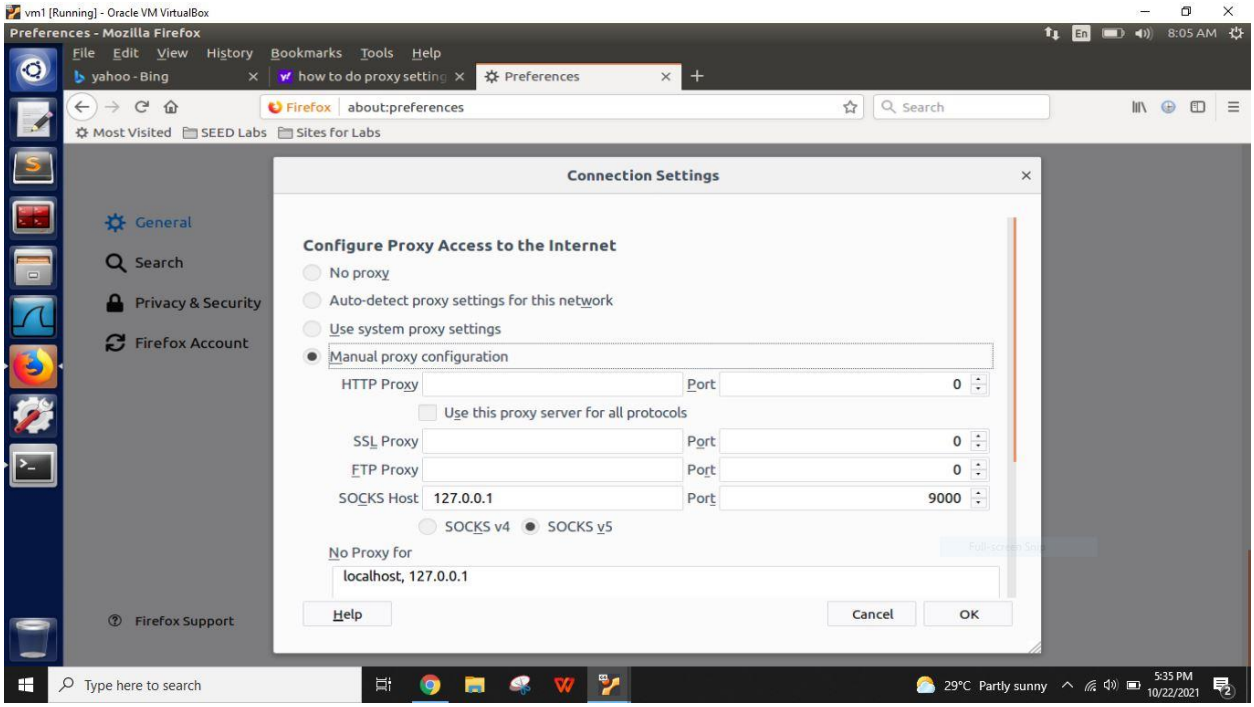
Later we visit the same site through browser and we can see the browser fails to load that web page



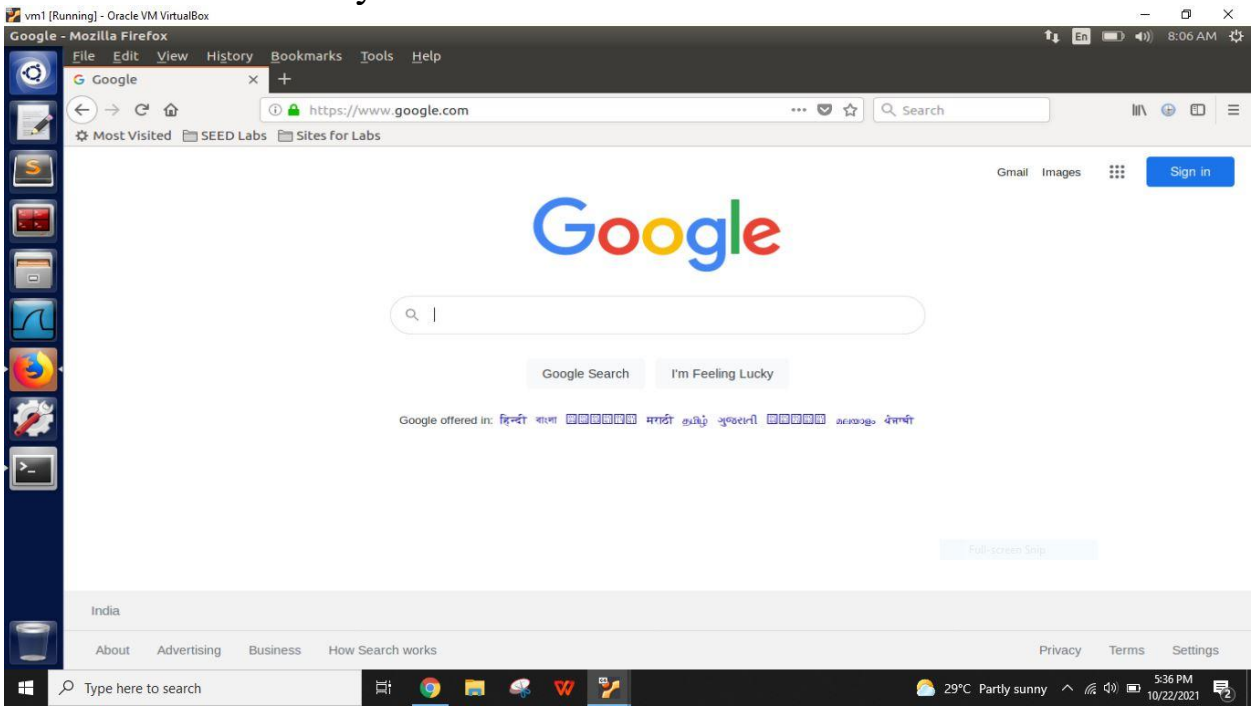

Now we are going to establish a ssh connection from vm1 to vm3



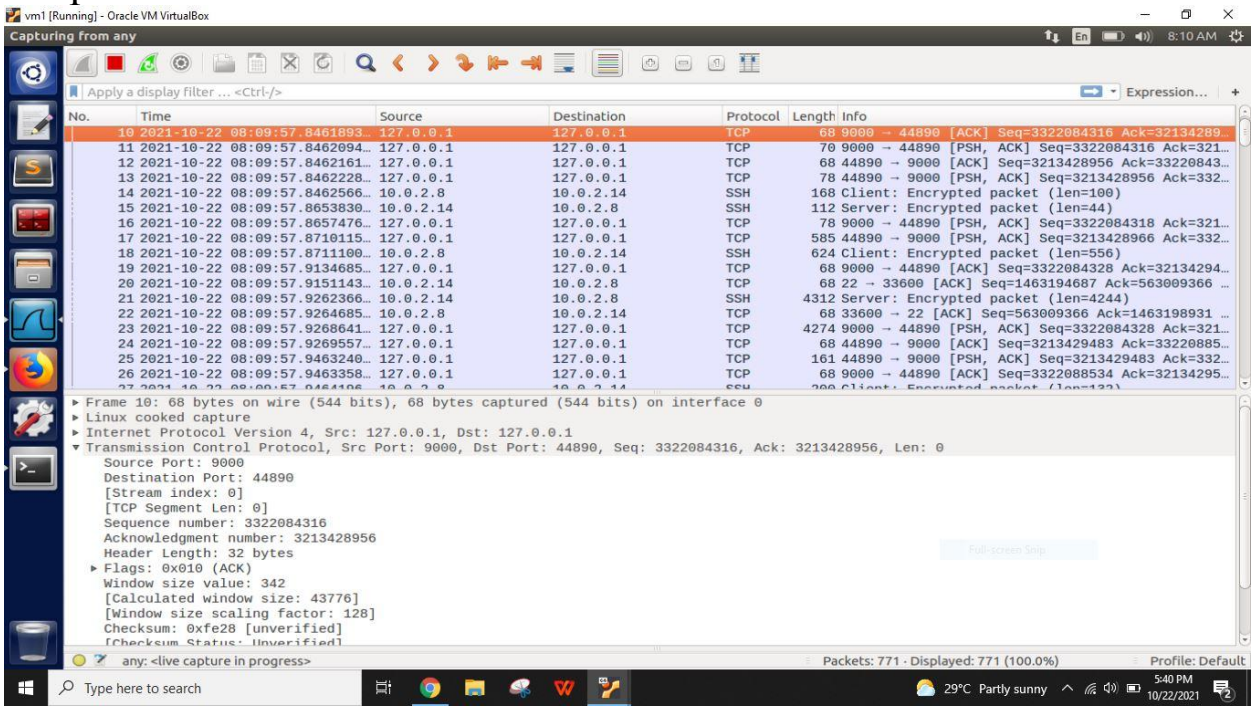
After successful ssh connection we are going to add manual proxy in the browser of vm1 as follows



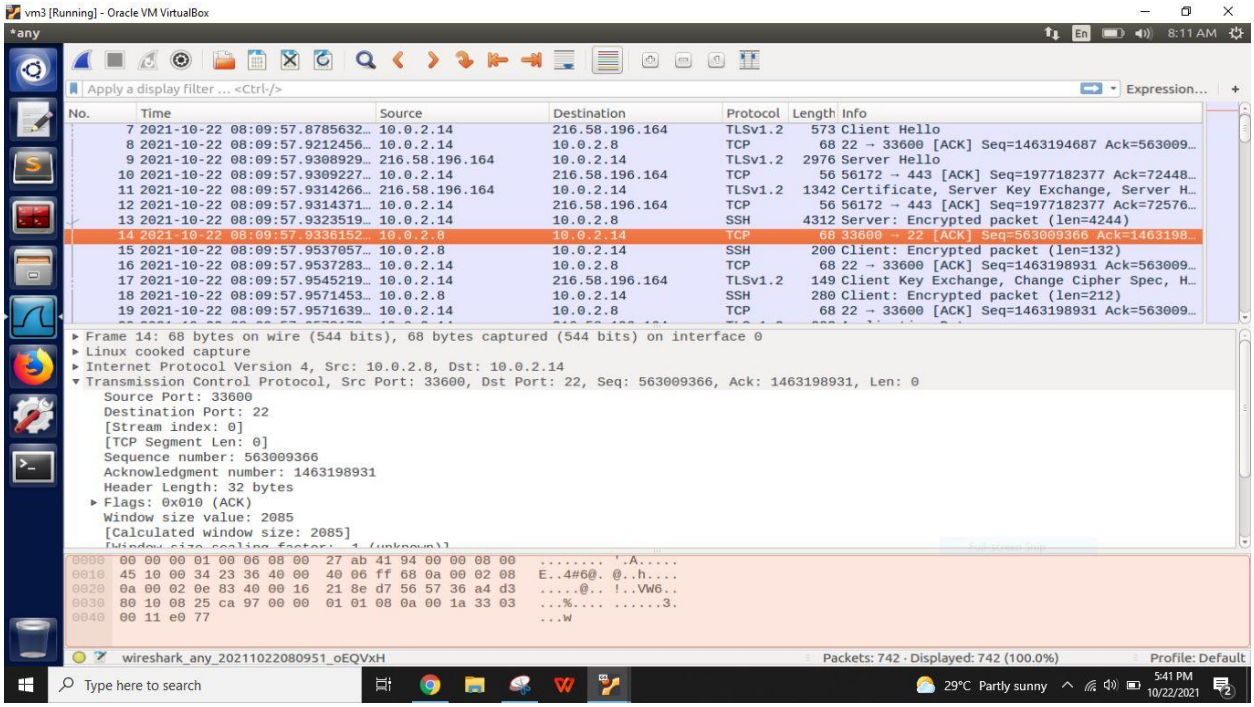

After adding the proxy we can see that the web page [www.google..com](http://www.google..com) is loaded successfully



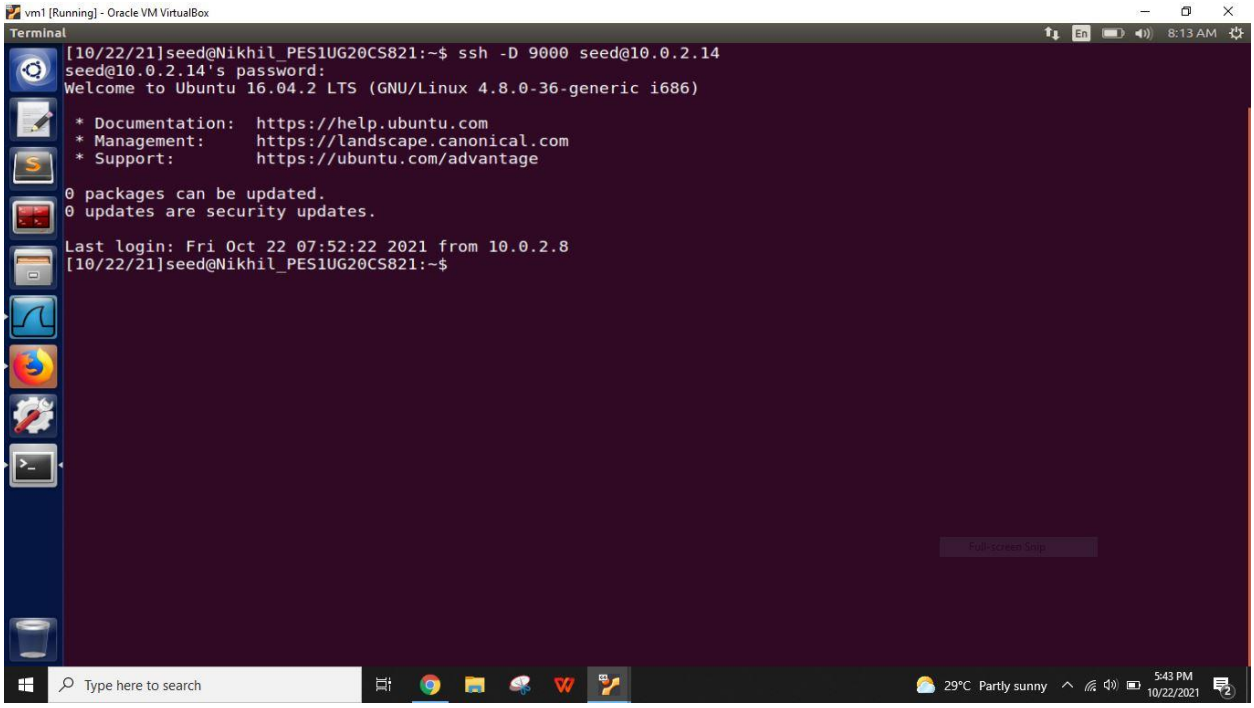
We can see the observation in vm1 wireshark which consists of tcp and ssh protocols with vm3



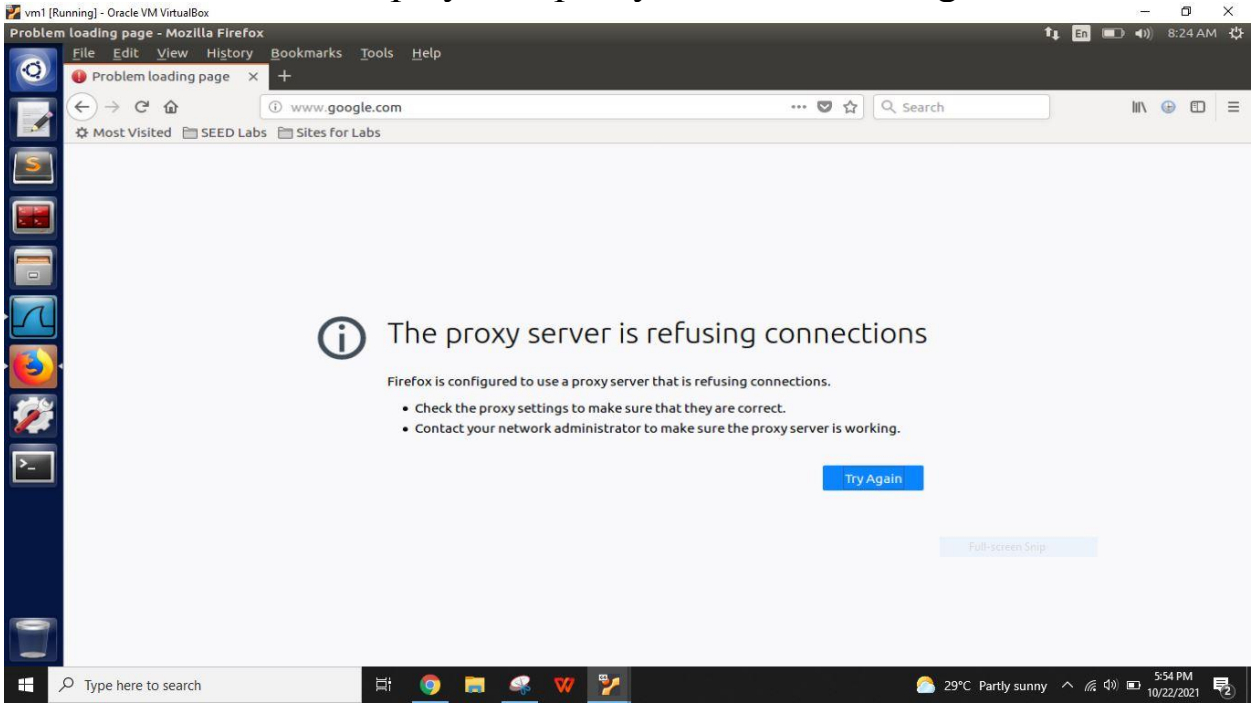

Same observation can be seen in vm3 with vm1



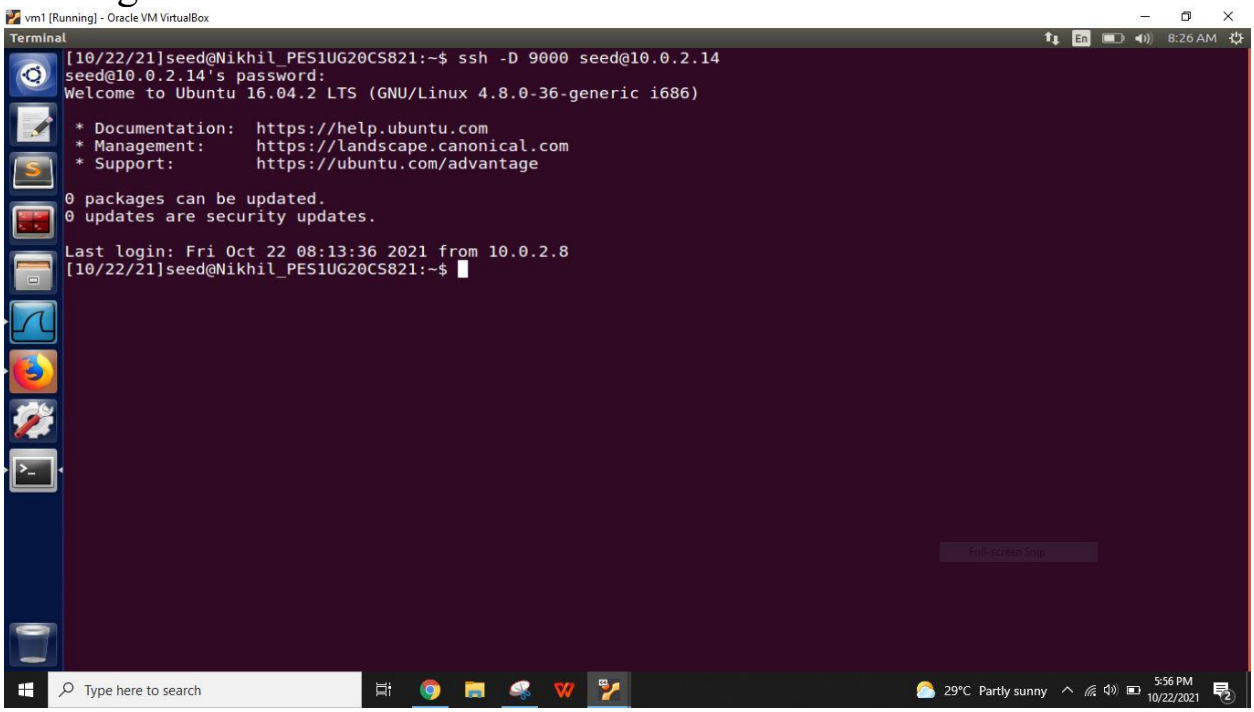

After we gets disconnect from the ssh connection in the vm1



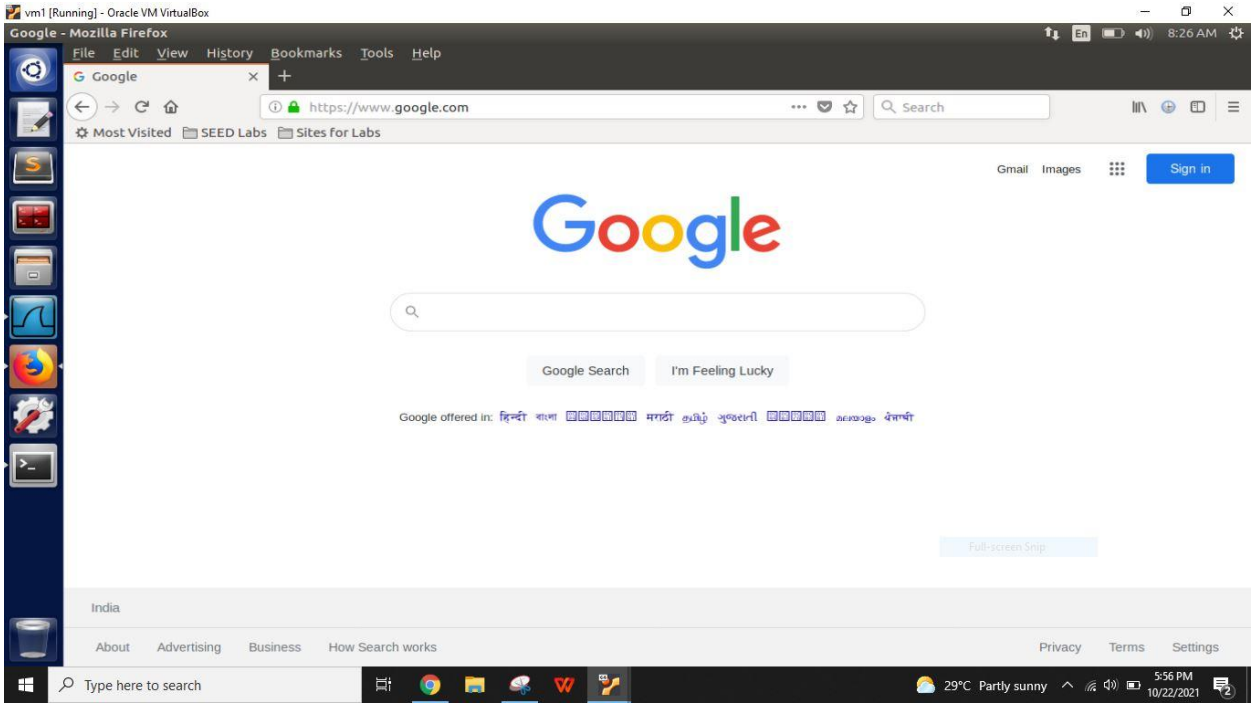

After getting disconnected from the ssh connection we are going to load that wepage again in the browser by clearing the cache and history.we can observer that it displays the proxy server is refusing connections



Now again we establish the ssh connection

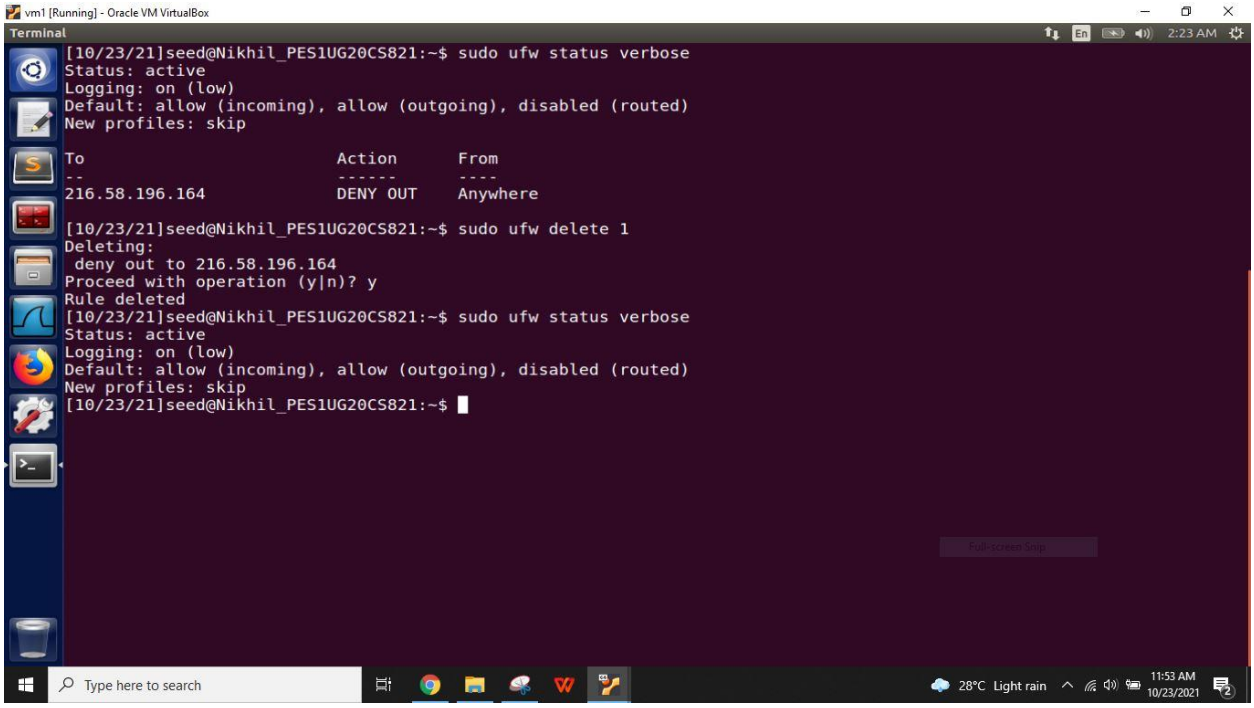



We reload the same page again and we can access the web page after the ssh connection

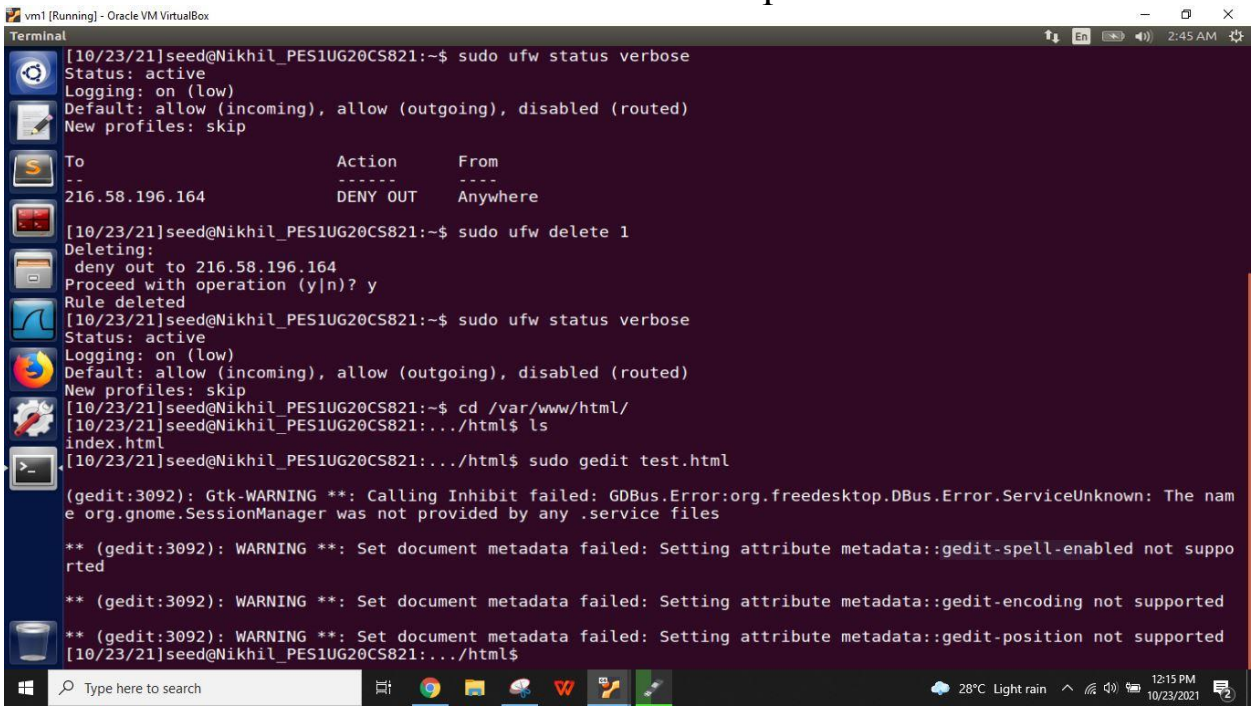


# Task 4: Evade Ingress Filtering

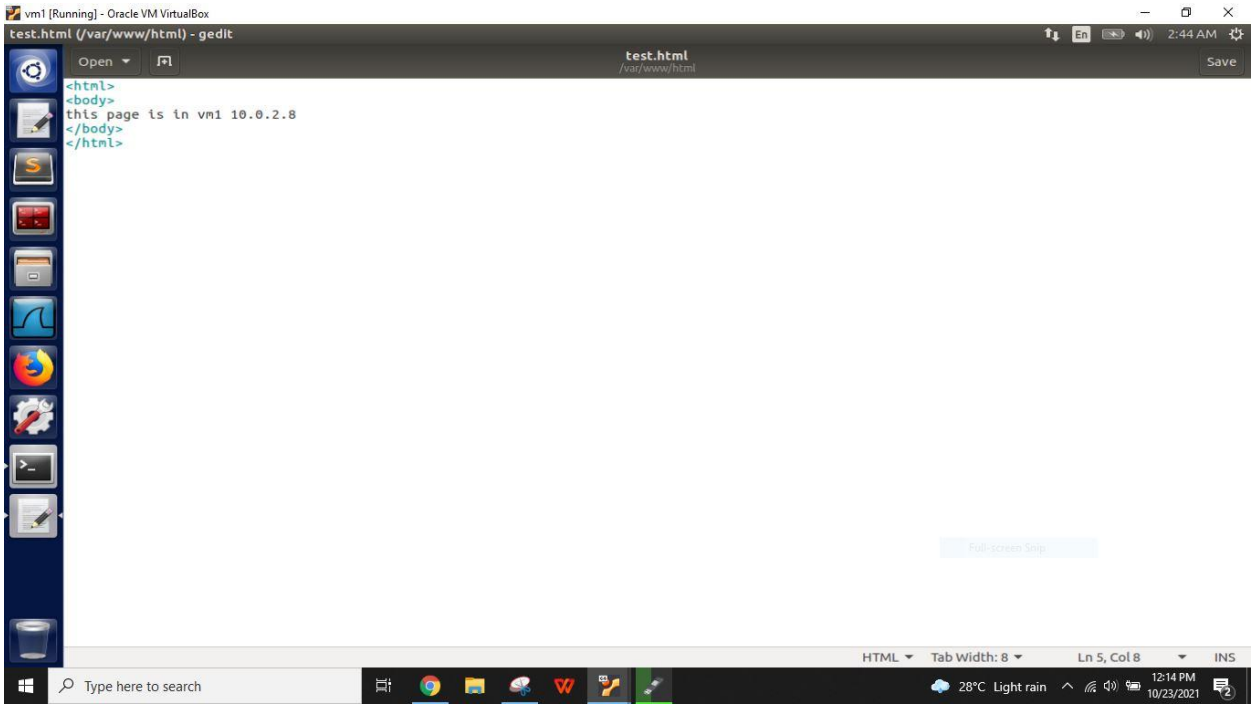
Delete all the previous rules we added for the previous tasks



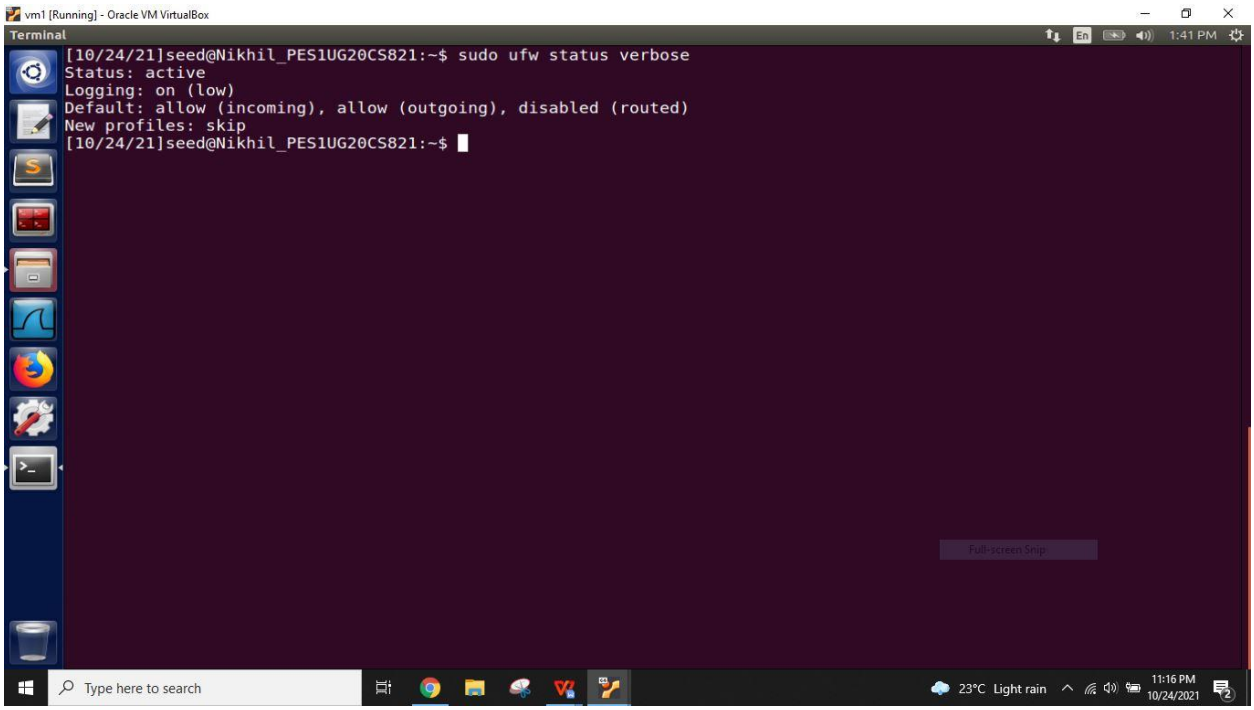

Create a new filed named as test.html in the path var/www/html



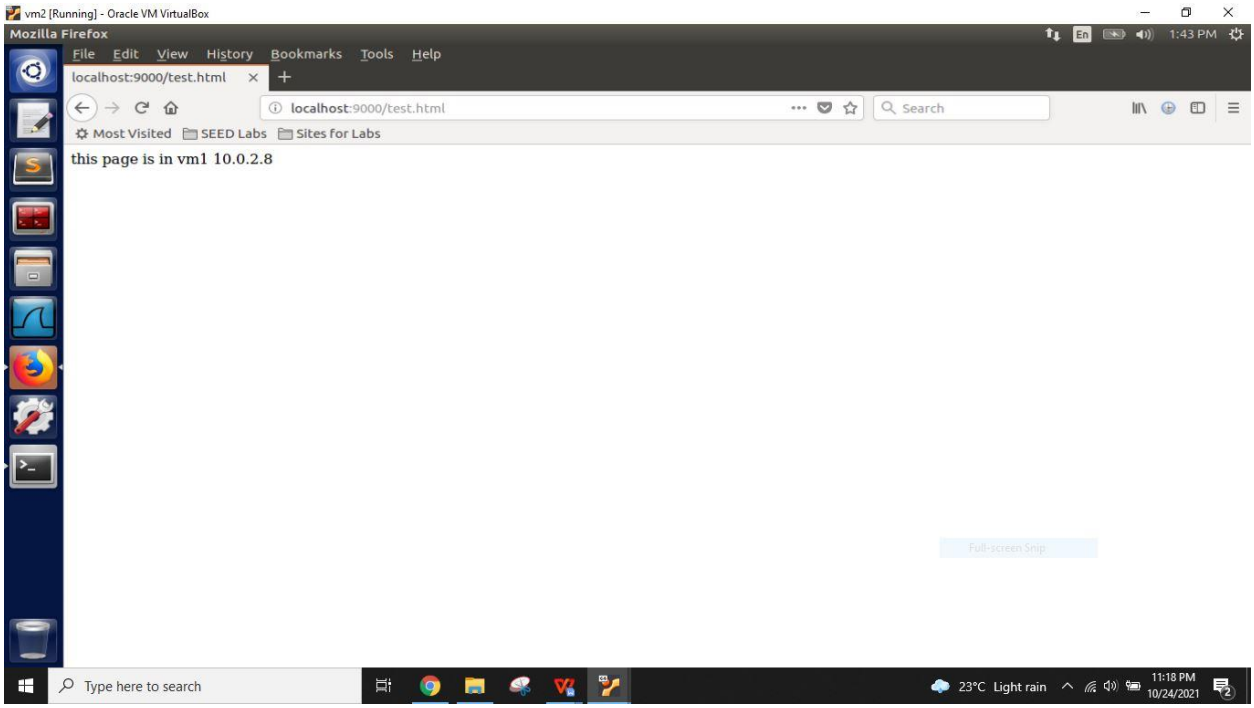
The content of the file test.html is as shown below



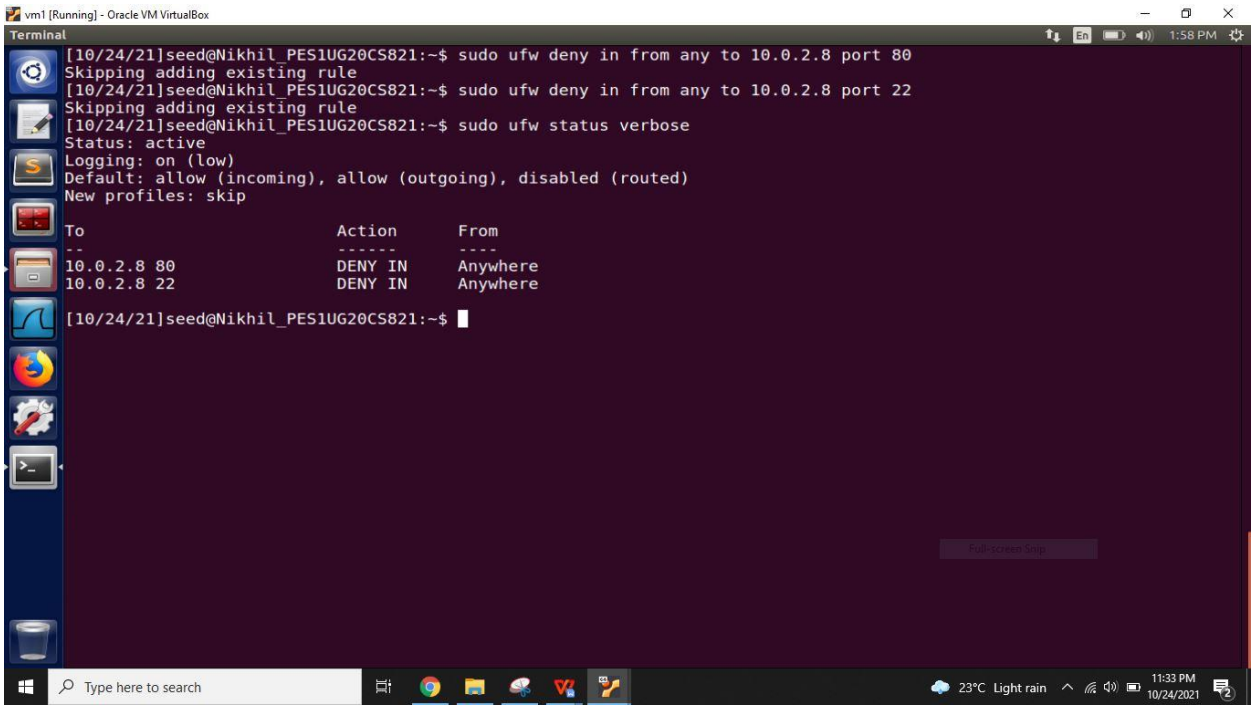

Now there is no firewall rules in the vm1



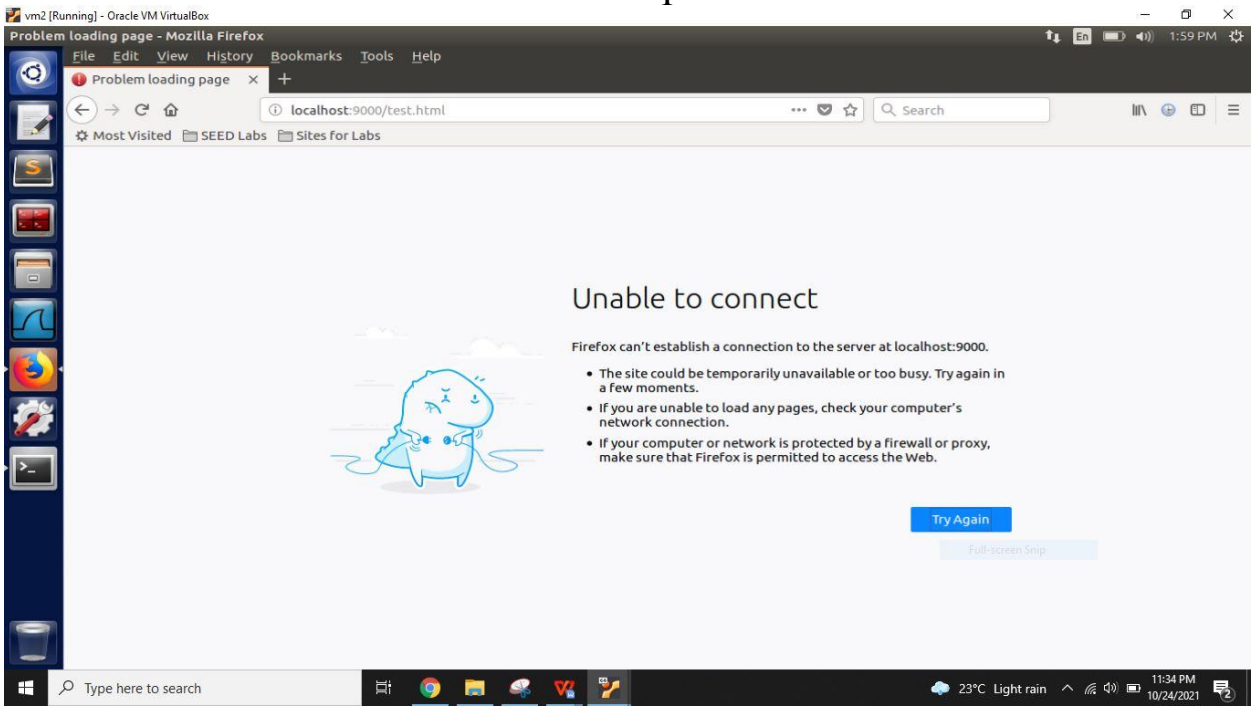
Because of there is no rules in the vm1 we can access the web page that is present in the vm1 form vm2 through the browser



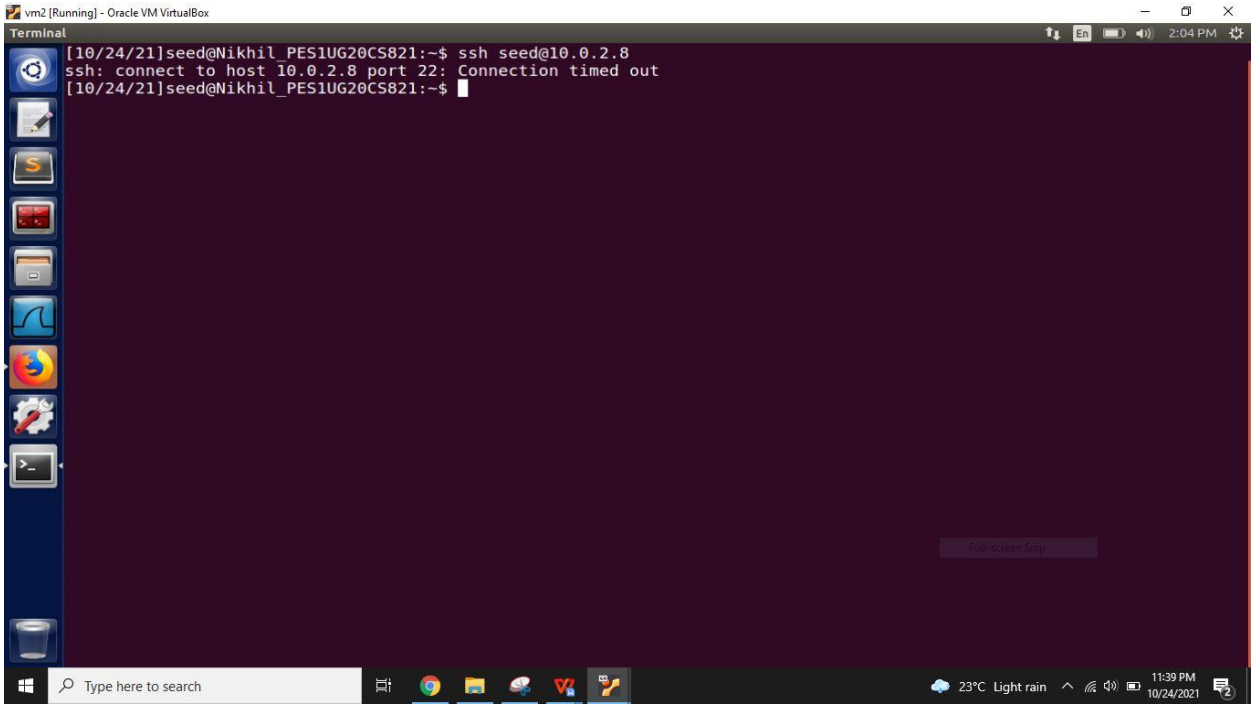

Now we add rules in the vm1 firewall to block the port 80 and 22



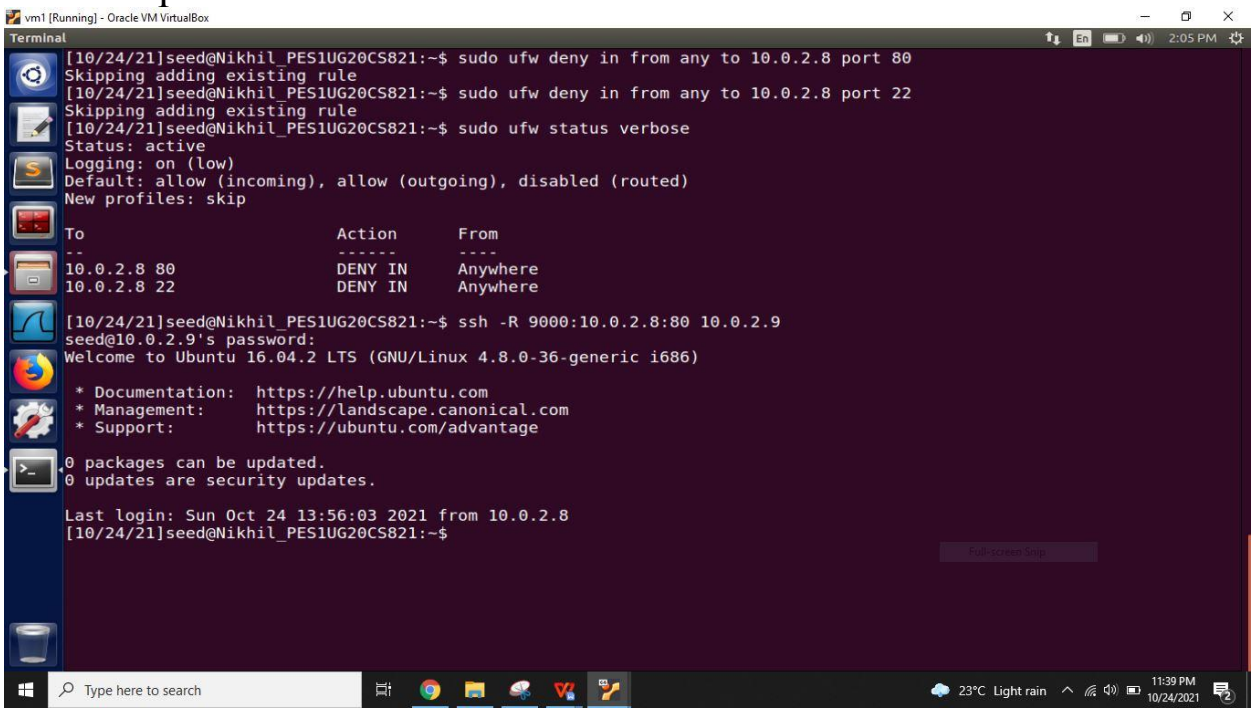
After adding the rules we go back to the vm2 and access the same file which is not able to access now as the port 80 from vm1 is blocked



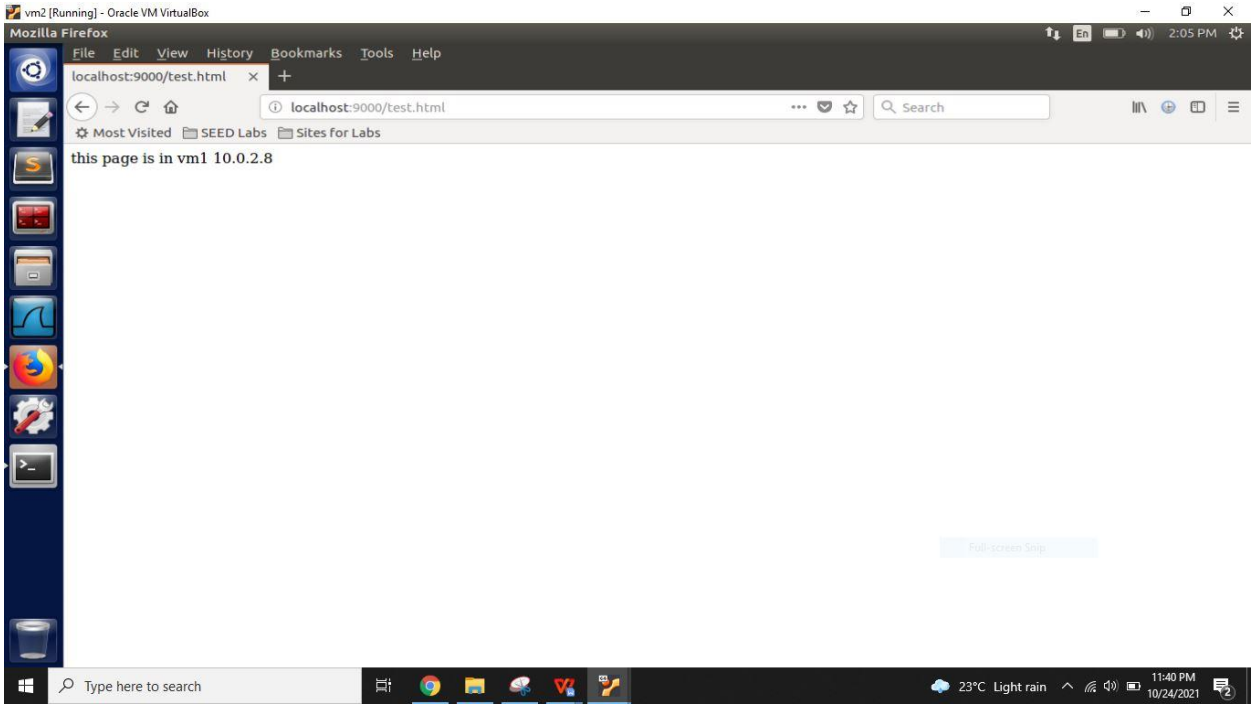

Also we cannot establish a ssh connection to vm1 from vm2 as the port 22 is blocked



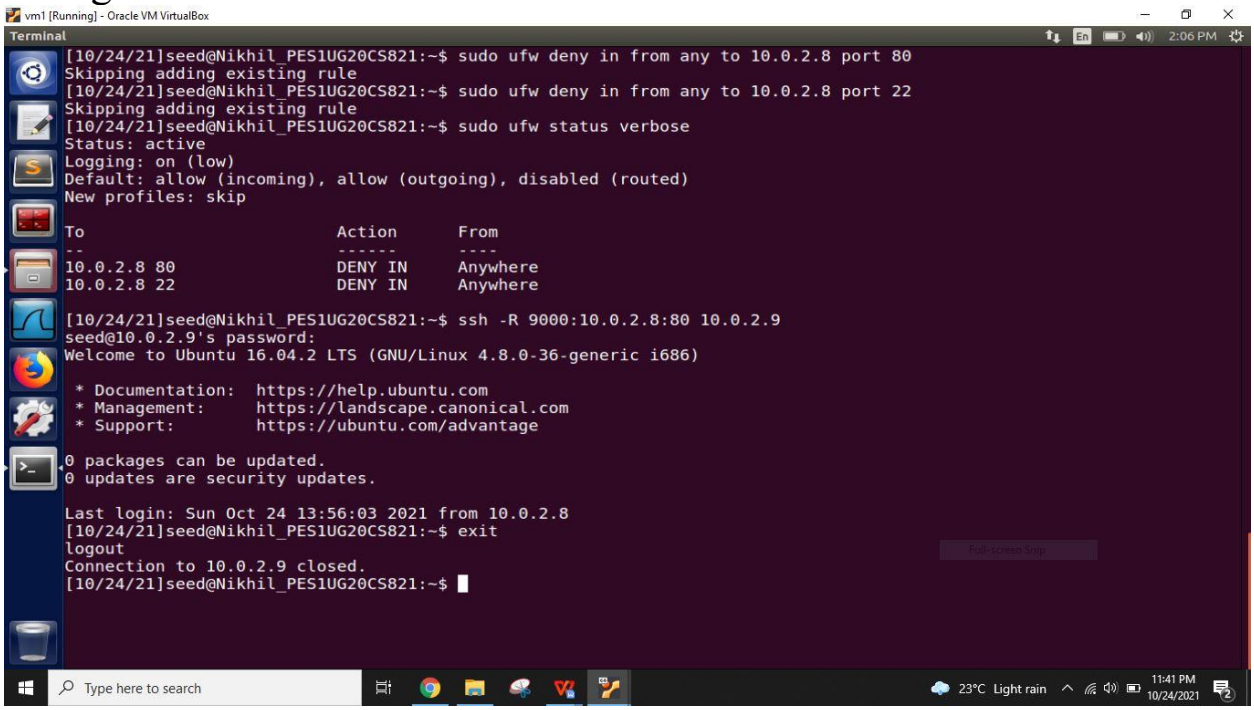
Now we perfrom reverse ssh from the vm1 to establish a reverse tunnel




With the tunnel connection we can access the webpage from the vm2 browser



Now we break the tunnel from vm1 to check whether we can access the file again from the vm2 browser




We can see again we cannot access the web page from the from the vm2 browser after the tunnel is broken.