Computer Networks Security
Assignment 4

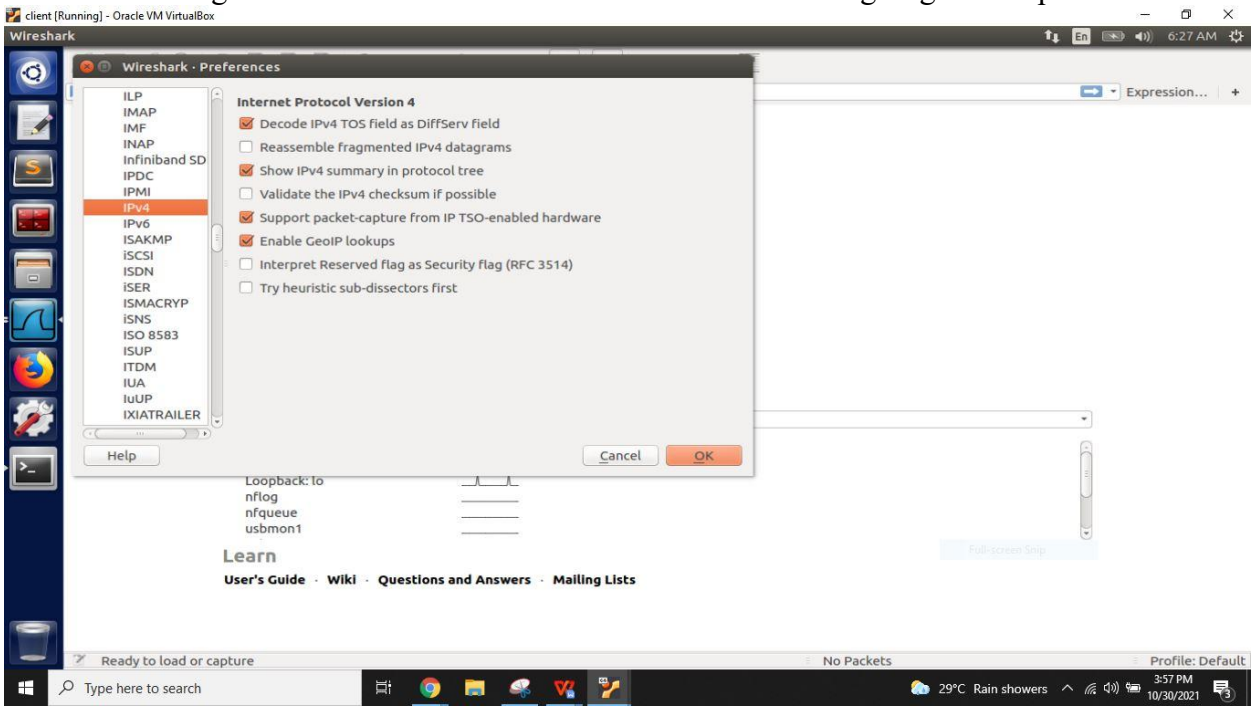
Name: Nikhil TM
SRN: PES1UG20CS821

Lab setup
Vm1:10.0.2.18
Vm2:10.0.2.19
Vm3:192.168.60.5

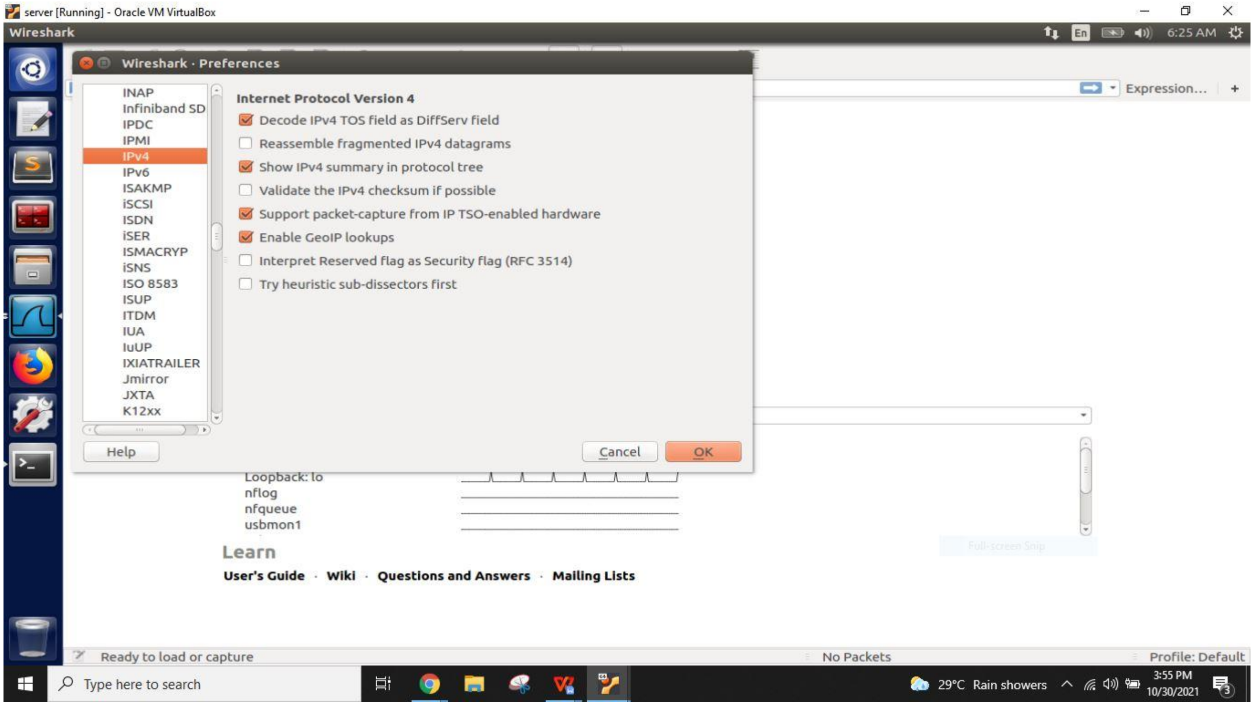
Task1: IP Fragmentation

Task 1a: UDP server

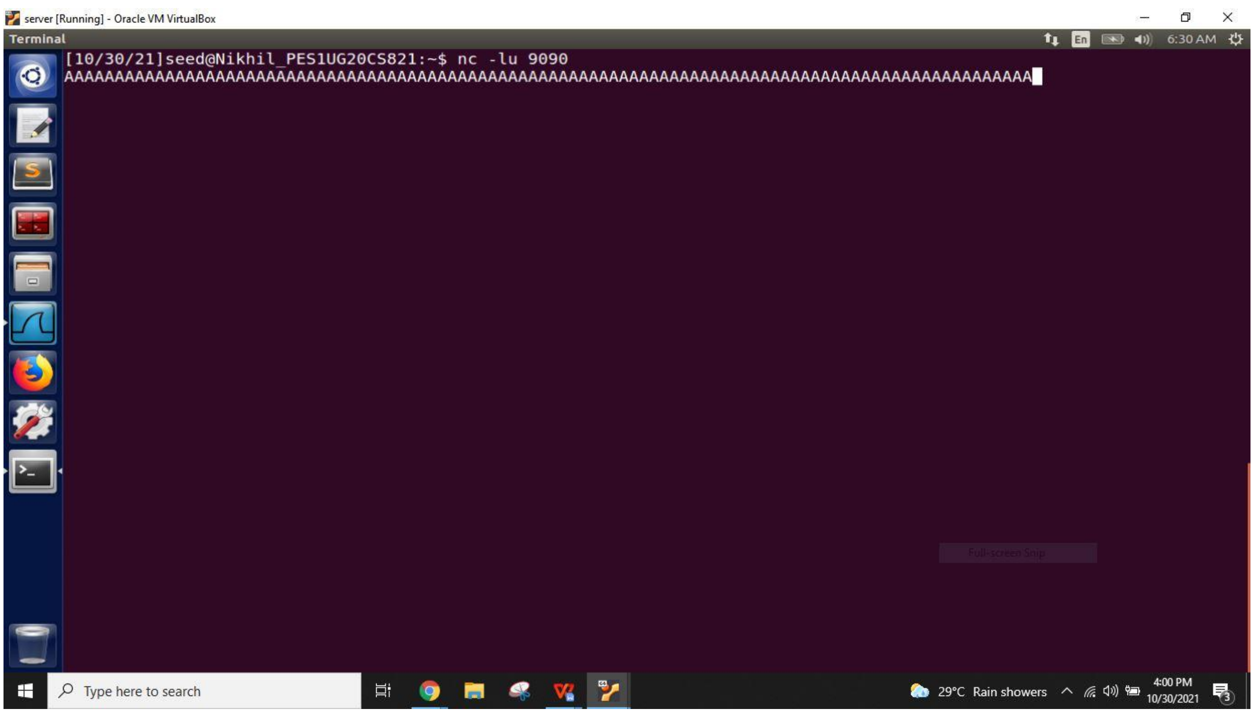
Wireshark configured on UDP Client. We disable the Reassembling fragmented packets feature



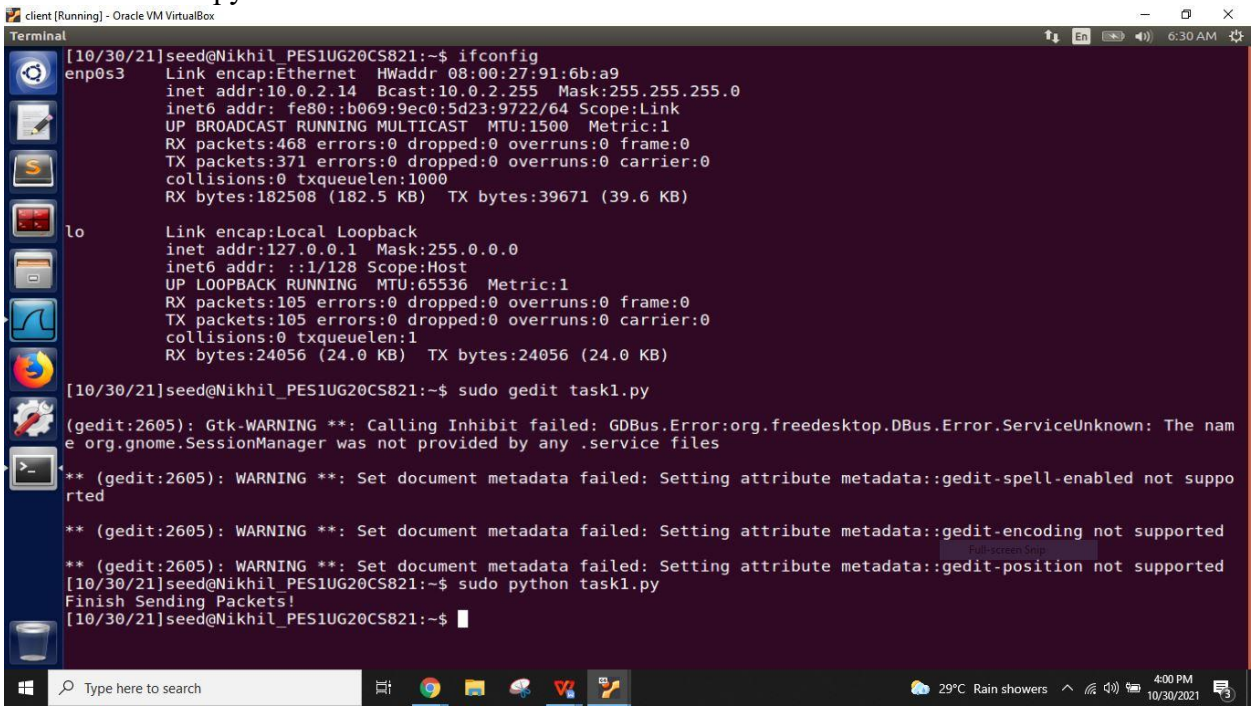
Wireshark configured on UDP Server also.We disable the Reassembling fragmented packets feature



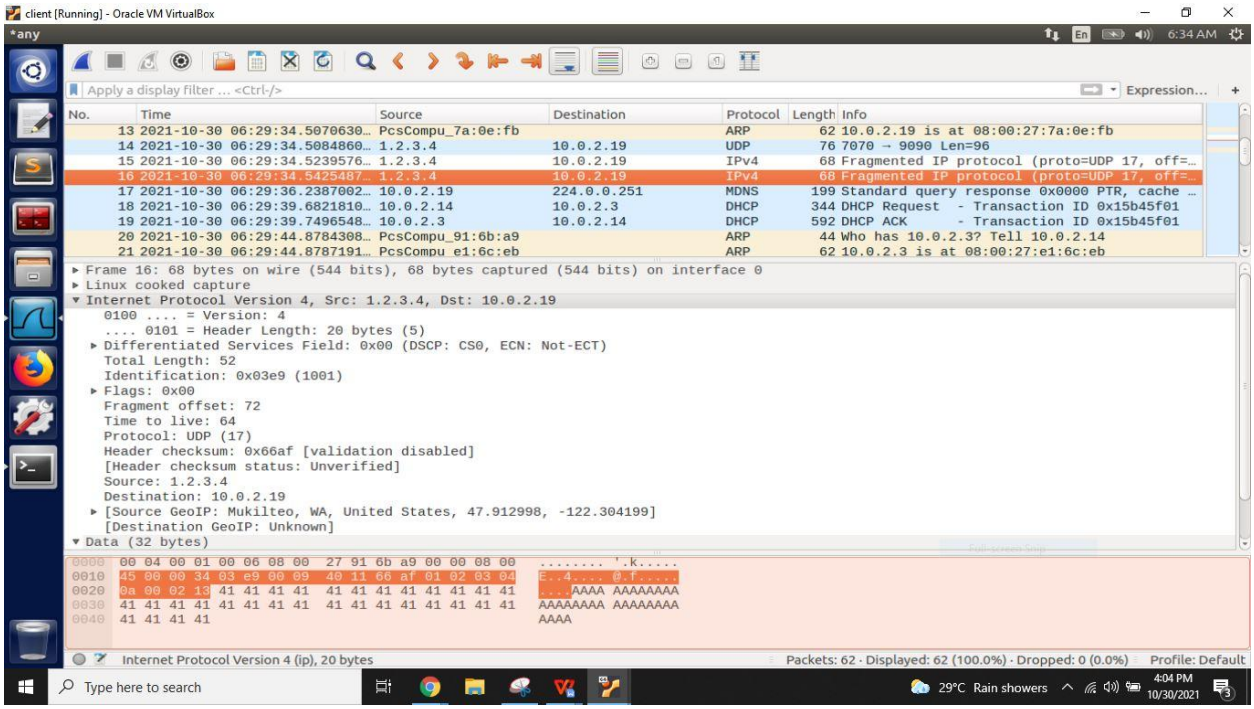
Listen to the port 9090 on UDP Server using netcat



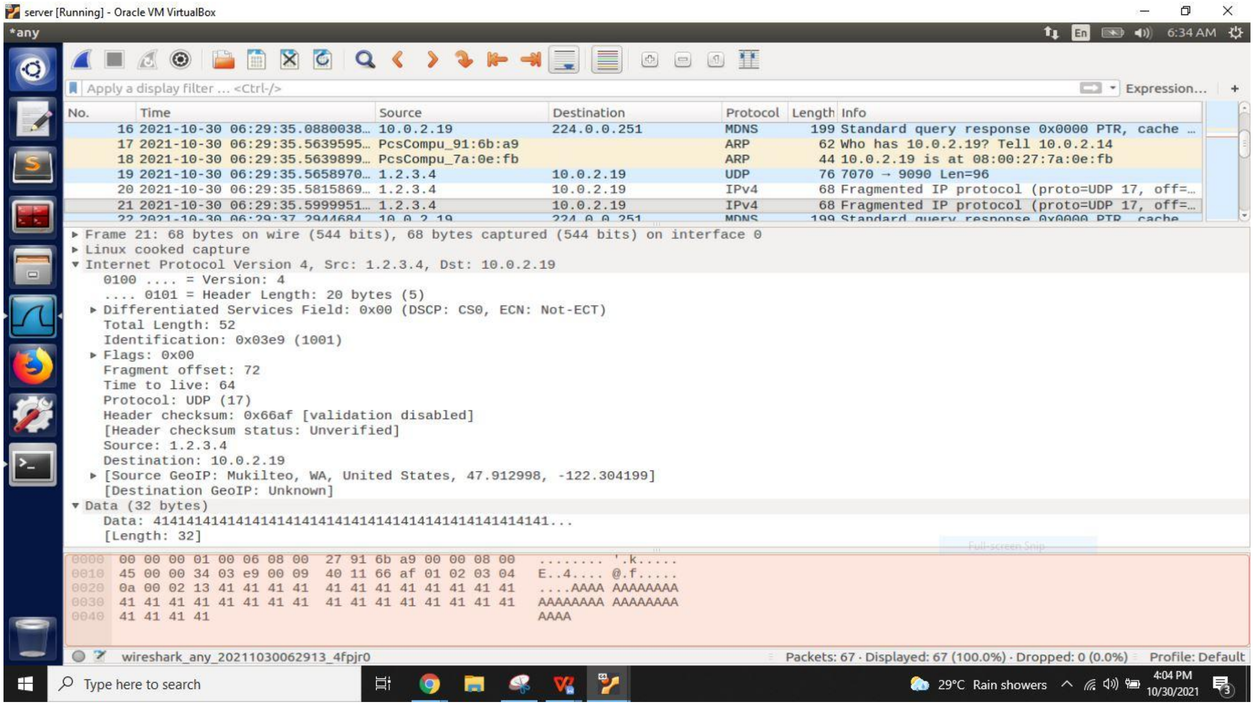
Run the Task1a.py on UDP client



Use wireshark to capture the packets we can see that packets from source 1.2.3.4 is captured by 10.0.2.19 in the UDP client

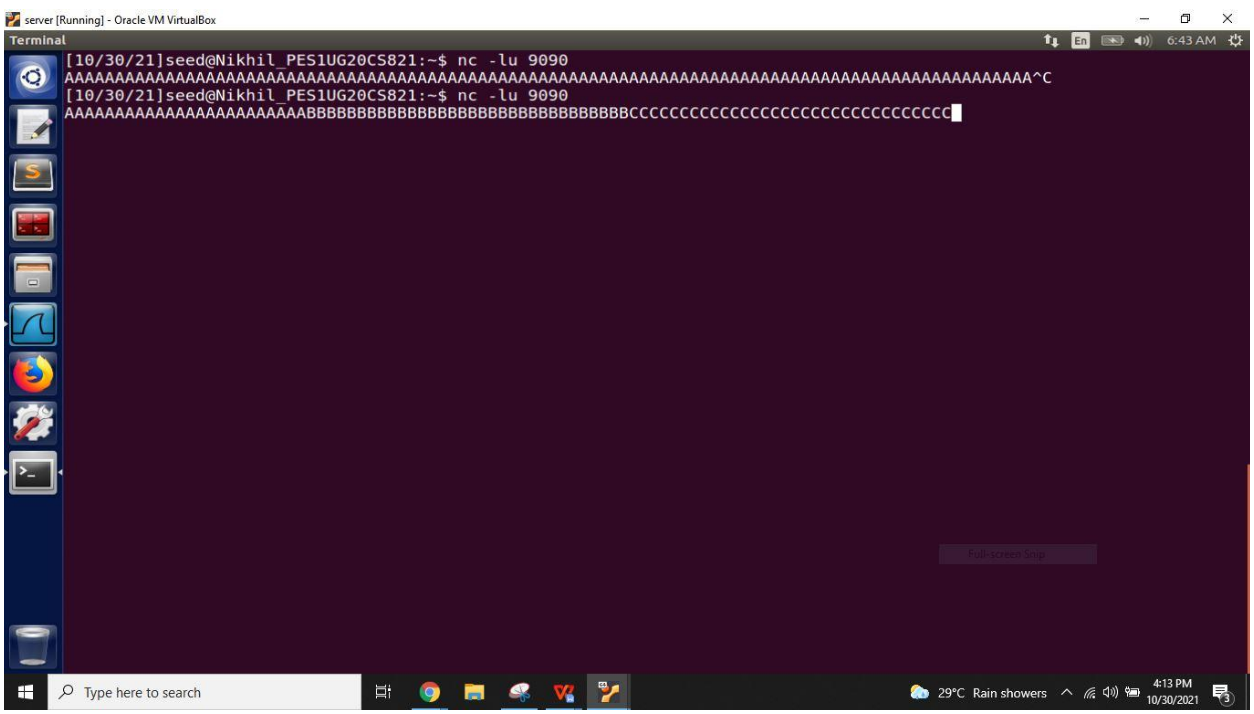


Same observation is seen in the UDP client

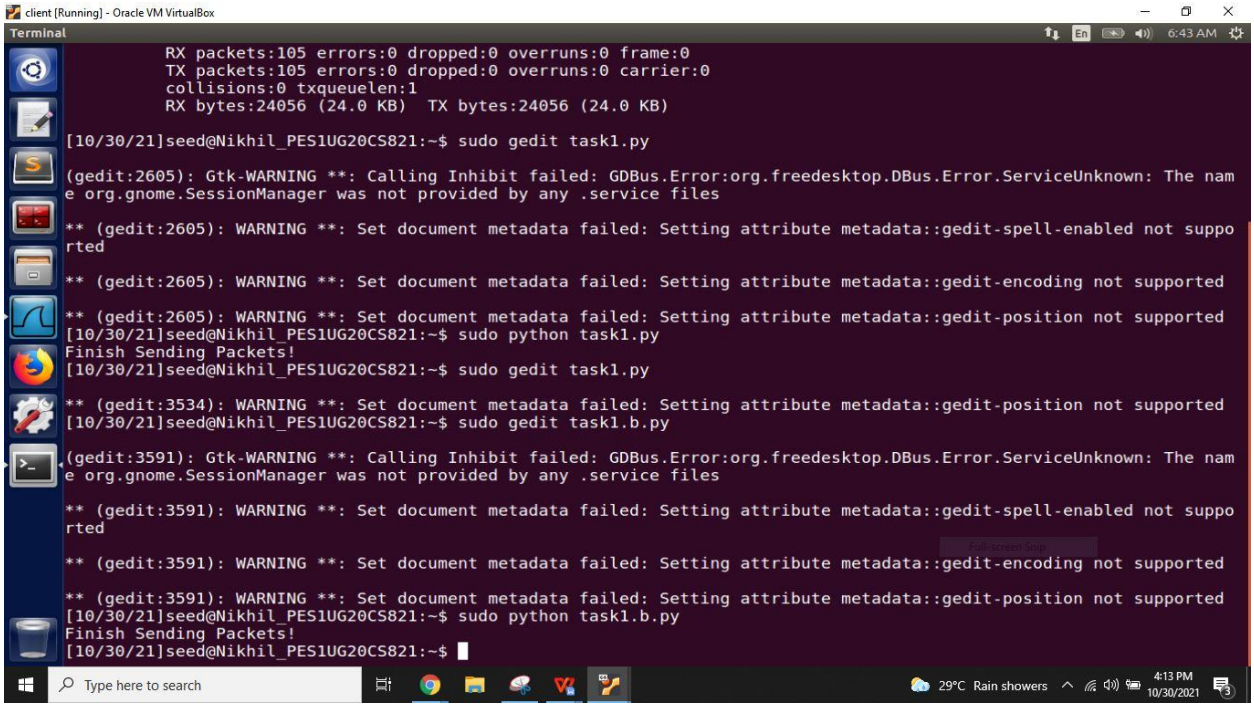


Task 1.b: IP Fragments with Overlapping Contents

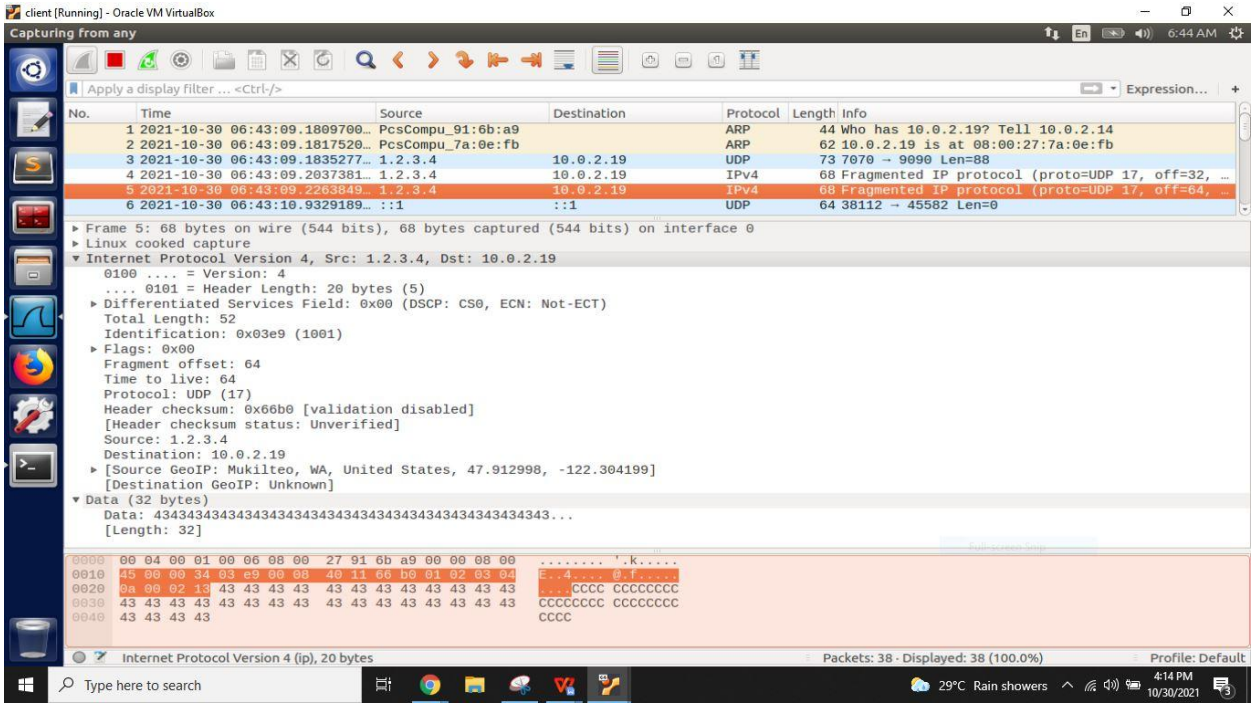
Listen to port 9090 from UDP server through netcat

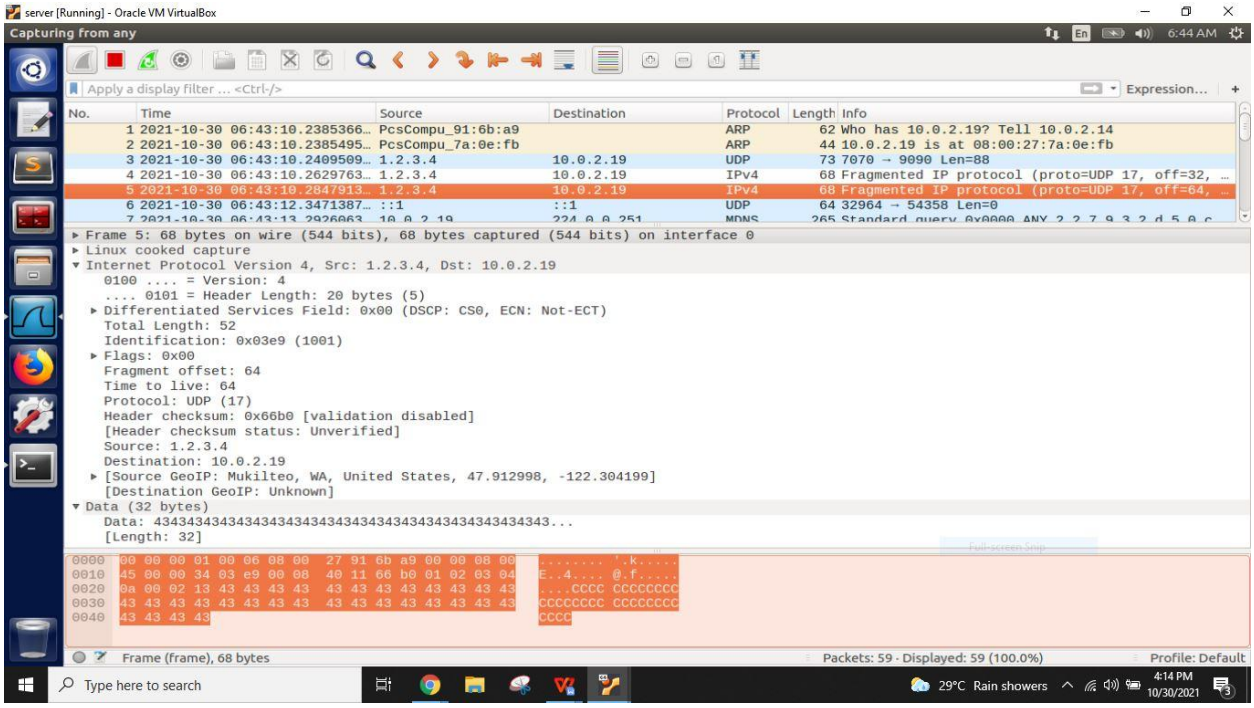


We considered k=15 and Running at the file task1.b on UDP Client.



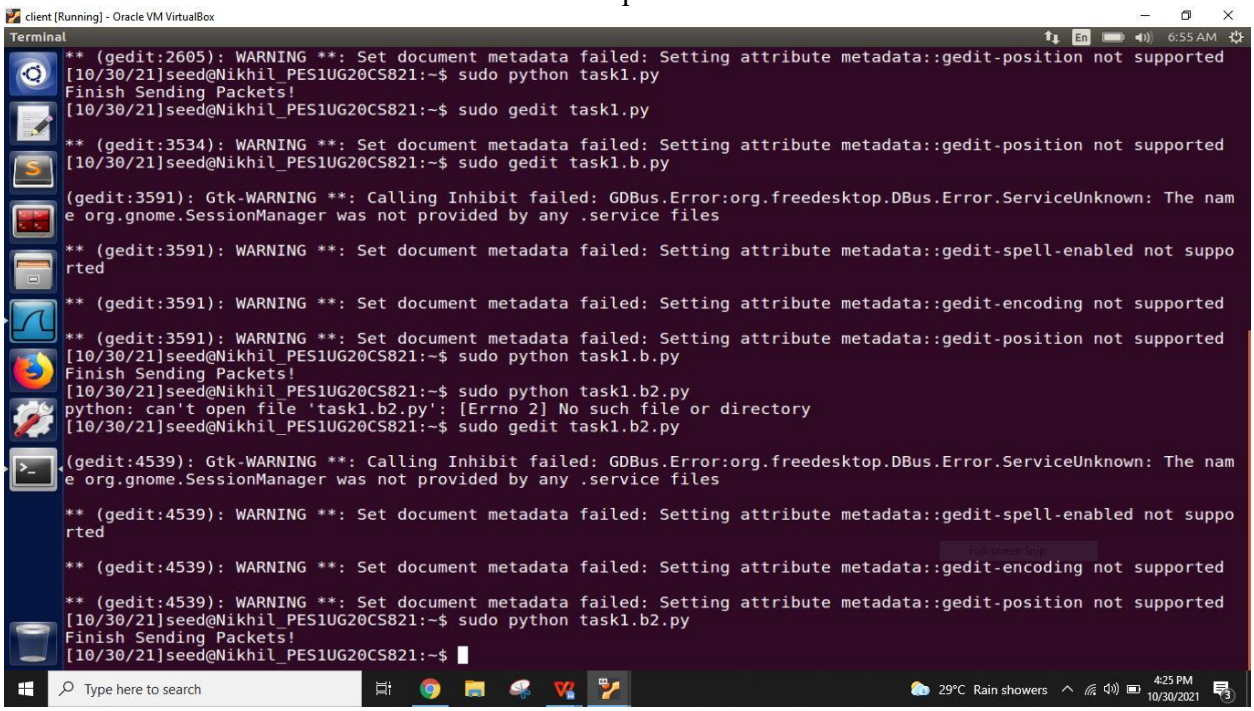
Those packets are captured in the wireshark on both UDP client and server and we can see the same observation on both of the systems ans we can see length is 32



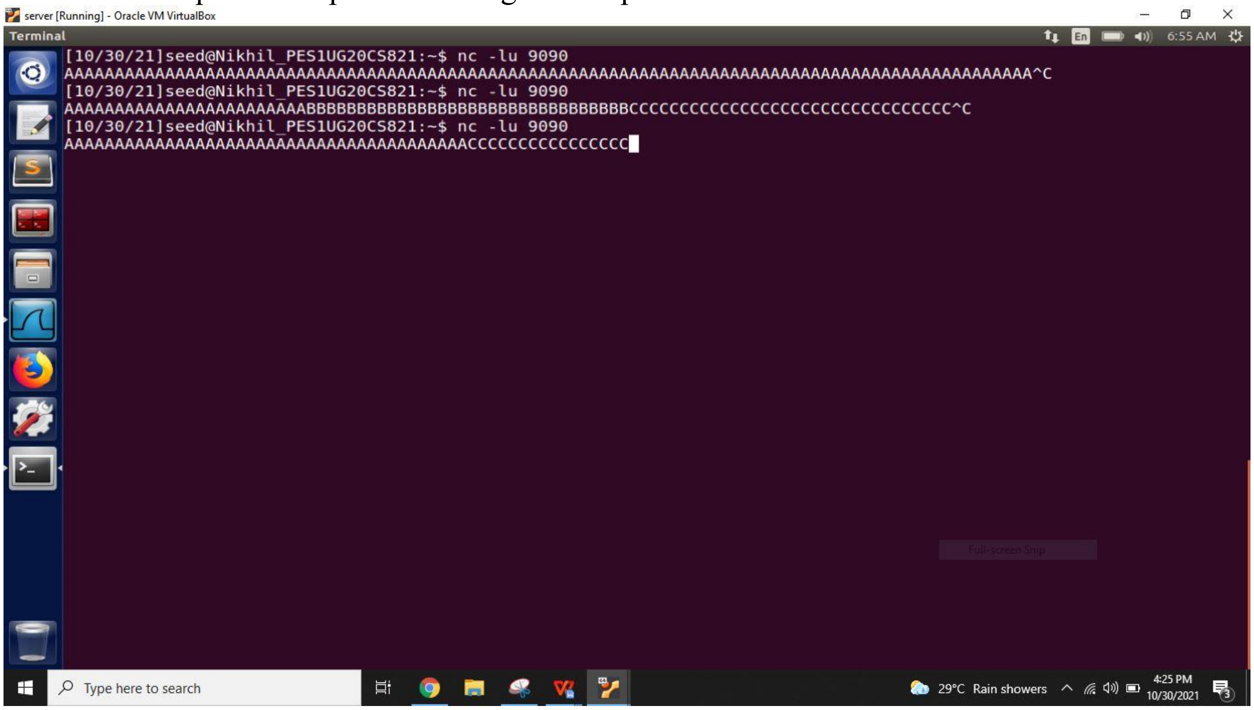


Now we send second fragment is first and then first fragment

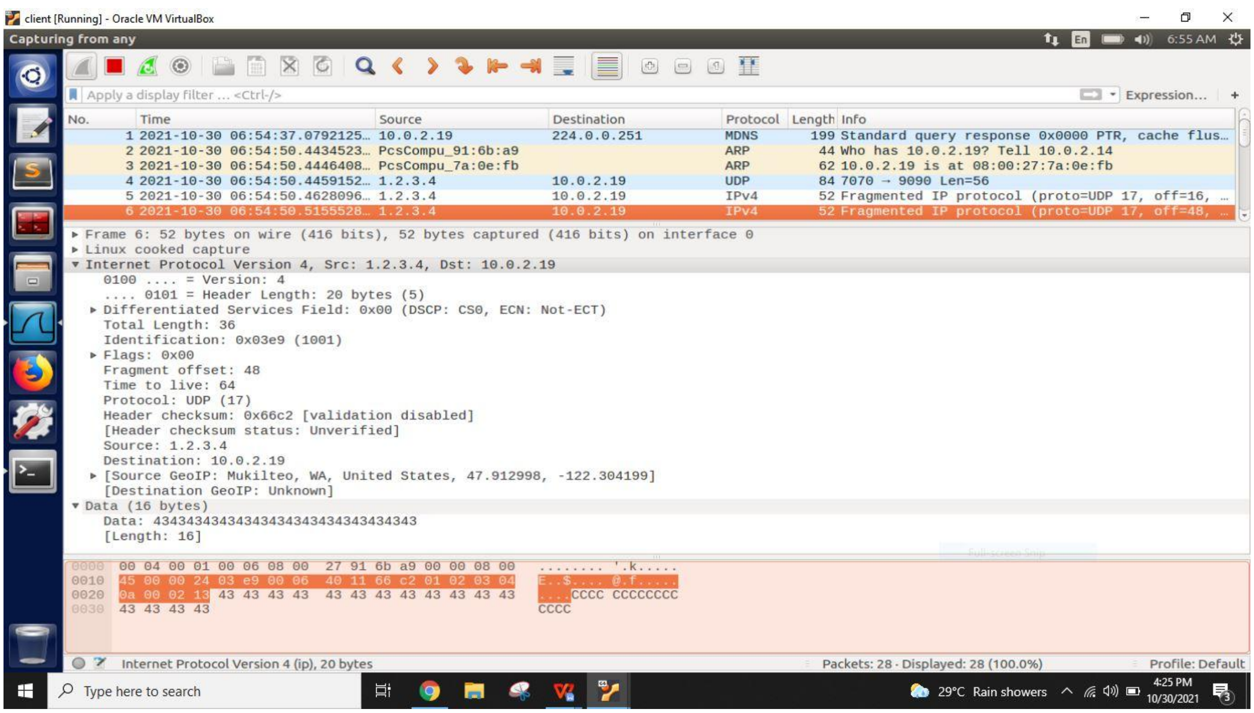
We run the code in the UDP client and send the packets

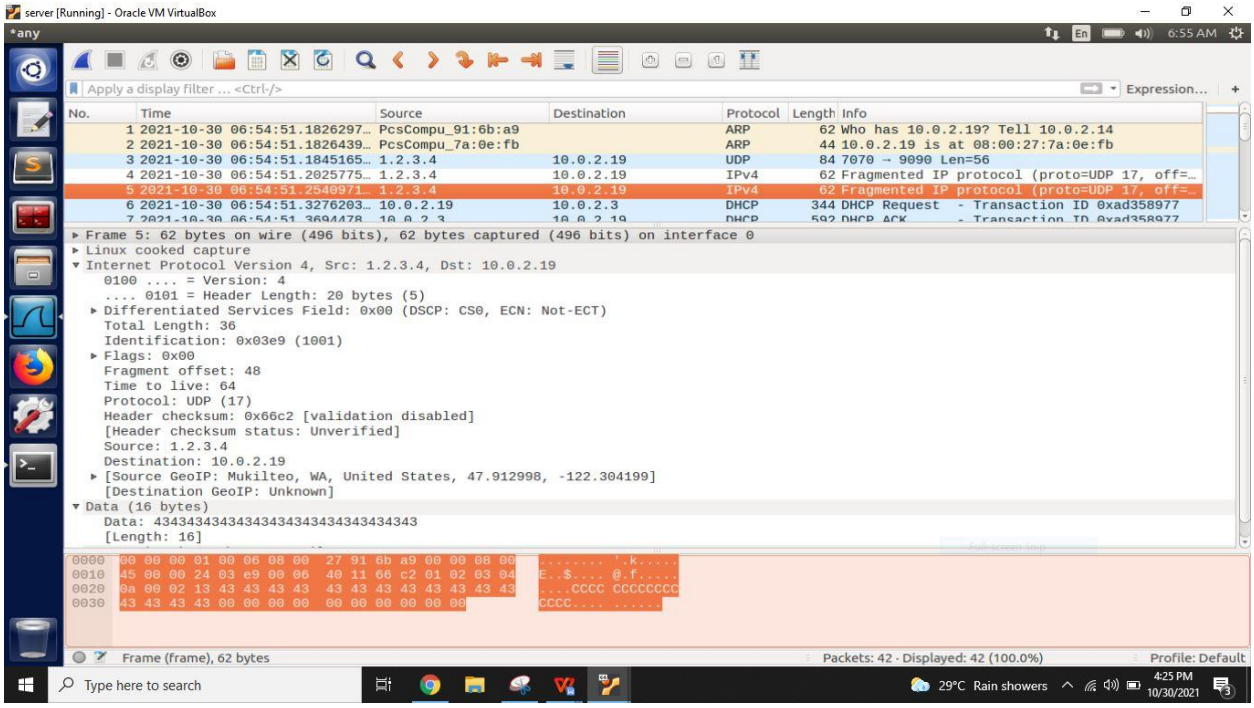


UDP Server captures the packets through 9090 port and observation can be seen



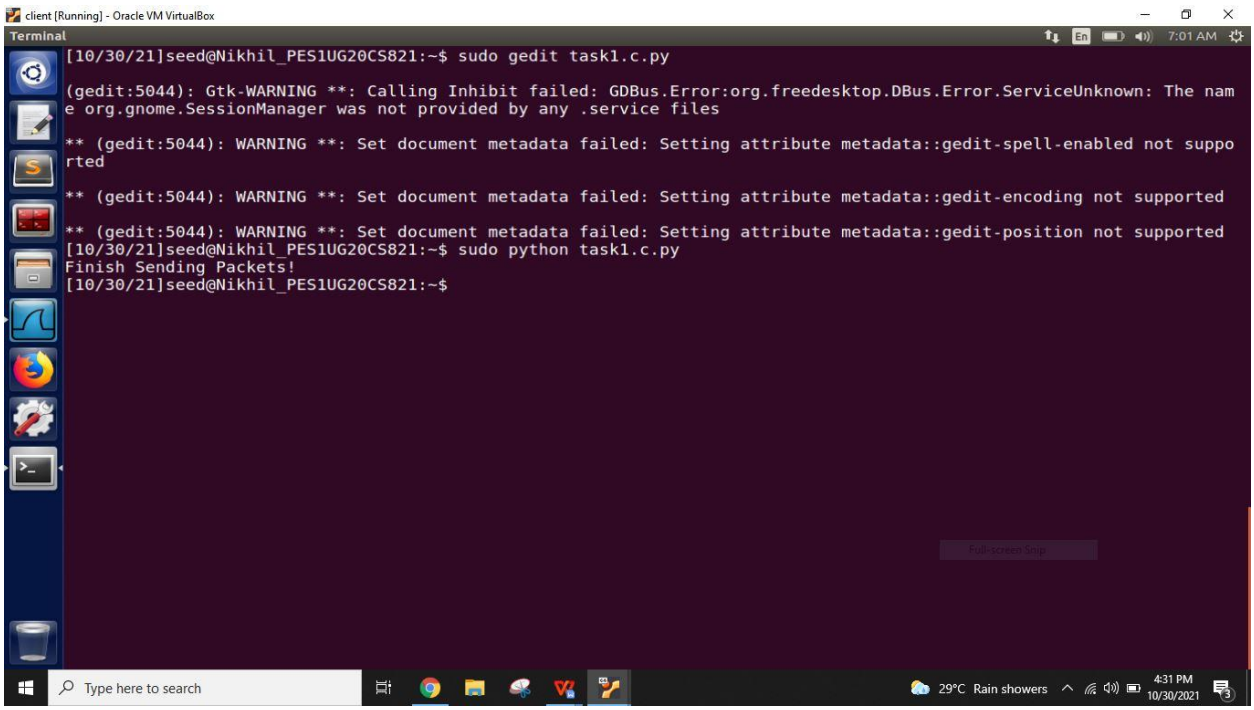
In this task also same observation see on both the client and server system which consists of UDP packets



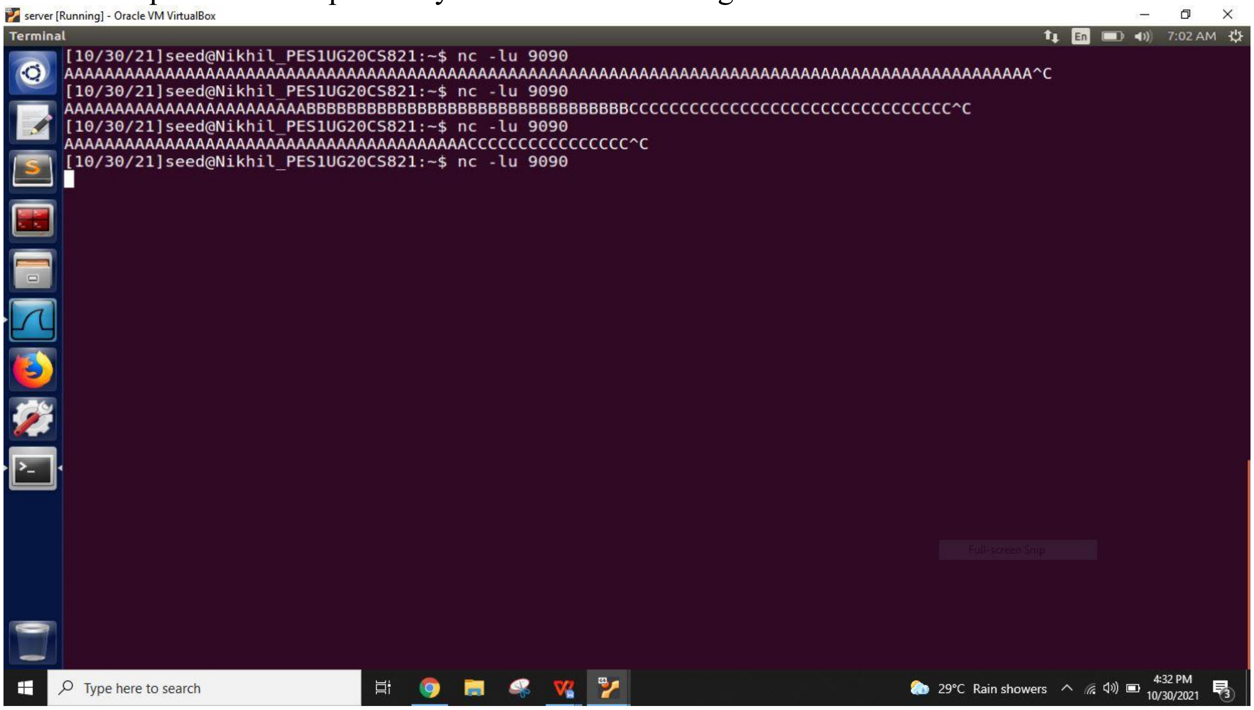


Task 1.c: Sending a Super-Large Packet

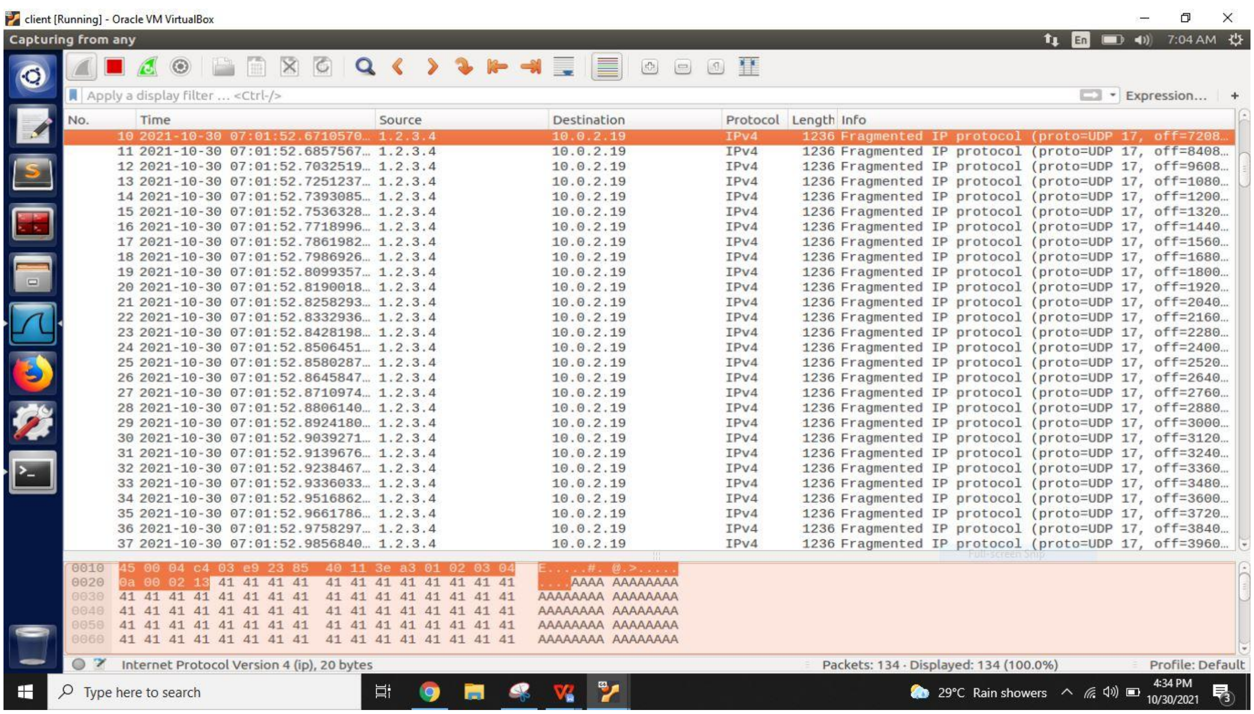
Send super large packet from the UDP client using the file task1.c.py



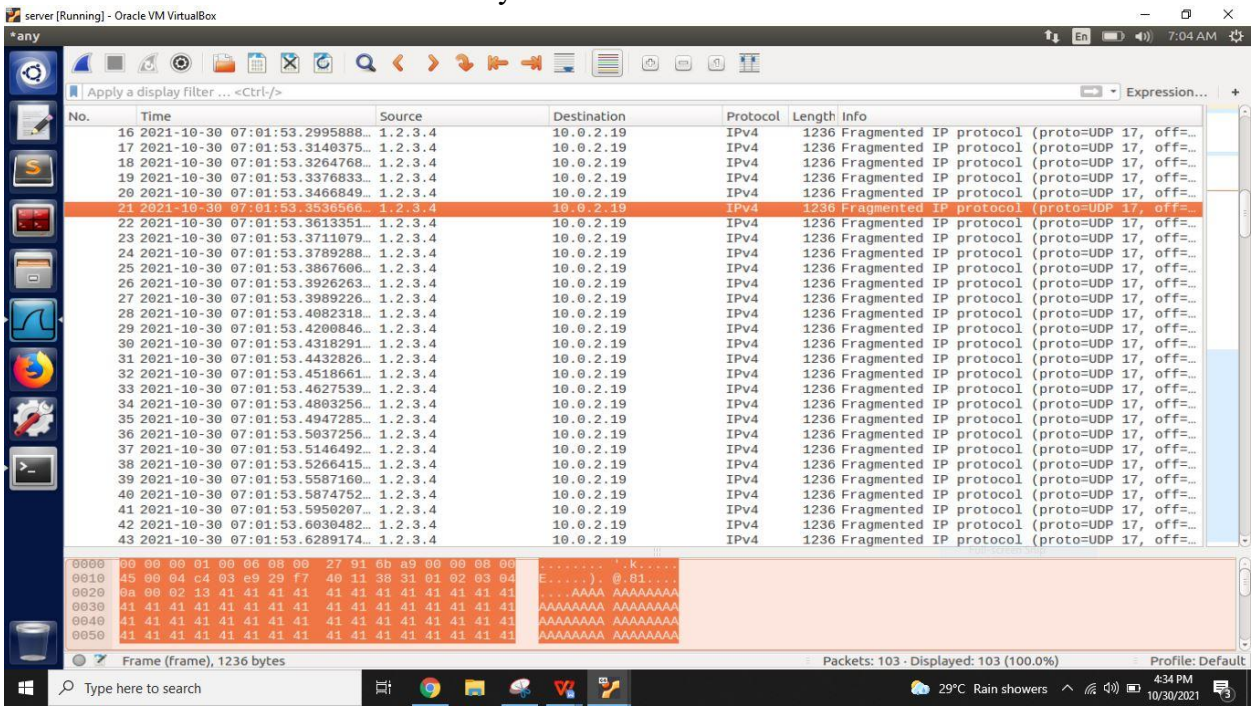
Those sent packets are captured by the UDP server using netcat



Wireshark observation clearly shows that the huge number of packets are sent to the server (10.0.2.19)

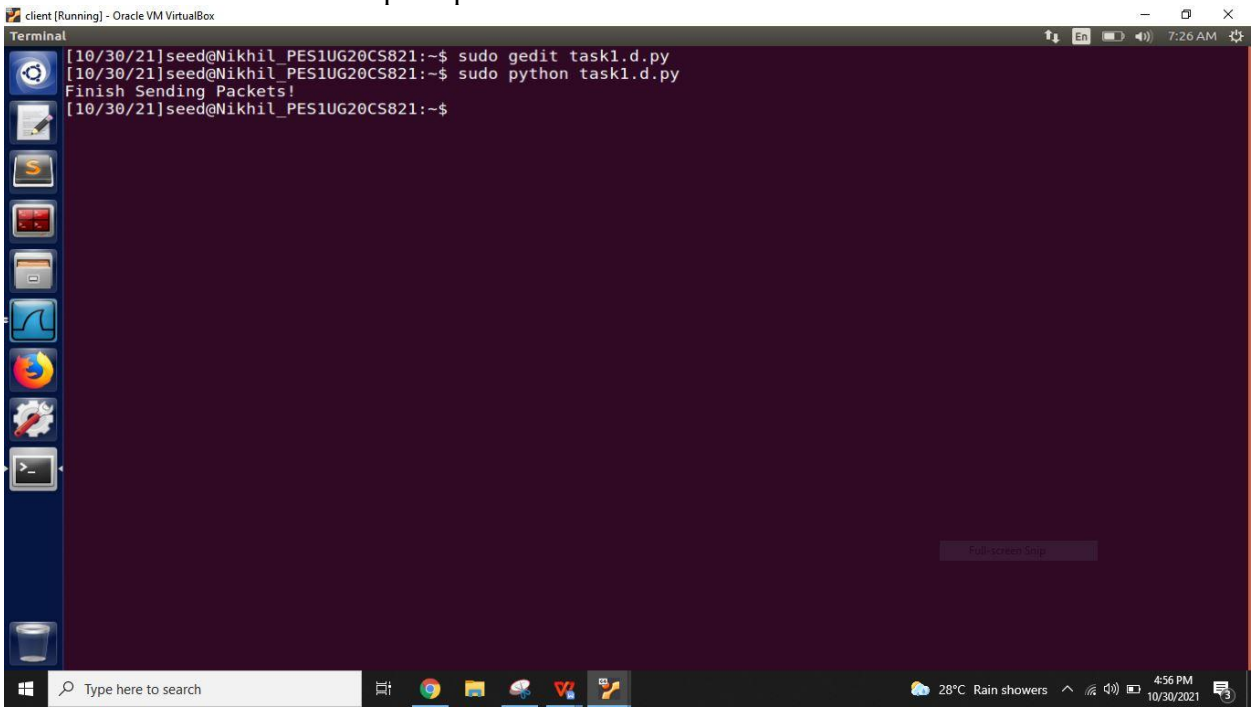


Same observation seen in both of the systems

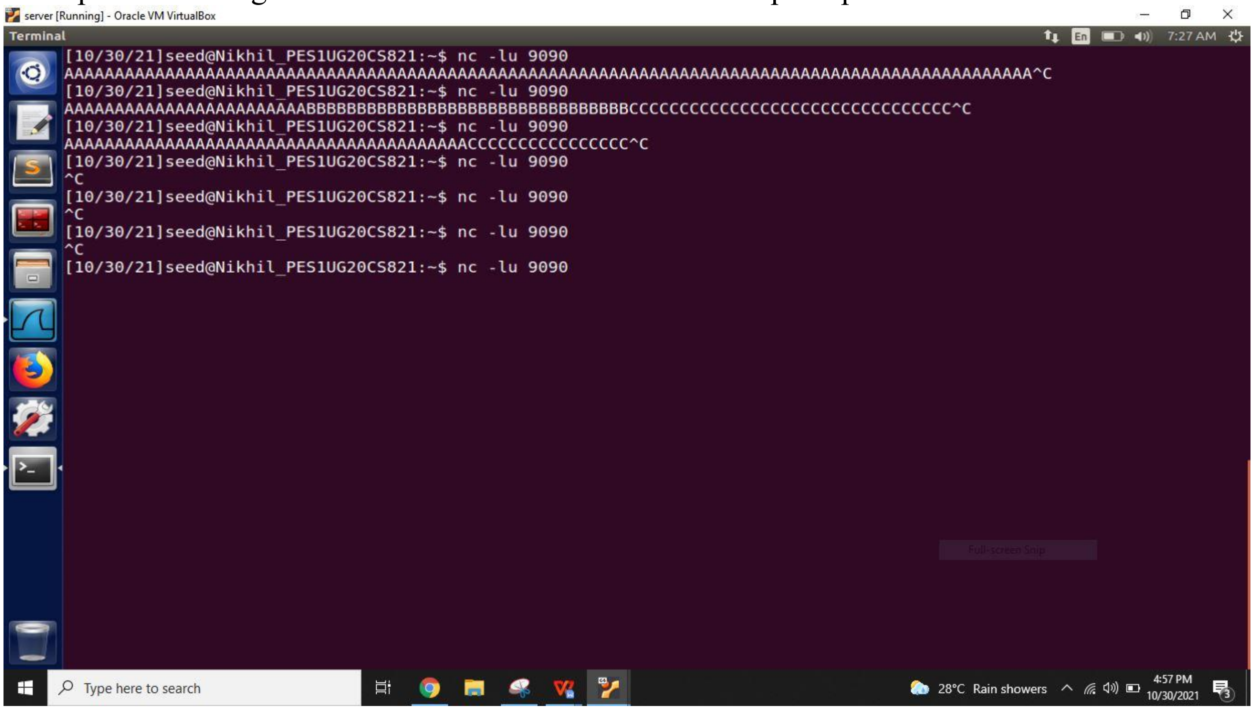


Task 1.d: Sending Incomplete IP Packet

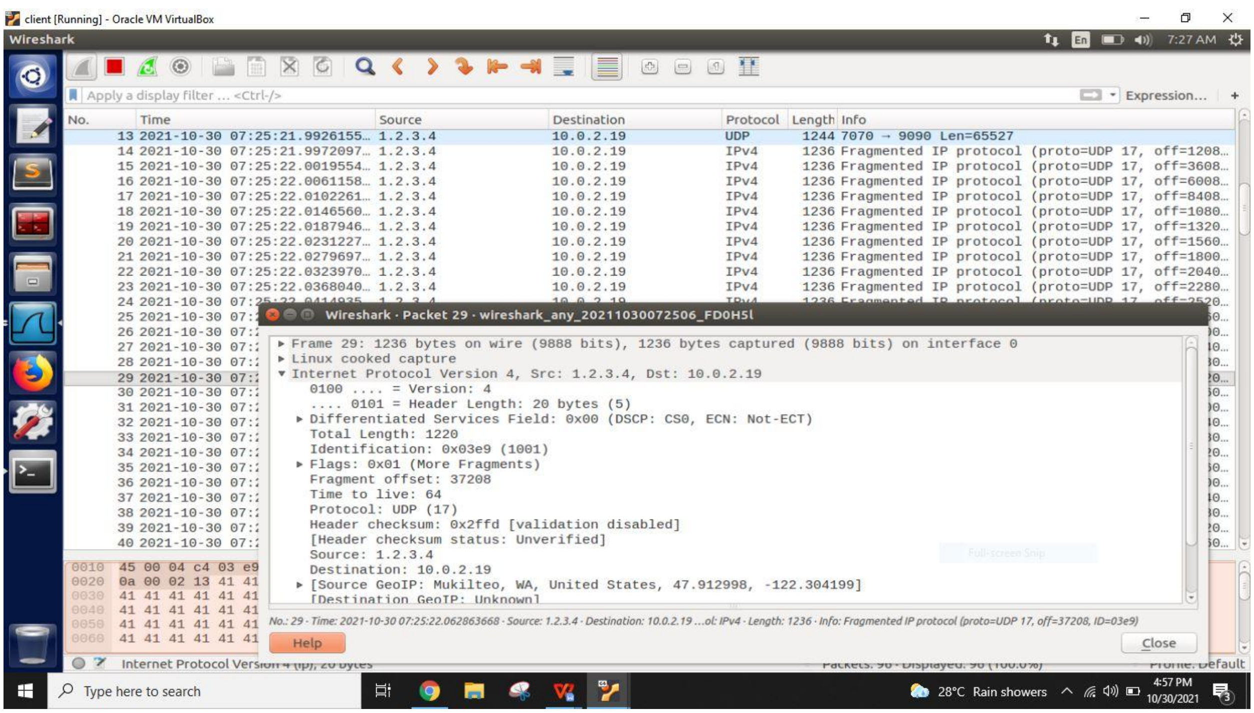
In this task we send an incomplete packet from the UDP client



As expected nothing is shown on the UDP server as no complete packet is sent



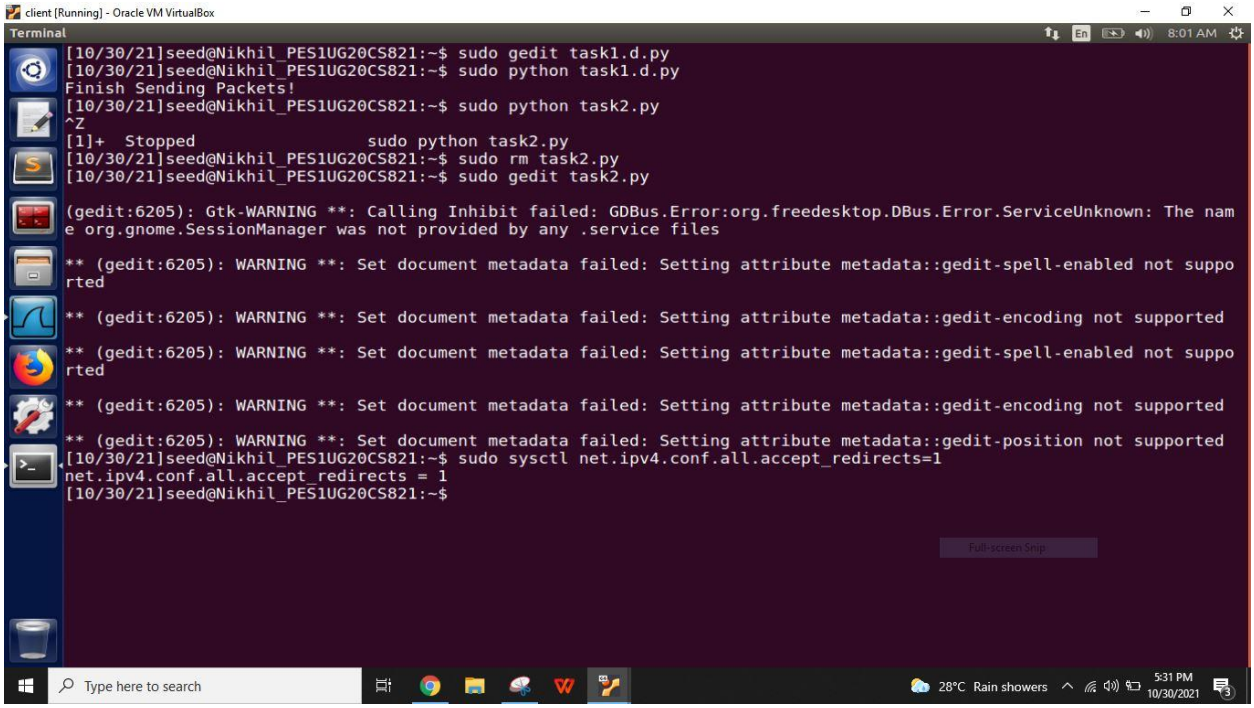
Wireshark has captured all the ip fragments



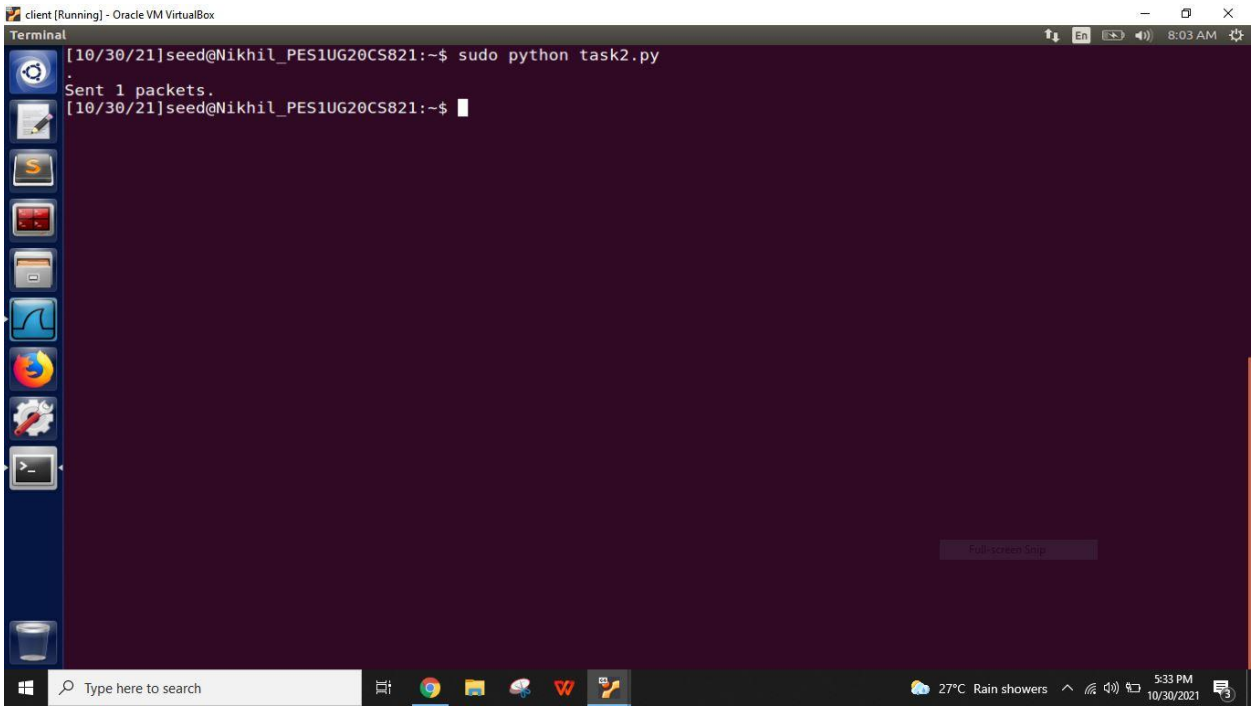
By sending incomplete packets we are going to flood the kernel memory and perform the dos attack

Task 2: ICMP Redirect Attack

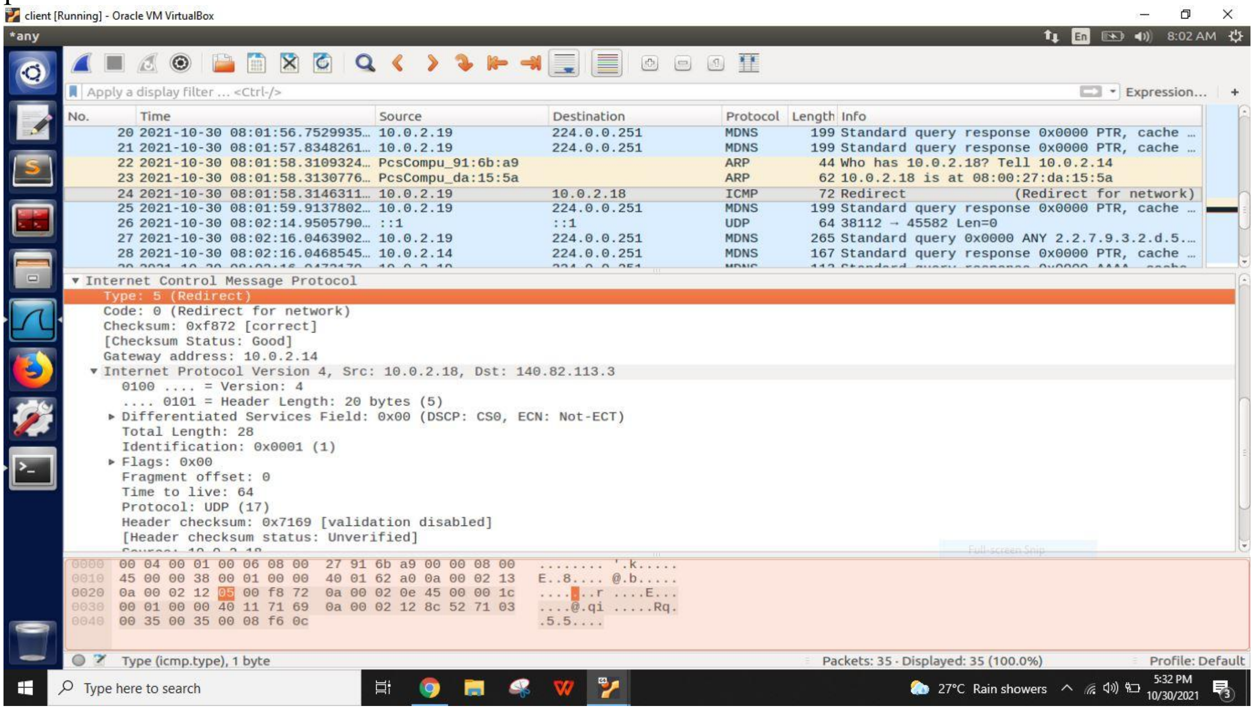
Remove the countermeasure



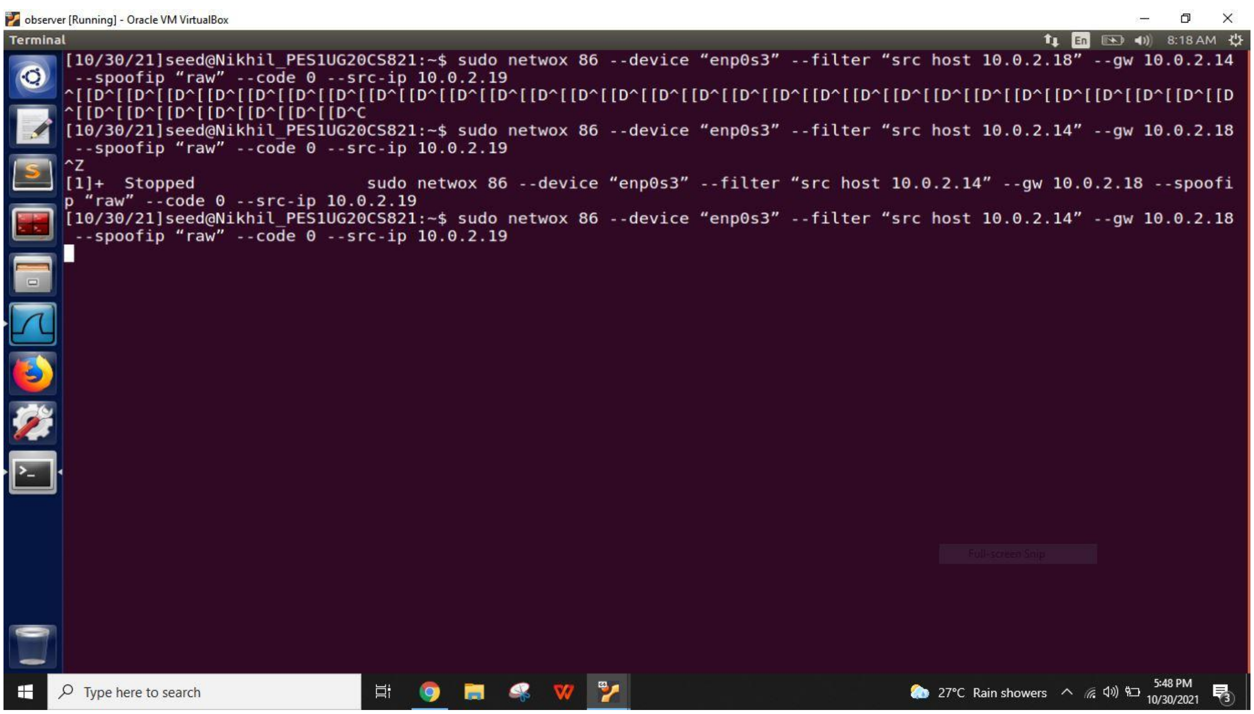
Send packet by running task2.py on the UDP client



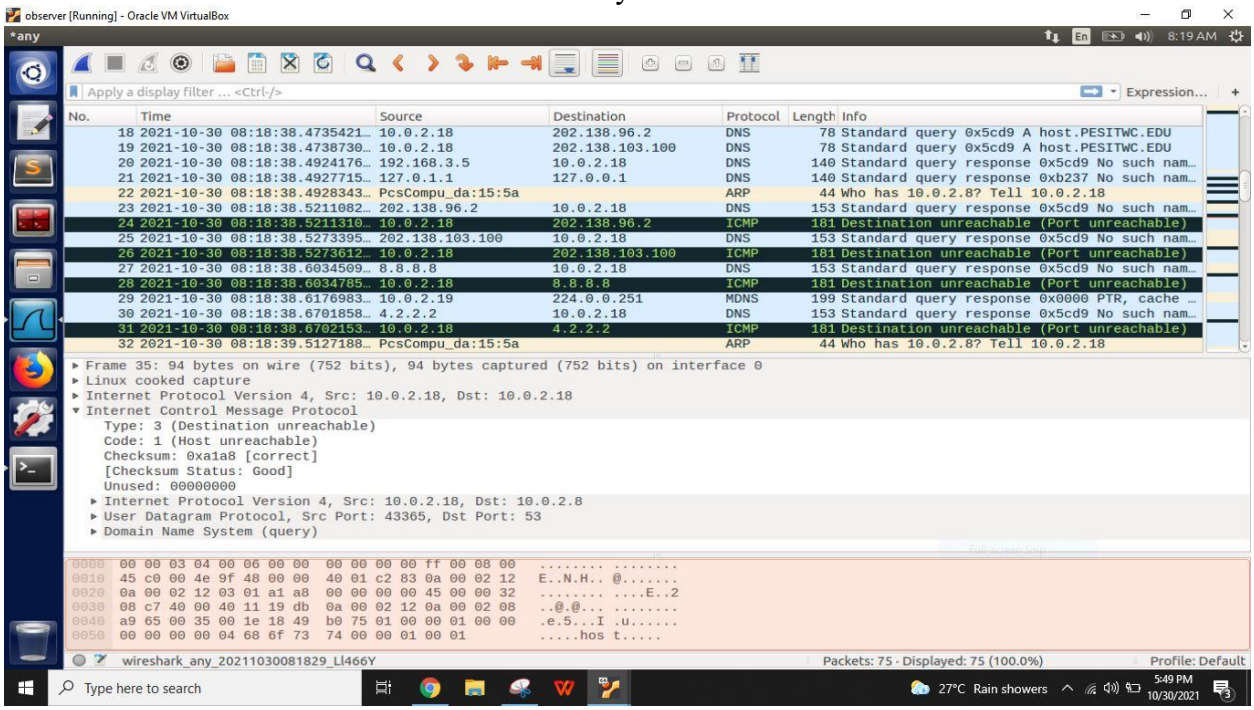
On wireshark we can see that icmp packet type is 5 which indicates the redirect of the icmp packet



Same above task we repeated but by using netwox tool. Run the netwox command in the UDP client

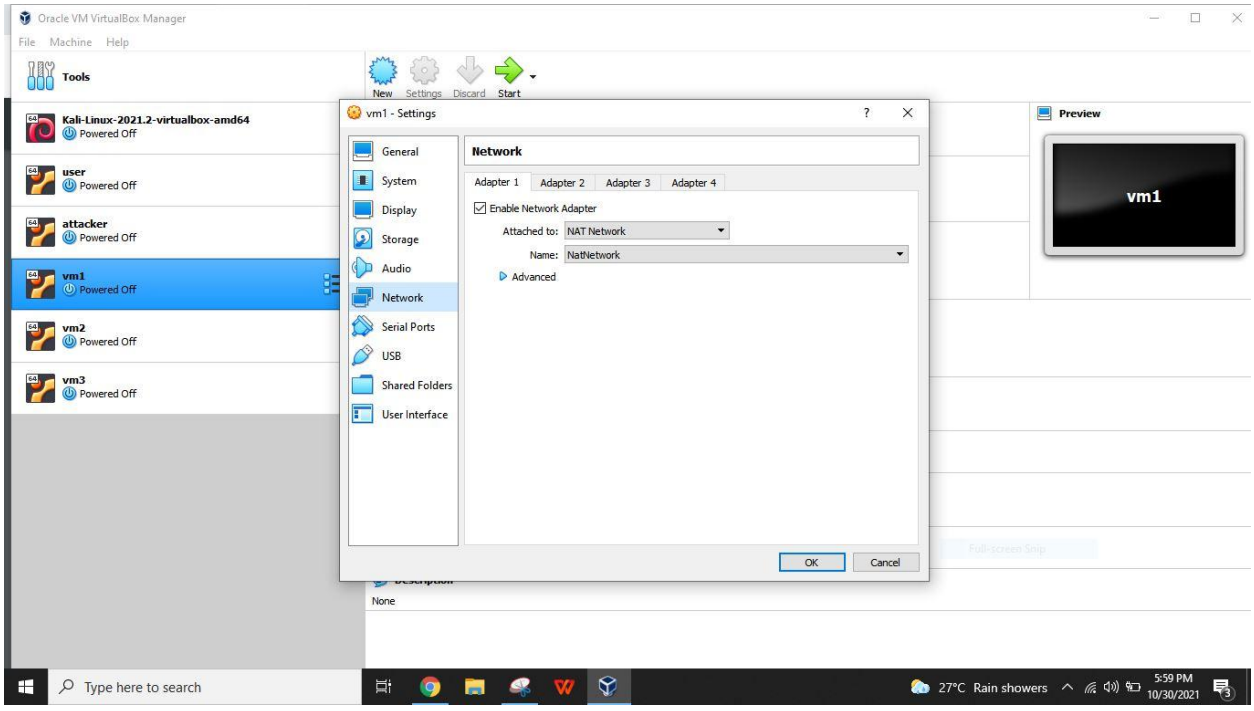


Wireshark observation is seen on the observer system



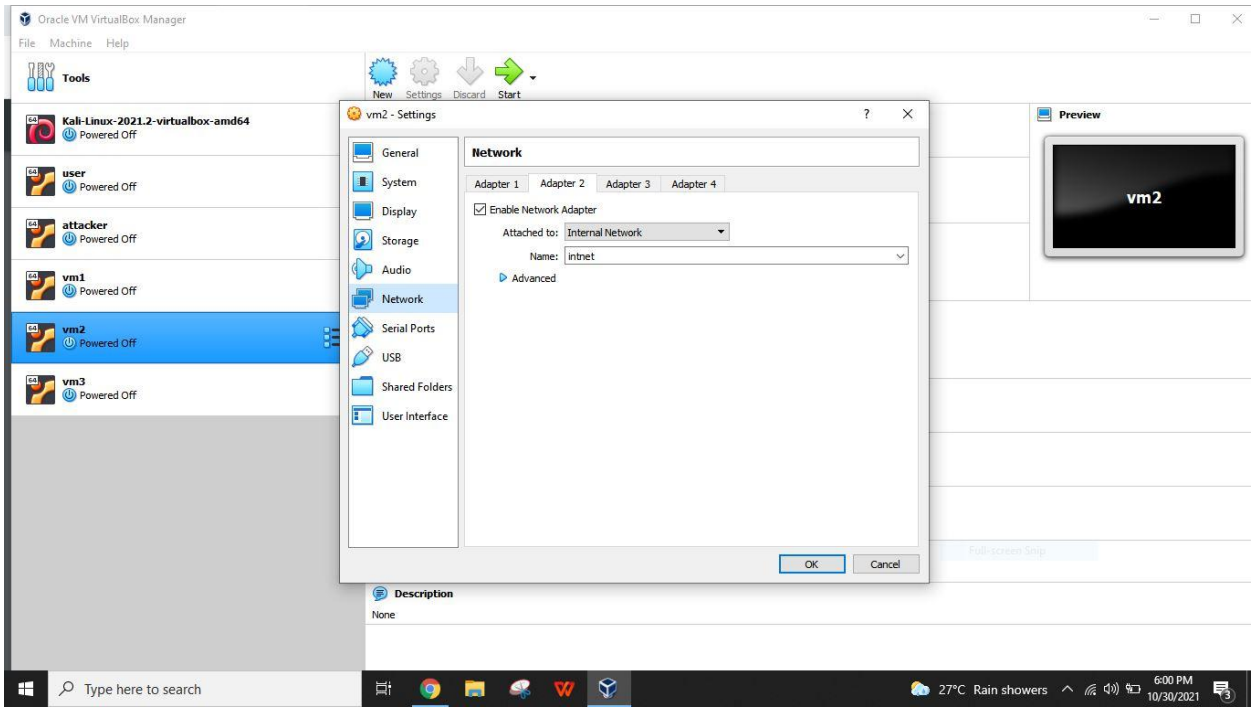
Task3: Routing and reverse Path filtering
Task 3a: Network setup

Set one nat network interface on the vm1

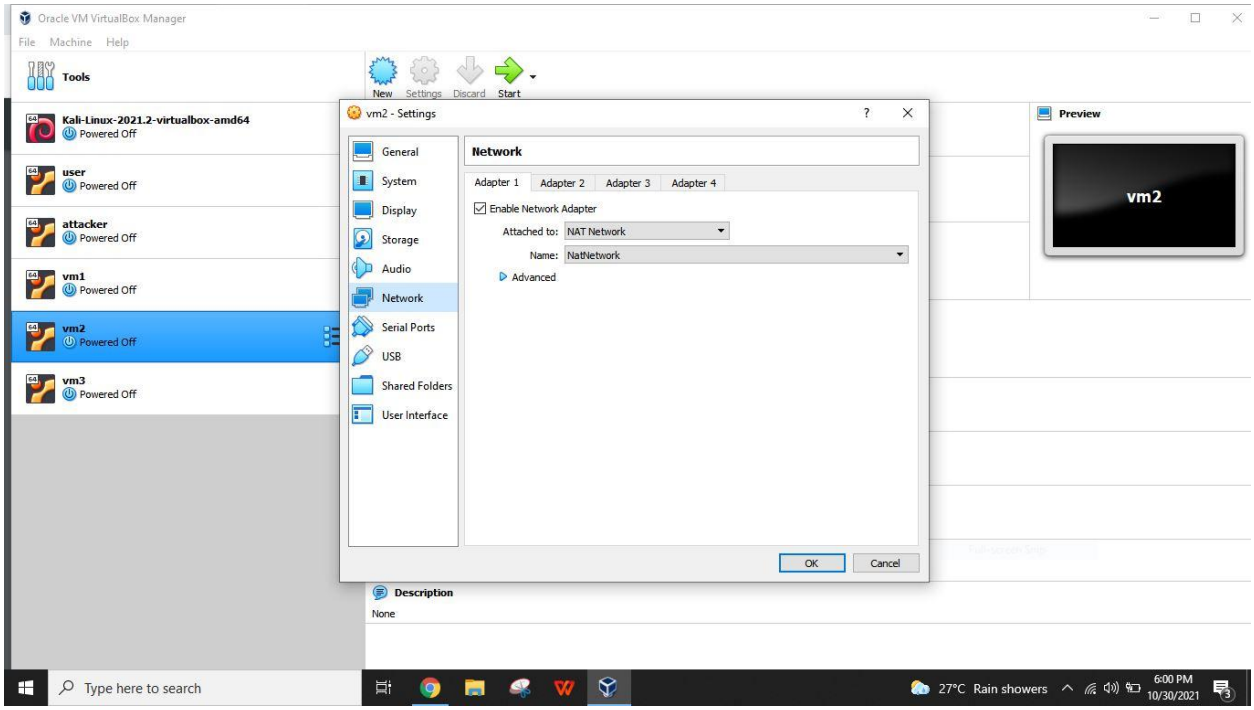


Set two interface in the vm2

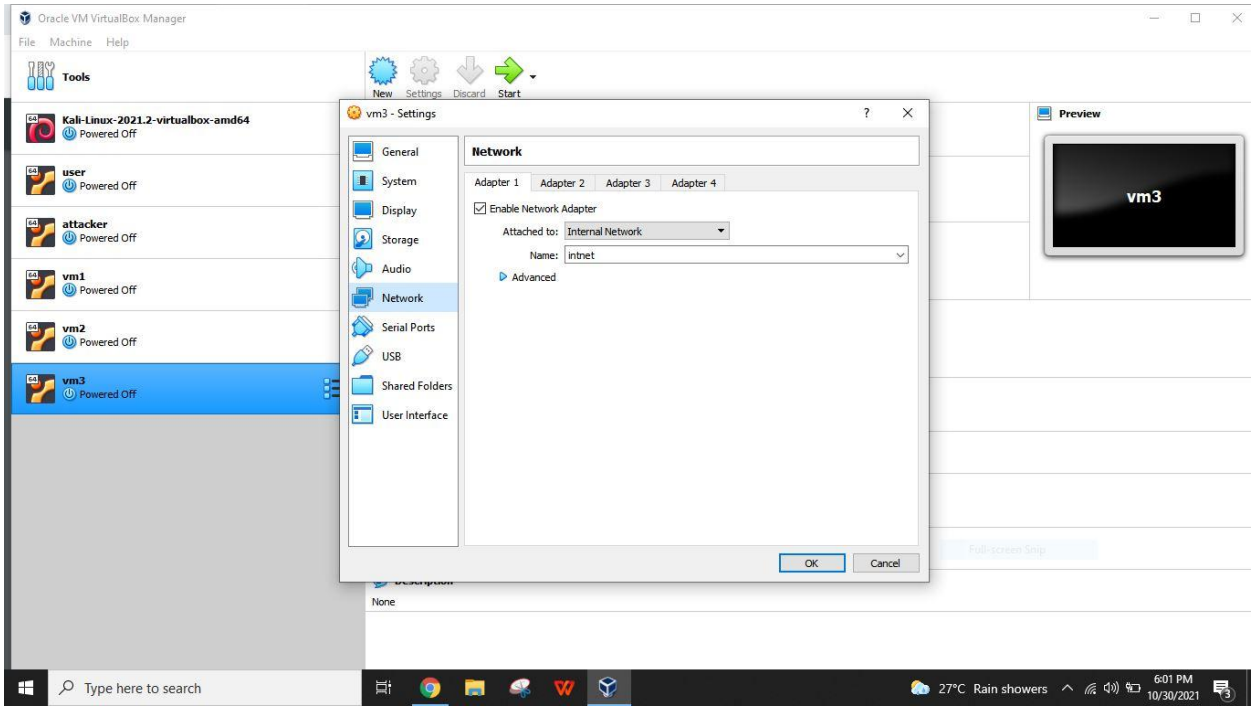
i) internal network



ii)NAT network

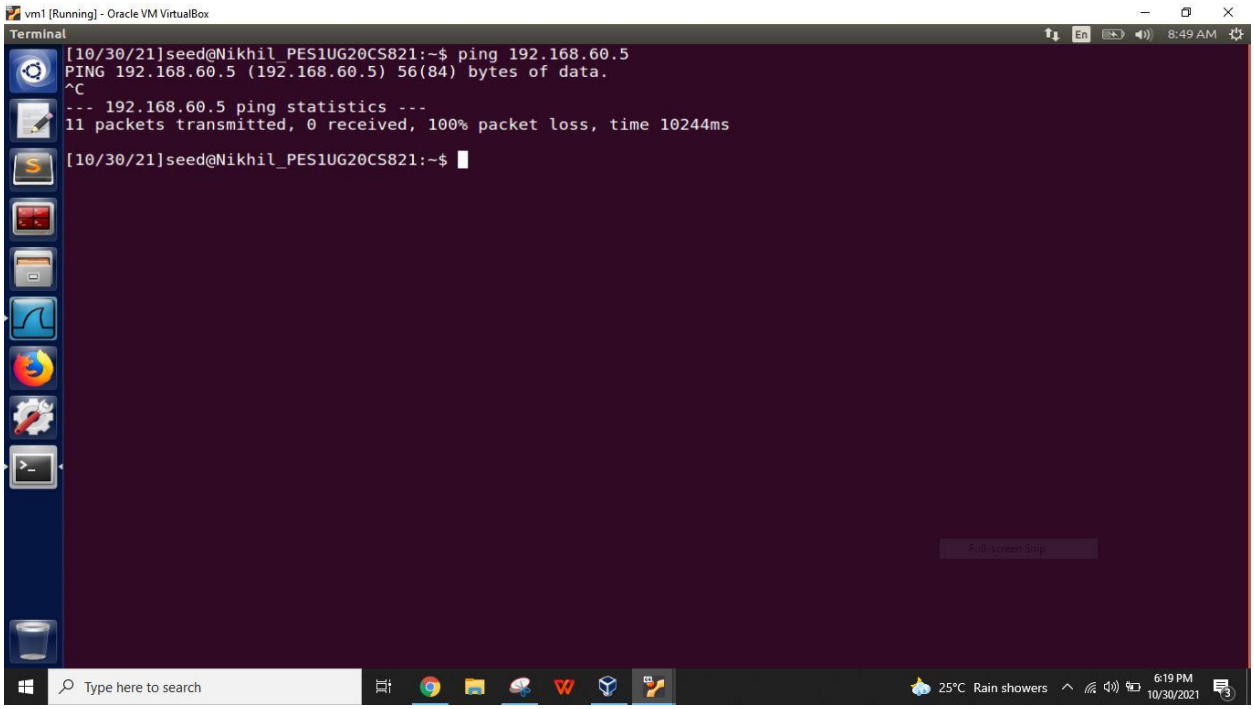


Only internal network interface is set on vm3

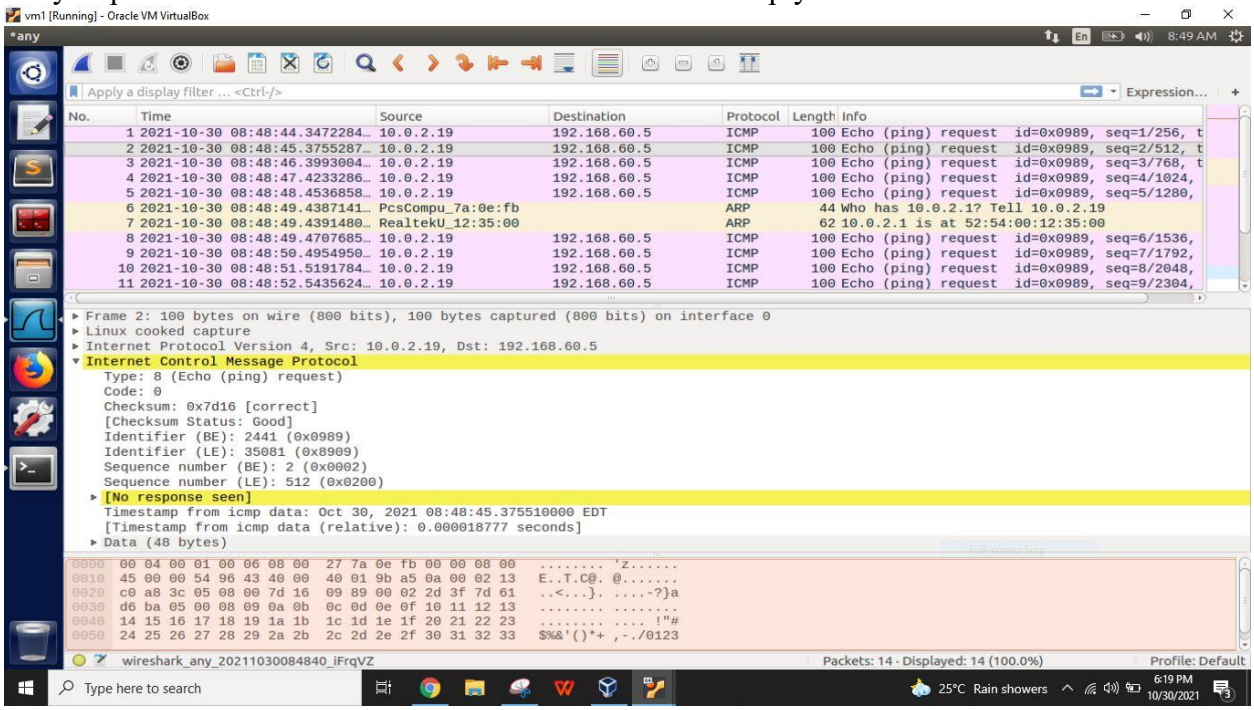


Task 3B: Routing setup

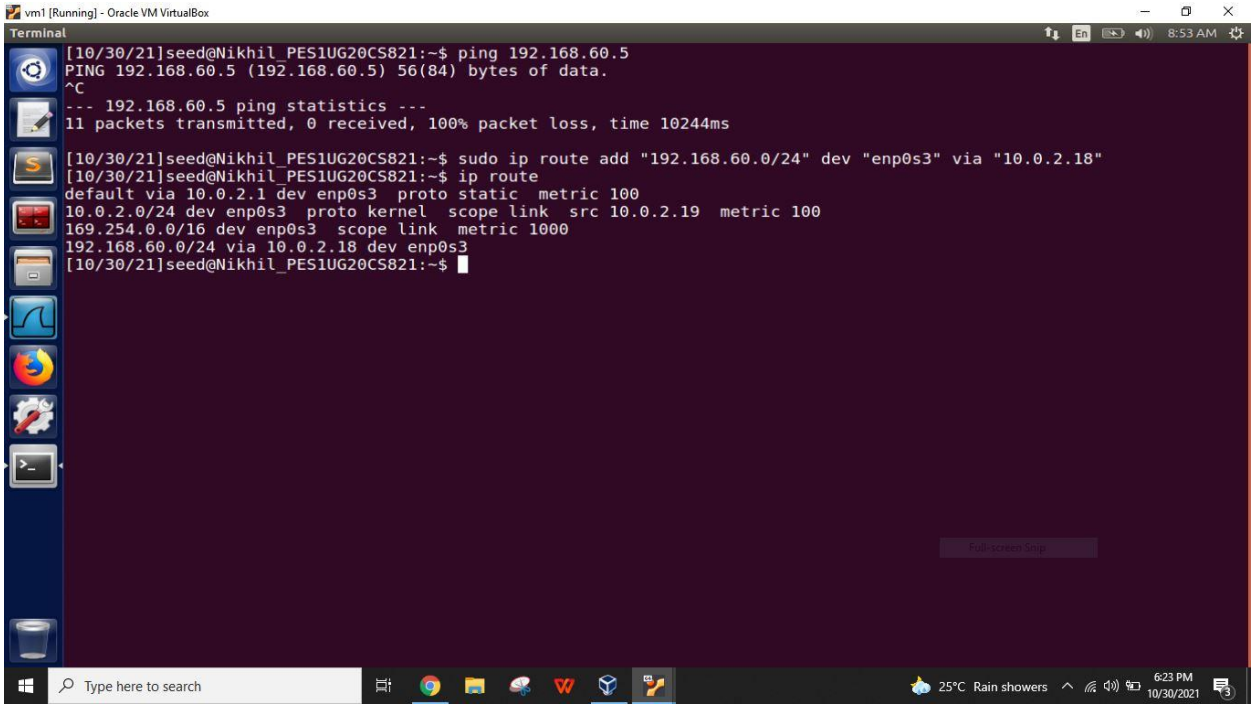
Vm1 cannot communicate with the vm3 direct as we haven't done any routing setup



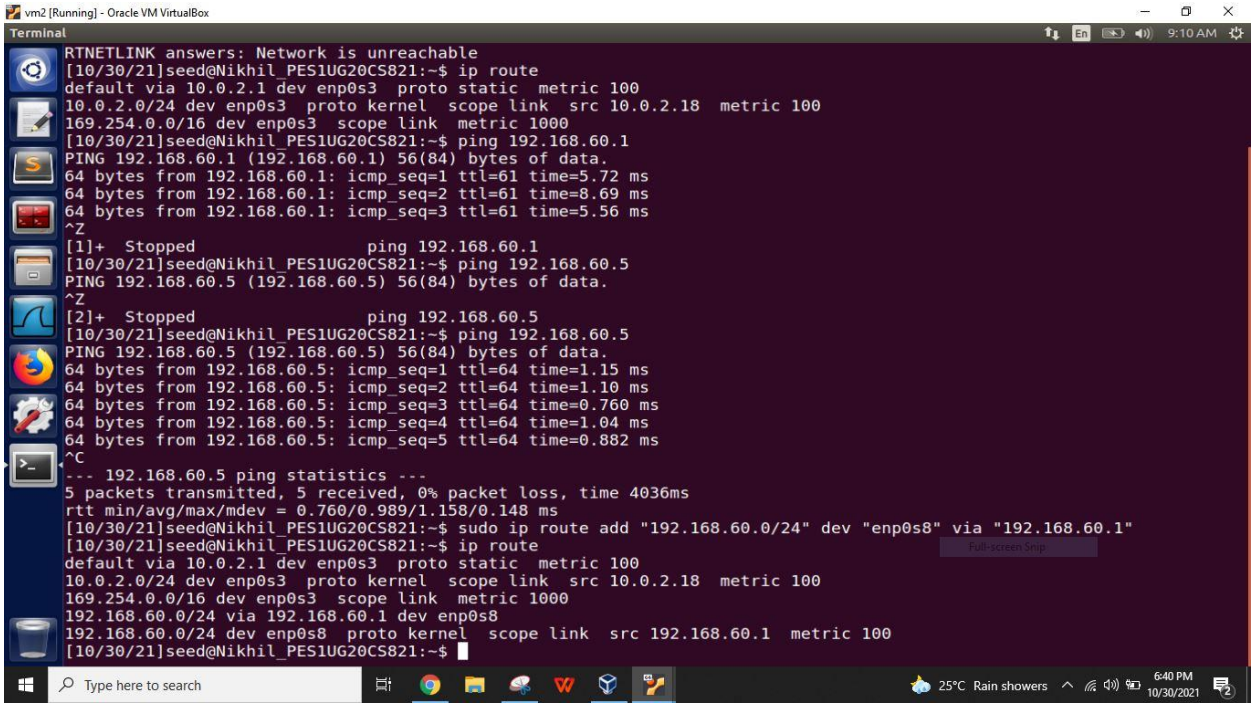
Only request can be seen in the wireshark observation no reply from the vm3



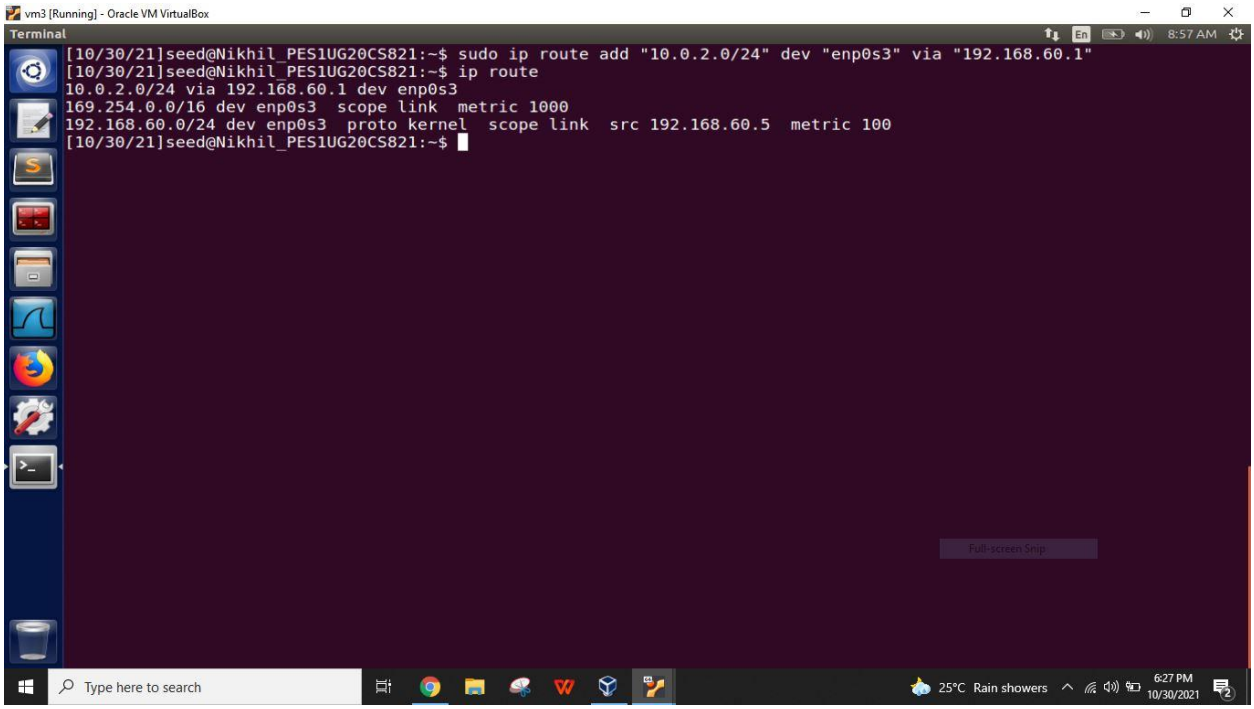
Now we set the routing path from vm1 to vm2



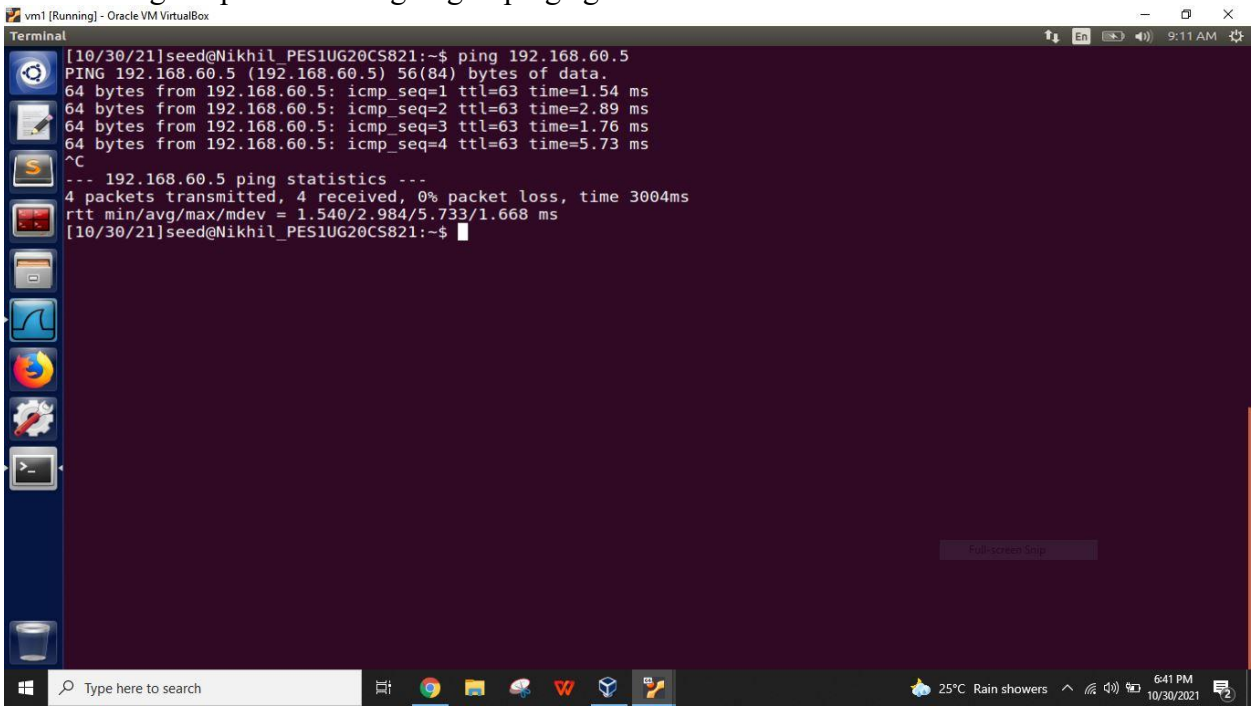
We set path from vm2 to vm3



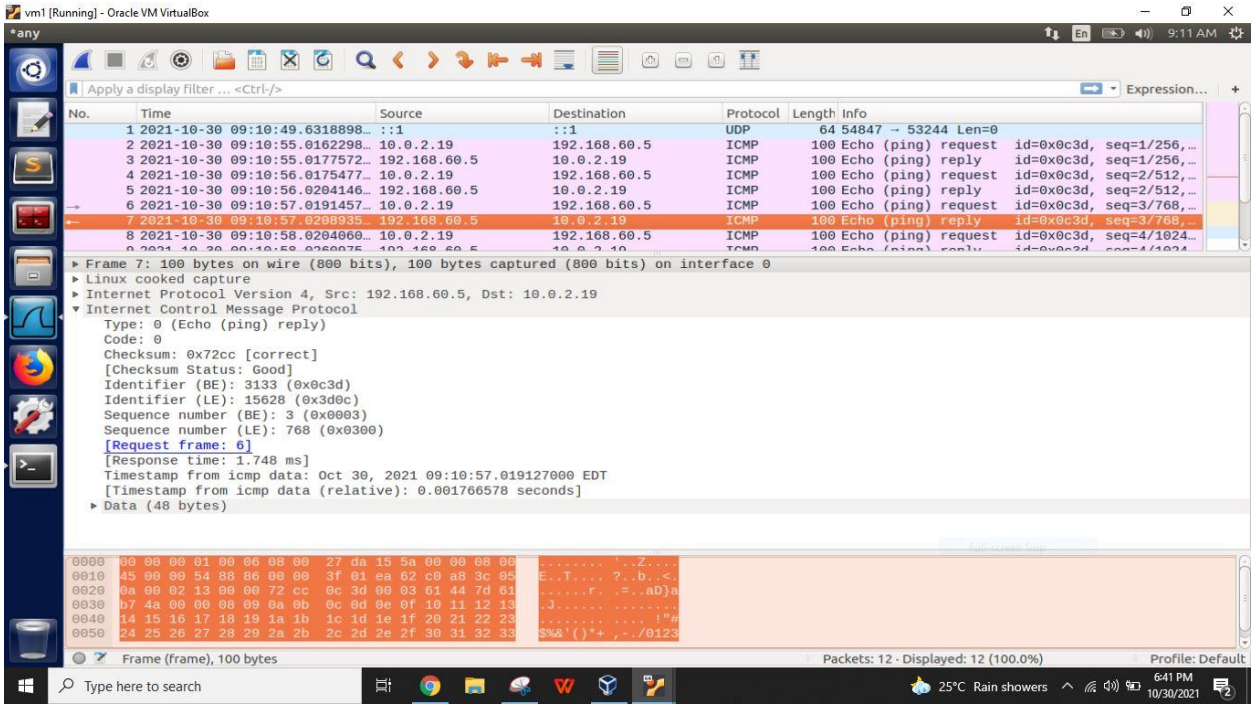
Same is done for vm3 also



After setting the patch we are going to ping again from vm1 to vm3

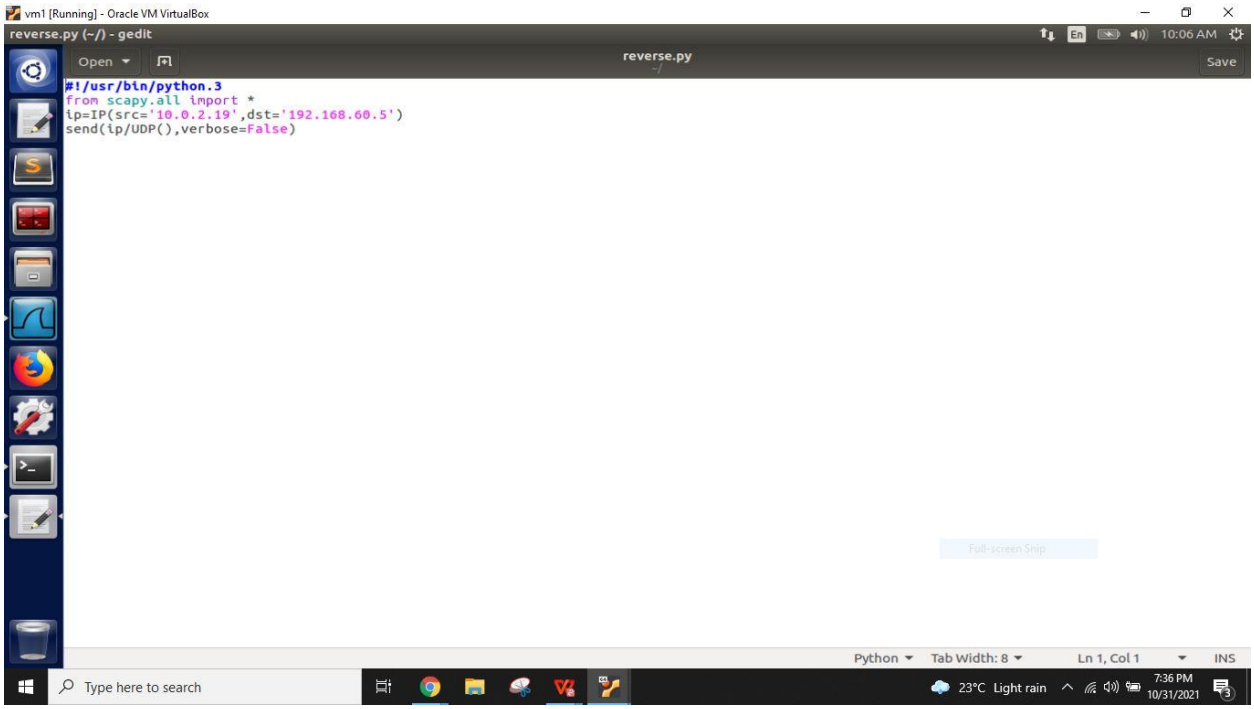


We can see both request and reply as the ip routing patch is set successfully

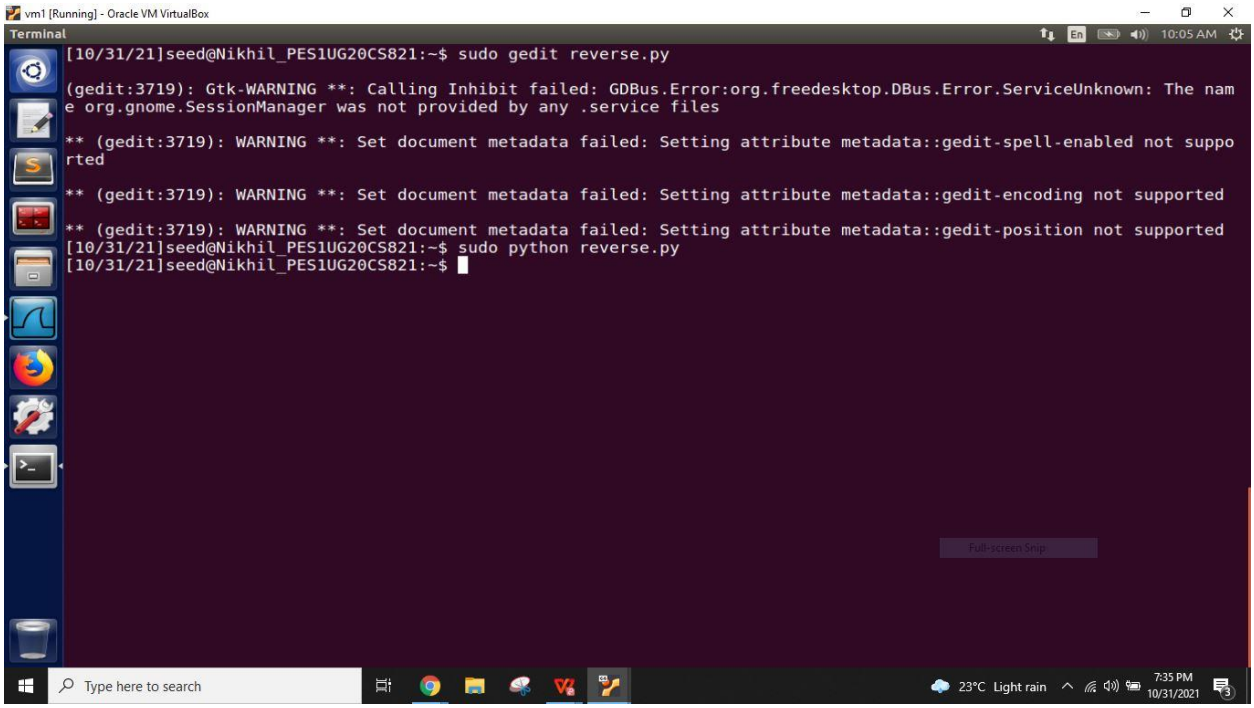


Task 3c: reverse path filtering

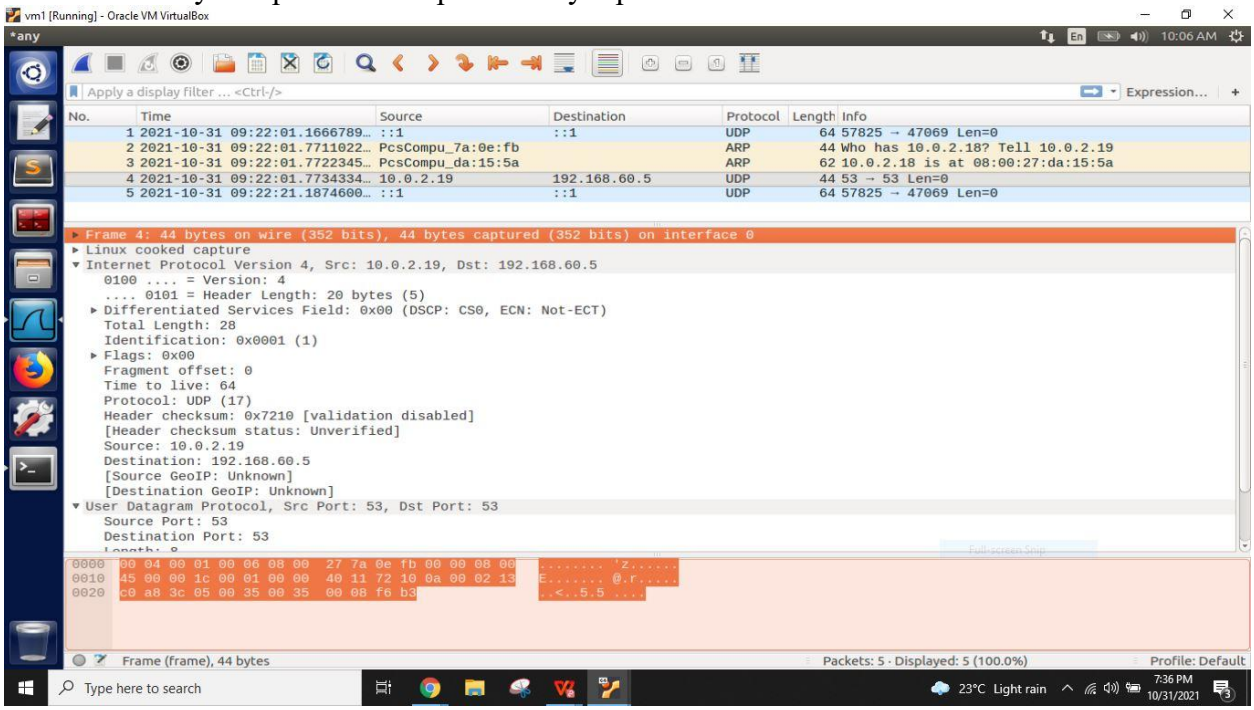
Save the below code in the vm1



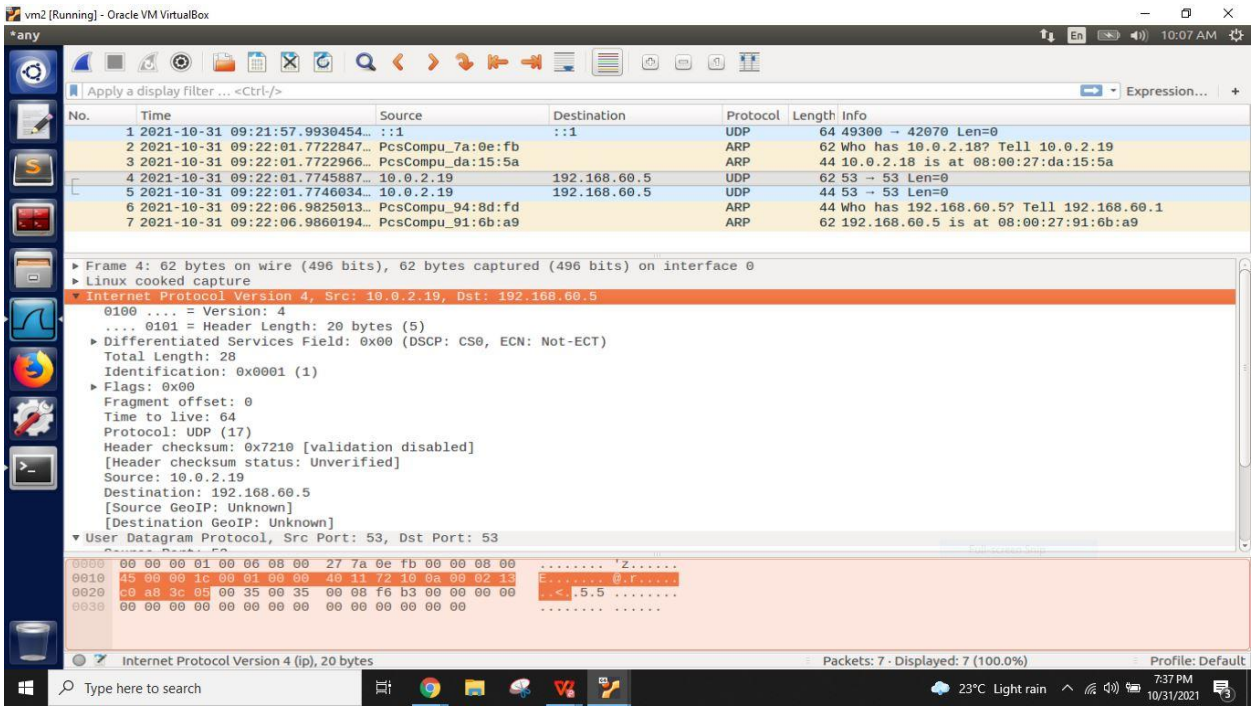
Save and run the above code in the vm1



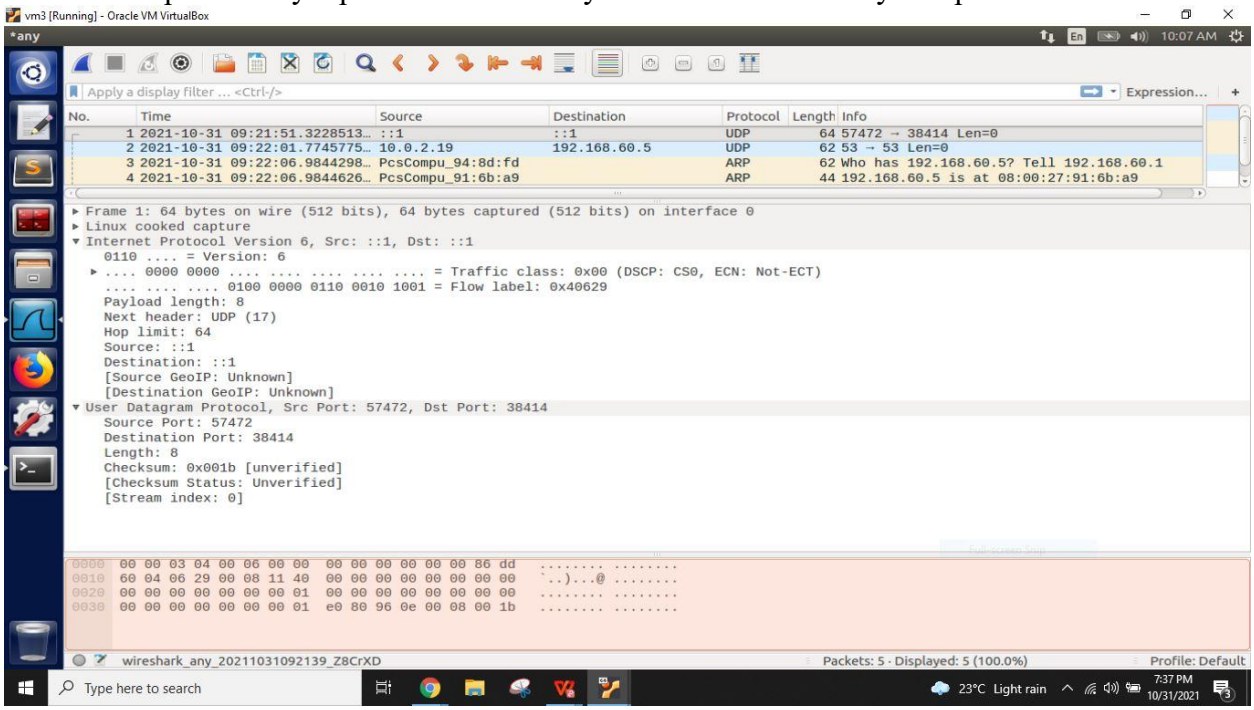
As we sent only one packet we captured only 1 packet from the vm1



Vm2 captures 2 consecutive packets but from different interface

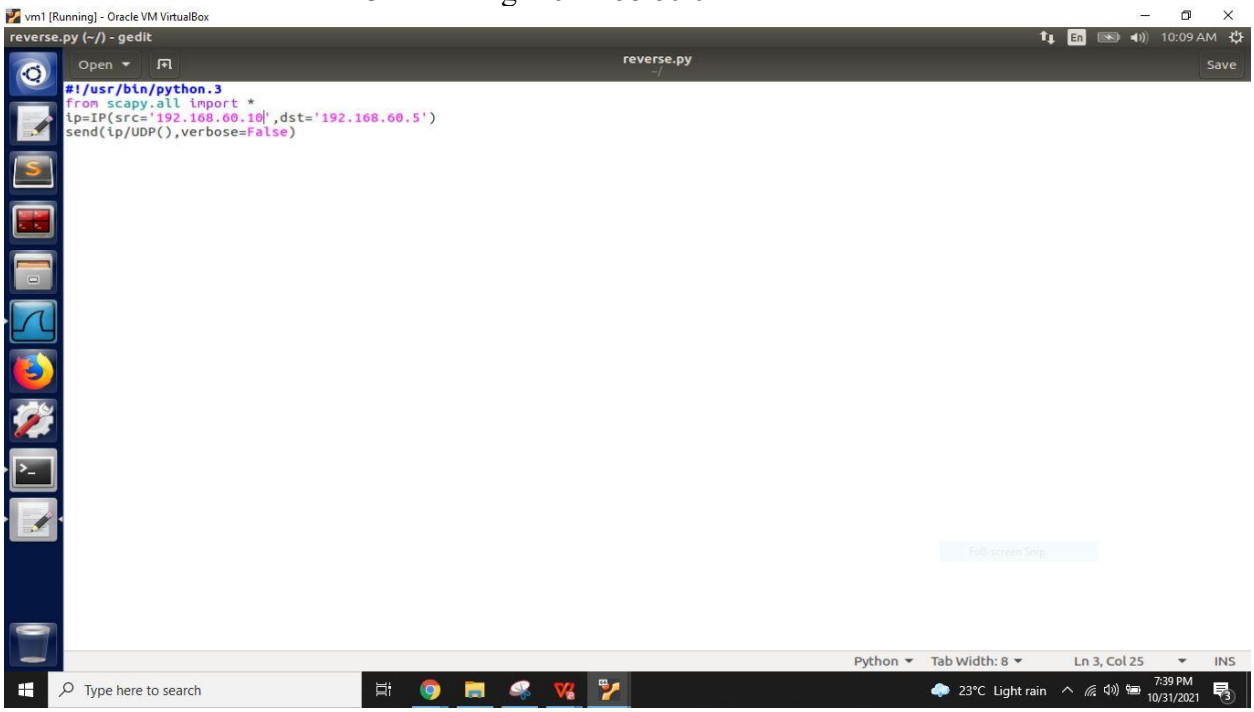


Vm3 also captures only 1 packet as it has only one interface and only one packet is sent

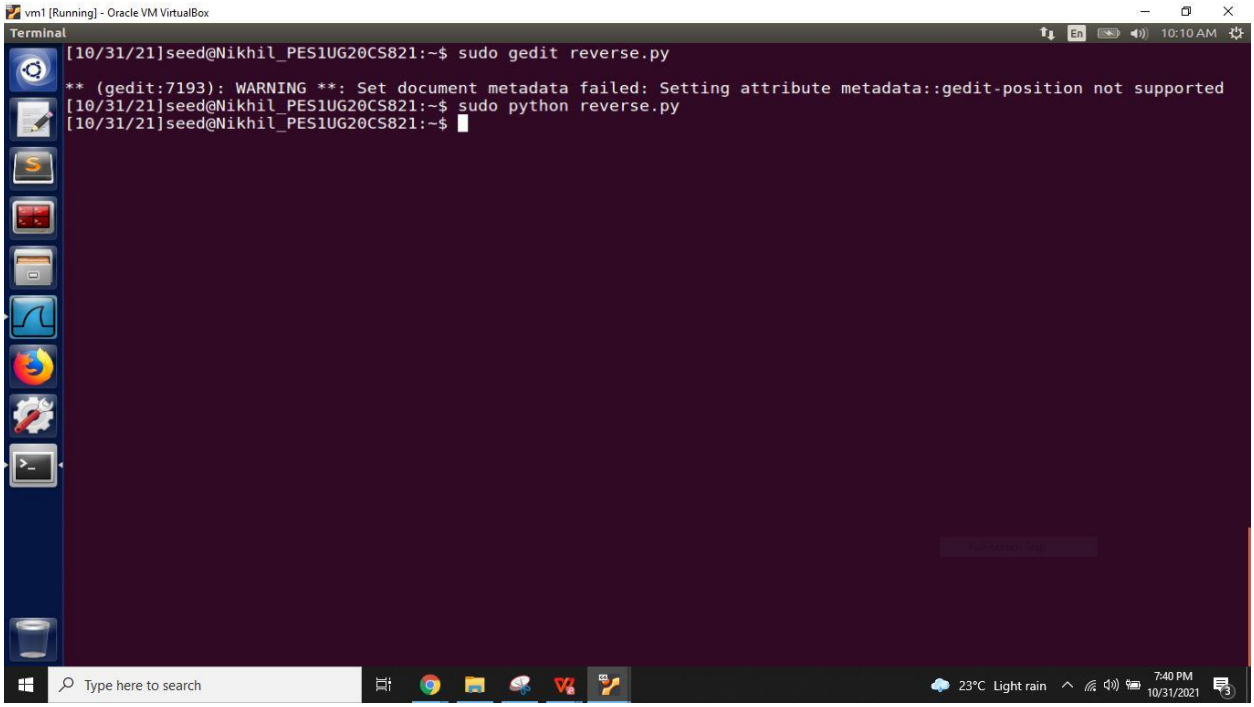


An IP address belonging to the network 10.0.2.0/24

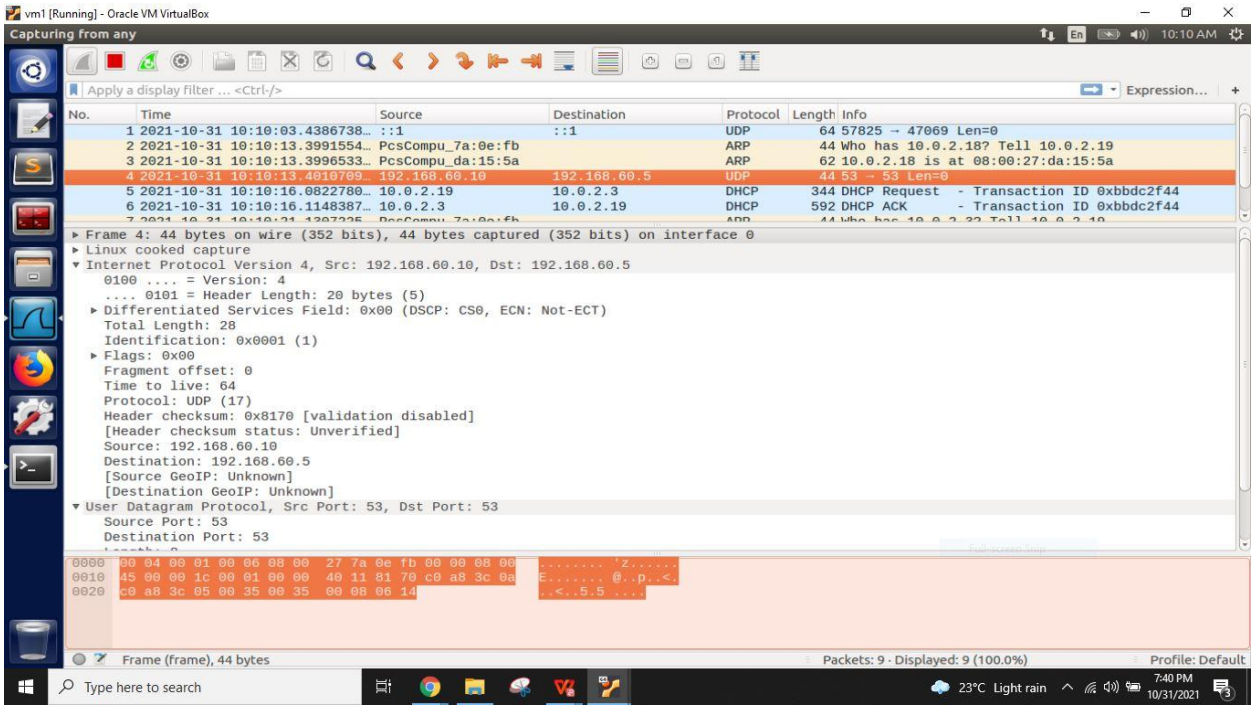
Packet sent from vm1 to vm3 from range 192.168.60.0/24



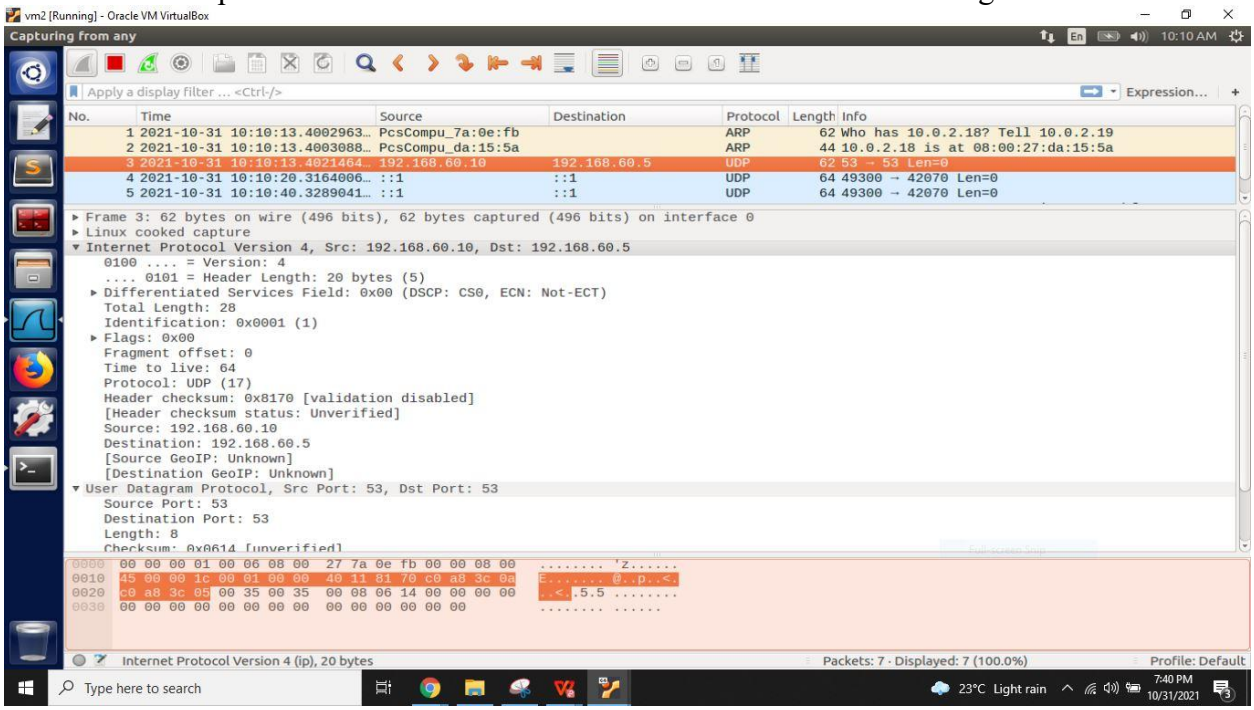
Save and run the above code in the vm1



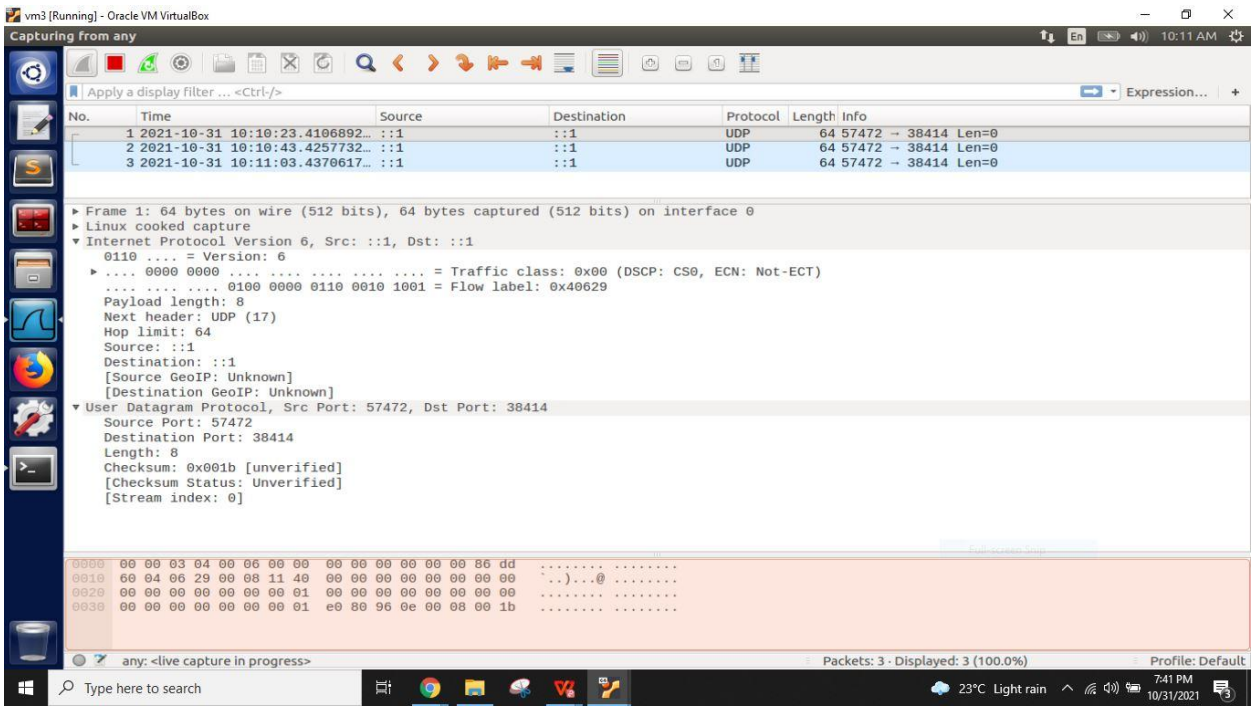
Capture those packets at all the 3 vm's
At vm1 we can see one packet



In vm2 also one packet is seen even it has 2 interface because we set the range

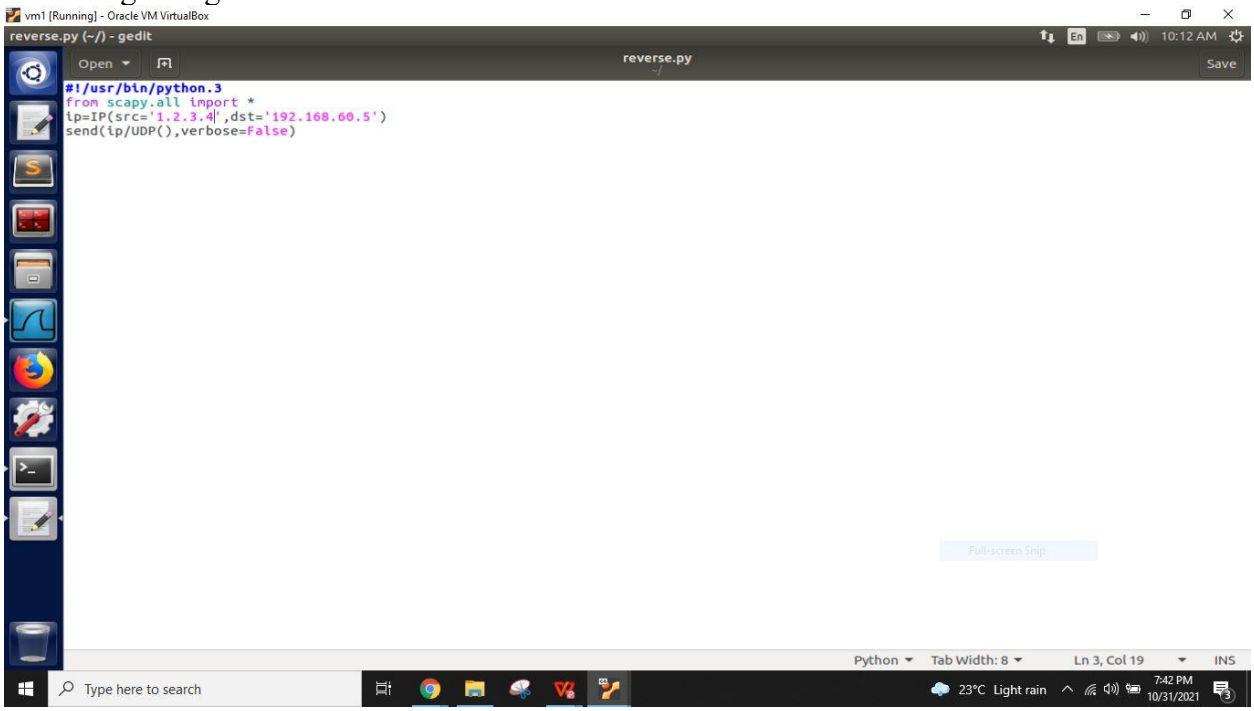


As expected we cannot see any packet in the vm3 as it has ip address out of range

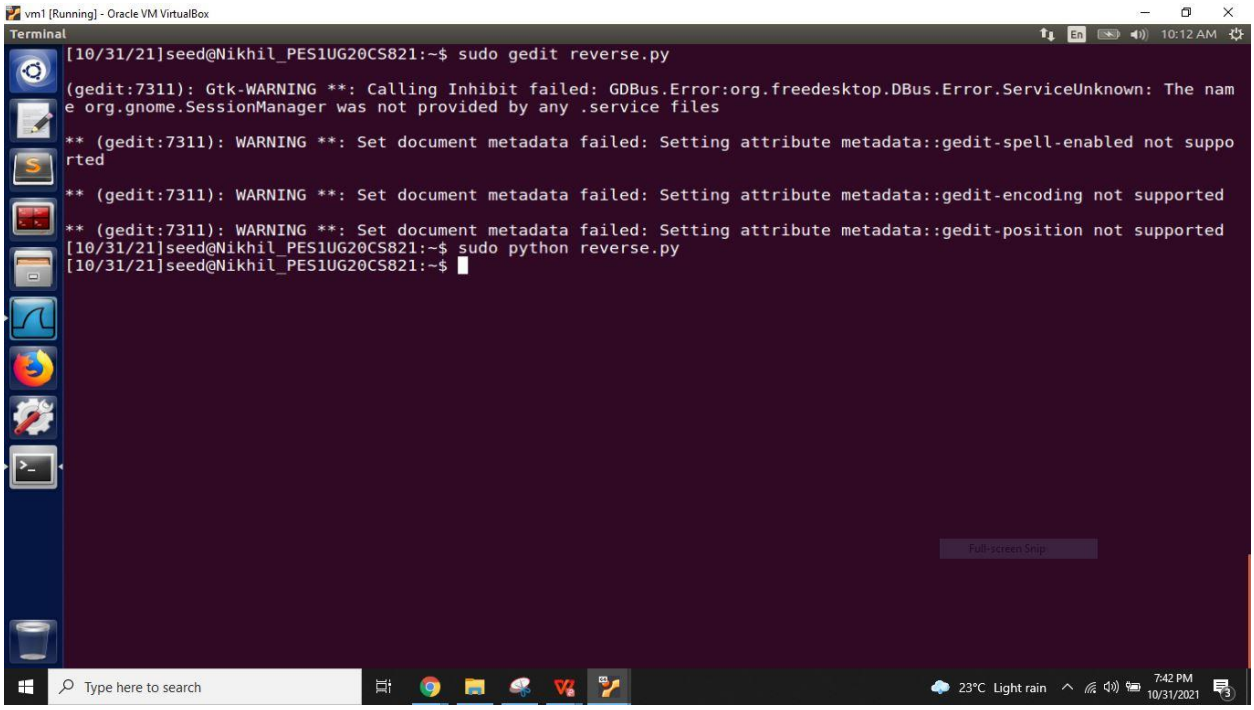


An IP address belonging to the Internet, such as 1.2.3.4.

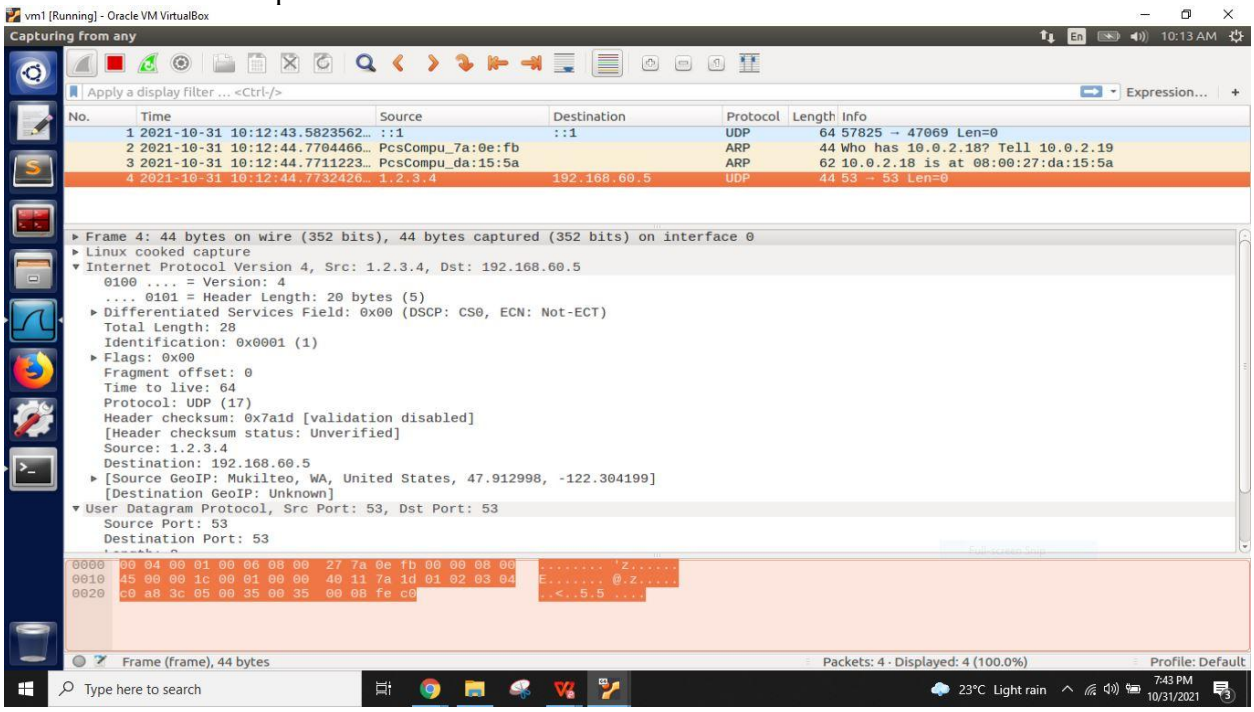
Following changes are made in the code



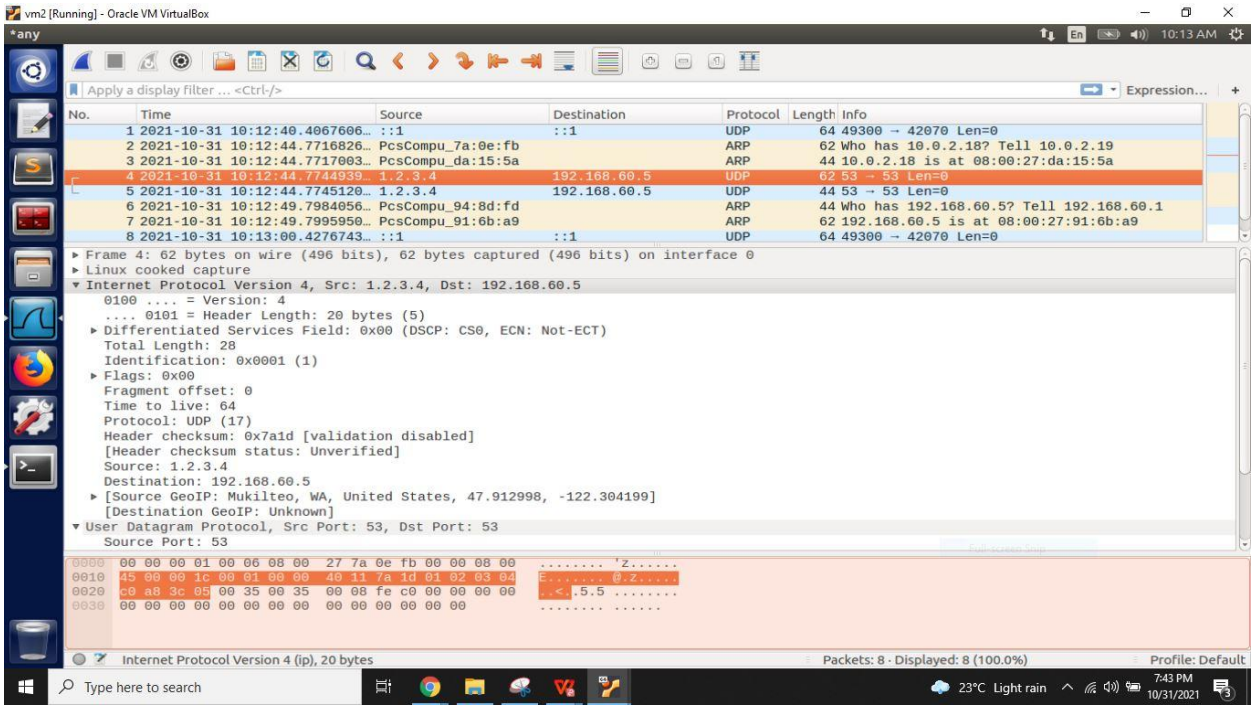
Save and run the above code in the vm1



Capture the packets in all the 3 vm's
In vm1 we can see 1 packet in the wireshark observation



In vm2 we can see 2 consecutive packets of both the interfaces



Only one packet is seen in the vm3