ICMP FLOODING ASSIGNMENT

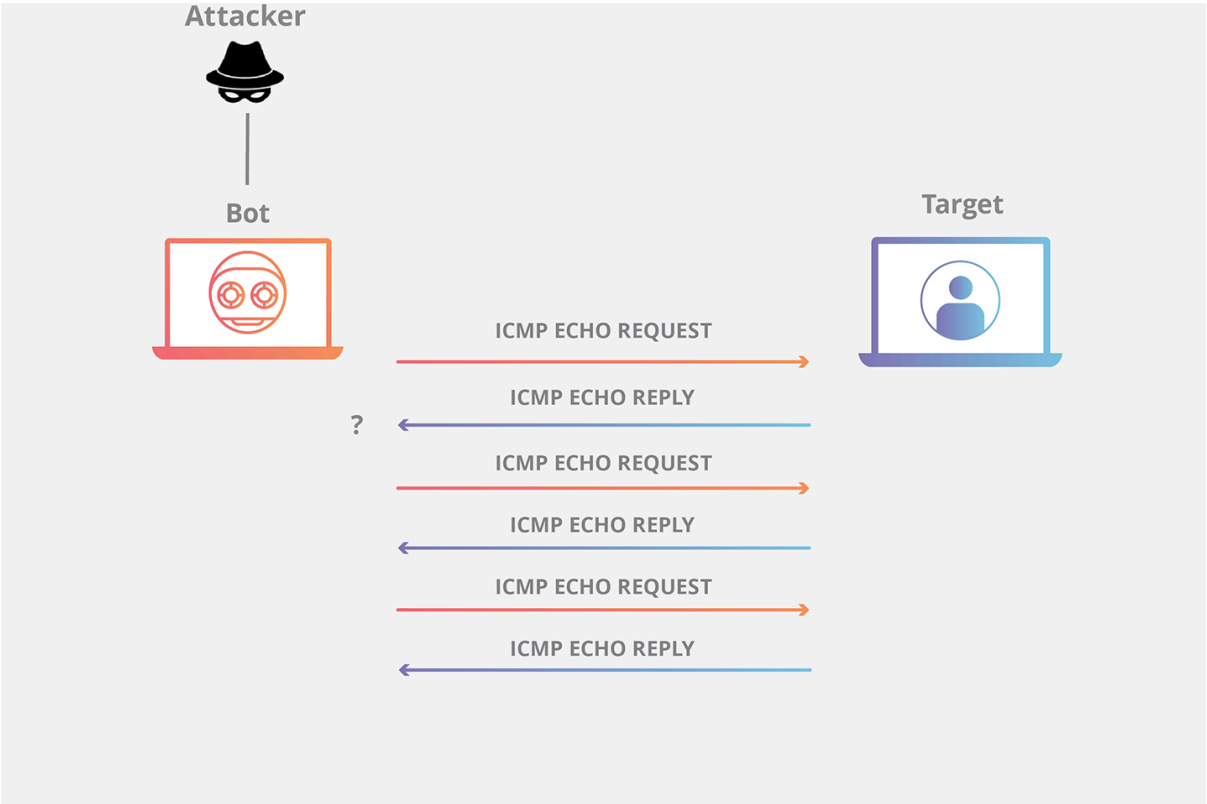
Name:- Etcharla Revanth Rao
Roll No:- 24CS06010

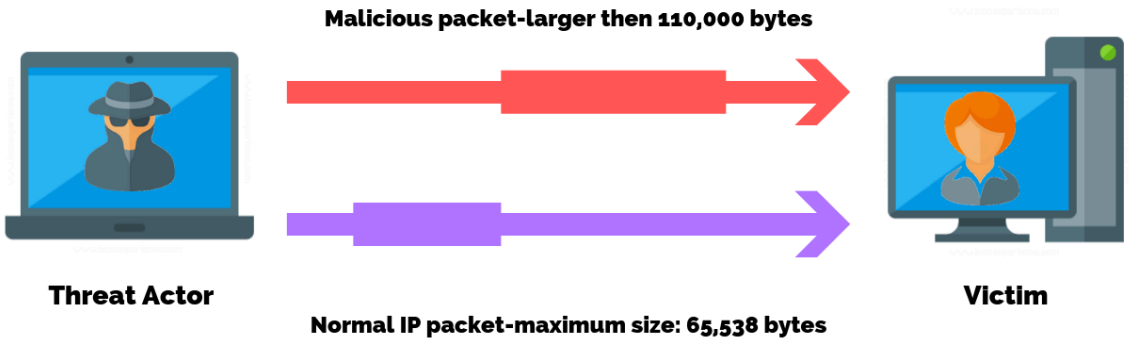
ICMP (Internet Control Message Protocol) is a vital part of the Internet Protocol (IP) suite that plays a key role in maintaining the health and efficiency of a network. Rather than transmitting user data, ICMP is used by network devices—such as routers and switches—to send diagnostic and error messages. These messages indicate whether data packets are successfully reaching their destination or if issues like unreachable hosts, timeouts, or routing problems are occurring.

For example, when you use the **ping** or **traceroute** commands, you're relying on ICMP to check connectivity and trace the path data takes across the network. In essence, ICMP acts like a behind-the-scenes communicator that helps troubleshoot and optimize network performance.

CMP Flooding Attack (Ping Flood) – Key Points

- **Type:** Denial-of-Service (DoS) attack.
- **Method:** Sends large numbers of ICMP Echo Request (ping) packets to the target.
- **Goal:** Overwhelm system resources (CPU, bandwidth, memory) and cause service disruption.
- **Flood Option:** Sends packets rapidly without waiting for replies.
- **Privileges Needed:** Often requires admin/root access to use flood mode.
- **Impact:** Slows down or crashes the target system/network.
- **Common Tool Used:** `ping -f` (on Unix/Linux systems).
- **Defense:** Rate-limiting ICMP traffic, firewalls, intrusion detection systems (IDS).





Environment Setup

Virtual Machines:

- **Attacker VM**: Ubuntu 22.04
- **Victim VM**: Ubuntu 14.04

Tools Used:

Attack Script: Python (using `scapy`, `random`, `time`)

- **Monitoring Tools**:
    - `htop` – CPU and RAM usage
    - `iftop` – Bandwidth monitoring
    - `ifstat` – Interface statistics

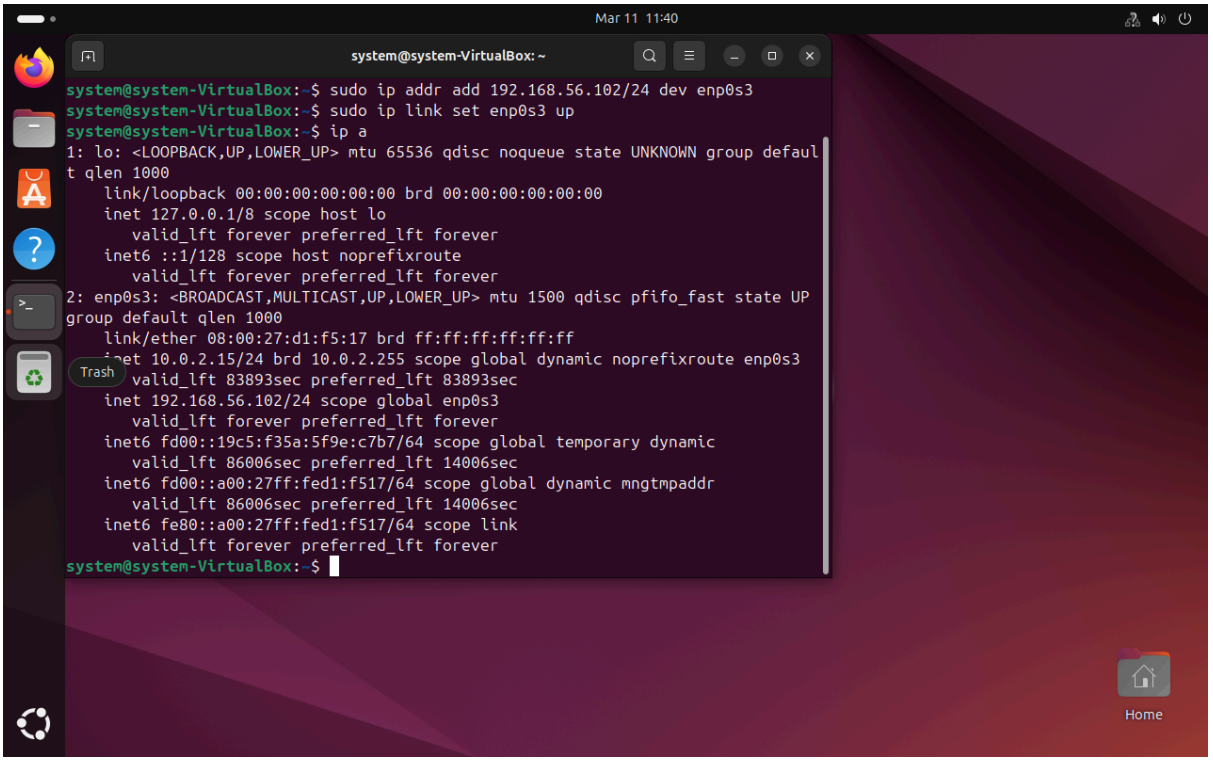
Install all the tools htop ,iftop,ifstat and wireshark .

Q1.Perform an ICMP Flood DoS Attack on a virtual machine. Measure packet reassembly and impact on CPU and RAM using htop and iftop or Windows Task manager.

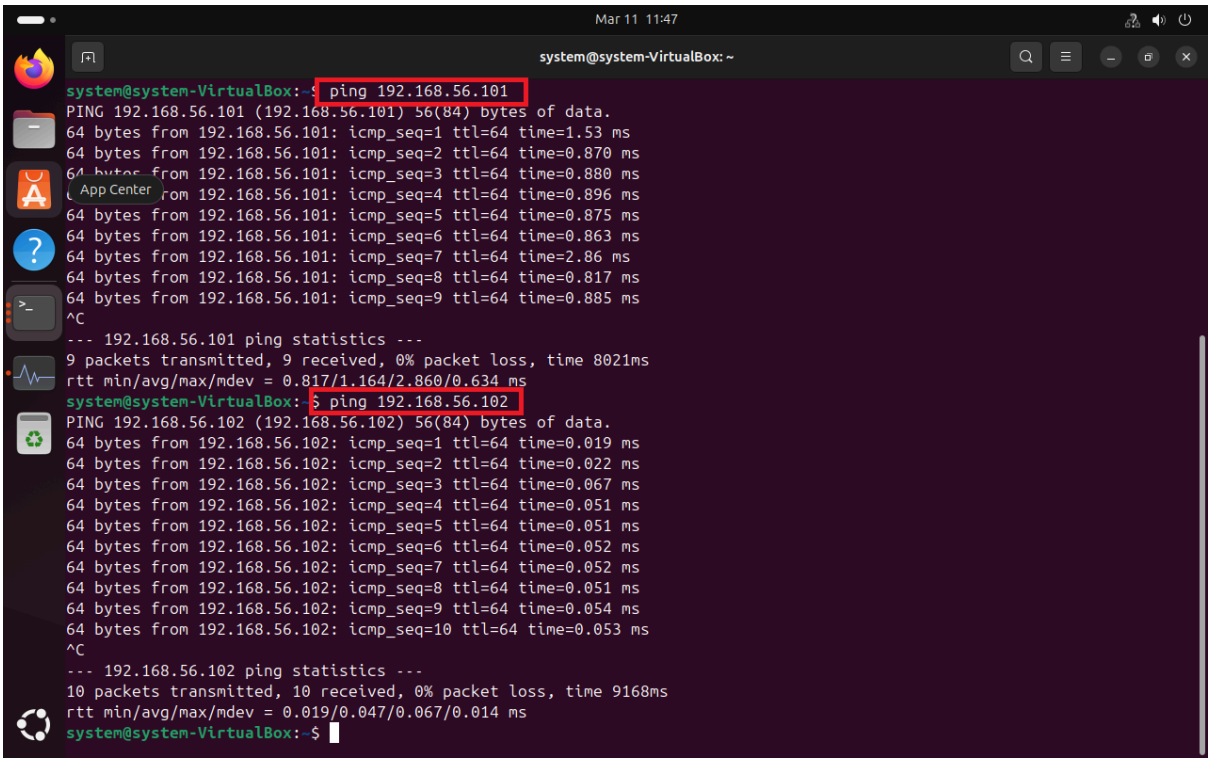
Ans:- Here i have used HOST ONLY mode in network setting for ICMP Flooding.

First step in the process is to connect two virtual machines for that will be assigning static ip to both attacker and victim vm's

sudo ifconfig <interface_name> <ip_address> netmask <netmask> up



For getting interface_name and other details we can use ***ip a.*** After using the above command you can check whether the IP changed to static or not using the same ***ip a.*** Command.

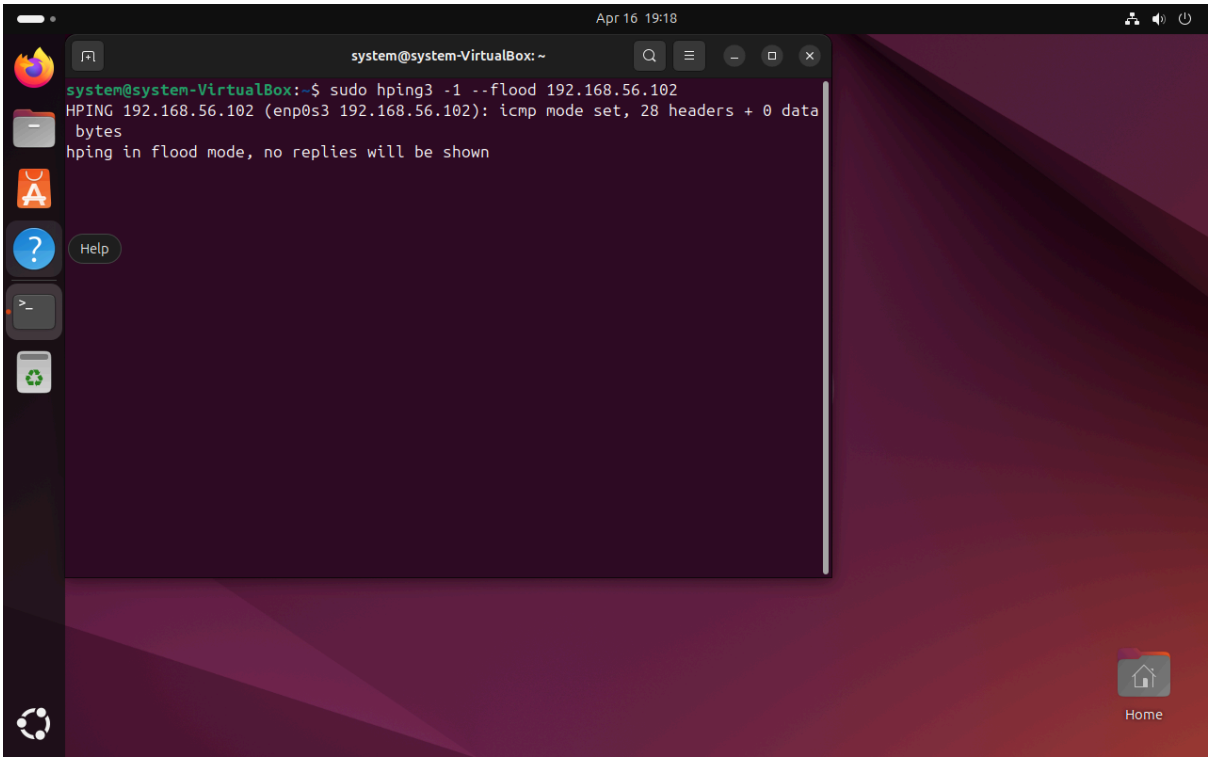


Using ***ping*** command i am able to see whether i am communicate or not ..

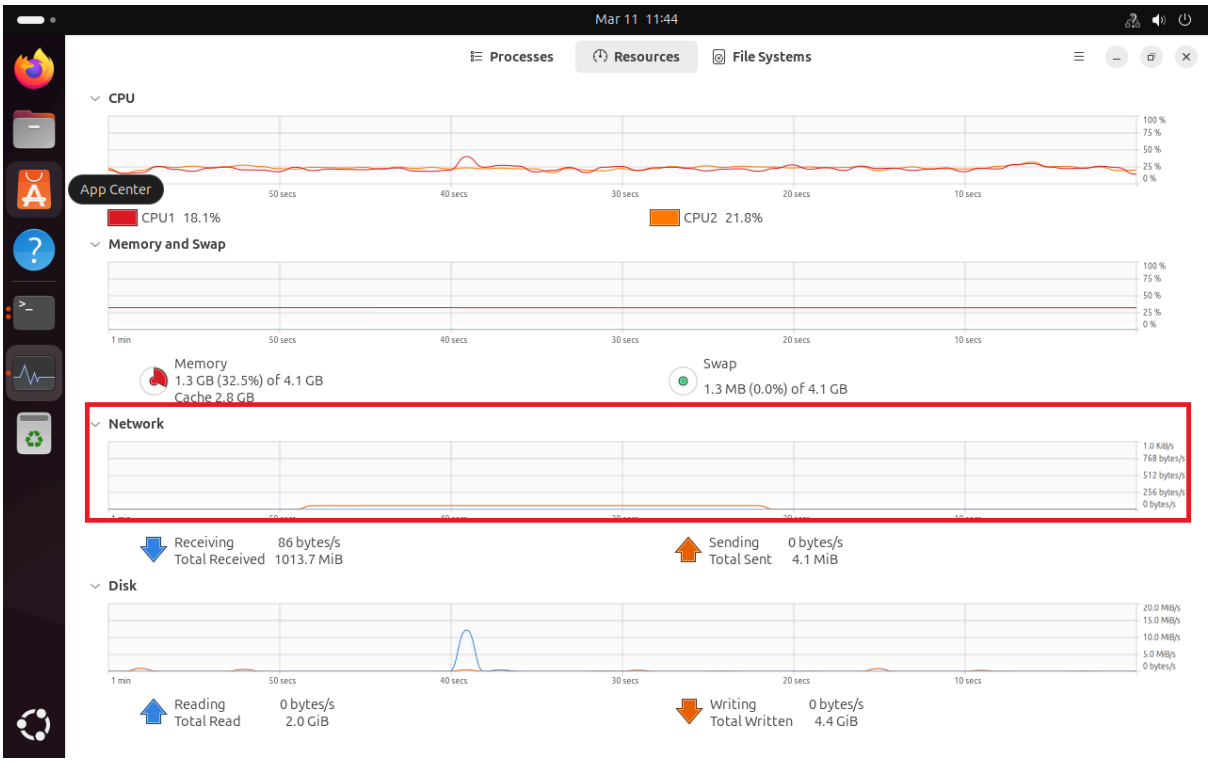
I have assigned Assigned static IP's **:** 192.168.56.101 for Attacker VM and 192.168.56.102 for Victims VM.

Now for creating flood attack we will attacker VM,

Run the command ***sudo hping3 -1 –flood 192.168.56.102 (IP address is victims)***



Initially before the attack the system monitor is shown as below :-

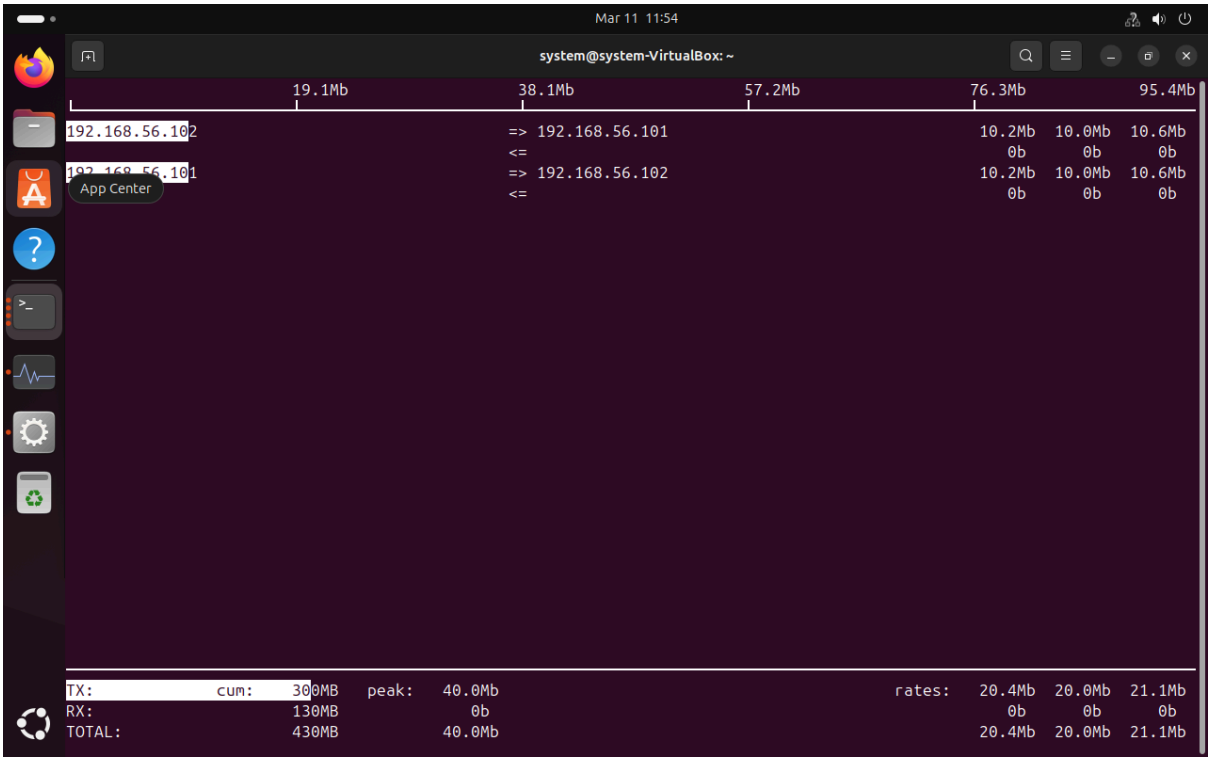


After the flood attack we see that the network usage is increased as 3.3Mb/s and CPU usage is also increased like CPU1 now increased from 18 to 100 percent.

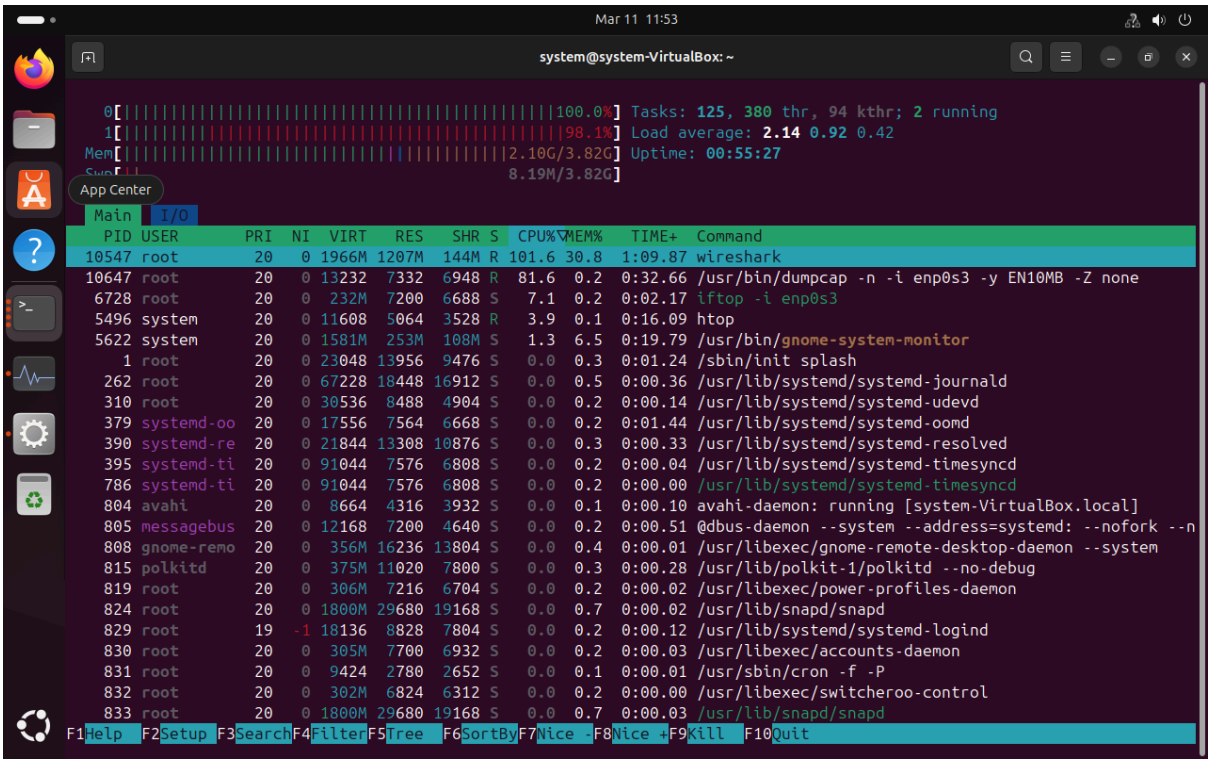
IFTOP

Using command: sudo iftop -i <network_interface>

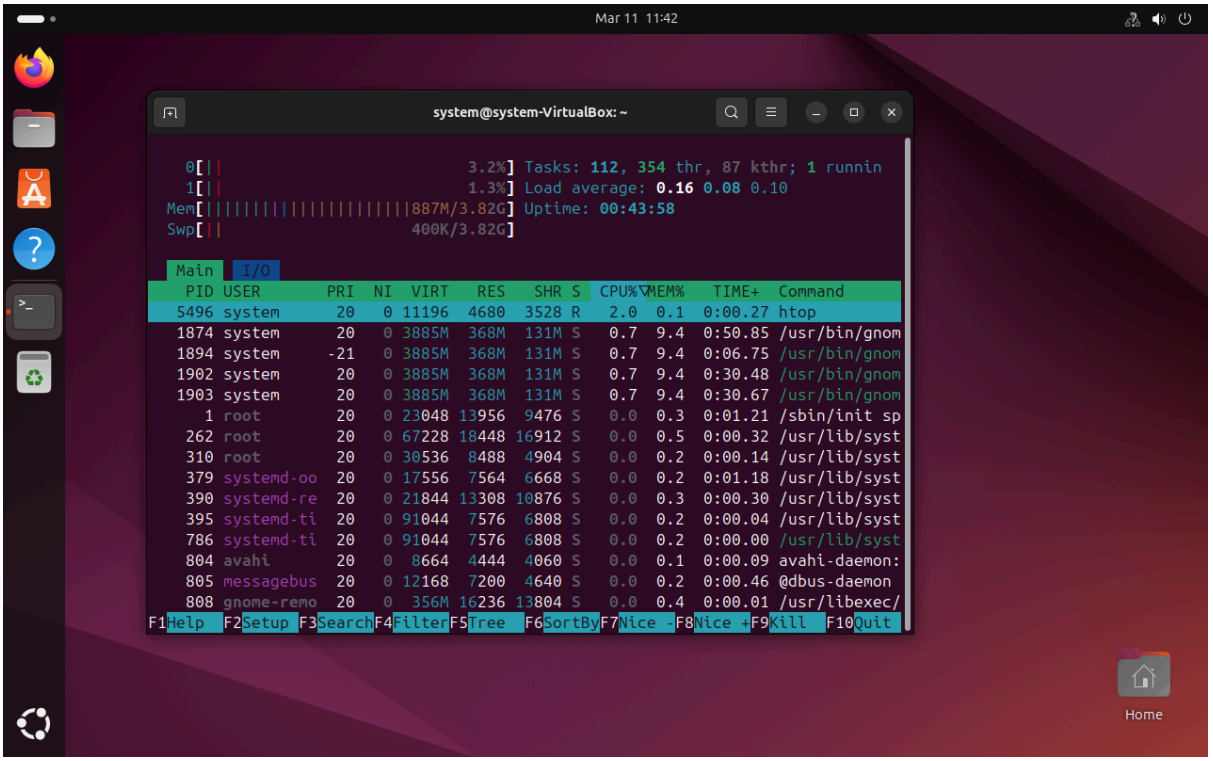
We can see the traffic is coming from attackers machine **192.168.56.101** and reply is also from victim machine **192.168.56.102**

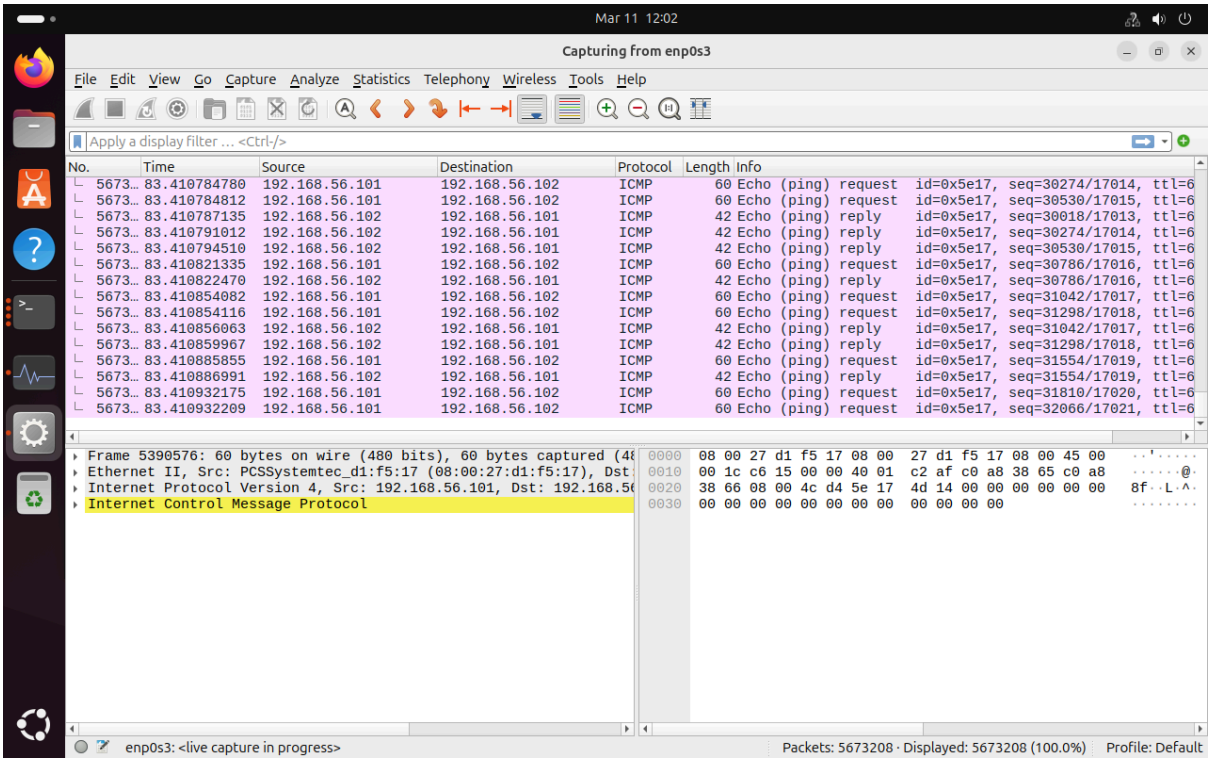


Htop before



Htop after





Q2. Write a script to perform the ICMP Attack where

a. Send packet bursts with randomized intervals using Python's time module.

b. Randomizes TTL values

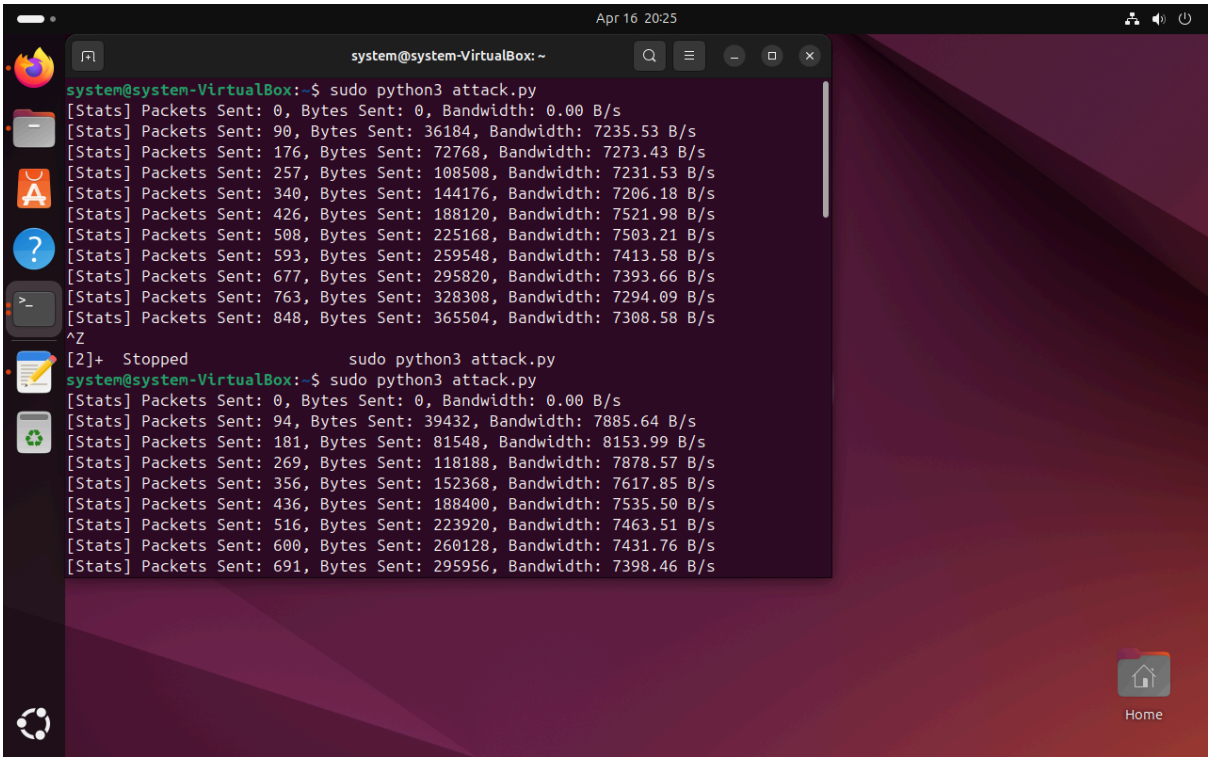
c. Alternates packet sizes

d. Spoofs IPs from a predefined IP pool

e. Calculate amplification factor and asymmetric bandwidth usage.

Ans:-

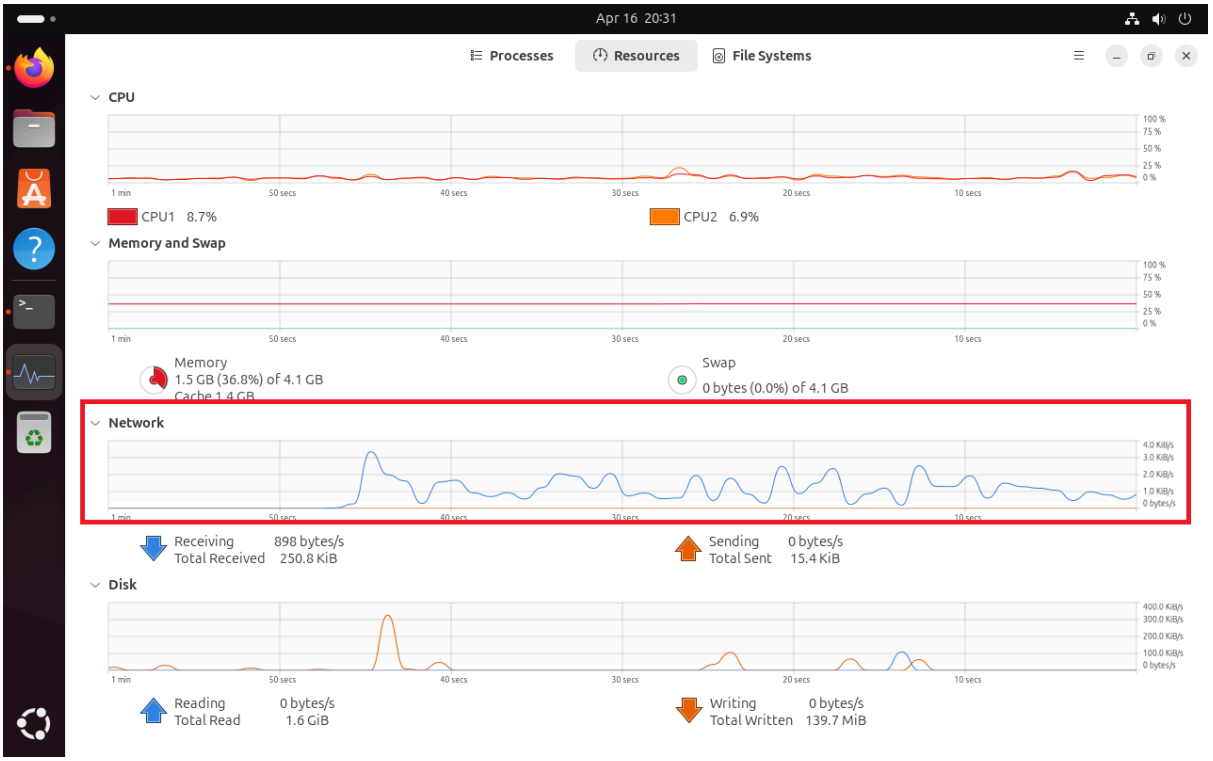
The script attack.py is run on the attack



CPU usage After :

Increase in CPU and RAM usage observed on the victim during the flood bursts.

Significant bandwidth was consumed, especially asymmetric, affecting victim's network availability.



Since we are IP spoofing with different IP address pool we can see that only packets are receiving 898 bytes/sec and not able to sent ~0kb/s.

IFTOP

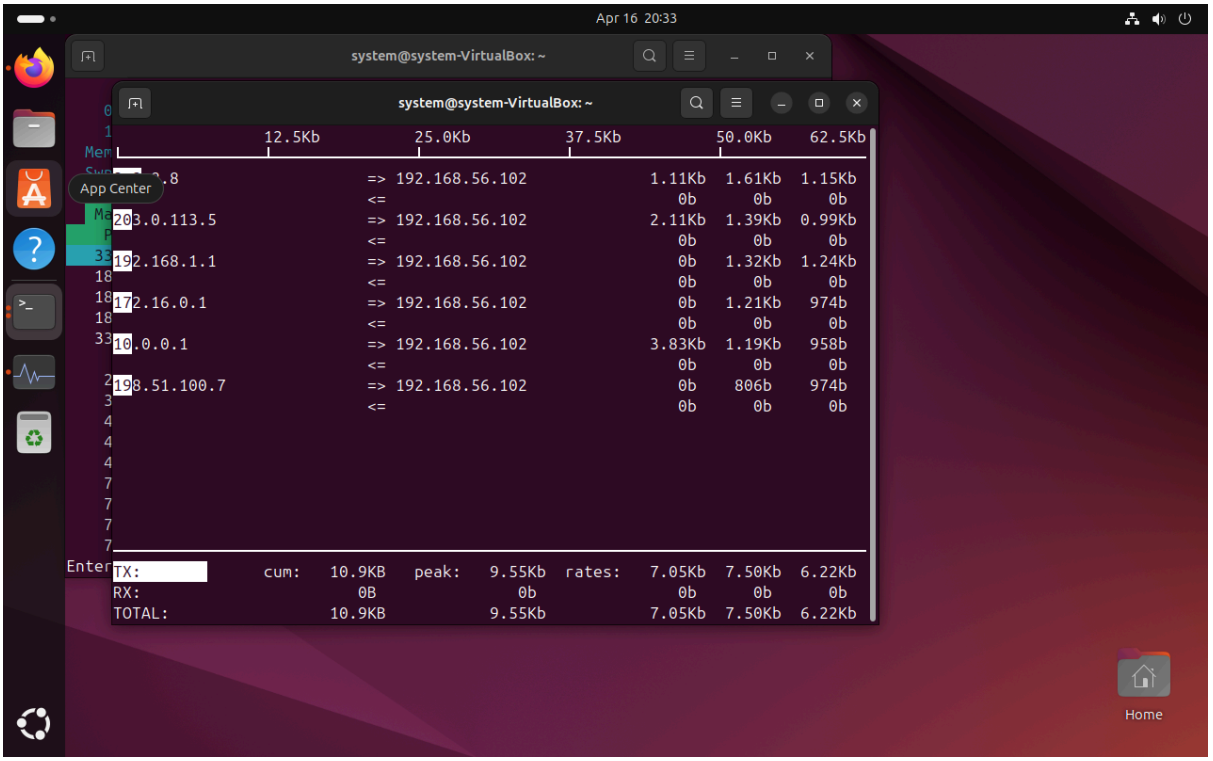
IP_POOL = [

"10.0.0.1", "172.16.0.1", "192.168.1.1",

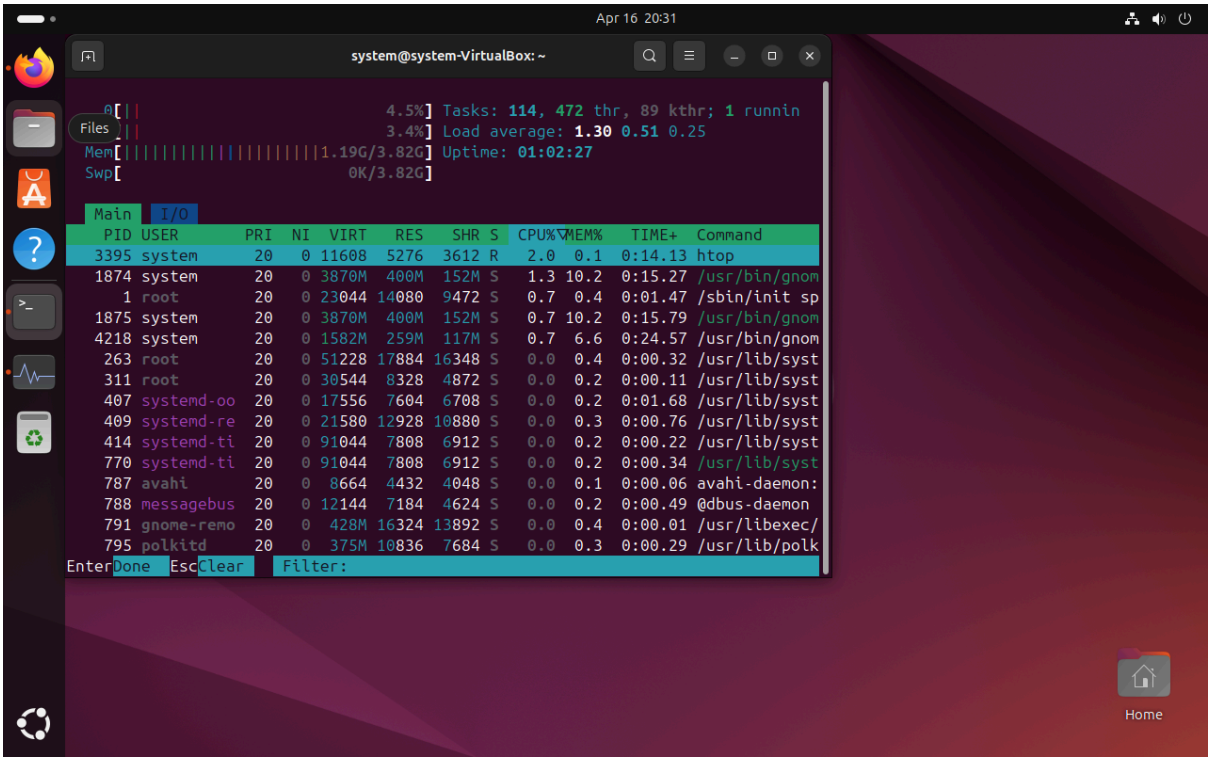
"203.0.113.5", "198.51.100.7", "8.8.8.8"

]

These IPs are used in the code. From the ***iftop*** output, traffic is seen coming from them, but no data is going back. This indicates they are spoofed IPs, used to simulate incoming traffic without real two-way communication.



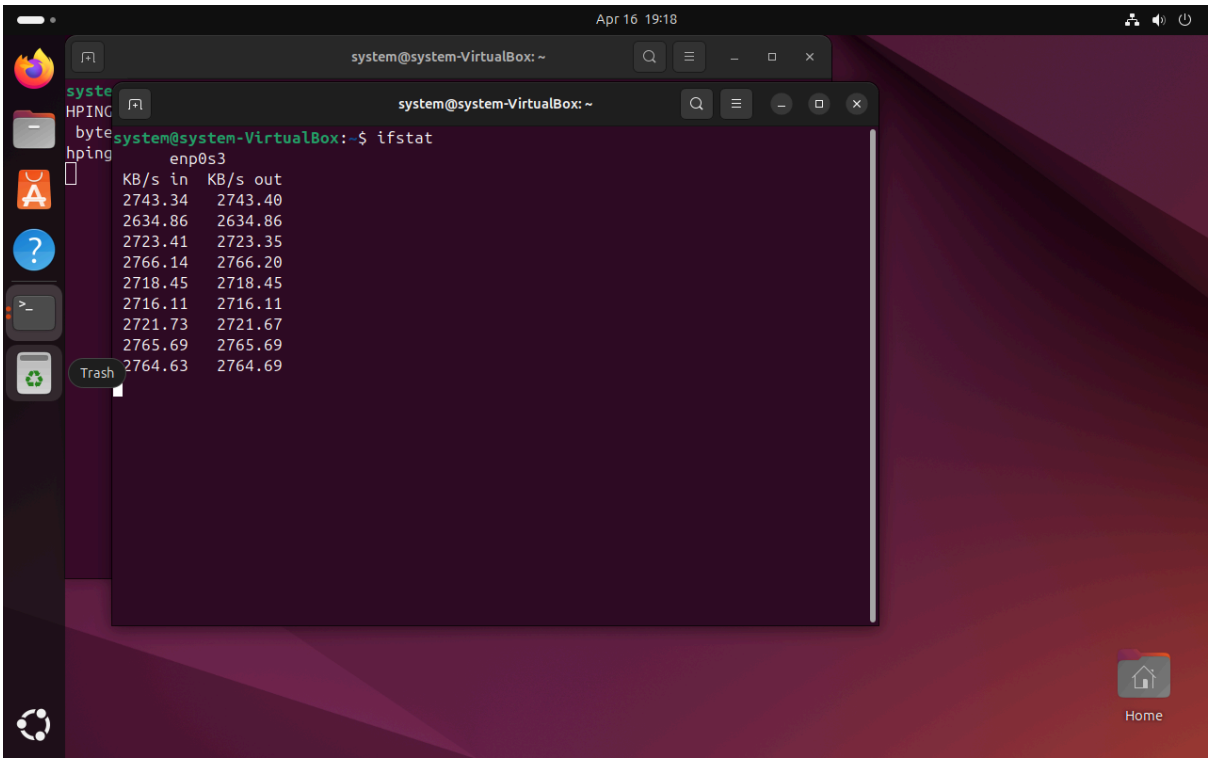
Htop

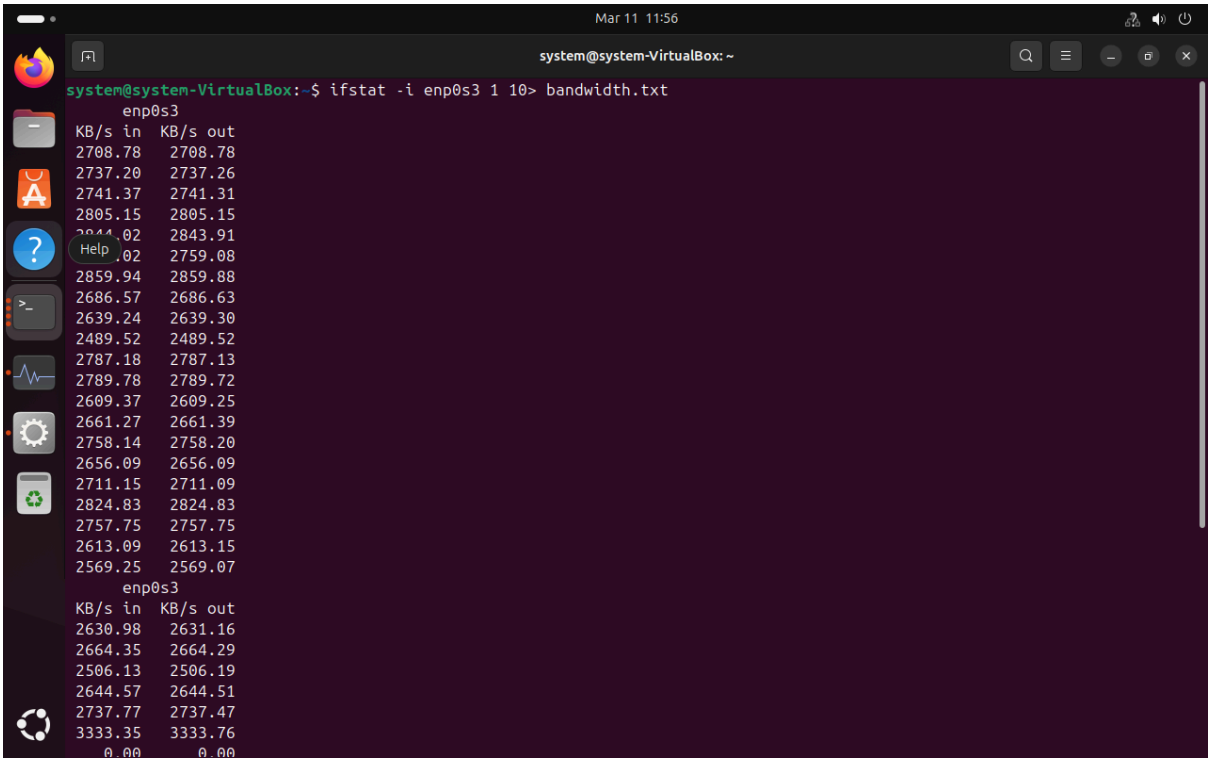


IFSTAT

Attackers VM

Command : sudo ifstat -i <network_interface>

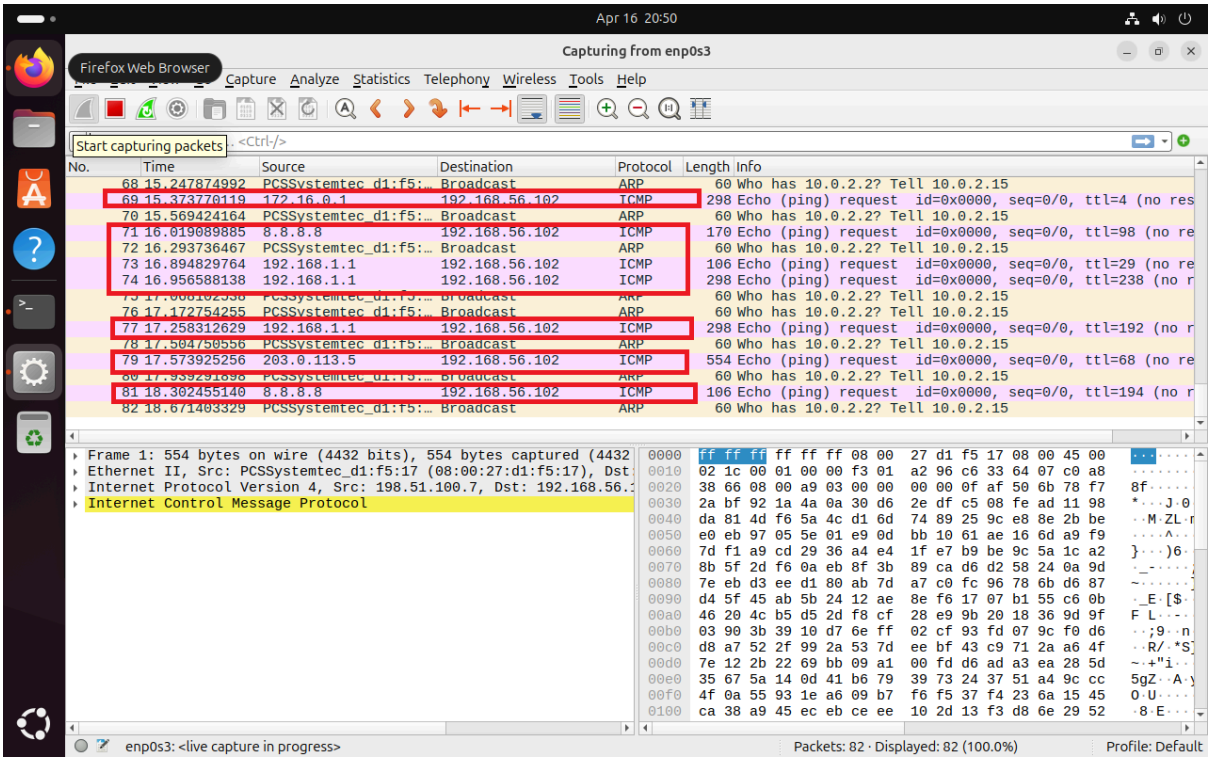




WireShark:

By analyzing the packets in Wireshark, we can observe several suspicious patterns:

- **Bursts of ICMP packets** sent at irregular, randomized intervals, suggesting an attempt to mimic normal traffic behavior.
- **Variable TTL (Time-To-Live) values**, likely used to bypass basic packet inspection or filtering mechanisms.
- **Alternating packet sizes**, which help in evading detection through packet size fingerprinting techniques.
- **Spoofed IP addresses** drawn from a fixed pool, making it harder to trace the origin and effectively anonymizing the attacker.



During the burst phase of the ICMP flood attack:
The attacker exhibited a **transmission-dominant profile**, generating outbound traffic at approximately **8.2 KB/s** while receiving negligible incoming data.

In contrast, the victim's virtual machine maintained a **reception-heavy behavior**, continuously ingesting traffic at around **7.06 KB/s** with no meaningful outbound communication.

This illustrates a **pronounced asymmetric bandwidth pattern**, a hallmark of **unidirectional flooding attacks**, where the attacker inundates the target with traffic, requiring no response to sustain the attack.

This scenario demonstrates a highly asymmetric bandwidth pattern, characteristic of one-way flooding attacks, where the attacker does not anticipate receiving any replies.

Attacker Transmission (TX) rate: 8.2 KB/s
Victim Reception (RX) rate: 7.06 KB/s

Amplification Factor is calculated as:
Attacker TX rate: 8.2 KB/s
Victim RX rate: 7.06 KB/s
Amplification Factor =
(Victim RX (Incoming)) ÷ (Attacker TX (Outgoing))
Amplification Factor = 7.06 / 8.2 ≈ 0.861