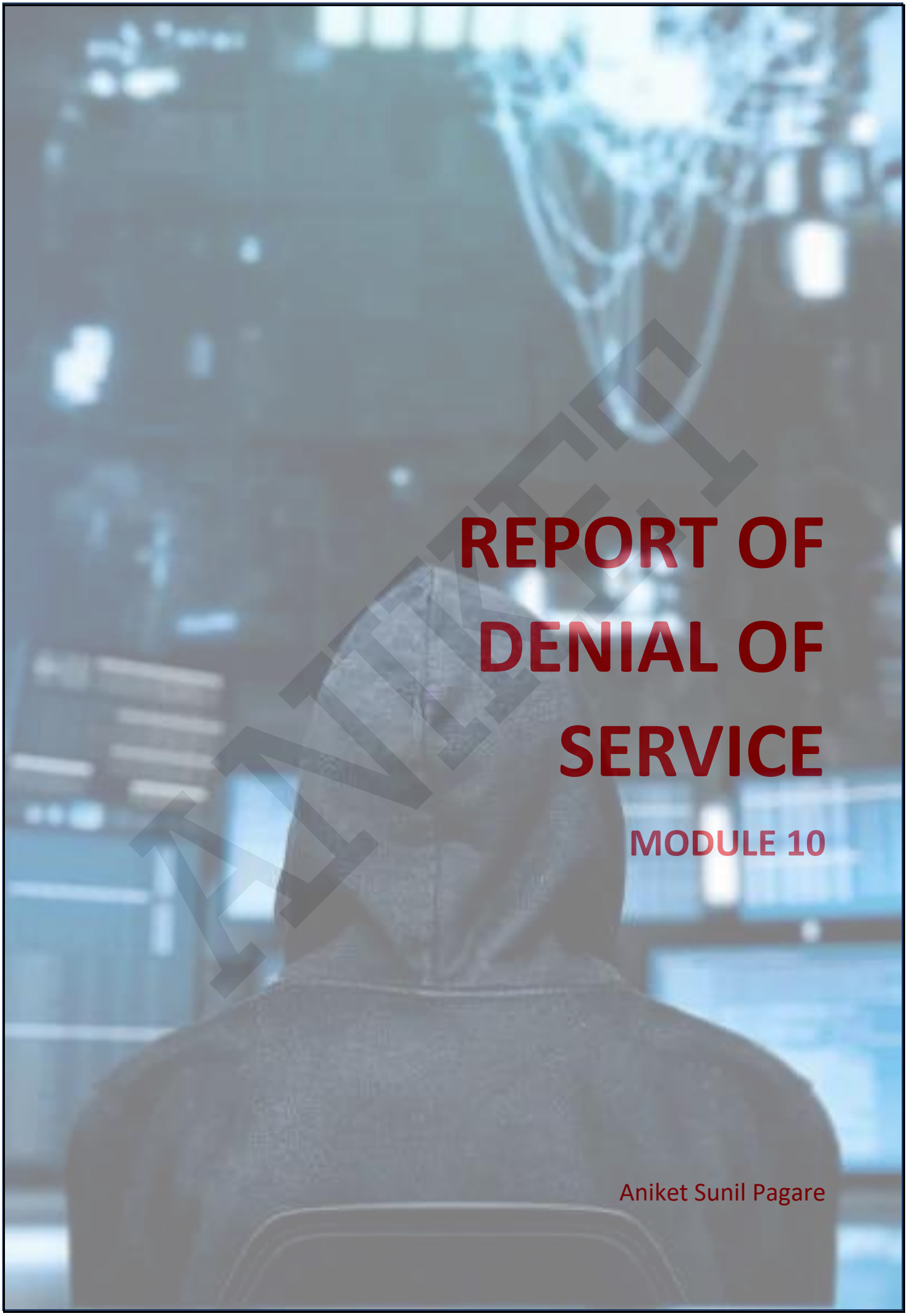


REPORT OF DENIAL OF SERVICE

MODULE 10

Aniket Sunil Pagare

Table of Contents: DoS & DDoS Attacks

- **4.3 Using HoneyBOT**
- **4.4 Using Wireshark**

---

5. Preventing DoS/DDoS Attacks

- **5.1 Network-Level Protection**
- **5.2 Use of Anti-DDoS Services**
- **5.3 Application-Level Defenses**
- **5.4 Monitoring and Detection Strategies**
- **5.5 Infrastructure Design Best Practices**
- **5.6 Using Reverse Proxies**
- **5.7 Having an Incident Response Plan**
- **5.8 Tools for DoS/DDoS Testing and Simulation**

DOS / DDOS Attack

1. DoS (Denial of Service) Attack:

A **DoS attack** is an attempt to make a machine or network resource **unavailable** to its intended users by **overloading it with traffic or crashing it** using malicious commands.

- **Type**: **Single system** attacker.
- **Goal**: Crash or overload the target.
- **Example**: Sending too many requests to a web server from a single computer, making it unable to respond to real users.

2. DDoS (Distributed Denial of Service) Attack:

A **DDoS attack** is a more advanced version of DoS, where the attack is launched from **multiple compromised systems** (called a **botnet**) simultaneously.

- **Type**: **Multiple systems** (usually infected computers).
- **Goal**: Same as DoS, but **harder to stop** due to distributed nature.
- **Example**: Thousands of infected computers flooding a server with traffic, causing downtime.

Objectives of DoS and DDoS Attacks

1. To disrupt the availability of services for legitimate users.
2. To overload system resources like CPU, memory, or bandwidth.
3. To crash or freeze the target system using malicious input or traffic.
4. To exhaust network capacity and slow down performance.
5. To distract security teams while launching more serious attacks.
6. To extort money by threatening continuous service outages.
7. To protest against organizations or governments for ideological reasons.

8. To test or probe the strength of an organization's cyber defenses.
9. To sabotage competitors by taking their services offline.
10. To take revenge on a target for personal or political motives.

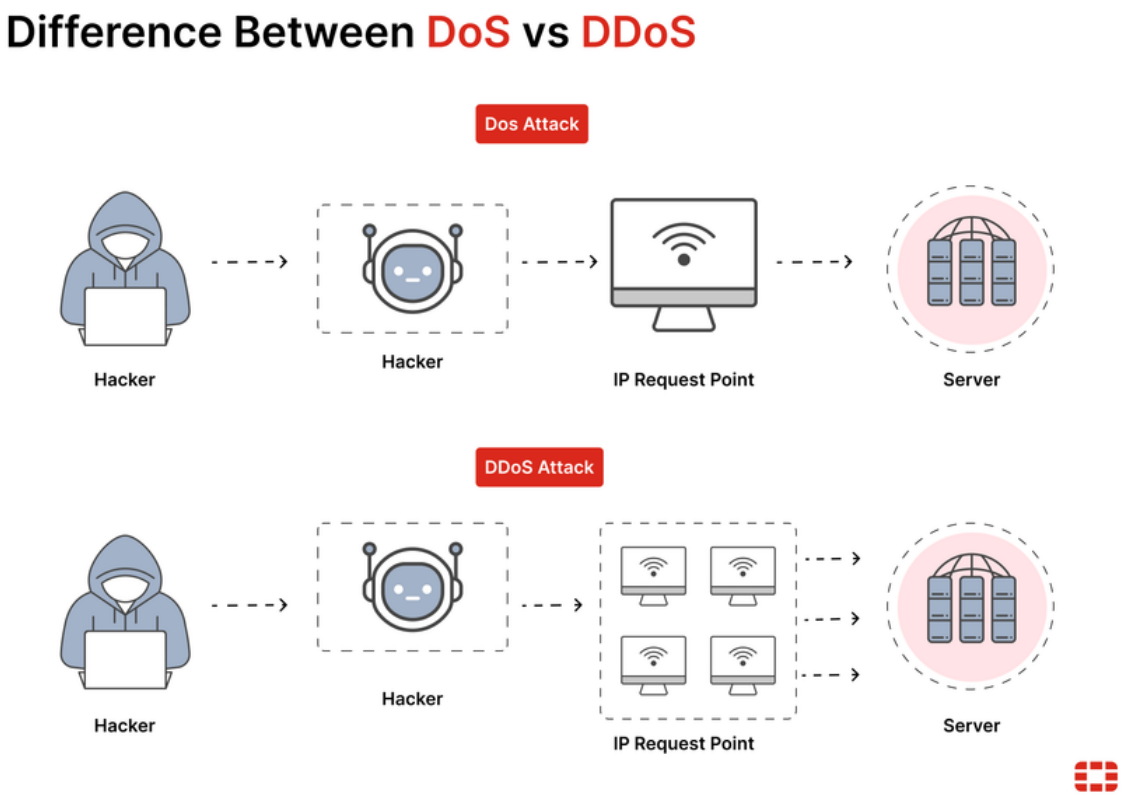


Figure of DOS/DDOS Attack 👆

Categories of DoS/DDoS Attacks

1. Volume-Based Attacks (Volumetric Attacks)

Goal: Consume the **entire bandwidth** of the target network/system.

- **Examples:**
  - **UDP Flood** – Sends massive UDP packets to random ports.
  - **ICMP Flood (Ping Flood)** – Overloads the system with ICMP Echo Requests (pings).

- **DNS Amplification** – Spoofs requests to DNS servers, causing them to flood the victim.

🔧 **Metrics:**

- Measured in **Gbps** or **pps** (packets per second).

---

2. Protocol Attacks (Network Layer Attacks)

Goal: Exploit weaknesses in **network protocols** to exhaust server or firewall resources.

🔹 **Examples:**

- **SYN Flood** – Sends a high volume of SYN packets to consume server memory/state.
- **Ping of Death** – Sends malformed or oversized packets that crash the system.
- **Smurf Attack** – Spoofs ICMP requests to a broadcast IP, amplifying the flood.
- **ACK Flood** – Overloads firewalls or stateful devices by sending ACK packets.

🔧 **Metrics:**

- Measured in **packets per second (pps)**.

---

3. Application Layer Attacks (Layer 7 Attacks)

Goal: Crash or slow down web applications by exhausting server-side resources.

🔹 **Examples:**

- **HTTP GET/POST Flood** – Sends a flood of HTTP requests to overload the web server.
- **Slowloris** – Sends partial HTTP requests slowly to keep connections open.
- **RUDY (R U Dead Yet?)** – Slowly sends POST requests to tie up server resources.

- **DNS Query Flood** – Floods DNS servers with seemingly legitimate queries.

🔧 **Metrics:**

- Measured in **requests per second (rps)**.

---

4. Resource Exhaustion Attacks

Goal: Drain **CPU, memory, or system limits** rather than just bandwidth.

🔹 **Examples:**

- **Fork Bomb** – Creates processes recursively to exhaust CPU.
- **Memory Leak Exploits** – Force programs to consume excessive RAM.

---

5. Multi-Vector Attacks

Goal: Combine multiple attack types **simultaneously** to bypass defenses.

🔹 **Examples:**

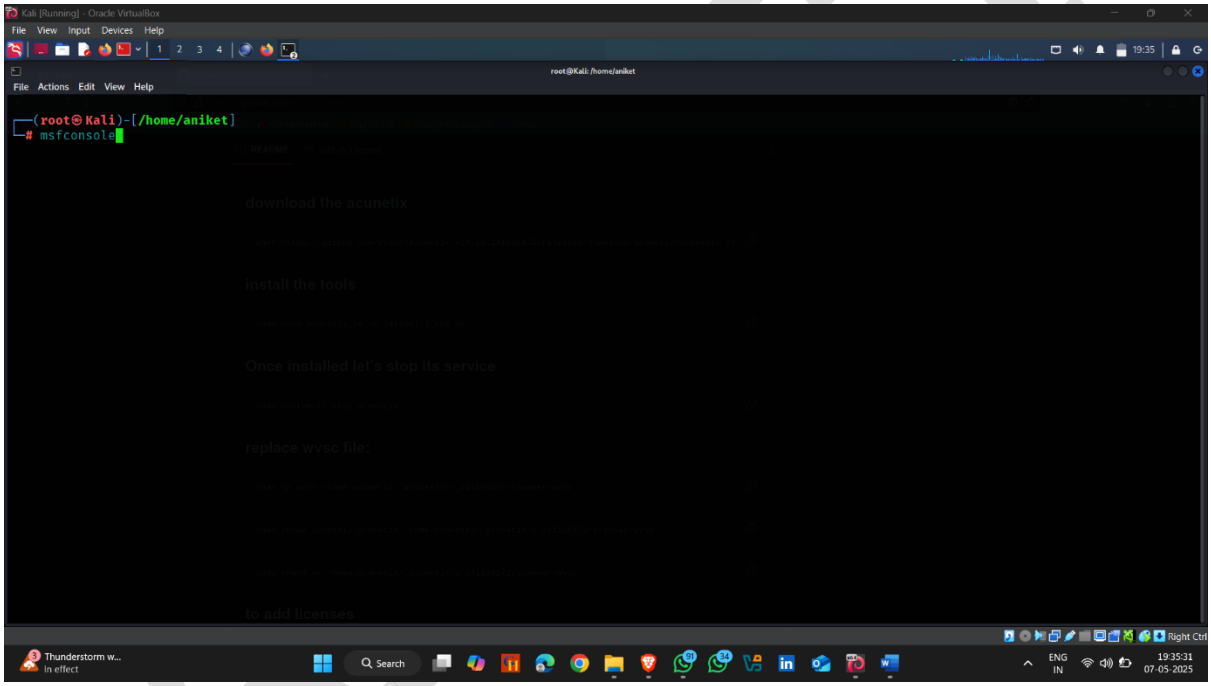
- A DDoS using both **SYN Flood (protocol layer)** and **HTTP Flood (application layer)**.
- Combining **DNS Amplification** and **Slowloris** attacks in a coordinated way.

1.Perform DOS/DDOS Using Metasploit

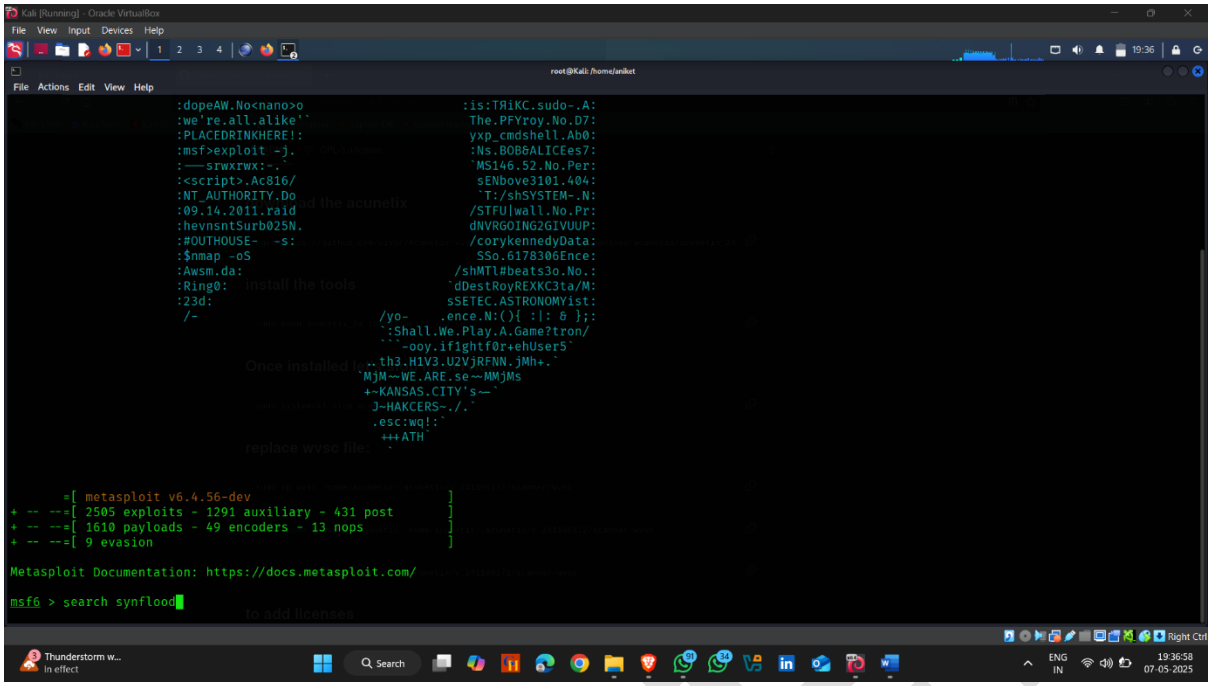
Metasploit is a powerful and widely used **penetration testing framework** that helps security professionals and ethical hackers **identify, exploit, and validate vulnerabilities** in systems and networks.

How to use it -:

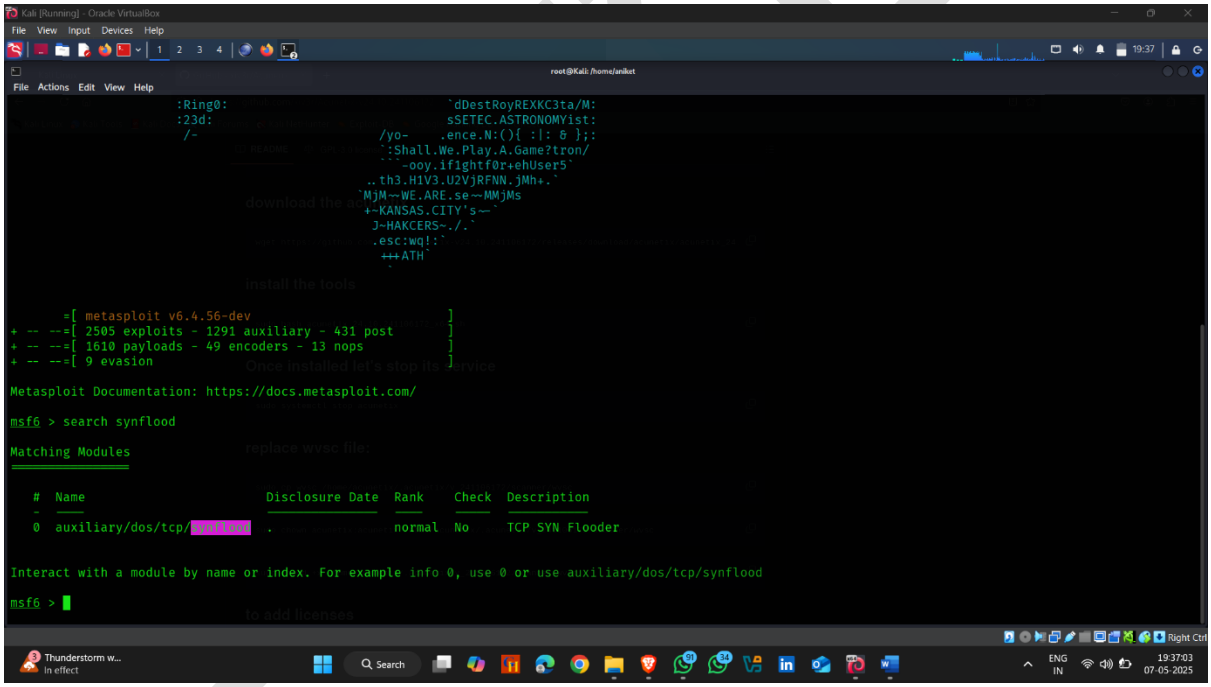
- Open kali linux terminal and type **msfconsole**



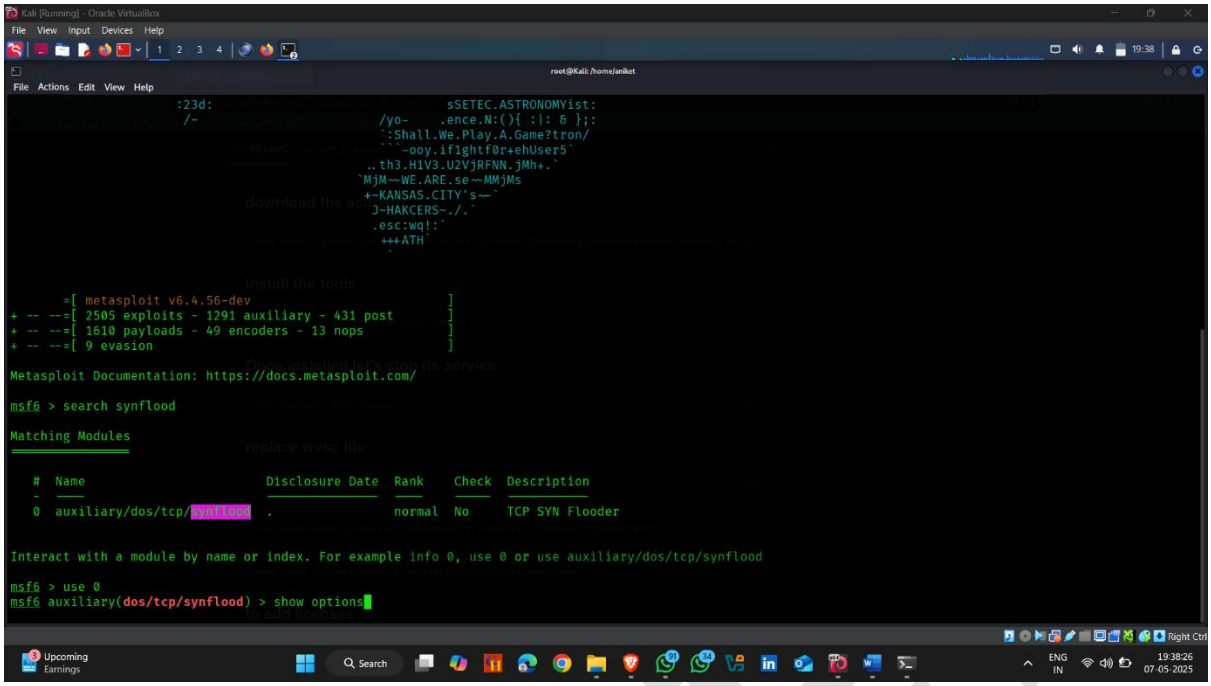
- Now search **synflood**



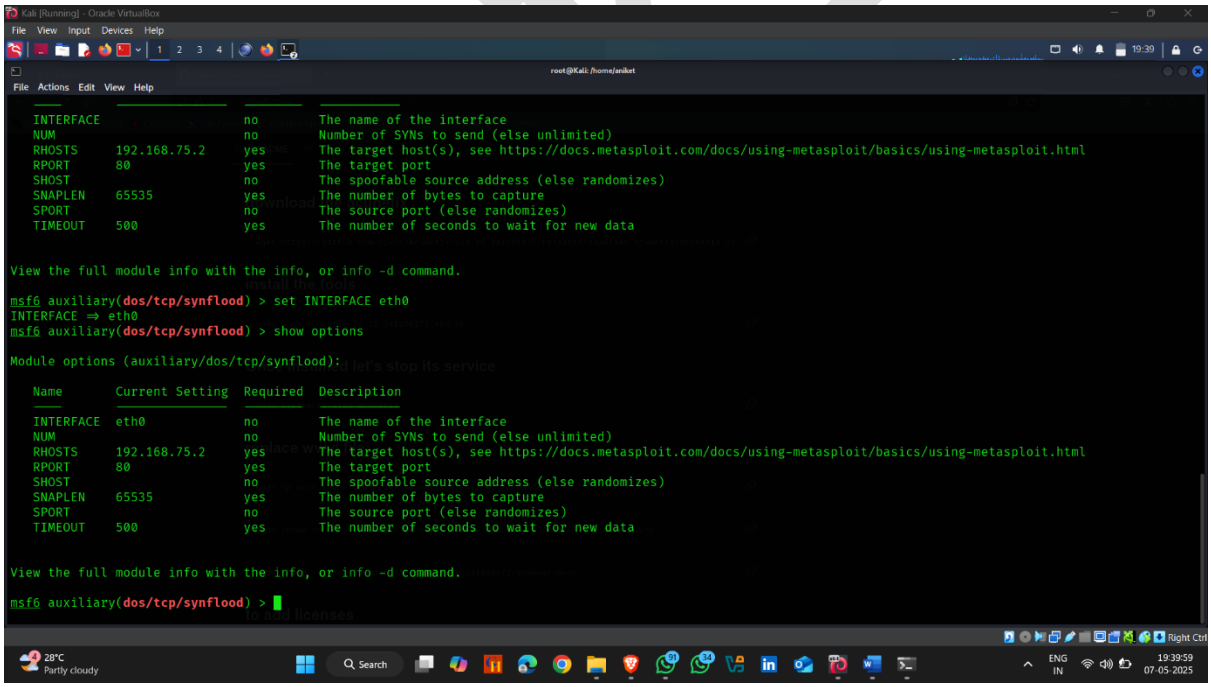
- Synflood auxiliary display



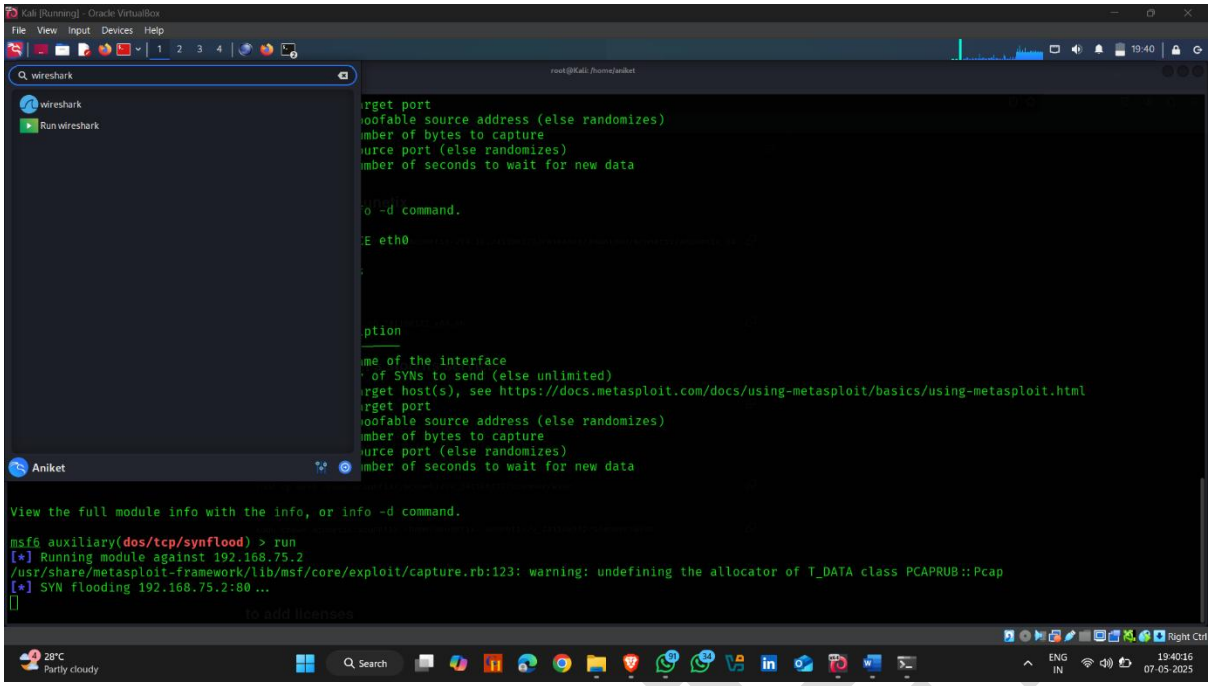
- Now use this auxiliary



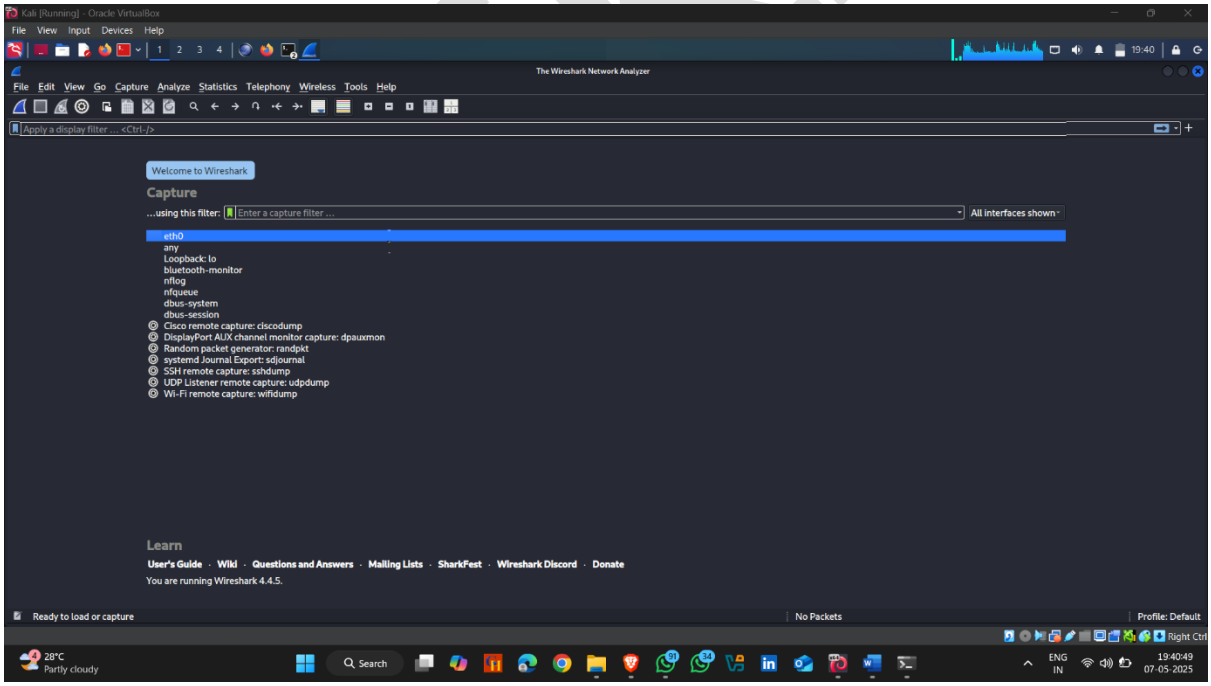
- Now set **RHOST and Network Interface**



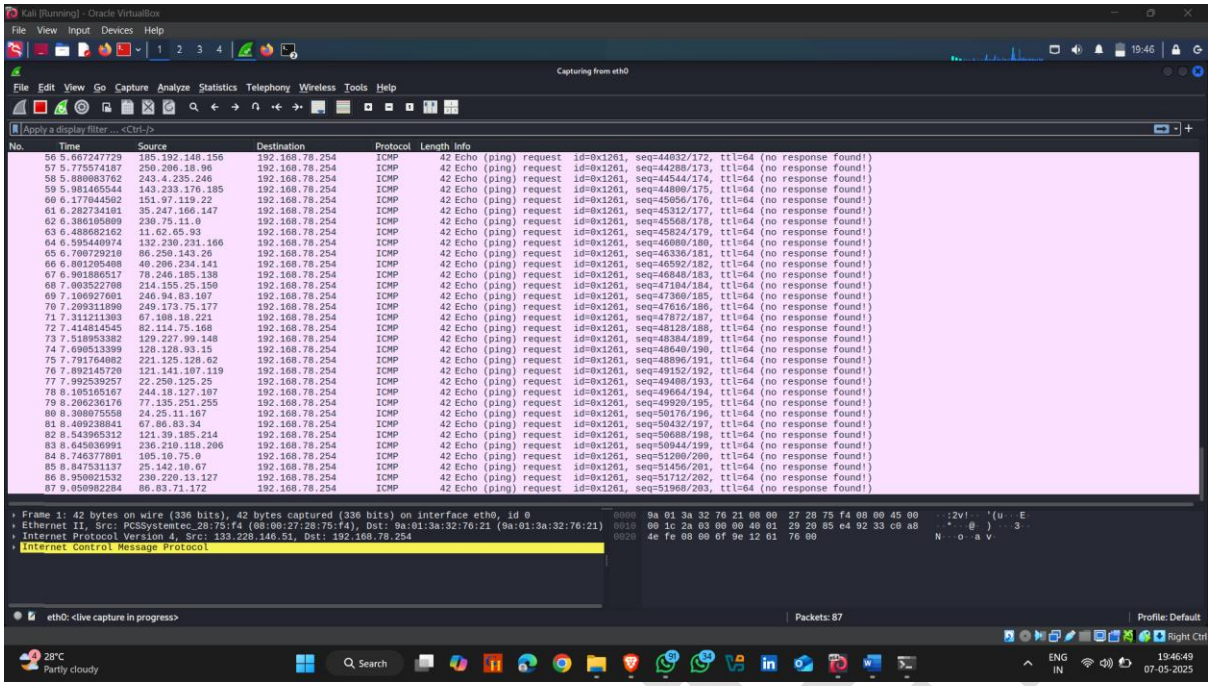
- Here , attack started , open wireshark to monitor attack



- Select eth0



- Attack started 👇



2.Perform DOS/DDOS Using Hping3

hping3 is a **command-line network tool** used for **packet crafting, scanning, firewall testing, and DoS simulation**. It is especially useful in penetration testing and network troubleshooting.

It's often referred to as a **"packet generator"** or **"network security auditing tool"**, similar to **ping**, but much more powerful.

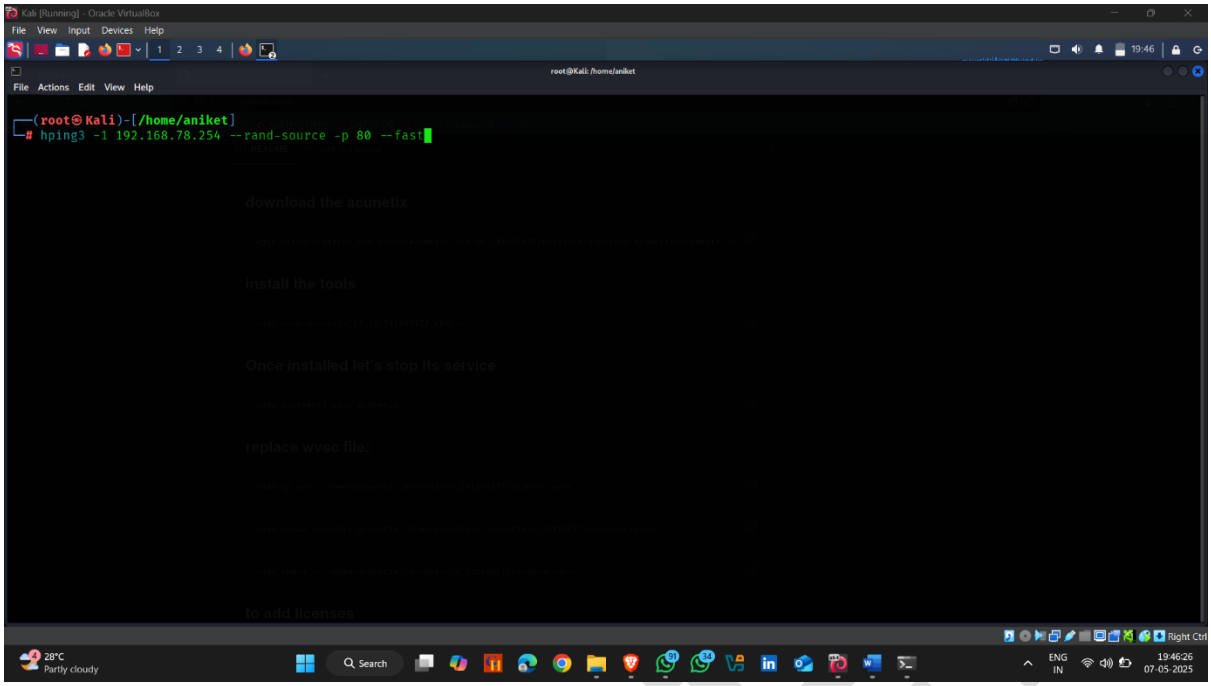
How to use it -:

- Open kali linux terminal and type hping3 command

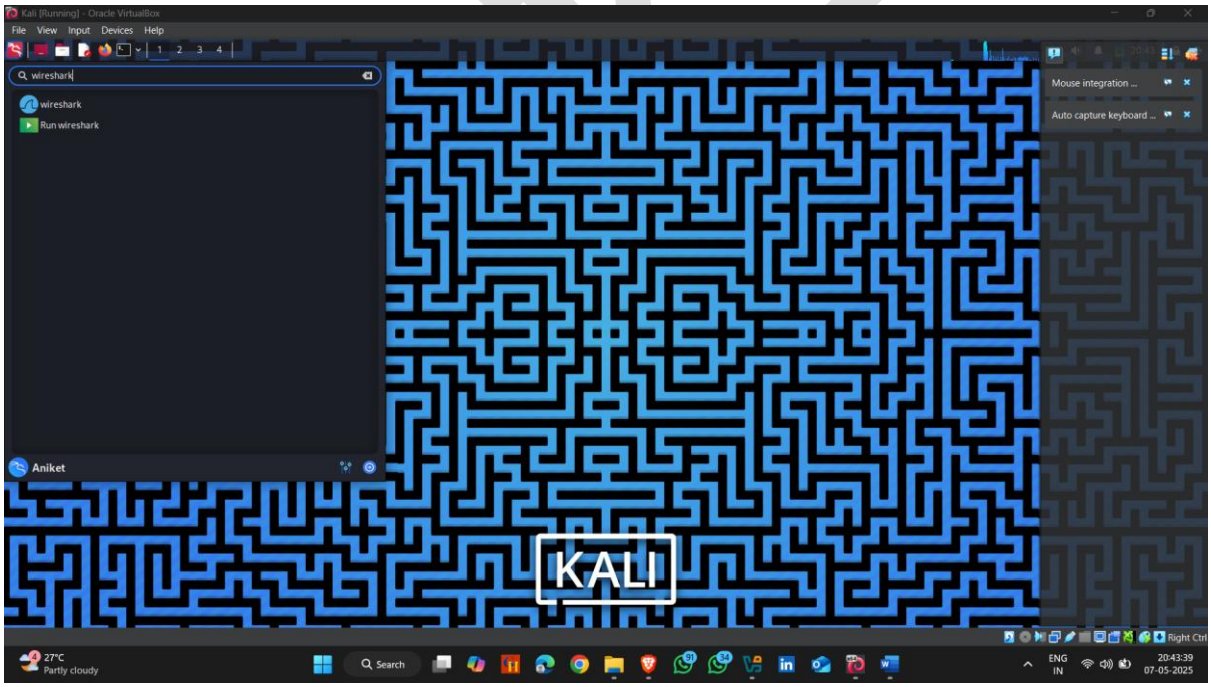
Command -: hping3 -1 <target ip> --rand-source -p 80 –fast

Explaination-:

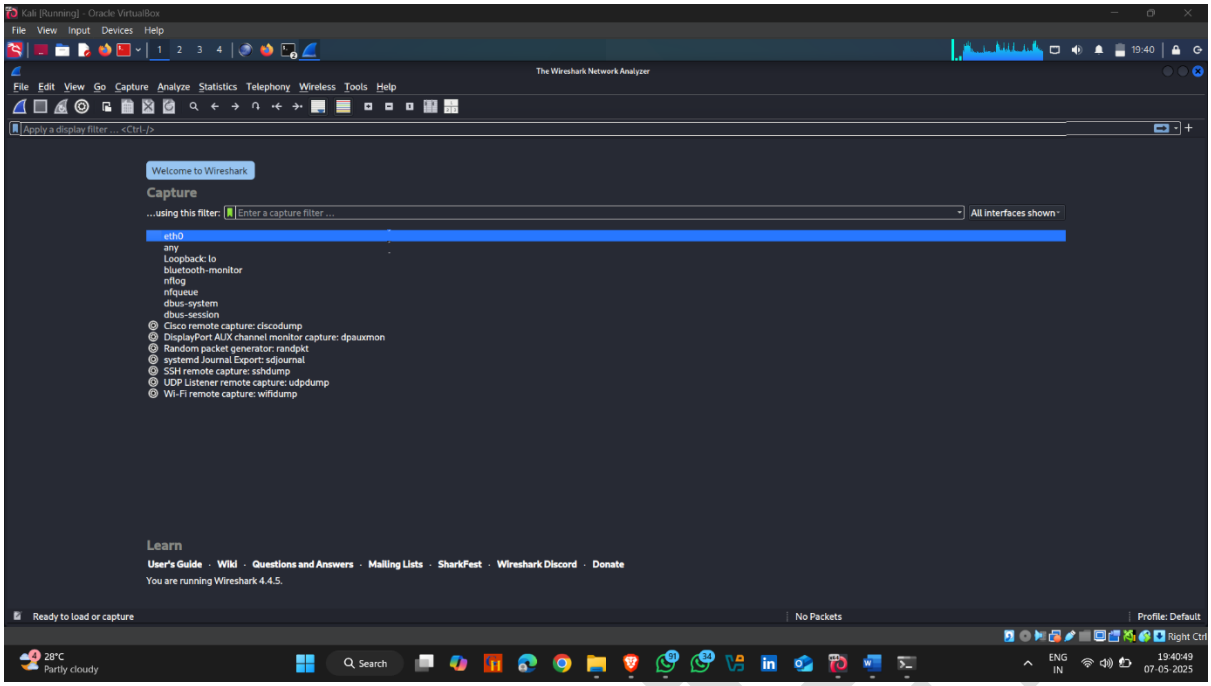
- -1 – for ICMP Packets
- --rand-source – Random ip sources
- -p – port
- 80 –port number



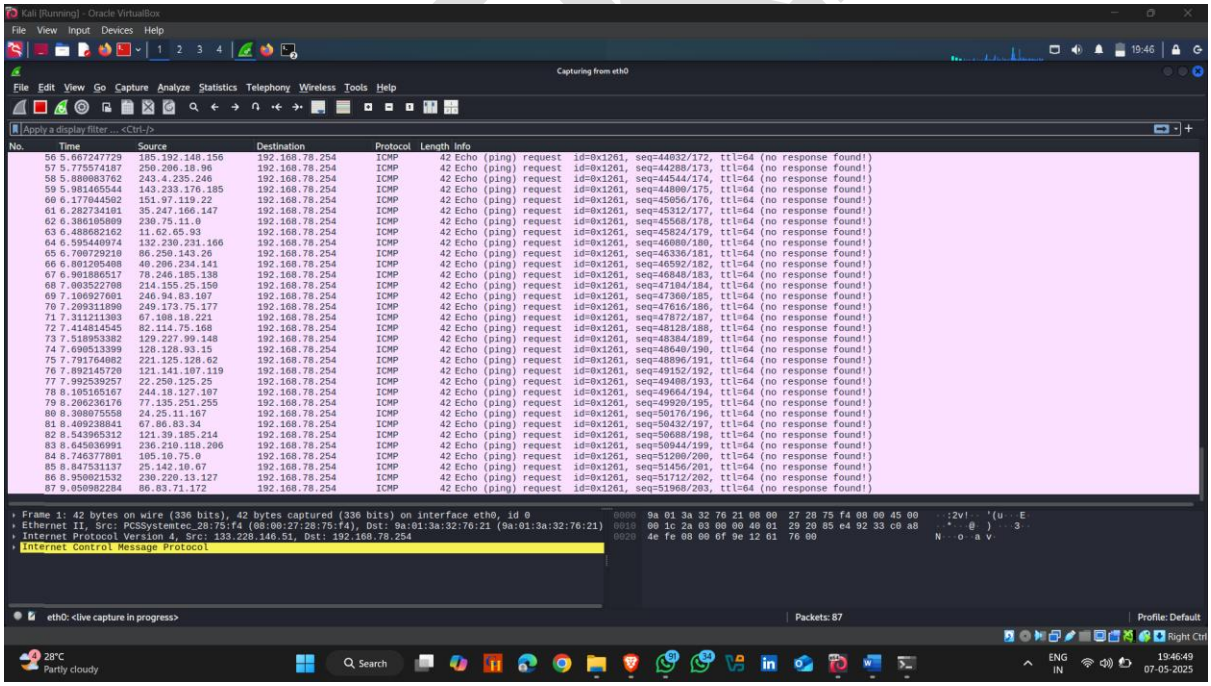
- Open wireshark to monitor attack



- Select eth0



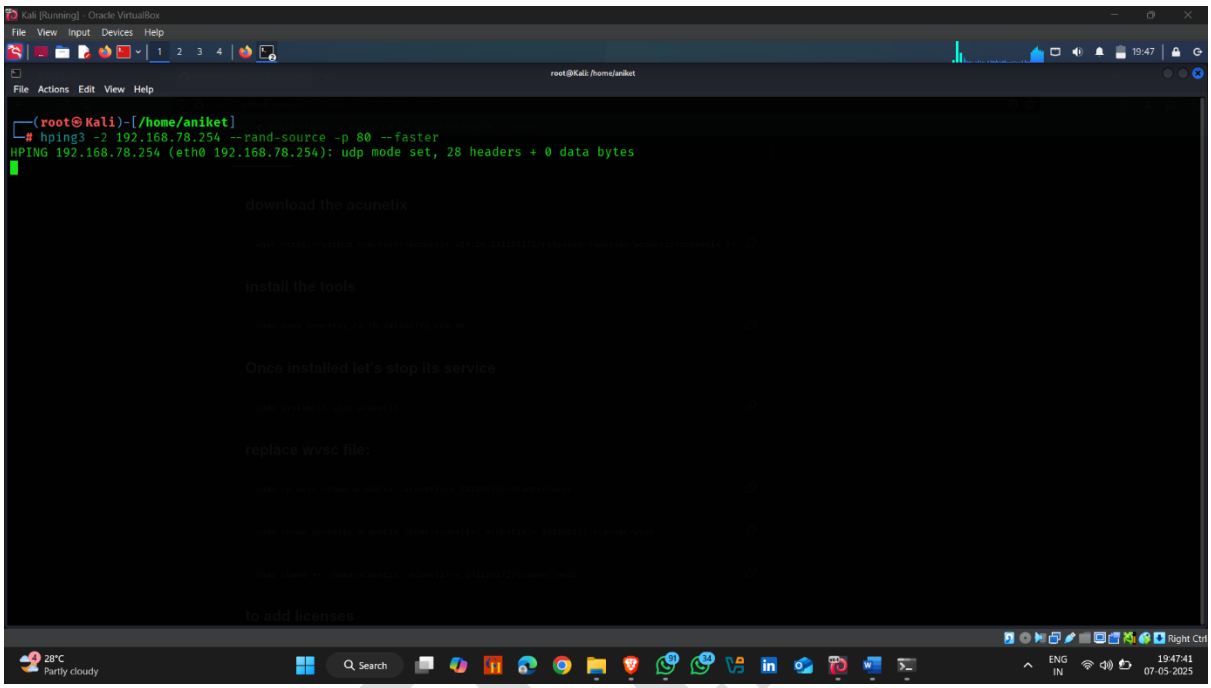
- Attack Started 👇



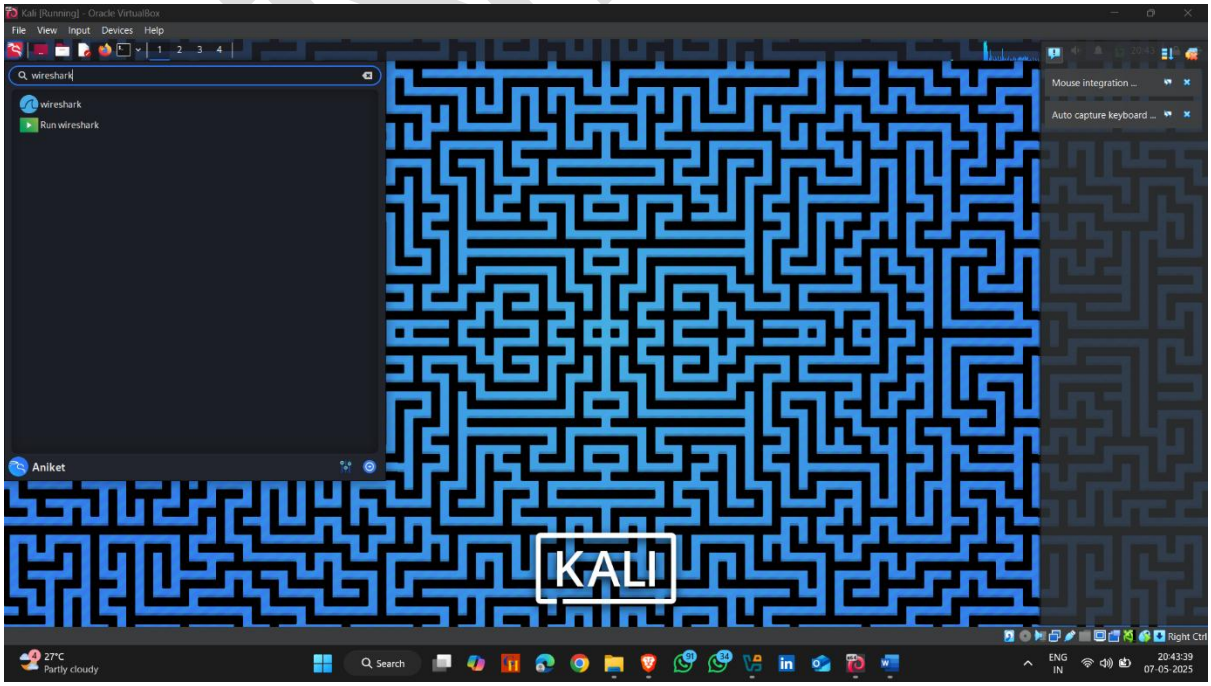
Next Command -: hping3 -2 <target ip> --rand-source -p 80 –faster

Explaination -:

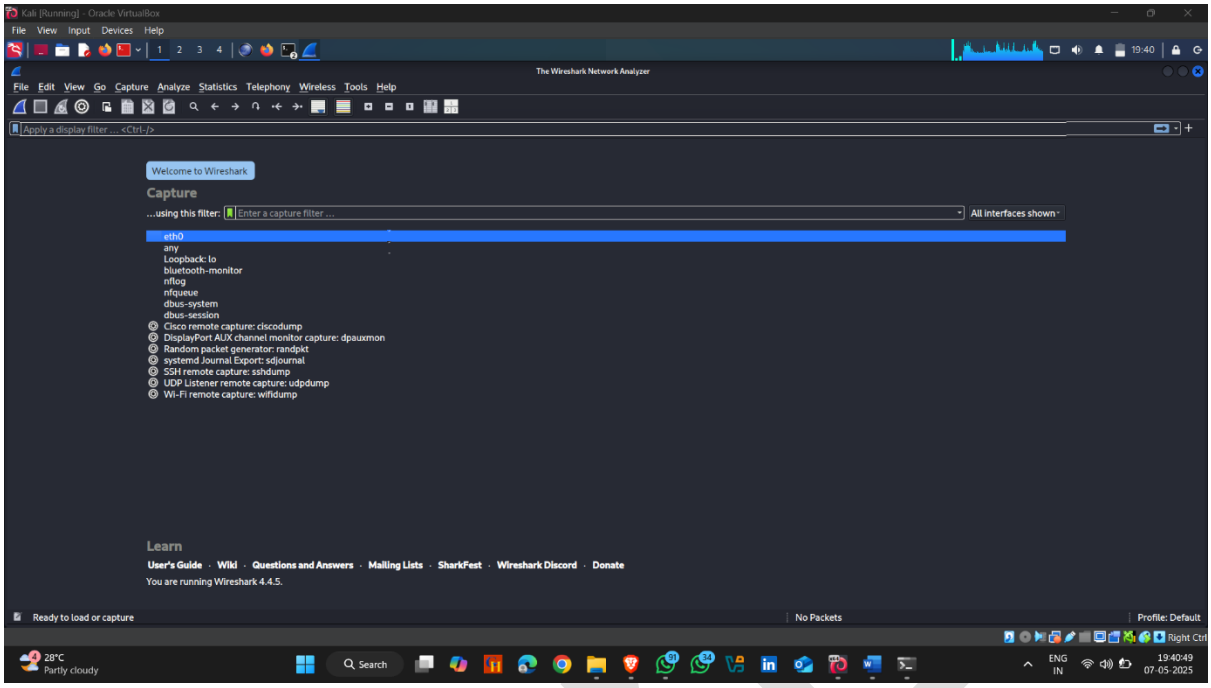
- **-2 – For Sending UDP Packet**
- **--rand-source – generate random ip address**
- **-p –port**
- **80 – port number**
- **--faster – increase speed of sending packets**



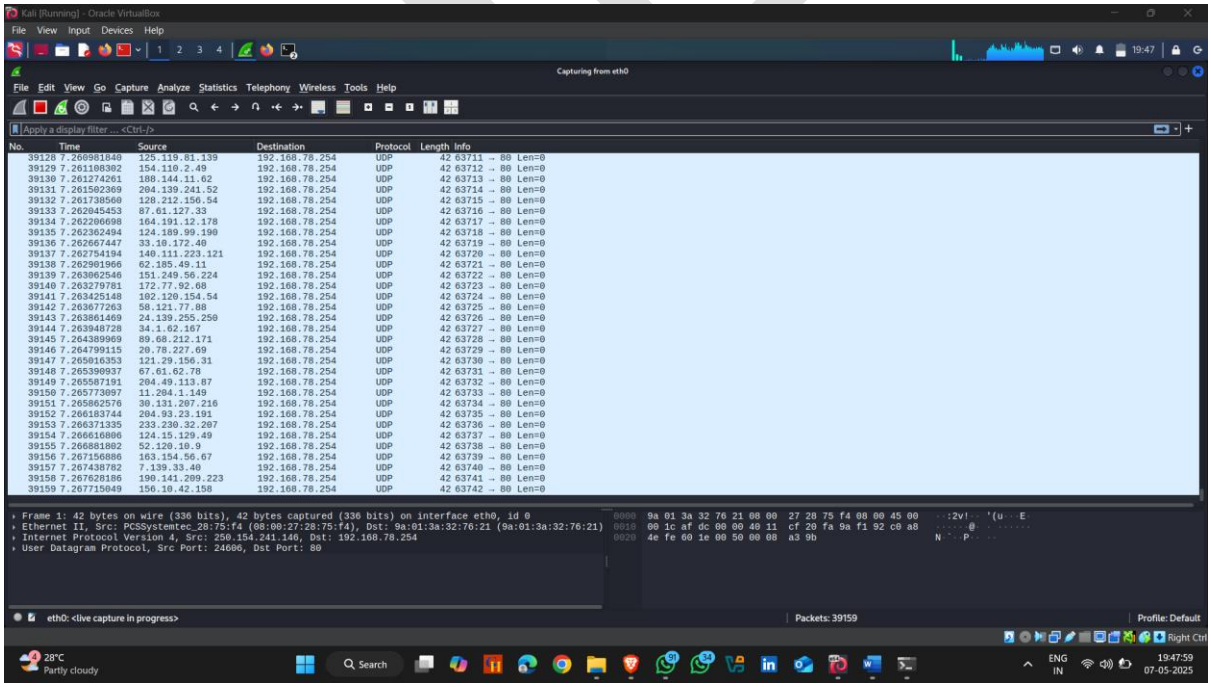
- Open Wireshark



- Select eth0



- Sending udp packets 👇

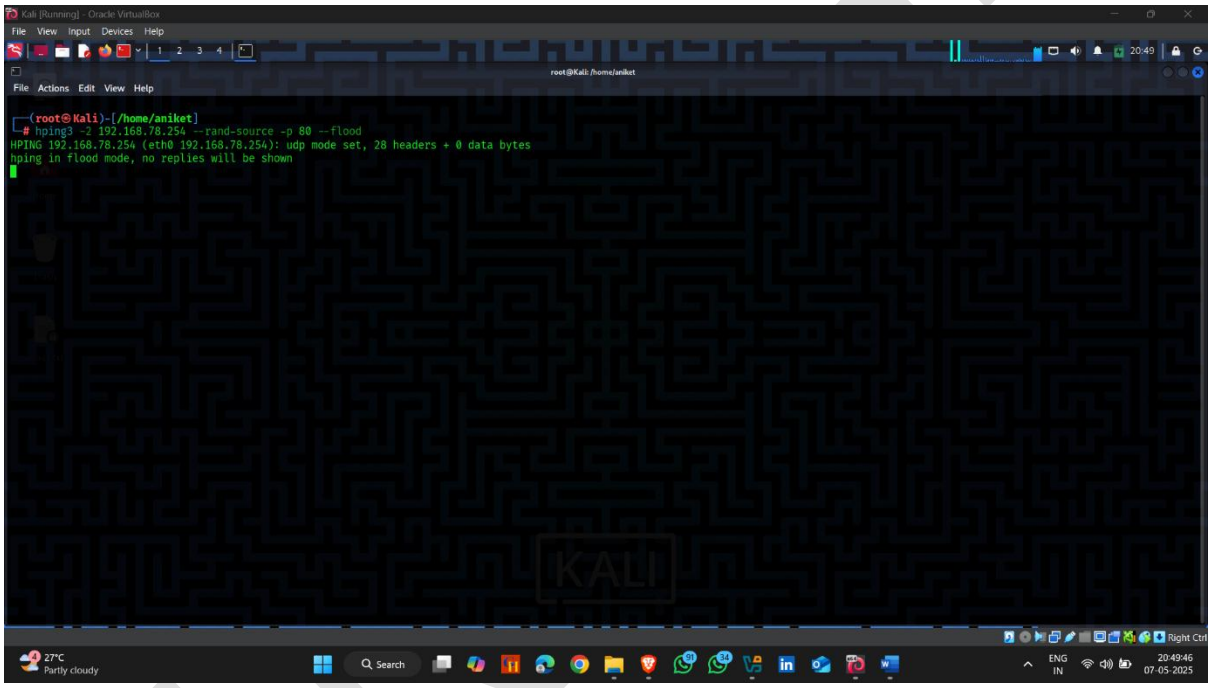


Next command using Hping3

Command -: hping3 -2 <target ip> --rand-source -p 80 –flood

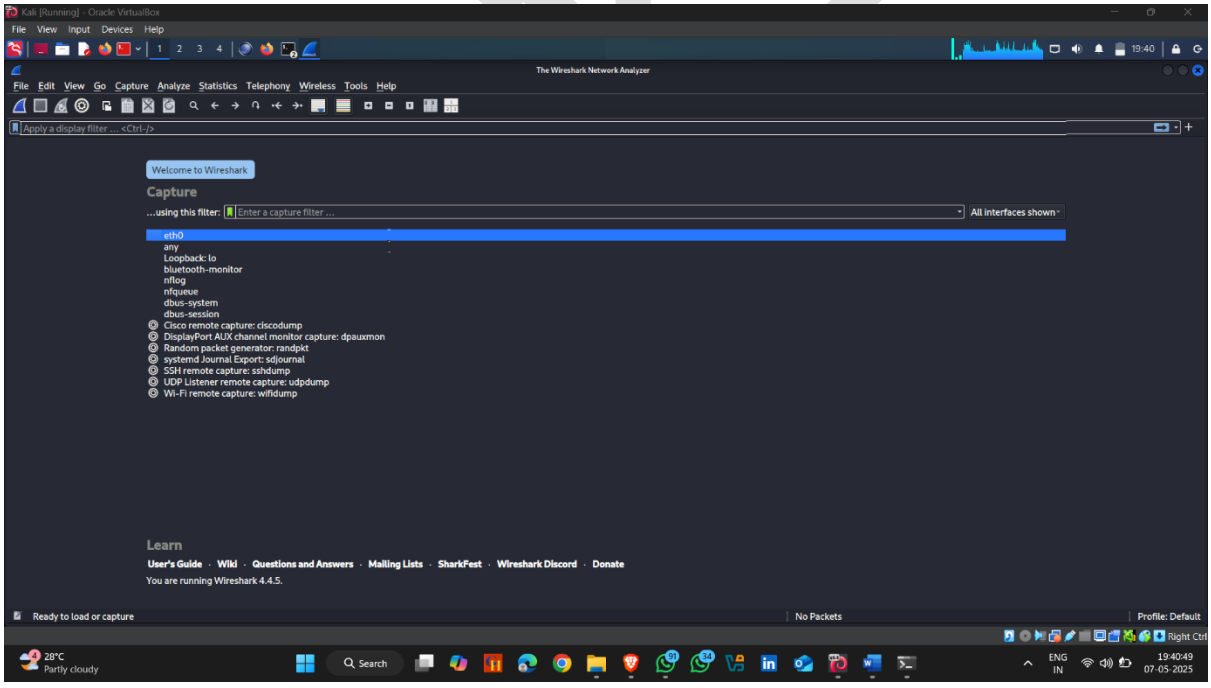
Explaination -:

- **-2 – For Sending UDP Packet**
- **--rand-source – generate random ip address**
- **-p –port**
- **80 – port number**
- **--flood – flooding of packets**

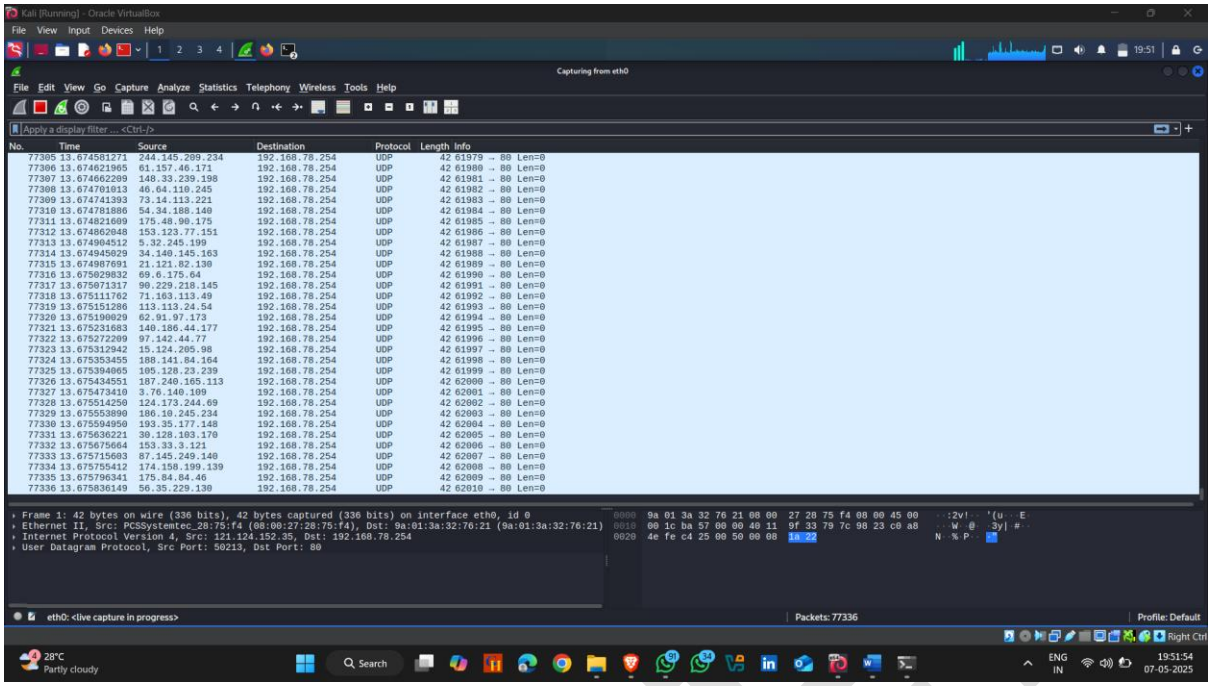


- Open wireshark for monitor network traffic

- Use eth0



- Attack Started 👇

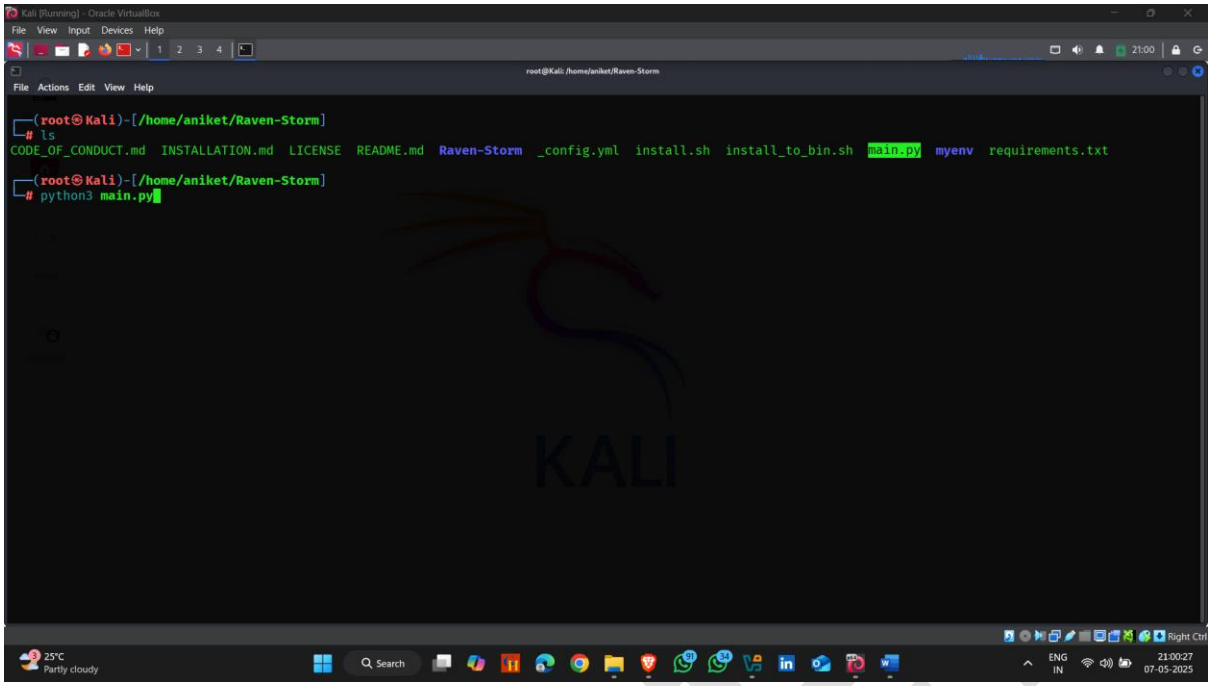


3.Perform DOS/DDOS Using Raven-Storm

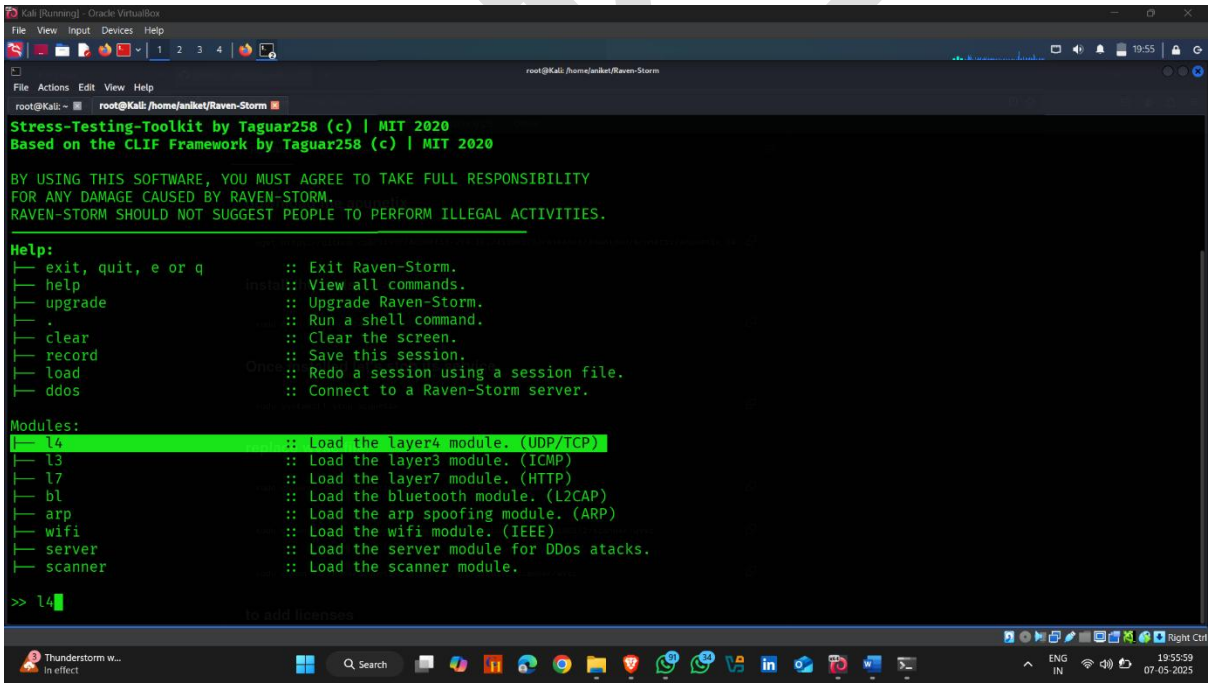
Raven-Storm is an **open-source DoS/DDoS attack tool** designed for **educational and stress-testing purposes**. It's written in **Python** and allows users to **simulate denial-of-service attacks** on local networks or lab environments.

How to use it -:

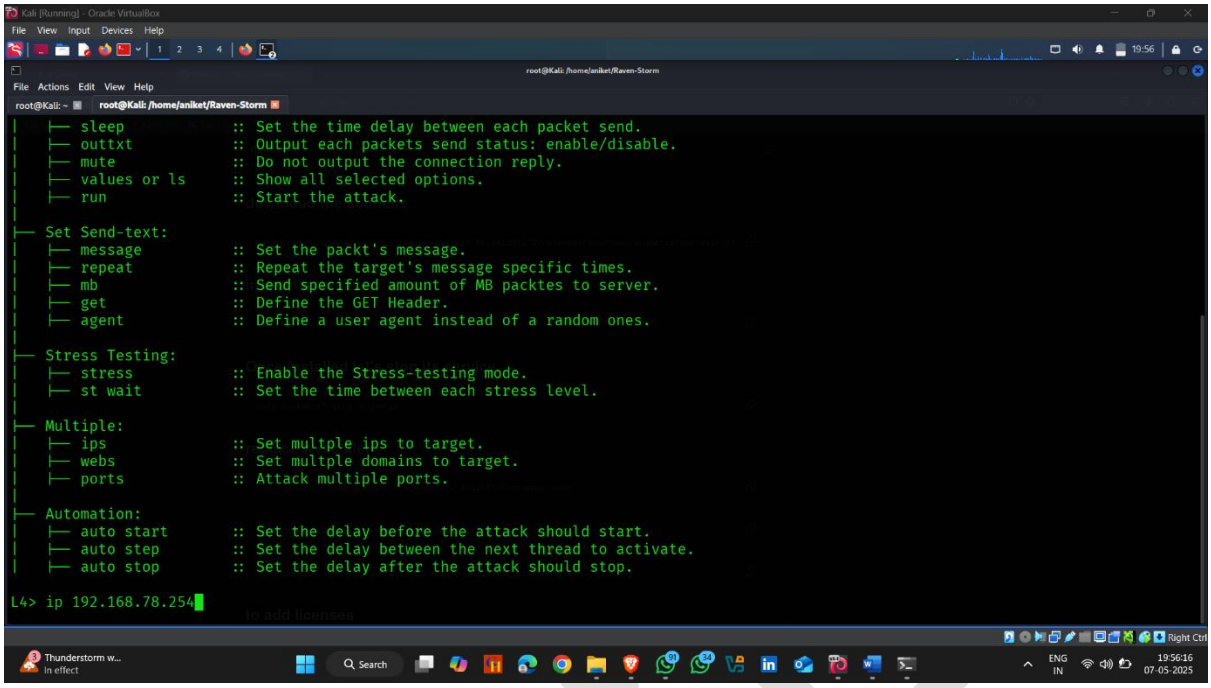
- First download tool from git hub
- After Download completed Open kali linux terminal and go to the **Raven-Storm Directory**
- Use **python3 main.py** to run raven-storm



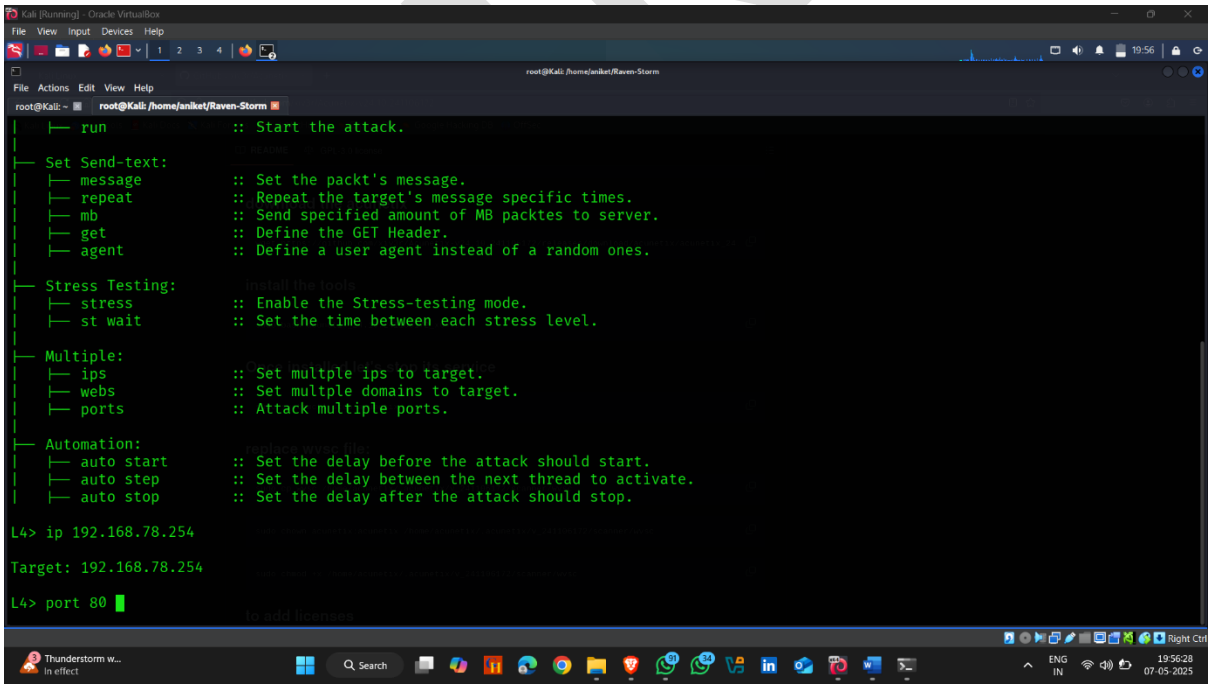
- Use L4 – for layer four attack (transport layer attack)



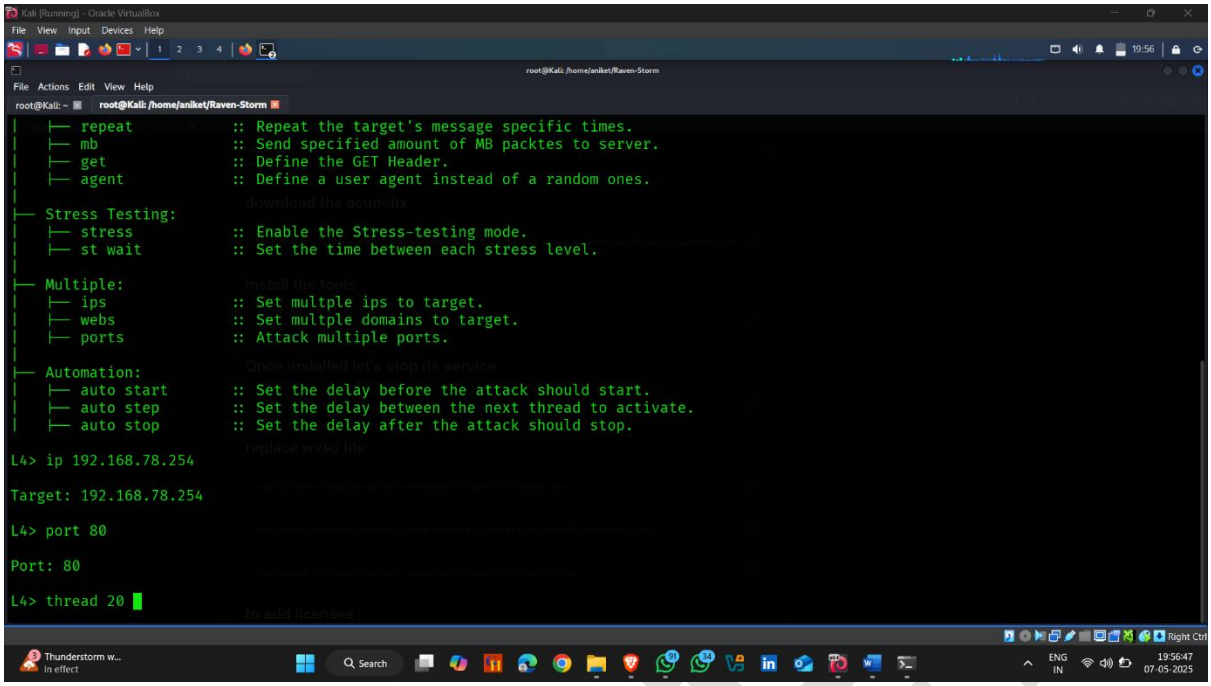
- Set target ip address



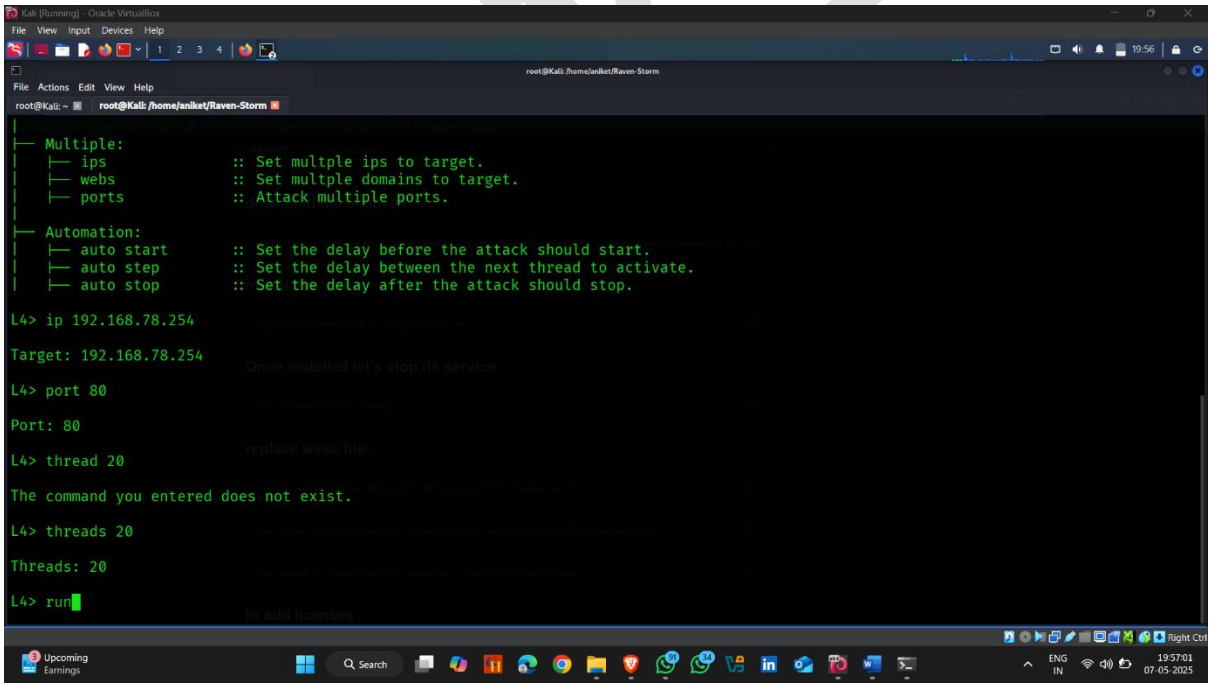
- Set port number



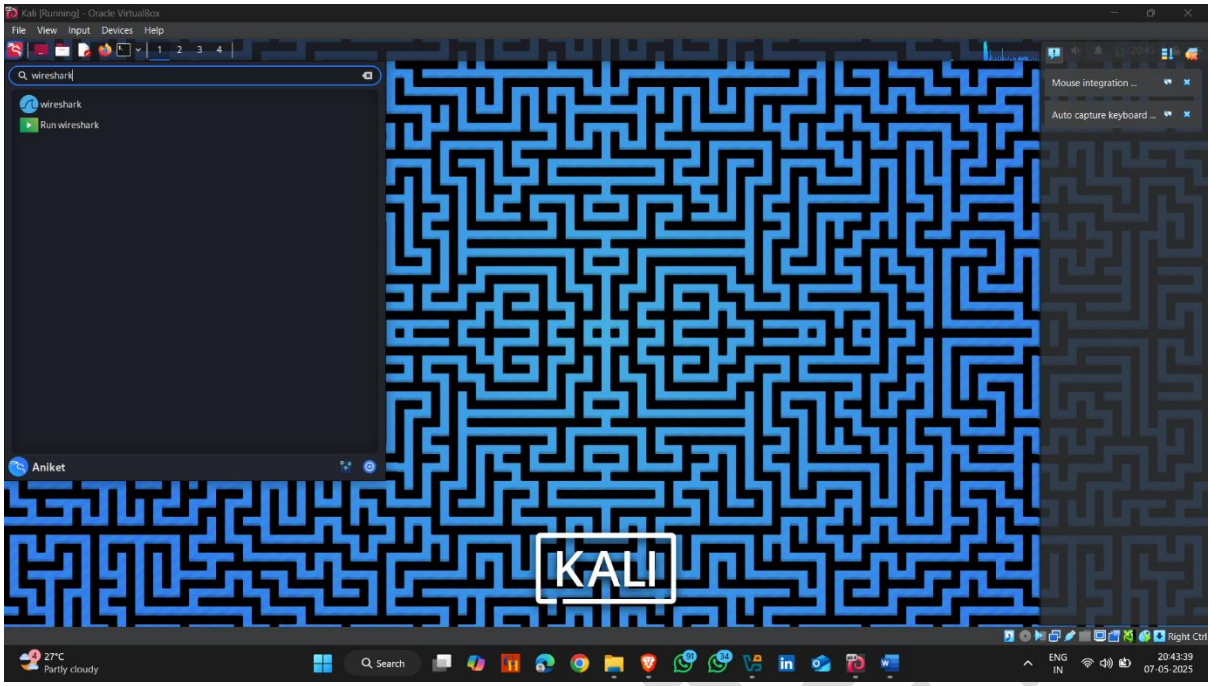
- Set threads



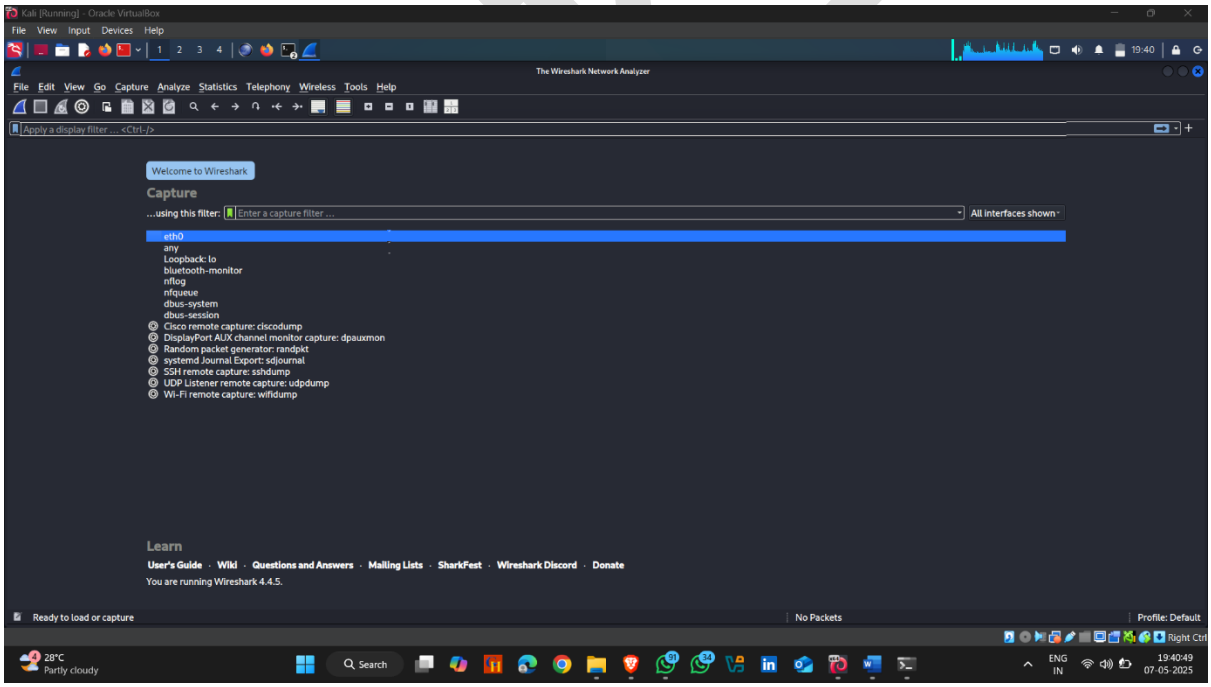
- Run



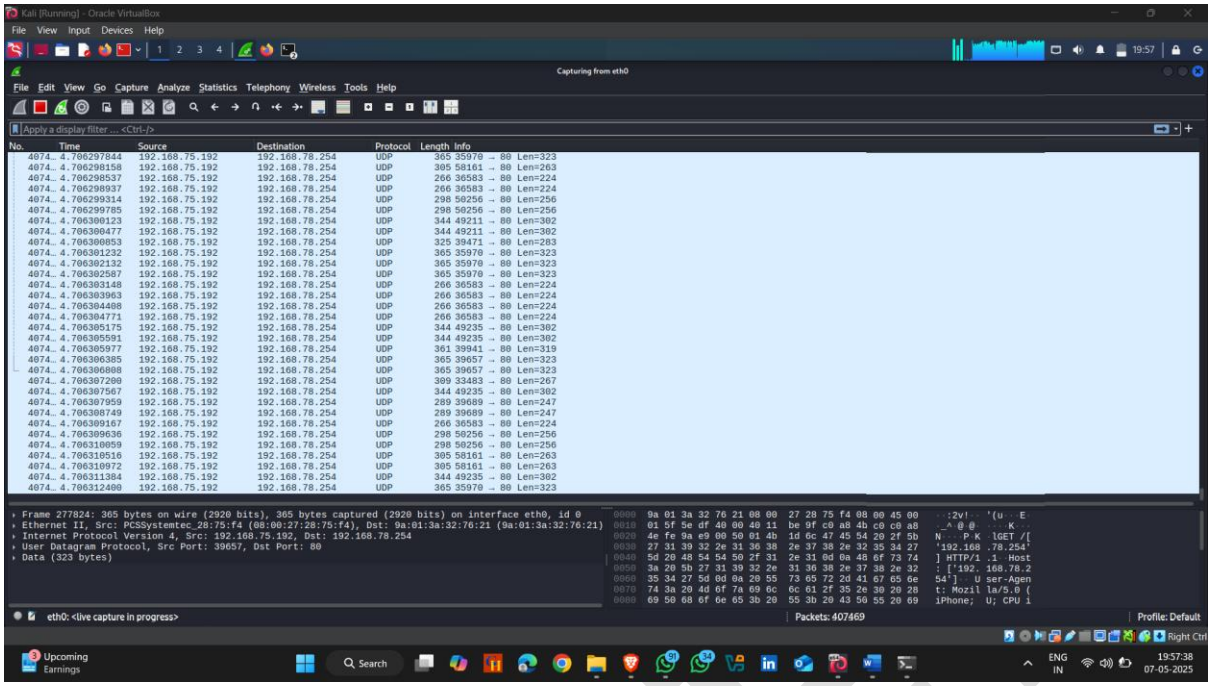
- Open wireshark



- Use eth0



- Attacker started 👇

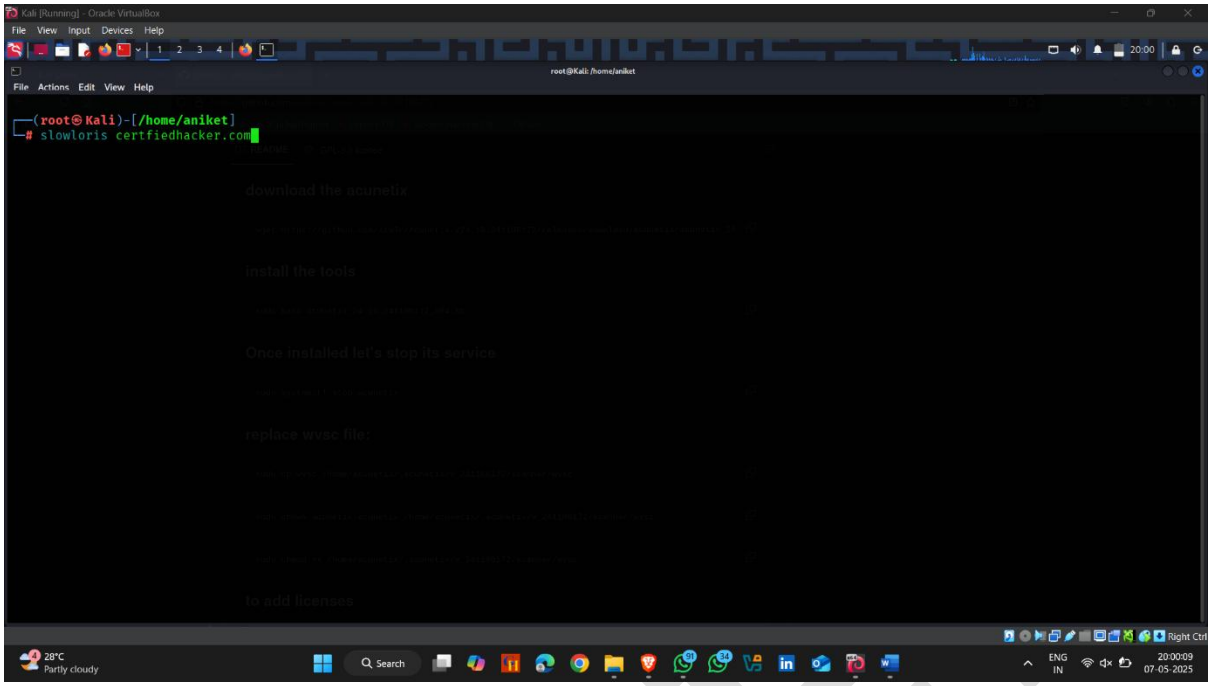


4.Perform DOS/DDOS Using Slowloris

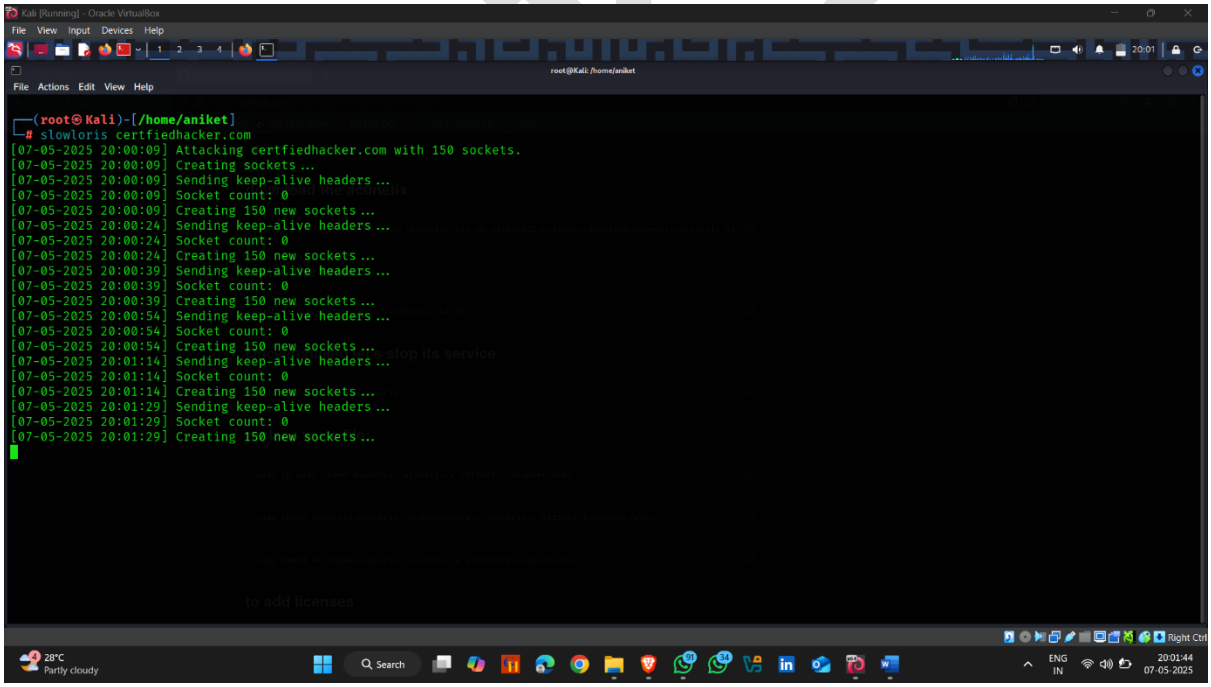
Slowloris is a **Denial-of-Service (DoS) attack tool** that targets web servers by **exploiting how they handle connections**. It allows **one single machine** to take down a web server by **keeping many connections open** and **slowly sending partial HTTP requests**, never completing them.

How to use it -:

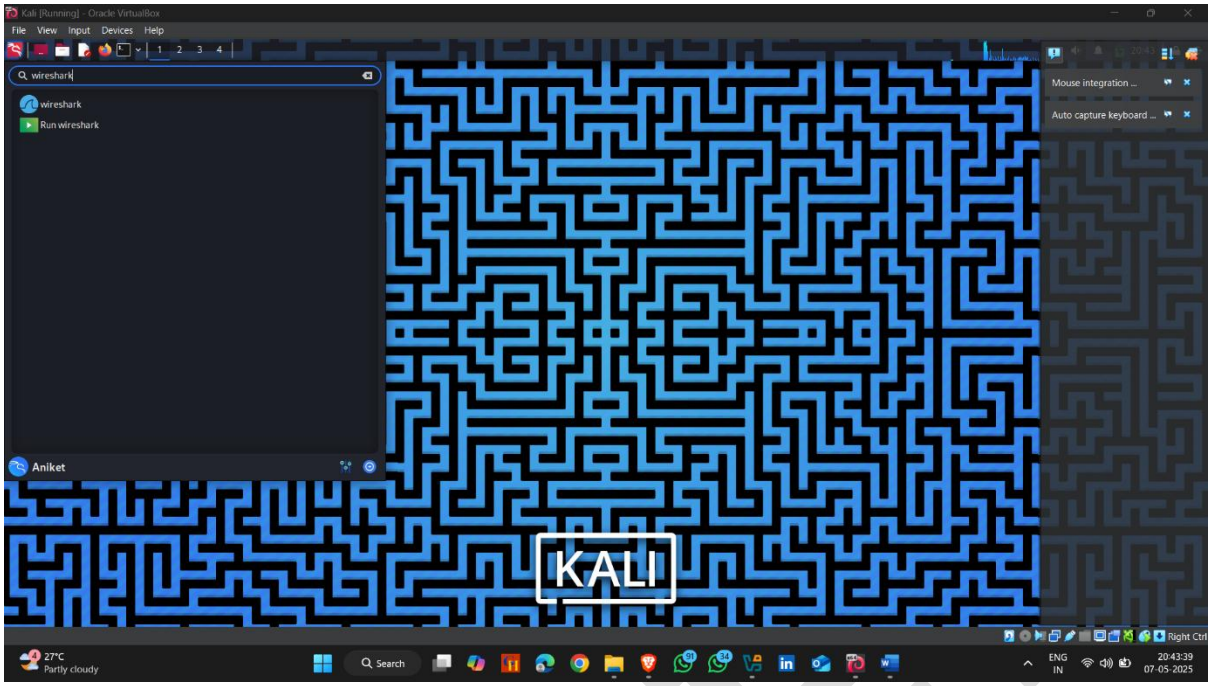
- Open kali linux terminal and type **slowloris <target url >**



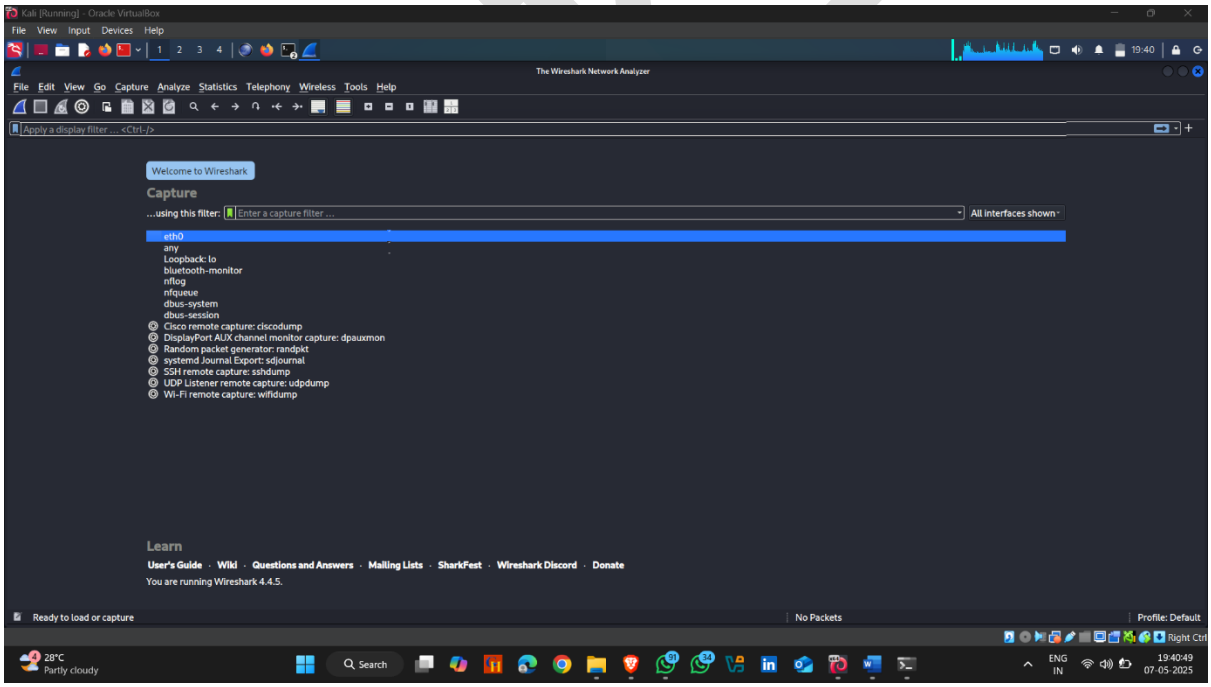
- Attack started 👇



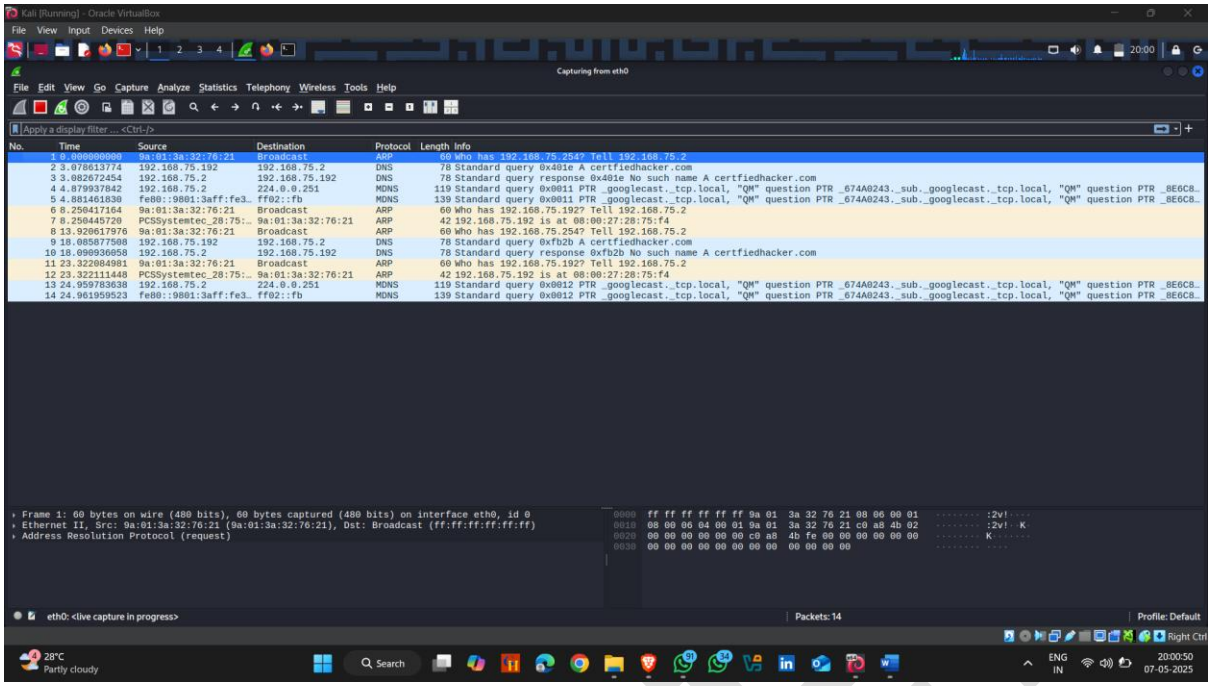
- Open wireshark



- Select network interface



- Attack Started 👇

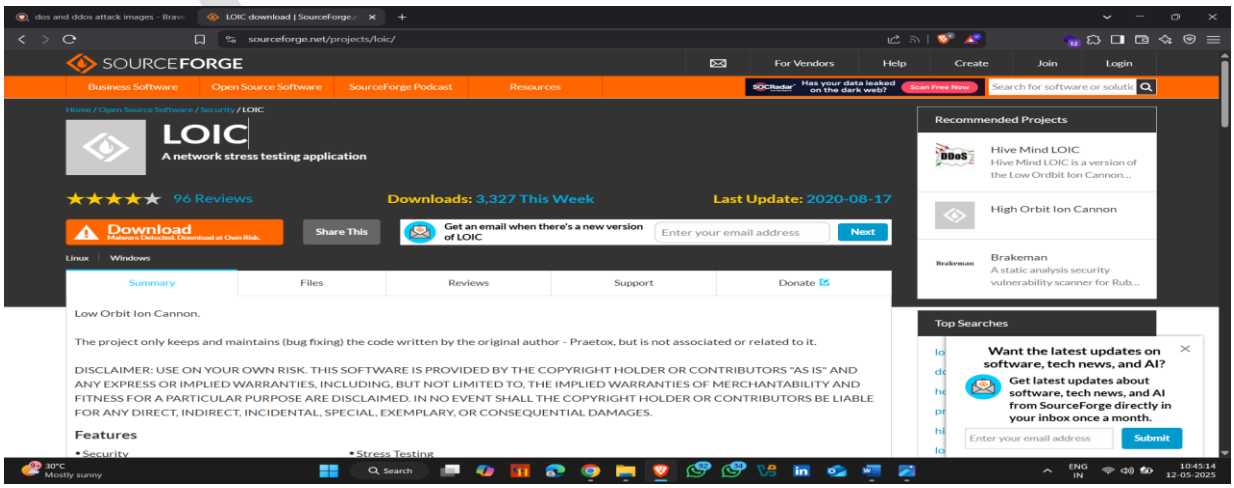


5.Perform DOS/DDOS Using LOIC

LOIC stands for **Low Orbit Ion Cannon** — it is an **open-source DoS (Denial of Service) attack tool** that is used to flood a target with **massive amounts of TCP, UDP, or HTTP requests**, causing the target server to **slow down or crash**.

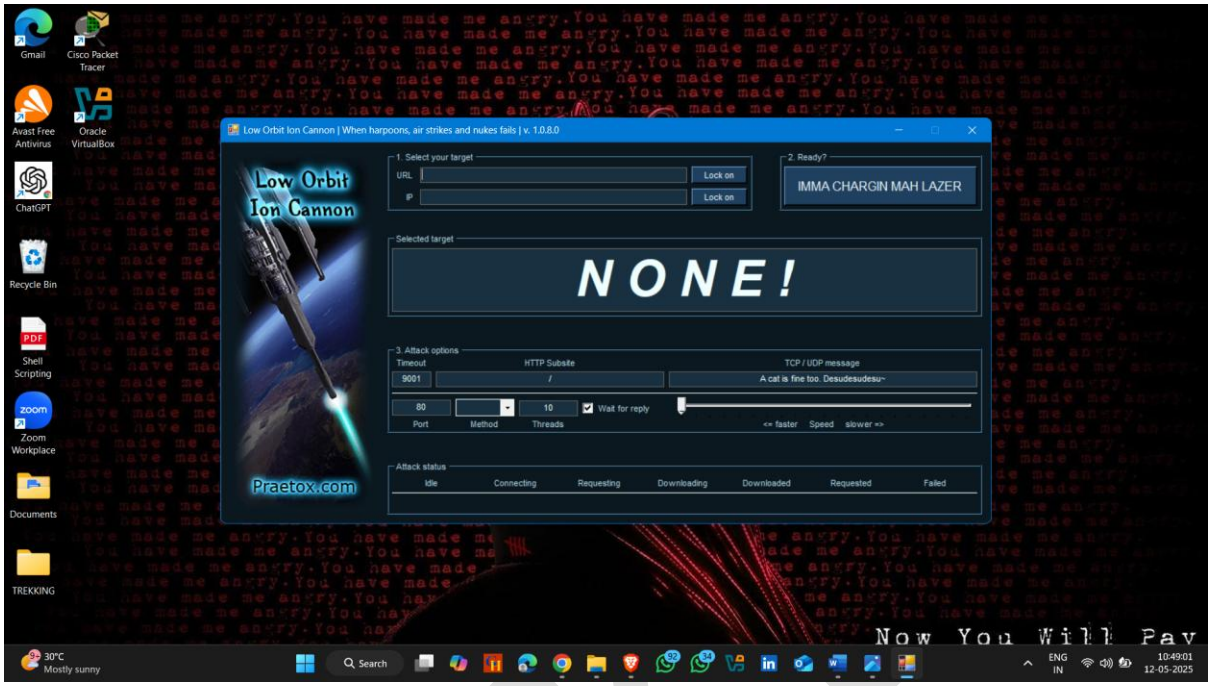
Originally developed for **network stress testing**, it became widely known when it was used by **hacktivist groups like Anonymous** in large-scale DDoS attacks.

Download Link -: https://sourceforge.net/projects/loic/

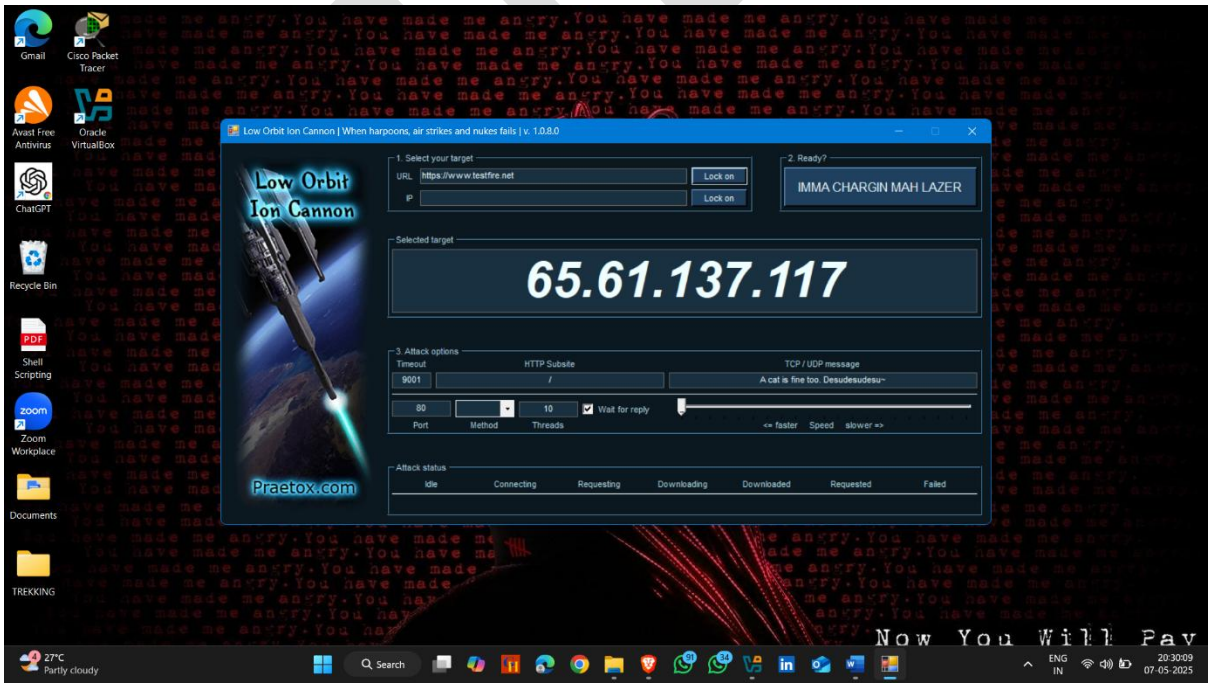


How to use it –:

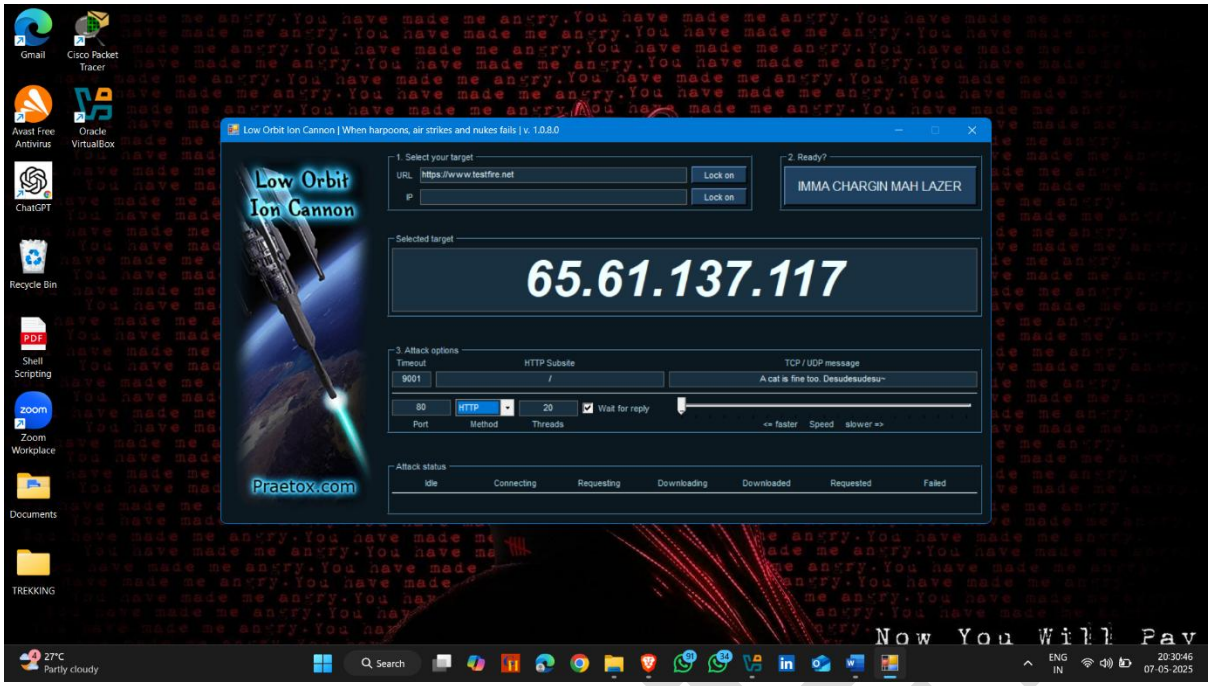
- After installation open the app



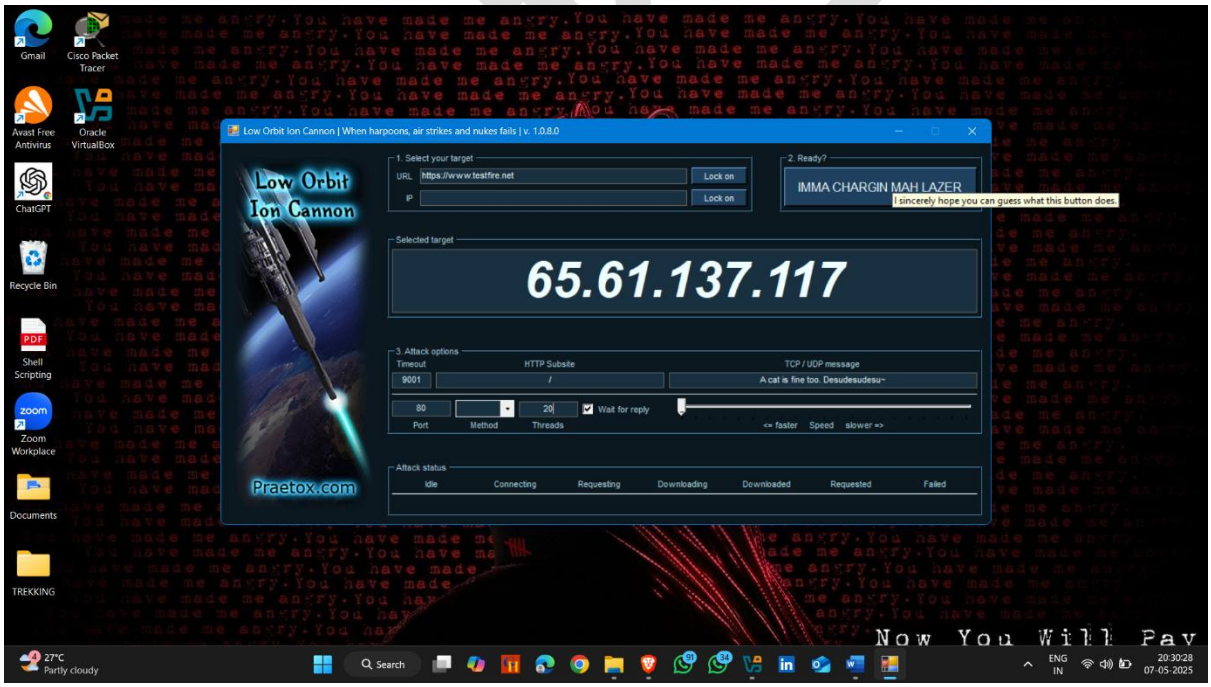
- Enter url and then click on **lock on**



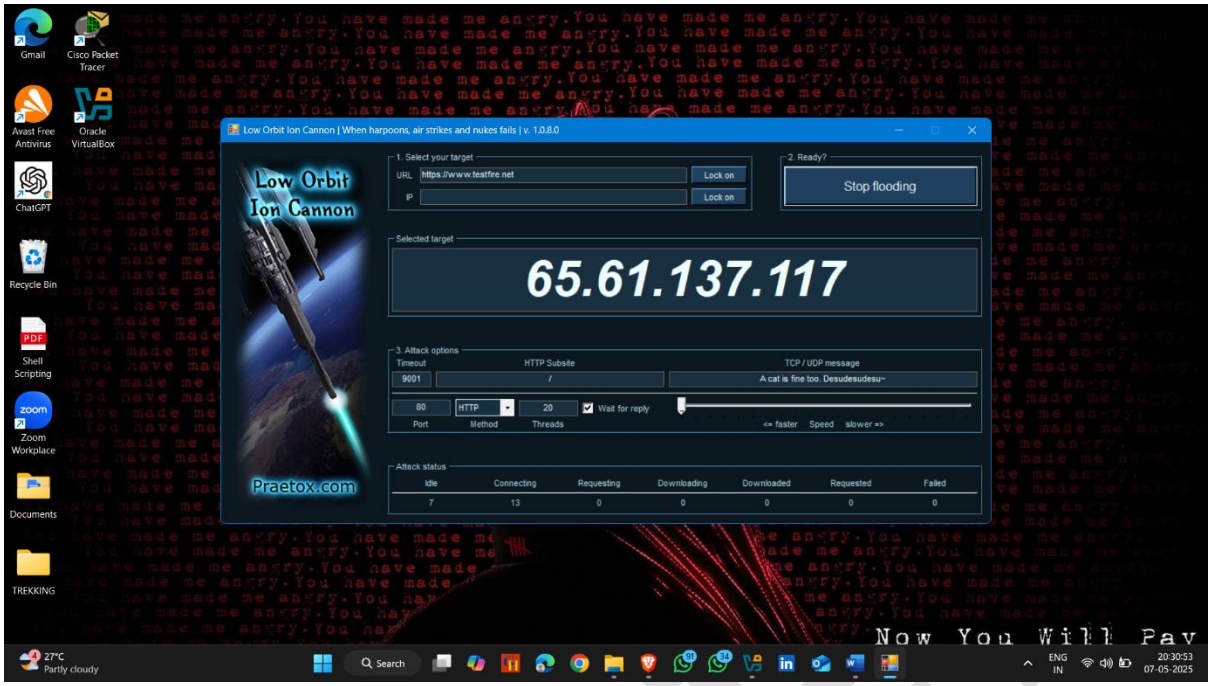
- Set Method



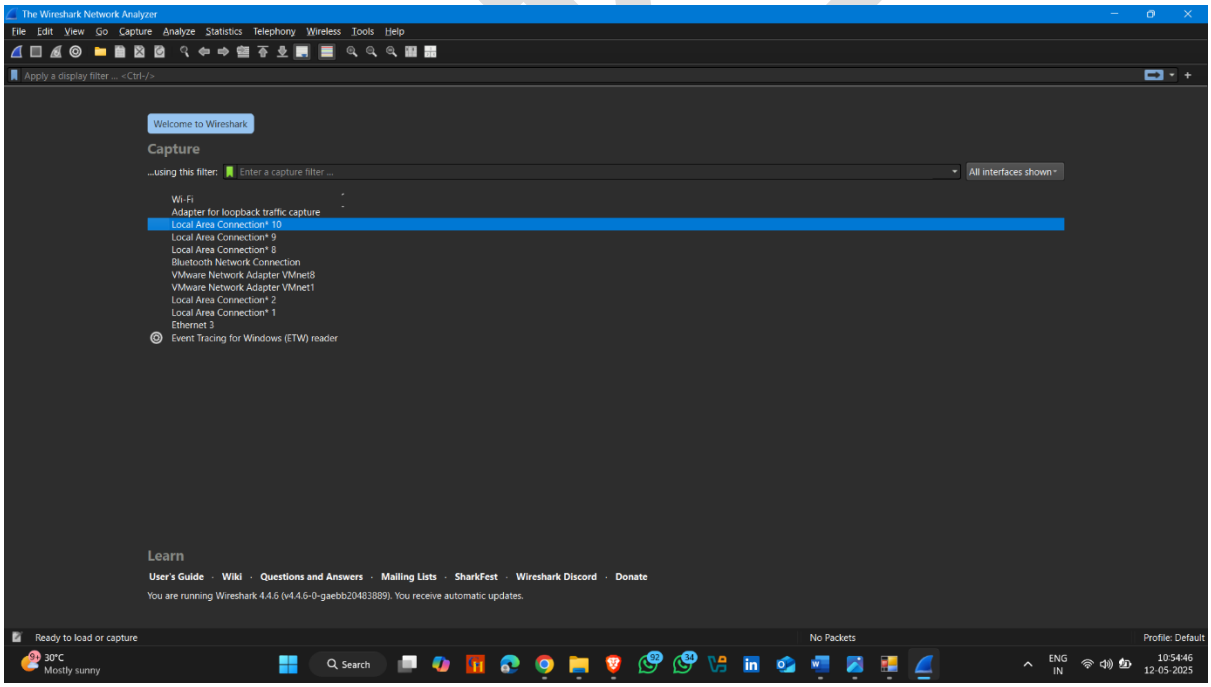
- Set threads and then click of **IMMA CHARGIN MAH LAZER**



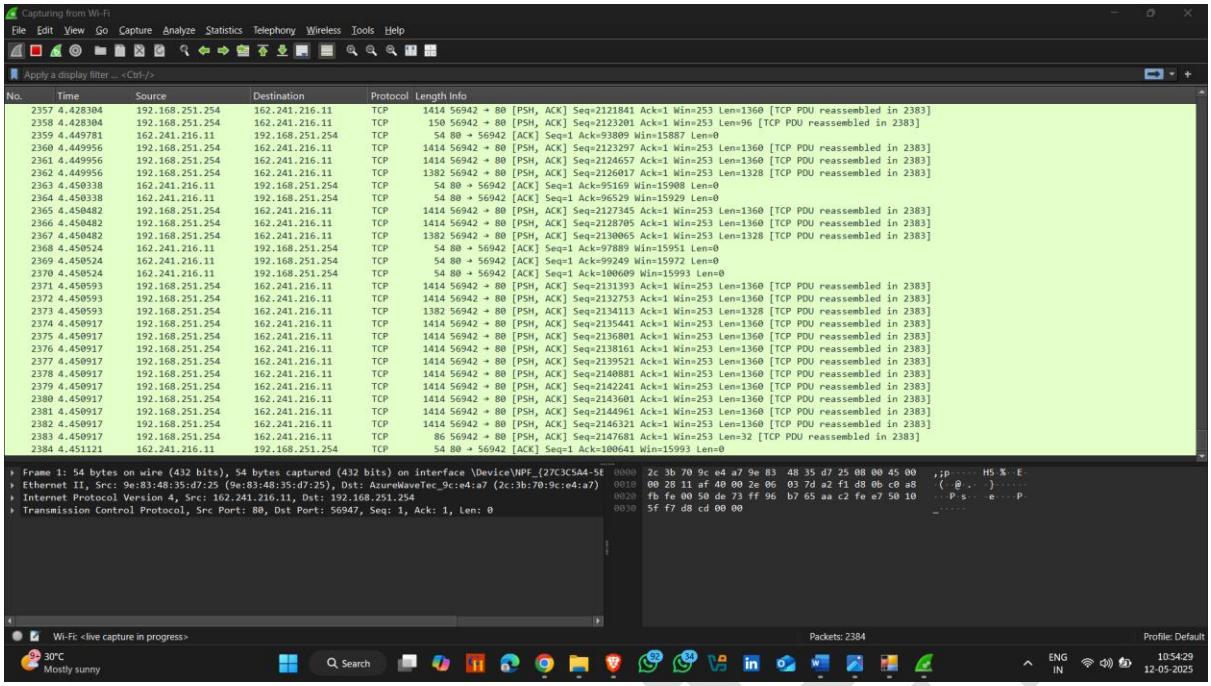
- Attack Started 👇



- Now open Wireshark for monitoring packets



- Attack started 👇

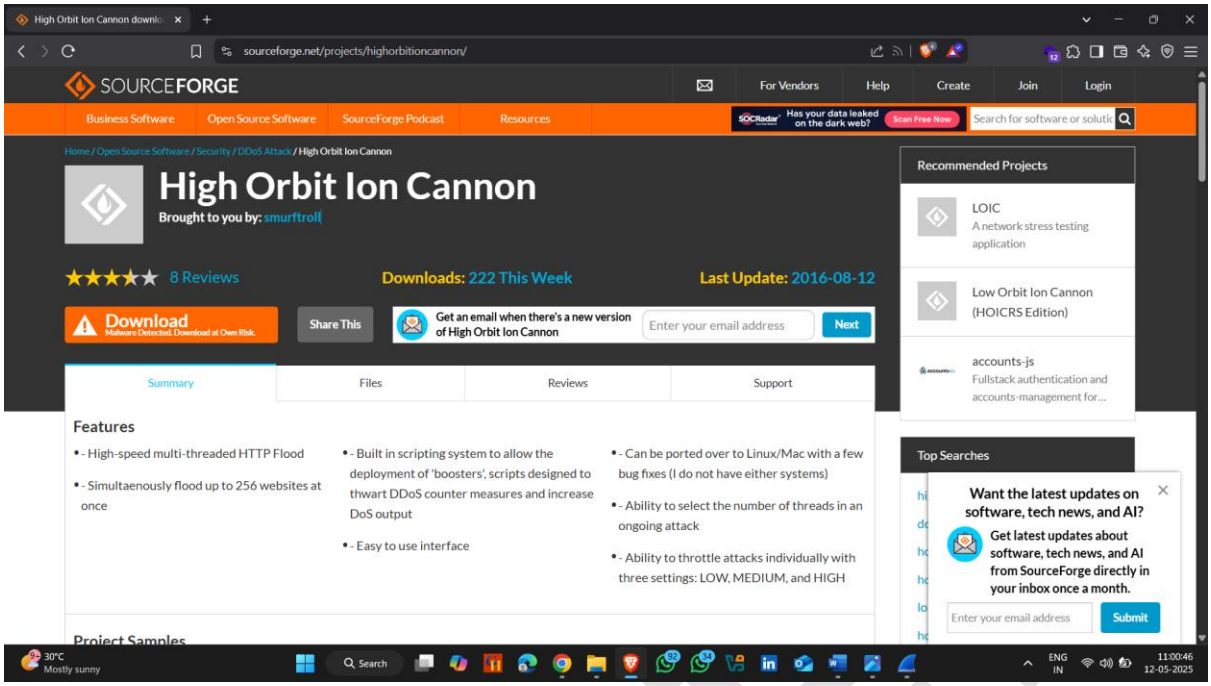


6.Perform DOS/DDOS Using HOIC

HOIC (High Orbit Ion Cannon) is a **powerful DoS (Denial of Service)** tool designed to launch **HTTP flood attacks** against web servers. It is an advanced version of the **LOIC (Low Orbit Ion Cannon)** tool but with a more **powerful and distributed attack mechanism**. It is primarily used for **flooding web servers** with HTTP requests to **overload** and **crash** the target system.

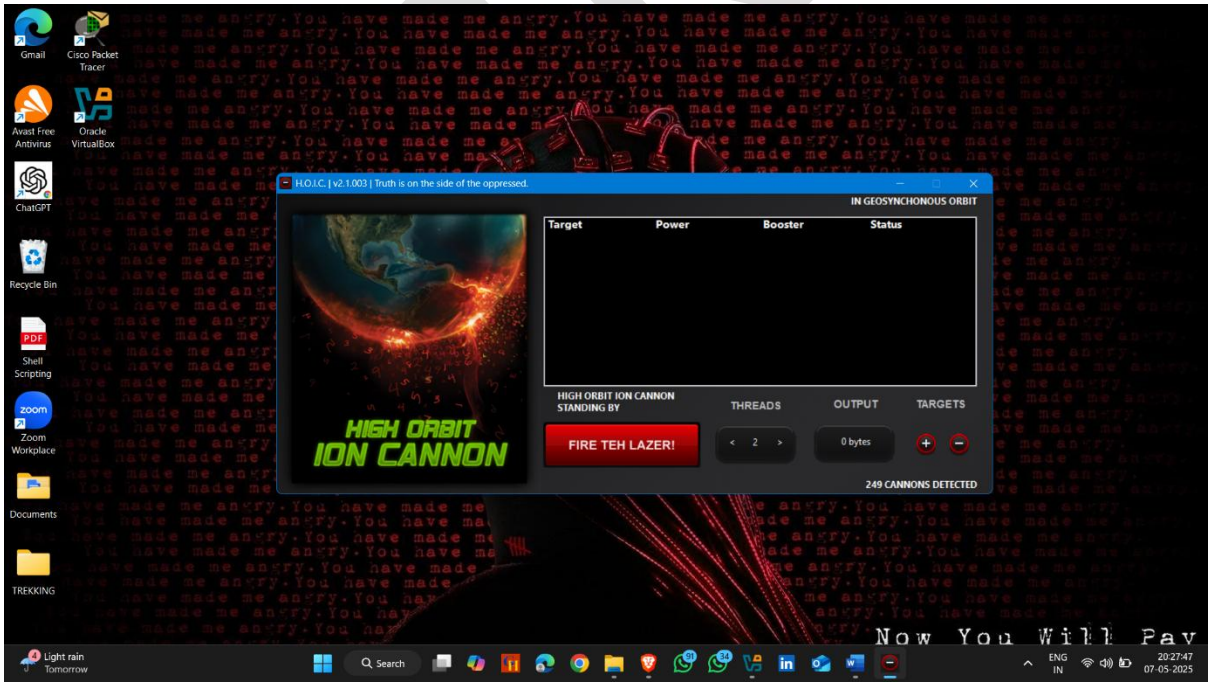
HOIC is popular in the **hacktivist** community and has been used in large-scale **DDoS** (Distributed Denial of Service) attacks.

Download Link -:
https://sourceforge.net/projects/highorbitioncannon/

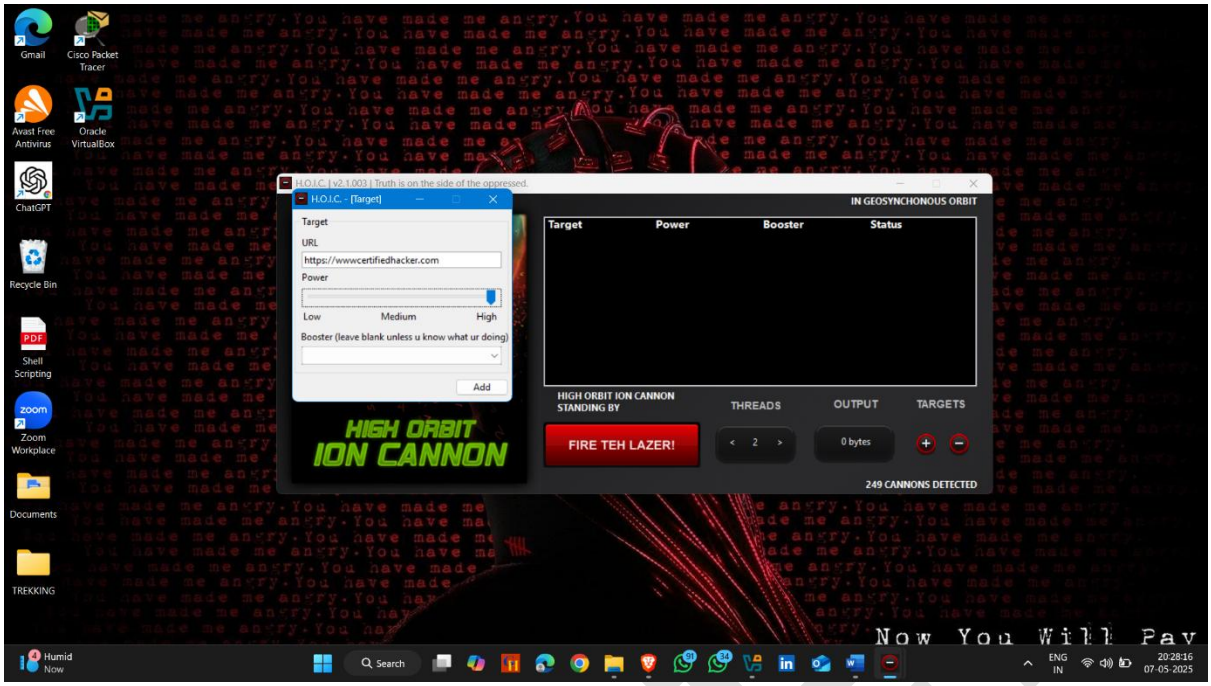


How to use it - :

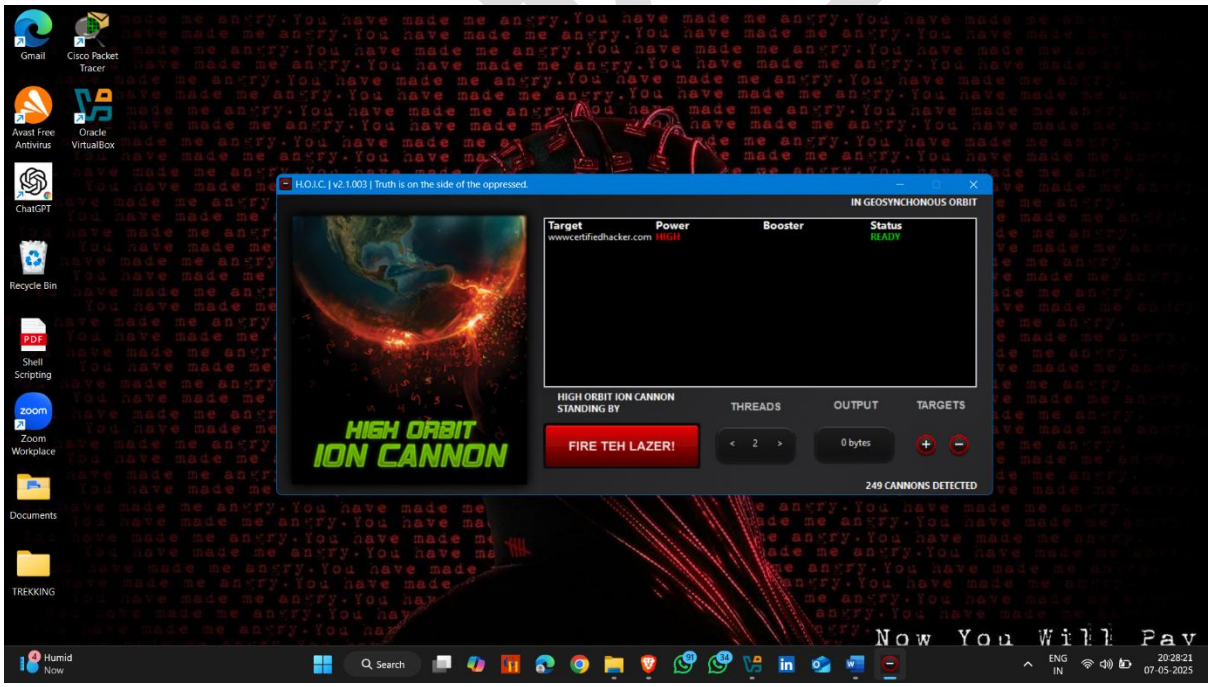
- After installation open the app



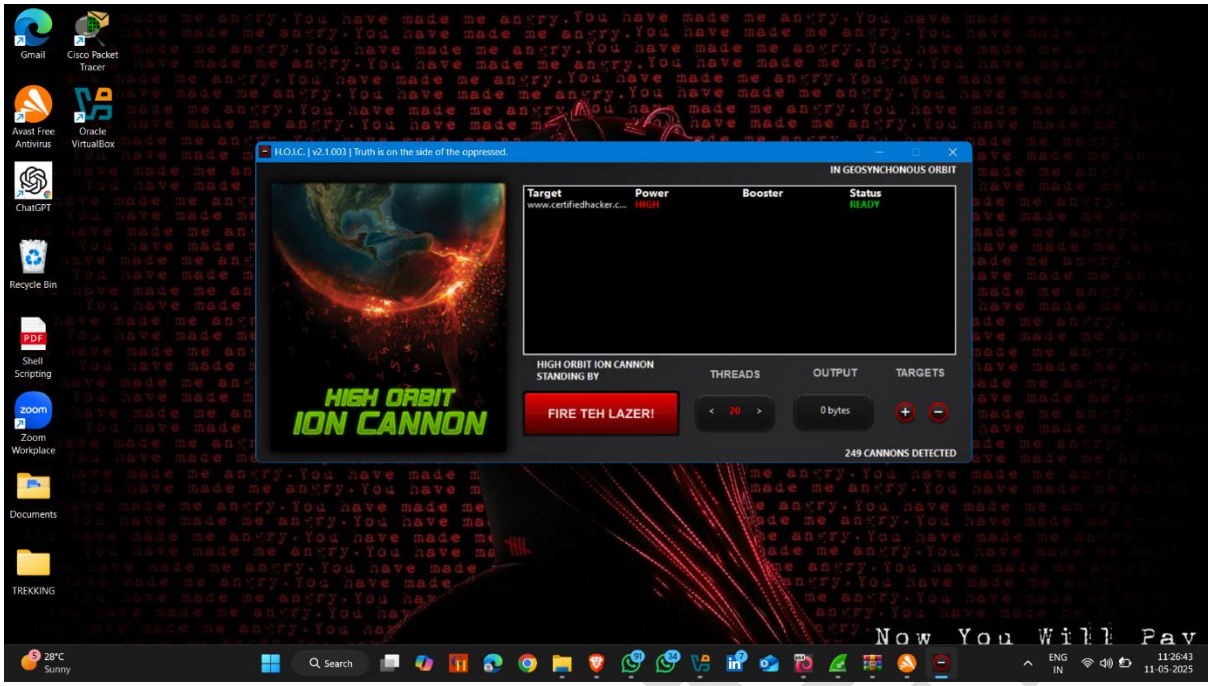
- Click on plus “ ➕ ” button and add target url
- Click on add



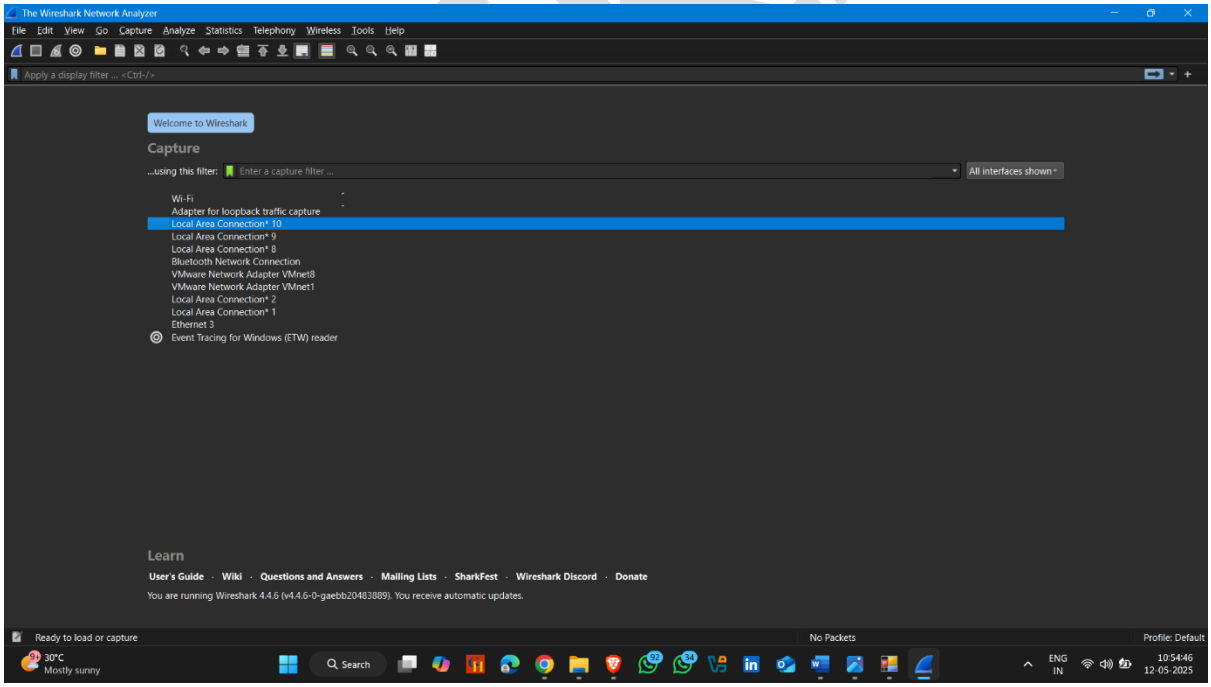
- Now adjust threads



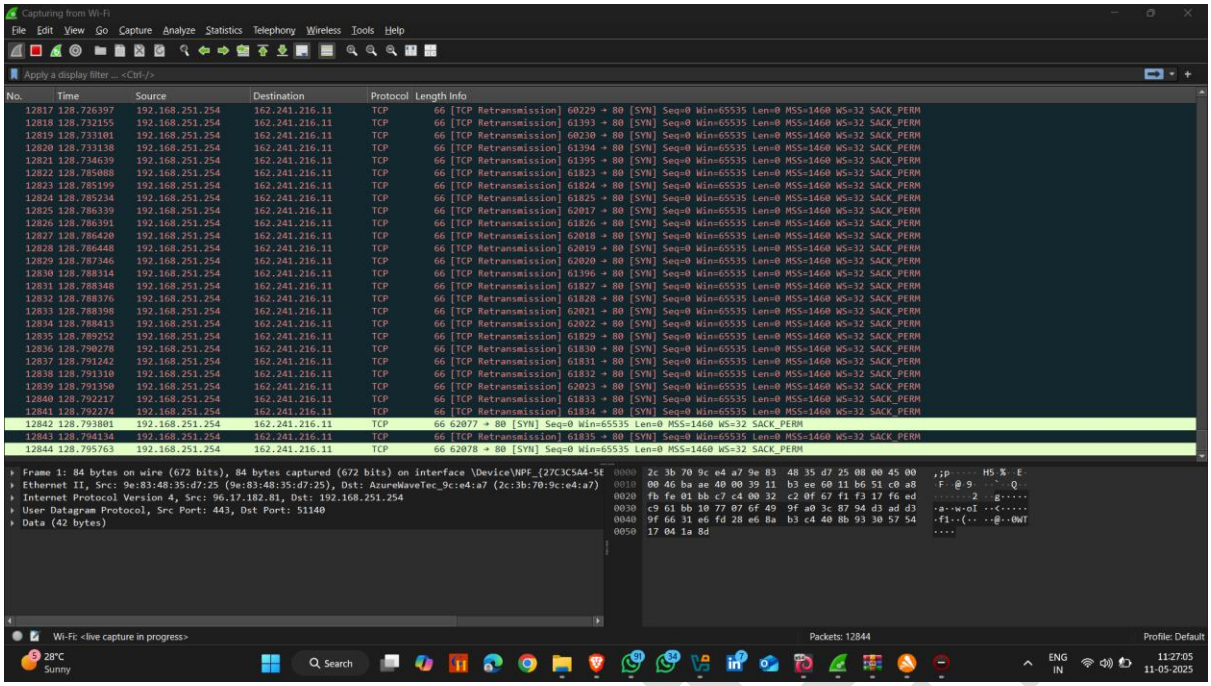
- Click on **FIRE THE LAZER !**



- Now open wireshark to monitor attack



- Attack started 👇

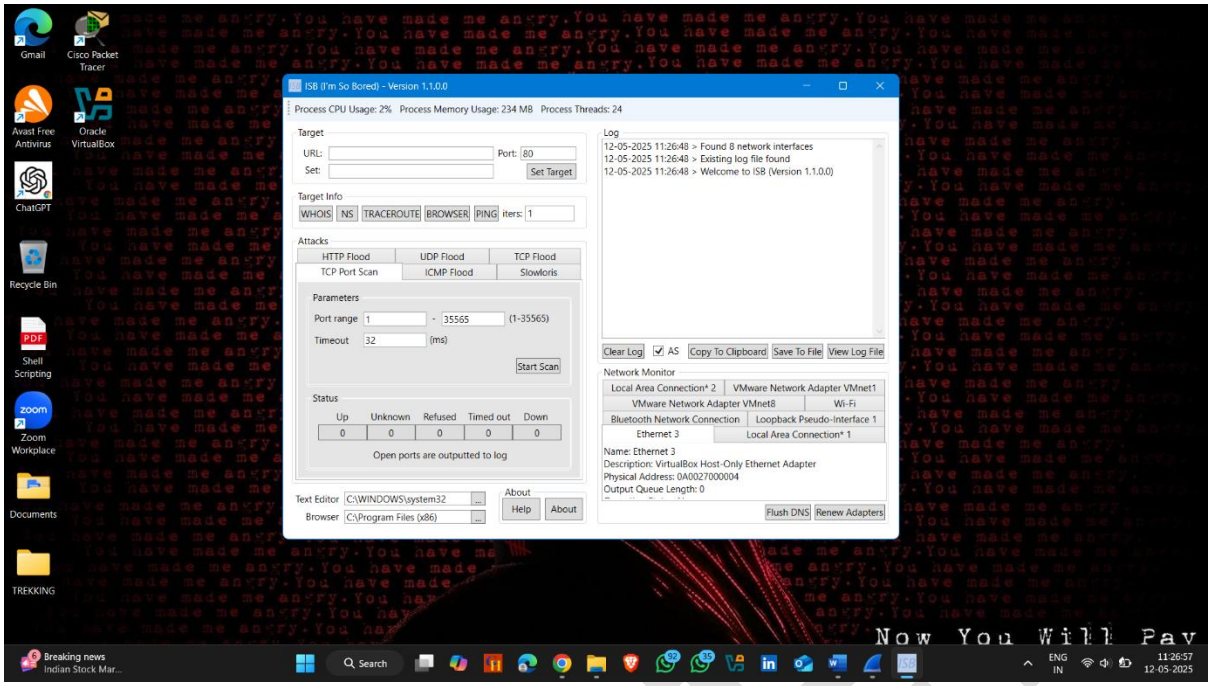


7.Perform DOS/DDOS Using ISB

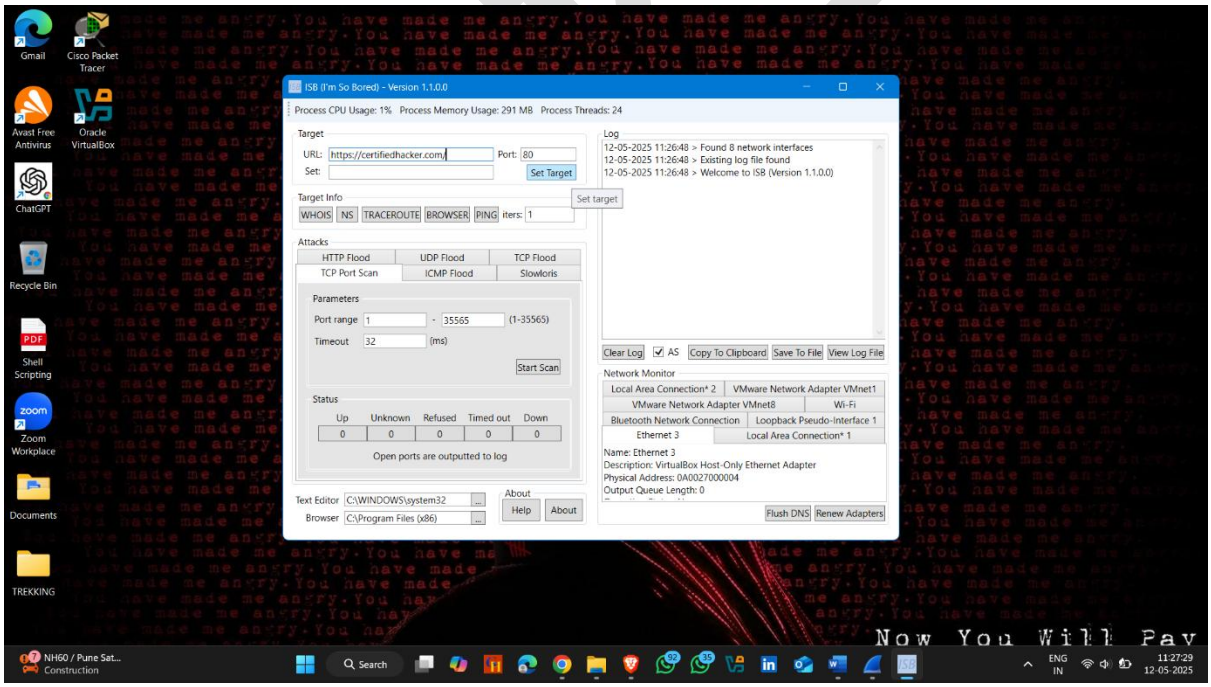
ISB, short for **"I'm So Bored"**, is an open-source **network stress-testing application** for Windows, developed by byte[size] Software. It's designed to simulate various types of network traffic to assess the resilience and performance of servers and network infrastructures under heavy load conditions.

How to use it -:

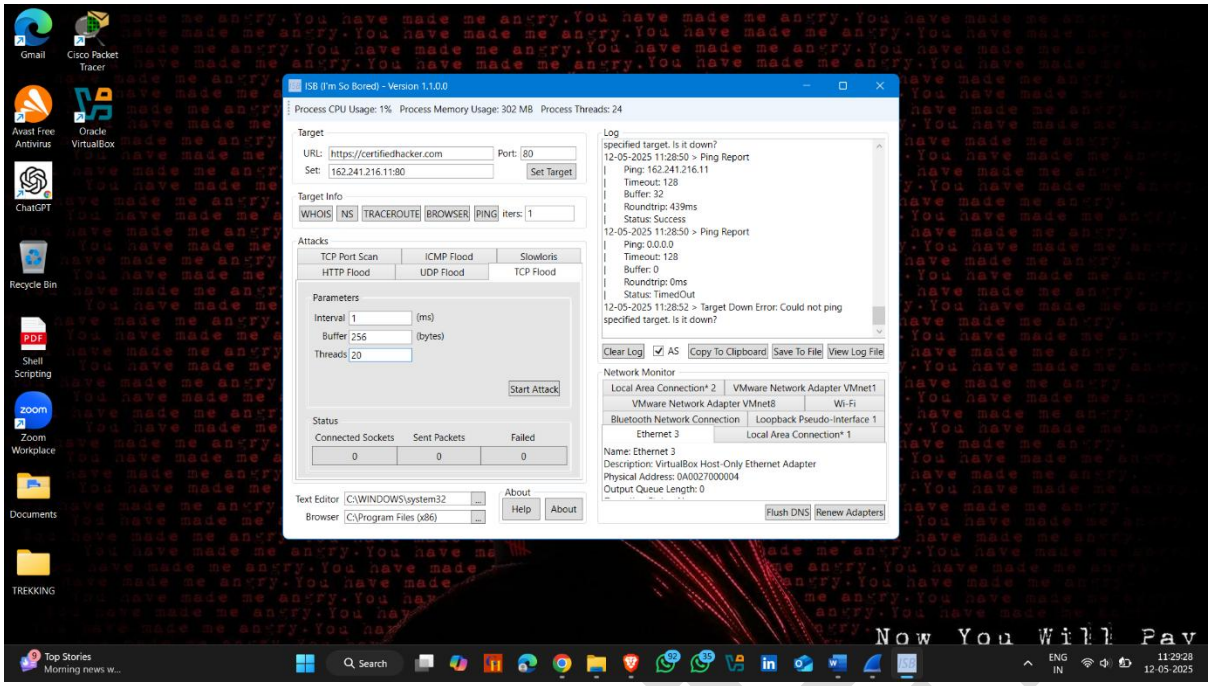
- Open ISB application



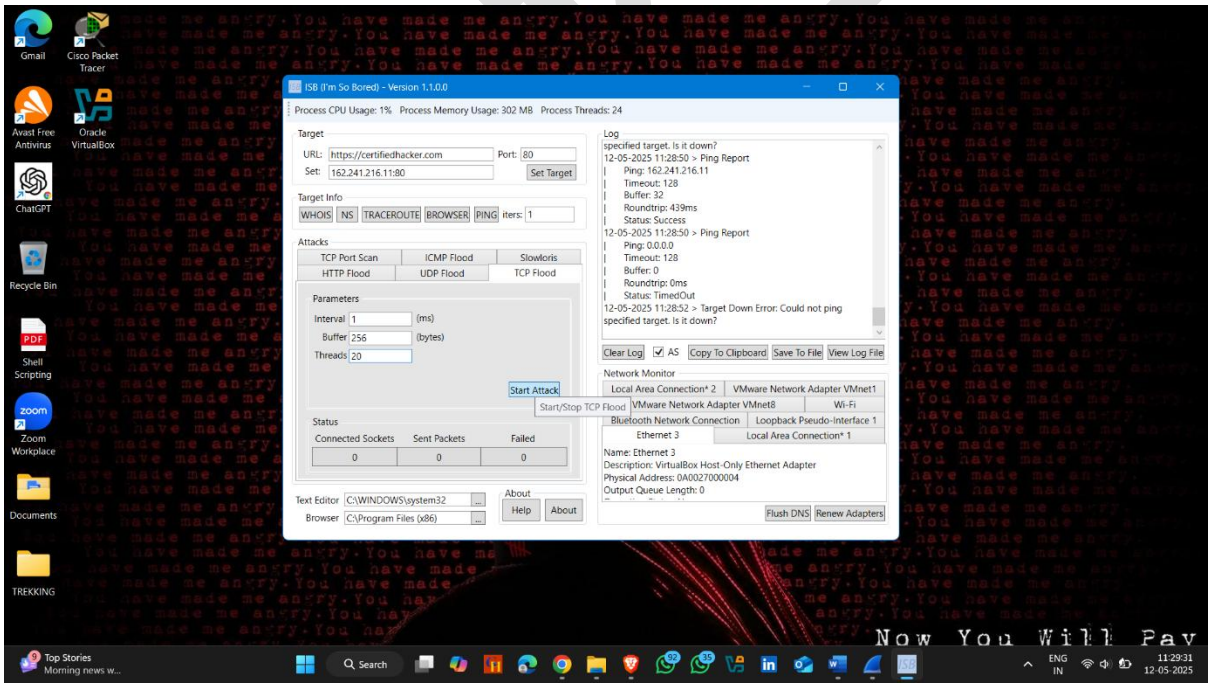
- Provide target url and click on set target



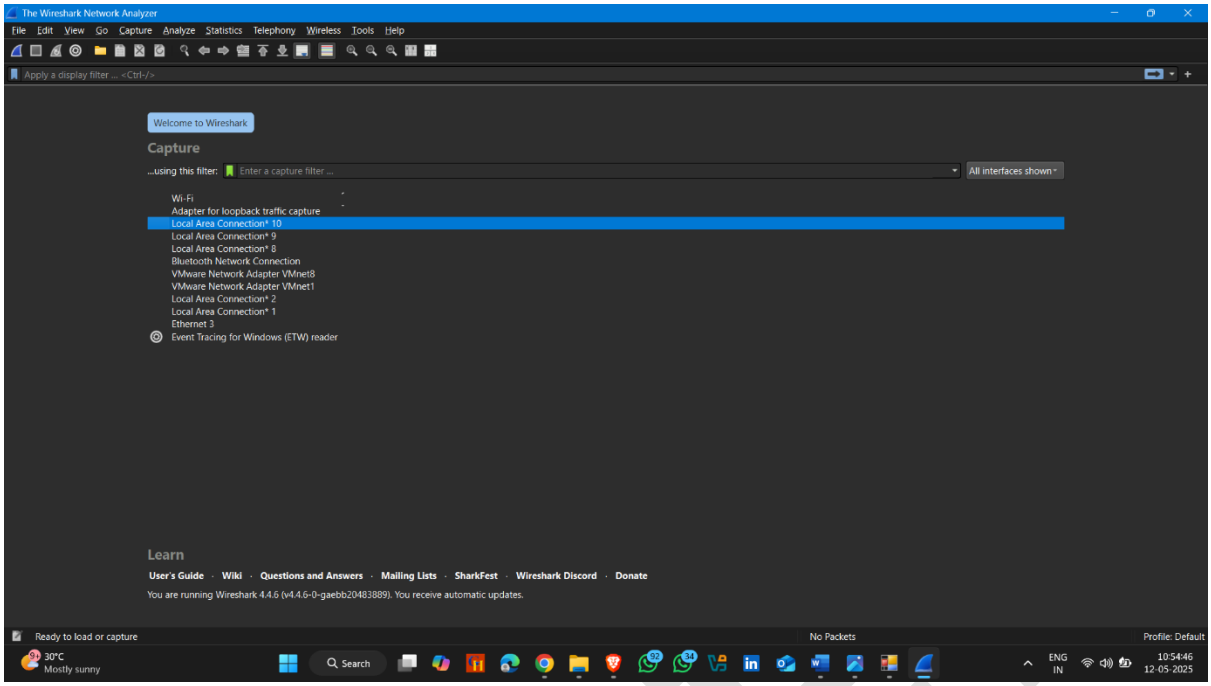
- Adjust threads



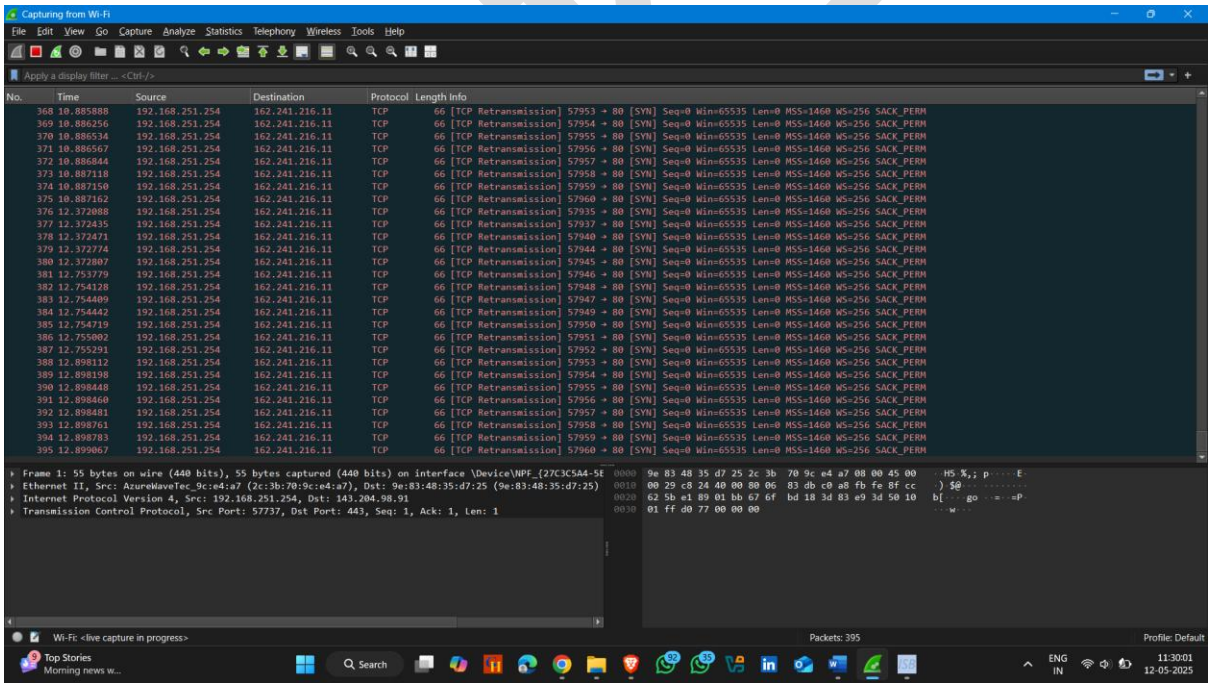
- Click on start attack



- Now open wiresharl to monitor attack



- Attack Started



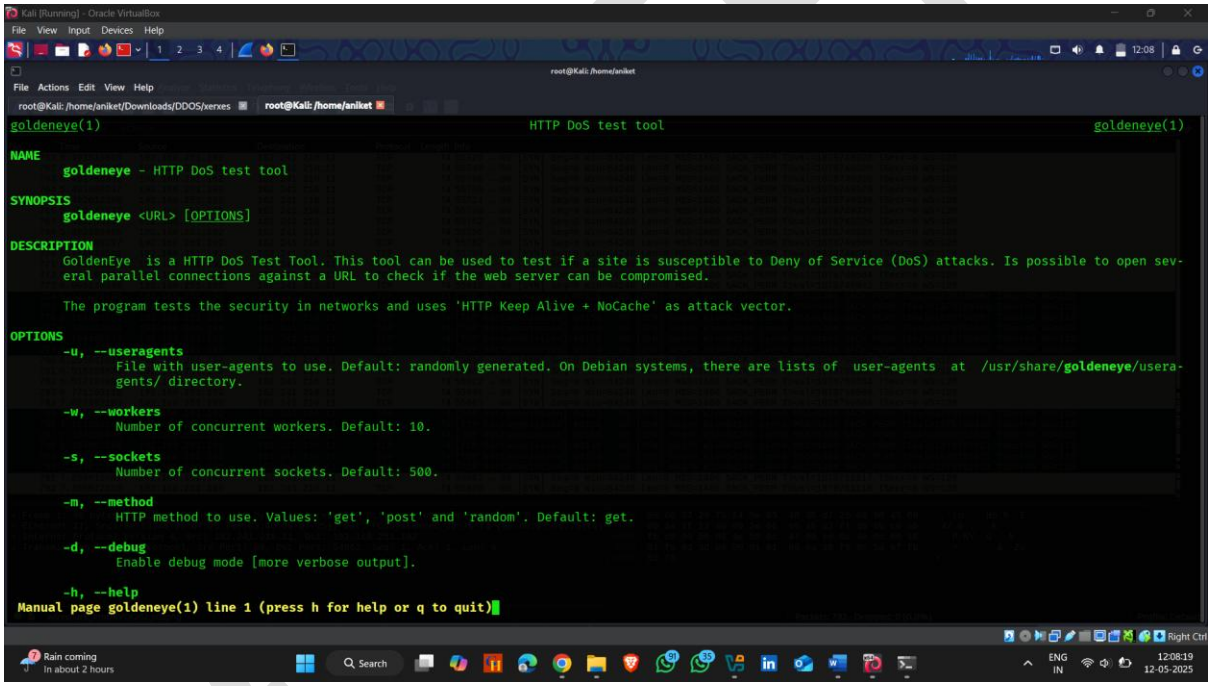
8.Perform DOS/DDOS Using Goldeneye

GoldenEye is a **Layer 7 (Application Layer) DoS testing tool** written in Python, designed to simulate **HTTP-based Denial of Service attacks** on web servers.

It's commonly used in **penetration testing labs** to test how a web server responds to large numbers of simultaneous HTTP requests.

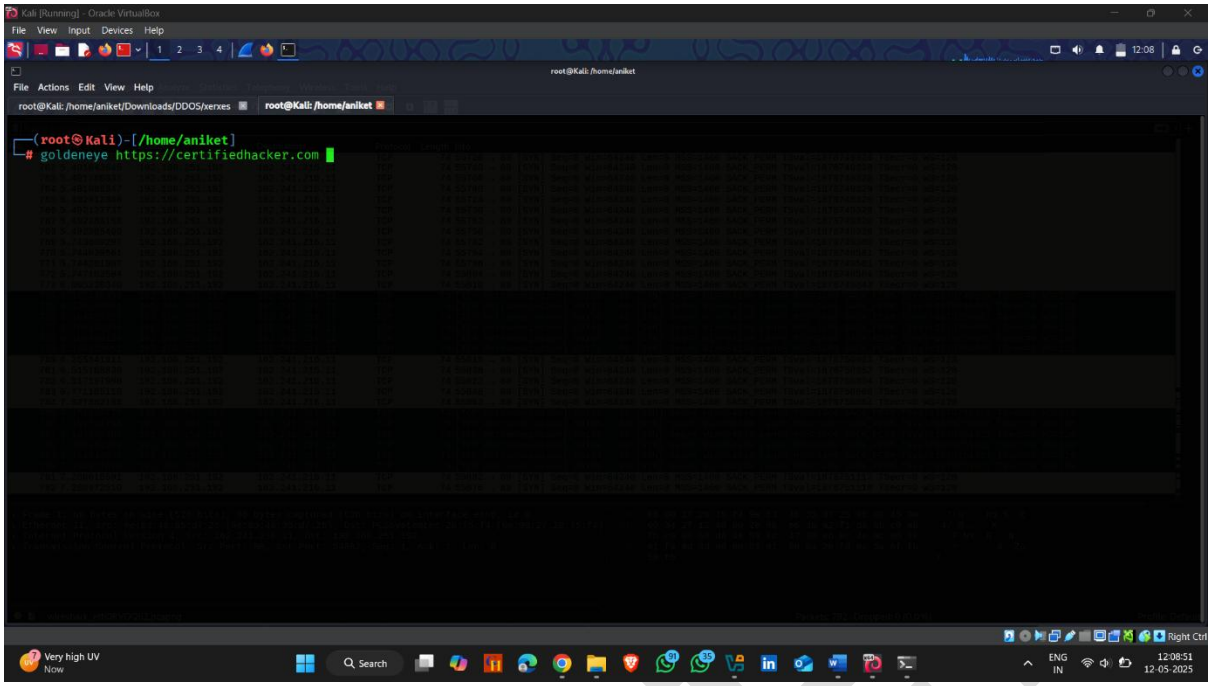
How to use it -:

- Open kali linux terminal and type man goldeneye – to get manual page about golden eye

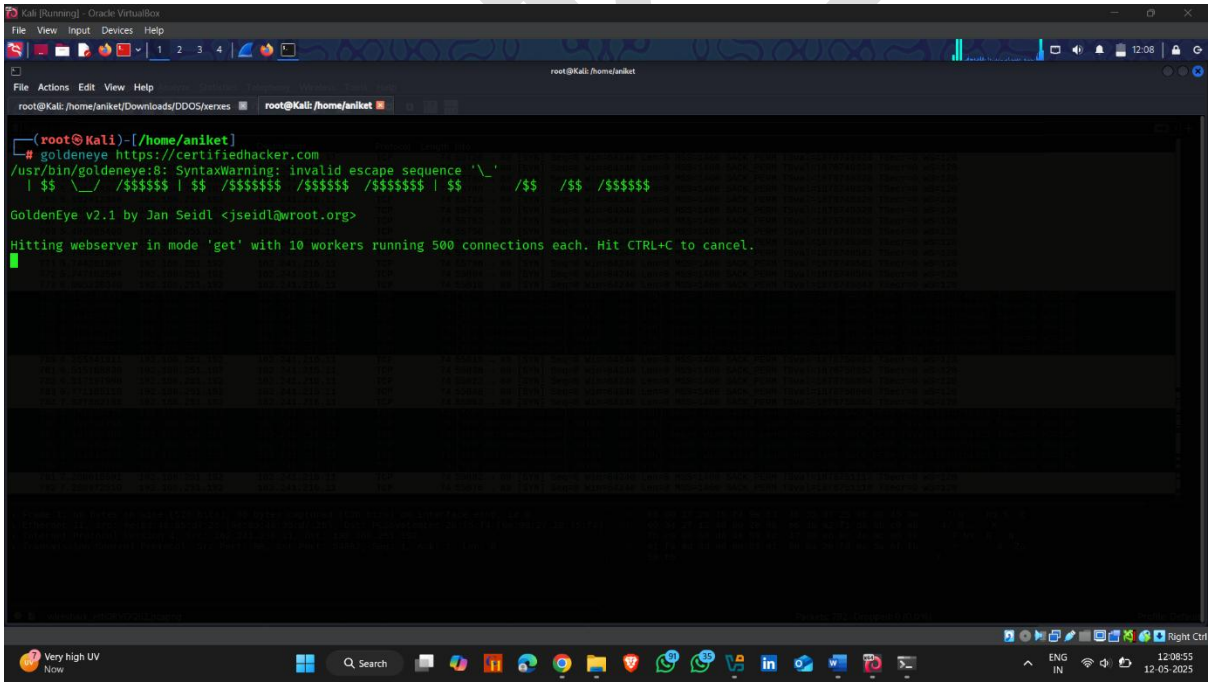


- now start attack using goldeneye

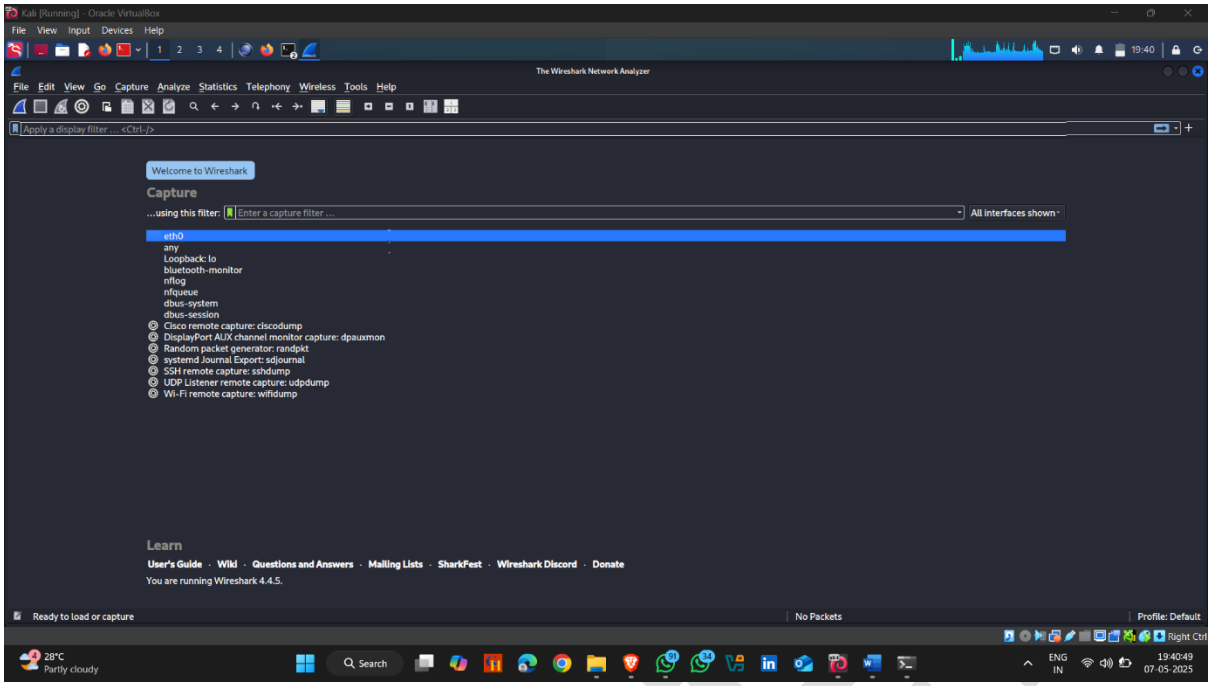
Command – goldeneye <target url >



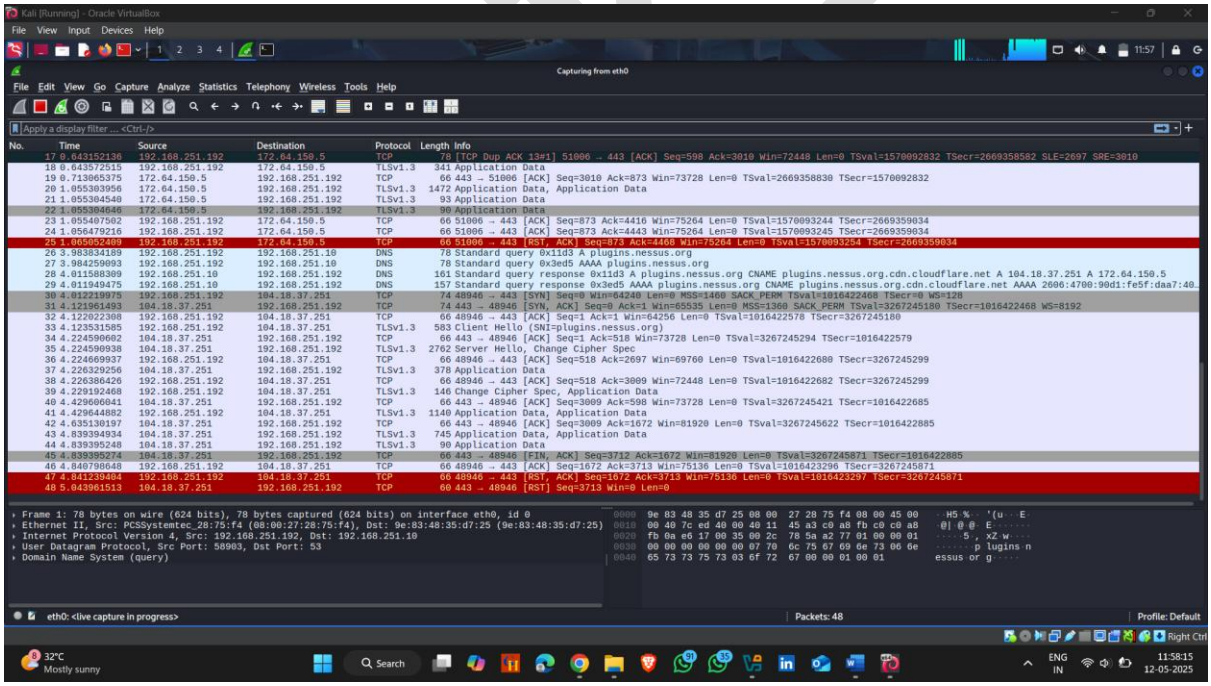
- attack started



- Now open wireshark
- Select eth0 network interface



- Attack started 👇



9.Perform DOS/DDOS Using Ping Of Death Attack

The **Ping of Death (PoD)** is a type of **Denial of Service (DoS) attack** where the attacker sends **malicious, oversized ICMP (ping) packets** to a target system, causing it to **crash, freeze, or reboot**.

🧠 **How Ping of Death Works**

- The standard **ICMP Echo Request (ping)** packet is **up to 65,535 bytes** (according to IP specification).
- But most systems **can't handle a full-size packet all at once**, especially older systems.
- The attacker **sends a ping packet larger than the system can process** by fragmenting it into small parts.
- When the system tries to **reassemble** the fragments into one big packet, it **overflows the memory buffer**, causing:
    - System crashes
    - Freezes
    - Reboots
    - Blue screens (on Windows)

How to use it -:

- Open command promt (cmd) and type command

Command –: ping -l 100 certifiedhacker.com -t

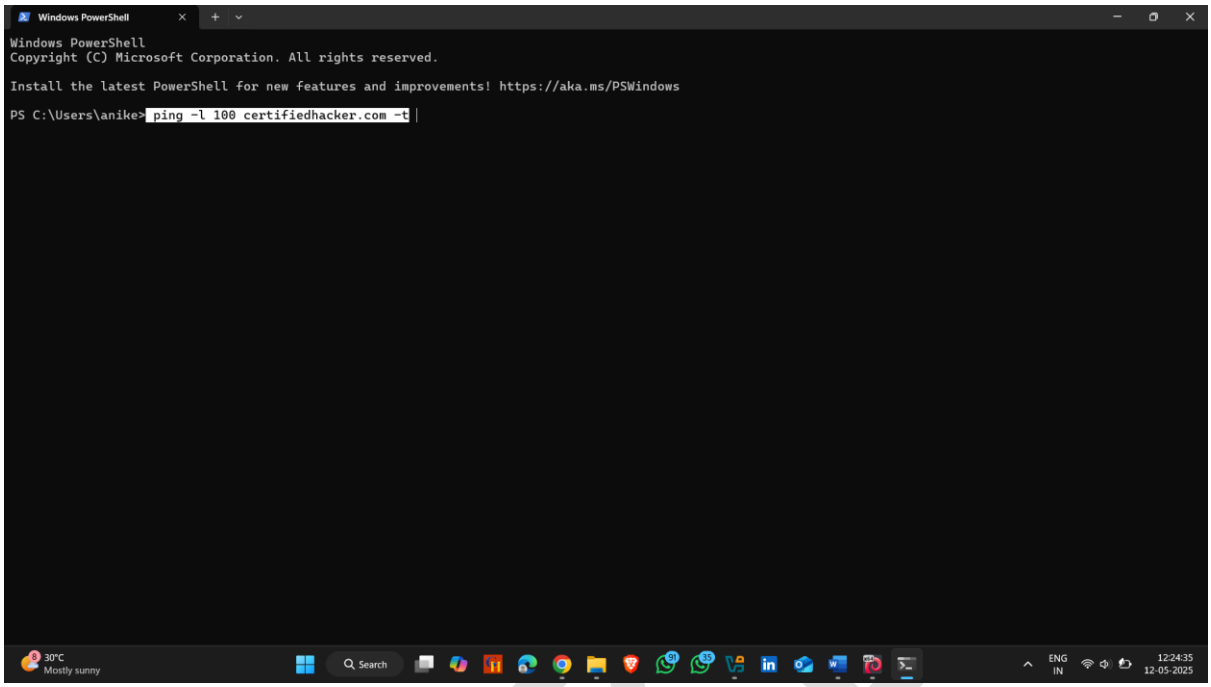
Explaination -:

Ping -: Sends ICMP Echo Request messages to a host

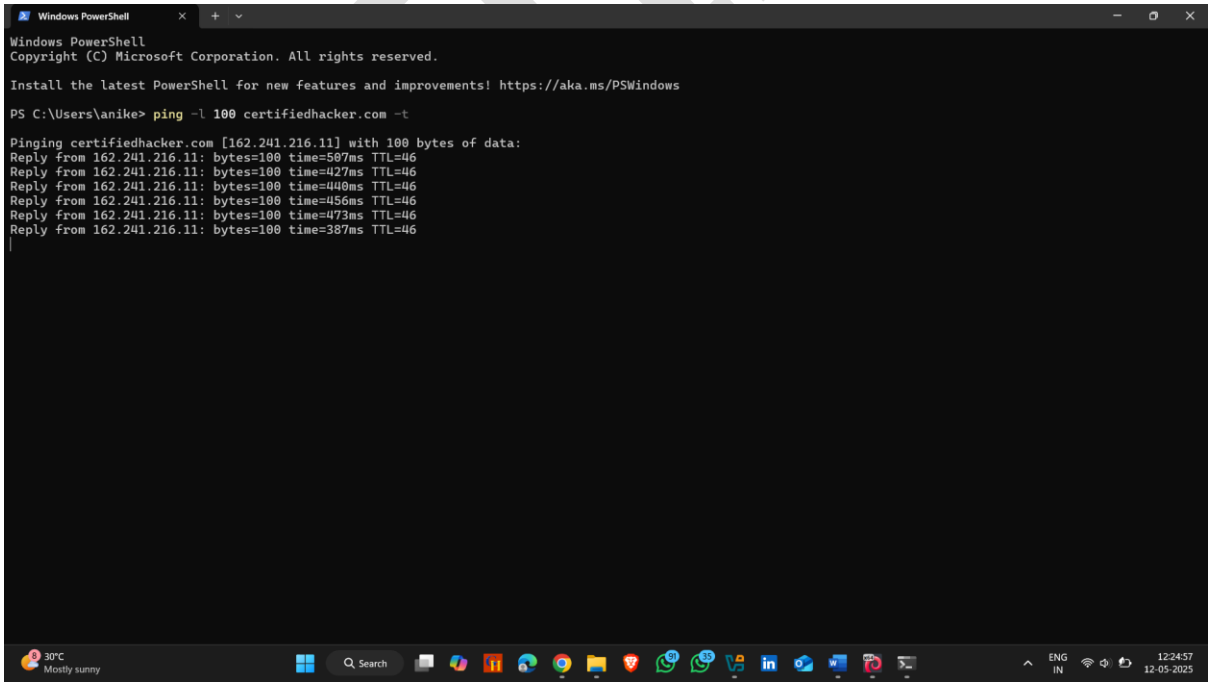
-l 100 -: Sets the size of the ping packet to 100 bytes

Certifiedhacker.com -: The Target Domain

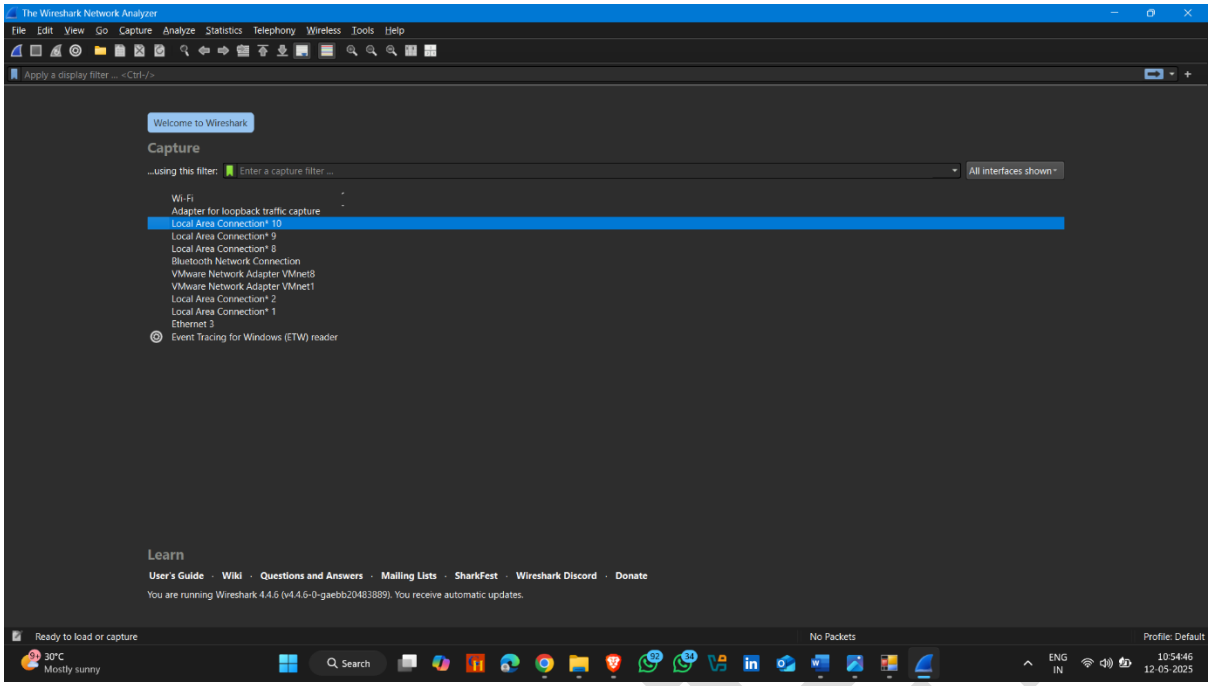
-t -: Makes the ping run continuously until manually stopped with ctrl+C



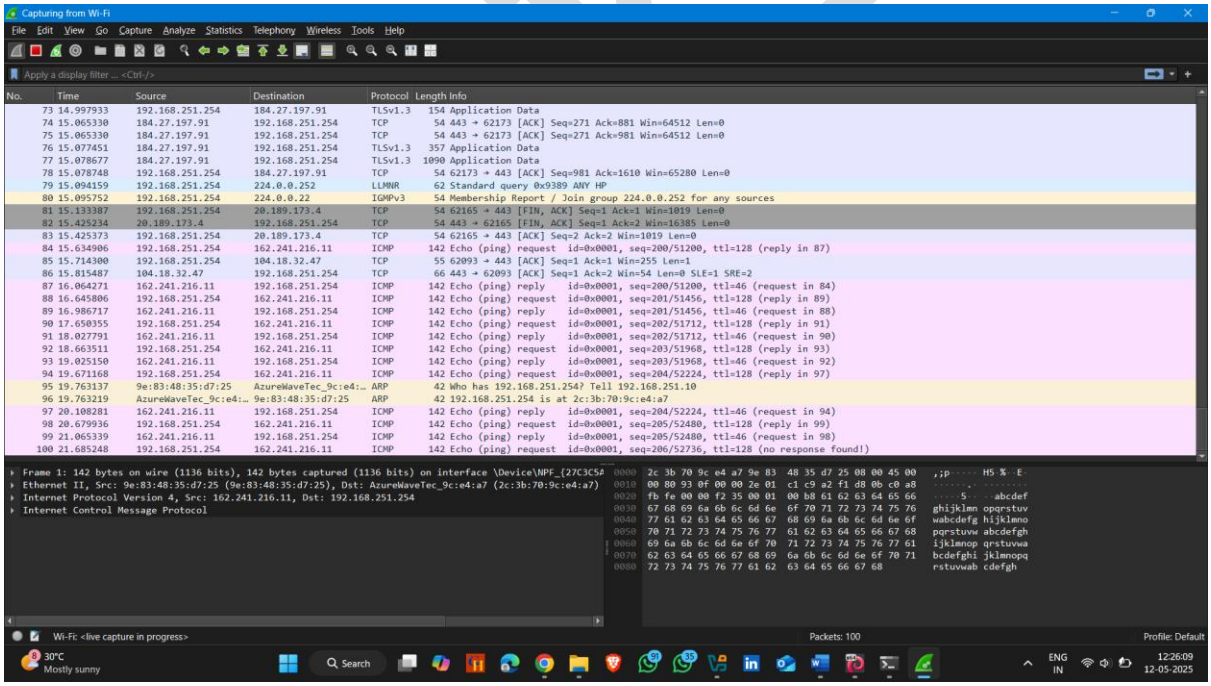
- Attack Started



- Open Wireshark


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark
Capture
...using this filter: Enter a capture filter ...
All interfaces shown
Wi-Fi
Adapter for loopback traffic capture
Local Area Connection* 10
Local Area Connection* 9
Local Area Connection* 8
Bluetooth Network Connection
VMware Network Adapter VMnet8
VMware Network Adapter VMnet1
Local Area Connection* 2
Local Area Connection* 1
Ethernet 3
Event Tracing for Windows (ETW) reader
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate
You are running Wireshark 4.4.6 (v4.4.6-0-gaebb20483889). You receive automatic updates.
Ready to load or capture
No Packets
Profile: Default
30°C
Mostly sunny
Search
ENG
IN
10:54:46
12-05-2025

- Attack Done 👇


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
73 14.997933 192.168.251.254 184.27.197.91 TLSv1.3 154 Application Data
74 15.065330 184.27.197.91 192.168.251.254 TCP 54 443 → 62173 [ACK] Seq=271 Ack=881 Win=64512 Len=0
75 15.065330 184.27.197.91 192.168.251.254 TCP 54 443 → 62173 [ACK] Seq=271 Ack=981 Win=64512 Len=0
76 15.077451 184.27.197.91 192.168.251.254 TLSv1.3 357 Application Data
77 15.078677 184.27.197.91 192.168.251.254 TLSv1.3 1090 Application Data
78 15.078748 192.168.251.254 184.27.197.91 TCP 54 62173 → 443 [ACK] Seq=981 Ack=1610 Win=65280 Len=0
79 15.094159 192.168.251.254 224.0.0.252 LLMNR 62 Standard query 0x9389 ANY HP
80 15.095752 192.168.251.254 224.0.0.22 IGMPv3 54 Membership Report / Join group 224.0.0.252 for any sources
81 15.333387 192.168.251.254 20.189.173.4 TCP 54 62165 → 443 [FIN, ACK] Seq=1 Ack=1 Win=1019 Len=0
82 15.425234 20.189.173.4 192.168.251.254 TCP 54 443 → 62165 [FIN, ACK] Seq=1 Ack=2 Win=16385 Len=0
83 15.425373 192.168.251.254 20.189.173.4 TCP 54 62165 → 443 [ACK] Seq=2 Ack=2 Win=1019 Len=0
84 15.634906 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=200/51200, ttl=128 (reply in 87)
85 15.714300 192.168.251.254 104.18.32.47 TCP 55 62093 → 443 [ACK] Seq=1 Ack=1 Win=255 Len=1
86 15.815487 104.18.32.47 192.168.251.254 TCP 66 443 → 62093 [ACK] Seq=1 Ack=2 Win=54 Len=0 SLE=1 SRE=2
87 16.064271 162.241.216.11 192.168.251.254 ICMP 142 Echo (ping) reply id=0x0001, seq=200/51200, ttl=46 (request in 84)
88 16.645806 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=201/51456, ttl=128 (reply in 89)
89 16.986717 162.241.216.11 192.168.251.254 ICMP 142 Echo (ping) reply id=0x0001, seq=201/51456, ttl=46 (request in 88)
90 17.650355 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=202/51712, ttl=128 (reply in 91)
91 18.027791 162.241.216.11 192.168.251.254 ICMP 142 Echo (ping) reply id=0x0001, seq=202/51712, ttl=46 (request in 90)
92 18.663511 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=203/51968, ttl=128 (reply in 93)
93 19.025150 162.241.216.11 192.168.251.254 ICMP 142 Echo (ping) reply id=0x0001, seq=203/51968, ttl=46 (request in 92)
94 19.671168 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=204/52224, ttl=128 (reply in 97)
95 19.763137 9e:83:48:35:d7:25 AzureWaveTec_9c:e4:... ARP 42 Who has 192.168.251.254? Tell 192.168.251.10
96 19.763219 AzureWaveTec_9c:e4:... 9e:83:48:35:d7:25 ARP 42 192.168.251.254 is at 2c:3b:70:9c:e4:a7
97 20.108281 162.241.216.11 192.168.251.254 ICMP 142 Echo (ping) reply id=0x0001, seq=204/52224, ttl=46 (request in 94)
98 20.679936 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=205/52480, ttl=128 (reply in 99)
99 21.065339 162.241.216.11 192.168.251.254 ICMP 142 Echo (ping) reply id=0x0001, seq=205/52480, ttl=46 (request in 98)
100 21.685248 192.168.251.254 162.241.216.11 ICMP 142 Echo (ping) request id=0x0001, seq=206/52736, ttl=128 (no response found!)
Frame 1: 142 bytes on wire (1136 bits), 142 bytes captured (1136 bits) on interface \Device\NPF_{27C3C5A
Ethernet II, Src: 9e:83:48:35:d7:25 (9e:83:48:35:d7:25), Dst: AzureWaveTec_9c:e4:a7 (2c:3b:70:9c:e4:a7)
Internet Protocol Version 4, Src: 162.241.216.11, Dst: 192.168.251.254
Internet Control Message Protocol
0000 2c 3b 70 9c e4 a7 9e 83 48 35 d7 25 08 00 45 00 ,;p..... H5.%..E.
0010 00 80 93 0f 00 00 2e 01 c1 c9 a2 f1 d8 0b c0 a8 ....
0020 fb fe 00 00 f2 35 00 01 00 b8 61 62 63 64 65 66 .....5.. ..abcdef
0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 ghijklmn opqrstuv
0040 77 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f wabcdefg hijklmno
0050 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 pqrstuvw abcdefgh
0060 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 61 ijklmnop qrstuvwa
0070 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 bcdefghi jklmnopq
0080 72 73 74 75 76 77 61 62 63 64 65 66 67 68 rstuvwab cdefgh
Wi-Fi: <live capture in progress>
Packets: 100
Profile: Default
30°C
Mostly sunny
Search
ENG
IN
12:26:09
12-05-2025

EXTRA ACTIVITY

1.Perform DOS/DDOS Using Macof

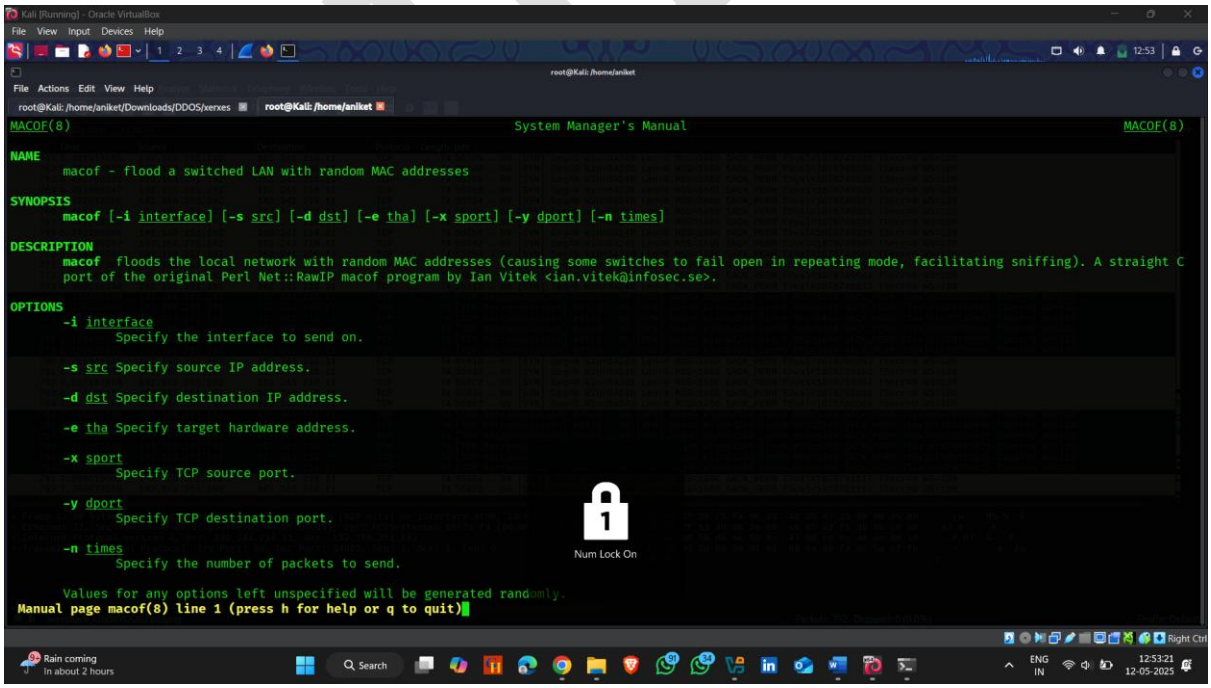
The **macof** tool in Kali Linux is a **network attack tool** that's part of the **dsniff suite**, and it's used to **flood a network switch's CAM table** (Content Addressable Memory) with fake MAC addresses.

🧠 Purpose of macof

macof generates a **huge number of fake MAC addresses** with random IP/MAC combinations and sends them over the network.

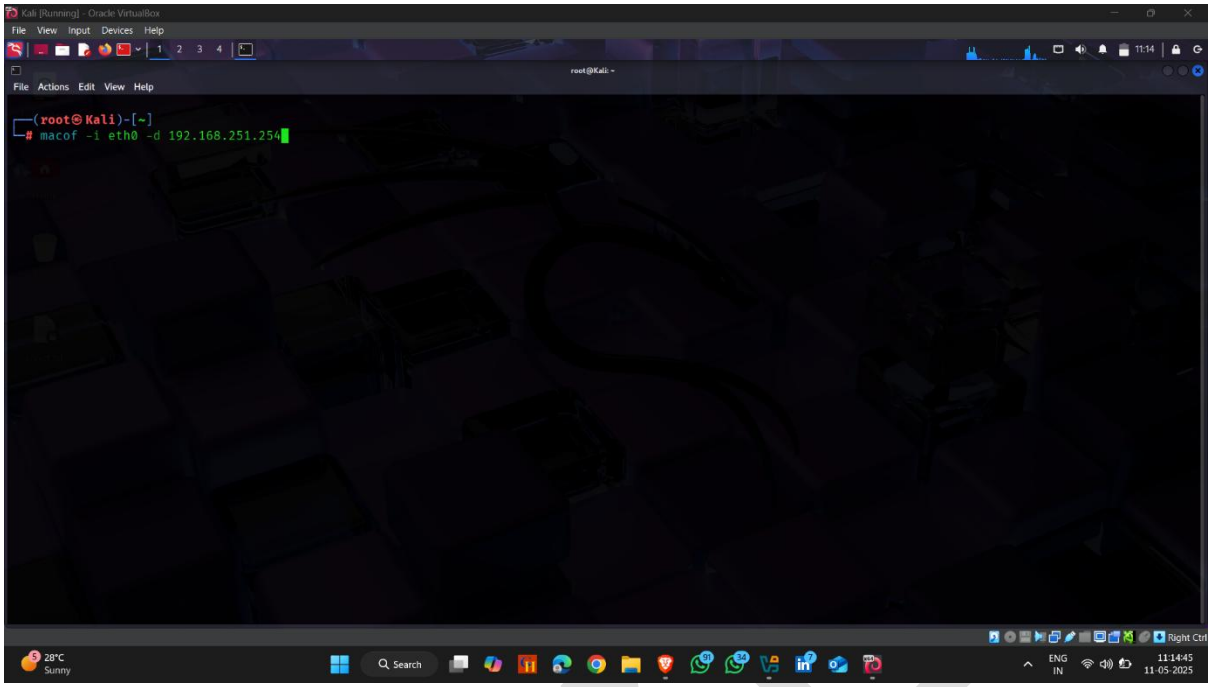
How to use it -:

- Open kali linux terminal and type man macof – to get manual page of macof

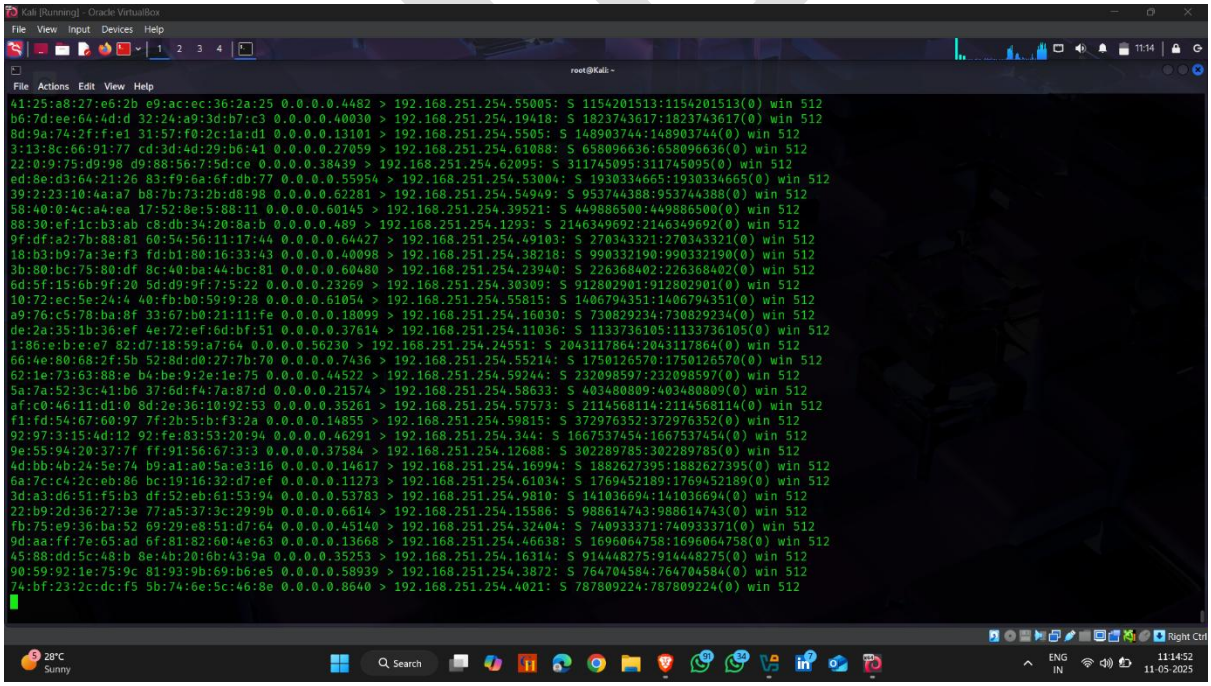


- Now start attack using macof

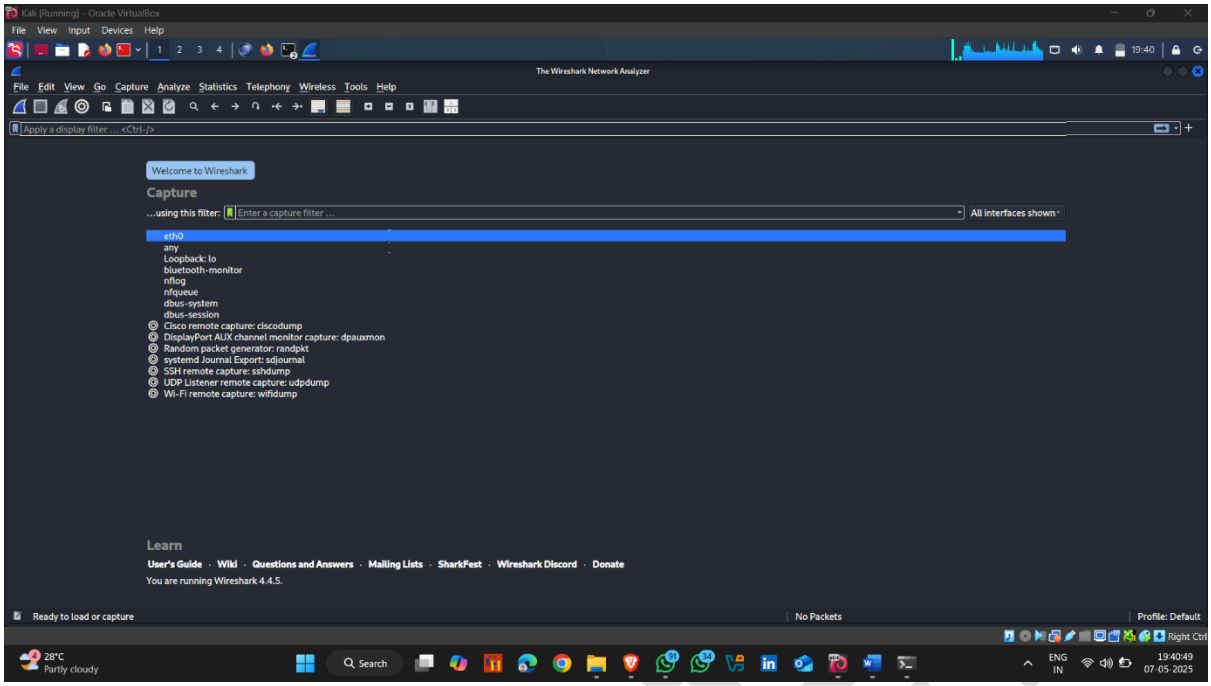
Command -: macof -I eth -d <target ip>

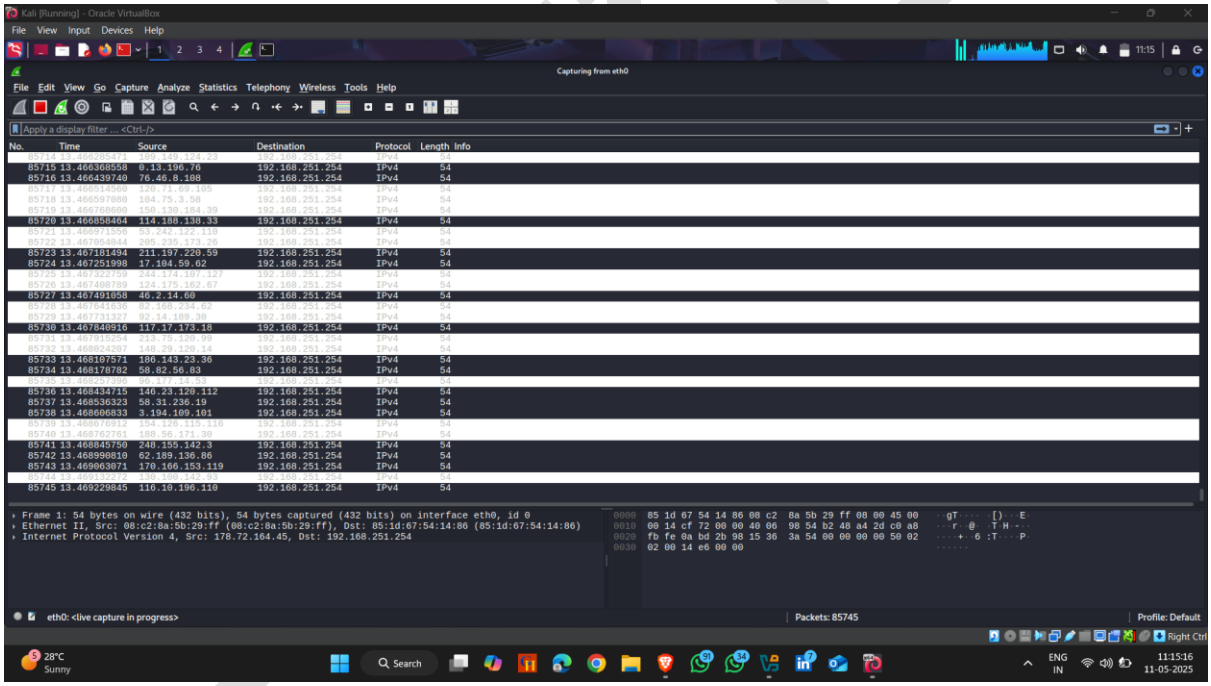


- Attack started 👇



- Open wireshark to see packets
- Select network interface


Kali [Running] - Oracle VirtualBox
File View Input Devices Help
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark
Capture
...using this filter: Enter a capture filter ...
All interfaces shown
eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate
You are running Wireshark 4.4.5.
Ready to load or capture
No Packets
Profile: Default
28°C
Partly cloudy
Search
ENG
IN
19:40:49
07-05-2025


Kali [Running] - Oracle VirtualBox
File View Input Devices Help
Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
85715 13.466368558 0.13.196.76 192.168.251.254 IPv4 54
85716 13.466439740 76.46.8.108 192.168.251.254 IPv4 54
85720 13.466858464 114.188.138.33 192.168.251.254 IPv4 54
85723 13.467181494 211.197.220.59 192.168.251.254 IPv4 54
85724 13.467251998 17.104.59.62 192.168.251.254 IPv4 54
85727 13.467491058 46.2.14.60 192.168.251.254 IPv4 54
85730 13.467640916 117.17.173.18 192.168.251.254 IPv4 54
85733 13.468107571 186.143.23.36 192.168.251.254 IPv4 54
85734 13.468178782 58.82.56.83 192.168.251.254 IPv4 54
85736 13.468434715 146.23.120.112 192.168.251.254 IPv4 54
85737 13.468536323 58.31.236.19 192.168.251.254 IPv4 54
85738 13.468606833 3.194.109.101 192.168.251.254 IPv4 54
85741 13.468845750 248.155.142.3 192.168.251.254 IPv4 54
85742 13.468990810 62.189.136.86 192.168.251.254 IPv4 54
85743 13.469063071 170.166.153.119 192.168.251.254 IPv4 54
85745 13.469229845 116.10.196.110 192.168.251.254 IPv4 54
Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0
Ethernet II, Src: 00:c2:8a:5b:29:ff (00:c2:8a:5b:29:ff), Dst: 85:1d:67:54:14:86 (85:1d:67:54:14:86)
Internet Protocol Version 4, Src: 176.72.164.45, Dst: 192.168.251.254
eth0: <live capture in progress>
Packets: 85745
Profile: Default
28°C
Sunny
Search
ENG
IN
11:15:16
11-05-2025

2.Perform DOS/DDOS Using Xerxes

Xerxes is a **Layer 7 (HTTP-level) Denial of Service (DoS) attack tool** designed to **flood a web server with requests**, overloading its resources and making it unresponsive.

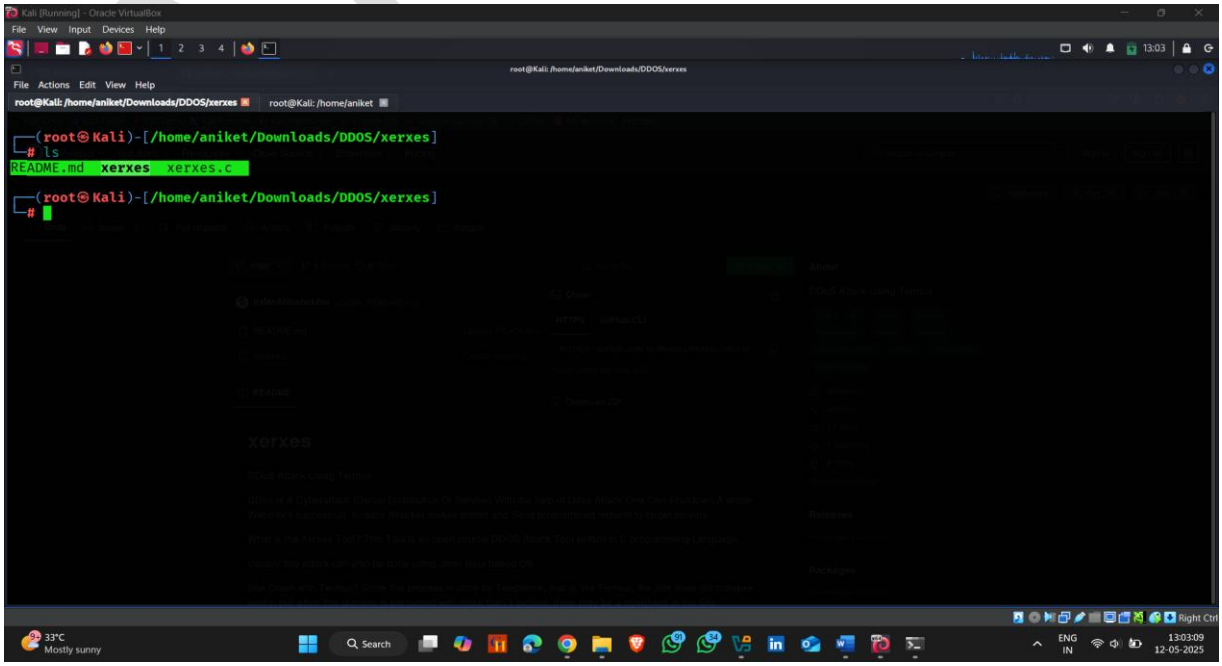
🧠 Key Features of Xerxes

- **Written in C**: Fast and lightweight.
- **Targets HTTP servers** (websites).
- Tries to **exhaust the target's connection capacity**.
- Works by keeping many **HTTP connections open simultaneously**.

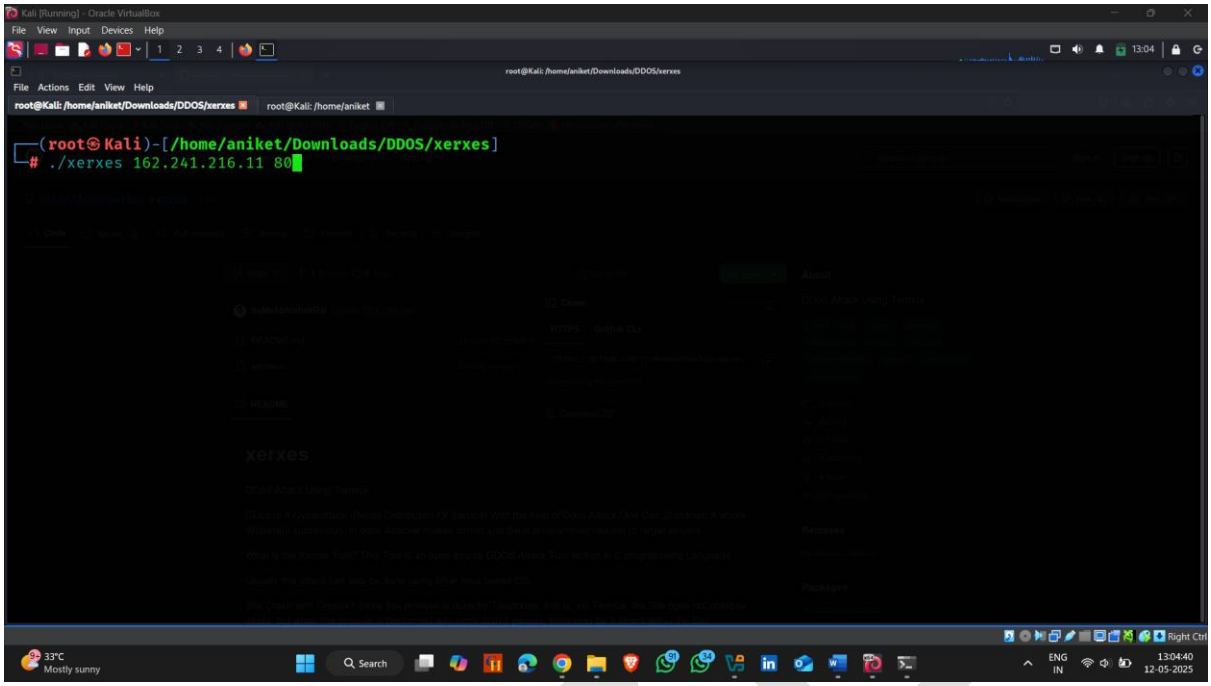
Download link-: https://github.com/ItsMeAbhishekRai/xerxes

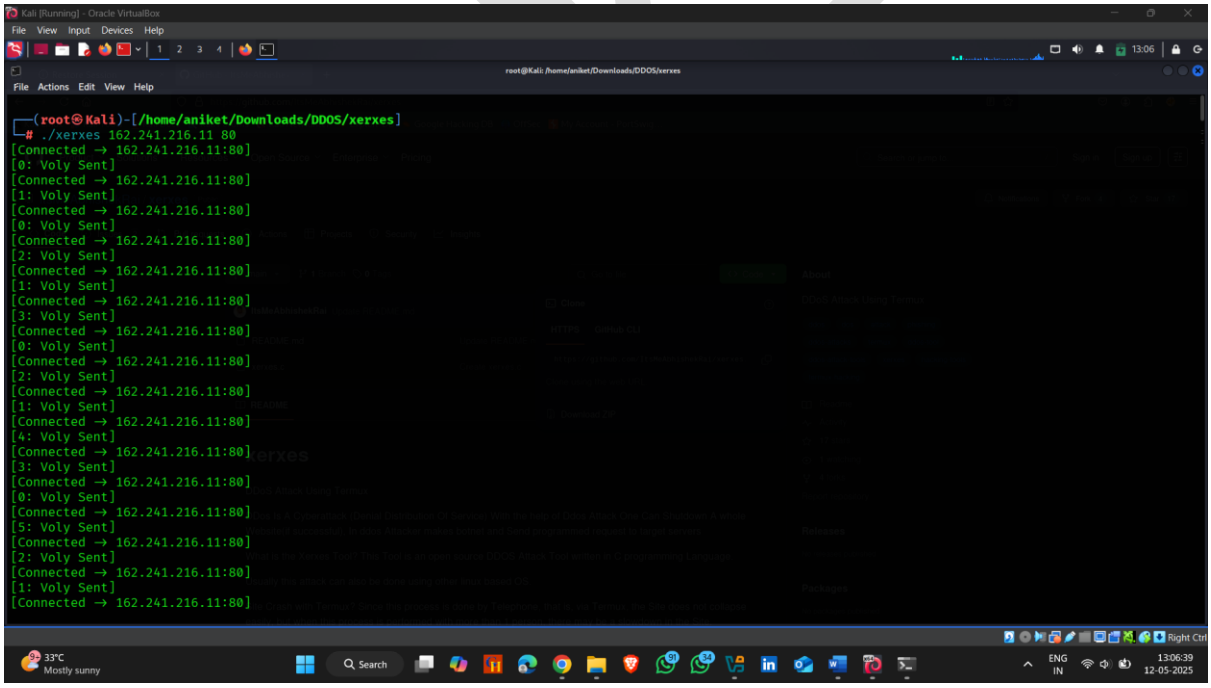
How to use it -:

- Downlaod tool from git hub
- Open kali linux terminal and go to the xerxes directory

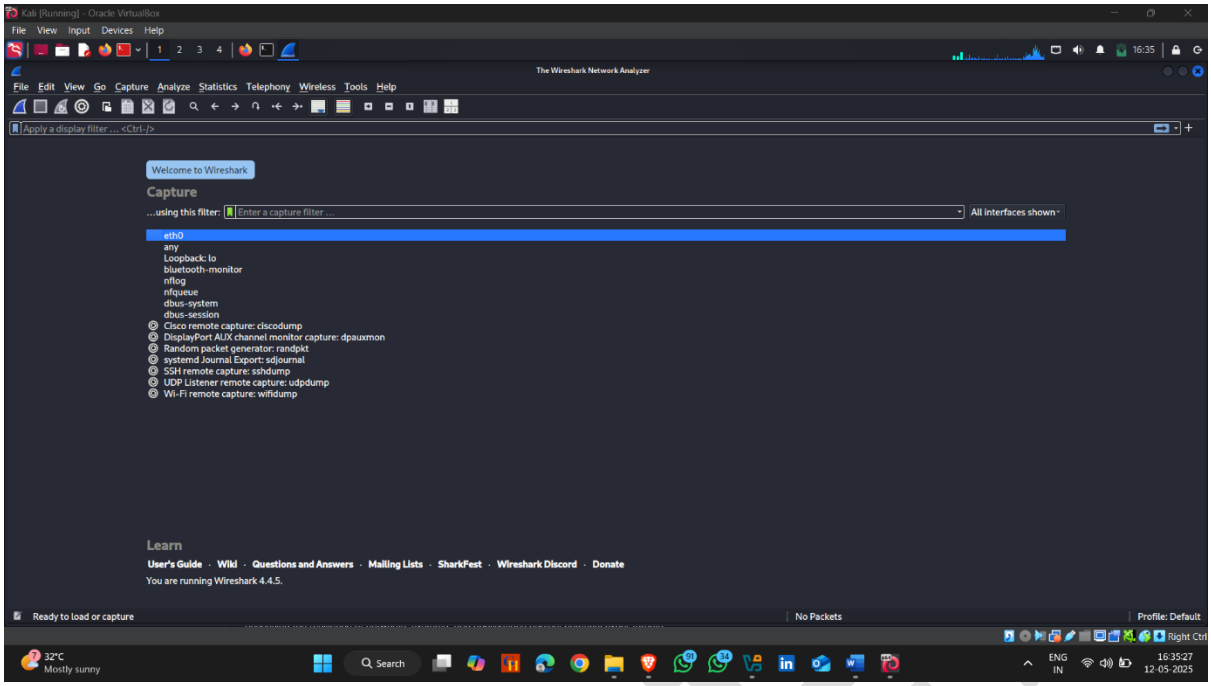


Command -: ./xerxes <target ip > <target port>

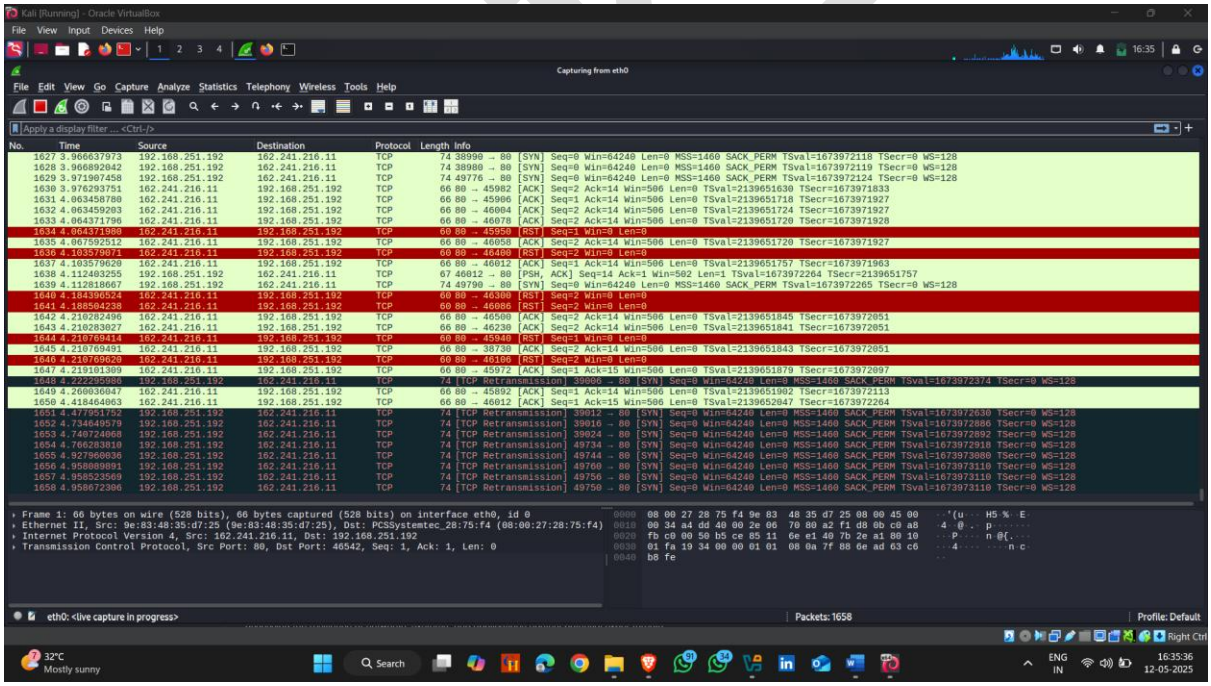




- Open Wireshark
- Select network interface



- Attack Started 👇



3.Perform DOS/DDOS Using Dosinator

Dosinator (also spelled **Dosinator**) is a **Python-based penetration testing tool** designed to simulate **DoS (Denial of Service)** and **DDoS (Distributed Denial of Service)** attacks. It's mainly used for **educational and testing purposes** in **cybersecurity training labs** and **VAPT environments**, not for illegal activity.

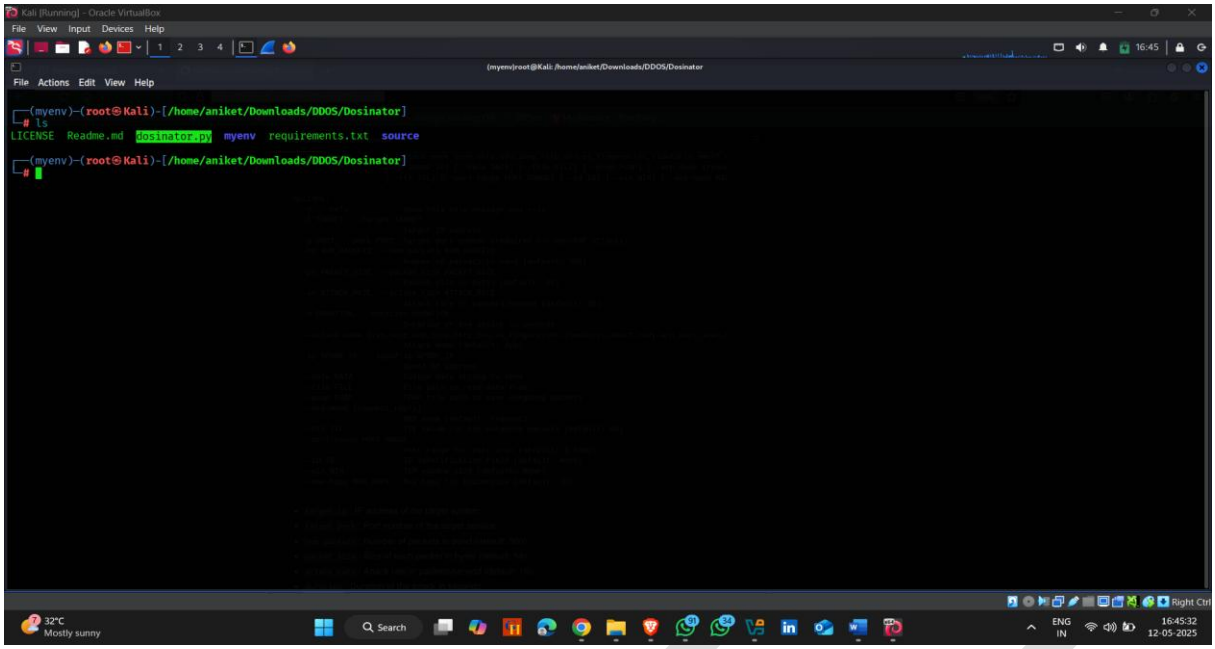
Key Features of D0sinator:

- **Multiple attack vectors**: Supports different types of flooding attacks like:
  - **UDP flood**
  - **TCP SYN flood**
  - **HTTP flood**

Download Link-: https://github.com/HalilDeniz/Dosinator

How to use it -:

- After installation Dosinator from git hub
- Open kali linux terminal and go to the dosinator directory

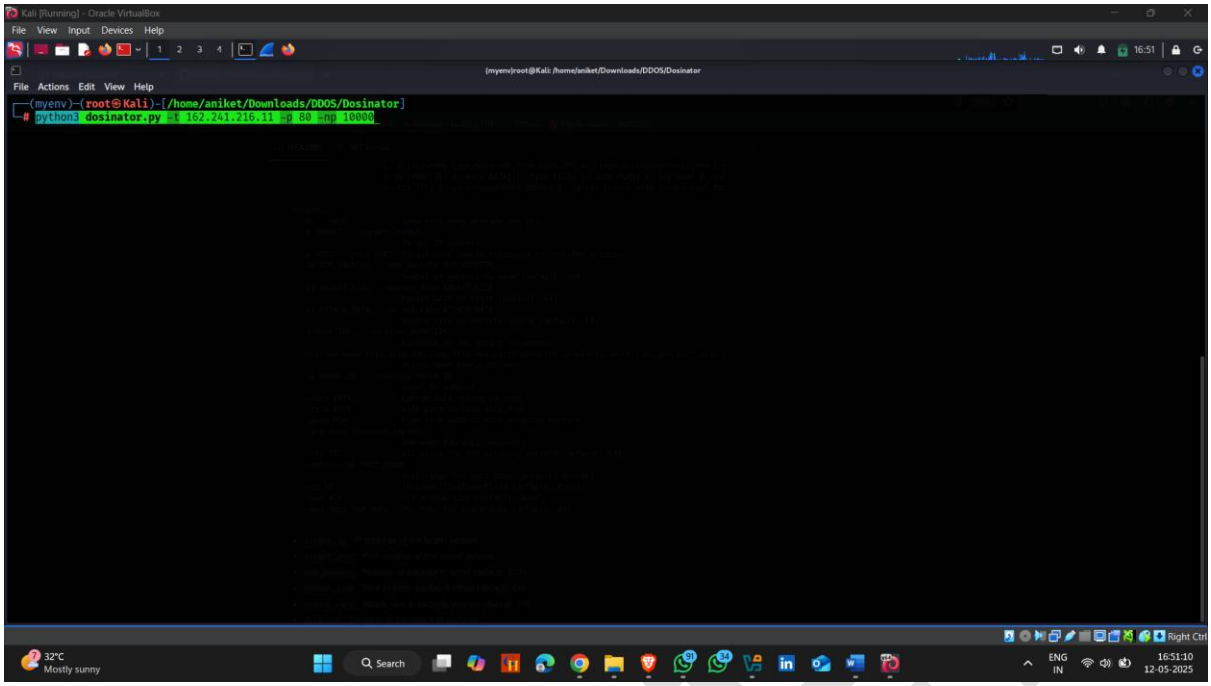


- Now run **dosinator.py** file

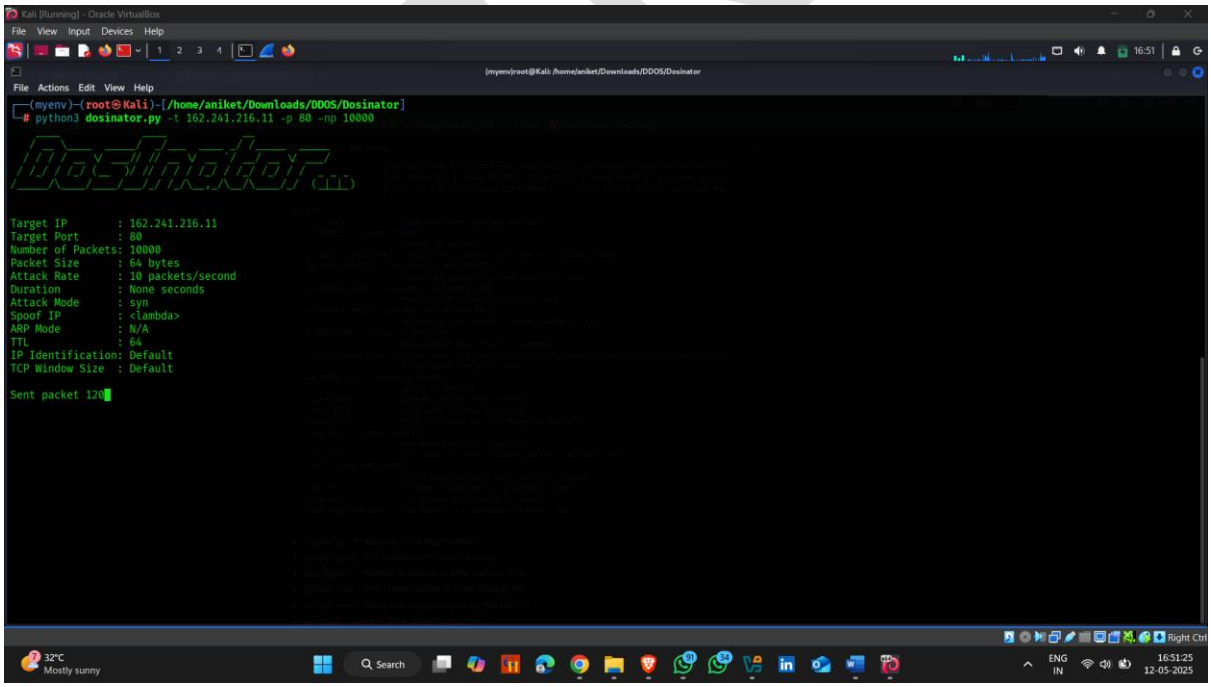
Command-: python3 dosinator.py -t 162.241.216.11 -p 80 -np 10000

Explaination-:

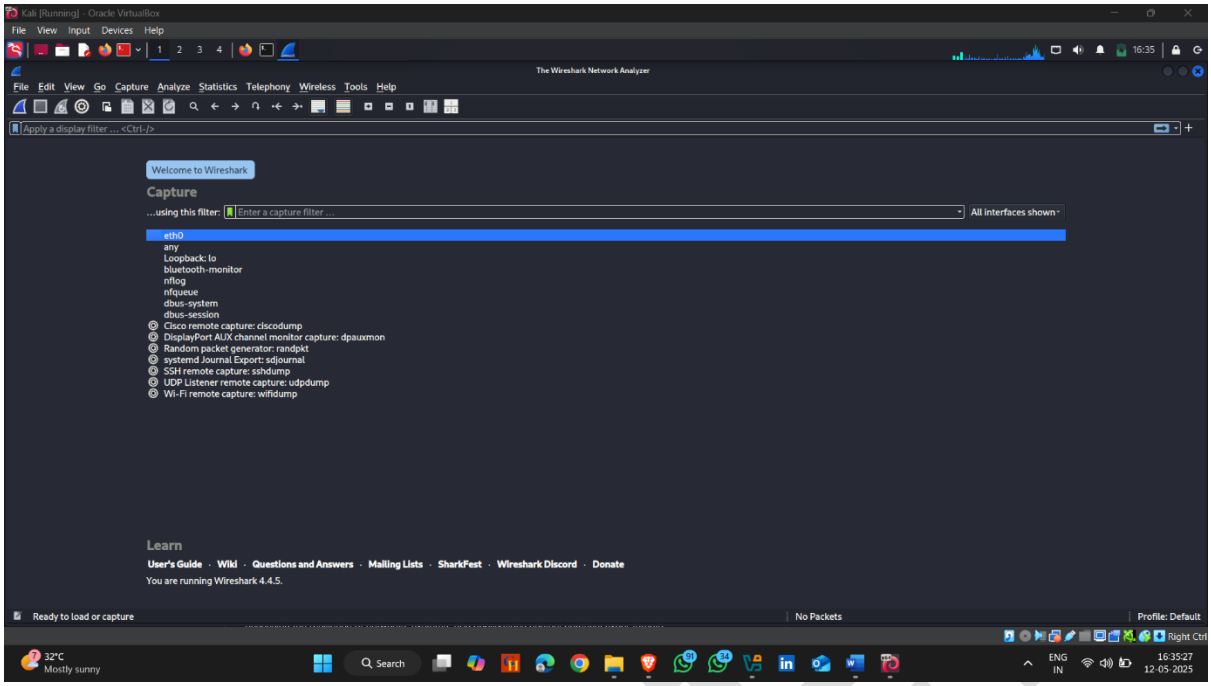
1. **Python3 – Python based tool**
2. **Dosinator.py – main file**
3. **-t – target**
4. **162.241.216.11 –target ip**
5. **-p – port**
6. **80 –port number**
7. **-np –number of packets**



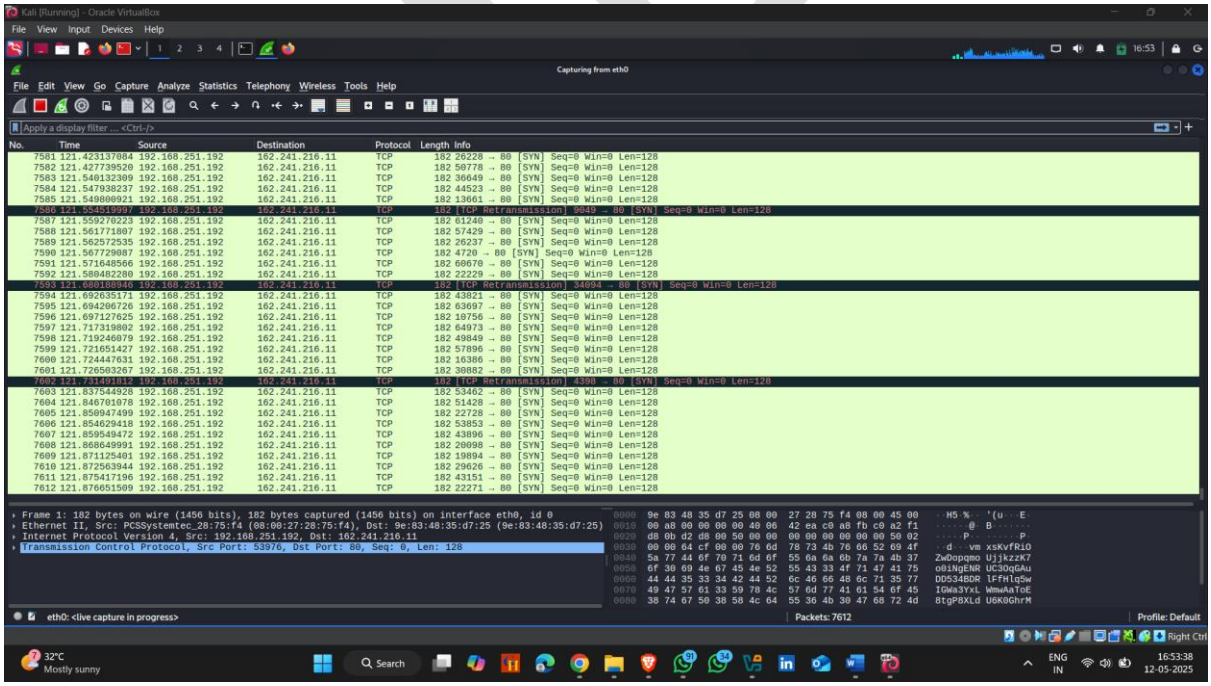
- Attack Started



- Open Wireshark for monitoring
- Select network interface



- **Attack Started** 👇



Detect And Monitor DOS/DDOS Attack

1.Perform Detect And Monitor DOS/DDOS Using Snort

Snort is an **open-source Intrusion Detection System (IDS)** and **Intrusion Prevention System (IPS)** developed by **Cisco (originally by Sourcefire)**. It's one of the most widely used tools for real-time **network traffic monitoring**, **packet analysis**, and **attack detection**.

🎯 **Why is Snort Used?**

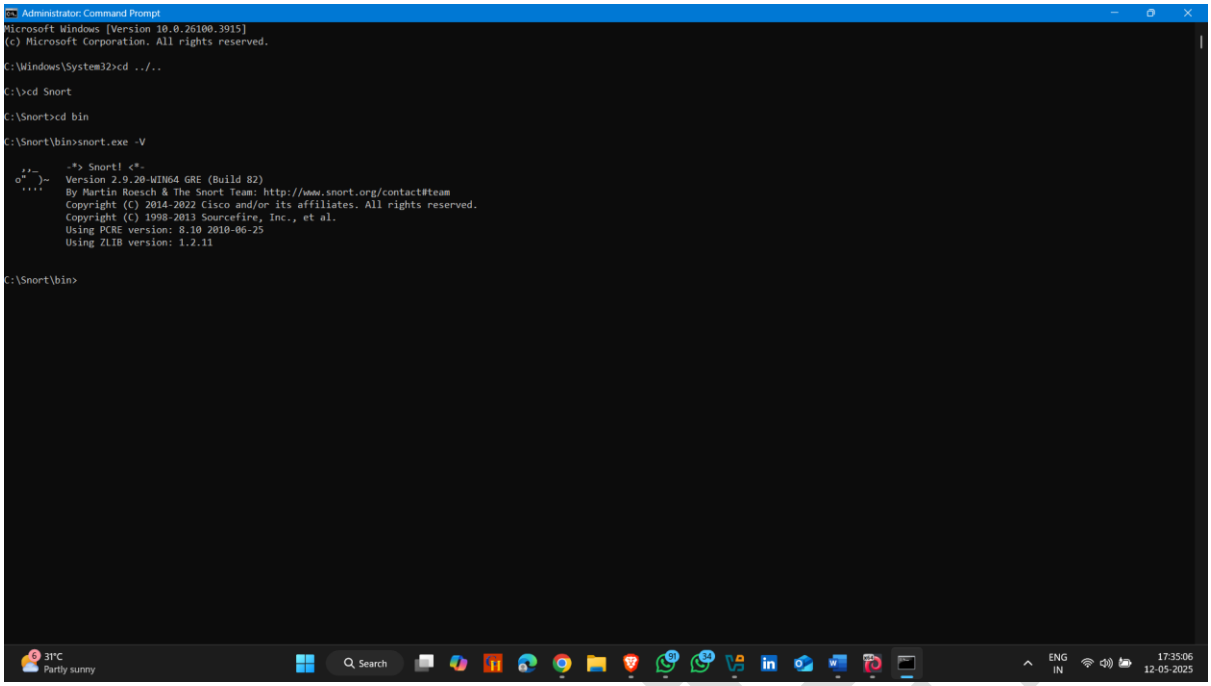
Snort is used primarily to **detect malicious activity** on networks and sometimes to **block** it, depending on how it's deployed.

Note -: Before run Snort ,kindly first Download and configure it

Youtube -: https://youtu.be/U6xMp-MIEfA?si=329iLmHly-I2xPrS

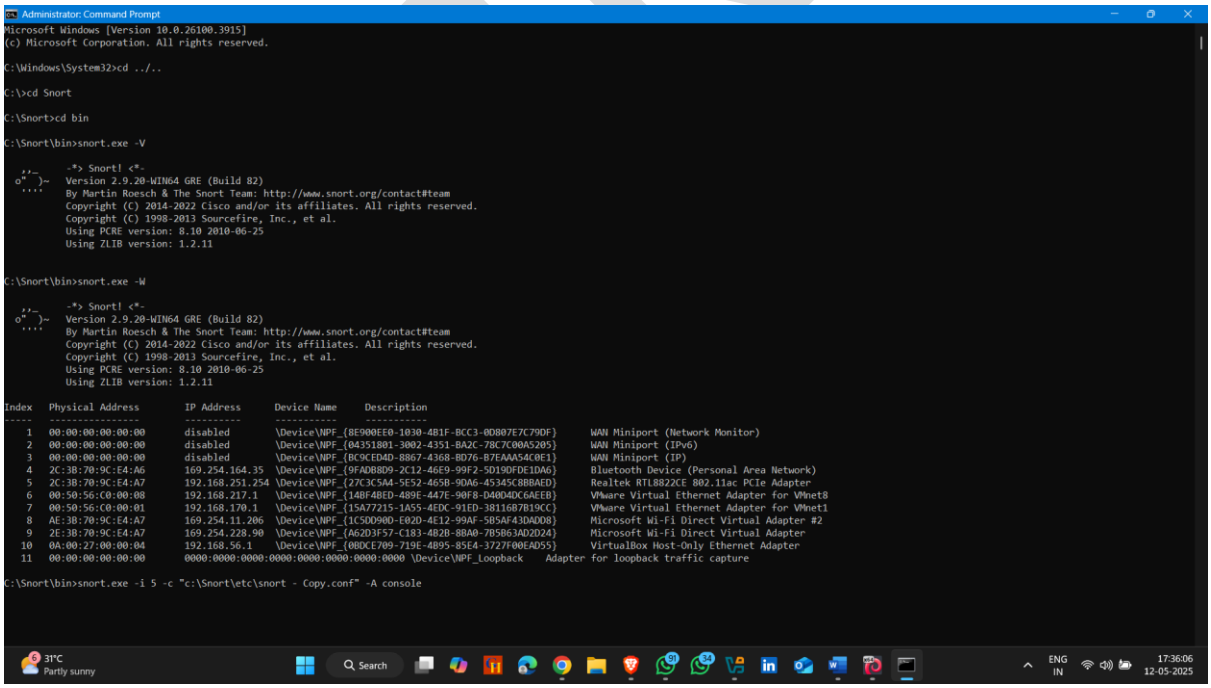
How to use it -:

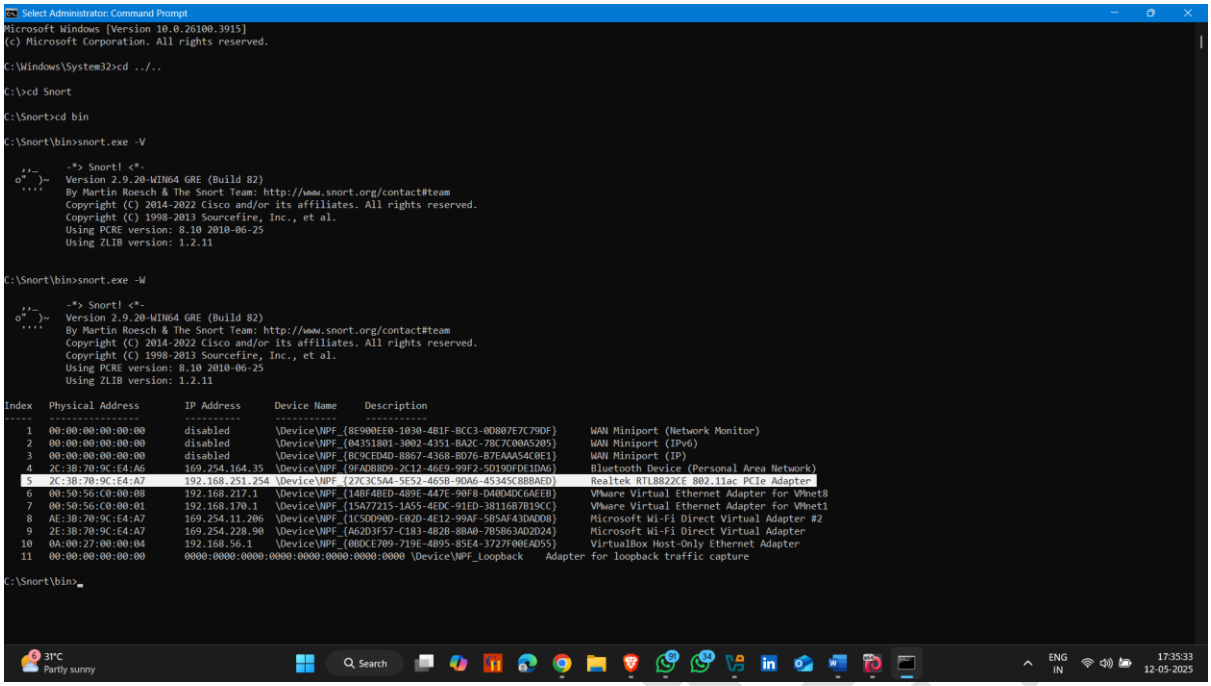
- Open command line Interface (CLI) as a administrator and go to the Snort directory



- Find network interface

Command -: snort.exe -W



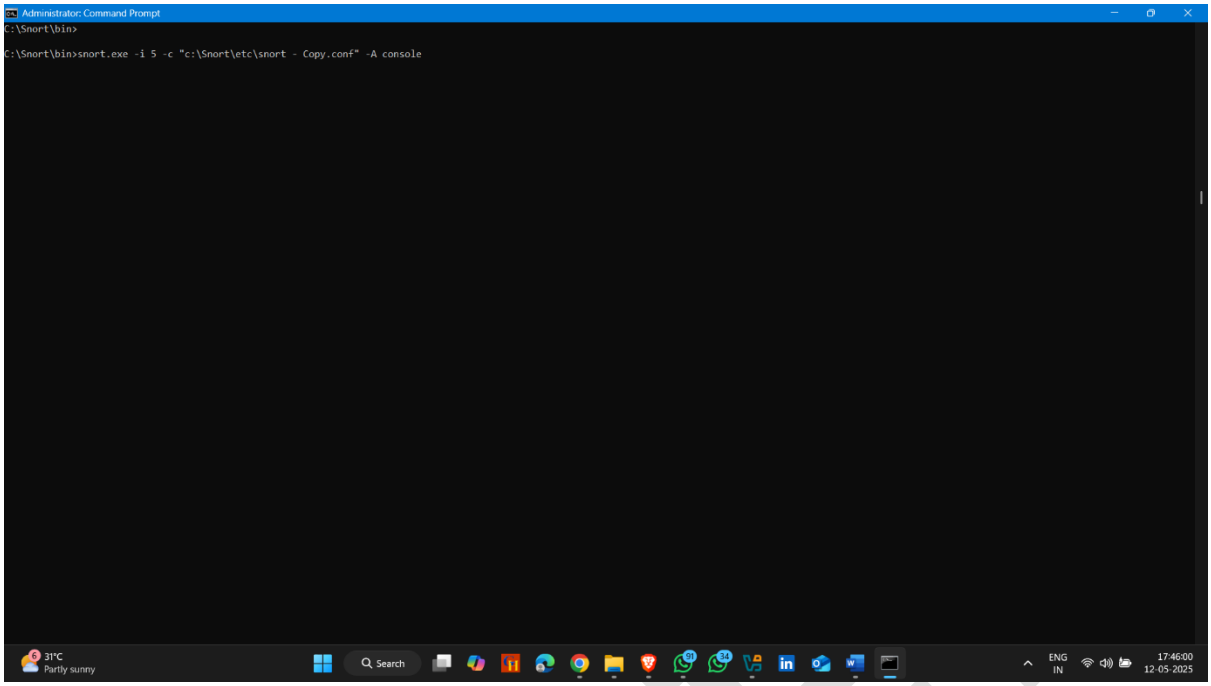


- **Now start Snort**

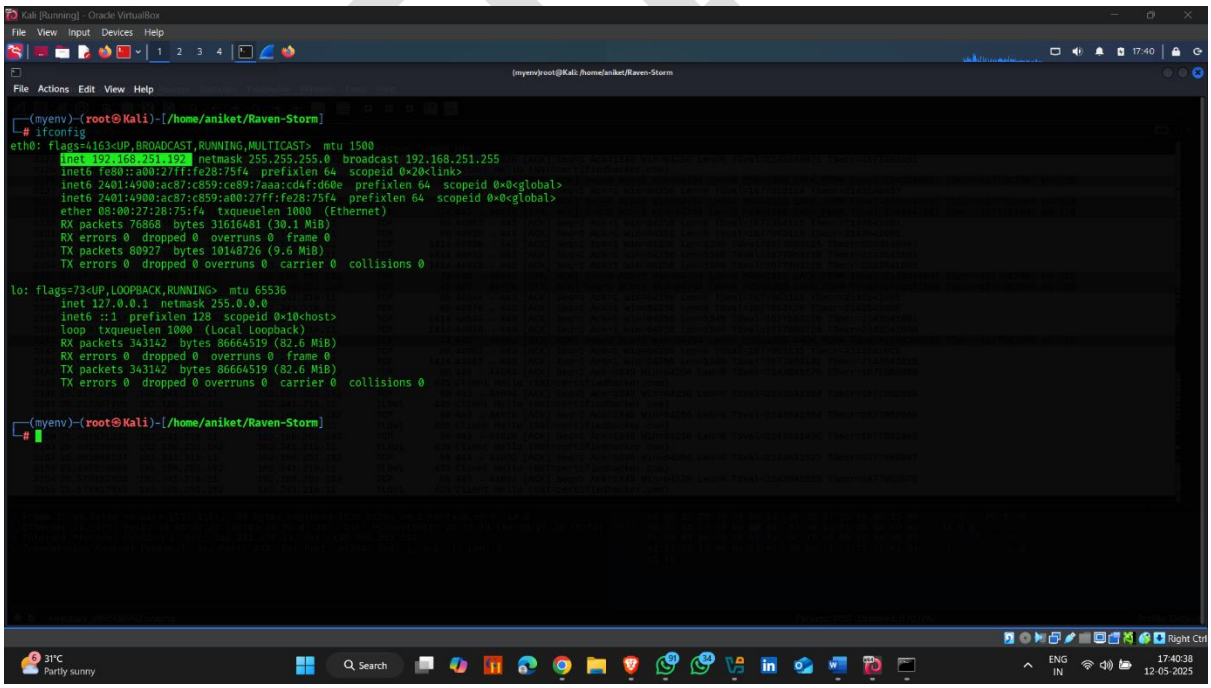
Command -: snort.exe -i 5 -c "c\Snort\etc\snort – copy.conf " -A console

Explaination -:

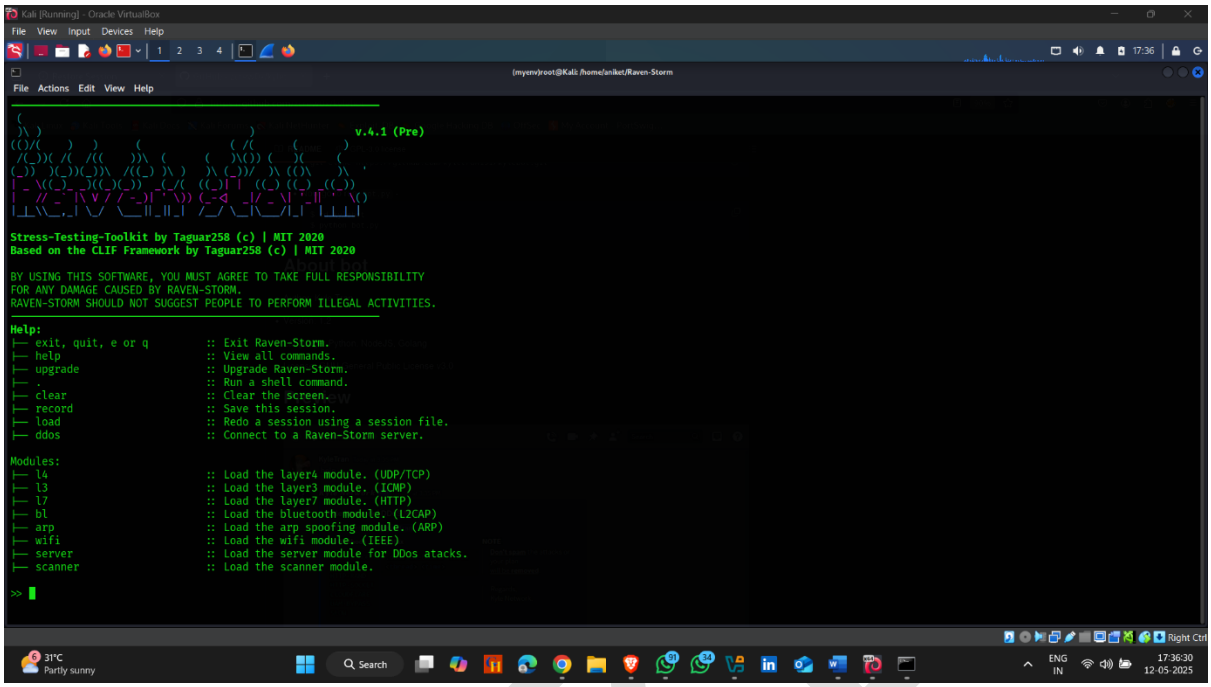
1. **Snort.exe -:** Launches Snort in Windows (the executable)
2. **-i 5 –:** Tells Snort to use network interface number 5 (you can list interfaces using snort.exe -W)
3. **-c "c\Snort\etc\snort – copy.conf" -:** Specifies the Snort configuration file to use; this path seems incorrect (should be C:\\Snort\\etc\\snort-copy.conf)
4. **-A console -: Outputs alerts to the console instead of log files**



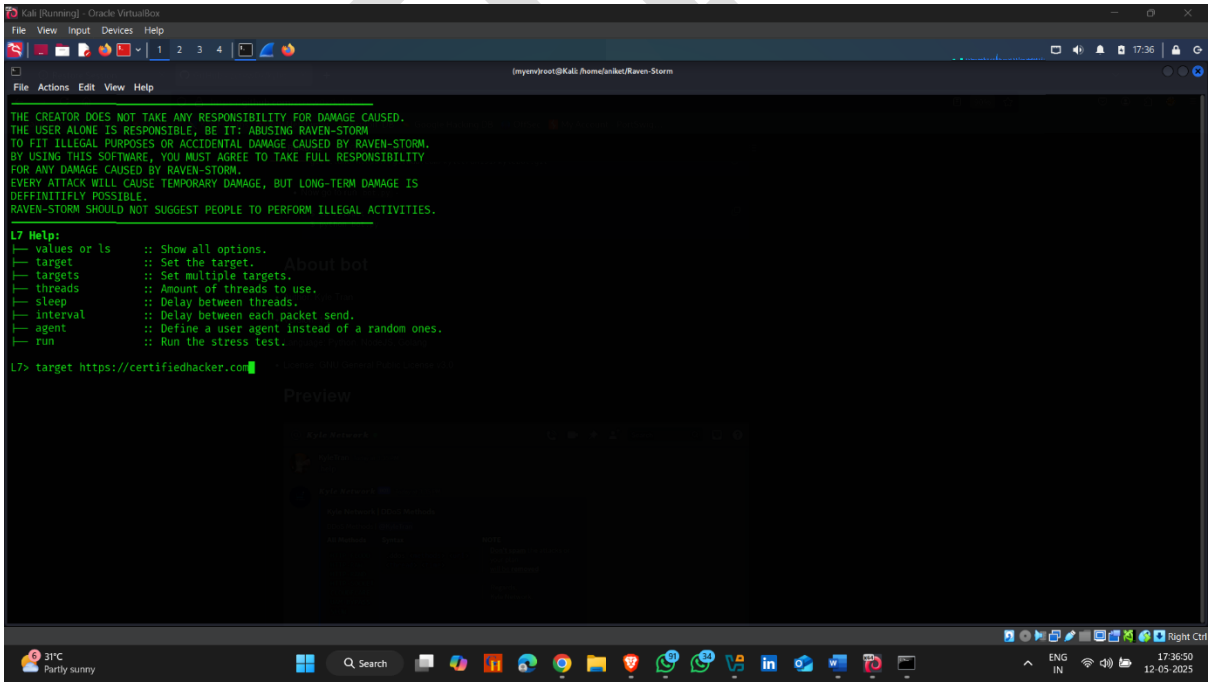
- Now open kali linux (VM) for Dos Attack
- Kali linux ip address 👇



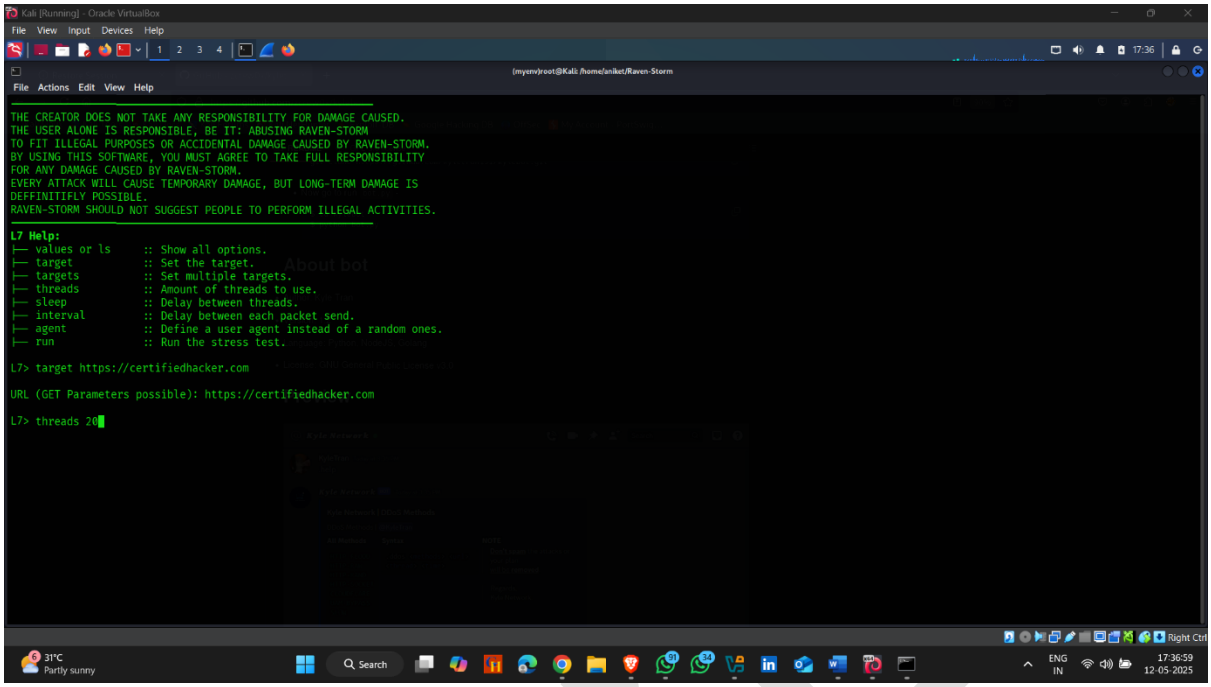
- **Tool used – Raven -Storm**
- Used L7 –Layer 7 attack



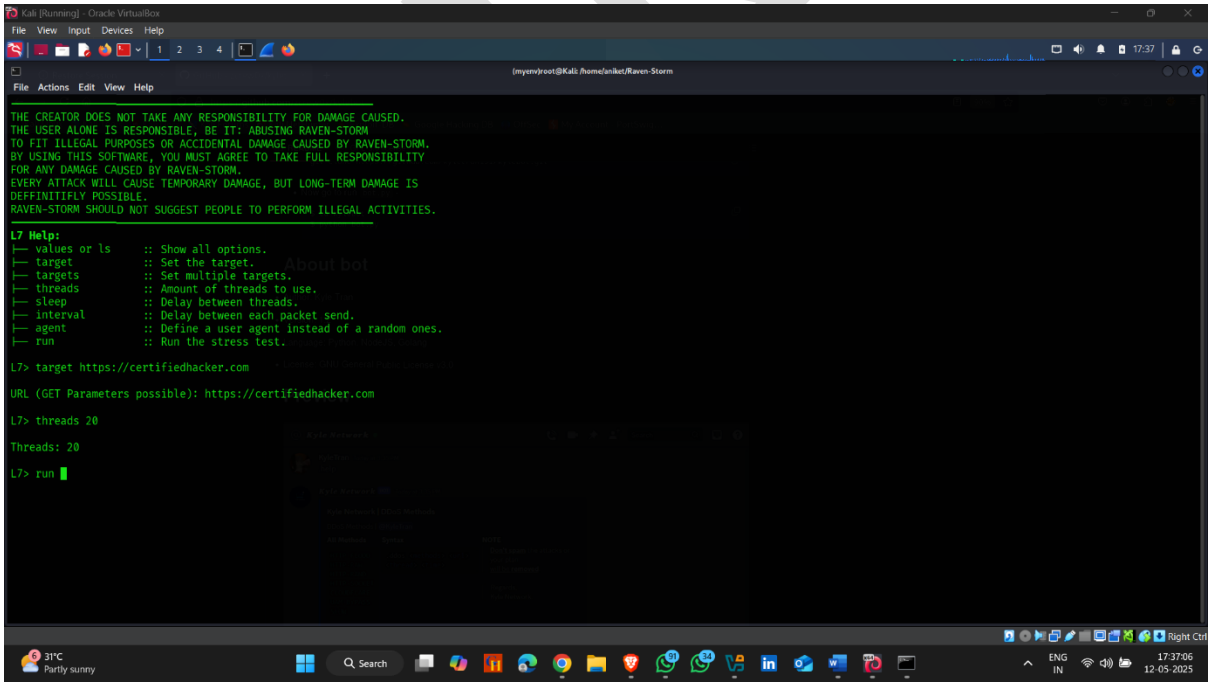
- Set target



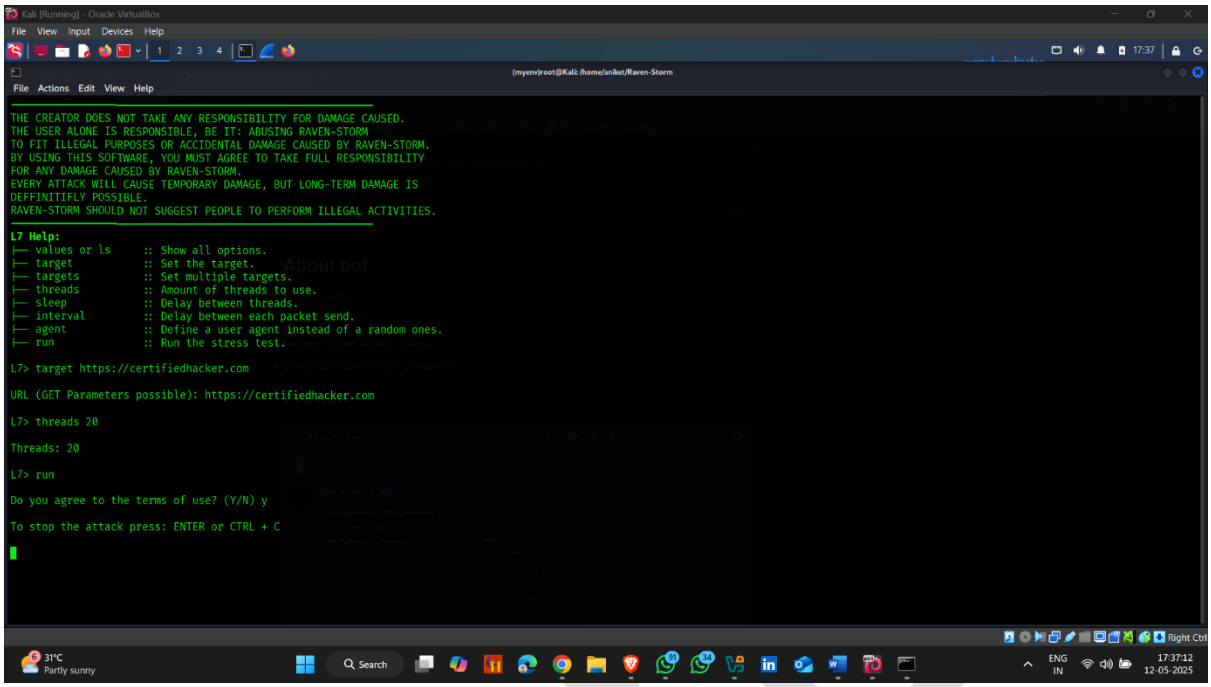
- Set threads



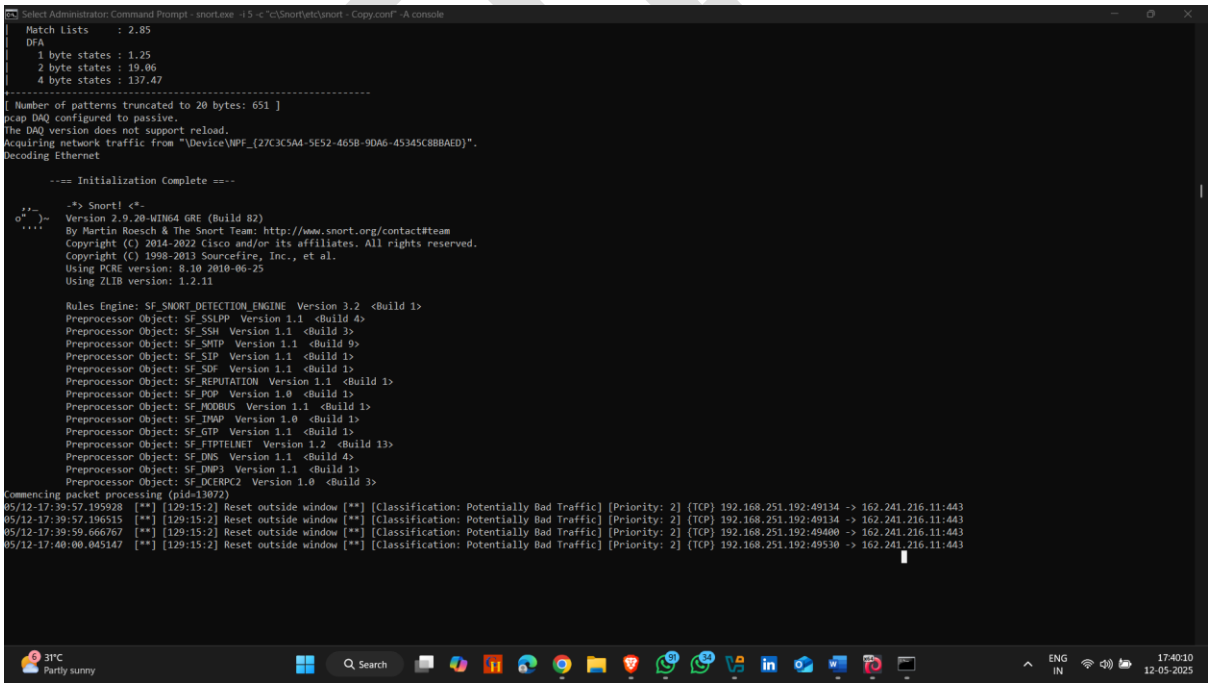
- Run

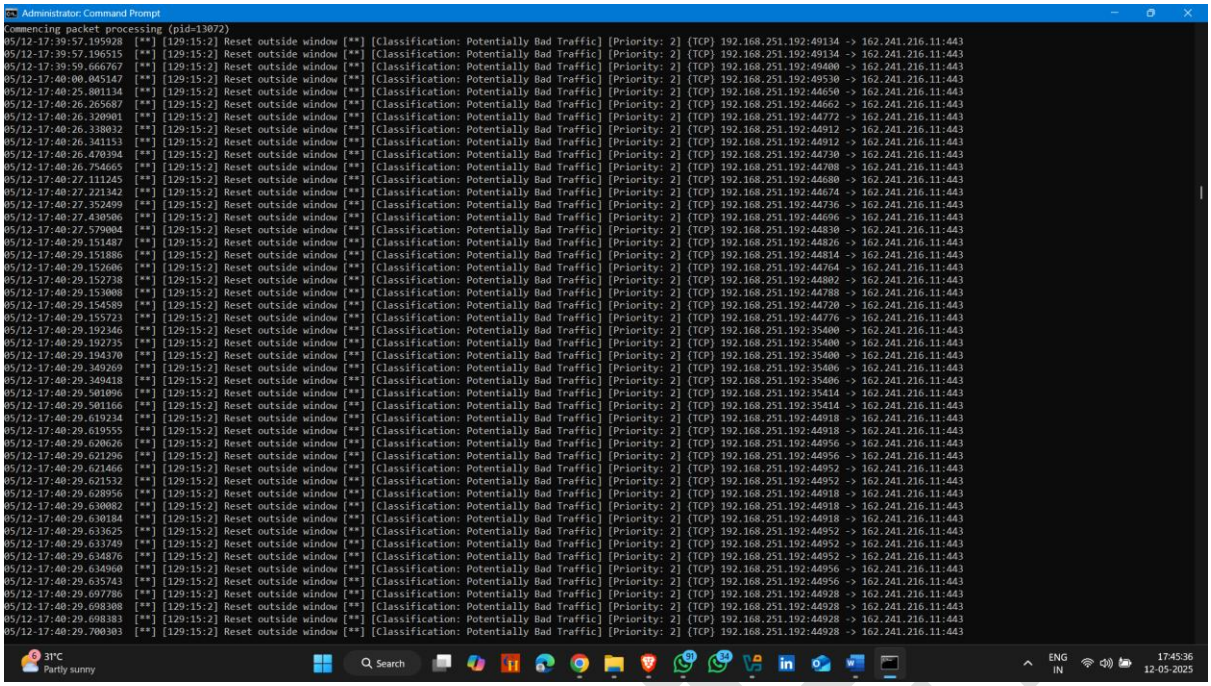


- Attack started 👇



- Now open Snort
- Snort detected attack





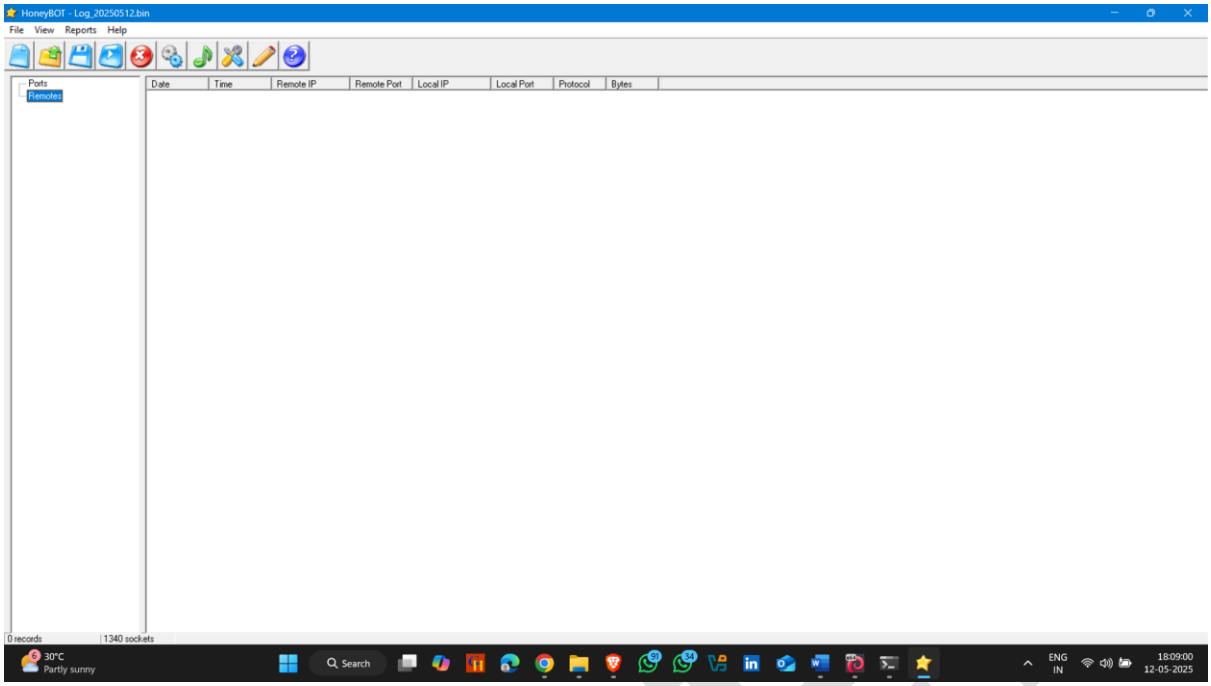
2.Perform Detect And Monitor DOS/DDOS Using HoneyBOT

HoneyBOT (short for *Honeypot BOT*) is a **Windows-based honeypot software** used for **network security monitoring** and **threat analysis**. It simulates vulnerable services on a system to **attract attackers**, allowing security professionals to **observe and analyze** their behavior **without risking a real system**.

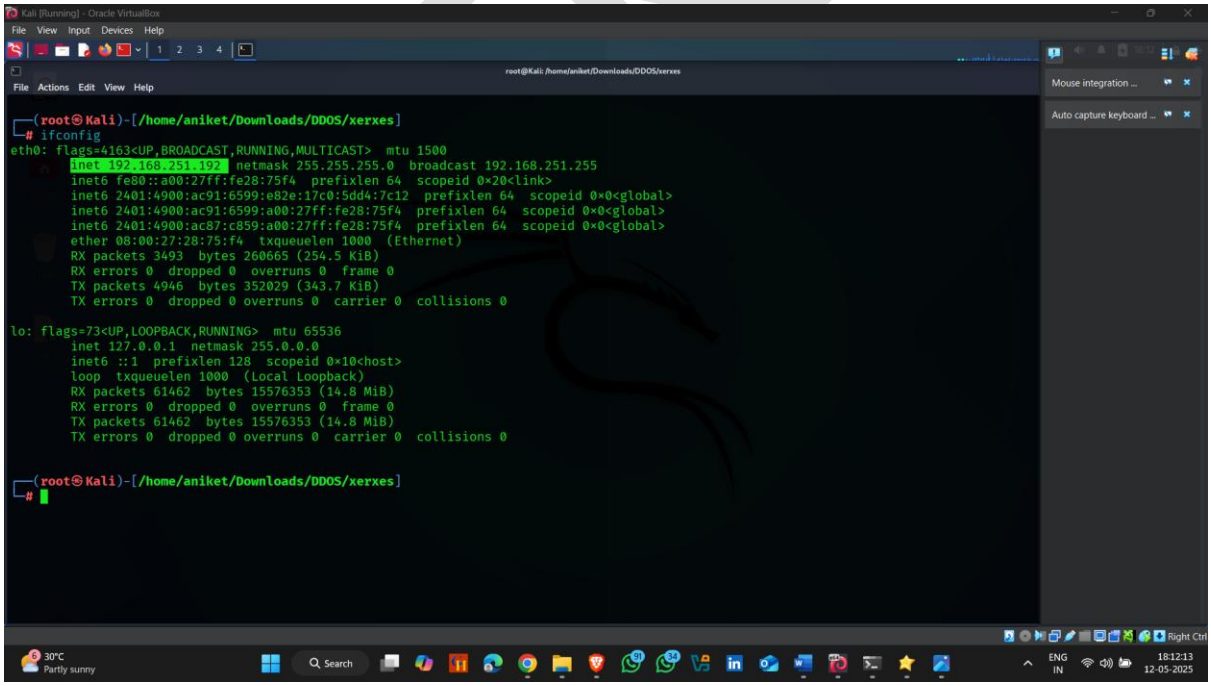
Download Link -:
https://honeybot.software.informer.com/download/

How to use it –:

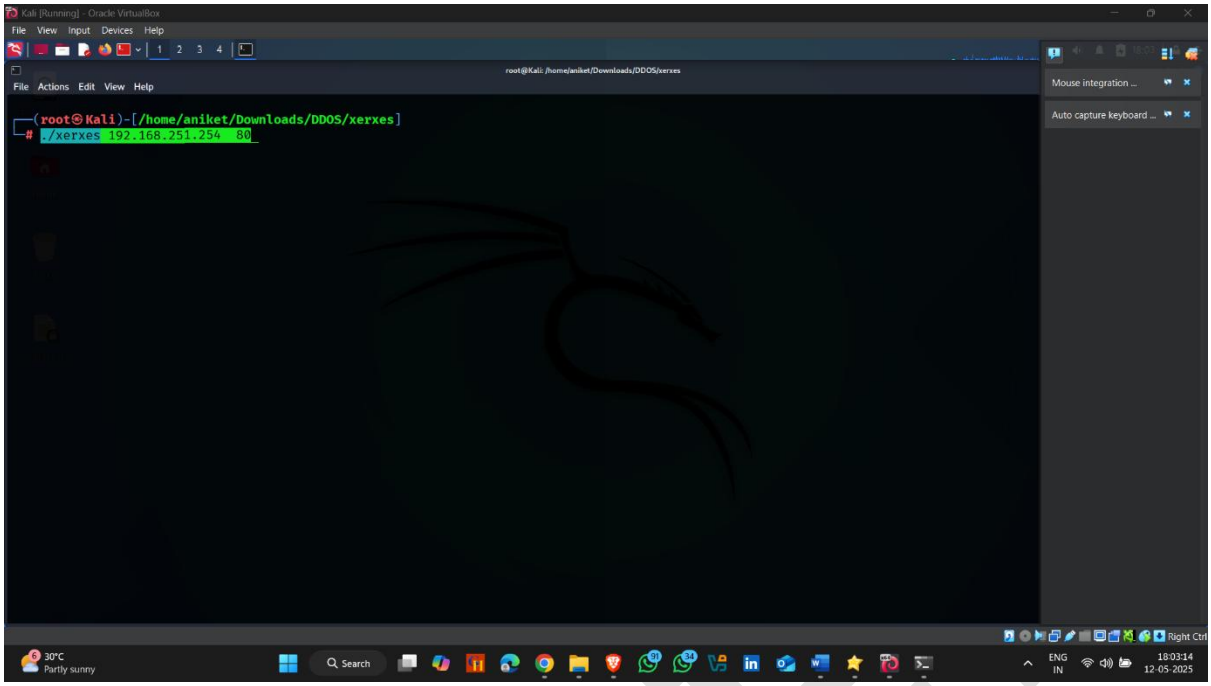
- After installation open the HoneyBOT application



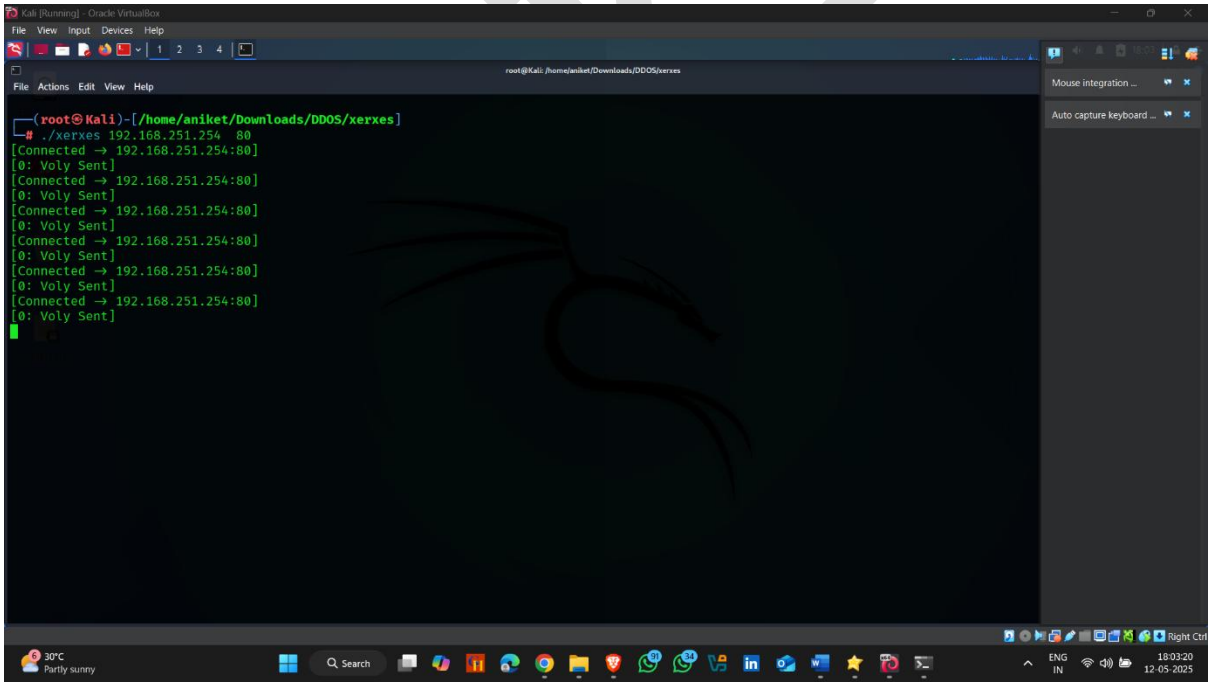
- Now open kali terminal virtual machine
- Kali linux ip address



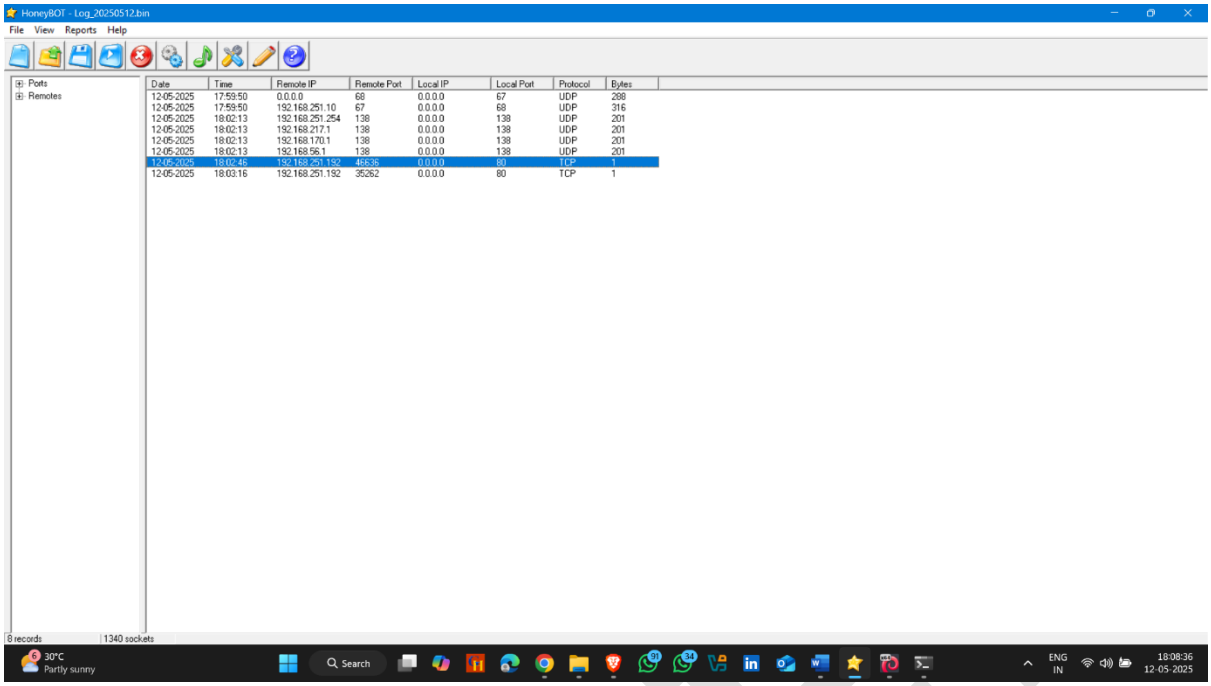
- Dos tool -; XerXes tool



- Attack Started 👇



- Back to the HoneyBOT application
- Attack detect



3.Perform Detect And Monitor DOS/DDOS Attack Using Wireshark

Wireshark is a **free and open-source network protocol analyzer**. It lets you **capture, inspect, and analyze** network traffic in real time at a very detailed level—**down to each packet**.

It's one of the most widely used tools in cybersecurity, networking, and digital forensics.

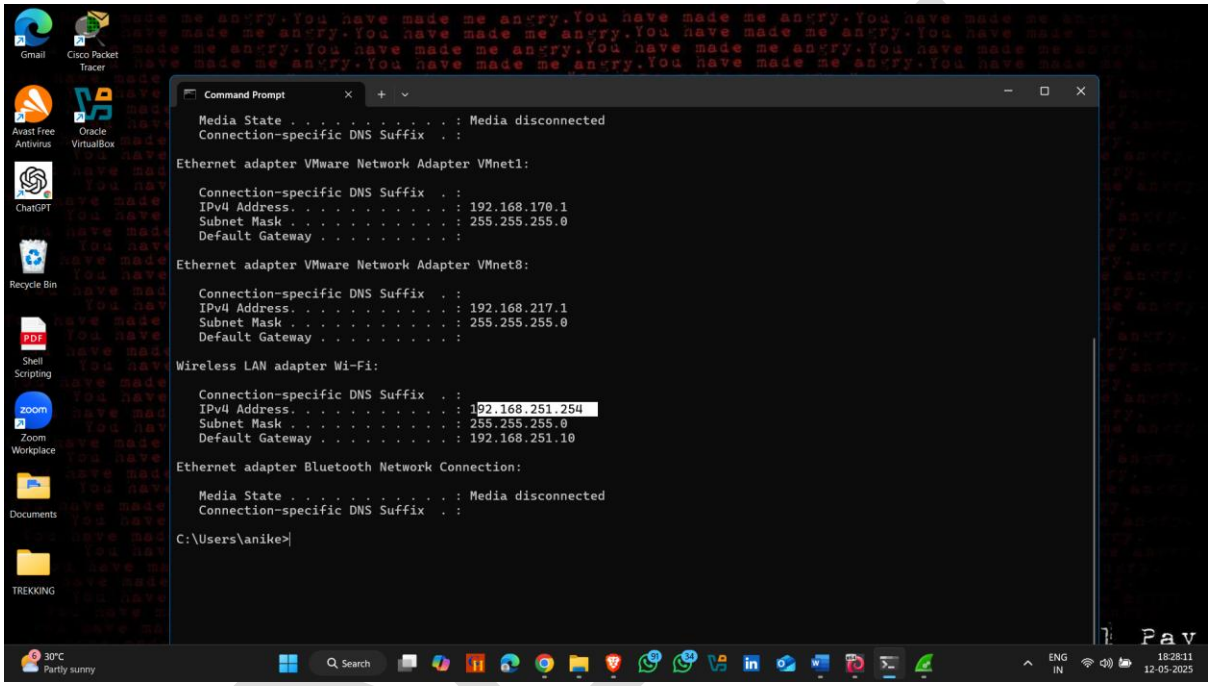
Purpose of Wireshark -:

- Wireshark is used to capture and analyze network traffic in real time.
- It helps identify network issues such as slow performance or packet loss.
- Security professionals use Wireshark to detect suspicious activity or attacks on the network.
- It is commonly used to study how network protocols like TCP, HTTP, and DNS function.

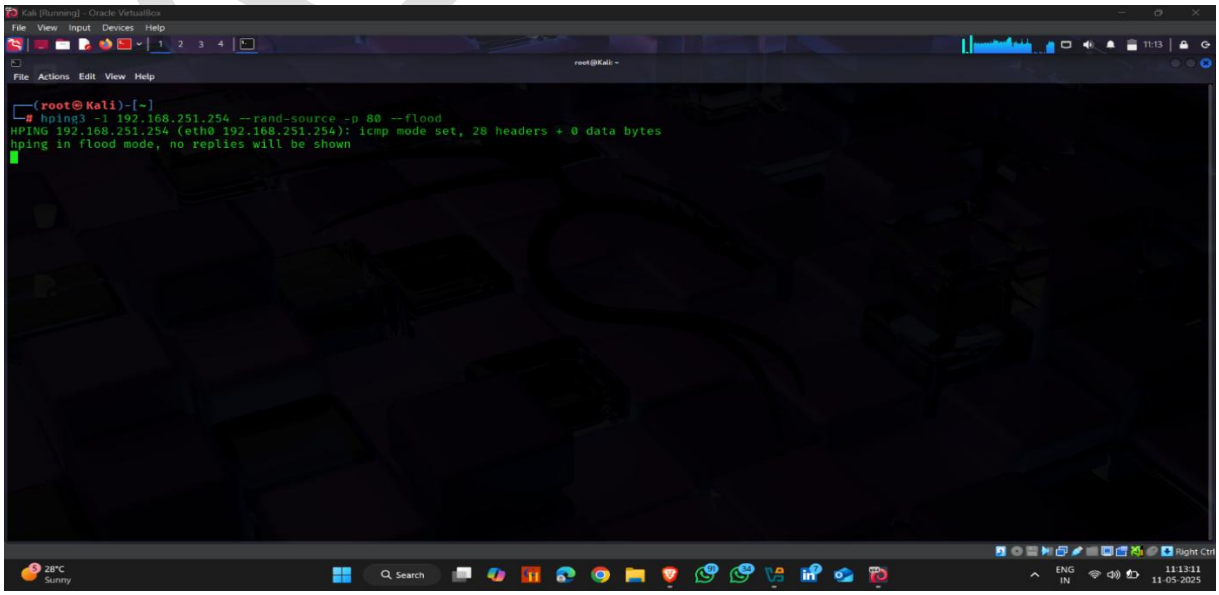
- Wireshark allows users to inspect packet contents to troubleshoot, investigate, or learn.

How to use it -:

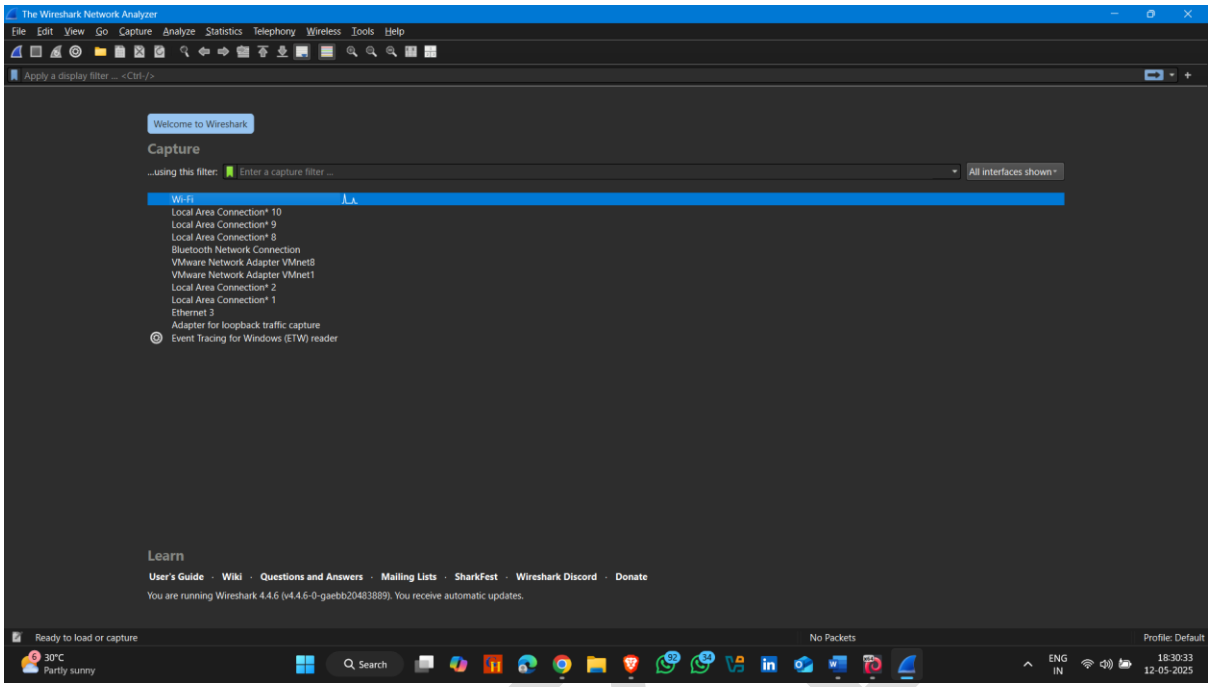
- **Target Ip address**



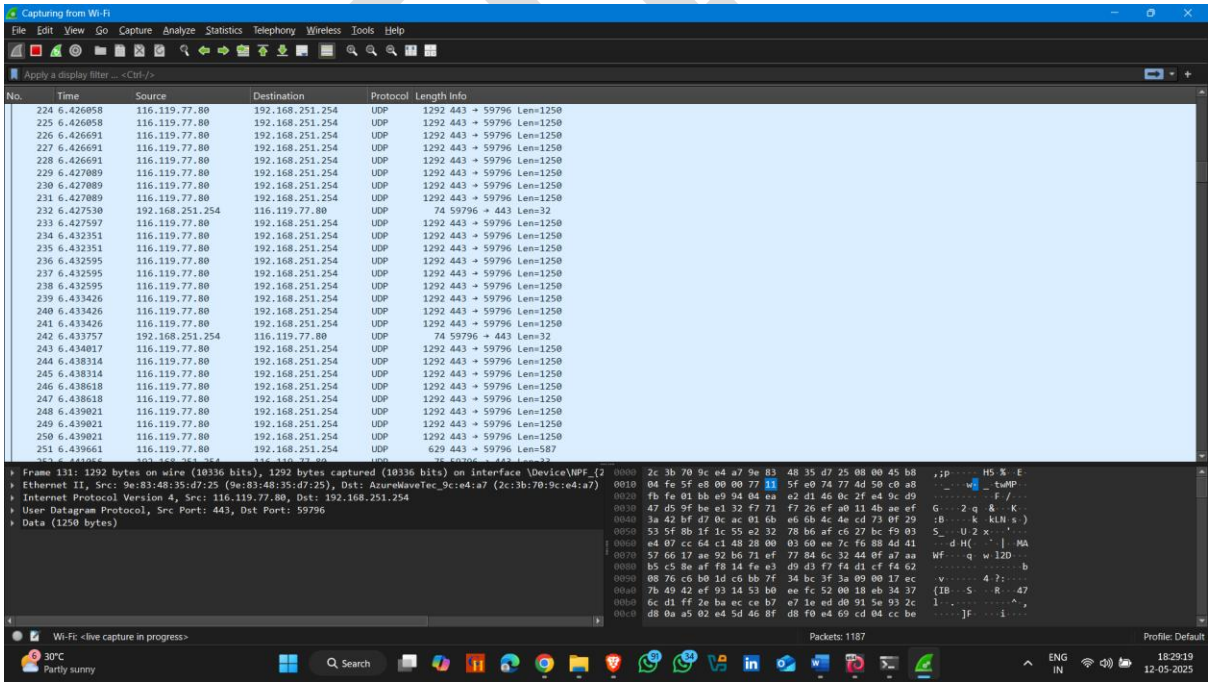
- Open kali linux terminal and perfom a attack
- Tool – using hping3
- Attack started



- Now open Wireshark on target machine
- Select network interface



- Capture incoming packet traffic



HOW TO PREVENT FROM DOS AND DDOS ATTACK

Preventing **DoS** and **DDoS** attacks requires a combination of **network hardening**, **traffic filtering**, **resource scaling**, and **incident response planning**. Here's a breakdown of effective prevention and mitigation strategies:

🛡️ 1. Network-Level Protection

- **Firewall Rules**: Block unused ports and restrict access to critical services.
- **Rate Limiting**: Limit the number of requests from a single IP address.
- **Geo-blocking**: Block or limit traffic from high-risk regions if not needed.
- **Intrusion Prevention Systems (IPS)**: Detect and stop suspicious traffic.

☁️ 2. Use Anti-DDoS Services

- **CDN & Cloud Protection**: Use services like:
  - **Cloudflare**
  - **AWS Shield**
  - **Google Cloud Armor**
  - **Azure DDoS Protection**
- These services absorb large-scale traffic before it reaches your server.

⚙️ 3. Application-Level Defenses

- **Web Application Firewall (WAF)**:
  - Blocks malicious HTTP traffic (e.g., HTTP floods).
  - Helps filter bad bots and SQL injection attempts.
- **Captcha/Challenge Pages**:
  - Stops automated traffic during attacks (e.g., Cloudflare "I'm Under Attack" mode).

🧠 4. Monitoring & Detection

- **Log Monitoring**:
  - Use tools like **Splunk**, **ELK Stack**, or **Graylog** to analyze traffic.
- **Traffic Anomaly Detection**:
  - Alerts if traffic spikes suddenly or unusual patterns are found.
- **Tools**:
  - Wireshark, NetFlow, SNORT, or Suricata can help detect unusual traffic.

⚙️ 5. Infrastructure Design

- **Redundancy & Load Balancing**:
  - Distribute traffic across multiple servers or data centers.
- **Auto-scaling**:
  - Cloud services can automatically scale up during spikes.
- **Separate Public & Private Services**:
  - Keep critical services (like admin panels or internal APIs) on separate networks.

🧩 6. Use Reverse Proxies

- Acts as a buffer between users and servers.
- Helps hide the real IP of your web server.

📋 7. Have an Incident Response Plan

- Define steps to follow during an attack.
- Have contact points for your ISP and DDoS mitigation providers.
- Set up alerting systems and regular backups.

🧪 Tools for Testing (Ethical Use Only)

- **LOIC**, **HOIC**, **Slowloris**, **Hping3**, **D0sinator** – for lab simulations.
- Test only in **controlled environments** (CTFs, home labs, VMs).

THANK YOU