MODULE: 3

Report on: Scanning Networks

-ATHARVA KATKAR

NETWORK SCANNING

Network scanning is a process used in computer networks to identify active devices, services, and potential vulnerabilities within a network. It involves sending data packets to target systems and analyzing their responses to gather information such as IP addresses, open ports, running services, and security risks.

Objectives –

- Discovery
- Security assessment
- Vulnerability detection
- Network mapping
- Performance analysis

Types of Network Scanning:

1. Port Scanning

Used to find:

- Open ports
- Closed ports
- Filtered (firewalled) ports

Tools: Nmap, Masscan

Examples:

- nmap -sS → SYN Stealth scan
- nmap -sU → UDP scan
- nmap -sV → Service version detection

2. Host Scanning (Ping Scanning)

Finds live hosts in a network.

Tools: Nmap, Angry IP Scanner

Examples:

- nmap -sn 192.168.1.0/24
- nmap -PR → ARP ping on LAN
- nmap -PE → ICMP ping

3. Vulnerability Scanning

Identifies weaknesses or CVEs.

Tools: Nessus, OpenVAS, Qualys

Finds:

- Missing patches
- Weak configurations
- Known vulnerabilities

4. Network Mapping (Topology Discovery)

Discovers network structure, routers, paths.

Tools: Nmap, traceroute

Why Network Scanning Is Important:

- Identify live hosts
- Identify open ports
- Identify services and versions
- Detect misconfigurations
- Prepare for exploitation

HOST DISCOVERY

Host Discovery is the process of identifying which systems (hosts) are alive and reachable on a network.

What Is Host Discovery?

Host Discovery (also called ping scanning) is the process of checking: Which devices are up and reachable on a network.It does not scan ports or services.

It only answers the question:

✓ Is the host alive?
✓ Is it reachable across the network?
✓ Does it respond to basic network probes?

The following are examples of host discovery techniques:

- . ARP ping scan
- . UDP ping scan
- . ICMP ping scan (ICMP ECHO ping, ICMP timestamp, ping ICMP, and address mask ping)
- TCP ping scan (TCP SYN ping and TCP ACK ping)
- IP protocol ping scan

Host Discovery using Nmap:

Test/Tool Shown:

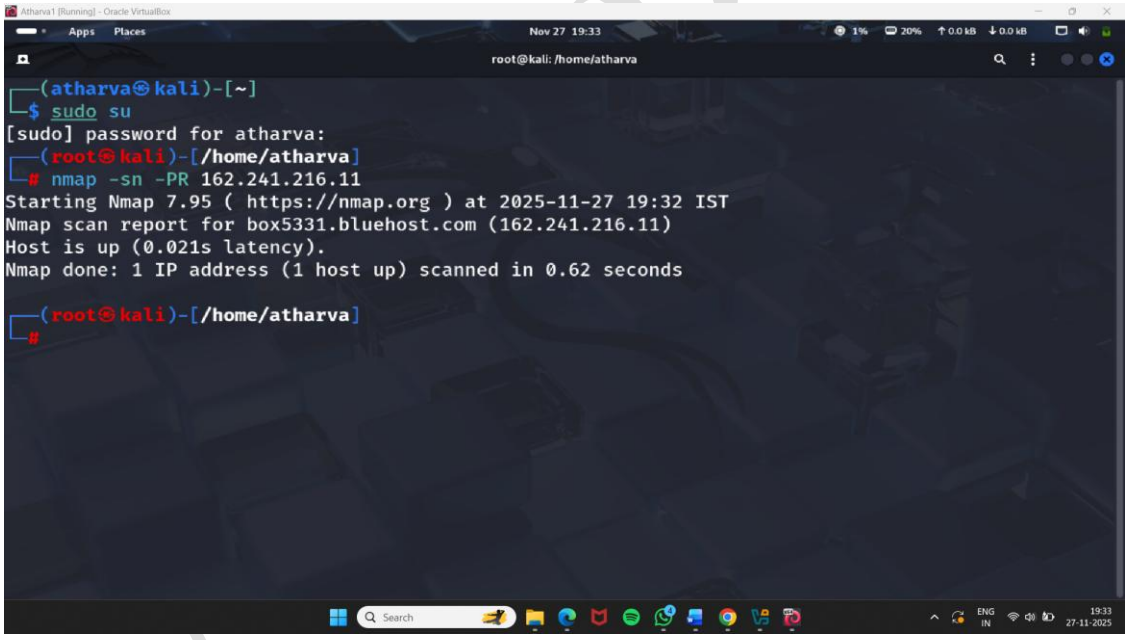
Nmap (nmap -sn -PR) on Kali Linux terminal

-sn: disables port scan an -PR: performs ARP ping scan.

Step Performed:

Executed the command: nmap -sn -PR 162.241.216.11

Output:



Fig(1)

- Nmap started normally.
- It resolved the IP address to a hostname: box5331.bluehost.com
- It detected: Host is up (0.021s latency)

Test/Tool Shown:

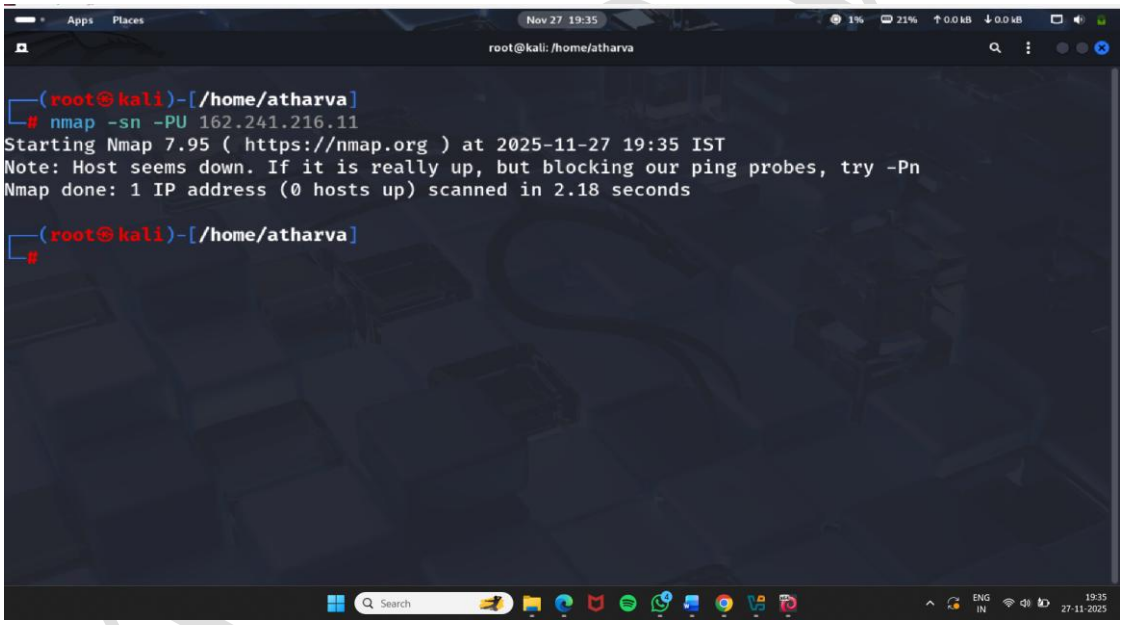
Nmap (nmap -sn -PU) on Kali Linux terminal

-PU: performs UDP ping scan

Step Performed:

Executed the command: nmap -sn -PU 162.241.216.11

Output:



Fig(2)

- Nmap attempted UDP ping packets.
- It returned the message:
  "Host seems down. If it is really up, but blocking our ping probes, try -Pn"
- Scan result: 0 hosts up

Test/Tool Shown:

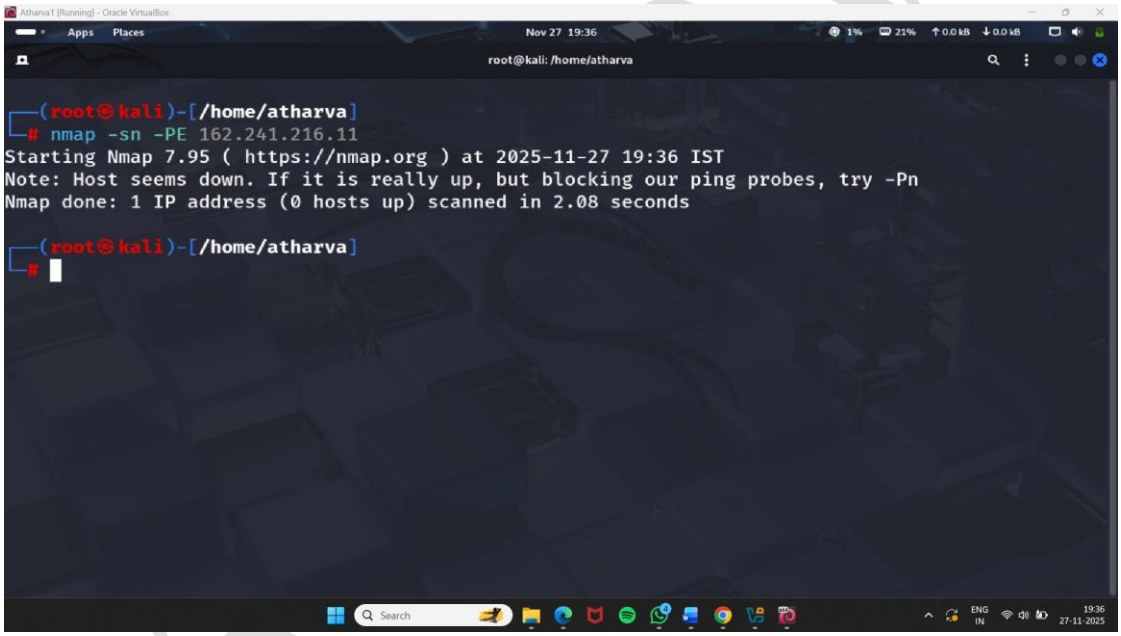
Nmap (nmap -sn -PE) – on Kali Linux terminal.

-PE : ICMP Echo ping (normal ping)

Step Performed:

Executed the command: nmap -sn -PE 162.241.216.11

Output:



Fig(3)

- Nmap attempted ICMP echo requests.
- Output displayed:
- "Host seems down. If it is really up, but blocking our ping probes, try -Pn"
- Final result: 0 hosts up

Test/Tool Shown:

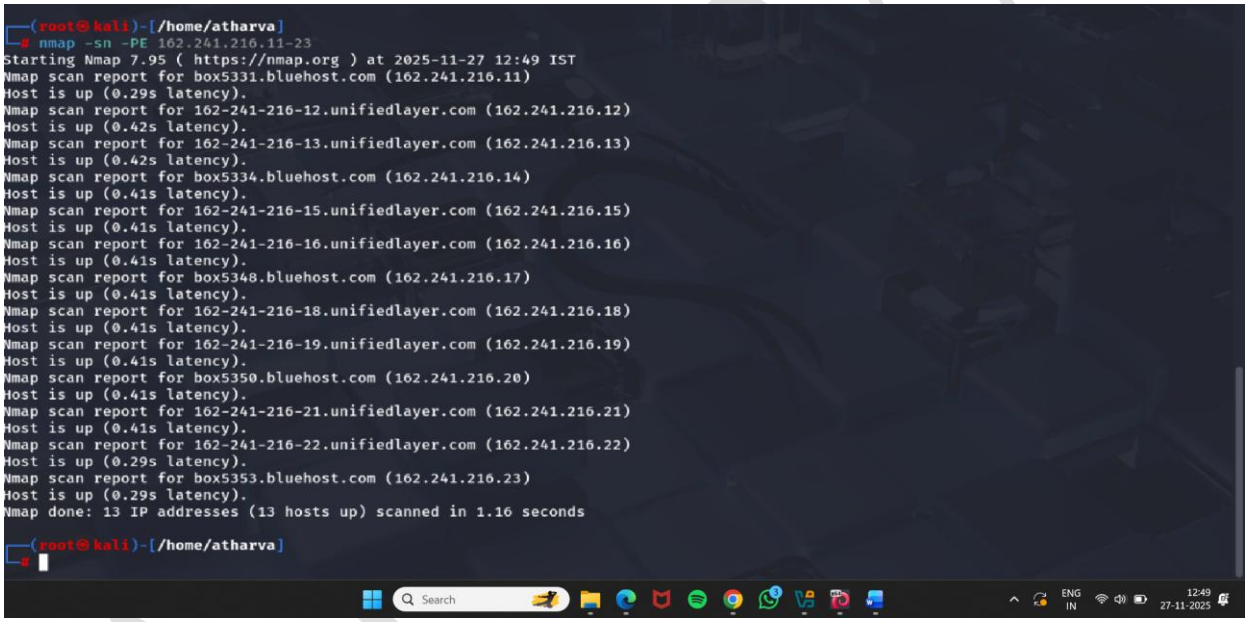
Nmap (nmap -sn -PE…….-23) – on Kali Linux terminal.

-PE: ICMP Echo ping

Step Performed:

Executed the command: nmap -sn -PE 162.241.216.11-23

Output:



Fig(4)

- Total Hosts Scanned: 13 IP addresses (the range .11 to .23 inclusive).
- Hosts Identified as Up: 13 hosts were found to be up.
- Scan Reports: Nmap successfully received ICMP Echo Replies from all 13 addresses.

Test/Tool Shown:

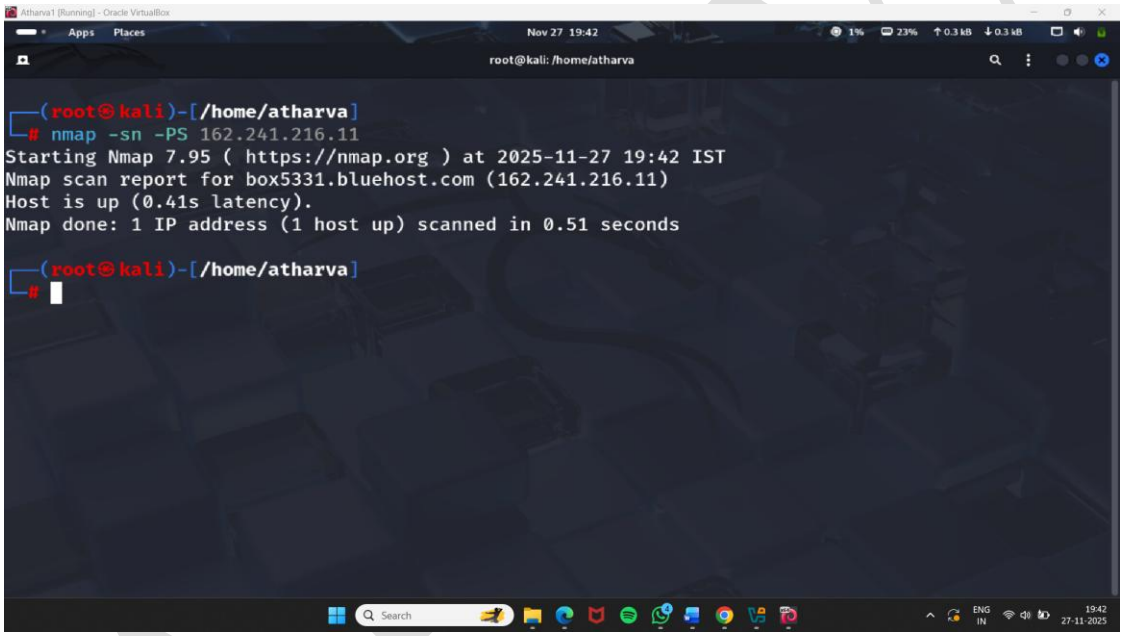
Nmap (nmap -sn -PS) – on Kali Linux terminal.

-PS: performs a TCP SYN Ping Scan

Step Performed:

Executed the command: nmap -sn -PS 162.241.216.

Output:



Fig(5)

- Nmap Scan Report: Nmap scan report for box5331.bluehost.com (162.241.216.11)
- Host Status: Host is up

❖ Mechanism: Nmap sent a TCP SYN probe to the host and received a response (likely a SYN/ACK), indicating the host is alive and a port (or the firewall) is responding to TCP requests.

- Final Result: Nmap done: 1 IP address (1 host up) scanned in 0.51 seconds.

Test/Tool Shown:

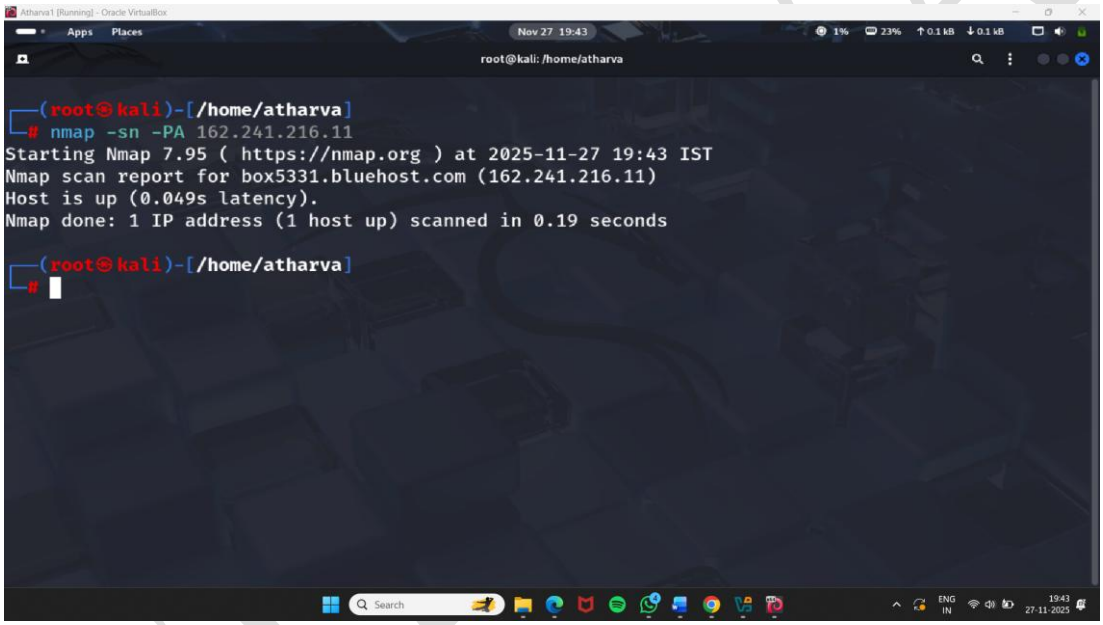
Nmap (nmap -sn -PA) – on Kali Linux terminal.

-PA: acknowledgement

Step Performed:

Executed the command: nmap -sn -PA 162.241.216.11

Output:



Fig(6)

- Nmap Scan Report: Nmap scan report for box5331.bluehost.com (162.241.216.11)
- Host Status: Host is up
- Mechanism: Nmap sent a TCP ACK probe to the target host. A responsive RST (Reset) packet indicates the host is alive, even if it's firewalled, as the packet successfully traversed the network stack. This is often used to bypass firewalls that block ICMP Echo requests (like the scenario described in your original "Observed Output").
- Final Result: Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds.

PORT & SERVICE DISCOVERY

Port and service discovery is the process of identifying open ports and services running on the target IP addresses/active

When analyzing a network, security professionals need to know:

- Which ports are open
- Which services are running
- Which versions or applications are active
- Which ports are filtered or blocked by firewalls

Port scanning techniques are categorized according to the type of protocol used for communication within the network:

- TCP Scanning
  - Open TCP scanning methods (TCP connect/full open scan)
  - Stealth TCP scanning methods (Half-open Scan, Inverse TCP Flag Scan, ACK flag probe scan, third party and spoofed TCP scanning methods)
- UDP Scanning
- SCTP Scanning
  - SCTP INIT Scanning
  - SCTP COOKIE/ECHO Scanning
- SSDP and List Scanning
- IPv6 Scanning

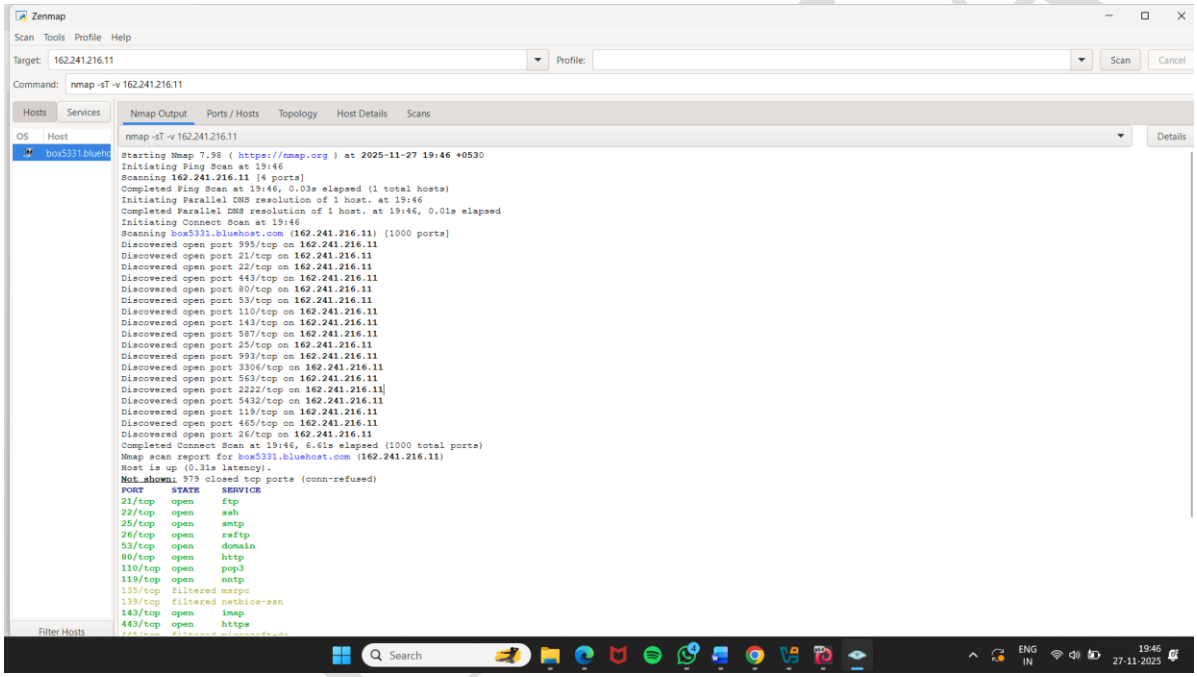
Test/Tool Shown:

- Zenmap (nmap -sT -v) on zenmap terminal.

Step Performed:

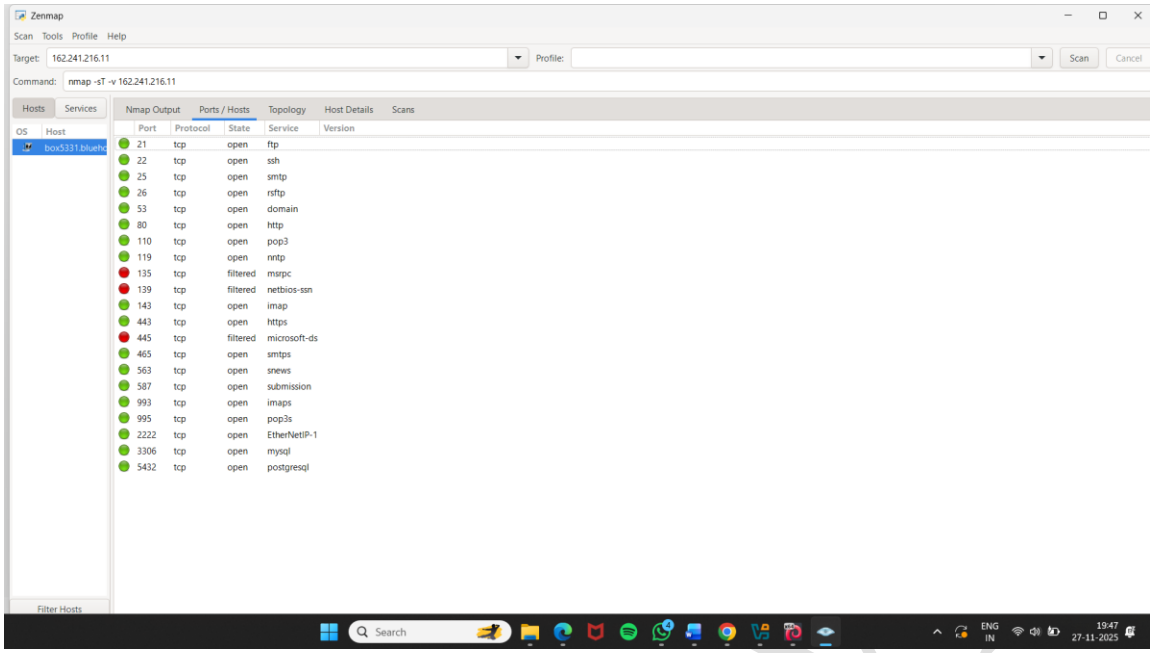
- A single target host 162.241.216.11 (box5331.bluehost.com) was scanned with Nmap from Zenmap using a TCP connect scan over the first 1000 TCP ports with DNS resolution and verbosity enabled.

Output:



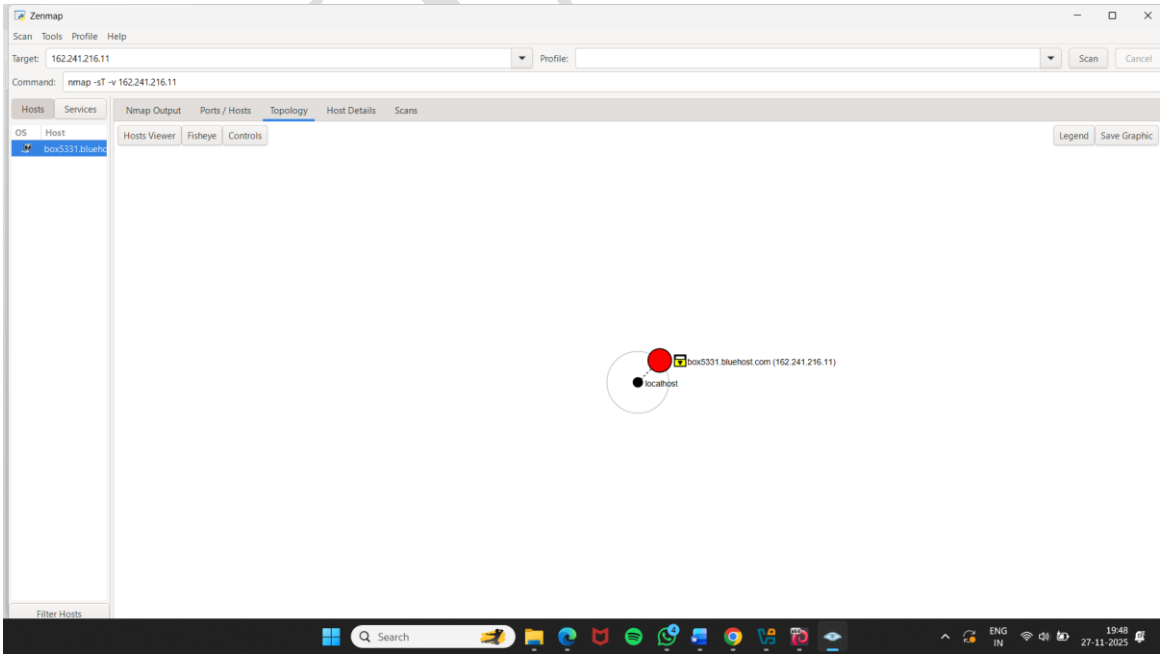
Fig(7)

- Host is up with low latency and resolves to hostname box5331.bluehost.com (an IP in a hosted/server network range).
- 16 TCP ports are reported open: 21, 22, 25, 53, 80, 110, 143, 443, 465, 587, 993, 995, 2222, 3306, 5432 plus 993/995 already counted, with services identified as ftp, ssh, smtp, domain (DNS over TCP), http, pop3, imap, https, smtps, submission, imaps, pop3s, EtherNetIP-1 (on 2222), mysql, and postgresql.
- Remaining scanned TCP ports (up to 1000) are shown as closed/filtered and not listed individually.



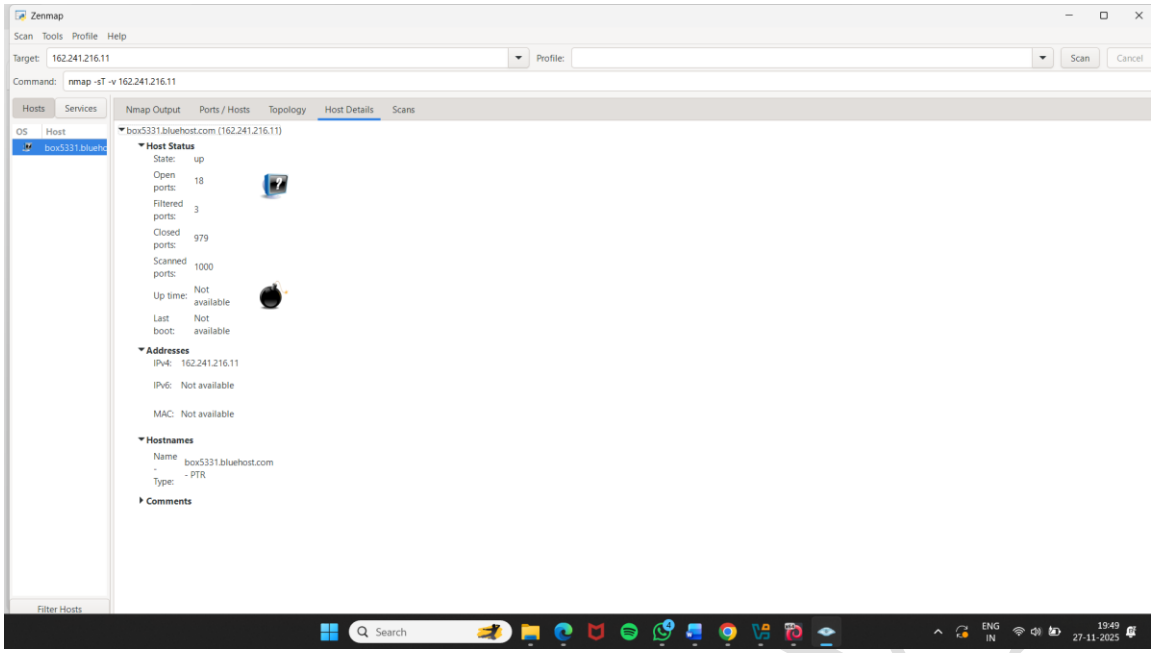
Fig(8)

The image shows the Ports / Hosts tab of the Zenmap graphical user interface, which is displaying the results of a successful Nmap port scan executed against the target IP address 162.241.216.11.

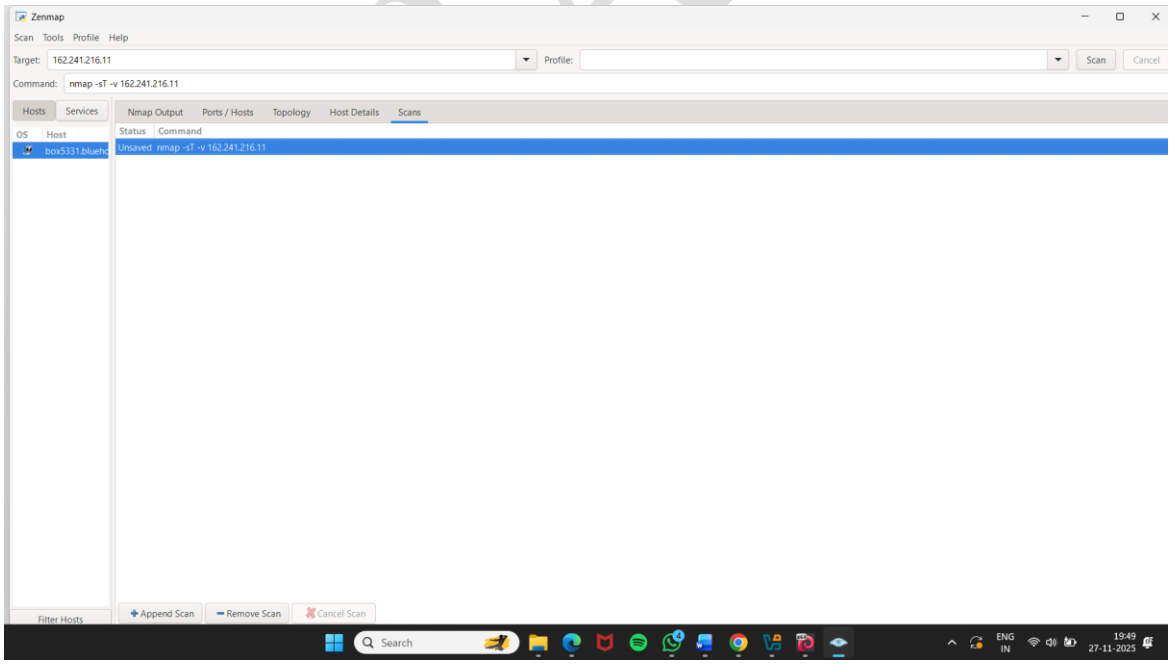


Fig(9)

The image displays the Topology tab within the Zenmap graphical user interface, showing a network visualization of the scan results.


Zenmap
Scan Tools Profile Help
Target: 162.241.216.11
Profile:
Scan
Cancel
Command: nmap -sT -v 162.241.216.11
Hosts
Services
OS
Host
box5331.bluehc
Nmap Output
Ports / Hosts
Topology
Host Details
Scans
box5331.bluehost.com (162.241.216.11)
Host Status
State: up
Open ports: 18
Filtered ports: 3
Closed ports: 979
Scanned ports: 1000
Up time: Not available
Last boot: Not available
Addresses
IPv4: 162.241.216.11
IPv6: Not available
MAC: Not available
Hostnames
Name - box5331.bluehost.com
Type: - PTR
Comments
Filter Hosts
Search
ENG
IN
19:49
27-11-2025

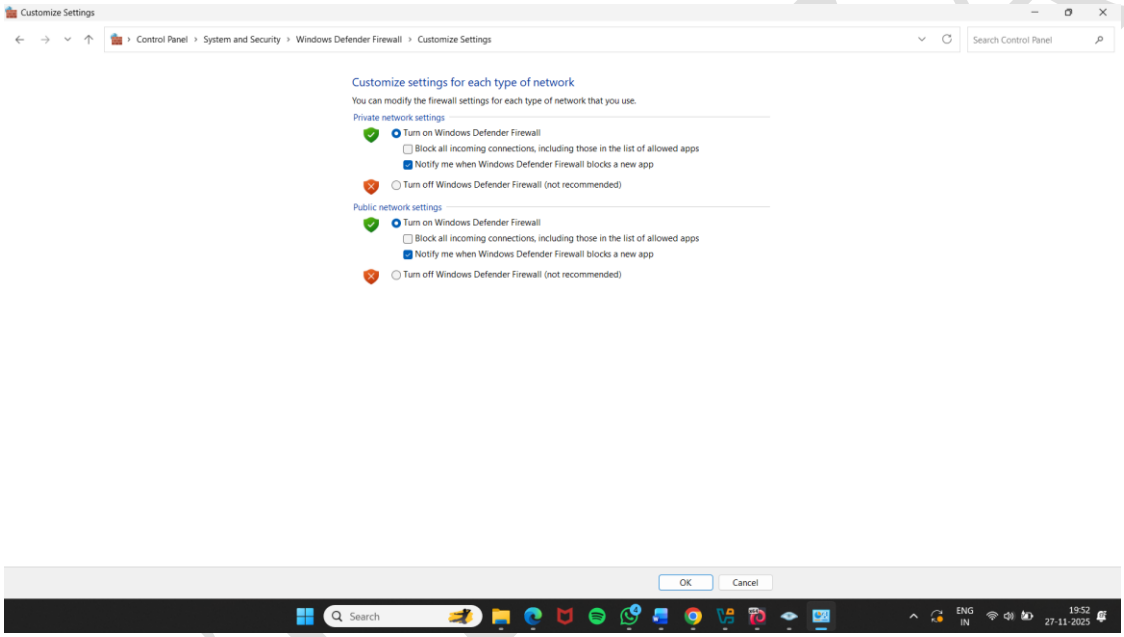
Fig(10)


Zenmap
Scan Tools Profile Help
Target: 162.241.216.11
Profile:
Scan
Cancel
Command: nmap -sT -v 162.241.216.11
Hosts
Services
OS
Host
box5331.bluehc
Nmap Output
Ports / Hosts
Topology
Host Details
Scans
Status
Command
Unsaved
nmap -sT -v 162.241.216.11
Filter Hosts
Append Scan
Remove Scan
Cancel Scan
Search
ENG
IN
19:49
27-11-2025

Fig(11)

In this sub-task, we have performed a **Stealth scan/TCP half-open scan, Xmas scan, TCP Maimon scan,** and **ACK flag probe scan** on a firewall-enabled machine.

Step: Navigate to Control Panel System and Security → Windows Defender Firewall → Turn Windows Defender Firewall on or off, enable Windows Firewall and click OK, as shown.



Fig(12)

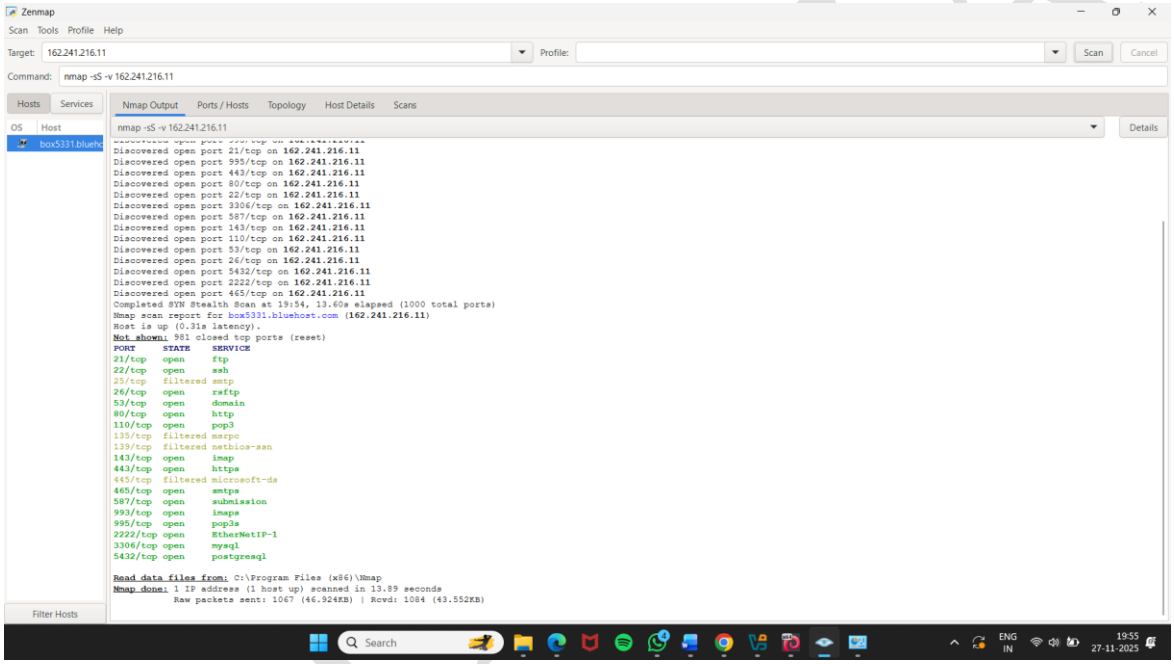
Test/Tool Shown:

Zmap (Zenmap GUI) – Nmap Stealth SYN Scan (-sS).

Step Performed:

A verbose SYN scan was executed to discover open TCP ports without completing full connections. (nmap -sS -v 162.241.216.11)

Output:



Fig(13)

Output:

The scan results appear, displaying all open TCP ports and services running on the target machine, as shown in the screenshot.

Note: The stealth scan involves resetting the TCP connection between the client and server abruptly before completion of three-way handshake signals, and hence leaving the connection half-open. This scanning technique can be used to bypass firewall rules, logging mechanisms, and hide under network traffic.

Test/Tool Shown:

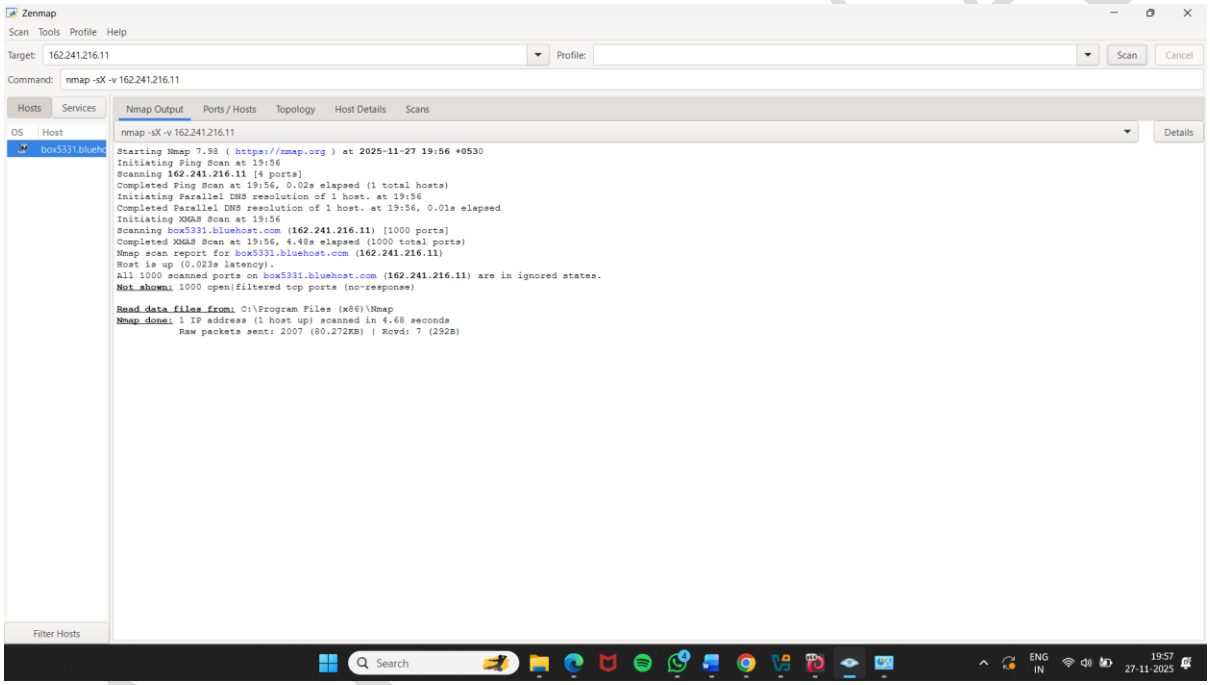
Zenmap – XMAS scan (-sX).

Step Performed:

An XMAS scan was run to probe firewall behavior using FIN, URG, PSH flags.

(nmap -sX -v 162.241.216.11)

Output:



Fig(14)

- All 1000 ports are in ignored states
- Not shown: 1000 open|filtered tcp ports (no-response)

Note: Xmas scan sends a TCP frame to a target system with FIN, URG, and PUSH flags set. If the target has opened the port, then you will receive no response from the target system. If the target has closed the port, then you will receive a target system reply with an RST.

Test/Tool Shown:

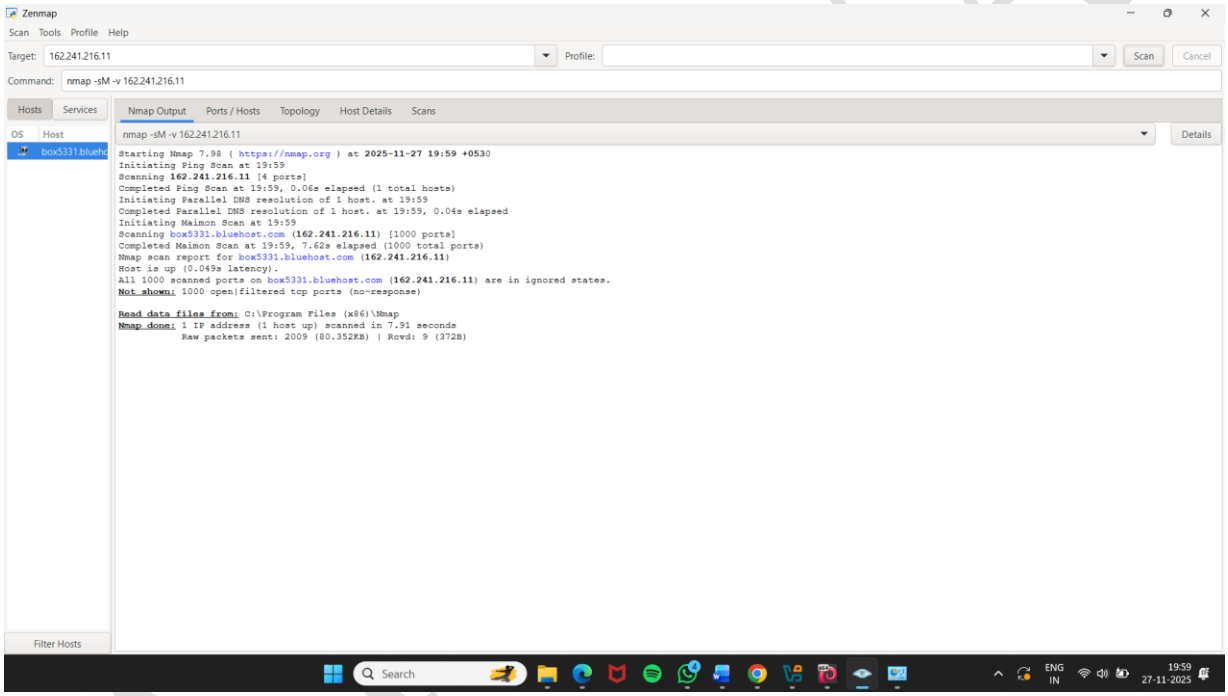
Zenmap – Maimon Scan (-sM).

Step Performed:

Executed a Maimon TCP FIN/ACK probe to detect filtering anomalies.

(nmap -sM -v 162.241.216.11)

Output:



Fig(15)

- All 1000 ports are in ignored states
- Server gave no responses.

Note: In the TCP Maimon scan, a FIN/ACK probe is sent to the target; if there is no response, then the port is Open | Filtered, but if the RST packet is sent as a response, then the port is closed.

Test/Tool Shown:

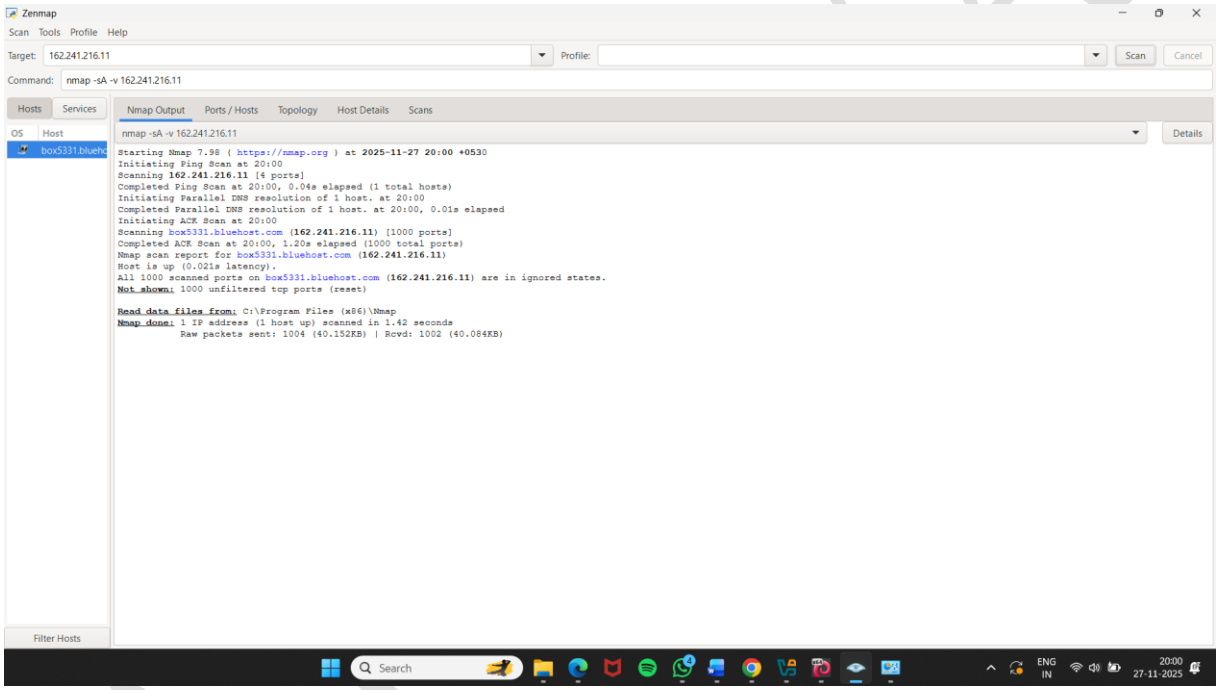
Zenmap – ACK scan (-sA).

Step Performed:

An ACK scan was run to detect firewall rules, not open ports.

(nmap -sA -v 162.241.216.11)

Output:



Fig(16)

- **All 1000 ports: unfiltered**
- Host responds with RST to ACK probes.

Note: The ACK flag probe scan sends an ACK probe packet with a random sequence number; no response implies that the port is filtered (stateful firewall is present), and an RST response means that the port is not filtered.

Test/Tool Shown:

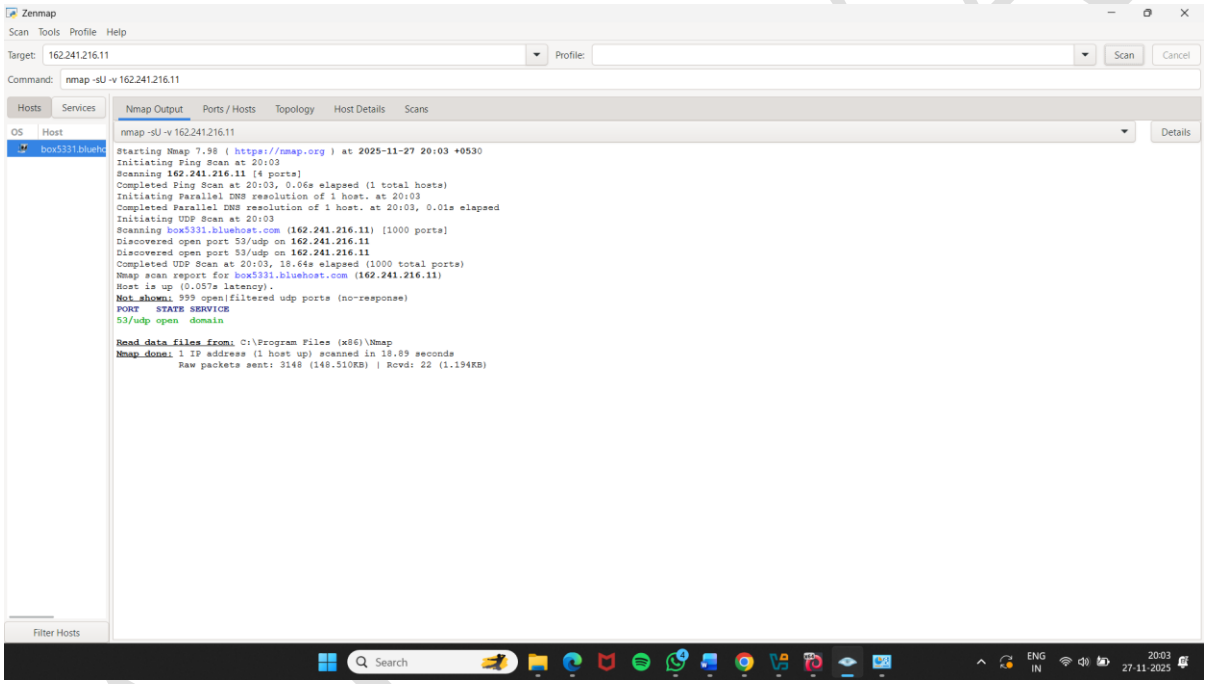
Zenmap – UDP scan (-sU).

Step Performed:

A verbose UDP scan was executed on all 1000 UDP ports.

(nmap -sU -v 162.241.216.11)

Output:



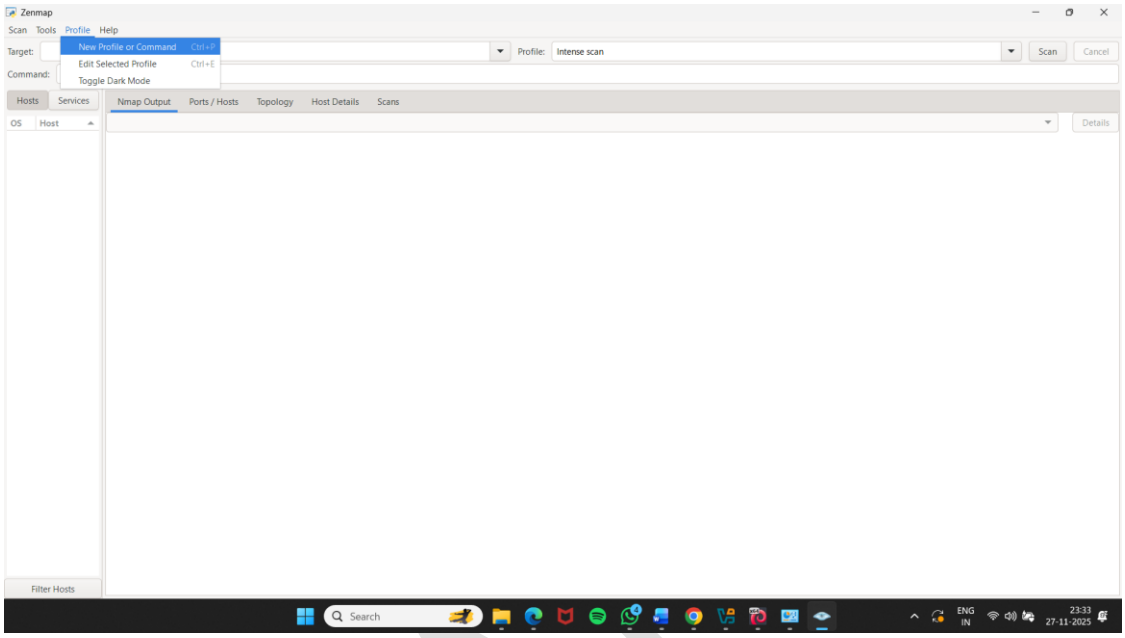
Fig(17)

- Only **port 53/udp (DNS)** is open.
- All others are **open|filtered** (typical for UDP).
- "No response" means UDP packets are dropped silently.

Note: The UDP scan uses UDP protocol instead of the TCP. There is no three-way handshake for the UDP scan. It sends UDP packets to the target host; no response means that the port is open. If the port is closed, an ICMP port unreachable message is received.

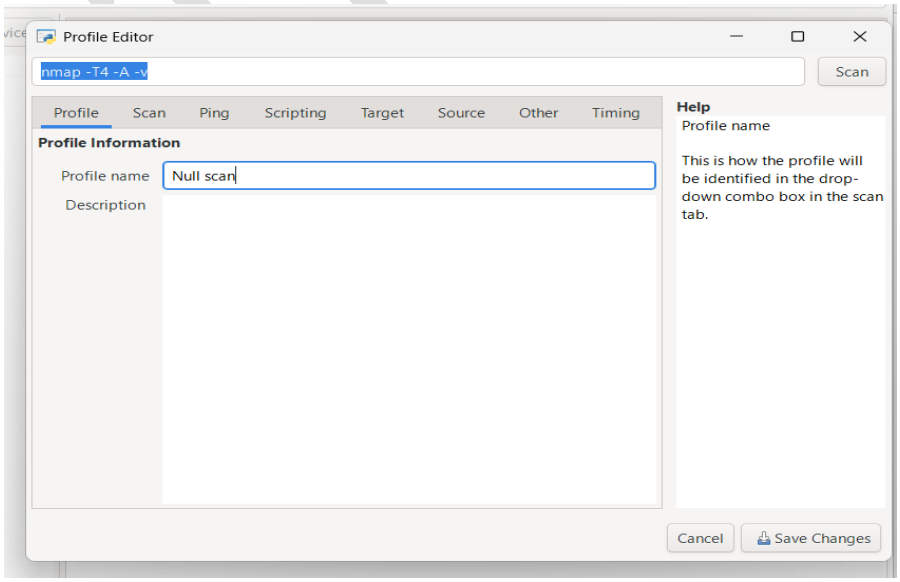
Creating a Scan Profile:

- Open Nmap - Zenmap GUI app.
- To create a scan profile; click Profile → New Profile or Command.



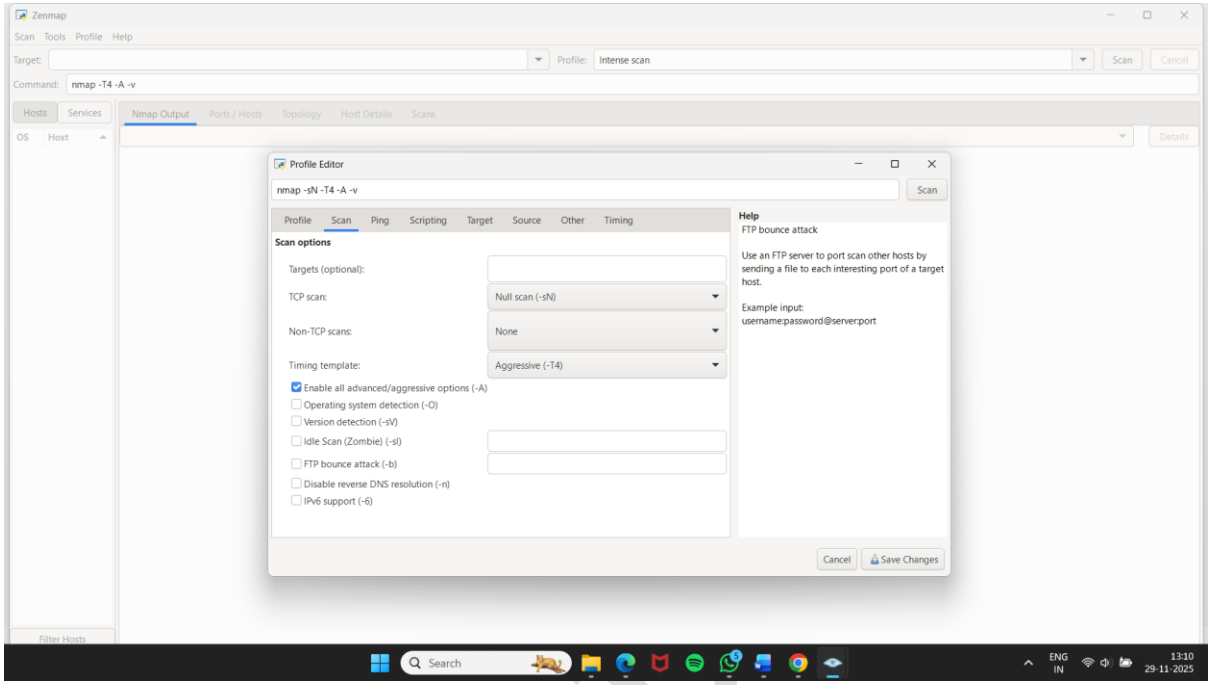
Fig(18)

- The Profile Editor window appears. In the Profile tab, under the Profile Information section, input a profile name (Null Scan) into the Profile name field.



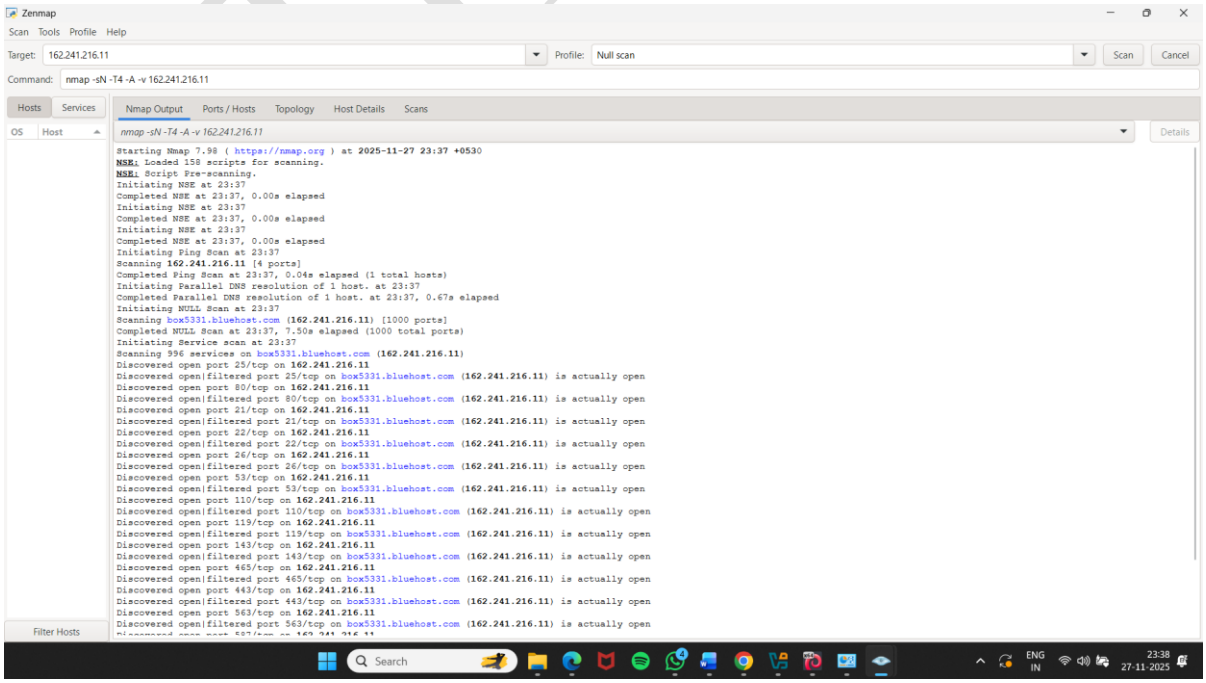
Fig(19)

- click the Scan tab and select the scan option (here, Null scan (-sN)) from the TCP scan drop-down list.



Fig(20)

- This will create a new profile and will thus be added to the profile list.
- In the main window of Zenmap, enter the target IP address (162.241.216.11) in the Target field to scan. Select the Null Scan profile, which you created from the Profile drop-down list, and then click Scan.



Fig(21)

Test/Tool Shown:

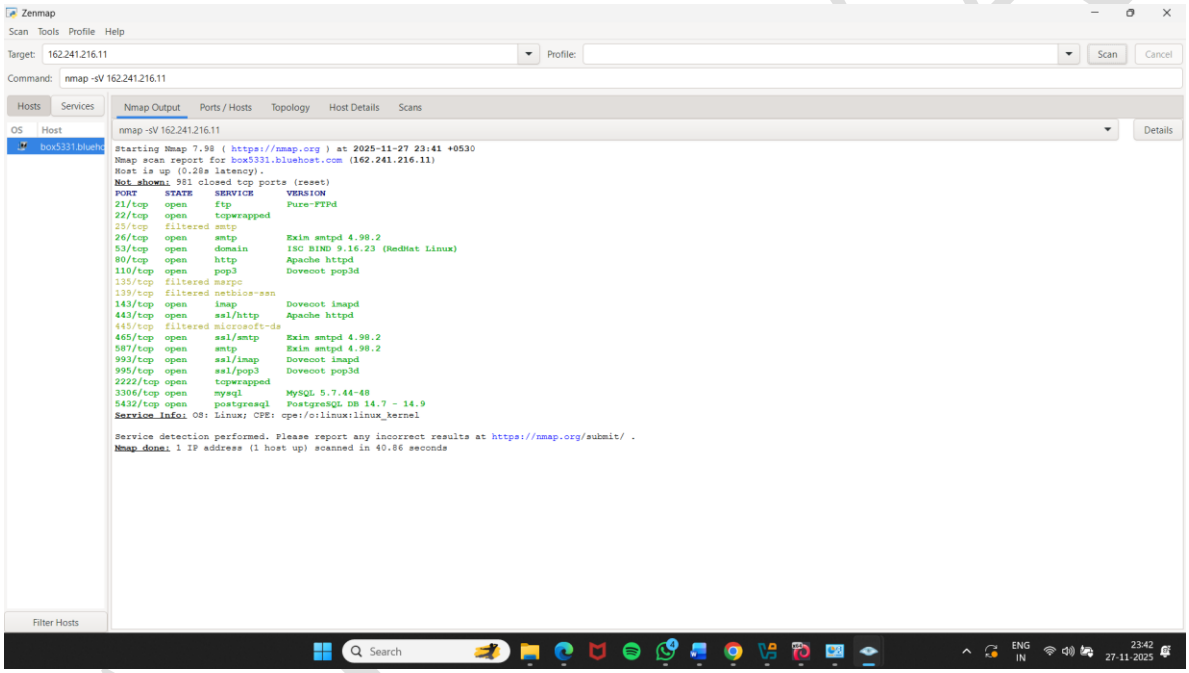
Zenmap (Nmap GUI) — Version Detection Scan (-sV).

Step Performed:

A service/version detection scan was run on the target host:

(nmap -sV 162.241.216.11)

Output:



Fig(22)

- Open Ports & Services Identified.
- 991 TCP ports were closed.
- Version detection succeeded on most open ports.
- Host appears to be a Bluehost shared hosting server.

Note: -A: enables aggressive scan. The aggressive scan option supports OS detection (-O), version scanning (-sV), script scanning (-sC), and traceroute (--traceroute). You should not use -A against target networks without permission.

OS DISCOVERY

OS discovery (or OS fingerprinting) is the process of finding out which operating system a remote machine is running — for example:

- Windows 10
- Windows Server
- Linux (Ubuntu, CentOS, Debian, etc.)
- macOS
- Network OS (Cisco, Juniper, MikroTik)

OR

Banner grabbing, or OS fingerprinting, is a method used to determine the OS that is running on a remote target system.

Kali Linux uses tools like Nmap, Netcat, and TTL analysis to make an educated guess about the OS based on how the target responds to network packets.

There are two types of OS discovery or banner grabbing techniques:

Active Banner Grabbing Specially crafted packets are sent to the remote OS, and the responses are noted, which are then compared with a database to determine the OS. Responses from different OSes vary, because of differences in the TCP/IP stack implementation.

Passive Banner Grabbing This depends on the differential implementation of the stack and the various ways an OS responds to packets. Passive banner grabbing includes banner grabbing from error messages, sniffing the network traffic, and banner grabbing from page extensions.

Test/Tool Shown:

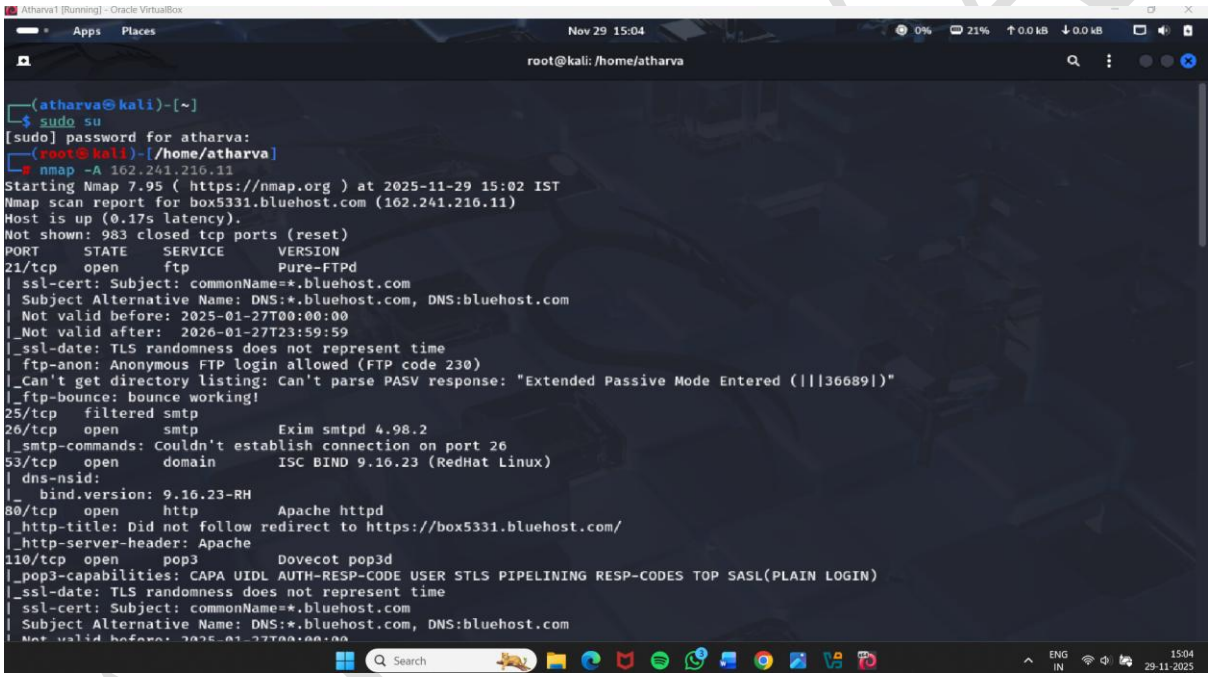
Nmap (Aggressive Scan -A)

Step Performed:

Executed: nmap -A 162.241.216.11

This performs: OS detection, Version detection, Script scanning, Traceroute.

Output:



Fig(23)

- Open ports & Services Detected
- The server is a Bluehost shared hosting machine**.**
- Many mail services (POP/IMAP/SMTP) are running.
- Both MySQL and PostgreSQL are publicly accessible, which is unsafe.
- FTP is active.

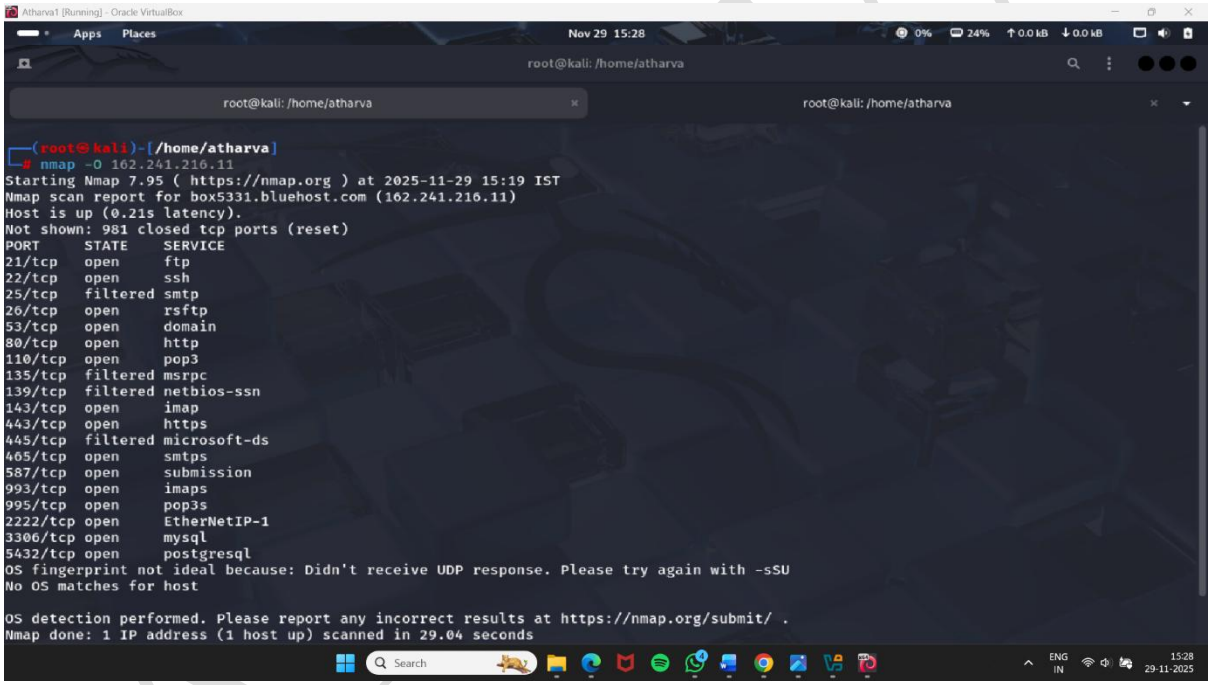
Test/Tool Shown:

Nmap OS Detection Scan (-O)

Step Performed:

Executed: nmap -O 162.241.216.11
This attempts to identify the OS via network fingerprinting

Output:



Fig(24)

- Open Ports Detected (same as aggressive scan, but without version banners)
- OS detection failed because the host's firewall blocks UDP probes.
- Filtered ports (25,135,139,445) indicate firewall protection is active.

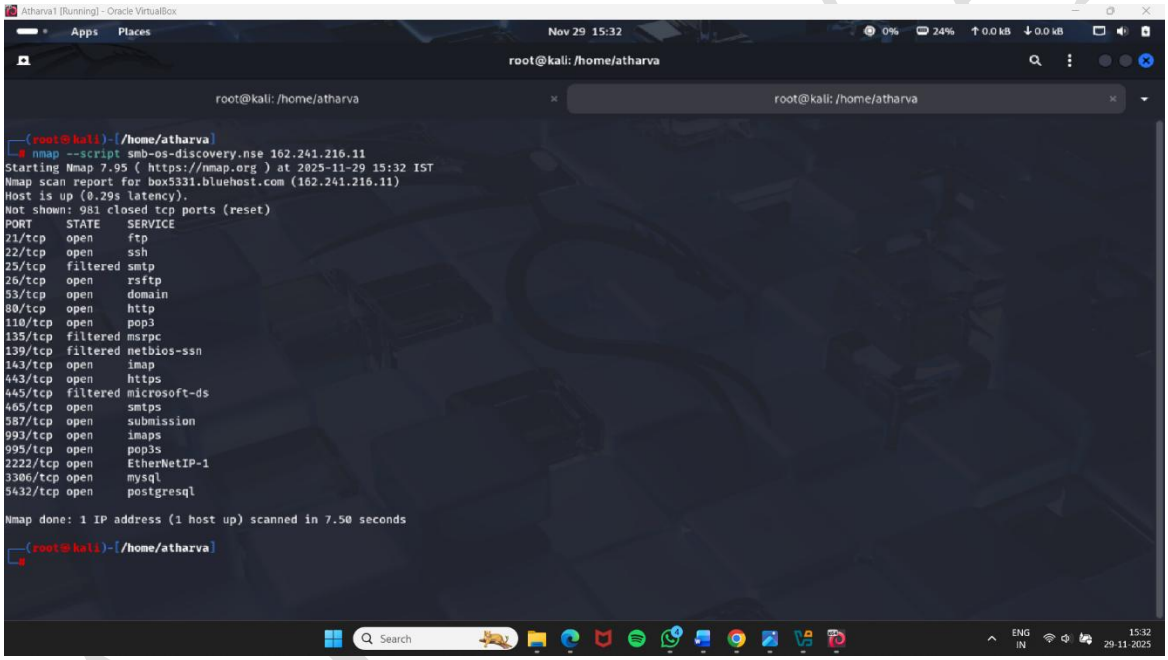
Test/Tool Shown:

nmap --script smb-os-discovery.nse

Step Performed:

Executed the command to detect SMB-related OS information, to enumerate Windows-style SMB shares or metadata, to check SMB version or domain/workgroup details.

Output:



Fig(25)

- Ports Detected.
- The script returned NO SMB information because port 445 (SMB) is filtered, not open.

Scan Beyond IDS and Firewall

Scanning beyond IDS and firewall is a process of sending intended packets to the target system in order to exploit IDS/firewall limitations.

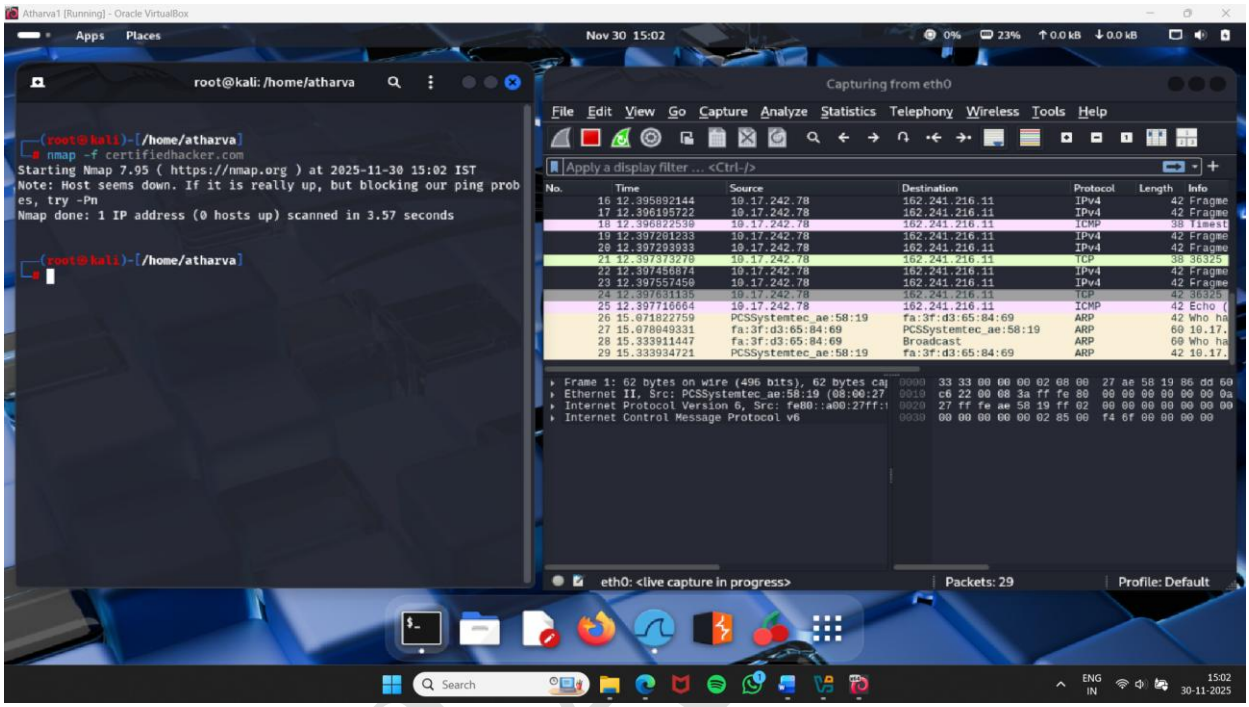
An Intrusion Detection System (IDS) and firewall are the security mechanisms intended to prevent an unauthorized person from accessing a network. However, even IDSs and firewalls have some security limitations. Firewalls and IDSs intend to avoid malicious traffic (packets) from entering into a network, but certain techniques can be used to send intended packets to the target and evade IDSs/firewalls.

Techniques to evade IDS/firewall:

- Packet Fragmentation
- Source Routing
- Source Port Manipulation
- IP Address Decoy
- IP Address Spoofing
- Creating Custom Packets
- Sending Bad Checksums
- Proxy Servers
- Randomizing Host Order

Test/Tool Shown:

1)Nmap (normal scan) + Wireshark packet capture



Fig(26)

Step Performed:

Command executed: nmap -f certifiedhacker.com
Fragmentation scan attempted.

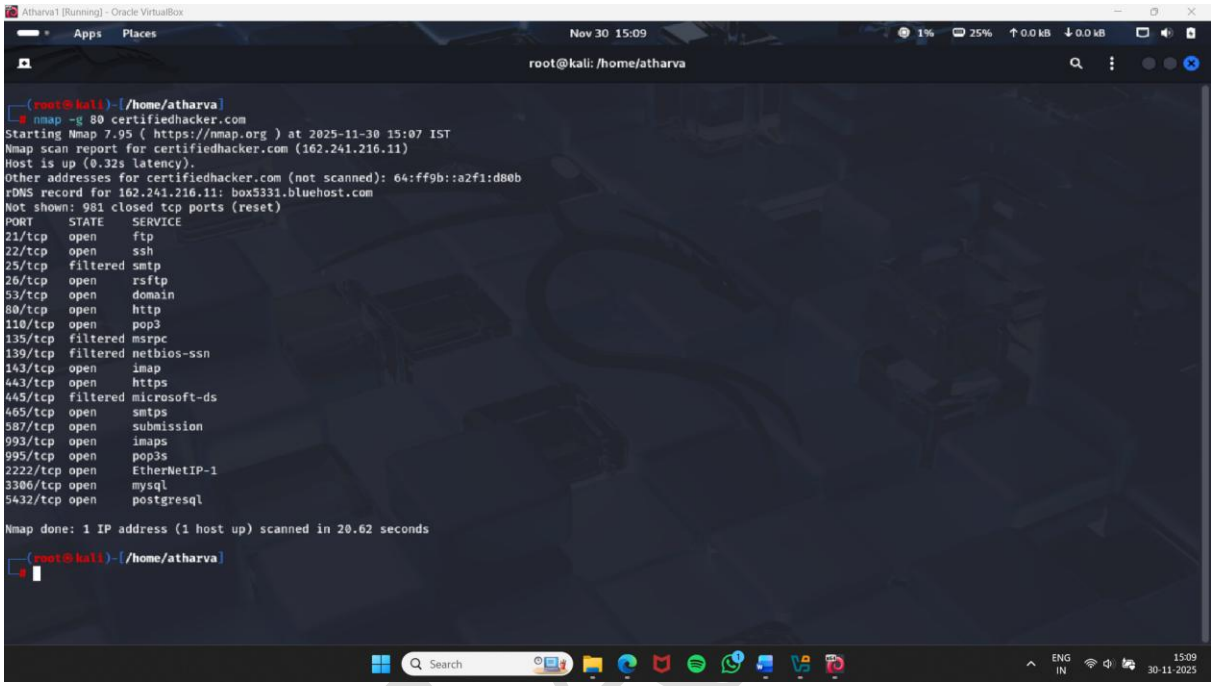
Output:

- Wireshark shows outbound packets from 10.17.242.78 → 162.241.216.11, but no meaningful replies.
- The target host blocked fragmented packets (-f) or ICMP echo requests.
- Host is alive, but firewall rejects Nmap's discovery probes.

Note: Packet fragmentation refers to the splitting of a probe packet into several smaller packets (fragments) while sending it to a network.

Test/Tool Shown:

2)Nmap verbose + packet tracing.



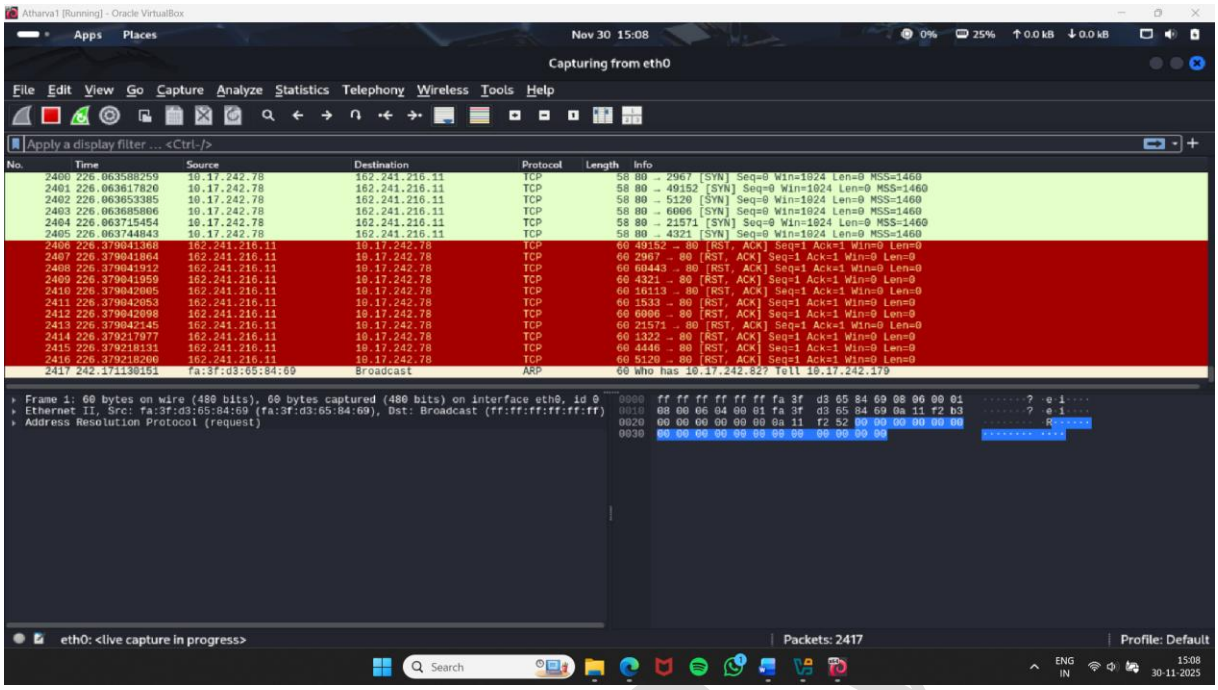
Fig(27)

Step Performed:

Command executed: nmap -v -g 80 certifiedhacker.com
(Source port spoofed as port 80)

Output:

- Nmap discovered multiple open ports:
- 80, 993, 110, 21, 995, 443, 587, 143, 3306, 26, 2222, 5432 etc.
- Many shared hosting servers allow traffic from port 80 due to “firewall trust” policy.

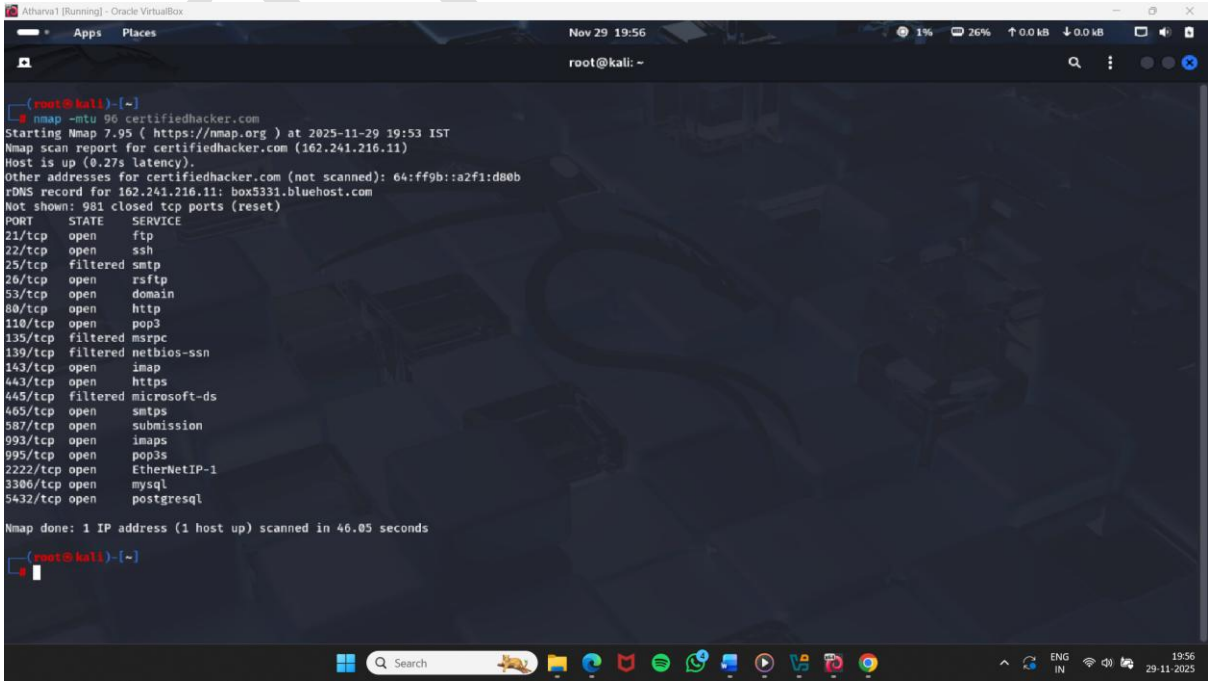


Fig(28)

Note: Source port manipulation refers to manipulating actual port numbers with common port numbers to evade IDS/firewall: this is useful when the firewall is configured to allow packets from well-known ports like HTTP, DNS, FTP, etc.

Test/Tool Shown:

3)Nmap with fragmentation MTU



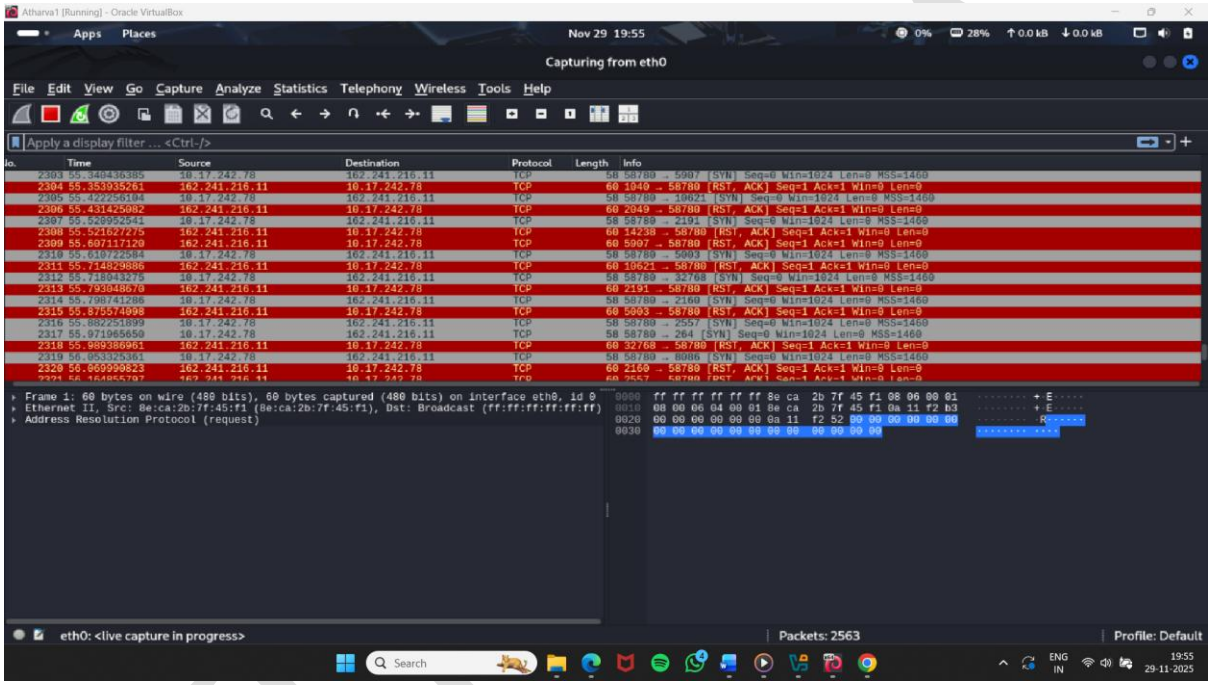
Fig(29)

Step Performed:

Command executed: nmap --mtu 96 certifiedhacker.com

Output:

- Host responded normally.
- Same ports open as previous scan.
- Smaller packets are transmitted instead of sending one complete packet at a time.

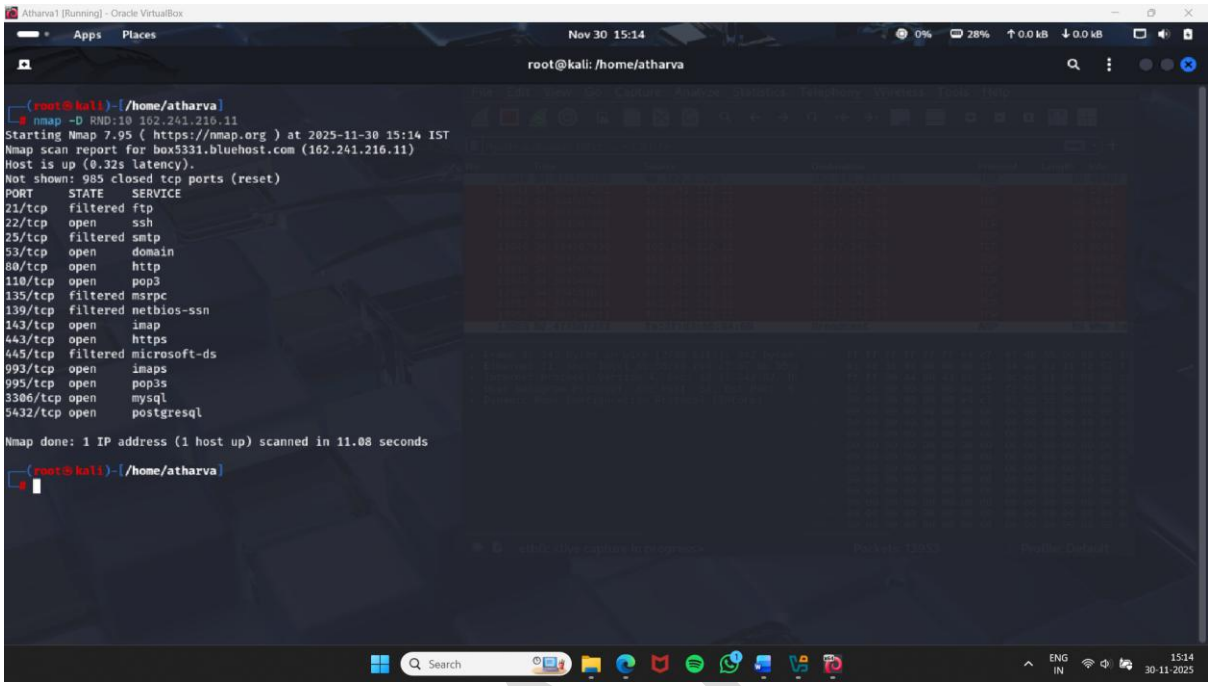


Fig(30)

Note: Using MTU, smaller packets are transmitted instead of sending one complete packet at a time. This technique evades the filtering and detection mechanism enabled in the target machine.

Test/Tool Shown:

4)Nmap decoy scan



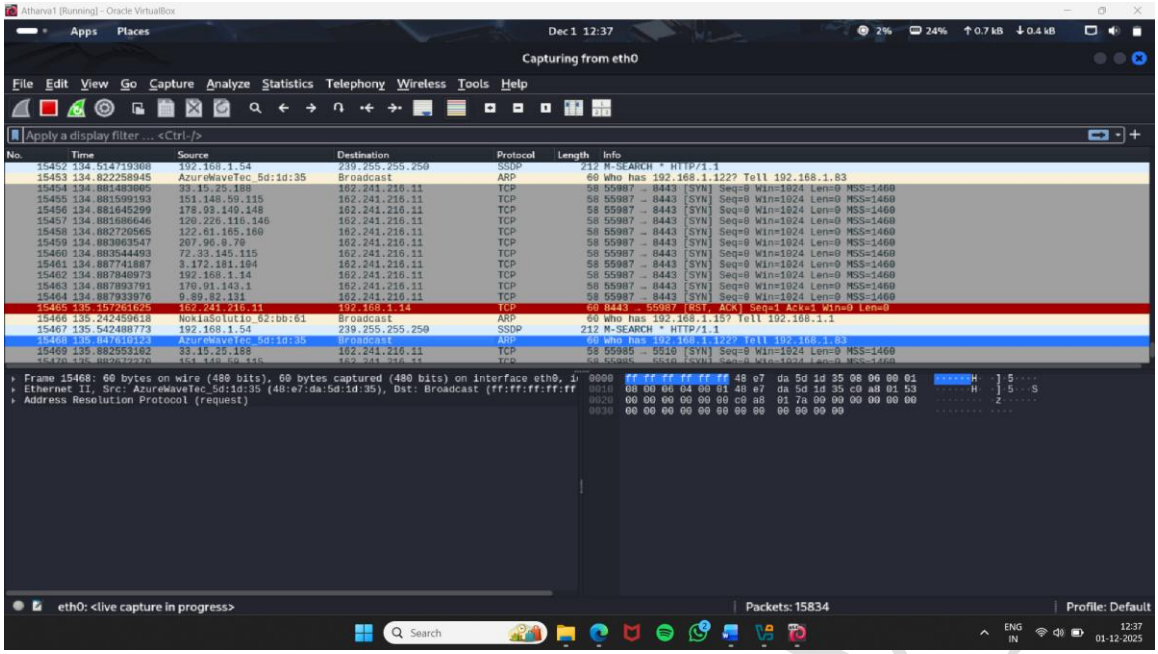
Fig(31)

Step Performed:

nmap -D RND:10 162.241.216.11

Output:

- Decoy scan completed.
- Same ports appear open.
- Nmap automatically generated a random number of decoys for the scan and randomly positions the real IP address between the decoy IP addresses.

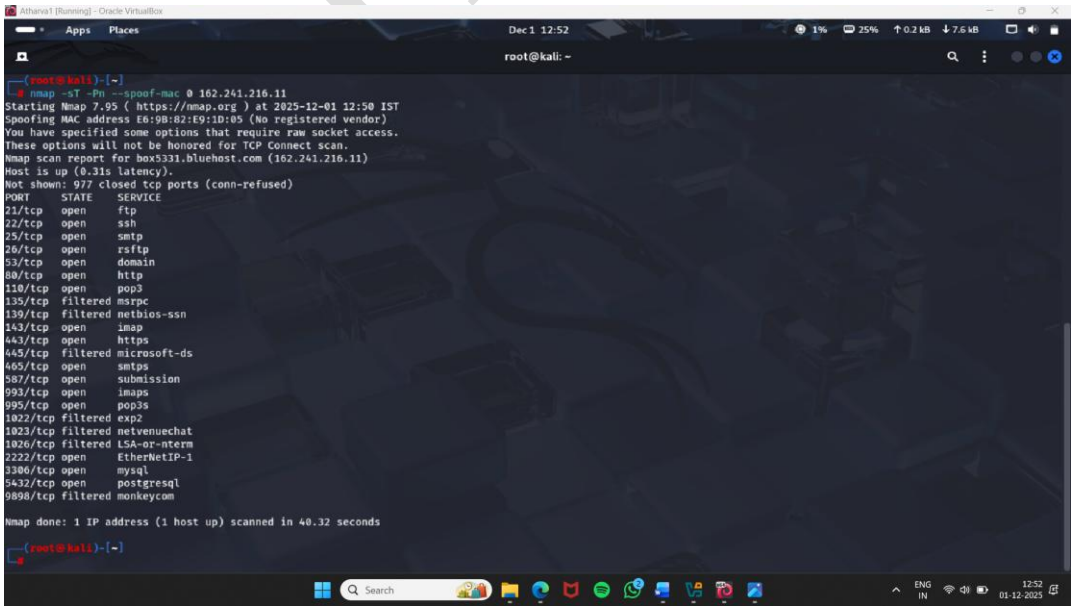


Fig(32)

Note: This technique makes it difficult for the IDS/firewall to determine which IP address was actually scanning the network and which IP addresses were decoys and you can observe the packets displaying the multiple IP addresses in the source section, as shown in the screenshot.

5) **Test/Tool Shown:**

Nmap 7.95 – TCP Connect Scan (-sT), no ping (-Pn), random MAC address spoofing.



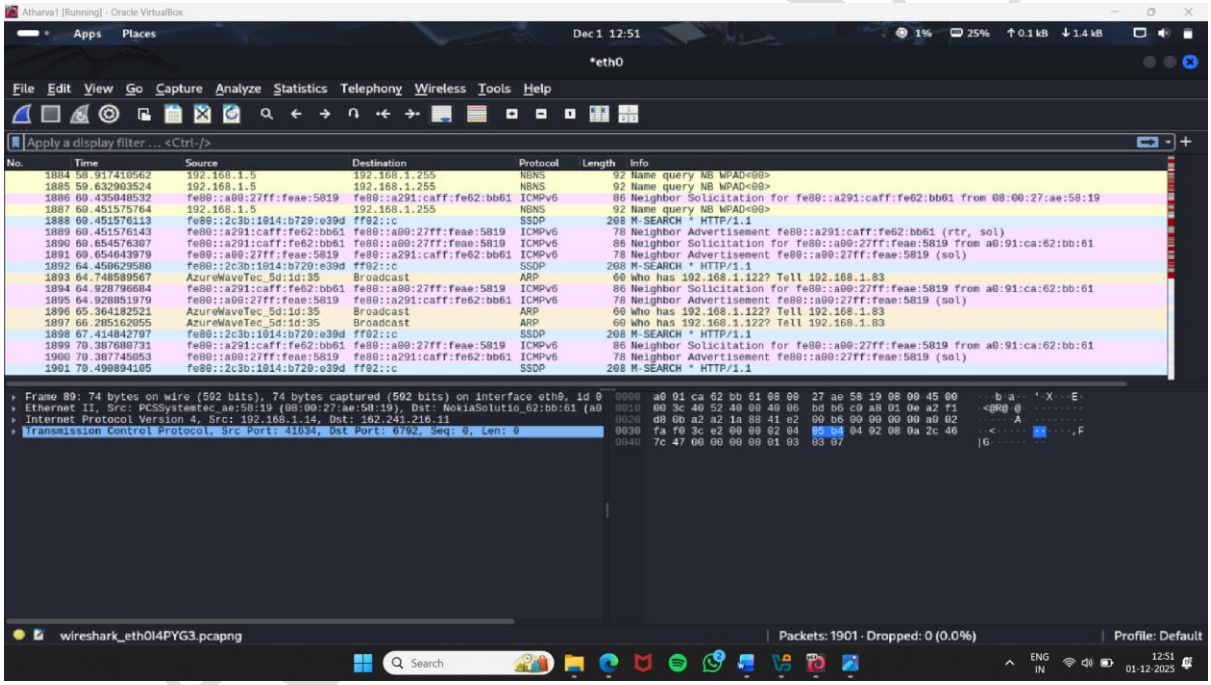
Fig(33)

Step Performed:

Executed a full TCP Connect scan on target **162.241.216.11**, attempting to spoof MAC.

Output:

- Multiple ports shown open: 21,22,25,53,80,110,135,143,443,445,587,993,995,2082,2222,3306,..
- Some ports filtered: 1022,1023,1026,1433,9898...
- Spoofing a MAC address with the MAC address of a legitimate user on the network.



Fig(34)

Note: MAC address spoofing technique involves spoofing a MAC address with the MAC address of a legitimate user on the network. This technique allows you to send request packets to the targeted machine/network pretending to be a legitimate host.

Additional Tools

hping3

hping3 is a network tool that's able to send custom TCP/IP packets and to display target replies like ping program does with ICMP replies .

What is hping3?

hping3 is an advanced network packet crafting tool that allows you to generate and manipulate packets for various network protocols, including TCP, UDP, ICMP, and RAW-IP. It is primarily used for network security testing, firewall testing, advanced network diagnostics, and even penetration testing. Think of it as a more powerful version of the traditional ping command, but with much more control over the packets being sent.

Main Features of hping3:

- TCP/IP Packet Manipulation: You can craft and send custom TCP, UDP, ICMP, and RAW-IP packets.
- Port Scanning: hping3 can perform basic to advanced port scanning.
- Firewall Testing: Test firewall rules by crafting specific packets and analyzing the responses.
- Network Trace Routing: Similar to traceroute, you can track the path packets take across the network.
- Advanced Packet Crafting: You have control over packet headers, flags (SYN, ACK, FIN), TTL, window size, and more.
- Flooding and DoS Simulation: Simulate high packet loads to test the resilience of servers.
- Data Transfer Testing: Send raw data over different protocols to test throughput and reliability.
- OS Fingerprinting: Infer the operating system of a remote host by analyzing packet responses.

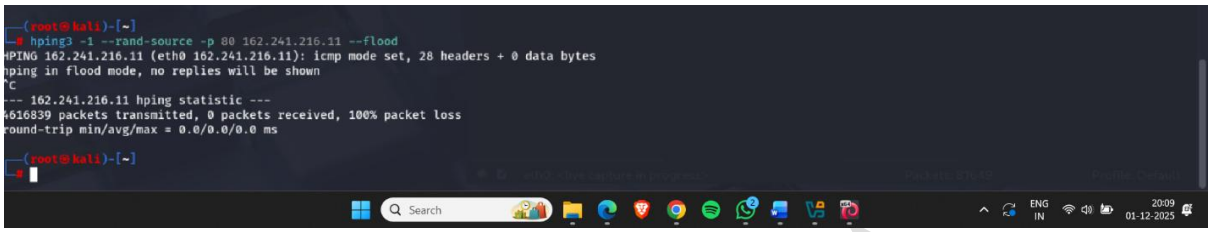
Basic Syntax of hping3: hping3 [options] [host]

Legal and Ethical Considerations:

- hping3 is a powerful tool often used in network testing and security auditing.
- Unauthorized usage on networks you do not own or have permission to test is illegal and violates cybersecurity laws.

Test/Tool Shown:

1)hping3 in ICMP mode (-1) with random source addresses.



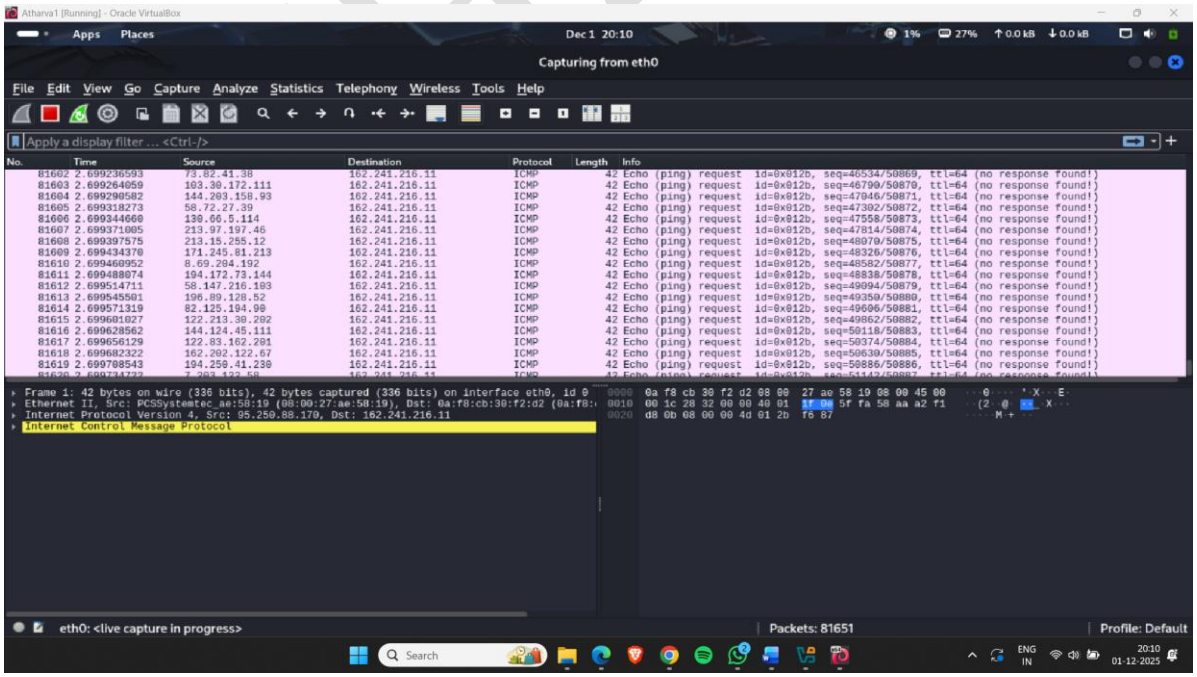
Fig(35)

Step Performed:

Command used: hping3 -1 --rand-source -p 80 162.241.216.11 –flood

Output:

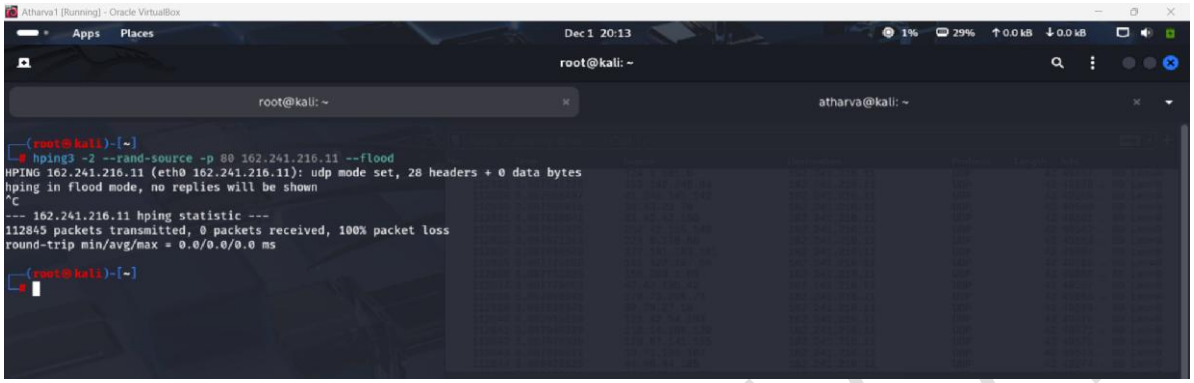
- ICMP mode initialized.
- No replies shown (expected in flood mode).
- Summary: **4616839 packets transmitted, 0 received, 100% loss**.
- Sent a massive, high-speed volume of ICMP Echo Requests (-1) to the target (162.241.216.11) while masking the true source by using randomly spoofed IP addresses (--rand-source).



Fig(36)

Test/Tool Shown:

2)hping3 – UDP mode (-2).



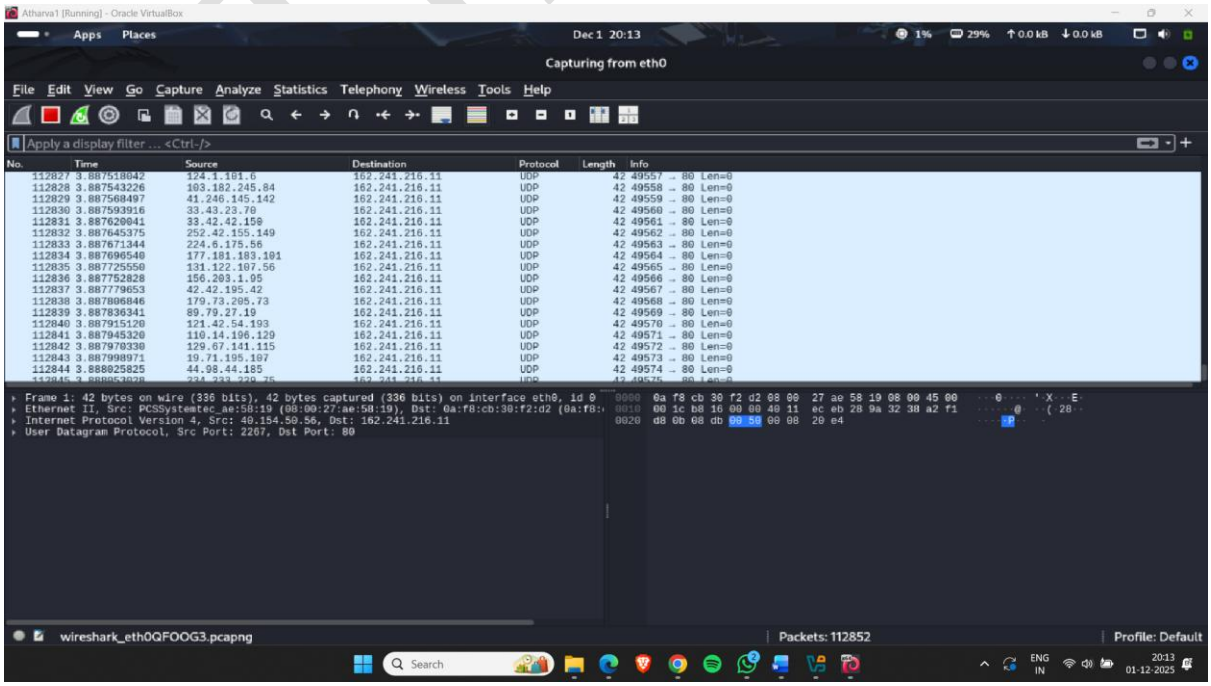
Fig(37)

Step Performed:

Command used: hping3 -2 --rand-source -p 80 162.241.216.11 –flood

Output:

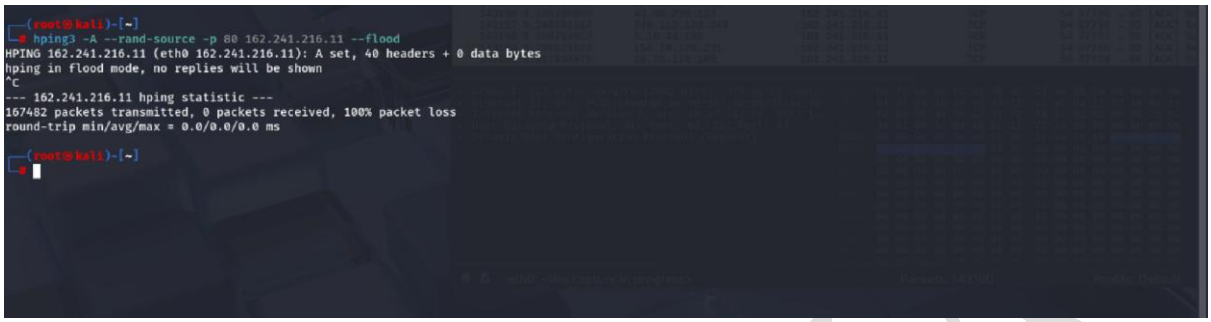
- UDP mode activated.
- 112845 packets transmitted.
- The terminal confirms it's sending packets in "udp mode" and "pinging in flood mode."



Fig(38)

Test/Tool Shown:

3)hping3 – TCP ACK mode (-A)



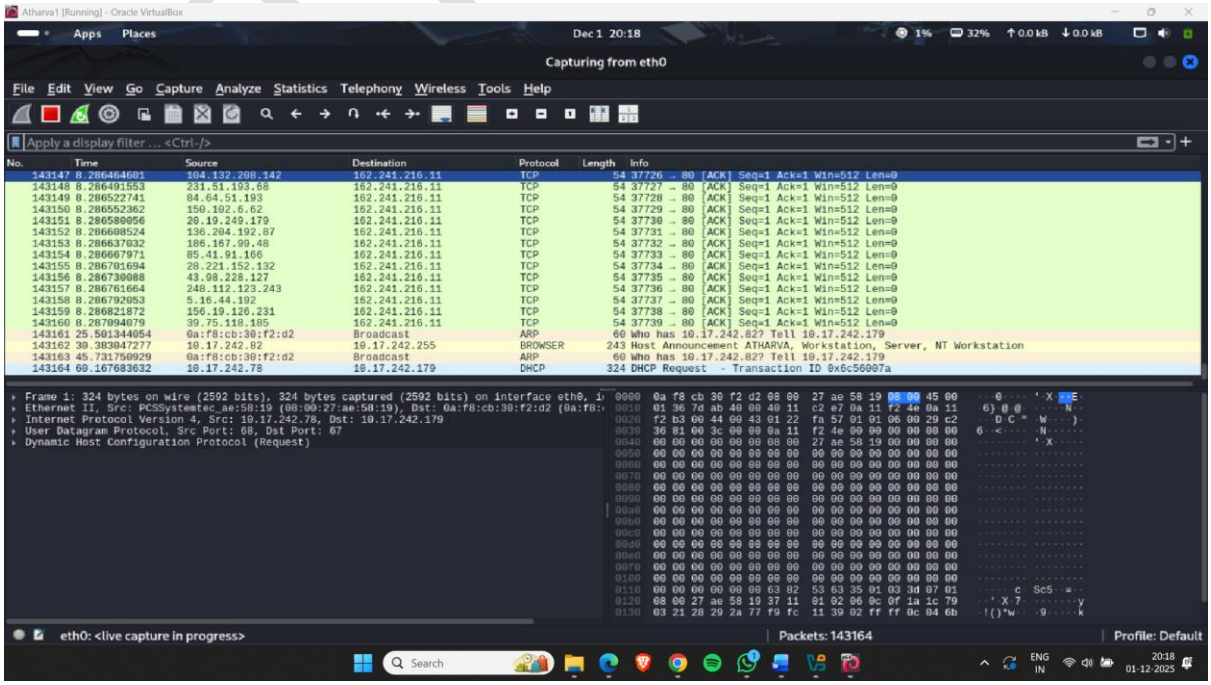
Fig(39)

Step Performed:

Command used: hping3 -A --rand-source -p 80 162.241.216.11 –flood

Output:

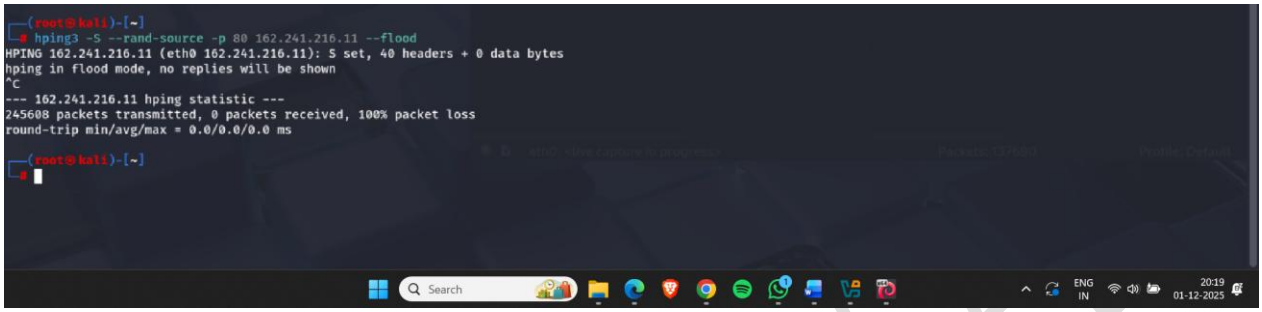
- TCP ACK packets sent.
- 167482 packets transmitted, 0 received
- Massive stream of TCP packets with the ACK flag set to the target on a specific port (like port 80).



Fig(40)

Test/Tool Shown:

4)hping3 – TCP SYN mode (-S)



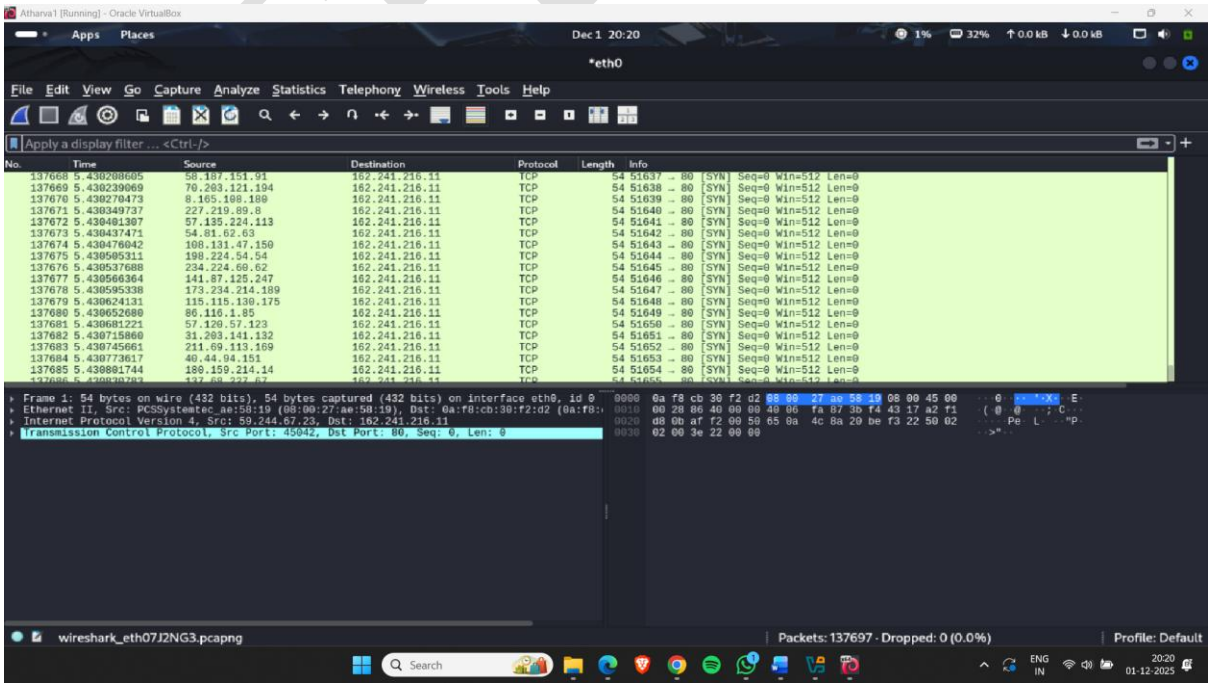
Fig(41)

Step Performed:

Command used: hping3 -S --rand-source -p 80 162.241.216.11 –flood

Output:

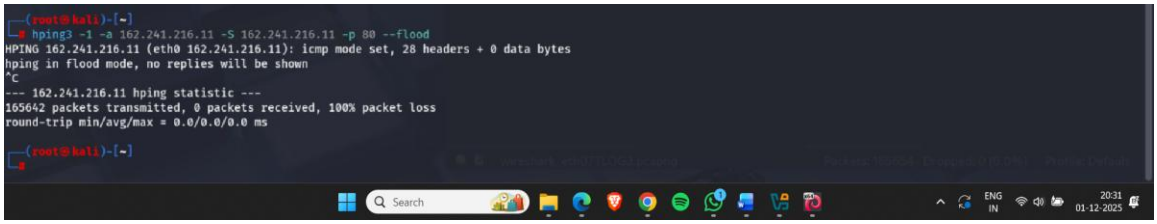
- SYN packets sent in flood mode.
- 245608 transmitted, 100% loss.
- [SYN] flag set, directed at port 80 on the target.



Fig(42)

Test/Tool Shown:

5)hping3 – ICMP + spoofed source (-a)



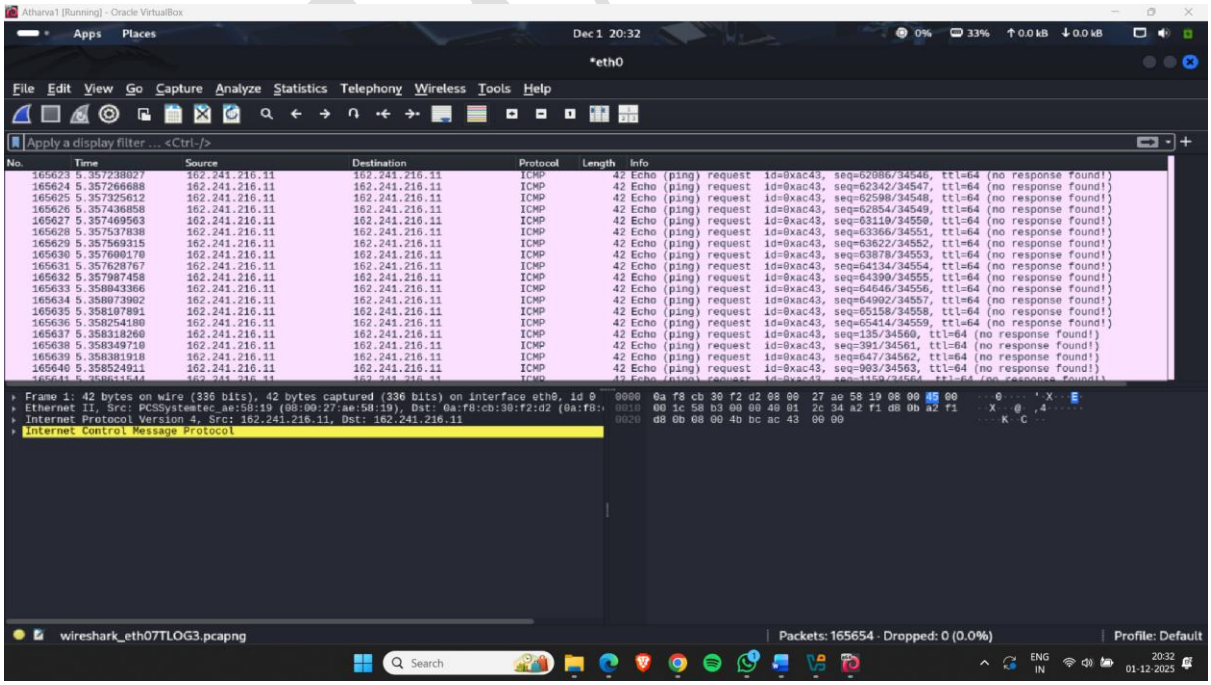
Fig(43)

Step Performed:

Command used: hping3 -1 -a 162.241.216.11 -S 162.241.216.11 -p 80 –flood

Observed Output:

- ICMP packets being sent.
- 165642 packets transmitted.
- -a 162.241.216.11 → Spoofed source IP
- -S 162.241.216.11 → Target IP (packets sent to own machine)



Fig(43)

Metasploit:

Metasploit is one of the most important and powerful tools in cybersecurity. It is used for penetration testing, exploit development, vulnerability assessment, and security research. Below is a clear, structured explanation that helps beginners and intermediate learners.

What is Metasploit?

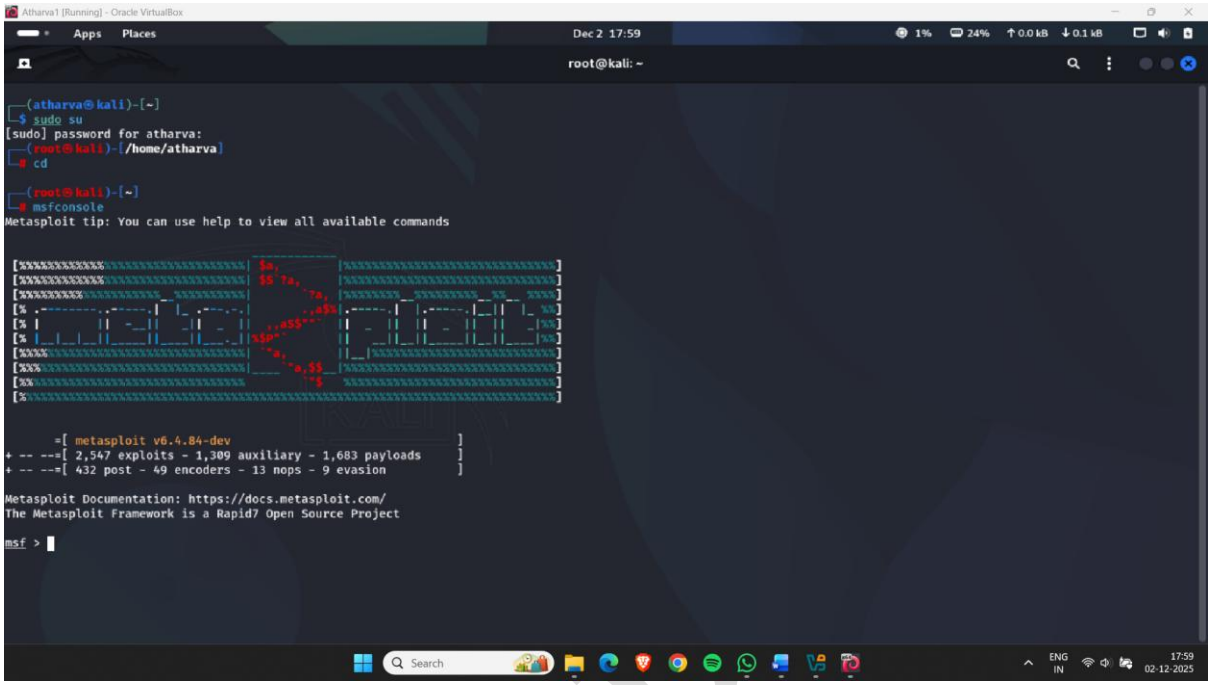
Metasploit is an open-source penetration testing framework used to:

- Find vulnerabilities
- Exploit vulnerabilities
- Test security defenses
- Create payloads
- Run post-exploitation modules
- Perform red-team assessments

Why Metasploit Is Important in Cybersecurity?

- Automates exploitation
- Contains huge exploit and payload library
- Helps professionals learn real-world attack techniques
- Supports post-exploitation and pivoting
- Useful for practical cybersecurity training

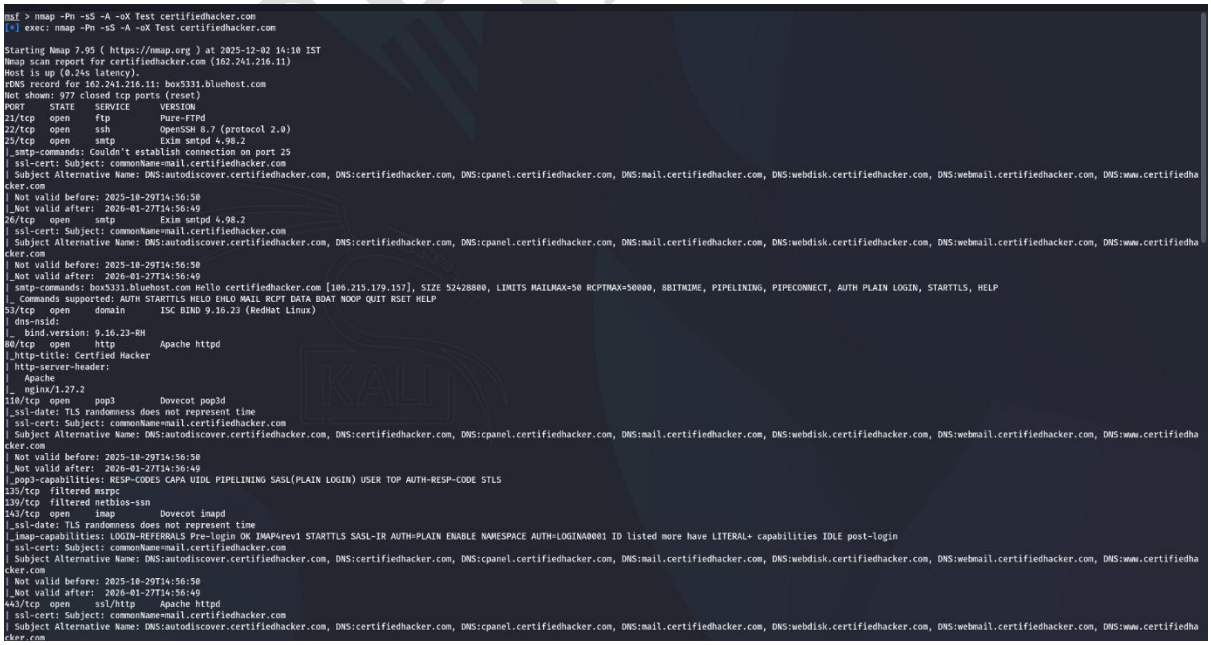
Step1: Execute command msfconsole to launch Metasploit.



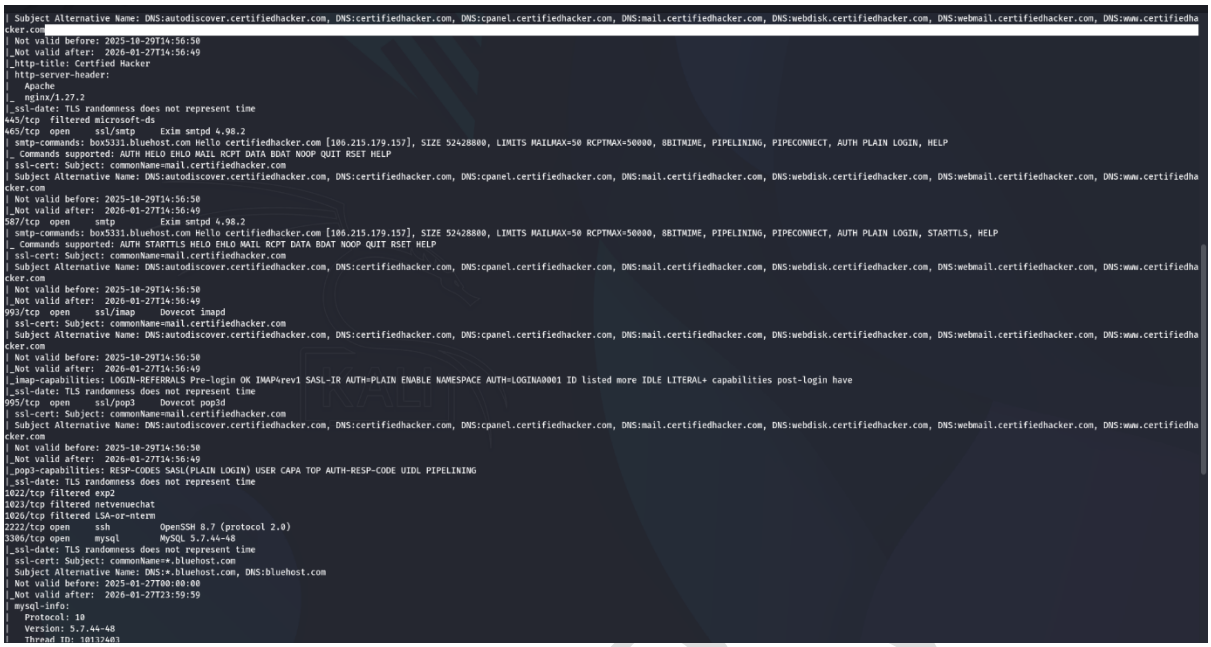
Fig(44)

Step2: An msf command line appears. Type nmap -Pn -sS -A -oX Test (target) and press Enter to scan .

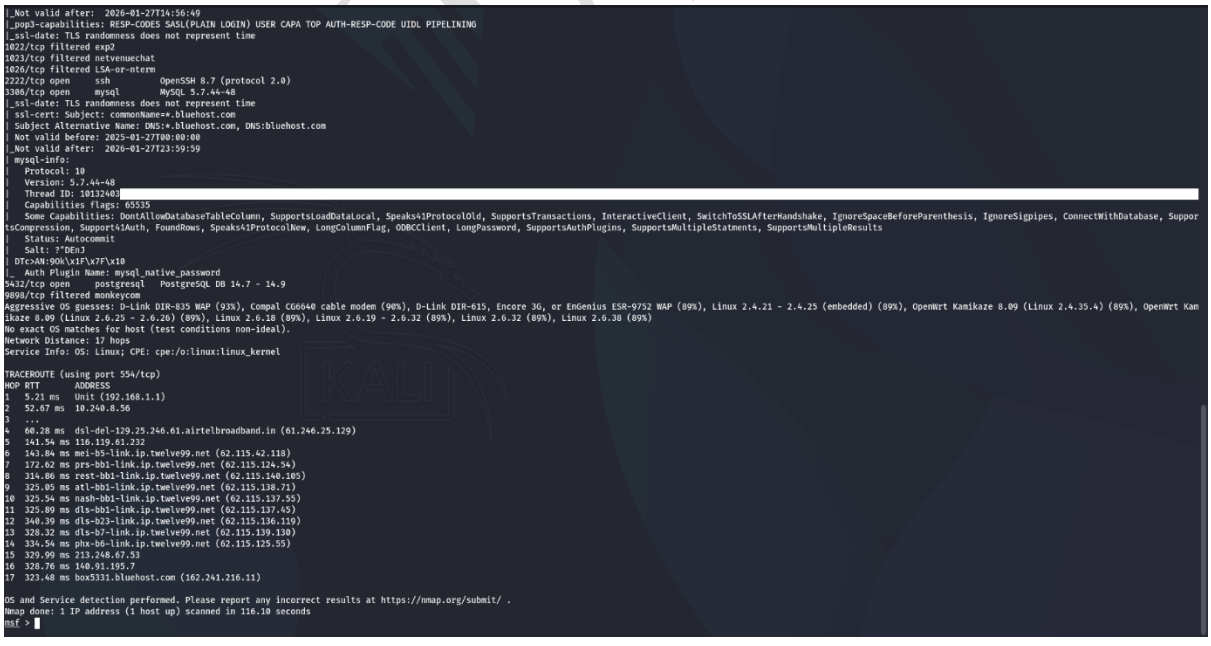
After the scan completes, Nmap displays the host information in the target network along with open ports, service and OS enumeration.



Fig(45)


| Subject Alternative Name: DNS:autodiscover.certifiedhacker.com, DNS:certifiedhacker.com, DNS:cpanel.certifiedhacker.com, DNS:mail.certifiedhacker.com, DNS:webdisk.certifiedhacker.com, DNS:webmail.certifiedhacker.com, DNS:www.certifiedhacker.com
| Not valid before: 2025-10-29T14:56:50
|_Not valid after: 2026-01-27T14:56:49
|_http-title: Certfied Hacker
| http-server-header:
| Apache
|_ nginx/1.27.2
|_ssl-date: TLS randomness does not represent time
445/tcp filtered microsoft-ds
465/tcp open   ssl/smtp   Exim smtpd 4.98.2
| smtp-commands: box5331.bluehost.com Hello certifiedhacker.com [106.215.179.157], SIZE 52428800, LIMITS MAILMAX=50 RCPTMAX=50000, 8BITMIME, PIPELINING, PIPECONNECT, AUTH PLAIN LOGIN, HELP
|_ Commands supported: AUTH HELO EHLO MAIL RCPT DATA BDAT NOOP QUIT RSET HELP
| ssl-cert: Subject: commonName=mail.certifiedhacker.com
| Subject Alternative Name: DNS:autodiscover.certifiedhacker.com, DNS:certifiedhacker.com, DNS:cpanel.certifiedhacker.com, DNS:mail.certifiedhacker.com, DNS:webdisk.certifiedhacker.com, DNS:webmail.certifiedhacker.com, DNS:www.certifiedhacker.com
| Not valid before: 2025-10-29T14:56:50
|_Not valid after: 2026-01-27T14:56:49
587/tcp open   smtp   Exim smtpd 4.98.2
| smtp-commands: box5331.bluehost.com Hello certifiedhacker.com [106.215.179.157], SIZE 52428800, LIMITS MAILMAX=50 RCPTMAX=50000, 8BITMIME, PIPELINING, PIPECONNECT, AUTH PLAIN LOGIN, STARTTLS, HELP
|_ Commands supported: AUTH STARTTLS HELO EHLO MAIL RCPT DATA BDAT NOOP QUIT RSET HELP
| ssl-cert: Subject: commonName=mail.certifiedhacker.com
| Subject Alternative Name: DNS:autodiscover.certifiedhacker.com, DNS:certifiedhacker.com, DNS:cpanel.certifiedhacker.com, DNS:mail.certifiedhacker.com, DNS:webdisk.certifiedhacker.com, DNS:webmail.certifiedhacker.com, DNS:www.certifiedhacker.com
| Not valid before: 2025-10-29T14:56:50
|_Not valid after: 2026-01-27T14:56:49
993/tcp open   ssl/imap   Dovecot imapd
| ssl-cert: Subject: commonName=mail.certifiedhacker.com
| Subject Alternative Name: DNS:autodiscover.certifiedhacker.com, DNS:certifiedhacker.com, DNS:cpanel.certifiedhacker.com, DNS:mail.certifiedhacker.com, DNS:webdisk.certifiedhacker.com, DNS:webmail.certifiedhacker.com, DNS:www.certifiedhacker.com
| Not valid before: 2025-10-29T14:56:50
|_Not valid after: 2026-01-27T14:56:49
|_imap-capabilities: LOGIN-REFERRALS Pre-login OK IMAP4rev1 SASL-IR AUTH=PLAIN ENABLE NAMESPACE AUTH=LOGINA0001 ID listed more IDLE LITERAL+ capabilities post-login have
|_ssl-date: TLS randomness does not represent time
995/tcp open   ssl/pop3   Dovecot pop3d
| ssl-cert: Subject: commonName=mail.certifiedhacker.com
| Subject Alternative Name: DNS:autodiscover.certifiedhacker.com, DNS:certifiedhacker.com, DNS:cpanel.certifiedhacker.com, DNS:mail.certifiedhacker.com, DNS:webdisk.certifiedhacker.com, DNS:webmail.certifiedhacker.com, DNS:www.certifiedhacker.com
| Not valid before: 2025-10-29T14:56:50
|_Not valid after: 2026-01-27T14:56:49
|_pop3-capabilities: RESP-CODES SASL(PLAIN LOGIN) USER CAPA TOP AUTH-RESP-CODE UIDL PIPELINING
|_ssl-date: TLS randomness does not represent time
1022/tcp filtered exp2
1023/tcp filtered netvenuechat
1026/tcp filtered LSA-or-nterm
2222/tcp open   ssh   OpenSSH 8.7 (protocol 2.0)
3306/tcp open   mysql   MySQL 5.7.44-48
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=*.bluehost.com
| Subject Alternative Name: DNS:*.bluehost.com, DNS:bluehost.com
| Not valid before: 2025-01-27T00:00:00
|_Not valid after: 2026-01-27T23:59:59
| mysql-info:
| Protocol: 10
| Version: 5.7.44-48
| Thread ID: 10132403

Fig(46)

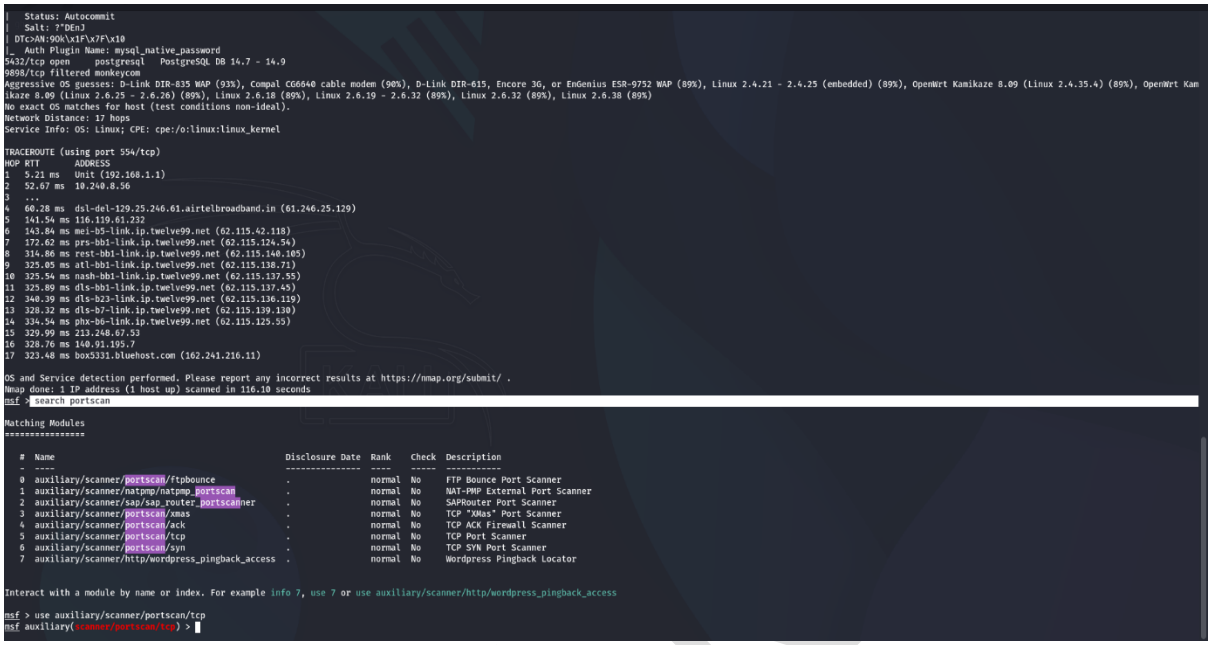

|_Not valid after: 2026-01-27T14:56:49
|_pop3-capabilities: RESP-CODES SASL(PLAIN LOGIN) USER CAPA TOP AUTH-RESP-CODE UIDL PIPELINING
|_ssl-date: TLS randomness does not represent time
1022/tcp filtered exp2
1023/tcp filtered netvenuechat
1026/tcp filtered LSA-or-nterm
2222/tcp open   ssh   OpenSSH 8.7 (protocol 2.0)
3306/tcp open   mysql   MySQL 5.7.44-48
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=*.bluehost.com
| Subject Alternative Name: DNS:*.bluehost.com, DNS:bluehost.com
| Not valid before: 2025-01-27T00:00:00
|_Not valid after: 2026-01-27T23:59:59
| mysql-info:
| Protocol: 10
| Version: 5.7.44-48
| Thread ID: 10132403
| Capabilities flags: 65535
| Some Capabilities: DontAllowDatabaseTableColumn, SupportsLoadDataLocal, Speaks41ProtocolOld, SupportsTransactions, InteractiveClient, SwitchToSSLAfterHandshake, IgnoreSpaceBeforeParenthesis, IgnoreSigpipes, ConnectWithDatabase, SupportsCompression, Support41Auth, FoundRows, Speaks41ProtocolNew, LongColumnFlag, ODBCClient, LongPassword, SupportsAuthPlugins, SupportsMultipleStatments, SupportsMultipleResults
| Status: Autocommit
| Salt: ?*DEnJ
| DTc>AN:9Ok\x1F\x7F\x10
|_ Auth Plugin Name: mysql_native_password
5432/tcp open   postgresql   PostgreSQL DB 14.7 - 14.9
9898/tcp filtered monkeycom
Aggressive OS guesses: D-Link DIR-835 WAP (93%), Compal CG6640 cable modem (90%), D-Link DIR-615, Encore 3G, or EnGenius ESR-9752 WAP (89%), Linux 2.4.21 - 2.4.25 (embedded) (89%), OpenWrt Kamikaze 8.09 (Linux 2.4.35.4) (89%), OpenWrt Kamikaze 8.09 (Linux 2.6.25 - 2.6.26) (89%), Linux 2.6.18 (89%), Linux 2.6.19 - 2.6.32 (89%), Linux 2.6.32 (89%), Linux 2.6.38 (89%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 17 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 554/tcp)
HOP RTT   ADDRESS
1   5.21 ms Unit (192.168.1.1)
2   52.67 ms 10.240.8.56
3   ...
4   60.28 ms dsl-del-129.25.246.61.airtelbroadband.in (61.246.25.129)
5   141.54 ms 116.119.61.232
6   143.84 ms mei-b5-link.ip.twelve99.net (62.115.42.118)
7   172.62 ms prs-bb1-link.ip.twelve99.net (62.115.124.54)
8   314.86 ms rest-bb1-link.ip.twelve99.net (62.115.140.105)
9   325.05 ms atl-bb1-link.ip.twelve99.net (62.115.138.71)
10 325.54 ms nash-bb1-link.ip.twelve99.net (62.115.137.55)
11 325.89 ms dls-bb1-link.ip.twelve99.net (62.115.137.45)
12 340.39 ms dls-b23-link.ip.twelve99.net (62.115.136.119)
13 328.32 ms dls-b7-link.ip.twelve99.net (62.115.139.130)
14 334.54 ms phx-b6-link.ip.twelve99.net (62.115.125.55)
15 329.99 ms 213.248.67.53
16 328.76 ms 140.91.195.7
17 323.48 ms box5331.bluehost.com (162.241.216.11)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 116.10 seconds
msf >

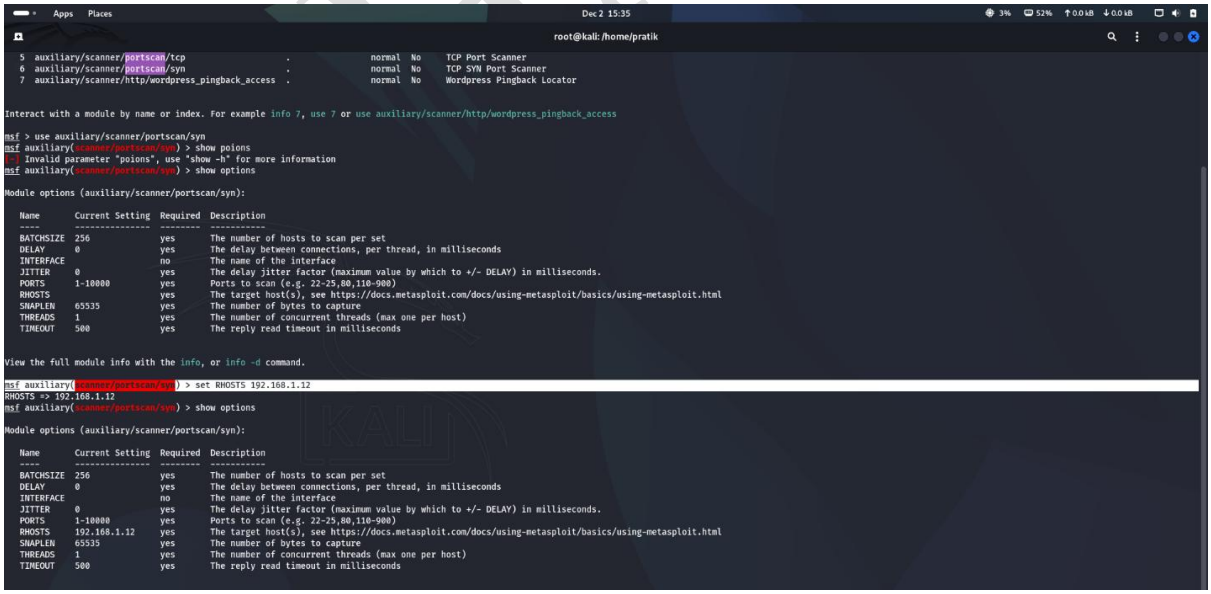
Fig(46)

Step4: Type search portscan and and press Enter. The Metasploit Metasploit port scanning modules appear, as shown in the screenshot



Fig(47)

Step5:Issue command: set RHOSTS 192.168.1.12



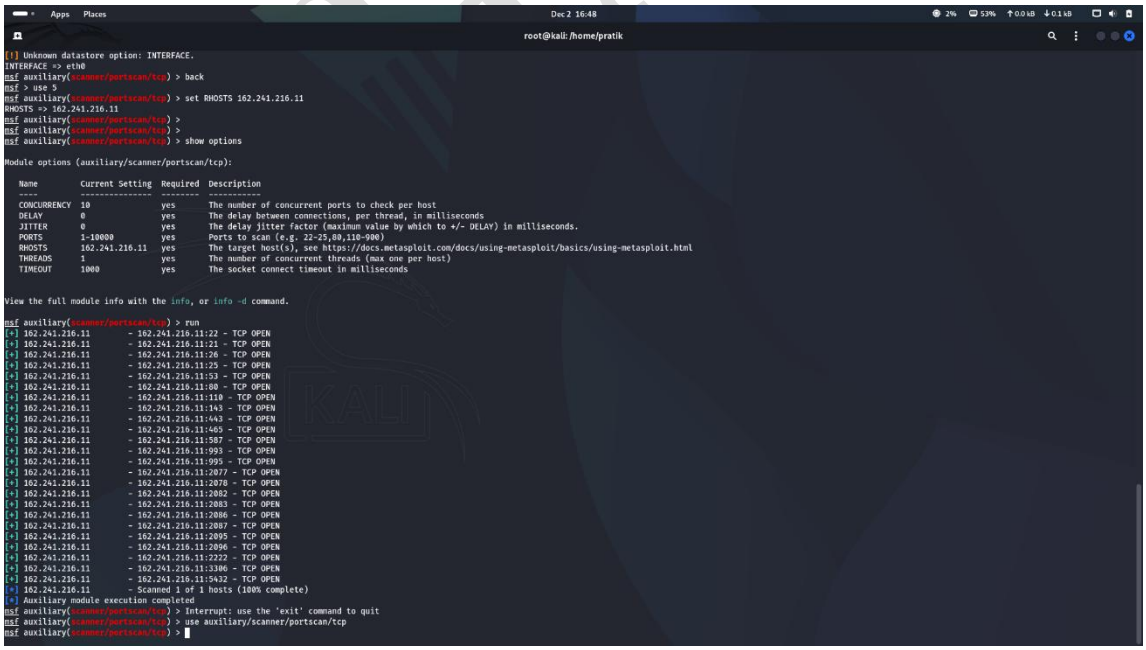
Fig(48)

Step6:Issue the below commands:

- set INTERFACE eth0
- set THREADS 50
- show options

Fig(49)

Step7: Issue command : set RHOSTS 162.241.216.11



Fig(50)