Project Title: Local Network Port Scanning and Packet Capture Analysis

Objective

To identify live hosts and open ports in a local subnet using nmap, and to capture & analyze the scan traffic using Wireshark. This is a basic yet essential activity in network reconnaissance and vulnerability assessment.

Tools Used

- **Kali Linux** – For scanning using nmap
- **Nmap** – For host discovery and port scanning
- **Wireshark** – For capturing and analyzing packets
- **GitHub** – For sharing code and analysis

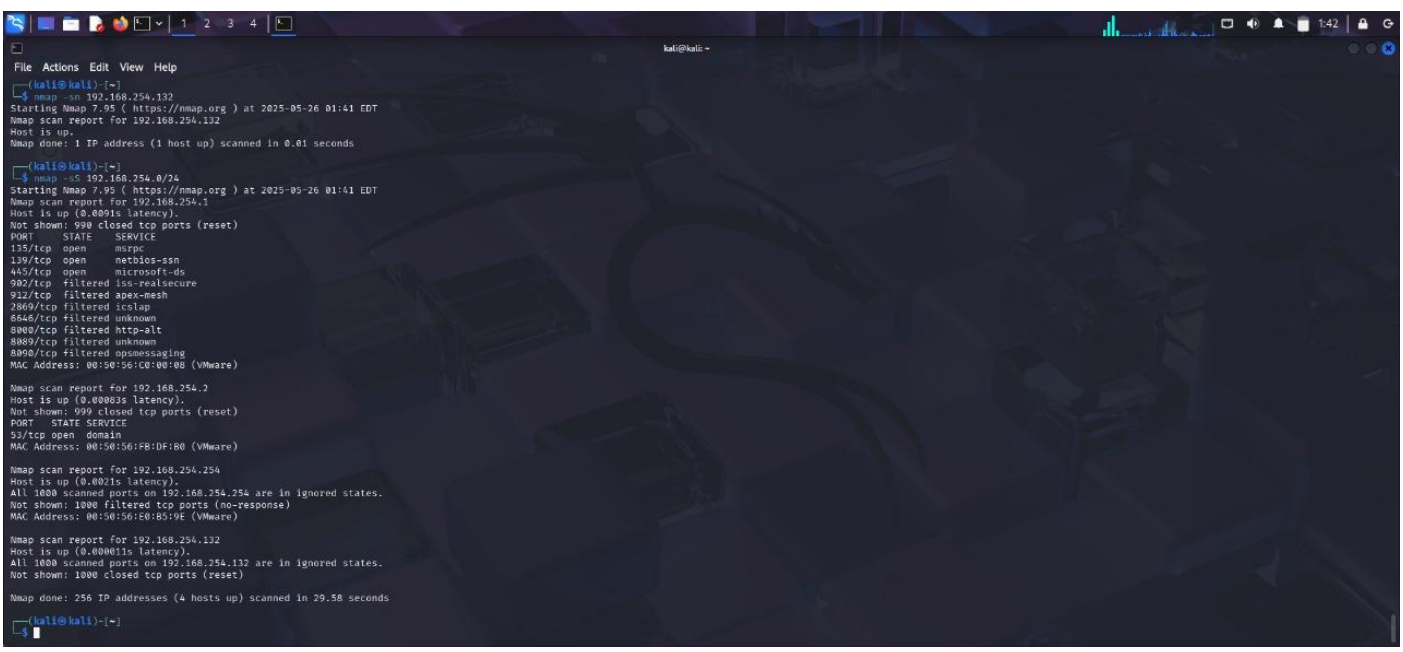
Screenshots and Proof of Work

1. Nmap Scans

From your first screenshot:

- **Command** used:

  nmap -sS 192.168.254.0/24



- This performs a **TCP SYN Scan** across the subnet.
- Multiple hosts were discovered.
- Open ports like 135/tcp, 445/tcp, and 902/tcp were identified.
- Ports like 443/tcp, 8080/tcp were filtered or closed on some hosts.
- MAC addresses and vendor info (e.g., VMware) were also captured.

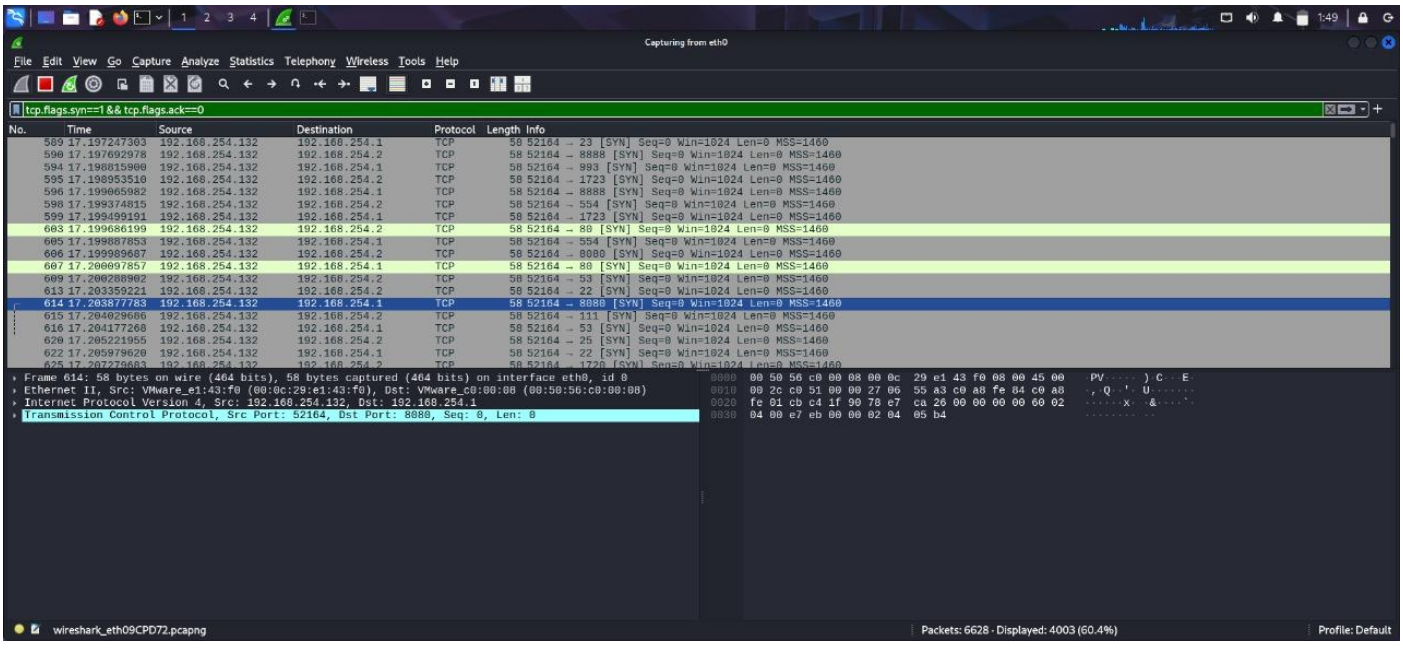
2. Wireshark Capture

From your second screenshot:

- **Display Filter Used**:

  tcp.flags.syn==1 && tcp.flags.ack==0

- This isolates **SYN packets** which are the first step in a TCP connection attempt and commonly used in SYN scanning.
- The capture shows:
  - ✓ Numerous SYN packets from 192.168.254.132 to ports like 23, 80, 8080, etc.
  - ✓ These packets were generated by your Nmap scan.
  - ✓ Destinations include other hosts on the network such as 192.168.254.1.



- **1. nmap -sn 192.168.254.132**

Purpose:

This command performs a Ping Scan (also known as a host discovery scan) to check if the host 192.168.254.132 is up (active).

Usage:

- nmap: The tool used for network scanning.
- -sn: "No port scan" – only pings the IP to check if the host is up, without scanning any ports.
- 192.168.254.132: Target IP address**.**

Result:

Host is up — confirms that the system at 192.168.254.132 is reachable.

- **2. nmap -sS 192.168.254.0/24**

*Purpose***:**

This command performs a TCP SYN scan on all devices in the 192.168.254.0/24 subnet, identifying open ports on each live device.

Usage:

- -sS: SYN scan (also known as half-open scan) – sends SYN packets and waits for SYN-ACK to determine open ports.
- 192.168.254.0/24: Scans all 256 IP addresses in the subnet (from 192.168.254.0 to 192.168.254.255).

Result:

Lists IPs that are up, their MAC addresses, and ports (open/filtered/closed).
Example:

445/tcp open microsoft-ds

Wireshark Capture Filters Used

Purpose:

To observe network packets generated by the nmap scans, especially SYN packets.

Used Filter:

tcp.flags.syn==1 && tcp.flags.ack==0

Why?:

- tcp.flags.syn==1: Captures TCP SYN packets (used to initiate a connection).
- tcp.flags.ack==0: Ensures that only initial SYN packets are shown (not responses like SYN-ACK).

Result:

Shows outgoing connection attempts (SYN packets) from the scanning machine to various ports.

Achieved

With nmap:

- Discovered live hosts in the subnet.
- Identified open and filtered ports.
- Determined operating systems and MAC addresses for some hosts.

With Wireshark:

- Captured the actual packets from the port scan.
- Verified the SYN scanning method visually.
- Observed how a port scan looks on the network (real-time).

Conclusion:

- This confirms a successful port scan and matching capture using Wireshark.