**Task 1: Scan Your Local Network for Open Ports**

**Objective: Learn to discover open ports on devices in your local network to understand network exposure.**

**Tools: Nmap**

### What is a Port?

A port is a virtual communication endpoint on a device used to distinguish different services and applications. Ports are identified by numbers ranging from 0 to 65535. Common ports include 80 (HTTP), 22 (SSH), and 443 (HTTPS).

### What is Port Scanning?

Port scanning is the process of sending requests to a range of ports on a host to determine which ports are open and what services might be running. It helps in identifying potential security vulnerabilities.

### TCP SYN Scan

A TCP SYN scan is a stealthy scanning technique where the scanner sends a TCP SYN packet to a port and waits for a response. If a SYN-ACK is received, the port is considered open. This scan does not complete the full TCP handshake, reducing the chance of detection.

### Open Ports and Security Risks

Open ports indicate active services that may be accessible from the network. Unsecured open ports can be exploited by attackers to gain unauthorized access or cause disruptions.

**Tools Used**

### Nmap

Nmap (Network Mapper) is a free and open-source tool for network discovery and security auditing. It allows scanning of IP addresses and ports to identify active devices and services running on a network.

### Wireshark (Optional)

Wireshark is a widely-used network protocol analyzer that captures and inspects network packets in real-time. It was optionally used to capture network traffic during the scanning process, enabling a deeper understanding of the TCP handshake and the behavior of the SYN scan.

**Methodology**

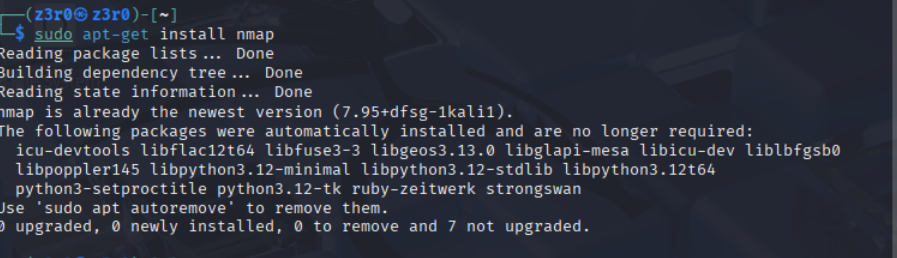
1. Install Nmap

I installed Nmap on my Linux VM using the package manager with the command:

sudo apt-get update

sudo apt-get install nmap

This ensured the latest stable version was installed.

****

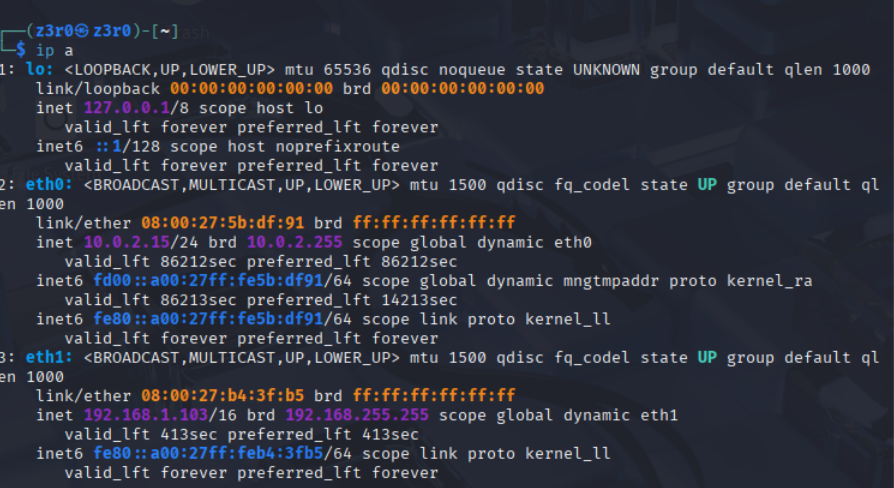
1. **Find Local IP Range**

To find the IP address and subnet of my VM’s network interface, I ran:

ip a

Ifconfig

· From the output, I identified my local subnet as 192.168.1.0/24.

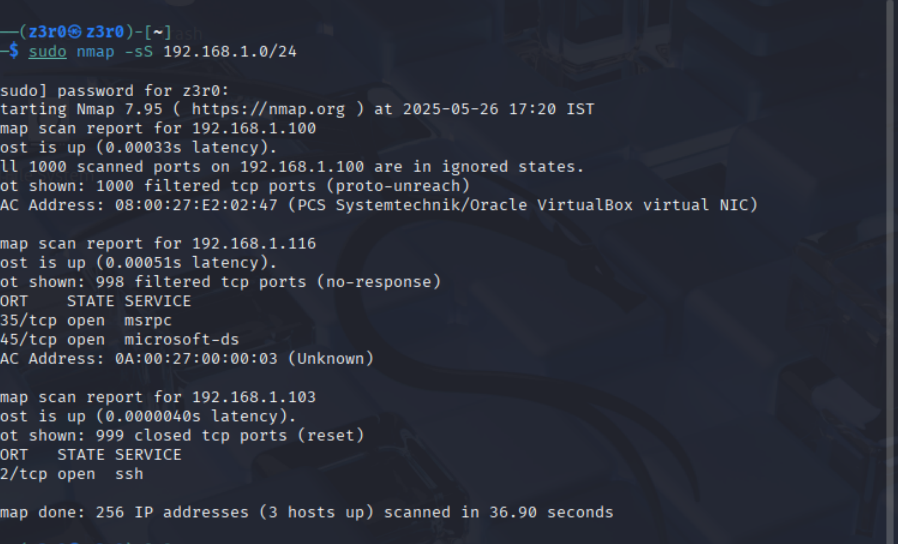


1. **Run TCP SYN Scan**

I performed a TCP SYN scan on the local network using:

sudo nmap -sS 192.168.1.0/24

The sudo is required because SYN scan needs elevated privileges.



The sudo is required because SYN scan needs elevated privileges.

1. **Record Open Ports and IP Addresses**

I analyzed the scan results to note active IP addresses and the ports open on those hosts.

| **IP Address** | **Open Ports** | **Common Services** |
| --- | --- | --- |

|  |  |  |
| --- | --- | --- |
| 192.168.1.103 | 22 | SSH (Secure Shell) |

|  |  |  |
| --- | --- | --- |
| 192.168.1.116 | 135, 445 | Msrpc, SMB (Windows File Sharing) |

1. **Packet Capture with Wireshark**

I optionally installed and used Wireshark to capture packets during the scan:

sudo apt-get install wireshark

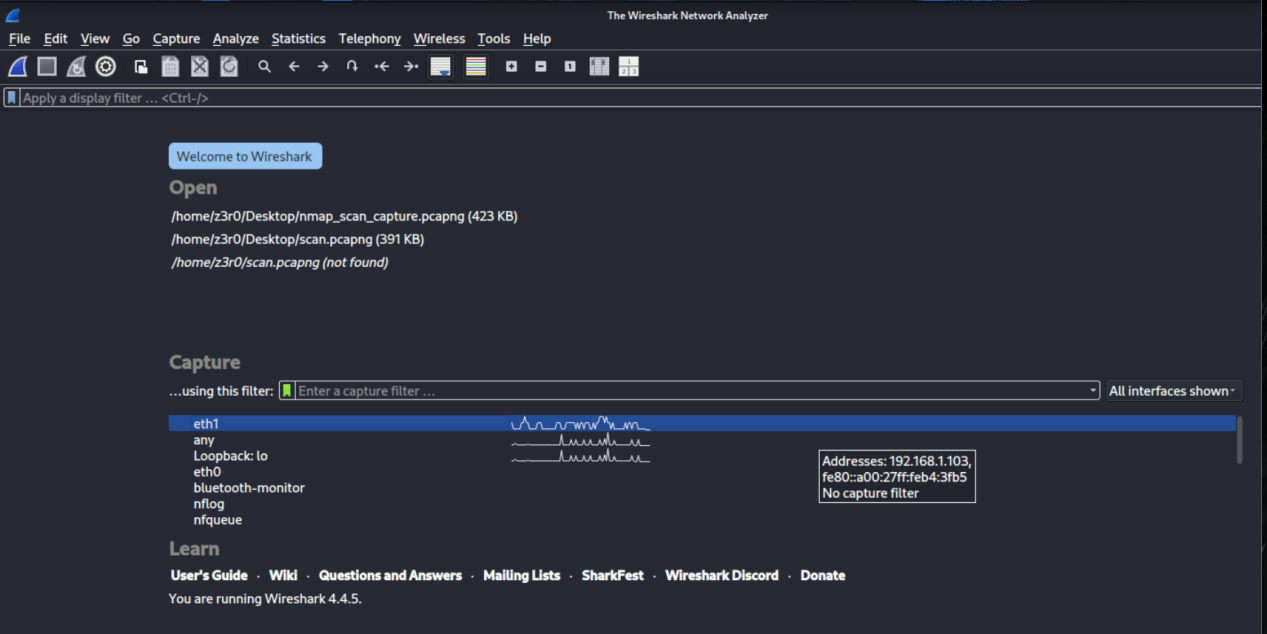
I captured traffic on the relevant network interface to observe SYN and SYN-ACK packets.

### ****Start Wireshark on eth1****

* Open a terminal and run:

sudo wireshark

* In Wireshark, select eth1 as the interface.
* Click the **shark fin icon** (top left) to start capturing packets.



### ****Run Nmap TCP SYN Scan in Another Terminal****

While Wireshark is capturing, open a second terminal and run the scan:

sudo nmap -sS 192.168.1.0/24

### ****Stop the Capture****

Once the scan finishes:

* Go back to Wireshark.
* Click the **red square "Stop" button** to stop the capture.

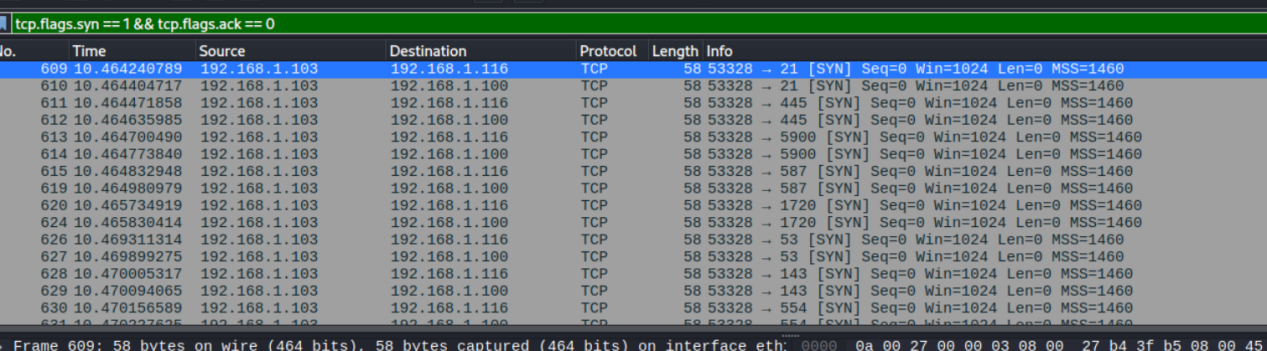
### ****Filter and Analyze SYN Packets****

To view just the SYN/SYN-ACK packets in Wireshark:

* In the **filter bar**, enter:

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

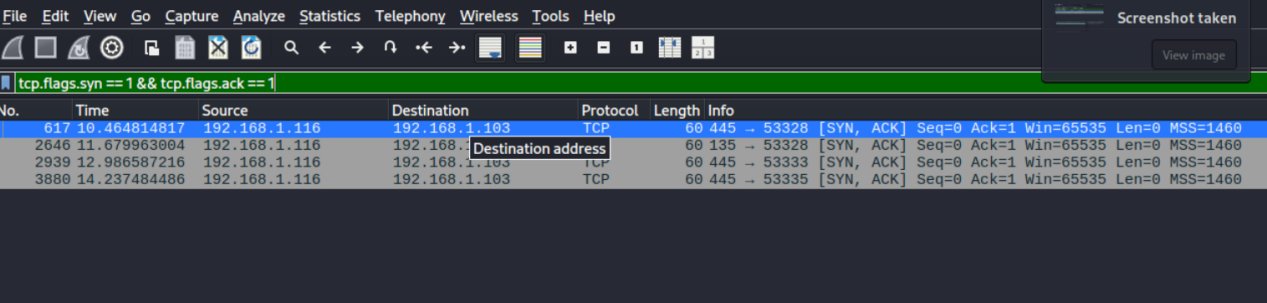
This shows SYN packets sent by Nmap (initial connection attempts).



To see SYN-ACK responses (indicating open ports), use:

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

This filtering helps you observe how Nmap detects open ports during a TCP SYN scan.



1. **Research Common Services Running on Open Ports**

| **IP Address** | **Open Ports** | **Common Services** | **Description** |
| --- | --- | --- | --- |
| 192.168.1.103 | 22 | SSH (Secure Shell) | Provides encrypted command-line access to remote machines. |
| 192.168.1.116 | 135, 445 | MSRPC (DCE/RPC), SMB (Server Message Block) | Used for Windows file sharing, printer sharing, and remote procedure calls. |

1. **Identify Potential Security Risks from Open Ports**

| **IP Address** | **Port** | **Service** | **Potential Security Risks** |
| --- | --- | --- | --- |
| 192.168.1.103 | 22 | SSH | - Brute-force attacks if weak credentials are used- Vulnerable to exploits if outdated. |
| 192.168.1.116 | 135 | MSRPC | - Often targeted by malware (e.g., WannaCry)- May allow remote code execution. |
| 192.168.1.116 | 445 | SMB | - Known for vulnerabilities (e.g., EternalBlue exploit)- Can be abused for lateral movement and data theft. |

### Research & Risk Analysis Summary

After conducting the network scan using Nmap, I identified two active hosts with open ports exposing commonly used services. The host at IP address 192.168.1.103 was running an SSH (Secure Shell) service on port 22, which is used for secure remote logins. While SSH is encrypted and secure by design, it remains a potential target for brute-force attacks if weak credentials or outdated versions are used.

The second host, 192.168.1.116, exposed ports 135 and 445. These correspond to MSRPC (Microsoft Remote Procedure Call) and SMB (Server Message Block) services typically found on Windows systems. These services are widely known attack surfaces—especially SMB, which has been exploited in high-profile attacks such as WannaCry and NotPetya. Improperly secured or unpatched systems using these services may allow attackers to execute code remotely, move laterally within a network, or exfiltrate data.

Recognizing these services and understanding their associated risks is essential for evaluating a network’s security posture and implementing necessary defenses such as patching, access control, and traffic monitoring.

### Conclusion

This task helped me learn how to scan a local network to find open ports and identify running services using Nmap and Wireshark. I discovered potential security risks like SSH brute-force attacks and vulnerabilities in Windows file-sharing services. Regular monitoring and securing open ports are essential to protect networks from threats.