Task 1

Network Reconnaissance and Open Port Scanning

Objective:

To understand network exposure by performing port scanning on the local network, identifying open ports and services, analyzing packets with Wireshark, and recognizing potential security risks.

Tools Used

- **Nmap** – Network mapper for scanning ports and IPs.
- **Wireshark** – Packet capturing tool for inspecting network traffic.

Step-by-Step Procedure

1. Install Nmap

- Downloaded Nmap 7.97 from the official website: https://nmap.org
- Installed it with default settings on a Windows system.

2. Identify Local IP Range

- Opened Command Prompt.
- Ran the command:

  ipconfig

- Found the local IP (e.g., 192.168.56.1) and determined the subnet range:
  ➤ **192.168.56.1/24**

3. Run a TCP SYN Scan Using Nmap

- In the command prompt, executed:

  nmap -sS 192.168.56.1/24

- Nmap began scanning 256 IP addresses in the local subnet to check for live hosts and open TCP ports using a **stealth SYN scan**.

4. Observe and Save Results

- Nmap returned the following results for 192.168.56.1:

```
C:\Users\Lenovo>nmap -sS 192.168.56.1/24
Starting Nmap 7.97 ( https://nmap.org ) at 2025-05-26 15:40 +0530
Nmap scan report for 192.168.56.1
Host is up (0.00058s latency).
Not shown: 994 closed tcp ports (reset)
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3306/tcp open  mysql
8080/tcp open  http-proxy
8090/tcp open  opsmessaging

Nmap done: 256 IP addresses (1 host up) scanned in 60.96 seconds
```

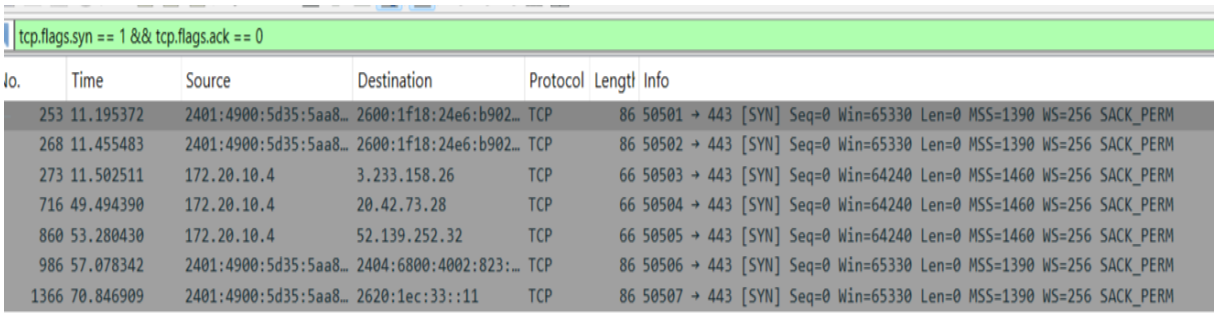
  This means 6 open ports were detected on the live host.

Screenshot saved as: open ports.png

5. Packet Capture with Wireshark

- Started Wireshark and selected the active network interface.
- Applied filter to observe SYN packets:

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



- Observed multiple SYN packets attempting to initiate connections.
- Analyzed packet details including:
  - Source & Destination IP
  - Protocol: TCP
  - Destination Port (e.g., 443 – HTTPS)
  - Flags: SYN
  - Sequence number and payload

Screenshot saved as: wireshark tcp packets.png

6. Research: Common Services Running on Detected Ports

Port	Service	Description
135	MSRPC	Remote procedure calls used by Windows for various communication tasks.
139	NetBIOS-SSN	Network Basic Input/Output System for Windows file and printer sharing.
445	Microsoft-DS	SMB over TCP; commonly used for Windows file sharing.
3306	MySQL	Open-source relational database used by many applications.
8080	HTTP-Proxy	Alternative port for web servers and admin panels.
8090	OpsMessaging	Often used by Java apps and enterprise services like Atlassian tools.

7. Identify Potential Security Risks from Open Ports:

Port 135 – Microsoft RPC (MSRPC):
This port is used by Microsoft's Remote Procedure Call services, which facilitate communication between applications on networked computers. Port 135 has historically been vulnerable, with one of the most notable exploits being MS03-026, used by the Blaster worm. If exposed to the internet, it can allow attackers to remotely execute code, initiate unauthorized communications, or manipulate system processes through DCOM interfaces.

Port 139 – NetBIOS Session Service:
Used primarily for file and printer sharing over a local network, this port can leak sensitive system information if left open. It allows access to shared folders and can be exploited to enumerate user accounts and shared resources. Attackers often target this port to gain unauthorized access, particularly in environments with weak access controls or unpatched systems.

Port 445 – Microsoft-DS (SMB):
This port is used for SMB over TCP/IP and is essential for network file sharing in Windows environments. However, it is one of the most commonly exploited ports due to its vulnerability to serious threats like EternalBlue. This vulnerability was famously weaponized in ransomware attacks such as WannaCry and NotPetya. An open Port 445 significantly increases the risk of malware propagation and unauthorized remote access.

Port 3306 – MySQL Database:
Port 3306 is the default port for MySQL servers. If exposed to the internet without proper authentication or firewall protection, it becomes a high-value target for attackers seeking to access or exfiltrate database contents. Common threats include brute-force attacks, unauthorized access, and SQL injection. A misconfigured MySQL instance can result in full compromise of sensitive backend data.

Port 8080 – HTTP Proxy / Web Admin Panels:
This port is typically used for web services or as an alternative HTTP port, often hosting admin dashboards or development servers. Services running on port 8080 are frequently found with minimal authentication or misconfigurations, making them vulnerable to brute-

force login attempts, directory traversal, or Cross-Site Scripting (XSS) attacks. If not properly secured, this port can offer an easy entry point for attackers.

Port 8090 – Ops Messaging / Internal Services:
Commonly used for enterprise services such as messaging platforms or development tools (e.g., Atlassian products), port 8090 may expose internal web applications to external access. If left unprotected, attackers can potentially access sensitive administrative functions, configuration interfaces, or internal communication systems, leading to information leaks or system compromise.

Mitigations:

- Close unnecessary ports using firewalls.
- Use strong passwords and 2FA for services.
- Regularly update and patch services.
- Limit access to internal IPs using ACLs or VPNs.

Files Included in GitHub Repo

- open ports.png – Nmap scan screenshot
- wireshark tcp packets.png – Wireshark capture
- README.md – This documentation

Key Learnings

- How to perform a SYN scan using Nmap
- How TCP handshakes and flags work
- How to observe SYN packets using Wireshark
- Identification of vulnerable services via open ports
- Basic network hardening strategies

Outcome

This task provided a hands-on understanding of basic network reconnaissance, service discovery, and packet inspection. It highlighted the importance of minimizing exposure to open ports and securing commonly exploited services.