Cyber Security Internship Task 1: Network Port Scanning Report

Objective

To perform a **local network port scan** using **Nmap**, analyze open ports, identify potential security risks, and document findings.

Tools Used

- **Nmap** (Network Mapper) – For port scanning
- **Wireshark** (Optional) – For packet capture analysis
- **GitHub** – For documentation and submission

Methodology

1. Installing Nmap

- Downloaded and installed **Nmap** from the official website.

2. Identifying Local IP Range

- Ran ipconfig (Windows) / ifconfig (Linux/Mac) to find the local subnet.

Interface eth0 (My main network interface):

- **IP Address (inet):** 192.168.xxx.xxx
- **Netmask:** 255.255.255.0
- **Broadcast Address:** 192.168.xxx.255
- **MAC Address:** 00:0c:xx:d4:xx:f8

My Local Network Range:

Given the IP 192.168.xxx.xxx and the netmask 255.255.255.0:

- **Network Address:** 192.168.xxx.x
- **CIDR Notation:** 192.168.xxx.x/24
- **Usable Host Range:** 192.168.190.1 to 192.168.190.254
- **Broadcast Address:** 192.168.190.255

3. Executing the Scan

- Performed a **TCP SYN Scan** (Stealth Scan) using:

  CMD: nmap -sS 192.168.1.0/24

Active Hosts on the Network:

IP Address	Status	Open Ports	MAC Address	Vendor
192.168.190.1	✅ Up	None (All filtered)	00:50:56:C0:00:08	VMware
192.168.190.2	✅ Up	53/tcp (DNS)	00:50:56:F5:CE:57	VMware
192.168.190.254	✅ Up	None (All filtered)	00:50:56:EE:52:74	VMware
192.168.xxx.xxx	✅ Up	None (All closed)	(This is MY **machine**)	—

Interpretation:

- 192.168.190.1 and 192.168.190.254 are likely the **default gateway or virtual network adapters** (possibly NAT/bridged configs in VMware).
- 192.168.190.2 is running a **DNS server** on port 53 — possibly a DHCP/DNS service of my virtual network.
- 192.168.xxx.xxx is my host machine.
- All MAC addresses point to **VMware**, indicating a **virtualized network** environment.

4. Analyzing Results

During the Nmap scan of the local network, several hosts were discovered. However, since the environment is virtualized using **VMware** and the operating system in use is **Kali Linux**, **no open ports** were found on the host machine (192.168.xxx.xxx) or on other VMware virtual interfaces, **except port 53**, which was open on 192.168.190.2 and running the **dnsmasq 2.51** DNS service.

This indicates a highly controlled or minimal service exposure within the local virtual network.

Commonly Encountered Ports and Their Services

(While not observed in this specific scan, these are frequently found in broader network scans):

Port	Protocol	Common Service	Description
22	TCP	SSH	Secure remote login
23	TCP	Telnet	Unencrypted remote login
25	TCP	SMTP	Simple Mail Transfer Protocol
53	TCP/UDP	DNS	Domain Name System (Observed)

Port	Protocol	Common Service	Description
80	TCP	HTTP	Web server (insecure)
110	TCP	POP3	Email client access
139	TCP	NetBIOS	Windows file/printer sharing
143	TCP	IMAP	Email retrieval protocol
443	TCP	HTTPS	Secure web traffic
445	TCP	SMB	Windows file sharing
3389	TCP	RDP	Remote Desktop Protocol

This scan's outcome reflects the current setup's minimal attack surface, which is ideal for secure, sandboxed testing in a Kali Linux VM environment.

5. Identifying Security Risks

While scanning the local virtual network hosted in **VMware** with **Kali Linux**, the environment was found to be secure with **minimal exposure** — only **port 53 (DNS)** was open on host 192.168.190.2, running dnsmasq 2.51. No unnecessary or insecure ports were found exposed in this controlled setup.

However, in broader real-world environments, the following ports and services are often seen as **potential security risks** if improperly configured or left exposed:

Examples of Risky or Outdated Services

Port	Service	Risk Description
21	FTP	Transmits credentials in plaintext; often outdated
23	Telnet	Unencrypted communication; replaced by SSH
25	SMTP	Can be misused for spam or relay attacks
110	POP3	Legacy protocol lacks encryption by default
139/445	NetBIOS/SMB	Vulnerable to exploits like EternalBlue
3389	RDP	A frequent target for brute-force and ransomware attacks

Even though such services were **not detected** in this scan, it's essential to always monitor for:

- **Outdated software versions** (e.g., dnsmasq 2.51 could have known vulnerabilities if not patched)
- **Open ports that are unused** or exposed to untrusted networks
- **Default credentials** or weak authentication mechanisms

This highlights the importance of regularly auditing network configurations, even in isolated virtual environments.

6. Optional: Wireshark Analysis

To further validate the Nmap scan behavior, **Wireshark** was used to **capture network packets** during the scanning process. The following observations were made:

- **Captured SYN packets** sent by Nmap to various IPs and ports across the local VMware network range.
- **SYN-ACK responses** were received **only from hosts with open ports**, notably from 192.168.190.2 on **port 53**, confirming the presence of the dnsmasq service.
- The scan exhibited **stealth behavior**, using **TCP SYN scan (-sS)** — only the initial SYN packet was sent, and no full TCP handshake (SYN-ACK followed by ACK) was completed, helping to **avoid detection by some firewalls and intrusion detection systems (IDS)**.
- No other significant traffic or unexpected responses were observed, indicating a clean and controlled virtual environment.

This packet-level analysis reinforces the minimal exposure observed during the Nmap scan and verifies the low-risk state of the current VMware-hosted network.

7. Save as a Text File

CMD: nmap -sS 192.168.190.0/24 -oN scan_results.txt

- -sS: SYN scan (stealth)
- 192.168.190.0/24: Your local subnet (adjust if needed)
- -oN scan_results.txt: Saves output in **normal text format**

This will save the result as scan_results.txt in your current directory.

8. Save as an HTML File (via XML + xsltproc)

First, save the output as XML:

CMD: nmap -sS 192.168.190.0/24 -oX scan_results.xml

Then convert to HTML:

CMD: xsltproc scan_results.xml -o scan_results.html

- xsltproc is a command-line tool to transform XML using XSLT.
- The resulting scan_results.html will be a nicely formatted HTML report.

If xsltproc is not installed, install it using:

CMD: sudo apt install xsltproc

Alternative: Save in All Formats at Once

CMD: nmap -sS 192.168.190.0/24 -oA scan_results

This will generate:

- scan_results.nmap (Normal)
- scan_results.xml (XML)
- scan_results.gnmap (Grepable)

Then, you can convert the .xml to .html using xsltproc as shown above.