

**Name:** Bint e Fatima Niazi

**Date:** July 2025

**Contact:**

-Email: fatimabinte4@gmail.com

-LinkedIn: linkedin.com/in/bint-e-fatima-niazi

-GitHub: github.com/fatima-niazi

Project: Nmap Port Scanning

# Overview:

This report documents hands-on experience using Nmap for network scanning. I explored both LAN-based and internet-based scanning techniques. Every command was tested, analyzed, and reviewed for real-world value. Screenshots are to be attached for demonstration purposes.

For educational purposes only.

# Tools & Environment

- Kali Linux   
- Nmap (network scanning tool)  
- Local network (LAN)  
- Internet (Scanme test target)  
- Python3 for local server

**Key Features of Nmap:**

1. **Network Discovery:**

Identify live hosts on a network.

Detects open ports and running services.

1. **Service Version Detection:**

Determines the version of services (e.g., web servers, databases) running on open ports.

1. **Operating System Detection:**

Identifies the operating system and its version on a target system.

1. **Vulnerability Detection:**

Can identify known vulnerabilities by integrating with tools like Nmap Scripting Engine (NSE).

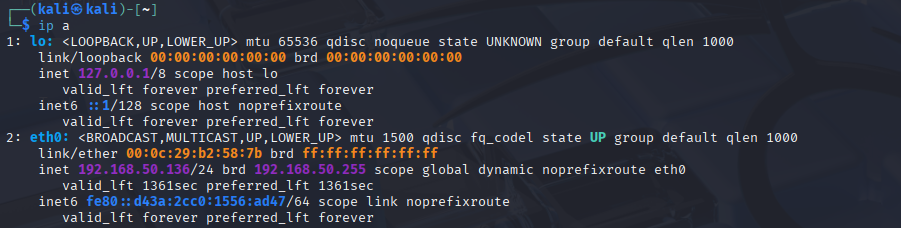
1. **Flexible and Scalable:**

Works efficiently on both small local networks and large enterprise environments.

# LAN-Based Scanning

* Command: ip a

Displays your IP address and network interfaces. Used to identify the local network range (e.g., 192.168.1.0/24).



* Command: nmap -sn 192.168.1.0/24

Performs a ping scan to detect live hosts on the LAN. Helps identify which devices are online.

A computer screen shot of a computer

AI-generated content may be incorrect.

* Command: nmap -sV -p 8080 127.0.0.1

Used after hosting a local web server using Python on port 8080. This checks if the local service is detected correctly.

A computer screen shot of a computer

AI-generated content may be incorrect.

Additionally, we hosted a local HTTP server using:

* Command: python3 -m http.server 8080

This was used to emulate a running service on the localhost to see how Nmap detects open ports.

A screen shot of a computer

AI-generated content may be incorrect.

# Internet-Based Scanning

* Command: nmap scanme.nmap.org

Performs a basic scan of the top 1000 ports on Scanme, the official test target by the Nmap team. Returned SYN stealth scan.

A screen shot of a computer program

AI-generated content may be incorrect.

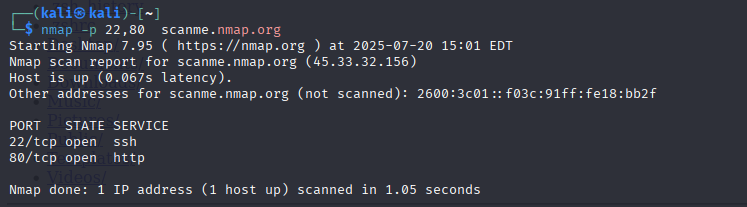
* Command: nmap -sV scanme.nmap.org

Attempts service and version detection on open ports. The scan was slow and eventually showed minimal additional info due to filtering.

Scan attempted but returned minimal output due to filtering.

* Command: nmap -sV -p 22,80 scanme.nmap.org

Scans only port 22 and 80 for services, improving speed. Again confirmed stealth scan but without much version info.



* Command: nmap -sV -Pn --max-retries 2 scanme.nmap.org

Skipped host discovery and reduced retries to increase efficiency. Still resulted in limited info due to rate-limiting/firewall behavior. A screen shot of a computer

AI-generated content may be incorrect.

# Conclusion

This task successfully demonstrated basic and intermediate Nmap usage. LAN scanning helped visualize host discovery and port detection, while Internet-based scanning revealed firewall limitations and stealth scanning in action. These are valuable reconnaissance skills in penetration testing.