**Project: Network Scanning and Discovery with Nmap**

# 1. Introduction

In this project, I performed network scanning and discovery using Nmap on a virtual network environment set up in VirtualBox. The goal was to identify open ports, services, and active devices on the network, focusing on practical network scanning and enumeration techniques. This project helps develop a fundamental understanding of network reconnaissance.

# 2. Tools Used

- Nmap: This is for network scanning and service enumeration.  
- VirtualBox: To create the virtual network environment for testing.  
- Kali Linux: Used as the scanning/attacker machine.

# 3. Environment Setup

The virtual environment was set up using VirtualBox with Kali Linux as the primary attack machine. The VirtualBox network was configured in NAT mode, providing the IP address 10.0.2.15 to the Kali Linux machine.

# 4. Basic Scan on Local Machine

I scanned the local machine (IP: 10.0.2.15) to identify open ports and services.

**Command:**

Nmap 10.0.2.15

**Output:**

Starting Nmap 7.94 (https://nmap.org) at 2024-10-02  
Nmap scan report for 10.0.2.15  
The host is up (0.00056s latency).  
Not shown: 999 closed ports.  
PORT STATE SERVICE  
22/tcp open ssh

**Analysis:**  
This scan identified that port 22 (SSH) was open on the local machine, indicating that the SSH service was running. No other significant ports were found open.

# 5. Scanning the Virtual Network

Next, I scanned the virtual network range 10.0.2.1-254 to detect other active devices and services within the virtual network.

**Command:**

Nmap 10.0.2.1-254

Output:

Nmap scan report for 10.0.2.1  
The host is up (0.00032s latency).  
Not shown: 1000 closed ports  
  
Nmap scan report for 10.0.2.3  
The host is up (0.00042s latency).  
Not shown: 1000 closed ports  
  
Nmap scan report for 10.0.2.15  
The host is up (0.00054s latency).  
PORT STATE SERVICE  
22/tcp open ssh

**Analysis:**  
This scan identified multiple devices on the virtual network, including my local machine. No additional services were found running on the other devices, except for the SSH service on 10.0.2.15.

# 6. SYN Scan (Stealth Scan)

I performed a SYN scan (stealth scan) on my local machine to detect any hidden services.

**Command:**

nmap -sS 10.0.2.15

**Output:**

Nmap scan report for 10.0.2.15  
The host is up (0.00047s latency).  
Not shown: 999 closed ports.  
PORT STATE SERVICE  
22/tcp open ssh

**Analysis:**  
The SYN scan confirmed the presence of SSH on port 22, with no other open ports detected. This scan is helpful for faster detection and bypassing certain firewall rules, but in this case, the results were the same as a standard scan.

# 7. Service and Version Detection

I performed a service and version detection scan to gather detailed information about the services running on the open ports.

Command:

nmap -sV 10.0.2.15

**Output:**

Nmap scan report for 10.0.2.15  
The host is up (0.00059s latency).  
Port State Service Version  
22/TCP open SSH OpenSSH 8.2p1 Debian 4 (protocol 2.0)

**Analysis:**  
The scan revealed that the SSH service running on port 22 used OpenSSH version 8.2p1. This information is useful for determining whether the service is vulnerable to specific exploits or misconfigurations.

# 8. Conclusion

This project provided hands-on experience with Nmap for network scanning and discovery. I identified open ports and services on my local machine and the virtual network and learned to use different scan techniques like SYN scans and service/version detection. This is a fundamental skill in network reconnaissance and vulnerability discovery.