CSE 406: Project Report

Network Scanning with Nmap

Submitted by:
Nafis Karim - 1805027
Mamun Munshi - 1805028

1 Introduction

Nmap stands for "Network Mapper," an open-source tool designed for network discovery, auditing, and security analysis. Created by Gordon Lyon, it was initially released in September 1997. Over the years, Nmap has become a de facto standard for network mapping and security scanning.

Nmap is often considered the 'Swiss Army knife' of network security, as it is highly flexible and capable of handling a wide variety of tasks. It can be used to discover devices running on a network and find open ports along with various attributes of the network.

1.1 Core Objectives

- **Network Discovery:** To find devices running on a network and map their ports.
- **Security Auditing:** To identify vulnerabilities within a given network.
- **Network Inventory:** To keep track of all hosts in a network.
- **Network Monitoring:** To monitor hosts or services in a network and provide insights into who is doing what.
- **Traffic Analysis:** To analyze network traffic and identify patterns or anomalies.
- **Performance Testing:** To measure the network's throughput, latency.
- **Compliance Checking:** To ensure that the network adheres to industry standards and regulations.
- **Troubleshooting:** To diagnose and resolve network issues.

1.2 Key Features:

- **Flexibility:** Highly adaptable, with options for different types of scans.
- **Open Source:** Free to use and modify, with a very active community.
- **Cross-Platform:** Available for Windows, Linux, macOS, and various other operating systems.
- **Scripting Engine:** Users can write their own scripts or use existing ones to extend Nmap's capabilities.
- **Rich Output Formats:** Offers multiple formats like interactive, XML, JSON, and even allows for graphical representations.
- **Portability:** Can be run from a portable USB device, without requiring installation.

2 Installation and Setup

Installing Nmap varies depending on the operating system in use. Below are the guidelines for installing Nmap on Windows, Linux, and macOS.

2.1 Windows

1. Visit the Nmap download page: `https://nmap.org/download.html`

2. Download the latest Windows installer (.exe file).

3. Run the installer and follow the on-screen instructions.

Note: The installer generally includes both Nmap and Zenmap, the graphical user interface for Nmap. You can choose to install both or only the command-line utility.

2.2 Linux

On a Debian-based system like Ubuntu, use the following command:

```
sudo apt-get update
sudo apt-get install nmap
```

For a Red Hat-based system like Fedora, use:

```
sudo dnf update
sudo dnf install nmap
```

2.3 macOS

For macOS, you can use Homebrew, a popular package manager. If you haven't installed Homebrew yet, you can get it from `https://brew.sh/`. Then, run:

```
brew update
brew install nmap
```

Note: Administrative privileges are generally required to perform scans, so you may need to use ‘sudo‘ on Linux and macOS.

3 Core Functions of Nmap

Nmap serves multiple purposes and offers a variety of features for probing computer networks. Below are some of its core functions:

- **Port Scanning:** Nmap is most famously used for identifying open ports on a target host. It supports various types of scans, including TCP, UDP, and SCTP scanning, to understand which ports are open, closed, or filtered.

- **Host Discovery:** Also known as ping scanning, this function identifies active hosts in the network. Nmap sends ICMP echo requests or TCP/UDP packets to various hosts and waits for a response to determine their status.
- **OS and Version Detection:** Nmap can guess the operating system and its version of a targeted machine. This is valuable for understanding the possible vulnerabilities of the target host.
- **Packet Crafting:** This advanced feature allows users to customize the packet sent to the target, providing a way to bypass firewalls or create specialized scan techniques.
- **Service Detection:** Beyond just the ports, Nmap can also identify the services running on them. This includes detecting the type of web server, SSH server, and more.
- **Firewall Evasion:** Nmap provides several techniques to bypass firewalls, like fragmentation scans and decoy scans, making it a versatile tool in penetration testing.
- **Vulnerability Scanning:** Using its scripting engine, Nmap can run a wide range of scripts that can detect vulnerabilities in the target system.
- **Rate Control:** Nmap allows you to control the speed of your scan. You can choose to make it fast to complete the scan as quickly as possible, or slow and stealthy to reduce detection chances.
- **Data Exfiltration:** Though not its primary use-case, Nmap can be used for data exfiltration using various Nmap Scripting Engine (NSE) scripts.
- **Topology Mapping:** Nmap can create a network map displaying how different nodes are interconnected.

4 Zenmap: The Official GUI for Nmap

Zenmap is the official graphical user interface (GUI) for Nmap, a free and open-source tool used for network discovery and security scanning. Zenmap is designed to make Nmap easier to use, while providing advanced users with the robust features they expect.

4.1 Features and Benefits

- **User-Friendly Interface:** Zenmap provides an intuitive graphical user interface that makes it easier for new users to get started with network scanning.
- **Saved Profiles:** Allows for the saving of scan profiles, beneficial for running the same types of scans regularly.

- **Real-Time Output:** Displays the scan output in real-time, providing ease of monitoring.
- **Topological Network Maps:** Capable of drawing a topological map of discovered hosts, aiding in understanding network architecture.
- **Comparative Analysis:** Features the ability to compare two scans to identify changes in network topology or open ports.
- **Scripting Support:** Offers the ability to run and customize Nmap Scripting Engine (NSE) scripts for more complex tasks.
- **Multi-platform:** Available for various operating systems including Windows, macOS, and Linux.

4.2 Common Uses

1. Network Inventory: Quick identification of active devices on the network.
2. Security Auditing: Regular scans to discover vulnerabilities.
3. Troubleshooting: Useful for diagnosing network issues.

4.3 Basic Usage

- **Quick Scan:** A simple scan for identifying live hosts, accessible via the "Quick Scan" profile.
- **Intense Scan:** A more detailed scan available through the "Intense Scan" profile.

4.4 Installation

Zenmap is often bundled with Nmap installations and can also be downloaded separately from the Nmap website. The installation process involves a simple installation wizard on Windows and package managers on Linux.

4.5 Comparison with Nmap CLI

While Zenmap offers a more user-friendly interface, it doesn't expose all the capabilities that Nmap has to offer. Advanced users may still prefer the Nmap command-line interface for greater control over scans.

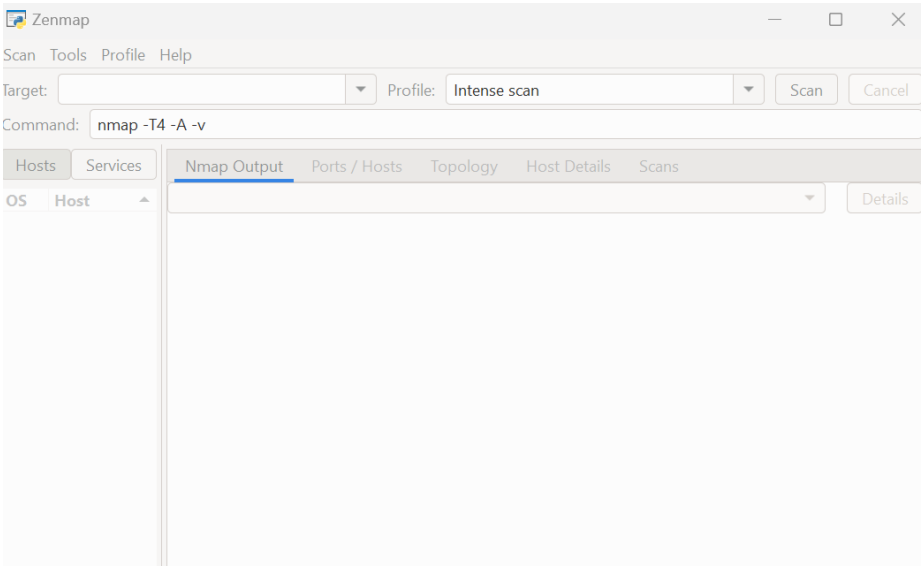


Figure 1: The zenmap gui

5 Basic Workflow of Nmap

The general workflow when using Nmap involves several key steps:

1. **Target Specification:** Choose the target IP addresses or subnets or even read them from a text file.

```
nmap 192.168.1.1-10
or
nmap -iL target_list.txt
```

2. **Scan Type:** Select the type of scan to perform (TCP, UDP, SYN, ACK, etc.).

```
nmap -sS 192.168.1.1
```

3. **Discovery Phase:** Conducts a preliminary scan to identify active hosts.

```
nmap -sn 192.168.1.0/24
```

4. **Port Scanning Phase:** Scans the target's ports to find open, closed, or filtered ports.

```
nmap -p 22,80,443 192.168.1.1
```

5. **OS Detection and Versioning:** Optionally identifies the OS and service versions running on the target.

```
nmap -O --osscan-guess 192.168.1.1
```

6. **Data Analysis:** Presents the data in a readable format or saves it for further analysis.
7. **Reporting:** Logs can be saved in various formats like XML, JSON, or standard text for record-keeping or further analysis.

```
nmap -oX output.xml 192.168.1.1
```

6 Basic Commands and Usage

Nmap offers a wide array of scan options. Here are some basic commands with explanations:

- **Ping Scan:** This scan is used to find which hosts are online in the network.

```
nmap -sn 192.168.1.0/24
```

- **TCP Scan:** The most basic form of TCP scanning involves the "three-way handshake" to establish a connection.

```
nmap -sT 192.168.1.1
```

- **UDP Scan:** Useful for scanning the User Datagram Protocol (UDP) services like DNS, SNMP, and DHCP.

```
nmap -sU 192.168.1.1
```

- **SYN Scan:** Also known as a "half-open" scan, this is a faster way to perform TCP scans without completing the three-way handshake.

```
nmap -sS 192.168.1.1
```

- **ACK Scan:** This scan can be used to map out firewall rule sets.

```
nmap -sA 192.168.1.1
```

- **Comprehensive Scan:** Combines different types of scans and additional information gathering for a detailed view.

```
nmap -sS -sU -T4 -A -v 192.168.1.1
```

- **Service and OS Detection:** Attempts to determine the service and operating system running on the target.

```
nmap -sV -O 192.168.1.1
```

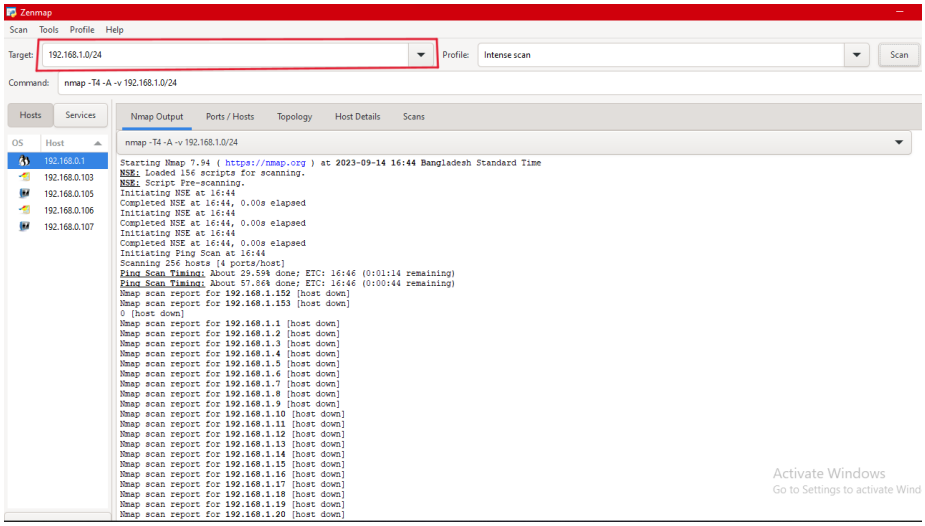
- **Output to File:** Store scan results in a text file for later analysis.

```
nmap -oN output.txt 192.168.1.1
```

7 Network Scanning Demonstration

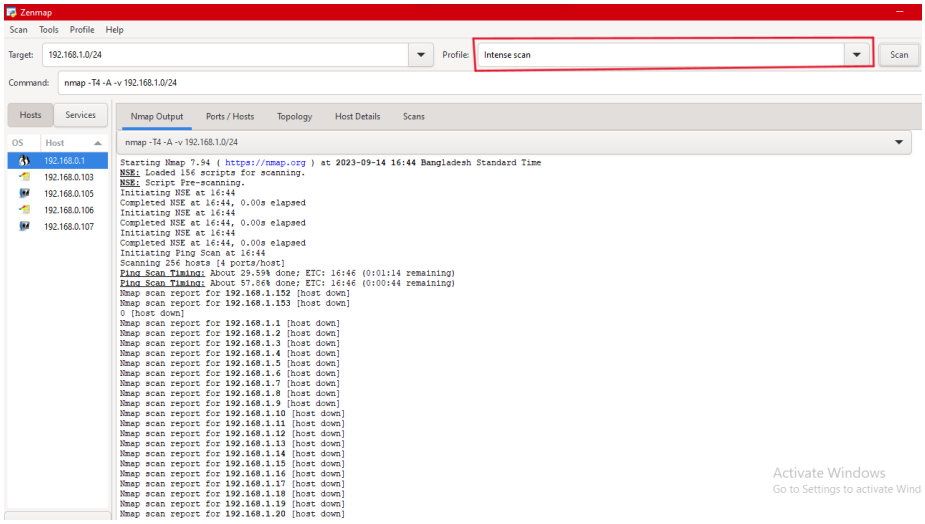
7.1 Specifying the target

Specify the network address that we want to scan:



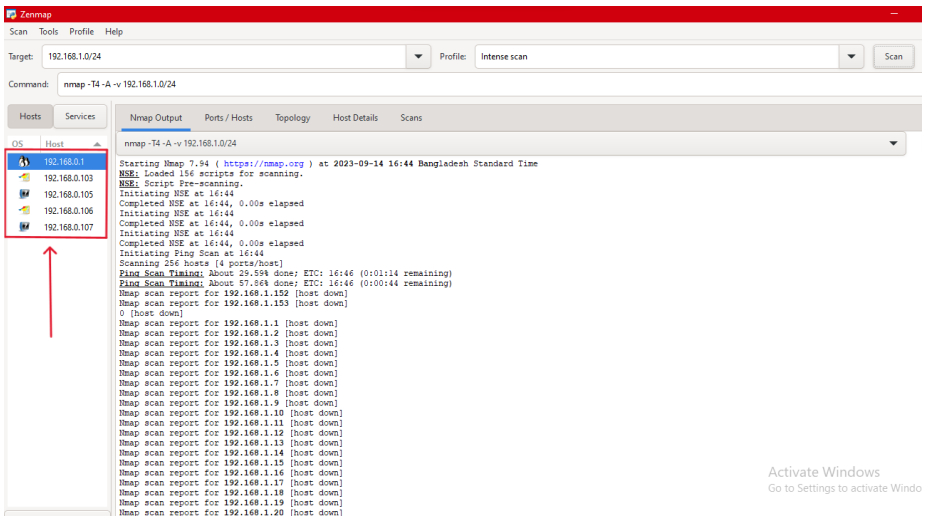
7.2 Specifying scan profile

we can choose the type of scan (for example: regular scan, intense scan or ping scan) and based on our choice, zenmap will automatically customize the command with necessary attributes.



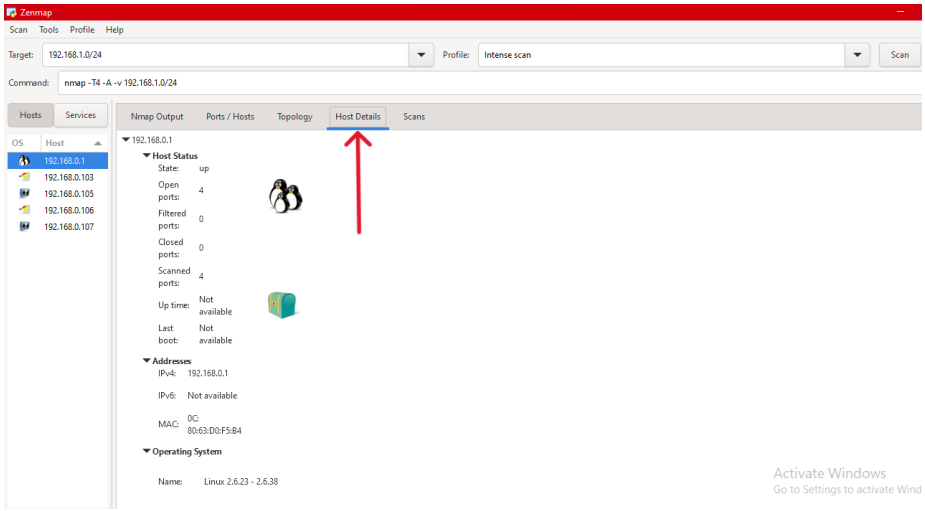
7.3 Exploring available devices

After the scan is completed, the output will describe the available devices. However, we can see a list of available devices at the left side.



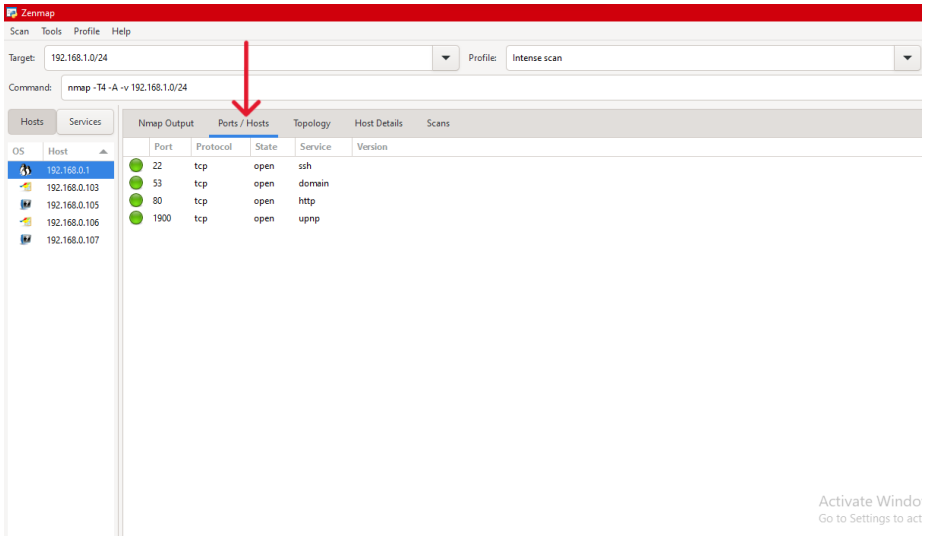
7.4 Host Details

After the scan is completed, the details of each host of the network can be obtained. Nmap can provide the number of open ports, the number of closed ports, IP address, MAC address, operating system etc of all available devices of the network.



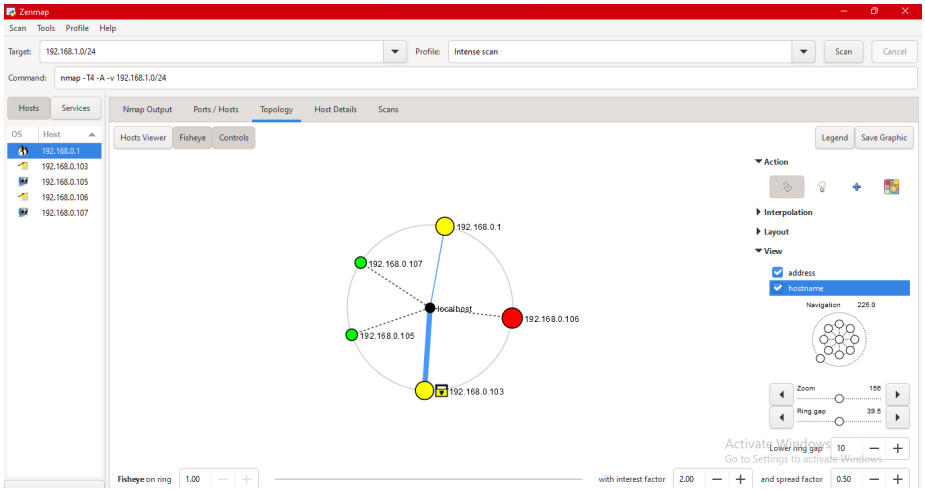
7.5 Port Details

Which service is running on which opened port can be obtained.



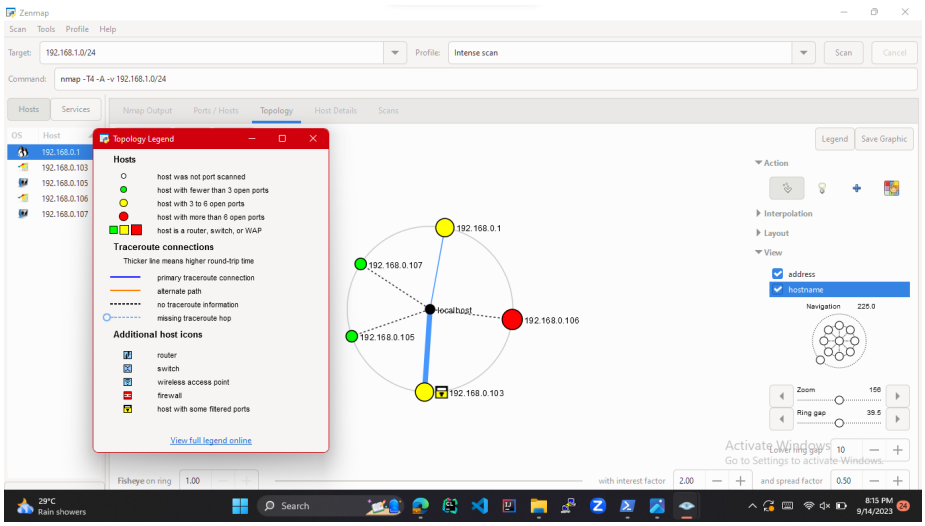
7.6 Displaying Network Topology

The complete topology of the network can be displayed visually. Therefore, the mutual connection between each device seem more understandable.



7.7 Connection Details

Analyzing the connection lines, their width and color, the details of the connection can be obtained.



8 Advanced Techniques

8.1 Script Scanning with NSE (Nmap Scripting Engine)

Description: The Nmap Scripting Engine (NSE) is one of Nmap's most versatile features. It allows users to write scripts that can perform a wide range of networking tasks, such as detecting vulnerabilities, gathering information, and even performing exploitation where applicable.

Example Commands:

- **To run all default scripts:**

```
nmap --script=default <Target IP>
```

- **To scan for SQL injection vulnerabilities:**

```
nmap --script=http-sql-injection <Target IP>
```

8.2 Firewall Evasion

Description: Firewalls often block incoming requests, making it difficult for a scan to pass through. Nmap provides several methods to bypass firewalls, such as sending fragmented packets or using specific TCP flags.

Example Commands:

- **For fragmented packets:**

```
nmap -f <Target IP>
```

- **Specify custom TCP flags:**

```
nmap --scanflags URGACKPSH <Target IP>
```

8.3 Decoy Scanning

Description: Decoy scanning is a way to hide the real source of the scan. The scanner sends out packets with multiple source IP addresses, effectively camouflaging the true origin.

Example Commands:

- **Using 10 random decoys:**

```
nmap -D RND:10 <Target IP>
```

8.4 Timing Templates

Description: Speed is essential but so is reliability. Timing templates in Nmap allow you to choose between the two. Ranging from -T0 (paranoid, slow) to -T5 (insane, fast), you can choose how aggressive your scan should be.

Example Commands:

- **Paranoid timing:**

```
nmap -T0 <Target IP>
```

- **Insane timing:**

```
nmap -T5 <Target IP>
```

8.5 Grepable Output

Description: When you're running automated scripts, human-readable output might not be ideal. Nmap can produce grepable output which is easy to parse but not necessarily easy for humans to read.

Example Commands:

- **To save output in grepable format:**

```
nmap -oG <output_file> <Target IP>
```

8.6 Polite Scanning

Description: In some cases, being less aggressive and not overwhelming the target is important. This mode is more bandwidth-efficient and less likely to trigger alarms.

Example Commands:

- **Least aggressive:**

```
nmap --max-parallelism 1 <Target IP>
```

8.7 Data Length Modification

Description: This technique involves adding random data to the packets to change their lengths, which might help to bypass certain firewall rules or Intrusion Detection Systems (IDS).

Example Commands:

- **Changing the data length:**

```
nmap --data-length 40 <Target IP>
```

8.8 Version Detection

Description: Version detection is crucial for understanding the network's architecture and potential vulnerabilities. Nmap can guess the versions of the services running on the target machine.

Example Commands:

- **To detect service versions:**

```
nmap -sV <Target IP>
```

9 Changing MAC Address Using Nmap

Changing the MAC address itself is not a functionality provided by Nmap. However, Nmap allows for MAC address spoofing during scanning. This can be useful for evasive scanning or penetration testing activities.

9.1 Using `--spoof-mac` with Nmap

You can use the `--spoof-mac` option followed by the MAC address you want to use for the scan.

Examples

1. To use a specific MAC address for the scan:

```
nmap --spoof-mac 00:11:22:33:44:55 <Target IP>
```

2. To use a random MAC address:

```
nmap --spoof-mac 0 <Target IP>
```

3. To use a MAC address from the same vendor as the scanning machine:

```
nmap --spoof-mac <Your Vendor> <Target IP>
```

Note: MAC spoofing might not be legal in some jurisdictions. We always need to ensure that we have proper authorization before conducting any network scans or other penetration testing activities.

10 Changing the Source Port in Nmap

In Nmap, you can specify the source port from which packets are sent using the `--source-port` or `-g` option followed by the port number. This can be helpful for bypassing firewall rules or other security mechanisms that may be in place on the target network.

10.1 Specify a Fixed Source Port

You can specify a fixed source port (e.g., port 53) as follows:

```
nmap --source-port 53 <Target IP>
```

10.2 Use a Privileged Source Port (0-1023)

If you have root privileges, you can specify a privileged source port (e.g., port 22):

```
nmap --source-port 22 <Target IP>
```

10.3 Use a Non-privileged Source Port (1024-65535)

Non-privileged source ports can also be specified:

```
nmap --source-port 12345 <Target IP>
```

11 Conclusion

This report has provided a comprehensive overview of Nmap, an invaluable tool for network security and administration. We have delved into its myriad capabilities, from simple tasks like host discovery and port scanning to more complex features like Script Scanning with NSE, Firewall Evasion techniques, and Decoy Scanning, among others.

Nmap has proven itself as an essential tool in the cybersecurity toolbox, offering not just flexibility but also robustness in gathering information about a network's architecture and potential vulnerabilities. Its open-source nature allows the community to contribute and keep the tool up-to-date with evolving security landscape.

However, it is crucial to note that while Nmap is powerful, it is not a standalone solution for network security. Its effectiveness is maximized when used in conjunction with other security measures and protocols. Moreover, ethical considerations must always be taken into account; unauthorized scanning is illegal and unethical.

In a constantly evolving digital landscape, tools like Nmap serve as both a shield and a lens, enabling administrators and security experts to protect and scrutinize their networks more effectively. As cyber threats become increasingly sophisticated, understanding and leveraging the capabilities of Nmap is more critical than ever.