Nmap

1. Introduction

- **Nmap:**
  Nmap, short for Network Mapper, is an open-source tool used to explore and audit networks. It helps security professionals and system administrators understand what devices are present in a network, what services they are running, and whether there are any potential vulnerabilities. At its core, Nmap sends specially crafted packets to target machines and analyzes their responses. From those responses, it can determine whether a host is active, what ports are open, and what services or applications are running on those ports.

- **Why Nmap is used?**
  Nmap is used to discover devices on a network, identify open ports, detect running services and their versions, determine operating systems, and assess potential vulnerabilities through its scripting engine. It helps security professionals map the attack surface, analyze network configurations, verify firewall rules, and detect weak or outdated services. Because of these capabilities, Nmap is commonly used in penetration testing, network auditing, troubleshooting, and overall security assessment.

- **Strengths:**

→ **Comprehensive Discovery:** Excels at quickly identifying live hosts, open ports, services running, and the operating system (OS fingerprinting) on a target network.

→ **Multiple Scan Techniques:** Offers a vast array of scanning methods (e.g., SYN, Connect, UDP, FIN, Xmas) that allow users to bypass or test different firewall rules and network defenses.

→ **Nmap Scripting Engine (NSE):** This engine significantly expands Nmap's capabilities beyond basic scanning to include vulnerability detection, service enumeration, brute-forcing, and advanced reporting.

→ **Open Source & Cross-Platform: I**t is free, open-source, and available on all major operating systems (Linux, Windows, macOS), fostering a large community that provides frequent updates and scripts.

→ **Speed and Scalability:** Designed for parallel scanning and uses a raw-packet approach, allowing it to scan large networks quickly and efficiently.

- **Limitations:**

→ **Potential for Disruption:** Aggressive or poorly configured scans can consume significant bandwidth, destabilize fragile, legacy, or industrial control systems (ICS), potentially causing crashes or Denial-of-Service (DoS).

→ **Easy to Detect:** While Nmap has stealth options, basic or aggressive scans are often easily logged and blocked by modern Intrusion Detection/Prevention Systems (IDS/IPS) and firewalls.

→ **False Positives/Negatives:** Due to complex network environments (NAT, firewalls), Nmap can sometimes incorrectly identify a port state or OS, requiring manual validation and a high degree of user expertise to interpret results correctly.

- **Note: F**or testing purpose, i am using target as metasploitable 2 Vm (192.168.1.76)

2. Basic Commands

- **Checking Nmap Version:**
  - ➔ **Command:** *nmap --version*
  - ➔ **Purpose:** To ensure you have the latest features and security updates, and to check that the tool is installed and running correctly.

```
┌──(paxton㉿kali)-[~]
└─$ nmap --version
Nmap version 7.95 ( https://nmap.org )
Platform: x86_64-pc-linux-gnu
Compiled with: liblua-5.4.7 openssl-3.4.1 libssh2-1.11.1 libz-1.3.1 libpcre2-10.
45 libpcap-1.10.5 nmap-libdnet-1.12 ipv6
Compiled without:
Available nsock engines: epoll poll select
```

- **Simple host scan:**
  - ➔ **Command:** *nmap 192.168.1.76*
  - ➔ **Purpose:** to quickly confirm a specific target is up, alive, and reachable on the network, and to get a basic list of its open ports.

```
┌──(paxton㉿kali)-[~]
└─$ nmap 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 10:23 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0044s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 4.41 seconds
```

- **Basic port scan:**
  - ➔ **Command:** *nmap -p 80,234,443,8080 192.168.1.76* and *nmap -p 60-80 192.168.1.76*
  - ➔ **Purpose:** To determine whether specific ports are open and listening for connections. This identifies the services running on a host (e.g., web server, mail server).

```
┌──(paxton㉿kali)-[~]
└─$ nmap -p 80,234,443,8080 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 10:45 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0099s latency).

PORT     STATE    SERVICE
80/tcp   open     http
234/tcp  filtered unknown
443/tcp  filtered https
8080/tcp filtered http-proxy

Nmap done: 1 IP address (1 host up) scanned in 1.42 seconds
```

```
┌──(paxton㉿kali)-[~]
└─$ nmap -p 60-80 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 10:47 +0545
Nmap scan report for 192.168.1.76
Host is up (0.00087s latency).

PORT   STATE    SERVICE
60/tcp filtered unknown
61/tcp filtered ni-mail
62/tcp filtered acas
63/tcp filtered via-ftp
64/tcp filtered covia
65/tcp filtered tacacs-ds
66/tcp filtered sqlnet
67/tcp filtered dhcps
68/tcp filtered dhcpc
69/tcp filtered tftp
70/tcp filtered gopher
71/tcp filtered netrjs-1
72/tcp filtered netrjs-2
73/tcp filtered netrjs-3
74/tcp filtered netrjs-4
75/tcp filtered priv-dial
76/tcp filtered deos
77/tcp filtered priv-rje
78/tcp filtered vettcp
79/tcp filtered finger
80/tcp open     http

Nmap done: 1 IP address (1 host up) scanned in 1.57 seconds
```

- **Scanning multiple Ips**
  - ➔ **Command:** *nmap 192.168.1.76 192.168.1.75* and *nmap 192.168.1.70-80*
  - ➔ **Purpose:** To efficiently gather information from a few non-contiguous targets without having to run the same command multiple times.

```
┌──(paxton㉿kali)-[~]
└─$ nmap 192.168.1.76 192.168.1.75
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 10:50 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0062s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap scan report for 192.168.1.75
Host is up (0.0055s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3306/tcp open  mysql

Nmap done: 2 IP addresses (2 hosts up) scanned in 6.52 seconds
```

- **Scanning Subnets**
  - ➔ **Command:** *nmap 192.168.1.0/24*
  - ➔ **Purpose:** To perform host discovery across an entire range or network segment (subnet) to quickly map all the devices that are currently active and available.

- **Scanning file of Ips:**
  - → **Command:** *nmap -iL ipListFile.txt*
  - → **Purpose:** To perform scanning on list of Ips saved in a file

```
┌──(paxton㉿kali)-[~]
└─$ nmap -iL ipListFile.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 11:09 +0545
Nmap scan report for 192.168.1.70
Host is up (0.0030s latency).
All 1000 scanned ports on 192.168.1.70 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 192.168.1.75
Host is up (0.0031s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3306/tcp open  mysql

Nmap scan report for 192.168.1.76
Host is up (0.025s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap scan report for 192.168.1.80
Host is up (0.0023s latency).
All 1000 scanned ports on 192.168.1.80 are in ignored states.
```

- **Other Commands:**
  - → **Command:** *nmap -p ftp* 192.168.1.76*
  - → **Purpose:** To perform scanning on ports running <ftp> related services.

```
┌──(paxton㉿kali)-[~]
└─$ nmap -p ftp* 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 12:22 +0545
Nmap scan report for 192.168.1.76
Host is up (0.00066s latency).

PORT     STATE    SERVICE
20/tcp   filtered ftp-data
21/tcp   open     ftp
574/tcp  filtered ftp-agent
989/tcp  filtered ftps-data
990/tcp  filtered ftps
8021/tcp filtered ftp-proxy

Nmap done: 1 IP address (1 host up) scanned in 1.33 seconds
```

```
┌──(paxton㉿kali)-[~]
└─$ nmap -p tcp* 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 12:24 +0545
Nmap scan report for 192.168.1.76
Host is up (0.00064s latency).

PORT     STATE    SERVICE
1/tcp    filtered tcpmux
475/tcp  filtered tcpnethaspsrv
1999/tcp filtered tcp-id-port
3805/tcp filtered tcpdataserver

Nmap done: 1 IP address (1 host up) scanned in 1.33 seconds
```

3. Port Scanning Techniques

- **TCP SYN Scan (-sS):**
  - ➔ **Command:** *nmap -sS 192.168.1.76*
  - ➔ **Working:** Also known as a "stealth scan" or half-open scan. Nmap sends a SYN packet and, if it receives a SYN/ACK (indicating an open port), it immediately sends a RST (reset) packet instead of completing the full three-way handshake.
  - ➔ **Best For:** Stealth and speed. It's the default and most popular scan because it is fast and often bypasses simple logging mechanisms on the target machine, as a full connection is never established.

```
┌──(paxton㉿kali)-[~]
└─$ nmap -sS 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 12:46 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0036s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 4.32 seconds
```

- **TCP Connect Scan (-sT):**
  - → **Command:** *nmap -sT 192.168.1.76*
  - → **Working:** The standard, full TCP three-way handshake scan. Nmap requests a full connection (SYN -> SYN/ACK -> ACK) and lets the operating system handle the process.
  - → **Best For:** Reliability when lacking privileges. It is used when the user running Nmap does not have the necessary raw socket privileges (e.g., as a non-root user), but it is noisier as the connection is fully logged by the target OS.

```
└─$ nmap -sT 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 12:47 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0048s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 4.72 seconds
```

- **UDP Scan (-sU):**
  - ➔ **Command:** *nmap -sU 192.168.1.76*
  - ➔ **Working:** Scans for open UDP ports. Since UDP is connectionless, Nmap sends a UDP packet. If an ICMP Port Unreachable error is received, the port is closed. If no response is received, the port is likely open or filtered.
  - ➔ **Best For:** Finding non-TCP services like DNS (port 53), SNMP (port 161), or DHCP. It is often slower and less definitive than TCP scans due to the lack of a handshake.

```
└─$ nmap -sU 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 12:53 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0065s latency).
Not shown: 996 open|filtered udp ports (no-response)
PORT     STATE SERVICE
53/udp   open  domain
111/udp  open  rpcbind
137/udp  open  netbios-ns
2049/udp open  nfs

Nmap done: 1 IP address (1 host up) scanned in 4.37 seconds
```

- **Idle Scan (-sI):**
  - ➔ **Command:** *nmap -sI <zombie ip eg. 192.168.8.134> 192.168.1.76*
  - ➔ **Working:** A highly advanced, truly blind port scan. It spoofs the IP address of a "zombie" host that is idle and IP ID sequenceable, using the changes in the zombie host's IP ID field to determine if the target port is open or closed.
  - ➔ **Best For:** Ultimate anonymity. This scan conceals the attacker's source IP address, as all packets appear to come from the "zombie" host. It's complex to set up and requires finding a suitable zombie.

4. Service, Version and OS Detection

- **Service Version detection (-sV):**
  - ➔ **Command:** *nmap -sV 192.168.1.76*
  - ➔ **Purpose:** It doesn't just say "HTTP is open," it says "Apache 2.4.49 is open." This precision allows you to look up known vulnerabilities (CVEs) specific to that exact version.

```
└─$ nmap -sV 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 13:07 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0036s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE      VERSION
21/tcp   open  ftp          vsftpd 2.3.4
22/tcp   open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet       Linux telnetd
25/tcp   open  smtp         Postfix smtpd
80/tcp   open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind      2 (RPC #100000)
139/tcp  open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec         netkit-rsh rexecd
513/tcp  open  login?
514/tcp  open  tcpwrapped
1099/tcp open  rmiregistry?
1524/tcp open  bindshell    Metasploitable root shell
2049/tcp open  nfs          2-4 (RPC #100003)
2121/tcp open  ftp          ProFTPD 1.3.1
3306/tcp open  mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc          VNC (protocol 3.3)
6000/tcp open  X11          (access denied)
6667/tcp open  irc          UnrealIRCd
8009/tcp open  ajp13        Apache Jserv (Protocol v1.3)
8180/tcp open  http         Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 162.12 seconds
```

- **OS detection (-O):**
  - ➔ **Command:** *nmap -O 192.168.1.76*
  - ➔ **Purpose:** Knowing the target's operating system (e.g., Windows, Linux, Cisco IOS) helps you tailor your approach, as different OSes have different security features, default file structures, and response behaviors.

```
┌──(paxton㉿kali)-[~]
└─$ nmap -O 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 13:11 +0545
Nmap scan report for 192.168.1.76
Host is up (0.00096s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge|VoIP adapter|general purpose
Running (JUST GUESSING): Oracle Virtualbox (98%), Slirp (98%), AT&T embedded (95%), QEMU (94%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:danny_gasparovski:slirp cpe:/a:qemu:qemu
Aggressive OS guesses: Oracle Virtualbox Slirp NAT bridge (98%), AT&T BGW210 voice gateway (95%), QEMU user mode network gateway (94
%)
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.25 seconds
```

- **Aggressive Mode (-A):**
  - ➔ **Command:** *nmap -A 192.168.1.76*
  - ➔ **Purpose:** It automatically combines -O (OS Detection), -sV (Version Detection), and a few other features like script scanning and traceroute.

5. NSE (Nmap Scripting Engine)

The Nmap Scripting Engine (NSE) is a core Nmap feature designed to significantly expand its capabilities beyond basic host and port discovery by allowing users to execute powerful, automated scripts written in the lightweight Lua programming language. This engine works by running scripts against discovered targets and services in categories like default, vuln, auth, and brute, enabling deeper interaction than standard scans. It transforms Nmap from a scanner into a versatile auditing tool capable of advanced service detection, automated vulnerability checking (identifying common security flaws and misconfigurations), security auditing, and credential testing, dramatically increasing the efficiency and investigative depth of any network assessment using a simple command like --script=<category or name>

- **Brute force script example:**
  - ➔ **Command:** *nmap --script=ssh-brute.nse -p 22 192.168.1.76*
  - ➔ **Purpose:** to perform a brute-force password guessing attack against the Secure Shell (SSH) service running on the target host.

```
┌──(paxton㉿kali)-[~]
└─$ nmap --script=ssh-brute.nse -p 22 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-25 09:41 +0545
NSE: [ssh-brute] Trying username/password pair: root:root
NSE: [ssh-brute] Trying username/password pair: admin:admin
NSE: [ssh-brute] Trying username/password pair: administrator:administrator
NSE: [ssh-brute] Trying username/password pair: webadmin:webadmin
NSE: [ssh-brute] Trying username/password pair: sysadmin:sysadmin
NSE: [ssh-brute] Trying username/password pair: netadmin:netadmin
NSE: [ssh-brute] Trying username/password pair: guest:guest
NSE: [ssh-brute] Trying username/password pair: user:user
NSE: [ssh-brute] Trying username/password pair: web:web
NSE: [ssh-brute] Trying username/password pair: test:test
NSE: [ssh-brute] Trying username/password pair: root:
NSE: [ssh-brute] Trying username/password pair: admin:
NSE: [ssh-brute] Trying username/password pair: administrator:
NSE: [ssh-brute] Trying username/password pair: webadmin:
NSE: [ssh-brute] Trying username/password pair: sysadmin:
NSE: [ssh-brute] Trying username/password pair: netadmin:
NSE: [ssh-brute] Trying username/password pair: guest:
NSE: [ssh-brute] Trying username/password pair: web:
NSE: [ssh-brute] Trying username/password pair: test:
NSE: [ssh-brute] Trying username/password pair: root:123456
NSE: [ssh-brute] Trying username/password pair: admin:123456
NSE: [ssh-brute] Trying username/password pair: administrator:123456
NSE: [ssh-brute] Trying username/password pair: webadmin:123456
NSE: [ssh-brute] Trying username/password pair: sysadmin:123456
NSE: [ssh-brute] Trying username/password pair: netadmin:123456
NSE: [ssh-brute] Trying username/password pair: guest:123456
```

- **ftp scripts:**
  - → **Command:** *nmap --script=ftp-vsftpd-backdoor.nse 192.168.1.76*
  - → **Purpose:** to test for the presence and exploitability of a malicious backdoor that was secretly inserted into the source code of vsFTPd version 2.3.4.

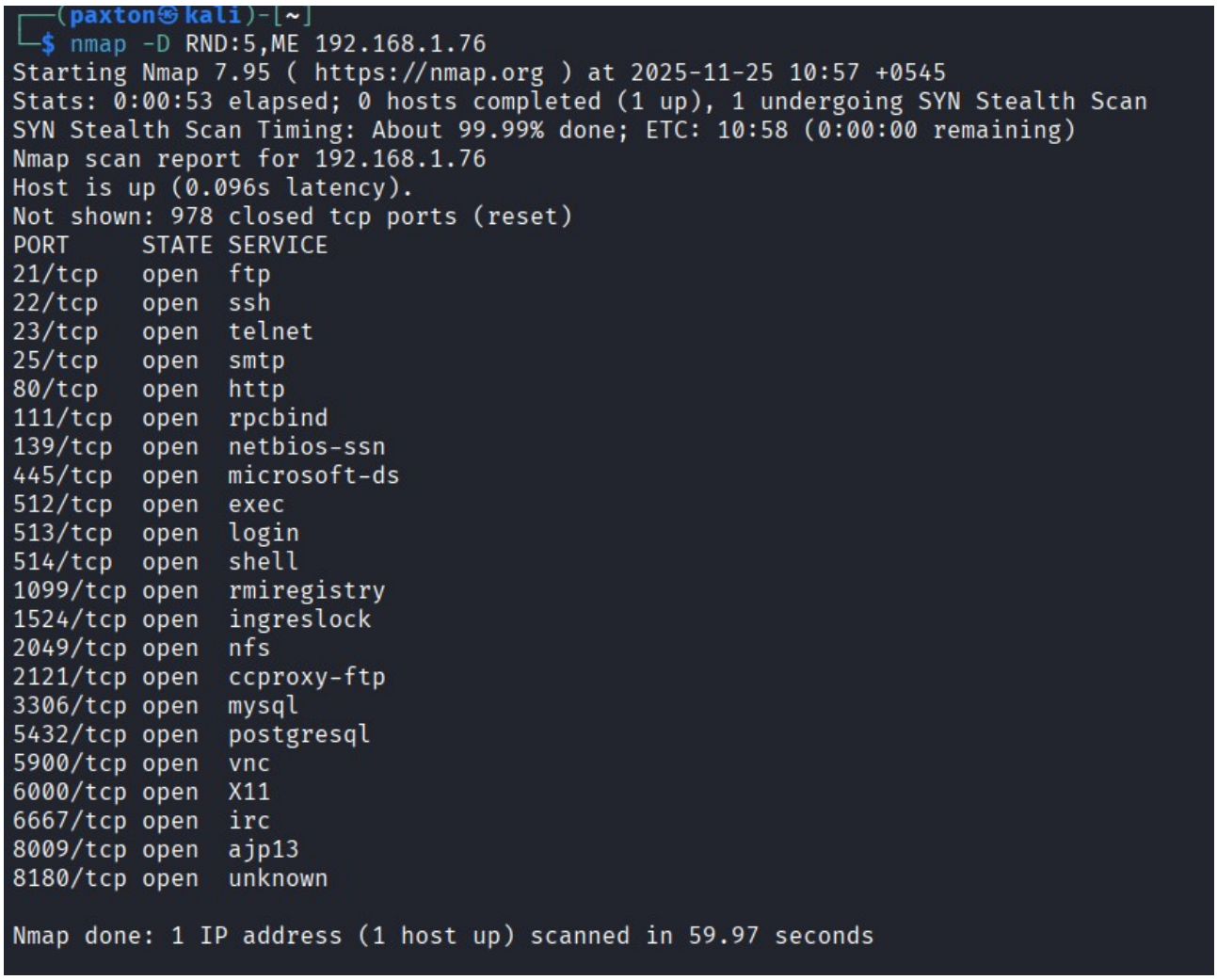
```
┌──(paxton㉿kali)-[~]
└─$ nmap --script=ftp-vsftpd-backdoor.nse 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-25 09:47 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0046s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|_      https://www.securityfocus.com/bid/48539
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 7.27 seconds
```

6. Firewall Evasion & Stealth

These commands are designed to make an Nmap scan less detectable by security systems or to confuse the target's logs.

I. **Decoy Scan: -D <decoy1>[,<decoy2>,...][,ME][,...]**
   - **Command:** *nmap -D RND:5,ME 192.168.1.76* (Scans with 5 random IPs and your real IP, which is represented by ME).
   - **Working:** Makes the target believe multiple hosts are scanning them simultaneously.Nmap sends scan probes from the real IP address *and* from the spoofed IP addresses (decoys) you provide. The target's security logs will show port scans coming from many different sources, making it difficult to pinpoint the real attacker's IP.



II. **IP Fragmentation: -f**
   - **Command :** *nmap -f 192.168.1.76*
   - **Working:** Fragment Packets. Causes the scan's IP packets to be split into tiny fragments. The idea is to split the TCP header over several small packets, making it difficult for simple packet filters or older IDSs to reassemble and detect the scan type.

```
┌──(paxton㉿kali)-[~]
└─$ nmap -f 192.168.1.76
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-25 11:11 +0545
Nmap scan report for 192.168.1.76
Host is up (0.0036s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 4.20 seconds
```

III. Source IP Address Spoofing

- **Command:** *nmap -S <spoofed_ip_address> <target_ip>*
- **Working:** Tells Nmap to use a different source IP address for the scan packets. This makes the target system believe a *completely different host* is performing the scan. You will not receive scan results back unless you are able to capture packets on the network or perform a more advanced technique like an Idle Scan (-sI).

IV. Timing Templates

- **Command:** *nmap -T<0-5> <target_ip> eg. nmap -T2 192.168.1.76*
- **-T<0-5> :** Adjust Timing/Speed. Controls how aggressive or slow the scan should be. Slower scans are generally stealthier.

  → **-T0 (Paranoid):** Extremely slow, waits 5 minutes between probes. Designed for maximum IDS evasion.
  → **-T1 (Sneaky):** Very slow, waits 15 seconds between probes. Also for IDS evasion.
  → **-T2 (Polite):** Slows down to use less network bandwidth and target resources.

- → **-T3 (Normal):** The default speed. Aims for a balance between speed and reliability.
- → **-T4 (Aggressive):** Recommended for most fast and reliable networks. Increases timeout and parallel scanning.
- → **-T5 (Insane):** Maximum aggressiveness, very fast, but may lead to inaccurate results or network congestion.

```
┌──(paxton㉿kali)-[~]
└─$ nmap 192.168.1.76 -T2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-25 11:34 +0545
Stats: 0:00:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.30% done
Stats: 0:00:08 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.80% done
Stats: 0:00:13 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 1.14% done; ETC: 11:53 (0:18:47 remaining)


┌──(paxton㉿kali)-[~]
└─$ nmap 192.168.1.76 -T3
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-25 11:35 +0545
Stats: 0:00:02 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 39.20% done; ETC: 11:35 (0:00:05 remaining)
Stats: 0:00:04 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 94.70% done; ETC: 11:35 (0:00:00 remaining)
Nmap scan report for 192.168.1.76
Host is up (0.0045s latency).
Not shown: 978 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 4.52 seconds
```

7. Output Formatting

Used to save output of scan in different formats in file

- **Command:** *nmap <file_format> <filename.txt> <target_ip> eg. Nmap -oN nmapScan1.txt 192.168.1.76*
- **Output format:**
  - **Normal Output:** *nmap -oN <filename.txt> <target_ip>*
    saves the scan results to the specified file (scan.txt) in a human-readable, plain text format that is very similar (though not identical) to what Nmap prints directly to your terminal.

  - **XML Output:** *nmap -oX <filename.xml> <target_ip>*
    Machine Readable. Best for piping scan data into other tools, scripts, databases, or graphical frontends like Zenmap.

  - **Grepable Output:** *nmap -oG <filename.gnmap> <target_ip>*
    Easy Parsing. Designed to be easily processed by Unix command-line tools like grep, awk, and cut (though it is officially deprecated in favor of XML).

  - **All format:** *nmap -oA <basename>*
    Saves the results in all three major formats (.nmap, .xml, and .gnmap) using the specified base filename.