

Lab 8: Network Scanning & Enumeration with Nmap

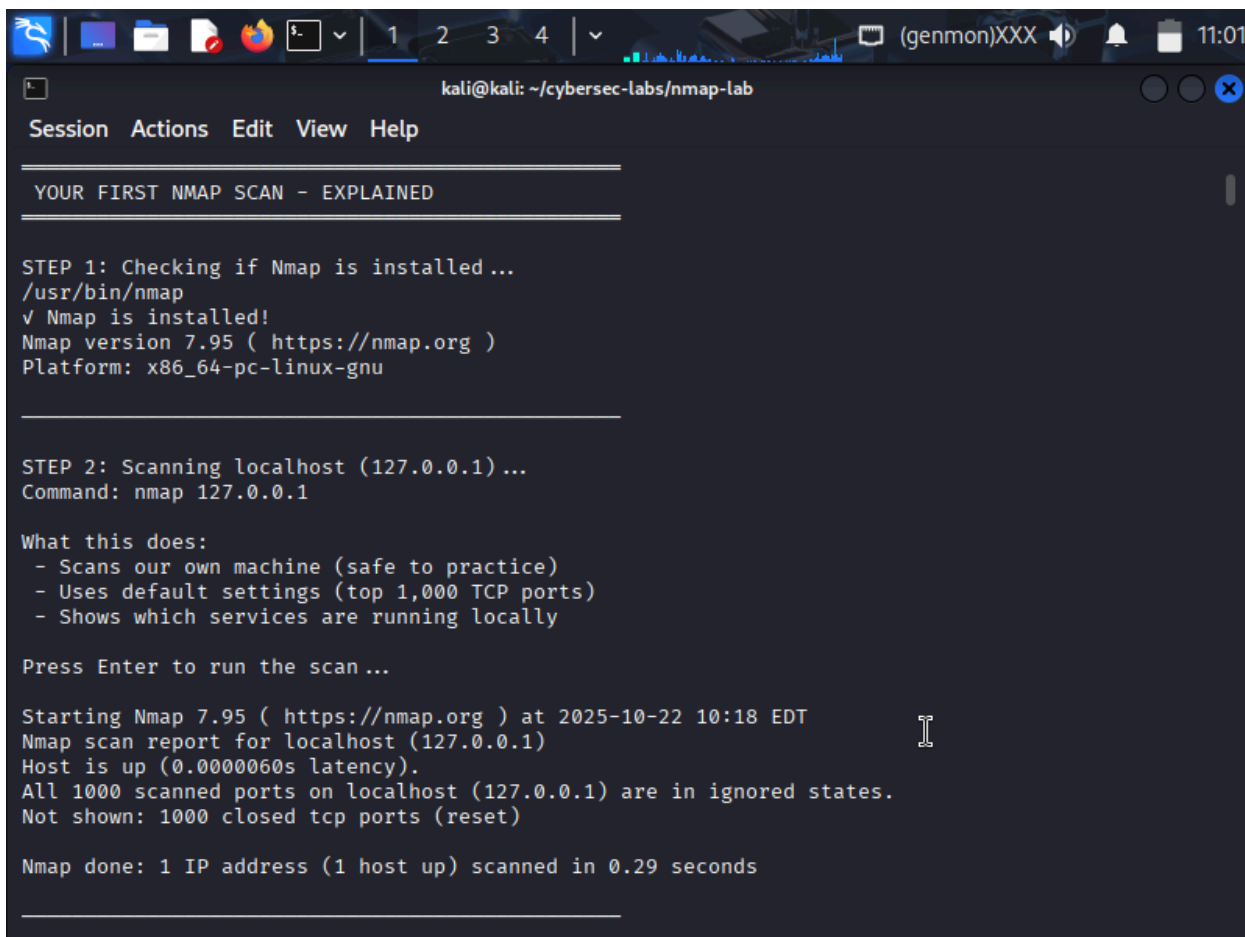
Matthew Cox - COMP 325

Introduction:

Network scanning and enumeration are foundational steps in penetration testing and security assessments. This lab utilises Nmap for network discovery and security auditing, teaching practical techniques for identifying live hosts, open ports, and running services, as well as determining service versions and operating systems. Through a series of hands-on exercises against Linux (Metasploitable) and Windows (Metasploitable3) targets, I will move from basic host discovery to advanced service enumeration and vulnerability detection using the Nmap Scripting Engine (NSE). Along the way, I will learn to interpret scan results, identify potential attack vectors, and choose appropriate scanning techniques.

Connectivity Verification

Before beginning scans, I verified connectivity to my Metasploitable3 VM. Several networking issues occurred during setup, but after correcting the virtual network settings, the host became reachable via ICMP (ping).



```
kali@kali: ~/cybersec-labs/nmap-lab
Session Actions Edit View Help

YOUR FIRST NMAP SCAN - EXPLAINED

STEP 1: Checking if Nmap is installed...
/usr/bin/nmap
√ Nmap is installed!
Nmap version 7.95 ( https://nmap.org )
Platform: x86_64-pc-linux-gnu

STEP 2: Scanning localhost (127.0.0.1)...
Command: nmap 127.0.0.1

What this does:
- Scans our own machine (safe to practice)
- Uses default settings (top 1,000 TCP ports)
- Shows which services are running locally

Press Enter to run the scan...

Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 10:18 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000060s latency).
All 1000 scanned ports on localhost (127.0.0.1) are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
```

Open port → Service is accepting connections

Filtered port → Nmap cannot determine the state due to firewall or packet filtering

-p flag → Specify ports instead of scanning only the top 1000

```
kali@kali: ~/cybersec-labs/nmap-lab
Session Actions Edit View Help

(kali@kali)~/cybersec-labs/nmap-lab
$ vim discover_targets.sh
(kali@kali)~/cybersec-labs/nmap-lab
$ chmod +x discover_targets.sh
(kali@kali)~/cybersec-labs/nmap-lab
$ ./discover_targets.sh
== Network Host Discovery ==
Scanning network for live hosts ...

== Quick Ping Scan ==
This will show all responding hosts ...
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 10:36 EDT
Nmap scan report for 192.168.1.0
Host is up (0.00032s latency).
Nmap scan report for 192.168.1.1
Host is up (0.00018s latency).
Nmap scan report for 192.168.1.2
Host is up (0.00044s latency).
Nmap scan report for 192.168.1.3
Host is up (0.00034s latency).
Nmap scan report for 192.168.1.4
Host is up (0.00018s latency).
Nmap scan report for 192.168.1.5
Host is up (0.00024s latency).
Nmap scan report for 192.168.1.6
Host is up (0.00055s latency).
Nmap scan report for 192.168.1.7
Host is up (0.00096s latency).
```

```
kali@kali: ~/cybersec-labs/nmap-lab
Session Actions Edit View Help

(kali@kali)~/cybersec-labs/nmap-lab
$ nmap $TARGET1_IP
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 11:11 EDT
Nmap scan report for 10.0.2.15
Host is up (0.0010s latency).
Not shown: 981 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-http
5985/tcp  open  wsman
7676/tcp  open  imapbrokerd
8080/tcp  open  http-proxy
8181/tcp  open  intermapper
8383/tcp  open  m2mservices
9200/tcp  open  wap-wsp
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49176/tcp open  unknown
MAC Address: 08:00:27:D7:CC:D8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds

(kali@kali)~/cybersec-labs/nmap-lab
$
```

```
kali@kali: ~/cybersec-labs/nmap-lab
Session Actions Edit View Help

(kali@kali)~/cybersec-labs/nmap-lab
$ nmap $TARGET2_IP
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 13:38 EDT
Nmap scan report for 10.0.2.15
Host is up (0.0010s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
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
8000/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:95:B6:E3 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

# Nmap done at Thu Oct 23 13:38:06 2025 -- 1 IP address (1 host up) scanned in 0.37 seconds

(kali@kali)~/cybersec-labs/nmap-lab
$
```

```
kali@kali: ~/cybersec-labs/nmap-lab
Session Actions Edit View Help

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ chmod +x banner_grab.sh

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ ./banner_grab.sh | tee banner_results.txt

Banner Grabbing Exercise

What is a banner?
- A message sent by a service when you connect
- Often reveals software name and version
- Can be grabbed manually with netcat or automatically with Nmap

— Manual FTP Banner (Target 1) —
Command: nc -w 2 192.168.1.100 21
What to expect: FTP server will announce itself

(UNKNOWN) [192.168.1.100] 21 (ftp) : Connection timed out

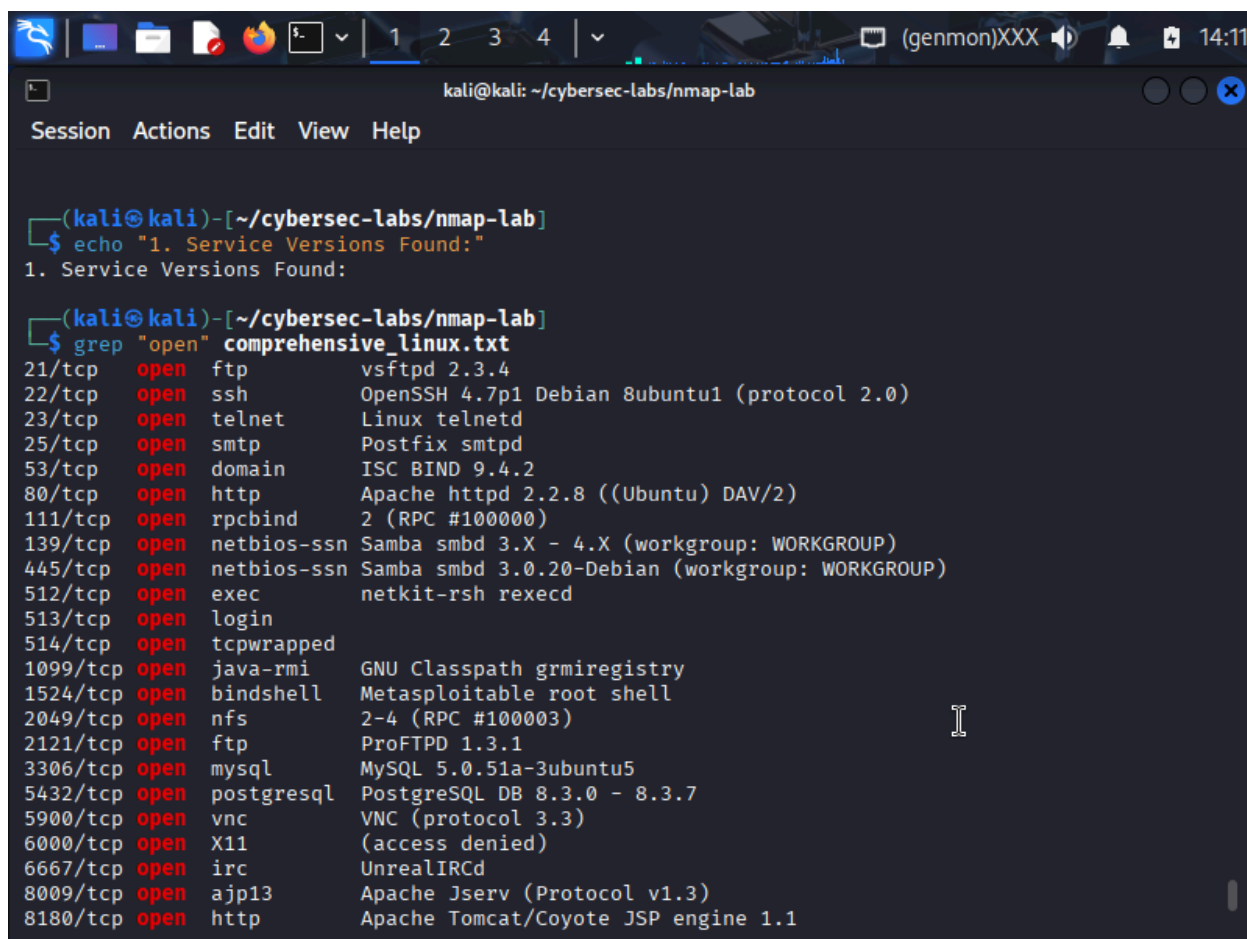
— Manual SSH Banner (Target 1) —
Command: nc -w 2 192.168.1.100 22
What to expect: SSH version string

(UNKNOWN) [192.168.1.100] 22 (ssh) : Connection timed out
./banner_grab.sh: line 30: syntax error near unexpected token `('
./banner_grab.sh: line 30: `echo "— Manual HTTP Banner (Target 1) —"'

(kali@kali)-[~/cybersec-labs/nmap-lab]
$
```

I used both Nmap's service detection and a custom banner-grabbing script: `nmap -sV <target>`

- Knowing specific versions (e.g., *Apache 2.2.8*) is more valuable than generic service names.
- `--version-intensity` ranges from 0 (fast/light) to 9 (aggressive/slow).



```
(kali@kali)-[~/cybersec-labs/nmap-lab]
$ echo "1. Service Versions Found:"
1. Service Versions Found:

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ grep "open" comprehensive_linux.txt
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
53/tcp    open  domain   ISC BIND 9.4.2
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 3.0.20-Debian (workgroup: WORKGROUP)
512/tcp   open  exec     netkit-rsh rexecd
513/tcp   open  login
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi  GNU Classpath grmiregistry
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
```

I used multiple script categories:

- default → Safe, commonly-used enumeration scripts
- safe → Non-intrusive, minimal risk
- vuln → Checks for known vulnerabilities

```
kali@kali: ~/cybersec-labs/nmap-lab
Session Actions Edit View Help
(kali@kali)-[~/cybersec-labs/nmap-lab]
$ echo "2. OS Detection:"
2. OS Detection:

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ grep -A 5 "OS details:" comprehensive_linux.txt
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE
: cpe:/o:linux:linux_kernel

Host script results:
|_smb2-time: Protocol negotiation failed (SMB2)

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ echo ""

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ echo "3. Traceroute Results:"
3. Traceroute Results:

(kali@kali)-[~/cybersec-labs/nmap-lab]
$ grep -A 10 "TRACEROUTE" comprehensive_linux.txt
TRACEROUTE
HOP RTT ADDRESS
1 1.13 ms 10.0.2.5

OS and Service detection performed. Please report any incorrect results at https://nmap.org/sub
mit/ .
# Nmap done at Thu Oct 23 14:04:47 2025 -- 1 IP address (1 host up) scanned in 22.41 seconds
```

- OS detection requires root privileges because raw packets must be crafted and analyzed.
- -A enables aggressive scanning, including -O, service detection, traceroute, and NSE scripts.
- A network distance of 1 hop indicates the target is on the same local network. Service banners often provide more reliable OS clues than TCP/IP fingerprinting alone.