Acknowledgment

This material is taken from /based on the forum “TryHackMe” (https://tryhackme.com/paths)

Nmap

Room 4: Nmap Post Port Scans

Task 1: Introduction

This room focuses on how Nmap can be used to:

- Detect versions of the running services (on all open ports)
- Detect the OS based on any signs revealed by the target
- Run Nmap’s traceroute
- Run select Nmap scripts
- Save the scan results in various formats

Task2: Service Detection

Once Nmap discovers open ports, you can probe the available port to detect the running service. Further investigation of open ports is an essential piece of information as the pentester can use it to learn if there are any known vulnerabilities of the service

Adding `-sV` to your Nmap command will collect and determine service and version information for the open ports. You can control the intensity with `--version-intensity LEVEL` where the level ranges between 0, the lightest, and 9, the most complete. `-sV --version-light` has an intensity of 2, while `-sV --version-all` has an intensity of 9.

It is important to note that using `-sV` will force Nmap to proceed with the TCP 3-way handshake and establish the connection. The connection establishment is necessary because Nmap cannot discover the version without establishing a connection fully and communicating with the listening service. In other words, stealth SYN scan `-sS` is not possible when `-sV` option is chosen.

The console output below shows a simple Nmap stealth SYN scan with the `-sV` option. Adding the `-sV` option leads to a new column in the output showing the version for each detected service. For instance, in the case of TCP port 22 being open, instead of `22/tcp open ssh`, we obtain `22/tcp open ssh OpenSSH 6.7p1 Debian 5+deb8u8 (protocol 2.0)`. Notice that the SSH protocol is guessed as the service because TCP port 22 is open; Nmap didn’t need to connect to port 22 to confirm. However, `-sV` required connecting to this open port to grab the service banner and any version information it can get, such as `nginx 1.6.2`. Hence, unlike the *service* column, the *version* column is not a guess.

Pentester Terminal

```
pentester@TryHackMe$ sudo nmap -sV MACHINE_IP
```

```
Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-10 05:03 BST
Nmap scan report for MACHINE_IP
Host is up (0.0040s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 6.7p1 Debian 5+deb8u8 (protocol 2.0)
25/tcp  open  smtp    Postfix smtpd
80/tcp  open  http    nginx 1.6.2
110/tcp open  pop3    Dovecot pop3d
111/tcp open  rpcbind 2-4 (RPC #100000)
MAC Address: 02:A0:E7:B5:B6:C5 (Unknown)
Service Info: Host:  debra2.thm.local; OS: 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 8.40 seconds
```

Note that many Nmap options require root privileges. Unless you are running Nmap as root, you need to use `sudo` as in the example above.

Start the VM. Once it is ready, open the terminal on the AttackBox to answer the following questions.

Task3: OS Detection and Traceroute

OS Detection

Nmap can detect the Operating System (OS) based on its behaviour and any telltale signs in its responses. OS detection can be enabled using `-O`; this is an *uppercase O* as in OS. In this example, we ran `nmap -sS -O MACHINE_IP` on the AttackBox. Nmap detected the OS to be Linux 3.X, and then it guessed further that it was running kernel 3.13.

```
Pentester Terminal
pentester@TryHackMe$ sudo nmap -sS -O MACHINE_IP

Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-10 05:04 BST
Nmap scan report for MACHINE_IP
Host is up (0.00099s latency).
Not shown: 994 closed ports
PORT    STATE SERVICE
```

```
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
110/tcp open  pop3
111/tcp open  rpcbind
143/tcp open  imap
MAC Address: 02:A0:E7:B5:B6:C5 (Unknown)
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3.13
OS details: Linux 3.13
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.91 seconds
```

The system that we scanned and attempted to detect its OS version is running kernel version 3.16. Nmap was able to make a close guess in this case. In another case, we scanned a Fedora Linux system with kernel 5.13.14; however, Nmap detected it as Linux 2.6.X. The good news is that Nmap detected the OS correctly; the not-so-good news is that the kernel version was wrong.

The OS detection is very convenient, but many factors might affect its accuracy. First and foremost, Nmap needs to find at least one open and one closed port on the target to make a reliable guess. Furthermore, the guest OS fingerprints might get distorted due to the rising use of virtualization and similar technologies. Therefore, always take the OS version with a grain of salt.

Traceroute

If you want Nmap to find the routers between you and the target, just add `--traceroute`. In the following example, Nmap appended a traceroute to its scan results. Note that Nmap's traceroute works slightly different than the `traceroute` command found on Linux and macOS or `tracert` found on MS Windows. Standard `traceroute` starts with a packet of low TTL (Time to Live) and keeps increasing until it reaches the target. Nmap's traceroute starts with a packet of high TTL and keeps decreasing it.

In the following example, we executed `nmap -sS --traceroute MACHINE_IP` on the AttackBox. We can see that there are no routers/hops between the two as they are connected directly.

```
Pentester Terminal
pentester@TryHackMe$ sudo nmap -sS --traceroute MACHINE_IP

Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-10 05:05 BST
Nmap scan report for MACHINE_IP
Host is up (0.0015s latency).
Not shown: 994 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
110/tcp open  pop3
111/tcp open  rpcbind
143/tcp open  imap
MAC Address: 02:A0:E7:B5:B6:C5 (Unknown)

TRACEROUTE
HOP RTT     ADDRESS
1   1.48 ms MACHINE_IP

Nmap done: 1 IP address (1 host up) scanned in 1.59 seconds
```

It is worth mentioning that many routers are configured not to send ICMP Time-to-Live exceeded, which would prevent us from discovering their IP addresses. For more information, visit the Active Reconnaissance room.

Summary

In this room, we learned how to detect the running services and their versions along with the host operating system. We learned how to enable traceroute and we covered selecting one or more scripts to aid in penetration testing. Finally, we covered the different formats to save the scan results for future reference. The table below summarizes the most important options we covered in this room.

Option	Meaning
`-sV`	determine service/version info on open ports
`-sV --version-light`	try the most likely probes (2)
`-sV --version-all`	try all available probes (9)

Option	Meaning
`-O`	detect OS
`--traceroute`	run traceroute to target
`--script=SCRIPTS`	Nmap scripts to run
`-sC` or `--script=default`	run default scripts
`-A`	equivalent to `-sV -O -sC --traceroute`
`-oN`	save output in normal format
`-oG`	save output in grepable format
`-oX`	save output in XML format
`-oA`	save output in normal, XML and Grepable formats