

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING
Department of Computer Engineering

Course , Subject & Experiment Details

Academic Year	2019-20	Estimated Time	02 - Hours
Course & Semester	T.E. (CMPN)- Sem VI	Subject Name & Code	CSS - (CSL604)
Module No.	0 5– Mapped to CO-3	Chapter Title	Network Security and Applications

Practical No:	7
Title:	Port scanning and OS fingerprinting using NMAP
Date of Performance:	
Date of Submission:	
Roll No:	
Name of the Student:	

Evaluation:

Sr. No	Rubric	Grade
1	On time submission Or completion (2)	
2	Preparedness(2)	
3	Skill (4)	
4	Output (2)	

Signature of the Teacher:

Date:

MNS

Title: Port scanning and OS fingerprinting using NMAP

Lab Scenario:

Network Mapped (Nmap) is a network scanning and host detection tool that is very useful during several steps of penetration testing. Nmap is not limited to merely gathering information and enumeration, but it is also powerful utility that can be used as a vulnerability detector or a security scanner. So Nmap is a multipurpose tool, and it can be run on many different operating systems including Windows, Linux, BSD, and Mac.

Lab Objectives:

- Detect the live host on the network (host discovery)
- Detect the open ports on the host (port discovery or enumeration)
- Detect the software and the version to the respective port (service discovery)
- Detect the operating system, hardware address, and the software version
- Detect the vulnerability and security holes (Nmap scripts)

Lab Environment:

To carry out this experiment you need:

- Install Kali linux as your Operating System.

Lab Tasks:

The usage of Nmap depends on the target machine because there is a difference between simple (basic) scanning and advance scanning. We need to use some advanced techniques to bypass the firewall and intrusion detection/preventative software to get the right result.

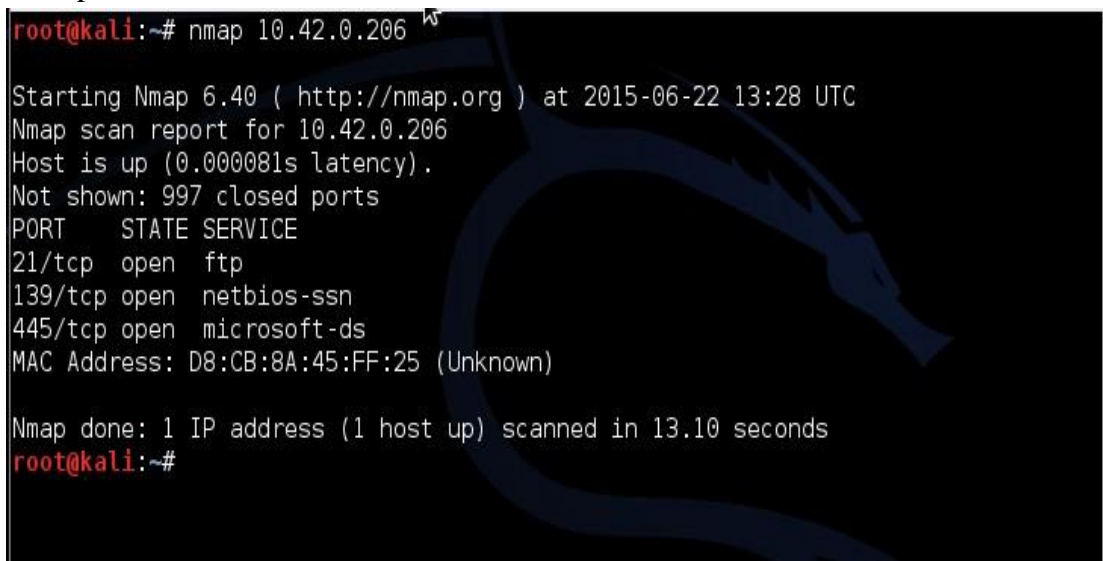
Nmap Scanning Commands:

If you want to scan a single system, then you can use a simple command

nmap targetIP

nmap target.com

#nmap 10.42.0.206

A terminal window with a Kali Linux dragon logo background. The prompt is root@kali:~#. The command nmap 10.42.0.206 has been executed. The output shows the Nmap version (6.40), the scan time (2015-06-22 13:28 UTC), and the scan report for 10.42.0.206. The host is up with a latency of 0.000081s. 997 closed ports are not shown. Open ports are listed: 21/tcp (ftp), 139/tcp (netbios-ssn), and 445/tcp (microsoft-ds). The MAC address is D8:CB:8A:45:FF:25 (Unknown). The scan took 13.10 seconds to complete.

```
root@kali:~# nmap 10.42.0.206
Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 13:28 UTC
Nmap scan report for 10.42.0.206
Host is up (0.000081s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: D8:CB:8A:45:FF:25 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 13.10 seconds
root@kali:~#
```

If you want to scan the entire subnet, then the command is

nmap target/subnet mask

nmap 10.42.0.0/24

It is very easy to scan a multiple targets, all you need to do is to separate each target via space:

nmap target target1 target2

nmap 192.168.1.1 192.168.1.8

Let suppose you have a list of a target machines. You can make Nmap scan for the entire list:

nmap -iL target.txt (Make sure to put the file on the same directory)

MNS

You can see that the below command with “-v” option is giving more detailed information about the remote machine.

```
#nmap -v 10.42.0.206
```

```
root@kali:~# nmap -v 10.42.0.206

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 14:51 UTC
Initiating ARP Ping Scan at 14:51
Scanning 10.42.0.206 [1 port]
Completed ARP Ping Scan at 14:51, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:51
Completed Parallel DNS resolution of 1 host. at 14:51, 13.00s elapsed
Initiating SYN Stealth Scan at 14:51
Scanning 10.42.0.206 [1000 ports]
Discovered open port 139/tcp on 10.42.0.206
Discovered open port 445/tcp on 10.42.0.206
Discovered open port 21/tcp on 10.42.0.206
Completed SYN Stealth Scan at 14:51, 0.06s elapsed (1000 total ports)
Nmap scan report for 10.42.0.206
Host is up (0.000096s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: D8:CB:8A:45:FF:25 (Unknown)

Read data files from: /usr/bin/../../share/nmap
```

In some cases we need to scan the entire subnet but not a specific IP addresses because it might be dangerous for us. In this scenario, use the Nmap command with the excluding parameter:

```
# nmap 10.42.0.0/24 --exclude 10.42.0.247
```

If you have a file that contains the list of IP addresses that you want to exclude, then you can call the file in the exclude parameter:

```
# nmap 10.42.0.0/24 --exclude file target.txt
```

If you want to scan a specific port on the target machines (for example, if you want to scan the HTTP, FTP, and Telnet port only on the target computer), then you can use the Nmap command with the relevant parameter:

```
# nmap -p80,21,23 192.168.1.1 // It scan the target for port number 80,21 and 23.
```

```

root@bt:~# nmap -p80,21,23 192.168.1.1

Starting Nmap 5.51 ( http://nmap.org ) at 2012-07-08 17:18 PKT
Nmap scan report for 192.168.1.1
Host is up (0.00064s latency).
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
80/tcp    open  http
MAC Address: 00:22:93:CF:EB:6D (ZTE)

Nmap done: 1 IP address (1 host up) scanned in 0.30 seconds

```

Nmap Scanning Techniques

There are so many scanning techniques available on Nmap, some of which will be discussed in the following segment:

TCP SYN Scan (-sS)

It is a basic scan, and it is also called half-open scanning because this technique allows Nmap to get information from the remote host without the complete TCP handshake process, Nmap sends SYN packets to the destination, but it does not create any sessions, As a result, the target computer can't create any log of the interaction because no session was initiated, making this feature an advantage of the TCP SYN scan.

```
# nmap -sS 10.42.0.206
```

```

root@kali:~# nmap -sS 10.42.0.206

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 13:44 UTC
Nmap scan report for networklab-ThinkCentre-E73 (10.42.0.206)
Host is up (0.000091s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: D8:CB:8A:45:FF:25 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
root@kali:~#

```

TCP connect() scan (-sT)

This is the default scanning technique used, if and only if the SYN scan is not an option, because the SYN scan requires root privilege. Unlike the TCP SYN scan, it completes the normal TCP three way handshake process and requires the system to call connect(), which is a part of the operating system. Keep in mind that this technique is only applicable to find out the TCP ports, not the UDP ports.

```
# nmap -sT 192.168.1.1
```

```
root@kali:~# nmap -sT 10.42.0.206

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 13:46 UTC
Nmap scan report for networklab-ThinkCentre-E73 (10.42.0.206)
Host is up (0.0016s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: D8:CB:8A:45:FF:25 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
root@kali:~#
```

UDP Scan (-sU)

As the name suggests, this technique is used to find an open UDP port of the target machine. It does not require any SYN packet to be sent because it is targeting the UDP ports. But we can make the scanning more effective by using -sS along with -sU. UDP scans send the UDP packets to the target machine, and waits for a response—if an error message arrives saying the ICMP is unreachable, then it means that the port is closed; but if it gets an appropriate response, then it means that the port is open.

```
# nmap -sU 10.42.0.206
```

FIN Scan (-sF)

Sometimes a normal TCP SYN scan is not the best solution because of the firewall. IDS and IPS scans might be deployed on the target machine, but a firewall will usually block the SYN packets. A FIN scan sends the packet only set with a FIN flag, so it is not required to complete the TCP handshaking.

```

root@kali:~# nmap -sF 10.42.0.206

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 14:27 UTC
Nmap scan report for 10.42.0.206
Host is up (0.00014s latency).
Not shown: 997 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered ftp
139/tcp   open|filtered netbios-ssn
445/tcp   open|filtered microsoft-ds
MAC Address: D8:CB:8A:45:FF:25 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 14.25 seconds
root@kali:~#

```

The FIN scan sends the packets containing only the FIN flag, where as the Null scan does not send any bit on the packet, and the xmas sends FIN, PSH, and URG flags.

Ping Scan (-sP)

Ping scanning is unlike the other scan techniques because it is only used to find out whether the host is alive or not, it is not used to discover open ports. Ping scans require root access s ICMP packets can be sent, but if the user does not have administrator privilege, then the ping scan uses connect() call.

nmap -sP 10.42.0.206

```

root@kali:~# nmap -sP 10.42.0.206

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 13:52 UTC
Nmap scan report for 10.42.0.206
Host is up (0.00024s latency).
MAC Address: D8:CB:8A:45:FF:25 (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 13.02 seconds
root@kali:~#

```

Version Detection (-sV)

Version detection is the right technique that is used to find out what software version is running on the target computer and on the respective ports. It is unlike the other scanning techniques because it is not used to detect the open ports, but it requires the information from open ports to detect the software version. In the first step of this scan technique, version detection uses the TCP SYN scan to find out which ports are open.

nmap -sV 10.42.0.206


```
root@kali:~# nmap -sV 10.42.0.206

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 13:55 UTC
Nmap scan report for 10.42.0.206
Host is up (0.000076s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.2
139/tcp   open  netbios-ssn  Samba smbd 3.X (workgroup: NETWORKLAB-THINKCENTRE-E73)
445/tcp   open  netbios-ssn  Samba smbd 3.X (workgroup: NETWORKLAB-THINKCENTRE-E73)
MAC Address: D8:CB:8A:45:FF:25 (Unknown)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.40 seconds
root@kali:~#
```

Idle Scan (-sI)

Idle scan provides complete anonymity while scanning. In idle scan, Nmap doesn't send the packets from your real IP address—instead of generating the packets from the attacker machine, Nmap uses another host from the target network to send the packets. Let's consider an example to understand the concept of idle scan:

nmap -sI zombie_host target_host # nmap -sI 10.42.0.75 10.42.0.206

```
root@kali:~# nmap -sI 10.42.0.75 10.42.0.206
WARNING: Many people use -Pn w/Idlescan to prevent pings from their true IP. On
the other hand, timing info Nmap gains from pings can allow for faster, more re
liable scans.

Starting Nmap 6.40 ( http://nmap.org ) at 2015-06-22 13:58 UTC
Idle scan using zombie 10.42.0.75 (10.42.0.75:443); Class: Incremental
Nmap scan report for 10.42.0.206
Host is up (0.044s latency).
Not shown: 997 closed|filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: D8:CB:8A:45:FF:25 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 20.36 seconds
root@kali:~#
```

The idle scan technique (as mentioned above) is used to discover the open ports on 10.42.0.206 while it uses the zombie_host (10.42.0.75) to communicate with the target host. So this is an ideal technique to scan a target computer anonymously.

There are many other scanning techniques are available like FTP bounce, fragmentation scan, IP protocol scan. and so on; but we have discussed the most important scanning techniques (although all of the scanning techniques can important depending on the situation you are dealing with).

In the next section of this article, I will discuss Nmap's operating system (OS) detection and discovery techniques.

OS Detection Nmap

One of the most important feature that Nmap has is the ability to detect remote operating systems and software. It is very helpful during a penetration test to know about the operating system and the software used by the remote computer because you can easily predict the known vulnerabilities from this information.

Nmap has a database called *nmap-os-db*, the database contains information of more than 2,600 operating systems. Nmap sends TCP and UDP packets to the target machine and then it examines the response by comparing the result with the database. The Nmap operating system discovery technique is slightly slower than the scanning techniques because OS detection involves the process of finding open ports.

The example above clearly demonstrates that the Nmap first discovers the open ports, then it sends the packets to discover the remote operating system. The OS detection parameter is *-O* (capital O).

```
root@bt:~# nmap -O 192.168.1.2

Starting Nmap 5.51 ( http://nmap.org ) at 2012-07-15 10:25 PKT
Nmap scan report for 192.168.1.2
Host is up (0.000073s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
111/tcp   open  rpcbind
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.19 - 2.6.36
Network Distance: 0 hops
```

Nmap OS fingerprinting technique discovers the:

- Device type (router, work station, and so on)
- Running (running operating system)
- OS details (the name and the version of OS)
- Network distance (the distance in hops between the target and attacker)

Suppose that the target machine has a firewall, IDS, and IPS all enabled. You can use the command *-PN* to ensure that you do not ping to find the remote operating system. The *-PN* tells Nmap not to ping the remote computer, since sometimes firewalls block the request.

```
# nmap -O -PN 192.168.1.1/24
```

The command informs the sender every host on the network is alive so there is no need to send a ping request as well. In short, it bypasses the ping request and goes on to discover the operating system.

The Nmap OS detection technique works on the basis of an open and closed port. If Nmap fails to discover the open and closed port, then it gives the error:

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port

```
root@bt:~# nmap -O 192.168.1.1
Starting Nmap 5.51 ( http://nmap.org ) at 2012-07-15 10:48 PKT
Nmap scan report for 192.168.1.1
Host is up (0.00066s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
80/tcp    open  http
MAC Address: 00:22:93:CF:EB:6D (ZTE)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
```

This is an undesirable situation, and it is good to limit the operating system scans if Nmap is not sure about the OS. If Nmap is not sure about the OS, then there is no need to detect by using `-oosscan_limit`.

```
root@bt:~# nmap -O --oosscan_limit 192.168.1.1
Starting Nmap 5.51 ( http://nmap.org ) at 2012-07-15 10:48 PKT
Nmap scan report for 192.168.1.1
Host is up (0.00072s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE
23/tcp    open  telnet
80/tcp    open  http
MAC Address: 00:22:93:CF:EB:6D (ZTE)

OS detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 5.62 seconds
```

If it is very difficult for Nmap to detect the remote OS accurately, you have the option of using Nmap's guess feature: `-oosscan-guess` finds the nearest match of the target operating system.

nmap -O --oosscan-guess 192.168.1.1

Practical and Real Time Applications

- **Nmap** is used for exploring networks, perform security scans, network audit and finding open ports on remote machine.

Conclusion:

The program was tested for different sets of inputs.

Program is working SATISFACTORY NOT SATISFACTORY
(Tick appropriate outcome)

Post Lab Assignment:

1. Write commands for the scenarios given below : consider host ip as :192.168.10.4/24

- a. Scan a host using TCP ACK (PA) and TCP Syn (PS) ping
- b. Scan a host using UDP ping
- c. Find out the most commonly used TCP ports using TCP SYN Scan
- d. Scan a firewall for security weakness

2. What is GUI alternative of nmap?