**FR. CONCEICAO RODRIGUES COLLEGE OF**

**ENGINEERING**

**Department of Computer Engineering**

**1 Course, Subject & Experiment Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Academic Year** | **2021-22** | | **Estimated** |  |
| Course &  Semester | T.E. (CMPN)- Sem VI | | Subject  Name  & Code | CSS - (CPC702) |
| Chapter No. | 02 – Mapped to CO- 1 | | Chapter Title | Basics of Cryptography |
| **Practical No:** | | | 3 | | |
| **Title:** | | | Port scanning and OS fingerprinting using NMAP | | |
| **Date of Performance:** | | | 11/02/2022 | | |
| **Date of Submission:** | | | 02/03/2022 | | |
| **Roll No:** | | | 8940 | | |
| **Name of the Student:** | | | Warren Fernandes | | |

**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:**

**M**

**NS**

**Title:** IPort 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.

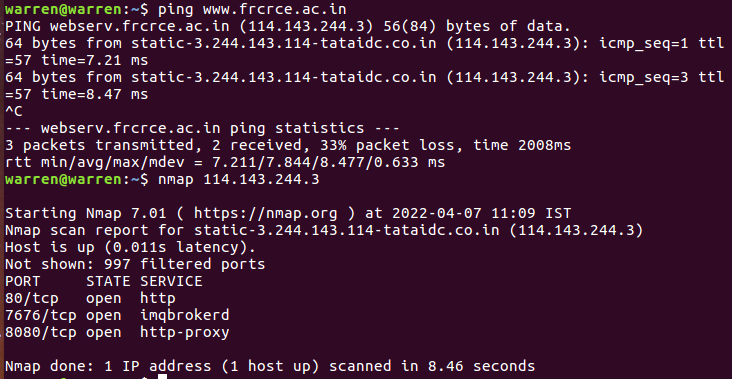
**Lab Tasks:**

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 103.250.36.83**



**M**

**NS**

If you want to scan the entire subnet, then the command is *nmap target/subnet mask*

# nmap 10.42.0.0/24

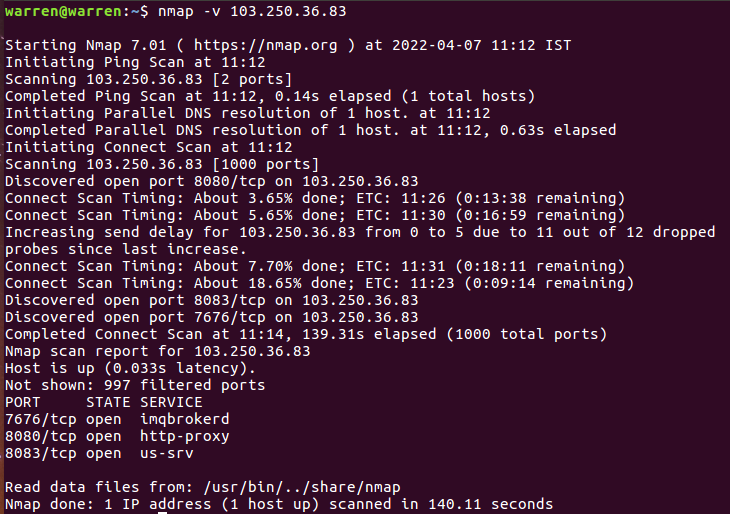
It is very easy to scan a multiple targets, all you need to do is to separate each target via space: *nmap target target1 target2*

# 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)

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



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*

**M**

**NS**

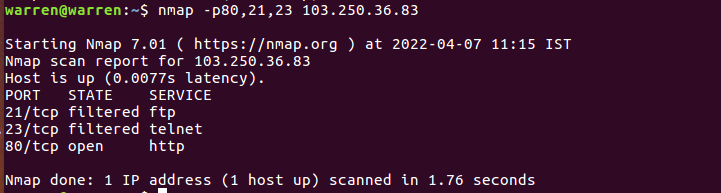
If you have a file that contains the list of IP addresses that you want to exclude, then you can call the file in the exclude parameter:

*# nmap 103.250.36.83 –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 -p80,21,23 103.250.36.83**

//It scan the target for port number 80,21 and 23



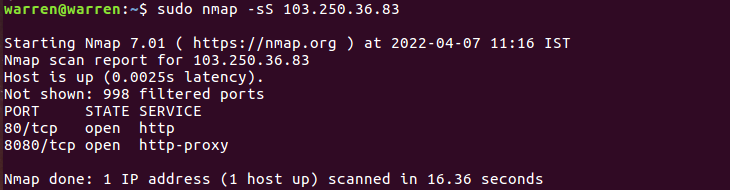
**Nmap Scanning Techniques**

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 103.250.36.83**

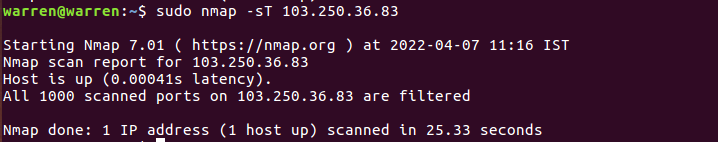


**M**

**NS**

# TCP connect() scan (-sT)

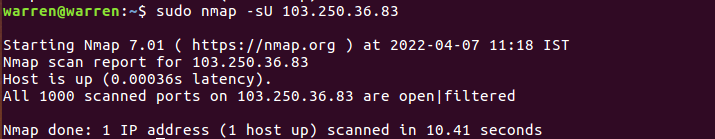
This the default scanning technique used, if and only if the SYN scan is not an option, because the SYN scan requires root privilege. Unlike the TCP SYN scan, it completes the normal TCP three way handshake process and requires the system to call connect(), which is a part of the operating system. Keep in mind that this technique is only applicable to find out the TCP ports, not the UDP ports. **# nmap -sT 103.250.36.83**



# 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 103.250.36.83**



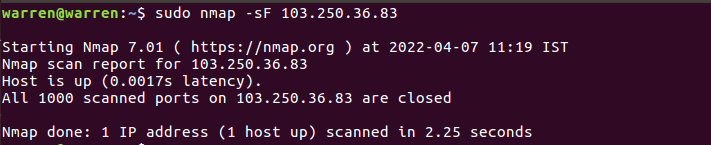
# FIN Scan (-sF)

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.

**# nmap -sF 103.250.36.83**

**M**

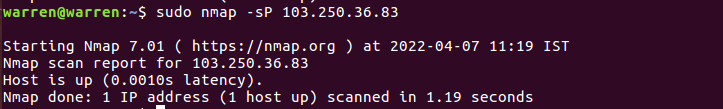
**NS**



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)

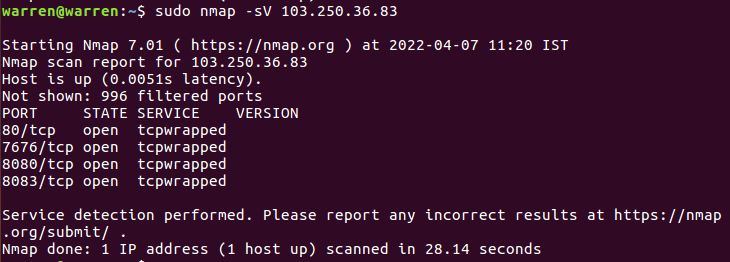
Ping scanning is unlike the other scan techniques because it is only used to find out whether the host is alive or not, it is not used to discover open ports. Ping scans require root access’s ICMP packets can be sent, but if the user does not have administrator privilege, then the ping scan uses connect() call. **# nmap -sP 103.250.36.83**



# 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 103.250.36.83s**



**M**

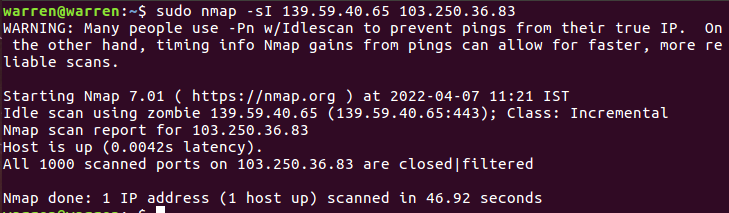
**NS**

# 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 139.59.40.65 103.250.36.83**



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

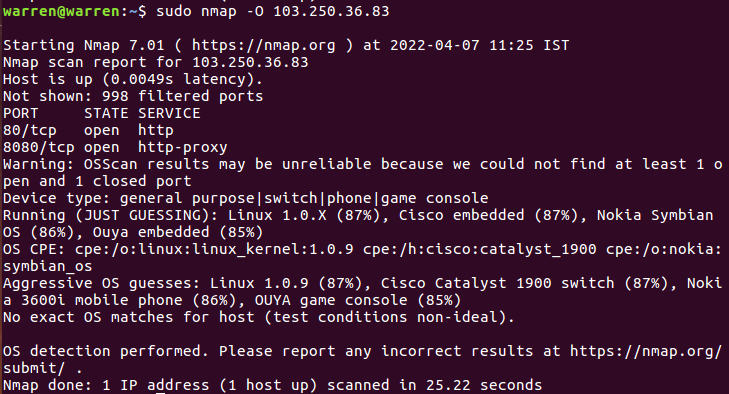
One of the most important feature that Nmap has is the ability to detect remote operating systems and software. It is very helpful during a penetration test to know about the operating system and the software used by the remote computer because you can easily predict the known vulnerabilities from this 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). **nmap -O 103.250.36.83**

**M**

**NS**



**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 103.250.36.83**

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.

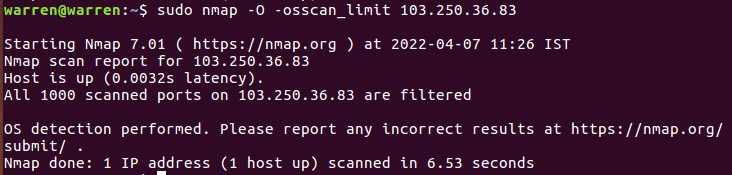
The Nmap OS detection technique works on the basis of an open and closed port. If Nmap fails to discover the open and closed port, then it gives the error:

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

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 – *osscan\_limit*.

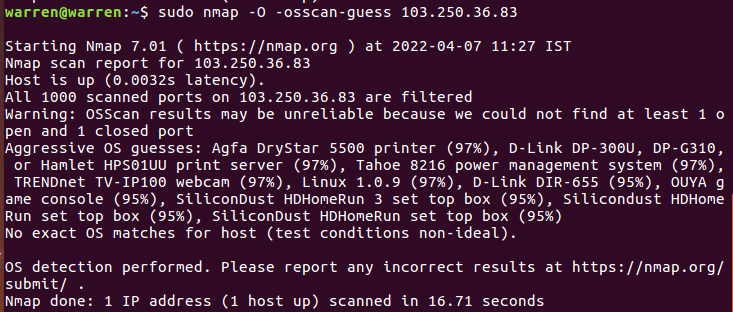
**M**

**NS**



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

**nmap -O --osscan-guess 103.250.36.83**



**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)

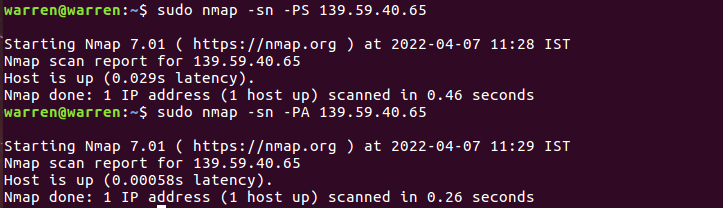
**M**

**NS**

**Post Lab Assignment:**

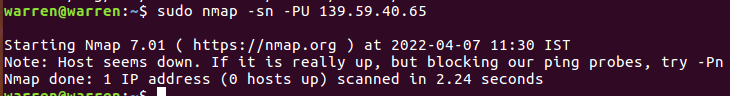
1. **Write commands for the scenarios given below consider host ip as: 139.59.40.65 a. Scan a host using TCP ACK (PA) and TCP Syn (PS) ping**

*# nmap -sn -PS 139.59.40.65 # nmap -sn -PA 139.59.40.65*



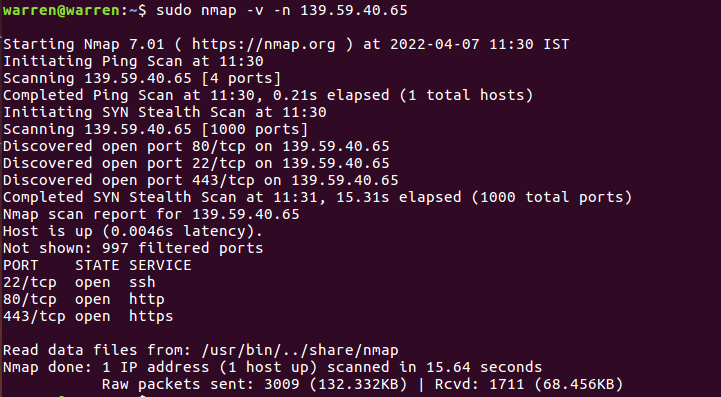
* 1. **Scan a host using UDP ping**

*# nmap -sn -PU 139.59.40.65*



* 1. **Find out the most commonly used TCP ports using TCP SYN Scan**

*nmap -v -n <target>*

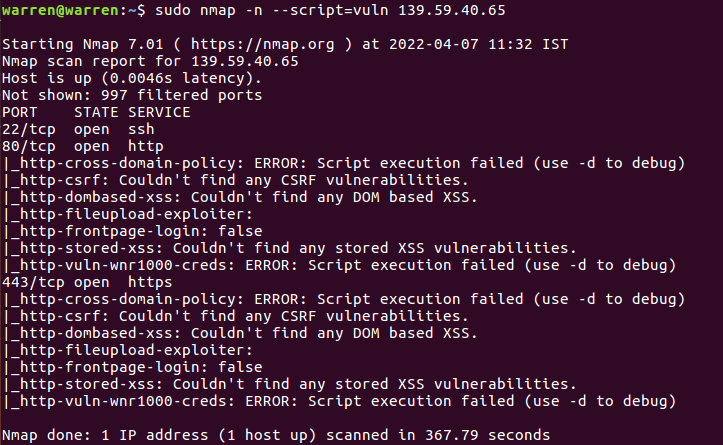


The most common TCP port which was found after scaning was port 443 and port 80 that are of HTTPS and HTTP respectively.

**M**

**NS**

* 1. **Scan a firewall for security weakness**



**M**

**NS**

1. **What is GUI alternative of nmap?**

The best alternative is Fing, which is free. Other great apps like Nmap are Angry IP Scanner (Free, Open Source), Zenmap (Free, Open Source), Advanced IP Scanner (Free) and Port Authority (Free, Open Source).

**M**

**NS**