CYBER SECURITY 2021

**P3: Scanning & Enumeration of ONE machine**

# **Step 0:**

Okay, first thing is studying some of enumeration tools. (for me I study nmap, nikto and wpscan)

# **Step 1: Prework, Study the tools**

Now I create a clone of my Kali virtual machine and change the network settings so that I have 2 machines with different IP and MAC address in my network:

A picture containing text, screenshot, monitor, indoor

Description automatically generated

Figure 1. Clone and main virtual machines

Next step is using **netdiscover** from first machine to scan my **own** network.

I use the command “sudo netdiscover -r 192.168.1.0/16” and found some live hosts:

A screen shot of a computer

Description automatically generated with low confidence

Figure 2. Basic scan with netdiscover

They are all my live devices in the network. I may use the parameter “-p” for passive mode (more quietly traffics) but it takes more time. Then I use the “-PN” parameter to get the header away and print the result directly to the terminal:

A screenshot of a computer

Description automatically generated with medium confidence

Figure 3. Use -PN in netdiscover

Interestingly, I can only find 3 hosts (my mobile phone was not found). Also, I can only use -PN parameters with network /16 (nothing can be found with /24).

Next, I use **nmap** to scan for open ports on one device, let say my clone virtual machine with IP 192.168.1.21:

“-Pn”: Treat all hosts as online -- skip host discovery

“-p 1000-4000”: specify port from 1000 to 4000

“-p –“: I do not know this

“-sV”: Probe open ports to determine service/version info

“-A”: Enable OS detection, version detection, script scanning, and traceroute (All)

“-O”: Just enable OS detection

“-sS”: use stealthy TCP SYN scan

“sT”: Connect scan

“-sU”: UDP Scan

“-sA”: Acknowledge scan (ACK)

Okay time for real example with virtual machines:

Graphical user interface, text

Description automatically generated

Figure 4. Scan for open ports

A picture containing text, indoor, screenshot, silver

Description automatically generated

Figure 5. Use TCP/IP fingerprint in OS and version scan.

When I use these “-sS -sT” parameters it yields nothing but the MAC address.

Text

Description automatically generated

Figure 6. Not quite sure...

And the -sU just stuck there forever…

I decide to move on to **enum4linux**. These are common parameters for this tool:

“-h”: Display the help message and exit (tried!)

“-U -o x.x.x.x”: List usernames, if the server allows and Pulls OS information using smbclient, this can pull the service pack version on some versions of Windows

“-a”: Do Everything, runs all options apart from dictionary-based share name guessing

Enough listing, time for real practicing with Kali.

Text

Description automatically generated

Figure 7. Using enum4linux

I don’t know for what reason, I cannot scan anything with enum4linux, even with any parameter…

OKAY, I have these machines in my network:

* The gateway (and WAP) NETGEAR 3c:37:86:0c:e0:a0
  + OS: no exact match
  + 5 open TCP ports
* Liteon Technology Corporation 80:30:49:ca:80:11 (my laptop)
  + Too many fingerprints match this host to give specific OS details
  + Cannot scan more…
* PCS Systemtechnik GmbH 08:00:27:fc:2c:26 (my main Kali virtual machine)
  + Same as my laptop…
* PCS Systemtechnik GmbH 08:00:27:63:c1:9d (my clone Kali)
  + Same as main Kali…..
* Sony Mobile Communications Inc 84:c7:ea:30:00:77 (my smartphone)
  + 1 service unrecognized despite returning data
  + No exact OS matches for host
  + 5 open TCP ports

# **Step 2:**

Installed web server: “sudo apt install apache2”

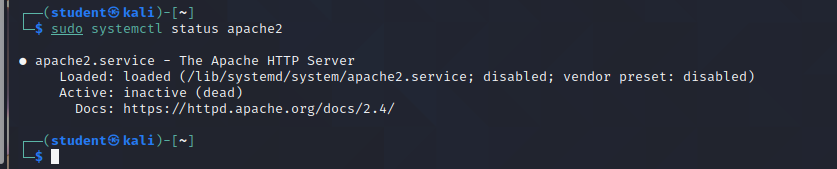


Figure 8. Check if the server is running

It’s dead now, let’s revive it. I found some useful material: <https://linoxide.com/find-how-to-stop-and-restart-apache-on-linux-systems>

Okay we can do magic with the command “sudo systemctl start apache2”

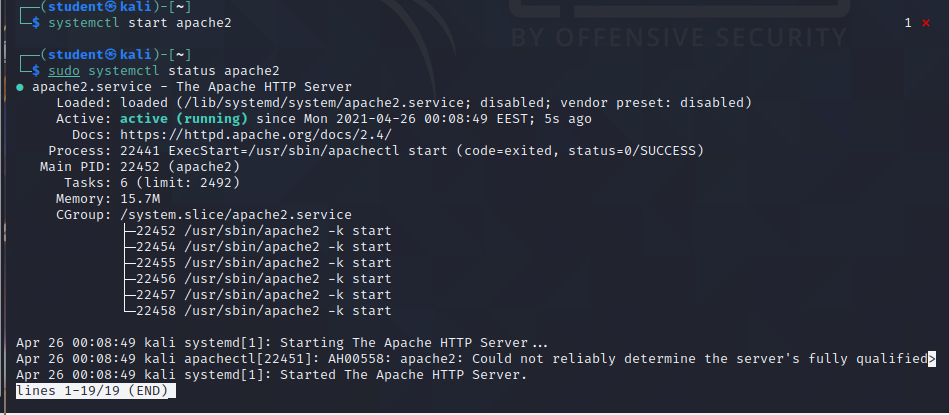


Figure 9. Make the server alive

Now anyone can access to my server:

Graphical user interface, text, email

Description automatically generated

Figure 10. Beautiful place to play with

It works!

Moreover, after booting up the server, I can scan more information about the machine:

Text

Description automatically generated

Figure 11. Scanning my virtual machine again

Nice scan. Server type is Linux 4.15 – 5.6 (Debian?), Apache 2.4.46

I added some files into “/var/www/html” directory:

Graphical user interface, text

Description automatically generated

Figure . Move some dummy files to my web server home page

# **Step 3: Information gathering exercise**

This step I need to scan a test machine on the Internet to gather as much information as possible.

The address for the machine is **cybertesting.duckdns.org**

First, ping and traceroute:

Text

Description automatically generated with low confidence

Figure . Ping and Traceroute

Seems that this domain is in Finland (or proxy?)

After using some whois service, I may assume that this is actually in Finland. I continue using nmap to scan more:

Device types: general purpose | storage-misc | firewall | webcam

OS: Linux

Running: Linux 4.X|5.X|2.6.X|3.X (95%), Synology DiskStation Manager 5.X (90%), WatchGuard Fireware 11.X (89%), Tandberg embedded (89%) (these are guessing)

Services: netbios-ssn Samba smbd, ssh OpenSSH 8.4 (protocol 2.0), http aiohttp 3.6.2

Open ports: 139, 222, 445, 5050 (all TCP)

I tried WPScan but it says the server is down ☹

Graphical user interface, text

Description automatically generated

Figure . Cannot scan with WPScan