DIFFERENT TOOLS AND METHODS FOR NETWORK SCANNING

Network scanning is a fundamental component of cybersecurity, allowing organizations to identify vulnerabilities, assess security postures, and protect against potential threats. A plethora of scanning tools are available, each designed to address specific needs and scenarios. In this comprehensive guide, we will explore various network scanning tools, categorizing them based on their functionalities and use cases.

1. Open Source Network Scanners:

a. Nmap (Network Mapper):

Technical Explanation: Nmap is a versatile open-source tool for network exploration and security auditing. It utilizes raw IP packets to determine available hosts, services, operating systems, and packet filters/firewalls.

Download Link: Nmap Download

Example Usage: `nmap -sP 192.168.1.0/24`

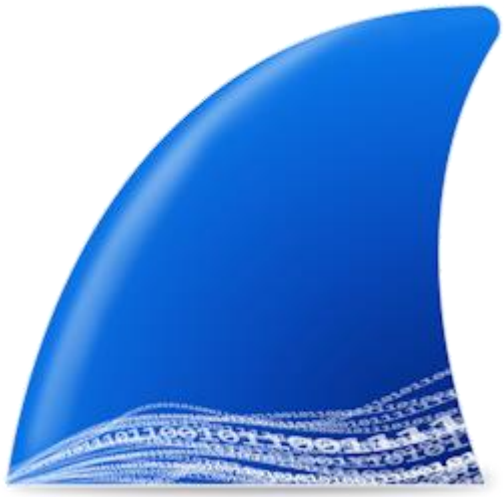
Advantages:

- Comprehensive protocol support
- Scriptable for advanced functionality
- Cross-platform (Linux, Windows, macOS)

Disadvantages:

- Requires understanding of networking concepts
- Some features may be complex for beginners

b. Wireshark:



Technical Explanation: Wireshark is a network protocol analyser that captures and analyses data on a computer network. It provides a human-readable display of the packet data and facilitates in-depth inspection and analysis.

Download Link: Wireshark Download

Example Usage: Capture and analyse network traffic in real-time.

Advantages:

- Rich protocol analysis capabilities
- Cross-platform and user-friendly GUI
- Extensive community support

Disadvantages:

- May capture more data than needed
- Steeper learning curve for in-depth analysis

c. Zenmap:

Technical Explanation: Zenmap is the graphical user interface for Nmap, providing a user-friendly way to visualize and interpret scan results. It simplifies Nmap usage while retaining advanced functionalities.

Download Link: Zenmap Download

Example Usage: GUI frontend for Nmap. Select a profile and target, then click “Scan.”

Advantages:

- Simplifies Nmap usage with a graphical interface
- Topology mapping for visual representation

Disadvantages:

- Limited compared to Nmap for advanced scripting

2. Port Scanners:

a. Masscan:

Technical Explanation: Masscan is designed for high-speed network scanning and can scan the entire IPv4 address space in minutes. It excels in asynchronous transmission and banner grabbing.

Download Link: Masscan on GitHub

Example Usage: `masscan 192.168.1.0/24 -p80,443`

Advantages:

- High-speed scanning
- Banner grabbing for identified services

Disadvantages:

- Limited scripting capabilities compared to Nmap
- Less user-friendly for beginners

b. Unicornscan:

Technical Explanation: Unicornscan is a lightweight and efficient asynchronous network reconnaissance tool. It emphasizes accuracy and resource efficiency.

Download Link: Unicornscan on GitHub

Example Usage: `unicornscan -IvR 192.168.1.1:1-1000`

Advantages:

- Lightweight and fast
- Module-based architecture for extensibility

Disadvantages:

- Limited community support
- Less feature-rich compared to some tools

3. Vulnerability Scanners:

a. Nessus:

Technical Explanation: Nessus is a widely-used vulnerability scanner that identifies vulnerabilities, misconfigurations, and security issues. It employs a plugin-based architecture and provides a vast database of security checks.

Download Link: Nessus Download

Example Usage: Create a scan policy, define targets, and launch the scan.

Advantages:

- Extensive vulnerability database
- Compliance checks for various standards

Disadvantages:

- Limited features in the free version
- Resource-intensive on larger networks

b. OpenVAS:

Technical Explanation: The Open Vulnerability Assessment System (OpenVAS) is an open-source vulnerability scanner. It focuses on ease of use, utilizing a plugin-based approach for conducting vulnerability assessments.

Download Link: OpenVAS Download

Example Usage: Use the web interface to configure and launch vulnerability scans.

Advantages:

- Open-source and free
- Web-based interface for ease of use

Disadvantages:

- Less polished interface compared to commercial alternatives
- Some features may require manual setup

4. Web Application Scanners:

a. OWASP ZAP (Zed Attack Proxy):

Technical Explanation: ZAP is a security testing tool focused on finding vulnerabilities in web applications during development and testing. It provides automated scanners, API support, and fuzz testing capabilities.

Download Link: ZAP Download

Example Usage: Configure target URL, start an automated scan, and review results.

Advantages:

- Active and passive scanning
- Fuzz testing for web applications

Disadvantages:

- Steeper learning curve for beginners
- Advanced features may be overwhelming

b. Nikto:

Technical Explanation: Nikto is a web server scanner that performs comprehensive tests against web servers. It checks for outdated software, misconfigurations, and potential security issues.

Download Link: Nikto on GitHub

Example Usage: `nikto -h example.com`

Advantages:

- Comprehensive web server scanning
- SSL/TLS scanning and support for multiple plugins

Disadvantages:

- Limited to web server scanning
- May produce false positives

5. Wireless Scanners:

a. Aircrack-ng:

Technical Explanation: Aircrack-ng is a suite of tools for assessing Wi-Fi network security. It includes packet capture, WEP and WPA/WPA2 key cracking, and various utilities for testing and analyzing wireless networks.

Download Link: Aircrack-ng Download

Example Usage: Various commands for capturing and cracking Wi-Fi passwords.

Advantages:

- Powerful Wi-Fi analysis tools
- Support for WEP and WPA/WPA2 cracking

Disadvantages:

- Requires a good understanding of Wi-Fi protocols
- Legal considerations and ethical use are crucial

b. Kismet:

Technical Explanation: Kismet is a wireless network detector, sniffer, and intrusion detection system. It passively captures and analyses wireless network traffic, providing information about detected networks, clients, and devices.

Download Link: Kismet Download

Example Usage: `kismet -c wlan0`

Advantages:

- Passive scanning without generating traffic
- Supports multiple capture sources

Disadvantages:

- Command-line interface may be less user-friendly
- Limited to Wi-Fi network analysis

6. Ping Sweep and ICMP Scanners:

a. Fping:

Technical Explanation: Fping is a fast and efficient command-line tool for pinging multiple hosts using ICMP Echo Requests. It supports parallel scanning, making it ideal for quickly probing large networks.

Download Link: Fping Download

Example Usage: `fping -a -g 192.168.1.0/24`

Advantages:

- Fast ICMP probing with parallel scanning
- Lightweight and efficient

Disadvantages:

- Limited to ICMP-based scanning
- No advanced features for in-depth analysis

b. Hping:

Technical Explanation: Hping is a command-line packet assembler and analyzer that can be used for host discovery, firewall testing, and advanced TCP/IP packet manipulation. It allows users to construct custom packets and send them to a target host.

Download Link: Hping Download

Example Usage: `hping3 -S -p 80 -c 5 192.168.1.1`

Advantages:

- Custom packet crafting and scripting
- Traceroute capabilities

Disadvantages:

- Command-line interface may be intimidating
- Requires understanding of TCP/IP protocols

7. Internet-Wide Scanning Services:

a. Shodan:

Technical Explanation: Shodan is a search engine designed to find specific types of devices connected to the internet. It looks into banners returned by devices, providing information about open ports, services, and potential vulnerabilities.

Access Link: Shodan

Example Usage: Search for devices or services, e.g., `apache country:US`

Advantages:

- Comprehensive search engine for internet-connected devices
- Banner grabbing and vulnerability search

Disadvantages:

- Free access is limited
- Some features are subscription-based

b. Censys:

Technical Explanation: Censys is a search engine for the internet's address space, offering information about hosts, websites, and certificates. It enables users to explore and analyse data related to domains, protocols, and digital certificates.

Access Link: Censys

Example Usage: Use the search bar to find information about IP addresses, websites, or certificates.

Advantages:

- Certificate transparency search
- Historical data analysis

Disadvantages:

- Free access is limited
- Some features are subscription-based

8. Penetration Testing Frameworks:

a. Metasploit Framework:

Technical Explanation: Metasploit is a powerful penetration testing framework that provides information about security vulnerabilities. It

includes tools for exploiting known vulnerabilities, creating custom exploits, and managing sessions on compromised systems.

Download Link: Metasploit Framework

Example Usage: Exploit a known vulnerability, e.g., `use exploit/windows/smb/ms08_067_netapi`

Advantages:

- Exploit development and penetration testing
- Large and active community

Disadvantages:

- Requires understanding of ethical hacking principles
- May be overwhelming for beginners

9. Packet Crafting and Manipulation:

a. Scapy:

Technical Explanation: Scapy is a powerful interactive packet manipulation program and library for Python. It allows users to craft custom packets, send them on the wire, capture them, and match requests and replies.

Download Link: Scapy Download

Example Usage: Craft custom packets, e.g., `send(IP(dst="example.com")/ICMP())`

Advantages:

- Python-based interactive packet manipulation
- Crafting custom packets for network discovery

Disadvantages:

- Command-line interface may be challenging for beginners

- Requires scripting skills for advanced use

In conclusion, the world of network scanning tools is vast and diverse, offering solutions for various cybersecurity needs. When utilizing these tools, it's crucial to act responsibly, ensuring that proper authorization is obtained before scanning networks. The information obtained through scanning should be used for the purpose of securing and improving the resilience of systems and networks.

SCRIPT SCANNING TO IDENTIFY VULNERABILITIES

Nmap vulnerability scan using NSE scripts

CVE stands for Common Vulnerabilities and Exposures. In plain English, that simply means it's a way to organize and categorize software vulnerabilities. This information can be highly useful for security researchers and penetration testers in their daily tasks.

Something we really love about the tool is its ability to expand its core features by using Nmap scripts. You might be wondering can Nmap find vulnerabilities. Yes it can! By combining these Nmap commands with a few NSE scripts, we're able to fetch the most popular CVEs from any target.

Two of the most popular vulnerability/CVE detection scripts found on Nmap NSE are nmap-vulners and vulscan, which will enable you to detect relevant CVE information from remote or local hosts.

Along with those two, the entire "vuln" category is an absolute treasure trove — a truly useful resource when using Nmap as a vulnerability scan engine.

Nmap vulscan

Vulscan queries its own local CVE databases, hosted on the client performing the scan. These local databases include the following files: scipvuldb.csv, cve.csv, securityfocus.csv, xforce.csv, expliotdb.csv, openvas.csv, securitytracker.csv, osvdb.csv.

In order to use this NSE script, we'll need to clone its GitHub repo.

Nmap-vulners

Nmap-vulners is one of the most famous vulnerability scanners in use. Let's explore how to install this tool, as well as how to perform a simple CVE scan.

The syntax we'll be using is pretty simple, calling the script by using --script and specifying the vulners engine, as shown here:

```
nmap --script nmap-vulners/ -sV 11.22.33.44
```

If you want to target specific posts, you simply need to add -p80 at the end, and replace "80" with the port you want to scan. And of course, replace 11.22.33.44 with your desired IP.

Nmap-vulners queries the Vulners exploit database every time we use the NSE script.

Nmap vuln

The way **NSE** scripts are defined is based on a list of predefined categories where each script belongs. These categories include: auth, broadcast, brute, default, discovery, dos, exploit, external, fuzzer, intrusive, malware, safe, version, and vuln.

Nmap script vuln is the one we'll be using to launch our next scan against vulnerable subdomains. The syntax is the same as that of the previous NSE scripts, with 'vuln' added after '--script', as you can see here:

```
nmap -Pn --script vuln 192.168.1.105
```

Running specific vulnerability scans with Nmap

The "nmap vuln" scan discussed above uses an entire category of scripts to test a vulnerable target against. In the case of "vuln", there are 50+ scripts in this category, as shown here: nmap vuln.

It is also possible to run scans using specific scripts within each category. Below are some examples of this.

Finding Vulnerable PHP versions

Now we look at 2 PHP scripts. The first checks for the version of PHP running(which only responds with versions up to 5.5.0) and the second looks to see if the PHP-CGI installation is vulnerable to CVE-2012-1823.

Run the first command like so:

```
nmap -sV --script=http-php-version testphp.vulnweb.com
```

Summary

Nmap's powerful scripts allow you to not only perform port scanning tasks, but also to discover CVEs in a matter of seconds. Thanks to Nmap, this becomes an easy task, even if you don't have advanced technical skills.

TOOLS AND TECHNIQUES TO ANALYSE NETWORK TRAFFIC

Packet Sniffing: In this approach, the data packets are captured and analysed while they are moving through the network. Adding packet sniffers to the specialized software allows to intercept and inspect packets, providing information about the network activity.

NetFlow Analysis: NetFlow is a networking protocol that collects data about IP packets as they flow through a networking device. The NetFlow analysis is responsible for collecting and analysing such data to get visibility into network traffic patterns, top talkers and bandwidth usage.

Firewall Logs: Firewalls perform as a barrier, by checking the network traffic in and out using the specified security policies. Examining firewall logs can give data concerning permitted and denied network connections, as well as possible security threats.

Intrusion Detection Systems (IDS): IDS stands for Intrusion Detection System which is a tool that is used to monitor the network for suspicious activities and then alert the users. IDS can be used to analyse network traffic behaviour and patterns. This way, it can detect unauthorized access attempts and potential security breaches.

Bandwidth Monitoring Tools: These instruments monitor how much data is transmitted over the network over a given period. Through tracking of bandwidth usage administrators can pinpoint bandwidth-consuming applications, improve performance of the network and detect suspicious activity.

Port Mirroring: Port mirroring, in other words, SPAN or port monitoring, is the process of duplicating data from one or more ports on a network switch to a designated port. This copy data will then be monitored using network monitoring tools.

Deep Packet Inspection (DPI): DPI is a technology that is used to check the contents of data packets when they are in motion through a network. It makes it possible to monitor the network traffic in detail, including the identification of applications, protocols and malware.

Flow-based Monitoring: Flow-oriented monitoring techniques, such as sFlow and J-Flow, collect and interpret the flow data records created by network devices. These logs store information about specific data channels, including source and destination IP addresses, ports, and protocols.

Application Performance Monitoring (APM): APM tools track the performance and uptime of apps that are networked. APM tools can detect application performance issues by analysing network traffic associated with specific applications. They can also be used to troubleshoot problems and enhance application performance.

Cloud-based Monitoring: Increasingly, businesses are adopting cloud services and virtual environments and thus cloud-based monitoring solutions provide visibility into the network traffic within cloud systems. They offer data on traffic patterns, performance metrics, and security issues in a cloud app or service.

User Activity Monitoring: The user activity monitoring tools track the network activity of individual users, including web browsing, file transfers, and application use. Through tracking user behaviour, companies can police security rules, detect insider threats and make sure they abide by the regulations.

Wireshark

First on our list is the most well-known tool—Wireshark. It's open-source and very advanced. Wireshark can be used for any type of traffic and any interface. It's a real powerhouse when it comes to traffic analysis. It can show all the traffic in real time, but you can also apply filters if you know what you are looking for. It also offers offline analysis from previously saved files and comes with a few options for generating interesting statistics about the traffic. Wireshark can run on Windows, Linux, and macOS systems. Disadvantages? It's quite complicated, so you need to spend some time learning it. It's also a great tool for ad-hoc analysis, but it's not a tool that you would use for company-wide implementation.

Ettercap

The next tool on our list, Ettercap, shares a few similarities with Wireshark. It's also open-source and can run on Windows, Linux, and macOS. However, while Wireshark works passively, Ettercap can not only analyze traffic but also manipulate it. Overall, Ettercap is more advanced than Wireshark. It's often used as a testing tool, but it has advanced network traffic capabilities too. It can identify malicious users and record their actions or block them from the network. Disadvantages? Similar to Wireshark, it requires quite some learning to understand all of its capabilities. Also, it's a command-line-only tool.

Kismet

Yet another popular traffic analysis tool is Kismet. For this one, let's start with its disadvantages. The biggest one is the fact that it can only work with WiFi networks, so you won't be able to use it on wired networks. This is quite a limitation, but if WiFi traffic analysis is what you need the most, that could be a good option for you. Bu default, Kismet only looks for packets of metadata, which is not a disadvantage on its own but something you should be aware of. It can, however, be switched to a full-packet data capture.

SolarWinds NetFlow Traffic Analyzer

One commonly used non-open-source NTA tool is the SolarWinds NetFlow Traffic Analyzer. It comes with many features, like identifying which endpoints generate heavy traffic on the network, generating reports, and alerting. Unlike the previous tools listed, the NetFlow Traffic Analyzer comes with a web interface so it can be installed once and used by the whole team.

Netreo Traffic Monitor

Last but definitely not least is Netreo Traffic Monitor, one of the most complete NTA solutions. It can gather data from many different devices, as it supports Netflow, sFlow, and IPFIX protocols. It can handle over a million connections per second, It's a real powerhouse when it comes to traffic analysis. Yet, at the same time, it's really easy to install and get started with. Configuration and maintenance are minimal.

For that reason, Netreo's Traffic Monitor is a really good option for pretty much any size company. It can automatically detect and instantly visualize new devices found on the network. So, unlike with other tools, you don't need to spend days or weeks crafting special filters and configuring alerts to get the visibility you need. Moreover, Netreo's NTA is suitable for modern environments. It gives you a unified analysis of all your networks, no matter if they are on-prem or in the cloud—something that's really important these days.

NMAP ADVANCED SCAN FOR TESTFIRE.NET

1- Simple Nmap scan

```
└─$ sudo su
[sudo] password for kali:
┌──(root💀kali)-[/home/kali]
└─# nmap www.testfire.net
Starting Nmap 7.91 ( https://nmap.org ) at 2024-07-09 01:04 EDT
Nmap scan report for www.testfire.net (65.61.137.117)
Host is up (0.023s latency).
Not shown: 996 filtered ports
PORT     STATE  SERVICE
80/tcp   open   http
443/tcp  open   https
8080/tcp open   http-proxy
8443/tcp closed https-alt

Nmap done: 1 IP address (1 host up) scanned in 20.48 seconds
```

2- Nmap SYN scan

```
└─# nmap -sS 65.61.137.117
Starting Nmap 7.91 ( https://nmap.org ) at 2024-07-09 01:30 EDT
Nmap scan report for 65.61.137.117
Host is up (0.033s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 13.02 seconds
```

3- Nmap UDP scan

```
└─# nmap -sU 65.61.137.117
Starting Nmap 7.91 ( https://nmap.org ) at 2024-07-09 01:26 EDT
Nmap scan report for 65.61.137.117
Host is up (0.0011s latency).
All 1000 scanned ports on 65.61.137.117 are open|filtered

Nmap done: 1 IP address (1 host up) scanned in 4.42 seconds
```

4- Nmap TCP scan

```
┌──(root💀kali)-[/home/kali]
└─# nmap -sT 65.61.137.117
Starting Nmap 7.91 ( https://nmap.org ) at 2024-07-09 01:06 EDT
Nmap scan report for 65.61.137.117
Host is up (0.00082s latency).
All 1000 scanned ports on 65.61.137.117 are filtered

Nmap done: 1 IP address (1 host up) scanned in 21.25 seconds
```

5- Nmap version discovery scan

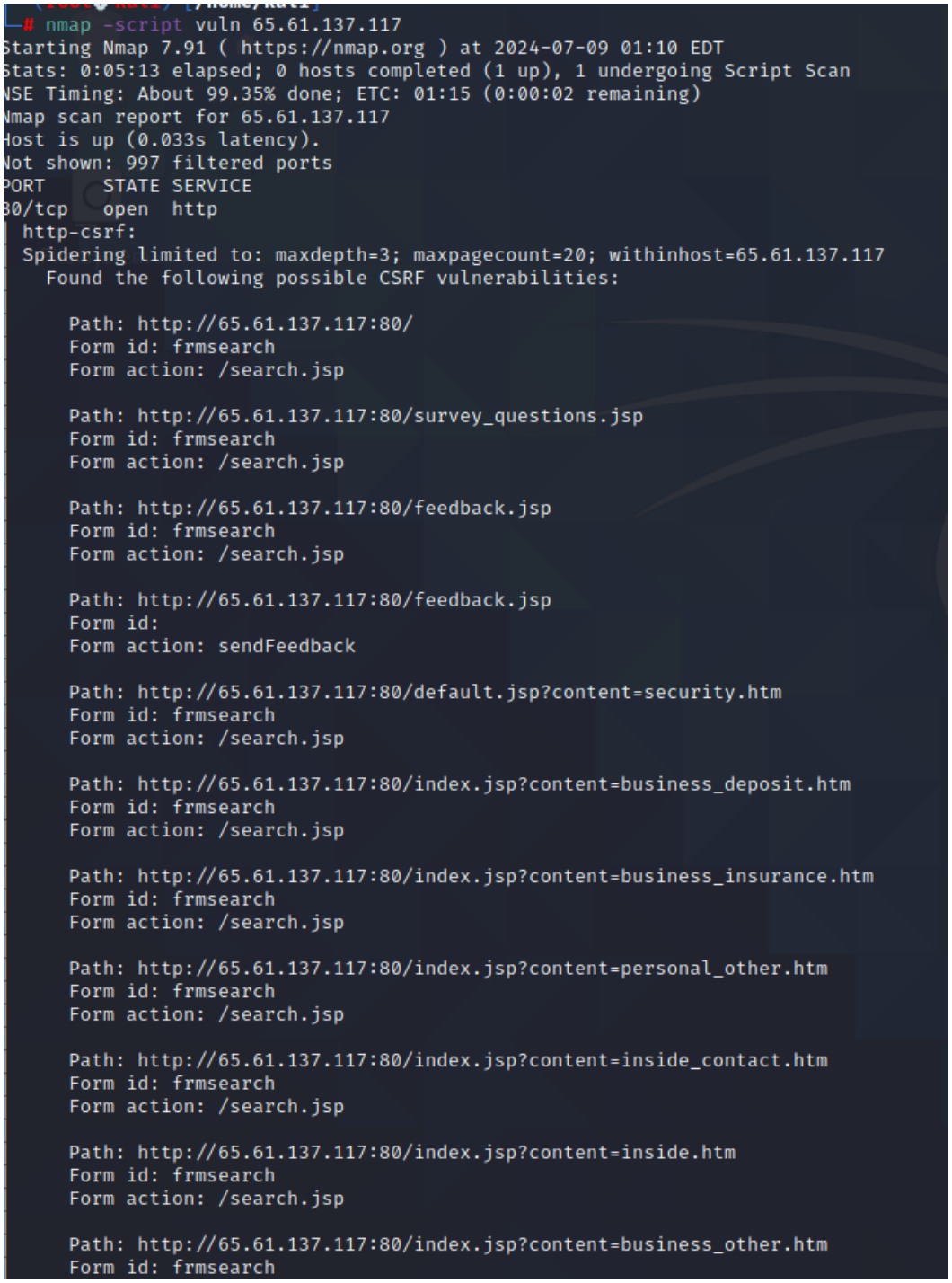
```
┌──(root💀kali)-[/home/kali]
└─# nmap -sV 65.61.137.117
Starting Nmap 7.91 ( https://nmap.org ) at 2024-07-09 01:05 EDT
Nmap scan report for 65.61.137.117
Host is up (0.035s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE    VERSION
80/tcp   open  http       Apache Tomcat/Coyote JSP engine 1.1
443/tcp  open  ssl/https?
8080/tcp open  http       Apache Tomcat/Coyote JSP engine 1.1

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.59 seconds
```

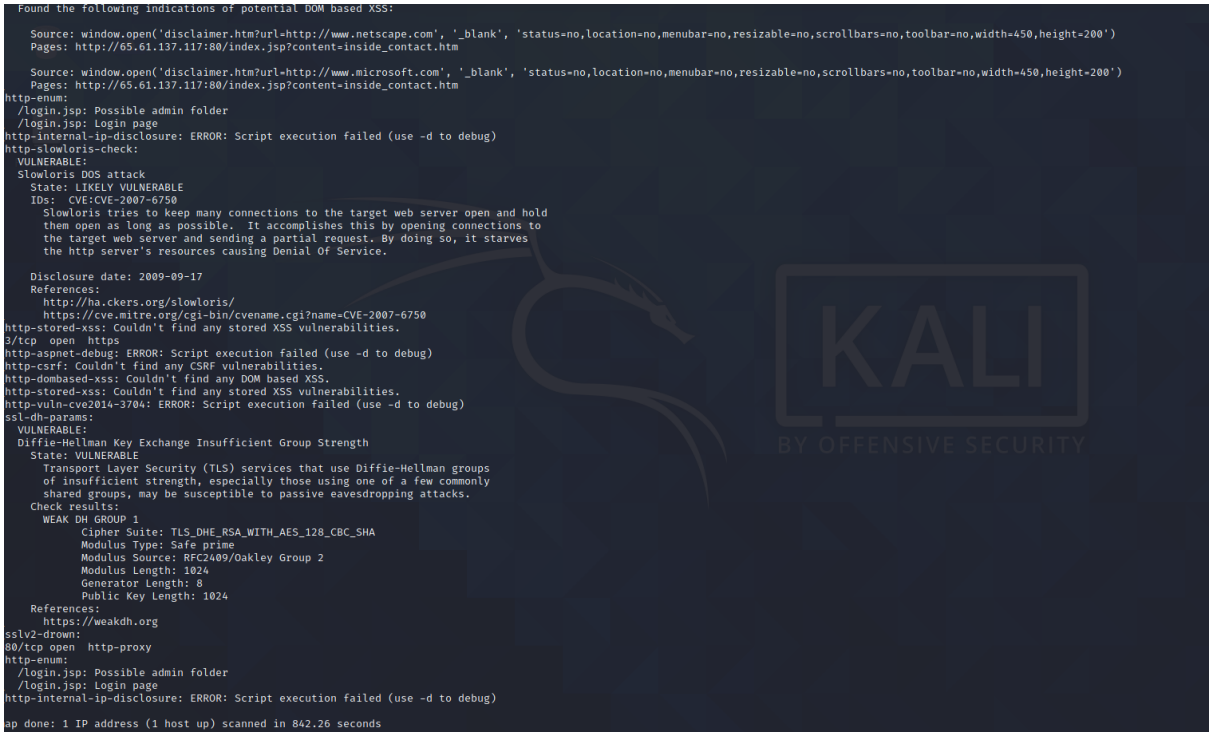
6- Nmap OS discovery scan

7- Nmap AGGRESSIVE scan

```
┌──(root💀kali)-[/home/kali]
└─# nmap -A 65.61.137.117
Starting Nmap 7.91 ( https://nmap.org ) at 2024-07-09 01:08 EDT
Stats: 0:00:00 elapsed; 0 hosts completed (0 up), 0 undergoing Script Pre-Scan
NSE Timing: About 0.00% done
Nmap scan report for 65.61.137.117
Host is up (0.055s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE    VERSION
80/tcp   open  http       Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
|_http-title: Altoro Mutual
443/tcp  open  ssl/https?
| ssl-cert: Subject: commonName=demo.testfire.net
| Subject Alternative Name: DNS:demo.testfire.net, DNS:altoromutual.com
| Not valid before: 2023-06-19T00:00:00
|_Not valid after:  2024-06-14T23:59:59
|_ssl-date: 2024-07-09T05:09:27+00:00; +2s from scanner time.
8080/tcp open  http       Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
|_http-title: Altoro Mutual
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge|general purpose
Running (JUST GUESSING): Oracle Virtualbox (98%), QEMU (92%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu
Aggressive OS guesses: Oracle Virtualbox (98%), QEMU user mode network gateway (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops

Host script results:
|_clock-skew: 1s

TRACEROUTE (using port 80/tcp)
HOP RTT     ADDRESS
1   1.27 ms 10.0.2.2
2   1.27 ms 65.61.137.117

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 44.91 seconds
```

8- Nmap vulnerability scan


└─# nmap -script vuln 65.61.137.117
Starting Nmap 7.91 (https://nmap.org) at 2024-07-09 01:10 EDT
Stats: 0:05:13 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.35% done; ETC: 01:15 (0:00:02 remaining)
Nmap scan report for 65.61.137.117
Host is up (0.033s latency).
Not shown: 997 filtered ports
PORT STATE SERVICE
80/tcp open http
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=65.61.137.117
| Found the following possible CSRF vulnerabilities:
|
| Path: http://65.61.137.117:80/
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/survey_questions.jsp
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/feedback.jsp
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/feedback.jsp
| Form id:
| Form action: sendFeedback
|
| Path: http://65.61.137.117:80/default.jsp?content=security.htm
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/index.jsp?content=business_deposit.htm
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/index.jsp?content=business_insurance.htm
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/index.jsp?content=personal_other.htm
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/index.jsp?content=inside_contact.htm
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/index.jsp?content=inside.htm
| Form id: frmsearch
| Form action: /search.jsp
|
| Path: http://65.61.137.117:80/index.jsp?content=business_other.htm
| Form id: frmsearch



REFERENCES

1-https://securitytrails.com/blog/nmap-vulnerability-scan

2-https://www.netreo.com/blog/network-traffic-analysis-tools/

3-https://medium.com/@ajithchandranr/comprehensive-list-of-network-scanning-tools-d03da398c1d4