Experiment No. 3

Title: Experimenting with Kali

Batch: A3 **Roll No: 16010421119** **Experiment no.: 3**

Aim: Experimenting with Kali.

Resources needed: Pentesting set up

Theory:

Kali Linux, a robust and specialized Linux distribution, stands as a beacon in cybersecurity, particularly for Vulnerability Assessment and Penetration Testing (VAPT). This purpose-built platform is meticulously crafted to equip security professionals and ethical hackers with a comprehensive suite of tools, allowing them to simulate real-world cyber threats in a controlled and ethical manner.

Kali Linux, derived from Debian, is tailored for VAPT, a proactive approach to securing information systems. The distribution integrates many pre-installed security tools covering every facet of the testing process. This includes reconnaissance, vulnerability identification, exploitation, post-exploitation analysis, and reporting. The ecosystem enables security experts tocomprehensively assess the resilience of networks, applications, and systems against potential threats.

Core Tools and Capabilities:

1. Nmap - Unveiling Network Landscapes:

Nmap, the cornerstone of network exploration, is instrumental in mapping out hosts, identifying open ports, and scrutinizing services. Its flexibility allows practitioners to conduct scans such as SYN scans for stealth, UDP scans for unconventional protocols, and version detection for granular insights into target systems.

2. OpenVAS - Unearthing Vulnerabilities:

OpenVAS, integrated into Kali Linux, transforms the vulnerability assessment landscape. By employing a database of known vulnerabilities, it systematically scans target systems, providing a detailed report on potential weaknesses. Security professionals can leverage this information to address and mitigate risks proactively.

3. Metasploit - The Art of Exploitation:

Metasploit, a potent penetration testing framework, enables security practitioners to simulate cyber-attacks. Its vast collection of exploits and payloads caters to a diverse range of targets. With Metasploit, ethical hackers can validate the effectiveness of security measures and develop strategies to fortify defenses.

4. Wireshark - Decrypting Network Traffic:

Wireshark, a network protocol analyzer, dissects packets traversing the network. It aids in understanding network behavior, identifying anomalies, and uncovering potential security threats. Security professionals can utilize Wireshark to intercept and analyze communication, enhancing their ability to detect and counteract malicious activities.

5. Aircrack-ng - Securing Wireless Networks:
In the realm of wireless security, Aircrack-ng takes center stage. This toolset empowers security experts to audit and secure wireless networks. From capturing Wi-Fi handshakes to exploiting vulnerabilities in wireless protocols, Aircrack-ng is pivotal in fortifying organizations against wireless threats.

Ethical Considerations:
The exploration of Kali Linux for VAPT demands a principled approach. Practitioners must operate within the bounds of legal and ethical frameworks. Gaining proper authorization, respecting privacy, and adhering to responsible disclosure practices are paramount. The objective is not to exploit for malicious intent but to fortify defenses and cultivate a proactive security posture.

Procedure:
Exploring network landscapes using Nmap involves a stepwise discovery, scanning, and analysis process.

Step 1: Install Nmap on Kali Linux
Ensure that Nmap is installed on the Kali Linux system. If not, install it using the following command:

```
sudo apt-get update
sudo apt-get install nmap
```

Step 2: Identify Target
Determine the target network or IP address range to scan. This could be a specific IP address, a range of IP addresses, or an entire subnet.

Step 3: Basic Ping Scan
Perform a basic ping scan to identify live hosts on the network. This helps in narrowing down the scope of the scan.

```
nmap -sn <target>
```

Replace <target> with the IP address or range to scan. This command sends ICMP echo requests to discover live hosts without performing detailed port scans.

Step 4: Port Scan for Common Ports
Conduct a port scan to identify open ports on live hosts. This command scans the 1,000 most common ports.

```
nmap -p 1-1000 <target>
```

Step 5: Intense Scan with Service Version Detection
Perform a more comprehensive scan, including service version detection. This provides details about the services running on open ports.

```
nmap -sV <target>
```

Step 6: Aggressive Scan with OS Detection
Execute an aggressive scan that includes operating system detection. This attempts to identify the operating system of the target hosts.

```
nmap -A <target>
```

Step 7: Output to a File

Save the results to a file for later analysis or reporting. Replace <output_file> with the desired file name.

```
nmap -A -oN <output_file> <target>
```

Step 8: Perform a Script Scan

Nmap has a variety of scripts that can provide additional information about the target. Use the following command to default scripts against the target.

```
nmap -sC <target>
```

Step 9: Explore UDP Ports

Include UDP port scanning to identify services running on UDP ports.

```
nmap -sU <target>
```

OpenVAS

Exploring vulnerabilities using OpenVAS involves a stepwise installation, configuration, and scanning process.

Step 1: Install OpenVAS on Kali Linux

Ensure that OpenVAS is installed on your Kali Linux system. You can install it using the following commands:

```
sudo apt-get update
sudo apt-get install openvas
```

During the installation, the prompt will be given to set up a password for the OpenVAS Administrator (admin).

Step 2: Configure OpenVAS

After installation, configure OpenVAS by running the following command:

```
sudo openvas-setup
```

Follow the prompts to set up the OpenVAS Manager, Scanner, and other components. This process may take some time as it downloads the necessary vulnerability databases.

Step 3: Start OpenVAS Services

Start the OpenVAS services with the following commands:

```
sudo systemctl start openvas-manager
sudo systemctl start openvas-scanner
sudo systemctl start openvas-gsa
```

Step 4: Access OpenVAS Web Interface

Open a web browser and navigate to the OpenVAS web interface using the following URL:

```
https://localhost:9392
```

Log in with the OpenVAS Administrator credentials set during the setup.

Step 5: Update OpenVAS Feeds

Update the vulnerability feeds to ensure that OpenVAS has the latest information. Go to the "Administration" tab and click on "Feeds." Click on the "Green Arrows" icon to update the feeds.

Step 6: Create a Target

Define a target for scanning. Go to the "Configuration" tab and click on "Targets." Click on the "Create Target" button and provide details such as the target's IP address or hostname.

Step 7: Create a Task

Create a scanning task associated with the target. Go to the "Scans" tab and click on "Tasks." Click the "Create Task" button, select the target, and configure scan parameters.

Step 8: Run the Scan

Initiate the vulnerability scan by selecting the created task and clicking the "Play" button. This will launch the scan against the specified target.

Metasploit

Using Metasploit for penetration testing involves a stepwise installation, exploration, and exploitation process.

Step 1: Install Metasploit on Kali Linux

Ensure that Metasploit is installed on the Kali Linux system. If not, install it using the following commands:

```
sudo apt-get update
sudo apt-get install metasploit-framework
```

Step 2: Start Metasploit Console

Launch the Metasploit console by entering the following command in the terminal:

```
msfconsole
```

This opens the Metasploit Framework console, providing access to various modules and functionalities.

Step 3: Explore Modules

Explore available modules using the search command. For example, to search for exploits related to the Apache web server, type:

```
search apache
```

Review the results and select a module based on target and scenario.

Step 4: Select and Load an Exploit Module

Choose an exploit module from the list and load it into the Metasploit console using the use command. Replace <exploit_module> with the name of the desired module:

```
use <exploit_module>
```

Step 5: Configure the Exploit

Configure the exploit by setting the required parameters. Use the show options command to view and set the necessary options. For example:

```
show options
set RHOSTS <target_IP>
set RPORT <target_port>
```

Step 6: Verify Exploit Configuration

Double-check configuration using the *show options* command to ensure all required parameters are set correctly.

Step 7: Exploit the Target. Execute the exploit by typing:

```
exploit
```

This launches the attack against the target system. Metasploit will attempt to exploit the specified vulnerability.

Step 8: Post-Exploitation

Upon successful exploitation, the post-exploitation phase starts. Use various Metasploit commands and modules to gather information, escalate privileges, and explore the compromised system.

```
sysinfo
getuid
```

Step 9: Explore Post-Exploitation Modules

Use the post command to explore post-exploitation modules. These modules help in privilege escalation, data exfiltration, and lateral movement.

```
use post/multi/recon/local_exploit_suggester
```

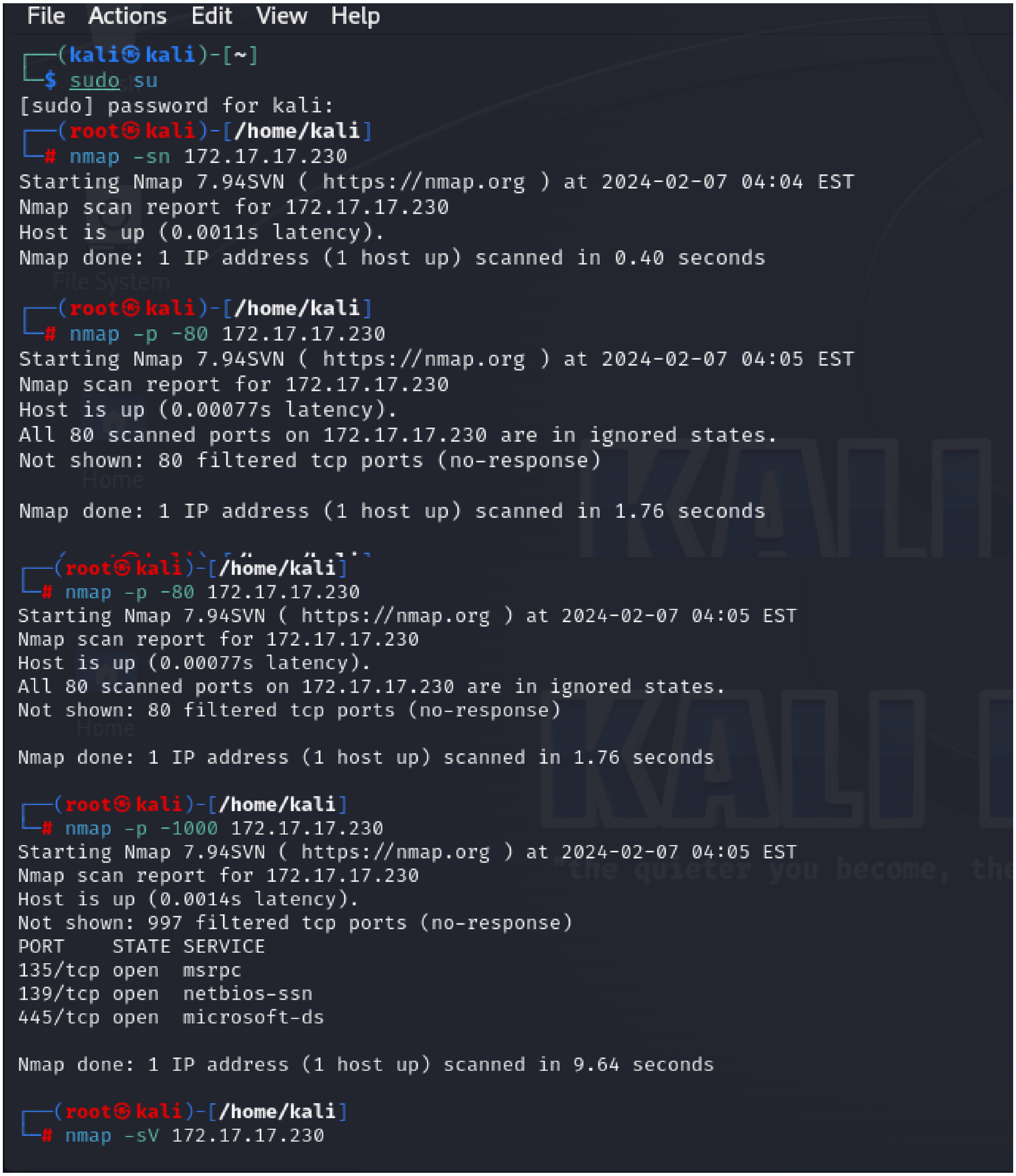
Step 10: Generate Reports

Document findings and generate reports summarizing the penetration test. Use the db_export command to export data to external tools for reporting.

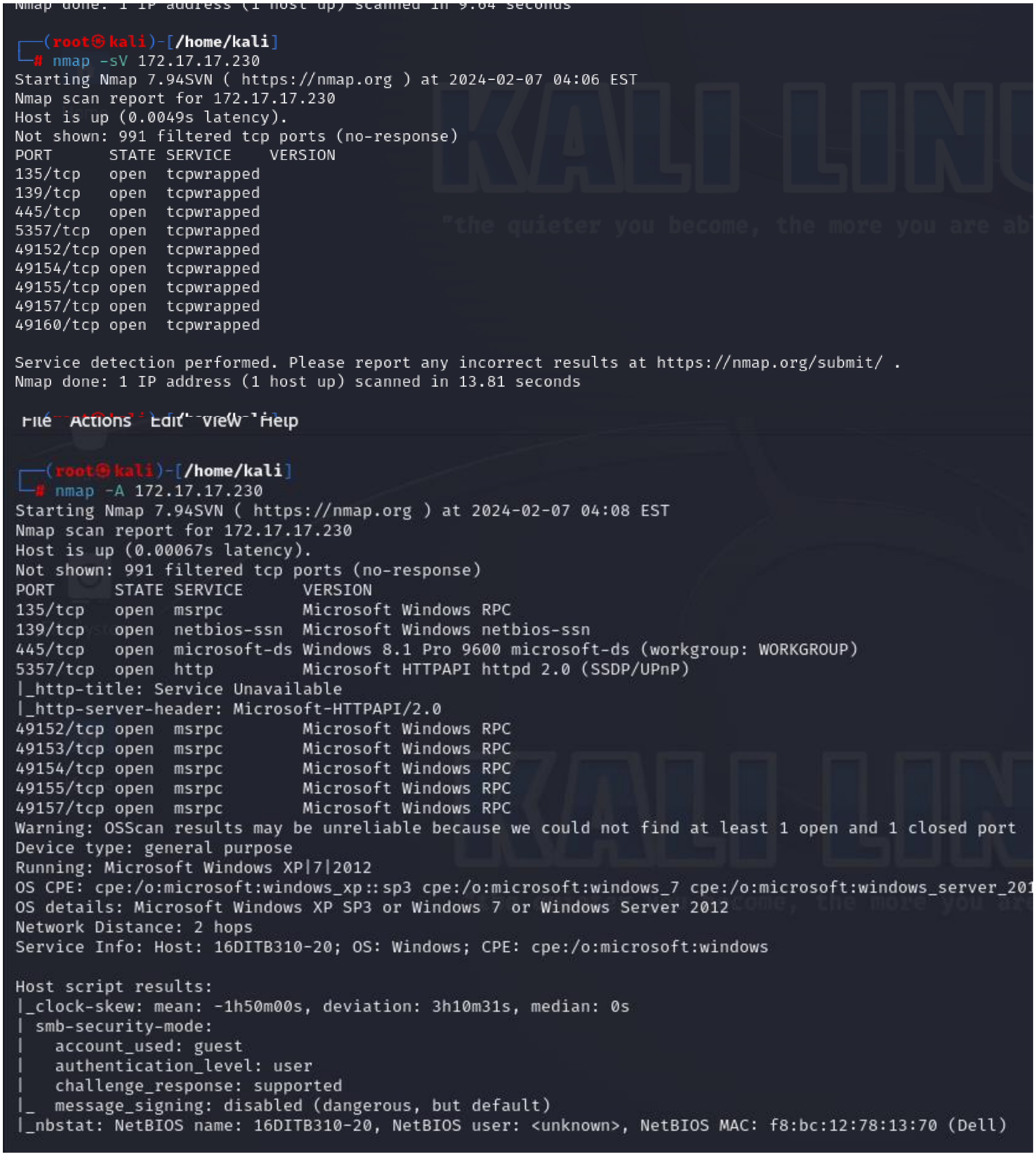
```
db_export -f xml -o /path/to/report.xml
```

Output(Code with result Snapshot)

- **Execute minimum 2 tools**

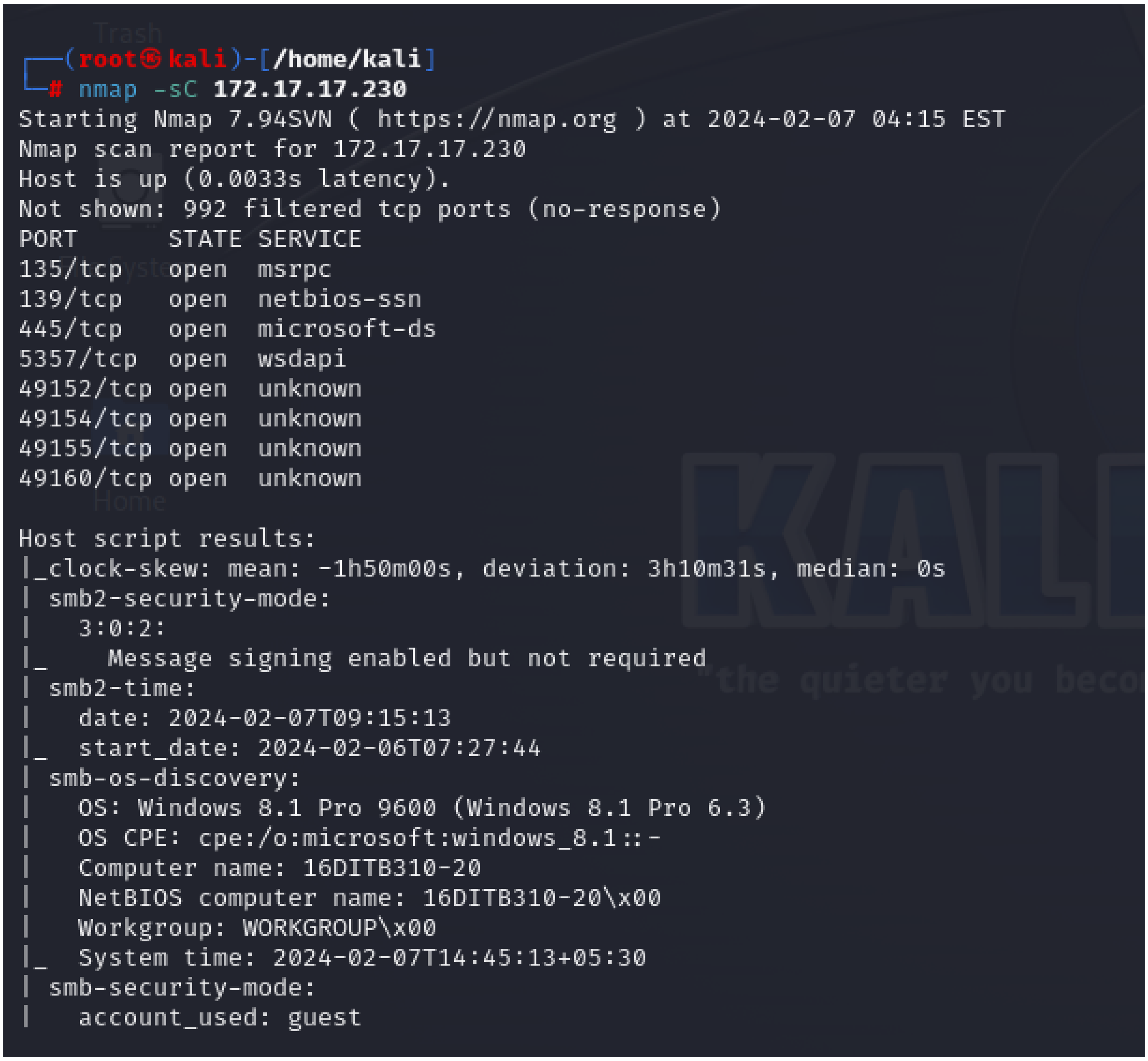


```
┌──(kali㉿kali)-[~]
└─$ sudo su
[sudo] password for kali:
┌──(root㉿kali)-[/home/kali]
└─# nmap -sn 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:04 EST
Nmap scan report for 172.17.17.230
Host is up (0.0011s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds

┌──(root㉿kali)-[/home/kali]
└─# nmap -p -80 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:05 EST
Nmap scan report for 172.17.17.230
Host is up (0.00077s latency).
All 80 scanned ports on 172.17.17.230 are in ignored states.
Not shown: 80 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 1.76 seconds

┌──(root㉿kali)-[/home/kali]
└─# nmap -p -80 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:05 EST
Nmap scan report for 172.17.17.230
Host is up (0.00077s latency).
All 80 scanned ports on 172.17.17.230 are in ignored states.
Not shown: 80 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 1.76 seconds

┌──(root㉿kali)-[/home/kali]
└─# nmap -p -1000 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:05 EST
Nmap scan report for 172.17.17.230
Host is up (0.0014s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 9.64 seconds

┌──(root㉿kali)-[/home/kali]
└─# nmap -sV 172.17.17.230
```



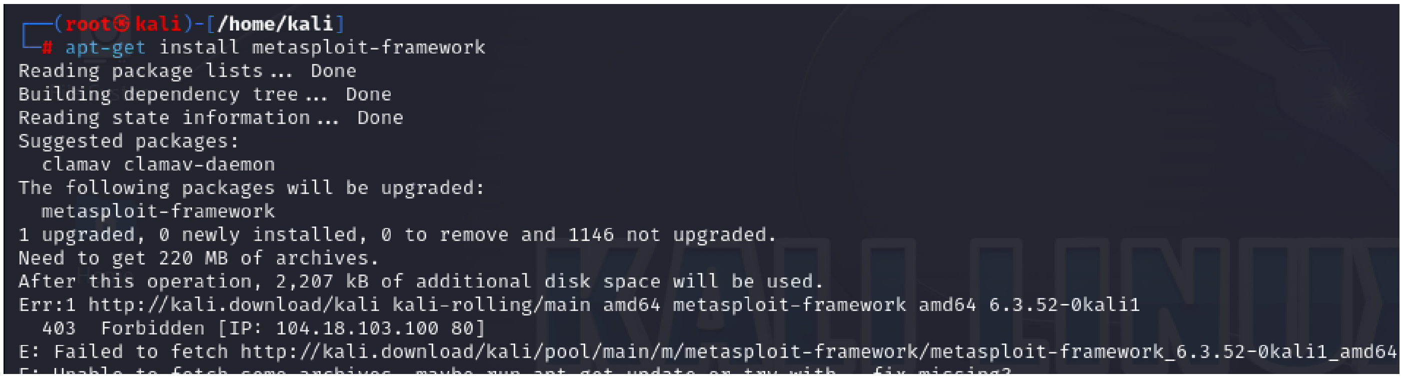
```
Nmap done: 1 IP address (1 host up) scanned in 9.64 seconds

┌──(root㉿kali)-[/home/kali]
└─# nmap -sV 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:06 EST
Nmap scan report for 172.17.17.230
Host is up (0.0049s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT      STATE SERVICE    VERSION
135/tcp   open  tcpwrapped
139/tcp   open  tcpwrapped
445/tcp   open  tcpwrapped
5357/tcp  open  tcpwrapped
49152/tcp open  tcpwrapped
49154/tcp open  tcpwrapped
49155/tcp open  tcpwrapped
49157/tcp open  tcpwrapped
49160/tcp open  tcpwrapped

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.81 seconds
```

```
┌──(root㉿kali)-[/home/kali]
└─# nmap -A 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:08 EST
Nmap scan report for 172.17.17.230
Host is up (0.00067s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Windows 8.1 Pro 9600 microsoft-ds (workgroup: WORKGROUP)
5357/tcp  open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Service Unavailable
|_http-server-header: Microsoft-HTTPAPI/2.0
49152/tcp open  msrpc        Microsoft Windows RPC
49153/tcp open  msrpc        Microsoft Windows RPC
49154/tcp open  msrpc        Microsoft Windows RPC
49155/tcp open  msrpc        Microsoft Windows RPC
49157/tcp open  msrpc        Microsoft Windows RPC
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Microsoft Windows XP|7|2012
OS CPE: cpe:/o:microsoft:windows_xp::sp3 cpe:/o:microsoft:windows_7 cpe:/o:microsoft:windows_server_201
OS details: Microsoft Windows XP SP3 or Windows 7 or Windows Server 2012
Network Distance: 2 hops
Service Info: Host: 16DITB310-20; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: -1h50m00s, deviation: 3h10m31s, median: 0s
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_nbstat: NetBIOS name: 16DITB310-20, NetBIOS user: <unknown>, NetBIOS MAC: f8:bc:12:78:13:70 (Dell)
```

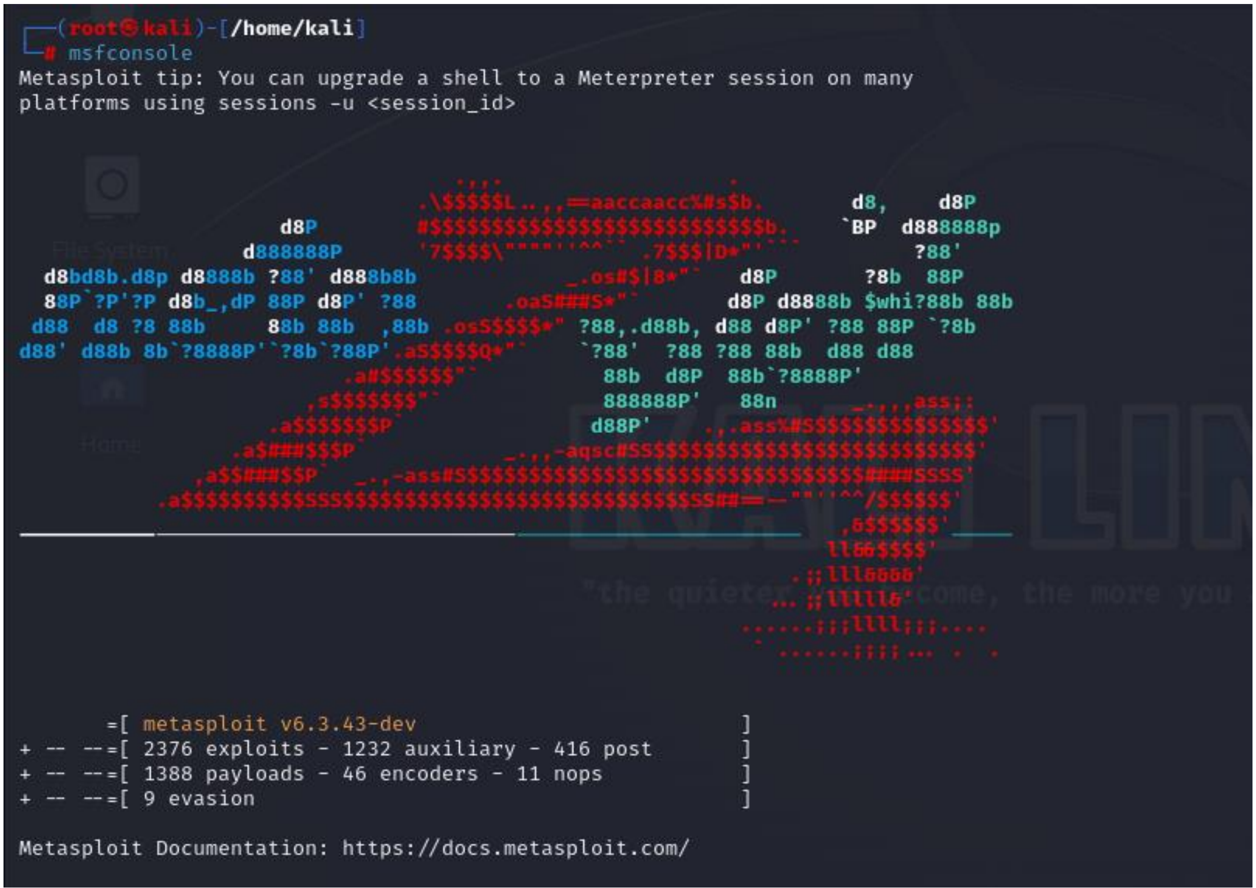


```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sC 172.17.17.230
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-07 04:15 EST
Nmap scan report for 172.17.17.230
Host is up (0.0033s latency).
Not shown: 992 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5357/tcp  open  wsdapi
49152/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49160/tcp open  unknown

Host script results:
|_clock-skew: mean: -1h50m00s, deviation: 3h10m31s, median: 0s
| smb2-security-mode:
|   3:0:2:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2024-02-07T09:15:13
|_  start_date: 2024-02-06T07:27:44
| smb-os-discovery:
|   OS: Windows 8.1 Pro 9600 (Windows 8.1 Pro 6.3)
|   OS CPE: cpe:/o:microsoft:windows_8.1::-
|   Computer name: 16DITB310-20
|   NetBIOS computer name: 16DITB310-20\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2024-02-07T14:45:13+05:30
| smb-security-mode:
|   account_used: guest
```

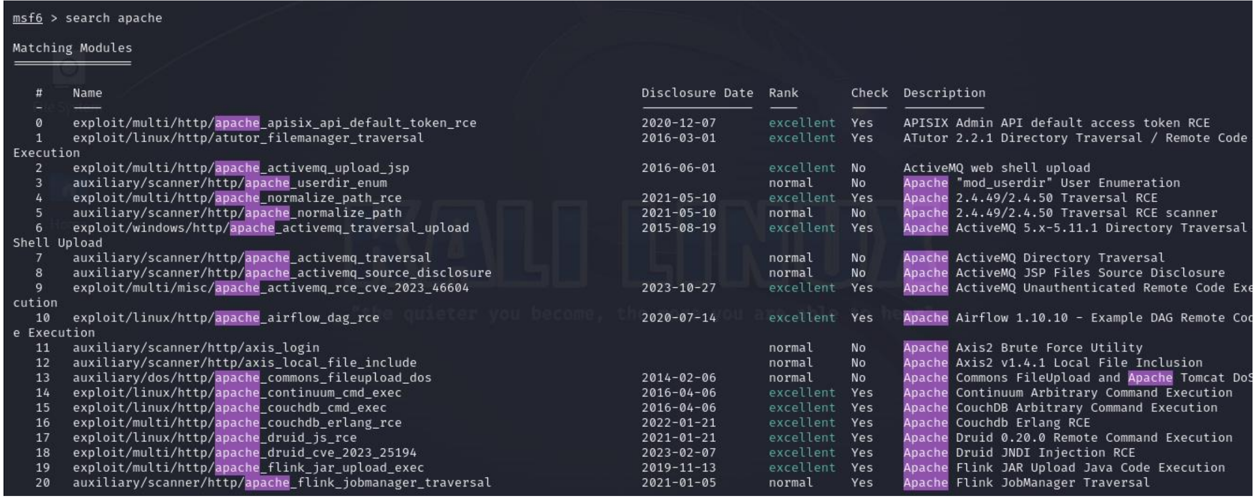
2. METASPLOIT



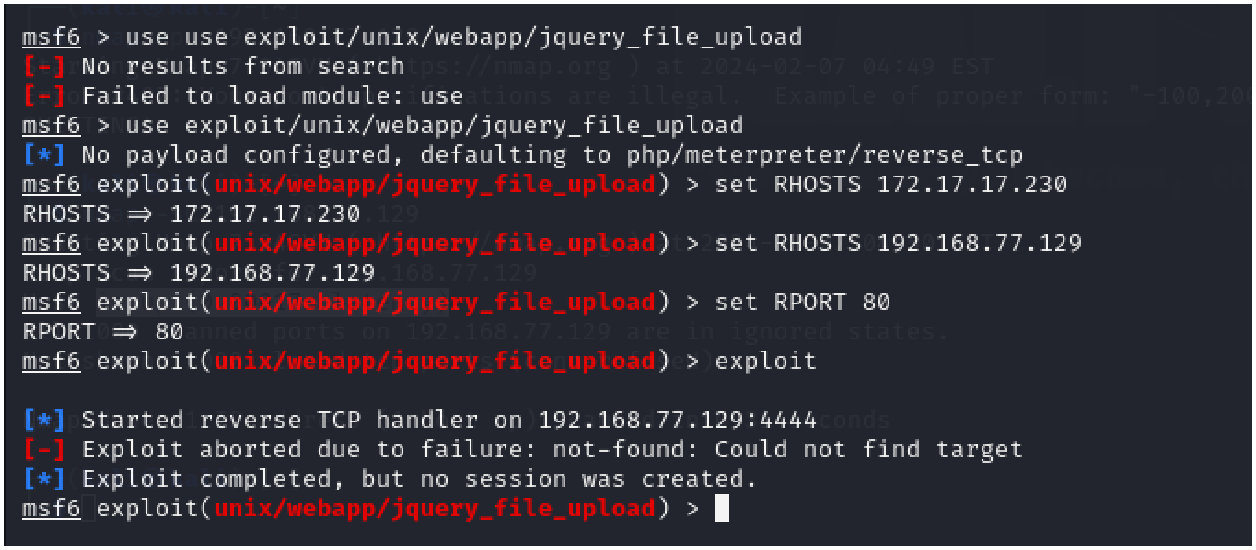
```
┌──(root㉿kali)-[/home/kali]
└─# apt-get install metasploit-framework
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  clamav clamav-daemon
The following packages will be upgraded:
  metasploit-framework
1 upgraded, 0 newly installed, 0 to remove and 1146 not upgraded.
Need to get 220 MB of archives.
After this operation, 2,207 kB of additional disk space will be used.
Err:1 http://kali.download/kali kali-rolling/main amd64 metasploit-framework amd64 6.3.52-0kali1
  403  Forbidden [IP: 104.18.103.100 80]
E: Failed to fetch http://kali.download/kali/pool/main/m/metasploit-framework/metasploit-framework_6.3.52-0kali1_amd64
E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?
```



```
┌──(root㉿kali)-[/home/kali]
└─# msfconsole
Metasploit tip: You can upgrade a shell to a Meterpreter session on many
platforms using sessions -u <session_id>

       =[ metasploit v6.3.43-dev                          ]
+ -- --=[ 2376 exploits - 1232 auxiliary - 416 post       ]
+ -- --=[ 1388 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/
```

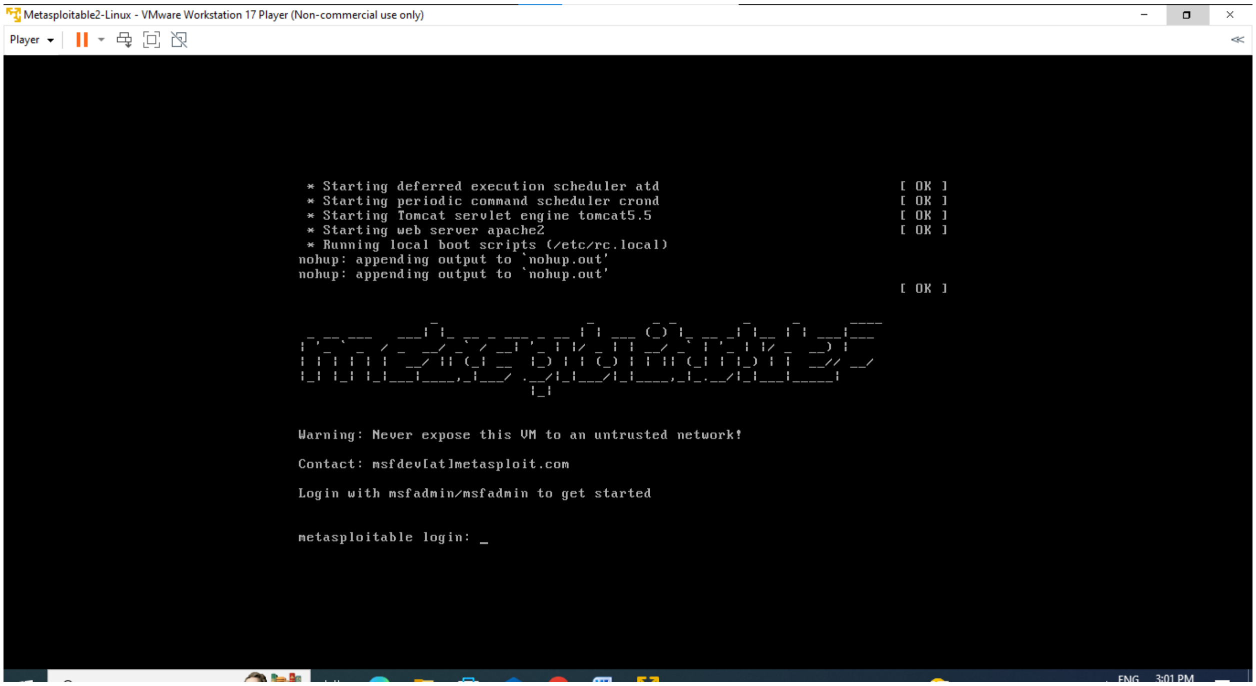


```
msf6 > search apache

Matching Modules
================

   #   Name                                                      Disclosure Date  Rank       Check  Description
   -   ----                                                      ---------------  ----       -----  -----------
   0   exploit/multi/http/apache_apisix_api_default_token_rce   2020-12-07       excellent  Yes    APISIX Admin API default access token RCE
   1   exploit/linux/http/atutor_filemanager_traversal           2016-03-01       excellent  Yes    ATutor 2.2.1 Directory Traversal / Remote Code
Execution
   2   exploit/multi/http/apache_activemq_upload_jsp             2016-06-01       excellent  No     ActiveMQ web shell upload
   3   auxiliary/scanner/http/apache_userdir_enum                                 normal     No     Apache "mod_userdir" User Enumeration
   4   exploit/multi/http/apache_normalize_path_rce              2021-05-10       excellent  Yes    Apache 2.4.49/2.4.50 Traversal RCE
   5   auxiliary/scanner/http/apache_normalize_path              2021-05-10       normal     No     Apache 2.4.49/2.4.50 Traversal RCE scanner
   6   exploit/windows/http/apache_activemq_traversal_upload     2015-08-19       excellent  Yes    Apache ActiveMQ 5.x-5.11.1 Directory Traversal
Shell Upload
   7   auxiliary/scanner/http/apache_activemq_traversal                           normal     No     Apache ActiveMQ Directory Traversal
   8   auxiliary/scanner/http/apache_activemq_source_disclosure                   normal     No     Apache ActiveMQ JSP Files Source Disclosure
   9   exploit/multi/misc/apache_activemq_rce_cve_2023_46604     2023-10-27       excellent  Yes    Apache ActiveMQ Unauthenticated Remote Code Exe
cution
   10  exploit/linux/http/apache_airflow_dag_rce                 2020-07-14       excellent  Yes    Apache Airflow 1.10.10 - Example DAG Remote Cod
e Execution
   11  auxiliary/scanner/http/axis_login                                          normal     No     Apache Axis2 Brute Force Utility
   12  auxiliary/scanner/http/axis_local_file_include                             normal     No     Apache Axis2 v1.4.1 Local File Inclusion
   13  auxiliary/dos/http/apache_commons_fileupload_dos          2014-02-06       normal     No     Apache Commons FileUpload and Apache Tomcat DoS
   14  exploit/linux/http/apache_continuum_cmd_exec              2016-04-06       excellent  Yes    Apache Continuum Arbitrary Command Execution
   15  exploit/linux/http/apache_couchdb_cmd_exec                2016-04-06       excellent  Yes    Apache CouchDB Arbitrary Command Execution
   16  exploit/multi/http/apache_couchdb_erlang_rce              2022-01-21       excellent  Yes    Apache Couchdb Erlang RCE
   17  exploit/linux/http/apache_druid_js_rce                    2021-01-21       excellent  Yes    Apache Druid 0.20.0 Remote Command Execution
   18  exploit/multi/http/apache_druid_cve_2023_25194            2023-02-07       excellent  Yes    Apache Druid JNDI Injection RCE
   19  exploit/multi/http/apache_flink_jar_upload_exec           2019-11-13       excellent  Yes    Apache Flink JAR Upload Java Code Execution
   20  auxiliary/scanner/http/apache_flink_jobmanager_traversal  2021-01-05       normal     Yes    Apache Flink JobManager Traversal
```



```
msf6 > use use exploit/unix/webapp/jquery_file_upload
[-] No results from search
[-] Failed to load module: use
msf6 > use exploit/unix/webapp/jquery_file_upload
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/jquery_file_upload) > set RHOSTS 172.17.17.230
RHOSTS => 172.17.17.230
msf6 exploit(unix/webapp/jquery_file_upload) > set RHOSTS 192.168.77.129
RHOSTS => 192.168.77.129
msf6 exploit(unix/webapp/jquery_file_upload) > set RPORT 80
RPORT => 80
msf6 exploit(unix/webapp/jquery_file_upload) > exploit

[*] Started reverse TCP handler on 192.168.77.129:4444
[-] Exploit aborted due to failure: not-found: Could not find target
[*] Exploit completed, but no session was created.
msf6 exploit(unix/webapp/jquery_file_upload) > 
```

METASPLOIT VIRTUAL MACHINE


Metasploitable2-Linux - VMware Workstation 17 Player (Non-commercial use only)
Player

```
 * Starting deferred execution scheduler atd                          [ OK ]
 * Starting periodic command scheduler crond                          [ OK ]
 * Starting Tomcat servlet engine tomcat5.5                           [ OK ]
 * Starting web server apache2                                        [ OK ]
 * Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out'
                                                                      [ OK ]

                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: _
```

Post Lab Questions:-

1. **You are tasked with securing a Wi-Fi network against potential attacks. You perform a wireless audit using Aircrack-ng as part of your security assessment.**

    1. **Identify Weaknesses**: Analyze the results of the wireless audit to identify potential weaknesses and vulnerabilities in the Wi-Fi network. Look for common issues such as weak encryption, default or easily guessable passwords, misconfigured access points, or rogue

devices.

2. **Implement Security Measures:**
   - Strengthen Encryption: Ensure that the Wi-Fi network is using strong encryption protocols such as WPA2 or WPA3.
   - Use Complex Passwords: Enforce the use of complex and unique passwords for Wi-Fi access, avoiding default or easily guessable passwords.
   - Disable WPS: If not needed, disable Wi-Fi Protected Setup (WPS) as it can be vulnerable to brute-force attacks.
   - Enable MAC Address Filtering: Restrict access to the Wi-Fi network by only allowing devices with approved MAC addresses.
   - Enable Intrusion Detection: Implement intrusion detection systems (IDS) or intrusion prevention systems (IPS) to monitor for suspicious activity on the Wi-Fi network.
3. **Regularly Monitor and Update:** Continuously monitor the Wi-Fi network for any signs of unauthorized access or suspicious behavior. Regularly update firmware and security patches for Wi-Fi routers and access points to mitigate known vulnerabilities.
4. **Educate Users:** Educate users about best practices for Wi-Fi security, such as avoiding public Wi-Fi networks, being cautious of phishing attempts, and not sharing sensitive information over unsecured networks.

2. **You are conducting a security assessment for an organization that relies heavily on wireless networks. Your goal is to identify potential vulnerabilities and weaknesses in their wireless infrastructure.**

- **Inventory of Wireless Infrastructure:** Conduct a comprehensive inventory of all wireless devices, access points, routers, and other wireless equipment used within the organization.
- **Wireless Site Survey:** Perform a wireless site survey to assess the coverage, signal strength, and potential interference in different areas of the organization's premises. Identify any dead zones or areas with weak wireless connectivity.
- **Vulnerability Assessment:** Use tools like Aircrack-ng, Wireshark, or Kismet to identify potential vulnerabilities in the wireless infrastructure. Look for security weaknesses such as open networks, weak encryption, rogue access points, or misconfigured settings.
- **Penetration Testing**: Conduct penetration testing to simulate real-world attacks and assess the effectiveness of security controls in place. Attempt to exploit identified vulnerabilities to gain unauthorized access to the wireless network or compromise sensitive data.
- **Risk Mitigation Recommendations**: Based on the findings of the security assessment, provide recommendations for mitigating identified risks and vulnerabilities. This may include implementing stronger encryption, updating firmware, configuring access controls, or improving security policies and procedures.
- **Documentation and Reporting**: Document the findings of the security assessment, including a summary of vulnerabilities, their potential impact, and recommended remediation steps. Present the findings to the organization's stakeholders, including management, IT staff, and security teams.

Outcomes:

CO1: Realize that premise of vulnerability analysis and penetration testing (VAPT)

Conclusion: (Conclusion to be based on the objectives and outcomes achieved)

Successfully Explored network landscapes using Nmap and Metasploit for penetration testing in Kali linux

Signature of faculty in charge with date

References:

1. https://www.guru99.com/kali-linux-tutorial.html
2. https://phoenixts.com/blog/learn-to-pen-test-with-kali-linux/
https://www.kali.org/docs/introduction/should-i-use-kali-linux/
