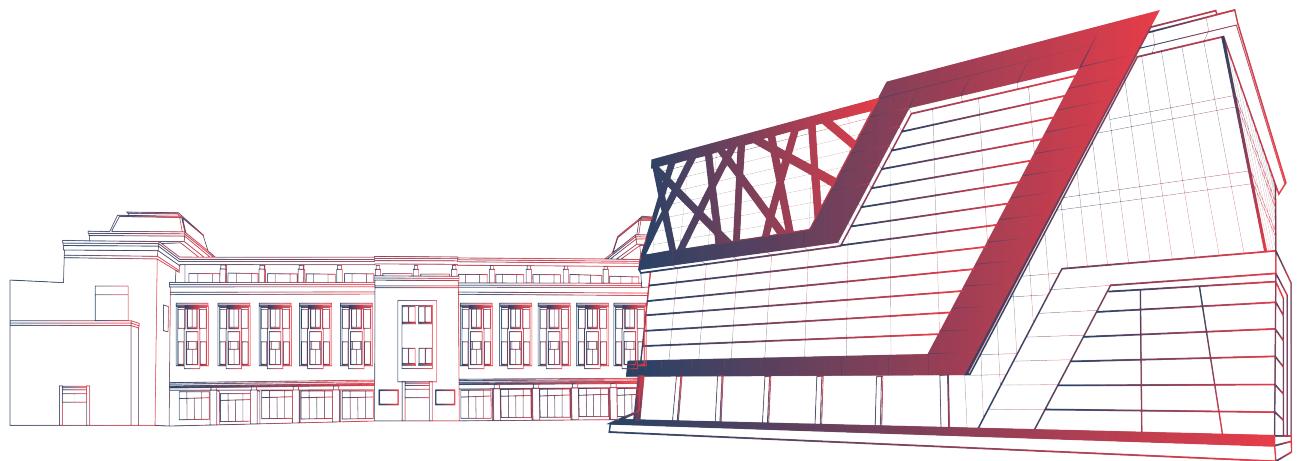


Wordlists Generator-Crunch

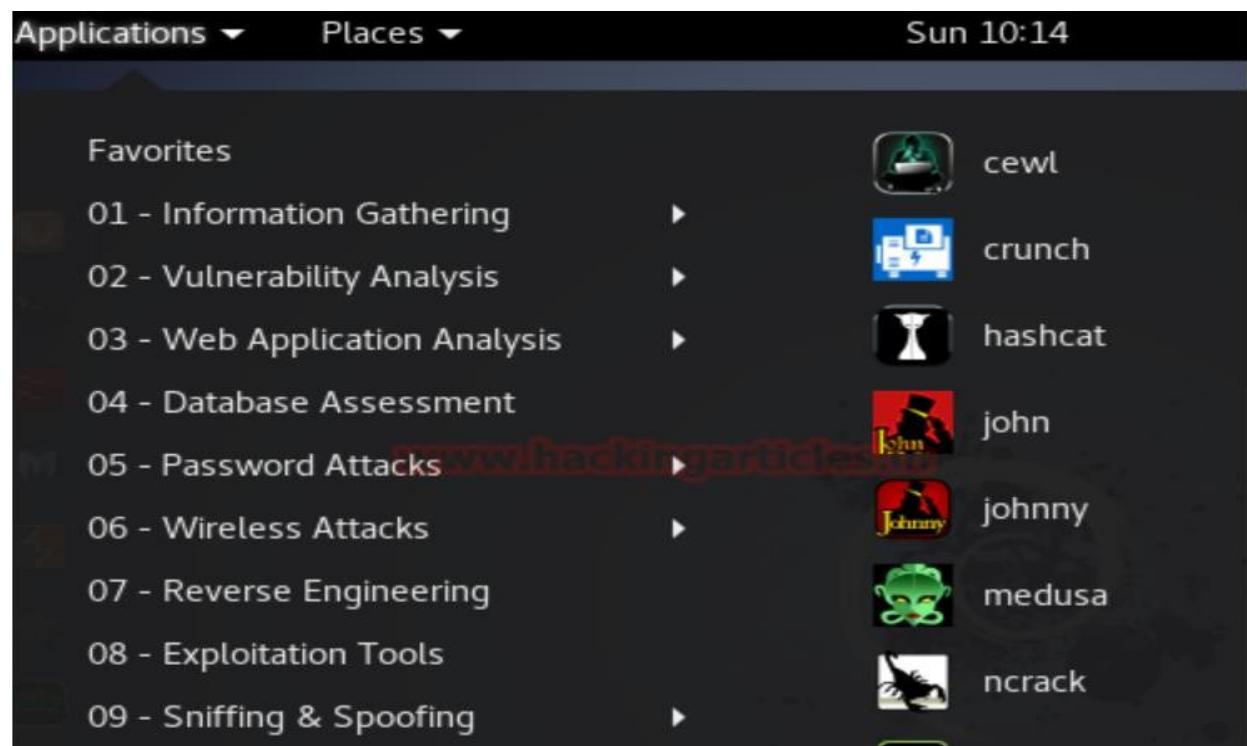


Wordlists Generator-Crunch

Crunch is a wordlist generator

- You can use a standard character set or
- A customized character set

Crunch is **pre-installed** on Kali



Wordlists Generator-crunch

Features:

- Crunch generates wordlists in **combination and permutation** ways
- It can breakup **output by number of lines or file size**
- It supports:
 - **number and symbols**
 - **upper and lower case characters separately**

Wordlists Generator-crunch

Usage Syntax –

Crunch

```
root@kali:~# crunch  
crunch version 3.6
```

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

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.

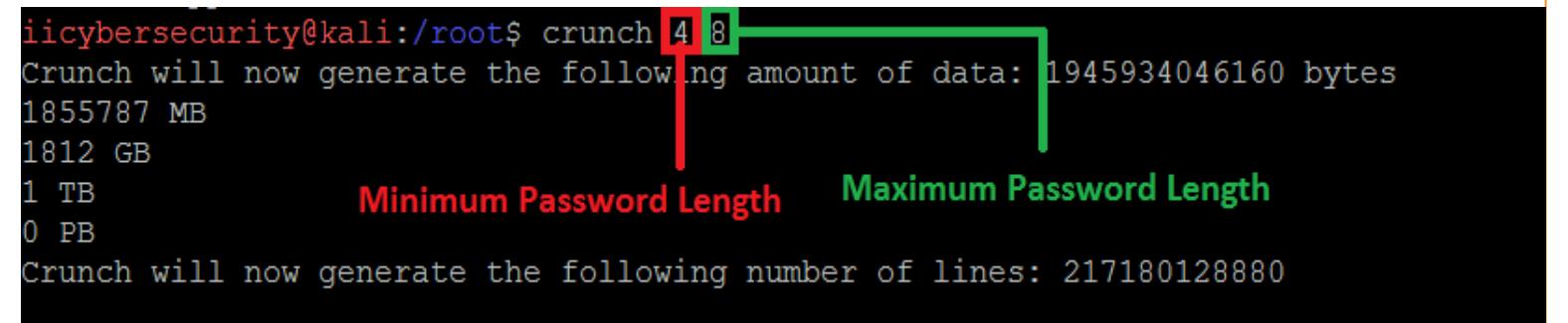
Wordlists Generator-crunch

Usage Syntax -

Crunch Min.Value Max.Value Characters

Example –

crunch 4 4 0123456789



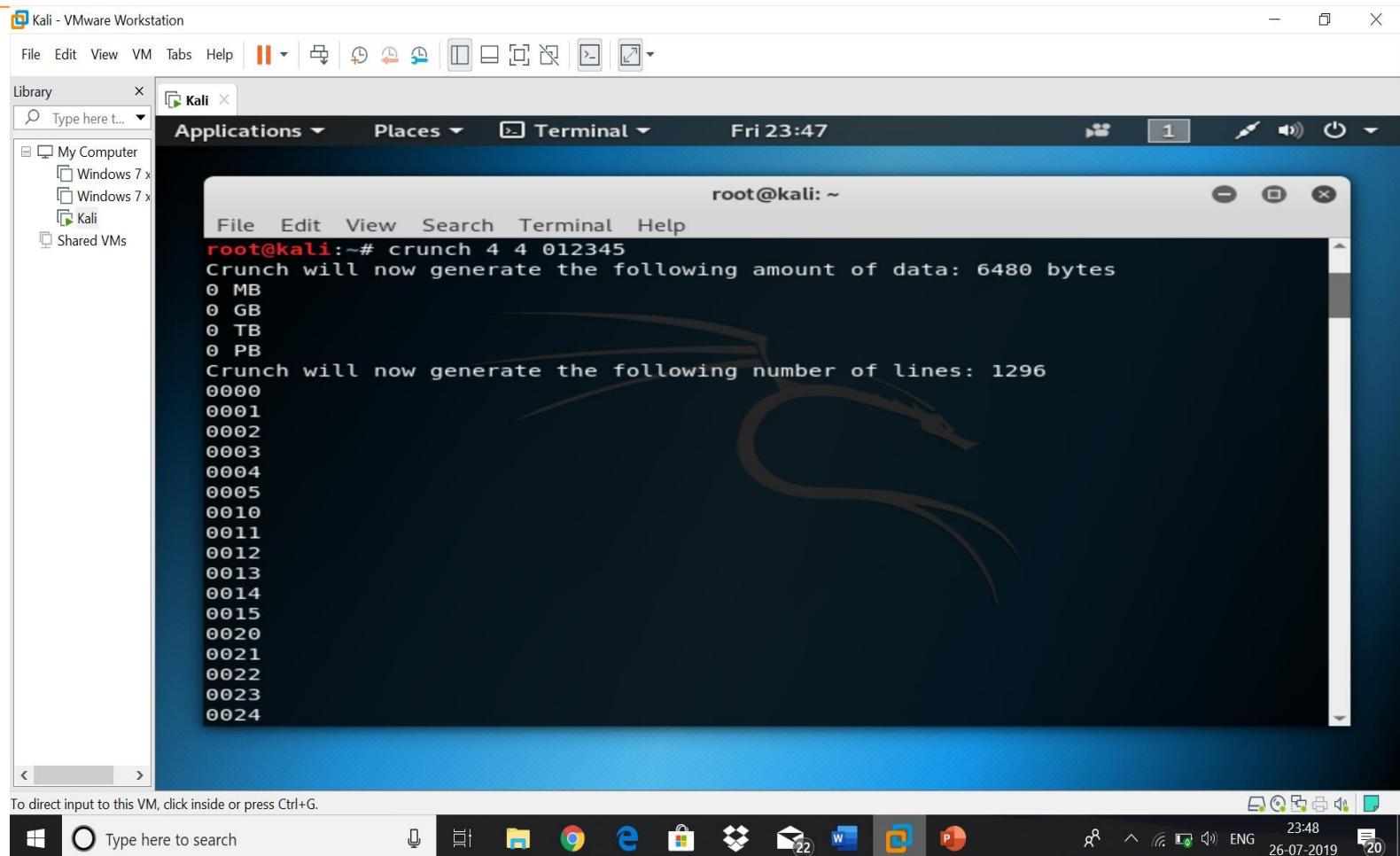
```
iicybersecurity@kali:/root$ crunch 4 8 0123456789
Crunch will now generate the following amount of data: 1945934046160 bytes
1855787 MB
1812 GB
1 TB
0 PB
Crunch will now generate the following number of lines: 217180128880
```

Wordlists Generator-crunch

crunch 4 4 012345

- Combinations will start from 0
- 1296 combinations

12345 are the numbers used to generate the list.



A screenshot of a Kali Linux desktop environment within a VMware Workstation window. The desktop includes a taskbar with icons for File Explorer, Edge browser, File Explorer, Mail, and others. A terminal window titled 'Kali' is open, showing the command 'crunch 4 4 012345' being run. The terminal output indicates that Crunch will generate 1296 lines of data, starting from 0 MB and ending at 0 PB. The desktop background features a dark blue gradient with a stylized orange and red swoosh graphic.

```
root@kali:~# crunch 4 4 012345
Crunch will now generate the following amount of data: 6480 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 1296
0000
0001
0002
0003
0004
0005
0010
0011
0012
0013
0014
0015
0020
0021
0022
0023
0024
```

Wordlists Generator-crunch

For saving it to a file:

```
crunch 4 4 0123456789 > filename.txt
```

OR

```
crunch 4 4 0123456789 -o /root/Desktop/wordlist.txt
```

- **-o /path/to/save/list** is used to specify the location to save the wordlist.

Crunch USING -i OPTION

- **-i** is used to inverts the output.
- If we don't use **-i**, crunch will generate list in **aa, ab, ac, ad, ae** but if you use **-i**, crunch will generate list in **aa, ba, ca, da, ea** as shown below.

```
root@kali:~# crunch 1 2 abcdefgh -i -o /root/wordlist2.txt
Crunch will now generate the following amount of data: 208 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 72
a
b
c
d
e
f
g
h
aa
ba
ca
da
```



Crunch USING -t OPTION

- **-t** allows you to specify a pattern
 - @ will insert lower case characters
 - , will insert upper case characters
 - % will insert numbers
 - ^ will insert symbols

Crunch USING -t OPTION

- @ : generate lowercase
- Type **crunch 10 10 -t @@@@ @@@@ 0316 -o /root/birth_datlist.txt**

```
root@kali:~# crunch 10 10 -t @@@@ @@@@ 0316 -o /root/birth_datlist.txt
Crunch will now generate the following amount of data: 3398073536 bytes
3240 MB
3 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 308915776
```

aaaaaa0316
aaaaab0316
aaaaac0316
aaaaad0316
aaaaae0316
aaaaaf0316
aaaaag0316
aaaaah0316
aaaaai0316
aaaaaj0316

Crunch USING -t OPTION

- , : generate uppercase
- Type **crunch 10 10 -t ,,,,0316 -o /root/birth_dat2list.txt**
- **0316** use as birthday dates to generate list.

```
root@kali:~# crunch 10 10 -t ,,,,0316 -o /root/birth_dat2list.txt
Crunch will now generate the following amount of data: 3398073536 bytes
3240 MB
```

AAAAAA0316
AAAAAAB0316
AAAAAAC0316
AAAAAAD0316
AAAAAAE0316
AAAAAAF0316
AAAAAG0316
AAAAAH0316
AAAAAI0316
AAAAAJ0316



Crunch USING -t OPTION

- % : generate numbers.
- Type **crunch 10 10 -t %%%%%%0618 -o /root/petnameslist.txt**
- **0618** use as birthday dates to generate list.

```
root@kali:~# crunch 10 10 -t %%%%%%0618 -o /root/petnameslist.txt
Crunch will now generate the following amount of data: 11000000 bytes
```

0000000618
0000010618
0000020618
0000030618
0000040618
0000050618
0000060618
0000070618
0000080618
0000090618



Crunch- In class Exercise

- Now, if I want to generate a file with **some specific characters** in a word we can go like
 - crunch 9 9 -t ^%hello%%
 - 1 imp Condition: **min and max length has to be of same size** when you are creating a pattern
 - That size should be equal to the pattern
 - e.g. `strlen(^%hello%%)` = 9

Heading Goes Here

Content goes here

And here

Here too

- Maybe here too
- Maybe here too
- Maybe here too