

Task 4: The Key Under the Mat - Basic Password Cracking

Name :- Rushikesh Sanjay Kumavat

INTERN ID:- SN1000726

DOMAIN:- Cyber Security

Aim :-

To understand password security by cracking a hashed password using John the Ripper and a dictionary attack.

OBJECTIVE :-

- Generate a password hash
- Store it in a file
- Use John the Ripper with rockyou wordlist
- Crack the password
- Analyze the weakness of weak passwords

Tools Used :-

- Kali Linux
- Metasploitable 2
- John the Ripper

Procedure :-

Step 1: Generate Password Hash

Command used:

➔ openssl passwd -1 password123

Output:

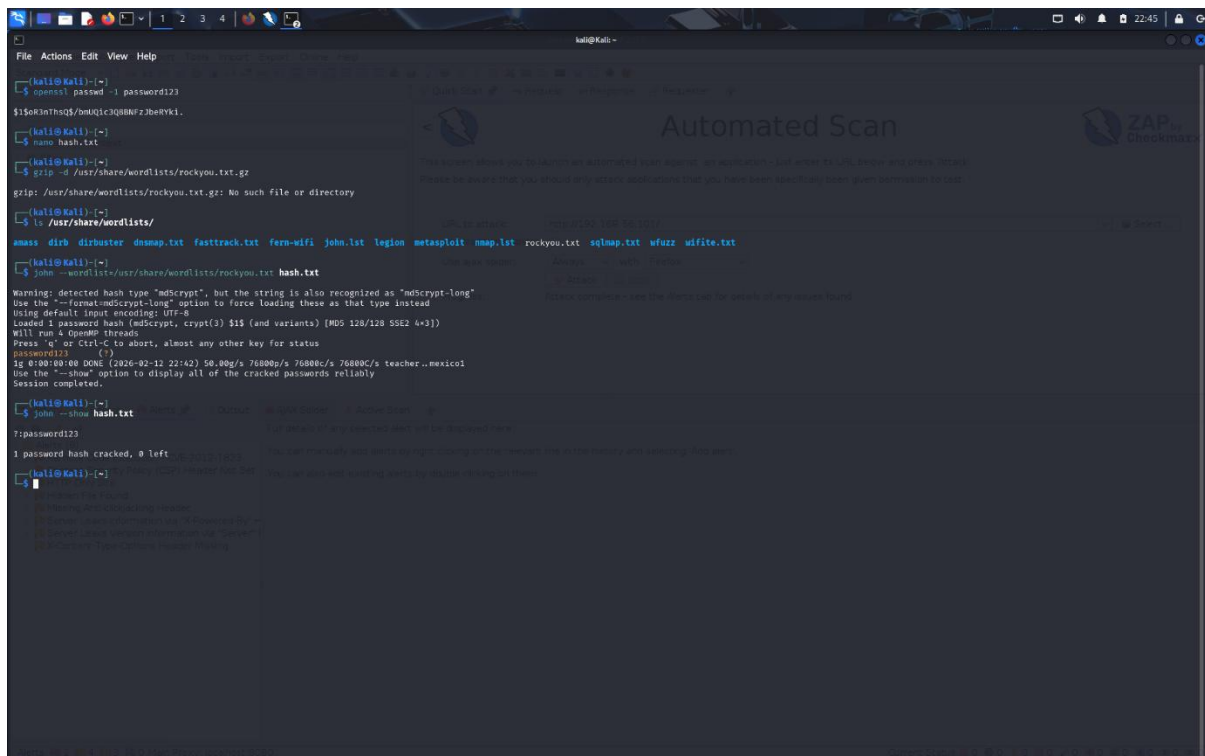
➔ \$1\$0R3nThsQ\$/bmUQic3Q8BNFzJbeRYki_

Step 2: Save Hash in File

Command:

➔ nano hash.txt

Paste the hash and save.



```
kali@kali:~$ openssl passwd -1 password123
$1$ok1nHsq$/bmq1c3Q8Mfz3beYk1..

kali@kali:~$ nano hash.txt

kali@kali:~$ gzip -d /usr/share/wordlists/rockyou.txt.gz
gzip: /usr/share/wordlists/rockyou.txt.gz: No such file or directory

kali@kali:~$ ls /usr/share/wordlists/
anass  dirb  dirbuster  dnsmap.txt  fasttrack.txt  fern-wifi  john.lst  legion  metasploit  nmap.lst  rockyou.txt  sqlmap.txt  wfuzz  wfite.txt

kali@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4x3])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password123:
1g 0:00:00.00 DONE (2026-02-12 22:42) 56.00q/s 76800p/s 76800c/s 76800C/s teacher..mexico!
Session completed.

kali@kali:~$ john --show hash.txt
7:password123

1 password hash cracked, 0 left

kali@kali:~$
```

Step 3: Verify Wordlist Location

Command:

➔ ls /usr/share/wordlists/

Confirmed:

➔ rockyou.txt

Step 4: Perform Dictionary Attack

Command:

➔ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt

John loaded hash type:

➔ md5crypt

Password Cracked:

➔ password123

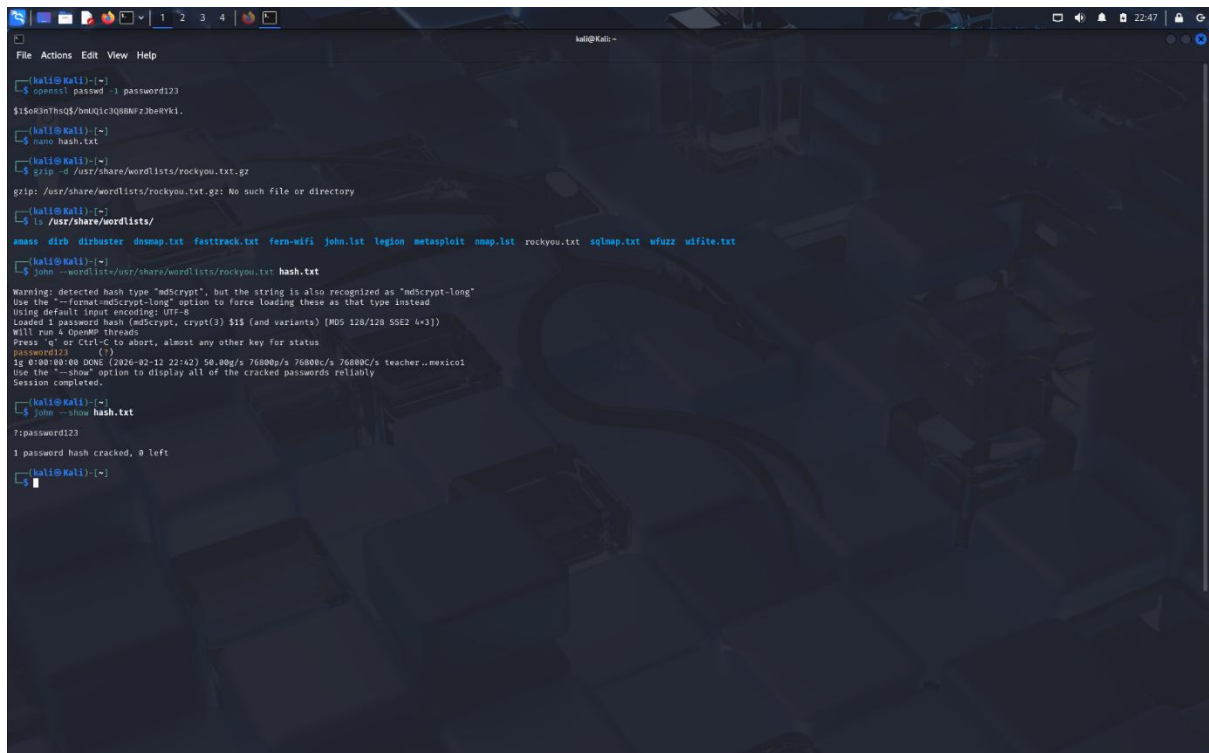
Step 5: Display Cracked Password

Command:

➔ john --show hash.txt

Output:

➔ password123



```
(kali@kali:~)$ openssl passwd -1 password123
$1$ok3nThsQ$/bmQ1c3Q8MFz3beYk1.

(kali@kali:~)$ nano hash.txt

(kali@kali:~)$ gzip -d /usr/share/wordlists/rockyou.txt.gz
gzip: /usr/share/wordlists/rockyou.txt.gz: No such file or directory

(kali@kali:~)$ ls /usr/share/wordlists/
unass  dirb  dirbuster  dsmap.txt  fasttrack.txt  fern-wifi  john.lst  legion  metasploit  nmap.lst  rockyou.txt  sqlmap.txt  wfuzz  wifite.txt

(kali@kali:~)$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt

Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4+3])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password123 (1)
1g 0:00:00.00 DONE (2026-02-12 22:42) 50.90g/s 76800p/s 76800c/s 76800C/s teacher..mexico!
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(kali@kali:~)$ john --show hash.txt
?:password123
1 password hash cracked, 0 left

(kali@kali:~)$
```

RESULT :-

The password hash was successfully cracked using a dictionary attack.
The original password was:

➔ password123

CONCLUSION :-

This task demonstrates:

- Weak passwords can be cracked quickly
- Dictionary attacks are very effective
- Proper password policies are necessary
- Hashing alone is not enough without strong passwords