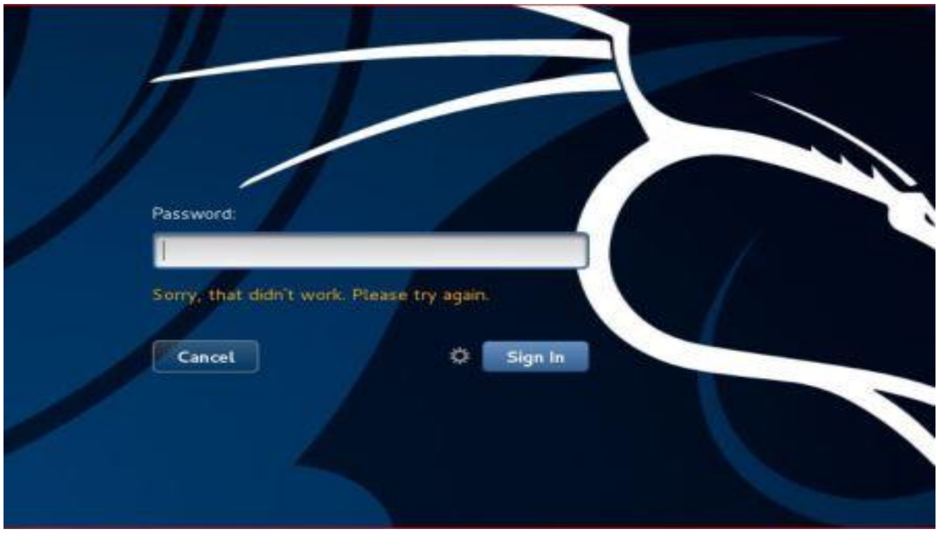
LAB MANUAL
ON
PASSWORD CRACKING OF KALI LINUX OPERATING SYSTEM


Password:

Sorry, that didn't work. Please try again.

ESTABLISHMENT OF ADVANCED LABORATORY FOR CYBER SECURITY TRAINING TO TECHNICAL TEACHERS

DEPARTMENT OF INFORMATION MANAGEMENT AND EMERGING ENGINEERING

MINISTRY OF ELECTRONICS AND INFORMATION TECHNOLOGY

GOVERNMENT OF INDIA

Principal Investigator: Prof. Maitreyee Dutta

Co Investigator: Prof. Shyam Sundar Pattnaik

PREPARED BY:

Prof. Maitreyee Dutta and Ms. Shweta Sharma (Technical Assistant)

Table of Contents

MANUAL-5: PASSWORD CRACKING OF KALI LINUX OPERATING SYSTEM

INTRODUCTION TO KALI LINUX OPERATING SYSTEM

- Kali Linux is a Debian-derived Linux distribution operating system which is designed for digital forensics and penetration testing.
- Kali Linux operating system is maintained and funded by Offensive Security.
- The first version (1.0) of Kali Linux operating system was released in March 2013 [1].
- This operating system has over 600 pre-installed penetration testing and security tools such as Nmap, John the Ripper, Aircrack-ng, Hashcat, Metasploit framework, and so on.

PASSWORD STORAGE IN KALI LINUX OPERATING SYSTEM

- Passwords are used to protect the system from an unauthorized access.
- Computers with Kali Linux operating system stores password in /etc/shadow file in the form of Message Digest 5

(MD5)/ Blowfish/ Secure Hash Algorithm (SHA-256/ SHA-512) hash.

- Passwords are stored in the form of hash due to its irreversible property. This means that password in plaintext can be converted to hash but a hash can't be converted back to plaintext.

PASSWORD CRACKING

- Password cracking in Kali Linux operating system is a process to recover passwords from a shadow file.
- The purpose of password cracking is to recover forgotten password. The forensic team can perform password cracking on a computer system to recover the data after getting the password.
- This is usually accomplished by recovering the passwords from data stored in the shadow file in the form of a hash value.

PASSWORD CRACKING TECHNIQUES

The password cracking techniques are discussed as follows:

- **BRUTE FORCE**: A brute force technique is an attempt to crack passwords using permutation and combination approach. This method takes a lot of time and memory consumption depending on the length and complexity of password.

- **DICTIONARY**: A dictionary technique is an attempt to store in-build passwords in a file known as dictionary. Instead of trying all combination of passwords, it creates a word-list of most common passwords and calculates the hash values while cracking the passwords. It will only able to crack the password if it is stored in dictionary file. This technique takes less time as compared to brute-force technique to crack the password.

- **RAINBOW TABLES**: This technique is same as dictionary, but instead of calculating hash vales during password cracking; it stores the in-built hash values of password in the tables. Thus, this technique takes less time as compared to brute-force and dictionary technique to crack the password.

JOHN-THE-RIPPER TOOL

- The John-the-ripper tool [2] is an open-source application and post-exploitation Kali Linux operating system tool that allows users to view authentication credentials.

- This tool provides hashes from shadow file of Kali Linux operating system to users.

- Kali Linux store password data in a shadow file in the form of a hash. The forensics team can use John-the-ripper tool to get the password in plain text and pass it to the target computer to login.

PASSWORD CRACKING WITH JOHN-THE-RIPPER TOOL

The password in plaintext from hash can be recovered with John-the-ripper tool with the following steps:

Step 1: Open Kali Linux operating system as shown in Figure 1.

Figure 1: Kali Linux operating system

<u>Step 2:</u> In Kali Linux operating system, open John-the-ripper tool. Go to Applications-> Password attacks-> john as shown in Figure 2.

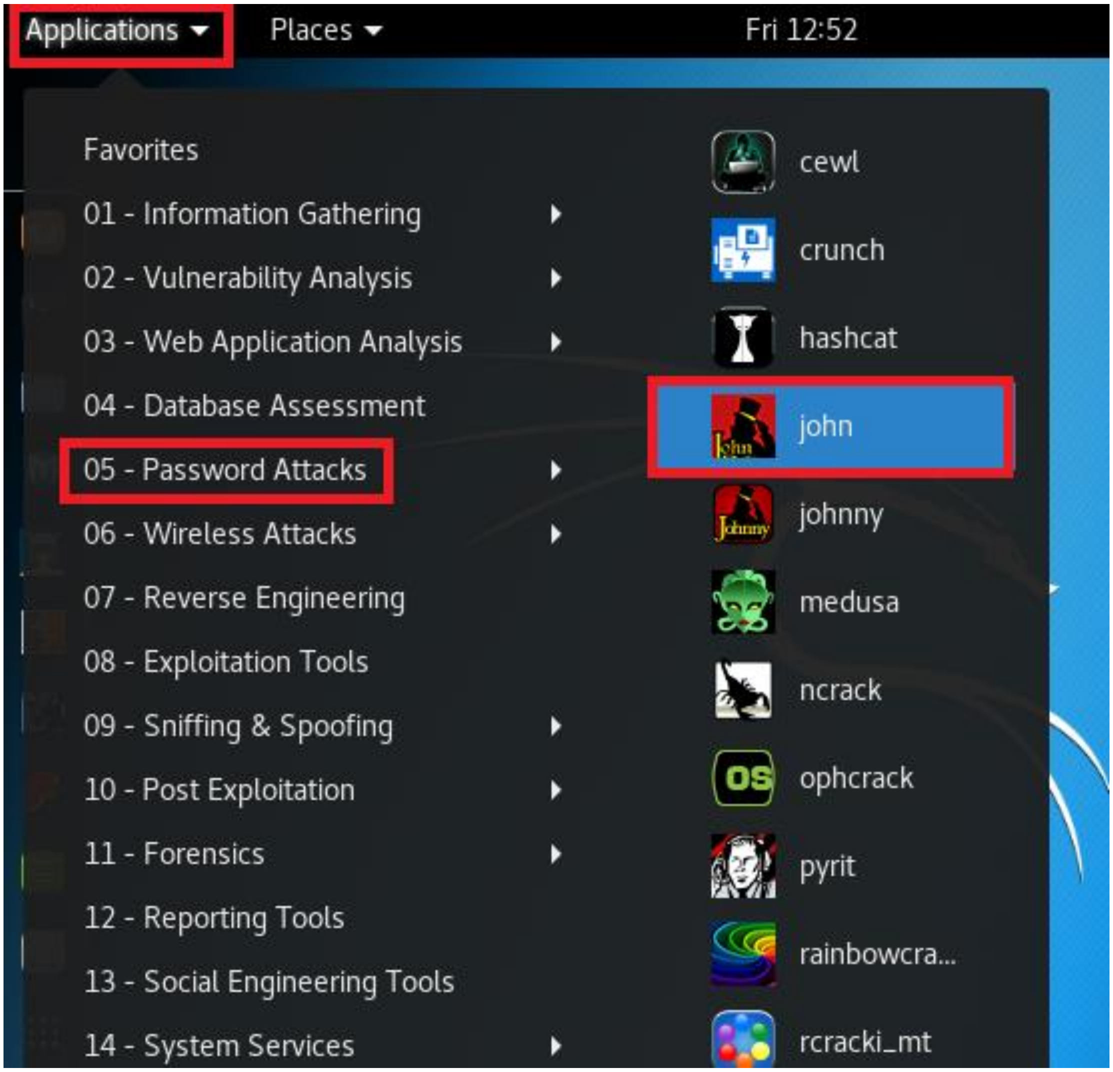
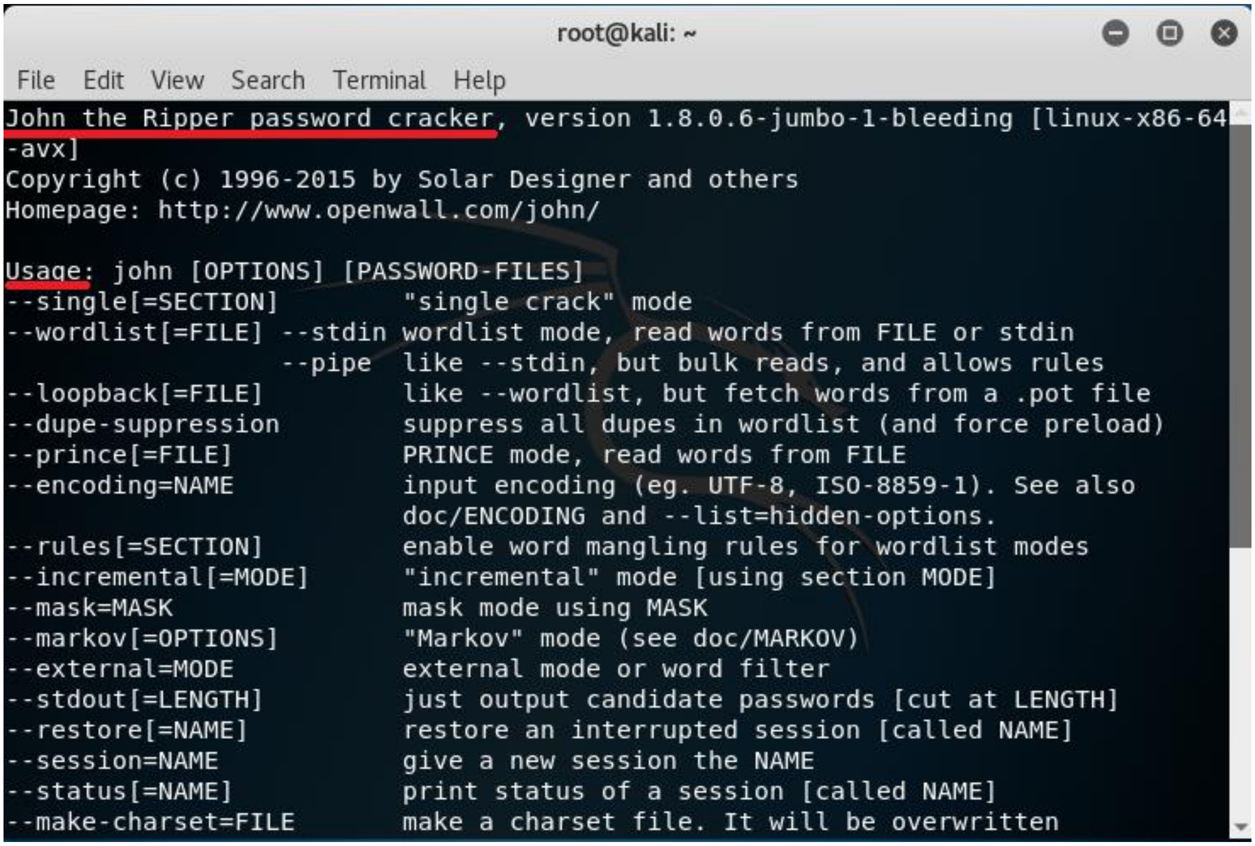


Figure 2: Opening John-the-Ripper tool

Step 3: A terminal with usage of John-the-ripper tool will open as shown in Figure 3 and Figure 4.

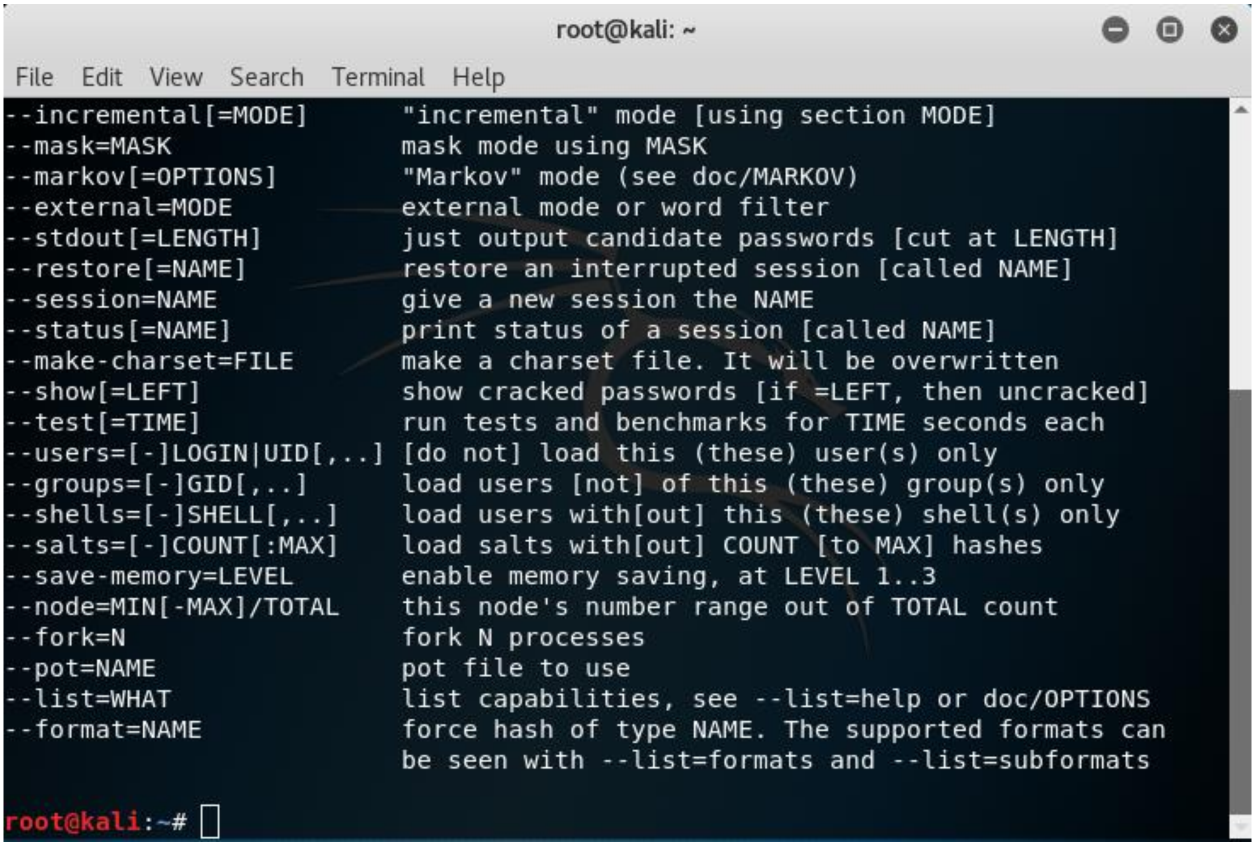


```
John the Ripper password cracker, version 1.8.0.6-jumbo-1-bleeding [linux-x86-64
-avx]
Copyright (c) 1996-2015 by Solar Designer and others
Homepage: http://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]
--single[=SECTION]        "single crack" mode
--wordlist[=FILE] --stdin wordlist mode, read words from FILE or stdin
                  --pipe  like --stdin, but bulk reads, and allows rules
--loopback[=FILE]         like --wordlist, but fetch words from a .pot file
--dupe-suppression        suppress all dupes in wordlist (and force preload)
--prince[=FILE]           PRINCE mode, read words from FILE
--encoding=NAME           input encoding (eg. UTF-8, ISO-8859-1). See also
                          doc/ENCODING and --list=hidden-options.
--rules[=SECTION]         enable word mangling rules for wordlist modes
--incremental[=MODE]      "incremental" mode [using section MODE]
--mask=MASK               mask mode using MASK
--markov[=OPTIONS]        "Markov" mode (see doc/MARKOV)
--external=MODE           external mode or word filter
--stdout[=LENGTH]         just output candidate passwords [cut at LENGTH]
--restore[=NAME]          restore an interrupted session [called NAME]
--session=NAME            give a new session the NAME
--status[=NAME]           print status of a session [called NAME]
--make-charset=FILE       make a charset file. It will be overwritten
```

Figure 3: John-the-Ripper tool in Terminal

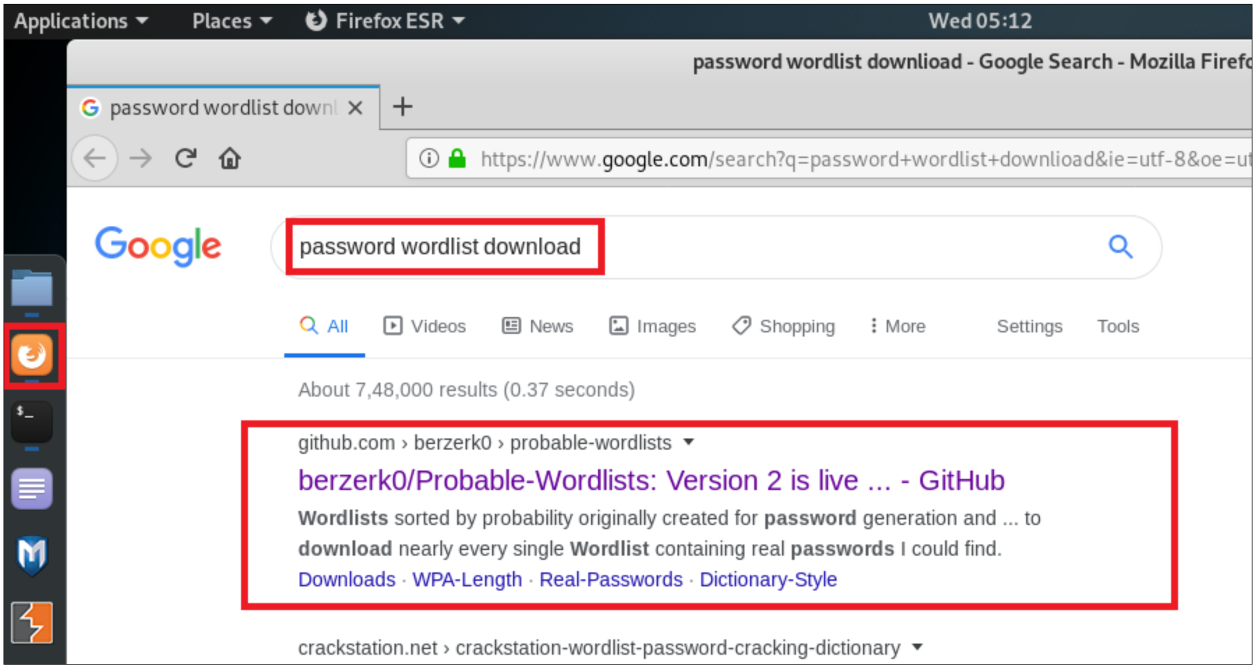
Step 4: Search the password wordlist by browsing Google search engine as shown in Figure 5. Open the GitHub website and download the ZIP file as shown in Figure 6.



```
--incremental[=MODE]       "incremental" mode [using section MODE]
--mask=MASK                mask mode using MASK
--markov[=OPTIONS]         "Markov" mode (see doc/MARKOV)
--external=MODE            external mode or word filter
--stdout[=LENGTH]          just output candidate passwords [cut at LENGTH]
--restore[=NAME]           restore an interrupted session [called NAME]
--session=NAME             give a new session the NAME
--status[=NAME]            print status of a session [called NAME]
--make-charset=FILE        make a charset file. It will be overwritten
--show[=LEFT]              show cracked passwords [if =LEFT, then uncracked]
--test[=TIME]              run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..]  [do not] load this (these) user(s) only
--groups=[-]GID[,..]       load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..]     load users with[out] this (these) shell(s) only
--salts=[-]COUNT[:MAX]     load salts with[out] COUNT [to MAX] hashes
--save-memory=LEVEL        enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL     this node's number range out of TOTAL count
--fork=N                   fork N processes
--pot=NAME                 pot file to use
--list=WHAT                list capabilities, see --list=help or doc/OPTIONS
--format=NAME              force hash of type NAME. The supported formats can
                           be seen with --list=formats and --list=subformats

root@kali:~# 
```

Figure 4: John-the-Ripper tool in Terminal


Applications ▾
Places ▾
Firefox ESR ▾
Wed 05:12
password wordlist download - Google Search - Mozilla Firefo
password wordlist downl ×
+
https://www.google.com/search?q=password+wordlist+downlioad&ie=utf-8&oe=ut
Google
All
Videos
News
Images
Shopping
More
Settings
Tools

password wordlist download

About 7,48,000 results (0.37 seconds)

github.com › berzerk0 › probable-wordlists ▾

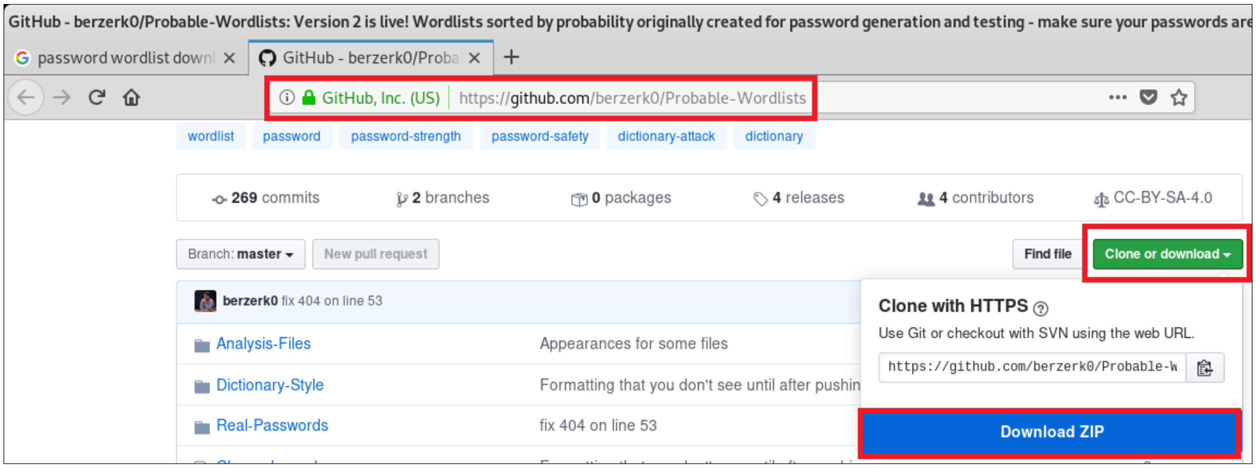
berzerk0/Probable-Wordlists: Version 2 is live ... - GitHub

Wordlists sorted by probability originally created for **password** generation and ... to **download** nearly every single **Wordlist** containing real **passwords** I could find.

Downloads · WPA-Length · Real-Passwords · Dictionary-Style

crackstation.net › crackstation-wordlist-password-cracking-dictionary ▾

Figure 5: Search password wordlist



GitHub - berzerk0/Probable-Wordlists: Version 2 is live! Wordlists sorted by probability originally created for password generation and testing - make sure your passwords are

https://github.com/berzerk0/Probable-Wordlists

berzerk0 fix 404 on line 53

Analysis-Files	Appearances for some files
Dictionary-Style	Formatting that you don't see until after pushin
Real-Passwords	fix 404 on line 53

Clone with HTTPS

Use Git or checkout with SVN using the web URL.

https://github.com/berzerk0/Probable-W

Figure 6: Download password wordlist

<u>Step 5:</u> Save and open the downloaded file as shown in Figure 7. Open the "Real-Passwords" folder to see the passwords wordlist as shown in Figure 8.

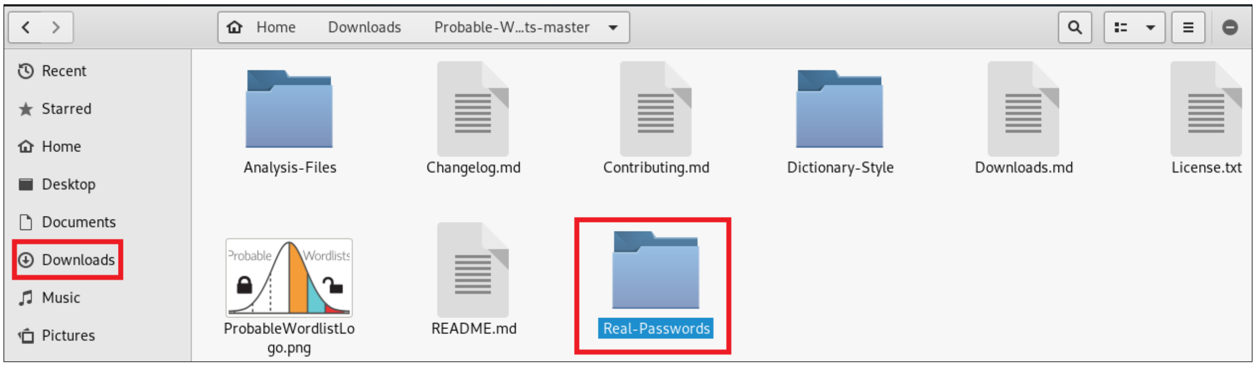
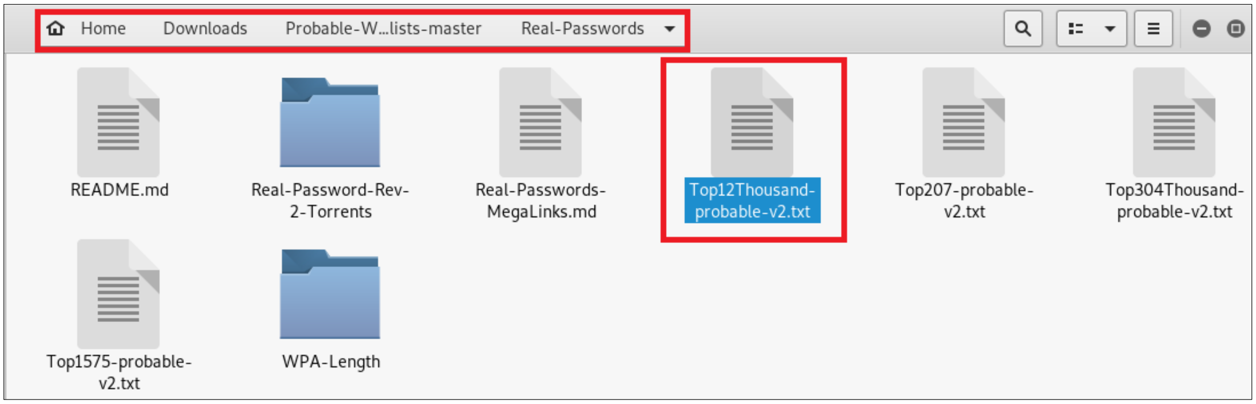


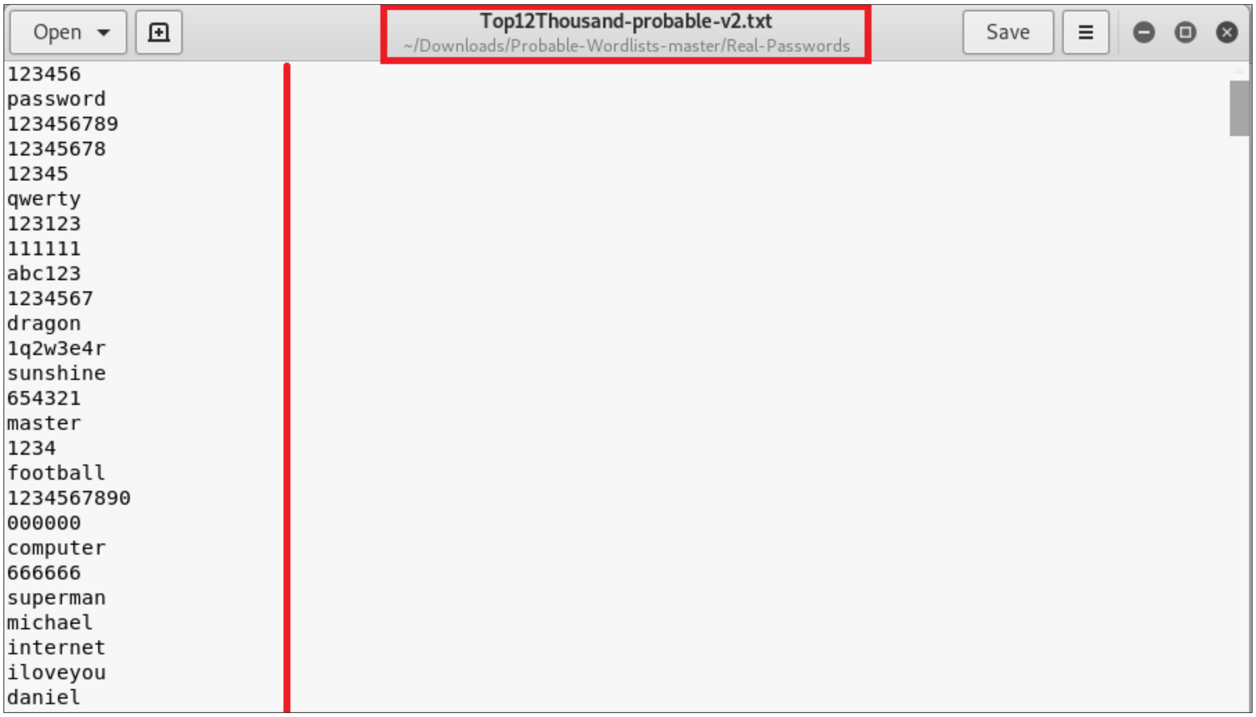
Figure 7: Password folder in downloaded file



Home
Downloads
Probable-W...lists-master
Real-Passwords
README.md
Real-Password-Rev-2-Torrents
Real-Passwords-MegaLinks.md
Top12Thousand-probable-v2.txt
Top207-probable-v2.txt
Top304Thousand-probable-v2.txt
Top1575-probable-v2.txt
WPA-Length

Figure 8: Password wordlist

Step 6: Open any password wordlist (e.g., Top12Thousand-probable-v2.txt file) as shown in Figure 9. Copy this file in Home directory and rename as "wordlist.txt" as shown in Figure 10.



Open
Save
Top12Thousand-probable-v2.txt
~/Downloads/Probable-Wordlists-master/Real-Passwords

```
123456
password
123456789
12345678
12345
qwerty
123123
111111
abc123
1234567
dragon
1q2w3e4r
sunshine
654321
master
1234
football
1234567890
000000
computer
666666
superman
michael
internet
iloveyou
daniel
```

Figure 9: Top 12 thousand most frequently used passwords

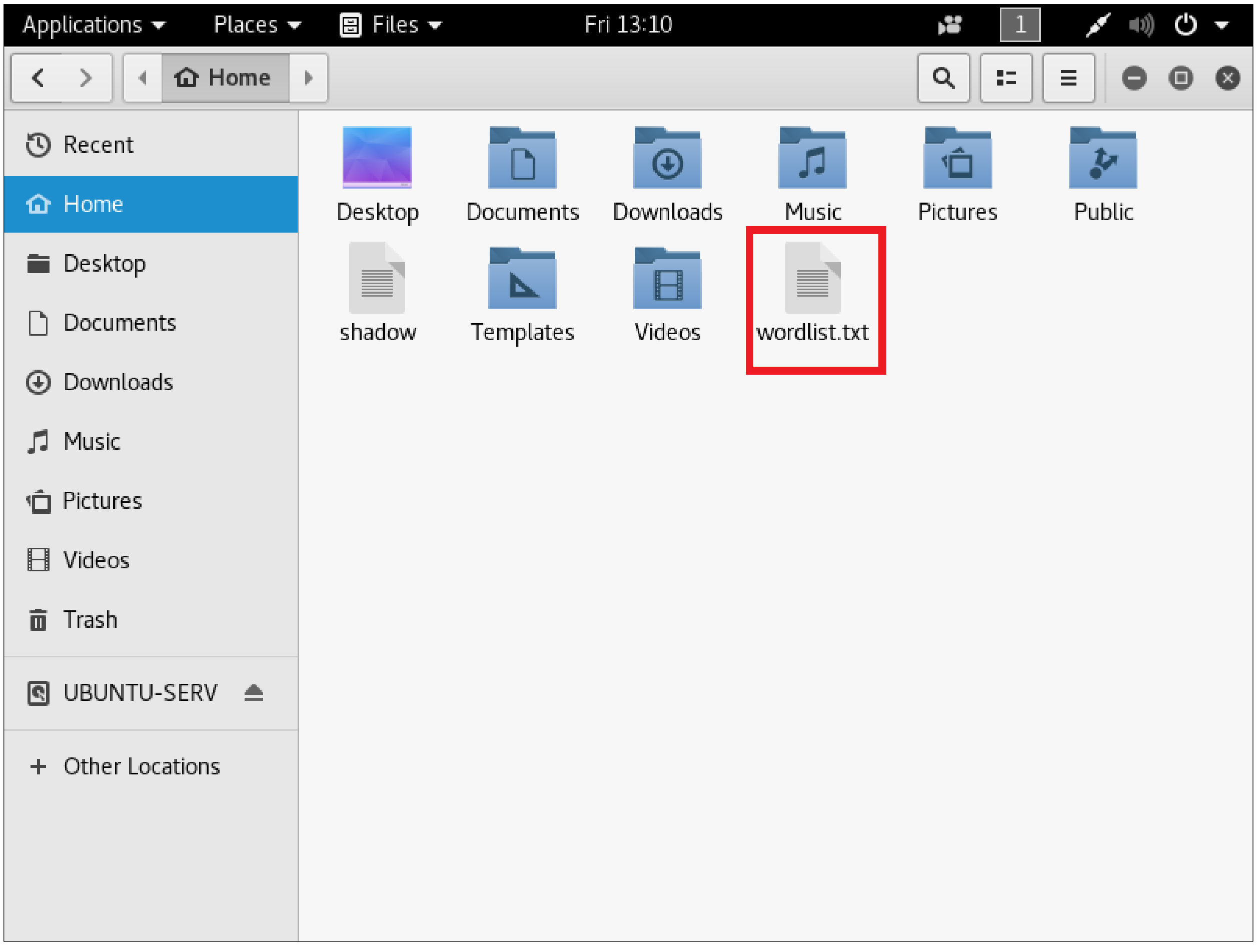
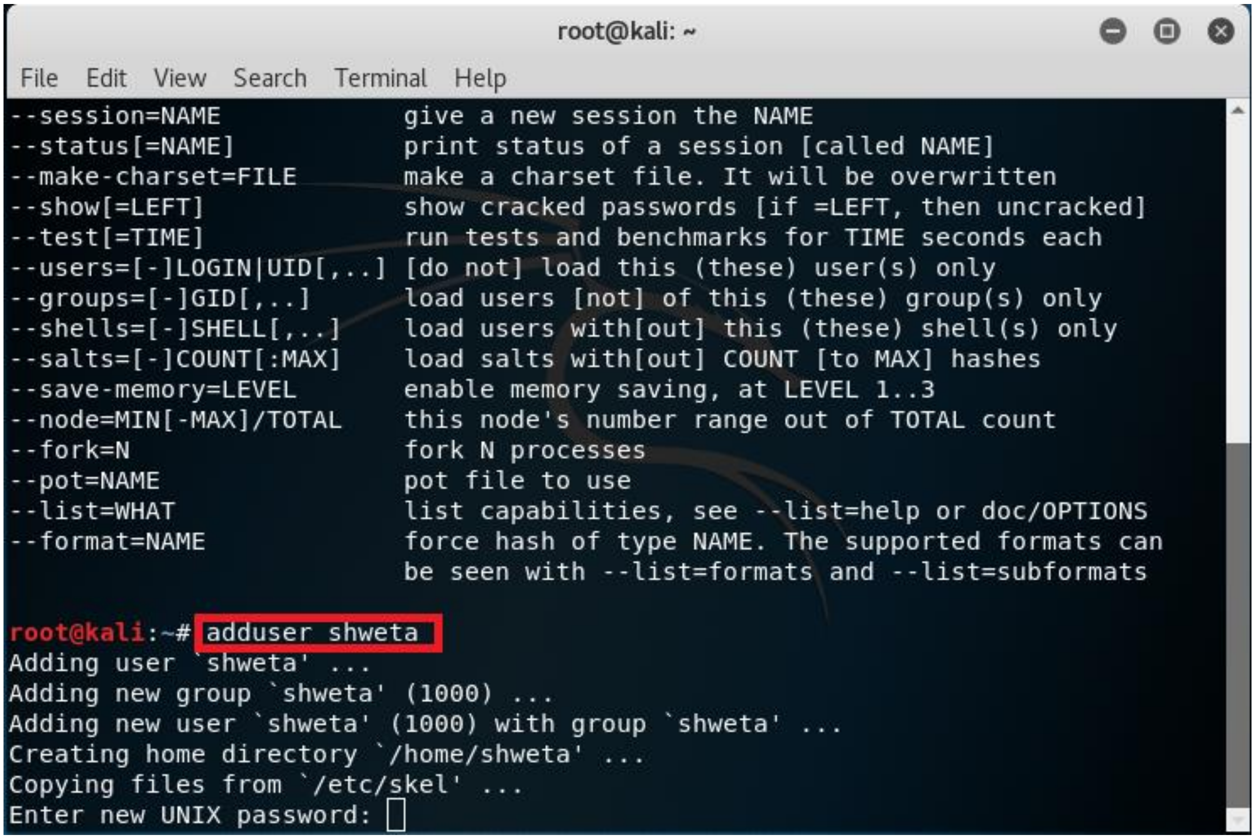


Figure 10: Wordlist file in Home directory

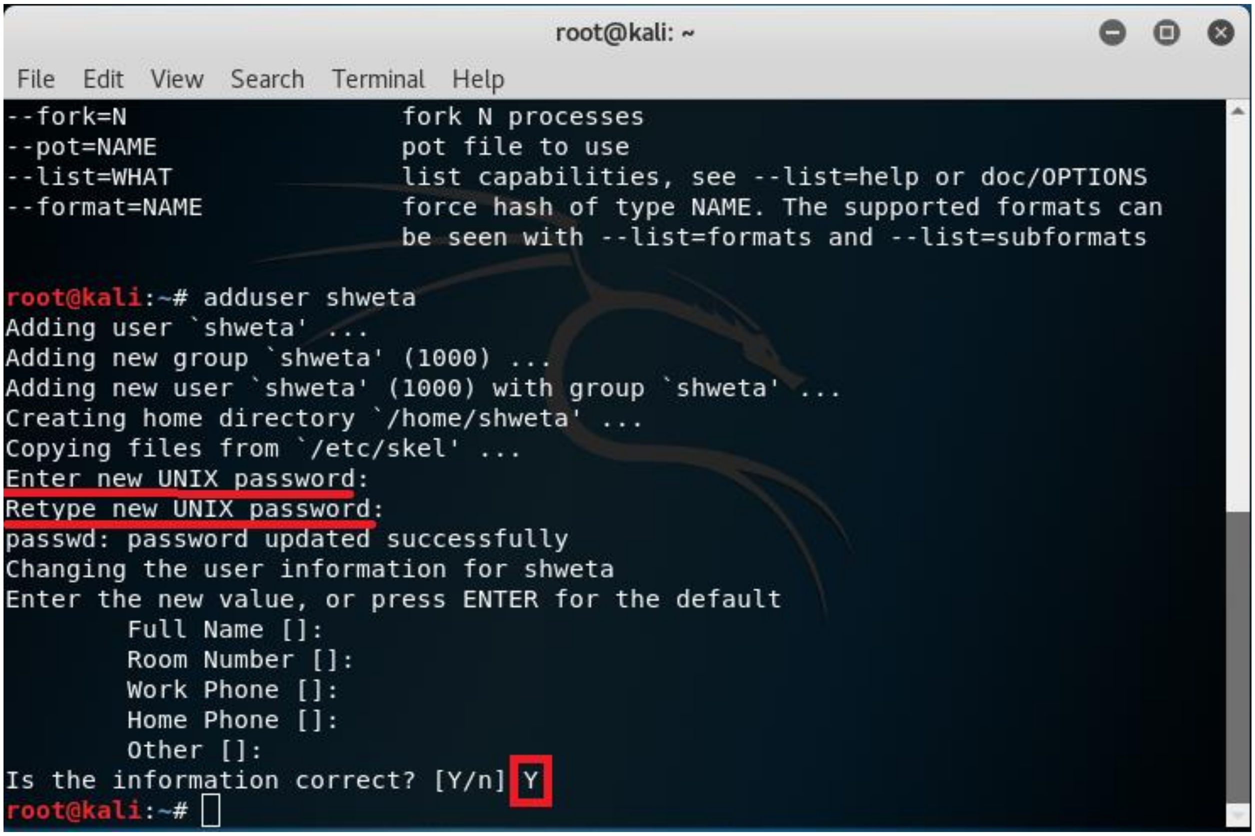
Step 7: Add new users in kali Linux operating system as shown in Figure 11, Figure 12, and Figure 13. Set a password and press 'Y' while creating new users.



```
--session=NAME             give a new session the NAME
--status[=NAME]            print status of a session [called NAME]
--make-charset=FILE        make a charset file. It will be overwritten
--show[=LEFT]              show cracked passwords [if =LEFT, then uncracked]
--test[=TIME]              run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..] [do not] load this (these) user(s) only
--groups=[-]GID[,..]       load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..]     load users with[out] this (these) shell(s) only
--salts=[-]COUNT[:MAX]     load salts with[out] COUNT [to MAX] hashes
--save-memory=LEVEL        enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL     this node's number range out of TOTAL count
--fork=N                   fork N processes
--pot=NAME                 pot file to use
--list=WHAT                list capabilities, see --list=help or doc/OPTIONS
--format=NAME              force hash of type NAME. The supported formats can
                           be seen with --list=formats and --list=subformats

root@kali:~# adduser shweta
Adding user `shweta' ...
Adding new group `shweta' (1000) ...
Adding new user `shweta' (1000) with group `shweta' ...
Creating home directory `/home/shweta' ...
Copying files from `/etc/skel' ...
Enter new UNIX password: 
```

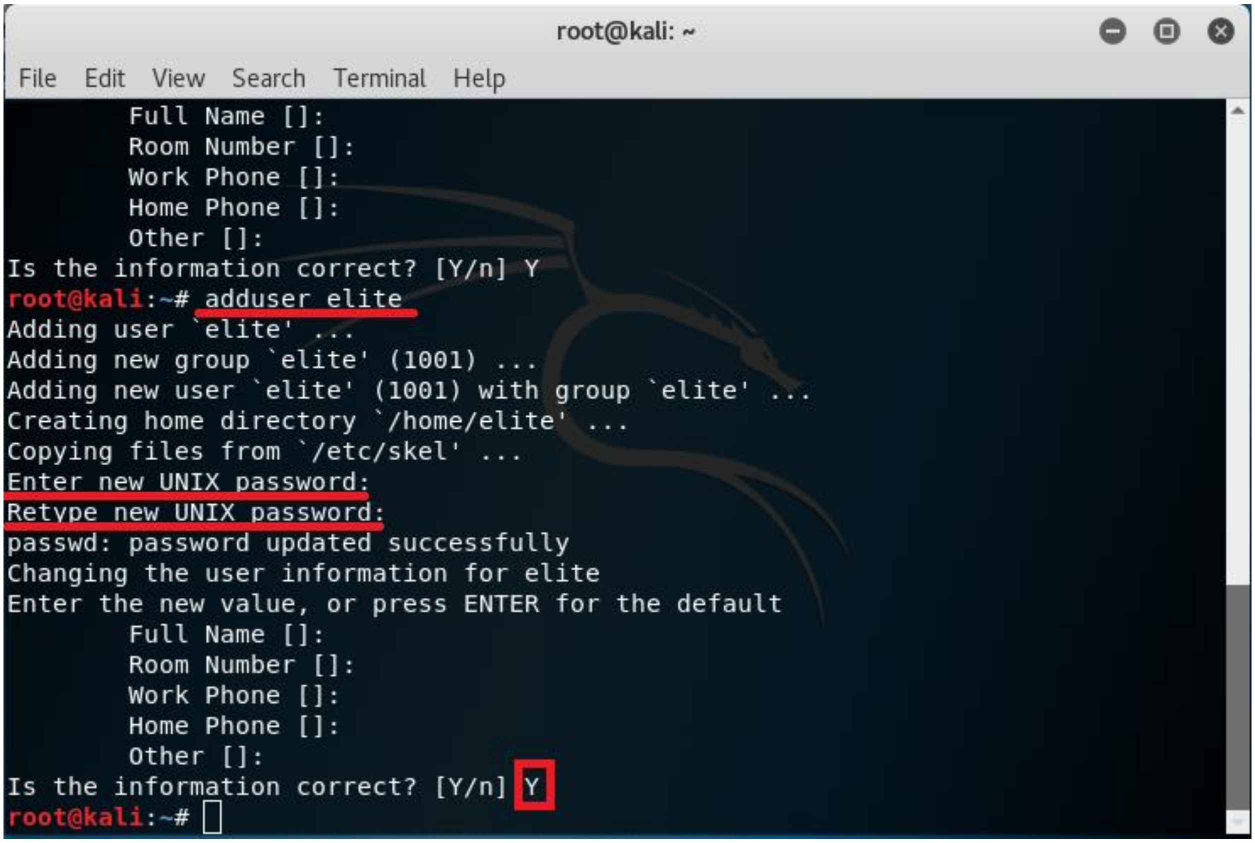
Figure 11: Adding new users in Kali Linux operating system



```
--fork=N                   fork N processes
--pot=NAME                 pot file to use
--list=WHAT                list capabilities, see --list=help or doc/OPTIONS
--format=NAME              force hash of type NAME. The supported formats can
                           be seen with --list=formats and --list=subformats

root@kali:~# adduser shweta
Adding user `shweta' ...
Adding new group `shweta' (1000) ...
Adding new user `shweta' (1000) with group `shweta' ...
Creating home directory `/home/shweta' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for shweta
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
root@kali:~# 
```

Figure 12: Adding new users in Kali Linux operating system



```
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
root@kali:~# adduser elite
Adding user `elite' ...
Adding new group `elite' (1001) ...
Adding new user `elite' (1001) with group `elite' ...
Creating home directory `/home/elite' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for elite
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
root@kali:~# 
```

Figure 13: Adding new users in Kali Linux operating system

<u>Step 8:</u> Go to Other Locations->Computer->etc folder to find the shadow file as shown in Figure 14, Figure 15, and Figure 16.

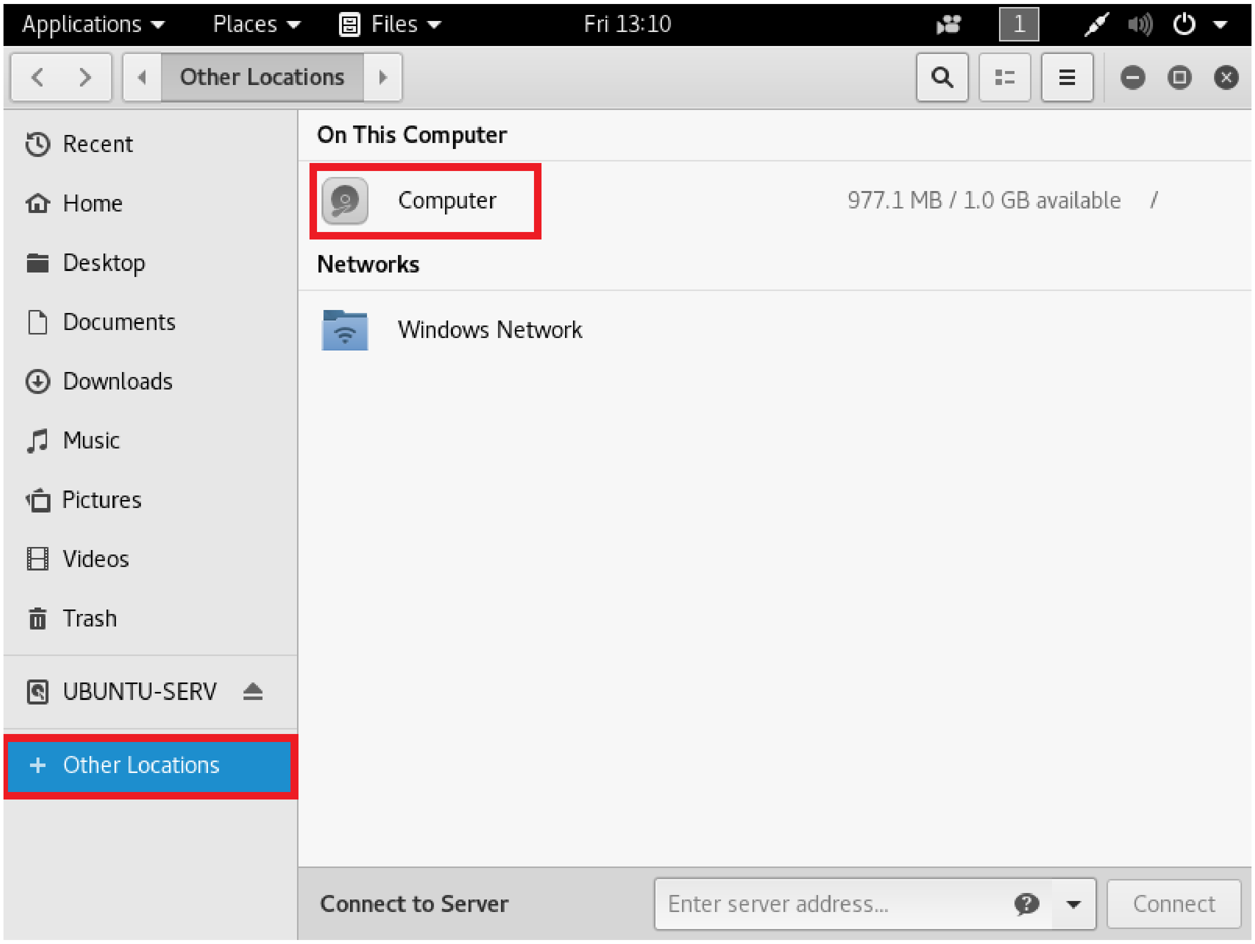


Figure 14: Opening other locations in Kali Linux operating system

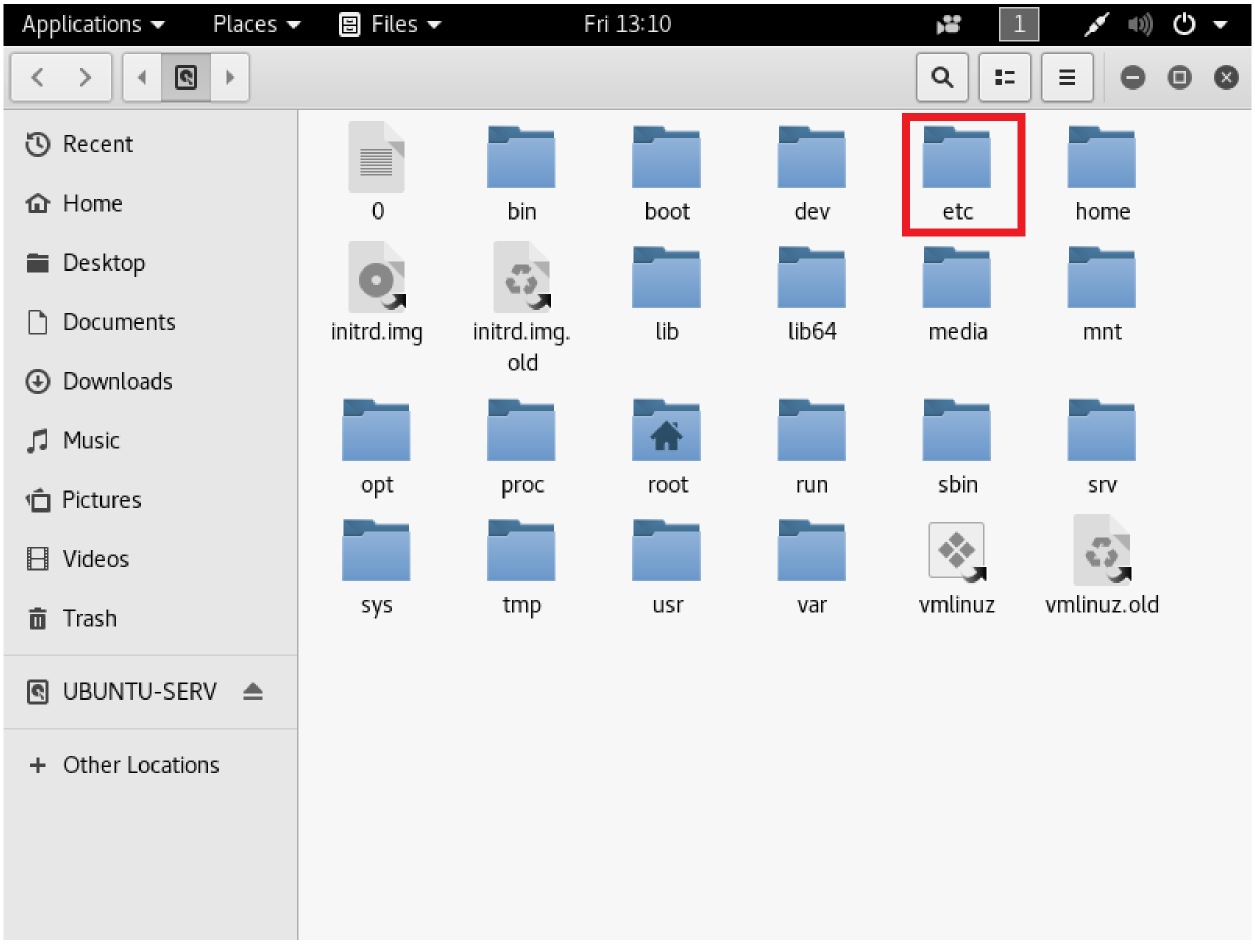
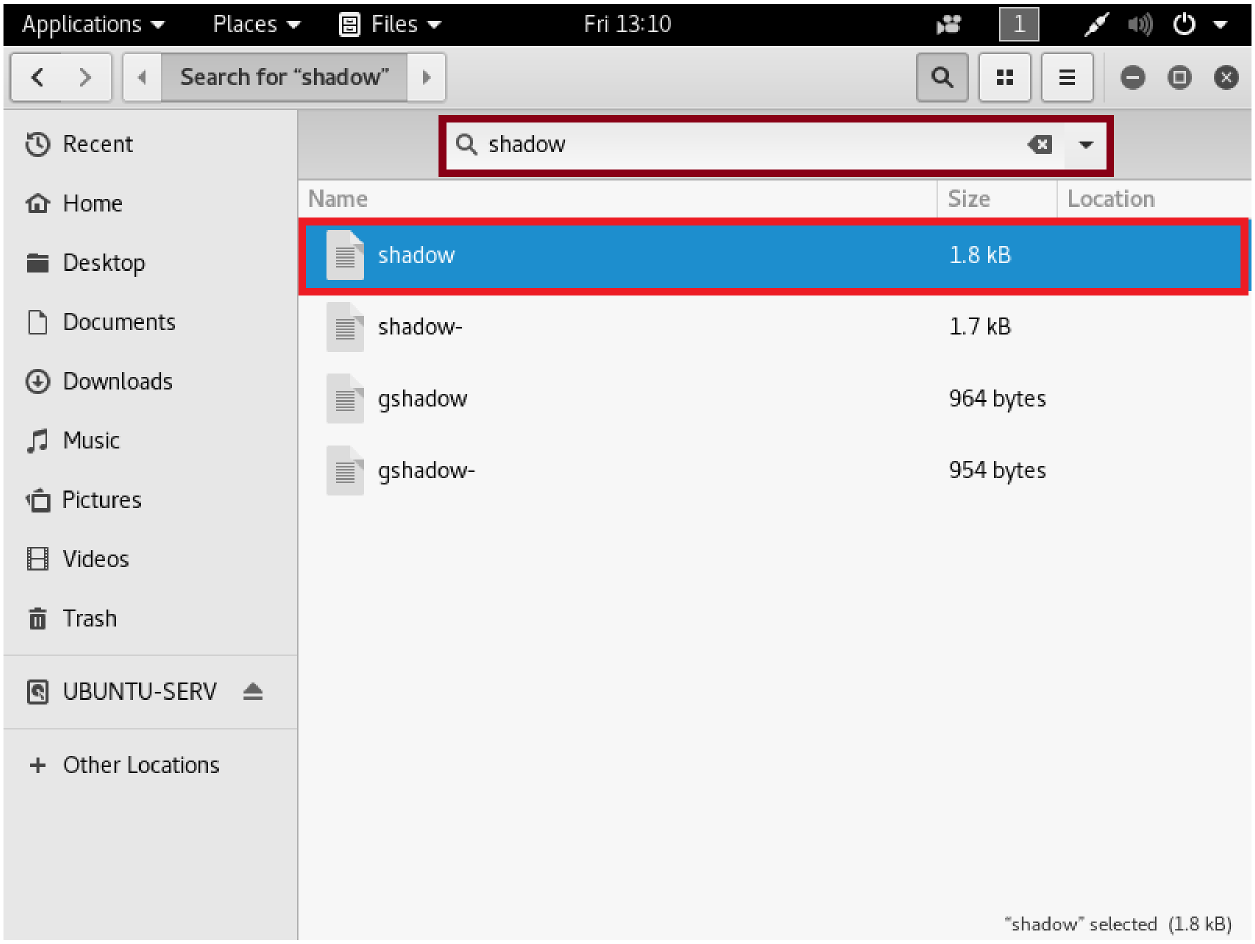


Figure 15: Opening etc folder in Kali Linux operating system



shadow

Name	Size	Location
shadow	1.8 kB	
shadow-	1.7 kB	
gshadow	964 bytes	
gshadow-	954 bytes	

"shadow" selected (1.8 kB)

Figure 16: Finding Shadow file

Step 9: Copy the shadow file and paste in Home directory as shown in Figure 17.

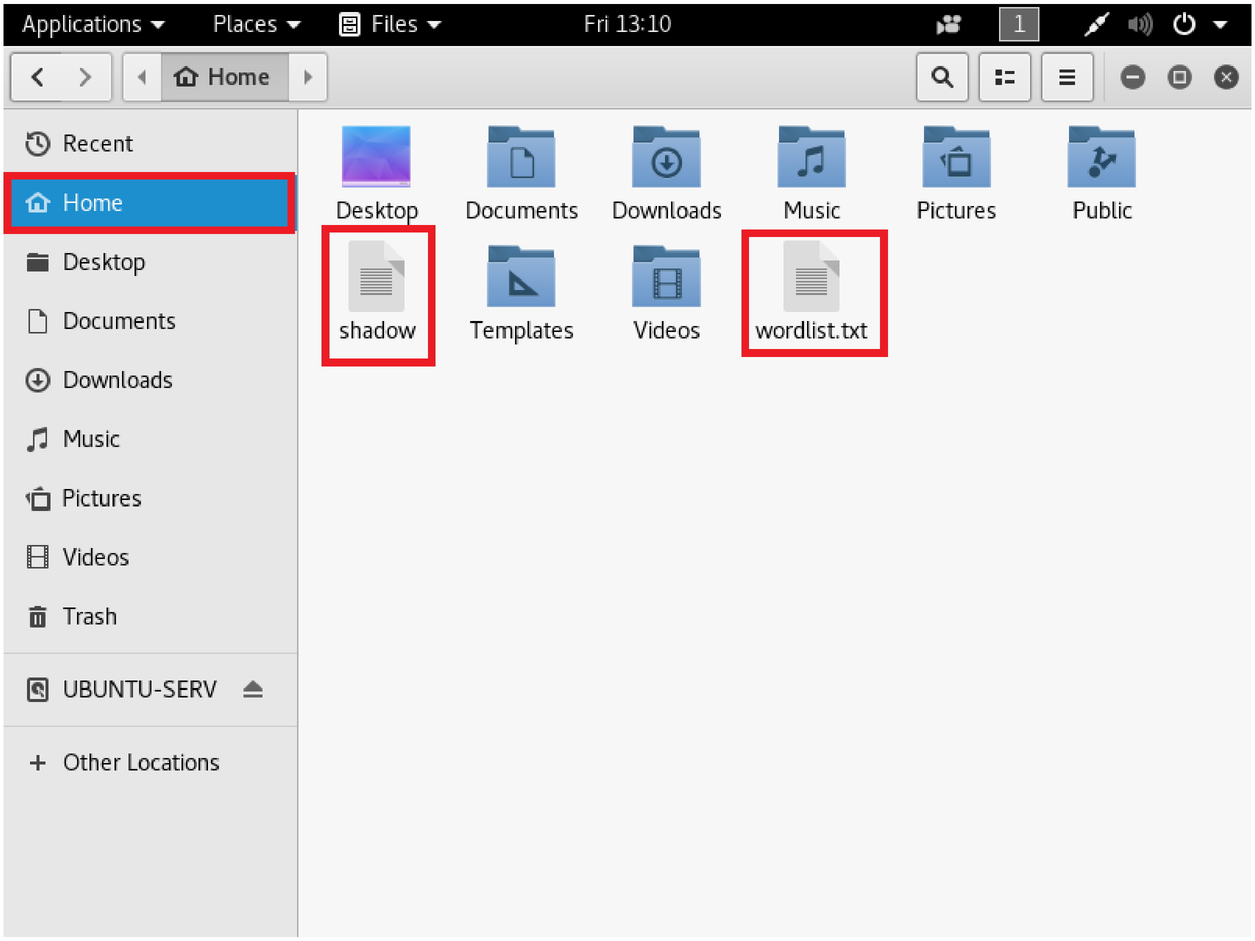
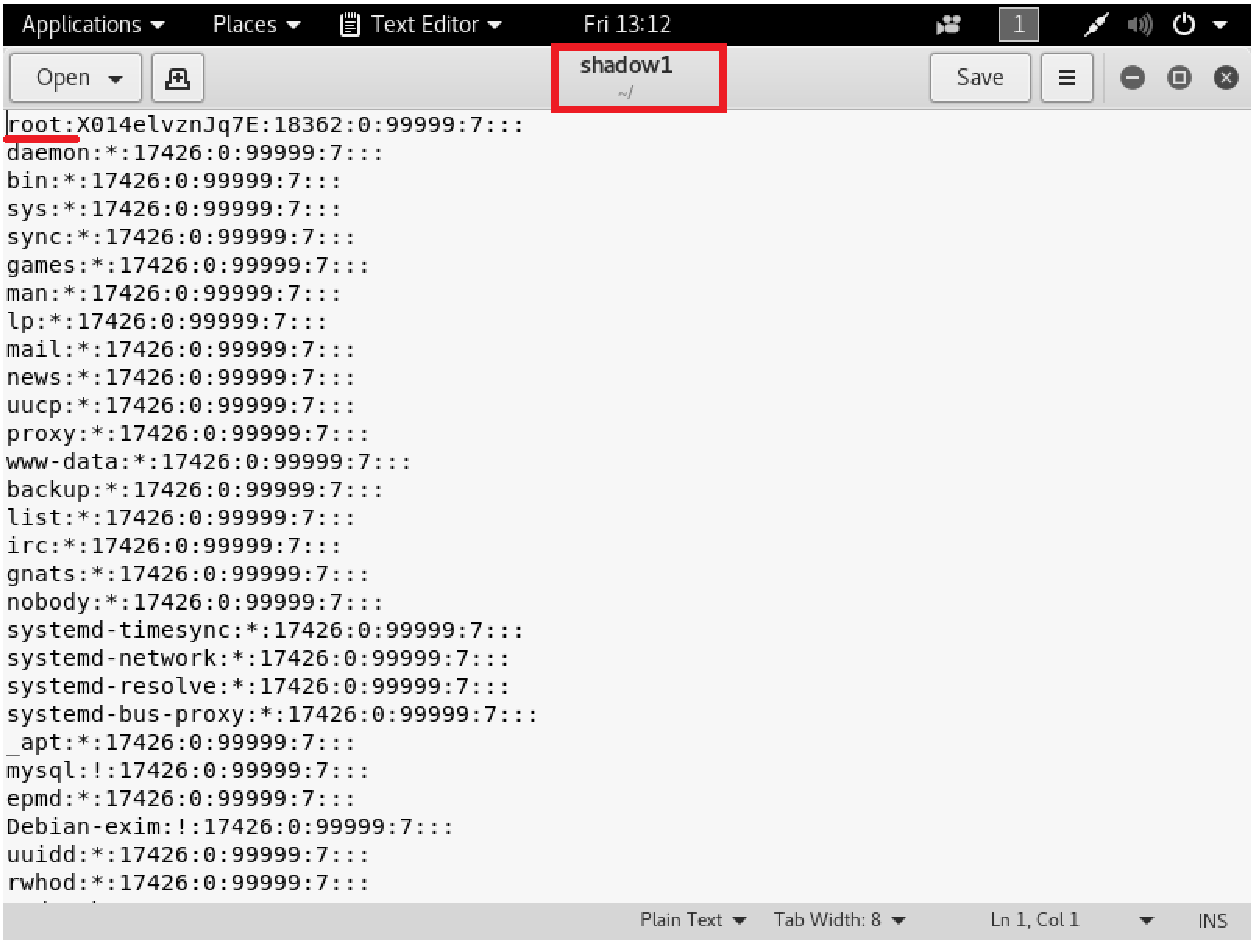


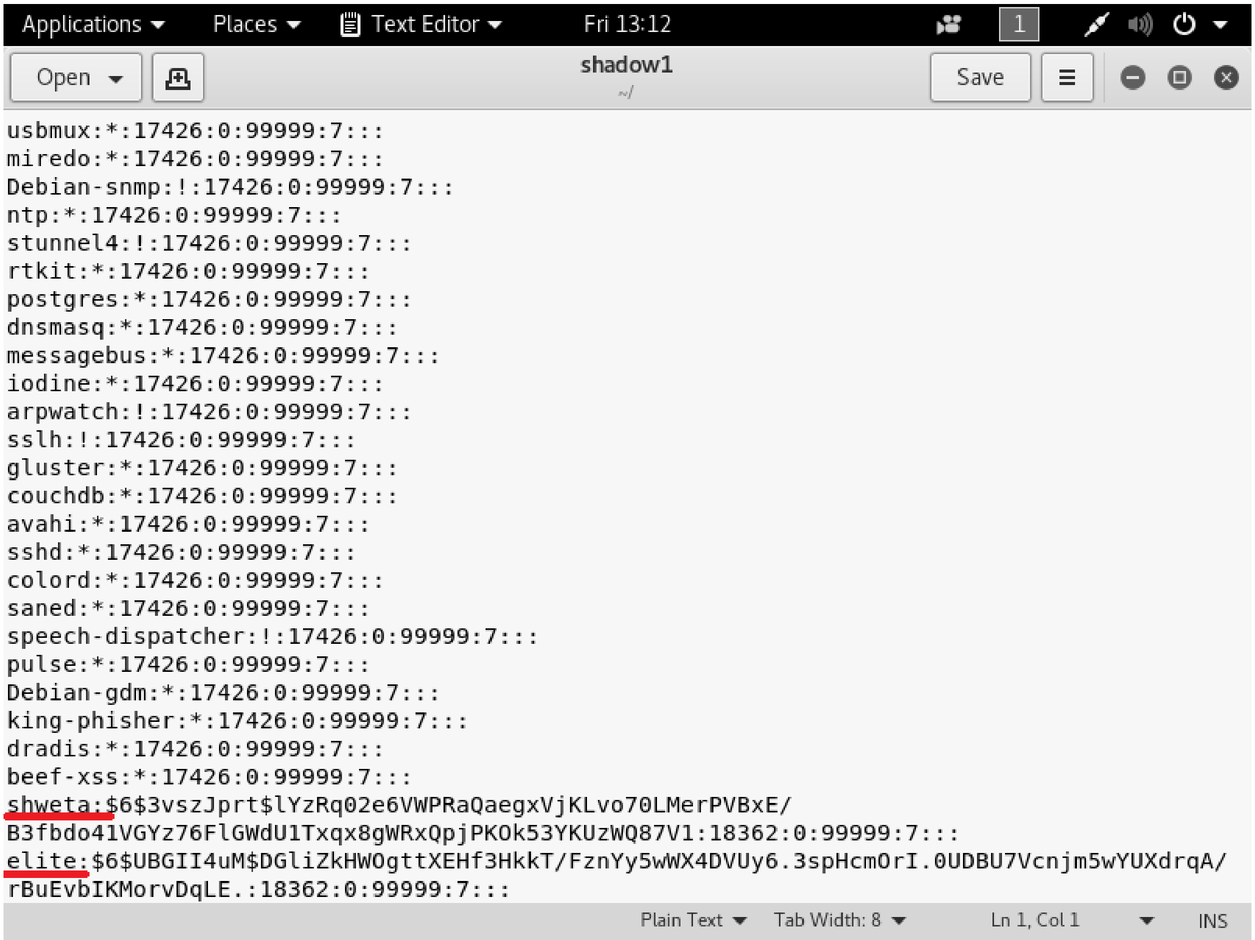
Figure 17: Copy of Shadow file and Wordlist in Home directory

Step 9: Rename the shadow file as shadow1 and open the file to find the usernames and password in the form of hash values as shown in Figure 18 and Figure 19.



```
root:X014elvznJq7E:18362:0:99999:7:::
daemon:*:17426:0:99999:7:::
bin:*:17426:0:99999:7:::
sys:*:17426:0:99999:7:::
sync:*:17426:0:99999:7:::
games:*:17426:0:99999:7:::
man:*:17426:0:99999:7:::
lp:*:17426:0:99999:7:::
mail:*:17426:0:99999:7:::
news:*:17426:0:99999:7:::
uucp:*:17426:0:99999:7:::
proxy:*:17426:0:99999:7:::
www-data:*:17426:0:99999:7:::
backup:*:17426:0:99999:7:::
list:*:17426:0:99999:7:::
irc:*:17426:0:99999:7:::
gnats:*:17426:0:99999:7:::
nobody:*:17426:0:99999:7:::
systemd-timesync:*:17426:0:99999:7:::
systemd-network:*:17426:0:99999:7:::
systemd-resolve:*:17426:0:99999:7:::
systemd-bus-proxy:*:17426:0:99999:7:::
_apt:*:17426:0:99999:7:::
mysql:!:17426:0:99999:7:::
epmd:*:17426:0:99999:7:::
Debian-exim:!:17426:0:99999:7:::
uuidd:*:17426:0:99999:7:::
rwhod:*:17426:0:99999:7:::
```

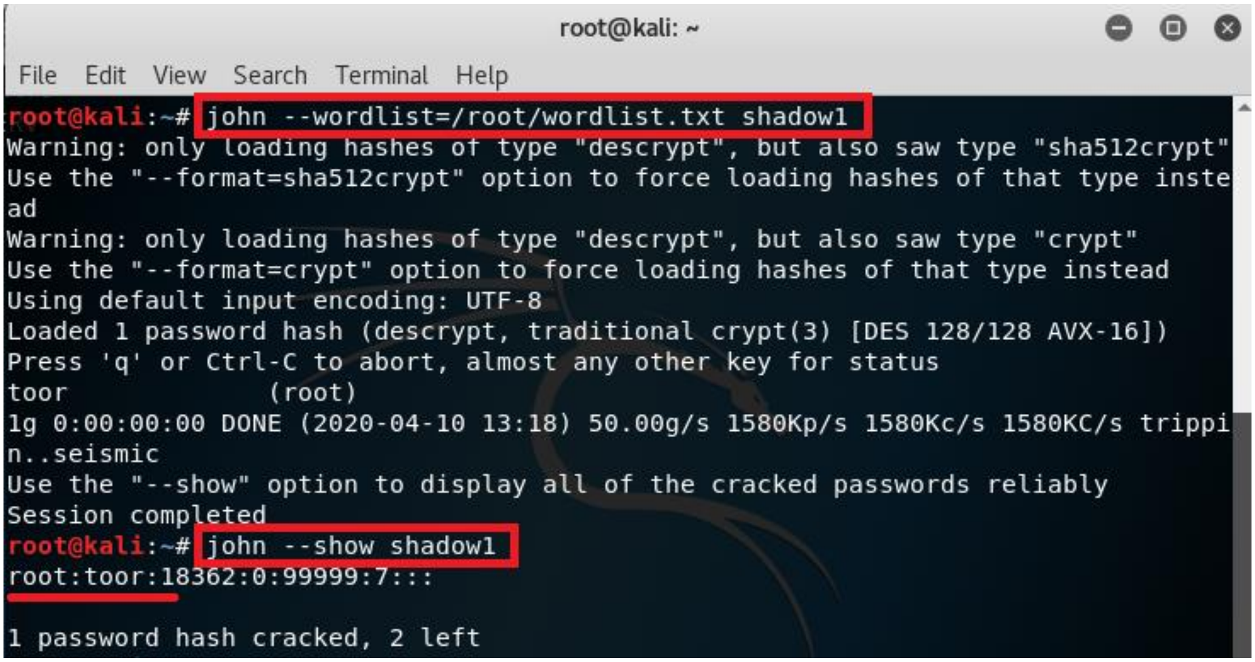
Figure 18: Opening Shadow file



```
usbmux:*:17426:0:99999:7:::
miredo:*:17426:0:99999:7:::
Debian-snmp:!:17426:0:99999:7:::
ntp:*:17426:0:99999:7:::
stunnel4:!:17426:0:99999:7:::
rtkit:*:17426:0:99999:7:::
postgres:*:17426:0:99999:7:::
dnsmasq:*:17426:0:99999:7:::
messagebus:*:17426:0:99999:7:::
iodine:*:17426:0:99999:7:::
arpwatch:!:17426:0:99999:7:::
sslh:!:17426:0:99999:7:::
gluster:*:17426:0:99999:7:::
couchdb:*:17426:0:99999:7:::
avahi:*:17426:0:99999:7:::
sshd:*:17426:0:99999:7:::
colord:*:17426:0:99999:7:::
saned:*:17426:0:99999:7:::
speech-dispatcher:!:17426:0:99999:7:::
pulse:*:17426:0:99999:7:::
Debian-gdm:*:17426:0:99999:7:::
king-phisher:*:17426:0:99999:7:::
dradis:*:17426:0:99999:7:::
beef-xss:*:17426:0:99999:7:::
shweta:$6$3vszJprt$lYzRq02e6VWPRaQaegxVjKLvo70LMerPVBxE/
B3fbdo41VGYz76FlGWdU1Txqx8gWRxQpjPKOk53YKUzWQ87V1:18362:0:99999:7:::
elite:$6$UBGII4uM$DGliZkHWOgttXEHf3HkkT/FznYy5wWX4DVUy6.3spHcmOrI.0UDBU7Vcnjm5wYUXdrqA/
rBuEvbIKMorvDqLE.:18362:0:99999:7:::
```

Figure 19: Opening Shadow file

Step 10: Write the command “john --wordlist=/root/wordlist.txt” to recover the hash of root and “*john --show shadow1*” to display the passwords in plaintext as shown in Figure 20.



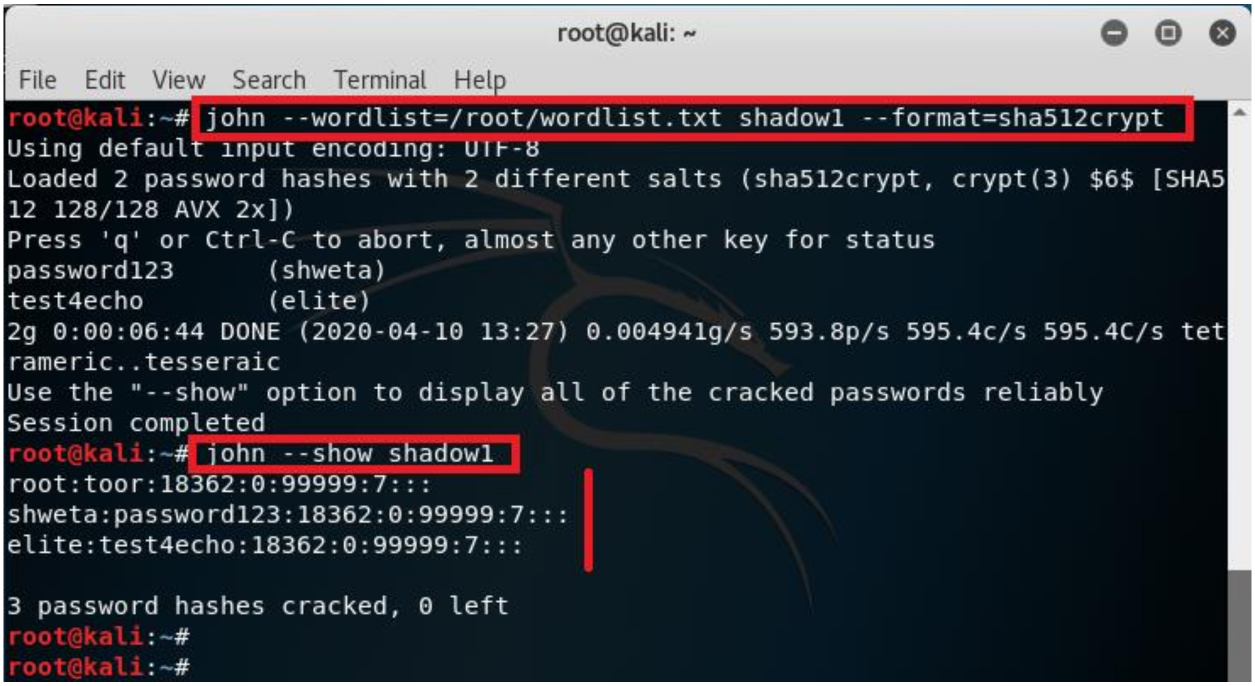
```
root@kali:~# john --wordlist=/root/wordlist.txt shadow1
Warning: only loading hashes of type "descrypt", but also saw type "sha512crypt"
Use the "--format=sha512crypt" option to force loading hashes of that type inste
ad
Warning: only loading hashes of type "descrypt", but also saw type "crypt"
Use the "--format=crypt" option to force loading hashes of that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (descrypt, traditional crypt(3) [DES 128/128 AVX-16])
Press 'q' or Ctrl-C to abort, almost any other key for status
toor             (root)
1g 0:00:00:00 DONE (2020-04-10 13:18) 50.00g/s 1580Kp/s 1580Kc/s 1580KC/s trippi
n..seismic
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~# john --show shadow1
root:toor:18362:0:99999:7:::

1 password hash cracked, 2 left
```

Figure 20: Cracking password of Root

<u>Step 10:</u> Write the command "john --wordlist= /root/wordlist.txt --format=sha512crypt" to recover the hash of other users and "*john --show shadow1*" to display the passwords in plaintext as shown in Figure 21.

The passwords in plaintext are displayed in the Figure 21 and highlighted in red rectangular box.



```
root@kali:~# john --wordlist=/root/wordlist.txt shadow1 --format=sha512crypt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA5
12 128/128 AVX 2x])
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (shweta)
test4echo        (elite)
2g 0:00:06:44 DONE (2020-04-10 13:27) 0.004941g/s 593.8p/s 595.4c/s 595.4C/s tet
rameric..tesseraic
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~# john --show shadow1
root:toor:18362:0:99999:7:::
shweta:password123:18362:0:99999:7:::
elite:test4echo:18362:0:99999:7:::

3 password hashes cracked, 0 left
root@kali:~#
root@kali:~#
```

Figure 21: Cracking password of other users

COUNTERMEASURES

The following countermeasures must be followed:

- **Strong Passwords:** Establish strong password using special characters, numbers, and lower and upper case alphabets.
- **Minimum Password Length:** The length of the password should be set to at least 14 characters. The long passwords are harder to crack than the short ones.
- **Dictionary words:** Do not use dictionary words such as password, qwerty, abc123, etc. These passwords can be cracked easily with tools. Do not rely on similar looking characters such as: 3 → E , 5 →S , ! → 1. These words are also stored in dictionary.

- **Minimum Password age:** The users must change the password after some time (90 days). This will reduce the risk of password cracking.
- **Stronger authentication method:** Use stronger authentication methods such as enable Gmail one time password feature to login in a new device.
- **Different passwords:** Use different passwords for different device or websites.
- **Sharing passwords:** Do not share passwords with anyone or change password immediately after usage, if shared.
- **Storing passwords:** Avoid storing passwords in an unsecured location such as desktop or mobile phones. An attacker can access those passwords by hacking the device. Try to remember the passwords.
- **Personal Information:** Do not use personal information such as date of birth, pet names, vehicle number, etc. An attacker can easily guess the password by knowing personal details through social engineering.

REFERENCES

[1] O. S. Limited, “Official Kali Linux Releases,” 2020. https://www.kali.org/kali-linux-releases/ (accessed Apr. 15, 2020).

[2] O. S. Limited, “john Package Description,” 2020. https://tools.kali.org/password-attacks/john (accessed May 20, 2020).