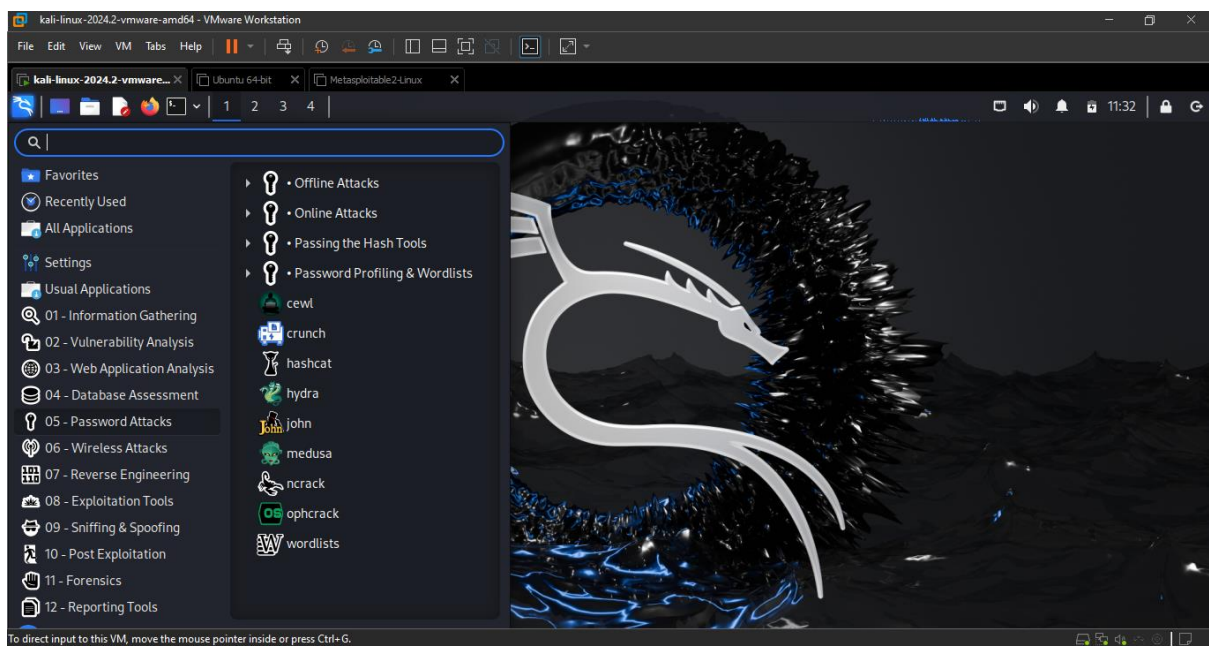


# John The Ripper

- The John-the-ripper tool 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.

## Steps:

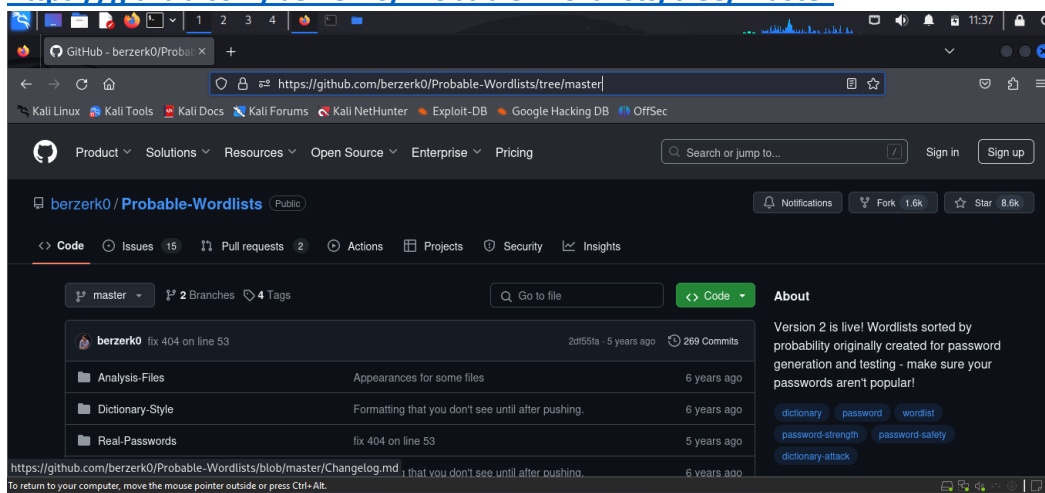
**Step 1:** Open Kali Linux operating system → Go to Applications → Password attacks → John.

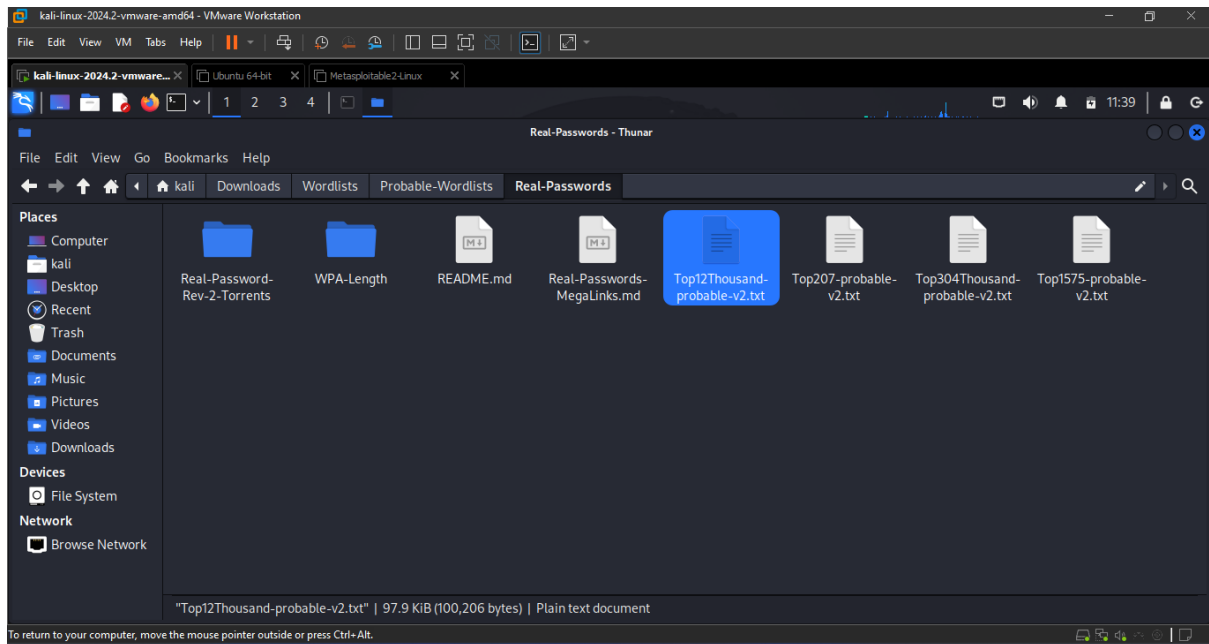


## Step 2:

Download the Wordlists from the below Website.

<https://github.com/berzerk0/Probable-Wordlists/tree/master>



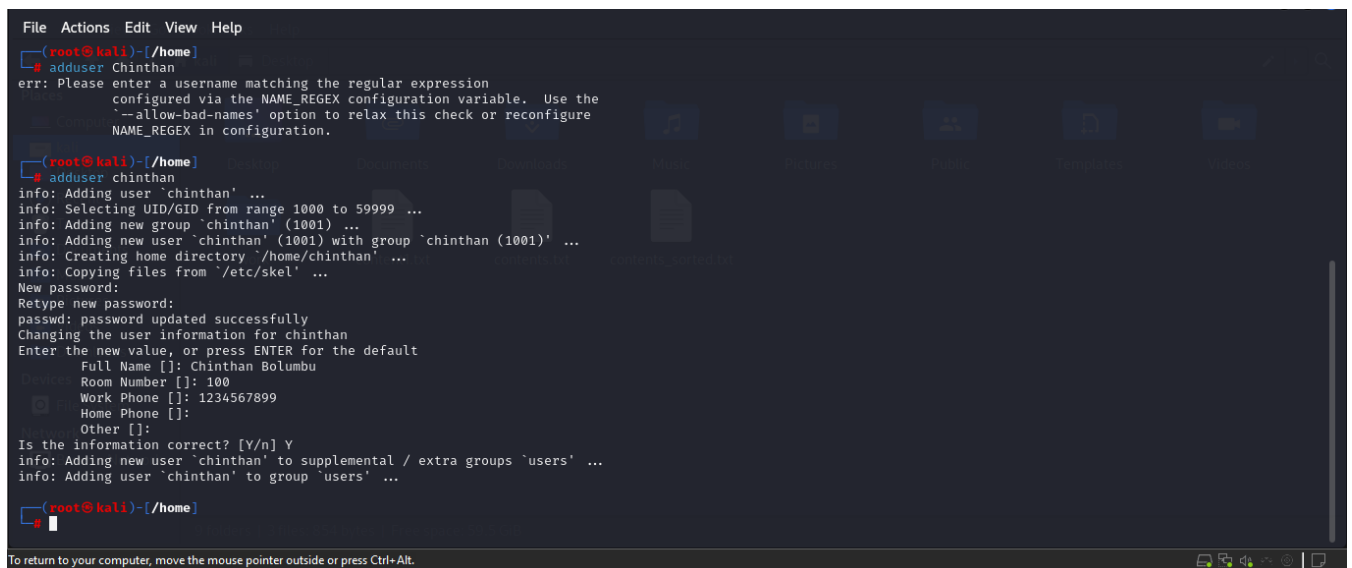


### Step 3:

Go to root-kali by `sudo su` and then store file 'wordlist.txt' there.

Copy Wordlists to Home Directory and start with `adduser` command in terminal.

Add new users in kali Linux operating system, set a password and press 'Y' while creating new users.

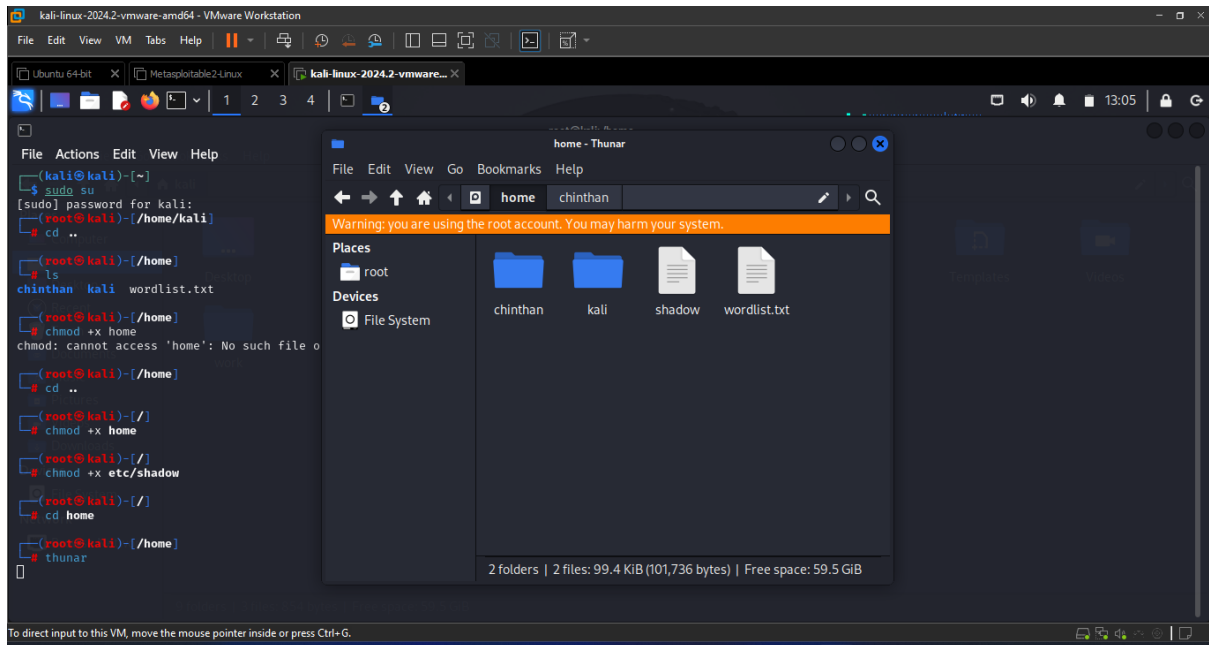


### Step 4:

Command `Thunar` to access home Directory.

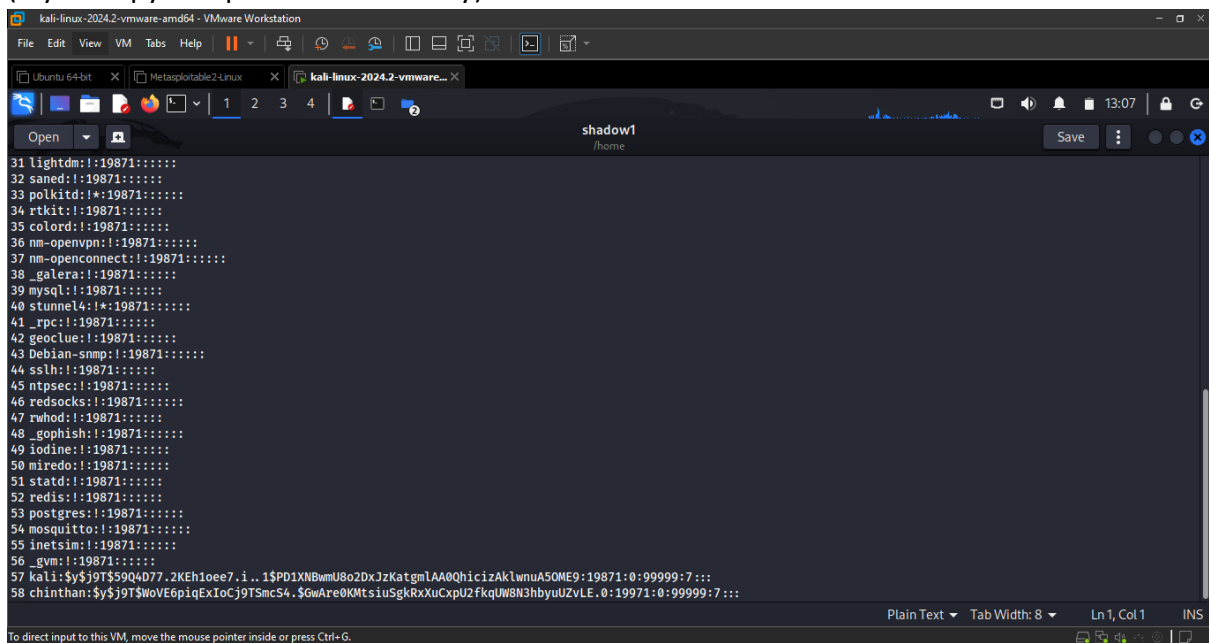
Copy the shadow file from file system and paste it to Home directory. OR

Command ``cp /etc/shadow ./shadow1.txt`` (Ignore Step 5)



### Step 5:

Rename Shadow file to Shadow1 and open the file and search for username and password.  
(if you Copy and paste in a directory)



### Step 6:

Command ``john -format=crypt pass.txt`` to start cracking.

```

(root@kali)-[/home/kali]
└─$ john --format=crypt pass.txt

Using default input encoding: UTF-8
Loaded 3 password hashes with 3 different salts (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descript 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0 for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
test02      (test02)
kali        (kali)
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
2g 0:00:02:57 17.21% 2/3 (ETA: 14:02:28) 0.01128g/s 169.3p/s 169.3c/s 169.3C/s NINA..PETER
2g 0:00:03:08 18.40% 2/3 (ETA: 14:02:21) 0.01061g/s 171.0p/s 171.0c/s 171.0C/s little2..patches2
2g 0:00:03:10 18.63% 2/3 (ETA: 14:02:19) 0.01051g/s 171.3p/s 171.3c/s 171.3C/s shawn2..sebastian2
2g 0:00:03:12 18.87% 2/3 (ETA: 14:02:17) 0.01040g/s 171.5p/s 171.5c/s 171.5C/s cassandra2..zeus2
2g 0:00:03:14 19.08% 2/3 (ETA: 14:02:17) 0.01029g/s 171.8p/s 171.8c/s 171.8C/s cunt2..kids2
2g 0:00:03:16 19.28% 2/3 (ETA: 14:02:16) 0.01019g/s 172.1p/s 172.1c/s 172.1C/s webster2..hal2
2g 0:00:03:17 19.43% 2/3 (ETA: 14:02:13) 0.01014g/s 172.2p/s 172.2c/s 172.2C/s dorothy!..emily!

```

- If the password set is easy then tool (John the ripper) will fetch the password from wordlist quickly.
- If the password set is slightly difficult then tool (John the ripper) will fetch the password from wordlist by checking each and every combinations.
- The orange coloured listed above is the username and the password associated, where left side is the password and right side is the username.
- Since there were 3 users set, 2 users had simple password and one user had difficult password where difficult password is taking time.
- The processes that are scheduled after the Orange colored text is that, it is trying to crack the password from predefined wordlist in every combinations taking more time.
- If the passwords are there in wordlists matching with users password then, passwords will be revealed.