

Beginners Guide for John the Ripper (Part 2)

June 9, 2018 By Raj Chandel

We learned most of the basic information on John the Ripper in our Previous Article which can be found [here](#). In this article, we will use John the Ripper to crack the password hashes of some of the file formats like zip, rar, pdf and much more.

To crack these password hashes, we are going to use some of the inbuilt and some other utilities which extract the password hash from the locked file. There are some utilities that come inbuilt with John which can be found using the following command.

```
locate *2john
```

As you can see that we have the following utilities, we will demonstrate some of them here.



```
root@kali:~# locate *2john
/usr/sbin/dm2john
/usr/sbin/gpg2john
/usr/sbin/hccap2john
/usr/sbin/keepass2john
/usr/sbin/keychain2john
/usr/sbin/keyring2john
/usr/sbin/kwallet2john
/usr/sbin/pfx2john
/usr/sbin/putty2john
/usr/sbin/pwsafe2john
/usr/sbin/racf2john
/usr/sbin/rar2john
/usr/sbin/ssh2john
/usr/sbin/zip2john
```

Cracking the SSH Password Hash

John the Ripper can crack the SSH private key which is created in RSA Encryption. To test the cracking of the private key, first, we will have to create a set of new private keys. To do this we will use a utility that comes with ssh, called “ssh-keygen”.

```
ssh-keygen
```

```

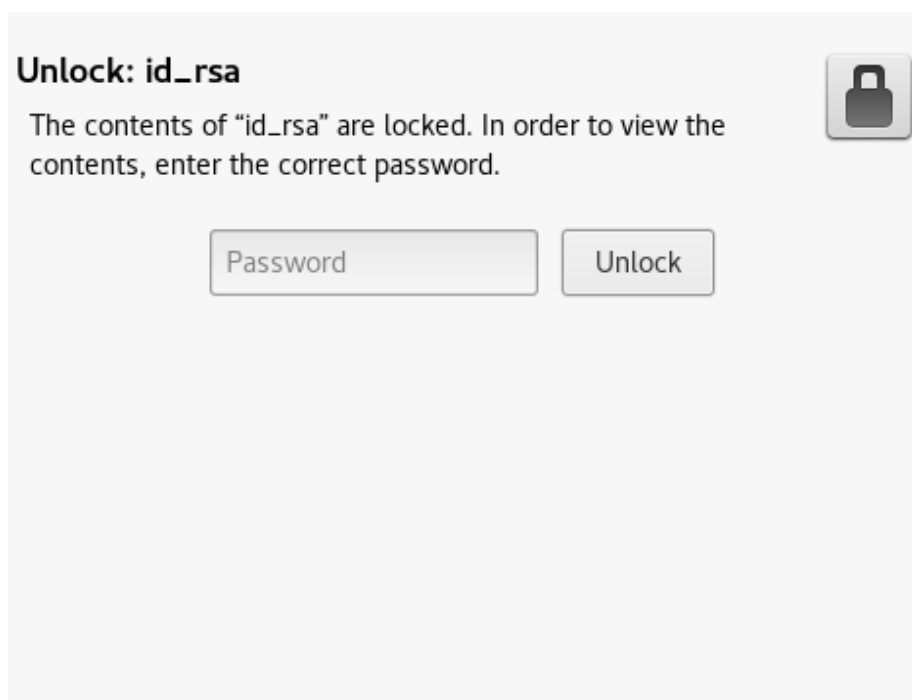
pavan@kali:~$ ssh-keygen ↵
Generating public/private rsa key pair.
Enter file in which to save the key (/home/pavan/.ssh/id_rsa):
Created directory '/home/pavan/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/pavan/.ssh/id_rsa.
Your public key has been saved in /home/pavan/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:dM3MSZNJPvG+YcrGSSzBnxXM61jQBbPv3VnU5GqFYLw pavan@kali
The key's randomart image is:
+---[RSA 2048]---+
|      oB*+oo|
|      . 0=*+B.|
|      . + 0o=.|
|      . . +E=o|
|      S . =+* o|
|      =.=.+|=
|      *  ..+|
|      .|
+-----[SHA256]-----+

```

After opening, it asks for the location at which we want the public/private RSA key pair to store? You can use any location or you can leave it as default.

After that it asks for the passphrase, after entering the password again, we successfully generate the RSA private key. (Refer the image)

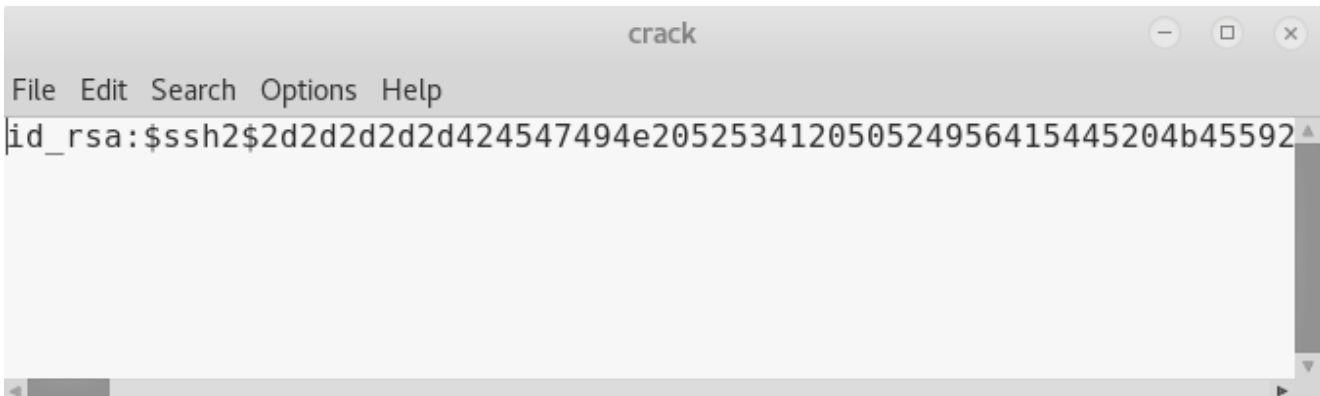
When you will try to open the file, you will be greeted by the following prompt.



Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called “ssh2john”.

Syntax: ssh2john [location of key]

```
ssh2john /home/pavan/.ssh/id_rsa > crack.txt
```



You can see that we converted the key to a crackable hash and then entered it into a text file named id_rsa.txt.

Now let's use John the Ripper to crack this hash.

```
john --wordlist=/usr/share/wordlists/rockyou.txt id_rsa.txt
```

Great! We have successfully cracked the passphrase used to create the private ssh key to be "password123"

```
pavan@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt id_rsa.txt
Created directory: /home/pavan/.john
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA 32/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
password123 (id rsa)
1g 0:00:00:00 DONE (2018-06-06 20:47) 3.448g/s 4772p/s 4772c/s
4772C/s password123
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

Cracking the KeepPass2 Password Hash

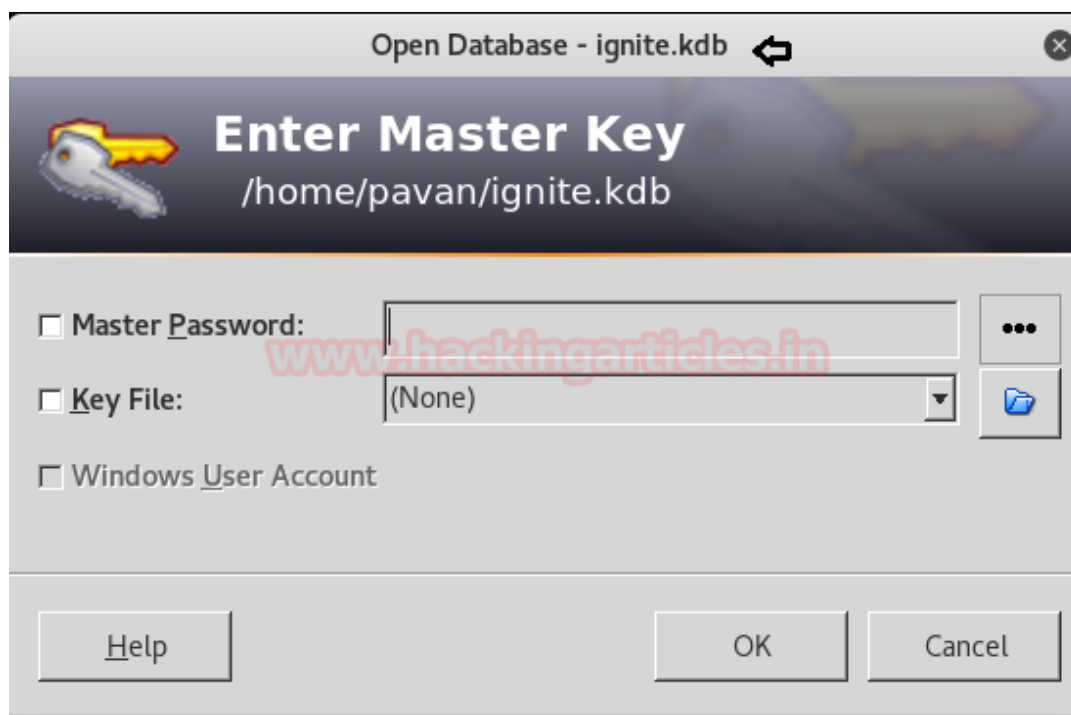
John the Ripper can crack the KeepPass2 key. To test the cracking of the key, first, we will have to create a set of new keys. To do this we will use a utility that is called "kpcli".

```
kpcli
```

```
pavan@kali:~$ kpcli ↵  
  
KeePass CLI (kpcli) v3.1 is ready for operation.  
Type 'help' for a description of available commands.  
Type 'help <command>' for details on individual commands.  
  
kpcli:/> saveas ignite.kdb  
Please provide the master password: *****  
Retype to verify: *****  
kpcli:/> exit
```

Now we will create a database file using the command “save as” and naming the database file as ignite.kdb and entering a passcode to secure it.

When you will try to open the file, you will be greeted by the following prompt.



Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called “keepass2john”.

Syntax: keepass2john [location of key]

```
keepass2john ignite.kdb > crack.txt
```



Now let's use John the Ripper to crack this hash.

```
john --wordlist=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the key to be "12345678"

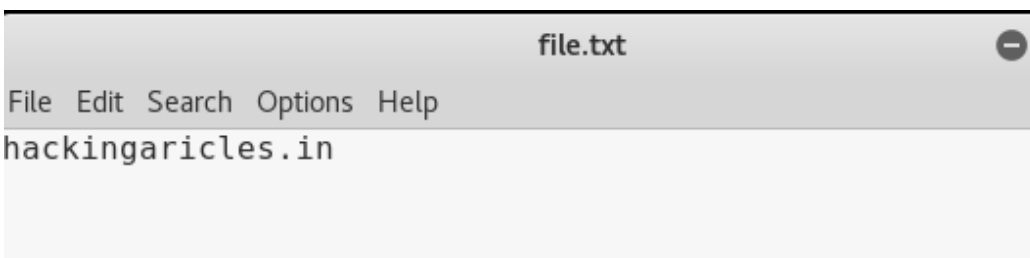
```
pavan@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt
crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Keepass [SHA256 AES 32/64 OpenSSL])
Press 'q' or Ctrl-C to abort, almost any other key for status
12345678 (ignite.kdb)
lg 0:00:00:00 DONE (2018-06-06 21:13) 3.225g/s 29.03p/s 29.03c
/s 29.03C/s 12345678
Use the "--show" option to display all of the cracked password
s reliably
Session completed
```

Cracking the RAR Password Hash

Now we will crack some compressed files, to do that we will have to create a file to be compressed so let's do that using echo command as shown in the given screenshot.

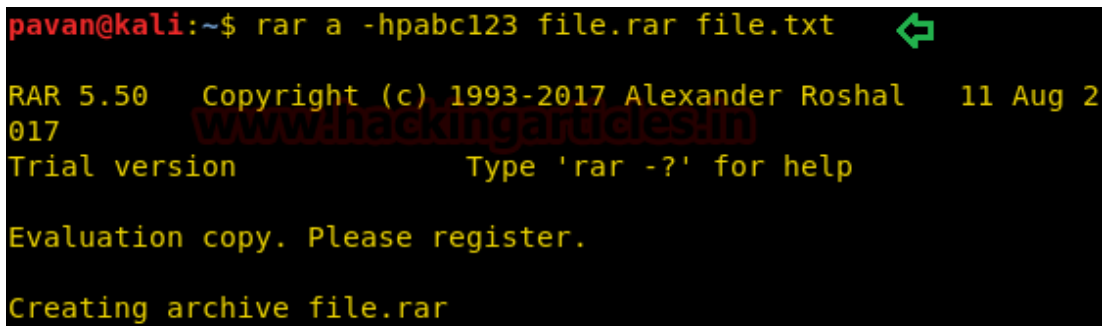
You can see that we created a file.txt which we will be using to create compressed files.

```
echo hackingarticles.in > file.txt
```



John the Ripper can crack the RAR file passwords. To test the cracking of the password, first, let's create a compressed encrypted rar file.

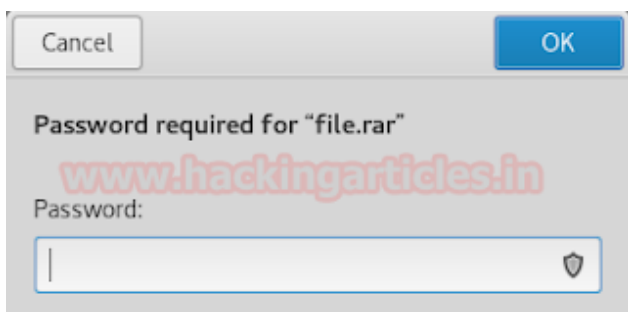
```
rar a -hpabc123 file.rar file.txt
```



```
pavan@kali:~$ rar a -hpabc123 file.rar file.txt
RAR 5.50 Copyright (c) 1993-2017 Alexander Roshal 11 Aug 2017
Trial version Type 'rar -?' for help
Evaluation copy. Please register.
Creating archive file.rar
```

- a = Add files to archive
- hp[password] = Encrypt both file data and headers

This will compress and encrypt our file.txt into a file.rar. So, when you will try to open the file, you will be greeted by the following prompt.



Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called "rar2john".

Syntax: rar2john [location of key]

```
rar2john file.rar > crack.txt
```



Now let's use John the Ripper to crack this hash.

```
john --wordlist=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the key to be “abc123”

```
pavan@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt
crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (RAR5 [PBKDF2-SHA256 128/128 SSE2 4x])
Press 'q' or Ctrl-C to abort, almost any other key for status
abc123 (file.rar)
lg 0:00:00:00 DONE (2018-06-06 21:20) 2.631g/s 31.57p/s 31.57c
/s 31.57C/s 12345678..daniel
Use the "--show" option to display all of the cracked password
s reliably
```

Cracking the ZIP Password Hash

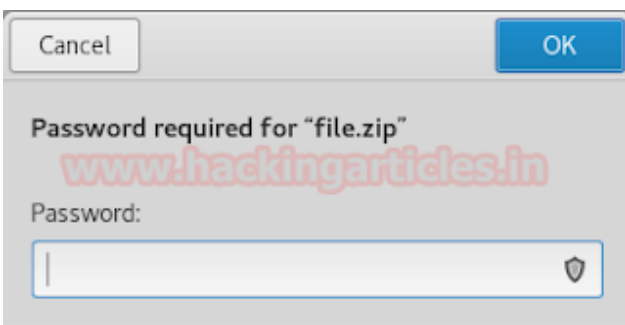
John the Ripper can crack the ZIP file passwords. To test the cracking of the password, first, let's create a compressed encrypted zip file.

```
zip -er file.zip file.txt
```

```
pavan@kali:~$ zip -er file.zip file.txt
Enter password:
Verify password:
adding: file.txt (stored 0%)
```

- e = Encrypt
- r = Recurse into directories

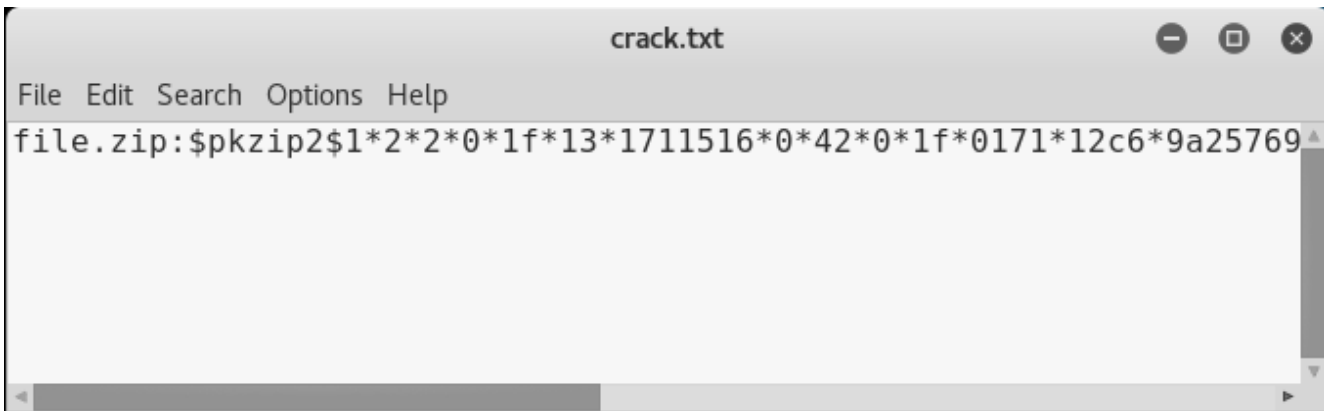
This will compress and encrypt our file.txt into a file.zip. So, when you will try to open the file, you will be greeted by the following prompt.



Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called “zip2john”.

Syntax: zip2john [location of key]

```
zip2john file.zip > crack.txt
```



Now let's use John the Ripper to crack this hash.

```
john --wordlist=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the key to be "654321"

```
pavan@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt
crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP)
Press 'q' or Ctrl-C to abort, almost any other key for status
654321 (file.zip)
lg 0:00:00:00 DONE (2018-06-06 21:33) 1.754g/s 35.08p/s 35.08c
/s 35.08C/s 654321..qwerty
Use the "--show" option to display all of the cracked password
```

Cracking the 7-Zip Password Hash

John the Ripper can crack the 7-Zip file passwords. To test the cracking of the password, first, let's create a compressed encrypted 7z file.

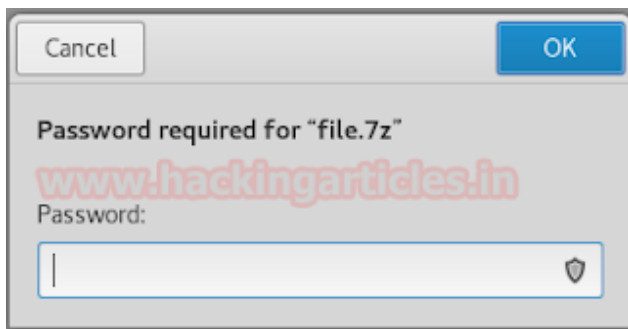
```
7z a -mhe file.7z file.txt -p"password"
```



```
pavan@kali:~$ 7z a -mhe file.7z file.txt -p"password"
7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 201
05-21
p7zip Version 16.02 (locale=en_IN,Utf16=on,HugeFiles=on,64 b
s,2 CPUs Intel(R) Pentium(R) CPU G2020 @ 2.90GHz (306A9),ASM
Scanning the drive:
1 file, 18 bytes (1 KiB)
Creating archive: file.7z
```

- a = Add files to archive
- m = Set compression Method
- h = Calculate hash values for files
- e = Encrypt file
- p = set Password

This will compress and encrypt our file.txt into a file.7z. So, when you will try to open the file, you will be greeted by the following prompt.



Now John cannot directly crack this key, first, we will change its format, which can be done using a john utility called “7z2john”. This is not inbuilt utility, It can be downloaded from here.

Syntax: zip2john [location of key]

```
python 7z2john.py file.7z > crack.txt
```



Now let's use John the Ripper to crack this hash.

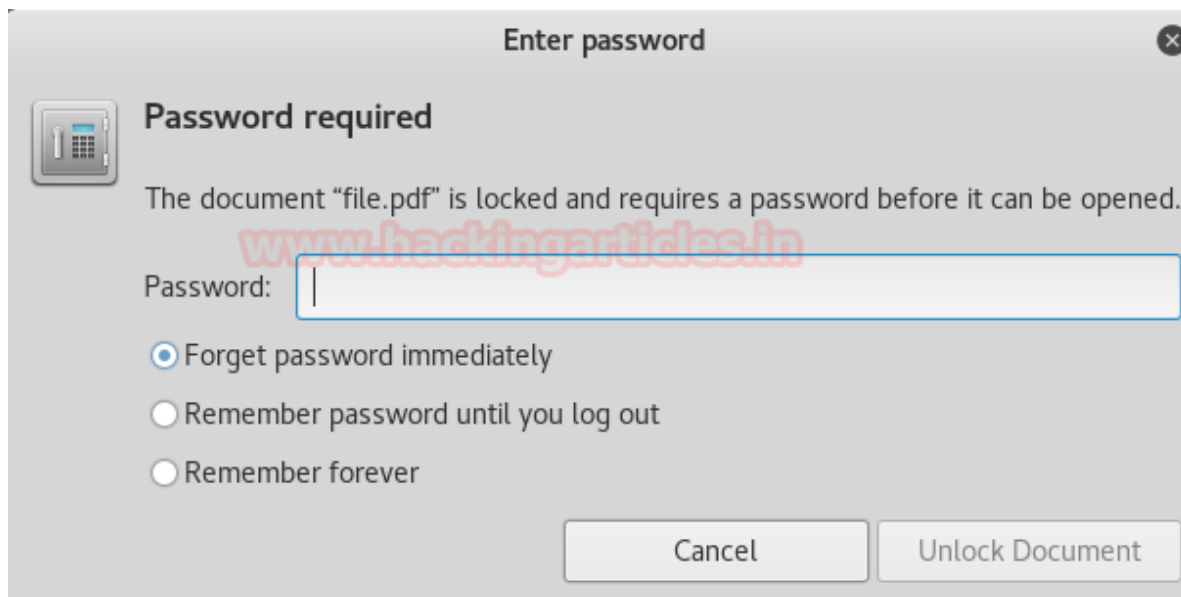
```
john -wordlist=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the key to be "password"

```
pavan@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt
crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (7z, 7-Zip [SHA256 AES 32/64])
Note: This format may emit false positives, so it will keep tr
ying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
password (file.7z)
lg 0:00:00:08 0.00% (ETA: 2018-06-16 12:26) 0.1114g/s 19.17p/s
```

Cracking the PDF Password Hash

John the Ripper can crack the PDF file passwords. You can encrypt your pdf online by using this website. This will compress and encrypt our pdf into a password protected file.pdf. So, when you will try to open the file, you will be greeted by the following prompt.



Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called "pdf2john". This is not an inbuilt utility, it can be downloaded from here.

Syntax: pdf2john [location of key]

```
python pdf2john.py file.pdf > crack.txt
```



Now let's use John the Ripper to crack this hash.

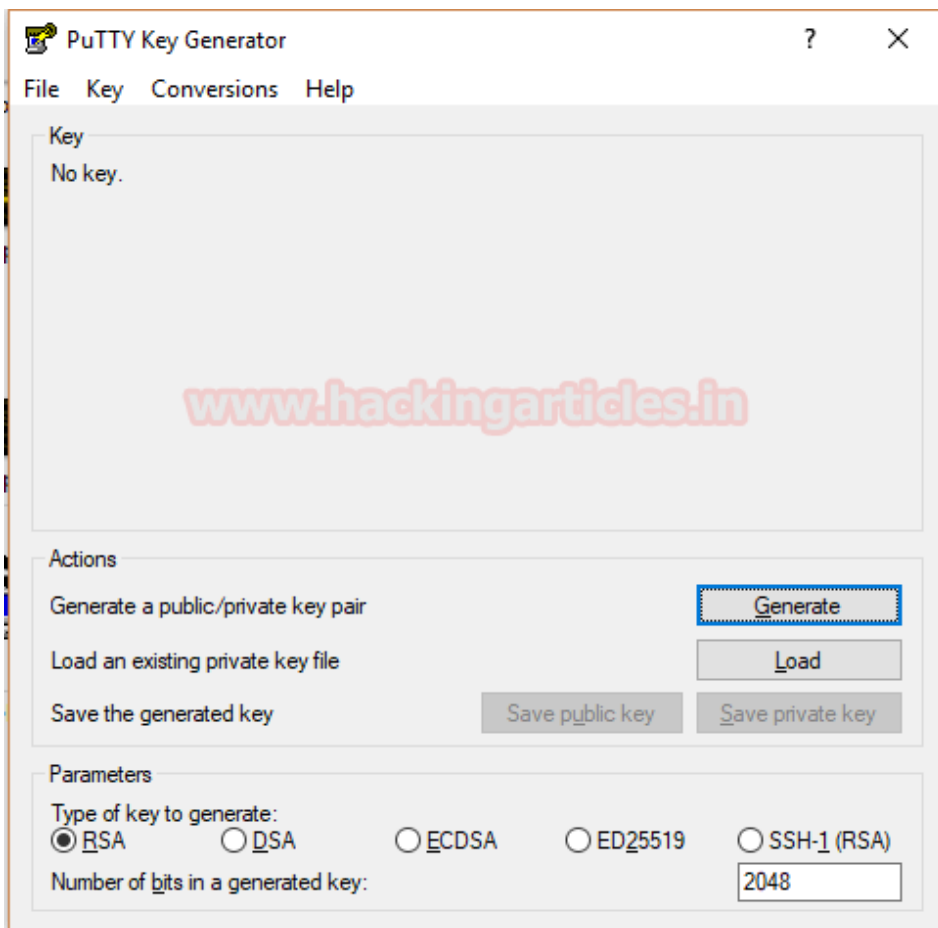
```
john -wordlist=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the key to be "password123".

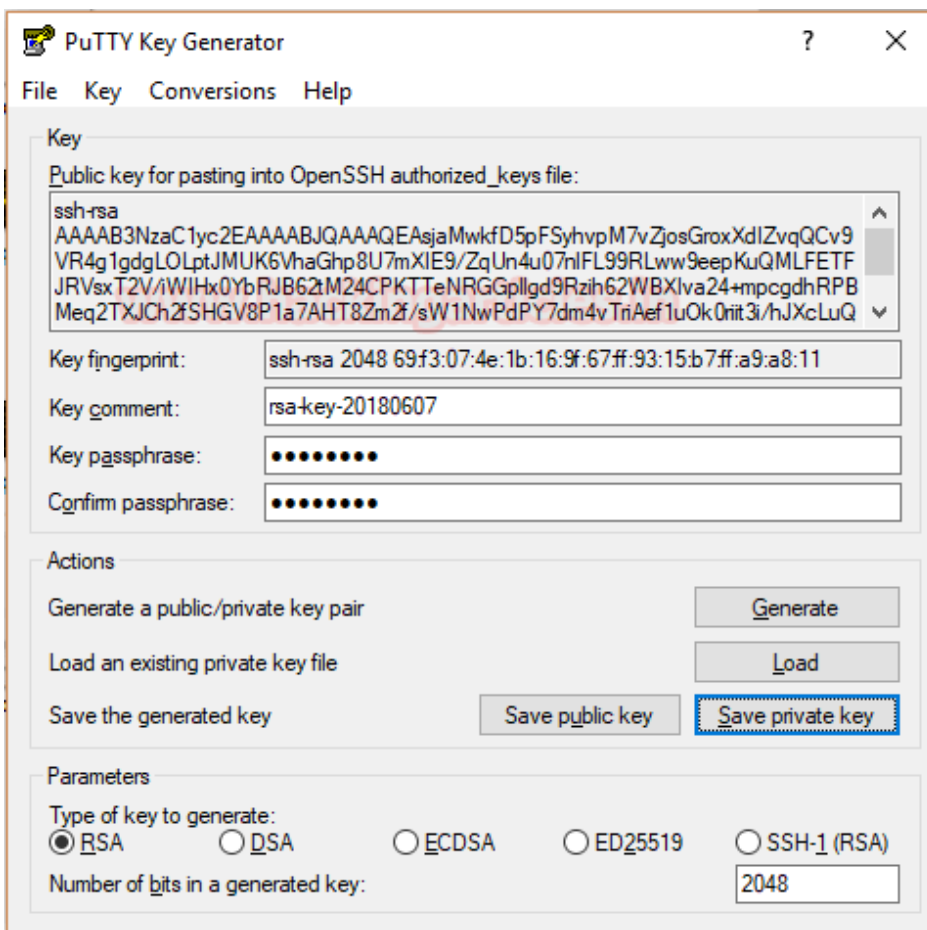
```
pavan@kali:~$ john --wordlist=/usr/share/wordlists/rockyou.txt
crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PDF [MD5 SHA2 RC4/AES 32/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
password123 (file.pdf)
lg 0:00:00:00 DONE (2018-06-06 22:57) 3.333g/s 4613p/s 4613c/s
4613C/s password123
Use the "--show" option to display all of the cracked password
s reliably
```

Cracking the PuTTY Password Hash

John the Ripper can crack the PuTTY private key which is created in RSA Encryption. To test the cracking of the private key, first, we will have to create a set of new private keys. To do this we will use a utility that comes with PuTTY, called "PuTTY Key Generator".



Click on “Generate”. After Generating the key, we get a window where we will input the key passphrase as shown in the image.



After entering the passphrase, click on Save private key to get a private key in the form of a .ppk file

After generating transfer this .ppk file to Kali Linux.

Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called “putty2john”.

Syntax: putty2john [location of key]

```
putty2john file.ppk > crack.txt
```



You can see that we converted the key to a crackable hash and then entered it into a text file named crack.txt.

Now let's use John the Ripper to crack this hash.

```
john -w=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the private PuTTY key to be “password”.

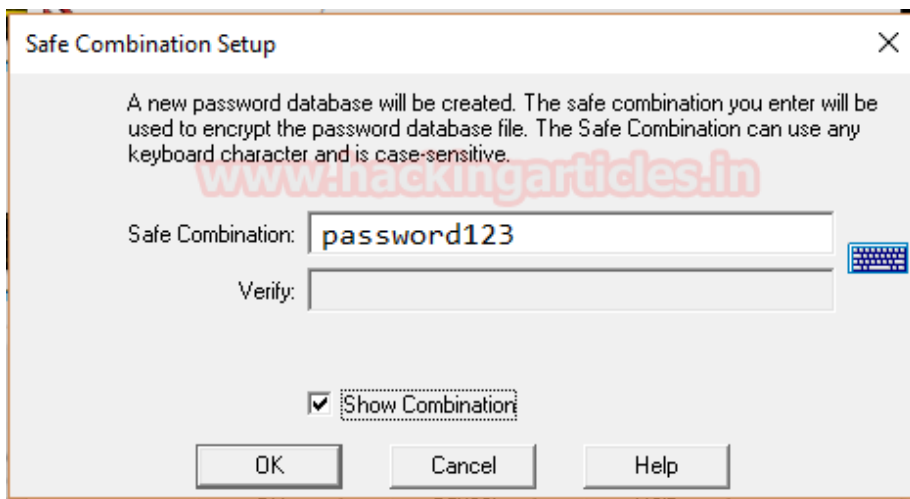
```
root@kali:~# john -w=/usr/share/wordlists/rockyou.txt crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PuTTY, Private Key [SHA1/AES 32/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
password (file)
lg 0:00:00:00 DONE (2018-06-07 02:16) 50.00g/s 200.0p/s 200.0c/s
rd
Use the "--show" option to display all of the cracked passwords
Session completed
```

Cracking the “Password Safe” Password Hash

John the Ripper can crack the Password Safe Software's key. To test the cracking of the key, first, we will have to create a set of new keys. To do this we will install the Password Safe Software on our Windows 10 System.



To get a new key, Click on “New”



In this prompt, check the Show Combination Box. After that Enter the passphrase you want to use to generate the key. This will generate a .psafe3 file.

After generating transfer this .safe3 file to Kali Linux.

Now John cannot directly crack this key, first, we will have to change its format, which can be done using a john utility called “pwsafe2john”.

Syntax: pwsafe2john [location of key]

```
pwsafe2john ignite.psafe3 > crack.txt
```



You can see that we converted the key to a crackable hash and then entered it into a text file named crack.txt.

Now let’s use John the Ripper to crack this hash.

```
john -w=/usr/share/wordlists/rockyou.txt crack.txt
```

Great! We have successfully cracked the passphrase used to create the private pwsafe key to be “password123”

```
root@kali:~# john -w=/usr/share/wordlists/rockyou.txt crack.txt
Using default input encoding: UTF-8
Loaded 1 password hash (pwsafe, Password Safe [SHA256 128/128 AVX
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (ignite)
1g 0:00:00:00 DONE (2018-06-07 02:14) 3.225g/s 4464p/s 4464c/s 446
password123
Use the "--show" option to display all of the cracked passwords re
Session completed
```