

Cracking a Password with John the Ripper.(JtR).

JtR is primarily a password/hash cracker used during penetration testing exercises that can help IT staff spot weak passwords and poor password policies.



In this example we will crack below MD5 hash.

2deb86960b7fb3a2b935beb0fbb9e26c

Can you crack this?

JtR is installed in Kali Linux. **Open your Kali Linux/Vmware/VBox**, You can also in- stall JtR on any Linux using below command.

sudo apt-get install john

Practical.

Create a text file in your folder named **password.txt** and put hash [**2deb86960b7fb3a2b935beb0fbb9e26c**] inside.

NB: Use the hash you stored in your database.

While on your folder, Right Click open in terminal.

Type: `john -h` Gives you help concerning JtR.

Type : `john --list=formats` Gives you all formats supported by JtR.

```
(kanav@Techofide)-[~/Desktop]
$ john --list=formats
descrypt, bsdicrypt, md5crypt, md5crypt-long, bcrypt, scrypt, LM, AFS,
tripcode, AndroidBackup, adxcrypt, agilekeychain, aix-ssh1, aix-ssh256,
aix-ssh512, andOTP, ansible, argon2, as400-des, as400-ssh1, asa-md5,
AxCrypt, AzureAD, BestCrypt, bfegg, Bitcoin, BitLocker, bitshares, Bitwarden,
BKS, Blackberry-ES10, WoWSRP, Blockchain, chap, Clipperz, cloudkeychain,
dynamic_n, cq, CRC32, sha1crypt, sha256crypt, sha512crypt, Citrix_NS10,
dahua, dashlane, diskcryptor, Django, django-scrypt, dmd5, dmg, dominosec,
dominosec8, DPAPImk, dragonfly3-32, dragonfly3-64, dragonfly4-32,
dragonfly4-64, Drupal7, eCryptfs, eigrp, electrum, EncFS, enpass, EPI,
EPiServer, ethereum, fde, Fortigate256, Fortigate, FormSpring, FVDE, geli,
gost, gpg, HAVAL-128-4, HAVAL-256-3, hdaa, hMailServer, hsrp, IKE, ipb2,
itunes-backup, iwork, KeePass, keychain, keyring, keystore, known_hosts,
krb4, krb5, krb5asrep, krb5pa-sha1, krb5tgs, krb5-17, krb5-18, krb5-3,
kwallet, lp, lpcli, leet, lotus5, lotus85, LUKS, MD2, mdc2, MediaWiki,
monero, money, MongoDB, scram, Mozilla, mscash, mscash2, MSCHAPv2,
mschapv2-naive, krb5pa-md5, mssql, mssql05, mssql12, multibit, mysqlna,
mysql-sha1, mysql, net-ah, nethalflm, netlm, netlmv2, net-md5, netntlmv2,
netntlm, netntlm-naive, net-sha1, nk, notes, md5ns, nsec3, NT, o10glogon,
o3logon, o5logon, ODF, Office, oldoffice, OpenBSD-SoftRAID, openssl-enc,
oracle, oracle11, Oracle12C, osc, ospf, Padlock, Palshop, Panama,
PBKDF2-HMAC-MD4, PBKDF2-HMAC-MD5, PBKDF2-HMAC-SHA1, PBKDF2-HMAC-SHA256,
PBKDF2-HMAC-SHA512, PDF, PEM, pfx, pgpdisk, pgpsda, pgpwde, phpass, PHPS,
PHPS2, pix-md5, PKZIP, po, postgres, PST, PuTTY, pwsafe, qnx, RACF,
RACF-KDFAES, radius, RAdmin, RAKP, rar, RAR5, Raw-SHA512, Raw-Blake2,
Raw-Keccak, Raw-Keccak-256, Raw-MD4, Raw-MD5, Raw-MD5u, Raw-SHA1,
Raw-SHA1-AxCrypt, Raw-SHA1-Linkedin, Raw-SHA224, Raw-SHA256, Raw-SHA3,
Raw-SHA384, ripemd-128, ripemd-160, rsvp, Siemens-S7, Salted-SHA1, SSHA512,
sapb, sapg, saph, sappse, securezip, 7z, Signal, SIP, skein-256, skein-512,
skey, SL3, Snefru-128, Snefru-256, LastPass, SNMP, solarwinds, SSH, sspr,
Stribog-256, Stribog-512, STRIP, SunMD5, SybaseASE, Sybase-PROP, tacacs-plus,
tcp-md5, telegram, tezos, Tiger, tc_aes_xts, tc_ripemd160, tc_ripemd160boot,
tc_sha512, tc_whirlpool, vdi, OpenVMS, vmx, VNC, vtp, wbb3, whirlpool,
whirlpool0, whirlpool1, wpapsk, wpapsk-pmk, xmpp-scram, xsha, xsha512, ZIP,
ZipMonster, plaintext, has-160, HMAC-MD5, HMAC-SHA1, HMAC-SHA224,
HMAC-SHA256, HMAC-SHA384, HMAC-SHA512, dummy, crypt
```


Our previous hash was an **MD5 hash**, Can you see the Raw-MD5 in above screen- shot among many others ?.

To crack a password, this is the JtR syntax

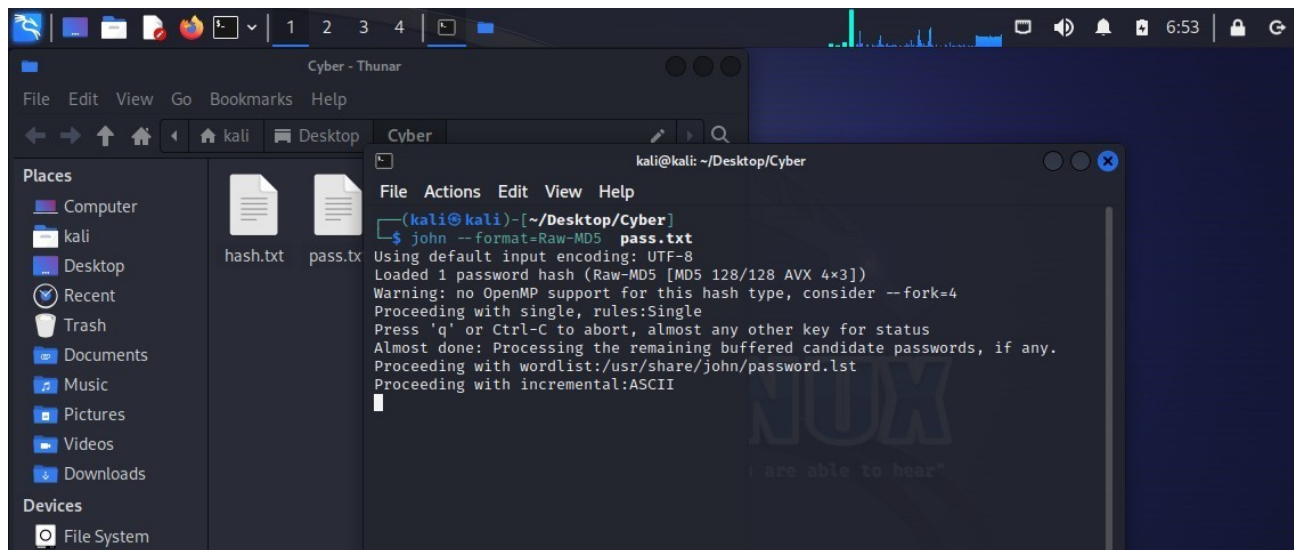
Command : `john --format=format name file_ containing_the_hashes`

Crack MD5 hash stored in pass.txt.

Command: **john --format = Raw-MD5 password.txt**

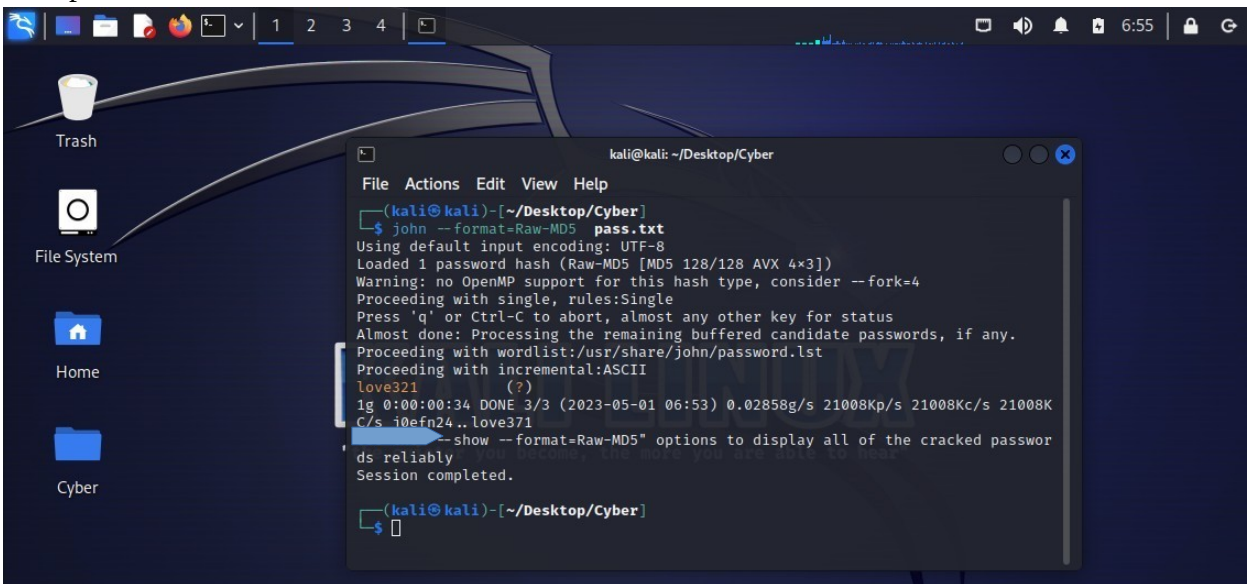
Press Enter.

Depending on Hash Complexity JtR will take some time. Give it 2 minutes



```
(kali@kali)-[~/Desktop/Cyber]
$ john --format=Raw-MD5 pass.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
```

Final Output. What was the Password in Plain Text? *Pass hidden in blue arrow.*



```
(kali@kali)-[~/Desktop/Cyber]
$ john --format=Raw-MD5 pass.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
love321 (?)
lg 0:00:00:34 DONE 3/3 (2023-05-01 06:53) 0.02858g/s 21008Kp/s 21008Kc/s 21008K
C/s i0efn24.. love371
--show --format=Raw-MD5" options to display all of the cracked password
s reliably
Session completed.

(kali@kali)-[~/Desktop/Cyber]
$
```

Remember if the password is long/hard it will also take long time to crack.

You can also crack password using HashCat Found in Kali Linux check below Link <https://www.freecodecamp.org/news/hacking-with-hashcat-a-practical-guide/>

Solutions to Protect Hashed Data.

1. Create Strong Passwords.
2. Use Latest hash algorithms such as Bcrypt, AES etc.
3. Use salt, this is an additional random string appended to the hash to make it more harder to crack.
4. Keep checking on blogs, tutorials, Twitter handles (<https://twitter.com/TheCyberSecHub>), articles and newsletters to learn on any advancements in data encryption.

Useful Links

<https://techofide.com/blogs/how-to-use-john-the-ripper-john-the-ripper-password-cracker-techofide/>
<https://www.openwall.com/john/doc/FAQ.shtml>
<https://zetcode.com/python/bcrypt/>