Procédure complète d'utilisation des outils de craquage de mots de passe : John the Ripper, Hashcat et Hydra

1. Introduction générale

Le craquage de mots de passe est une étape cruciale dans les tests d'intrusion et les audits de sécurité. Il permet d'identifier la faiblesse des mots de passe utilisés dans un système. Trois outils sont largement utilisés dans le domaine de la cybersécurité : John the Ripper, Hashcat et Hydra. Ce document propose une procédure détaillée pour chacun d'eux avec une explication de leur rôle, un exemple pratique et un glossaire de commandes utiles.

2. John the Ripper

2.1 But de l'outil

John the Ripper est un outil en ligne de commande conçu pour détecter les mots de passe faibles. Il est principalement utilisé pour craquer des mots de passe hachés présents dans des fichiers, par exemple les fichiers `/etc/shadow` sous Linux. Il supporte de nombreux formats de hash (MD5, SHA, etc.).

2.2 Exemple simple

Préparation : créer un fichier contenant un hash, par exemple :

echo -n "password" | md5sum > hash.txt

Nano pour n'avoir que le hash

Utilisation :

john --format=raw-md5 --wordlist=/usr/share/wordlists/rockyou.txt hash.txt

```
┌──(kali㉿kali)-[~]
└─$ john --format=raw-md5 --wordlist=/usr/share/wordlists/rockyou.txt hash.txt

Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 SSE2 4×3])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
password         (?)
1g 0:00:00:00 DONE (2025-05-23 14:01) 50.00g/s 9600p/s 9600c/s 9600C/s 123456..november
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

```
┌──(kali㉿kali)-[~]
└─$ echo -n "admin1234" | md5sum > hash2.txt

┌──(kali㉿kali)-[~]
└─$ nano hash2.txt

┌──(kali㉿kali)-[~]
└─$ john --format=raw-md5 --wordlist=/usr/share/wordlists/rockyou.txt hash2.txt

Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 SSE2 4×3])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
admin1234        (?)
1g 0:00:00:00 DONE (2025-05-23 14:03) 25.00g/s 7963Kp/s 7963Kc/s 7963KC/s adrian33..ab2007
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

2.3 Glossaire

john <fichier> : Lance le craquage du fichier contenant les hashes

--show <fichier> : Affiche les mots de passe déjà trouvés

--wordlist=<fichier> : Utilise un fichier de mots comme dictionnaire

--format=<type> : Force un format de hash (ex : raw-md5, sha512crypt...)

3. Hashcat

3.1 But de l'outil

Hashcat est un puissant outil de craquage de mots de passe utilisant la puissance du GPU pour effectuer des attaques sur des fichiers de hash. Il prend en charge des centaines de types de hash et permet des attaques par dictionnaire, brute force, combinées, par masque, etc.

3.2 Exemple simple

Créer un fichier contenant un hash (MD5 de 'password') :

echo -n "password" | md5sum > hash.txt (on utilisera le fichier précédent ici)

Utilisation avec dictionnaire :

hashcat -m 0 -a 0 hash.txt /usr/share/wordlists/rockyou.txt

```
┌──(kali㉿kali)-[~]
└─$ hashcat -m 0 -a 0 hash.txt /usr/share/wordlists/rockyou.txt --force

hashcat (v6.2.6) starting
```

```
Host memory required for this attack: 0 MB

Dictionary cache built:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344392
* Bytes.....: 139921507
* Keyspace..: 14344385
* Runtime...: 1 sec

5f4dcc3b5aa765d61d8327deb882cf99:password

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 0 (MD5)
Hash.Target......: 5f4dcc3b5aa765d61d8327deb882cf99
Time.Started.....: Fri May 23 14:05:21 2025, (0 secs)
Time.Estimated...: Fri May 23 14:05:21 2025, (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   392.6 kH/s (0.12ms) @ Accel:256 Loops:1 Thr:1 Vec:4
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 1024/14344385 (0.01%)
Rejected.........: 0/1024 (0.00%)
Restore.Point....: 0/14344385 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: 123456 → bethany
Hardware.Mon.#1..: Util: 11%
```

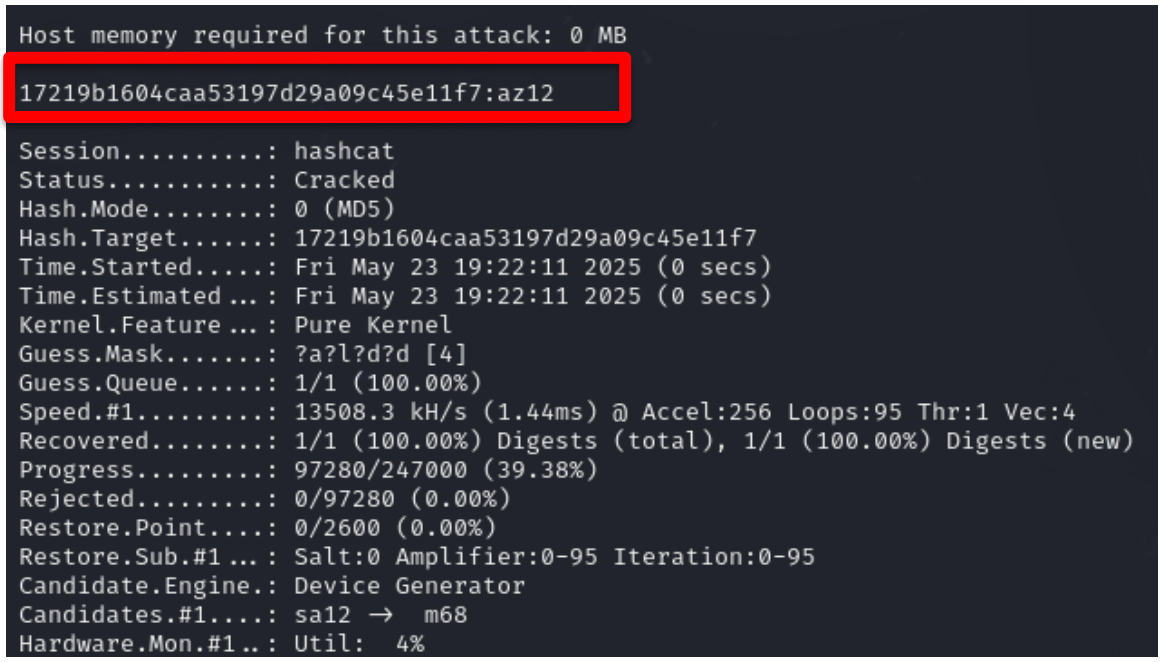
On peut également attaquer en brute force pure (selon les ressources de votre machine)

avec un nouveau fichier hash3.txt :

hashcat -m 0 -a 3 hash2.txt ?a?l?d?d

```
┌──(kali㉿kali)-[~]
└─$ echo -n "az12" | md5sum > hash3.txt

┌──(kali㉿kali)-[~]
└─$ hashcat -m 0 -a 3 hash3.txt ?a?l?d?d
```



3.3 Glossaire

-m <mode> : Spécifie le type de hash (ex: 0 pour MD5, 1000 pour NTLM)

-a <mode> : Type d'attaque (0 = dictionnaire, 3 = brute force)

--force : Ignore les avertissements liés au GPU

--help : Affiche l'aide de hashcat

?a ?l ?d ?d : Masque pour tester toutes les combinaisons de 4 caractères parmi :

?a : lettres (maj/min), chiffres et symboles

?l : lettre minuscule

?d : chiffre

4. Hydra

4.1 But de l'outil

Hydra (ou THC-Hydra) est un outil utilisé pour effectuer des attaques par force brute ou dictionnaire sur des services réseau. Il prend en charge de nombreux protocoles comme SSH, FTP, HTTP, MySQL, RDP, etc. Il est particulièrement utile pour tester les failles d'authentification sur les services exposés.

4.2 Exemple simple

Lancer une attaque sur un service SSH cible (192.168.1.10) :

hydra -l admin -P /usr/share/wordlists/rockyou.txt ssh://192.168.1.10

4.3 Glossaire

-l <login> : Spécifie le nom d'utilisateur

-L <fichier> : Spécifie un fichier contenant une liste de logins

-p <motdepasse> : Spécifie un mot de passe unique à tester

-P <fichier> : Spécifie un fichier de mots de passe

-t <nombre> : Nombre de threads à utiliser

ssh://IP : Définit le protocole et la cible