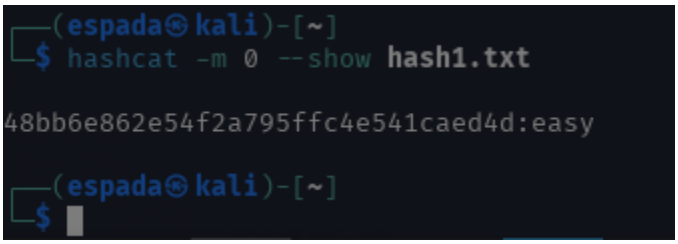
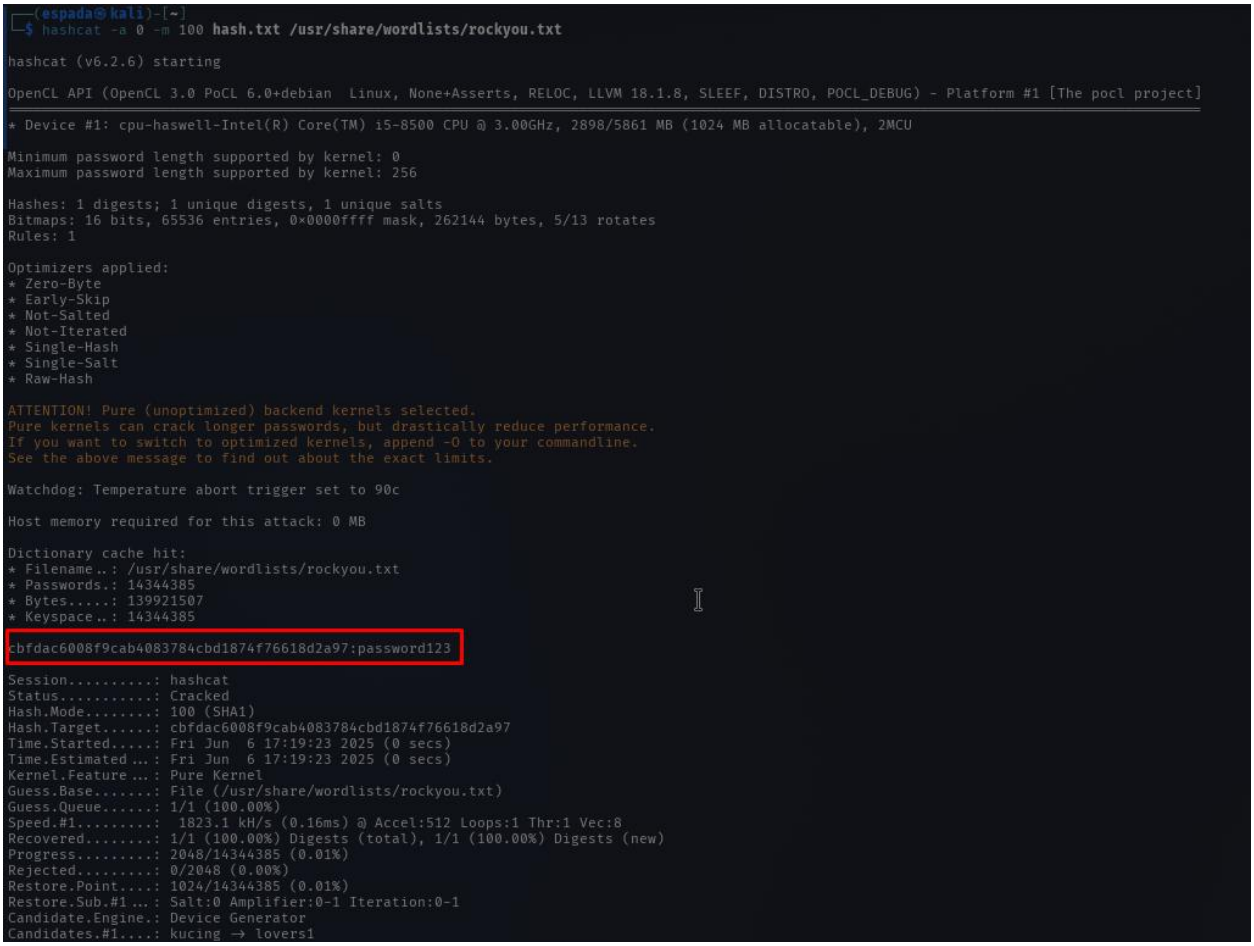
Formando: Daniel Espada

48bb6e862e54f2a795ffc4e541caed4d (md5) -> easy

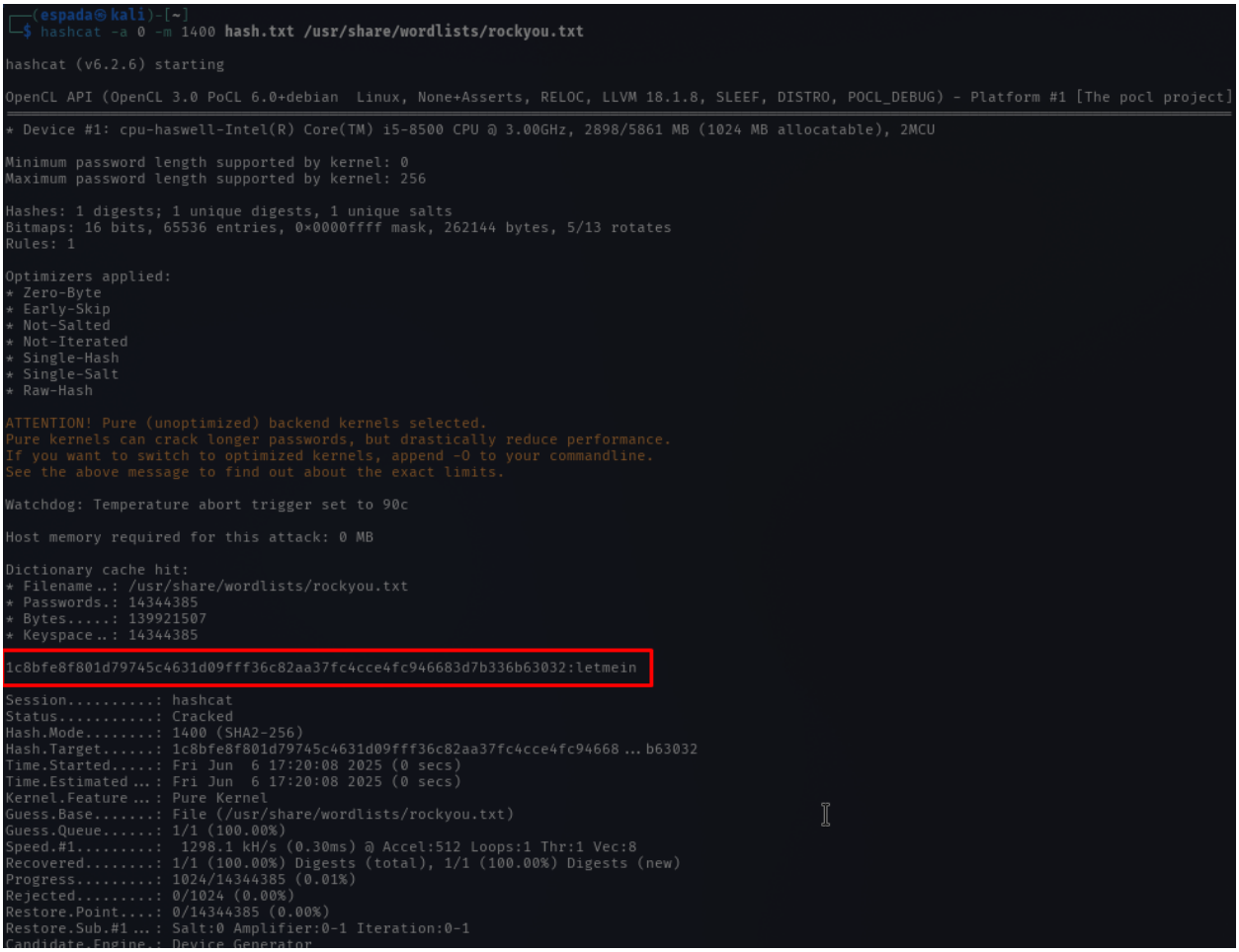


Ja tinha feito o comando "hashcat -a 0 -m 0 hash1.txt /usr/share/wordlists/rockyou.txt" e não tirei print, ou seja, tentei fazer de novo para tirar print e dá erro de que todas as hashes já tinham sido encontradas, por isso fiz este comando

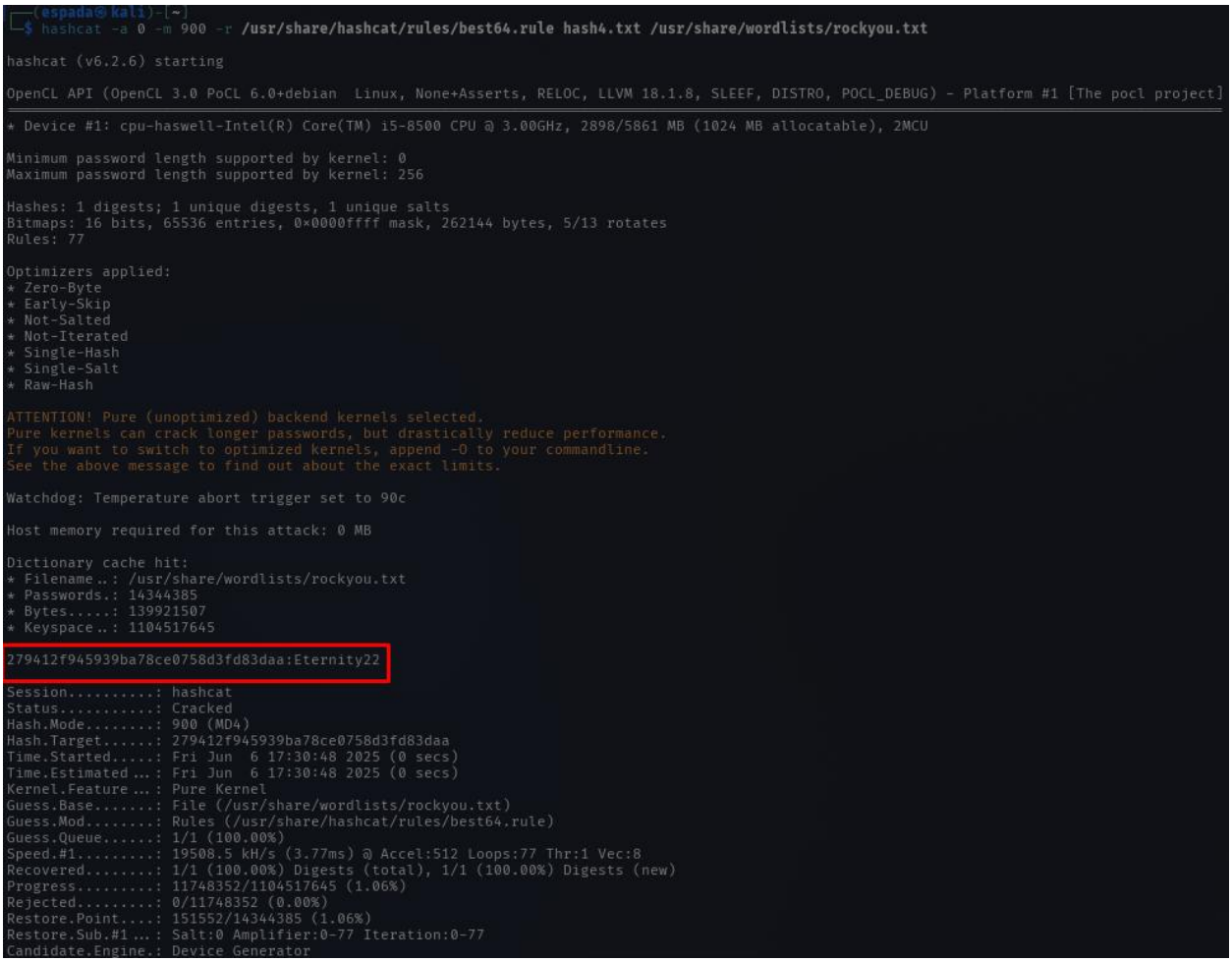
CBFDAC6008F9CAB4083784CBD1874F76618D2A97 (sha*) -> password123



1C8BFE8F801D79745C4631D09FFF36C82AA37FC4CCE4FC946683D7B336B63032 (sha*)
-> letmein



279412f945939ba78ce0758d3fd83daa (md4) -> Eternity22



$2y$12$Dwt1BZj6pcyc3Dy1FWZ5ieeUznr71EeNkJkUlypTsgbX1H68wsRom (consultar página de ajuda do hashcat) -> bleh

```
┌──(espada㉿kali)-[~]
└─$ cat /usr/share/wordlists/rockyou.txt | grep '^....$' | tee rockyou_4char.txt
```

Como soube que a password era so 4 letras, passei todas as palavras que continham 4 letras para um novo ficheiro e corri a partir desse ficheiro. Ps.: o comando foi disponibilizado por um colega.

```
└─$ hashcat -a 0 -m 3200 hash5.txt rockyou_4char.txt
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 6.0+debian  Linux, None+Asserts, RELOC, LLVM 18.1.8, SLEEF, DISTRO,
=============================================================================================
* Device #1: cpu-haswell-Intel(R) Core(TM) i5-8500 CPU @ 3.00GHz, 2898/5861 MB (1024 MB allocat

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 72

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

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 0 MB

Dictionary cache built:
* Filename..: rockyou_4char.txt
* Passwords.: 18152
* Bytes.....: 91391
* Keyspace..: 18152
* Runtime...: 0 secs

Cracking performance lower than expected?

* Append -w 3 to the commandline.
  This can cause your screen to lag.

* Append -S to the commandline.
  This has a drastic speed impact but can be better for specific attacks.
  Typical scenarios are a small wordlist but a large ruleset.

* Update your backend API runtime / driver the right way:
  https://hashcat.net/faq/wrongdriver

* Create more work items to make use of your parallelization power:
  https://hashcat.net/faq/morework

$2y$12$Dwt1BZj6pcyc3Dy1FWZ5ieeUznr71EeNkJkUlypTsgbX1H68wsRom:bleh
```