

Digital Forensics – Lab#11

Ahmad Abdullah (i22-1609)

1. Hashid gave MD5 hash family

```
(kali@kali)~[~/Downloads]
$ hashid 48bb6e862e54f2a795ffc4e541caed4d
Analyzing '48bb6e862e54f2a795ffc4e541caed4d'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] Snefru-128
[+] NTLM
[+] Domain Cached Credentials
[+] Domain Cached Credentials 2
[+] DNSSEC(NSEC3)
[+] RAdmin v2.x
```

Using man hashcat it gave the number that we need to use for md5 which is 0.

```
0 = MD5
10 = md5($pass.$salt)
20 = md5($salt.$pass)
30 = md5(unicode($pass).$salt)
40 = md5($salt.unicode($pass))
50 = HMAC-MD5 (key = $pass)
60 = HMAC-MD5 (key = $salt)
100 = SHA1
110 = sha1($pass.$salt)
120 = sha1($salt.$pass)
130 = sha1(unicode($pass).$salt)
140 = sha1($salt.unicode($pass))
150 = HMAC-SHA1 (key = $pass)
160 = HMAC-SHA1 (key = $salt)
200 = MySQL323
300 = MySQL4.1/MySQL5
400 = phpass, MD5 Wordpress, MD5 phpBB3, MD5 Joomla
500 = md5crypt, MD5 Unix, FreeBSD MD5, Cisco-IOS MD5
900 = MD4
1000 = NTLM
1100 = Domain Cached Credentials (DCC), MS Cache
1400 = SHA256
1410 = sha256($pass.$salt)
1420 = sha256($salt.$pass)
1430 = sha256(unicode($pass).$salt)
1431 = base64(sha256(unicode($pass)))
1440 = sha256($salt.unicode($pass))
1450 = HMAC-SHA256 (key = $pass)
1460 = HMAC-SHA256 (key = $salt)
```

```

1600 = md5($pass), MD5(APR), Apache MD5
1700 = SHA512
1710 = sha512($pass.$salt)
1720 = sha512($salt.$pass)
1730 = sha512(unicode($pass).$salt)
1740 = sha512($salt.unicode($pass))
1750 = HMAC-SHA512 (key = $pass)
1760 = HMAC-SHA512 (key = $salt)
1800 = SHA-512(Unix)
2400 = Cisco-PIX MD5
2410 = Cisco-ASA MD5
2500 = WPA/WPA2
2600 = Double MD5
3200 = bcrypt, Blowfish(OpenBSD)
3300 = MD5(Sun)
3500 = md5(md5(md5($pass)))
3610 = md5(md5($salt).$pass)
3710 = md5($salt.md5($pass))
3720 = md5($pass.md5($salt))
3800 = md5($salt.$pass.$salt)
3910 = md5(md5($pass).md5($salt))
4010 = md5($salt.md5($salt.$pass))

```

Using the command and giving some seconds, it gave us

```

(kali@kali)~[~/Downloads]
$ hashcat -m 0 -a 0 -o cracked.txt hash.txt ../Downloads/rockyou.txt

hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 6.0+debian Linux, None+Asserts, RELOC, LLVM 17.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The

* Device #1: cpu-sandybridge-Intel(R) Core(TM) i7-6820HQ CPU @ 2.70GHz, 1793/3650 MB (512 MB allocatable), 3MCU

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

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

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

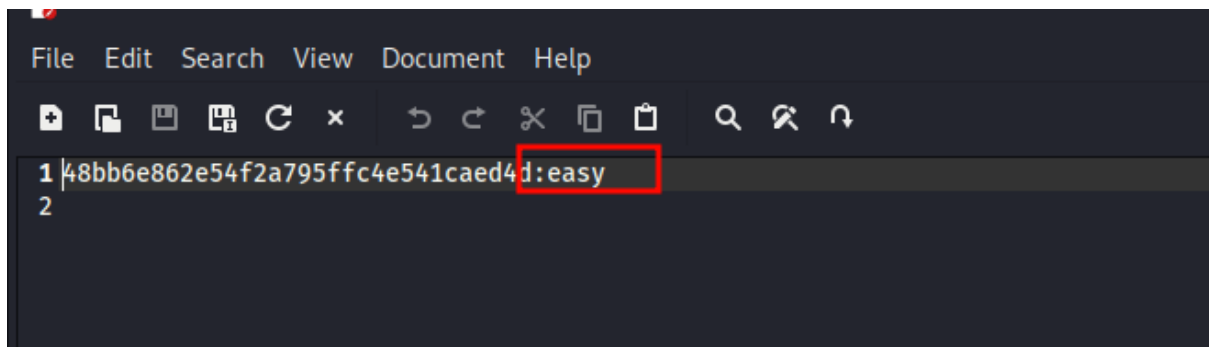
Host memory required for this attack: 0 MB

Dictionary cache hit:
* Filename..: ../Downloads/rockyou.txt
* Passwords.: 14344385
* Bytes.....: 139921507
* Keyspace..: 14344385

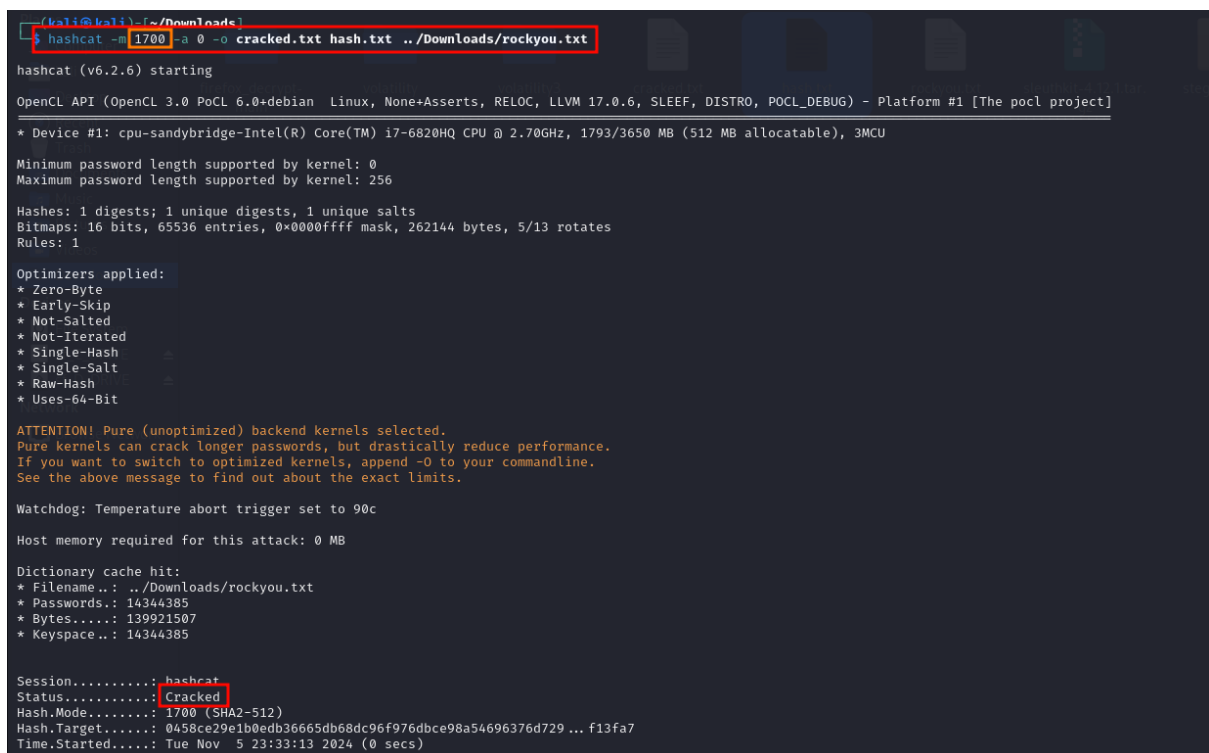
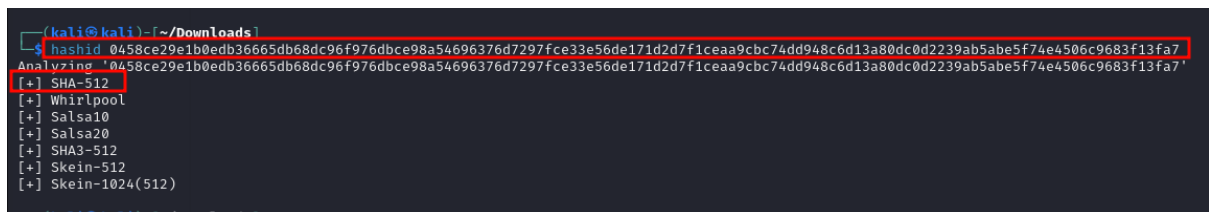
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 0 (MD5)
Hash.Target.....: 48bb6e862e54f2a795ffc4e541caed4d

```

I used the command -o cracked.txt to output it in the file rather than showing the password in the terminal. The password is 'easy'.



2. Hashid identified this hash as SHA-512 so we used 1700 number for this and rest of the command was same.



Password is: michael1997



3. Hashid gave Snefru-256 which did not work so instead I used SHA-256 code 1400 in hashcat to crack the hash.

```
(kali@kali) - [~/Downloads]
$ hashid 11adeb3106116457ba233b1ef0989ff6b15f590cfe1ab0a7ce00401c429bd58c
Analyzing 11adeb3106116457ba233b1ef0989ff6b15f590cfe1ab0a7ce00401c429bd58c
[+] Snefru-256
[+] SHA-256
[+] RIPEMD-256
[+] Haval-256
[+] GOST R 34.11-94
[+] GOST CryptoPro S-Box
[+] SHA3-256
[+] Skein-256
[+] Skein-512(256)
```

```
(kali@kali) - [~/Downloads]
$ hashcat -m 1400 -a 3 -o cracked.txt hash.txt ?u?d?l?s

hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 6.0+debian Linux, None+Asserts, RELOC, LLVM 17.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]

* Device #1: cpu-sandybridge-Intel(R) Core(TM) i7-6820HQ CPU @ 2.70GHz, 1793/3650 MB (512 MB allocatable), 3MCU

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

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates

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

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 1400 (SHA2-256)
Hash.Target....: 11adeb3106116457ba233b1ef0989ff6b15f590cfe1ab0a7ce0...9bd58c
Time.Started...: Tue Nov 5 23:50:10 2024 (0 secs)
Time.Estimated...: Tue Nov 5 23:50:10 2024 (0 secs)
Kernel.Feature...: Pupe Kernel
Guess.Mask.....: ?u?d?l?s [5]
Guess.Queue....: 1/1 (100.00%)
Speed.#1.....: 7445.2 kH/s (1.87ms) @ Accel:256 Loops:26 Thr:1 Vec:8
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 19968/2230800 (0.90%)
Rejected.....: 0/19868 (0.00%)
```

Password: N00b_

```
File Machine View Input Devices Help

~/Downloads/cracked.txt - Mousepad

File Edit Search View Document Help

1 48bb6e862e54f2a795ff4e541caed4d:easy
2 0458ce29e1b0edb36665db68dc96f976dbce98a54696376d7297fce33e56de171d2d7f1ceaa9cbc74dd948c6d13a80dc0d2239ab5abe5f74e4506c9683f13fa7:michael1997
3 11adeb3106116457ba233b1ef0989ff6b15f590cfe1ab0a7ce00401c429bd58c:N00b_
4
```

4. Hashid could not identify so I used GPT to see which type of hash has \$6\$ at the beginning and it said SHA-512 and the number for that is 1800.

```
(kali@kali) ~/Downloads
$ hashcat -m 1800 -a 0 -o cracked.txt hash.txt ../Downloads/rockyou.txt

hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 6.0+debian Linux, None+Asserts, RELOC, LLVM 17.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]

* Device #1: cpu-sandybridge-Intel(R) Core(TM) i7-6820HQ CPU @ 2.70GHz, 1793/3650 MB (512 MB allocatable), 3MCU

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

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt
* Uses-64-Bit

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 hit:
* Filename..: ../Downloads/rockyou.txt
* Passwords.: 14344385
* Bytes.....: 139921507
* Keyspace..: 14344385
```

```
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 1800 (sha512crypt $6$, SHA512 (Unix))
Hash.Target.....: $6$sup3rstr0ngs4lt$fZt5XYt.hdLFCs7Y0LSIXT.0cDaNIhtP...b0QSW1
Time.Started.....: Tue Nov 5 23:55:35 2024 (5 mins, 4 secs)
Time.Estimated...: Wed Nov 6 00:00:39 2024 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (../Downloads/rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 817 H/s (11.14ms) @ Accel:128 Loops:256 Thr:1 Vec:4
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 248960/14344385 (1.74%)
Rejected.....: 0/248960 (0.00%)
Restore.Point...: 248832/14344385 (1.73%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:4864-5000
Candidate.Engine.: Device Generator
Candidates.#1...: beetle2 -> batman1234
Hardware.Mon.#1..: Util: 93%

Started: Tue Nov 5 23:54:36 2024
Stopped: Wed Nov 6 00:00:41 2024
```

It takes about 4 minutes, and it cracked.

```
1 48bb6e862e54f2a795ffc4e541caed4d:easy
2 0458ce29e1b0edb36665db68dc96f976dbce98a54696376d7297fce33e56de171d2d7f1ceaa9cbc74dd948c6d13a80dc0d2239ab5abe5f74e4506c9683f13fa7:michael1997
3 11adeb3106116457ba233b1ef0989ff6b15f590cfe1ab0a7ce00401c429bd58c:N00b
4 $6$sup3rstr0ngs4lt$fZt5XYt.hdLFCs7Y0LSIXT.0cDaNIhtP5QdRdYP6OD349oD8hR9mEYueBRxaSAEHTAJ385wYNYEELJkb0QSW1:batman1234
5
```