

Usage of HASHCAT

The project consists of two text(.txt) files

dictionary.txt

This file consists a list of 47603 passwords made of english alphabets

```
1  'spring
2  'buster+'
3  ++++
4  +monday
5  +rose
6  +rosemg      "rosemg": Unknown word.
7  ----
8  -abril      "abril": Unknown word.
9  -ali
10 -amcld      "amcld": Unknown word.
11 -asce      "asce": Unknown word.
12 -assd      "assd": Unknown word.
13 -aug
14 -beloi      "beloi": Unknown word.
15 -beloit
16 -boston
17 -charl      "charl": Unknown word.
18 -dan
19 -desig      "desig": Unknown word.
20 -emh
21 -ever
22 -ignet      "ignet": Unknown word.
23 -jacs      "jacs": Unknown word.
24 -love
25 -mar
26 -march
27 -nov
28 -oct
29 -param
30 -patriot
31 -pens
32 -piv-
33 -year
34 =bart
35 =damian
36 =dublin
37 =nov
38 =oct
```

hashes.txt

For the respective passwords in dictionary.txt, hashes.txt consists of hashed password we got after hashing the passwords using SHA-1 Algorithm

```
1 9016d0cb7d22e428be77b03ae951118c2ee1913f
2 810051d89a4bee82273ada8686dd123d07ac1cc7
3 549715a6088e5adccd374811dc1aa7ae52ab3b78
4 6ca336109e35500683a7820f8f10d5e42e19f540
5 6ba596e10d551d4939b2b27d75bf9fde9f523a1a
6 d2f2cfaf712363d050b67dafe869a3a42e703dcb
7 af8207aece6ab93d4889751950d782c3fac325e3
8 53836a7ca47ab82b53d3e70bc41d05dad13840a4
9 4327683576f07f105ecd51d8a5a37a854168016f
10 af2ae4ee650e3f8f8fafcfbfe3e8a20910c8fba6a
11 602a8ba026d7542eababec8abfd07eb34e03bd43
12 7cf1e601990a70934382dc14a52453c00148bc3c
13 0450c2a8769b9341b0f924add9d7dd4ffe86e8f4
14 5d99ffaf849233380c573cd237879399669c2970
15 8fb5b77b3b9d6ab921f7b16896039d2e50719fbe
16 18197ba3790682f65b8095e5db79e35d5b2e8db0
17 30a91cdc90892e6db92ca366f8d10909031e92ed
18 3beadd01d2551f158e15adae00aea62d35bc96de
```

recovered.txt

After running HASHCAT on hashes.txt and dictionary.txt, here's what we get:

```
1 636f5dbc72c1a3d15a8005bf85e3c58fc76ca95d:alyssa
2 aef1a877b7d98f0272ecf9326eacd002f2890331:alyssab "alyssa
3 91bb850e8a411b83cd5662e117d457406ecb78c1:alyssad "alyssa
4 91059c2e24104c575b8968cca4df5a6ce7eac458:alyssak "alyssa
5 e7db7332c269553b6566f3ed74c6d518ee6b9424:alyssam "alyssa
6 af0e9fb3353d3aa6d43818d3e5d9d22ee5b154a2:alyssas "alyssa
7 203d2071a3026e3281e772709575894b1229d3a9:alyse "alyse"
8 e68558bc56c50a15dc92806d8135c0e66e92c901:alyssia "alyssi
9 35b5a3d5e653d52115c6dccdf741e8dcf7c5de02:amaa "amaa": Ur
10 6aec8e9125279a8b5229ff09a7eccd46443c4d2e:amaama "amaama"
11 607503d3b0e5c60e3c7aa92d39fc541fec623059:amab "amab": Ur
12 e033aff4209e9fa62fd11042a6be65044dbe658d:amable "amable"
13 bba4d2e285812a355f740a4b2dc4deab72a15ef5:amac "amac": Ur
14 8ab3f2ccf50720826f294f45bd471c3e61e329a0:amad "amad": Ur
15 de3ece35d710816cd0441a75269a4bb13d5895d3:amada "amada":
16 cc8caff161a4c910313fb796a47f735f30c15cb4:amadeo "amadeo"
17 ebd6ca3abb01f628385fa8ba6227d076719a97be:amader "amader"
18 c2139c65e0b627c867ad32317e4e51b5203b60c9:amadeus
```

hash.py

```
import hashlib

with open("dictionary.txt", "r") as f:

    passwords = f.read().splitlines()

with open("hashes.txt", "w") as f:

    for password in passwords:

        sha1_hash = hashlib.sha1(password.encode()).hexdigest()

        f.write(sha1_hash + "\n")

total_hashes = len(passwords)
```

```

cracked_hashes = set()

with open("recovered.txt", "r") as f:

    for line in f:

        if ":" in line:

            hash_part = line.split(":")[0].strip()

            cracked_hashes.add(hash_part)

success_rate = (len(cracked_hashes) / total_hashes) * 100

print(f"Total hashes: {total_hashes}")

print(f"Unique hashes cracked: {len(cracked_hashes)}")

print(f"Success rate: {success_rate:.2f}%")

```

Command to use HASHCAT

```

C:\hashcat-6.2.6>hashcat.exe -a 0 -m 100 "D:\SNU\6th Sem\FIS\Hashcat\hashes.txt" "D:\SNU\6th Sem\FIS\Hashcat\dictionary.txt" --outfile="D:\SNU\6th Sem\FIS\Hashcat\recovered.txt" --force

```

```

Watchdog: Temperature abort trigger set to 90c

```

```

Host memory required for this attack: 1172 MB

```

```

Dictionary cache built:

```

```

* Filename..: D:\SNU\6th Sem\FIS\Hashcat\dictionary.txt
* Passwords.: 47603
* Bytes.....: 322069
* Keyspace..: 47603
* Runtime...: 0 secs

```

```
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 100 (SHA1)
Hash.Target.....: D:\SNU\6th Sem\FIS\Hashcat\hashes.txt
Time.Started.....: Sun Mar 30 23:45:27 2025, (10 mins, 5 secs)
Time.Estimated...: Sun Mar 30 23:55:32 2025, (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (D:\SNU\6th Sem\FIS\Hashcat\dictionary.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 323.0 kH/s (0.20ms) @ Accel:512 Loops:1 Thr:64 Vec:1
Speed.#2.....: 375 H/s (1.74ms) @ Accel:128 Loops:1 Thr:64 Vec:1
Speed.#*.....: 323.4 kH/s
Recovered.....: 47603/47603 (100.00%) Digests (total), 47603/47603 (100.00%) Digests (new)
Remaining.....: 0 (0.00%) Digests
Recovered/Time...: CUR:0,N/A,N/A AVG:702.08,N/A,N/A (Min,Hour,Day)
Progress.....: 47603/47603 (100.00%)
Rejected.....: 0/47603 (0.00%)
Restore.Point...: 0/47603 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Restore.Sub.#2...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1...: "spring -> beltran
Candidates.#2...: beltire -> ~bruins
Hardware.Mon.#1..: Temp: 55c Util: 7% Core: 990MHz Mem:5000MHz Bus:4
Hardware.Mon.#2..: N/A
```

Success rate in recovering hashed password

```
● PS D:\SNU\6th Sem\FIS\Hashcat> & C:/Users/6th Sem/FIS/Hashcat/hash.py"
Total hashes: 47603
Unique hashes cracked: 47603
Success rate: 100.00%
○ PS D:\SNU\6th Sem\FIS\Hashcat> □
```