

# 1845479-HW04

Carmignani Federico

November 28, 2021

## Abstract

The goal of this homework is to determine passwords given salts and SHA256 hashes of the salted passwords.

The solution has to be sent to `cns@diag.uniroma1.it` from the institutional email.

The password to unzip the file with the ciphertext published by Classroom is: 'U9(\*\$!'.

The deadline is December 1<sup>th</sup> at 11:59 pm.

## 1 What is the challenge

The challenge consists in an offline attack to find passwords, given a passwords' file.

The objective is to determine user passwords given a file containing just the salts (12 bits) and the SHA256 hashes of the salted passwords.

SHA256 is a 256-bit hash algorithm (Secure Hash Algorithm) which is used for cryptographic security. The hash algorithm is performed giving as argument the sum of the salt and the password. Therefore, the task is to try the password such that the hash contained in the file is equal to the hash computed with this password and the salt present in the file, and this has to be repeated for each line of the file.

## 2 How to solve the challenge

The format of a single line in the file is: `< salt_in_cleartext >< hash_of_the_salted_password > .`

The salt in clear text is in a text format, instead the hash is hexadecimal.

They are not divided by any special or tabular character.

There would be many possibilities to solve the challenge, but many of them are not practically possible, for example reversing the SHA256. The choice taken has been the dictionary attack, using the famous 'rockyou' dictionary with the most popular passwords used in the world. Another possibility could also be to use specific tools like Crackstation. It is a dictionary attack that tries all the words contained in the dictionary and it sees if the hash calculated using the salt in the specific row of the file is the same of that at the end of the salt.

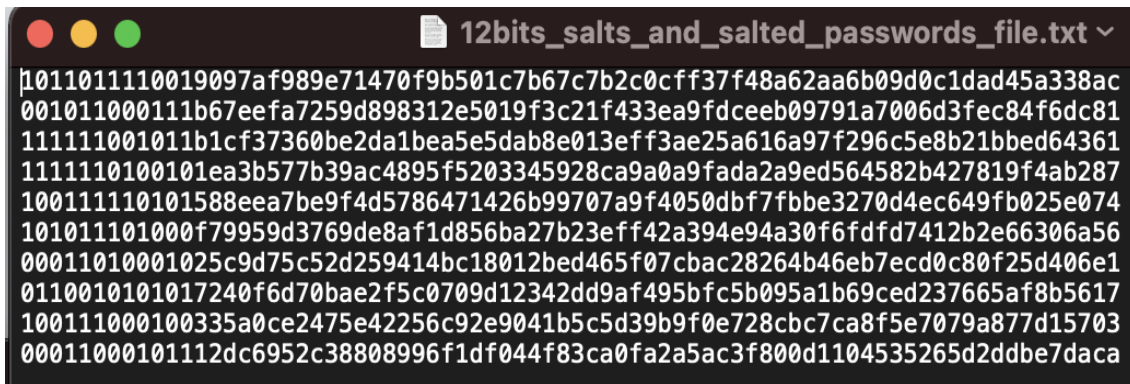
So, in the worst case, the program stopped after that every password has been tried for each line of the file.

## 3 How to use and run the program

The program performs what described before automatically. It has to be executed using Spider or another IDE to run the program or also in the terminal using Python interpreter with this command: 'python 1845479-HW04.py', the important thing is that the file with the salts and hashes and the file of the dictionary is in the same folder of the Python script. The library hashlib is imported to use the SHA256 function, at the beginning the file is read line by line, the salt and the hash are taken and every password in the 'rockyou' dictionary is tried, in case of success the passwords are printed.

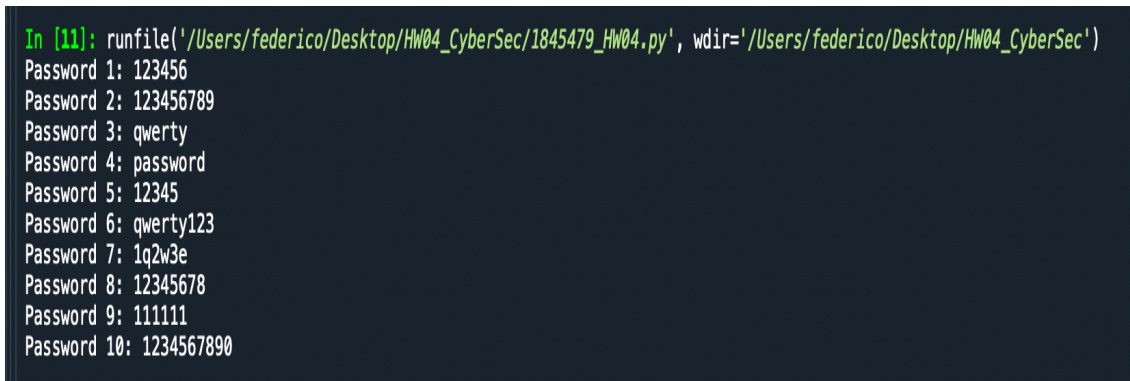
### 3.1 The code

#libraries



```
1011011110019097af989e71470f9b501c7b67c7b2c0cff37f48a62aa6b09d0c1dad45a338ac
001011000111b67eefa7259d898312e5019f3c21f433ea9fdceeb09791a7006d3fec84f6dc81
11111001011b1cf37360be2da1bea5e5dab8e013eff3ae25a616a97f296c5e8b21bbcd64361
111110100101ea3b577b39ac4895f5203345928ca9a0a9fada2a9ed564582b427819f4ab287
10011110101588eea7be9f4d5786471426b99707a9f4050dbf7fbbe3270d4ec649fb025e074
101011101000f79959d3769de8af1d856ba27b23eff42a394e94a30f6fdfd7412b2e66306a56
00011010001025c9d75c52d259414bc18012bed465f07cbac28264b46eb7ecd0c80f25d406e1
0110010101017240f6d70bae2f5c0709d12342dd9af495bfc5b095a1b69ced237665af8b5617
100111000100335a0ce2475e42256c92e9041b5c5d39b9f0e728cbc7ca8f5e7079a877d15703
00011000101112dc6952c38808996f1df044f83ca0fa2a5ac3f800d1104535265d2ddbe7daca
```

Figure 1: Demo 1.



```
In [11]: runfile('/Users/federico/Desktop/HW04_CyberSec/1845479_HW04.py', wdir='/Users/federico/Desktop/HW04_CyberSec')
Password 1: 123456
Password 2: 123456789
Password 3: qwerty
Password 4: password
Password 5: 12345
Password 6: qwerty123
Password 7: 1q2w3e
Password 8: 12345678
Password 9: 111111
Password 10: 1234567890
```

Figure 2: Demo 2.

```
import hashlib

def find_passwords():
    num=0
    #opening the file with salts and hashes and take them line by line
    for line in open("12bits_salts_and_salted_passwords_file.txt","r"):
        salt = line[:12]
        hash = line[12:].strip()
        #passwords from a dictionary are tested and in positive case are printed
        for password in open("rockyou.txt", "r"):
            if (hashlib.sha256((salt + password).encode("utf-8")).hexdigest().encode("utf-8")
                == hash.encode("utf-8")):
                num+=1
                print("Password "+str(num)+":",password.strip())
                break
    return

find_passwords()
```

### 3.2 The program in use

The file with salts and hashes is shown in Figure 1.  
After many attempts the passwords have been found and are printed, an illustration is given in Figure 2, where the Python console shows the prints.