Project Report

On

Designing of a smartest strategy to Crack Passwords

By

Priyanka Nagarapu (U25229125)

Livya Harika Choday (U98578275)

Sneha Potlapally (U76253166)

ABSTRACT: With millions of Internet users increasing day by day, maintaining highly reliable and secure data communication between different corporations is very important. The wide-ranging applications and the Internet's numerous benefits contribute to its immense utility. However, numerous security threats threaten the success and effectiveness of the internet and various information systems. SHA-1 is one of the important cryptographic Hash functions for enhancing security. In this project, we break down the SHA-1 code in order to decrypt the data. To do that, together with the hashed data, we are given a set of more than 5000 words file dictionary. Brute force attack method is implemented to break the code here.

INTRODUCTION: Rapid computer network system development brings great convenience to users as well as new security threats. It refers to the system's network system reliability, confidentiality, integrity, and data information availability. The problem of network security generally includes security of the network system and data security. It refers to the system's network system reliability, confidentiality, integrity, and data information availability.

Cryptography is a type of feature that provides secure communication. There are usually 3 types of methods for pursuing security goals such as Symmetric Cryptography, Asymmetric Cryptography, and other Hash functions. The dispatch or plain text is enciphered or hashed in the schemes mentioned above. The output acquired is expressed as coded text or hash function. This result is then transformed into the party that receives it. In the case of keyed scheme, the receiver would decipher the text with the key, and it could not decipher the message for the hash function unless they knew that the source data was being transformed.

Although there are many attacks, Brute force attack is a type of attack where the intruder made an effort to decipher the data on any possible alphabet associations, numbers, symbols, etc. They intrude the encrypted text in keyed schemes and make an effort to obtain the plain text, format it and measure it with the native hash data for hashed scheme.

we tried to decipher the group of passcodes in a file along with the additional dictionary file compromising feasible words that might have been used as the original passcode.The indigenous passcodes are hashed SHA1 where the data of significantly arbitrary length is transfigured to a message of meaningless fixed length. We used different attacking plans such as hashing the message in the dictionary file and contrasting the consequences with the password dataset given.

ATTACKING STRATEGIES

Brute-force attack:

It is a password speculating method. We know that attacks are exceptionally basic against sites and web servers. They are a standout amongst the most well-known vectors used to bargain sites.

The procedure is straightforward, and the attackers fundamentally attempt numerous permutations of usernames and passwords until they discover one that works.

Dictionary attack:

A dictionary attack is an attack that endeavors to speculate the key of a ciphertext by attempting a wide range of normal passwords and conceivable passwords that are probably going to be utilized by people. A dictionary attack utilizes what is known as a dictionary, which stores commonEnglish words, expressions, and passwords prepared to figure as the key.Dictionary attacks are more effective than a Brute Force Attack as they don't need to attempt such huge numbers of mixes – yet with the drawback that if the key isn't contained in the dictionary, it will never effectively discover it.

In this project, we have used brute force method to crack all the passwords because it tries every permutation and combinations and all the passwords can be cracked by using this method, but it takes more time if length of the password is more. Firstly, we import hashlib, which converts passwords and guesses into their respective hash functions.

- Secondly, the itertools package which is used to implement a number of iterators building blocks.

We used 5 cases to implement our project.

CASE 1: STRAIGHT SEARCH

In this method we compare the hashes in the password file to the hashes of the words in the dictionary file. In this method, we can find the passwords directly from the dictionary itself.

Decoded SHA1 Message	**Hashed Message**
password	5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8
university	f9f914060ccb1e10d551ad49016b1a6658d6edec
notwithstanding	2da33edd1e7f7c2c750282073cbfecc251561792
sounding	42445617d7b2749b167789e2a0a0fd83fda36d39

CASE 2: Number Combinations Using Permutations

It generates hash function for different number combinations and compare with the password file.

Decoded SHA1 message	Hashed message
123456	7c4a8d09ca3762af61e59520943dc26494f8941b
0000	39dfa55283318d31afe5a3ff4a0e3253e2045e43
1234567	20eabe5d64b0e216796e834f52d61fd0b70332fc
123	40bd001563085fc35165329ea1ff5c5ecbdbbeef
20161021	8cb3002abcc525d5aa6f9c1fe08c87f5d1fb5e07

CASE 3: Two Word Combinations

In this method we combine two words into one word and generate the hash. This hash is now compared with the password file.

Decoded SHA1 message	Hashed message
lovedollars	f2c8ee589d1ae0a578dc276ce7a3712e9b020a9b
washingtonartists	e2af32446bea52f4cc6f4920158c846248aca6a0
callhoney	a3866018a493c776081bfcaddb63ce004a47befo
sillymind	56afd90d4beaa873618ef1decf12f7bd35e07df5

CASE 4: Combinations of words and numbers

We combine both words and numbers and then generate the hash for entire combination and compare the obtained hash with the password.

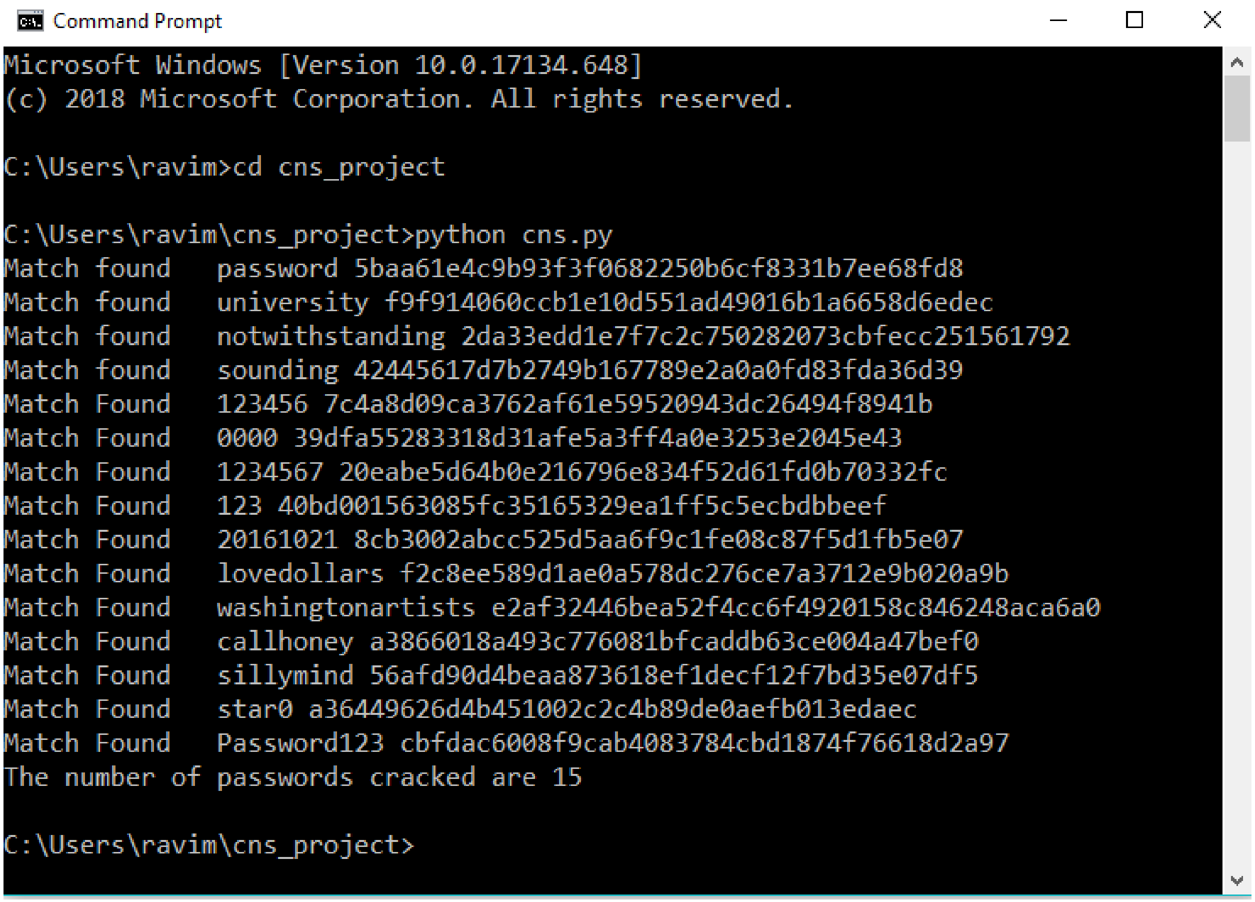
Decoded SHA1 message	Hashed message

star0	a36449626d4b451002c2c4b89de0aefb013edaec
password123	cbfdac6008f9cab4083784cbd1874f76618d2a97

CASE 5: Three Word Combinations and Three word plus Number combinations:

We tried to combine three words as well as three word plus number combinations (for example 'applebearhead123') then generated the hash for the combinations and tried to search the password in the dictionary file. Unfortunately, due to large number of permutations that are formed due to three for loops and iter methods for number this part of code crashed. We tried to run it in three different computers, and we were able to crack the password after the presentation.

Output:



```
Microsoft Windows [Version 10.0.17134.648]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ravim>cd cns_project

C:\Users\ravim\cns_project>python cns.py
Match found   password 5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8
Match found   university f9f914060ccb1e10d551ad49016b1a6658d6edec
Match found   notwithstanding 2da33edd1e7f7c2c750282073cbfecc251561792
Match found   sounding 42445617d7b2749b167789e2a0a0fd83fda36d39
Match Found   123456 7c4a8d09ca3762af61e59520943dc26494f8941b
Match Found   0000 39dfa55283318d31afe5a3ff4a0e3253e2045e43
Match Found   1234567 20eabe5d64b0e216796e834f52d61fd0b70332fc
Match Found   123 40bd001563085fc35165329ea1ff5c5ecbdbbeef
Match Found   20161021 8cb3002abcc525d5aa6f9c1fe08c87f5d1fb5e07
Match Found   lovedollars f2c8ee589d1ae0a578dc276ce7a3712e9b020a9b
Match Found   washingtonartists e2af32446bea52f4cc6f4920158c846248aca6a0
Match Found   callhoney a3866018a493c776081bfcaddb63ce004a47bef0
Match Found   sillymind 56afd90d4beaa873618ef1decf12f7bd35e07df5
Match Found   star0 a36449626d4b451002c2c4b89de0aefb013edaec
Match Found   Password123 cbfdac6008f9cab4083784cbd1874f76618d2a97
The number of passwords cracked are 15

C:\Users\ravim\cns_project>
```

Conclusion:

Here it is safe to say that for shortest passwords Brute-Force is most efficient way to crack password. But it takes significant amount of time especially to crack Case 3 and Case 5. We would optimize the code so that we can avoid the for loops and find a more optimized way to crack all the 20 passwords.