Honey Encryption based Password Manager

[1]Nahri Syeda Noorunnisa, [2]Dr. Khan Rahat Afreen
[1]M.E. Student, Department of Computer Science and Engineering,
Marthawada Shikshan Prasark Mandal's Deogiri Institute of Engineering & Management Studies,
Aurangabad Maharashtra state, India 2018-19
[2] Associate Professor, Department of Computer Science and Engineering,
Marthawada Shikshan Prasark Mandal's Deogiri Institute of Engineering & Management Studies,
Aurangabad Maharashtra state, India 2018-19

Abstract: It is seen from the recent data breaches that a large number of users use weak passwords and also reuse the same on different websites. Cryptographic hash function is used to protect the passwords before storing them in the database. But when the passwords are weak, these cryptographic hash functions can get compromised as there are many hash cracking tools easily available. The security level of Password-Based Encryption (PBE) scheme is also affected due to selection of weak passwords as the message in PBE is encrypted under a password. PBE is used to protect sensitive data and mostly used in Password Managers. A software application like a Password Manager (PM) help users compiles small database of login credentials such as passwords and their associated accounts. It is encrypted with a master password and thus this master password becomes vulnerable to brute force cracking. The proposed approach aims to develop a secure Password Manager that uses Honey Encryption technique for preventing the passwords from brute face attacks. In this paper we present a framework, which uses Honey Encryption technique for securing the stored passwords with the combined technique of One Time Pad, for additional security.

***IndexTerms* – Password based Encryption, Password Managers, Brute Force Attack, Honey Encryption, One Time Pad.**

I. INTRODUCTION

Data breaches are increasingly becoming common and harmful. Security breaches proved that many users around the world use passwords which are easy to remember. It is observed that majority users just select one easy to remember password and use it for all websites and even note it down for reference, thus making it more vulnerable to attacks.

The largest collection of stolen emails and passwords in the hacking history was recently discovered and revealed by a security researcher, Troy Hunt. The collection is ominously called "Collection #1" and bundles some 773 million emails, 21 million passwords. [1] Few of the biggest data breaches include Yahoo - 3 billion user accounts hacked, Marriott International - 500 million customers data leaked, Adult Friend Finder - More than 412.2 million accounts compromised, eBay -145 million users information leaked, etc. [2] These and similar other breaches [3,4,5,6] provided proof that users repeatedly use easy to guess login credentials. If these login names and passwords become easy to guess then it becomes more likely that the attackers or hackers can break into these accounts using various attacking techniques such as Dictionary attacks, Brute force, and other readily available password cracking tools. The vulnerability of weak passwords also affects the security provided by Password-Based Encryption (PBE) technique as it uses password to encrypt and thus suffers from the threat of brute force attacks. This technique is mostly used with Password Managers, which help users store their passwords in a secure way. With the increasing user base of Password Managers such as Dashlane, KeyPass [7] etc. they are rapidly becoming vulnerable to attacks.

The selections of weak passwords lead to the development of Honey Encryption concept. In computer security, the term 'honey' is used to denote false resource also known as decoys or honeywords which are used to misguide an attacker. Honey Encryption technique creates a cipher text, decryption of which with an incorrect key or password yields a valid looking bogus message which makes it difficult for the attacker to understand when the attack is successful. In the proposed system, an attempt is made to develop a Password Manager which uses the technique of Honey Encryption. An additional level of security is added by using One Time Pad. The system is developed such that the brute force attack will give the attacker similar looking data, making it difficult to guess whether the attack is successful or not.

II. LITERATURE SURVEY

The compromise of the database of passwords in the breaches mentioned previously, revealed that storing the passwords in plaintext form is vulnerable to attacks. Passwords are stored in the form of cryptographic hash function to ensure their security. Hashing is done by using a cryptographic hash function which is irreversible. Simply storing the passwords in hash form does not meet the actual security needs as there are many techniques to break the hashes and obtain the plaintext. Different techniques of password cracking:

- Dictionary attacks: uses a file containing a list of passwords.
- Brute-Force attack: adversary tries every possible combination of characters.
- Rainbow Table attack: large sets of precompiled tables filled with hash values that are pre-matched to possible plaintext passwords.

Hashing uses the method of salting, where the hashes are randomized by appending or pre-appending a salt, a random string. [8] As a result, the same password hash gets transformed into a completely different string every time. Though hashing is said to be irreversible, there are many tools available to crack the hashes. The 2014 breach of Yahoo, [9] revealed that the one of the major reason behind the disclosure of hashed passwords are the hash cracking tools. John The Ripper, Ophcrack, Brutus [10] are some of the available Hash cracking tools.

2.1 Password Based Encryption

The traditional password-based encryption also gets tampered due to the weak selection of passwords. Password Based Encryption (PBE) is a symmetric cryptographic method [11] that uses a password-like key to perform the encryption and the decryption process.

The technique of PBE can be explained as consisting of an encryption function enc() and a corresponding decryption function dec(). A message M is encrypted under a password P as,

$$\text{Ciphertext } C = \text{enc } P(M)$$

The message can be decrypted as,

$$M = \text{dec } P(C)$$

Decrypting with an incorrect password $P' \neq P$, dec P(C) outputs an error message, which makes it clear that we have entered an incorrect password. [12]

The traditional approach thus notifies the unsuccessful attempt by giving an invalid-looking output. Thus the adversaries making password-guess attempts against a PBE cipher text knows when they have decrypted successfully.

III. Password Managers

With the increasing development of hacking techniques, the only option for true security is a strong password with random characters, such as "$^uH0!jkR". Such random passwords are harder to crack. At the same time, they are difficult to memorize especially when the user uses different passwords for every account the user has. The Password Managers thus help the user by securely storing all the passwords and their associated accounts. [12] Password Managers are very helpful as they combine security with convenience by storing all your credentials in one place, allowing you to use strong, complex passwords that user don't have to remember.

In a Password Manager, the confidential data of the user is stored in an encrypted file. The encrypted data can be accessed by only one password which is the user selected master password. Hence the user needs to remember only a single password, the one used to unlock the so-called vault. A few most popular password vaults available are 1Password, Dashlane, KeePass, and LastPass. [3]

A security researcher and professor at Carnegie Mellon University, Lujo Bauer, say that "Password Managers are not a magic pill, but they offer a much better combination of security and convenience to the users that they have without them". [13] Though the Password Manager greatly reduces the burden of remembering the passwords, they themselves need to be super secure. If an encrypted vault of passwords is leaked, then offline brute force attacks can lead to compromise of the passwords stored in the vault.

3.1 Honey Encryption

Ari Juels and Thomas Ristenpart, proposed the technique of Honey Encryption which is best suited where encrypted data is obtained from passwords. [14] The main motive of Honey Encryption is to baffle the attacker, making it difficult to know when he has guessed the right key or password. In a conventional cryptography technique, decryption of a ciphertext with the wrong key presents the attacker with an invalid message, thus the attacker can find and eliminate wrong keys via a brute-force attack. However, if a data is encrypted using Honey Encryption, the output of decryption under the wrong choices of an attacker will give plausible looking invalid data, thus misguiding the attacker. For example, if an attacker tries to get a mobile number by making 200 attempts, then for all the 200 attempts he will be getting 200 fake numbers. As each decryption will be looking plausible like others, the attacker in no manner will be able to distinguish the correct and incorrect data.

Honey encryption comes along with a concept called distribution-transforming encoder (DTE), a model of message distribution. [14] Honey encryption manages plaintext space via DTE. Let the probability distribution over the message space be p over the message M. The distribution transforming encodes the message M as a Q bit seed $S \in \{0, 1\}Q$ and decodes the message by inverse DTE method, decode (S) =M. DTE The internal structure of the HE includes DTE encryption and DTE decryption that defines the complete functioning of Honey Encryption.

```
$H \leftarrow \mathrm{Enc}(K, M)$
$S \leftarrow \$\ encode\ (M)$
$RS \leftarrow \$\ \{0, 1\}\ n$
$S' \leftarrow H(RS, K)$
$C \leftarrow S' \oplus S$
```

```
$H \leftarrow \mathrm{Dec}(K, (RS, C))$
$S' \leftarrow H(RS, K)$
$S \leftarrow C \oplus S'$
$M \leftarrow decode(S)$
Return $M$
```

Figure 1 Honey Encryption and Decryption with DTE

Table 1 Honey Encryption Symbols

H	Cryptographic Hash Function
S	Seed
M	Message
K	Key
RS	Random String
C	Cipher text

3.1.1 Related Word on Honey Encryption

Abiodun Esther Omolara, Aman Jantan, Oludare Isaac Abiodun and Howard Eldon Poston, proposed an approach [15] where they attempted to apply Honey Encryption to natural language message to produce convincing fake messages for documents such as Email. This paper [16] presents a tool named “GenoGuard”, for providing strong protection for genomic data using the concept of Honey Encryption. It proved that, decryption under any key will yield a plausible genome sequence. Another reference paper [17] presents a HE-based statistical scheme and designed a Honey Chatting application, which is said to be robust to eavesdropping. Fancy Arora in the paper [18] explained how Honey Encryption can be used to provide security in cloud computing in addition to other concepts like Homomorphic Encryption and Elliptic Curve Cryptography.

3.2 One Time Pad

One Time Pad (OTP) is said to be a system with perfect secrecy. OTP also known as Vernam Cipher is a provably secure cryptosystem, developed by Gilbert Vernam in 1918. For the encryption in OTP, plaintext characters are XOR with one time pad key characters:

$$C = (P \oplus K)$$

Where; P:plaintext characters, K:key characters, C:ciphertext character.

For decryption cipher is Xor-ed with the message,

$$P = (C \oplus K)$$

Encryption in this algorithm is to combine each character in the plaintext with the characters on the keys. Therefore, the key length must be at least equal to the length of the plaintext. Vernam-cipher is said to be “Perfectly Secure” i.e.: “Shannon Secure”, as stated by Claude Shannon in – “Communication Theory of Secrecy Systems”, [19] because no information of the plaintext is provided by the generated ciphertext. It can be perfectly secure or unbreakable only if the following rules are followed:

- Key must be truly random
- Length of the key must be as long as the plaintext
- Key must be kept secret

Along with perfect secrecy, OTP is said to be immune to brute-force attacks as compared to the conventional symmetric encryption. [20] Even if few parts of the plaintext are known to the attacker, brute force attacks would not be able to break the system, as the attacker won’t be able to gain any knowledge about that part of the key which is required for the decryption of remaining message. The known parts will reveal only the parts of key that corresponds to them, and to be noted, all parts of the key are independent. No part of key is dependent on other; they strictly correspond to one-to-one basis.

IV. Proposed System

Users believe that their information is secure if they store it in a Password Manager and lock it. But once the master password becomes available to the attackers, they can decrypt the Password Manager database and all your data gets compromised, the login credentials and the other stored documents. Passwords are very vulnerable to brute force attack. The brute attack can be prevented by utilizing or requiring strong passwords, asking security questions, tricking the attack software etc. and a few other defense strategies.

In the proposed methodology, a Password Manager is developed using the concept of Honey Encryption and One-time pad. Log in to the system with the correct master password will give the correct information. But if the invalid master password is used then plausible looking false data will be available. In this system, an attacker using an invalid password will get the invalid data which are the decoys, looking similar to the original one, thus misguiding the attacker.

The proposed methodology consists of the following phases:-

- User registers and sets the master password which is system generated.
- Master Password is then Honey Encrypted with the help of mapping.
- Additional security is added by encrypting the HE cipher with OTP technique converting it to Vernam cipher, then storing into database.
- To login, user will enter a Master password which is used to decrypt Vernam cipher, which is again decrypted by HE to get the original master password.
- If the user entered master password matches the decrypted password, original data gets visible.
- If the user entered Master Password does not match the decrypted password, decoys get visible.

The following diagram shows the complete process of the proposed system with honey encoding and Vernam Cipher.

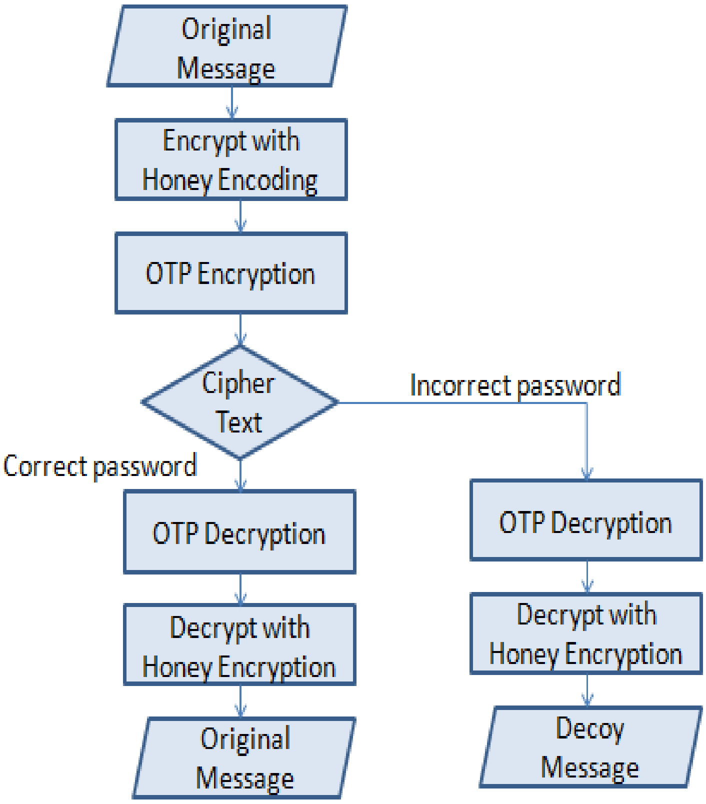

Figure 2 Flow Chart of Proposed System

V. Comparative Analysis

The reference paper, [21] shows the comparative analysis of HE with AES and HE with Blowfish. As per the paper, the authors said that the execution time of Honey Encryption with Blowfish has less execution time as compared to Honey Encryption with AES.

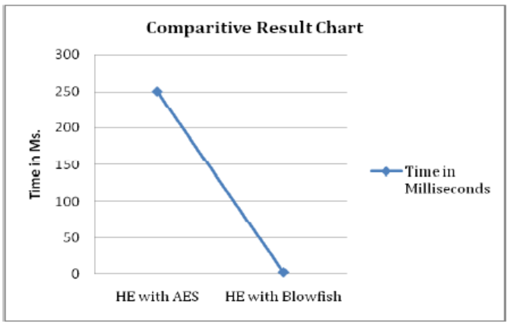

Figure 3 Execution time chart of He with AES and HE with Blowfish

We extended this comparative analysis [21] to Honey Encryption with OTP. It can be stated that the time taken by OTP is almost the same as Blowfish.

Table 2 Comparative Analysis of Execution Time

Techniques	Overall Execution time to Encrypt and Decrypt (in milliseconds)
Honey Encryption with AES	250
Honey Encryption with Blowfish	2

Honey Encryption with OTP	> = 2

We tried to execute Honey Encryption with OTP with variable lengths of password (starting with 8 characters in length) and we got variable timings. The minimum time it took was 2 milliseconds.

VI. CONCLUSION AND FUTURE WORK

We tried to design a secure framework by implementing the proposed concept of Honey Encryption by Ari Juels and Thomas Ristenpart to Password Manager. Using a secure Password Manager can help you use strong passwords that are less likely to get hacked through brute force attacks. The user won’t have to remember them all and can securely store them all in one place for convenience. Also, an additional level of security is added by combining Honey Encryption with Vernam cipher. When combining HE with AES, Blowfish, and OTP, it is observed that the time take by OTP is almost same with Blowfish.

In using the Honey Encryption technique, the challenging part is to create the decoy space, which affects the security provided by HE. Though HE can fool the attackers but its capability of protecting sensitive data varies for different applications as the message spaces vary. In future work, a tool can be implemented for decoy creation and also such that decoys are not duplicated.

REFERENCES

[1] Why You Should Use A Password Manager. [Online]. Available: https://www.popularmechanics.com/technology/security/a26629/use-password-manager/ .

[2] The 18 biggest data security breaches of 21st century | CSO Online. [Online]. Available: https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html.

[3] RockYou hack exposes names, passwords of 32M accounts | Computerworld. [Online]. Available: http://www.computerworld.com/article/2522045/security0/rockyou-hack-exposes-names--passwords-of-30m-accounts.html.

[4] Yahoo Hacked, 45,000 passwords posted online – CNN.com. [Online]. Available: http://edition.cnn.com/2012/07/12/tech/web/yahoo-users-hacked/

[5] Number of Adobe Accounts Hacked Now to 150M, Check Yours. [Online]. Available: http://petapixel.com/2013/11/07/number-adobe-accounts-hacked-now-150m-check/

[6] More than 6 million LinkedIn passwords stolen. [Online]. Available: http://money.cnn.com/2012/06/06/technology/linkedin-password-hack/

[7] Best Password Manager 2019 [Online]. Available: https://www.tomsguide.com/us/best-password-managers,review-3785.html.

[8] Pritesh N. Patel, Jigisha K. Patel and Paresh V. Virparia, “A Cryptography Application using Salt Hash Technique”, Volume 2, Issue 6, June 2013

[9] Yahoo Hacked and How to Protect Your Passwords. [Online]. Available: http://www.forbes.com/sites/jameslyne/2014/01/31/yahoo-hacked-and-how-to-protect-your-passwords/

[10] Hack Like a Pro: How to Crack Passwords, Part 1 (Principles & Technologies). [Online]. Available: http://null-byte.wonderhowto.com/ how-to/hack-like-pro-crack-passwords-part-1-principles-technologies-0156136/

[11] Hanna Willa Dhany, Fahmi Izhari, Hasanul Fahmi, Tulus, Sutarman, “Encryption and Decryption using Password Based Encryption, MD5, and DES”, ASSEHR, Volume 141, 2017.

[12] Ambarish Karole, Nitesh Saxena, and Nicolas Christin, “A Comparative Usability Evaluation of Traditional Password Managers”

[13] Everything You Need to Know About Password Managers – Consumer Reports. [Online] Available: https://www.consumerreports.org/digital-security/everything-you-need-to-know-about-password-managers/

[14] Ari Juels, Thomas Ristenpart, “Honey Encryption- Encryption beyond Brute Force Barrier” IEEE Security and Privacy July/August 2014

[15] Abiodun Esther Omolara, Aman Jantan, Oludare Isaac Abiodun and Howard Eldon Poston, “A Novel Approach for the Adoption of Honey Encryption to Support Natural Language Message”, International MultiConference of Engineers and Computer Scientists, 2018

[16] Zhicong Huang, Erman Ayday, Jacques Fellay Jean-Pierre Hubaux, Ari Juels, “GenoGuard: Protecting Genomic Data against Brute-Force Attacks”, IEEE Symposium on Security and Privacy, 2015

[17] Joo-Im Kim, Ji Won Yoon, “Honey chatting: A novel instant messaging system robust to eavesdropping over communication”, IEEE International Conference on Acoustics, Speech and Signal Processing, 2016

[18] Fancy Arora, “Security in Cloud Computing using Honey Encryption”, ResearchGate, May 2017.

[19] Claude E. Shannon, “Communication Theory of Secrecy Systems”

[20] One-time pad – Wikipedia. [Online] Available: https://en.wikipedia.org/wiki/One-time_pad#Perfect_secrecy

[21] Rasmita Sahu, Mohd Shajid Ansari, “Securing Messages from Brute Force Attack by Combined Approach of Honey Encryption and Blowfish”, IRJET, Volume 04, Issue 09, September 2017.