HONEYWORDS PASSWORD PROTECTION

Prachi Tawde[#1], Ronald Nazareth[#2], Meet Waghela[#3], Melvita Gonsalves[#4]
[#]*Information Technology, Mumbai University*
[1]prachi.xie@gmail.com
[2]ronaldnazareth2@gmail.com
[3]meetwaghela97@gmail.com
[3]gonsalves.melvita@yahoo.com

Abstract— **Username is a distinctive identity for each user. Username is used to find a particular user uniquely among several others. In any security system a username-password checking is very important. So, to safeguard the password from an intruder we implement, for each user account a valid password which is then converted to honeywords and are appended to the database. The valid password is also hashed and stored to the database to avoid further security issues. Honeywords is basically fake passwords which are associated to every user's account. If we choose the honeywords properly the intruder cannot guess whether he has a real password or a honeyword. Here if the attacker gets the stored hash file and tries to decrypt it, he will get a honeyword instead of a valid password and if the attacker tries to enter a honeyword an alarm is set off and user is made aware about the threat to his account. Moreover, if the attacker inputs a password which is not a honeyword, after three attempts the attacker is allowed to login but is directed to a decoy page (dummy page). In this study we will examine in detail about the honeywords system and present some comments to focus on the weak points. We also focus on pragmatic password, reducing the storage cost of password, and an alternate way to choose the distinct password from existing user passwords.**

Keywords— **honeywords; username-password; hashing login.**

I. INTRODUCTION

Many real world systems select password based encryption algorithm and so the maintenance of password file in the database becomes a very important challenge in different areas. Hence, the password files play a significant role in millions of users and organizations such as Yahoo, RockYou, LinkedIn, eHarmony and Adobe since the disclosure of password makes the user aim of many possible cyber-attacks.[2],[3],[4] Hashing and salting algorithms are used by many organizations to safeguard the password files. For example, the SHA-1 algorithm without a salt were being used by LinkedIn and the eHarmony passwords were also stored using unsalted MD5 hashes.[2],[5] Attacker can easily get the password file by using simple password cracking skills.

Honeyword generation method is one of the techniques to protect against the stolen password file from attackers. In this approach, the list of passwords which contains the real user's password along with honeywords from honey generation algorithm are stored by the system. In order to get access to the data, the attacker who steal databases of user logins and passwords only have to predict (or guess) a single correct password. The attackers gets the confidential passwords when the database or password file becomes readable by using brute force attack. To rush up the process, attackers have ingress to advanced software that can send thousands of passwords every minute to applications in an endeavor to decode the data. Using excessive speed, multicore processors also reduces the time it can take to crack encryption. To store the user's real password with honeyword and the classification of honeywords and user's real password is the fundamental innovation of the honeywords generation scheme. The contemporary existing honeywords generation method has inadequacy in password storage and old password management problem. [2]

So we initiate the following facts to subdue the vulnerability of the existing system.

- We initiate honeywords generation methods to lessen storage overhead problem, typo safety problem and our initiated method is flatness.
- Moreover we initiate hashing and salting algorithm with very less time complexity for safeguarding stored passwords. [6]

II. PROBLEM DEFINITION

A. *Existing System*

Many password schemes with different degrees of resistance to shoulder surfing have been proposed, seeing that most users are more familiar with textual passwords than pure graphical passwords, proposed a text-based shoulder surfing resistant graphical password scheme. The user has to mix his textual password on the login screen to get the session password. However, the login process of scheme is complex and tedious. And then, several text based shoulder surfing resistant graphical password schemes have been proposed. [8],[9] Unfortunately, none of existing text-based shoulder surfing resistant graphical password schemes is both secure and efficient enough.

Existing System Disadvantages

1. Existing system have less security.
2. The correct combination of characters breaks the password.
3. This method would possess so much difficulty to adopt for the non-math oriented people.
4. Existing system can easily get hack.
5. Username & password are easily writable.
6. Easy to share username and password.

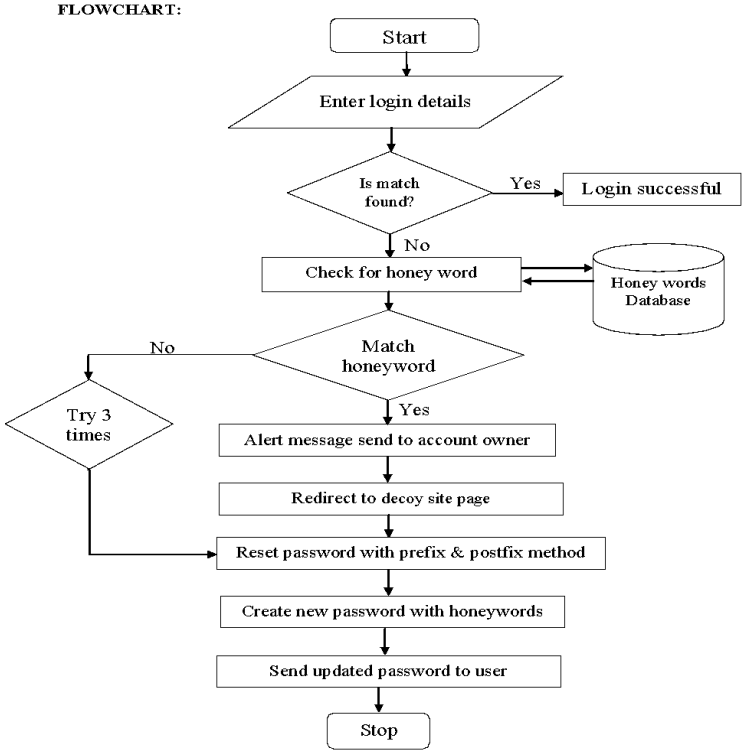
7. The session passwords does not provide better security against brute force attacks.
8. Authentication results are slow and not accurate.

B. Aim and Objectives

The idea is to insert false passwords called as honeywords into the system. These fake words or honeywords are associated to each user's account. All the honeywords along with the valid password is stored in the database. If the attacker gets the password list he cannot be sure whether the password is fake or genuine. In this system the cracked password file can then be detected by the system administrator, if a login attempt is made using a honeyword by the attacker. A dedicated server will be made to detect the use of honeywords while login. The database containing honeywords will be appended to the server to distinguish between valid passwords and honeywords. The honeywords system does not prevent the hacker from entering the system but it will alert the administrator about the breach that has occurred.

We are generating honeywords. Honeywords are generated from the real password and in case any hacker tries to hack into the account by guessing the password, the main user is sent alerts in form of a mail or some message so he knows that somebody is trying to log into his or her account. The hacker is given access when he enters a honeyword as the password, but he is shown decoy files and the real files are safe with the user. Following are Modules used in the proposed system.[10],[11]

FLOWCHART:



Proposed System Advantages

1. The system is easy to learn for users and familiar with textual password so the user can easily and efficiently complete the login process
2. Flexibility: It can be easily integrated with other authentication systems
3. A scenario offers as almost unlimited combination of possibilities.
4. It is more efficient password system against attack like shoulder surfing or brute force attack for the password.
5. This scheme can be easily used by any type of user which widens the scope of applicability of our scheme.
6. The session passwords provide better security against brute force attacks as password changes for every session.
7. This scheme will be usable anywhere and at any time with a low error rate as well as a faster authentication result.
8. The proposed scheme will benefit from the argument that people are better in recognizing images. Therefore, pass images should be easy to remember.
9. The system will provide a strong line of defense against shoulder surfing brute force, intersection and educated guess attacks.

III. HONEYWORDS GENERATION METHODS

Honeyword generation algorithms can be classified into two types.

A. Legacy-UI Procedures

The password-change UI is consistent i.e. it takes the same password entered by the user for honeyword generation. [7,][12]To generate the honeywords, Chaffing by tweaking concept is used. In this method, the user password is given as an input to the generator algorithm, it tweaks the selected positions of the correct password to produce the honeywords. Each character of the selected positions is replaced by randomly chosen character and a set of honeywords are formed. Consider a method Gen(m;d) in generator algorithm wherein d is the number of characters to be replaced. For example if tweaking last 2 positions

from the real password, the d = 2. If user password is abc216 and d = 2, then honeywords abc289, abc245 may be generated. [7],[12] Generation algorithm should be such that the real password should not be identifiable from honeywords.

B. Modified-UI Procedures

The password-change UI is altered to allow better honey-word generation. The user's actual password is altered to end with a randomly chosen value to form a new user password. Take-a-tail method is an example of this category. For example, abcd23 is user's entered password then system generates '@12' as a tail. So now user's new password becomes abcd23@12. Above method has highest security standard but is very poor in usability standard. It is very difficult for user to remember the system generated information for his different accounts.[7],[12]

C. Hybrid Generation methods

- Combining several methods can result in better honeywords
- Combine both legacy-UI techniques:
- Require the user to use digits at the end of the password
- Chaffing-with-a-password to generate new random words
- Chaffing-by-tweaking-digits on all words

abacad513	snurfle672	zinja897
Abacad941	snurfle134	zinja320
abacad004	snurfle845	zinja461
abacad752	snurfle772	zinja389

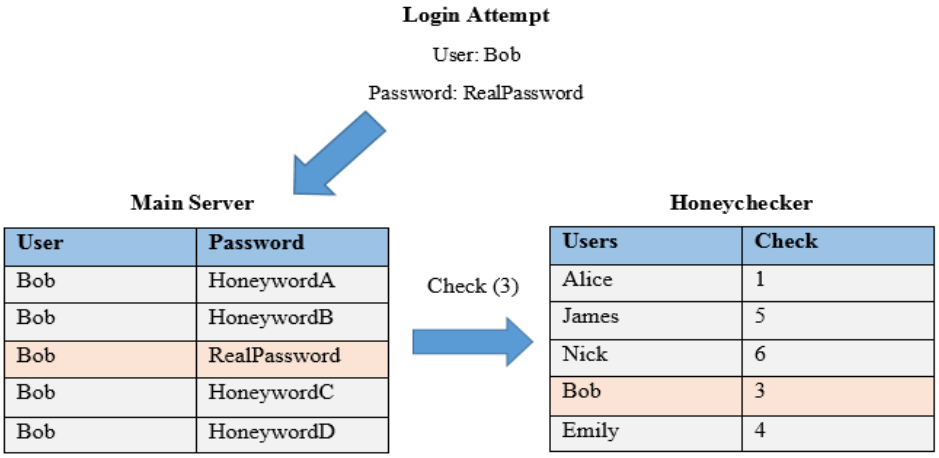
IV. MODULE DESCRIPTION

A. Registration

Here the user is going to register into system. After registration the user's password will be hashed and additional honeywords will be generated and stored into the table.

B. Login

Here user is going to Login into the System. If password matches with password provided by the user at the time of registration, then user is allowed to login.



User	Password
Bob	HoneywordA
Bob	HoneywordB
Bob	RealPassword
Bob	HoneywordC
Bob	HoneywordD

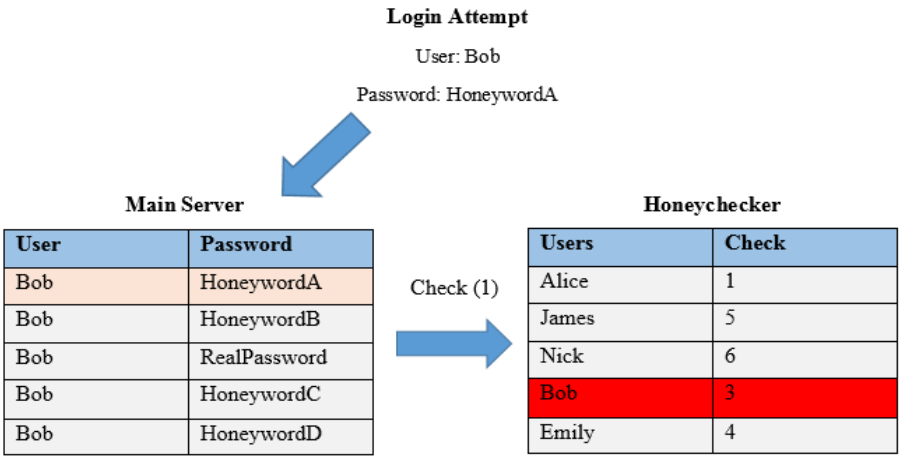
Users	Check
Alice	1
James	5
Nick	6
Bob	3
Emily	4

C. Honeyword Generator

In this phase honeywords are generated by using chaffing with tweaking concept. Here honeywords are generated to ensure security of the system. Total 20 honeywords will be generated randomly and are stored in the same file.20 honeywords as in there are 19 honeywords generated randomly and 1 is the actual password which the user will enter while logging into the system. So total 20 honeywords are generated and this honeywords are stored randomly. If any hacker is trying to get access into the valid user's account then he has to give 3 attempts to get access to the system. If any of the password entered by hacker matches with the honeyword then he will get access to the system but system will show him the decoy files (dummy files) which are fake files. Though the hacker gets fake files but the notification is sent to the valid user by mail that someone has tried to login into your account.

D. Hacker

Here the hacker will try to breach the system. If the attacker tries to login into the system and he enters a honeyword he will be redirected the decoy page (dummy page). Similarly if the attacker tries to enter a combination of different passwords which are not honeywords, after three attempts he will be allowed to login but will be redirected to the decoy page (dummy page).



E. Log Creation

Log creation is done for each user action to the system and is stored into the database.

F. Notification

A Notification Program is an additional service offered in our proposed System to notify the legitimate user about the hacker trying to access a file without his permission. Here the automated system notifies using techniques like SMS and Email. The permission to access this service is offered to all the users. The advantage of implementing our proposed service is to enhance the overall performance of the system and provides user satisfaction.

V. CONCLUSION

Password security has always been a realm of active research. Honeyword based authentication have proved preferable results in this domain. The big dissimilarity between the traditional methods and when honeywords are used is that a victorious brute-force password attack does not gives the attacker reliance that he can log in into system victoriously without being noticed. Honey encryption technique using honeywords becomes very interesting and challenging technique in security area because it can give various advantages over password based schemes.

In this paper, we present a novel honeywords generation method which has much lesser storage space and it can also lessen the majority of the drawbacks of the existing honeywords generation techniques. The proposed hashing and salting technique has much lesser time complexity than Advanced Encryption Standard (AES), Data Encryption Standard (DES) or any other current technique. The practice of decoy data mechanism will protect the confidential data of the authorized users from the attacker. In honeyword based authentication approach, it is sure that the hacker will be detected. The main goal of project is to confirm whether data access is permitted or not when irregular information entry is noticed. Misleading the hacker with lure data safeguards from the exploit of the user's real data. The admin keeps the data of the tracked IP's with them and uses them to obstruct entry on their network. Use of honeywords is very helpful and works for each user's account. In future, we will apply this honeywords generation method and hashing and salting algorithm in actual application system that is required for security such as message transmission process.

REFERENCES

[1] Juels, Ari, and Ronald L. Rivest. "Honeywords: Making password-cracking detectable." Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security. ACM, 2013.

[2] Moe, Khin Su Myat, and Thanda Win. "Improved hashing and honey-based stronger password prevention against brute force attack." Electronics and Smart Devices (ISESD), 2017 International Symposium on. IEEE, 2017.

[3] Mirante, D and Justin,C, "Understanding Password Database Compromise", Technical Report TR-CSE-2013-02, Department of Computer Science and Engineering Polytechnici Institute of NYU, 2013.

[4] Vence, A, "If your password is 123456, just make it hackme", The New York Times 20, 2010.

[5] Brown,K , "The danger of weak hashes", Technical report, SANS Institute InfoSec Reading Room, 2013.

[6] Arias, D., Arias, D. and Kogan, E. (2018). Adding Salt to Hashing: A Better Way to Store Passwords. [online] Auth0 - Blog. Available at: https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/ [Accessed 20 Oct. 2018].

[7] Erguler, Imran. "Achieving flatness: Selecting the honeywords from existing user passwords." IEEE Transactions on Dependable and Secure Computing 13.2 (2016): 284-295.

[8] Ankush, Doke Ashvini, and Shaikh Saddam Husain. "Authentication Scheme for Shoulder surfing using Graphical and Pair Based scheme." International Journal 2.10 (2014).

[9] Ms. Shraddha Gajare, Mrs. Vaishali Londhe, Mrs. Nilima Nikam , " A Theoretical Approach to Simple 4-D Authentication Scheme Resistant to Shoulder Surfing Attack" , International Journal of Application or Innovation in Engineering & Management (IJAIEM) , Volume 4, Issue 10, October 2015 , pp. 014-018 , ISSN 2319 - 4847.

[10] "Detecting Data Breaches with Honeywords." InfoSec Resources, 2 July 2018, resources.infosecinstitute.com/detecting-data-breaches-with-honeywords/.

[11] JOURNAL OF INFORMATION, KNOWLEDGE AND RESEARCH IN COMPUTER ...www.ejournal.aessangli.in/ASEEJournals/CE174.pdf.

[12] Naik, Ms Komal, Varsha Bhosale, and Vinayak D. Shinde. "Generating Honeywords From Real Passwords With Decoy Mechanism."