Data Security using Honeywords

Dipali Dhumal[1] Prof. Shyam Gupta[2]
[1]Student [2]Professor
[1,2]Department of Computer Engineering
[1,2]Siddhant College of Engineering Sudumbare, Pune, India

Abstract— Now a days, password files has a lot of security problem because of affected millions of users and many companies. Password file is generally stored in encrypt format, if a password file is stolen, then using the password cracking techniques and decryption technique it is easy to find most of the plaintext and encrypt passwords. For trouble shoot this here we create the honeyword password i.e. a False password by using a perfectly flat honeyword generation method and try to attract unauthorized user. Hence that time we detect the unauthorized user. Here also protect the original data from unauthorized user. In propose work, it generates honeyword passwords i.e. untrue passwords by using flawlessly honeyword generation way, and also it tries to invite prohibited or unauthorized users. Also in proposed going to authenticate the users using hashing algorithm which gives us more correctness to select authenticate users. Hence notice the illegal users. Here also protects original info from illegal users using other file format.

Key words: Honeywords, Honeypot, Login, OTP, Authentication, Password Cracking, Passwords, Decoy, Documents

I. INTRODUCTION

In many 1companies and software industries store their data in database. The entry point of a system which is required user name and password are stored in encrypt form in database. Once a password file is stolen, by using the password cracking technique it is easy to find plaintext passwords for avoiding it, there are two issues that should be considered to overcome these security problems: first passwords protected and secure using the appropriate algorithm, second point is that a secure system should detect the entry of unauthorized user in the System. In the proposed system we focus on the honeywords i.e. fake passwords and accounts. The administrator purposely creates user accounts and detects a password, if any one of the honeypot passwords get used it is easily to detect the admin. If each user incorrect login attempts with some passwords lead to Honeypot accounts, i.e. malicious behavior is recognized in proposed system , We create the password in plane text, and stored it with the fake password set. Using honeyword approach and give some remarks about the security of the system. When unauthorized user attempts to enter the system and get access the database, the alarm is triggered and gets notification to the administrator, since that time unauthorized user get decoy documents i.e. Fake database.

II. MOTIVATION

There are many motivational scenarios relating to passwords, including the following:

A. Stolen file of password hashes

An adversary is somehow able to steal the file of password hashes, and solve for many passwords using bruteforce computation. He may more generally be able to steal the password hashless on many systems or on one system at various times.

B. Easily Guessable Passwords

A substantial number of users choose passwords so poorly that an adversary can successfully impersonate at least some users of a system by attempting logins with common password.

C. Visible passwords

A user's password is compromised when an adversary views it being entered (shoulder-surng), or an adversary sees it on a yellow stickie on a monitor. A one-time password generator 3 such as RSA's SecurID token provides good protection against this threat.

III. EXISTING SYSTEM

The decoy password i.e honey words to detect attacks against hash password database. For each user account the legitimate password stored in form of honey words. If attacker Attack on password i.e honey words it cannot be sure it is real password or honeyword. It is much easier to crack a password hash with the advancements in the processing unit technology. Entering with a honeyword to login will trigger an alarm notifying the administrator about a password file breach and trigger an alarm.

IV. PROPOSED SYSTEM

In proposed system, the generated honeyword passwords i.e. untrue passwords is generated by using hybrid generation. It also tries to invite prohibited or unauthorized users with the questions asked during the authorization process. In proposed work it will authenticate the users using hashing algorithm which gives us more correctness to select authenticate users. Hence this project notices the illegal users. This project is using SHA-1 algorithm for the authentication process for the users. Here below is the detail algorithm.

A. System Architecture

The fig shows proposed system work flow that gives fake document to unauthorized user .The administrator purposely creates user accounts and detects a password disclosure, if any one of the honeypot passwords get used it is easily to detect the admin. According to the study, for each user incorrect login attempts with some passwords lead to Honeypot accounts, i.e. malicious behavior is recognized. In proposed system, we create the password in plane text, and stored it with the fake password set. We analyze the

honeyword approach and give some remarks about the security of the system. When unauthorized user attempts to enter the system and get access the database, the alarm is triggered and gets notification to the administrator, since that time unauthorized user get decoy documents. I.e. fake database.

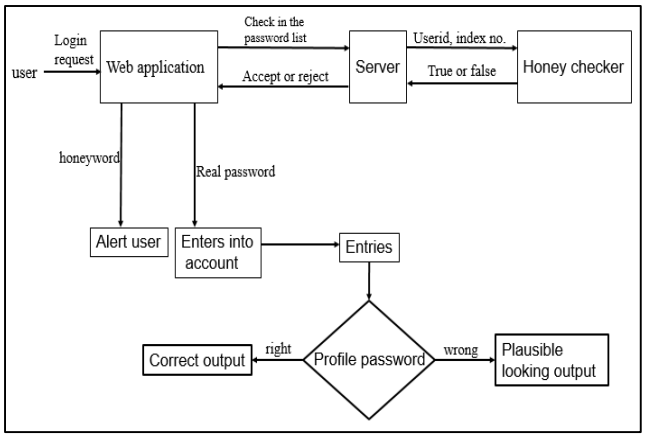

Fig. 1: System Architecture

The contribution of our approach is twofold. First, this method requires less storage compared to the original study. Second, in the previous sections we argue that effectiveness of the honey word system directly depends on how Gen() flatness is provided and how it is close to human behavior in choosing passwords. Within our approach passwords of other users are used as the fake passwords, so guess of which password is fake and which is correct becomes more complicated for an adversary.

B. Models Used

1) Communication Module

- Communication will be performed with the help of socket programming.
- Communication Module will communicate with the Requester and the Provider.
- Requester-Emulator for Cloud Computing.
- Provider-Servers connected via LAN.

2) Security Module

- Security Module will be divided in Authentication Module and Access Control List.
- Authentication Module will be responsible for Authentication services.
- The Access Control List will contain the list of users who will be allowed to perform login. The client can perform a login as administrator or either as a regular user.

3) Decoy Documents

- Validating whether data access is authorized when abnormal information access is detected.
- Confusing the attacker with bogus information.

4) Honey word Generation Methods and Discussions

Categorize the honey word generation methods into two groups. The first category consists of the legacy-UI (user interface) procedures.Second includes modified-UI procedures whose password-change UI is modified to allow better password/honey word generation.Take-a-tail method is given as an example of the modified-UI procedures category. According to this a randomly selected tail is produced for the user to append this suffix to her entered password and the result becomes her new password. For instance, let a user enter password games01, and then system let propose ’413’ as a tail. So the password of the user now becomes games01413. Although this method strengthens the password, to our point of view, it is impractical – some users even forget the passwords that they determined. Therefore in the remaining parts, the analysis that we conducted is limited with the legacy-UI procedures.

5) Hybrid Method

Another method is using combining the strength of different honeyword generation methods, e.g. chaffing-with-a-password-model and chaffing-by-tweaking digits. By using this technique, random password model will yield seeds for tweaking-digits to generate honeywords. For example let the correct password be apple1903. Then the honeywords angel2562 and happy9137 should be produced as seeds to chaffing-by-tweaking digits. For t = 3 and k = 4 for each seed, the sweet word table given below may be attained:

V. ALGORITHM

A. Proposed Algorithm

1) Step1: Start
2) Step2: Enter the user name
3) Step3: if (username!=true) go to step8
4) Step4: Enter the password
5) Step5: if(password!=true) go to step8
6) Step2: Enter the answer of Question
7) Step3: if(answer!=true) go to Step8
8) Step6: Enter the OTP
9) Step7: if(OTP!=true) go to step8
10) Step8: Create the honeyword i.e; false password using Chang-by-tweaking Algorithm.
11) Step 9: If Step 8 followed, decoy document delivered to used and user is attacker else original document delivered to user and user is authenticated.

B. DES Algorithm

The increasing volume, value and confidentiality of these records regularly transmitted and stored by commercial and government agencies has led to heightened recognition and concern over their exposures to unauthorized access and use. This misuse can be in the form of theft or defalcations of data records representing money, malicious modification of business inventories or the interception and misuse of confidential information about people. The need for protection is then apparent and urgent.

VI. MATHEMATICAL MODEL

A. Set Theory

1) U is the set of Users U= u1, u2, u3-
   Where U is main set of Users like u1, u2, u3…ui
2) g is the set of password, g= g1, g2, g3
   Where g is set of Cluster Head g1, g2, g3.., gi
3) X is Honey Set correspond to useers U.
4) H(g) is the Hash value function correspond to the password.

System firstly checks whether entered password, g, iscorrect for the corresponding username u.

To accomplish .this, firstly the Xi of the corresponding ui is attained from the F1 file. Then, the hash values stored inF2file for the respective indices in Xi are compared with H(g) to find a match.

– If a match is not obtained, then it means that g is neither the correct password, nor one of the honeywords, i.e. login fails. On the other hand, if H(g) is found in the list, then the main server checks whether the account is a honeypot. If it is a honeypot, then it follows a predefined security policy against the password disclosure scenario.

VII. RESULT

Fig. 2: User login

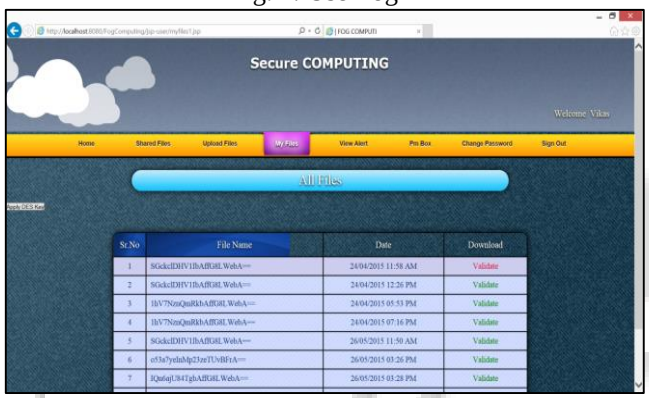

Fig. 3: Validation of file

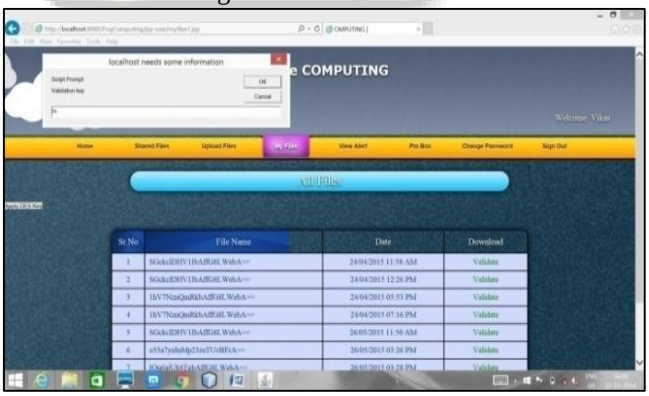

Fig. 4: Validation key for authentication using SHA

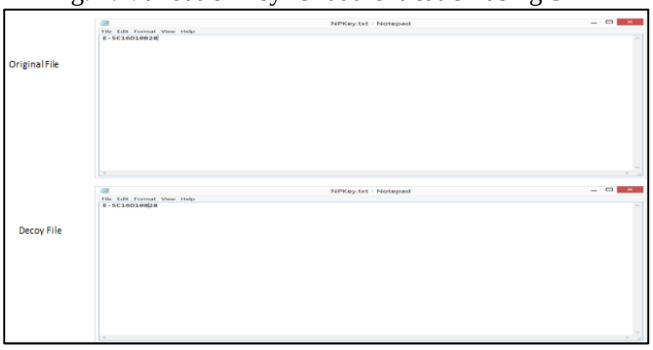

Fig. 5: Download file fake file

VIII. CONCLUSION

In this study, analyzed the security of the honey word system and addressed a number of flaws that need to be handled before successful realization of the scheme. In this respect, we have pointed out that the strength of the honey word system directly depends on the generation algorithm, i.e. flatness of the generator algorithm determines the chance of distinguishing the correct password out of respective sweet words. Another point that we would like to stress is that defined reaction policies in case of a honey word entrance can be exploited to realize a DoS attack. This will be a serious if honey word given the respective password is not negligible. To combat such a problem, also known as DoS resistance, low probability of such an event must be guaranteed. This is achieved by employing unpredictable honey words to minimize this risk. Hence, we have noted that the security policy should keep balance between DoS vulnerability and effectiveness of honey words. Furthermore, we have demonstrated the weak and strong points of each method.

REFERENCES

[1] Imran Erguler,"Achieving Flatness: Selecting the Honeywords from Existing User Passwords", DOI 10.1109/TDSC.2015.2406707, IEEE Transactions on Dependable and Secure Computing.
[2] D. Mirante and C. Justin, "Understanding Password Database Compromises," Dept. of
[3] Computer Science and Engineering Polytechnic Inst. of NYU, Tech. Rep. TRCSE-2013-02, 2013.
[4] A. Vance, "If Your Password is 123456, Just Make It Hackme," The New York Times, vol. 20, 2010.
[5] K. Brown, "The Dangers of Weak Hashes," SANS Institute InfoSec Reading Room,Tech. Rep., 2013.
[6] M. Weir, S. Aggarwal, B. de Medeiros, and B. Glodek, "Password Cracking Using Probabilistic Context-Free Grammars, in Security and Privacy, 30th IEEE Symposium on. IEEE, 2009, pp. 391405.

