Improving Password Security using Honey words and Honey Encryption

Rupesh G. Mahajan[1] Amol I. Parde[2] Anand N. Varale[3] Harshal S. Kamble[4] Swapnil N. Jagtap[5]
[2,3,4,5] Students, [1]Professor
[1,2,3,4,5] Computer Engineering
[1,2,3,4,5] Dr. D.Y. Patil Institute of Technology, Pimpri Pune, India

Abstract : Security is important , but it's easy to overlook the little things like having effective passwords , During this paper we discuss about the honey word mechanism to detect an adversary who attempts to login with cracked passwords. New password is the combination of existing user passwords called honey words. Fake password is nothing but the honey words basically, for each username a set of sweet words is constructed such that only one element is the correct password and the others are honey words (decoy passwords). Hence, when an adversary tries to enter into the system with a honey word, an alarm is triggered to notify the administrator about a password leakage. 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. honeys words it cannot be sure it is real password or honey word.

Index Terms **- Authentication Processing, Password Cracking, Honeypot, Honey Indexing, Honey Sets**

I. INTRODUCTION

In this study there are three security concepts that should be considered to overcome these problems: First passwords must be protected by taking appropriate precautions and storing with their hash values computed through complex mechanisms. Hence, for an adversary it must be hard to invert hashes to acquire plaintext passwords. The second point is that a secure system should detect whether a password file disclosure incident happened or not to take appropriate actions. In this study, we focus on the latter issue and deal with fake passwords or accounts as a simple and cost effective solution to detect compromise of passwords. Third one is that server should send a message to a secure server which is called "honey checker", for the user and the sweet word. The honey checker will determine whether the submitted word is a password or a honey word. If a honey word is submitted, then it will raise an alarm or take an action that is previously chosen. The honey checker cannot know anything about the user's password or honey words. It maintains a single database that contains only the order of the true password among the user's sweet words.

HONEYWORDS GENERATION METHODS:

A. Chaffing with Toughnut

In this method, the system intentionally injects some special honeywords, named as tough nuts, such that inverting hash values of those words is computationally infeasible, e.g. fixed length random bit strings should be set as hash value of a honeyword. Moreover, it is noted that number and positions of tough nuts are selected randomly. By means of this, it is expected that the adversary cannot seize whole sweetword set and some sweetwords will be blank for her, thereby deterring the adversary to realize her attack. It is discussed that in such a situation the adversary may pause before attempting login with cracked passwords.

B. Chaffing with Tweaking

In this method, user password seeds the generator algorithm which tweaks selected character positions of the real password to produce the honeywords. For instance, each character of user password in predetermined positions is replaced by a randomly chosen character of the same type: digits are replaced by digits, letters by letters, and special characters by special characters. Number of positions to be tweak, denoted as t should depend on system policy etc. As an example t = 3 and tweaking last t characters may be a method for generator algorithm Gen (k, t). Another approach named in the study as"chaffing-by-tweaking-digits" is executed by tweaking the last t positions that contain digits. For example, by using last technique for the password 42hungry and t = 2, the honeywords 12hungry and 58hungry may be generated.

C. Tail

This is 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 sugarword table given below may be attained:

II. RELATED WORK

A.Existing Work:

Discloser of password files is a severe security problem that has affected millions of users and companies like Yahoo, RockYou, LinkedIn, eHarmony and Adobe since leaked passwords make the users target of many possible cyber-attacks. These recent events have demonstrated that the weak password storage methods are currently in place on many web sites. The encryption of the password is stored in the database. Here the security measures are up to the encryption only and not the further security measures are taken. The user's password is saved as a plaintext in the encrypted form. As the hacker stoles the database, the account can easily accessed as after decryption.

B. Proposed Work:

Here the focus is on security issue and dealing with fake passwords or accounts as a simple and cost effective solution to detect compromise of passwords. Honeypot is one of the methods to identify occurrence of a password database breach. In this approach, the administrator purposely creates deceit user accounts to lure adversaries and detects a password disclosure, if any one of the honeypot passwords get used. Proposed model is based on use of honey words to detect password-cracking. We propose to use indexes that map to valid passwords in the system. The contribution of our approach is twofold. First, this method requires less storage compared to the original study. 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.

III. PROBLEM STATEMENT

To design and develop an alternative approach that selects the honey words from existing user passwords in the system in order to provide realistic honey words perfectly flat honey word generation method.

IV. LITERATURE SURVEY

1) Title: Password Cracking Using Probabilistic Context- Free Grammars.
Choosing the most effective word-mangling rules to use when performing a dictionary-based password cracking attack can be a difficult task. To provide a more effective way to crack passwords as compared to traditional methods by testing and most effective when tailoring one's attack against different sources by training it on passwords of a relevant structure.
Year: 2010

2) Title: Examination of a new defense mechanism: Honey words.
The decoy passwords i.e honey words to detect attacks against hash password database. For each user account the legitimate password stored in form of honey words.It is much easier to crack a password hash with the advancements in the graphical processing unit.
Year: 2011

3) Title: A large-scale study of web password habits.
Reporting the results of a large scale study of password use and password re-use habits. The study involved half a million users over a three month period .Client component on users' machines recorded a variety of password strength.
Year: 2010

4) Title: Improving Security Using Deception
As the convergence between physical and digital worlds continues at a rapid pace, much of our information is becoming available online. Here to identify the areas of worth, investigations are done.
Year: 2011

5) Title: Guess again: Measuring password strength by simulating password-cracking algorithms
The comparative strength of different composition policies were found with several notable results about the security of the password. The effectiveness of a dictionary check depends heavily on the choice of the dictionary.
Year: 2012

V. SYSTEM ARCHITECTURE

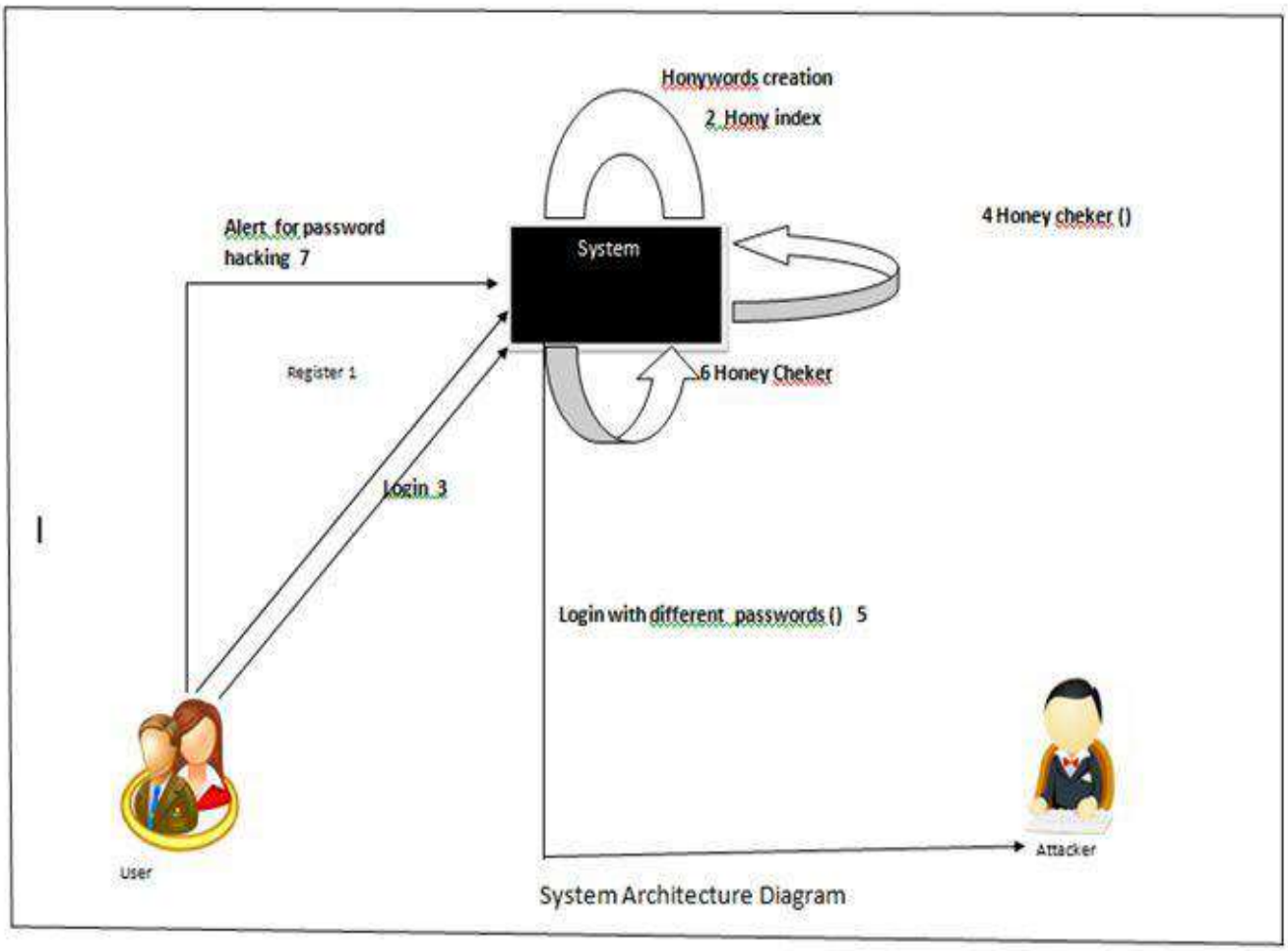
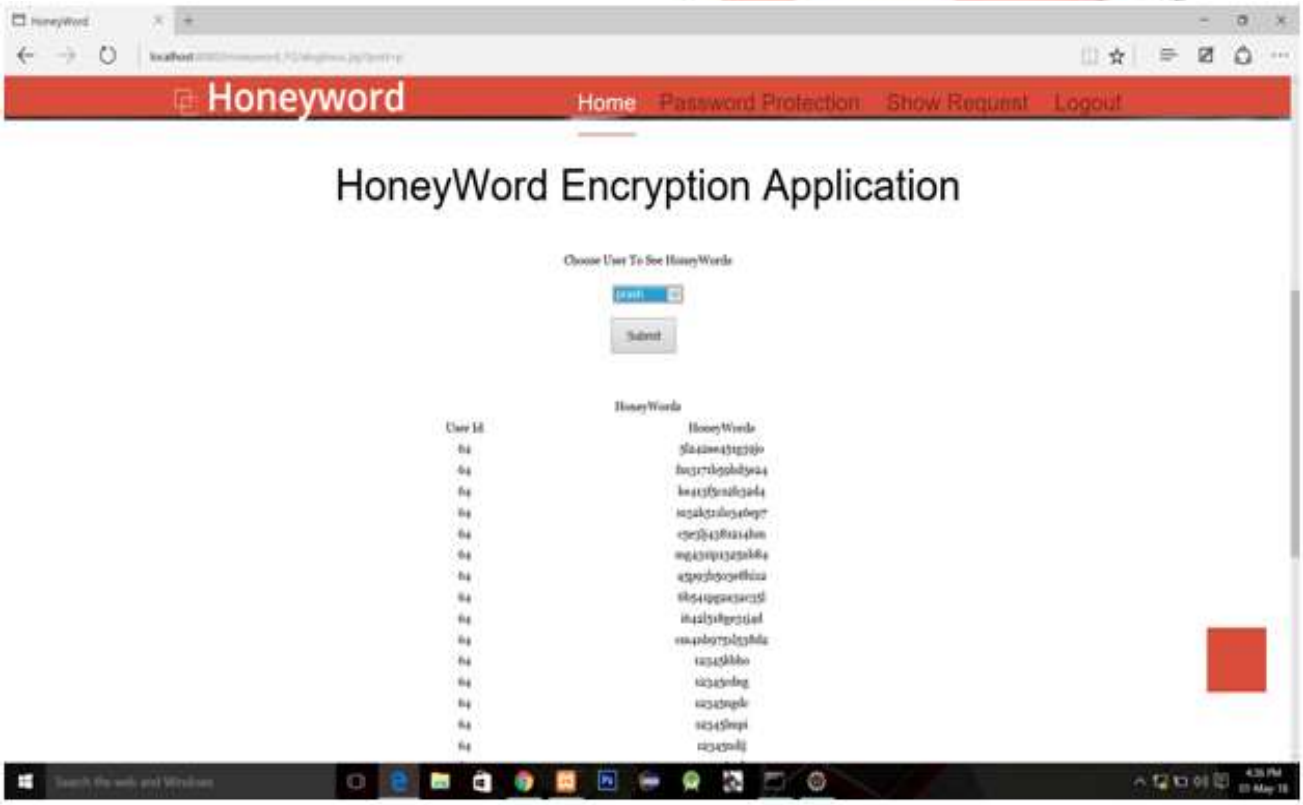


Fig: System Architecture

VI. RESULT

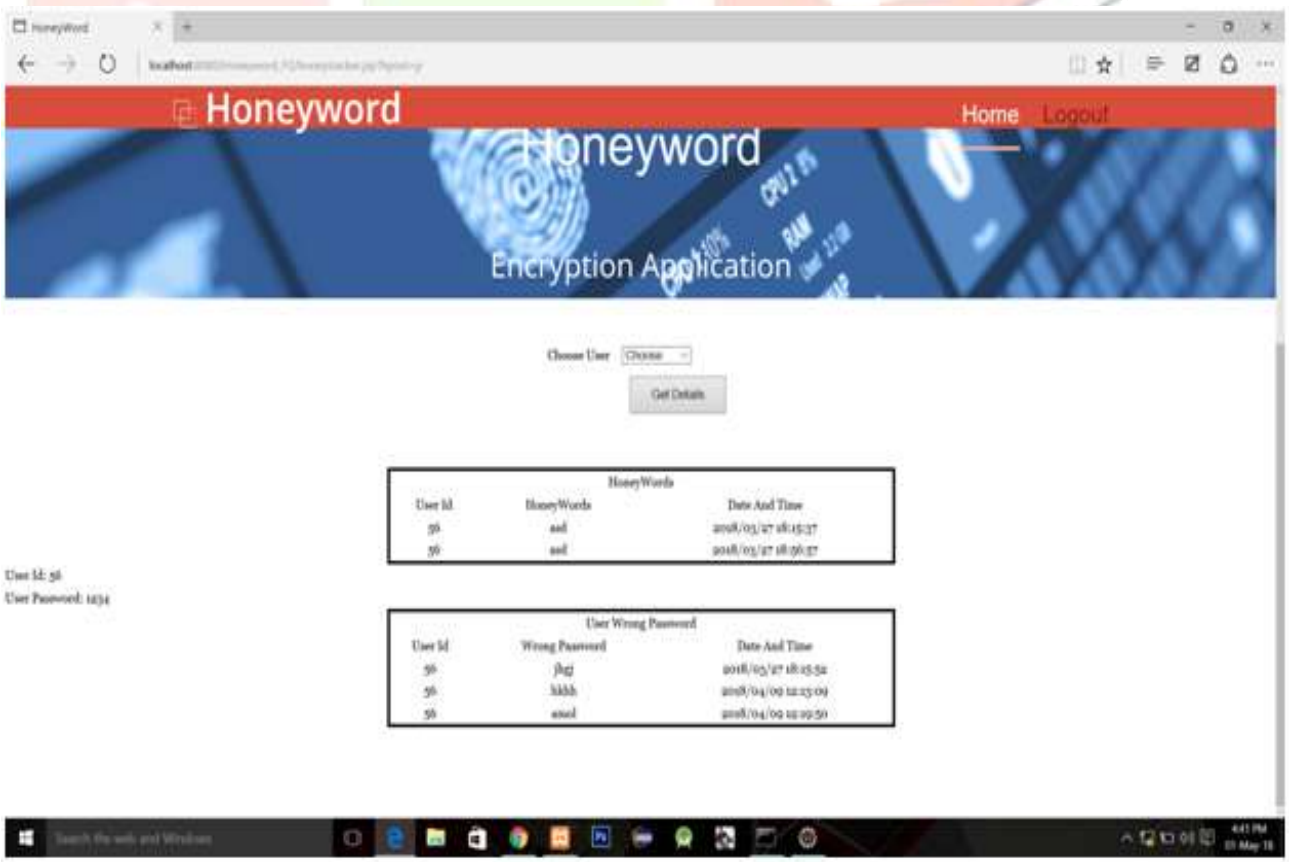
Screenshot 1

Screenshot 2

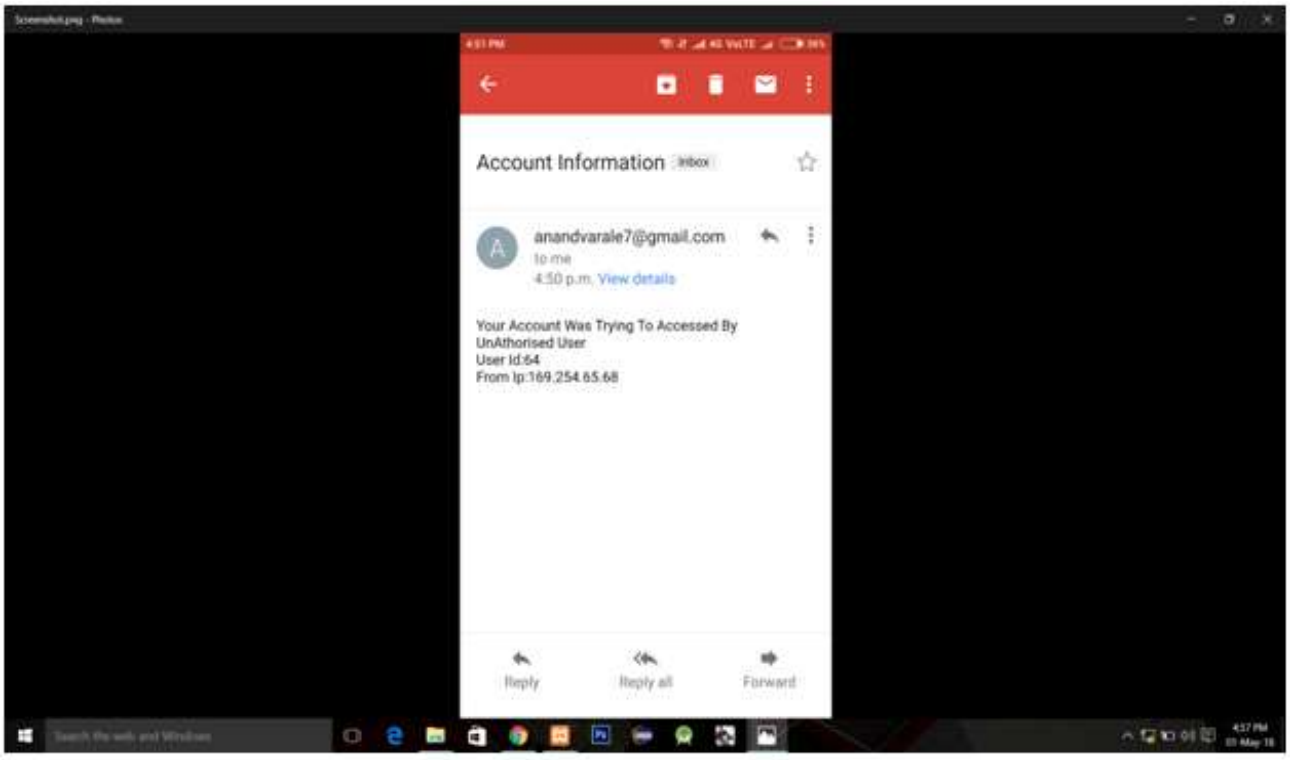


Screenshot 3

Screenshot 4



Screenshot 5



Screenshot 6

VII. CONCLUSION

We propose monitoring data access patterns by profiling user behavior to determine if and when a malicious insider illegally accesses someone's documents in a system service. If the malicious behavior is observed, the user is informed and requested to change password which is used for improving the security and preventing the documents of the user. User account is secured by applying various ways to keep the account safe. For more security purpose, user is informed about the changes occurred in the account.

VIII. FUTURE SCOPE

In the future, we would like to refine our model by involving hybrid generation algorithms to also make the total hash inversion process harder for an adversary in getting the passwords in plaintext form from a leaked password hash file. Hence, by developing such methods both of two security objectives increasing the total effort in recovering plaintext passwords from the hashed lists and detecting the password disclosure can be provided at the same time.

REFERENCES

[1] Imran Erguler ,Achieving Flatness: Selecting the Honeywords from Existing User Passwords , Year: 2016, Volume: 13, Pages: 284 - 295

[2] H. Bojinov, E. Bursztein, X. Boyen, and D. Boneh. Kamouflage:, "Loss-resistant password management."

[3] 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. 391–405.

[4] F. Cohen, "The use of deception techniques: Honeypots and decoys.

[5] M. H. Almeshekah, E. H. Spafford, and M. J. Atallah, Improving security using deception, Center for Education and Research Information Assurance and Security.

[6] C. Herley and D. Florencio, Protecting financial institutions from brute-force attacks, in Proc. 23rd Int. Inform. Security Conf., 2008, pp. 681685.

[7] J. Bonneau. Guessing human-chosen secrets. Technical Report UCAM-CLTR-819, University of Cambridge, Computer Laboratory, May 2012

[8] Z. A. Genc, S. Kardas, and K. M. Sabir, "Examination of a New Defense Mechanism: Honeywords," Cryptology ePrint Archive, Report 2013/696, 2013.
[9] A. Juels and R. L. Rivest, "Honeywords: Making Passwordcracking Detectable," in Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, ser. CCS '13. New York, NY, USA: ACM, 2013, pp. 145–160.
[10] D. Mirante and C. Justin, "Understanding Password Database Compromises," Dept. of Computer Science and Engineering Polytechnic Inst. of NYU, Tech. Rep. TR-CSE-2013-02, 2013.