Security Enhancement by Achieving flatness in Honeywords From Existing User Passwords

[1]Ankit Bhanushali, [2]Atish Chavan, [3]Sandip Maurya

[1,2,3]UG Student, Department of Information Technology, Pillai HOC College Of Engineering and Technology, Rasyani, Panvel, India.

Abstract - Every year new approach against cyber security threats are introduced. But simultaneously the adversary also create new techniques those overcome these efforts. So considering for security and data protection as a priority new techniques are needed. So, there is one of the important security issues is with disclosure of password file In each user account the legitimate password is stored with distinct honey words in order to sense impersonation. If honey words are selected properly, a cyber attacker who steals list of hashed passwords cannot be sure if it is the real password or a honey word for any account .In addition ,entering with a honeyword to login will trigger a caution educate the chairman about the secret word record an infraction. The simple but clever idea behind this system is insertion of false passwords called as honeywords associated with each users account. In this analyse the honey word system and present some remarks to highlight possible weak points an any attacker who's able to steal a copy of a password database won't know if the information it contains is real or fake.

Keyword- Authentication, honeypot, honeywords, login, passwords, password cracking.

I. INTRODUCTION

A simple but clever idea behind the study is the insertion of false passwords called as honey words associated with each users account[1]. When an adversary gets the password list, he/she recovers many password candidates for each account and he/she cannot be sure about which word is genuine. Hence, the cracked password can be detected by the system administrator if a login attempt is done with a honey word by the adversary. [2] We use the notations and definitions to simplify the description of the honey word scheme.

In this way, there are two issues that should be considered to overcome these security problems: First, passwords must be protected by taking appropriate precautions and storing with their hash values computed through salting or some other complex mechanisms [3]. 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 le disclosure incident happened or not to take appropriate actions [4]. In this study, we focus on the later issue and deal with fake passwords or accounts as a simple and cost effective solution to detect compromise of passwords.[5] Honeypot is one of the methods to identify occurrence of a password database breach. In this method, the administrator purposely creates deceit user accounts to lure adversaries and detects a password disclosure, if any one of the honey pot passwords get used [6].To design the secure environment using honeywords, it overcome password-crack detection problem and security policies should reduce the cyber-attacks. This system selects the honeyword from existing password of the user and reduces the storage cost of the honeyword scheme.

II. LITERATURE SURVEY

This idea has been modified by C.Herley and. D.Florencio[7] to protect online banking accounts from password brute-force attacks. According to the study, for each user incorrect login attempts with some passwords lead to honeypot accounts, i.e. malicious behaviour is recognized. For instance, there are 108 possibilities for a 8-digit password and let system links 10000 wrong password to honeypot accounts, so the adversary performing the brute-force attack 10000 times more likely to hit a honeypot account than the genuine account. Use of decoys for building theft-resistant was introduced by Bojinov et al. in called as Camouflage. . In this model, the fake password sets are stored with the real user password set to conceal the real passwords, thereby forcing an adversary to carry out a considerable amount of

DOI : 10.18231/2454-9150.2018.0046

online work before getting the correct information. Recently, Juels and Rivest have presented the honeyword mechanism to detect an adversary who attempts to login with cracked passwords. The most important concept is information security requirement in this which is secured using some authentication method. Various authentication method are existing such as Patterns, Passwords, PIN's etc.. Now-a-days most generally used technic for authentication is passwords. Security of password is an important part in security. A password is a secret word, which a user must input during a login, this word is match only after that it is possible to get access.

Generally disclosure of password files is a several security problem that has affected millions of users and many companies and software industries store their data in database, Like facebook, Yahoo, RockYou, Gmail and Adobe[1],[2]. Generally user name and passwords are stored in a database. Since stolen passwords make the users target of many possible attacks. These recent events have proved that the weak password storage methods are currently used by many people on websites. For example, the LinkedIn passwords were using the SHA-1 algorithm without a salt and similarly the passwords in the eHarmony system were also stored using unsalted MD5 hashes[3].Once a password file is leakage, attacker by using the password cracking technique it is easy to capture most of the plaintext passwords[4].

In this respect, there are two issues that should be considered to avoid these security problems: First, passwords must be protected by taking proper caution and storing with their hash values computed through some other correct complex mechanisms. Hence, for an advance it must be hard to include hashes value in plaintext passwords. The second point is that a secure system should detect whether a password file leakage incident happened or not to take appropriate actions. 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 [5], [6]. In the proposed system we focus on the honeyindexand deal with fake passwords or accounts as a simple and cost effective solution to detect compromise of passwords.This paper describes the study of password used and password reused habits. They measured average number of passwords and average number of accounts each user has, as well as measured number of times user enters pass-word per day. They calculated this data and estimated password strength, password vary by site and number of times user forgotten password. In their findings, it showed users choose weak password; they measured exactly how number of distinct passwords used by a client vs. age of client in days also, number of sites per password vs. age of client in days. They also analysed password strength. We are able to estimate the number of accounts that users maintain the number of passwords they type per day, and the per cent of phishing victims in the population.

III. 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. Our proposed model is still using honeywords concept. However, instead of generating honeywords we generate honey indexes of existing passwords. For achieving this, for each account we assign index number, which we call honey indexes. Moreover, hash of the correct password is saving with the correct index in a list. On the other side, in another list ui is stored with a honey index set which is consisting of the honey indexes and also the correct index. Honeyindex set is created as when any new user registered it takes some honey indexes and then merge the new honeyindex of that new user and shuffled all the honeyindexes after shuffling we get the new honeyindex set which will get stored in the list. The contribution of our approach: First, this model requires less storage compared to the original study. Second, effectiveness of the honeyword system directly depends on how Gen() flatness is creating the honeywords and how it depends on human behaviour in choosing passwords. In our approach indexes of passwords of other users are used as the fake indexes in honeyindex set, so it’s difficult to find which

DOI : 10.18231/2454-9150.2018.0046

password is wrong and which is correct becomes more complicated for an adversary.

A] SYSTEM ARCHITECTURE

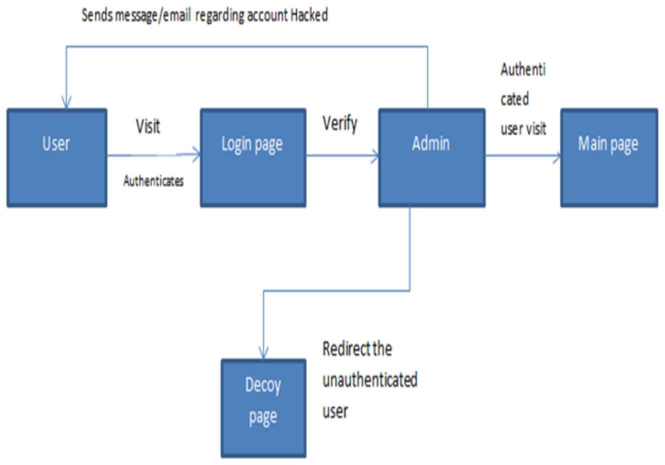


Fig.1:System Architecture

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.

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.

IV. APPLICATIONS

1. This system can be useful to avoid DoS attack.
2. This system helps to prevent brute force attack which is very common.
3. This system has advantages over other systems as it is easy to generate honeywords and low complexity level.
4. This system can be used at various banking operations also for security purpose.
5. This system saves our files from hacker.

This system gives instant alert to user or admin in case of unauthorized access to user account.

V. SYSTEM TESTING ANALYSIS

5.1 Test Cases:

Test case	Honeyword generation
Objective	System will generate the honeywords using some methods such as chaffing by tweaking, chaffing with password model, chaffing with tough nuts.
Expected Result	Successful.

Test case	Notification
Objective	System will notify user for malicious login attempt
Expected Result	Successful.

5.2 GUI Testing

Test case	Login Screen- Sign up
Objective	Click on sign up button then check all required/ mandatory fields with leaving all fields blank
Expected Result	All required/ mandatory fields should display with symbol "*". Instruction line "* field(s) are mandatory" should be displayed
Test case	Create a Password >>Text Box Confirm Password >>Text Box
Objective	Check the validation message for Password and Confirm Password field
Expected Result	Correct validation message should be displayed accordingly or "Password and confirm password should be same" in place of "Password mismatch".

DOI : 10.18231/2454-9150.2018.0046

VI. RESULT

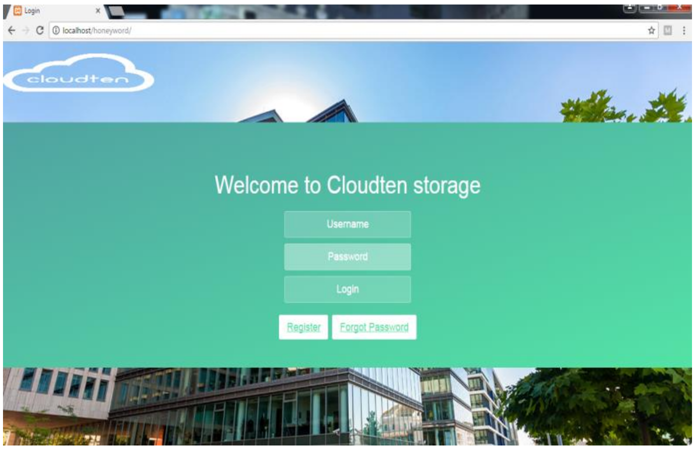
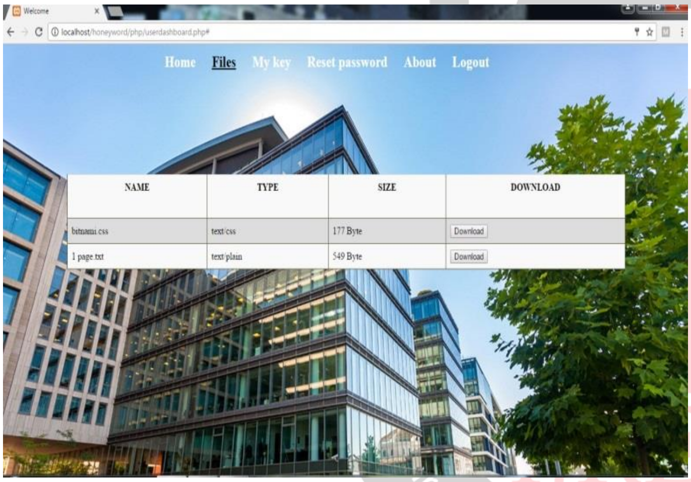


Fig2:User Login

User login: A login, logging in or logging on is the entering of identifier information into a system by a user in order to access that system (e.g. ,a computer or a website).It is an integral part of computer security.



NAME	TYPE	SIZE	DOWNLOAD
bitnami.css	text/css	177 Byte	Download
1 page.txt	text/plain	549 Byte	Download

Fig 3: User Files

User Files: Here after the complete uploading of file by user , view and retrieve option is allocated and the details of the files is also present.

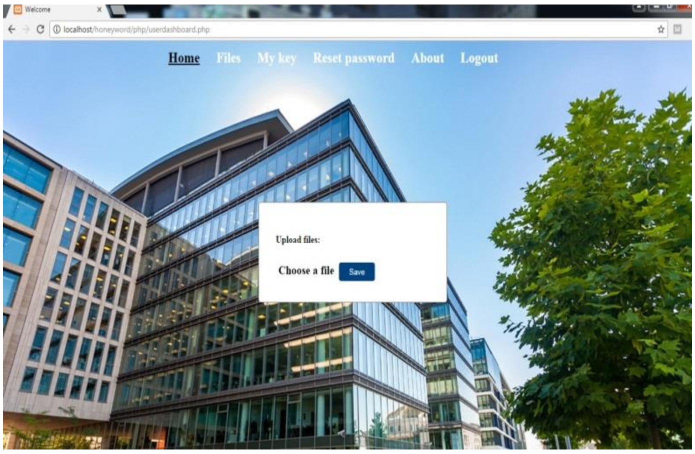
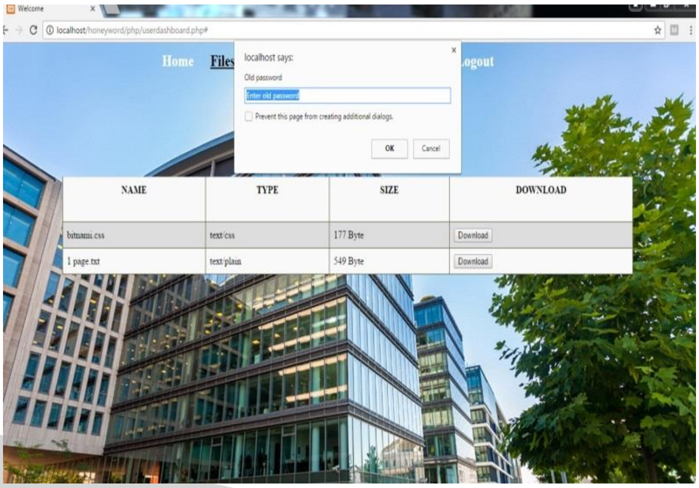


Fig 4:Upload files

Upload files: After successful login , it is possible for the user to upload files and even can retrieve them with proper authentication.



localhost says:

Old password

Enter old password

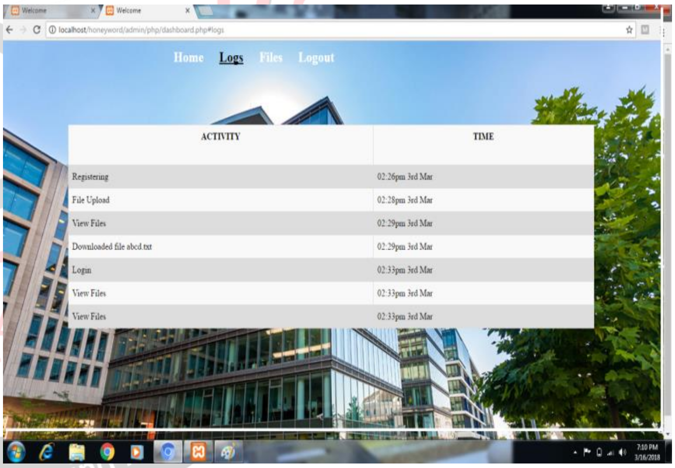
□ Prevent this page from creating additional dialogs.

OK Cancel

NAME	TYPE	SIZE	DOWNLOAD
bitnami.css	text/css	177 Byte	Download
1 page.txt	text/plain	549 Byte	Download

Fig 5: Retrival/Download of files

Retrival/Download of files: The system is based on providing high level of security if the user wants to retrieve the data then it needs to provide a unique key which will be different on every time the user retrieves.

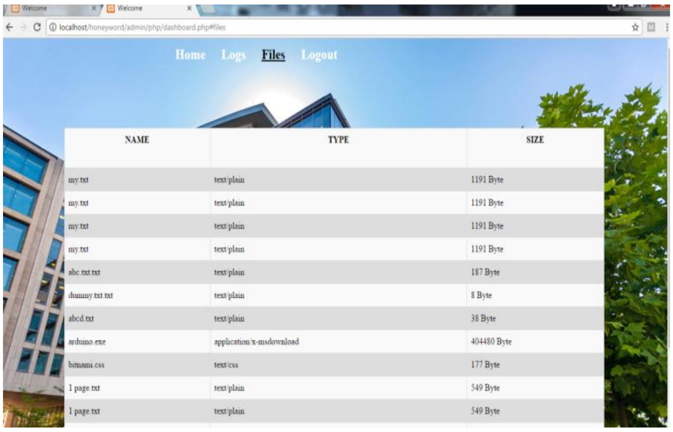


ACTIVITY	TIME
Registering	02:26pm 3rd Mar
File Upload	02:28pm 3rd Mar
View Files	02:29pm 3rd Mar
Downloaded file abcd.txt	02:29pm 3rd Mar
Login	02:33pm 3rd Mar
View Files	02:33pm 3rd Mar
View Files	02:33pm 3rd Mar

Fig 6: Admin Log's

Admin Log's The system describes admin model in which the admin can view the logs and details of each file that is uploaded by the user but without knowing the legitimate user of the file.

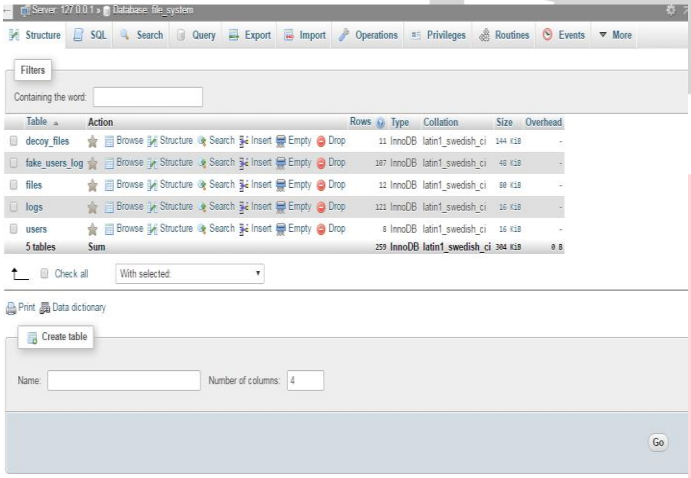
DOI : 10.18231/2454-9150.2018.0046



NAME	TYPE	SIZE
my.txt	text/plain	1191 Byte
my.txt	text/plain	1191 Byte
my.txt	text/plain	1191 Byte
my.txt	text/plain	1191 Byte
abc.txt.txt	text/plain	187 Byte
dummy.txt.txt	text/plain	8 Byte
abcd.txt	text/plain	38 Byte
arduino.exe	application/x-msdownload	404480 Byte
bitnami.css	text/css	177 Byte
1 page.txt	text/plain	549 Byte
1 page.txt	text/plain	549 Byte

Fig 7: Files Stored

Files Stored The Admin panel have the responsibility to contain all the files simultaneously. But the current system restricts the admin to see the content and uploaded by which user.



Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
decoy_files	Browse Structure Search Insert Empty Drop	11	InnoDB	latin1_swedish_ci	144 KiB	-
fake_users_log	Browse Structure Search Insert Empty Drop	107	InnoDB	latin1_swedish_ci	48 KiB	-
files	Browse Structure Search Insert Empty Drop	12	InnoDB	latin1_swedish_ci	80 KiB	-
logs	Browse Structure Search Insert Empty Drop	121	InnoDB	latin1_swedish_ci	16 KiB	-
users	Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 KiB	-
5 tables	Sum	259	InnoDB	latin1_swedish_ci	304 KiB	0 B

Name:
Number of columns: 4

Fig 8: Database Struture

Database Struture Admin plays a role of database administrator which is responsible for all the files, accounts stored, and provides security and helps to maintain all the records.

VI. CONCLUSION

As per the analysed data it deals with security of the honeyword system This system helps to user and admin. User gets instant alert when some hacker tried to access his account . Also hacker will see the list of decoy files in the system. So he feels that he have hacked the account.

According to the analysis presented a new approach to make the generation algorithm as close as to human nature by generating honeywords with randomly picking passwords that belong to other users in the system. As per the study, compared the proposed model with other methods with respect to DoS resistance, flatness, and storage cost and usability properties.

REFERENCES

[1] 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.

[2] A. Vance, If Your Password is 123456, *Just Make ItHackme*, The New York Times, vol. 20, 2010.

[3] K. Brown, *The Dangers of Weak Hashes*, SANS Institute InfoSec Reading Room, Tech. Rep., 2013.

[4] 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.

[5] F. Cohen, *The Use of Deception Techniques: Honeypots and Decoys*, Handbook of Information Security, vol. 3, pp. 646655, 2006.

[6] *Improving Security using Deception*, Center for Education and Research Information Assurance and Security, Purdue University, Tech. Rep. CERIAS Tech Report 2013-13, 2013

[7] C. Herley and D. Florencio, *Protecting nancial institutions from brute-force attacks*, in SEC08, 2008,pp.681685.

DOI : 10.18231/2454-9150.2018.0046