International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2494-9150 Vol-02, Issue 10, Jan 2017

Achieving Flatness: Selecting Honeywords From Existing User Passwords

[1]Prashant Muthiya, [2]Sachin Padvi, [3]Devendra Patil, [4]Dipak Patil

[1,2,3,4]UG Students, Department of Computer Engineering, Late G.N. Sapkal Knowledge Hub, Nashik, Maharashtra, India.

[1]pmuthiya29@gmail.com, [2]sachinpadvi08@gmail.com , [3]dipakpatil085@gmail.com , [4]devendra2india@gmail.com

Abstract : **Each user account the legitimate password is stored with several honey words in order to sense impersonation. If honey words are selected properly, a cyber attacker who steals a le of hashed passwords cannot be sure if it is the real password or a honey word for any account. Moreover, entering with a honey word to login will trigger an alarm notifying the administrator about a password le breach. The simple but clever idea behind this system is insertion of false passwords called as honeywords associated with each users account. In this system scrutinize 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 wont know if the information it contains is real or fake.**

Keywords : Authentication, honeypot, honeywords, login, passwords, pass-word cracking.

I. INTRODUCTION

Basically, 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, she recovers many password candidates for each account and she cannot be sure about which word is genuine. Hence, the cracked password les can be detected by the system administrator if a login attempt is done with a honey word by the adversary. We use the notations and de nitions to simplify the description of the honey word scheme [2]. In this respect, 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 [1]. In this study, we focus on the latter issue and deal with fake pass-words 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 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 reduce the storage cost of the honeyword scheme [5].

II. LITRATURE SURVEY

A. *Guess again (again and again)*

Measuring password strength by simulating password-cracking algorithms.

This paper describes the effects of password-composition policies on the guess ability of passwords. Seven different password-composition policies are used online to apply on a dataset of 1200 plaintext passwords. They have developed approaches for calculating time consumed to guess each password they collected. They have implemented guess-number calculator to evaluate the effectiveness of password-guessing attacks. Results also reveal important information about conducting guess-resistance analysis. Effective attacks on passwords created under complex or rare-in-practice composition policies re-quire access to abundant, closely matched training data. Shannon entropy provides only a rough correlation with guess resistance and is unable to correctly predict quantitative differences in guess ability among password sets [1].

B. The Science of guessing: analyzing an anonymized corpus of 70 million passwords

The Science of guessing: analyzing an anonymized corpus of 70 million pass-words

This paper describes the evaluation of large password data sets by collecting a massive password data set legitimately and analyzing it in a mathematically rigorous manner. In previous paper, Shannon entropy and guessing entropy not worked with any realistically sized sample, therefore, they developed partial guessing metrics including a new variant of guess-work parameterized by an attackers desired success rate. In their study most troublesome is how little password distributions seem to vary, with all populations of users [2].

C. A Large-Scale Study of Web Password Habits

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 weak. They measured 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 analyzed password strength. We are able to estimate the number of accounts that users maintain the number of passwords they type per day, and the percent of phishing victims in the overall population [3][4].

D. An In-Depth Analysis of Spam and Spammers

An In-Depth Analysis of Spam and Spammers

This paper describes the characteristics of spam and technology used by spammers. They observed that spammers use software tools to send spam with attachment. To track and represent the characteristics of spam and spammers they setup a spam trap in their mail server. The paper is discussed in two types i.e. first type spam with attachment and second type is spam without attachment. They concluded, for spam without attachment, senders use non sophisticated methods but for spam with attachment, senders use sophisticated software to spam end users [5].

E. Examination of a New Defense Mechanism: Honeywords

Examination of a New Defense Mechanism: Honeywords

This paper describes hash passwords are used to improve security. For user authentication false passwords are added in hashed password i.e.honeywords. They analyzed the honeyword system according to both functionality and the security perspective. They also elaborated how the system will respond to six password related attacks. Improvements for honeywords is described briefly i.e. number of honeywords, typo-safe honeyword generation and old passwords problem. Assumptions are illustrated to an active attack against honeyword system. They concluded that honeyword system is the powerful defense mechanism where an adversary steals the file of password hashes and inverts most or many of the hashes [6][7].

III. PROPOSED SYSTEM

Following Figure shows the system architecture which having application side and client side. At application side User authentication, file Upload, get encryption and decryption key will be done [1].

For eg. To check whether SQL injection attacks are possible, the vulnerability scanners send modified requests and analyze the responses returned by the server. A server may respond with a rejection page or with an execution page. A rejection page corresponds to the detection of syntactically incorrect or in-valid inputs. An execution page is returned by the server as a consequence of a successful execution of the request. This page legitimate use of the web site, but may also result from a successful exploitation of an injection attack [5].

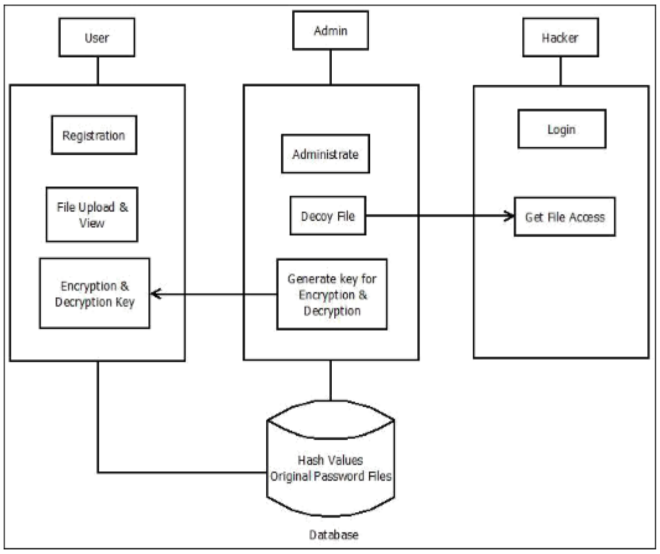

Fig. 1 Proposed system Block Diagram

We propose a simple method for improving the security of hashed passwords: the maintenance of additional honeywords(false passwords) associated with each users account [3]. An adversary who steals a file of hashed passwords and inverts the hash function cannot tell if he has found the password or a honeyword. The at-tempted use of a honeyword for login sets off an alarm. An auxiliary server (the honey checker) can distinguish the user password from honeywords for the login routine, and will set off an alarm if a honeyword is submitted [4].

IV. Conclusion

The security of the honeyword system and addressed a number 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., the generator algorithm determines the chance of distinguishing the correct password out of respective sweetwords.

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 Crack-ing Using Probabilistic Context-Free Grammars*, in Security and Pri-vacy, 30th IEEE Sympo-sium 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] M. H. Almeshekah, E. H. Spafford, and M. J. Atallah, *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 financial institutions from brute-force attacks*, in SEC08, 2008, pp. 681685.