ISSN:0975-9646

A Novel Approach: Proactive Password Checking based on User Profiling

Monika Pathak,Sukhdev Singh
Assistant Professor
Multani Mal Modi College,
Patiala

Abstract-Now days, Proactive Password checking tools are used to choose safe Credentials (Id/Password). It is a well-known problem in computer security that human chosen passwords are insecure because these passwords taken from small domain. A small password domain enables hackers to crack credentials by trying all possible id/passwords, until they find the correct one. This attack is known as a "dictionary attack". Proactive password checking enforces password policies and prevents users from choosing easily guessable passwords. Proactive password checking method provides an automatic tool which check whether the given credential is safe or not and find the relationship between user profile and credentials. In this paper, we discussed about the threats of user profile and credentials. The user profiling is a most frequent activity over the internet by which user has asked to provide personal information. Proactive password checking technique is introduced to solve these problems. It is used to determine whether the password choice is acceptable or not. Different techniques, mechanism and tools of proactive password checking are also discussed in this paper. The main concern is to check the level of association between different fields of user profile with credentials.

Keywords
Proactive Password Checking, User profiling, Safe Credentials, Naïve Pattern Matching, Authentication.

1. INTRODUCTION

Password security is a severe problem. Due to the limitation of human memory, people choose easily guessable passwords (e.g. phone numbers, birthdays, name of family members or friends, or frequently used words in human languages) which create serious security problems. Generally user believed that difficult passwords were secure but difficult to remember and easy-to remember passwords were insecure. A small password domain [1] enables hackers to attempt to login to accounts by trying all possible passwords, until they find the correct one. This attack is known as a "dictionary attack". Safe credentials are one of the key requirements in Electronic commerce which is buying and selling of goods and services across the Internet. It includes transaction of money and transfer of confidential information over the Internet. SSL (Secure Sockets Layer) protocols are used to make transactions of sensitive data over the internet, but another equally important issue for security is protection from attacks on your web site and user credential. The main concern is of security issues. There are number of methods [2] to protect the web site, transfer of confidential information, but there is a lack of security of credential of site login or credit card numbers for user end. This paper explores the rule of proactive password checking in authentication and issues related to safe profiling.

2. ROLE OF PROACTIVE PASSWORD CHECKING IN AUTHENTICATION

Authentication [3] is a process of identifying an individual based on a username and password. This process consists of obtaining the authentication information from an individual, analysis the data and determining if it is associated with that individual. It means the computer store some information about the individual. Authentication is case-insensitive; a password guessing attack against it doesn't need to consider whether letters in the password are uppercase or lowercase. When a hacker cracks passwords, the following two methods [4] can be used:

1. To do a dictionary attack, which tries each of a list of word and other possible weak passwords, and simple transformations such as capitalizing, prefixing, suffixing or reversing a word as a candidate until the hashed value of the candidate matches a password hash.
2. To launch a brute force attack to search the whole key space, which is commonly huge? Hackers, however, always prefer to use dictionary attack.

Passwords are the example of authentication mechanism based on what people know: user supplies the password and the computer validates it. If the one that associated the user then user's identity is authenticated. If not, the password is rejected and authentication failed. The most common type of attack is password guessing. Attackers can guess passwords locally or remotely approach.

Password guessing [5] is not very difficult as we think because most of the passwords are taken from dictionaries. Dictionary attack is a method of breaking into a password-protected computer or server by systematically entering every word in a dictionary as a password. It can also be used in an attempt to find the key necessary to decrypt an encrypted message or document. Dictionary attacks work because many computer users and businesses insist on using ordinary words as passwords.

Web based user profile is one of the most frequent activity on internet. The user is asked to provide personal information in the form of profile. On the other hand, generally users create passwords based on personal information like birth city, pet name, surname or combination of these. It is severe problem because hackers can easily guess user's password based on personal information. The present study is focused on safe web based profiling and safe credentials.

To solve these problems, the concept of proactive password checking technique is used. Where proactive password checking is a technique to verify credentials based on some rules so that safe credentials can be generated. Proactive password policies can be defined on the basis of user profile such that if credentials are strongly associated with user profile then credential may be rejected.

In literature [6-10] most of the proactive password techniques are based on the dictionary attack. They check each user-chosen password candidate against a dictionary of weak passwords. If a candidate matches a dictionary item, or anyone of its variants that are generated by common transformations, then the candidate is an unacceptable password and rejected. *Crack [11]* is one of the most popular password cracking software and has used as a dictionary-based crack in tool. A *cracklib [12]* supported system can prevent all weak passwords that can be guessed by *crack.*

The proactive password techniques based on dictionaries faced many problems like requirement of more storage space, long response time due to huge size of dictionaries, requirement of high processing system. It also requires implementing complicated compression algorithm like Directed Acyclic Word Graph. To overcome the problem of comparison with huge data storage, the *cracklib* technique is used which is based on a file indexing to access dictionary words. In summary, the algorithms used by *cracklib* to optimize dictionary storage and checking speed were very intuitive and worked efficiently when the dictionary file was of a modest size.

The concept of decision tree is used in *ProCheck* [13] which compress huge sized high dictionary and improve checking speed in many folds.

In this paper, we have discussed a proactive password which bases on some pattern matching rules to check whether the password is safe or not on the bases of user profile details. In literature no such system is found.

3. Different Techniques In Proactive Password Checking

Richard et al.[14] have introduced two categories of credit card fraud detection.These categories are behavioral fraud and application fraud. Application fraud occurs when individuals obtain new credit cards from issuing companies using false personal information and then spend as much as possible in a short space of time. Behavioral fraud occurs when details of legitimate cards have been obtained fraudulently. Most credit card fraud is behavioral. This technique detects the behavioral fraud through the analysis of longitudinal data. Where longitudinal data is consider as a record of observations based on behavioral parameters.

Victor et al.[15] have provided a case study on authentication counter which measures design model and methodology for e-commerce systems. The proposed system provides different layers of security such as:

Design Time: Impose authentication secure measures while capturing data at design time and block unwanted and malicious user requirements.

High level documentation: Security measures can be directly incorporated into high-level design documents of e-commerce systems.

Counter measures: Contain effective counter measures against the set of all known security attacks related to authentication

These convert the model into a flow chart for implementation purposes so that it can easily implemented.

Guidelines: These help in avoiding security pitfalls during system implementation.

Marchany et al. [16] have introduced some of the important issues related to network and computer security. These threats cover both hackers as well as the e-commerce site. A comparative analysis of different security threats has been discussed the security weaknesses.

William et al. [17] have introduced the Biometric Authentication system which is a method of establishing a person's identity. A Biometric Authentication system is an efficient way to replace the traditional password based authentication system because it cannot be shared, lost, or guessed.

Eugene et al.[18] have provided a method which ensure password security by comparing user choices against a list of unacceptable words. This method is space efficient because it provides efficient sorting method in dictionary search.

Joseph et al. [19] have introduced a system which can defend against internal malware threats in a mobile network by using remote security scanner to check the vulnerabilities of the machines. This maximizes the usefulness of the machine by preventing attacks .Positive results have been obtained by using policy management by network manager.

4. Mechanism Of Proactive Password Checking

We are discussing the mechanism that allows a security system to check the security level of credentials created by new user.

This would protect the misuse of user personal information.

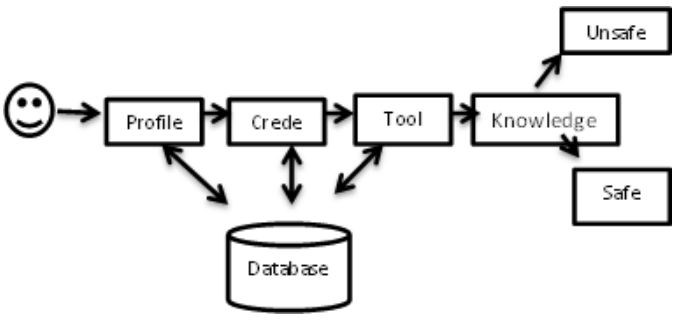

Figure1: Mechanism for Proactive Password Checking

The diagram (Fig.1) describes that different user's profile with credentials is stored into database. Pattern matching tools can be further applied to produce the desired knowledge. The co-relation technique has been applied to find the necessary relationship between the personal data and the password. This knowledge helps to determine how user profile is associated with the credentials and how security is enhanced with safe credentials. If there is

a strong relation between user profile and credentials then it will be considered as unsafe credentials.

"More secure is the credentials (password) more safe is the e-commerce business".

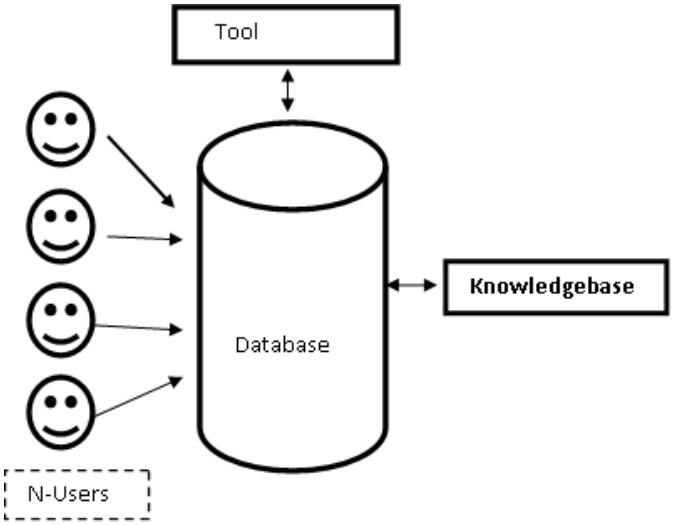

Figure2: Demonstrates data communication

The diagram (Fig.2) shows the execution flow between different user's profiles with credentials is stored into database.

Data collection has been done through web based user-friendly interface. Collected data is stored into the permanent storage. This helps the researcher to retrieve it whenever it is required. Later on data can be consider for analysis. Testing design is prepared to check the stability of the data. Data is collected and tested with persons having different level of computer knowledge. Analysis of the result has been done by some rules, which define rate from the scale of R1 to R5 where R1 means high level of insecurity and R5 means high level of security.

5. TOOL, IMPLEMENTATION AND ANALYSIS FOR PROACTIVE PASSWORD CHECKING

Pattern matching is used to compare two or more strings and find out the correlation between them. In literature, there are n number of techniques are reported. For the present study, we consider naïve algorithm [20] as described below.

```
Naive pattern matching algorithm
 i=1
while i <= n-m+1 do begin//represent phase
 j=1
 while (j<=m) and (pat[j]=text[i]) do begin//represent steps
 j=j+1
 i=i+1
 end
if j<=m then i=i-j+2
else Print("found at  Location", i-j+1)
end
```

With the help of following diagram Naïve Algorithm is demonstrated.

Practical Implementation: The Naive Algorithm

The naive algorithm is to search every location of the string *t* for the pattern *p*.

Let us consider two strings as

p = AAATA

t= AATAAAATA:

PHASE 1

Step1

A	*A*	*A*	*T*	*A*				
A	*A*	*T*	*A*	*A*	*A*	*A*	*T*	*A*

Step2

A	***A***	*A*	*T*	*A*				
A	***A***	*T*	*A*	*A*	*A*	*A*	*T*	*A*

Step3 (Failed)

A	***A***	~~*A*~~	*T*	*A*				
A	***A***	~~*T*~~	*A*	*A*	*A*	*A*	*T*	*A*

PHASE 2.

Step1

	A	*A*	*A*	*T*	*A*			
A	***A***	*T*	*A*	*A*	*A*	*A*	*T*	*A*

Step2 (Failed)

	A	~~*A*~~	*A*	*T*	*A*			
A	***A***	~~*T*~~	*A*	*A*	*A*	*A*	*T*	*A*

PHASE 3

Step1 (Failed)

		~~*A*~~	*A*	*A*	*T*	*A*		
A	*A*	~~*T*~~	*A*	*A*	*A*	*A*	*T*	*A*

PHASE 4

Step1

			A	*A*	*A*	*T*	*A*	
A	*A*	*T*	***A***	*A*	*A*	*A*	*T*	*A*

Step2

			A	***A***	*A*	*T*	*A*	
A	*A*	*T*	***A***	***A***	*A*	*A*	*T*	*A*

Step3

			A	***A***	***A***	*T*	*A*	
A	*A*	*T*	***A***	***A***	***A***	*A*	*T*	*A*

Step4 (Failed)

			A	***A***	***A***	~~*T*~~	*A*	
A	*A*	*T*	***A***	***A***	***A***	~~*A*~~	*T*	*A*

PHASE 5

Step1

				A	*A*	*A*	*T*	*A*
A	*A*	*T*	*A*	***A***	*A*	*A*	*T*	*A*

Step2

				A	***A***	*A*	*T*	*A*
A	*A*	*T*	*A*	***A***	***A***	*A*	*T*	*A*

Step3

				A	***A***	***A***	*T*	*A*
A	*A*	*T*	*A*	***A***	***A***	***A***	*T*	*A*

Step4

				A	***A***	***A***	***T***	*A*
A	*A*	*T*	*A*	***A***	***A***	***A***	***T***	*A*

Step5

				A	***A***	***A***	***T***	***A***
A	*A*	*T*	*A*	***A***	***A***	***A***	***T***	***A***

As shown in the above diagrams, the naïve algorithm uses different phases to compare pattern strings. The following table shows number of pattern successfully matched in each phase.

Table 1

Phase	No. of Pattern	Description
1	*2*	*2 character string matched*
2	*1*	*1 character string matched*
3	*0*	*0 character string matched*
4	*3*	*3 character string matched*
5	*5*	*5 character string matched*

According to the analysis of above table, Phase 5 of the algorithm shows maximum number of string pattern matched with each other.

The above algorithm is implemented for proactive password checking tool where the algorithm actually compare different fields of profile with credentials (Id and Password). The algorithm finds correlation and degree of correlation in terms of pattern matching as shown in table 1.

6. CONCLUSION

Proactive password checking enforces the password policies and prevents users from choosing easily guessable passwords. When a user chooses a password, the proposed proactive checker will determine whether his password choice is acceptable or not. The user will be immediately provided password security by selecting good passwords. It is easy for proactive password checking to block all possible weak passwords. With the help of Pattern matching technique, Proactive password checking allows users to choose secure password on the basis of comparison of password with his/her profile. The attributes of profile of user determine the relationship with fields of credentials. If there is high level of association between credential and attributes of the profile, then credential is consider as unsafe and vice versa.

7. FUTURE SCOPE

Present research work can be used in future in following different situation and fields.

1. Prevent weak password selection: Present research work prevents the selection of weak password, which can be helpful for new users to decide secure password.
2. Integration with Anti-hacking System: It can be integrating with anti-hacking system so that user cannot provide personal information from which password guessing can be carried out.
3. Safe Profiling: User profiling is required at different situations, current study can be used to make safe profiling process can be integrated with this method. So that user may not provide sensitive information.
4. Integration with Fraud detection tool: Present research work can be integrated with Fraud detection tool. The study can be used to find retaliation ship between two or more independent variables.
5. Profile Comparing: Current study can be used to compare two or more profiles and patterns.
6. Compatibility of Profiles: The present research work can be used for finding compatibility of two or more profiles using pattern matching component.

Present work may be improved if we use advance comparison techniques like Ontology matching and advanced data structure.

REFERENCES

1. Mark Merkow, Jim Breithaupt, "Information Security Principles and Practices", Pearson Prentice Hall, 2006.
2. Michall E., Whitman and Herbert J. Maiiord, “Information Security”, Thomson, Inc., 2003.
3. J. Bonneau , S. Preibusch, “The password thicket: technical and market failures in human authentication on the web”, In Workshop on the Economics of Information Security, 2010.
4. F Bergadano, “High dictionary compression for proactive password checking”, ACM trans. on info and system security Vol.1, No.1, Nov. 1998.
5. J.Bonneau," The science of guessing: analyzing an anonymized corpus of 70 million passwords", in IEEE Symposium on Security and Privacy, pages 538-552,2012.
6. F Bergadano, “Proactive password checking with decision trees”, in proceedings of ACM conference on computer and communications security, 1997.
7. C. Davies,R. Ganesan. BApasswd: A new proactive password checker. In 16th National Computer Security Conference, pages 1--15, Baltimore, MD, Sept. 1993
8. DVKlein.Foiling,“The Cracker: A Survey of, and Improvements to Unix Password Security” , in proceedings of the USENIX Security Workshop. Portland, Oregon: USENIX Association, summer 1990.
9. Wilson, S. 1997. “Certificates and trust in electronic commerce”, Information Management & Computer Security, Vol. 5(5), 175-181, MCB University Press.
10. Burton Bloom. Space/time trade-offs in hash coding with allowable errors, CACM, 13(7): 422-426, July 1979
11. T. Raleigh and R. Underwood, ‘‘CRACK: A Distributed Password Advisor,’’ USENIX UNIX.Security, Workshop Proceedings, August 1988.
12. Alec Muffett, “CrackLib: a proactive password sanity library”.http://www.users.dircon.co.uk/~crypto/download/cracklib, 2.7.txt.
13. F Bergadano, “High dictionary compression for proactive password checking", ACM trans. on info and system security Vol.1, No.1, Nov. 1998.

14. Richard J. Bolton, “Unsupervised Profiling Methods for Fraud Detection”, 2001.
15. Victor D. Sawma.,” E-Commerce Authentication , An Effective Countermeasures Design Model”, School of Information Technology and Engineering, University of Ottawa.
16. Marchany, R. , Tront J, "E-commerce Security Issues", in the proceedings of 35th Annual Hawaii International Conference on System Sciences , Page-193,IEEE,2002.William Stallings, “Cryptography and network Security”, 3rd edition, Prentice Hall,2003.
17. Eugene H. Spafford, “Observing Reusable Password Choices”,Purdue Technical Report, Department of Computer Sciences, Purdue University, 1992.
18. Randy C. Marchany, Joseph G. Tront, “E-Commerce Security Issues”, in the of proceedings 35th International Conference on System Sciences Page-.2,- 2002.
19. Rish, Irina, "An empirical study of the naive Bayes classifier", in the proceedings of International Joint Conference on Artificial Intelligence 2001.
20. Yanhong Cui, Renkuan Guo, "Education Technology and Training" in the proceedings of International Workshop on Geoscience and Remote Sensing, 2008.