International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 2, February- 2013

Secured Password Authentication based on images

Gayatthri. M and Ranjana. P
School of Computing Sciences, Hindustan University

Abstract

Password schemes that encourage the use of strong passwords have failed. During past, preferring images for passwords was not appreciated since that involves complex techniques. In recent years, the usage of image as password has increased consistently. Most of the passwords are insecure, whether text or image, since appropriate securities are not imposed. In this paper, an extended ObPwd generates strong passwords from images on a computer.In general, a common way to make password secure is by using stenographic techniques along with other processes. It is well known that image passwords are more secure than text passwords. When image is chosen as password, it will definitely be a challenge for the hacker to hack the password since trial and error combinations can be implemented. An extended ObPwd concept is introduced in which image is chosen initially for generating password that is done along with the salting technique. The password is generated from the image selected and the final output is salted. The image selected is generally in the form of pixels. Hence, these pixels must be converted to bits using binary conversion to enhance password generation. Extended ObPwd can be implemented in places where security is more needed and in places where transactions have to be done in a safe and secured fashion. Once the login process is successful with the acceptance of password combination, the transactions and other processes can be done securely. These processes of image selection and implementation of salting technique enhances security to a greater extent since transactions are more prone to attacks. In general transactions, the flow involves simple processes. Transactions are made more safe and secure and the possibility of hacking is reduced to a greater extent. Thus, extended ObPwd is enhanced to be applied in places where security is much poor that leads to insecure transactions that will be a great boon to mankind.

Keywords: *Extended Object Based Password(ExObPwd), Hashing,Password salting,RSA algorithm,Authentication.*

1. Introduction

Text passwords remain ubiquitous, despite endless criticism. People consistently choose weak passwords for many reasons, including users trying to manage on average 25 password-protected accounts. Losing strategies include blaming users, and imposing complex password rules. Some claim that choosing weak password is a rational economic response. Some argue that strong passwords are nonessential for preventing automated online dictionary attacks like password-protected sites that can present challenge CAPTCHAs after failed attempts lock out the targeted account temporarily. However, the latter can affect legitimate users, and CAPTCHA schemes are regularly defeated by improved attacks in the artificial intelligence arms-race, by human solvers, or bypassed due to implementation flaws. A recent exploration of the feasibility of online dictionary attacks highlights the critical security vulnerability of the human-generated password. To address these issues, an object- based password scheme is introduced to generate passwords used infrequently, used in common web authentication, or used to access encryption keys. The basic idea is as follows. Many users currently possess a large collection of digital content such as photos, audio recordings, videos, documents, and e-mail messages. ObPwd generates a password from such items by computing a hash from the user-selected object then converting the hash bit string, by known techniques to an appropriate password format like a string of keyboard characters or a word sequence. In place of remembering exact passwords, users only need a strategy to remember which password object they chose. ObPwd requires no modifications to password system interfaces and the system side (remote or local) need not be aware of ObPwd, facilitating deployment. Our contributions include the basic design and an object-based replacement for text password schemes that provides results and interpretation including exploration of performance, user acceptance, and multi password interference. An extended scheme in this paper from the object password scheme provides additional security and makes a great impact on the security issues. Numerin techniques can be implied in combination but that may sometimes lead to

disaster. Therefore, combination of techniques should be carefully handled and only then the security can be enhanced.

2. Literature Survey

2.1. Password managers

Password managers are used to make the passwords more secure. Those managers use Password Hash and Password Multiplier. These support security but they face usability issues. Password Hash are browser plug-ins that generate strong passwords and prevent Java script attacks. Password Multipliers are plug-ins that protect against phishing attacks. These usability studies uncover problems so that they can be corrected.[1]

2.2. User choice in graphical passwords schemes

The user selection of passwords in two graphical password schemes are permitted. One based on entropy and the other based on high correlation with race. The graphical password schemes generally require a different posture towards password selection than text password, where selection by user remains the norm.[2]

2.3. Multiple Graphical Passwords

The study of multiple graphical passwords is to systematically examine frequency of access to a graphical password, the interference resulting from interleaving access to multiple graphical passwords, and patterns of access while training multiple graphical passwords. The findings regarding interference, training, and long-term recall motivate future field studies examining how people typically acquire and learn new passwords.[5]

2.4. Persuasion based Text passwords

Influencing users to create more secure passwords remains an open problem. Persuasive Text Passwords (PTP), a text password creation system was developed which leverages persuasive technology principles to influence users in creating more secure passwords without sacrificing usability. After users choose a password during creation, PTP improves its security by placing randomly-chosen characters at random positions into the password identifying the point where the limits of human memory lead users to employ coping mechanisms when dealing with randomness in passwords.[8]

2.5. S/KEY Password system

Eavesdropping on network connections is a form of attack to obtain login id's and passwords of legitimate users. So, a prototype software system, the S/KEYTM one-time password system is developed to counter this type of attack The user's secret password never crosses the network during login or when executing other commands requiring authentication. No secret information is stored anywhere, including on the host being protected and the underlying algorithm may be made public. The remote end (client) of this system can run on any locally available computer and the host end (server) can be integrated into any application requiring authentication.[12]

3. Existing System

In the existing system, CAPTCHA concepts are used. In password schemes, the use of strong passwords have failed. CAPTCHA schemes are regularly defeated by improved attacks in the artificial intelligence arms-race, by human solvers or bypassed due to implementation flaws. Also, the ObPwd scheme that was implemented earlier does not involve salting technique. It only involves image conversion and generating the password. Another drawback of text password is that they can be retrieved through techniques like Brute force, Caesar technique and so on. All these issues makes the password weak and also easily retrievable.

4. Proposed System

In the proposed model, an Extended Object-Based Password (ExObpwd) concept is introduced. This process aims at file based authentication scheme in which user selects an image and it is given as a input to the server for authentication which are then converted to bytes. These bytes are encrypted using RSA algorithm and then hashed. After the generation of sequence number, a part of it is salted and then authenticated.

5. Extended Obpwd System

5.1. System Architecture

Figure 1: Represents the architecture of an extended Object-based Password system . It consist of Six Modules. They are,

- User registration
- Byte conversion
- Hashing
- Salting and Desalting
- Random Sequence
- Authentication
- Sequence checking

The proposed algorithm efficiently handles the security issues. The entire processing of the system is depicted through the architecture phase. The flow for the entire process is manipulated. The literature reviews are made after which the modules are identified, apart from which the functions of the modules are stated. The proposed algorithm fits itself perfectly by improving the efficiency of the password and also makes the authentication more safe and secure. The extended ObPwd makes the transactions more convenient by presenting the password more securely. Generally, generating passwords from images are more secure. In addition to it, salting the password makes the account much secure.

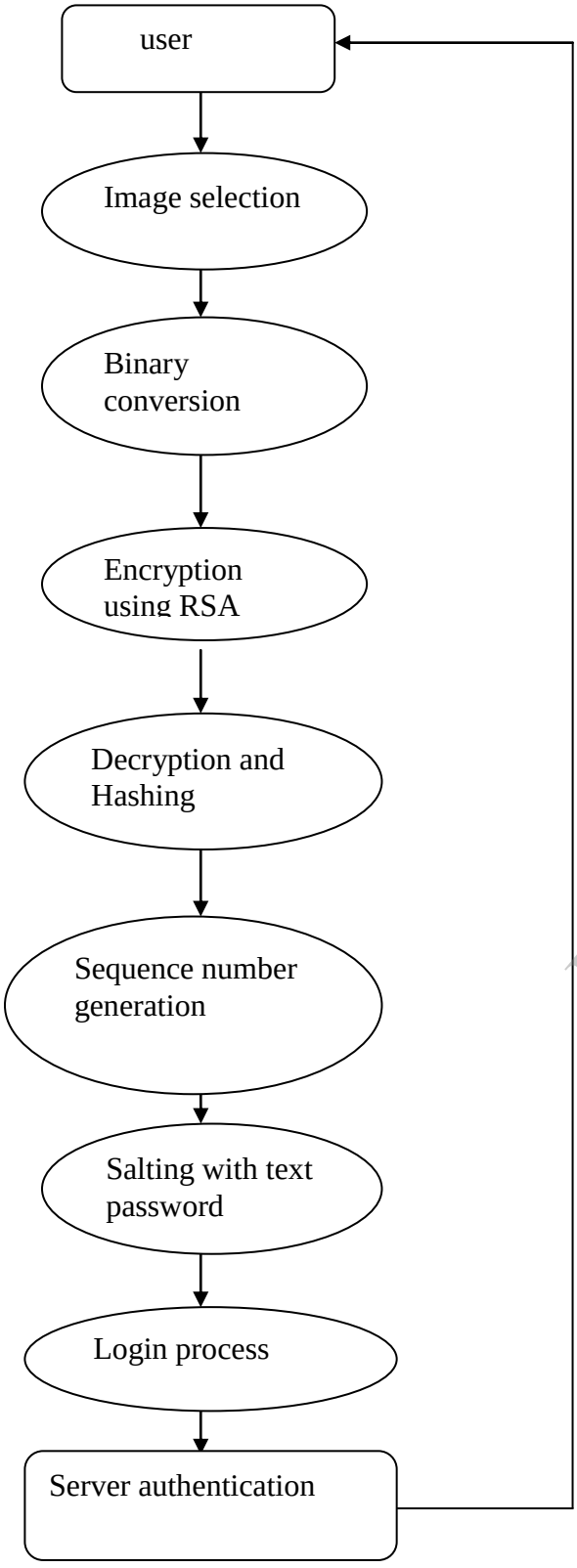


Figure 1: Extended ObPwd architecture

5.2. Module Description

5.2.1. User Registration

In this project, user registration is done along with the image selection. The user needs to select the image along with the password. After registration, the unique username ID is generated to the individual user and that is used for the further login process. This is termed as Object based Password (Obpwd).

5.2.2. Byte Conversion

The input image which is in the form of pixels currently is received and converted to corresponding bytes. These bytes are then encrypted using RSA algorithm. This encryption is done in order to avoid hacking.

5.2.3. Hashing

Hashing is a process which includes the conversion of the bytes into the hash code bytes. This conversion is done by using SHA-1 algorithm. The sequence of hashed bytes is sent to the user. The user need to select a sequence of the hashed bytes randomly which is subjected for comparison and authentication.

5.2.4. Salting and Desalting

Salting is the process of adding bits to that of the random number. This process is done by the user and the result is sent back to the server. This bit concatenation is similar to that of the addition of the parity bit. The server after receiving the salted code, desalts the code and thus the original sequence is regained. Salting process makes the ObPwd, an Extended ObPwd.

5.2.5. Random Sequence

The encrypted code which is converted into hashed bytes will generate a sequence which is random in nature. The random sequence comprises of the byte codes which are arranged in a frequently changing manner.

5.2.6. Authentication

After desalting, the original hash code is regained. Initial authentication is done when the sequence number is generated. Further authentication is proceeded when salting combination is provided. Both the user and the server must be authenticated inorder to avoid data loss or data leakage.

5.2.7. Sequence checking

Sequence checking is nothing but the same hash input should not be given to the server again. The random number should differ each time.

5.3. ExObPwd Algorithm

The algorithm that explains the processes taking place in this architecture is as follows:

Step 1: Start the process with the usual user registration.
Step 2: Browse the image for password generation with path specified to enhance security. The path is specified as:
String path1 = "C:\\Program Files\\Apache Software Foundation\\Tomcat 5.5\\webapps\\SecuredPassword \\image\\"+id+".jpg";
Step 3: Perform the byte conversion.
Step 4: The bytes are then encrypted and decrypted using RSA algorithm.
RSA rsa = new RSA();
rsa.setSeeds(61,31);
rsa.setEncKey(43);
rsa.keyGen();
String chiper = rsa.encryptToNumeralString buffer.toByteArray()).toString());
String key = rsa.getDecKeyPairAsString();
String desc = rsa.decryptFromNumeralString(chiper);
Step 5: These encrypted bytes are then hashed using SHA-1 algorithm.
SHA256 sha = new SHA256();
str = sha.hash(chiper.getBytes());
Step 6: The sequence number generated is verified and then salting technique is executed.
Step 7: The combination is executed as password.
Step 8: Stop the process through login or logout.

5.4. Impact of ObPwd

- Currently used security systems are not efficient as the proposed system.
- In the existing system, only text passwords are used. Graphical passwords are used in some security concerns.
- Now a days, hacking passwords became common.
- This system provides additional security to the account through advanced authentication.
- The major causes are caused by artificial intelligence and human solvers.
- Passwords cannot be retrieved through Brute force or Caesars techniques.
- This system is user friendly when these kind of security systems are used.
- ExObPwd system can be used in places where additional security is needed in a greater extent.

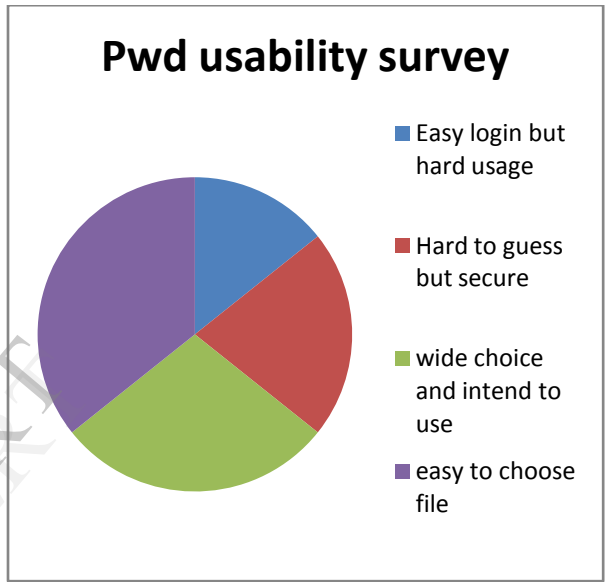
6. Analysis Report

The level of usability is based on four factors. They are,

- ease of thinking of a file to select for the password;
- ease of locating the file chosen;
- ease of creating the password using ObPwd;
- ease of logging into the website with the chosen password.

Based on the usability survey carried out in a small session ,the user's suggestion's of using ObPwd are,

About 20% of users – easy to login, hard to use and disliked using.
About 30% of users – thought hard to guess password and thought more secure.
About 40% of users – liked wider choice.
About 20 to 50% of users – easy to choose file.
About 20 to 40% - intend to use.



7. Conclusion and Future Enhancement

7.1. Conclusion

An extended ObPwd is a more secure system compared with the existing system. This system can be implemented in places where security is much poor or additional security is needed. This concept can be used extensively in the field of banking since transactions are more fraudulent. Hacking of password is reduced to a greater extent since an extended scheme provides a strong protection.

7.2. Future Enhancement

Images are currently implemented in ExObPwd system. In future, all types of digital objects can be implemented depending upon the usability. In this, image passwords are made in combination with salting technique. In further developments, image passwords can be used in combination with other password protecting techniques. This system is currently tested in banking applications.

REFERENCES

[1] S. Chiasson, P. van Oorschot, and R. Biddle, "A usability study and critique of two password managers," in *Proc. USENIX Security Symp*., Vancouver, Canada, Aug. 2006.

[2] D. Davis, F. Monrose, and M. K. Reiter, "On user choice in graphical password schemes," in *Proc. USENIX Security Symp.*, SanDiego, CA, Aug. 2004.

[3] M. Mannan and P. van Oorschot, "Digital objects as passwords," in *Proc. USENIX Workshop on Hot Topics (HotSec'08*), San Jose, CA, Jul. 2008.

[4] M. Mannan, T.Whalen, R. Biddle, and P. van Oorschot, The Usable Security of Passwords Based on Digital Objects: From *Design and Analysis to User Study Comp. Sci.,* Carleton Univ., Tech. Rep. TR-10-02, 2010.

[5] K. M. Everitt, T. Bragin, J. Fogarty, and T. Kohno, "A comprehensive study of frequency, interference, and training of multiple graphical passwords," in Proc. *Conf. Human Factors in Computing Syst. (CHI'09*), Boston, MA, Apr. 2009.

[6] D. Florêncio and C. Herley, "A large-scale study of web e password habits," in Proc. *World Wide Web Conf.* (WWW'07), Banff, Alberta, Canada, May 2007.

[7] D. Florêncio, C. Herley, and B. Coskun, "Do strong web passwords accomplish anything?," in Proc. *USENIX Workshop on Hot Topics in Security (HotSec'07*), Boston, MA, Aug. 2007.

[8] A. Forget, S. Chiasson, P. van Oorschot, and R. Biddle, "Improving text passwords through persuasion," in Proc. *Symp. Usable Privacy and Security (SOUPS'08*), Pittsburgh, PA, Jul. 2008.

[9] R. Biddle, S. Chiasson, and P. van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Computing Surveys*, to be published.

[10] W. E. Burr, D. F. Dodson, and W. T. Polk, Electronic authentication guidelines *NIST Special Publication* 800-63, Apr.2006[Online].Available:http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf

[11] S. Chiasson, A. Forget, E. Stobert, P. van Oorschot, and R. Biddle, "Multiple password interference in text and click-based graphical passwords," in Proc. *ACM Computer and Communications Security(CCS'09)*, Chicago, IL, Nov. 2009.

[12] N. Haller, "The S/KEY one-time password system," in *Proc. Network and Distributed System Security Symp. (NDSS'94)*, San Diego, CA, Feb. 1994.

[13] C. Herley, "So long, and no thanks for the externalities: The rational rejection of security advice by users," *in Proc. New Security Paradigms Workshop (NSPW'09*), Oxford, U.K., Sep. 2009.

[14] R. W. Picard, Affective Computing MIT Media Lab, *Perceptual Computing Group, Tech*. Rep., 1995.