Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 5, May 2015, pg.47 – 52

RESEARCH ARTICLE

Security and Usability Analysis of Password Agent

Seema[1], Nidhi Sharma[2], Alok Sharma[3]

M.Tech Scholar, Computer Engineering, MDU, India

Assistant Professor in CSE Dept, MDU, India

Assistant Professor in BITS Bhiwani, MDU India

kharod.seema30@gmail.com, nidhisharma1725@gmail.com, malok1231@rediffmail.com

Abstract - Password plays an important role in online authentication. But it suffer from two interactable problems ,one is password cracking and second is password theft. Password agent mechanism contains strong hashing scheme which provides stronger protection against password theft and password cracking. Password Agent generates strong passwords by enhancing the hash function with a large random salt. This paper describes comparative security and usability analysis of Password Agent with different hashing mechanism.

Keywords - LPWA (Lucent Personal Web Assistant), PwdHash, Passpet, Pwd Multiplier, UserID

I. INTRODUCTION

Password Agent, is an automatic password management system with enhanced hashing, which consists of a Salt Repository server and a browser plug-in Agent for securing online passwords. The Salt Repository stores a list of salts for each registered user while the Agent provides the user interface, salt retrieval, and hashing functionality. When a plain-text password needs to be protected for a specific website, the user simply activates the Agent and enters the plain-text password. The Agent automatically concatenates the plaintext password and the website specific salt to generate the site password via a hash function.

In figure, each enterprise network maintains a Salt Repository providing salt storage services for its users. To achieve high reliability and scalability, it is possible that multiple servers function as the Salt Repository within one enterprise network [1].

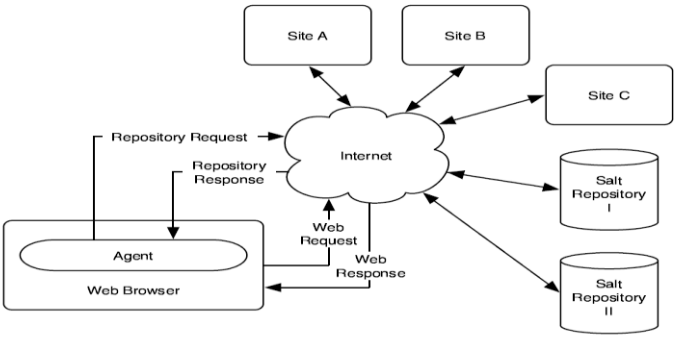


Figure 1: The architecture of PasswordAgent.

Password Agent generates strong passwords by enhancing the hash function with a large random salt. With salt repository, it gains a much stronger security guarantee than existing mechanisms. Password salting is a way of making passwords more secure by adding a random string of characters to passwords before their hash is calculated, which makes them harder to reverse. The random string of characters should be a mix of letters, numbers and other characters.

II. RELATED WORK

In this, we highlight the contributions of the Lucent Personal Web Assistant (LPWA) and three recent systems: PwdHash, Password Multiplier, and Passpet .These existing systems simplify the password hashing in online user authentication, and they are most related to our proposed Password Agent. It is an HTTP proxy providing data anonymity services to users. LPWA generates secure, consistent, and pseudonymous usernames, passwords, and email aliases for different websites based on three inputs: a UserID, a universal password to the proxy, and destination website address. Using LPWA, users can protect their real identities. It was successful, but it has limitations. LPWA does not support HTTPS. It also requires users to fully trust the proxy server, which knows all the login credentials to the destination servers, resulting in security and privacy concerns[2].

PwdHash is a browser extension that transparently creates a different password for each site, improving web password security and defending against phishing attacks. PwdHash uses the destination domain name as a salt and sends a hashed password to the remote site. However, PwdHash is vulnerable to two major kinds of attacks. One is a dictionary attack on the hashed passwords. This vulnerability is due to three factors: a phishing site can obtain hashed passwords.The second vulnerability is its susceptibility to advanced phishing attacks, such as using Flash objects or focus stealing. Flash objects and focus stealing are a form of phishing that allows keyboard strokes to be intercepted before other browser plugins have a chance to handle them[3].

As a browser extension, Password Multiplier can generate strengthened passwords for an arbitrary number of accounts while requiring the user to memorize only a single short password. It also uses the same three inputs as LPWA: a UserID, a master password, and a destination domain name. Password Multiplier is using a strengthened hash function to generate high-entropy passwords. But the main problem with Password Multiplier is that all the derived passwords will be known to attackers if the master password is stolen. Today, it is possible for an attacker to steal a master password through a keylogger or other spyware. Moreover, changing a password for a specific site is complicated because Password Multiplier requires users to remember additional information. Changing the master password also becomes tedious because the password on every site needs to be updated[4].

Built upon Password Multiplier and Petname Tool, Passpet turns a single master password into distinct passwords for different websites and uses petnames to help users recognize phishing attempts. In order to generate correct passwords, Passpet relies on a remote server to store site label files. But Passpet has the same drawback as Password Multiplier in terms of master password vulnerability. In addition, its remote storage server is vulnerable to various malicious attacks, which is acknowledged by the authors[5].

At present, to overcome from these problems which occurs in these hashing mechanism, Password Agent is come into existence and it improves password security.

III. SECURITY ANALYSIS

The primary goal of Password Agent is to improve user security. Here we compare the security of Password Agent with those of LPWA, PwdHash, Password Multiplier, and Passpet in ten different aspects as follows:

Unique Passwords: Each password hashing solution generates a unique password for each site, even if the plain-text password is the same.

Offline Attacks: Password Agent is less vulnerable to offline attacks. Because the salt list is not stored locally, launching an offline attack to retrieve the salt list is difficult. Password Multiplier and Passpet are also resistant to offline attacks as long as the local machine remains uncompromised. With Password Agent, even if the Agent password is stolen, only the salt list is revealed. The attacker would still need to launch an online attack against the target site to determine the site password.

Compromised Plain-text Password: When the plain-text password is compromised, only Password Agent still provides user protection. An attacker would be unable to use the compromised password, because the random site salt is not known. PwdHash does not have this feature, as the salt is the site's domain name, allowing an attacker to utilize the compromised password to access the site. Password Multiplier and Passpet both use one plain-text password as a master password to generate all of the site passwords..

Compromised Site Password: All password hashing schemes claim to protect users when a site password is compromised. However, because PwdHash uses MD5 and a known salt, the domain name, it is possible to launch a brute force attack on the compromised password. In contrast, Password Agent defends against offline attacks with a large random salt.

Basic Phishing Protection: The hash-based password generation allows all schemes to provide a basic level of phishing protection. Because each site password is unique, using any of these password generation tools on a phishing site will not immediately expose the login of the target site. LPWA, PwdHash, Password Multiplier, and Passpet all suffer from this problem. However, Password Agent offers the additional security with random salts, so even a stolen plain-text password will not give an attacker access to a login.

Advanced Phishing Protection: Password Agent provides early warning against phishing sites. If a user attempts to enter a protected password on an unregistered site, an information dialog notifies the user. This allows users to check if they are on a phishing site. Displaying security information in the browser chrome, Password Agent prevents its user interface from being spoofed by web pages. Because web pages do not have access to the browser chrome, it is difficult to place a fake login button or security indicator.

Shoulder Surfing Protection: Password Agent makes shoulder surfing—watching a user type in a password much more difficult to succeed, because it requires the observation of two separate events, the typing of the Agent password and the typing of the site password. Since the Agent password is entered only when the user begins a session, an attacker is forced to hover around the victim for longer periods of time, increasing the chance of detection. Other schemes, however, only require one password, making the attacker easier to succeed.

Secured Remote Storage: The Salt Repository of Password Agent is secure, and does not leak any useful information to attackers. But, Passpet leaks not only whether a username exists but also how large k1 is ,where k1 is the number of iterations of a hash function used for generating the site password. The smaller the k1, the weaker the password.With this knowledge, an attacker can target a user with a small k1 value and launch a brute force attack. Both Password Agent and Passpet store only encrypted data and guarantee the integrity of the data with a MAC. Even in a situation in which a Salt Repository is compromised, the leaked information would not be useful because the attacker would have to brute force the salt list and then launch an online attack against the site specific passwords.

Adaptation to Faster Computers: Password Agent can adapt to faster computers and the associated greater power of attackers in launching dictionary/brute force attacks, by increasing the salt size. This is a minor change to the Agent implementation. The user simply regenerates a longer salt while keeping the plain-text password intact. The newly-generated site password is stronger, and no extra memory burden is placed on the user. Both Passpet and Password Multiplier can increase the number of iterations to make it harder for an attacker to compute the site password.

Data Anonymity: Only LPWA has data anonymity as its goal. The others, including Password Agent, focus solely on password protection. LPWA enables a user to browse, hold accounts, and email without ever revealing personal identification information.

TABLE I
SECURITY COMPARISON OF PASSWORD AGENT WITH FOUR OTHER TOOLS

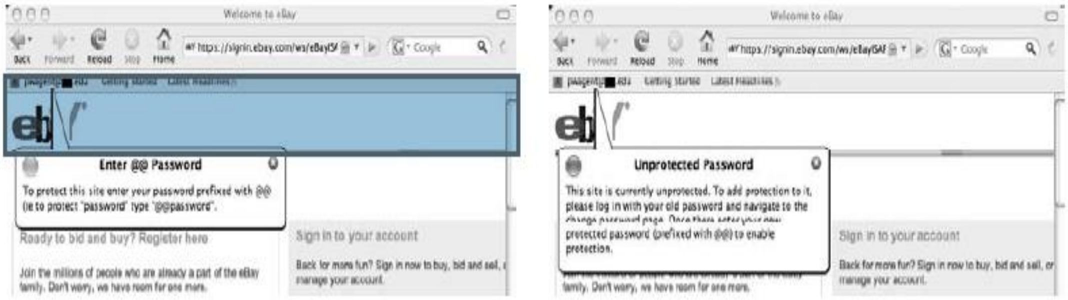
	Security	LPWA	PwdHash	Password Multiplier	Passpet	Password Agent
1.	Unique Password for Each Site	yes	yes	yes	yes	yes
2.	Resist Offline Attacks	-	-	no	yes	yes
3.	Protect Compromised Plain-text Password	no	no	no	no	yes
4.	Protect Compromised Site Password	yes	yes	yes	yes	yes
5.	Basic Phishing Protection	yes	yes	yes	yes	yes
6.	Advanced Phishing Protection	no	no	no	yes	yes
7.	Enhance Shoulder Surfing Protection	no	no	no	no	yes
8.	Secured Remote Storage	-	-	-	no	yes
9.	Adaptation to Faster Computers	no	no	yes	yes	yes
10.	Provide Data Anonymity	yes	no	no	no	no

IV. USABILITY ANALYSIS

Usability is a key factor in any software system. A simple usability flaw might render a cryptographically secure system useless. Care is taken in the development of Password Agent to address usability concerns that exist in previous password hashing solutions. The specific usability benefits of Password Agent are detailed as follows.

Ease of Site Password Updating: Password Agent allows users to change their site passwords exactly like they normally do, via the change password page of the website. By changing it to a new protected password, users maintain all the benefits of Password Agent without any complicated processes. PwdHash has the same function. In contrast, Password Multiplier forces users to append information to the domain name being hashed and additional information also that are using for logins. Passpet uses a similar mechanism, in which users can change the label of a site to change the password.

Notification of Protected Sites: Only Password Agent and Passpet notify users when a site requires protected passwords. Password Agent displays a “notification bubble”, which informs the user of the status of the site and how to login as shown in Figures 2(a) and 2(b). Both PwdHash and Password Multiplier fail to indicate whether a site is expecting a protected or plain-text password. Users who enter an incorrect password will often proceed to enter many of their other passwords, including plain-text passwords. This leads to multiple passwords being exposed, a situation that is even worse than if no password protection is used[6].



Enter @@ Password

To protect this site enter your password prefixed with @@ (ie to protect "password" type "@@password".

Ready to bid and buy? Register here

Join the millions of people who are already a part of the eBay family. Don't worry, we have room for one more.

Sign in to your account

Back for more fun? Sign in now to buy, bid and sell, or manage your account.

(a)

Unprotected Password

This site is currently unprotected. To add protection to it, please log in with your old password and navigate to the change password page. Once there enter your new protected password (prefixed with @@) to enable protection.

family. Don't worry, we have room for one more.

Sign in to your account

Back for more fun? Sign in now to buy, bid and sell, or manage your account.

(b)

Figure 2: User focused password fields: (a) on a protected site, (b) on an unprotected site.

Changing Master Password: The user can change the master password for the Salt Repository at any point without changing the password on any individual site. By entering the old and new Agent Password, the salt list can be decrypted and then re-encrypted with the new password. Because the same salts are used to generate the password, the site password remains the same. This is more convenient than in Passpet and Password Multiplier, where a change to the master password requires the user to login into each individual site and manually change the password.

Site Specific Password Requirements: Many sites have different password requirements, including size and acceptable characters. Only Password Agent and PwdHash examine the user's plain-text password for clues to the expected composition of a password. Any errors with the plain-text password are mirrored in the site password.

Minimal Changes to Browsing Paradigm: Password Agent makes only minimal changes to the normal interaction between a user and a web browser similar to PwdHash. The only two changes include: (1) the user must log into the Agent when beginning a session, and (2) the protected password must start with @@ (or the user must activate Password Agent via the F2 key). Password Multiplier and Passpet both require obvious deviations from the normal user login.

Ease of Switching Storage Servers: Password Agent is can easily transfer the salt list from one repository to another. In contrast, Passpet uses the storage server address as part of the master password generation, thus any change in the storage server address forces users to create a new master password and update all their site passwords.

TABLE III
USABILITY COMPARISON OF PASSWORD AGENT WITH FOUR OTHER TOOLS

	Usability	LPWA	PwdHash	Password Multiplier	Passpet	Password Agent
1.	Allow Easy Site Password Update	yes	yes	no	yes	yes
2.	Notify if Site is Protected	no	no	no	yes	yes
3.	Support all Site Specific Password Requirements	no	yes	no	no	yes
4.	Minimal Change to Browsing Paradigm	yes	yes	no	no	yes

V. CONCLUSION

Password Agent is an automatic password management system with enhanced hashing includes a Salt Repository and a browser plug-in Agent, and it provides a convenient and secure password protection service in an automatic manner. It automatically secures the user's plain-text password by rendering a unique site password for each website visited. Under the security guarantee, a user's site password is robustly defended against password cracking and theft. This paper clearly indicates the security and usability benefits of Password Agent.

REFERENCES

[1] Benjamin Strahs Chuan Yue Haining Wang Department of Computer Science .The College of William and Mary William 23187, US{*bgstra,cyue,hnw}@cs.wm.edu*

[2] GABBER, E., GIBBONS, P. B., KRISTOL, D. M., MATIAS, Y., AND MAYER, A. Consistent, yet anonymous, Web access with LPWA. Commun. ACM 42, 2 (1999), 42–47.

[3] ROSS, B., JACKSON, C., MIYAKE, N., BONEH, D., AND MITCHELL, J. C. Stronger password authentication using browser extensions. In Proceedings of the 14th USENIX Security Symposium (2005), pp. 17–32.

[4] *HALDERMAN, J. A., WATERS, B., AND FELTEN, E. W. A* convenient method for securely managing password. In Proceedings of 14th international conference on World Wide Web (2005), pp. 471-479.

[5] Petname Tool. *http://petname.mozdev.org/*.

[6] *CHIASSON, S., VAN OORSCHOT, P., AND BIDDLE, R.* A usability study and critique of two password managers. In Proceedings of the 15th USENIX Security Symposium (2006), pp. 1–16.

[7] *HERLEY, C., VAN OORSCHOT, P., AND PATRICK*, A. S. Passwords: If we're so smart, why are we still using them. In Proceedings of the Financial Cryptography and Data Security Conference (2009).