A Remote User Authentication Scheme Based on Strong Graphical Passwords

一個以強圖形化通行碼為基礎的遠端使用者身份認證設計

Wei-Chi Ku (顧維祺), Kuo-Tsai Chang (張國財), and Feng-Yuan Yang (楊豐源)
Dept of Computer Science and Information Engineering
Fu Jen Catholic University
輔仁大學資訊工程學系
E-mail: wcku@csie.fju.edu.tw

摘要

傳統使用強通行碼的遠端使用者身份認證設計之共同缺點為使用者必須記憶難以記憶的文字通行碼，故而侷限此類設計的應用。由於圖形化通行碼對使用者而言較容易記憶且可抵擋傳統的字典攻擊的優點，在本文中，我們提出了一個使用強圖形化通行碼的遠端使用者身份認證設計，此設計除了具有較傳統強通行碼遠端使用者身份認證設計更高的實用性外，我們並證明此設計具有良好的安全性與可恢復性。

關鍵詞：圖形化通行碼、遠端使用者身份認證、可恢復性、強通行碼。

Abstract

Conventional remote user strong-password authentication schemes have the common drawback that the user has to memorize a hard-to-remember textual password, and therefore their applications are restricted. To solve this problem, we propose a remote user authentication scheme using strong graphical passwords in this paper. As graphical passwords are easy to remember for the user and conventionally dictionary attacks on graphical passwords are infeasible, the practicability of the proposed scheme is improved. Next, we show that the proposed scheme can withstand the replay attack, the password-file compromise attack, the denial-of-service attack, the predictable *n* attack, and the insider attack. In particular, the proposed scheme is easily reparable.

Keywords: graphical password, remote user authentication, reparability, strong password.

1. Introduction

Password authentication is regarded as one of the simplest and most convenient remote user authentication mechanisms. To prevent direct wiretapping attacks in open network environments, many modern password authentication schemes use one-time passwords. Existing one-time password authentication schemes can be categorized into two types [1][8][9][12][16][24][25]: one requires only weak passwords and the other must use strong passwords. Weak-password authentication schemes usually lead heavy computational load to the whole application system because of using public-key cryptographic techniques [11]. In contrast, the computational load of most strong-password authentication schemes is lighter because of using only simple operations, e.g., one-way hash function and XOR operation. In addition, strong-password authentication schemes have another advantage over weak-password authentication schemes in that their implementations are easier and cost less, and therefore are especially suitable for some constrained environments.

Up to now, many strong-password authentication schemes have been proposed, e.g., Lamport's scheme [16], S/KEY [9], CINON [24], PERM [25], SAS [23], OSPA [17], and Lin-Shen-Hwang's scheme [18]. Unfortunately, none of these previous schemes is secure enough [5][17][21][25][27]. As analyzed in [5], the weaknesses of most previous strong-password authentication schemes are mainly due to two unsolved problems. First, if the adversary has stolen the verifier of a user, he can use it to impersonate the user to login. Secondly, the integrity of the messages transmitted from the user to the server for updating the user's verifier is not well protected so that the adversary can modify the transmitted messages without being de-

tected by the server. Without considering security issues, conventional strong-password authentication schemes, which use textual passwords, have the common drawback that the user has to memorize a hard-to-remember password. Recently, quite a few of graphical passwords, e.g., [3], [7], [13], and [28], have been proposed to solve this problem. The appeal of graphical passwords is primarily due to one's great memory for pictures over texts [19]. It is widely recognized that graphical passwords are easy to remember for the user and conventionally dictionary attacks on graphical passwords are infeasible because there are no existing workable dictionaries for graphical information. Herein, we will propose a strong-password authentication scheme that can solve the previously described two security problems. In particular, strong graphical passwords are used to improve the practicability of the proposed scheme by decreasing the memory burden of the user.

The sequel is organized as follows. In Section 2, we review several strong graphical password schemes. Section 3 describes the proposed scheme. Section 4 analyzes security of the proposed scheme. Finally, a conclusion is given in Section 5.

2. Graphical Passwords

Because weak textual passwords are susceptible to the dictionary attack and strong textual passwords are hard to remember, graphical passwords have been proposed as an alternative to textual passwords [26]. Psychological studies [4][14][20] showed that people recall and recognize pictures with higher probability than texts. Clearly, graphical password schemes can naturally resist conventional dictionary attacks [26]. And, if the number of possible pictures is sufficiently large and the diversity of picture-based passwords can be captured, the password space of a graphical password scheme may exceed that of textual password schemes.

In 1996, Blonder [3] initially proposed a graphical password scheme in which a password is a sequence of clicks at points in a predetermined image. In 1999, Jermyn et al. [13] proposed the DAS scheme, in which a password is a picture drawn on a two-dimensional grid. The coordinates of the touched grids are recorded in temporal order of the drawing. Once same cells are crossed with same order, the user is authenticated. In 2000, Dhamija and Perrig [7] proposed a graphical password scheme, Déjà Vu. In their scheme, which involves three phases: the portfolio creation phase, the training phase, and the authentication phase. In the portfolio creation phase, the user chooses a subset of images to be used for his password. In the training phase, the user becomes more familiar with the subset of images. In the authentication phase, the user picks out his portfolio images from a display of images consisting of his portfolio and decoys. The Passface™ scheme [22] is also a graphical password scheme in which the user can click the correct faces that are previously chosen by himself to authenticate him to the system. In 2002, De Angeli et al. [6] proposed a graphical password scheme, VIP, which designed to provide a promising and easy-to-use alternative to PIN approach. In 2005, Wiedenbeck et al. [28] proposed an improved version of Blonder's scheme, the PassPoints scheme, in which the user can use any image provided by the system or chosen by the user. The only requirement in practice is that the image be intricate and rich enough so that lots of possible click points are available. Unlike Blonder's scheme, the PassPoints scheme does not need artificial predefined click regions with well-marked boundaries.

3. The Proposed Scheme

In the proposed scheme, the PassPoints scheme [28], which is an improved version of Blonder's graphical password scheme [3], will be employed to implement the strong graphical password. The system or the user could provide the image, and the user may choose any place in the image as a click point. The user's password consists of any sequence of click points chosen by the user. In order to login, the user has to click close to the chosen click points within some tolerance distance. The image is discretized into squares that are large enough so that we can expect the user can hit the same square. However, this leaves the possibility that the user may choose a click point that happens to be close to an edge of a discretization square in which the tolerance distance of the click point will pass its own square boundary. Such a problem can be solved by using three discretization grids simultaneously [2]. If the image size is $a \times b$ pixels and the square size is $v \times w$ pixels. The number of squares is $(a \times b)/(v \times w)$, and the password space of z clicks at points is $P((a \times b)/(v \times w), z)$. However, in practice, not all areas of the image of PassPoints have memorable features. If only half areas are used, the password space is $P((a \times b)/(2 \times (v \times w)), z)$. In the proposed scheme, it is assumed that the image size is 1024×768 pixels, the square size is 14×14 pixels, and the number of clicks is 5, and therefore the size of the password space is $P(2006, 5) = 2^{54}$, which is larger than 2^{53}, the password space of textual passwords with 8 characters or less constructed from the printable ASCII codes, i.e., such PassPoints passwords can be used as strong passwords in the proposed scheme.

Figure 1 shows an example of the PassPoints password with five clicks.

Figure 1: An example of the PassPoints password.

The notations used in the proposed scheme are summarized as follows:

- *A* denotes the user.
- *ID* denotes the identity of *A*.
- *GPW* denotes the encoded graphical password of *A*.
- *S* denotes the remote server.
- *x* denotes the secret key of *S* used for generating a unique storage key for each user.
- *T* denotes the latest time *A* initially registers or re-registers to *S*.
- *N* denotes a sequence number starting from 1 in *U*'s initial registration.
- $\|$ denotes the concatenation operation.
- $\oplus$ denotes the bitwise XOR operation.
- $h(\ )$ represents a cryptographic hash function.
- 'U_1=>U_2: *data*' represents U_1 sends *data* to U_2 through a secure channel.
- '$U_1 \rightarrow U_2$: *data*' represents U_1 sends *data* to U_2 through a common channel.

Registration Protocol

This phase is invoked when *A* initially registers or re-registers to *S*.

Step R1. $A \rightarrow S$: registration request.

Step R2. *S* sets *T* to the value of his current time-stamp. If it is *A*'s initial registration, S sets *N* to 1. Otherwise, S sets $N = N + 1$.

Step R3. *S* =>*A* : *T*, *N*.

Step R4. *A* enters his graphical password by clicking points in the image to generate the corresponding code *GPW*, and then computes the verifier $V = h^2(S \| GPW \| N \| T)$.

Step R5. *A* =>*S*: *V*.

Step R6. *S* computes the storage key $k_A^{(T)} = h(ID \| h(x \| T))$, and then computes the sealed verifier $sv^{(N)} = h^2(S \| GPW \| N \| T) \oplus k_A^{(T)}$. Next, *S* stores $sv^{(N)}$, *T*, and *N* in his password file.

Login Protocol

This protocol is invoked whenever *A* logins *S*. Assume that $N = n$ and $T = t$.

Step L1. $A \rightarrow S$: *ID*, *rc*.

// *rc* is a random nonce selected by *A*. //

Step L2. *S* retrieves *t* from his password file and computes $k_A^{(t)} = h(ID \| h(x \| t))$, and then uses the computed $k_A^{(t)}$ to derive the verifier $h^2(S \| GPW \| n \| t)$ from the stored sealed verifier $sv^{(n)}$ $(= h^2(S \| GPW \| n \| t) \oplus k_A^{(t)})$. Next, *S* computes $h(h^2(S \| GPW \| n \| t) \oplus rc)$.

Step L3. $S \rightarrow A$: *n*, *rs*, $h(h^2(S \| GPW \| n \| t) \oplus rc)$, *t*.

// *rs* is a random nonce selected by *S*. //

Step L4. *A* enters his graphical password by clicking points in the image to generate the corresponding code *GPW*. Next, *A* computes $V = h^2(S \| GPW \| n \| t)$ and $h(V \oplus rc)$. If the computed $h(V \oplus rc)$ equals the received $h(h^2(S \| GPW \| n \| t) \oplus rc)$, *A* authenticates *S*. Otherwise, *A* terminates this session. Then, *A* computes

$$d_1 = h^2(S \| GPW \| n \| t) \oplus h(S \| GPW \| n \| t)$$

$$d_2 = h(S \| GPW \| n \| t) \oplus h^2(S \| GPW \| n + 1 \| t)$$

$$d_3 = h(h^2(S \| GPW \| n + 1 \| t) \| rs)$$

Step L5. $A \rightarrow S$: d_1, d_2, d_3.

Step L6. *S* uses previously derived verifier to compute

$$u_1 = d_1 \oplus h^2(S \| GPW \| n \| t)$$

$$u_2 = d_2 \oplus u_1$$

If $h(u_1)$ equals the retrieved $h^2(S \| GPW \| n \| t)$ and $h(u_2 \| rs) = d_3$ holds, *S* authenticates *A*. Otherwise, *S* rejects *A*'s login request and terminates this session. Then, *S* computes $sv^{(n+1)} = u_2 \oplus k_A^{(t)}$ $(= h^2(S \| GPW \| n + 1 \| t) \oplus k_A^{(t)})$, replaces $sv^{(n)}$ with $sv^{(n+1)}$, and sets $N = n + 1$ for *A*'s next login.

4. Security Analysis

Because the graphical passwords used in the pro-

posed are assumed to be strong, the proposed scheme can resist conventionally dictionary attacks. In addition, we will show that the proposed scheme can resist the replay attack, the password-file compromise attack [1], the denial-of-service attack, the predictable *n* attack [21], and the insider attack [15]. Furthermore, we will also explain that the proposed scheme is reparable [10].

4.1 Resistance to Replay Attack

Suppose that $N = n$ and the adversary has captured all A's past authentication messages $\{d_1^{(i)}, d_2^{(i)}, d_3^{(i)}\}$ for $i = 1, 2, \ldots,$ and $n-1$. Since A's current verifier stored in S is $h^2(S \| GPW \| n \| t)$, the adversary cannot login S by using $\{d_1^{(i)} (= h^2(S \| GPW \| i \| t) \oplus h(S \| GPW \| i \| t)), d_2^{(i)}, d_3^{(i)}\}$, where $1 \le i \le n-1$. Alternatively, if the adversary replaces the transmitting $d_2^{(n)}$ and $d_3^{(n)}$ with $d_2^{(i)}$ and $d_3^{(i)}$, where $i = 1, 2, \ldots,$ and $n-1$, during A's login, S will detect this fraudulence because $h((d_2^{(i)} \oplus (d_1^{(n)} \oplus h^2(S \| GPW \| n \| t))) \| rs^{(n)})$ does not equal $d_3^{(i)} (= h(h^2(S \| GPW \| i + 1 \| t) \| rs^{(i)}))$. Note that even if the adversary could fool S into replacing A's verifier $h^2(S \| GPW \| n \| t)$ with $h^2(S \| GPW \| i \| t)$, where $1 \le i \le n-1$, by some means, the adversary cannot impersonate A to login S because $rs^{(n)} \ne rs^{(i)}$, which implies $h((d_2^{(i)} \oplus (d_1^{(i)} \oplus h^2(S \| GPW \| i \| t))) \| rs^{(n)}) \ne d_3^{(i)}$. On the other hand, since the adversary doesn't know $h^2(S \| GPW \| n \| t)$, he cannot generate and send the correct $h(h^2(S \| GPW \| n \| t) \oplus rc)$ to A in Step L3 after receiving the rc sent from A in Step L1. That is, the adversary cannot successfully impersonate S to cheat A by mounting such a replay attack. Therefore, the proposed scheme can resist the replay attack.

4.2 Resistance to Password-File Compromise Attack

Suppose that S's password file was compromised to the adversary, say E, i.e., E has obtained A's $T (= t)$, $N (= n)$, and the sealed verifier $sv^{(n)} (= h^2(S \| GPW \| n \| t) \oplus k_A^{(t)})$. Clearly, t and n are not secrets. As $sv^{(n)} = h^2(S \| GPW \| n \| t) \oplus k_A^{(t)}$, E can derive $h^2(S \| GPW \| n \| t)$ from $sv^{(n)}$ only if he knows $k_A^{(t)} = h(ID \| h(x \| t))$, which implies that he knows x. As assumed, x is under strict protection, and therefore, the proposed scheme can resist the password-file compromise attack. In practice, if x, the top secret of the system, is compromised, the whole system should be reinitialized, and all users should choose their new graphical passwords and register to S again.

4.3 Resistance to Denial-of-Service Attack

To prevent the denial-of-service attack, S has to make sure that the computed u_2, which will be used as A's next verifier, is authentic. Since d_3 can protect the integrity of d_1 and d_2, which are used to compute u_2, any unauthorized modification on d_1, d_2, or d_3 will be detected by S. As the adversary cannot disable A's account, the proposed scheme can resist the denial-of-service attack.

4.4 Resistance to Forgery Attack

To mount a forgery attack on the proposed scheme, the adversary must generate the authentication message corresponding to the given n and rs. Since the adversary knows neither GPW nor $h(S \| GPW \| n \| t)$, he cannot produce the correct $\{d_1, d_2, d_3\}$ that will be accepted by S. Hence, the proposed scheme can resist the forgery attack.

4.5 Resistance to Predictable n Attack

Because the n used in the proposed scheme is predictable, the adversary may try to mount a bigger n attack or a smaller n attack as follows. Upon observing A's login request message sent in Step L1, the adversary impersonates S to reply n' $(> n)$ and rs', which is randomly selected, to A in Step L3. Then, A will be fooled into generating and sending $\{d_1^{(n')}, d_2^{(n')}, h(h^2(S \| GPW \| n' + 1 \| t) \| rs')\}$ to the adversary. However, since $rs^{(n')}$ will be randomly selected by S, it will not equal rs', i.e., $d_3^{(n')} (= h(h^2(S \| GPW \| n' + 1 \| t) \| rs^{(n')}))$ will not equal $h(h^2(S \| GPW \| n' + 1 \| t) \| rs')$. Thus, the proposed scheme can resist the bigger n attack. In addition, since the adversary cannot benefit by using n'' $(< n)$ to fool A into replying the corresponding authentication message, the smaller n attack is meaningless to the proposed scheme. That is, the proposed scheme can resist the predictable n attack.

4.6 Reparability

Suppose that the adversary has captured all A's past authentication messages. If the adversary also knows an ever used verifier, say $h^2(S \| GPW \| i \| t)$, where $1 \le i \le n-1$, for A by some means, he can compute all A's verifiers, including the current one $h^2(S \| GPW \| n \| t)$. In this case, the adversary can compute $h^2(S \| GPW^* \| n \| t)$, where GPW^* is a graphical password selected by the adversary, and then fool S into replacing A's verifier $h^2(S \| GPW \| n \| t)$ with the computed $h^2(S \| GPW^* \| n \| t)$. Then, the adversary can impersonate A to login S. However, once A finds this fraudulence, he can re-registers to S so that S will renew A's verifier with $h^2(S \| GPW \| n + 1 \| t')$, where t' is the value of S's current timestamp. Because the adversary cannot compute A's new verifier from A's compromised verifiers, his login request will be rejected. In addition, since S will also renew the storage key $k_A^{(t)}$ with $k_A^{(t')} = h(ID \| h(x \| t'))$, the adversary cannot compute $h^2(S \| GPW \| n + 1 \| t')$ even if he has ever obtained the old storage key $k_A^{(t)}$ and stolen A's updated sealed verifier

$sv^{(n+1)} = h^2(S \parallel GPW \parallel n + 1 \parallel t') \oplus k_A^{(t')}$. Therefore, the improved scheme is easily reparable [10].

4.7 Resistance to Insider Attack

In practice, it is likely that *A* uses the same graphical password *GPW* to access several servers for his convenience. If the insider of *S* has obtained *GPW*, he can impersonate *A* to access other servers. Actually, the insider of *S* cannot obtain *GPW* directly in that *A* will not reveal *GPW* to *S* in the registration protocol and the login protocol. Furthermore, as *GPW* is equivalent to a strong password, the insider of *S* cannot derive *GPW* by performing a conventionally dictionary attack on what he has received from *A*. Alternatively, since the insider of *S* knows *A*'s verifier $h^2(S \parallel GPW \parallel n \parallel t)$, he can try to impersonate *A* to login another server, say S^*. However, $h^2(S \parallel GPW \parallel n \parallel t)$ will not equal $h^2(S^* \parallel GPW^* \parallel n^* \parallel t^*)$ even if $GPW = GPW^*$, $n = n^*$, and $t = t^*$, and therefore, the insider of *S* cannot successfully impersonate *A* to login S^*. Thus, the proposed scheme can resist the insider attack.

5. Conclusion

We have proposed a remote user authentication scheme using strong graphical passwords. The proposed scheme can withstand the replay attack, the password-file compromise attack, the denial-of-service attack, the predictable *n* attack, and the insider attack. In particular, the proposed scheme is easily reparable. Note that the proposed scheme is secure under the assumption that the easy-to-remember PassPoints password is strong. Although conventional dictionary attacks on graphical passwords are infeasible because there are no existing workable dictionaries for graphical information, the resistance of the proposed scheme to specific graphical dictionary attacks has to be studied in future research. Furthermore, the resistance to shoulder surfing attacks may be considered in the improved version of the proposed scheme.

References

[1] S. Bellovin and M. Merritt, "Augmented encrypted key exchange: A password-based protocol secure against dictionary attacks and password-file compromise," *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 244–250, 1993.

[2] J. C. Birget, D. Hong, N. Memon, "Robust discretization, with an application to graphical password," *Cryptologye Print Archive*, 2003. http://eprint.iacr.org/2003/168

[3] G. Blonder, "Graphical passwords," *United States Patent* 5559961, 1996.

[4] M. W. Calkins. "Short studies in memory and association," *Psychological Review*, vol. 5, pp. 451–462, 1898.

[5] C. M. Chen and W. C. Ku, "Stolen-verifier attack on two new strong-password authentication protocols," *IEICE Transactions on Communications*, vol. E85-B, no. 11, pp. 2519–2521, Nov. 2002.

[6] A. De Angeli, M. Coutts, L. Coventry, G. I. Johnson, 2002 "VIP: a visual approach to user authentication," *Proceeding of the Working Conference on Advanced Visual Interfaces AVI*. ACM Press, New York, pp. 316–323. 2002.

[7] R. Dhamija and A. Perrig. Déjà Vu: "A user study using images for authentication," *Proceedings of the 9th USENIX Security Symposium*, 2000.

[8] L. Gong, "Optimal authentication protocols resistant to password guessing attacks," *Proceedings of the 8th IEEE Computer Security Foundation Workshop*, pp. 24–29, 1995.

[9] N. M. Haller, "The S/KEY (TM) one-time password system," *Proceedings of the Internet Society Symposium on Network and Distributed System Security*, pp. 151–158, 1994.

[10] T. Hwang and W. C. Ku, "Reparable key distribution protocols for Internet environments," *IEEE Transactions on Communications*, vol. 43, no. 5, pp. 1947–1949, May 1995.

[11] IEEE P1363.2 / D11, "Standard specifications for password-based public-key cryptographic techniques," *IEEE P1363 working group*, Aug. 2003.

[12] D. Jablon, "Strong password-only authenticated key exchange," *ACM Computer Communications Review*, vol. 20, no. 5, pp. 5–26, 1996.

[13] Jermyn, A. Mayer, F. Monrose, M. Reiter, and A. Rubin, "The design and analysis of graphical passwords," *Proceedings of the 8th USENIX Security Symposium*, 1999.

[14] E. A. Kirkpatrick, "An experimental study of memory," *Psychological Review*, vol. 1, pp. 602–609, 1894.

[15] W. C. Ku, C. M. Chen, and H. L. Lee, "Cryptanalysis of a variant of Peyravian-Zunic's password authentication scheme," *IEICE Transactions on Communications*, vol. E86-B, no. 5, pp. 1682–1684, May 2003.

[16] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, Nov. 1981.

[17] C. L. Lin, H. M. Sun, and T. Hwang, "Attacks and solutions on strong-password authentication," *IEICE Transactions on Communications*, vol. E84-B, no. 9, pp. 2622–2627, Sept. 2001.

[18] C. W. Lin, J. J. Shen, and M. S. Hwang, "Security enhancement for optimal strong password authentication protocol," *ACM Operating Systems Review*, vol. 37, no. 2, pp. 7–12, April 2003.

[19] S. Madigan, "Picture memory," *Imagery, Memory and Cognition*, pp. 65–89, 1983.

[20] S. Madigan and V. Lawrence, "Factors affecting item recovery and hypermnesia in free recall," *American Journal of Psychology*, vol. 93, pp. 489−504, 1980.

[21] C. J. Mitchell and L. Chen, "Comments on the S/KEY user authentication scheme," *ACM Operating Systems Review*, vol. 30, no. 4, pp. 12−16, Oct. 1996.

[22] Real User Corporation, "The Science Behind Passfaces," June 2004. *Available at http://www.realuser.com/published/ScienceBehindPassFaces.pdf*

[23] M. Sandirigama, A. Shimizu, and M. T. Noda, "Simple and secure password authentication protocol (SAS)," *IEICE Transactions on Communications*, vol. E83-B, no. 6, pp. 1363–1365, June 2000.

[24] A. Shimizu, "A dynamic password authentication method by one-way function," *IEICE Transactions*, vol. J73-D-I, no. 7, pp. 630–636, July 1990.

[25] A. Shimizu, T. Horioka, and H. Inagaki, "A password authentication methods for contents communication on the Internet," *IEICE Transactions on Communications*, vol. E81-B, no. 8, pp. 1666–1673, Aug. 1998.

[26] J. Thorpe and P. van Oorschot, "Graphical dictionaries and the memorable space of graphical passwords," *Proceedings of the 13th USENIX Security Symposium*, 2004.

[27] T. Tsuji and A. Shimizu, "An impersonation attack on one-time password authentication protocol OSPA," *IEICE Transactions on Communications*, vol. E86-B, no. 7, pp. 2182–2185, July 2003.

[28] S. Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy and N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system," *International Journal of Human-Computer Studies*, vol. 63, pp. 102–127, July 2005.