Weaknesses of Lee-Li-Hwang's Hash-Based Password Authentication Scheme

Wei-Chi Ku Chien-Ming Chen Hui-Lung Lee
Department of Computer Science and Information Engineering
Fu Jen Catholic University
510 Chung Cheng Rd., Hsinchuang, Taipei County, Taiwan 242, R.O.C.
Email: wcku@csie.fju.edu.tw

Abstract

Many password authentication schemes employ hash functions as their basic building blocks to achieve better efficiency. In 2000, Peyravian and Zunic proposed a hash-based password authentication scheme that is efficient and can be easily implemented. Recently, Lee, Li, and Hwang demonstrated that Peyravian-Zunic's hash-based password authentication scheme is vulnerable to the off-line guessing attack, and then proposed an improved version. In this article, we show that their improved scheme is still vulnerable to the off-line guessing attack, the denial-of-service attack, and the stolen-verifier attack.

Keywords: Password authentication, hash function, guessing attack, denial-of-service attack, stolen-verifier attack

I Introduction

Password authentication is regarded as one of the simplest and most convenient authentication mechanisms. Conventional static password authentication methods can not resist direct wiretapping attacks, and thus, are unsuitable for open network environments. To meet today's security requirements, many password authentication methods using dynamic, or one-time, passwords have been proposed. Existing dynamic one-time password authentication schemes can be categorized into two types, one [3] employs cryptosystems, either public-key cryptosystems or secret-key cryptosystems, and the other [4,9,11,15,16,17] employs only simple operations, e.g., one-way hash function [12,14,18] and XOR (exclusive-or) operation. Although the latter type, the hash-based password authentication scheme, usually requires that users should choose strong passwords, which can not be easily guessed, however, it has the advantage over the former type in that its computation is lighter, design is simpler, and implementation is easier, and therefore is especially suitable for certain constrained environments.

The first well-known hash-based password authentication scheme was proposed by Lamport [9]. This scheme allows the server to authenticate the user in a way that neither eavesdropping on an authentication exchange nor reading server's database enables someone to impersonate the user. However, high hash overhead and the necessity for password resetting decrease its suitability for practical use. Additionally, Lamport's scheme is vulnerable to the replay attack. Later, Haller [4]

proposed a deployed version of Lamport's scheme, the S/KEY. Like Lamport's scheme, S/KEY is also vulnerable to the replay attack. To eliminate the drawbacks of Lamport's scheme and S/KEY, Shimizu [16] proposed a one-time password authentication scheme, CINON. The one-time characteristic is gained by using two variable random numbers that are changed at each authentication. However, the user has to either memorize two variable random numbers or carry with some sort of portable storage tokens, e.g., floppy disks or IC cards. This inconvenience obstructs the deployment of CINON. Next, Shimizu *et al.* [17] proposed a token-free one-time password authentication scheme, PERM. The user doesn't need to either memorize any random number or carry with a portable storage token. Instead, a random number is stored in the server for authenticating the user. It is only when the server receives the correct reply corresponding to the sent random number, he will believe that the user is authentic and then refresh the stored random number. Unfortunately, PERM is subject to the man-in-the-middle attack in that the adversary can impersonate user by modifying two consecutive sessions between the user and the server. However, by using weak passwords, none of the above mentioned hash-based password authentication schemes can resist the password guessing attack.

In 2000, Sandirigama *et al.* [15] proposed a simple hash-based strong-password authentication scheme, SAS, which was intended to be superior to several well-known similar schemes, e.g., S/KEY, CINON, and PERM, in storage utilization, processing time, and transmission overhead. However, SAS has been found to be vulnerable to the replay attack and the denial-of-service attack [7,11]. In addition, Lin *et al.* [11] also proposed a refined scheme, the OSPA (Optimal Strong-Password authentication) scheme, which was asserted to be secure against the stolen-verifier attack, the replay attack, and the denial-of-service attack. Unfortunately, the OSPA scheme is also found to be vulnerable to the stolen-verifier attack [2] and the man-in-the-middle attack [19]. Independently, Peyravian and Zunic [13] also proposed a hash-based password authentication scheme. Since the associated operations are relatively simple, their scheme is efficient and can be easily implemented. Later, Hwang and Yeh [5] showed that Peyravian-Zunic's scheme is vulnerable to the off-line guessing attack, the server spoofing attack, and the stolen-verifier attack, and then proposed a modified version, which additionally uses the public-key cryptosystem. Clearly, Hwang-Yeh's scheme violates the original expectation that only simple operations are used. Moreover, it has been found [8] that Hwang-Yeh's scheme has several weaknesses. Recently, Lee, Li, and Hwang [10] proposed another improvement of the Peyravian-Zunic's scheme, and claimed that their scheme is secure against the off-line guessing attack. Unfortunately, we find that their improved scheme, which is referred to as Lee-Li-Hwang's scheme hereafter, is still vulnerable to the off-line guessing attack, the denial-of-service attack, and the stolen-verifier attack [1,2,11]. In this article, we will describe the weaknesses of Lee-Li-Hwang's scheme.

II Review of Lee-Li-Hwang's Scheme

In 2002, Lee, Li, and Hwang [10] proposed a hash-based password authentication scheme, Lee-Li-Hwang's scheme, which is claimed to be an improved version of the Peyravian-Zunic's scheme. Before demonstrating the weaknesses of Lee-Li-Hwang's scheme, we first brief review it for readers' convenience.

Let C represent the client, S represent the server, and E represent the adversary. Notations id and pw denote the identity and the password of C, respectively. And, r_c and r_s denote the random numbers

generated by C and S, respectively. H denotes a hash function. Notation $\oplus$ represents the bitwise XOR operation. Initially, S stores $hpw = H(id, pw)$ as the verifier for pw. The scheme involves two protocols, the protected password transmission protocol and the protected password change protocol, which can be briefly described as in the following.

Protected Password Transmission Protocol

The protected password transmission protocol is invoked whenever C wants to access the resources at S by using pw.

Step 1. $C \rightarrow S$: $id, r_c \oplus hpw$

Step 2. $C \leftarrow S$: $r_s \oplus hpw$

Step 3. $C \rightarrow S$: $id, H(hpw, r_c, r_s)$

Step 4. $C \leftarrow S$: *access granted/denied*

If the $H(hpw, r_c, r_s)$ received in Step 3 equals the expected one, S accepts C's request and sends 'access granted' to C in Step 4.

Protected Password Change Protocol

The protected password change protocol is invoked whenever C wants to change pw with a new one, say pw_{new}.

The steps are the same as within the protected password transmission protocol except that Step 3 is replaced by Step 3′ as in the following:

Step 3′. $C \rightarrow S$: $id, H(hpw, r_c, r_s), hpw_{new} \oplus H(hpw, r_c+1, r_s)$

where $hpw_{new} = H(id, pw_{new})$. If the received $H(hpw, r_c, r_s)$ in Step 3′ equals the excepted one, S accepts C's request. Next, S retrieves hpw_{new} from the received $hpw_{new} \oplus H(hpw, r_c+1, r_s)$, replaces the verifier hpw with hpw_{new}, and sends 'access granted' to C in Step 4.

III Weaknesses of Lee-Li-Hwang's Scheme

In this section, we respectively show that Lee-Li-Hwang's scheme is vulnerable to the off-line guessing attack, the denial-of-service attack, and the stolen-verifier attack [1,2,11].

Off-Line Guessing Attack

Suppose that the adversary E has intercepted $id, r_c \oplus hpw, r_s \oplus hpw$, and $H(hpw, r_c, r_s)$ in a previous run of the protected password transmission protocol. E can guess a password pw', and then compute

$$hpw' = H(id, pw').$$

Next, he can compute

$$H(hpw', r'_c, r'_s),$$

where

$$r'_c = hpw' \oplus (r_c \oplus hpw)$$

$$r'_s = hpw' \oplus (r_s \oplus hpw),$$

and then compare the result with the intercepted $H(hpw, r_c, r_s)$. If these two values are equal, E has correctly guessed C's password, i.e., $pw' = pw$. Otherwise, E can try another guess for pw again. Thus, Lee-Li-Hwang's scheme can not effectively resist the off-line guessing attack as its authors claimed.

Note that the above attack is based on the assumption that pw is a *weak password* [3,6], which can be easily guessed. If pw is a *strong password* [2,11], which can not be guessed easily, such an attack may be infeasible. However, since it is usually difficult to memorize a strong password, C probably has to additionally use a tamper-resistant storage token to carry pw securely. If so, the advantage of using a password authentication scheme is diminished.

Denial-of-Service Attack

Whenever C wants to change his password pw with a new one, say pw_{new}, the protected password change protocol is invoked. During Step 3′, E can replace the transmitting

$$hpw_{new} \oplus H(hpw, r_c+1, r_s)$$

with any equal-sized number, say r_E . The id and $H(hpw, r_c, r_s)$ sent in Step 3′ are left unchanged. Then, S will be fooled into believing that the entire message received in Step 3′ is really sent by C because the received $H(hpw, r_c, r_s)$ equals the expected one. Consequently, S will send 'access granted' to C in Step 4 and change the verifier hpw with

$$r_E \oplus H(hpw, r_c+1, r_s),$$

which clearly does not equal hpw_{new} (= $H(id, pw_{new})$). From now on, C's succeeding requests for either accessing the resources or changing password will be denied by S.

Hence, the adversary can easily lock the account of any client without using cryptographic techniques. Since this attack does not require password guessing, Lee-Li-Hwang's scheme is vulnerable to the denial-of-service attack regardless of whether pw is strong or weak.

Stolen-Verifier Attack

If pw is a weak password and E has stolen the verifier hpw (= $H(id, pw)$), E can find pw by employing the off-line guessing attack, in which each guess for pw can be verified with $H(id, pw)$. In this case, Lee-Li-Hwang's scheme is vulnerable to the stolen-verifier attack [6, 11, 12]. On the other hand, if the client chooses a strong password as pw, such a simple form of the stolen-verifier attack will be inhibited.

However, we find that Lee-Li-Hwang's scheme still suffers from another form of the stolen-verifier attack even if pw is a strong password. In the protected password transmission protocol, E can randomly select r_E, and send id and

$$r_E \oplus hpw$$

to S in Step 1. Next, S retrieves r_E from the second item of the received message by using hpw, and then sends

$$r_s \oplus hpw$$

to E in Step 2. E retrieves r_s from the received message by using the stolen hpw, computes

$$H(hpw, r_E, r_s),$$

and then sends id and the computed result to S in Step 3. Since the received message equals the expected one, S accepts E's request and sends 'access granted' to E in Step 4.

In addition, the stolen-verifier attack can also be mounted on the protected password change protocol in the same way except for Step 3′ as follows. E can select a password, say pw_E, compute

$$H(hpw, r_E, r_s)$$

$$hpw_E \oplus H(hpw, r_E+1, r_s),$$

where

$$hpw_E = H(id, pw_E),$$

and then send id accompanied with these two computed results to S in Step 3′. As the received $H(hpw, r_E, r_s)$ equals the expected one, S will be fooled into changing C's verifier hpw with hpw_E. Therefore, E can impersonate C to access the resources at S and/or change C's password at will.

IV Conclusion

To achieve better efficiency, many password authentication schemes employ hash functions as their basic building blocks. So far, many hash-based password authentication schemes have been proposed.

Unfortunately, most of these schemes have been found insecure. Herein, we have shown that a new hash-based password authentication scheme, Lee-Li-Hwang's scheme, is vulnerable to the off-line guessing attack, the denial-of-service attack, and the stolen-verifier attack. Moreover, even if strong passwords instead of weak passwords are used, the denial-of-service attack and the stolen-verifier attack still work.

References

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

[2] C. M. Chen and W. C. Ku, "Stolen-verifier attack on two new strong-password authentication protocols," *IEICE Transactions on Communications*, vol. E58-B, no. 11, pp. 2519–2521, Nov. 2002.

[3] Draft D2002-12-20 of IEEE P1363.2 (Standard specifications for public key cryptographic: password-based techniques), *IEEE P1363 working group*, 2002.

[4] N. M. Haller, "A one-time password system," *RFC 1704*, 1994.

[5] J. J. Hwang and T. C. Yeh, "Improvement on Peyravian-Zunic's password authentication schemes," *IEICE Transactions on Communications*, vol. E85-B, no. 4, pp. 823–825, 2002.

[6] S. Keung and K.Y. Siu, "Efficient protocols secure against guessing and replay attacks," in *Proceedings of the 4th International Conference on Computer Communications and Networks*, pp. 105–112, 1995.

[7] W. C. Ku and C. M. Chen, "Cryptanalysis of a one time password authentication protocols," in *Proceedings of the 2001 National Computer Symposium*, Taiwan, pp. F046–F050, Dec. 2001.

[8] 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.

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

[10] C. C. Lee, L. H. Li, and M. S. Hwang, "A remote user authentication scheme using hash functions," *ACM Operating Systems Review*, vol. 36, no. 4, pp. 23–29, Oct. 2002.

[11] 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.

[12] National Institute of Standards and Technology, "Secure hash standard," *FIPS Publication 180-1*, April 1995.

[13] M. Peyravian and N. Zunic, "Methods for protecting password transmission," *Computers & Security*, vol. 19, no. 5, pp. 466–469, 2000.

[14] R. Rivest, "The MD5 message-digest algorithm," *RFC 1321*, April 1992.

[15] 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.

[16] A. Shimizu, "A dynamic password authentication method by one-way function," *IEICE Transac-*

tions, vol. J73-D-I, no. 7, pp. 630–636, July 1990.

[17] 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.

[18] W. Stallings, Cryptography and Network Security: Principles and Practice, Second Edition, Prentice-Hall, 1999.

[19] T. Tsuji and A. Shimizu, "An impersonation attack on one-time password authentication protocol OSPA," *to appear in IEICE Transactions on Communications*, vol. E86-B, no. 7, July 2003.