Enhancing of a Password-Based Authentication Scheme Using Smart Cards*

Youngsook Lee[1,**] and Dongho Won[2,***]

[1] Department of Cyber Investigation Police, Howon University, Korea
ysooklee@howon.ac.kr
[2] Department of Computer Engineering, Sungkyunkwan University, Korea
dhwon@security.re.kr

Abstract. A password based remote user authentication scheme is a two-party protocol whereby an authentication server in a distributed system confirms the identity of a remote individual logging on to the server over an untrusted, open network. This paper discusses the security of Chen et al.'s remote user authentication scheme making use of smart cards. They have recently presented an improved version of Lin, Shen, and Hwang's scheme. But, unlike their claims, in Chen et al.'s scheme, if an attacker gains access to some user's smart card and extracts the information stored in the smart card, he/she can easily find out the user's password. We show this by mounting a dictionary attack on the scheme. In addition, Chen et al.'s scheme does not support its main security goal of authenticating between a remote individual and the server. This is shown via a sever impersonation attack on the scheme. Motivated by these security flaws, we propose a more secure remote user authentication scheme that achieves both two-factor security and mutual authentication.

Keywords: Authentication scheme, smart card, dictionary attack, impersonation attack, two-factor security.

1 Introduction

In 2000, Sandirigama et al. [17] proposed a simple and secure password authentication scheme, so called SAS, which exhibits lower storage, processing and transmission overheads. However, Lin et al. [13] have pointed out that Sandirigama et al.'s scheme is vulnerable to the replay attack and the denial of service attack. To fix these security problems, Lin et al. have presented a new scheme called OSPA (Optimal Strong Password Authentication), and have claimed, among others, that their proposed scheme achieves the property of the strong authentication. Unlike their claim, however, Chen and Ku [2] showed that neither the SAS protocol nor the OSPA could withstand a stolen verifier attack. Recently, Lin, Shen, and Hwang [12] proposed a modified OSPA to repair the security flaw of OSPA. But, unfortunately, this enhancement has still security problem.

* This work was supported by Howon University in 2009.
** The first author.
*** Corresponding author.

R. Meersman, T. Dillon, P. Herrero (Eds.): OTM 2009, Part II, LNCS 5871, pp. 879–886, 2009.
© Springer-Verlag Berlin Heidelberg 2009

For preventing these security problems, Chen et al. [5] has recently presented an improved version of Lin, Shen, and Hwang's [12] scheme.

However, despite many impressive achievements [7,1,15,8,11], it still remains a challenging problem to design a secure remote user authentication scheme [3,9,18,6]. Our observation is that the above stated remote user authentication scheme, Chen et al. [5]'s scheme, does not satisfy its fundamental goal of a password security. We demonstrate this by showing that the scheme is vulnerable to a password guessing attack in which an attacker U_a can easily find out the password of the user U_i. In addition, Chen et al.'s scheme does not support its main security goal of authenticating between a remote individual and the server. We show this by mounting a server impersonation attack on Chen et al.'s scheme. Motivated by this observation, we propose a more secure remote user authentication scheme that achieves both two factor security and mutual authentication.

2 Review of Chen et al.'s Authentication Schemes

Chen et al. [5] has recently presented an improved version of Lin, Shen, and Hwang's [12] scheme. Besides preventing the the denial-of-service attack, the scheme intends to improve more efficiently than Lin, Shen, and Hwang's scheme. We begin by describing the top level structure of the scheme. This scheme by proposed Chen et al. consists of two phases: registration phase and authentication phase. The registration phase is performed only once per user when a new user registers itself with the remote server. The authentication phase is carried out whenever a user wants to gain access to the server.

Before the registration phase is performed for the first time, the server S decides on the following system parameters: a one-way hash function h, a cryptographic key x. The key x is kept secret by the server.

2.1 Registration Phase

The registration of a new user U_i to the server S proceeds as follows:

Step 1. A user U_i, who wants to register with the server S, chooses its password PW_i and a random nonce N_1, computes $Z_i = h^2(PW_i||N_1)$, and submits a registration request, consisting of its identity ID_i, Z_i and N_1, to the remote server S via a secure channel.

Step 2. Upon receiving the request, the server S first stores Z_i in its database, computes $K_i = h(x||ID_i)$, and issues a smart card containing $\langle N_1, K_i, h(\cdot)\rangle$ to the user U_i.

2.2 Authentication Phase

When U_i wants to log in to the server, it inserts its smart card into a card reader and enters its identity ID_i and password PW_i. With the user input, the scheme enters the authentication phase during which the server and the smart card perform the following steps:

Step 1. Firstly, the smart card, that wants to identify this transaction, generates a random nonce N_2 and sends the message $\langle ID_i, N_2 \rangle$ to the server S.

Step 2. When $\langle ID_i, N_2 \rangle$ arrives, S first checks that ID_i is valid. If it is untrue, S rejects the request. Otherwise, the server chooses a random nonce R, computes $A_i = R \oplus h(x||ID_i)$ and $B_i = h(N_2||R)$, and sends the response message $\langle A_i, B_i \rangle$ to the user U_i.

Step 3. Upon receiving $\langle A_i, B_i \rangle$, the smart card computes $R' = A_i \oplus K_i$ and then checks that B_i is equals to $h(N_2||R')$. If it is untrue, the smart card aborts the scheme. Otherwise, given PW_i, the smart card again generates a new random nonce N_3, and computes $Z_i = h^2(PW_i||N_1)$, $C_1 = h(PW_i \oplus N_1) \oplus h(Z_i \oplus R')$, $C_2 = h^2(PW_i \oplus N_3) \oplus h(PW_i \oplus N_1)$, and $C_3 = h^3(PW_i \oplus N_3)$. The smart card then sends the login request message $\langle C_1, C_2, C_3 \rangle$ to the server S.

Step 4. After receiving $\langle C_1, C_2, C_3 \rangle$, S first computes $D_1 = C_1 \oplus h(Z_i \oplus R)$ and $D_2 = C_2 \oplus D_1$. Then S verifies that: (1) Z_i equals $h(D_1)$ and (2) C_3 equals $h(D_2)$. If one of these conditions is untrue, S rejects the login request. Otherwise, S accepts the login request and substitutes D_2 for Z_i stored in the its database.

3 Attacks on Chen et al.'s Scheme

There are two threats in between a remote user and an authentication server communications: (1) eavesdropping - an attacker intercepts or reads the message exchanged between the remote user and the server; (2) impersonation - an attacker disguises a legitimate remote user or the authentication server. It is a real threat to communications because of the possibility of losing some user's smart card or extracting the secret values stored in it [10,16].

Accordingly, there are two basic security requirements for a password-based scheme in order to authenticate remote user using smart cards. One is that the password should not be transmitted in clear text between the user and the server, otherwise it has the risk of being disclosed. The other is that the secret information stored in the smart cards should be well protected so that the password would not be leaked even if the smart cards are lost or stolen and all the values stored in it are revealed.

But unfortunately, Chen et al.'s [5] scheme does not achieve its fundamental goal of a password security [19]. We demonstrate this by showing that the scheme is vulnerable to a password guessing attack in which an attacker U_a can easily find out the password of the user U_i. In addition, Chen et al.'s scheme does not support its main security goal of authenticating between a remote individual and the server. We show this by mounting a server impersonation attack on Chen et al.'s scheme. In describing these attacks, we assume that the attacker has gained access to the U_i's smart card and extracted the secret values stored in it by monitoring its power consumption.

3.1 Password Guessing Attack on Chen et al.'s Scheme

In Chen et al.'s scheme, an attacker U_a can easily figure out the password of some user U_i by obtaining the extracted values N_1 and K_i from its smart card. The attack is described step by step as follows:

1. After accepting the server as authentic by verifying the authenticity of the received message $\langle A_i, B_i \rangle$ from S, U_i sends $\langle C_1, C_2, C_3 \rangle$ as a login request to the server S.
2. However, at that moment, U_a reads these exchanged messages between the server and the remote user and thereby obtains all of them. With $K_i (=h(x \,||ID_i))$ extracted from U_i's smart card and A_i obtained from the server's response message, U_a recovers R as $R = A_i \oplus K_i$.
3. Finally, U_a, who has extracted N_1 from U_i's smart card and obtained C_1 from the U_i's login message, can find out PW_i by employing a dictionary attack, in which each guess PW_i' for PW_i can be verified by computing $C_1' = h(PW_i' \oplus N_1) \oplus h(h(PW_i' \oplus N_1) \oplus R)$ and by checking the equality $C_1' \stackrel{?}{=} C_1$

3.2 Impersonating S to U_i on Chen et al.'s Scheme

Unfortunately, Chen et al.'s scheme [5] does not achieves server-to-user authentication, In this subsection, we show this by a server impersonation attack. We present a server impersonation attack where an attacker can easily impersonate the remote server S to the user U_i. Before describing the attack, we note that the secret values stored in the smart card could be extracted by monitoring its power consumption [10,16]. We now proceed to describe the server impersonation attack.

1. As usual, the authentication phase begins when U_i sends the message $\langle ID_i, N_2 \rangle$ to the server S.
2. But, the attacker U_a posing as S intercepts this message and sends immediately back to U_i a forged response message as follows: U_a first generates a random nonce M, computes $A_i{=}M \oplus h(x||ID_i)$ and $B_i{=}h(N_2||M)$ by using the value $h(x||ID_i)$ extracted from U_i's smart card and the nonce N_2 contained in U_i's login message, and then sends $\langle A_i, B_i \rangle$ in response to U_i's request.
3. The forged response $\langle A_i, B_i \rangle$ will pass the verification test by U_i since B_i is equal to $h(N_2||M)$ by computing $M = A_i \oplus h(x||ID_i)$. Hence, U_i, who could not detect the attack, believes U_a as the authentic server and proceeds the scheme specification as usual.

4 The Proposed Scheme

In this section we propose a password-based authentication scheme which enhances on previous scheme, Chen et al. [5]'s scheme. As mentioned earlier, our scheme improves over Chen et al.'s scheme in four ways: (1) it does not require

the server to maintain a password table for verifying the legitimacy of login users, (2) it can withstand the password guessing attack even though the attacker has stolen some user's smart card or gained access to it and extracted the secret values stored in it by monitoring its power consumption, (3) it is provably secure against a powerful active attacker under the computational Diffie-Hellman assumption, and (4) as a result, it achieves both two factor security and mutual authentication.

Before the registration phase is performed for the first time, the server S decides on the following system parameters: a one-way hash function h, a cryptographic key x, a large number p, and a generator g of $\mathbb{Z}_p^*$. The key x is kept secret by the server. In describing the scheme, we will omit 'mod p' from expressions for notational simplicity.

4.1 Description of the Scheme

Registration Phase. The registration of our proposed authentication scheme proceeds as follows:

Step 1. A new user U_i, who wants to register with the server S, chooses its password PW_i, computes $Z_i = h(PW_i)$ and submits a registration request, consisting of its identity ID_i and Z_i, to the remote server S via a secure channel.

Step 2. Upon receiving the request, S first computes

$$K_i = h(ID_i, x) \quad \text{and} \quad N_i = g^{K_i} \oplus h(PW_i)$$

and issues a smart card containing $\langle N_i, h(\cdot), g, p \rangle$ to the user U_i.

Authentication Phase. When U_i wants to log in to the server, it inserts its smart card into a card reader and enters its identity ID_i and password PW_i. With the user input, the scheme enters the authentication phase during which the server and the smart card perform the following steps:

Step 1. Given PW_i, the smart card chooses a random number $a \in_R \mathbb{Z}_p^*$ and computes $A_i = N_i \oplus h(PW_i)$, $C_1 = g^a, C_2 = A_i^a$, and $C_3 = h(ID_i, C_1, C_2)$. The smart card then sends the login request message $\langle ID_i, C_1, C_3 \rangle$ to the server S.

Step 2. After receiving $\langle ID_i, C_1, C_3 \rangle$, S first computes $B_i = C_1^{h(ID_i, x)}$. Then S verifies that: (1) ID_i is valid and (2) C_3 equals $h(ID_i, C_1, B_i)$. If one of both conditions is untrue, S rejects the login request. Otherwise, S chooses a random number $b \in_R \mathbb{Z}_p^*$, computes $D_1 = g^b$, $D_2 = {C_1}^b$, and $D_3 = h(D_1, D_2, B_i)$, and sends the response message $\langle D_1, D_3 \rangle$ to U_i.

Step 3. Upon receipt of the response $\langle D_1, D_3 \rangle$, user U_i computes $E_1 = {D_1}^a$ and checks that D_3 is equal to $h(D_1, E_1, C_2)$. If the verification fails, then U_i aborts the protocol. Otherwise, U_i computes $V_i = h(ID_i, E_1)$ and sends V_i to the server S.

Step 4. When S receives V_i from U_i, it verifies the correctness of V_i by checking that V_i is equal to $h(ID_i, D_2)$. If correct, S accepts the login request; otherwise, rejects it.

4.2 Security Analysis

We now analyze the security of the proposed protocol, considering password guessing attack, impersonation attacks, and replay attack.

Password guessing attack. The vulnerability of Chen et al.'s scheme to the password guessing attack is due to the following fact: to find out the password of the user, they suffice to obtain the information stored in its smart card and read the exchanged message between the server and the remote user. More concretely, the problem with Chen et al.'s scheme is that whoever obtains these values of N_1 and K_i stored in U_i's smart card, the server's response A_i, and the part of the user U_i's login message C_1 can break the user U_i's password PW_i. But, our proposed scheme effectively defeats these kind of attacks mentioned above. Even if the attacker obtains the information (i.e., N_i) stored in the smart card and the exchanged message between the server and the user, he/she can no longer find out the password of the user U_i. In the proposed protocol, the only information related to passwords is $N_i (= g^{K_i} \oplus h(PW_i))$, but because K_i is the secret information that the server only knows, this value does not help the attacker to verify directly the correctness of guessed passwords. Thus, off-line guessing attacks would be unsuccessful against the proposed protocol.

Impersonation attacks. Our scheme can resist two impersonation attacks, a server impersonation attack and a user impersonation attack. Although the attacker obtains the information (i.e., N_i) stored in the smart card, he/she can no longer forge a valid response message $\langle D_1, D_3 \rangle$ or a valid login request message $\langle ID_i, C_1, C_3 \rangle$. Forging a response message is impossible because computing $D_3 = h(D_1, D_2, B_i)$ requires the knowledge of $B_i = C_1^{h(ID_i, x)}$ which in turn needs the server's secret value x. Forging a login request message is also infeasible. This is because no one can compute $C_2 = {A_i}^a$ without knowing $A_i (= N_i \oplus h(PW_i))$ or equivalently knowing $g^{h(ID_i, x)}$. Clearly, computing A_i requires either the server's secret value x or U_i's password PW_i. But since the attacker knows neither x nor PW_i, he/she cannot compute A_i. Therefore, the attacks given in Chen et al.'s scheme will no longer be applied against our revised scheme.

Replay attack. In this attack, an attacker tries to replay messages partly or completely obtained in previous session. If an attacker can impersonate an authentication server or a legitimate user through this replay, then the scheme is said to be vulnerable to a replay attack. Our authentication scheme provides protection against this kind of replay attacks. Firstly, it is impossible for an attacker to impersonate S to U_i. Since U_i chooses a random number a anew for each challenge C_1 and C_2, the server's response for one session cannot be replayed for any other session. Following a similar reasoning as above, an attacker

is also unable to impersonate U_i to S by replaying any of user's response sent for previous sessions. Hence, our proposed scheme is resistant to replay attacks.

5 Comparison

In this section, we introduce the following seven security requirements for evaluating remote user authentication protocols using smart cards.

- R1. The passwords or verification tables are not stored inside the computer.
- R2. The passwords are not transmitted in plain text on network.
- R3. The protocol should provide protection against replay attack.
- R4. No one can impersonate a legitimate user to login the sever.
- R5. No one can impersonate an authentication server.
- R6. The protocol can achieve mutual authentication between an authentication server and a remote user.
- R7. The password cannot be broken by password guessing attack even if a user's smart card is lost.

Table 1. Comparison of security requirements between our protocols and previously published two protocols

	R1.	R2.	R3.	R4.	R5.	R6.	R7.
Lin et al. [14]	N	Y	Y	Y	Y	N	N
Chen et al. [5]	N	Y	Y	Y	N	N	N
Chang and Wu [4]	Y	N	N	N	N	N	N
Our Proposed	Y	Y	Y	Y	Y	Y	Y

Y: Yes, N: No

In the Table 1, we compare the proposed protocol with previously published authentication protocols using the seven security requirements. It is easy to see that our proposed authentication protocol can achieve all of the seven security requirements. However, our proposed scheme increases the computational cost considerably in comparison with previously published three protocols.

References

1. Carlsen, U.: Cryptographic protocol flaws: know your enemy. In: Proceedings of the 7th IEEE Computer Security Foundations Workshop, pp. 192–200 (1994)
2. Chen, C.-M., Ku, W.-C.: Stolen-verifier attack on two new strong-password authentication protocols. IEICE Trans. on Communications E85-B(11) (2002)
3. Chang, C.-C., Wu, T.-C.: Remote password authentication with smart cards. IEE Proceedings E-Computers and Digital Techniques 138(3), 165–168 (1991)
4. Chang, C.-C., Wu, C.-C.: A novel key agreement scheme in a multiple server environment. In: Lee, J.K., Yi, O., Yung, M. (eds.) WISA 2006. LNCS, vol. 4298, pp. 187–197. Springer, Heidelberg (2007)

5. Chen, T.-H., Lee, W.-B., Horng, G.: Ssecure SAS-like password authentication schemes. Computer Standards & Interfaces 27, 25–31 (2004)
6. Chien, H.-Y., Jan, J.-K., Tseng, Y.-M.: An efficient and practical solution to remote authentication: smart card. Computers & Security 21(4), 372–375 (2002)
7. Diffie, W., van Oorschot, P.C., Wiener, M.J.: Authentication and authenticated key exchange. Designs, Codes and Cryptography 2(2), 107–125 (1992)
8. Hsu, C.-L.: Security of Chien et al.'s remote user authentication scheme using smart cards. Computer Standards and Interfaces 26(3), 167–169 (2004)
9. Hwang, M.-S., Li, L.-H.: A new remote user authentication scheme using smart cards. IEEE Trans. on Consumer Electronics 46(1), 28–30 (2000)
10. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999)
11. Ku, W.-C., Chang, S.-T., Chiang, M.-H.: Weaknesses of a remote user authentication scheme using smart cards for multi-server architecture. IEICE Trans. on Commmunications E88-B(8), 3451–3454 (2005)
12. Lin, C.-W., Shen, J.-J., Hwang, M.S.: Security enhancement for optimal strong-password authentication protocol. ACM Operating Systems Rievew 37(2) (2003)
13. Lin, C.L., Sun, H.M., Hwang, T.: Attacks and solutions on strong password authentication. IEICE Trans. on Communications E84-B(9), 2622–2627 (2001)
14. Lin, C.-W., Tsai, C.-S., Hwang, M.S.: A new strong-password authentication scheme using one-way hash functions. Journal of Computer and Systems Scienes International 45(4), 623–626 (2006)
15. Lowe, G.: An attack on the Needham-Schroeder public-key authentication protocol. Information Processing Letters 56(3), 131–133 (1995)
16. Messerges, T.-S., Dabbish, E.-A., Sloan, R.-H.: Examining smart card security under the threat of power analysis attacks. IEEE Trans. on Computers 51(5), 541–552 (2002)
17. Sandirigama, M., Shimizu, A., Noda, M.T.: Simple and secure password authentication protocol (Sas). IEICE Trans. on Communications E83-B(6), 1363–1365 (2000)
18. Sun, H.-M.: An efficient remote user authentication scheme using smart cards. IEEE Trans. on Consumer Electronics 46(4), 958–961 (2000)
19. Tian, X., Zhu, R.W., Wong, D.S.: Improved efficient remote user authentication schemes. International Journal of Network Security 4(2), 149–154 (2007)