Improvement of the ElGamal Based Remote Authentication Scheme Using Smart Cards

Yung-Cheng Lee*[1], Yi-Chih Hsieh[2], Pei-Ju Lee[3] and Peng-Sheng You[4]

[1] Department of Security Technology and
Management, WuFeng University, Chiayi, Taiwan
*yclee@wfu.edu.tw.
[2] Department of Industrial Management
National Formosa University, Yunlin, Taiwan
[3] School of Information Science and Technology
University of Pittsburgh, 135 N Bellefield, Pittsburgh, PA 15260
[4] Graduate Institute of Marketing and Logistics/Transportation
National Chiayi University, Chiayi, Taiwan

ABSTRACT
Nowadays, we can easily obtain variety of services through networks. But due to the open environment, networks are vulnerable to many security threats. The remote user authentication scheme is one of the most widely used mechanisms for servers to authorize users to access the services. In 2009, Ramasamy and Muniyandi proposed a discrete logarithm based remote authentication scheme with smart cards. Their scheme provides mutual authentication and withstands the denial of service attack, forgery attack and parallel session attack. In this article, we show that their scheme is not a practical solution for remote access. It lacks key agreement mechanism and users cannot choose or update passwords freely. Moreover, their scheme cannot resist the stolen-verifier attack, off-line guessing attack, impersonation attack and smart-card-loss-attack. We propose an improved scheme to remedy the drawbacks. The improved scheme has the merits of providing mutual authentication and key agreement, while forward and backward secrecy are ensured as well. The users can choose and update their passwords freely. Furthermore, the scheme can also withstand many attacks such as the smart-card-loss-attack, the replay attack, the off-line guessing attack, the insider attack, the impersonation attack and the parallel session attack.

Keywords: Remote authentication, smart cards, discrete logarithm problem.

1. Introduction

Network is one of the best platforms for people to obtain variety of online services. However, due to the open environment, many network systems are vulnerable to lots of attack. The remote user authentication scheme is one of the most convenient and widely used authentication mechanisms for servers to identify users over insecure communication channels [1-8]. In the remote user authentication scheme, the remote users access to the system for services after they are authenticated. A remote user authentication scheme comprises users, servers and insecure communication channels. Thus the security and vulnerabilities of the remote user authentication scheme depend on these three components [9]. An adversary or even the legitimate user can launch variety of attacks such as the guessing attack, replay attack, impersonation attack, insider attack, etc. on the system. An attacker can intercept the transmitted information in secure or insecure channel and then use the intercepted information to attack the system.

Since smart card is one of the most reliable and efficient tools for authentication, there are many remote user authentication schemes identify the users by using smart cards [1, 3, 6, 7, 9-13]. Lamport [14] proposed the first well-known remote password authentication scheme in 1981. But the scheme has the drawbacks of high hash overhead and the server does not need to store a password verification table. Chien et al. [10] proposed a user authentication scheme in 2002. But Ku and Chen [12] showed that the scheme is vulnerable to the reflection attack and the insider attack. Ku and Chen proposed an enhancement to solve the problems. Their scheme has the merits of mutual authentication, no verification table and resisting the parallel session attack. Later, Yoon et al. [7] indicated that Ku and Chen's improved scheme was also susceptible to the parallel session attack, and then they proposed an improved scheme to fix the flaws. However, in 2009, Hsiang and Shin [11] showed that Yoon et al.’s improved scheme is vulnerable to the parallel session attack, masquerading attack and password guessing attack. They also proposed an improved scheme to remedy the drawbacks.

Hwang and Li [15], based on the ElGamal’s [16] public key scheme, proposed a remote user authentication scheme with smart cards in 2000. The scheme withstands the replay attack and the server does not need to maintain a password table for verifying the legitimacy of the login users. The security of the scheme relies on the difficulty of solving discrete logarithms problem. But Chan and Cheng [17] indicated that the scheme is insecure since a legal user can generate a valid pair of identity and password without knowing the server's secret key. Shen et al. [18] also shown that the Hwang and Li's scheme is vulnerable to the masquerade attack, and then proposed a modified scheme to fix the drawback. Shen et al.'s scheme prevents the masquerade attack by hiding user's identity. However, Leung et al. [19] shown that Shen et al.'s scheme also cannot resist the impersonation attack. Awasthi and Lal [20] proposed a modified scheme to enhance the security of Hwang-Li’s scheme. Awasthi and Lal declared that any attacker cannot obtain the passwords corresponding to the previously registered identities with the revealed secret key. But Kumar [9] indicated that Awasthi and Lal’s scheme does not provide forward security to the authentication server.

Yoon et al. [21] based on generalized ElGamal signature scheme proposed a smart card remote user authentication scheme. The scheme provides mutual authentication and session key agreement; the user can choose and update his password freely without the assistance of the server. However, Tian et al. [22] indicated that Yoon et al.’s scheme does not provide two-factor security. Two-factor security means a system can guarantee the security of the scheme when either the user’s smart card or his password is stolen, but not both. Tian et al. shown that if an attacker obtains the secret in the smart card, he can forge user's valid login request without knowing the user’s password. Tian et al. also propose two more efficient smart card remote user authentication schemes to remedy the flaws.

Ramasamy and Muniyandi [13], based on ElGamal cryptosystem, proposed a remote authentication scheme with smart card in 2009. Their scheme provides mutual authentication and withstands the denial of service attack, forgery attack and parallel session attack. In this paper, we will show that their scheme has the following weaknesses: (1) It cannot withstand the stolen-verifier attack; (2) It lacks key agreement mechanism; (3) Users cannot choose and update their passwords freely; (4) The scheme is not a practical solution for remote authentication due to the long bit-length password which is difficult to memorize; (5) It is vulnerable to the off-line guessing attack if password is small. (6) It cannot resist the impersonation attack. (7) It cannot resist the smart-card-loss-attack.

We propose an improved scheme to remedy the drawbacks. The improved scheme has the merits of providing mutual authentication and ensures forward and backward secrecy. It can also withstand variety of attacks such as the smart-card-loss-attack, the replay attack, the off-line guessing attack, the insider attack, the impersonation attack and the parallel session attack.

The rest of the paper is organized as follows. All notations used throughout this article are described in Section 2. Ramasamy and Muniyandi's scheme and its weakness are briefly described in Section 3. Our improved scheme and its security analysis along with performance discussion are presented in Section 4. Finally, we make conclusions in Section 5.

2. Preliminaries and notations

Generally, a smart card based remote user authentication scheme comprises the following five phases: (1) the registration phase, (2) the login phase, (3) the authentication phase, (4) the key agreement phase and (5) the password changing phase. In the registration phase, the user sends a registration request along with related information to the server via a secure channel. The server identifies the user and generates related messages to store in the smart card, and then delivers the card to the user. In the login phase, the user attaches his smart card into a card reader and keys in password to login to the system. In the authentication phase, the server checks the validity of the login request. If the user is authenticated, the server allows the user to access the system online. In the key agreement phase, the user and the server cooperatively negotiate a common session key for future secure communication. In the password changing phase, remote users freely update their passwords. The mutual authentication refers to users authenticating themselves to a server and that server authenticating itself to the user in such a way that both parties are assured of the others' identity. Many remote authentication schemes provide mutual authentication feature to enhance the security.

If the user wants to join the authentication system, he should register himself to the server. When a user wants to obtain services, he has to login to the system. The server authorizes the user to access the system after he is authenticated. All notations used in this paper are listed in Table 1.

Notations	Description
U_i	A legitimate user.
ID_i	The identity of the user U_i .
PW_i	The password of the user U_i .
AS	The authentication server.
x_S	The server's secret key.
r	A random number.
p	*A* large prime number.
SK	The common session key for the server and user.
$h(\cdot)$	A secure one-way hash function.
T	A timestamp.
$\oplus$	Bitwise exclusive-or (XOR) operation.
$\Vert$	The concatenation operation.
$A \Rightarrow B : \{M\}$	The entity *A* sends message *M* to the receiver *B* via a secure channel.
$A \rightarrow B : \{M\}$	The entity *A* sends message *M* to the receiver *B* through a public channel.

Table 1. The notations.

3. Ramasamy and Muniyandi’s remote mutual authentication scheme using smart cards

Ramasamy and Muniyandi proposed an ElGamal based remote authentication scheme with smart card in 2009 [13]. The security of the scheme depends on solving the discrete-logarithm problem. The scheme comprises the registration phase, the login phase and the authentication phase. The scheme and its weaknesses are described in Section 3.1 and 3.2, respectively.

3.1 Ramasamy and Muniyandi's scheme

(1) The registration phase

If a user U_i wants to join the system, the steps of the registration phase are as follows.

Step R-1. $U_i \Rightarrow AS : \{ID_i, \text{Registration request}\}$

The user submits ID_i along with the registration request to the server in person or through a secure channel.

Step R-2. $AS \Rightarrow U_i : \{PW_i, \text{Smart card}\}$

Upon receiving ID_i and the registration request, the authentication server AS computes the password PW_i for U_i by:

$$PW_i = (ID_i \oplus T_R)^{x_S} \bmod p \qquad (1)$$

Where T_R is the registration timestamp and x_S is the server's secret key. The AS installs $\{PW_i, h(\cdot), p\}$ into a smart card and sends it along with the password PW_i to the user through a secure channel.

(2) The login phase

If a user wants to access the system for services, he should login to the system. The steps of the login phase are as follows.

Step L-1. The user keys in ID_i and PW_i .

The user U_i attaches the smart card to a card reader and keys in his ID_i and PW_i.

Step L-2. $U_i \rightarrow AS: \{ID_i, C_1, C_2, T\}$.

The smart card generates a random number *r* and computes $\{C_1, t_1, M, C_2\}$ by:

$$C_1 = PW_i^r \bmod p \tag{2}$$

$$t_1 = h(T \oplus PW_i) \bmod (p-1) \tag{3}$$

$$M = PW_i^{t_1} \bmod p \tag{4}$$

$$C_2 = M \times (C_1^{t_1}) \bmod p \tag{5}$$

Where *T* is the current timestamp. Next, the user sends $\{ID_i, C_1, C_2, T\}$ to the server.

(3) The authentication phase

In the authentication phase, the server and the user check the received information to authenticate each other. The steps of the authentication phase are as follows.

Step A-1. The server checks ID_i and *T*.

On receiving $\{ID_i, C_1, C_2, T\}$, the server checks the identity ID_i and the timestamp *T*. The steps continued if ID_i is existed in the verification table and *T* is in a valid time interval; otherwise, the login steps are terminated.

Step A-2. The server authenticates the user.

If ID_i and *T* are valid, the server computes $PW_i = (ID_i \oplus T_R)^{x_S}$ and $t_1 = h(T \oplus PW_i) \bmod (p-1)$. Next, the server checks whether the following equation holds.

$$C_2(C_1^{t_1})^{-1} = (PW_i)^{t_1} \bmod p \tag{6}$$

The user is authenticated if above equation is satisfied.

Step A-3. $AS \rightarrow U_i: \{C_3, T_S\}$.

After the user is authenticated, the authentication server computes t_2 and C_3 by:

$$t_2 = h(T_S \oplus PW_i) \bmod (p-1) \tag{7}$$

$$C_3 = C_1^{t_2} \bmod p \tag{8}$$

Where T_S is the current timestamp. Then, the server sends $\{C_3, T_S\}$ to the user.

Step A-4. The user authenticates the server.

After receiving $\{C_3, T_S\}$, the user checks the freshness of the timestamp T_S. If T_S is in a valid time interval, the user computes $t_2' = h(T_S \oplus PW_i) \bmod (p-1)$ and $C_3' = C_1^{t_2'}$. Then the user checks whether C_3' is equal to C_3. The server is authenticated by the user if $C_3 = C_3'$.

3.2 The weaknesses of Ramasamy and Muniyandi's scheme

Though Ramasamy and Muniyandi's scheme provides mutual authentication and withstands the denial of service attack, forgery attack and parallel session attack. However, their scheme contains weaknesses as follows.

(1) The users cannot update their password freely

In general, for the security of a remote password authentication scheme, it is necessary to update password frequently. However, in Ramasamy and Muniyandi's scheme, due to the password is computed by the server with $PW_i = (ID_i \oplus T_R)^{x_S}$, the user should repeat the registration steps if he wants to update his password. That is, the user should register himself again to the authentication server through secure channel. This makes the scheme is inconvenient for users on updating passwords. Moreover, due to the password is computed only by the server, it will cause lots of disputes if the password is disclosed.

(2) It is not a practical solution for authentication if the bit-length of password is quite long

In Ramasamy and Muniyandi's scheme, it is a discrete logarithm problem to find x_S because the password is computed by the authentication server with $PW_i = (ID_i \oplus T_R)^{x_S} \pmod p$, where p is a large prime. The large prime will cause long bit-length password. However, the long bit-length of password is difficult to be memorized. This makes it is not easy for users to key in password for login. Thus the scheme is not a practical solution for authentication.

(3) It is vulnerable to the off-line guessing attack if the password is small

On the contrary, for the ease of memorizing, we always choose a small password. However, the scheme is vulnerable to the off-line guessing attack if the password is small. The steps of the off-line guessing attack are as follows.

Step G-1. The adversary intercepts $\{ID_i, C_1, C_2, T\}$ from Step L-2.

Step G-2. The adversary guesses the password.

In this step, the adversary tries to guess the password. Suppose he tries to guess PW_i' is the correct password. Then the adversary computes $D = C_2(C_1^{h(T \oplus PW_i')})^{-1}$ and $D' = (PW_i')^{h(T \oplus PW_i')}$ with the intercepted information $\{ID_i, C_1, C_2, T\}$ and PW_i'. Then the adversary checks whether the equation $D = D'$ holds; the guessing steps are repeated if it isn't. The correct password is obtained if $D = D'$. The password will be found easily if its bit-length is not long enough. So Ramasamy and Muniyandi's scheme is vulnerable to the off-line guessing attack if the password is small.

Similarly, due to $C_3 = C_1^{t_2} = C_1^{h(T_S \oplus PW_i)}$, the adversary also can use the information $\{C_3, T_S\}$ in Step A-3 to guess the password.

(4) It cannot resist the impersonation attack

With the off-line guessing attack described above, an adversary will obtain the password PW_i. With PW_i, the adversary can compute $\{ID_i, C_1, C_2, T\}$ by Eq.[2-5]. With $\{ID_i, C_1, C_2, T\}$, an adversary can impersonate a legitimate user to login to the system. Similarly, an adversary can also compute correct information $\{C_3, T_S\}$ in Step A-3 for authentication. Thus Ramasamy and Muniyandi's scheme is vulnerable to the impersonation attack.

(5) It lacks key agreement mechanism

The key agreement mechanism provides users and server to negotiate a common session key; thereafter users and server can communicate securely with the session key. Generally, a remote authentication scheme should provide user and server a key agreement protocol to establish a common session key after authentication is obtained. However, Ramasamy and Muniyandi's scheme lacks key agreement mechanism which makes it is inconvenient to use.

(6) It cannot resist the stolen-verifier attack

In Ramasamy and Muniyandi's scheme, the server stores $\{x_S, ID_i, T_R\}$ in the verification table. The server should properly protect the verification table to avoid secret information being disclosed. However, the existence of the verification table will cause the adversary intends to steal the secret information for benefits. If an adversary obtains $\{x_S, ID_i, T_R\}$, he can impersonate as the legitimate user to login the system. Thus Ramasamy and Muniyandi's scheme is also vulnerable to the stolen-verifier attack.

(7) It cannot resist the smart-card-loss-attack

Smart-card-loss-attack means an attacker can launch various attacks such as the off-line guessing attack if he/she obtains a legitimate user's smart card [23]. In the login phase of Ramasamy and Muniyandi's scheme, the user keys in ID_i and PW_i. However, ID_i and PW_i are not verified by the smart card. Thus an adversary can try to guess the password arbitrarily if he obtains the card. Since the bit-length of the password is always quite short, the password will be guessed quickly. Thus the scheme cannot resist the smart-card-loss-attack.

4. Our improved scheme

In this section, we propose an improved scheme to enhance the security. The scheme comprises the registration phase, the login phase, the authentication phase, the key agreement phase and the password updating phase. The improved scheme is described in Section 4.1. Following, the security analysis and the performance comparison are discussed in Section 4.2 and 4.3, respectively.

4.1 The improved scheme

(1) The registration phase

If a user wants to join the system, he should register himself to the authentication server. The steps of the registration phase are as follows.

Step IR-1. $U_i \Rightarrow AS : \{ID_i, \mathrm{P}W_i\}$.

The user U_i first chooses his identity ID_i and password $\mathrm{P}W_i$, and then submits $\{ID_i, \mathrm{P}W_i\}$ to the authentication server *AS* in person or through a secure channel.

Step IR-2. $AS \Rightarrow U_i$: Smart card.

Upon receiving $\{ID_i, \mathrm{P}W_i\}$, *AS* computes *S* and *R* by:

$$S = h(ID_i \oplus h(x_S)) \tag{9}$$

$$R = S \oplus PW_i \tag{10}$$

AS stores $\{S, R, h(\cdot)\}$ in a smart card and sends it to the user. Note that the server does not need to store user's password.

(2) The login phase

If U_i wants to access the system for services, the steps of the login phase are as follows.

Step IL-1. The user keys in ID_i and PW_i.

The user U_i attaches the smart card to a card reader and keys in ID_i and PW_i.

Step IL-2. The smart card checks PW_i.

The smart card computes $S' = R \oplus PW_i$ and checks whether *S'* is equal to *S*. The steps are terminated if $S' \neq S$.

Step IL-3. $U_i \rightarrow AS : \{ID_i, C_1, C_2, T_U\}$.

If $S' = S$ the smart card generates a random number *r* and computes $\{C_1, A, M, C_2\}$ by:

$$C_1 = S^r \bmod p \tag{11}$$

$$A = h(T_U \oplus S) \bmod (p - 1) \tag{12}$$

$$M = S^A \bmod p \tag{13}$$

$$C_2 = M\,(C_1^{\,A}) \bmod p \tag{14}$$

Where T_U is the current timestamp. Next, the user sends $\{ID_i, C_1, C_2, T_U\}$ along with the login request to the server.

(3) The authentication phase

On receiving the login request and $\{ID_i, C_1, C_2, T_U\}$, the server checks the received information to authenticate the user by the following steps.

Step IA-1. The server checks ID_i and T_U.

The server checks the identity ID_i and the timestamp T_U. The authentication steps are stopped if ID_i does not exist in the verification table or T_U is not fresh.

Step IA-2. The server authenticates the user.

If ID_i and T_U are valid, the server computes $S' = h(ID_i \oplus h(x_S))$ and $A' = h(T_U \oplus S') \bmod (p - 1)$. Then the server verifies whether $C_2(C_1^{\,A'})^{-1}$ and $S^{A'}$ are equal or not. If $C_2(C_1^{\,A'})^{-1} = S^{A'}$, the server will authenticate the user.

Step IA-3. $AS \rightarrow U_i : \{C_3, T_S\}$.

After the user is authenticated, the server computes B and C_3 by

$$B = h(T_S \oplus S) \bmod (p - 1) \tag{15}$$

$$C_3 = C_1^B \bmod p \tag{16}$$

Where T_S is the current timestamp. Next, the authentication server sends $\{C_3, T_S\}$ to the user.

Step IA-4. The user authenticates the server.

After receiving $\{C_3, T_S\}$, the user checks the freshness of the timestamp T_S. If it is fresh then the smart card computes B' and C_3' by.

$$B' = h(T_S \oplus S) \bmod (p - 1) \tag{17}$$

$$C_3' = C_1^{B'} \bmod p \tag{18}$$

Finally, the smart card compares C_3' with C_3. The mutual authentication is obtained if $C_3' = C_3$.

(4) The key agreement phase

The server and the user will obtain a common session key *SK* if the mutual authentication is obtained. The common session key is negotiated by the server and the user with the following equation respectively.

$$SK = h(h(S) \oplus T_S) \tag{19}$$

From now on, the server and the user can communicate securely by using the common session key *SK*.

(5) The password updating phase

The scheme provides users to choose and update passwords freely. Since the updating steps are only performed in the smart card, it is very convenient for users to change their password. The steps of the password updating phase are as follows.

Step IU-1. The user attaches his smart card to a card reader and keys in his identity ID_i and password PW_i.

Step IU-2. The smart card checks PW_i.

The smart card computes $S' = R \oplus PW_i$ and checks whether S' is equal to S or not. The password updating request is rejected if $S' \neq S$.

Step IU-3. The smart card replaces R with R_{new}.

If $S' = S$, the user selects and keys in a new password PW_{i_new}. The smart card computes R_{new} by

$$R_{new} = R \oplus PW_i \oplus PW_{i\_new} \tag{20}$$

Next, the smart card replaces R with R_{new}. From now on, the user can use the new password to login to the system.

4.2 Discussion and security analysis

The improved scheme has the merits of providing mutual authentication and ensuring forward and backward secrecy. The users can choose and update passwords freely. The information stored in the authentication server database only includes the secret x_S and user's identity ID_i, where x_S can be protected with cryptographic mechanisms and ID_i is public, thus the stolen-verifier attack can be avoided.

Furthermore, the scheme can withstand lots of attacks such as the smart-card-loss-attack, the replay attack, the off-line guessing attack, the insider attack, the impersonation attack and the parallel session attack. The merits and security features are described as follows.

(1) It withstands the smart-card-loss-attack

Through the smart-card-loss-attack, if an unauthorized person obtains the legitimate user's smart card, he can guess the password and impersonate the legitimate user to login to the system. The smart-card-loss-attack is caused by the smart card which always generates and outputs fixed information for the same input. The smart-card-loss-attack is always a huge threat for all cardholders. However, most of the smart card based schemes suffer from this attack.

In the login phase of the proposed scheme, the user sends $\{ID_i, C_1, C_2, T_U\}$ to the authentication server, where $C_1 = PW_i^r$ and $C_2 = M(C_1^A) \bmod p$. Due to the ever changing random number *r*, the transmitted information C_1 and C_2 are always updated on each login session. If an adversary obtains the smart card and tries to guess the password, the trial will fail due to the ever changed information makes the adversary cannot verify his guess. So the smart-card-loss-attack can be avoided even if the adversary obtains the smart card.

(2) It withstands the replay attack

In the login phase, the user sends $\{ID_i, C_1, C_2, T_U\}$ to the server for authentication. Since the information contains the timestamp T_U, the replay message will be detected if the adversary retransmits the intercepted message. Similarly, if an adversary replays the message $\{C_3, T_S\}$ to the user in Step IA-3, the user can also detect the attack by checking the freshness of the timestamp T_S. So the proposed scheme resists the replay attack.

(3) The proposed scheme provides forward and backward secrecy

Though the session key is important secret information shared between the user and the server, but the session key may be disclosed in some scenario. Thus it is important to develop a system that a compromise of the current key should not compromise any future key or earlier key. In the proposed scheme, the common session key is computed by $SK = h(h(S) \oplus T_S)$. Due to the ony way hash function, the hashed secret $h(S)$ cannot be disclosed even if the SK is compromised. Without knowing $h(S)$, the future or the earlier session keys cannot be obtained. So it is infeasible to obtain any future key or earlier key even if the current session key is compromised.

(4) The proposed scheme provides mutual authentication

In a client-server system, it is important for a user authenticating himself to a server, and vice-versa. After mutual authentication, both parties are assured of the others' identity.

In Step IA-2, upon receiving $\{ID_i, C_1, C_2, T_U\}$, the server authenticates the user by checking whether $C_2(C_1^{A'})^{-1} = S^{A'}$ holds. Similarly, in Step IA-4, the user authenticates the server if $C_3' = C_3$ is verified. Thus, the proposed scheme provides mutual authentication that will enhance the security.

(5) It resists the off-line guessing attack

Suppose that an adversary wants to guess the password off-line by using the intercepted information $\{ID_i, C_1, C_2, T_U\}$. Due to $C_1 = PW_i^r$, $A = h(T_U \oplus S)$, $M = S^A$ and $C_2 = M(C_1^A)$, it is infeasible to find the password PW_i such that $C_2(C_1^A)^{-1} = S^A$ since it is a discrete logarithm problem. Thus, the improved scheme can resist the off-line guessing attack.

(6) It resists the insider attack

A malicious insider is a legitimate user whose actions are in contradiction to the system policy, or is a masquerader who owns a legitimate user's identity and impersonates another user for malicious purposes. In the improved scheme, the secret parameters *S* and *R* are stored in a tamper-free smart card; no malicious insider can obtain the secrets even if he owns the card. Without knowing *S* and *R*, a malicious insider cannot compute correct C_1 and C_2 to impersonate a legitimate user to login the system. Suppose that the adversary owns a legitimate user's smart card. The probability for the malicious insider to correctly guess the password is $1/2^{|PW|}$ which is very small, where $|PW|$ is the bit-length of the password. The system also can prevent the attack by limiting the error trial times. That is a malicious insider cannot fabricate false verification information to login to the system. Therefore, the scheme can resist the insider attack.

(7) It resists the impersonation attack

If an adversary wants to impersonate a legitimate user to login to the system, since the secret parameters S and R are unknown, he cannot compute correct C_1 and C_2 to successfully login to the system. Similarly, the adversary cannot obtain B since S is unknown and $B = h(T_S \oplus S)$; therefore the adversary cannot compute correct C_3 with $C_3 = C_1^B$ in Step IA-3 for authentication. So the improved scheme resists the impersonation attack.

(8) It withstands the parallel session attack

In the parallel session attack, an attacker can masquerade as the legitimate user by sending eavesdropped communication information between the user and the server. In the improved scheme, if an adversary intercepts the login or authentication information $\{ID_i, C_1, C_2, T_U\}$ or $\{C_3, T_S\}$ in previous session and resends it to the other party, the attack will be blocked since the verification steps in authentication phase will fail.

4.3 Performance comparison

The performance comparison of several ElGamal signature based password authentication schemes and our scheme are listed in Table 2. In the table, H means hash function and E stands for exponentiation operation.

	Hwang-Li [14]	Awasthi-Lal [19]	Kumar [8]	Ramasamy-Muniyandi [12]	Our scheme
Registration phase	1E	1H 1E	2E	1E	1H
Login phase	1H 3E	2H 3E	1H 3E	1H 2E	1H 3E
Authentica-tion phase	1H 3E	1H 3E	1H 2E	2H 3E	1H 2E
Mutual authentica-tion	No	No	No	No	Yes

Table 2. Comparison of several ElGamal signature based schemes and our scheme.

Note that, in our improved scheme, the computation complexity does not include the authentication steps for the user to validate the server. It is shown that our scheme uses less computational resources than others, thus our scheme is more suitable to be used in a smart card based scenario. In practice, the smart card only needs to store S, R and a one way hash function $h(\cdot)$; so our scheme does not need large storage space in the smart card. Furthermore, our scheme also provides mutual authentication which is important on ensuring the security of remote authentication schemes.

5. Conclusions

In this paper, we show that Ramasamy and Muniyandi's remote authentication scheme lacks key agreement mechanism and users cannot update their password freely. Moreover, their scheme cannot withstand the stolen-verifier attack, the off-line guessing attack and the smart-card-loss-attack; and it is not a practical solution for remote accessing. We propose an improved scheme to fix the flaws. The improved scheme has the following merits:

(1) The scheme provides mutual authentication and key agreement.

(2) The scheme ensures forward and backward secrecy.

(3) The users can choose and update their passwords freely.

(4) The improved scheme withstands lots of attacks such as the smart-card-loss-attack, the replay attack, the off-line guessing attack, the insider attack, the impersonation attack and the parallel session attack.

Acknowledgments

This work was partially supported by the National Science Council of the Republic of China under the contract number NSC 101-2632-E-274-001-MY3.

References

[1] C. C. Chang and T. C. Wu, “Remote password authentication with smart cards,” IEE Proc E- Comput Digit Tech, vol. 138, no. 3, pp. 165-168, 1991.

[2] B. T. Hsieh et al., “On the security of some password authentication protocols,” Informatica, vol. 14, no. 2, pp. 195-204, 2003.

[3] M. S. Hwang and L. H. Li, “A new remote user authentication scheme using smart cards,” IEEE T Consum Electr, vol. 1, no. 46, pp. 28-30, 2000.

[4] C. W. Lin et al., “A new strong password authentication scheme using one-way hash functions,” J Comput Sys Sc Int, vol. 45, no. 4, pp.623-626, 2006.

[5] X. Tian et al., “Improved efficient remote user authentication schemes,” Int J Net Sec, vol. 4, no. 2, pp. 149-154, 2007.

[6] W. H. Yang and S. P. Shieh, “Password authentication schemes with smart card,” COMPSEC, vol. 8, no. 18, pp. 727-733, 1999.

[7] E. J. Yoon et al., “Further improvement of an efficient password based remote user authentication scheme using smart cards,” IEEE T Consum Electr, vol. 50, no. 2, pp. 612-614, 2004.

[8] R. Martínez-Peláez et al., “Security improvement of two dynamic ID-based authentication schemes by Sood-Sarje-Singh,” J Appl Res Technol, vol. 11, no. 5, pp. 755-763, Oct. 2013.

[9] M. Kumar, “Some remarks on a remote user authentication scheme using smart cards with forward secrecy," IEEE T Consum Electr, vol. 50, no. 2, pp. 615-618, 2004.

[10] H. Y. Chien et al., “An efficient and practical solution to remote authentication: smart card,” COMPSEC, vol. 4, no. 21, pp. 372-375, 2002.

[11] H. C. Hsiang and W. K. Shih, “Weaknesses and improvements of the Yoon-Ryu-Yoo remote user authentication scheme using smart cards,” Comput Commun, vol. 32, no. 4, pp. 649-652, 2009.

[12] W. C. Ku and S. M. Chen, “Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards,” IEEE T Consum Electr, vol. 50, no. 1, pp. 204-207, 2004.

[13] R. Ramasamy, A. P. Muniyandi, “New remote mutual authentication scheme using smart cards,” T data privacy, vol. 2, pp. 141-152, 2009.

[14] L. Lamport, “Password authentication with insecure communication,” Commun ACM, vol. 24, no. 11, pp. 770-772, 1981.

[15] M. S. Hwang and L. H. Li, “A new remote user authentication scheme using smart cards,” IEEE T Consum Electr, vol. 46, no. 1, pp. 28-30, 2000.

[16] T. ElGamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” IEEE T Inform Theory, vol. 31, no. 4, pp. 469-472, 1985.

[17] C. K. Chan and L. M. Cheng, “Cryptanalysis of a remote user authentication scheme using smart cards,” IEEE T Consum Electr, vol. 46, pp. 992-993, 2000.

[18] J. J. Shen et al., “A modified remote user authentication scheme using smart cards,” IEEE T Consum Electr, vol. 49, no. 2, pp. 414-416, 2003.

[19] K. C. Leung et al., “Cryptanalysis of a remote user authentication scheme using smart cards”, IEEE T Consum Electr, vol. 49, no. 4, pp. 1243-1245, 2003.

[20] A. K. Awasthi and S. Lal, “A remote user authentication scheme using smarts cards with forward secrecy,” IEEE T Consum Electr, vol. 49, no. 4, pp. 1246-1248, 2003.

[21] E. J. Yoon et al., “Efficient remote user authentication scheme based on generalized ElGamal signature scheme,” IEEE T Consum Electr, vol. 50, no. 2, pp. 568-570, 2004.

[22] X. Tian et al., “Improved efficient remote user authentication schemes,” Int J Net Sec, vol.4, no.2, pp.149-154, 2007.

[23] Y. C. Lee, “Smart - card - loss - attack and improvement of Hsiang et al.’s authentication scheme,” J Appl Res Technol, vol. 11, no. 4, pp. 597-603, Aug. 2013.