Turk J Elec Eng & Comp Sci, Vol.19, No.3, 2011, © TÜBİTAK
doi:10.3906/elk-1004-16

Remote mutual authentication and key agreement scheme based on elliptic curve cryptosystem

Eun-Jun YOON
School of Electrical Engineering and Computer Science, Kyungpook National University,
1370 Sankyuk-Dong, Buk-Gu, Daegu 702-701, SOUTH KOREA
e-mail: ejyoon@knu.ac.kr

Received: 07.04.2010

Abstract

Remote mutual authentication is an important part of security, along with confidentiality and integrity, for systems that allow remote access over untrustworthy networks, like the Internet. In 2006, Shieh-Wang pointed out the weakness of Juang's remote mutual authentication scheme using smart card and further proposed a novel remote user authentication scheme using smart card. However, this paper demonstrates that Shieh-Wang's scheme still does not provide perfect forward secrecy and is vulnerable to a privileged insider's attack. We also present an improved scheme based on the Elliptic Curve Diffie-Hellman problem (ECDHP) and secure one-way hash function, in order to isolate such security problems.

Key Words: *Authentication, password, key agreement, cryptanalysis, smart card, elliptic curve cryptosystem*

1. Introduction

Remote mutual authentication is a mechanism for two communicating parties to mutually authenticate each other through an insecure communication channel. In addition, a smart card based remote mutual authentication scheme is very practical to authenticate remote users [1, 2]. Since Lamport [3] proposed a remote authentication scheme in 1981, many researchers have proposed new schemes to improve the efficiency and security [4, 5, 6, 7, 8, 9, 10, 11, 12, 13].

In 2000, Sun [4] proposed a cost effective unilateral remote authentication scheme in which only a server can authenticate a user's legitimacy. In 2002, Chien-Jan-Tseng [5] proposed an efficient remote mutual authentication scheme using smart card allowing server and user to authenticate each other. The advantages in the scheme include freely chosen passwords, no verification tables, low communication and computation costs. However, as demonstrated by Hsu [6], Chien-Jan-Tseng's scheme is vulnerable to the parallel session attack. Thereafter, in 2004, Juang [7] proposed another improved scheme preserving all the advantages of Chien-Jan-Tseng's scheme. Unlike Chien-Jan-Tseng's scheme, Juang's scheme is a nonce based authentication and key agreement scheme. Therefore, no synchronized clocks are required in the scheme. In addition, Juang's scheme generates a session key for the user and server in their subsequent communication.

Recently, Shieh-Wang [8], however, pointed out another weakness of Juang's scheme and then proposed an improvement of the scheme to improve the weakness. Shieh-Wang claimed that their scheme not only preserves all the advantages of Juang's scheme but also improves its efficiency.

Nevertheless, this paper demonstrates that Shieh-Wang's scheme still does not provide perfect forward secrecy [14] and is vulnerable to a privileged insider's attack [15, 16]. We also present an improved scheme based on Elliptic Curve Diffie-Hellman problem (ECDHP) and secure one-way hash function, in order to isolate such security problems. The Elliptic Curve cryptosystems [17, 18], which are based on the Elliptic Curve Discrete Logarithm Problem (ECDLP) over a finite field, have some advantages over other cryptosystems: The key size can be much smaller than those of the other cryptosystems since only exponential-time attacks have been known to occur so far, if the curve is carefully chosen [19], and that the Elliptic Curve Discrete Logarithms might still be intractable even if factoring and the multiplicative group discrete logarithm turn out to be tractable problems. As a result, the improved scheme has the following merits: (1) The scheme provides not only perfect forward secrecy but also explicit mutual authentication between the user and a remote server. (2) The scheme does not require time synchronization or a delay-time limitations by using timestamp between the user and the remote system. (3) In order to prevent the problems of clock synchronization or a delay-time limitations, the proposed scheme adopts a nonce-based scheme [20] instead of a timestamp-based scheme. (4) The security of the proposed scheme is based on Elliptic Curve Diffie-Hellman problem (ECDHP) [21] and one-way hash function to suitable for light-weight authentication and key agreement. (5) The scheme resists the privileged insider's attack. (6) The scheme provides secure password change scheme without helping of the remote server.

The remainder of this paper is organized as follows¿ Section 2 briefly reviews Shieh-Wang's scheme. Section 3 demonstrates the security weaknesses of Shieh-Wang's scheme. The proposed authentication scheme is presented in Section 4, while Sections 5 and 6 discusses the security and performance of the proposed scheme, respectively. The conclusion is given in Section 7.

2. Review of Shieh-Wang's scheme

This section briefly reviews Shieh-Wang's a remote mutual authentication and key agreement scheme using smart card with secure one-way hash function [8]. Some of the notations used in this paper are defined as follows.

- U_i: user i
- ID_i: identity of U_i
- PW_i: password of U_i
- x: the secret key maintained by the server
- $h(\cdot)$: a secure one-way hash function
- $\oplus$: exclusive-or operation
- $||$: string concatenation operation
- q: the order of the underlying finite field F_q

- E: a suitably chosen Elliptic Curve defined over F_q
- P: a base point in the generator point E
- n: the prime order of P
- O: the point at infinity, where $nP = O$ and $P \neq O$.

Figure 1 shows Shieh-Wang's scheme and the scheme consists of two phases: the registration, and the login and key agreement.

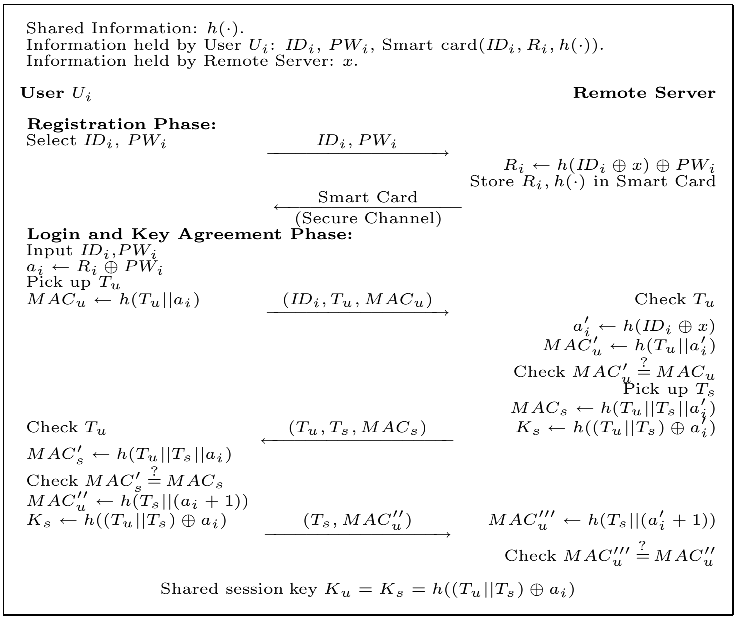

Figure 1. Shieh-Wang's remote mutual authentication and key agreement scheme.

2.1. Registration phase

Assume a user U_i submits his/her identity ID_i and password PW_i to the server over a secure channel for registration. If the request is accepted, the server computes $R_i = h(ID_i \oplus x) \oplus PW_i$ and issues U_i a smart card containing R_i and $h(\cdot)$.

2.2. Login and key agreement phase

When the user U_i wants to login to the server, he/she first inserts his/her smart card into a card reader then inputs his/her identity ID_i and password PW_i. The smart card then performs the following steps to begin an access session:

1. Compute $a_i = R_i \oplus PW_i$.
2. Acquire current time stamp T_u, store T_u temporarily until the end of the session, and compute $MAC_u = h(T_u||a_i)$.

3. Send the message (ID_i, T_u, MAC_u) to the server and wait for response from the server. If no response is received in time or the response is incorrect, report login failure to the user and stop the session.

After receiving the message (ID_i, T_u, MAC_u) from U_i, the server performs the following steps to assure the integrity of the message, respond to U_i, and challenge U_i to avoid replay:

1. Check the freshness of T_u. If T_u has already appeared in a current executing session of user U_i, reject U_i's login request and stop the session. Otherwise, T_u is fresh.

2. Compute $a'_i = h(ID_i \oplus x)$, $MAC'_u = h(T_u || a'_i)$, and check whether MAC'_u is equal to the received MAC_u. If it is not, reject U_i's login and stop the session.

3. Acquire the current time stamp T_s. Store temporarily paired time stamps (T_u, T_s) and ID_i for freshness checking until the end of the session. Compute $MAC_s = h(T_u || T_s || a'_i)$ and session key $K_s = h((T_u || T_s) \oplus a'_i)$. Then, send the message (T_u, T_s, MAC_s) back to U_i and wait for response from U_i. If no response is received in time or the response is incorrect, reject U_i's login and stop the session.

On receiving the message (T_u, T_s, MAC_s) from the server, the smart card performs the following steps to authenticate the server, achieve session key agreement, and respond to the server:

1. Check if the received T_u is equal to the stored T_u to assure the freshness of the received message. If it is not, report login failure to the user and stop the session.

2. Compute $MAC'_s = h(T_u || T_s || a_i)$ and check whether it is equal to the received MAC_s. If not, report login failure to the user and stop. Otherwise, conclude that the responding party is the real server.

3. Compute $MAC''_u = h(T_s || (a_i + 1))$ and session key $K_s = h((T_u || T_s) \oplus a_i)$, then send the message (T_s, MAC''_u) back to the server. Note that, in the message (T_s, MAC''_u), T_s is a response to the server.

When the message (T_s, MAC''_u) from U_i is received, the server performs the following steps to authenticate U_i and achieve key agreement:

1. Check if the received T_s is equal to the stored T_s. If it fails, reject U_i's login request and stop the session.

2. Compute $MAC'''_u = h(T_s || (a'_i + 1))$ and check whether it is equal to MAC''_u. If it is not, reject U_i's login request and stop the session. Otherwise, conclude that U_i is a legal user and permit the user U_i's login. At this moment, mutual authentication and session key agreement between U_i and the server are achieved. From now on, the user U_i and the server can use the session key K_s in their further secure communication until the end of the access session.

3. Weaknesses of Shieh-Wang's scheme

This section shows that Shieh-Wang's remote mutual authentication and key agreement scheme does not provide perfect forward secrecy [14] and is vulnerable to a privileged insider attack [15, 16]. In addition, the scheme has a time synchronization problem [7].

3.1. Perfect forward secrecy problem

Perfect forward secrecy [14] is a very important security requirement for evaluating a strong protocol. A protocol with perfect forward secrecy assures that even if one entity's long-term key (e.g. user password or server's secret key) is compromised, it will never reveal any old fresh session keys used before. For example, the well-known Diffie-Hellman key agreement scheme can provide perfect forward secrecy.

However, Shieh-Wang's scheme does not provide it because once the secret key x of the server is disclosed, all previous fresh session keys K_s will also be opened and hence previous communication messages will be learned. In the Shieh-Wang's scheme, suppose an attacker E obtains the secret key x from the compromised server and intercepts transmitted values (ID_i, T_u, T_s), from an open network. It is easy to obtain the information since its are exposed over an open network. Then, E can easily compute $a_i = h(ID_i \oplus x)$ by using the obtained ID_i. Finally, E can compute the shared session key $K_s = h((T_u||T_s) \oplus a_i)$ by using a_i, T_u and T_s. By using the K_s, E can eavesdrop all previous communication messages. Obviously, Shieh-Wang's scheme does not provide perfect forward secrecy.

3.2. Privileged insider's attack

In practice, a user uses the same password to access several servers for his/her convenience. In the registration phase of Shieh-Wang's scheme, U_i's password PW_i will be revealed to the remote server because it is transmitted directly to the server. Then, the privileged insider of the remote server may try to use PW_i to impersonate U_i to login the other servers that U_i has registered with outside this system [15]. If the targeted outside server adopts the normal password authentication scheme, it is possible that the privileged insider of the server can successfully impersonate U_i to login it by using PW_i. Although, it is also possible that all the privileged insiders of the server are trusted and U_i does not use the same password to access several servers, the implementers and the users of the scheme should be aware of such a potential weakness. Obviously, Shieh-Wang's scheme is vulnerable to a privileged insider attack.

3.3. Time synchronization problem

The schemes based on timestamps must overcome the problems of clock synchronization and delay-time limitation so that we better implement them in fast local area networks. Because Shieh-Wang's scheme also used timestamps to resist replay attacks, the scheme can lead to serious clock synchronization problems, namely that the user's time and the server's time must differ only in a small range [7]. For example, in a large-scale network, it is almost impossible to maintain the synchronization of clocks among all entities in the network and to guarantee the delay time of transmission. Therefore, we proposed a nonce-based and simplified scheme to avoid these clock synchronization and delay-time limitation problems.

4. Proposed scheme

This section proposes an improvement of Shieh-Wang's scheme so that they can withstand the above mentioned problems. The proposed scheme consists of three phases: the registration, the login and key agreement, and the password change. Figure 2 shows the proposed remote mutual authentication and key agreement scheme. It works as follows.

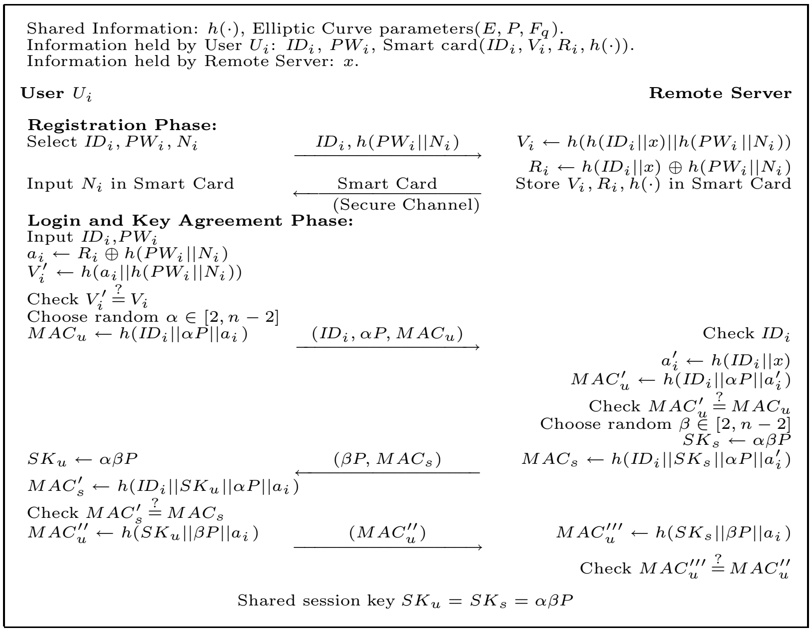

Figure 2. Proposed remote mutual authentication and key agreement scheme.

4.1. Registration phase

When a new user U_i wants to registration, the proposed registration phase performs the following steps:

1. U_i freely chooses his/her identity ID_i, password PW_i and a random number N_i. Then, U_i computes $h(PW_i||N_i)$ and submits it with ID_i to the remote server for registration. These private data must be sent in person or over a secure channel.

2. If the request is accepted, the server computes $V_i = h(h(ID_i||x)||h(PW_i||N_i))$ and $R_i = h(ID_i||x) \oplus h(PW_i||N_i)$, and issues U_i a smart card containing V_i, R_i and $h(\cdot)$.

3. After receiving the smart card, U_i enters N_i into his/her smart card.

4.2. Login and key agreement phase

When the user U_i wants to login to the server, he/she first inserts his/her smart card into a card reader then inputs his/her identity ID_i and password PW_i. The smart card then performs the following steps to begin an access session:

1. Compute $a_i = R_i \oplus h(PW_i||N_i)$.

2. Compute $V'_i = h(a_i||h(PW_i||N_i))$ and check whether it is equal to the stored V_i. If not, report password PW_i is incorrect to the user. This verification process performs only three times that can withstand password guessing attack by using the stolen or lost smart card.

3. Choose a random number $\alpha \in [2, n-2]$, and compute αP and $MAC_u = h(ID_i||\alpha P||a_i)$.

4. Send the message $(ID_i, \alpha P, MAC_u)$ to the server.

After receiving the message $(ID_i, \alpha P, MAC_u)$ from U_i, the server performs the following steps to assure the integrity of the message, respond to U_i, and challenge U_i to avoid replay:

1. Check the correctness of ID_i. If it is incorrect, reject U_i's login request and stop the session.
2. Compute $a_i' = h(ID_i||x)$, $MAC_u' = h(ID_i||\alpha P||a_i')$, and check whether MAC_u' is equal to the received MAC_u. If it is not, reject U_i's login and stop the session.
3. Choose a random number $\beta \in [2, n-2]$, compute βP, session key $SK_s = \alpha\beta P$ and $MAC_s = h(ID_i||SK_s||\alpha P||a_i')$. Then, send the message $(\beta P, MAC_s)$ back to U_i and wait for response from U_i.

On receiving the message $(\beta P, MAC_s)$ from the server, the smart card of U_i performs the following steps to authenticate the server, achieve session key agreement, and respond to the server:

1. Compute session key $SK_u = \alpha\beta P$ and $MAC_s' = h(ID_i||SK_u||\alpha P||a_i)$, and check whether it is equal to the received MAC_s. If not, report login failure to the user and stop. Otherwise, conclude that the responding party is the real server.
2. Compute $MAC_u'' = h(SK_u||\beta P||a_i)$ and send the message (MAC_u'') back to the server. Note that, in the message (MAC_u'') is a response to the server.

When the message (MAC_u'') from U_i is received, the server performs the following steps to authenticate U_i and achieve key agreement:

1. Compute $MAC_u''' = h(SK_s||\beta P||a_i')$.
2. Check whether MAC_u''' is equal to MAC_u''. If it is not, reject U_i's login request and stop the session. Otherwise, conclude that U_i is a legal user and permit the user U_i's login. At this moment, mutual authentication and session key agreement between U_i and the server are achieved. From now on, the user U_i and the server can use the session key $SK_u = SK_s = \alpha\beta P$ in their further secure communication until the end of the access session.

4.3. Password change scheme

The password change scheme is invoked whenever a user U_i wants to change his password PW_i. By invoking this scheme, U_i can easily change his password without taking any assistance from the remote server. Figure 3 shows the proposed password change scheme and it works as follows.

1. U_i inserts his/her smart card into a card reader then inputs his/her identity ID_i and password PW_i.
2. The smart card computes $a_i = R_i \oplus h(PW_i||N_i)$.
3. The smart card computes hash value $V_i' = h(a_i||h(PW_i||N_i))$ and verifies it with stored V_i. If it holds, the smart card proceeds to the next step; otherwise, terminates the operation. This verification process performs only three times that can withstand password guessing attack by using stolen or lost smart card.

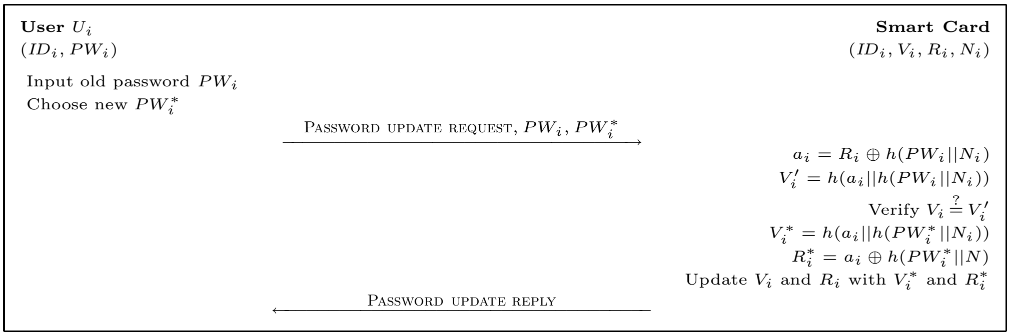

Figure 3. Password change scheme.

4. U_i submits a new password PW_i^*.
5. The smart card computes $V_i^* = h(a_i||h(PW_i^*||N_i))$ and $R_i^* = a_i \oplus h(PW_i^*||N)$.
6. The password has been changed now with the new password PW_i^* and the smart card replaced the previously stored V_i and R_i values by V_i^* and R_i^* values.

5. Security analysis

This section analyzes the security of the proposed remote mutual authentication and key agreement scheme. First, we define the security terms [14, 17, 18, 23, 24, 25] needed to conduct an analysis of the proposed scheme. They are as follows.

Definition 1 *A weak secret key (user's password PW_i) is the value of low entropy $W(k)$, which can be guessed in polynomial time.*

Definition 2 *A strong secret key (server's secret key x) is the value of high entropy $S(k)$, which cannot be guessed in polynomial time.*

Definition 3 *The Elliptic Curve Discrete Logarithm Problem ($ECDLP$) is as follows: given a public key point $V = \alpha P$, it is hard to compute the secret key α.*

Definition 4 *The Elliptic Curve Diffie-Hellman Problem ($ECDHP$) is as follows: given point elements αP and βP, it is hard to find $\alpha\beta P$.*

Definition 5 *A secure one-way hash function $y = h(x)$ is one where given x to compute y is easy, and given y to compute x is hard.*

The following eight security properties [14, 22, 24] must be considered for the proposed protocol: a replay attack,a guessing attack, a reflection and parallel session attack, a privileged insider attack, a mutual authentication, a perfect forward secrecy, a fast wrong password detection and a secure password change. Regarding the above mentioned definitions, the followings are used to analyze the eight security properties of the proposed scheme.

1. *The proposed scheme can resist a replay attack:* When the server receives the message $(ID_i, \alpha P,\ MAC_u)$, it includes a fresh Diffie-Hellman element αP from U_i. Therefore, the server must send back the received T to U_i including MAC_s as a response. When U_i receives the message $(\beta P, MAC_s)$, it includes the fresh Diffie-Hellman element αP in the MAC_s. Note that αP is fresh on each session. Besides, MAC_u and MAC_s guarantee their integrity and source, respectively. In addition, it is impossible to create corresponding responses and their message authentication codes, MAC_s and MAC''_u, without knowing the shared secret value a_i between U_i and the server. Therefore, except for U_i and the server, no one can pass the challenges.

2. *The proposed scheme can resist a guessing attack:* Assume a user lost his/her smart card and it is found by an attacker or an attacker steals a user's smart card. The attacker, however, cannot impersonate a legitimate user U_i by using the smart card because no one can reveal the PW_i from value R_i in the smart card without knowing the system's secret key x. Furthermore, the server's secret key x is protected by the secure one-way hash function $h(\cdot)$. It is computationally infeasible to derive x from the value $h(ID_i||x)$. In the same way, the shared secret a_i between U_i and the server cannot be derived from the message authentication code MAC_u, MAC_s, or MAC''_u. Therefore, a_i is safely shared only between U_i and the server.

3. *The proposed scheme can resist a reflection attack and a parallel session attack [6]:* In the proposed scheme, the reflection attack and a parallel session attack will fail because of the asymmetric structure of the message authentication codes MAC_u and MAC''_u. Note that $MAC_u \leftarrow h(T||\alpha P||a_i)$ and $MAC''_u \leftarrow h(SK_u||\beta P||a_i)$. Therefore, the proposed scheme can resist a reflection attack and a parallel session attack.

4. *The proposed scheme can resist a privileged insider attack:* Since U_i registers to the server by presenting $h(PW_i||N_i)$ instead of PW_i, the insider of the server cannot directly obtain PW_i without knowing of random nonce N_i. Therefore, the proposed scheme can withstand the insider attack.

5. *The proposed scheme provides the mutual authentication:* Mutual authentication between U_i and the server is achieved, because U_i and the server authenticate each other with the message authentication codes MAC_s, and MAC''_u, respectively. Since nobody can create the correct message authentication codes without knowing the shared secret value a_i between U_i and the server, a_i is used to confirm the legitimacy of each party. In other words, it is infeasible for an intruder or a pretended server to masquerade as a legal party. Also, the proposed scheme uses the Elliptic Curve Diffie-Hellman key exchange algorithm in order to provide mutual explicit key authentication. Then, the key is explicitly authenticated by a mutual confirmation session key, $SK = \alpha\beta P$.

6. *The proposed scheme provides a perfect forward secrecy:* In the proposed scheme, since the Elliptic Curve Diffie-Hellman key exchange algorithm is used to generate a session key $SK = \alpha\beta P$, perfect forward secrecy is ensured because an attacker with a compromised server's secret key x is only able to obtain the αP and βP from an earlier session. In addition, it is also computationally infeasible to obtain the session key $\alpha\beta P$ from αP and βP, as it is a ECDLP and a ECDHP.

7. *The proposed scheme provides a fast wrong password detection:* In Shieh-Wang's scheme, if user U_i input a wrong password PW_i by mistake, this wrong password will be detected by the remote server in the login and key agreement phase. Therefore, Shieh-Wang's scheme is slow to detect the user's wrong password.

In contrast to Shieh-Wang's scheme, in the proposed scheme, if user U_i inputs the wrong password by mistake, this wrong password will be quickly detected by a smart card since the smart card can verify $V_i' = V_i$ using the stored K_i in step 2 of the login and key agreement phase. Therefore, the proposed scheme provides fast wrong password detection.

8. *The proposed scheme provides a secure password change:* Shieh-Wang's scheme does not provide password change scheme. The proposed password change scheme is simple and secure. Because the smart card can verify V_i^* using the stored V_i in step 3 of the password change scheme, when the smart card was lost or steal, unauthorized users cannot change the password of the card without knowing the U_i's password PW_i. Therefore, the proposed password change scheme provides secure password change.

We compared the proposed scheme with other related schemes [9, 10, 11, 12, 13] as well as Shieh-Wang's scheme [8]. Table 1 shows the comparison results of the security properties of the proposed scheme and various other remote authentication schemes based on smart cards. As show in Table 1, in contrast to related schemes, the proposed scheme is more secure and practical for smart card-based remote mutual authentication and key agreement.

Table 1. Security properties of the proposed scheme with other related schemes.

Security properties	Liaw et al. [9]	Cheng et al. [10]	Wang et al. [11]	Yang et al. [12]	Xu et al. [13]	Shieh-Wang [8]	Proposed scheme
Replay attack	Secure	Secure	Secure	Secure	Secure	Secure	Secure
Guessing attack	Secure	Insecure	Insecure	Secure	Secure	Secure	Secure
Reflection attack	Secure	Secure	Insecure	Secure	Secure	Secure	Secure
Parallel session attack	Secure	Insecure	Secure	Secure	Secure	Secure	Secure
Privileged insider attack	Insecure	Insecure	Secure	Secure	Insecure	Insecure	Secure
Mutual authentication	Provide	Provide	Provide	Provide	Provide	Provide	Provide
Explicit mutual authentication	No provide	No provide	No provide	No provide	No provide	No provide	Provide
Session key agreement	Provide	Provide	Provide	Provide	Provide	Provide	Provide
Perfect forward secrecy	Provide	No provide	No provide	Provide	No provide	No provide	Provide
Wrong password detection	Slow	Fast	Fast	Slow	Slow	Slow	Fast
Secure password change	No provide	Provide	Provide	No provide	No provide	No provide	Provide
Time synchronization	No required	Required	Required	No required	Required	No required	No required
No verification table	Yes	No	Yes	Yes	Yes	Yes	Yes

6. Performance comparisons

This section analyzes the efficiency of the proposed scheme. Table 2 provides computational costs of the proposed scheme with various other related schemes [9, 10, 11, 12, 13] as well as Shieh-Wang's scheme [8] in regards to the registration, login, authentication and key agreement, and password change phases.

In the registration phase, the 3 time one-way function operation and 1 time exclusive-OR operation are required to resist an insider attack. In the authentication and key agreement phase, the 4 times modular addition operations and 6 times one-way function operations are required to provide session key agreement and perfect forward secrecy. In the password change phase, the 4 time one-way function operations and 2 time exclusive-OR operations are required to resist a stolen or lost smart card attack. The symmetric key computations and hash functions are faster than the asymmetric key computations.

Table 2. Computational costs of the proposed scheme with other related schemes.

	Registration phase	Login phase	Authentication and key agreement phase	Password change phase
Proposed scheme	$3T(f)$ $1T(\oplus)$	$3T(f)$ $1T(\oplus)$	$4T(MA)$ $6T(f)$	$4T(f)$ $2T(\oplus)$
Shieh-Wang's scheme [8]	$1T(f)$ $1T(\oplus)$	$1T(f)$ $1T(\oplus)$	$8T(f)$ $3T(\oplus)$	No support
Liaw et al.'s scheme [9]	$1T(f)$ $1T(\oplus)$	$1T(f)$ $1T(\oplus)$	$4T(ME)$ $2T(f)$ $6T(S)$ $2T(\oplus)$	$2T(\oplus)$
Cheng et al.'s scheme [10]	$2T(f)$ $1T(\oplus)$	$(n+1)T(f)$ $2T(\oplus)$	$(n+3)T(f)$ $3T(\oplus)$	$3T(f)$ $5T(\oplus)$
Wang et al.'s scheme [11]	$3T(f)$ $3T(\oplus)$	$4T(f)$ $5T(\oplus)$	$4T(f)$ $5T(\oplus)$	$4T(f)$ $4T(\oplus)$
Yang et al.'s scheme [12]	$5T(f)$ $3T(\oplus)$	$1T(f)$ $1T(\oplus)$ $1T(ME)$	$3T(ME)$ $4T(A)$	$2T(f)$ $2T(\oplus)$
Xu et al.'s scheme [13]	$1T(ME)$ $2T(f)$ $1T(\oplus)$	$2T(ME)$ $3T(f)$ $1T(\oplus)$	$4T(ME)$ $6T(f)$	No support

$T(f)$: computation cost of one-way function; $T(\oplus)$: computation cost of exclusive-OR operation or addition operation; $T(S)$: computation cost of symmetric encryption; $T(A)$: computation cost of asymmetric encryption; $T(MA)$: computation cost of modular addition; $T(ME)$: computation cost of modular exponentiation.

On a typical workstation, the asymmetric key computations can be performed 2 times per second, symmetric key computations can be performed 2,000 times per second and hash function can be performed 20,000 times per second. To provide the computational efficiency, we can change the the Diffie-Hellman key exchange algorithm with nonce-based key exchange algorithm in the proposed scheme. In this case, the proposed scheme cannot provide the perfect forward secrecy. But, the computation costs are very low because only a few hashing function computations are needed like other related schemes.

In addition, other security requirements including session key agreement can still be satisfied unlike other related schemes. Therefore, as in Tables 1 and 2, we can see that the proposed scheme not only is secure to various cryptographic attacks, but also has the reasonable computational costs.

7. Conclusion

This paper demonstrated that Shieh-Wang's scheme does not provide perfect forward secrecy and is vulnerable to a privileged insider's attack, and then an improved scheme based on Elliptic Curve Diffie-Hellman problem and one-way hash function was presented in order to resolve such problems. As a result, in contrast to Shieh-Wang's scheme, the proposed scheme is able to provide greater security and practicality.

Acknowledgments

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency (NIPA-2011-(C1090-1121-0002)).

References

[1] P. Peyret, G. Lisimaque, T.Y. Chua, "Smart cards provide very high security and flexibility in subscribers management," IEEE Transactions on Consumer Electronics, Vol. 36, No. 3, pp. 744–752, 1990.

[2] D. Sternglass, "The future is in the pc cards," IEEE Spectrum, Vol. 29, No. 6, pp. 46–50, 1992.

[3] L. Lamport, "Password authentication with insecure communication," Communications of the ACM, Vol. 24, pp. 770–772, 1981.

[4] H.M. Sun, "An efficient remote user authentication scheme using smart cards," IEEE Transactions on Consumer Electronics, Vol. 46, No. 4, pp. 958–961, November 2000.

[5] H.Y. Chien, J.K. Jan, Y.H. Tseng, "An efficient and practical solution to remote authentication: smart card," Computers and Security. Vol. 21, No. 4, pp. 372–375, 2002.

[6] C.L. Hsu, "Security of Chien et al's remote user authentication scheme using smart card," Computer Standards and Interfaces, Vol. 26, pp. 167–169, 2004.

[7] W.S. Juang, "Efficient password authenticated key agreement using smart cards," Computers and Security, Vol. 23, pp. 167–173, 2004.

[8] W.G. Shieh, J.M. Wang, "Efficient remote mutual authentication and key agreement," Computers and Security, Vol. 25, pp. 72–77, 2006.

[9] H.T. Liaw, J.F. Lin, W.C. Wu, "An efficient and complete remote user authentication scheme using smart cards," Mathematical and Computer Modelling, vol. 44, pp. 223–228, 2006.

[10] T.F. Cheng, J.S. Lee, C.C. Chang, "Security enhancement of an IC-card-based remote login mechanism Computer Networks," Vol. 51, No. 9, pp. 2280–2287, 2007.

[11] X.M. Wang, W.F. Zhang, J.S. Zhang, M.K. Khan, "Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards," Computer Standards & Interfaces, Vol. 29, No. 5, pp. 507–512, 2007.

[12] G. Yang, D.S. Wong, H. Wang, X. Deng, "Two-factor mutual authentication based on smart cards and passwords," Journal of Computer and System Sciences, Vol. 74, No. 7, pp. 1160–1172, 2008.

[13] J. Xu, W.T. Zhu, D.G. Feng, "An improved smart card based password authentication scheme with provable security," Computer Standards & Interfaces, Vol. 31, pp. 723–728, 2009.

[14] A.J. Menezes, P.C. Oorschot, S.A. Vanstone, Handbook of Applied Cryptography, CRC Press. New York, 1997.

[15] W.C. Ku, H.M. Chuang, M.J. Tsaur, "Vulnerabilities of Wu-Chieu's improved password authentication scheme using smart cards," IEICE Trans. Fundamentals, Vol. E88-A, No. 11, pp. 3241–3243, November 2005.

[16] E.J. Yoon, K.Y. Yoo, "Two security problems of efficient remote mutual authentication and key agreement," IEEE Proceedings of Future Generation Communication And Networking (FGCN'07), Vol. 2, pp. 66–70, Jeju, Korea, 2007.

[17] V. Miller, "Uses of elliptic curves in cryptography," Proceedings of Crypto'85, Santa Barbara, USA, pp. 417–426, 1986.

[18] N. Koblitz, "Elliptic Curve Cryptosystems," Mathematics of Computation, Vol. 48, pp. 203–209, 1987.

[19] N. Koblitz, "CM-curves with good cryptographic properties," Proceedings of Crypto'91, Santa Barbara, USA, 1992.

[20] R.M. Needham, M.D. Schroeder, "Using encryption for authentication in large networks of computers," Communications of the ACM, Vol. 21, No. 12, pp. 993–999, 1978.

[21] Certicom Research, Standard for efficient cryptography, SEC 1: EC Cryptography, Ver. 1.0, 2000.

[22] C. Boyd, A. Mathuria, Protocols for Authentication and Key Establishment, Springer-Verlag, 2003.

[23] W. Diffie, M. Hellman, “New directions in cryptography,” IEEE Transaction on Information Theory, Vol. IT-22, No. 6, 1976. 644-654

[24] B. Schneier, Applied Cryptography-Protocols: Algorithms and Source Code in C, 2nd edi., John Wiley & Sons Inc., 1995.

[25] A. Menezes, T. Okamoto, S. Vanstone, “Reducing elliptic curve logarithms to logarithms in a finite field,” IEEE Transactions on Information Theory, Vol. 39, pp. 1639–1646, 1993.

[26] G. Frey, H. Ruck, “A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves,” Mathematics of Computation, Vol. 62, pp. 865–874, 1994.