Secure Remote User Authentication Scheme Using Bilinear Pairings

Eun-Jun Yoon[1], Wan-Soo Lee[2], and Kee-Young Yoo[2,⋆]

[1] Faculty of Computer Information, Daegu Polytechnic College,
395 San 130 Manchon-Dong, SooSung-Gu, Daegu 706-711, South Korea
ejyoon@tpic.ac.kr
[2] Department of Computer Engineering, Kyungpook National University,
1370 Sankyuk-Dong, Buk-Gu, Daegu 702-701, South Korea
Tel.: +82-53-950-5553; Fax: +82-53-957-4846
complete2@infosec.knu.ac.kr, yook@knu.ac.kr

Abstract. In 2006, Das et al. proposed a remote user authentication scheme using the properties of bilinear pairings. The current paper, however, demonstrates that Das et al.'s scheme is still vulnerable to an impersonation attack and an off-line password guessing attack. Furthermore, we present an improved authentication scheme based on bilinear computational Diffie-Hellman problem and one-way hash function to the schemes, in order to isolate such problems.

Keywords: Authentication, Password, Key agreement, Cryptanalysis, Smart card, Bilinear pairings.

1 Introduction

Remote user authentication is an important part of security, along with confidentiality and integrity, for systems that allow remote access over untrustworthy networks, like the Internet. As such, a remote password authentication scheme authenticates the legitimacy of users over an insecure channel, where the password is often regarded as a secret shared between the remote system and the user. With knowledge of the password, the user can use it to create and send a valid login message to a remote system in order to gain access. Meanwhile, the remote system also uses the shared password to check the validity of the login message and to authenticate the user.

ISO 10202 standards have been established for the security of financial transaction systems that use integrated circuit cards (IC cards or smart cards). The smart card originates from the IC memory card which has been in the industry for about 10 years [1][2]. The main characteristics of a smart card are its small size and low-power consumption. In general, a smart card contains a microprocessor which can quickly manipulate logical and mathematical operations, RAM, which is used as a data or instruction buffer, and ROM which stores the

---

⋆ Corresponding author.

D. Sauveron et al. (Eds.): WISTP 2007, LNCS 4462, pp. 102–114, 2007.

user's secret key and the necessary public parameters and algorithmic descriptions of the executing programs. The merits of a smart card regarding password authentication are its simplicity and its efficiency in terms of the log-in and authentication processes.

In 2000, Joux [3] discovered the bilinear computational Diffie-Hellman problem of the groups over elliptic curves. This hard problem can be considered as a new security assumption to develop cryptosystems. Since then, several variant security schemes have been presented [4][5][6][7]. Bilinear pairings are an effective method to reduce the complexity of the discrete log problem in a finite field and provides a good setting for the bilinear computational Diffie-Hellman problem.

In 2006, Das et al. [8] proposed a remote user authentication scheme using the properties of bilinear pairings that can prohibit the scenario of many logged in users with the same login-ID, and provide a flexible password change option to the registered users without any assistance from the remote system. The current paper, however, demonstrates that Das et al.'s scheme is still vulnerable to an impersonation attack [9], where an attacker easily masquerade as another legal users in order to access the resources of a remote system, and an off-line password guessing attack [10], where an attacker can easily guess a legal users's password and can impersonate an legal users. Furthermore, we present an improved authentication scheme based on bilinear computational Diffie-Hellman problem [3] and one-way hash function [9] to the schemes, in order to isolate such problems. As a result, the proposed scheme is more secure than Das et al.'s scheme. Also, it provides mutual authentication between the user and remote system and it has the same advantages of other schemes. In addition, the proposed scheme does not require time synchronization or delay-time limitations between the user and remote system, unlike Das et al.'s scheme.

The remainder of this paper is organized as follows: In the next section, we give some preliminaries of bilinear pairings. Section 3 briefly reviews Das et al.'s scheme and then Section 4 demonstrates the security weakness of Das et al.'s scheme. The proposed authentication scheme is presented in Section 5, while Sections 6 discusses the security of the proposed protocol. The conclusion is given in Section 7.

2 Preliminaries

This section summarizes the underlying primitives used throughout this paper. This primitive include modified Weil pairing, bilinear computational Deffie-Hellman assumption, symmetric encryption scheme, one-way hash function and map-to-point function [3][7][8].

2.1 Bilinear Pairings

Suppose G_1 is an additive cyclic group generated by P, whose order is a prime q, and G_2 is a multiplicative cyclic group of the same order. A map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ is called a bilinear mapping if it satisfies the following properties:

1. Bilinear: $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$, for all $P, Q \in G_1$ and all $a, b \in Z_q^*$.
2. Non-degenerate: there exists $P, Q \in G_1$ such that $\hat{e}(P, Q) \neq 1$.
3. Computable: there is an efficient algorithm to compute $\hat{e}(P, Q)$ for all P, $Q \in G_1$.

We note that G_1 is the group of points on an elliptic curve and G_2 is a multiplicative subgroup of a finite field. Typically, the mapping $\hat{e}$ will be derived from either the Weil or the Tate pairing on an elliptic curve over a finite field.

2.2 Mathematical Problems

Definition 1. *Discrete Logarithm Problem (DLP): Given* $Q, R \in G_1$*, find an integer* $x \in Z_q^*$ *such that* $R = xQ$*.*

The MOV and FR reductions: Menezes et al. [11] and Frey and Ruck [12] show a reduction from the DLP in G_1 to the DLP in G_2. The reduction is: Given an instance $Q, R \in G_1$, where Q is a point of order q, find $x \in Z_q^*$, such that $R = xQ$. Let T be an element of G_1 such that $g = \hat{e}(T, Q)$ has order q, and let $h = \hat{e}(T, R)$. Using bilinear property of $\hat{e}$, we have $\hat{e}(T, R) = \hat{e}(T, Q)^x$. Thus, DLP in G_1 is no harder than the DLP in G_2.

Definition 2. *Bilinear Computational Diffie-Hellman Problem (BCDHP): Given* (P, aP, bP) *for* $a, b \in Z_q^*$*, compute* abP*.*

The advantage of any probabilistic polynomial-time algorithm $\mathcal{A}$ in solving the BCDHP in G_1, is defined as $Adv_{\mathcal{A},G_1}^{CDH} = Prob[\mathcal{A}(P, aP, bP, abP) = 1 : a, b \in Z_q^*]$. For every probabilistic algorithm A, $Adv_{\mathcal{A},G_1}^{CDH}$ is negligible.

3 Review of Das et al.'s Scheme

This section briefly reviews Das et al.'s authentication scheme [8]. Das et al.'s scheme consists of mainly three phases: Setup, registration, and authentication phase. Figure 1 shows Das st al.'s authentication scheme. The scheme works as follows:

3.1 Setup Phase

Let G_1 is an additive cyclic group of order prime q, and G_2 is a multiplicative cyclic group of the same order. Let P is a generator of G_1, $\hat{e} : G_1 \times G_1 \in G_2$ is a bilinear mapping and $H : \{0, 1\}^* \rightarrow G_1$ is a cryptographic hash function. The remote system RS selects a secret key s and computes the public-key as $Pub_{RS} = sP$. Then, the RS publishes the system parameters $< G_1, G_2, \hat{e}, q, P, Pub_{RS}, H(\cdot) >$ and keeps s secret.

3.2 Registration Phase

This phase is executed by the following steps when a new user wants to register with the RS:

R1. Suppose a new user U_i wants to register with the RS, then U_i submits his identity ID_i and password PW_i to the RS.
R2. On receiving the registration request, the RS computes $Reg_{ID_i} = s \cdot H(ID_i) + H(PW_i)$.
R3. The RS personalizes a smart card with the parameters ID_i, Reg_{ID_i}, $H(\cdot)$ and sends the smart card to U_i over a secure channel.

3.3 Authentication Phase

This phase is executed every time whenever a user logs into the RS. The phase is further divided into the login and verification phases. In the login phase, user sends a login request to the RS. The login request comprises with a dynamic coupon, called DID, which is dependent on the user's ID, password and RS's secret key. The RS allows the user to access the system only after successful verification of the login request.

Login Phase: The user U_i inserts the smart card in a terminal and keys ID_i and PW_i. If ID_i is identical to the one that is stored in the smart card, the smart card performs the following operations:

L1. Computes $DID_i = T \cdot Reg_{ID_i}$, where T is the user system's timestamp.
L2. Computes $V_i = T \cdot H(PW_i)$.
L3. Sends the login request $\{ID_i, DID_i, V_i, T\}$ to the RS over a public channel.

Verification Phase: Let the RS receives the login message $\{ID_i, DID_i, V_i, T\}$ at time T^* $(\geq T)$. The RS performs the following operations to verify the login request:

V1. Verifies the validity of the time interval between T^* and T. If $(T^* - T) \leq \Delta T$, the RS proceeds to the Step (V2), where ΔT denotes the expected valid time interval for transmission delay. Otherwise, rejects the login request. We note that at the time of registration, the user and the RS have agreed on the accepted value of the transmission delay ΔT.
V2. Checks whether $\hat{e}(DID_i - V_i, P) = \hat{e}(H(ID_i), Pub_{RS})^T$. If it holds, the RS accepts the login request; otherwise, rejects it.

3.4 Password Change Phase

When a user U_i wants to change his password, U_i can change his password without taking any assistance from the RS by invoking this phase. Das et al.'s password change phase works as follows:

Shared Information: $G_1, G_2, \hat{e}, q, P, Pub_{RS}, H(\cdot)$.
Information held by User U_i: ID_i, PW_i, Smart card($ID_i, Reg_{ID_i}, H(\cdot)$).
Information held by Remote System RS: s.

User U_i **Remote System**

Registration Phase:
Select ID_i, PW_i $\xrightarrow{ID_i, PW_i}$
$Reg_{ID_i} \leftarrow s \cdot H(ID_i) + H(PW_i)$
Store $ID_i, Reg_{ID_i}, H(\cdot)$ in Smart Card
$\xleftarrow[\text{(Secure Channel)}]{\text{Smart Card}}$

Login and Verification Phase:
Input ID_i,PW_i
Pick up T
$DID_i \leftarrow T \cdot Reg_{ID_i}$
$V_i \leftarrow T \cdot H(PW_i)$ $\xrightarrow{\{ID_i, DID_i, V_i, T\}}$ Check $(T^* - T) \leq \Delta T$
Check $\hat{e}(DID_i - V_i, P) \overset{?}{=} \hat{e}(H(ID_i), Pub_{RS})^T$

Fig. 1. Das et al.'s Authentication Scheme

P1. U_i attaches the smart card to a terminal and keys ID_i and PW_i. If ID_i is identical to the one that is stored in the smart card, proceeds to the Step (P2); otherwise, terminates the operation.
P2. U_i submits a new password PW_i^*.
P3. The smart card computes $Reg_{ID_i}^* = Reg_{ID_i} - H(PW_i) + H(PW_i^*) = s \cdot H(ID_i) + H(PW_i^*)$.
P4. The password has been changed now with the new password PW_i^* and the smart card replaced the previously stored Reg_{ID_i} value by $Reg_{ID_i}^*$ value.

4 Cryptanalysis of Das et al.'s Scheme

This section demonstrates that Das et al.'s authentication scheme is vulnerable to some attacks.

4.1 Impersonation Attack

This subsection demonstrates that Das et al.'s scheme is vulnerable to an impersonation attack, where an attacker can easily impersonate other legal users to access the resources at a remote system. Suppose that an attacker E has eavesdropped a valid message (ID_i, DID_i, V_i, T) from an open network. It is easy to obtain the information since it is exposed over an open network. Then, in the Login Phase, an impersonation attack proceeds as follows:

(1) E chooses a timestamp T' and computes $r = T'/T$, where T' is E's the current date and time for succeeding with Step (V2) of the Authentication Phase.
(2) E computes $DID'_i = r \cdot DID_i$ and $V'_i = r \cdot V_i$.
(3) E sends a forged message (ID_i, DID'_i, V'_i, T') to RS.
(4) It is easy to check whether RS will accept this forged message, as $\hat{e}(DID'_i - V'_i, P) \overset{?}{=} \hat{e}(H(ID_i), Pub_{RS})^{T'}$. Its correctness easy to see that the Verification Step (V2) of E's forged login request is verified by the following:

$$\begin{aligned}
\hat{e}(DID'_i - V'_i, P) &= \hat{e}(r \cdot DID_i - r \cdot V_i, P) \\
&= \hat{e}(r \cdot T \cdot Reg_{ID_i} - r \cdot T \cdot H(PW_i), P) \\
&= \hat{e}(r \cdot T \cdot (s \cdot H(ID_i) + H(PW_i)) - r \cdot T \cdot H(PW_i), P) \\
&= \hat{e}(r \cdot T \cdot s \cdot H(ID_i), P) \\
&= \hat{e}(s \cdot H(ID_i), P)^{r \cdot T} \\
&= \hat{e}(H(ID_i), sP)^{T'} \\
&= \hat{e}(H(ID_i), Pub_{RS})^{T'}
\end{aligned}$$

(5) Finally, RS will accept the attacker's login request, making Das et al.'s scheme insecure.

4.2 Off-Line Password Guessing Attack

In the login phase of Das et al.'s scheme, suppose that an attacker E has eavesdropped a valid message (ID_i, DID_i, V_i, T) from an open network. Then, in order to obtain the password PW_i of user U_i, the off-line password guessing attack proceeds as follows:

(1) E makes a guess at the secret password PW'_i.
(2) E computes $T \cdot H(PW'_i)$, where T is intercepted U_i's current timestamp.
(3) E checks if $V_i = T \cdot H(PW'_i)$.
(4) If the computed value is the same as V_i, then E guesses the legitimate user U_i's password PW_i. Otherwise, E repeatedly performs Steps (1), (2) and (3) until $V_i = T \cdot H(PW'_i)$.

If a user loses his smart card and it is found out by an attacker or an attacker steals a user's smart card, then the attacker can easily impersonate the legitimate user U_i by using the guessed password PW'_i in the Login Phase. Furthermore, if some users employ the same password for multiple accounts, those will be compromised as well. As a result, Das et al.'s scheme is vulnerable to an off-line password guessing attack.

5 Proposed Scheme

This section proposes an improvement of Das et al.'s scheme so that they can withstand the above mentioned attacks. In addition, the proposed scheme provides mutual authentication between the user and a remote system and does not require time synchronization or a delay-time limitations between the user and the remote system. In order to prevent the problems of clock synchronization or a delay-time limitations, the proposed scheme adopts a nonce-based protocol [13] instead of a timestamp-based protocol. The security of the proposed scheme is based on Discrete Logarithm Problem (DLP), Bilinear Computational Diffie-Hellman problem (BCDHP) (Definitions 1, 2 in Section Preliminaries) and one-way hash function, and consists of setup, registration, and authentication phases. Figure 2 shows the proposed authentication scheme. The scheme works as follows:

5.1 Setup Phase

Let G_1 is an additive cyclic group of order prime q, and G_2 is a multiplicative cyclic group of the same order. Let P is a generator of G_1, $\hat{e} : G_1 \times G_1 \in G_2$ is a bilinear mapping, $H : \{0,1\}^* \rightarrow G_1$ is a cryptographic hash function and $F(\cdot)$ is a collision resistant one-way hash function with an output size of 512 bits, e.g. SHA-512 [9]. The remote system RS selects a secret key s. Then, the RS publishes the system parameters $< G_1, G_2, \hat{e}, q, P, H(\cdot), F(\cdot) >$ and keeps s secret.

5.2 Registration Phase

This phase is executed by the following steps when a new user wants to register with the RS:

R1. Suppose a new user U_i wants to register with the RS, then U_i selects his identity ID_i, password PW_i and random number N freely.
R2. U_i computes $F(PW_i|N)$, where $|$ is a concatenation operation, and then submits ID_i and $F(PW_i|N)$ to the RS.
R3. On receiving the registration request, the RS computes $U = H(ID_i, ID_s)$, $K_i = s \cdot U$, $VK_i = F(K_i)$ and $Reg_{ID_i} = K_i + H(F(PW_i|N))$, where ID_s is the RS's identity.
R4. The RS personalizes a smart card with the parameters U, VK_i, Reg_{ID_i}, $H(\cdot)$, $F(\cdot)$ and sends the smart card to U_i over a secure channel.
R5. U_i enters N into his smart card.

5.3 Authentication Phase

This phase is executed every time whenever a user logs into the RS. The phase is further divided into the login and session key agreement phases.

Login Phase: If the user U_i wants to login, U_i inserts the smart card in a terminal and keys ID_i and PW_i. Then, the smart card performs the following operations:

L1. Extracts K_i from the smart card by computing $Reg_{ID_i} - H(F(PW_i|N))$.
L2. Computes hash value $F(K_i)$ and verifies it with stored VK_i. If it holds, the card performs next Step. Otherwise, the card rejects U_i's login request. This verification process performs only three times that can withstand password guessing attack by using stolen or lost smart card.
L3. Chooses a fresh random value $a \in Z_q^*$, and computes $C_1 = aP$.
L4. Sends a login request message $\{ID_i, C_1\}$ to RS.

Session Key Agreement Phase: Upon receiving the authentication request message $\{ID_i, C_1\}$, the remote system and smart card execute the following steps for mutual authentication and session key agreement between the user U_i and the remote system.

K1. The system verifies the format of ID_i. If the format is incorrect, the system rejects the login request. Otherwise, the system computes $U = H(ID_i, ID_s)$ and $K_i^* = s \cdot U$. Then, the system chooses a fresh random value $b \in Z_q^*$, and computes $C_2 = bP$, $sk = \hat{e}(C_1, bU) = \hat{e}(aP, bU) = \hat{e}(P, U)^{ab}$ and $C_3 = F(ID_i, K_i^*, sk, C_1)$. The system sends back the message $\{C_2, C_3\}$.
K2. Upon receiving the message $\{C_2, C_3\}$, the smart card computes $sk^* = \hat{e}(C_2, aU) = \hat{e}(bP, aU) = \hat{e}(P, U)^{ab}$ and $C_3^* = F(ID_i, K_i, sk^*, C_1)$. Then, the smart card compares C_3 and C_3^*. If they are equal, the user U_i believes that the responding part is the real system, otherwise the user U_i interrupts the connection. Finally, the smart card computes $C_4 = F(ID_i, K_i, sk^*, C_2)$ and sends this authentication token to the system for mutual authentication and session key agreement.
K3. Upon receiving the message $\{C_4\}$, the system computes $C_4^* = F(ID_i, K_i^*, sk, C_2)$ and compares C_4 and C_4^*. If they are equal, the system can ensure that the user U_i is legal.

After mutual authentication and session key agreement between the user and the remote system, sk and sk^* are used as a session key, respectively.

5.4 Password Change Phase

This phase is invoked whenever a user U_i wants to change his password. By invoking this phase, U_i can easily change his password without taking any assistance from the RS. The phase works as follows:

P1. U_i attaches the smart card to a terminal and keys ID_i and PW_i.
P2. The smart card computes $K_i = Reg_{ID_i} - H(F(PW_i|N))$.
P3. The smart card computes hash value $F(K_i)$ and verifies it with stored VK_i. If it holds, the smart card proceeds to the Step (P4); otherwise, terminates the operation. This verification process performs only three times that can withstand password guessing attack by using stolen or lost smart card.
P4. U_i submits a new password PW_i^*.
P5. The smart card computes $Reg_{ID_i}^* = K_i + H(F(PW_i^*|N))$.
P6. The password has been changed now with the new password PW_i^* and the smart card replaced the previously stored Reg_{ID_i} value by $Reg_{ID_i}^*$ value.

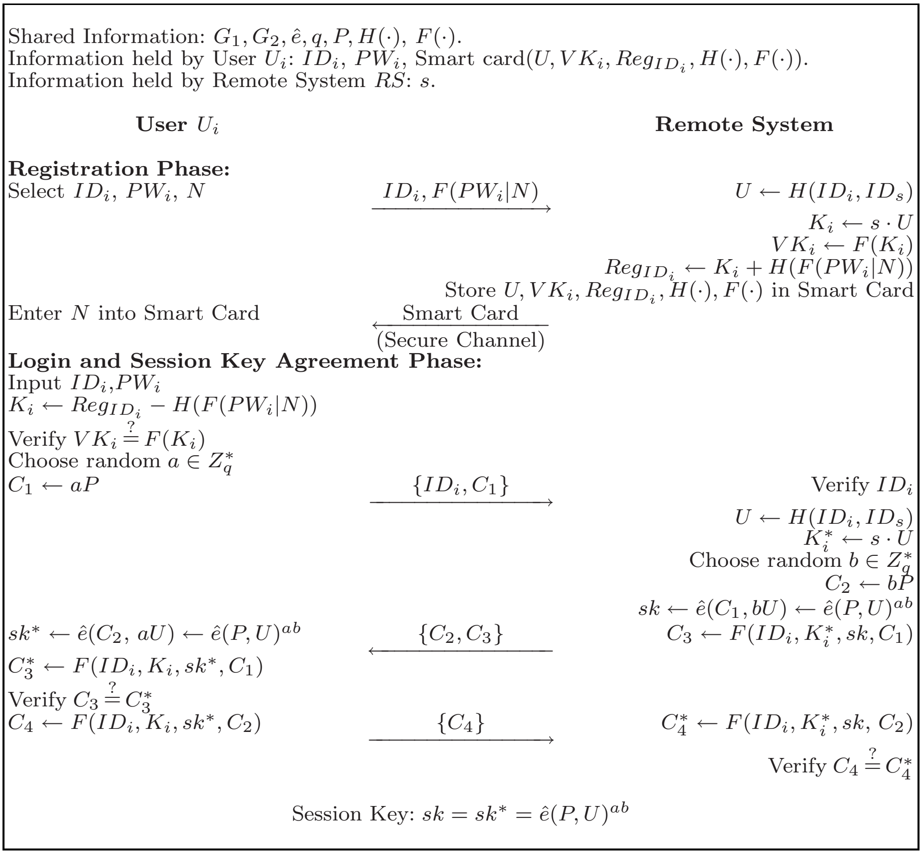

Fig. 2. Proposed Authentication Scheme

6 Security Analysis

This section provides the proof of correctness of the proposed scheme. Here, nine security properties: passive attack, active attack, guessing attack, insider attack, known-key attack, secure password change, fast wrong password detection, mutual authentication and perfect forward secrecy, would be considered for the proposed scheme [9].

(1) The proposed scheme can resist a passive attack. If an attacker, called E, who eavesdrops on a successful proposed scheme run can make a guess at the session key by using only information obtainable over a network and a guessed value of the remote system's secret key s, E could break a Bilinear Computational Diffie-Hellman Problem (BCDHP) (Definition 2 in Section Preliminaries). The reason will be clear. Such a problem can be reduced to the computing of a keying material $\hat{e}(P, U)^{ab}$ from the value C_1 and C_2 in

the scheme. Thus, we claim that it is as difficult as to break the BCDHP. Without the ability to compute the keying material $\hat{e}(P,U)^{ab}$, the messages C_3 and C_4 do not leak any information to the passive attacker. Since the user U_i and the remote system do not leak any information either, the proposed scheme can resist a passive attack.

(2) The proposed scheme can resist an active attack. Active attacks can take many different forms, depending on what information is available to the attacker. An attacker who knows the remote system's secret key s can easily pretend to be U_i and communicate with the system. Similarly, an attacker with s can masquerade as the system when U_i tries to contact him. A man-in-the middle attack, which requires an attacker to fool both sides of a legitimate conversation, cannot be carried out by an attacker who does not know the system's secret key s. For example, suppose that attacker E wants to fool the system into thinking he is talking to U_i. First, E can compute $C_1' = eP$, where e is a fresh random value, and send it to the system. Then, the system will compute $sk = \hat{e}(C_1, bU) = \hat{e}(P,U)^{ae}$, $C_2 = bP$ and $C_3 = F(ID_i, K_i^*, sk, C_1')$, and send C_2 and C_3 to E. When E receives C_2 and C_3 from the remote system, E has to make $C_4' = F(ID_i, K_i', sk', C_2)$ and send it to the system. Since the problem is combined with the BCDHP and a secure one-way hash function, in order to compute valid C_4', E cannot guess sk' or K_i' from C_3. Thus, the proposed scheme can withstand the man-in-the-middle attack.

(3) The proposed scheme can resist guessing attack. Assume a user loses his 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 Reg_{ID_i} in the smart card without knowing the system's secret key s. Since the smart card verifies computed value $F(K_i)$ with stored VK_i, an attacker can perform a password guessing attack by using stolen or lost smart card. However, in the proposed scheme, this verification process performs only three times that can withstand the attack. Therefore, no one can get a legitimate user U_i's password PW_i. Even if an attacker has $K_i = s \cdot H(ID_i)$, it is extremely hard for any attacker to derive s from $K_i = s \cdot H(ID_i)$ because of Discrete Logarithm Problem (DLP) (Definition 1 in Section Preliminaries). Therefore, the proposed scheme can withstand the guessing attack.

(4) The proposed scheme can resist insider attack. In many scenarios, the user uses a common password to access several systems for his convenience. If the user login request is password-based and the RS maintains password or verifier table for login request verification, an insider of RS could impersonate user's login by stealing password and gets access of the other systems. In the registration phase of Das et al.'s scheme, user U_i's password PW_i will be revealed to remote server RS after Step (R2). If U_i uses PW_i to access several servers for his convenience, the insider of RS can impersonate U_i to access other servers. In the proposed scheme, since U_i registers to RS by presenting $ID_i, F(PW_i|N)$ instead of ID_i, PW_i, the insider of RS cannot

directly obtain PW_i without knowing of random nonce N. Therefore, the proposed scheme can withstand the insider attack.

(5) The proposed scheme can resist the known-key attack. Known-key security means that each run of a key agreement protocol between two entities U_i and a remote system should produce unique secret keys; such keys are called session keys. If the session key sk is revealed to a passive attacker E, E does not learn any new information from combining sk with publicly-visible information. This is true because the messages C_3 or C_4 do not leak any information to the attacker. We have already established that E cannot make meaningful guesses at the session key sk from the guessed passwords, and there does not appear to be an easy way for E to carry out an off-line password guessing attack. It means that the attacker, having already obtained some past session keys, cannot compromise current or future session keys. Thus, it can resist the known-key attack.

(6) The proposed scheme provides secure password change In Das et al.'s scheme, when a smart card is stolen, an unauthorized user can easily change a new password for the card in password-change phase. First, an unauthorized user inserts U_i's smart card into the smart card reader of a terminal, enters the ID_i and PW_e, where PW_e is the unauthorized user's arbitrary password, and requests a change of passwords. Since ID_i is public value and the entered ID_i is identical to the one that is stored in the smart card, the smart card will proceed to the Step (P2) of password change phase. Next, the unauthorized user enters an arbitrary new password PW_e^* and then the smart card computes $Reg_{ID_i}^* = Reg_{ID_i} - H(PW_e) + H(PW_e^*)$, which yields $s \cdot H(ID_i) + H(PW_i) - H(PW_e) + H(PW_e^*)$, and then replaces he previously stored Reg_{ID_i} with $Reg_{ID_i}^*$ without any checking. If a malicious user stole user U_i's smart card for a short time and change an arbitrary new password as above described, then the legal user U_i's succeeding login requests will be denied unless he re-registers with the remote server again because $\hat{e}(DID_i - V_i, P) \neq \hat{e}(H(ID_i), Pub_{RS})^T$ in the verification phase. So considered, Das et al.'s password change phase is insecure. However, the proposed scheme provides secure password change. Because the smart card can verify K_i using the stored $F(K_i)$ in Step (P3) of the password change phase, when the smart card was stolen, unauthorized users cannot change the password of the card without knowing the U_i's password PW_i. Therefore, the proposed scheme provides secure password change.

(7) The proposed scheme provides fast wrong password detection In Das et al.'s scheme, if user U_i input a wrong password by mistake, this wrong password will be detected by the remote system in the authentication phase. Therefore, Das et al.'s scheme is slow to detect the user's wrong password. In contrast to Das et al.'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 $F(K_i) = VK_i$ using the stored VK_i in Step (L2) of the login phase. Therefore, the proposed scheme provides fast wrong password detection.

(8) The proposed scheme provides the mutual authentication. Mutual authentication means that both the user and remote system are authenticated to each other within the same protocol, while explicit key authentication is the property obtained when both implicit key authentication and key confirmation hold. As such, the proposed scheme uses the Diffie-Hellman key exchange algorithm in order to provide mutual authentication. Then, the key is explicitly authenticated by a mutual confirmation session key, $\hat{e}(P, U)^{ab}$.

(9) The proposed scheme provides perfect forward secrecy. Perfect forward secrecy means that if a long-term private key (e.g. user password PW_i or system's private key s) is compromised, this does not compromise any earlier session keys. In the proposed scheme, since the Diffie-Hellman key exchange algorithm is used to generate a session key $\hat{e}(P, U)^{ab}$, perfect forward secrecy is ensured because an attacker with a compromised system's secret key s is only able to obtain the aP and bP from an earlier session. In addition, it is also computationally infeasible to obtain the session key $\hat{e}(P, U)^{ab}$ from aP and bP, as it is a DLP and a BCDHP.

The security properties of Das et al.'s scheme and the proposed scheme are summarized in Table 1.

Table 1. A comparison of security properties

Security properties	Das et al.'s Scheme	Proposed Scheme
Passive attack	Secure	Secure
Active attack	Insecure	Secure
Guessing attack	Insecure	Secure
Stolen smart card attack	Insecure	Secure
Insider attack	Insecure	Secure
Secure password change	Not Provide	Provide
Mutual authentication	Not Provide	Provide
Session key distribution	Not Provide	Provide
Perfect forward secrecy	Not Provide	Provide
Wrong password detection	Slow	Fast
Timestamp	Required	Not Required

7 Conclusion

The current paper demonstrated that Das st al.'s scheme is vulnerable to an impersonation attack and an off-line password guessing attack. Furthermore, we presented an improved authentication scheme based on bilinear computational Diffie-Hellman problem and one-way hash function to the schemes, in order to isolate such problems. As a result, the proposed scheme is more secure than

Das et al.'s scheme and it provides mutual authentication between the user and remote system. In addition, the proposed scheme does not require time synchronization or delay-time limitations between the user and remote system. However, security of our protocol is not still proved formally. This is our future work.

Acknowledgements

This research was supported by the MIC of Korea, under the ITRC support program supervised by the IITA (IITA-2006-C1090-0603-0026).

References

1. Peyret, P., Lisimaque, G., Chua, T.Y.: Smart Cards Provide Very High Security and Flexibility in Subscribers Management. IEEE Transactions on Consumer Electronics. Vol. 36. No. 3. (1990) 744-752
2. Sternglass, D.: The Future Is in the PC Cards. IEEE Spectrum. Vol. 29. No. 6. (1992) 46-50
3. Joux, A.: A One Round Protocol for Tripartite Diffie-Hellman. in: Proceedings of Algorithmic Number Theory Symposium Lecture Notes in Computer Science. Vol. 1838. Springer-Verlag. Berlin. (2000) 385-394
4. Boneh, D., Franklin, M.: Identity-based Encryption from the Weil Pairing. in: Advances in Cryptology-Crypto 2001. LNCS 2139. Springer-Verlag. (2001) 213-229
5. Smart, N.P.: An Identity Based Authentication Key Agreement Protocol Based on Pairing. Electron. Lett. Vol. 38. (2002) 630-632
6. Paterson, K.G.: ID-based Signature from Pairings on Elliptic Curves. Electron. Lett. Vol. 38. No. 18. (2002) 1025-1026
7. Wen, H.A., Lee, T.F., Hwang, T.: Provably Secure Three-party Password-based Authenticated Key Exchange Protocol Using Weil Pairing. IEE Proc.-Commun. Vol. 152. No. 2. (2005) 138-143
8. Das, M. L., Saxena, A., Gulati, V.P., Phatak, D.B.: A Novel Remote User Authentication Scheme Using Bilinear Pairings. Computers & Security. Vol. 25. No. 3. (2006) 184-189
9. Menezes, A.J., Oorschot, P.C., Vanstone, S.A.: Handbook of Applied Cryptograph. CRC Press. New York. (1997)
10. Ding, Y., Horster, P.: Undetectable On-line Password Guessing Attacks. ACM Operating Systems Review. Vol. 29. No. 4. (1995) 77-86
11. Menezes, A., Okamoto, T., Vanstone, S.: Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field. IEEE Transactions on Information Theory. Vol. 39. (1993) 1639-1646
12. Frey, G., Ruck, H.: A Remark Concerning m-divisibility and the Discrete Logarithm in the Divisor Class Group of Curves. Mathematics of Computation. Vol. 62. (1994) 865-874
13. Needham, R.M., Schroeder, M.D.: Using Encryption for Authentication in Large Networks of Computers. Communications of the ACM. Vol. 21. No. 12. (1978) 993-999