ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2011, Vol. 40, No. 3

A ROBUST REMOTE USER AUTHENTICATION SCHEME USING SMART CARD

Chun-Ta Li

Department of Information Management, Tainan University of Technology
529 Zhongzheng Road, Tainan City 71002, Taiwan, R.O.C.
E-mail: th0040@mail.tut.edu.tw

Cheng-Chi Lee

Department of Library and Information Science, Fu Jen Catholic University
510 Jhongjheng Road, New Taipei City 24205, Taiwan, R.O.C. and
Department of Information and Communication Engineering, Asia University
500 Lioufeng Raod, Wufeng Shiang, Taichung, Taiwan, R.O.C.
Corresponding E-mail: cclee@mail.fju.edu.tw

crossref http://dx.doi.org/10.5755/j01.itc.40.3.632

Abstract. Remote user authentication is important to identify whether communicating parties are genuine and trustworthy using the password and the smart card between a login user and a remote server. A number of password-based authentication schemes using smart cards have been proposed in recent years. We find that two most recent password-based authentication schemes (Hsiang and Shih 2009, Chen and Huang 2010) assume that the attacker cannot extract the secret information of the smart card. However, in reality, the authors in (Kocher et al. 1999 and Messerges et al. 2002) show that the secrets stored in the card can be extracted by monitoring its power consumption. Therefore, these schemes fail to resist smart card security breach. As the main contribution of this paper, a robust remote user authentication scheme against smart card security breach is presented, while keeping the merits of the well-known smart card based authentication schemes.

Key words: Cryptanalysis, Network security, Smart card, User authentication.

1. Introduction

With the significant advances in communication networks over the last couple of decades, remote user authentication based on passwords [7, 10, 14, 17] or biometrics [15, 16] over insecure networks is the conventional method of authentication and has already been accepted warmly. Typically a network of remote servers are responsible for managing and supplying network services to login users for which user authentication protocols have been provided during a login procedure.

In 1981, Lamport [14] first proposed a remote user authentication scheme, remote server maintains verification tables and authenticates the validity of a login user by verifying his/her identity and password. However, it is ineffective for remote server to maintain the verification tables due to the size of the verification tables are proportional to the number of login users. Therefore, in 1998, Jan and Chen [10] proposed a password authentication scheme without maintaining verification tables in the system. In 2000, Hwang and Li [7] proposed a remote authentication scheme using smart cards based on ElGamal's [6] public-key cryptosystem, remote server only keeping one secret key and no maintained verification tables required. Without storing verification tables in servers, it may prevent possible stolen-verifier problems. However, in contrast to a hashing function based cryptography such as SHA-1 [19], the main disadvantage of a public-key based cryptography is high computational cost and more inefficient to implement. Due to the power constraint of the smart cards, Sun [23] proposed an enhanced version of Hwang-Li's authentication scheme based on use of a lightweight hashing function. However, in Sun's scheme, a login user cannot freely choose the password he/she wants. That is, the design in Sun's scheme is actually unrealistic and lack of user friendliness. In 2002, Chien et al. proposed a friendly user authentication scheme [5], allowing users to freely choose passwords and providing mutual authentication between login user and remote server. However, in 2004, Hsu pointed out that Chien et al.'s scheme is vulnerable to parallel session attacks [9] and the attacker can replay a legal user's previous login message to pass server's authentication. In 2005, Chen and Yeh proposed an efficient nonce-based authentication scheme with session key agreement [2]. In contrast to timestamp-based approaches, nonce-based approaches prevent serious time synchronization problems. Moreover, in Chen-Yeh's authentication scheme, a common session key will be agreed for securing later communications after the authentication phase. In 2008, Juang et al. [11] proposed a user authentication and key agreement scheme using smart cards based on elliptic curve cryptosystems [12] and can prevent insider attack. However, Sun et al. [24] found that Juang et al.'s scheme is vulnerable to session-key problem, inability of password-update operation and inefficiency of double secret keys.

Recently, Liao et al. [18] proposed nine requirements for rating performance of a new password authentication scheme in terms of security, friendliness and efficiency. A new password authentication scheme using smart cards should satisfy the following requirements: (1) without maintaining verification tables; (2) users can freely choose and update passwords; (3) resistance to password disclosure to the server; (4) prevention of masquerade attacks; (5) resistance to replay, modification, parallel session and stolen-verifier attacks; (6) a easy-to-remember password; (7) low communication cost and computation complexity; (8) achieve mutual authentication between login users and remote servers; (9) resistance to guessing attacks even if the smart card is lost or stolen by attackers. Besides requirements stated in reference [18], we list four additional requirements to solve all problems in smart card-based authentication schemes, including: (10) session key agreement; (11) revocation of smart card; (12) resistance to insider attacks; (13) prevention of smart card security breach attacks. For Requirement (13), it is important to note that secret information stored in a smart card can be extracted by analyzing and monitoring its power consumption [13, 20]. Obviously, if a legal user's smart is lost and it is picked up by a malicious attacker or an attacker steals user's smart card, the user's sensitive password may be derived out by an attacker. After that, there is no way to prevent the attacker from masquerading as the legal user. To the best of our knowledge, we find that until now no smart-card-oriented authentication schemes can resist smart card security breach attacks. In this paper, we focus on the security of password authentication schemes for the merit that the design scheme achieves Requirement (13) and we will propose a robust remote user authentication scheme with better security strength while keeping the above-mentioned requirements.

The remainder of the paper is organized as follows. Section 2 is a brief review of one related authentication scheme and we analyze two schemes to show their security weaknesses in Section 3. The new remote user authentication scheme against smart card security breach is proposed in Section 4. Our proposed scheme is compared with other related works in Section 5 and Section 6 concludes the paper.

2. Review of Related Works

A number of password-based remote authentication scheme using smart cards have been proposed in recent years. In this section, we review two most recent password-based remote authentication schemes [4, 8]. For convenience of description, we will list the common notations used throughout this paper in Table 1. In the very beginning of two remote authentication schemes, the login user U_i chooses his/her identity ID_i and password PW_i, and the remote server S holds a master secret key X, which is kept secret and only known by the server.

2.1. A Review of Hsiang and Shih's scheme

Hsiang and Shih proposed an improved version of Yoon et al.'s remote user authentication scheme [25] and Hsiang and Shih's scheme [8] is composed of three phases, registration, authentication and password update.

2.1.1. Registration Phase

The registration phase consists of two steps:

(R.1) $U_i \Longrightarrow S : ID_i, H(PW_i), H(b \oplus PW_i)$
User U_i computes $H(b \oplus PW_i)$ and submits $\{ID_i, H(PW_i), H(b \oplus PW_i)\}$ to the remote authentication server S, where b is a random number.

(R.2) $S \Longrightarrow SC_i : C_1, H(\cdot)$
If it is U_i's initial registration, the server S creates an entry for U_i and records $N = 0$ in the account database. Otherwise, S sets $N = N + 1$ in U_i's entry. Moreover, S computes $P = H(EID \oplus X)$, $R = P \oplus H(b \oplus PW_i)$, and $V = H(P \oplus H(PW_i))$ and stores V, R, and $H(\cdot)$ into the smart card SC_i, where $EID = (ID||N)$. Finally, S releases SC_i to U_i and the registration phase is completed after U_i stores b into SC_i.

2.1.2. Authentication Phase

When U_i wants to login to the server, U_i performs the following operations and the authentication phase consists of two steps.

(A.1) $SC_i \longrightarrow S : ID_i, T_{U_i}, C_2$
The user U_i enters ID_i and PW_i and the smart card SC_i computes $C_1 = R \oplus H(b \oplus PW_i)$ and $C_2 = H(C_1 \oplus T_{U_i})$, where T_{U_i} is the current timestamp generated by U_i. Then, SC_i submits $\{ID_i, T_{U_i}, C_2\}$ to the server.

(A.2) $S \longrightarrow SC_i : T_S, C_3$
Upon receiving the login request, S verifies the validity of T_{U_i}. If it is invalid, S rejects

Table 1. Notations used throughout this paper

U_i	The login user
PW_i	The password of U_i
ID_i	The identity of U_i
SC_i	The smart card of U_i
S	The remote server
X	The master secret key stored in S
N	The number of times U_i re-registers to S
T_i	The current timestamp generated by an entity i
RN_i	The random number generated by an entity i
SK	The common session key
$\oplus$	The bitwise XOR operation
$H(\cdot)$	A collision free one-way hash function
$\|\|$	String concatenation
$E_K(.)/D_K(.)$	The symmetric encryption/decryption function with key K
$\Longrightarrow$	A secure channel
$\longrightarrow$	A public channel

U_i's login request; otherwise, S checks if the hashed value $H(H(EID \oplus X) \oplus T_{U_i})$ is equal to received C_2. If it does not hold, SC_i terminates communication; otherwise, S succeeds to authenticate U_i and submits T_S and $C_3 = H(H(EID \oplus X) \oplus H(T_S))$ to SC_i, where T_S is the current timestamp generated by S. Upon receiving the message from S, SC_i verifies the validity of T_S. If it is invalid, U_i terminates communication; otherwise, U_i checks if the hashed value $H(C_1 \oplus H(T_S))$ is equal to received C_3. If it holds, U_i succeeds to authenticate the remote server S.

2.1.3. Password Update Phase

In this phase, U_i enters his/her identity ID_i, the original password PW_i, and the new password PW_i'. Next, SC_i computes $P' = R \oplus H(b \oplus PW_i)$ and $V' = H(P' \oplus H(PW_i))$ and checks if the value V' is equal to stored V. If it does not hold, SC_i rejects U_i's password update request; otherwise, SC_i computes $R_{new} = P' \oplus H(b \oplus PW_i')$ and $V_{new} = H(P' \oplus H(PW_i'))$ and replaces (R, V) with (R_{new}, V_{new}).

2.2. A Review of Chen and Huang's scheme

In 2010, Chen and Huang proposed a user-participation-based authentication scheme [4] that benefits from combining CAPTCHA (Completely Automated Public Turing Test to Tell Computers and Humans Apart) [22] and visual secret sharing [21]. Chen and Huang's authentication scheme is composed of three phases, registration, authentication and password update.

2.2.1. Registration Phase

This phase is to issue a smart card SC_i to the login user U_i and the transmitted messages in this phase is over a secure channel. The registration phase consists of two steps:

(R.1) $U_i \Longrightarrow S : ID_i, H(PW_i \oplus n)$
User U_i computes $H(PW_i \oplus n)$ and submits ID_i and $H(PW_i \oplus n)$ to the remote authentication server S, where n is an initial nonce.

(R.2) $S \Longrightarrow SC_i : C_1, H(\cdot)$
The server S computes $C_1 = H(ID_i \oplus X) \oplus H(PW_i \oplus n)$ and stores C_1 and $H(\cdot)$ into the smart card SC_i. Then, S releases SC_i to U_i and the registration phase is completed after U_i stores n into SC_i.

2.2.2. Authentication Phase

If user U_i wants to login to the server, U_i should first insert his/her SC_i into a card reader and enters the identity ID_i and password PW_i. Then, SC_i should do the following operations and the authentication phase consists of three steps.

(A.1) $SC_i \longrightarrow S : ID_i, E_{C_2 \oplus R_1}(S_1), R_1, H(S_1||R_1||C_2)$

User U_i generates a visual secret sharing image S_1 and a random number R_1 and computes $C_2 = C_1 \oplus H(PW_i \oplus n)$, $H(S_1||R_1||C_2)$ and $E_{C_2 \oplus R_1}(S_1)$. Then, SC_i submits $\{ID_i, E_{C_2 \oplus R_1}(S_1), R_1, H(S_1||R_1||C_2)\}$ to the server.

(A.2) $S \longrightarrow SC_i : E_{H(ID_i \oplus X) \oplus R_2}(S_2), R_2, H(S_2||R_2||C_2)$

Upon receiving the login request, S computes symmetric key $H(ID_i \oplus X) \oplus R_1$ to derive S_1' by decrypting $E_{C_2 \oplus R_1}(S_1)$ and checks if the hashed value $H(S_1'||R_1||C_2)$ is equal to received $H(S_1||R_1||C_2)$. If it does not hold, the login request is rejected; otherwise, S generates a CAPTCHA image $IMG_m = CAPTCHA(m)$ and computes $S_2 = VC(S_1', IMG_m)$, where IMG_m is a CAPTCHA image with visually recognizable distorted digits or characters of a random message m, CAPTCHA($\cdot$) is a function to generate CAPTCHA image with input of a set of digits or characters, and $VC(\cdot,\cdot)$ is a visual secret sharing function used to derive another sharing image with the input of a secret image and a predefined share image. Moreover, S generates a random number R_2 and computes $E_{H(ID_i \oplus X)\oplus R_2}(S_2)$ and $H(S_2||R_2||C_2)$. Finally, S sends $\{E_{H(ID_i \oplus X)\oplus R_2}(S_2), R_2, H(S_2||R_2||C_2)\}$ to SC_i.

(A.3) $SC_i \longrightarrow S : m'$

Upon receiving the message from S, SC_i computes symmetric key $C_2 \oplus R_2$ to derive S_2' by decrypting $E_{H(ID_i \oplus X)\oplus R_2}(S_2)$ and checks if the hashed value $H(S_2'||R_2||C_2)$ is equal to received $H(S_2||R_2||C_2)$. If it does not hold, SC_i terminates communication; otherwise, SC_i recognizes the message m' appearing on the stacked image by superimposing S_1 and S_2' and submits m' to S. Finally, S verifies if received m' is equal to m or not. If it holds, S succeeds to authenticate the user U_i.

2.2.3. Password Update Phase

In this phase, the user U_i can change the original password PW_i to the new password PW_i' any time he/she wants. First, U_i enters ID_i, PW_i and new password PW_i' and SC_i computes $H(PW_i \oplus n)$, $H(PW_i' \oplus n)$ and $C_1' = C_1 \oplus H(PW_i \oplus n) \oplus H(PW_i' \oplus n)$ and replaces C_1 with C_1'.

3. The Various Kinds of Attacks with Smart Card Security Breach

In this section, we show some attacks with smart card security breach in two authentication schemes [4, 8] and we assume that an attacker U_A may have the capabilities shown in Table 2. Let us consider the following scenarios. If a user's smart card is lost and it is picked up by an attacker or an attacker steals user's smart card. The secrets stored in the smart card can be extracted by monitoring its power consumption [13, 20], then the attacker can easily derive user's password and masquerade as a legitimate user.

3.1. Off-line Password Guessing Attack

3.1.1. Cryptanalysis of Hsiang and Shih's Scheme

Similarly, according to C-2 (ii) of Table 2, in Hsiang and Shih's scheme [8], the attacker U_A can breach the secrets $R = P \oplus H(b \oplus PW_i)$, $V = H(P \oplus H(PW_i))$, b and $H(\cdot)$ are stored in the smart card and use the breached secrets R, V, b and $H(\cdot)$ to perform the following steps:

Step 1. Select a guessed password PW_i^*.

Step 2. Compute $H(b \oplus PW_i^*)$

Step 3. Compute $P' = R \oplus H(b \oplus PW_i^*)$ and $V' = H(P' \oplus H(PW_i^*))$.

Step 4. Compare V to V'.

A match in Step 4 above indicates the correct guess of user's password. Therefore, U_A succeeds to guess the low-entropy password PW_i and Hsiang and Shih's scheme is also vulnerable to off-line password guessing attack.

3.1.2. Cryptanalysis of Chen and Huang's Scheme

According to C-3 of Table 2, in Chen and Huang's scheme [4], the attacker U_A breaches the secrets $C_1 = H(ID_i \oplus X) \oplus H(PW_i \oplus n)$, n and $H(\cdot)$ are stored in the smart card and eavesdrops the messages $E_{C_2 \oplus R_1}(S_1)$, R_1 and $H(S_1||R_1||C_2)$. Then, the attacker can use the breached secrets C_1, n and $H(\cdot)$ and the eavesdropped messages $E_{C_2 \oplus R_1}(S_1)$, R_1 and $H(S_1||R_1||C_2)$ to perform the following steps:

Step 1. Select a guessed password PW_i^*.

Step 2. Compute $H(PW_i^* \oplus n)$

Step 3. Compute $C_2' = C_1 \oplus H(PW_i^* \oplus n)$ and $C_2' \oplus R_1$.

Step 4. Compute $S_1' = D_{C_2' \oplus R_1}(E_{C_2 \oplus R_1}(S_1))$ and $H(S_1'||R_1||C_2')$.

Step 5. Compare $H(S_1||R_1||C_2)$ to $H(S_1'||R_1||C_2')$.

A match in Step 5 above indicates the correct guess of user's password. Therefore, the attacker U_A succeeds to guess the low-entropy password PW_i and Chen and Huang's scheme cannot resist off-line password guessing attack.

3.2. Masquerading Attack

Once the attacker U_A has correctly derived the user's password PW_i, he/she can also use the stored information on the stolen or lost smart card to forge a valid login request to masquerade as a legal user.

Table 2. An attacker has the following capabilities of doing security attacks

C-1	The attacker U_A has total control over the communication channel between U_i and S such as eavesdropping, intercepting, inserting, deleting, and modifying any transmitted messages in the public channel.
C-2	The attacker may either (i) obtain U_i's PW_i, or (ii) extract the secret information of the smart card, but cannot achieve both (i) and (ii).
C-3	We focus on the security of authentication schemes for the case that the attacker has Capabilities C-1 and C-2 (ii).

3.2.1. Cryptanalysis of Hsiang and Shih's Scheme

Similarly, during the authentication phase of Hsiang and Shih's scheme, the attacker U_A can use the information on the lost or stolen smart card to make a valid login request to masquerade as a legal user with ease. For example, U_A is able to compute $C_1^* = R \oplus H(b \oplus PW_i^*)$ and $C_2^* = H(C_1^* \oplus T_{U_A})$ by using the current timestamp T_{U_A} and the derived password PW_i^* on the lost or stolen smart card. Hence, U_A can successful make a valid login request message to masquerade as a legal user U_i by sending $\{ID_i, T_{U_A}, C_2^*\}$ to the server S.

3.2.2. Cryptanalysis of Chen and Huang's Scheme

The detailed description of this attack is as follows. Firstly, during the authentication phase of Chen and Huang's scheme, the attacker U_A generates a visual secret sharing image S_1^* and a random number R_1^*. Then, U_A can use the information on the lost or stolen smart card to make a valid login request to masquerade as a legal user with ease. For example, U_A is able to compute $C_2^* = C_1 \oplus H(PW_i^* \oplus n)$, $H(S_1^*||R_1^*||C_2^*)$ and $E_{C_2^* \oplus R_1^*}(S_1^*)$ by using the derived password PW_i^* on the lost or stolen smart card. Thus, U_A can successful make a valid login request message to masquerade as a legal user U_i by sending $\{ID_i, E_{C_2^* \oplus R_1^*}(S_1^*), R_1^*, H(S_1^*||R_1^*||C_2^*)\}$ to the server S.

4. The Proposed Scheme

In this section, we describe a robust remote user authentication scheme which resolves all the above security flaws of smart card security breach. In general, some remote user authentication schemes consists of registration, authentication, and password update phase. In our proposed we divide the authentication phase into the login and verification phase. Thus, our scheme consists of registration, login, verification, password update, and smart card revocation phases. Figure 1 shows the entire protocol structure of the new user authentication scheme.

4.1. The Registration Phase

(R.1) $U_i \Longrightarrow S : ID_i, H(H(PW_i \oplus RN_1))$
To register, the user U_i chooses his/her identity ID_i and password PW_i and generates a random number RN_1. Then, U_i computes $H(H(PW_i \oplus RN_1))$ and sends ID_i and $H(H(PW_i \oplus RN_1))$ over a secure communication channel to S.

(R.2) $S \Longrightarrow SC_i : ID_i, C_1, H(\cdot)$
Upon receiving ID_i and $H(H(PW_i \oplus RN_1))$, S maintains a account table (AT) for a registration service and the format of AT is shown as follows:

User identity	Registration times	Verification parameter
ID_i	$N = 0$	$H(H(PW_i \oplus RN_1))$

where the 1st field of AT records the user's identity, the 2nd field of AT records $N = 0$ if it is U_i's initial registration, otherwise, S sets $N = N + 1$ in the existing field for U_i, and the 3rd field records U_i's verification parameter $H(H(PW_i \oplus RN_1))$ for a later login request.

Finally, S computes $C_1 = H(ID_i||X||N) \oplus H(H(PW_i \oplus RN_1))$ and stores $\{ID_i, C_1, H(\cdot)\}$ into the smart card SC_i and releases it to U_i.

(R.3) $U_i \Longrightarrow SC_i : ID_i, C_1, H(\cdot), RN_1$
Upon receiving SC_i, U_i stores RN_1 into SC_i and U_i finishes the registration procedure. Note that U_i's SC_i contains $\{ID_i, C_1, H(\cdot), RN_1\}$ and U_i does not need to remember RN_1 after finishing this phase. Note that the bit length of random numbers RN_i and S's master secret key X are assumed to be 256. That is, RN_i and X are two high entropy random numbers.

4.2. The Login Phase

When U_i wants to login S, the following operations will perform:

(L.1) $U_i \Longrightarrow SC_i : ID_i, PW_i, RN_2$
U_i inserts his/her SC_i into the smart card reader

User		Server
Registration Phase:		
Select ID_i, PW_i, RN_1		
	$\{ID_i, H(H(PW_i \oplus RN_1))\}$ $\longrightarrow$	
		$C_1 = H(ID_i\|\|X\|\|N) \oplus H(H(PW_i \oplus RN_1))$
		Store $C_1, ID_i, H(\cdot)$ into SC_i
	$\{SC_i\}$ $\longleftarrow$	
Store RN_1 into SC_i		
Login Phase:		
Enter ID_i, PW_i, RN_2		
$C_2 = H(PW_i \oplus RN_1)$		
$C_3 = C_1 \oplus H(C_2)$		
$C_4 = C_3 \oplus C_2$		
$C_6 = E_{K_{U_i}}(C_5, RC)$		
	$\{ID_i, C_4, C_6\}$ $\longrightarrow$	
Verification Phase:		
		Verify ID_i
		$C_7 = H(ID_i\|\|X\|\|N)$
		$C_8 = C_4 \oplus C_7$
		$C_9 = H(C_8)$
		Abort if $C_9 \neq H(H(PW_i \oplus RN_1))$
		$(C_5, RC) = D'_{K_{U_i}}(C_6)$
	$\{E_{K_S}(RC, RS, C_5)\}$ $\longleftarrow$	
$(RC, RS, C_5) = D'_{K_S}(E_{K_S}(RC, RS, C_5))$		
Verify RC and C_5		
Replace (RN_1, C_1) with $(RN_2, C_3 \oplus C_5)$		
	$\{H(RS)\}$ $\longrightarrow$	
		Verify $H(RS)$
Compute $SK = H(RC \oplus RS)$		Compute $SK = H(RC \oplus RS)$
Password Update Phase:		
Enter ID_i, PW_i, PW'_i, RN_3		
$C_2 = H(PW_i \oplus RN_2)$		
$C_3 = C_1 \oplus H(C_2)$		
$C_4 = C_3 \oplus C_2$		
$C'_5 = H(H(PW'_i \oplus RN_3))$		
$C_6 = E_{K_{U_i}}(C'_5, RC)$		
	$\{ID_i, C_4, C_6\}$ $\longrightarrow$	
		Verify ID_i
		Abort if $C_9 \neq H(H(PW_i \oplus RN_2))$
		$C'_5 = H(H(PW'_i \oplus RN_3))$
Replace (RN_2, C_1) with $(RN_3, C_3 \oplus C'_5)$		

Fig. 1. Proposed user authentication scheme

and enters ID_i, PW_i and a new random number RN_2, where RN_2 is used for next login request. Then, SC_i generates a random number RC and computes $C_2 = H(PW_i \oplus RN_1)$, $C_3 = C_1 \oplus H(C_2)$, $C_4 = C_3 \oplus C_2$, and $C_6 = E_{K_{U_i}}(C_5, RC)$, where $C_5 = H(H(PW_i \oplus RN_2))$ and $K_{U_i} = H(C_2||C_3)$.

(L.2) $SC_i \longrightarrow S : ID_i, C_4, C_6$
SC_i sends $\{ID_i, C_4, C_6\}$ over a public communication channel to the remote server S.

4.3. The Verification Phase

Upon receiving the login request from U_i, the remote server S and the smart card SC_i performs the following operations:

(V.1) $S \longrightarrow SC_i : E_{K_S}(RC, RS, C_5)$
If ID_i is invalid, S rejects U_i's login request. Otherwise, S computes $C_7 = H(ID_i||X||N)$, $C_8 = C_4 \oplus C_7$, and $C_9 = H(C_8)$ and compares the third entry $H(H(PW_i \oplus RN_1))$ to the computed C_9. If equal, S successfully authenticates U_i and computes symmetric key $K'_{U_i} = H(C_8||C_7)$, which equals to $K_{U_i} = H(C_2||C_3)$, to obtain (C_5, RC) by decrypting $D'_{K_{U_i}}(C_6)$. Then, S replaces the third entry $H(H(PW_i \oplus RN_1))$ with $C_5 = H(H(PW_i \oplus RN_2))$ and sends $E_{K_S}(RC, RS, C_5)$ over a public communication channel to the smart card SC_i, where RS is a random number generated by S and $K_S = H(C_7||C_8)$. Finally, the format of AT is shown as follows:

User identity	Registration times	Verification parameter
ID_i	$N = 0$	$H(H(PW_i \oplus RN_2))$

(V.2) $SC_i \longrightarrow S : H(RS)$
Upon receiving the message from S, SC_i computes symmetric key $K'_S = H(C_3||C_2)$, which equals to $K_S = H(C_7||C_8)$, to obtain (RC, RS, C_5) by decrypting $D'_{K_S}(E_{K_S}(RC, RS, C_5))$. Then, SC_i verifies if generated (RC, C_5) equals received (RC, C_5). If not equivalent, SC_i terminates communication; otherwise, SC_i now successfully authenticates S and replaces original RN_1 and C_1 with new RN_2 and $C_3 \oplus C_5$, respectively. Finally, SC_i sends a response $H(RS)$ to S and S can make sure that it is communicating with a legitimate U_i. Note that both U_i and S can compute the agreed session key $SK = H(RC \oplus RS)$ for securing future communications.

4.4. Password Update Phase

This phase is extremely similar to the login and verification phases of the proposed scheme and U_i is strongly recommended not to use any previous parameters for his/her update request, e.g. random number RN_2. When a user U_i wants to update his/her password PW_i with a new password PW'_i, U_i inserts his/her SC_i into the smart card and enters his/her ID_i, the original password PW_i, the new password PW'_i, and a new random number RN_3. Then, SC_i computes $C_2 = H(PW_i \oplus RN_2)$, $C_3 = C_1 \oplus H(C_2)$, $C_4 = C_3 \oplus C_2$, and $C_6 = E_{K_{U_i}}(C'_5, RC)$, where $C'_5 = H(H(PW'_i \oplus RN_3))$ and $K_{U_i} = H(C_2||C_3)$. Finally, SC_i sends $\{ID_i, C_4, C_6\}$ over a public communication channel to the remote server S. Upon receiving the message, S performs Step (V.1) and finally the format of AT is shown as follows:

User identity	Registration times	Verification parameter
ID_i	$N = 0$	$C'_5 = H(H(PW'_i \oplus RN_3))$

Note that the new password PW'_i and the new random number RN_3 stored in S's AT are simultaneous updated. Moreover, SC_i replaces original RN_2 and C_1 with new RN_3 and $C_3 \oplus C'_5$, respectively. Now, the new password PW'_i and the new random number RN_3 are successfully updated and this phase is terminated.

4.5. Smart Card Revocation Phase

In case of stolen or lost smart card, invalid user may impersonate a legal registered user to login into the remote server by using the login parameters stored in the stolen or lost smart card. As a result, there should be provision in the scheme for revoking the illegal use of stolen or lost smart card. U_i notified the remote server of the revocation and S verified the validity of U_i by checking his/her personal credentials, e.g. national ID card. If U_i is legal, U_i generates a new password PW_i" and a new random number RN_4 and performs the same steps of the registration phase. That is, the value of N is incremented by one and the format of AT is shown as follows:

User identity	Registration times	Verification parameter
ID_i	$N = N + 1$	$H(H(PW_i" \oplus RN_4))$

Note that U_i can re-register to S without changing his/her ID_i. Finally, S releases the new smart card SC'_i to U_i and U_i stores RN_4 into SC'_i and this phase is finished.

5. Analysis of The Proposed Scheme

The proposed authentication scheme benefits from the protection of smart cards to prevent the secret information for an attacker to steal and guess the real secrets stored in the stolen smart card or in the exchange of authentication messages. In the following subsections, we give an in-depth analysis of the proposed scheme in terms of security and functionality properties.

5.1. Security Analysis

Proposition 1. *The present scheme is secure against off-line password guessing attack with smart card security breach.*

Proof. With the assumption that the attacker has Capabilities C-1 - having collecting the transmitted messages $\{ID_i, C_4 = H(ID_i||X||N) \oplus H(PW_i||RN_i)$, $C_6 = E_{K_{U_i}}(H(H(PW_i \oplus RN_{i+1}))), E_{K_S}(RC, RS,$ $H(H(PW_i \oplus RN_{i+1}))), H(RS)\}$ and C-2 (ii) - having extracted the secrets $\{ID_i, C_1 = H(ID_i||X||N) \oplus$ $H(H(PW_i \oplus RN_{i+1})), H(\cdot), RN_{i+1}\}$ stored in the lost or stolen smart card, where $i = 1, 2, 3, \ldots$, $K_{U_i} = H(H(PW_i \oplus RN_i))||H(ID_i||X||N))$ and $K_S = H(H(ID_i||X||N)||H(PW_i||RN_i))$.

Throughout the proposed scheme, U_i's password PW_i makes four appearances as $C_4 = H(ID_i||X||N)$ $\oplus H(PW_i||RN_i), C_6 = E_{K_{U_i}}(H(H(PW_i \oplus RN_{i+1})))$, $E_{K_S}(RC, RS, H(H(PW_i \oplus RN_{i+1})))$ and $C_1 =$ $H(ID_i||X||N) \oplus H(H(PW_i \oplus RN_{i+1}))$. However, for each new login request, the previous random number RN_i stored in the smart card have to be replaced with new random number RN_{i+1}. Therefore, an attacker cannot launch off-line password guessing attack without knowing the previous secret RN_i and our proposed authentication scheme can resist off-line password guessing attack with smart card security breach. □

Proposition 2. *The proposed scheme can withstand masquerade attack with smart card security breach.*

Proof. As is described in Table 2 of Section 3, let us assume an attacker U_A has extracted smart card's secrets and has got the transmitted messages between U_i and S. U_A inserts U_i's SC_i into the card reader and then enters the guessing password PW_i^* and a random number RN_i^*. As described above, throughout the proposed scheme, if any trial value of the password is used, U_A has only one chance to guess the original password to pass server's validation. Once U_A's guessing password is wrong, the server can immediately detect the validity of fake login request and terminate U_A's login session. In this case, U_A cannot masquerade as a legal user to send a valid login request message and the masquerade attack cannot work in the proposed scheme. □

Proposition 3. *The proposed scheme is able to provide mutual authentication and a agreed session key between U_i and S in every login session.*

Proof. By the proposed scheme, let us assume that A and B be the two communication parties, namely the login user and the remote server. Let $A \xleftrightarrow{SK} B$ denotes the agreed session key SK shared between A and B. Hence, the mutual authentication is achieved between A and B if there exists a session key SK, then A would believe $A \xleftrightarrow{SK} B$, and B would believe $A \xleftrightarrow{SK} B$. As a result, we have stated that a strong mutual authentication should satisfy the following equations:

$$A \ \ believes\ \ B \ \ believes\ \ A \xleftrightarrow{SK} B. \quad (1)$$

$$B \ \ believes\ \ A \ \ believes\ \ A \xleftrightarrow{SK} B. \quad (2)$$

In Step (L.2) of the login phase, after B receives the login request $\{A, C_4 = H(A||X||N) \oplus$ $H(PW_A \oplus RN_i), C_6 = E_{K_A}(H(H(PW_A \oplus RN_{i+1}))$, $RC)\}$, B will verify $H(PW_A \oplus RN_i)$ by computing $C_4 \oplus H(A||X||N)$ and check whether the hashed value $H(C_4 \oplus H(A||X||N))$ is equal to $H(H(PW_A \oplus RN_i))$. If it holds, B could decrypt C_6 and gets RC in Step (V.1) of the verification phase. B then generates RS and submits $E_{K_S}(RC, RS, C_5 = H(H(PW_A \oplus RN_{i+1}))$ to A. After A receives the response message, A will verify $H(H(PW_A \oplus RN_{i+1}))$ and RC by computing $D_{H(H(A||X||N))||H(PW_A \oplus RN_i))}(E_{K_S}(RC, RS, C_5 =$ $H(H(PW_A \oplus RN_{i+1})))$. If these values are valid, A then computes the session key $SK = H(RC \oplus RS)$ and believes $A \xleftrightarrow{SK} B$. Since RC is chosen by A, A believes B believes $A \xleftrightarrow{SK} B$.

Also, in Step (V.2) of the verification phase, a response $H(RS)$ will be sent to B. After B received the response message from A, B use RS to compute $H(RS)$ and check whether the hashed value contains a response RS. If it holds, B believes $A \xleftrightarrow{SK} B$. Since RS is chosen by B, B believes A believes $A \xleftrightarrow{SK} B$. Finally, after Equations (1) and (2) are satisfied, and together they accomplish the mutual authentication and dynamic session key agreement in the proposed scheme. □

Proposition 4. *The proposed scheme is safe against the stolen-verifier and insider attack.*

Proof. In order to verify the validity of login user, the server may maintain some verification or password tables in its database. However, the server is always targets of malicious attacks and the malicious insider U_A may obtain the server's verification table. Thus, U_A can try to impersonate the legal user to access any remote server. In the registration phase of the proposed scheme, a legal user U_i only sends his/her identity ID_i and $H(H(PW_i \oplus RN_i))$ to S, that is, PW_i will not be revealed to S. Besides, in the login phase, a user is required to submit $C_4 = H(ID_i||X||N) \oplus H(PW_i \oplus RN_i)$ with his/her own known secret $H(PW_i \oplus RN_i)$. Moreover, the value of $H(ID_i||X||N)$ is only known by U_i and S. Therefore, due to the property of one-way hashing function it is computationally infeasible for U_A to compute $H(PW_i \oplus RN_i)$ given the stolen values of ID_i and $H(H(PW_i \oplus RN_i))$. Since U_A cannot invert U_i's secret $H(PW_i \oplus RN_i)$ from the stolen $H(H(PW_i \oplus RN_i))$, our scheme can withstand the stolen-verifier and insider attack. □

Proposition 5. *The proposed scheme can resist the server secret key guessing attack.*

Proof. An legal user U_i may try to derive the server's master secret key X from $C_3 = C_1 \oplus H(H(PW_i \oplus RN_i)) = H(ID_i||X||N)$. In fact, due to it is computationally infeasible to invert the one-way hashing function and S's secret key is assumed to be 256 bits, this attack will fail and any kind of server secret key guessing attack will not occur in our proposed scheme. □

5.2. Functionality Analysis

In this subsection, the functionality comparisons among related authentication schemes and our proposed scheme are summarized in Table 3. From Table 3, it is clear that our proposed scheme satisfies all the security and functionality requirements, with particular focus on smart card security breach. To the best of our knowledge, until now no smart card based user authentication scheme that can simultaneously resist smart card security breach and eavesdropping attack has been proposed. Among these vulnerabilities, these schemes [1–5, 8, 18, 23] are insecure against the password guessing and masquerade attacks with smart card security breach. After the attacker collects the login request message during one of the past sessions of a login user, with the secrets stored in the stolen smart card, the attacker can derive the user's password by launching an off-line guessing attack. Moreover, the attacker can use the derived password to masquerade as legal user to access the resources of the remote server. We observe that the key point to prevent proposed weakness is to provide user password verification in server side not in user side. Thus, the attacker cannot use the secret information stored in the stolen smart card to launch off-line password-guessing attacks. Once the password verification is done by the remote server, the remote server must maintain some secrets for verifying the validity of login user, and this design may suffer from stolen-verifier or insider attacks. The key of attacking succeed is what kind of information revealed from the stolen verification table. In the proposed scheme, user password PW_i is mixed with a random number RN_i to form the verifier $V = H(H(PW_i \oplus RN_i))$ and then stored in the server side. If the malicious insider is able to steal the verifier V from the server, he/she still cannot derive PW_i, RN_i and $H(PW_i \oplus RN_i)$ from V, in which is mainly based on the protection of one-way property of $H(\cdot)$.

On the other hand, in case of a legal user's smart card is lost or stolen by the attacker, he/she can notify the remote server to revoke his/her smart card. In the proposed scheme, it provides a functionality for revoking the illegal use of lost or stolen smart card.

6. Conclusions

This paper proposed a robust user authentication scheme using smart cards. We have showed that the proposed scheme avoids smart card security breach attacks and maintains the merits of related works such as provision of mutual authentication, prevention of password guessing attack, detection of masquerade attack, session key agreement, and so on. In our future works, a formal security proof and a experimental simulation would have been a better picture to demonstrate the feasibility of the proposed scheme and the proposed scheme can be further extended with the countermeasure against the Denial-of-Service (DoS) attacks.

Acknowledgment

The authors would like to express their appreciation to the anonymous referees for their valuable suggestions and comments. This research was partially supported by the National Science Council, Taiwan, R.O.C., under contracts no: NSC 99-2221-E-165-001 and NSC 99-2221-E-030-022.

References

[1] T. H. Chen, W. B. Lee, and G. Horng, "Secure SAS-like password authentication schemes," *Computer Standards and Intefaces*, 27(1):25–31, 2004.

[2] Y. C. Chen and L. Y. Yeh, "An efficient nonce-based authentication scheme with key agreement," *Applied Mathematics and Computation*, 169(2):982–994, 2005.

[3] T. H. Chen and W. B. Lee, "A new method for using hash functions to solve remote user authentication," *Computers and Eletrical Engineering*, 34(1):53–62, 2008.

Table 3. Functionality comparisons of our scheme with other authentication schemes

	F1	F2	F3	F4	F5	F6	F7	F8
Sun (2000) [23]	NO	NO	NO	NO	NO	YES	NO	NO
Chien et al. (2002) [5]	YES	NO	YES	NO	NO	YES	NO	NO
Chen et al. (2004) [1]	YES	NO	YES	NO	NO	NO	NO	NO
Chen and Yeh (2005) [2]	YES	YES	YES	NO	NO	YES	NO	NO
Liao et al. (2006) [18]	YES	NO	YES	YES	NO	YES	NO	NO
Chen and Lee (2008) [3]	YES	NO	YES	YES	NO	YES	NO	NO
Hsiang and Shih (2009) [8]	YES	NO	YES	YES	YES	YES	NO	NO
Chen and Huang (2010) [4]	YES	NO	YES	YES	NO	YES	NO	NO
Our scheme	YES	YES	YES	YES	YES	YES	YES	YES

F1: Mutual authentication; F2: Session key agreement; F3: Freely choose password; F4: Friendly update password; F5: Smart card revocation; F6: Resistance to stolen-verifier attack; F7: Resistance to password guessing attack with smart card security breach; F8: Resistance to masquerade attack with smart card security breach.

[4] T. H. Chen and J. C. Huang, "A novel user-participating authentication scheme," *The Journal of Systems and Software*, 83(5):861–867, 2010.

[5] H. Y. Chien, J. K. Jan, and Y. M. Tseng, "An efficient and practical solution to remote authentication: smart card," *Computers and Security*, 21(4):372–375, 2002.

[6] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, IT-31(4):469–472, 1985.

[7] Min-Shiang Hwang and Li-Hua Li, "A new remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, 46(1):28–30, 2000.

[8] H. C. Hsiang and W. K. Shih, "Weaknesses and improvements of the Yoon-Ryu-Yoo remote user authentication scheme using smart cards," *Computer Communications*, 32(4):649–652, 2009.

[9] C. L. Hsu, "Security of Chien et al.'s remote user authentication scheme using smart cards," *Computer Standards and Interfaces*, 26(3):167–169, 2004.

[10] J. K. Jan and Y. Y. Chen, "Paramita wisdom password authentication scheme without verification tables," *The Journal of Systems and Software*, 42(1):45–57, 1998.

[11] Wen-Shenq Juang, Sian-Teng Chen and Horng-Twu Liaw, "Robust and Efficient Password-Authenticated Key Agreement Using Smart Cards," *IEEE Transactions on Industrial Electronics*, 55(6):2551–256, 2008.

[12] K. Koblitz, "Elliptic curve cryptosystems," in *Mathematics of Computation*, 48(177):203–209, 1987.

[13] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proceedings of Advances in Cryptology*, pages 388-397, 1999.

[14] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, 24(11):770–772, 1981.

[15] C. T. Li and M. S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards," *Journal of Network and Computer Applications*, 33(1):1–5, 2010.

[16] C. T. Li and M. S. Hwang, "An online biometrics-based secret sharing scheme for multiparty cryptosystem using smart cards," *International Journal of Innovative Computing, Information and Control*, 6(5):2181–2188, 2010.

[17] Chun-Ta Li and Cheng-Chi Lee, "A novel user authentication and privacy preserving scheme with smart cards for wireless communications," *Mathematical and Computer Modelling*, article in press, 2011.

[18] I. E. Liao, C. C. Lee, and M. S. Hwang, "A password authentication scheme over insecure networks," *Journal of Computer and System Sciences*, 72(4):727–740, 2006.

[19] W. Mao, "Modern Cryptography: Theory and Practice," Prentice Hall PTR, 2003.

[20] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Transactions on Computers*, 51(5):541–552, 2002.

[21] M. Naor and A. Shamir, "Visual cryptography," *Lecture notes in Computer Science*, vol. 950, pages 1–12, 1995.

[22] C. Pope and K. Kaur, "Is it human or computer? Defending e-commerce with captchas," *IT Professional*, 7(2):42–49, 2005.

[23] H. M. Sun, "An efficient remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, 46(4):958–961, 2000.

[24] D. Z. Sun, J. P. Huai, J. Z. Sun, J. X. Li, J. W. Zhang and Z. Y. Feng, "Improvements of Juang 's Password-Authenticated Key Agreement Scheme Using Smart Cards," *IEEE Transactions on Industrial Electronics*, 56(6):2284–2291, 2009.

[25] E. J. Yoon, E. K. Ryu, K. Y. Yoo, "Further improvement of an efficient password based remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, 50(2):612–614, 2004.

Received February 2011.