# Efficient and Complete Remote Authentication Scheme with Smart Cards

Wen-Gong Shieh[1] and Wen-Bing Horng[2]
[1]Department of Information Management, Chinese Culture University
55 Hwa-Kang Road, Yang-Ming-Shan, Taipei 11114, Taiwan, R.O.C.
[2]Department of Computer Science and Information Engineering, Tamkang University
151 Ying-Chuan Road, Tamsui, Taipei, 25137, Taiwan, R.O.C.
[1]wgshieh@faculty.pccu.edu.tw, [2]horng@mail.tku.edu.tw

***Abstract*—A complete remote authentication scheme should provide the following security properties: (1) mutual authentication, (2) session key exchange, (3) protection of user anonymity, (4) support of immediate revocation capability, (5) low communication and computation cost, (6) resistance to various kinds of attacks, (7) freely choosing and securely changing passwords by users, and (8) without storing password or verification tables in servers. However, none of the existing schemes meets all the requirements. In this paper, along the line of cost effective approach using hash functions for authentication, we propose an efficient and practical remote user authentication scheme with smart cards to support the above complete security properties.**

***Index Terms*—Mutual authentication, revocation, session key exchange, smart card, user anonymity**

## I. Introduction

With the prevalence of computer networks all over the world, a lot of network services are provided by remote servers. To access these services, traditional remote user authentication is usually a convenient and simple way for the server to validate the user's legitimacy [11]. The smart card is an emerging technology that can enhance remote user authentication. Thus, many researchers have proposed various password based authentication schemes using smart cards [2–10, 12, 13, 17–20].

In 2000, Hwang and Li [8] proposed a novel remote user authentication scheme using smart cards based on the ElGamal public key cryptosystems without storing verification tables in servers. Later, Sun [19] proposed an efficient remote user authentication scheme based on hash functions. However, both Hwang-Li's and Sun's schemes do not allow users to choose their passwords freely. To cope with this problem, in 2002, Hwang et al. [7] proposed another cost effective remote user authentication scheme using smart cards, allowing users freely to select and to change their own passwords.

The above three authentication schemes are unilateral. However, as reported in [1, 14], many Internet frauds occurred in electronic commerce by using smart cards. Therefore, mutual authentication between server and user to authenticate each other is crucial in secure communication.

In 2002, Chien et al. [2] proposed a very efficient remote mutual authentication scheme with smart cards based on hash functions. However, as demonstrated by Hsu [6], Chien et al.'s scheme is susceptible to the parallel session attack. In 2004, Ku and Chen [10] further pointed out that this scheme is vulnerable to the reflection attack and the insider attack and that it lacks reparability. They then presented improvements over these security defects as well as a naive procedure for freely changing users' passwords. Later, Yoon et al. [20] further indicated that Ku-Chen's scheme is still vulnerable to the parallel session attack and is insecure in changing passwords. They also provided an improved version to get rid of these security flaws. However, their scheme is vulnerable to the off-line password guessing attack in the improved password change phase and is inefficient in preventing the parallel session attack.

Once the user has successfully logged into the server, an attacker might eavesdrop on the line to intercept subsequent transmitted messages. To keep the communication confidential, in 2004, Juang [9] proposed an authentication scheme to provide session key agreement based on symmetric encryption. However, this scheme is vulnerable to the known-plaintext attack and the replay attack as indicated by Shieh and Wang [18]. Later, in 2006, Liaw et al. [13] also proposed a scheme for session key exchange based on modular exponentiation. However, it is vulnerable to the known-plaintext attack and the man-in-the-middle attack and is insecure during the password change phase, as illustrated by Shieh and Horng [17].

For most of the existing authentication schemes, when a user wants to login to a server, his identity ID is transmitted in plaintext form at each time. An attacker may trace a particular user according to the transmitted ID and launch some attack actions. Therefore, protecting user's privacy in networks becomes an important issue. In 2004, Das et al. [4] first presented a dynamic ID-based unilateral authentication scheme to protect user anonymity based on hash functions, in which the remote server does not know the login user's real ID. Latter, in 2005, Chien and Chen [3] proposed a remote mutual authentication scheme to preserve user anonymity based on modular exponentiation, where the user's real ID is known to the server.

To improve the security of the server for handling the lost smart card problem, in 2005, Fan et al. [5] proposed a new scheme to provide server managers the capability to revoke users' authority immediately based on symmetric encryption.

ISI 2008, June 17-20, 2008, Taipei, Taiwan
1-4244-2415-3/08/$20.00 ©2008 IEEE

According to the above analysis, a complete and efficient remote user authentication scheme with smart cards must have the following properties:

(1) The remote servers do not need to store password or verification tables.
(2) The users can freely choose and change their own passwords.
(3) The scheme must be efficient and practical.
(4) The scheme must resist various kinds of attacks, such as replay attacks, stolen-verifier attacks, modification attacks, insider attacks, offline password guessing attacks, reflection attacks, and parallel session attacks.
(5) The scheme must provide mutual authentication.
(6) The scheme must provide session key agreement.
(7) The scheme can protect user anonymity.
(8) The scheme can support immediate revocation for lost smart cards.

However, none of the existing schemes possesses all the above eight security properties. In this paper, along the line of cost effective approach using hash functions only as in [2, 7, 10, 19, 20], we propose a complete and efficient remote user authentication scheme to satisfy all the above requirements for security properties, without resorting to costly modular exponentiation or symmetric encryption.

The rest of the paper is organized as follows. In Section II, we propose our authentication scheme. In Section III, we analyze the security of the proposed scheme. In Section IV, we compare our scheme with other related schemes. Finally, we conclude this article in the last section.

## II. Proposed Authentication scheme

In this section, we propose a complete and efficient authentication scheme with smart cards based on secure one-way hash functions to meet all the eight requirements discussed in Section I. The scheme consists of five different phases: the registration phase, the login phase, the verification phase, the password change phase, and the revocation phase.

### A. Registration Phase

Suppose that $x$ and $p$ are two permanent secret keys stored in a remote server $S$, and $h(\cdot)$ is a public secure one-way hash function [15], where $p$ is used to provide user anonymity. Let $n$ denote the number of times a user $U$ re-registers to $S$, which is stored in the account database $DB$ of $S$. The registration phase proceeds as follows:

(1) If $U$ wishes to register to $S$, he/she first selects his identity $ID$ and password $PW$. (If he/she wants to re-register to $S$, he/she only needs to select a new password $PW$.) Then, he/she chooses a random number $b$ and computes $EPW = h(b \| PW)$.
(2) $U$ submits $ID$ and $EPW$ to $S$ via a secure channel.
(3) If $S$ accepts $U$'s registration (or re-registration) request, $S$ performs the following steps:
   (a) If $U$ is a new user to $S$, set $n = 0$; otherwise, if $U$ wants to re-register to $S$, retrieve $n$ by $ID$ from $DB$ and increment $n$ by 1. Store ($ID$, $n$) to $DB$.
   (b) Compute $V = h(x \| EID)$, where $EID = (ID \| n)$.
   (c) Compute $R = V \oplus EPW$.
(4) $S$ issues $U$ a smart card containing $R$ and $p$ over a secure channel.
(5) $U$ enters and stores $b$ into his new smart card so that he/she does not need to remember $b$.

Note that as in most of the proposed authentication schemes, the smart card is considered as a secure device. In our scheme, we assume that the values of $R$, $p$, and $b$ stored in the smart card cannot be retrieved by any holder of the smart card.

### B. Login (*Password*) Change Phase

If $U$ wishes to login to $S$ (or to change his password), he/she inserts his smart card into the card reader of a terminal and inputs his $ID$ and $PW$. The smart card performs the following steps, as shown in Fig. 1:

(1) Set $O = 1$ for the login request (or $O = 0$ for the password change request).
(2) Compute $V = R \oplus EPW$, where $EPW = h(b \| PW)$.
(3) Acquire the current timestamp $T$ and generate a random number $r$.
(4) Compute $C_1 = h(r \| O \| T \| V)$. Note that $C_1$ is a *message authentication code* (*MAC*) acting as a challenge message to the server $S$.
(5) Compute an anonymous identity $AID = ID \oplus h(p \| T \| r)$.
(6) Send the message ($AID$, $T$, $r$, $O$, $C_1$) to $S$.

### C. Verification Phase

On the receipt of $U$'s login (or password change) request message ($AID$, $T$, $r$, $O$, $C_1$), $S$ performs the following steps to authenticate $U$:

(1) Verify the freshness of $T$. If it fails, reject $U$'s request and stop.
(2) Compute $ID = AID \oplus h(p \| T \| r)$ and check the validity of $ID$. If it fails, reject $U$'s request and stop.
(3) Retrieve $n$ by $ID$ from $DB$ and compute $EID = (ID \| n)$.
(4) Compute $V = h(x \| EID)$ and check whether $h(r \| O \| T \| V) = C_1$. If they are equal, $S$ believes $U$ is a legal user because $U$ has the shared secret $V$; otherwise, reject $U$'s request and stop.
(5) Compute $C_2 = h(C_1 \| V)$ and send it to $U$, where $C_2$ is a MAC acting as a response message to $C_1$ for mutual authentication.
(6) If $O = 1$, then accept $U$'s login request, compute a session key $SK = h(C_2 \| V)$, and continue the current session; otherwise, terminate the communication for the password change request.

Upon receiving the reply message $C_2$ from $S$, the smart card performs the following steps to authenticate $S$:

(1) Check whether $h(C_1 \| V) = C_2$. If it fails, give up the login (or password change) request. Otherwise, the authentication is successful because $S$ contains the MAC $C_1$ and the shared secret $V$.

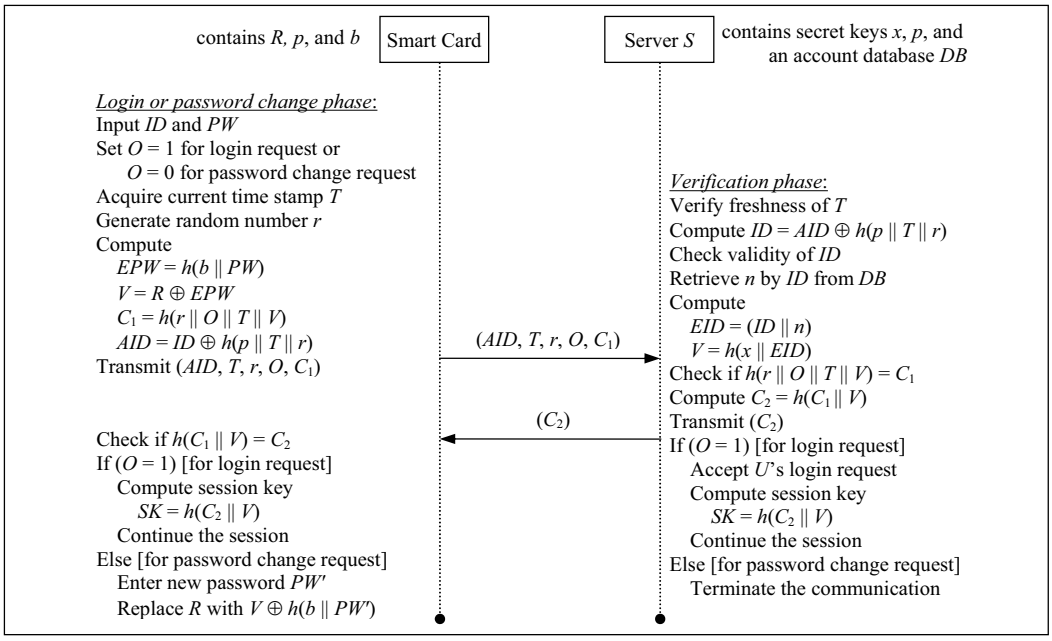


Fig. 1. The login phase, verification phase, and password change phase of the proposed scheme.

(2) If $O = 1$, compute a session key $SK = h(C_2 \| V)$, and continue the current session.
(3) Otherwise, perform the following steps to change password:
   (a) Ask $U$ to input a new password $PW'$.
   (b) Replace $R$ with $V \oplus h(b \| PW')$.

### D. Revocation Phase

If $U$ loses his smart card, the server manager may change his $n$ value stored in $DB$ of $S$ to immediately revoke the access authority of $U$ to maintain user's immediate security. In our scheme, $n$ is incremented by one. Note that the server manager does not need to delete $U$'s entry from $DB$. Later, the revoked user $U$ may re-register to $S$ without changing his $ID$. Though $n$ is stored in $DB$, $S$ only needs to keep its integrity but not its confidentiality.

## III. Security Analysis and Discussion

1. *No password or verification table.* In this scheme, only the values of $x$ and $p$ are kept secretly in $S$; there are no password or verification tables stored in $S$. Thus, our scheme will not suffer from the stolen-verifier attack and the modification attack.
2. *Freely choosing and securely changing passwords.* In some remote authentication schemes as in [8, 19], strong passwords are assigned to users by the server. These passwords are usually too long (for example, 1024 bits) to remember. In our scheme, users can freely choose their own passwords (i.e., weak passwords for easily remembering) during the registration phase. In addition, our scheme also provides a secure password change phase with the help of the server verification to let users change their own passwords at will, unlike the insecure password change procedures in [7, 10, 20].
3. *Efficiency and practicability.* The smart card needs only five hash operations during the login (or change password) phase, verification phase, and session key generation. Since the computation cost in the smart card is extremely low, the proposed scheme guarantees user efficiency. Even in the server, only four hash operations and one database retrieval operation are performed. Thus, server efficiency is also guaranteed. These low computation costs in both the smart card and the server make the scheme more practical.
4. *Resistance to the parallel session attack and the reflection attack.* Parallel session attacks in [2, 10] and reflection attacks in [2] are resulted from symmetric computations of MACs on both the server and the smart card. Since we employ asymmetric computations of MACs $C_1$ and $C_2$, our scheme can withstand these two kinds of attacks.
5. *Resistance to the replay attack.* Because our scheme uses timestamp-based approach, the server will check the freshness of each login (or password change) request message to avoid the replay attack, as does in Ku-Chen's [10] scheme. In addition, our scheme also applies challenge/response mechanism to avoid the replay attack. That is, $C_1$, created by the smart card, is a fresh challenge containing fresh timestamp $T$ and $C_2$ is a response containing the fresh $C_1$. A replayed $C_2$ containing an old $C_1$ will never pass the check performed by the smart card.
6. *Resistance to the offline password guessing attack.* In the login (or change password) request message, $C_1 = h(r \| O \| T \| V)$ does not contain any password information, but $V = R \oplus EPW = R \oplus h(b \| PW)$ resulting in $C_1 = h(r \| O \| T$

$\| R \oplus h(b \| PW)$) may be used to guess password. However, $R$ and $b$ are kept secretly inside the smart card and cannot be retrieved by any cardholder. Therefore, although $T$ and $O$ can be intercepted from the network, correct $PW$ in $C_1 = h(r \| O \| T \| R \oplus h(b \| PW))$ cannot be guessed without knowing $R$ and $b$. Similarly, correct $PW$ cannot be guessed from the reply message $C_2 = h(C_1 \| V)$ during the authentication phase.

7. *Resistance to the insider attack.* During the registration phase, if $U$'s password $PW$ is revealed to $S$, the insider of $S$ may directly obtain $PW$ and impersonate $U$ to access $U$'s other accounts in other servers if $U$ uses the same password for the other accounts. This causes the so-called insider attack. Even if $h(PW)$ is presented to $S$, an offline password guessing attack might be performed on $h(PW)$ by an insider of $S$. Like [10], our scheme asks $U$ to present $h(b \| PW)$ instead of $h(PW)$ to $S$. Since $b$ is only known to $U$, the insider of $S$ cannot launch any offline password guessing attack on $h(b \| PW)$ to obtain $PW$ unless he/she knows $b$.
8. *Mutual authentication.* In this scheme, mutual authentication between $U$ and $S$ is achieved by sending MACs $C_1$ and $C_2$. $U$'s smart card sends $C_1$ as a challenge message to $S$ to be authenticated. Since this scheme assumes the information stored in the smart card cannot be retrieved, nobody can create the correct MAC $C_1$ without knowing $R$, $b$, and $PW$. If $U$ inputs correct password $PW$, then $C_1$ contains the correct shared secret $V$ and will pass the validation of $S$ because only $S$ and $U$ know $V$. Thus, $S$ believes that $U$ is a legal user. Similarly, $S$ sends $C_2$ as a response message to $U$'s smart card for authentication. Because $C_2$ contains the shared secret $V$ and the fresh challenge $C_1$, it will pass the validation of $U$'s smart card. Thus, $U$ believes $S$ is a legal server.
9. *Session key agreement.* This scheme provides a session key agreement during the verification phase. A session key $SK = h(C_2 \| V)$ is generated in both S and $U$. Since $C_2$ contains $C_1$ that in turn contains timestamp $T$, $SK$ will be different for each login session. Because $R$ cannot be retrieved and password $PW$ is only known to $U$, $SK$ will not be known by attackers. Thus, the communication parties can use it securely to encrypt and decrypt subsequent messages.
10. *User anonymity.* Each time when $U$ wants to login, our scheme will compute an anonymous identity $AID = ID \oplus h(p \| T \| r)$ for $U$. Note that the $AID$ will be different for each login (or password change) request since the timestamp $T$ (acquired from the terminal) and random number $r$ (generated by the smart card itself) are different at each request. Since $p$ is a shared secret key, $h(p \| T \| r)$ works as a session key used to hide the user $U$'s identity $ID$ during the transmission of the login (or password change) request message, and finally $ID$ will be known to $S$. Because $p$ is irretrievable from the smart card, there is no straight forward way to compute $ID$ from the intercepted $AID$, $T$, and $r$ from the transmitted message. However, if an attacker has intercept $U$'s login request message and has stolen $U$'s smart card, he may use the smart card trying to derive $U$'s $ID$ from $AID \oplus (AID^* \oplus ID^*)$, where $ID^*$ is a forged identity provided by the attacker to the smart card, and $AID^*$ is computed by the smart card and can be intercepted by the attacker. This is because $AID \oplus ID = h(p \| T \| r) = AID^* \oplus ID^*$. Although the timestamp $T$ can be provided outside from the attacker to the smart card, the random number $r$ cannot be obtained from the attacker; it must be generated by the smart card itself. Therefore, it is impossible for an attacker to derive $ID$ in this way. Thus, user anonymity is protected.

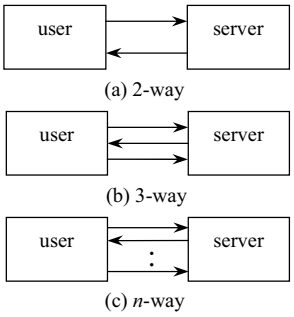


Fig. 2. Three kinds of mutual authentication mechanisms.

11. *Revocation.* The value of $n$ is protected in $R$ stored in $U$'s smart card. If $U$'s smart card is lost or stolen, or whenever necessary, the server manager will revoke the smart card by incrementing the value of $n$ by one in $DB$. Since the revoked smart card contains the old $n$ value, using the old smart card will not pass the verification phase performed by $S$. Of course, $U$'s account is still kept in $DB$, $U$ can re-register to $S$ to obtain a new smart card with a new $n$ value.
12. *Efficient 2-way mutual authentication.* For mutual authentication schemes, several messages are transmitted to verify the legitimacy of both the user and the server before successful login. Fig. 2 illustrates three possible mutual authentication schemes. Fig. 2(a) is a 2-way mutual authentication; it is usually used in the timestamp-based approach, because only a pair of challenge/response messages is transmitted. Fig. 2(b) is a 3-way authentication; it is usually used in the nonce-based approach, since two pairs of challenge/response messages are transmitted, in which the second message acts both as a response message and as a challenge message. An $n$-way authentication, as illustrated in Fig. 2(c), includes more messages for session key exchange, in addition to challenge/response messages, such as in [12]. It is obvious that 2-way mutual authentication schemes are the most efficient (fastest) mechanisms in terms of communication time because only two messages are used for authentication. In our scheme, a session key is also generated during the efficient 2-way mutual authentication.

TABLE I
COMPARISONS OF COMPUTATION AND COMMUNICATION COSTS

| **Authentication Scheme** | *E1* | *E2* | *E3* | *E4* | *E5* | *E6* |
|---|---|---|---|---|---|---|
| Our scheme | 128 | 2-way + K | 513 | 2H | 2H | 5H |
| Shieh-Horng (2007) [17] | 128 | 3-way + K | 1654 | 1H | 2E + 1S + 2H | 2E + 1S + 5H |
| Liao et al. (2006) [12] | 128 | 3-way | 640 | 1E + 1H | 1E + 3H | 1E + 3H |
| Liaw et al. (2006) [13] | 128 | 5-way + K | 2560 | 1H | 3E + 1S + 1H | 3E + 1S + 2H |
| Shieh-Wang (2006) [18] | 128 | 2-way + K | 512 | 1H | 3H | 3H |
| Fan et al. (2005) [5] | 128 | 3-way | 1536 | 1S + 1H | 4H | 1S + 4H |
| Chien-Chen (2005) [3] | 128 | 2-way + K | 4416 | 2H | 2E + 2S | 2E + 2S + 1H |
| Juang (2004) [9] | 128 | 2-way + K | 832 | 1H | 3S + 1H | 3S + 2H |
| Das et al. (2004) [4] | 128 | 1-way | 448 | 2H | 4H | 3H |
| Yoon et al. (2004) [20] | 128 | 2-way | 512 | 2H | 3H | 3H |
| Ku-Chen (2004) [10] | 128 | 2-way | 512 | 1H | 3H | 3H |
| Chien et al. (2002) [2] | 128 | 2-way | 512 | 1H | 2H | 3H |
| Hwang et al. (2002) [7] | 128 | 1-way | 320 | 1H | 2H | 2H |
| Sun (2000) [19] | 128 | 1-way | 320 | 1H | 1H | 2H |
| Hwang-Li (2000) [8] | 1024 | 1-way | 2240 | 1E | 3E + 1H | 3E + 1H |

*E1*: password length (in bit); *E2*: method of authentication; *E3*: communication cost (in bit); *E4*: computation cost of registration; *E5*: smart card computation cost of authentication; *E6*: server computation cost of authentication; E: modular exponentiation; S: symmetric encryption/decryption; H: hashing operation; K: session key agreement.

## IV. COMPARISONS OF RELATED SECURITY SCHEMES

### A. Communication and Computation Costs

In Table I, we evaluate the efficiency of our two proposed schemes and other related schemes in terms of communication and computation costs. We assume that the length of the prime number $p$ in Hwang-Li's [8] scheme, Chien-Chen's [3], Liaw et al.'s [13], and Liao et al.'s [12] schemes is 1024 bits in order to make the discrete logarithm and factoring problems infeasible. We assume that the block size of secure symmetric cryptosystems [16] is 128 bits and the output size of secure one-way hash function [15] is 128 bits. For comparison, we also assume that, without loss of generality, the lengths of IDs and freely chosen PWs are 128 bits, and the sizes of timestamps and random numbers are 64 bits. In Hwang-Li's [8] and Sun's [19] schemes, user passwords are generated, respectively, by discrete logarithm and hash function, while in other schemes, passwords are chosen freely by users.

For the method of authentication field (*E2*), "1-way" stands for unilateral authentication, "$n$-way" ($n > 1$) for mutual authentication, and "K" for additionally providing session key agreement during authentication. The communication cost (*E3*) of each scheme includes all the transmitted messages for authentication (including the login phase, the verification phase, and session key agreement if provided). For the computation costs of registration (*E4*), smart card (*E5*), and server (*E6*), we only consider hash functions, symmetric encryptions/decryptions, and exponential operations because other operations, such as exclusive-or operations and string concatenations, are much cheaper operations. Of these three compared operations, the cost of exponential operation is much more expensive than that of symmetric encryption/ decryption, which in turn is more expensive than that of hash function. Note that since some schemes [3, 9, 13, 17, 18] provide session key agreements, the computation cost fields (*E4*, *E5*, and *E6*) also include operations for generating session keys because in some of these schemes, session key generation is embedded in the authentication procedure. From Table I, it is obviously that our proposed scheme provides efficient 2-way authentication with session key agreement and the costs of communication and computation both on smart card and server are very low as compared to other schemes.

### B. Security Features

In Table II, we compare our schemes with other related authentication schemes according to the following thirteen different security features:

- *F1*: no password or verification table
- *F2:* freely choosing password
- *F3:* freely changing password
- *F4:* providing mutual authentication
- *F5:* providing session key agreement
- *F6:* protecting user anonymity
- *F7:* lost card revocation capability
- *F8:* secure password change phase (for freely changing password)
- *F9:* resistance to offline password guessing attacks
- *F10:* resistance to insider attacks (not revealing to the administrator of the server)
- *F11:* resistance to parallel session attacks (for mutual authentication)
- *F12:* resistance to reflection attacks (for mutual authentication)

Note that feature *F8* (i.e., secure password change phase) requires that the password change phase only allow legal users to change their passwords and can withstand the offline password guessing attack. In this table, Hwang et al. [7], Ku and Chen [10], Das et al. [4], and Liaw et al. [13], and Liao et al. [12] proposed a naive but insecure password change phase because their procedures do not check the legitimacy of the cardholder. Therefore, it is impossible for an attacker to make sure whether he/she has guessed a correct password offline in those schemes. However, for Yoon et al.'s [20] scheme, although they checked users' legitimacy, their scheme is vul-

TABLE II
COMPARISONS OF SECURITY FEATURES

| Authentication Scheme | *F1* | *F2* | *F3* | *F4* | *F5* | *F6* | *F7* | *F8* | *F9* | *F10* | *F11* | *F12* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Our scheme | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| Shieh-Horng (2007) [17] | ○ | ○ | ○ | ○ | ○ | × | × | ○ | ○ | × | ○ | ○ |
| Liao et al. (2006) [12] | ○ | ○ | ○ | ○ | ○ | × | × | × | × | × | ○ | ○ |
| Liaw et al. (2006) [13] | ○ | ○ | ○ | ○ | ○ | × | × | × | ○ | × | ○ | ○ |
| Shieh-Wang (2006) [18] | ○ | ○ | × | ○ | ○ | × | × | – | ○ | × | ○ | ○ |
| Fan et al. (2005) [5] | ○ | ○ | × | ○ | × | × | ○ | – | ○ | × | ○ | ○ |
| Chien-Chen (2005) [3] | ○ | ○ | × | ○ | ○ | ○ | × | – | ○ | × | ○ | ○ |
| Juang (2004) [9] | ○ | ○ | × | ○ | ○ | × | × | – | ○ | × | ○ | ○ |
| Das et al. (2004) [4] | ○ | ○ | ○ | × | × | ○ | × | × | ○ | × | – | – |
| Yoon et al. (2004) [20] | ○ | ○ | ○ | ○ | × | × | × | × | × | ○ | ○ | ○ |
| Ku-Chen (2004) [10] | ○ | ○ | ○ | ○ | × | × | × | × | ○ | ○ | × | ○ |
| Chien et al. (2002) [2] | ○ | ○ | × | ○ | × | × | × | – | ○ | × | × | × |
| Hwang et al. (2002) [7] | ○ | ○ | ○ | × | × | × | × | × | ○ | × | – | – |
| Sun (2000) [19] | ○ | × | × | × | × | × | × | – | ○ | × | – | – |
| Hwang-Li (2000) [8] | ○ | × | × | × | × | × | × | – | ○ | × | – | – |

○: supported, ×: not supported; –: not relevant.

nerable to offline password guessing attacks. Thus, their scheme is insecure in their password change phase. Liao et al.'s [12] scheme is also vulnerable to offline password guessing attacks. Note that Fan et al.'s [5] claimed that their scheme could withstand offline password guessing attacks even if the contents of the smart card could be retrieved. However, Fan et al.'s scheme is still vulnerable to offline password guessing attacks if the contents of the smart card can be revealed. From Table II, it is apparently that our second scheme satisfies all the security requirements as analyzed in Section I.

## V. CONCLUSION

In this paper, we have proposed a complete and efficient remote user authentication scheme with smart cards. This scheme possess the following properties: (1) no password or verification table, (2) freely choosing and changing password, (3) low communication and computation cost, (4) providing mutual authentication, (5) generating session key, (6) offering revocation capability, (7) protecting user's anonymity, and (8) resisting various kinds of attacks, such as the offline password guessing attack. One important characteristic of the scheme is that it uses hash functions only, without using costly modular exponentiations or symmetric encryption/decryption, which makes our scheme become efficient and practical.

## REFERENCES


[1] N. Aoskan, H. Debar, M. Steiner, and M. Waidner, "Authenticating public terminals," *Comput. Network*, vol. 31, pp. 861–870, 1990.

[2] H. Y. Chien, J. K. Jan, and Y. M. Tseng, "An efficient and practical solution to remote authentication: smart card," *Comput. Secur.*, vol. 21, pp. 372–375, 2002.

[3] H. Y. Chien and C. H. Chen, "A remote authentication scheme preserving user anonymity," in *Proc. 19th Int. Conf. Advanced Information Networking and Applications*, Taipei, Taiwan, 2005, pp. 245–248.

[4] M. L. Das, A. Saxena, and V. P. Gulati, "A dynamic ID-based remote user authentication scheme," *IEEE Trans. Consumer Electron.*, vol. 50, pp. 629–631, 2004.

[5] C. I. Fan, Y. C. Chan, and Z. K. Zhang, "Robust remote authentication scheme with smart cards", *Comput. Secur.*, vol. 24, pp. 619–628, 2005.

[6] C. L. Hsu, "Security of Chien et al.'s remote user authentication scheme using smart cards," *Comput. Stand. Interfaces*, vol. 26, pp. 167–169, 2004.

[7] M. S. Hwang, C. C. Lee, and Y. L. Tang, "A simple remote user authentication scheme," *Math. Comput. Model.*, vol. 36, pp. 103–107, 2002.

[8] M. S. Hwang and L. H. Li, "A new remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. 46, pp. 28–30, 2000.

[9] W. S. Juang, "Efficient password authenticated key agreement using smart cards," *Comput. Secur.*, vol. 23, pp. 167–173, 2004.

[10] W. C. Ku and S. M. Chen, "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. 50, pp. 204–207, 2004.

[11] L. Lamport, "Password authentication with insecure communication," *Commun. ACM*, vol. 24, pp. 770–772, 1981.

[12] I. E. Liao, C. C. Lee, and M. S. Hwang, "A password authentication scheme over insecure networks," *J. Comput. Syst. Sci.*, vol. 72, pp. 727–740, 2006.

[13] H. T. Liaw, J. F. Lin, and W. C. Wu, "An efficient and complete remote user authentication scheme using smart cards," *Math. Comput. Model.*, vol. 44, pp. 223–228, 2006.

[14] D. McElroy and E. Turban, "Using smart cards in electronic commerce," *Int. J. Inform. Manag.*, vol. 18, pp. 61-72, 1998.

[15] NIST FIPS PUB 180-2, *Secure Hash Standard*, National Institute of Standards and Technology, U.S. Department of Commerce, DRAFT, 2002.

[16] NIST FIPS PUB 197, *Announcing the Advanced Encryption Standard (AES)*, National Institute of Standards and Technology, U.S. Department of Commerce, November 2001.

[17] W. G. Shieh and W. B. Horng, "An improvement of Liaw-Lin-Wu's efficient and complete remote mutual authentication with smart cards," *WSEAS Trans. Info. Sci. Appl.*, vol. 4, pp. 1200–1205, 2007.

[18] W. G. Shieh and J. M. Wang, "Efficient remote mutual authentication and key agreement," *Comput. Secur.*, vol. 25, pp. 72–77, 2006.

[19] H. M. Sun, "An efficient remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. 46, pp. 958–961, 2000.

[20] E. J. Yoon, E. K. Ryu, and K. Y. Yoo, "Further improvement of an efficient password based remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. 50, pp. 612–614, 2004.