International Journal of Innovative Computing, Information and Control
Volume **8**, Number **8**, August **2012**

ICIC International ©2012 ISSN 1349-4198
pp. **5499–5511**

A SMART CARD BASED AUTHENTICATION SCHEME FOR REMOTE USER LOGIN AND VERIFICATION

Zi-Yao Cheng[1], Yun Liu[1], Chin-Chen Chang[2,3,4] and Shih-Chang Chang[4]

[1]Department of Electronic and Information Engineering
Key Laboratory of Communication and Information Systems
Beijing Municipal Commission of Education
Beijing Jiaotong University
No. 3, Shang Yuan Cun, Hai Dian District, Beijing 100044, P. R. China
{ 09111024; liuyun }@bjtu.edu.cn

[2]Department of Information Engineering and Computer Science
Feng Chia University
No. 100, Wenhwa Road, Seatwen, Taichung 40724, Taiwan
alan3c@gmail.com

[3]Department of Biomedical Imaging and Radiological Science
Chinese Medical University
No. 91, Hsueh-Shih Road, Taichung 40402, Taiwan

[4]Department of Computer Science and Information Engineering
National Chung Cheng University
No. 160, San-Hsing, Ming-Hsiung, Chiayi 621, Taiwan
chang.coby@gmail.com

Received April 2011; revised September 2011

Abstract. *With the advancement of Internet network technologies, remote user authentication schemes using smart cards have been widely adopted. In order to satisfy the requirements of a remote user authentication scheme, the smart card has become an essential device, one that is widely used because of its low computation cost and expedient portability. To achieve computation efficiency and system security, many researchers have focused on this field and published corresponding literature. Recently, Chen et al. proposed security enhancement on an improvement on two remote user authentication schemes using smart cards. They claimed their method does not have the security weaknesses of Wang et al.'s scheme such as impersonation attack and parallel session attack, and preserves important criteria through which a legal user can negotiate a specific session key with his remote authentication server by executing mutual authentication. Meanwhile, the scheme can provide high-level perfect forward secrecy. However, there is much room for security enhancement in Chen et al.'s scheme. In this paper, we suggest that serious vulnerabilities still threaten security requirements, and that security enhancements still cannot withstand known-key attack and off-line guessing attack. Accordingly, we propose an enhanced scheme to remedy these security weaknesses and prove that this scheme is more secure and efficient for network application with merits in its properties.*
Keywords: Mutual authentication, Cryptanalysis, Smart card, Security, Key agreement

1. **Introduction.** As far as current Internet technologies are concerned, providing concise and secure services has been extensively investigated for a long time. In this context, a remote authentication scheme has become essential, in which a remote user with a computer can receive quality service and secure communication from a homologous server that requires authentication from the user.

It is generally known that the first proposed remote authentication scheme was based on a password to identify a legitimate user over even an insecure channel [1-3], and this is the subject of a published research by Lamport in 1981 [4]. It has been claimed that there is a potential security threat caused by a stored verifier table on a remote authentication system, because the verifier table risks being modified by an adversary and has high maintenance cost, even though all secret passwords can be encrypted to the threat of disclosure. Later, Hwang and Li [5] presented the weakness of Lamport's scheme and proposed a new scheme based on the ElGamal public-key encryption system [6] to solve corresponding problem. In this novel method, there is no need to maintain any verifier table to achieve remote user authentication. In view of the low cost and capacity of cryptosystems, Sun [7] developed an authentication scheme to enhance the performance efficiency of Hwang and Li's scheme by involving only several one-way hash operations, so that the scheme could serve as an ideal substitute for high-cost modular exponentiations. Nevertheless, these two mentioned schemes could not provide users with a free choice of passwords and mutual authentication.

Since the smart card is with the tamper-resistant properties, it can solve the problem of maintaining the verifier table on the server side. In a smart card based authentication system, only the user was required to hold a smart card, which was issued by the server for more convenient communication and which contained all kinds of stored secret information. Many related studies [8-12] have been investigated and the smart card has become essential in remote authentication schemes. More specifically, Chien et al. [13] proposed an effective solution for remote authentication schemes by using smart cards. Their contributions contain several aspects such as mutual authentication between the user and the server, free choice of passwords, and the requirement of only one-way hash operations. Besides, there is no need to process extra computation cost for maintaining the verifier table which achieves the requirements of low cost. This complements the attributes of cryptographic capacity and portability. However, Chen et al.'s scheme has serious security weaknesses, in which it cannot protect against inside attack, guessing attack and reflection attack. In 2004, Ku and Chen proposed an improved scheme [14] to overcome these weaknesses, but Yoon et al. [15] claimed that Ku and Chen's scheme was still vulnerable to parallel attack; especially, they maintained that their scheme was unfeasible when the user arbitrarily changed his password. Then, Yoon et al. proposed an improvement to enhance Ku and Chen's scheme. Unfortunately, Wang et al. [16] found that an adversary could threaten both these schemes [14,15] by achieving guessing attack, forgery attack and denial of service (DoS) attack; consequently, they proposed an efficient enhancement based on these two schemes.

In 2010, Chen et al. [17] pointed out that Wang et al.'s scheme could not withstand impersonation attack [18] and parallel session attack [14]; hence, they proposed an improved approach over Wang et al.'s scheme. After an in-depth analysis, we found that Chen et al.'s scheme is actually not as secure as they claimed, since it is still susceptible to known-key attack and off-line guessing attack. Hence, we propose a novel scheme to defend against the mentioned security weaknesses. Furthermore, our proposed scheme has better computation efficiency, which has become clear by comparing previous works with ours. In addition, our scheme has the following properties:

P1. Freely chosen and exchanged password: A legal user can freely choose and change his password [13].
P2. No verification table: There is no need to maintain a verification table on the server side [5].

P3. No adversary can derive the known-key in the scheme: No one can utilize the secret information of a legal user to derive the session key.
P4. No malicious user can guess the secret long-term key of the server: The secret long-term key is protected against off-line guessing attack to prevent malicious users from imitating the authentication server.
P5. Mutual authentication: Both the legal user and the remote server can authenticate each other successfully [13].
P6. Session key agreement: The legal user and the remote server can negotiate a session key and utilize it to process subsequent communication [16].
P7. Perfect forward secrecy: Even if an adversary can obtain contiguous knowledge of the long-term key, he cannot derive the previous session keys.
P8. Efficiency and practicability: We ensure that our proposed scheme has higher computation efficiency by a comparison of performance, and is more practical for use in networking environments.

The rest of this paper is organized as follows. In Section 2, we review Chen et al.'s scheme and demonstrate its security weaknesses. In Section 3, we present our proposed scheme, and in Section 4, we illustrate the security analysis. In Section 5, we compare the performance of our scheme with those of Wang et al. and Chen et al. Finally, our concluding remarks are shown in Section 6.

2. **Related Works.**

In this section, we review Chen et al.'s authentication scheme and then show that their scheme cannot protect against known-key attack and off-line guessing attack. The details and weaknesses of Chen et al.'s scheme are demonstrated in Subsections 2.1 and 2.2, respectively.

We first introduce the notations throughout this paper as follows:

- U: the user
- ID: the identity of user
- PW: the password of user U
- S: the remote server
- x: the permanent private key of the remote server U
- $h(\cdot)$: a one-way hash function without a cryptographic key
- $h_p(\cdot)$: a one-way hash function with a cryptographic key p
- $\Rightarrow$: a secure channel
- $\rightarrow$: a common channel
- $\|$: a concatenation operator which combines two strings into one

2.1. **Review of Chen et al.'s scheme.**

In this subsection, we briefly review the specific procedures of Chen et al.'s scheme. This scheme includes four phases: the registration phase, the login phase, the verification phase, and the password change phase.

2.1.1. *Registration phase.* We illustrate the procedures of this phase in Figure 1 and show the details as follows. Whenever U initially registers with S, the registration phase is invoked:

Step 1: U chooses a random number b and computes $h(b \oplus PW)$, then sends it with his ID to the server S; $U \Rightarrow S : ID$, $h(b \oplus PW)$.
Step 2: S calculates the following parameters: $p = h(ID \oplus x)$, $R = p \oplus h(b \oplus PW)$, $V = h_p(h(b \oplus PW))$, and the server S stores the data $\{V, R, h(\cdot), h_p(\cdot)\}$ on a new smart card, and issues the smart cart to user U.
Step 3: U enters b into his smart card so that it contains $\{V, R, h(\cdot), h_p(\cdot), b\}$.

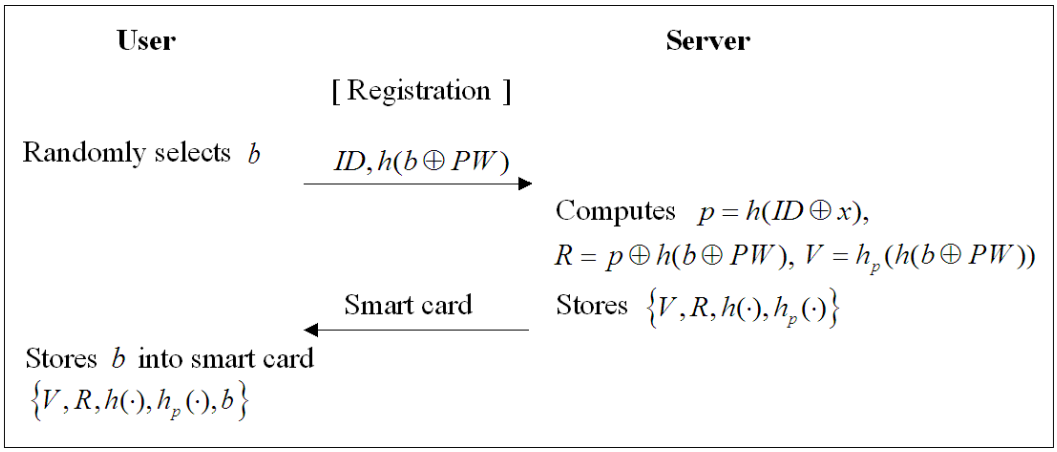

FIGURE 1. Registration phase of Chen et al.'s scheme

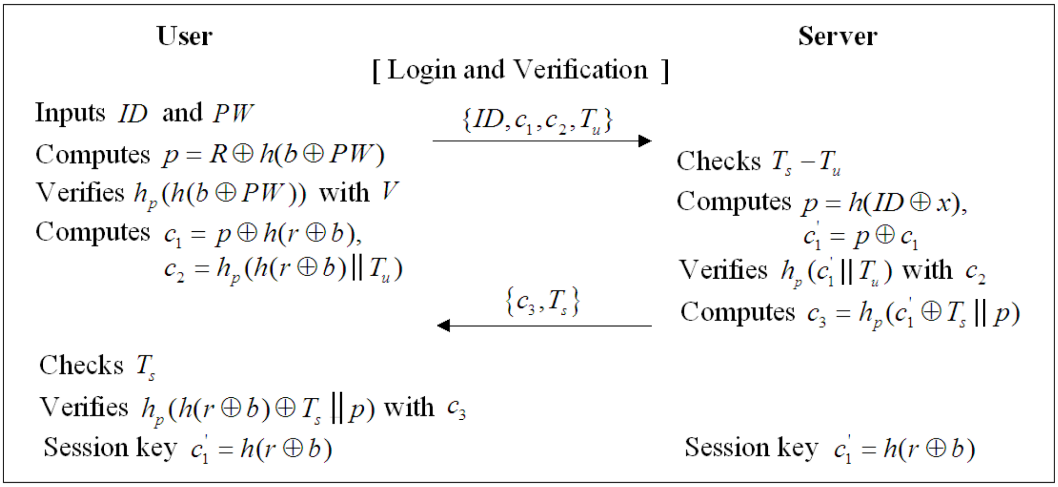

FIGURE 2. Login and verification phase of Chen et al.'s scheme

2.1.2. *Login phase.* When U attempts to login to the server S, he should execute the following steps and this phase is depicted in Figure 2.

Step 1: U inserts the smart card into the smart card reader and inputs his ID and PW.

Step 2: The smart card computes $p = R \oplus h(b \oplus PW)$ and checks whether $h_p(h(b \oplus PW))$ equals V. If so, the smart card continues to calculate $c_1 = p \oplus h(r \oplus b)$, $c_2 = h_p(h(r \oplus b) \parallel T_u)$, where r is a random number generated by the smart card and T_u is the current timestamp of U.

Step 3: U sends a login request message to the server S; $U \rightarrow S : \{ID, c_1, c_2, T_u\}$.

2.1.3. *Verification phase.* Upon receiving the login request message, the following steps can be depicted in Figure 2 and the details can be shown:

Step 1: S checks the validity of ID and whether $T_s > T_u$, where T_s is the current timestamp of the server. If one of them does not hold, then S rejects the login request; otherwise, S checks whether $T_s - T_u$ is within a valid time interval ΔT. If not, S rejects the login request.

Step 2: If $T_s - T_u$ is really within the interval ΔT, S computes $p = h(ID \oplus x)$ and $c_1' = p \oplus c_1$ in order to check whether $h_p(c_1' \parallel T_u)$ equals the original c_2. If so, the

validity of U is authenticated and $S \rightarrow U : \{c_3, T_s\}$, where $c_3 = h_p(c_1{}' \oplus T_s \parallel p)$; otherwise, S rejects the login request.

Step 3: After receiving $\{c_3, T_s\}$, U checks the validity of T_s and whether $T_s > T_u$. If it does not hold, U terminates the connection; otherwise, U checks whether $h_p(h(r \oplus b) \oplus T_s \parallel p)$ equals the received c_3. If so, the validity of S is authenticated.

Step 4: Moreover, U and S establish a common session key $c_1{}' = h(r \oplus b)$ for private communication.

2.1.4. *Password change phase.* U can freely change his password PW to PW_{new} in this phase as follows:

Step 1: U inserts the smart card into the smart card reader, inputs his ID and PW and requests to change his password. Thus, the smart card computes $p^* = R \oplus h(b \oplus PW)$, $V^* = h_{p^*}(h(b \oplus PW))$.

Step 2: The smart card checks whether V^* equals the original V stored in the smart card. If so, then U selects a new password PW_{new}; otherwise, the smart card rejects the password change request.

Step 3: The smart card computes $R_{new} = p^* \oplus h(b \oplus PW_{new})$ and $V_{new} = h_{p^*}(h(b \oplus PW_{new}))$, then stores them and replaces the original R and V, respectively.

2.2. **Weaknesses of the reviewed scheme.** Chen et al. claimed that their method is an enhanced version of Wang et al.'s scheme that can withstand impersonation attack [18] and parallel session attack [14]. In this sub-section, we show that Chen et al.'s scheme is still vulnerable to known-key attack and off-line guessing attack.

2.2.1. *The known-key attack.* A similar description of the known-key attack was presented [19]. We assume that an adversary compromises the parameter $c_1{}'$ such as $c_1{}' = p \oplus c_1$; he can easily intercept the parameter c_1 from the login request message $\{ID, c_1, c_2, T_u\}$, and then derive the secret parameter $p = c_1{}' \oplus c_1$. Thus, the adversary can utilize the derived parameter p and select two random numbers r' and b' to perform the following computations: $c_1^* = p \oplus h(r' \oplus b')$, $c_2^* = h_p(h(r' \oplus b') \parallel T_u')$. As a result, we can see that the adversary can execute the following procedures by sending a fabricated login request message $\{ID, c_1^*, c_2^*, T_u'\}$ to the server S. After receiving the adversary's login message, the verification phase is followed step by step:

Step 1: S checks either if the format of ID is invalid or $T_u' = T_s'$, where T_s' is the current timestamp of the server. Due to the transmission delay or the adversary delay on purpose, T_u' cannot be equal to T_s'. Hence, the adversary can smoothly pass this step.

Step 2: S computes $p = h(ID \oplus x)$ and $c_1^{''} = p \oplus c_1^* = h(r' \oplus b')$. Upon calculating the result, S can get the verification $c_2^{''} = h_p(c_1^{''} \parallel T_u') = h_p(h(r' \oplus b') \parallel T_u') = c_2^*$ in Chen et al.'s scheme. It is clear that the identity of U can be authenticated.

Step 3: S responds to the message $\{c_3^*, T_s'\}$ to U, where $c_3^* = h_p(c_1^{''} \oplus T_s' \parallel p)$ and T_s' is the current timestamp of the server S. Upon receiving the message from U, the verification of S is achieved. This way, user U and server S have a mutual authentication.

Thus, they obtain a new session key $c_1^{''} = h(r' \oplus b')$ so that the known-key attack happens in this scheme.

2.2.2. *Off-line guessing attack.* In fact, it is intractable for an adversary to attack the cryptosystem by extracting the secret information stored on the smart card. However, Assume that a malicious (legitimate) user U can obtain the parameter p in the login

phase, the malicious user U' can easily achieve an off-line guessing attack by utilizing the parameter p. The attack works as follows:

Step 1: U' can make use of the derived information to guess the long-term key x of the authentication server S, since he can assume the long-term key is x' and then computes $p' = h(ID \oplus x')$.

Step 2: U' can check whether the p' is equal to the derived p. If so, the malicious user U' has correctly guessed the private long-term key of the server S.

Consequently, the malicious user can easily imitate a legal server in the next session. Hence, this scheme has definitely suffered from the risk of this guessing attack.

3. The Proposed Scheme.

In this section, we propose a robust and secure remote user authentication scheme to overcome the weaknesses of Chen et al.'s scheme. Taking computation efficiency into consideration, we execute our proposed scheme by utilizing simple one-way hash functions. There are four phases accordingly and all these phases work as follows:

3.1. Registration phase.

This phase is invoked whenever U initially registers or reregisters with S. Suppose x is the long-term key of the authentication server S. As shown in Figure 3, the following steps are performed in this phase:

Step 1: U chooses a random number b and computes $h(b \oplus PW)$, then sends it with his ID to the server S; $U \Rightarrow S : ID$, $h(b \oplus PW)$.

Step 2: S calculates the following parameters such as: $p = h(ID \oplus x) \parallel h(x)$, $R = p \oplus h(b \oplus PW)$, $V = h_p(h(b \oplus PW))$, and the server S stores the data $\{V, R, h(\cdot), h_p(\cdot)\}$ on a new smart card , and issues the smart cart to user U.

Step 3: U enters b into his smart card so that it contains $\{V, R, h(\cdot), h_p(\cdot), b\}$.

3.2. Login phase.

This phase is depicted in Figure 4. When U intends to login S, the following computations should be performed:

Step 1: U inserts the smart card into the smart card reader and inputs his ID and PW.

Step 2: The smart card computes $p = R \oplus h(b \oplus PW)$ and checks whether $h_p(h(b \oplus PW))$ equals V. If so, the smart card continues to calculate $c_1 = R \oplus h(b \oplus PW)$, $c_2 = h_p(c_1 \parallel T_u)$, where T_u is the current timestamp of U.

Step 3: U sends a login request message to the server S; $U \rightarrow S : \{ID, c_2, T_u\}$.

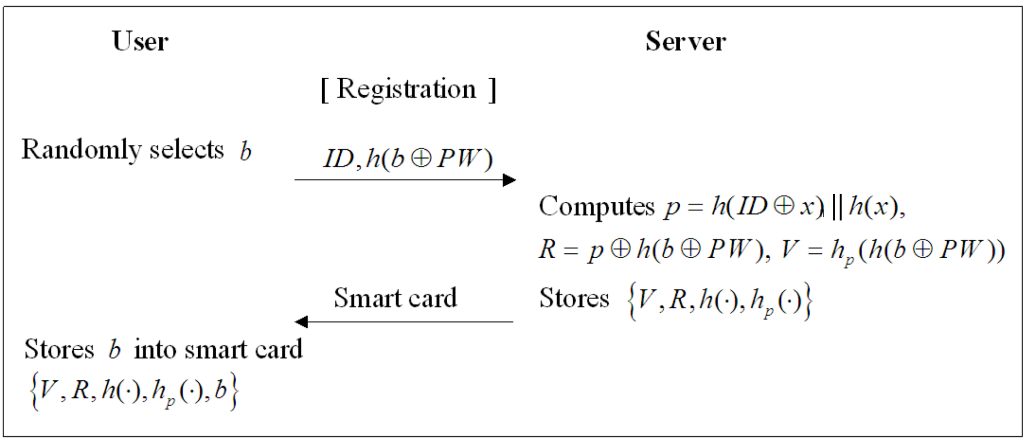

FIGURE 3. Registration phase of our proposed scheme

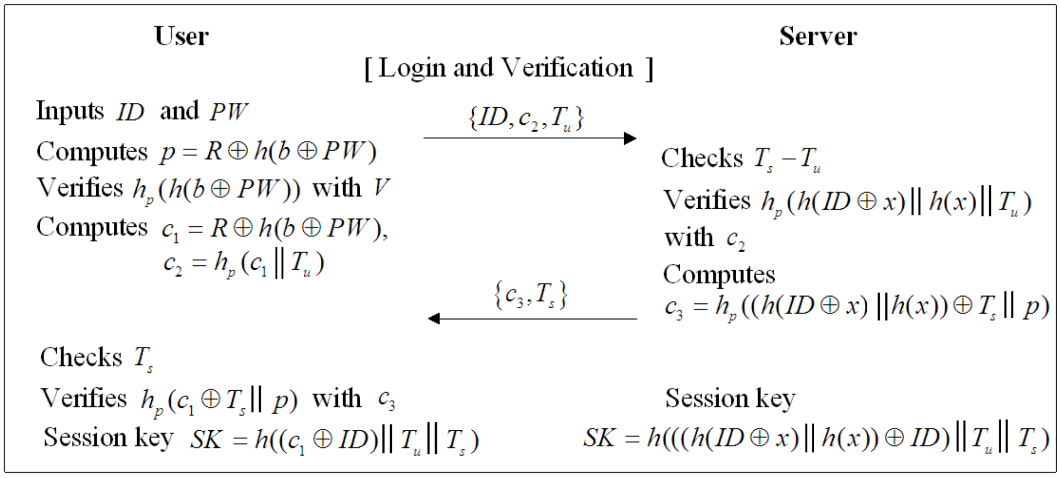

FIGURE 4. Login and verification phase of our proposed scheme

3.3. **Verification phase.** Upon receiving the login request message, the server S and the user U should perform the following steps to achieve mutual authentication and compute a session key. The details of this phase are shown in Figure 4.

Step 1: S checks the validity of ID and whether $T_s > T_u$, where T_s is the current timestamp of the server. If one of them cannot hold, then S rejects the login request; otherwise, S checks whether $T_s - T_u$ is within a valid time interval ΔT. If not, then S rejects the login request.

Step 2: If the $T_s - T_u$ is really within the interval ΔT, S computes $h_p(h(ID \oplus x) \parallel h(x) \parallel T_u)$ in order to check whether the result equals c_2. If so, the validity of U is authenticated and $S \rightarrow U : \{c_3, T_s\}$, where $c_3 = h_p((h(ID \oplus x) \parallel h(x)) \oplus T_s \parallel p)$; otherwise, S rejects the login request.

Step 3: After receiving $\{c_3, T_s\}$, U checks the validity of T_s and whether $T_s > T_u$. If it does not hold, U terminates the connection; otherwise, U checks whether $h_p(c_1 \oplus T_s \parallel p)$ equals the received c_3. If so, the validity of S is authenticated.

Step 4: Moreover, U and S establish a common session key $SK = h((c_1 \oplus ID) \parallel T_u \parallel T_s) = h(((h(ID \oplus x) \parallel h(x)) \oplus ID) \parallel T_u \parallel T_s)$ for private communication.

3.4. **Password change phase.** U can freely change his password PW to PW_{new} in this phase as follows:

Step 1: U inserts the smart card into the smart card reader , inputs his ID and PW and requests to change his password. Thus, the smart card computes $p^* = R \oplus h(b \oplus PW)$, $V^* = h_{p^*}(h(b \oplus PW))$.

Step 2: The smart card checks whether V^* equals the original V stored in the smart card. If so, then U selects a new password PW_{new}; otherwise, the smart card rejects the password change request.

Step 3: The smart card computes $R_{new} = p^* \oplus h(b \oplus PW_{new})$ and $V_{new} = h_{p^*}(h(b \oplus PW_{new}))$, then stores them and replaces the original R and V, respectively.

4. **Security Analysis.** In this section, we present the logic analysis based on BAN logic to prove the authority of authentication procedure and the correctness of our scheme execution. The details will be shown in Subsection 4.1. Then, we discuss several significant attacks and demonstrate the security strength of our proposed scheme in Subsection 4.2.

Moreover, we show that our proposed scheme enhances the security of Chen et al.'s scheme and withstands its corresponding weaknesses.

4.1. **BAN logic demonstration for our proposed scheme.** We use BAN logic to verify our remote user authentication scheme using smart cards. Our scheme not only provides the mutual authentication requirement but also achieves to establish a common session key between the user and the server. According to the analytical procedures of BAN logic, each round of the scheme has to be transformed into the idealized form. Next, we briefly describe basic notations of BAN logic as follows.

$P \overset{K}{\leftrightarrow} Q$: P and Q may communicate with each other using the shared key K. The key K will never be discovered by any principal except P or Q.

$P \overset{X}{\Leftrightarrow} Q$: Formula X is a secretly known only to P and Q. Only P and Q may use X to prove their identities to one another.

$\{X\}_K$: This represents Formula X encrypted under the key K.

$< X >_Y$: This represents Formula X combined with Formula Y.

Then, we provide the following logical postulates to present that U and S can mutually authenticate and cooperate to obtain a session key.

S **believes** ID,

S **believes fresh** (T_u),

U **believes fresh** (T_s),

U **believes** $U \overset{SK}{\leftrightarrow} S$,

U **believes** S **believes** $U \overset{SK}{\leftrightarrow} S$,

S **believes** $U \overset{SK}{\leftrightarrow} S$,

S **believes** U **believes** $U \overset{SK}{\leftrightarrow} S$.

In our scheme, there are two messages that used to achieve the mutual authentication and key agreement requirements. These messages are shown in Figure 4. Then, we idealize the scheme as follows.

Message 1. $U \to S$: ID, $h_p(< c_1 >_{T_u})$, T_u.

Message 2. $S \to U$: $h_p(< p \oplus T_s >_p)$, T_s.

Before starting to analyze our scheme, we first make the following assumptions:

A 1. U **believes** $U \overset{h()}{\Leftrightarrow} S$.

A 2. U **believes** $U \overset{h_p()}{\Leftrightarrow} S$.

A 3. U **believes fresh** (T_s).

A 4. S **believes** (U **controls** ID).

A 5. S **believes** $U \overset{h()}{\Leftrightarrow} S$.

A 6. S **believes** $U \overset{h_p()}{\Leftrightarrow} S$.

A 7. S **believes fresh** (T_u).

A 8. U **believes** (S **controls** $U \overset{SK}{\leftrightarrow} S$).

A 9. S **believes** (U **controls** $U \overset{SK}{\leftrightarrow} S$).

Then, we analyzed the idealized form of our proposed scheme using the above assumptions and rules of BAN logic. Details of the logic proof are presented as follows.

S receives Message 1. The rules show that

S **sees** $\{ID, h_p(< c_1 >_{T_u}), T_u\}$. (Statement 1)

We break conjunctions and produce

S **believes** U **said** ID, (Statement 2)

S **believes** U **said** $h_p(< c_1 >_{T_u})$, (Statement 3)

and

S **believes** U **said** T_u. (Statement 4)

By A 4 and Statement 2, we apply the nonce-verification rule to deduce

S **believes** ID. (Statement 5)

By A 6 and Statement 3, we apply the message-meaning rule to derive

S **believes** U **said** $< c_1 >_{T_u}$. (Statement 6)

By A 7 and Statement 6, the nonce-verification rule applies and yields

S **believes** $< c_1 >_{T_u}$. (Statement 7)

By A 7 and Statement 4, we apply the nonce-verification rule to deduce

S **believes** T_u. (Statement 8)

Then, U receives Message 2. The annotation rule yields that

U **sees** $\{h_p(< p \oplus T_s >_p), T_s\}$. (Statement 9)

We break conjunctions and produce as following:

U **believes** S **said** $h_p(< p \oplus T_s >_p)$, (Statement 10)

and

U **believes** S **said** T_s. (Statement 11)

By A 2 and Statement 10, the message-meaning rule to obtain

U **believes** S **said** $< p \oplus T_s >_p$. (Statement 12)

By A 3 and Statement 12, we apply the nonce-verification rule to deduce

U **believes** $< p \oplus T_s >_p$. (Statement 13)

By A 3 and Statement 11, the nonce-verification rule applies and yields

U **believes** T_s. (Statement 14)

Finally, we apply the message-meaning rule to derive

U **controls** $U \overset{SK}{\leftrightarrow} S$ (Statement 15)

and

S **controls** $U \overset{SK}{\leftrightarrow} S$. (Statement 16)

By A 8 and Statement 16, the jurisdiction rule applies to deduce

U **believes** $U \overset{SK}{\leftrightarrow} S$. (Statement 17)

By A 9 and Statement 15, we apply the jurisdiction rule to derive

S **believes** $U \overset{SK}{\leftrightarrow} S$. (Statement 18)

Based on Statement 7 and Statement 13, we prove our proposed scheme can achieve the mutual authentication requirement. Due to the results of Statement 17 and Statement 18, we also prove our proposed scheme can establish a common session key between U and S.

4.2. **Protection against possible attacks.** In this subsection, we show our proposed scheme can withstand all these possible attacks as follows so that it successfully remedied the security drawbacks of Chen et al.'s scheme.

4.2.1. *The known-key attack.* Chen et al.'s scheme is vulnerable to the known-key attack since an adversary can easily intercept a legal user's login request message $\{ID, c_1, c_2, T_u\}$ and get the parameter c_1, when the $c_1{}'$ has been compromised. Upon getting the parameters, the secret information $p = h(ID \oplus x)$ can be derived by computing $c_1{}' \oplus c_1$. Nevertheless, it is impossible for the adversary to intercept any secret information from the user's login request message in our proposed scheme, since the login request message just includes $\{ID, c_2, T_u\}$. This is because c_1 is protected in the secure one-way hash function belonging to c_2, where $c_2 = h_p(c_1 \parallel T_u)$. Moreover, a legal user's smart card has no need to select random number r to continue the following verification phase. It is no longer possible to reveal any secret information to the adversary. Hence, an adversary cannot obtain validation from the authentication server S. Therefore, we surmount the weakness of Chen et al.'s scheme, because our proposed scheme prevents an adversary from deriving the secret information and sending a fabricated login request message $\{ID, c_2^*, T_u{}'\}$ to obtain a new session key. The know-key attack can be prevented as demonstrated in the above proof.

4.2.2. *Off-line guessing attack.* As aforementioned in Subsection 2.2.2, where a malicious user U' can derive the essential parameter p in Chen et al.'s scheme, it is obvious that the malicious user can premeditate imitating a legal server by guessing the private long-term key x. If the malicious user U' attempts to achieve this purpose in our proposed scheme, he needs to obtain the parameter p in the login phase by calculating $p = R \oplus h(b \oplus PW)$, then execute the operation of an off-line guessing attack. However, after obtaining the parameter p, the malicious user's purpose of off-line guessing attack will fail, because the malicious user U' cannot achieve his purpose by using his own identity ID and the derived parameter p. The reason is that he first assumes a long-term key x' and computes the equation $p' = h(ID \oplus x') \parallel h(x')$. Then, he checks whether the equation equals the original p or not. However, the malicious user U' cannot successfully perform the off-line guessing attack without knowing the hash value $h(x)$. In general, the off-line guessing attack can be achieved because an adversary can guess one part of the secret information by utilizing the other known part. Nevertheless, the parameter p contains the long-term key x and the corresponding hash value $h(x)$ in our proposed scheme so that the malicious user U' cannot guess a correct value of long-term key x' to make $p' = h(ID \oplus x') \parallel h(x')$ equal the original p. Therefore, the off-line guessing attack on Chen et al.'s scheme has been defeated in our proposed scheme.

4.2.3. *Replay attack.* An adversary can intercept either the login request message $\{ID, c_2, T_u\}$ or the response message $\{c_3, T_s\}$ that are transmitted among a legal user U and the authentication server S. However, both of these messages include the corresponding timestamps T_u and T_s, respectively. If the adversary replays his intercepted message, the server S should check the validity of the corresponding ID and T_u. Unfortunately, $T_s - T_u$ cannot be within a valid time interval ΔT. Similarly, it cannot be validated in Step 3 of the verification phase when the adversary might replay the response message $\{c_3, T_s\}$, because he cannot pass the time interval validation. Therefore, the adversary makes such a replay attack very hard.

4.2.4. *Impersonation attack.* An adversary forges a legal user's login request message into $\{ID, c_2{}', T_u{}'\}$ and transmits it to the remote server S. After receiving the message $\{ID, c_2{}', T_u{}'\}$, S should check whether $c_2{}'$ equals the result of $h_p(h(ID \oplus x) \parallel h(x) \parallel T_u{}')$ or not. However, the adversary cannot acquire the value of $h(ID \oplus x) \parallel h(x)$, so that he cannot be validated by the server S in the verification phase. Similarly, it is intractable for the adversary to forge the authentication server S by transmitting an impersonation

response message $\{c_3', T_s'\}$. That is because that the adversary cannot be validated since the equation $c_3' = h_p(c_1 \oplus T_s' \parallel p)$ cannot hold. Meanwhile, the c_1 and p are unavailable parameters for the adversary in our scheme. Therefore, the impersonation attempts of adversaries cannot be achieved.

4.2.5. *Parallel attack.* In Chen et al.'s scheme, an adversary who attempts to masquerade as a legal user U by eavesdropping on communication between the server S and U cannot make a parallel attack among the two different sessions, because c_2 and c_3 have disparate functions. Thus, we inherit this advantage in our proposed scheme, in which the adversary cannot start a new session with server S by sending a fabricated login request message $\{ID, c_3, T_s\}$. Because in Step 2 of the authentication phase, S computes $h_p(h(ID \oplus x) \parallel h(x) \parallel T_u)$ to check whether the result equals the received c_2. However, it is obvious that when $c_3 = h_p((h(ID \oplus x) \parallel h(x)) \oplus T_s \parallel p)$, the result does not equal the value of c_2. Therefore, the adversary cannot make such a parallel attack.

4.2.6. *Perfect forward secrecy.* It is an essential security property to ensure that an adversary cannot derive the session keys used previously, even if he obtains the contiguous knowledge of the long-term key. We assume that the adversary has corrupted a legal user U and acquired the long-term key x. However, $SK = h((c_1 \oplus ID) \parallel T_u \parallel T_s)$ is protected by a one-way hash function, and the equation contains an unavailable value of $c_1 = h(ID \oplus x) \parallel h(x)$ with regards to the adversary. Moreover, due to the different login and authentication processes, the corresponding timestamps T_u and T_s should be updated accordingly. Therefore, there is no way for the adversary to derive the session keys in our scheme. In this way, our proposed scheme can achieve perfect forward secrecy.

5. **Performance and Property Analysis.** In this section, we compare the computation cost with previous works such as Wang et al.'s scheme [16] and Chen et al.'s scheme [17] to estimate the performance of our proposed scheme. The detailed comparison is depicted in Table 1. We note that "Hash" means a one-way hash operation and "Exc" denotes an exclusive-or operation. It is obvious that the computation capability of one-way hash function is most practical in terms of efficiency. In our proposed scheme, we utilize nearly all one-way hash functions to enhance system efficiency and simultaneously remedy the security weaknesses of Chen et al.'s scheme.

Table 1. Performance comparison between our scheme and previous schemes

Items	Wang et al. [16]	Chen et al. [17]	Our scheme
Register phase	$3Hash + 3Exc$	$3Hash + 3Exc$	$4Hash + 3Exc$
Login phase	$4Hash + 5Exc$	$4Hash + 4Exc$	$3Hash + 2Exc$
Verification phase	$4Hash + 5Exc$	$4Hash + 3Exc$	$4Hash + 4Exc$
Password change phase	$4Hash + 4Exc$	$4Hash + 4Exc$	$4Hash + 4Exc$
Total	$15Hash + 17Exc$	$15Hash + 14Exc$	$15Hash + 11Exc$

From the viewpoint of system efficiency, the computation cost of the registration phase in our proposed scheme requires an extra one-way hash operation to calculate the parameter $p = h(ID \oplus x) \parallel h(x)$ so that our remedy is resistant to off-line guessing attack. In the login and verification phases, we utilize only seven one-way hash operations and eight exclusive-or operations which are lower than the computation cost of two comparison targets. Because we try to avoid known-key attack occurring, we don't use the random number r to compute the essential parameter c_1. Note that this step remedy is superior to previous works on computation efficiency. In the password change phase, we require the same computation cost as the other two comparison schemes.

TABLE 2. Property comparison between our scheme and previous schemes

Items	Wang et al. [16]	Chen et al. [17]	Our scheme
P1	Yes	Yes	Yes
P2	Yes	Yes	Yes
P3	No	No	No
P4	No	No	No
P5	Yes	Yes	Yes
P6	Yes	Yes	Yes
P7	No	Yes	Yes
P8	No	No	No

Consequently, we not only achieve the goal of remedying Chen et al.'s security weaknesses but also require lower computation cost totally in our proposed scheme, which compares favorably with the relevant schemes. In Table 2, we show a comparison of the properties we have mentioned in Section 1 between our scheme and the related works. It is obvious that we really propose a novel scheme to remedy the security drawback of Chen et al.'s scheme, and it also satisfies all the above-mentioned properties. In brief, due to the analysis of our proposed scheme, which focuses on the security and performance aspects, our scheme proves to be more secure and efficient than the schemes proposed previously.

6. **Conclusions.** In this paper, we propose a remote user authentication that is novel, has high-level of security, and is efficient for smart cards use. According to the above analysis, we not only enhance Chen et al.'s scheme but also provide evidence that our proposed scheme requires lower computational load than the related works. Moreover, we demonstrate that our new scheme has advanced security features and performance, which have been summarized as properties that distinguished our scheme from previous ones. Therefore, our proposed scheme is more secure and practical for the remote user authentication environment. Meanwhile, we need to develop our system to implement in the progressive applications of wireless communications, such as Wi-Fi, WiMAX, and Mobile networks. Wi-Fi is an advanced technology of its Alliance and the trademark for products that belong to a class of Wireless LAN equipments using the IEEE 802.11 family of standards. WiMAX refers to interoperable implementations of the IEEE 802.16 family standards that provide fixed and mobile Internet access. In the future, we will make further improvements on our scheme to satisfy the different network environments.

Acknowledgment. This work is partially supported by National High Technology and Development Program (863 Program) of China under Grant No. 2011AA010104-2, National Natural Science Foundation of China under Grant 61071076, the Academic Discipline and Postgraduate Education Project of Beijing Municipal Commission of Education, the Fundamental Research Funds for the Central Universities under Grant 2012YJS023. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] T.-C. Wu and H.-S. Sung, Authenticating passwords over an insecure channel, *Computer & Security*, vol.15, no.5, pp.431-439, 1996.

[2] M. Peyravian and N. Zunic, Methods for protecting password transmission, *Computer & Security*, vol.19, no.5, pp.466-469, 2006.

[3] C.-C. Chang, C.-Y. Lee and Y.-C. Chiu, Enhanced authentication scheme with anonymity for roaming service in global mobility networks, *Computer Communications*, vol.32, no.4, pp.611-618, 2009.
[4] L. Lamport, Password authentication with insecure communication, *Communications of the ACM*, vol.24, no.11, pp.770-772, 1981.
[5] M.-S. Hwang and L.-H. Li, A new remote user authentication scheme using smart cards, *IEEE Trans. on Consumer Electronics*, vol.46, no.1, pp.28-30, 2000.
[6] T. ElGamal, A public-key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Trans. on Information Theory*, vol.31, no.4, pp.469-472, 1985.
[7] H. M. Sun, An efficient remote use authentication scheme using smart cards, *IEEE Trans. on Consumer Electronics*, vol.46, no.4, pp.958-961, 2000.
[8] W.-S. Juang, Efficient multi-server password authenticated key agreement using smart cards, *IEEE Trans. on Consumer Electronics*, vol.50, no.1, pp.251-255, 2004.
[9] W.-S. Juang, S.-T. Chen and H.-T. Liaw, Robust and efficient password-authenticated key agreement using smart cards, *IEEE Trans. on Consumer Electronics*, vol.55, no.6, pp.2551-2556, 2008.
[10] S.-K. Kim and M.-G. Chung, More secure remote user authentication scheme, *Computer Communications*, vol.32, no.6, pp.1018-1021, 2009.
[11] J.-Y. Liu, A.-M. Zhou and M.-X. Gao, A new mutual authentication scheme based on nonce and smart card, *Computer Communications*, vol.31, no.10, pp.2205-2209, 2008.
[12] D.-Z. Sun, J.-P. Huai, J.-Z. Sun and J.-X. Li, Cryptanalysis of a mutual authentication scheme based on nonce and smart cards, *Computer Communications*, vol.32, no.6, pp.1015-1017, 2009.
[13] H.-Y. Chien, J.-K. Jan and Y.-M. Tseng, An efficient and practical solution to remote authentication: smart card, *Computer & Security*, vol.21, no.4, pp.372-375, 2002.
[14] W.-C. Ku and S.-M. Chen, Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards, *IEEE Trans. on Consumer Electronics*, vol.50, no.1, pp.204-207, 2004.
[15] 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. on Consumer Electronics*, vol.50, no.2, pp.612-614, 2004.
[16] X.-M. Wang, W.-F. Zhang, J.-S. Zhang and M. K. Khan, Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards, *Computer Standard & Interfaces*, vol.29, no.5, pp.507-512, 2007.
[17] T.-H. Chen, H.-C. Hsiang and W.-K. Shih, Security enhancement on an improvement on two remote user authentication schemes using smart cards, *Future Generation Computer Systems*, vol.27, no.4, pp.377-380, 2011.
[18] C.-K. Chan, Cryptanalysis of a remote user authentication scheme using smart cards, *IEEE Trans. on Consumer Electronics*, vol.46, no.4, pp.992-993, 2000.
[19] R.-C. Wang, W.-S. Juang and C.-L. Lei, Robust authentication and key agreement scheme preserving the privacy of secret key, *Computer Communications*, vol.34, no.3, pp.274-280, 2011.