Computer Standards & Interfaces 31 (2009) 1118–1123

Contents lists available at ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi

# Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment

Han-Cheng Hsiang [*,1], Wei-Kuan Shih

*Department of Computer Science, National Tsing Hua University, No. 101, Kuang Fu Rd, Sec. 2, 300 HsingChu, Taiwan, ROC*

ARTICLE INFO

*Article history:*
Received 27 August 2008
Accepted 13 November 2008
Available online 16 December 2008

*Keywords:*
Authentication
Dynamic ID
Multi-server
Mutual authentication
Password
Smart card

ABSTRACT

Recently, Liao and Wang proposed a secure dynamic ID based remote user authentication scheme for multi-server environment, and claimed that their scheme was intended to provide mutual authentication, two-factor security, replay attack, server spoofing attack, insider and stolen verifier attack, forward secrecy and user anonymity. In this paper, we show that Liao and Wang's scheme is still vulnerable to insider's attack, masquerade attack, server spoofing attack, registration center spoofing attack and is not reparable. Furthermore, it fails to provide mutual authentication. To remedy these flaws, this paper proposes an efficient improvement over Liao–Wang's scheme with more security. The computation cost, security, and efficiency of the improved scheme are well suited to the practical applications environment.

## 1. Introduction

With the rapid growth of network technologies, the remote servers provide resources to be accessed over open network around the world. Mainly via the convenience of the Internet, distant users can share information with each other. In distributed environment, secure communication in insecure communication networks is a very important issue. Hence, user authentication and secret key distribution become the most important security service for communication networks. A common feature of conventional password authentication schemes is that a verification table, which contains the verifiers of users' passwords, should be securely stored in the server. If the verifier is stolen or modified by the adversary, the system will be breached. In 1990, Hwang et al. [1] initially proposed a non-interactive password authentication scheme and its enhanced version, which additionally uses smart cards. Since then many password authentication schemes using smart cards have been proposed [2–11], and each has its advantages and disadvantages. However, all these schemes are designed for the single-server architecture. If conventional password authentication methods are applied to multi-servers environment, each network user does not only need to log into various remote servers repetitively but also need to remember different identifications and passwords for accessing different servers. It is inefficient and easily evokes the compromise of the identities and passwords. Therefore, Lee and Chang proposed a user identification and key distribution scheme in 2000 [12]. Their scheme is based on the difficulty of factorization and hash function, and agrees with the multi-server environment.

Since then, several papers have been devoted to the study of accessing the resources of multi-server environments [12–21]. Among these schemes, based on the computation complexity, the smart card-based authentication schemes are divided into two types, namely hash-based authentication and public-key based authentication.

In 2001, Tsaur proposed a remote user authentication scheme based on RSA cryptosystem and Lagrange interpolating polynomial for multi-server environments [13]. In the same year, Li et al. proposed a remote password authentication scheme by using neural networks [14]. However, it is impractical to spend too much time and cost on training and maintaining neural networks. Later, Lin et al. (2003) proposed a new efficient remote user authentication scheme based on the simple geometric properties of the Euclidean [15]. Many studies have shown the weakness of the above schemes and have proposed new schemes to improve the efficiency and security of the above schemes [16–18]. In 2004, Juang pointed out that Lin et al.'s scheme was not efficient enough for the authentication process, and then proposed an efficient multi-server user authentication and key agreement based on hashing function and symmetric-key cryptosystem [19]. His scheme obtains the shared secret key between the network user and the service provider, and then mitigates the load of each registered server for maintaining the encrypted keys table. However, Juang's scheme neither provides the smart card for the mechanism of checking identity and password in the login phase nor updates user's password without the help of registration center. Thus, it will easily suffer online guessing attack after losing the smart card. Besides, if the secret parameters of the smart card are extracted with some ways [20], Juang's scheme cannot withstand offline dictionary attack and is not reparable. To reduce the computation cost, Chang-Lee (2004) proposed an efficient scheme, which assumes that

[*] Corresponding author.
E-mail address: shc@rtlab.cs.nthu.edu.tw (H.-C. Hsiang).

[1] The author is also with Department of Information Management, Vanung University of Science and Technology, Chungli 320, Taiwan, ROC.

0920-5489/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.csi.2008.11.002

the secret key $x$ of registration center is distributed to each registered server via secure channel [20]. However, the proposed scheme cannot withstand the insider attack, spoofing attack and registration center spoofing attack.

Recently, Liao and Wang proposed a secure dynamic ID based remote user authentication scheme for multi-server environment [22]. The protocol only uses hashing functions to implement a robust authentication scheme for the multi-server environment, and provides a secure method to update password without the help of any third trusted party. They claimed that their scheme was intended to provide mutual authentication, two-factor security, replay attack, server spoofing attack, insider and stolen verifier attack, forward secrecy and user anonymity. Unfortunately, we find that Liao–Wang's scheme is vulnerable to an insider's attack, masquerade attack, server spoofing attack, registration center spoofing attack and is not reparable. Furthermore, it fails to provide mutual authentication. To remedy these flaws, this paper proposes an efficient improvement over Liao–Wang's scheme with more security.

The rest of this paper is organized as follows. In Section 2, a brief review of Liao–Wang's scheme is given. In Section 3, describes a cryptanalysis of Liao–Wang's scheme. Section 4 shows the details of the proposed scheme. Section 5 makes the security analysis of the proposed scheme. Section 6 compares the performance and functionality of the proposed scheme with the related schemes. Finally, some concluding remarks are made in the last section.

## 2. Review of Liao–Wang's scheme

We first review Liao–Wang's scheme [22] before demonstrating its weaknesses. The notations used in Liao–Wang's scheme are summarized in Table 1. Their scheme only uses simple hashing functions to complete the mutual authentication and session key agreement and introduces dynamic ID to achieve user's anonymity [7]. Besides, their scheme is a nonce-based scheme to avoid the time-synchronization problem. Consider the multi-server environment containing three participants, the user ($U_i$), the service provider ($S_j$) and the registration center ($RC$).

It is assumed that $RC$ is a trusted party responsible for the secret keys distribution between $U_i$ and $S_j$. $RC$ chooses the master secret key $x$ and a secret number $y$ to distribute among the involved parties via secure channel. Let $ID_i$ denote a unique identification of $U_i$ and $SID_j$ denotes a unique identification of $S_j$. There are four phases in Liao–Wang's scheme — the registration phase, the login phase, the mutual verification session key agreement phase and the password change phase. Different phases of work are described as follows:

### 2.1. Registration phase

This phase is invoked whenever the user $U_i$ wants to access the resources of the service provider $S_j$, he has to submit his identity $ID_i$ and password $PW_i$ to $RC$. Then, $RC$ performs the following steps:

Step R1. $U_i$ freely selects a password $PW_i$.
Step R2. $U_i \Rightarrow RC$: $ID_i$, $PW_i$.
Step R3. $RC$ computes $T_i = h\ (ID_i||x)$, $V_i = T_i \oplus h\ (ID_i||PW_i)$, $B_i = h\ (PW_i) \oplus h(x)$ and $H_i = h(T_i)$.
Step R4. $S \Rightarrow U$: a smart card containing ($V_i$, $B_i$, $H_i$, $h(\cdot)$, $y$).

### 2.2. Login phase

When $U_i$ wants to login $S$, he keys his identity $ID_i$, password $PW_i$ and the server identity $SID_j$ in order to login the service provider $S_j$, and then the smart card performs the following steps:

Step L1. $U_i$'s smart card performs the following computations:

$T_i^* = V_i \oplus h(ID_i||PW_i)$ and $H_i^* = h(T_i^*)$. Then checks whether $H_i^*$ and $H_i$ is equal or not. If yes, the legitimacy of the user can be assured and proceeds to the next step; otherwise, reject the login request.

**Table 1**
Notations.

| Notation | Meaning |
|---|---|
| $U_i$ | The $i_{\text{th}}$ user |
| $S_j$ | The $j_{\text{th}}$ server |
| $RC$ | The registration center |
| $ID_i$ | The identification of $U_i$ |
| $PW_i$ | The password of $U_i$ |
| $SID_j$ | The identification of $S_j$ |
| $CID_i$ | The dynamic ID of $U_i$ |
| $h(\cdot)$ | A secure one-way hash function |
| $x$ | The secret key maintained of registration center |
| $\oplus$ | Exclusive- or operation |
| $\|\|$ | String concatenation operation |
| $\Rightarrow$ | A secure channel. |
| $\rightarrow$ | A common channel. |

Step L2. Generate nonce $N_i$ and compute ($CID_i$, $P_{ij}$, $Q_i$) in accordance with the following equations:

$$CID_i = h(PW_i) \oplus h(T_i |\ y |\ N_i)$$
$$P_{ij} = T_i \oplus h\left(y |\ N_i |\ SID_j\right)$$
$$Q_i = h(B_i |\ y |\ N_i).$$

Step L3. $U \rightarrow S_j$: $CID_i$, $P_{ij}$, $Q_i$, $N_i$.

### 2.3. Mutual verification and session key agreement phase

Upon receiving the login request message {$CID_i$, $P_{ij}$, $Q_i$, $N_i$}, the service provider $S_j$ authenticates the user $U_i$ with the following steps:

Step V1. Compute $T_i = P_{ij} \oplus h(y||N_i||SID_j)$, $h(PW_i) = CID_i \oplus h(T_i||y||N_i)$ and $B_i = h(PW_i) \oplus h(x)$.
Step V2. Compute $h(B_i||N_i||y)$, and then compares it with $Q_i$. If they are not equal, the server $S_j$ rejects the login request and terminates this session.
Step V3. Generate nonce $N_j$ and computes $M'_{ij} = h(B_i||N_i||y||SID_j)$, and then send back the message ($M'_{ij}$, $N_j$) to the user $U_i$.

Upon receiving the acknowledgement message ($M'_{ij}$, $N_j$), the user $U_i$ performs the following steps:

Step V4. Computes $h(B_i||N_i||y||SID_j)$ and compares it with $M'_{ij}$. If they are equivalent, it indicates that the legality of the service provider $S_j$ is authenticated; otherwise, the connection is interrupted.
Step V5. Computes $M_{ij}'' = h(B_i||N_j||y||SID_j)$, and then sends back $M_{ij}''$ to the service provider $S_j$.

Upon receiving the message $M_{ij}''$, the service provider $S_j$ responds to the following step:

Step V6. Compute $h(B_i||N_j||y||SID_j)$ and compares it with $M_{ij}''$. If it is held, the identity of $U_i$ can be assured.

After finishing mutual authentication phase, the user $U_i$ and the service provider $S_j$ computes $h(B_i||N_i||N_j||y||SID_j)$ as the session key $SK$.

### 2.4. Password change phase

This phase is invoked whenever $U$ wants to change his password $PW$ with a new one, say $PW_{\text{new}}$.

When the user $U_i$ wants to update his password without the help of RC, he inserts his smart card to card reader and inputs ($ID_i$, $PW_i$) corresponding to the smart card.

Step C1. $U_i$ inserts his smart card into the smart card reader, enters $ID_i$, $PW_i$, and requests to change password.

Step C2. Upon receiving the request of changing passwords and $ID_i$, $PW_i$, $U_i$'s smart card performs the following computations:

$T_i^* = V_i \oplus h(ID_i||PW_i)$ and $H_i^* = h(T_i^*)$. Then checks whether $H_i^*$ and $H_i$ is equal or not. If not, the smart card rejects the password change request. Otherwise $U_i$ chooses a new password $PW_{\mathrm{inew}}$.

Step C3. $U_i$'s smart card computes $V_{\mathrm{inew}} = T_i \oplus h(ID_i||PW_{\mathrm{inew}})$ and $B_{\mathrm{inew}} = B_i \oplus h(PW_i) \oplus h(PW_{\mathrm{inew}})$. The parameter $V_{\mathrm{inew}}$, $B_{\mathrm{inew}}$ is stored in the smart card to replace $V_i$, $B_i$ respectively.

## 3. Cryptanalysis of Liao–Wang's Scheme

In this section, we will show that Liao–Wang's scheme is vulnerable to insider attack, masquerade attack, poor reparability and fails to provide Mutual authentication. Besides, their scheme has the problems of server spoofing and registration center spoofing. It shows some issues in the scheme, if a user loses his/her smart card and it is found out by an attacker or an attacker steals user's smart card, and extracts the stored values through some ways [21], then the attacker can easily impersonate legitimate user $U_i$ without knowing any password. Obviously, Liao–Wang's scheme does not provide the two-factor security which they claimed.

### 3.1. Poor reparability

Although the tamper resistance of smart cards was widely assumed in their applications, such an assumption may be difficult in practice. Many researches have shown that the secrets stored in a smart card can be breached by analyzing the leaked information, or monitoring the power consumption [21].

We shall prove that Liao–Wang's scheme cannot withstand the masquerade attack and poor reparability. The adversary can masquerade as the legal user to login the remote system without knowing the password $PW_i$, if the adversary has obtained the $y$ and $B_i$ stored in the user's smart card. He can forge a login message that can pass $S_j$'s authentication as follows.

Suppose that the adversary has intercepted the message transmitted in Step L3, i.e., $\{CID_i, P_{ij}, Q_i, N_i\}$, during one of $U_i$'s past logins. Then, the adversary can compute

$$T_i = P_{ij} \oplus h\left(y|\ N_i|\ SID_j\right),$$
$$h(PW_i) = CID_i \oplus h(T_i|\ y|\ N_i).$$

Once the adversary obtains $T_i$ and $h(PW_i)$, he first randomly chooses a nonce, $N_{\mathrm{a}}$, and computes $CID_i^* = h(PW_i) \oplus h(T_i||y||N_{\mathrm{a}})$, $P_{ij}^* = T_i \oplus h(y||N_{\mathrm{a}}||SID_j)$ and $Q_i^* = h(B_i||y||N_{\mathrm{a}})$. Then sends an imitative login message $\{CID_i^*, P_{ij}^*, Q_i^*, N_{\mathrm{a}}\}$ to the service provider $S_j$.

When $S_j$ receives $\{CID_i^*, P_{ij}^*, Q_i^*, N_{\mathrm{a}}\}$, he will compute $T_i = P_{ij}^* \oplus h(y||N_{\mathrm{a}}||SID_j)$, $h(PW_i) = CID_i^* \oplus h(T_i||y||N_{\mathrm{a}})$ and $B_i = h(PW_i) \oplus h(x)$.

Next, $S_j$ computes $h(B_i||N_i||y)$, and then compares it with $Q_i^*$. Because the computed result equals the received $Q_i^*$, $S_j$ will accept the adversary's login request.

However, since $y$ is commonly used for all users rather than specifically used for only $U_i$, such a deception cannot be forbidden unless $S_j$ replaces $y$ with a new one and updates the $y$ stored in the smart card of each user. It will be unreasonable and inefficient if $y$ should be changed to recover the security of $U_i$ only. Obviously, the reparability [2] of Liao–Wang's scheme is poor.

### 3.2. Masquerade attack

We shall prove that Liao–Wang's scheme cannot withstand the other masquerade attacks, if an adversary Eve is a legal user of the system. The adversary Eve can masquerade as any legal user $U_i$ to login the remote server $S_j$ without knowing the password $PW_i$ at anytime. He can forge a login message that can pass $S_j$'s authentication. A more detailed description of the attack is as follows:

Suppose that Eve has intercepted the message transmitted in Step L3, i.e., $\{CID_i, P_{ij}, Q_i, N_i\}$, during one of $U_i$'s past logins. Since Eve is a legal user, he can obtain $y$ and $B_{\mathrm{e}}$ from his smart card. Then, Eve can compute $h(x) = h(PW_{\mathrm{e}}) \oplus B_{\mathrm{e}}$, which $PW_{\mathrm{a}}$ is Eve's password.

Next, Eve computes

$$T_i = P_{ij} \oplus h\left(y||N_i||SID_j\right),$$
$$h(PW_i) = CID_i \oplus h(T_i||y||N_i)$$

and

$$B_i = h(PW_i) \quad h(x).$$

Once Eve obtains $T_i$, $h(PW_i)$ and $B_i$, he first randomly chooses a nonce $N_{\mathrm{a}}$, and computes $CID_i^* = h(PW_i) \oplus h(T_i||y||N_{\mathrm{a}})$, $P_{ij}^* = T_i \oplus h(y||N_{\mathrm{a}}||SID_j)$ and $Q_i^* = h(B_i||y||N_{\mathrm{a}})$. Then sends an imitative login message $\{CID_i^*, P_{ij}^*, Q_i^*, N_{\mathrm{a}}\}$ to the service provider $S_j$.

When $S_j$ receives $\{CID_i^*, P_{ij}^*, Q_i^*, N_{\mathrm{a}}\}$, he will compute $T_i = P_{ij}^* \oplus h(y||N_{\mathrm{a}}||SID_j)$, $h(PW_i) = CID_i^* \oplus h(T_i||y||N_{\mathrm{a}})$ and $B_i = h(PW_i) \oplus h(x)$.

Next, $S_j$ computes $h(B_i||N_{\mathrm{a}}||y)$, and then compares it with $Q_i^*$. Because the computed result equals the received $Q_i^*$, $S_j$ will accept the adversary's login request.

Eve also only can replay the intercepted message $\{CID_i, P_{ij}, Q_i, N_i\}$ to $S_j$. Since Liao–Wang's scheme does not check the transmitted nonce value $N_i$. The replayed login request will pass $S_j$'s authentication because Eve knows $B_i$ and $y$, he can compute $M_{ij}'' = h(B_i||N_j||y||SID_j)$ to respond to the server $S_j$. So he will pass $S_j$'s authentication.

### 3.3. Server spoofing and registration center spoofing attacks

In Liao–Wang's scheme, service provider $S_j$ only need to know $y$ and $h(x)$ for verifying legitimacy of the users. As mentioned above, if the adversary Eve is a legal user, he also can impersonate as any service provider $S_j$ to cheat $U_i$, since he has known $y$ and $h(x)$, and can construct the session key $SK$ with the knowledge of $B_i$, $y$ and $h(x)$. After communicating with the masqueraded service provider, the legal user $U_i$ will be fooled into believing Eve as the legitimate server and establishing the session key with her, which implies that all the messages can be decrypted by Eve. Hence, Liao–Wang's scheme is vulnerable to server spoofing attack.

Besides, Liao and Wang's multi-server authentication scheme assumes that the registration center $RC$ and all system servers are trustworthy, and $RC$ chooses the master secret key $x$ and a secret number $y$ to distribute amongst the involved parties via secure channel. Sharing the same secret key enables each server to masquerade as other servers or as the registration center $RC$ in the real network environments. The multi-server authentication scheme therefore is vulnerable to security breaches arising from server spoofing attack and registration center spoofing attack. Obviously, Liao–Wang's scheme does not suffer from server spoofing attack and registration spoofing attack.

### 3.4. Insider's attack

In real environments, it is likely that the user uses the same password to access several servers for his convenience. If a privileged insider of $RC$ has learned the user's password, he may try to impersonate $U_i$ to access other servers. In Step R1 of the registration phase, $U_i$'s password $PW_i$ will be revealed to $RC$ because it is transmitted directly to $RC$. Then, the privileged insider of $RC$ may try to access the servers outside this system. If the targeted outside server adopts the normal password authentication scheme, it is possible that the privileged insider of RC can successfully impersonate $U_i$ to login it by using $PW_i$. Although it is also possible that all the privileged insiders of $RC$ are trusted and $U_i$ does not use the same password to access several servers, the implementers and the users of the scheme should be aware of such a potential weakness. For this reason, in several password authentication schemes [9–11], the user's password is not exposed to others including the registration center and the servers. Clearly, Liao–Wang's scheme is vulnerable to an insider attack.

### 3.5. Failure to provide mutual authentication

In Liao–Wang's scheme, if a legal user $U_i$ wants to login, the service provider $S_j$, $U_i$ cannot pass $S_j$'s authentication anytime. Since the scheme uses a wrong computation in Srep V2 of the mutual verification and session key agreement phase for verifying $U_i$'s login request, it will never finish the mutual authentication. The more detailed description of the security flaw is as follows:

In normal condition, a legal user $U_i$ wants to login the service provider $S_j$, $U_i$ sends the login message $\{CID_i, P_{ij}, Q_i, N_i\}$ to $S_j$. After receiving $U_i$'s login request, the $S_j$ performs the following steps.

1. Compute $T_i = P_{ij} \oplus h(y||N_i||SID_j)$, $h(PW_i) = CID_i \oplus h(T_i||y||N_i)$ and $B_i = h(PW_i) \oplus h(x)$.
2. Compute $Q_i' = h(B_i||N_i||y)$, and then compares it with $Q_i$. Since $Q_i = h(B_i||y||N_i)$ and $Q_i' = h(B_i||N_i||y)$, they are not equal. The server $S_j$ will reject the login request and terminate this session.

Obviously, Liao–Wang's scheme fails to provide mutual authentication.

## 4. The improved scheme

In this section, we propose an improvement on Liao–Wang's scheme, which keeps the merits of original scheme and can withstand the security flaws described in previous section. To resist the server spoofing attack and the registration center spoofing attack inherent in Liao–Wang's scheme, we assumed that only $RC$ knows the master secret key $x$ and two secret numbers $r$ and $y$. When the registration center $RC$ permits the entry of a service provider $S_j$, $RC$ uses $SID_j$ to compute the shared secret key $h(SID_j||y)$, and sends $h(SID_j||y)$ to $S_j$ via the secure channel. This shared key is used to confirm the legitimacy of the service provider and the registration center.

There are four phases in our improved scheme — the registration phase, the login phase, the mutual verification session key agreement phase and the password change phase. Different phases of work are described as follows:

### 4.1. Registration phase

This phase is invoked whenever the user $U_i$ wants to access the resources of the service provider $S_j$, he has to submit his identity $ID_i$ and password $PW_i$ to $RC$. Then, $RC$ performs the following steps:

Step R1. $U_i$ freely selects a password $PW_i$, a arbitrary number $b$ and computes $h(b \oplus PW_i)$.
Step R2. $U_i \Rightarrow RC$: $ID_i$, $h(b \oplus PW_i)$.
Step R3. $RC$ computes $T_i = h(ID_i||x)$, $V_i = T_i \oplus h(ID_i||h(b \oplus PW_i))$, $A_i = h(h(b \oplus PW_i)||r) \oplus h(x \oplus r)$, $B_i = A_i \oplus h(b \oplus PW_i)$, $R_i = h(h(b \oplus PW_i)||r)$ and $H_i = h(T_i)$.
Step R4. $S \Rightarrow U$: a smart card containing $(V_i, B_i, H_i, R_i, h(\cdot))$.
Step R5. $U$ enters $b$ into his smart card.

Note that $U$'s smart card contains $V_i$, $B_i$, $b$, $H_i$, $R_i$ and $h(\cdot)$.

### 4.2. Login phase

When $U_i$ wants to login $S_j$, he keys his identity $ID_i$, password $PW_i$ and the server identity $SID_j$ in order to login the service provider $S_j$, and then the smart card performs the following steps:

Step L1. $U$'s smart card performs the following computations:

$T_i = V_i \oplus h(ID_i||h(b \oplus PW_i))$ and $H_i^* = h(T_i)$

Then checks whether $H_i^*$ and $H_i$ is equal or not. If yes, the legitimacy of the user can be assured and proceeds to the next step; otherwise, rejects the login request.

Step L2. Generate nonce $N_i$ and compute $(CID_i, P_{ij}, Q_i, V_i)$ in accordance with the following equations:

$$\begin{aligned}
&A_i = B_i \oplus h(b \oplus PW_i) \\
&CID_i = h(b \oplus PW_i) \oplus h(T_i||A_i||N_i) \\
&P_{ij} = T_i \oplus h\left(A_i||N_i||SID_j\right) \\
&Q_i = h(B_i||A_i||N_i) \\
&D_i = R_i \oplus SID_j \oplus N_i \\
&C_0 = h\left(A_i||N_i + 1||SID_j\right).
\end{aligned}$$

Step L3. $U \rightarrow S_j$: $CID_i$, $P_{ij}$, $Q_i$, $D_i$, $C_0$, $N_i$.

### 4.3. Mutual verification and session key agreement phase

Upon receiving the login request message $\{CID_i, P_{ij}, Q_i, D_i, C_0, N_i\}$, the service provider $S_j$ authenticates the user $U_i$ with the following steps:

Step V1. $S_j$ generates nonce $N_{jr}$ and computes $M_{jr} = h(SID_j||y) \oplus N_{jr}$, then sends the message $\{M_{jr}, SID_j, D_i, C_0, N_i\}$ to the registration center $RC$.

Upon receiving the message $\{M_{jr}, SID_j, D_i, C_0, N_i\}$, $RC$ performs the following step:

Step V2a. $RC$ computes $N_{jr}' = M_{jr} \oplus h(SID_j||y)$, $R_i' = D_i \oplus SID_j \oplus N_i$ and $A_i' = R_i' \oplus h(x \oplus r)$.
Step V2b. Compute $C_0' = h(A_i'||N_i + 1||SID_j)$, and then compares it with $C_0$. If they are not equal, the registration center $RC$ rejects the authentication request and terminates this session.
Step V2c. $RC$ generates nonce $N_{rj}$ and computes $C_1 = h(N_{jr}'||h(SID_j||y)||N_{rj})$, $C_2 = A_i \oplus h(h(SID_j||y) \oplus N_{jr}')$, then sends back the message $\{C_1, C_2, N_{rj}\}$ to $S_j$.

Upon receiving the send back message $\{C_1, C_2, N_{rj}\}$, the service provider $S_j$ performs the following steps:

Step V3. Compute $C_1' = h(N_{jr}||h(SID_j||y)||N_{rj})$, and then compares it with $C_1$. If they are not equal, the server $S_j$ reports a RC authentication error message and terminates this session.
Step V4. Compute $A_i = C_2 \oplus h(h(SID_j||y) \oplus N_{rj})$, $T_i = P_{ij} \oplus h(A_i||N_i||SID_j)$, $h(b \oplus PW_i) = CID_i \oplus h(T_i||A_i||N_i)$ and $B_i = A_i \oplus h(b \oplus PW_i)$.
Step V5. Compute $h(B_i||A_i||N_i)$, and then compares it with $Q_i$. If they are not equal, the server $S_j$ rejects the login request and terminates this session.
Step V6. Generate nonce $N_j$ and computes $M_{ij}' = h(B_i||N_i||A_i||SID_j)$, and then sends back the message $(M_{ij}', N_j)$ to the user $U_i$.

Upon receiving the acknowledgement message $(M_{ij}', N_j)$, the user $U_i$ performs the following steps:

Step V7. Compute $h(B_i||N_i||A_i||SID_j)$ and compare it with $M_{ij}'$. If they are equal, it indicates that the legitimacy of the service provider $S_j$ is authenticated; otherwise, the connection is interrupted.
Step V8. Compute $M_{ij}'' = h(B_i||N_j||A_i||SID_j)$, and then send back $M_{ij}''$ to the service provider $S_j$.

Upon receiving the message $M_{ij}''$, the service provider $S_j$ responds to the following step:

Step V9. Compute $h(B_i||N_j||A_i||SID_j)$ and compare it with $M_{ij}''$. If it is on hold, the identity of $U_i$ can be assured.

After finishing mutual authentication phase, the user $U_i$ and the service provider $S_j$ compute $h(B_i\ ||A_i||N_i||N_j\ ||SID_j)$ as the session key $SK$.

### 4.4. Password change phase

This phase is invoked whenever $U$ wants to change his password $PW$ with a new one, say $PW_{\text{new}}$.

When the user $U_i$ wants to update his password without the help of $RC$, he inserts his smart card to card reader and inputs $(ID_i, PW_i)$ corresponding to the smart card.

Step C1. $U$ inserts his smart card into the smart card reader, enters $ID_i$, $PW_i$, and requests to change password.

Step C2. Upon receiving the request of changing passwords and $ID_i$, $PW_i$, $U_i$'s smart card performs the following computations:

$$T_i = V_i \quad h(ID_i||h(b\oplus PW_i)) \text{ and } H_i^* = h(T_i)$$

Then checks whether $H_i^*$ and $H_i$ is equal or not. If not, the smart card rejects the password change request. Otherwise $U$ chooses a new password $PW_{\text{inew}}$.

Step C3. $U_i$'s smart card computes $V_{\text{inew}} = T_i \oplus h(ID_i||h(b\oplus PW_{\text{inew}}))$ and $B_{\text{inew}} = B_i \oplus h(b\oplus PW_i)\oplus h(b\oplus PW_{\text{inew}})$. The parameter $V_{\text{inew}}$, $B_{\text{inew}}$ is stored in the smart card to replace $V_i$, $B_i$ respectively.

## 5. Security analysis

In this section, we will only discuss the enhanced security and efficiency of our improved scheme. The others are the same as original Liao–Wang's scheme in literature [22].

### 5.1. Masquerade attacks resistance and reparability

The adversary cannot masquerade as the legal user to login the remote server $S_j$ if the adversary has the user's smart card, and obtains the $V$, $B_i$, $b$, $R_i$, $H_i$, and $h(\cdot)$ stored in the user's smart card. He cannot breach $r$ or the secret key $x$ from stolen smartcard or intercept the information. The adversary couldn't forge a login message to pass $S_j$'s authentication, because he does not know $x$, $r$ and $U_i$'s password $PW_i$. He cannot use the $V$, $B_i$, $b$, $R_i$, $H_i$, and $h(\cdot)$ that he has obtained without knowing the $PW_i$, $x$ and $r$ to compute the correct values of $A_i$, $T_i$ and $B_i$.

Besides, if an adversary Eve is a legal user of the system, the adversary Eve cannot masquerade as any legal user $U_i$ to login the remote server $S_j$ without knowing the password $PW_i$. He cannot forge a login message that can pass $S_j$'s authentication. A more detailed description of the attack is as follows:

Suppose that Eve has intercepted the message transmitted in Step L3, i.e., $\{CID_i, P_{ij}, Q_i, V_i, N_i\}$, during one of $U_i$'s past logins. Although Eve is a legal user, he can only obtain $B_e$ from his smart card. Eve cannot conduct $h(x)$ from the $V$, $B_e$, $b$, $H_i$, and $h(\cdot)$ in his smart card. So he cannot compute the correct values of $A_i$, $T_i$ and $B_i$ from the intercepted message $\{CID_i, P_{ij}, Q_i, V_i, N_i\}$ without knowing $PW_i$ and $h(x)$.

Thus, in the improved scheme, the adversary cannot masquerade as the legal user to access the remote system. Therefore, the improved scheme is reparable and can resist the masquerade attacks.

### 5.2. Server spoofing and registration center spoofing

As mentioned above, if the adversary Eve is a legal user, he cannot impersonate as any service provider $S_j$ to cheat $U_i$, since he cannot construct the session key $SK$ with the knowledge of $B_i$, $A_i$. After communicating with the masqueraded service provider, the legal user can detect immediately and terminate the session. Hence, our improved scheme can protect the user from cheating by the masqueraded service provider.

Besides, it is impossible for an attacker to masquerade as the server to cheat a legal user or the registration center. Because none of the servers store any user authentication key in it, and none of the servers know $h(x)$ and $h(x\oplus r)$. If the server wants to authenticate any user, the server must be authenticated by the registration center first, and then obtain the user authentication key $A_i$ from the registration center. If an adversary wishes to cheat the registration center, he must have $h(SID_j||y)$. If the adversary wishes to masquerade as $S_j$ without knowing $h(SID_j||y)$, it will be detected immediately and terminate the session in steps V2a and V2b of the mutual verification and session key agreement phase.

Similarly, if an adversary wishes to masquerade at the registration center to cheat the server, it will be infeasible because each server $S_j$ has a $h(SID_j||y)$. The server can use $h(SID_j||y)$ to verify the registration center in Step V3 of the Mutual verification and session key agreement phase.

Thus, the improved scheme can prevent server spoofing attack and registration center spoofing attack.

### 5.3. Insider's attack resistance

In our improved scheme, $U_i$ registers to $S_j$ by presenting $h(b\oplus PW_i)$ instead of $PW_i$, the insider of $RC$ cannot directly obtain $PW_i$. Moreover, as $b$ is not revealed to $RC$, the insider of $RC$ cannot obtain $PW_i$ by performing an off-line guessing attack on $h(b\oplus PW_i)$. The improved scheme also does not maintain any verifier table. Thus, the improved scheme can resist the insider attack [10].

## 6. Performance and functionality analysis

In this section, we summarize some performance issues of our improved scheme. We compare the improved scheme with the related schemes in terms of computation costs. We mainly focus on the computations of login and verification phases since the two phases are the main body of an authentication scheme. To analyze the computational complexity of the schemes, we define the notation $T_{\text{h}}$ and $T_{\text{sym}}$ as the time complexity for hashing function and symmetric encryption/decryption. Because exclusion-OR operation requires very few computations, it is usually neglected considering its computational cost. Under the above assumptions, the time complexity associated with the different operations can be roughly expressed as $T_{\text{sym}} > T_{\text{h}}$.

Table 2 shows the performance comparison of our scheme and the related schemes. The computation cost of the user and the service provider is defined by the time spent by the user and the service provider in the process of authentication. According to the above definition, in Liao–Wang's scheme, both the computation cost of the user and that of the service provider are $9T_{\text{h}}$ and $6T_{\text{h}}$ respectively. In our improved scheme, both the computation costs of the user and the service provider are $9T_{\text{h}}$ and $7T_{\text{h}}$ respectively. In addition, our scheme can achieve the mutual authentication between server and the registration center in mutual verification and session key agreement phase, which requires four extra hash operations. This is because our scheme provides two-variant hashing operations for resisting insider attack, masquerade attack, poor reparability, server spoofing attack, registration center spoofing attack and advance collision attacks [23,24]. Even if all the information stored in smart card or transmitted via insecure channel is extracted by adversary, the improved scheme is still secure. Besides, our scheme can achieve mutual authentication. Obviously, it is worth achieving such a high level of security at the cost of only five extra hashing operations.

**Table 2**
Performance comparison of our scheme and the related schemes.

| | | Ours | Liao–Wang | Chang-Lee | Jung |
|---|---|---|---|---|---|
| Communication cost of authentication | User | $9T_{\text{h}}$ | $9T_{\text{h}}$ | $4T_{\text{h}}+3T_{\text{sym}}$ | $3T_{\text{h}}+3T_{\text{sym}}$ |
| | Server | $7T_{\text{h}}$ | $6T_{\text{h}}$ | $4T_{\text{h}}+3T_{\text{sym}}$ | $3T_{\text{h}}+6T_{\text{sym}}$ |
| | *RC* | $4T_{\text{h}}$ | 0 | 0 | $1T_{\text{h}}+2T_{\text{sym}}$ |
| Total | | $20T_{\text{h}}$ | $15T_{\text{h}}$ | $8T_{\text{h}}+6T_{\text{sym}}$ | $7T_{\text{h}}+11T_{\text{sym}}$ |

**Table 3**
Comparison between the related schemes and our scheme.

| | Ours | Liao–Wang | Chang-Lee | Jung |
|---|---|---|---|---|
| Single registration | Yes | Yes | Yes | Yes |
| No verification table | Yes | Yes | Yes | Yes |
| Mutual authentication | Yes | No | Yes | Yes |
| Securely chosen password | Yes | Yes | No | No |
| Session key agreement | Yes | Yes | Yes | Yes |
| User's anonymity | Yes | Yes | No | No |
| No time synchronization | Yes | Yes | Yes | Yes |
| Two factor security | Yes | No | No | No |
| Prevention of registration center spoofing | Yes | No | No | Yes |
| Prevention of server spoofing | Yes | No | No | Yes |

Moreover, we summarize the functionality of the proposed scheme and make comparisons with some related schemes in Table 3. It demonstrates that our schemes can achieve the essential requirements for a secure and efficient remote user authentication for multi-server environment [19].

## 7. Conclusion

We have analyzed the security of Liao–Wang's secure dynamic ID based remote user authentication scheme for multi-server environment. Although Liao and Wang claimed their scheme can resist against various known attacks, we have shown that the scheme is indeed completely vulnerable to insider's attack, masquerade attacks, server spoofing attack, registration center spoofing attack and is not reparable. In Liao–Wang's scheme, the adversary can easily impersonate any user to log into the server without knowing the legal user's password at any time, if the legal user's smart card was lost or if the adversary also is a legal user. Besides, it fails to provide mutual authentication. Hence, we have proposed an improved scheme that addresses the known security problems. Compared with the Liao–Wang's scheme, the proposed scheme inherits the merits, enhances the security, and can achieve mutual authentication. In addition, the proposed scheme will avoid the adversary to breach the secret key from stolen smartcard or intercept the information when the smartcard was carelessly lost. Therefore, the proposed scheme is well suited to the practical applications environment.

## References

[1] T. Hwang, Y. Chen, C.S. Laih, Non-interactive password authentication without password tables, IEEE Region 10 Conference on Computer and Communication System, vol. 1, Sept. 1990, pp. 429–431.
[2] T. Hwang, W.C. Ku, Reparable key distribution protocols for Internet environments, IEEE Trans. Consum. Electron. 43 (5) (1995) 1947–1949.
[3] H.M. Sun, An efficient remote user authentication scheme using smart cards, IEEE Trans. Consum. Electron. 46 (4) (2000) 958–961.
[4] J.J. Shen, C.W. Lin, M.S. Hwang, A modified remote user authentication scheme using smart cards, IEEE Trans. Consum. Electron. 49 (2) (2003) 414–416.
[5] Amit K. Awashti, SunderLal , An enhanced remote user authentication scheme using smart cards, IEEE Trans. Consum. Electron. 50 (2) (2004) 583–586.
[6] C. Chang, K.F. Hwang, Some forgery attacks on a remote user authentication scheme using smart cards, Informatics 14 (3) (2003) 289–294.
[7] M.L. Das, A. Saxena, V.P. Gulati, A dynamic ID-based remote user authentication scheme, IEEE Trans. Consum. Electron. 50 (2) (2004) 629–631.
[8] W.C. Ku, S.T. Chang, Impersonation attack on a dynamic ID-based remote user authentication scheme using smart cards, IEICE Trans. Commun. 5 (2005) 2165–2167.
[9] M.S. Hwang, C.C. Lee, Y.L. Tang, A simple remote user authentication scheme, Math. Comput. Model. 36 (1-2) (2002) 103–107.
[10] W.C. Ku, S.M. Chen, Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards, IEEE Trans. Consum. Electron. 50 (1) (2004) 204–207.
[11] C. Lee, M.S. Hwang, W.P. Yang, A flexible remote user authentication scheme using smart cards, ACM Oper. Syst. Rev. 36 (3) (2002) 46–52.
[12] W.B. Lee, C.C. Chang, User identification and key distribution maintaining anonymity for distributed computer network, Comput. Syst. Sci. 15 (4) (2000) 211–214.
[13] W.J. Tsuar, C.C. Wu, W.B. Lee, A flexible user authentication for multi-server internet services, Networking-JCN2001LNCS, vol. 2093, Springer-Verlag, 2001, pp. 174–183.
[14] L. Li, I. Lin, M. Hwang, A remote password authentication scheme for multi-server architecture using neural networks, IEEE Trans. Neural Netw. 12 (6) (2001) 1498–1504.
[15] C. Lin, M.S. Hwang, L.H. Li, A new remote user authentication scheme for multi-server architecture, Future Gener. Comput. Syst. 1 (19) (2003) 13–22.
[16] W.J. Tsuar, An enhanced user authentication scheme for multi-server internet services, Appl. Math. Comput. 170 (2005) 258–266.
[17] T.S. Wu, C.L. Hsu, Efficient user identification scheme with key distribution preserving anonymity for distributed computer networks, Comput. Secur. 23 (2004) 120–125.
[18] Y. Yang, S. Wang, F. Bao, J. Wang, R. Deng, New efficient user identification and key distribution scheme providing enhanced security, Comput. Secur. 23 (8) (2004) 697–704.
[19] W.S. Juang, Efficient multi-server password authenticated key agreement using smart cards, IEEE Trans. Consum. Electron. 50 (1) (2004) 251–255.
[20] C. Chang, J.S. Lee, An efficient and secure multi-server password authentication scheme using smart cards, IEEE. Proceeding of the International Conference on Cyberworlds, 2004.
[21] T.S. Messergers, E.A. Dabbish, R.H. Sloan, Examining smart card security under the threat of power analysis attacks, IEEE Trans. Comput. 51 (5) (2002) 541–552.
[22] Y.P. Liao, S.S. Wang, A secure dynamic ID based remote user authentication scheme for multi-server environment, Comput. Stand. Interfaces (2007), doi:10.1016/j.csi.2007.10.007.
[23] X. Wang, F. Guo, X. Lai, H. Yu, Collisions for Hash Functions MD4,MD5, HAVAL-128 and RIPEMD, Rump Session of Crypto'04 and IACR Eprint Archive, 2004.
[24] X. Wang, H.B. Yu, How to break MD5 and other hash functions, Advances in Cryptology Eurocrypt'05, Springer-Verlag, 2005, pp. 19–35

**Han-Cheng Hsiang** received the M.S. degree in Information and Computer Engineering from the Chung Yuan Christian University, Chungli, Taiwan in 1997. He completed his Ph.D. degree in Computer Science from the National Tsing Hua University, Hsinchu, Taiwan. His research activities are mainly focused on cryptography, information security, electronic commerce and mobile communications security.

**Wei-Kuan Shih** is a Professor at the Department of Computer Science, National Tsing Hua University. He completed his Ph.D. degree in the University of Illinois, Urbana-Champaign. His current research interests include real-time systems, wireless, Internet technology, multimedia system, and information security.