Cryptanalysis and Improvement of an Authentication Scheme for Telecare Medical Information Systems

Yun Zhao1*, Wenbo Shi2
1School of Information Engineering, Guangdong Medical College, Guangdong, China
2Department of Electronic Engineering, Northeastern University at Qinhuangdao, Qinhuangdao, China
*Corresponding author, e-mail: zhaoyun2012@hotmail.com

Abstract
The telecare medical information system (TMIS) could improve quality of medical care since it allows patients to enjoy health-care delivery services in their home. However, the privacy and security influence the development of the TMIS since it is employed in open networks. Recently, Wu and Xu proposed a privacy authentication scheme for the TMIS and claimed that their scheme could overcome weaknesses in previous schemes. However, we will demonstrate that their scheme is venerable to the server spoofing attack and cannot provide user anonymity. To overcome weaknesses in their scheme, we also propose a new authentication scheme for the TMIS. Analysis shows that our scheme not only overcome weaknesses in Wu et al.’s scheme, but also has better performance.

Keywords: mutual authentication, anonymity, smart card, telecare medical information system

1. Introduction
With the rapid development of technologies in wireless communication, low-power integrated circuits and wearable medical sensors, the telecare medical information system (TMIS) is widely used to improve quality of medical care. Through TMIS, patients could login the remote server and enjoy various medical services almost anywhere at any time. Therefore, the TMIS could bring great convenience to people’s life. How to address the privacy and security in the TMIS has attracted wide attention since the data transmitted in the TMIS is very sensitive and important.

The anonymity authentication scheme could provide mutual authentication between the user and the remote server and user anonymity. Then, it is very suitable for solve security problem in the TMIS. In 1981, Lamport [1] proposed the first authentication scheme for secure communication in open networks. However, Lamport’s scheme is vulnerable to the stolen verifier table attack. Since then, many authentication schemes [2-10] have been proposed for different applications. However, those schemes [1-10] are not for the TMIS since their performance is not satisfactory. To solve the problem, Wu et al. [11] proposed the first authentication scheme for TMIS. Unfortunately, He et al. [12] pointed out that Wu et al.’s scheme is vulnerable to the impersonation attacks and the insider attack. He et al. [12] also proposed a new authentication scheme for TMIS. Later, Wei et al. [13] pointed out that Wu et al.’s scheme and He et al.’s scheme cannot provide two-factor security. Wei et al. also proposed an improved scheme and claimed that their scheme could withstand various attacks. However, Zhu [14] demonstrated that Wei et al.’s scheme is vulnerable to the off-line password guessing attack.

All those authentication schemes for TMIS cannot provide user anonymity since users’ identities are transmitted in plaintext format. Das et al. [15] proposed a dynamic ID-based authentication scheme to protect user's anonymity. In 2012, Chen et al. [16] proposed a dynamic ID-based authentication scheme for TMIS. However, Cao et al. [17], Xie et al. [18], Lin [19] and Jiang et al. [20] pointed out that Chen et al.’s scheme had weaknesses such as off-line password guessing attack, tracking attack, lack of privacy protection and so on. Jiang et al. [20] also proposed an improved scheme to overcome weaknesses in previous schemes. However, Wu and Xu [21] pointed out that Jiang et al.’s scheme has useless identity and is vulnerable to off-line password guessing attack, user impersonation attack and Denial of Service (DoS)
attack. Wu and Xu also proposed an improved scheme to protect the user's privacy in TMIS. Unfortunately, we will demonstrate that Wu and Xu’s scheme is venerable to the server spoofing attack and cannot provide user anonymity. We also proposed a new authentication scheme for TMIS to overcome weaknesses in their scheme. Analysis shows that our scheme not only overcome in Wu et al.’s scheme, but also has better performance.

The organization of the paper is described as follows. In Section “Review of Wu and Xu’s scheme”, we briefly review their scheme. Then the Section “Security analysis of Wu and Xu’s scheme” analyzes security of Wu and Xu’s scheme. In Section “Our proposed scheme”, we propos a new authentication scheme for TMIS. Sections “Security analysis” and “Performance analysis” analyze the security and performance of our scheme separately. Some conclusions are proposed in the last section.

2. Review of Wu and Xu’s scheme

In this section, we will give a brief review of Wu and Xu et al.’s scheme. For convenience, some notations are defined as follows.

a) $iU$: a user;

b) $iID$: the identity of $iU$;

c) $iPW$: the password of $iU$;

d) $S$: the remote server for the system;

e) $x$: the secret key of $S$;

f) $T_i$: the timestamp generated by $iU$;

g) $T_s$: the timestamp generated by $S$;

h) $N$: the registration times of $iU$;

i) $sk$: the session key generated between $iU$ and $S$;

j) $E_k(M)$: Encryption of a message $M$ using the key $k$;

k) $D_k(C)$: Decryption of a message $C$ using the key $k$;

l) $h()$: a secure one-way hash function;

m) $\oplus$: the bitwise XOR operation;

n) $||$: the concatenation operation;

Wu and Xu’s scheme consists of five phases, i.e. the registration phase, the login phase, the authentication phase, the password change phase and the lost smart card revocation phase. The details are described as follows:

2.1. Registration Phase

In this phase, $iU$ could register or re-register at the remote $S$ through the following step.

1) $iU$ generates a random number $r_i$, chooses his identity $ID_i$, password $PW_i$, computes $HPW_i = h(r_i || PW_i)$ and sends the message $\{ID_i, HPW_i\}$ to $S$ through a secure channel.

2) After receiving $\{ID_i, HPW_i\}$, $S$ checks the validity of $ID_i$. If it is not valid, $S$ rejects the session; otherwise, $S$ checks the account records in database. If $U_i$ is a new user, $S$ adds the tuple $(ID_i, N = 0)$ into the database; otherwise, $S$ sets $N = N + 1$ and stores it. Then $S$ computes $J_i = h(ID_i || N)$, $L_i = J_i \oplus RPW_i$ and $E_i = h(x) \oplus h(RPW_i || ID_i)$. At last, $S$ stores $\{L_i, E_i, h(\cdot), E_i(\cdot), D_i(\cdot)\}$ into a smart card and sends it to $U_i$ through a secure channel.

3) After receiving the smart card, $U_i$ inputs $r_i$ into it.

2.2. Login Phase

When wanting to login at $S$ and enjoy services, as shown in Figure 1, $U_i$ will carry out the following steps.
1) \( U_i \) inserts his smart card into a card reader and inputs his identity \( ID_i \) and password \( PW_i \).

2) The smart card computes \( HPW_i = h_r(T_i \| PW_i) \), \( J_i = L_i \oplus RPW_i \), \( AID_i = E_i \oplus h(RPW_i \| ID_i) \oplus h(T_i) \oplus ID_i \), \( B_i = E_{i} \oplus h(RPW_i \| ID_i) \oplus T_i \), \( V_i = h(T_i \| J_i) \) and \( C_i = E_{h(T_i)}(AID_i \| T_i \| V_i) \), where \( T_i \) is the timestamp generated by \( U_i \). At last, \( U_i \) sends the message \( m_i = \{B_i, C_i\} \) to \( S \).

2.3. Authentication Phase

\[
\text{Figure 1. Wu and Xu’s Scheme}
\]

As shown in Figure 1, \( U_i \) and \( S \) could authenticate each other through executing the following steps.

1) After receiving \( m_i = \{B_i, C_i\} \), \( S \) computes \( T'_i = B_i \oplus h(x) \) and checks whether \( T'_i \) is fresh. If it is not fresh, \( S \) stops the request; otherwise, \( S \) computes \( ID'_i = AID'_i \oplus h(x) \oplus h(T'_i) \) and checks \( ID'_i \). If \( ID'_i \) and \( ID_i \) are equal, \( S \) checks whether \( T''_i = AID'_i \oplus h(x) \oplus h(T_i) \) is in the account table. If it is not, \( S \) stops the session; otherwise, \( S \) computes \( J'_i = h(x \| ID'_i \| N) \) and
checks whether $V'_i$ and $h(T'_i || J'_i)$ are equal. If they are not equal, $S$ stops the request; otherwise, $S$ computes $B_i = h(x) \oplus T_i$, $C_i = E_{h(x)}(V'_i || T_i)$, $sk = h(J'_i || T'_i || ID'_i)$ and sends the message $m_i = \{B_i, C_i\}$ to $U_i$, where $T_i$ is the timestamp generated by $S$.

2) After receiving $m_i = \{B_i, C_i\}$, $U_i$ computes $T'_i = B_i \oplus E_i \oplus h(RPW'_i || ID_i)$ and checks the freshness of $T'_i$. If it is not fresh, $U_i$ stops the session; otherwise, $U_i$ computes $(V'_i || T'_i) = D_{h(x)}(C_i)$. Then $U_i$ checks whether the equations $V'_i = V_i$ and $T'_i^* = T_i^*$ hold. If either of them does not hold, $U_i$ stops the session; otherwise, $U_i$ computes $sk = h(J_i || T_i || T'_i || ID_i)$.

### 2.4. Password Change Phase

When $U_i$ wants to change his password, he inserts his smart card into a card reader and inputs his identity $ID_i$, the old password $PW_i$, and a new password $PW_{i\text{new}}$. Then step 2) of the login phase and step 1) of the authentication phase are executed.

After receiving $m_i = \{B_i, C_i\}$, $U_i$ computes $T'_i = B_i \oplus E_i \oplus h(RPW'_i || ID_i)$ and checks the freshness of $T'_i$. If it is not fresh, $U_i$ stops the session; otherwise, $U_i$ computes $(V'_i || T'_i) = D_{h(x)}(C_i)$. Then $U_i$ checks whether the equations $V'_i = V_i$ and $T'_i^* = T_i^*$ hold. If either of them does not hold, $U_i$ stops the session; otherwise, $U_i$ computes $HPW_{i\text{new}} = h(r_i \parallel PW_{i\text{new}})$, $L_i = L_i \oplus R PW_i \oplus R PW_{i\text{new}}$ and $E_{i\text{new}} = E_i \oplus h(RPW'_i || ID_i) \oplus h(RPW_{i\text{new}} || ID_i)$. At last, $U_i$ replaces $L_i$ and $E_i$ with $L_{i\text{new}}$ and $E_{i\text{new}}$ separately.

### 2.5. Lost Smart Card Revocation Phase

When $U_i$ loses his smart card, he can re-register at $S$ through the secure channel as the registration phase. $S$ verifies $U_i$, makes $N = N + 1$ and stores $(ID_i, N)$ into the account table. At last, $S$ issues a new smart card to $U_i$.

### 3. Security Analysis of Wu and Xu’s Scheme

Wu and Xu claimed that their scheme could withstand various attacks. However, in this section, we will show their scheme cannot provide user’s anonymity and is vulnerable to the server spoofing attack.

Suppose $U_a$ is a malicious user. Then he could get a smart card containing the message $\{L_i, E_i, h(\cdot), E_{i\text{new}}(\cdot), D_{\cdot}(\cdot)\}$ through registering at $S$, where, $L_a = h(x || ID_a || N) \oplus R PW_a$, $E_a = h(x) \oplus h(RPW_a || ID_a)$ and $HPW_a = h(r_a \parallel PW_a)$. Since the messages are transmitted in public channel, we could assume that $U_a$ has total control over the channel, i.e. he could intercept, insert and modify messages transmitted between the user and the server.

#### 3.1. User Anonymity

User anonymity is very important if the TMIS since the leakage of user’s identity could influence user’s privacy. Wu and Xu claimed that their scheme could provide user’s anonymity. However, in this section, we will show a malicious $U_a$ could get other user’s identity. The detail is described as follows.

1) $U_a$ extracts $\{L_i, E_i\}$ from his smart card, where $HPW_a = h(r_a \parallel PW_a)$, $L_a = h(x || ID_a || N) \oplus R PW_a$ and $E_a = h(x) \oplus h(RPW_a || ID_a)$.

2) $U_a$ computes $HPW_a = h(r_a \parallel PW_a)$ and $h(x) = E_a \oplus h(RPW_a || ID_a)$.

3) $U_a$ intercepts $m_i = \{B_i, C_i\}$ sent $U_i$, where $AID_i = h(x) \oplus h(T_i) \oplus ID_i$, $C_i = E_{h(x)}(AID_i || T_i || V_i)$, $B_i = h(x) \oplus T_i$ and $V_i = h(T_i || J_i)$. 

Cryptanalysis and Improvement of an Authentication Scheme for Telecare… (Yun Zhao)
4) \( U_a \) computes \( T_j = B_i \oplus h(x) \), \( (AID_j \parallel T_j \parallel V_i) = D_{h(T_j)}(C_i) \) and \( ID_i = AID_j \oplus h(x) \oplus h(T_j) \).

From the above description, we know that \( U_a \) could get the identity of \( U_i \) easily. Therefore, Wu and Xu’s scheme cannot provide user anonymity.

3.2. Server Spoofing Attack

Wu and Xu claimed that their scheme could withstand various attacks. In this subsection, we will show their scheme is vulnerable to the server spoofing attack, i.e. a malicious user \( U_a \) could impersonate the server to another user \( U_i \). The details are described as follows.

1) \( U_a \) extracts \( \{L_i,E_i\} \) from his smart card, where \( HPW_a = h(r_a \parallel PW_a) \), \( L_a = h(x \parallel ID_a \parallel N) \oplus RPW_a \) and \( E_a = h(x) \oplus h(RPW_a \parallel ID_a) \).

2) \( U_a \) computes \( HPW_a = h(r_a \parallel PW_a) \) and \( h(x) = E_a \oplus h(RPW_a \parallel ID_a) \).

3) \( U_a \) intercepts \( m_1 = \{B_i,C_i\} \) sent \( U_i \), where \( AID_j = h(x) \oplus h(T_j) \parallel ID_j \), \( C_i = E_{h(T_i)}(AID_j \parallel T_i \parallel V_i) \), \( B_i = h(x) \oplus T_i \) and \( V_i = h(T_i \parallel J_i) \).

4) \( U_a \) computes \( T'_j = B_i \oplus h(x) \), \( (AID'_j \parallel T'_j \parallel V'_i) = E_{h(T'_j)}(C_i) \), \( B_2 = h(x) \oplus T'_j \), \( C_2 = E_{h(T'_j)}(V'_i \parallel T'_j) \), and sends the message \( m_2 = \{B_2,C_2\} \) to \( U_i \).

It is easy to verify that the message \( m_2 = \{B_2,C_2\} \) could pass \( U_i \)’s verification. Therefore, \( U_a \) could impersonate \( S \) to \( U_i \) successfully and Wu and Xu’s scheme is vulnerable to the server spoofing attack.

4. Our Proposed Scheme

To overcome weaknesses in Wu and Xu’s scheme, we proposed an improved authentication scheme for TMIS. Our scheme also consists of five phases, i.e. the registration phase, the login phase, the authentication phase, the password change phase and the lost smart card revocation phase. The details are described as follows:

4.1. Registration Phase

In this phase, \( U_i \) could register or re-register at the remote \( S \) through the following step.

1) \( U_i \) generates a random number \( r_i \), chooses his identity \( ID_i \), password \( PW_i \), computes \( HPW_i = h(r_i \parallel PW_i) \) and sends the message \( \{ID_i,HPW_i\} \) to \( S \) through a secure channel.

2) After receiving \( \{ID_i,HPW_i\} \), \( S \) checks the validity of \( ID_i \). If it is not valid, \( S \) rejects the session; otherwise, \( S \) checks the account records in database. If \( U_i \) is a new user, \( S \) adds the tuple \( (ID_i,N = 0) \) into the database; otherwise, \( S \) sets \( N = N+1 \) and stores it. Then \( S \) generates a pseudo identity \( pid_i \), computes \( I_i = h(x \parallel pid_i) \), \( J_i = h(x \parallel ID_i \parallel N) \), \( E_i = I_i \oplus h(RPW_i \parallel ID_i) \) and \( L_i = J_i \oplus RPW_i \). At last, \( S \) stores \( \{pid_i,L_i,E_i,h(\cdot),E_k(\cdot),D_k(\cdot)\} \) into a smart card and sends it to \( U_i \) through a secure channel.

3) After receiving the smart card, \( U_i \) inputs \( r_i \) into it.

4.2. Login Phase

When wanting to login at \( S \) and enjoy services, as shown in Fig. 1, \( U_i \) will carry out the following steps.

1) \( U_i \) inserts his smart card into a card reader and inputs his identity \( ID_i \) and password \( PW_i \).
2) The smart card computes $\text{HPW}_j = h(r_j \parallel PW_j)$, $J_j = L_j \oplus \text{RPW}_j$, $I_j = E_j \oplus h(\text{RPW}_j \parallel ID_j)$, $V_j = h(ID_j \parallel T_j \parallel J_j)$ and $C_j = E_j(ID_j \parallel T_j \parallel V_j)$, where $T_j$ is the timestamp generated by $U_j$. Then, $U_j$ sends the message $m_i = \{\text{pid}_i, C_j\}$ to $S$.

4.3. Authentication Phase

![Diagram of Authentication Phase](image)

As shown in Figure 2, $U_j$ and $S$ could authenticate each other through executing the following steps.

1) After receiving $m_i = \{\text{pid}_i, B, C_j\}$, $S$ computes $I_i = h(x \parallel \text{pid}_i)$, $(ID_i' \parallel T'_i \parallel V'_i) = D_j(C_j)$ and checks whether $T'_i$ is fresh. If it is not fresh, $S$ stops the request; otherwise, $S$ checks whether $ID_i'$ is in the account table. If it is not in the account table, $S$ stops the session; otherwise, $S$ computes $J'_i = h(x \parallel ID_i' \parallel N)$ and checks whether $V'_i$ and $h(ID_i' \parallel T'_i \parallel J'_i)$ are equal. If they are not equal, $S$ stops the request; otherwise, $S$ generates a new pseudo identity $\text{pid}_i^\text{new}$, computes $I_i^\text{new} = h(x \parallel \text{pid}_i^\text{new})$, $V'_i = h(ID_i' \parallel T'_i \parallel \text{pid}_i^\text{new} \parallel I_i^\text{new} \parallel T'_i)$, $C_2 = E_j(V'_i \parallel T_i \parallel \text{pid}_i^\text{new} \parallel I_i^\text{new})$, $sk = h(J'_i \parallel T'_i \parallel ID_i')$ and sends the message $m_2 = \{C_2\}$ to $U_j$, where $T'_i$ is the timestamp generated by $S$. 

Cryptanalysis and Improvement of an Authentication Scheme for Telecare… (Yun Zhao)
2) After receiving $m_2 = \{C_z\}$, $U_i$ computes $(V'_i \| T'_i \| \text{pid}^{\text{new}}_i \| I_i^{\text{new}}) = D_i(C_z)$ and checks whether $T'_i$ is fresh. If it is not fresh, $U_i$ stops the session; otherwise, $U_i$ checks whether the equation $V'_i = h(ID_i \| T_i \| V_i \| \text{pid}^{\text{new}}_i \| I_i^{\text{new}} \| T_i)$ holds. If it does not hold, $U_i$ stops the session; otherwise, $U_i$ replaces $\text{pid}_i$ and $E_i$ with $\text{pid}^{\text{new}}_i$ and $I_i^{\text{new}} \oplus h(\text{RPW}_i \| ID_i)$ separately. At last, $U_i$ computes the session key $sk = h(J_i \| T_i \| T'_i \| ID_i)$.

4.4. Password Change Phase

When $U_i$ wants to change his password, he inserts his smart card into a card reader and inputs his identity $ID_i$, the old password $PW_i$ and a new password $PW^{\text{new}}_i$. Then step 2) of the login phase and step 1) of the authentication phase are executed.

After receiving $m_2 = \{C_z\}$, $U_i$ computes $(V'_i \| T'_i \| \text{pid}^{\text{new}}_i \| I_i^{\text{new}}) = D_i(C_z)$ and checks whether $s^{T'}_i$ is fresh. If it is not fresh, $U_i$ stops the session; otherwise, $U_i$ checks whether the equation $s^{T'}_i = h(ID_i \| T_i \| V_i \| \text{pid}^{\text{new}}_i \| I_i^{\text{new}} \| T_i)$ holds. If it does not hold, $U_i$ stops the session; otherwise, $U_i$ computes $\text{HPW}^{\text{new}}_i = h(v_i \| PW^{\text{new}}_i)$, $L^{\text{new}}_i = L_i \oplus \text{RPW}_i \oplus \text{RPW}^{\text{new}}_i$, $E^{\text{new}}_i = I^{\text{new}}_i \oplus h(\text{RPW}^{\text{new}}_i \| ID_i)$ and replaces $L_i$ and $E_i$ with $L^{\text{new}}_i$ and $E^{\text{new}}_i$ separately.

4.5. Lost Smart Card Revocation Phase

When $U_i$ loses his smart card, he can re-register at $S$ through the secure channel as the registration phase. $S$ verifies $U_i$, makes $N = N + 1$ and stores $(ID_i, N)$ into the account table. At last, $S$ issues a new smart card to $U_i$.

5. Security Analysis

In this section, we will analyze the security of our scheme. We will show our scheme could withstand general attacks and provide common security features.

5.1. User Anonymity

The user’s identity $ID_i$ is included in the ciphertext $C_i = E_i(ID_i \| T_i \| V_i)$, where $V_i = h(ID_i \| T_i \| J_i)$, $I_i = h(x \| \text{pid}^i)$ and $T_i$ is the timestamp generated by $U_i$. Without the knowledge of the server’s secret key $x$, the adversary, including the malicious user, cannot compute $I_i$ and decrypt $C_i$. Therefore, our scheme could provide the user anonymity.

5.2. Mutual Authentication

Without the knowledge $J_i = h(x \| ID_i \| N)$, any adversary including the malicious user cannot generate $V'_i = h(ID_i \| T'_i \| J'_i)$. Then, he cannot generate a legal message $m_i = \{\text{pid}_i, C_i\}$, where $C_i = E_i(ID_i \| T_i \| V_i)$ and $T_i$ is the timestamp generated by $U_i$. Therefore, $S$ could authenticate $U_i$ by checking whether $V'_i$ and $h(ID'_i \| T'_i \| J'_i)$ are equal in Step 1) of the authentication scheme.

Without the knowledge the server’s secret key $x$, any adversary including malicious cannot compute $I_i = h(x \| \text{pid}^i)$ and get $(ID_i \| T'_i \| V'_i)$ by decrypting $C_i$. Then, he cannot generate a legal message $m_2 = \{C_z\}$, where $C_z = E_i(V'_i \| T'_i \| \text{pid}^{\text{new}}_i \| I^{\text{new}}_i)$, $V'_z = h(ID_i \| T'_i \| V'_i \| \text{pid}^{\text{new}}_i \| I^{\text{new}}_i \| T'_i)$ and $T'_z$ is the timestamp generated by $S$. Therefore, $U_i$ could authenticate $S$ by checking whether the equation $V'_z = h(ID_i \| T'_i \| V'_i \| \text{pid}^{\text{new}}_i \| I^{\text{new}}_i \| T'_i)$ holds in Step 2) of the authentication phase.
5.3. Privileged Insider Attack

In the registration phase of our scheme, the user \( U_i \) sends the message \( \{ID_i, HPW_i \} \) to the server, where \( HPW_i = h(r_i \| PW_i) \) and \( r_i \) is a random number generated by \( U_i \). Then, the privileged insider of the server cannot get \( U_i \)’s password \( PW_i \) since it is protected by the secure hash function and the random number. Therefore, our scheme could withstand the privileged insider attack.

5.4. Man-in-the-middle Attack

From the analysis in Section 5.2, we know that our scheme could provide mutual authentication between the user and the server. Therefore, our scheme could withstand the man-in-the-middle attack.

5.5. Replay Attack

The adversary may intercept the message \( m_i = \{pid_i, C_i \} \) and replay it to the server, where \( HPW_i = h(r_i \| PW_i) \), \( J_i = L_i \oplus RPW_i \), \( I'_i = E_i(h(RPW_i \| ID_i)) \), \( V_i = h(ID_i \| T_j \| J_i) \), \( C_i = E_i(ID_i \| T_j \| V'_i) \) and \( T_j \) is the timestamp generated by \( U_i \). In the Step 1) of the authentication, the server will check the freshness of \( T_j \), then he could find the attack easily.

The adversary may intercept the message \( m_2 = \{C_2 \} \) and replay it to the user, where \( C_2 = E_{I'_j}(V'_i \| T_j \| pid_i^{new} \| I_i^{new}) \), \( V'_i = h(ID'_i \| T'_j \| V'_i \| pid_i^{new} \| I_i^{new} \| T_j) \) and \( T_j \) is the timestamp generated by \( S \). In the Step 2) of the authentication phase, the user will check the freshness of \( T_j \), then he could find the attack easily.

5.6. Impersonation Attack

To impersonate the user to the server, the adversary has to generate a legal message \( m_i = \{pid_i, C_i \} \), where \( C_i = E_i(ID_i \| T_j \| V'_i) \), \( V'_i = h(ID_i \| T_j \| J_i) \) and \( T_j \) is the current timestamp. However, any adversary including the malicious user cannot generate \( V'_i \) if he does not know the value of \( J_i = h(x \| ID_j \| N) \). Therefore, our scheme could withstand the impersonation attack.

5.7. Server Spoofing Attack

To impersonate the server to the user, the adversary has to generate a legal message \( m_2 = \{C_2 \} \) when he intercepts the message \( m_i = \{pid_i, C_i \} \), where \( C_i = E_i(ID_i \| T_j \| V'_i) \), \( V'_i = h(ID_i \| T_j \| J_i) \), \( C_2 = E_{I'_j}(V'_i \| T'_j \| pid_i^{new} \| I_i^{new}) \), \( V'_i = h(ID'_i \| T'_j \| V'_i \| pid_i^{new} \| I_i^{new} \| T_j) \).

However, any adversary including malicious cannot compute \( I_j = h(x \| pid_i) \) and get \( (ID'_j \| T'_j \| V'_i) \) by decrypting \( C_i \) if he does not know the server’s secret key \( x \). Therefore, our scheme could withstand the server spoofing attack.

5.8. Stolen Verifier Attack

In our scheme, the server just maintains a table of tuple \( (ID_j, N) \) and there is no user’s password is stored in the table. Therefore, our scheme could withstand the stolen verifier attack.

5.9. Modification Attack

The adversary may intercept the message \( m_i = \{pid_i, C_i \} \) and resend it after modifying it at his will. However, the user could find the attack by checking whether \( V'_i \) and \( h(ID'_i \| T'_j \| J'_i) \) are equal. By the similar method, we could show the user could find the modification of \( m_2 = \{C_2 \} \). Therefore, our scheme could withstand the modification attack.
5.10. Stolen Smart Card Attack

The adversary may steal the user’s smart card and extract the stored information \( \{pid, I, E\} \) through the side channel attack, where \( I = h(x \| pid) \), \( J = h(x \| ID) \) and \( E = I \oplus h(RPW, \| ID) \). However, there is no message related \( pid \) and \( E \) could be found since the user will change \( pid \) and \( E \) in every session. Therefore, he cannot verify whether his guess is correct and our scheme could withstand the stolen smart card attack.

6. Performance Analysis

In this section, we will analyze the performance of our scheme. To the best of our knowledge, Jiang et al.’s scheme [20] and Wu and Xu’s scheme [21] are more suitable for TMIS than other schemes. We will also compare our scheme with that two schemes. For convenience, some notations are defined as follows.

a) \( T_H \): the running time of a hash function operation;

b) \( T_S \): the running time of a symmetric encryption/decryption operation;

c) \( T_{XOR} \): the running time of a bitwise XOR operation;

We just need to compare the performance in the login and authentication phase of different schemes since other phases are executed only one time. The performance comparisons are listed in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>Jiang et al.’s scheme</th>
<th>Wu and Xu’s scheme</th>
<th>Our scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>( 3T_H + 1T_S + T_{XOR} )</td>
<td>( 6T_H + 2T_S + 5T_{XOR} )</td>
<td>( 5T_H + 2T_S + 3T_{XOR} )</td>
</tr>
<tr>
<td>Server</td>
<td>( 3T_H + 3T_S )</td>
<td>( 5T_H + 2T_S + 3T_{XOR} )</td>
<td>( 5T_H + 2T_S )</td>
</tr>
<tr>
<td>Total</td>
<td>( 6T_H + 4T_S + T_{XOR} )</td>
<td>( 11T_H + 4T_S + 8T_{XOR} )</td>
<td>( 10T_H + 4T_S + 5T_{XOR} )</td>
</tr>
</tbody>
</table>

In the login and authentication phase of Jiang et al.’s scheme [20], \( 3T_H + 1T_S + T_{XOR} \) and \( 3T_H + 3T_S \) are needed at the side of user and server separately. In the login and authentication phase of Wu and Xu’s scheme [21], \( 6T_H + 2T_S + 5T_{XOR} \) and \( 5T_H + 2T_S + 3T_{XOR} \) are needed at the side of user and server separately. In the login and authentication phase of our scheme, \( 5T_H + 2T_S + 3T_{XOR} \) and \( 5T_H + 2T_S \) are needed at the side of user and server separately. Besides, the running time of a bitwise XOR operation could be ignored when compared with that of a hash function or a symmetric encryption/decryption operation. Therefore, Jiang et al.’s scheme has better performance than Wu and Xu’s scheme and our scheme. However, Wu and Xu pointed out that Jiang et al.’s scheme has useless identity and is vulnerable to off-line password guessing attack, user impersonation attack and DoS attack. Therefore, Jiang et al.’s scheme is not suitable for practical applications. We have demonstrated that Wu and Xu et al.’s scheme is venerable to the server spoofing attack and cannot provide user anonymity. Besides, our scheme has better performance than Wu and Xu et al.’s scheme. Therefore, we could conclude that our scheme is more suitable for TIMS.

7. Conclusion

In this paper, we demonstrate that Wu and Xu’s scheme cannot withstand the server spoofing attack and cannot provide the user anonymity. To overcome those weaknesses, we proposed an improved authentication scheme for TMIS. Security analysis shows our scheme could withstand general attacks and overcome the drawback of Wu and Xu’s scheme. Performance analysis shows our scheme also has better performance than Wu and Xu’s scheme. Therefore, we could conclude that our scheme is more suitable for TMIS.
Acknowledgements
The authors would like to thank the anonymous referees for their invaluable comments. This research was supported by National Natural Science Foundation of China (no.61202447), the Natural Science Foundation of Hebei Province of China (no. F2013501066) and the Northeastern University at Qinhuangdao Science and Technology Support Program (no. xnk201307).

References