# Cryptanalysis and Improvement of a Biometric-based Authentication Scheme for Multi-server Architecture

Tao Wan[1], Xiaochang Liu[1], Weichuan Liao[2], and Nan Jiang[1]
*(Corresponding author: Tao Wan)*

School of Information Engineer, East China Jiaotong University, China[1]
School of Science, East China Jiaotong University, China[2]
(Email: wantao217@163.com)

*(Received Nov. 30, 2018; Revised and Accepted Aug. 5, 2019; First Online Sept. 21, 2019)*

## Abstract

In recent year, with the increasing amount of wireless technologies, biometric-based authentication schemes for multi-server architectures have become more crucial and widely developed. In 2016, Wang *et al.* demonstrated that Mishra *et al.*'s protocol has several drawbacks and proposed an improved authentication scheme of biometric-based architecture using smart card and password. They claimed that their scheme achieves intended security requirements and is more appropriate for practical applications. In this paper, we indicate that their scheme cannot resist session key disclosure, smart card forgery attack, server spoofing attack, user impersonation attack, DoS attack, and no provision of user anonymity. Furthermore, we propose a robust biometric-based authentication scheme using public-key encryption techniques to remove these defects. The performance and functionality comparison shows that our proposed scheme provides the best secure functionality and is computational efficient.

*Keywords: Authentication; Biometric; Multi-Server; Security; Smart Card*

## 1 Introduction

With the swift expansion of communication technologies and mobile devices, an increasing number of remote user authentication schemes are usually used to provide services to users. Earlier authentication methods were limited to single-server architecture. However, users need to obtain different services from multiple servers, they not only have to register to different servers, but also need to remember a large number of identities and passwords. Obviously, it is very difficult and unsafe for users to remember and manage multiple information. As a scalable solution, multi-server architecture has been introduced, where the users can register only once at the registration server and avail the services of all associated application servers. Several authors have suggested various authentication protocols for multi-server architecture during the past decade [1, 3, 4, 6, 8, 22, 27] .

Password, smart card and biometrics based authentication verifies the legitimacy of each user and offers the access to network resources. The first remote user password based authentication method was proposed by Lamport [12] . Unfortunately, password based authentication method is vulnerable to some attacks, especially, password guessing attack. Hence, the password with smart card methods have proposed. However, several researches indicated that password with smart card methods are still prone to numerous attacks [9,13,18,21,29] . To solve these problems, many researches have combined the biometric, password and smart card to enhance the security of authentication schemes [14, 17, 19, 23].

In 2009, Wang *et al.* [28] proposed a dynamic ID-based remote user authentication scheme and claimed that their scheme provides user's anonymity. Unfortunately, in 2011, Khan *et al.* [11] presented that Wang's protocol is prone to user anonymity, session key disclosure attack and smart card stolen attack. Furthermore, they proposed an enhanced authentication scheme to overcome the weaknesses of Wang *et al.*'s scheme and is more secure and efficient for practical application environment. In 2012, Chen *et al.* [2] proved that Khan *et al.*'s scheme is still vulnerable to insider attack. To remedy these, they proposed an enhanced authentication scheme and demonstrated their scheme is more secure. In 2013, Jiang *et al.* [10] observed that Chen *et al.*'s scheme achieves neither anonymity nor untraceability, and is sensitive to the identity guessing attack and tracking attack. Then, they proposed an enhanced authentication scheme which achieves user anonymity and untraceablity and claimed that it is a secure and efficient authentication scheme with user privacy preservation which is practical for TMIS. However, Wu and Xu *et al.* [30] proved that Jiang *et al.*'s scheme still cannot resist off-line password guessing attack, user impersonation attack, denial-of-service attack and so on. They even put forward an improved mutual authentication scheme used for a telecare medical information system. Chuang and Chen *et al.* [5] proposed an efficient and secure dynamic ID-based authentication scheme for TMI systems and demonstrated their scheme overcomes several drawbacks. In 2014, Mishra *et al.* [16] pointed out several drawbacks of Chuang and Chen's protocol, such as, server spoofing attack and Denial-of-Service attack. Furthermore, they proposed an efficient improvement on Chuang and Chen's scheme. In 2016, Wang *et al.* [24] proved that Mishra *et al.*'s protocol was vulnerable to masquerade attack, replay attack and Denial-of-Service attack. They proposed a novel biometric-based multi-server architecture and key-agreement scheme. But, we identify that Wang *et al.*'s scheme is still vulnerable to the server spoofing attack and user impersonation attack. Besides, their scheme cannot resist to session key disclosure, smart card forgery attack, DoS attack and fails to provide user anonymity.

The remainder of this manuscript is organized as follows. We introduce the one-way secure hash function, threat model and biometrics-based fuzzy extractor in Section 2. We review the robust smart card authentication scheme for multi-server architecture proposed by Wang *et al.* in Section 3. We analyze the security flaws of Wang *et al.*'s scheme in Section 4. We present a proposed protocol in Section 5. We compare the performance of our proposed scheme with the previous schemes in Section 6. We conclude this paper in Section 7.

# 2 Preliminaries

During this section, we briefly describe some concepts relating to secure hash function, threat models and biometrics-based fuzzy extractor as follows.

## 2.1 One-way Secure Hash Function

A one-way secure hash function $h : \{0,1\}^* \rightarrow \{0,1\}^n$ is considered as cryptographically secure and deterministic algorithm, which takes arbitrary size string $x$ as input and produces a fixed length value $V = h(x) \in \{0,1\}^n$. A secure hash function has the following attributes:

- It is computationally easy to find $V = h(x)$, given $h(\cdot)$ and $x$.
- It is computationally infeasible to compute $x$, given $V$ and $h(\cdot)$.
- For given hash code $V = h(x)$ and hash function $h(x)$, it is infeasible to find the input $x'$ such that $h(x') = h(x)$. This property is known as weak collusion resistance property.
- It is difficult to find two inputs $x_1 \neq x_2$ such that $h(x_1) = h(x_2)$. This property is known as strong collusion resistance property.

## 2.2 Treat Model

For the analysis of security of Wang *et al.*'s scheme and the proposed scheme in this paper, we consider a widely accepted threat model to inspect the security of the proposed protocol that has been considered in most of the existing authentication protocols [7,25]. More details about these threat models are described as below.

- An attacker might be a malicious user or malicious server.
- An attacker can extract the information from the smart card by examining the power consumption or leaked information.
- An attacker is able to eavesdrop all the communications between the parties involved such as a user and a server over a public channel.
- An attacker can trap, insert, modify, resend and delete the eavesdropped transmitted messages.
- An attacker may try to trace the actions of a particular user when any of the transmitted parameter is constant.
- In some situation, an attacker may know the previously established session keys. This presumption help us deal with session key disclosure.

## 2.3 Biometrics-based Fuzzy Extractor

Here, we briefly discuss the preliminaries about biometrics-based fuzzy extractor used in our scheme. The fuzzy extractor converts the biometric information into two values, which consists of two procedures, namely, *Gen* and *Rep*. More details illustrated as following:

- *Gen* is a generation procedure, which on input biometric data $BIO_i$, outputs an extracted string $P_i$ and auxiliary string $R_i$, where $Gen(BIO_i) \rightarrow (R_i, P_i)$.
- *Rep* is a deterministic generation reproduction procedure that allows to recover $R_i$ from the corresponding auxiliary string $P_i$ and any vector $BIO_i^*$ close to $BIO_i$, where $Rep(BIO_i^*, P_i) \rightarrow R_i$.

  The uniqueness property of a biometric allows its applications in authentication protocols.

# 3 Review of Wang *et al.*'s Scheme

In this section, we briefly review Wang *et al.*'s biometric-based authentication scheme for multi-server. Three roles participate in this scheme: The user $U_i$, the server $S_j$ and the registration center $RC$. There are five phases relating to Wang *et al.*'s scheme, ie. server registration phase, user registration phase, login and authentication phase, password change phase and revocation/re-registration phase. The details are described in the following subsections. Table 1 lists the notations used in this scheme.

Table 1: Notations used in the paper

| Symbols | Their meaning |
|---|---|
| $RC$ | *The registration center* |
| $U_i$ | *The $i_{th}$ user* |
| $ID_i$ | *The $U_i$'s identity* |
| $S_j$ | *The $j_{th}$ application server* |
| $SID_j$ | *The $S_j$'s identity* |
| $PW_i$ | *The user $U_i$'s password* |
| $PSK$ | *Per shared key* |
| $x$ | *Master secret key* |
| $h(\cdot)$ | *A secure one-way hash function* |
| $\|\|$ | *Concatenation operation* |
| $\oplus$ | *XOR operation* |
| $SK_{ij}$ | *Section key shared between $U_i$ and $S_j$* |

### 3.1 Server Registration Phase

This phase is executed between the application server $S_j$ and the registration center $RC$. This registration phase consists of the following steps:

**Step S1:** The server $S_j$ first sends a registration request to the registration center $RC$.

**Step S2:** Receiving the registration request from the remote server $S_j$, the registration center $RC$ assigns the value $PSK$ to the remote server $S_j$.

### 3.2 User Registration Phase

When a user wishes to access any services provided by the registered servers, he/she must first register himself/herself. This registration phase consists of the following steps:

**Step U1:** The user $U_i$ chooses an identity $ID_i$, password $PW_i$. Then the user $U_i$ imprints his personal biometric information $BIO_i$ at a sensor. The sensor sketches $BIO_i$ to extract an unpredictable binary string $R_i$ and an auxiliary binary string $P_i$ from $Gen(BIO_i) \rightarrow (R_i, P_i)$. Then, sensor stores $P_i$ in the memory.

**Step U2:** The user $U_i$ computes $RPW_i = h(PW_i||R_i)$ and sends $\{ID_i, RPW_i\}$ to $RC$ via a secure channel. $RC$ adds a novel entry $< ID_i, N_i = 1 >$ to the database, where $N_i$ means the times of user registration.

**Step U3:** The registration center $RC$ computes

$$\begin{aligned} A_i &= h(ID_i||x||T_r), \\ B_i &= RPW_i \oplus h(A_i), \\ C_i &= B_i \oplus h(PSK), \\ D_i &= PSK \oplus A_i \oplus h(PSK), \\ V_i &= h(ID_i||RPW_i), \end{aligned}$$

where $T_r$ is the time of user registration time.

**Step U4:** The registration center $RC$ securely issues the smart card containing $\{B_i, C_i, D_i, V_i\}$ to the user $U_i$.

**Step U5:** After receiving the issued smart card, the user $U_i$ stores the $P_i$ into the smart card.

### 3.3 Login and Authentication Phase

When a legal user $U_i$ wants to access the resources provided by remote server $S_j$, he/she first attaches the smart card to a device reader, and inputs his/her identity $ID_i$ and password $PW_i$, and imprints the biometrics $BIO_i^*$ at the sensor. Sensor sketches $BIO_i^*$ and recovers $R_i$ from $Rep(BIO_i^*, P_i) \rightarrow R_i$. Then, as illustrated in Figure 1, the login and authentication mechanism is performed as follows:

**Step V1:** The user $U_i$ computes $RPW_i = h(PW_i||R_i)$ and checks whether $h(ID_i||RPW_i)$ is equal to $V_i$. If it holds, the smart card further calculates $h(PSK) = B_i \oplus C_i$, then generates a random nonce $N_1$ and computes

$$\begin{aligned} AID_i &= ID_i \oplus h(N_1), \\ M_1 &= RPW_i \oplus N_1 \oplus h(PSK), \\ M_2 &= h(AID_i||N_1||RPW_i||SID_j||T_i). \end{aligned}$$

The user $U_i$ sends the login request message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$ to the server $S_j$, where $T_i$ means the timestamp.

**Step V2:** Upon receiving the message from the user $U_i$, the server $S_j$ checks whether $T_i - T_j$ is less than $\triangle T$, where $T_j$ is a timestamp. If not, the communication is simply terminated. Otherwise, the server $S_j$ computes

$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1 &= RPW_i \oplus M_1 \oplus h(PSK). \end{aligned}$$

and verifies whether $h(AID_i||N_1||RPW_i||SID_j||T_i)$ is equal to $M_2$. If it holds, the server $S_j$ generates a random number $N_2$, and computes

$$\begin{aligned} SK_{ij} &= h(AID_i|SID_j||N_1||N_2), \\ M_3 &= N_2 \oplus h(AID_i||N_1) \oplus h(PSK), \\ M_4 &= h(SID_j||N_2||AID_i). \end{aligned}$$

**Step V3:** Furthermore, the server $S_j$ sends the response message $\{SID_j, M_3, M_4\}$ to $U_i$. Upon getting the response message, the user $U_i$ computes

$$\begin{aligned} N_2 &= M_3 \oplus h(AID_i||N_1) \oplus h(PSK), \\ K_{ij} &= h(AID_i||SID_j||N_1||N_2), \\ N_1 &= B_i \oplus M_1 \oplus h(PSK). \end{aligned}$$

and verifies whether $h(SID_j||N_2||AID_i)$ is equal to $M_4$. If not, the communication is simply terminated. Otherwise, the user $U_i$ computes $M_5 =$

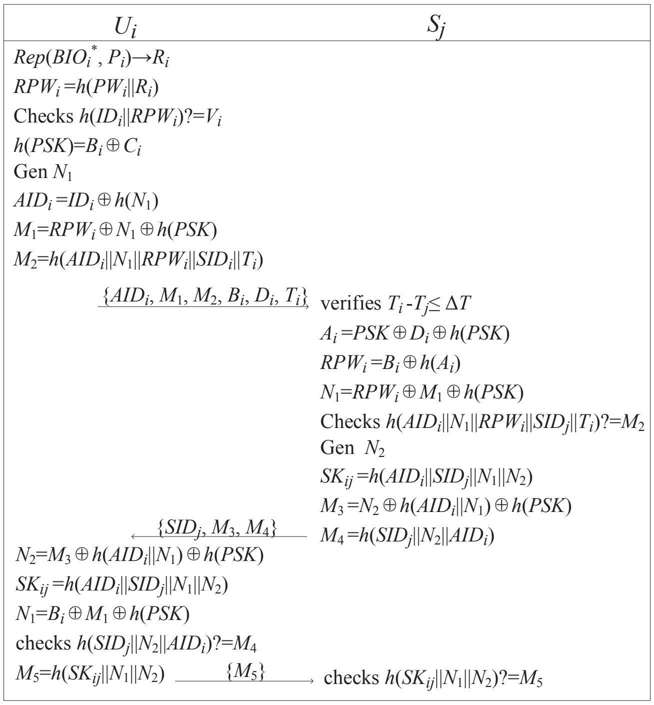


Figure 1: User login and authentication on Wang *et al.*'s Scheme

$h(SK_{ij}||N_1||N_2)$. Then user $U_i$ transmits the message $\{M_5\}$ to the server $S_j$.

**Step V4:** Upon getting the message $\{M_5\}$, the server $S_j$ checks whether $h(SK_{ij}||N_1||N_2)$ is similar to $M_5$. If this condition holds, the server $S_j$ and the user $U_i$ communicates with session key $SK_{ij}$.

### 3.4 Password Change Phase

This phase is invoked whenever $U_i$ wants to change his password $PW_i$ to a new password $PW_i^{new}$ .

**Step P1:** The user $U_i$ inserts his smart card and inputs his identity $ID_i$ and password $PW_i$, and imprints his biometrics $BIO_i^*$ at sensor. Then the sensor sketches $BIO_i^*$ and recovers $R_i$ from $Rep(BIO_i^*, P_i) \rightarrow R_i$.

**Step P2:** The smart card calculates $RPW_i = h(PW_i||R_i)$ and checks whether $h(ID_i||RPW_i)$ is similar to $V_i$. If it holds, smart card asks $U_i$ for a new password.

**Step P3:** The user $U_i$ input the new password $PW_i^{new}$ and the smart card further computes

$$\begin{aligned} RPW_i^{new} &= h(PW_i^{new}||R_i), \\ B_i^{new} &= B_i \oplus RPW_i \oplus RPW_i^{new}, \\ C_i^{new} &= C_i \oplus RPW_i \oplus RPW_i^{new}, \\ V_i^{new} &= h(ID_i||RPW_i^{new}). \end{aligned}$$

**Step P4:** The smart card then replaces $B_i$ with $B_i^{new}$, $C_i$ with $C_i^{new}$, and $V_i$ with $V_i^{new}$ in the memory.

### 3.5 User Revocation/Re-registration Phase

If the user $U_i$ wants to revoke his privilege, he needs to send a revocation request message, his smart card and verification message $\{RPW_i\}$ to the registration center $RC$ via a secure channel. The detailed procedure of this phase is shown as follows.

**Step R1:** $RC$ checks whether $U_i$ is valid. If it holds, $RC$ modifies the corresponding entry by setting $< ID_i, N_i = 0 >$.

**Step R2:** $RC$ executes the steps described in the section of user registration phase and replaces$< ID_i, N_i = N_i + 1 >$ with $< ID_i, N_i >$ to help $U_i$ re-register.

## 4 Security Analysis of Wang *et al.*'s Scheme

In Wang *et al.*'s scheme, the security analysis of scheme demonstrated that their scheme satisfies the desirable security requirements. Unfortunately, we find that their scheme still has many vulnerabilities. If an attacker colludes with a registered but malicious server and eavesdrops messages between the user $U_i$ and the server $S_j$, he can launches session key disclosure, smart card forgery attack, server spoofing attack and user impersonation attack. He also can forge a current timestamp and initiate DoS attack that attempt to make network resource or machines unavailable. Moreover, a user's behavior is tracked because smart card data $B_i$ in the public channel, which can be easily eavesdropped by adversaries. The details are as follows.

### 4.1 Session Key Disclosure

In Wang *et al.*'s scheme, the registration center $RC$ shares the same pre-shared $PSK$ with all the servers. Once the attacker $Z$ colludes with the registered but malicious server, he can obtain the pre-shared key $PSK$ and launch the session key disclosure. Now we show the reason why Wang *et al.*'s scheme cannot resist to session key disclosure. The attacker intercepts messages $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$, $\{SID_j, M_3, M_4\}$ and calculates the following operations:

$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1 &= RPW_i \oplus M_1 \oplus h(PSK), \\ N_2 &= M_3 \oplus h(AID_i||N_1) \oplus h(PSK), \\ SK_{ij} &= h(AID_i||SID_j||N_1||N_2), \end{aligned}$$

Now, the attacker $Z$ easily derives the current session key $SK_{ij}$ shared between $U_i$ and $S_j$. After that, $S_k$ can decrypt all encrypted information between $U_i$ and $S_j$. Hence, Wang *et al.*'s scheme is vulnerable to session key disclosure.

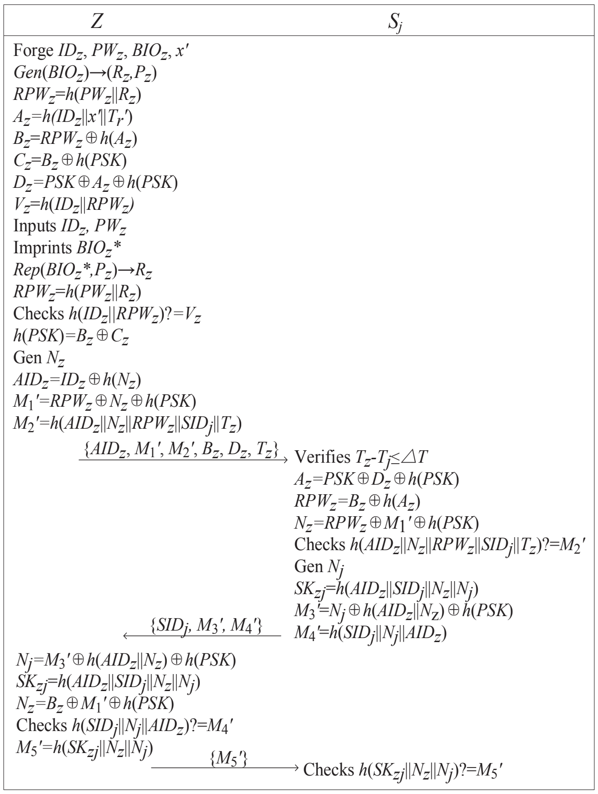


Figure 2: Smart card forgery attack on Wang *et al.*'s Scheme

## 4.2 Smart Card Forgery Attack

As shown in Wang *et al.*'s scheme, any server has the same pre-shared key $PSK$. Under the condition that the attacker $Z$ colludes with a malicious $S_k$, they can forge a smart card to log into any server ($e.g.$, $S_j$) as shown in Figure 2. The procedure is as following:

- $Z$ forges a new identity $ID_z$, password $PW_z$ and personal biometric $BIO_z$, and forges a master key $x'$. Sensor sketches $BIO_z$, extracts $(R_z, P_z)$ from $Gen(BIO_z)$, and stores $P_z$ in the memory.

- $Z$ computes
$$\begin{aligned} RPW_z &= h(PW_z||R_z), \\ A_z &= h(ID_z||x'||T_r'), \\ B_z &= RPW_z \oplus h(A_z), \\ C_z &= B_z \oplus h(PSK), \\ D_z &= PSK \oplus A_z \oplus h(PSK), \\ V_z &= h(ID_z||RPW_z), \end{aligned}$$
then the forged smart card containing $\{B_z, C_z, D_z, V_z, P_z\}$.

- $Z$ inserts the forged smart card and input identity $ID_z$, password $PW_z$ and personal biometric $BIO_z$, sensor sketches $BIO_z$ recovers $R_z$ from $Rep(BIO_z, P_z) \rightarrow R_z$.

- $Z$ computes $RPW_z = h(PW_z||R_z)$ and checks whether $h(ID_z||RPW_z)$ is equal to $V_z$. Obviously, $h(ID_z||RPW_z)$ is equal to $V_z$. Then, $Z$ computes $h(PSK) = B_z \oplus C_z$, generates a random number $N_z$, computes
$$\begin{aligned} AID_z &= ID_z \oplus h(N_z), \\ M_1' &= RPW_z \oplus N_z \oplus h(PSK), \\ M_2' &= h(AID_z||N_z||RPW_z||SID_j||T_z). \end{aligned}$$
Then, the forged smart card send the request message $\{AID_z, M_1', M_2', B_z, D_z, T_z\}$ to $S_j$ via a public channel.

Upon receiving the message $\{AID_z, M_1', M_2', B_z, D_z, T_z\}$, $S_j$ verifies whether $T_z - T_j$ is less than $\triangle T$. If the condition holds, the server $S_j$ computes
$$\begin{aligned} A_z &= PSK \oplus D_z \oplus h(PSK), \\ RPW_z &= B_z \oplus h(A_z), \\ N_z &= RPW_z \oplus M_1' \oplus h(PSK), \end{aligned}$$
and checks whether $h(AID_z||N_z||RPW_z||SID_j||T_z)$ is equal to $M_2'$. The server $S_j$ generates a random number $N_j$ , and computes
$$\begin{aligned} SK_{zj} &= h(AID_z||SID_j||N_z||N_j), \\ M_3' &= N_j \oplus h(AID_z||N_z) \oplus h(PSK), \\ M_4' &= h(SID_j||N_j||AID_z). \end{aligned}$$
Finally, $S_j$ sends the message $\{SID_j, M_3', M_4'\}$ to the attacker $Z$.

When receiving the replay message $\{SID_j, M_3', M_4'\}$, $Z$ computes
$$\begin{aligned} N_j &= M_3' \oplus h(AID_z||N_z) \oplus h(PSK), \\ SK_{zj} &= h(AID_z||SID_j||N_z||N_j) \end{aligned}$$
and $N_z = B_z \oplus M_1' \oplus h(PSK)$. Obviously, $h(SID_j||N_j||AID_z)$ is equal to $M_4'$. The attacker $Z$ computes $M_5' = h(SK_{zj}||N_z||N_j)$ and sends $\{M_5'\}$ to the server $S_j$.

At last, the attacker successfully logs into the server $S_j$ using the forged smart card. Therefore, Wang *et al.*'s scheme cannot resist smart card forgery attack.

## 4.3 Server Spoofing Attack

In the server registration phase, $RC$ transmits the same pre-shared key $PSK$ to every server, thus an authorized but malicious server $S_k$ can impersonate as any server ($e.g.$, $S_j$) to deceive any legal user after he intercepts the request message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$. $S_k$ masquerades as the server $S_j$ to spoof $U_i$ in the following way.

- $S_k$ can retrieve
$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1 &= RPW_i \oplus M_1 \oplus h(PSK), \end{aligned}$$

by capturing the message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$.

- $S_k$ generates $N_2'$, calculates

$$\begin{aligned} SK_{ij}' &= h(AID_i||SID_j||N_1||N_2'), \\ M_3' &= N_2' \oplus h(AID_i||N_1) \oplus h(PSK), \\ M_4' &= h(SID_j||N_2'||AID_i), \end{aligned}$$

then sends $\{SID_j, M_3', M_4'\}$ to $U_i$ via a public channel.

- After receiving the message, $U_i$ computes

$$\begin{aligned} N_2' &= h(AID_i||N_1) \oplus M_3' \oplus h(PSK), \\ SK_{ij}' &= h(AID_i||SID_j||N_1||N_2'). \end{aligned}$$

Then the user $U_i$ verifies the condition

$$h(SID_j||N_2'||AID_i)? = M_4'.$$

Evidently, this condition holds. The user $U_i$ mistakenly thinks that he is communicating with $S_j$.

At last, the authorized malicious server $S_k$ can successfully launch the server spoofing attack.

### 4.4 User Impersonation Attack

As shown in Wang *et al.*'s scheme, the user $U_i$ transmits the request message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$ to the server $S_j$, $S_j$ can retrieve the user's identity $ID_i = AID_i \oplus h(N_1)$ through computing

$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1 &= RPW_i \oplus M_1 \oplus h(PSK). \end{aligned}$$

Once the server reveals $ID_i$ and $RPW_i$ to the attacker $Z$, $Z$ can impersonate as the user, the details are shown as below.

- The attacker $Z$ generates a random number $N_1'$ and computes

$$\begin{aligned} AID_i' &= ID_i \oplus h(N_1'), \\ M_1' &= RPW_i \oplus N_1' \oplus h(PSK), \\ M_2' &= h(AID_i'||N_1'||RPW_i||SID_j||T_i'). \end{aligned}$$

Finally, $Z$ delivers his login request message $\{AID_i', M_1', M_2', B_i, D_i, T_i'\}$ to the server $S_j$.

- Upon the server $S_j$ receiving the message, $S_j$ checks whether $T_j$-$T_i'$<=$\triangle T$ is valid. If the condition holds, $S_j$ computes

$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1' &= RPW_i \oplus M_1' \oplus h(PSK). \end{aligned}$$

$S_j$ checks whether $h(AID_i'||N_1'||RPW_i||SID_j||T_i)$ is similar to $M_2'$.

- The server $S_j$ generates a random number $N_2$, computes

$$\begin{aligned} SK_{ij}' &= h(AID_i'||SID_j||N_1'||N_2), \\ M_3' &= N_2 \oplus h(AID_i'||N_1') \oplus h(PSK), \\ M_4' &= h(SID_j||N_2||AID_i'), \end{aligned}$$

and sends $\{SID_j, M_3', M_4'\}$ to $Z$ over a public channel.

- The attacker $Z$ computes

$$\begin{aligned} N_2 &= M_3' \oplus h(AID_i'||N_1') \oplus h(PSK), \\ SK_{ij}' &= h(AID_i'||SID_j||N_1'||N_2), \\ N_1' &= B_i \oplus M_1' \oplus h(PSK). \end{aligned}$$

Obviously, $h(SID_j||N_2\ ||AID_i')$ is equal to $M_4'$. Then, the attacker $Z$ calculates $M_5' = h(SK_{ij}'||N_1'\ ||N_2)$ and sends $\{M_5'\}$ to $S_j$ via a public channel.

- The server $S_j$ checks whether $h(SK_{ij}'||N_1'||N_2)$ is equal to $M_5'$. If it holds, $S_j$ uses the session key $SK_{ij}'$ to communicate with $Z$ and believes that he is the legal user $U_i$.

Thus, Wang *et al.*'s scheme cannot resist to user impersonation attack.

### 4.5 Denial of Service Attack

From the login and authentication phase of Wang *et al.*'s scheme, we find that any attacker $Z$ who colludes with the malicious server can easily forge a login request message and replay it to the server $S_j$. In Wang *et al.*'s scheme, the attacker can launch DoS attack as described below:

- Upon intercepting the message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$, the attacker $Z$ computes

$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1 &= RPW_i \oplus M_1 \oplus h(PSK). \end{aligned}$$

- The attacker $Z$ generates a current timestamp $T_i'$ and calculates $M_2' = h(AID_i||N_1||RPW_i||SID_j||T_i')$. $Z$ sends $\{AID_i, M_1, M_2', B_i, D_i, T_i'\}$ to $S_j$.

- Upon receiving the message from $Z$, $S_j$ computes

$$\begin{aligned} A_i &= PSK \oplus D_i \oplus h(PSK), \\ RPW_i &= B_i \oplus h(A_i), \\ N_1 &= RPW_i \oplus M_1 \oplus h(PSK) \end{aligned}$$

and verifies whether

$$h(AID_i||N_1||RPW_i||SID_j||T_i')$$

is similar to $M_2'$. Obviously, the verification holds.

- $S_j$ generates a number $N_j$ and computes

$$SK_{ij} = h(AID_i||SID_j||N_1||N_j),$$
$$M_3 = N_2 \oplus h(AID_i||N_1) \oplus h(PSK),$$
$$M_4 = h(SID_j||N_2||AID_i).$$

- $S_j$ sends message $\{SID_j, M_3, M_4\}$ to the user $U_i$. The attacker $Z$ will intercept the message to terminate the communication.

By this way, the attacker can launch DoS attack on the server $S_j$, which will result in the computing and communication loss of the server.

### 4.6 No Provision of User Anonymity

The user anonymity is a desirable property for remote user authentication. Generally, the scheme with user anonymity contains two aspects of content, one is the user's real identity cannot be revealed by the attacker, another is that the user cannot be traced by the attacker. In Wang *et al.*'s scheme, Any server authenticated with the user can recover the identity of the user.Any server authenticated with the user can recover the identity of the user through computing

$$A_i = PSK \oplus D_i \oplus h(PSK),$$
$$RPW_i = B_i \oplus h(A_i),$$
$$N_1 = RPW_i \oplus M_1 \oplus h(PSK),$$
$$ID_i = AID_i \oplus h(N_1),$$

which $D_i$ and $AID_i$ are intercepted from the message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$. Thus, the identity of the user is leaked to the server. Moreover, in each login phase, the user $U_i$ submits the login request message $\{AID_i, M_1, M_2, B_i, D_i, T_i\}$ to the server $S_j$. On this message, $B_i = RPW_i \oplus h(A_i)$ and $D_i = PSK \oplus A_i \oplus h(PSK)$ are unique for each user. The attacker can distinguish whether two sessions are launched by the same user. Therefore, the attacker can trace the user by $B_i$ and $D_i$. Accordingly, Wang *et al.'s* scheme fails to preserve user anonymity.

## 5 The Proposed Protocol

In this section, based on the cryptanalysis of Wang *et al.*'s scheme, we present our robust biometrics-based multi-server authentication scheme with smart card using public-key encryption technique, where $Pub_{sj}$ is the public key of $S_j$, $Pri_{sj}$ is the secret key of $S_j$. The proposed scheme consists of three phases: Registration phase, login and authentication phase and password change phase . There are also three participants: The user $U_i$, the server $S_j$ and the registration center $RC$.

### 5.1 Registration Phase

In our proposed protocol, the registration phase consists of two sub-phases, the server registration phase and the user registration phase. In this phase, the server and the user should register themselves to the registration center $RC$ and obtains secret information to initial system.

#### 5.1.1 Server Registration Phase

The server $S_j$ sends a registration request to $RC$ in order to become an authorized server. This registration process consists of following steps:

**Step S1:** The server $S_j$ sends a registration request message $\{SID_j\}$ to $RC$.

**Step S2:** The registration center $RC$ replies with $\{h(PSK||SID_j)\}$ to the server $S_j$, which can be used in further phases of authentication.

#### 5.1.2 User Registration Phase

When a user wants to access the services of servers, he must register himself, as shown in Figure 3. This registration process according to the following steps:

**Step R1:** The user $U_i$ freely selects his identity $ID_i$, which uniquely identities the user's identity, password $PW_i$ and scans his biometrics $BIO_i$ at sensor terminal to gets $R_i$ from $Gen(BIO_i) \rightarrow (R_i, P_i)$. Then the user $U_i$ generates a random number $b_i$ and computes $AID_i = h(ID_i||b_i)$ and $RPW_i = h(PW_i||R_i||b_i)$. At last, the user $U_i$ sends a request message $\{AID_i, RPW_i\}$ to $S_j$ via a secure channel.

**Step R2:** Upon getting the message, $RC$ computes

$$B_{ij} = h(AID_i||h(PSK||SID_j)),$$
$$C_{ij} = B_{ij} \oplus RPW_i,$$
$$V_i = h(AID_i||RPW_i).$$

**Step R3:** The $RC$ selects a base point $G$ and stores $\{< SID_j, C_{ij} >, V_i, G, h(\cdot)\}$ into the smart card and delivers it to the user $U_i$ via a secure channel.

**Step R4:** Upon getting the message, the user $U_i$ stores $\{b_i, P_i\}$ into the smart card.

### 5.2 Login and Authentication Phase

When a user $U_i$ wants to access the services of remote server $S_j$, he launches the login request by inserting smart card , and inputting $ID_i$ and $PW_i$. Next, the user $U_i$ imprints his biometric information $BIO_i$ at a sensor. After that, sensor sketches user $U_i$'s biometric information $BIO_i$ and recovers the unpredictable binary string $R_i$ from $Rep(BIO_i, P_i) \rightarrow R_i$. Then, as shown in Fig 4, the login and authentication procedure is performed as follows:

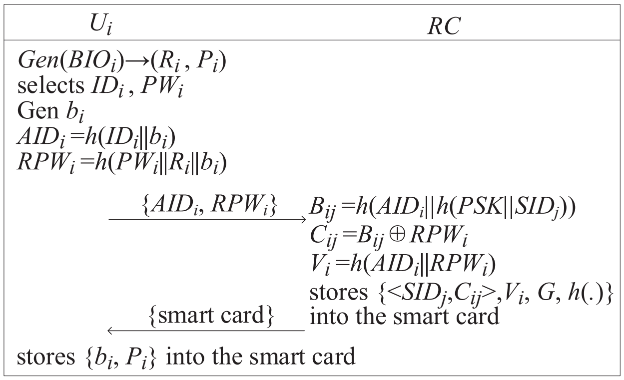

Figure 3: User registration phase of our scheme

**Step L1:** The smart card computes $AID_i = h(ID_i||b_i)$, $RPW_i = h(PW_i||R_i||b_i)$, and verifies whether $V_i$ is equal to $h(AID_i||RPW_i)$. If $V_i$ is invalid, smart card terminates the communication; otherwise, the user $U_i$ generates a random number $N_1$ and calculates

$$\begin{aligned} B_{ij} &= RPW_i \oplus C_{ij}, \\ D_i &= N_1 \cdot G, \\ F_{ij} &= B_{ij} \oplus D_i, \\ M_1 &= E_{Pub_{sj}}(AID_i||T_i), \\ M_2 &= h(AID_i||B_{ij}||D_i||T_i). \end{aligned}$$

Then the user $U_i$ sends the login request message $\{F_{ij}, M_1, M_2, T_i\}$ to the server $S_j$, where $T_i$ is a current timestamp.

**Step L2:** Upon receiving the message from the user $U_i$, the server $S_j$ checks whether $T_i$ - $T_j$ is less than $\triangle T$, where $\triangle T$ is the time interval and $T_j$ is the time when $S_j$ receives the login request message. The server $S_j$ computes

$$\begin{aligned} AID_i||T_i &= D_{Pri_{sj}}(M_1), \\ B_{ij} &= h(AID_i||h(PSK||SID_j)), \\ D_i &= B_{ij} \oplus F_{ij}, \end{aligned}$$

and verifies whether the condition $M_2$ is equal to $h(AID_i||B_{ij}||D_i||T_i)$. If the condition holds, the server $S_j$ authenticates the user $U_i$, otherwise the process can be terminated.

**Step L3:** The server $S_j$ further generates a random number $N_2$ and computes

$$\begin{aligned} D_j &= N_2 \cdot G, \\ P_j &= N_2 \cdot D_i, \\ SK_{ij} &= h(AID_i||SID_j||P_j||D_j), \\ M_3 &= h(SK_{ij}||AID_i||D_j). \end{aligned}$$

Furthermore, the server $S_j$ sends the response message $\{M_3, D_j\}$ to the user $U_i$.

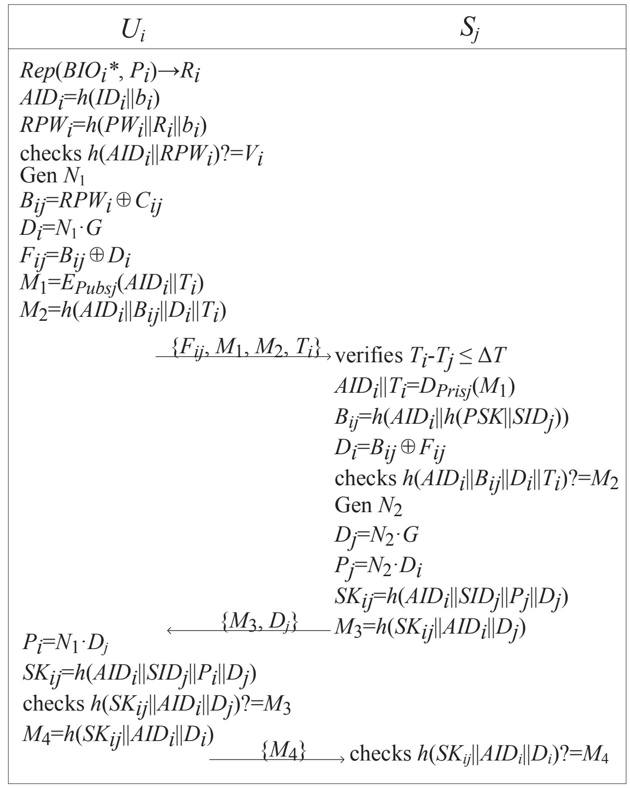

Figure 4: Login and authentication phase of our scheme

**Step L4:** After receiving the message $\{M_3, D_j\}$, the user $U_i$ computes

$$\begin{aligned} P_i &= N_1 \cdot D_j, \\ SK_{ij} &= h(AID_i||SID_j||P_i||D_j), \end{aligned}$$

and verifies whether the condition $M_3$ is similar to $h(SK_{ij}||AID_i||D_j)$. If the condition holds, the user $U_i$ authenticates the remote server $S_j$, otherwise the process is terminated. Then, the user computes $M_4 = h(SK_{ij}||AID_i||D_i)$ and sends the message $\{M_4\}$ to the server $S_j$.

**Step L5:** Upon receiving the message $\{M_4\}$, the server $S_j$ verifies whether $M_4$ is equal to $h(SK_{ij}||AID_i|| D_i)$. If not , the server $S_j$ terminates the communication. Otherwise, the user $U_i$ and the server $S_j$ can use the current session key $SK_{ij}$ for securing communication.

## 5.3 Password Change Phase

This procedure invokes when a user $U_i$ wishes to update his password. The user $U_i$ can change his password as follows:

**Step P1:** The user $U_i$ inputs $ID_i$ and $PW_i$, and imprints his biometrics $BIO_i$. The sensor sketches $BIO_i$ and recovers $R_i$ from $Rep(BIO_i, P_i)$→$R_i$.

**Step P2:** The smart card computes

$$AID_i = h(ID_i||b_i),$$
$$RPW_i = h(PW_i||R_i||b_i),$$

and then verifies whether $V_i$ is similar to $h(AID_i||RPW_i)$. If this verification is valid, the smart card asks user $U_i$ for a new password. Otherwise, password change phase is terminated immediately by the smart card.

**Step P3:** The user $U_i$ chooses a new password $PW_i^{new}$ and generates a random number $b_i^{new}$. Then $U_i$ computes

$$AID_i^{new} = h(ID_i||b_i^{new}),$$
$$RPW_i^{new} = h(PW_i^{new}||R_i||b_i^{new}),$$
$$C_i^{new} = B_{ij} \oplus RPW_i^{new},$$
$$V_i^{new} = h(AID_i^{new}||RPW_i^{new}).$$

**Step P4:** In the memory, smart card respectively replaces $C_i$ with $C_i^{new}$ and $V_i$ with $V_i^{new}$.

# 6 Analysis of the Proposed Protocol

In this section, we first present security analysis of our scheme, and then analyze its performance efficiency by comparing it with previous related works.

## 6.1 User Anonymity

In our scheme, the real identity of user is not revealed throughout all the phases of communication. In the user registration phase, $U_i$ submits $AID_i = h(ID_i||b_i)$ to $RC$, which the real identity is protected with a one-way hash function and random number $b_i$. During the login phase, the messages $\{F_{ij}, M_2, T_i\}$, $\{M_3, D_j\}$ and $\{M_4\}$ are converted as dynamic in the form of $D_i = N_1 \cdot G$ and $D_j = N_2 \cdot G$, where $N_1$ and $N_2$ are random numbers. The message $\{M_1\}$ is converted as dynamic by freshness timestamp $T_i$. All the messages between the user and the server are dynamic and dose not disclose the identity of $U_i$. Hence, our scheme can provide user anonymity.

## 6.2 Resistance to User Impersonation Attack

Consider a scenario where the attacker $U_z$ acts as a legitimate one and proceeds with the authentication procedures. If the attacker $U_z$ wants to impersonate a legitimate user $U_i$, he requires to build a login request message $\{F_{ij},\ M_1,\ M_2,\ T_i\}$, where $F_{ij} = B_{ij} \oplus D_i$, $M_1 = E_{Pub_{sj}}(AID_i||T_i)$ and $M_2 = h(AID_i||B_{ij}||D_i||T_i)$. However, the attacker cannot compute $D_i = N_1 \cdot G$ because $N_1$ is the user generated random number. Moreover, in order to compute $AID_i$ and $B_{ij}$, the attacker requires user's identity $ID_i$ and password $PW_i$, which are unobtainable. So our scheme is secure against the user impersonation attack.

## 6.3 Resistance to Server Spoofing Attack

In the proposed scheme, if the malicious server $S_k$ wants to authenticate with the user $U_i$ by impersonating as the server $S_j$, $S_k$ needs to compute $B_{ij} = h(AID_i||h(PSK||SID_j))$. Although $S_k$ can capture parameters $AID_i$ and $SID_j$, it is impossible for $S_k$ to retrieve the pre-share key $PSK$ from the registration center $RC$. Because on the server registration phase, the registration center $RC$ transmits $h(PSK||SID_j)$ to $S_k$, rather than $PSK$. Therefore, our proposed scheme withstands the server spoofing attack.

## 6.4 Resistance to Session Key Disclosure

In our scheme, the session key is defined as $SK_{ij} = h(AID_i||SID_j||P_j||D_j) = h(AID_i||SID_j||P_i||D_j)$, where $P_j = P_i = N_1 \cdot N_2 \cdot G$ and $D_j = N_2 \cdot G$ with randomly chosen number $N_1$ and $N_2$ . We can see $N_1$ and $N_2$ are random nonce generated by user and server. Obviously, attacker cannot get $N_1$ and $N_2$. Moreover, the attacker cannot get $AID_i$ due to only the server $S_j$ can decrypt the message $M_1 = E_{Pub_{sj}}(AID_i||T_i)$ using the private key $Pri_{sj}$ of the server. Thus, our scheme can resist session key disclosure.

## 6.5 Resistance to Smart Card Forgery Attack

In our proposed scheme, the smart card contains $\{C_{ij}, V_i, b_i, P_i\}$. If the attacker attempts to forge smart card, he forges a new identity $ID_z$, password $PW_z$ and personal biometric $BIO_z$. Sensor sketches $BIO_z$, extracts $(R_z, P_z)$ from $Gen(BIO_z) \to (R_z, P_z)$, and stores $P_z$ into smart card. The attacker generates a random number $b_z$, and calculates $AID_z = h(ID_z||R_z||b_z)$ and $RPW_z = h(PW_z||R_z||b_z)$. To forge parameter $C_{zj}$, the attacker attempt to compute $B_{zj} = h(AID_z||h(PSK||SID_j))$. Unfortunately, the attacker cannot retrieve $PSK$ since $RC$ calculates $h(PSK||SID_j)$ for each $S_j$. So, the attacker cannot forge $C_{zj}$. Thus, our scheme can resist smart card forgery attack.

## 6.6 Resistance to Privileged Insider Attack

During user registration phase of our proposed scheme, $U_i$ dose not submits identity $ID_i$ and password $PW_i$ in plaintext form to the registration server $RC$. $U_i$ submits $AID_i = h(ID_i||b_i)$ and $RPW_i = h(PW_i||R_i||b_i)$ to $RC$, where $b_i$ is a random number generated by the user $U_i$. Hence, an insider cannot obtain the original credentials of any user. In this way, our proposed protocol attains resistance to privileged insider attacks.

Table 2: Efficiency Comparison

| | User side | Server side | Total | Times(ms) |
|---|---|---|---|---|
| Reddy et al.[20] | $8T_h$+$2T_{epm}$ | $5T_h$+$1T_{epm}$ | $13T_h$+$3T_{epm}$ | 6.693 |
| Lu et al.[14] | $4T_h$+$3T_{re}$ | $14T_h$+$3T_{rd}$ | $18T_h$+$3T_{re}$+$3T_{rd}$ | 12.1689 |
| Mishra et al.[15] | $6T_h$+$2T_{epm}$ | $10T_h$+$1T_{epm}$ | $16T_h$+$3T_{epm}$ | 6.7148 |
| Wang et al.[24] | $12T_h$ | $8T_h$ | $20T_h$ | 0.08 |
| Our scheme | $9T_h$+$1T_{re}$ | $5T_h$+$1T_{rd}$ | $14T_h$+$1T_{re}$+$1T_{rd}$ | 4.0533 |

## 6.7 Resistance to Replay Attack

If the attacker intercepts the communication message $\{F_{ij}, M_1, M_2, T_i\}$ between $U_i$ and $S_j$, he/she transmits $\{F_{ij}, M_1, M_2, T_i'\}$ to the server $S_j$, where $T_i'$ is a current timestamp. Upon receiving the response message, $S_j$ computes $M_2' = h(AID_i||B_{ij}||D_i||T_i')$ and verifies whether $M_2'$ is equal to $M_2$. Here, $S_j$ identifies it as a fake response from the malicious user due to $M_2' \neq M_2$ and terminates the session immediately. Hence, our protocol is secure against replay attack.

## 6.8 Resistance to Password Guessing Attack

The attacker may try to guess the password $PW_i$ from the extracted smart card stored parameters $\{C_{ij}, V_i, h(\cdot)\}$. The stored parameter contains the password $PW_i$ in the form $RPW_i = h(PW_i||R_i||b_i)$, where $R_i$ froms $Gen(BIO_i) \rightarrow (R_i, P_i)$. The attacker attempts to verify the condition $V_i ? = h(AID_i||RPW_i)$ while constantly guessing $PW_i$. The attacker needs the value of $ID_i$ and $R_i$ of $U_i$ in order to achieve the password guessing attack. However, the value of $R_i$ is nowhere stored and the attacker cannot know $ID_i$. As a result, he cannot guess $PW_i$. Therefore, our scheme resist to password guessing attack.

## 6.9 Perfect Forward Secrecy

The session key of the proposed protocol is computed as $S_{ij} = h(AID_i||SID_j||P_j||D_j) = h(AID_i||SID_j||P_i||D_j)$, where $P_j = P_i = N_1 \cdot N_2 \cdot G$ and $D_j = N_2 \cdot G$. Although the long term key is compromised with the attacker, he still cannot construct a valid session key due to following reason. The parameter $P_i$, $P_j$ and $D_j$ are dynamic due to its association with random generated number $N_1$ and $N_2$, which is not possible to extract. Therefore, the proposed protocol provides perfect forward secrecy.

## 6.10 Performance and Functionality Comparisons

In this section, we compare our proposed protocol with several related schemes [14, 15, 20, 24]. In Table 2, we provide the comparison based on the key security of these schemes, while we compare their efficiency in terms of computation. According to Kilinc *et al.*'s [31] estimation, the average running time of $T_h$ is about 0.0004ms, $T_{re}$ is 3.8500, $T_{rd}$ is 0.1925ms and $T_{epm}$ is 2.229ms. Table 2 illustrates the comparative performance of our improved scheme and previously proposed schemes. From that, we can see our proposed scheme is more efficient than Reddy *et al.*'s scheme, Lu *et al.*'s scheme and Mishra *et al.*'s scheme. The following notations are used in Table 2.

- $T_h$: The execution time of one-way hash;
- $T_{re}$: RSA encryption;
- $T_{rd}$: RSA decryption;
- $T_{epm}$: The time for executing a scalar multiplication operation of elliptic curve.

We perform a comparative functional analysis of previous schemes, which is illustrated in Table 3. For fair comparison, we use the objective third-party evaluation metrics, where refer to Wang *et al.*'s scheme [26]. As illustrated in Table 3, our scheme provides all the 15 criteria while maintaining reasonable efficiency, all the other schemes fail to achieve at least one critical criterion. Thus, we can find that our proposed scheme is more secure and provides more functionality requirements than the other related schemes.

# 7 Conclusions

In this paper, we analyzed Wang *et al.*'s smart card based multi-server authentication scheme. Our analysis reveals its inherent security vulnerabilities, i.e., session key disclosure, smart card forgery attack, server spoofing attack, user impersonation attack, DoS attack and no provision of user anonymity. In addition, this paper proposed a robust biometrics-based multi-server authentication scheme with smart cards using public-key encryption techniques. The mutual authentication of the proposed protocol achieved significant features such as biometric authentication, public-key encryption techniques, with less computational and communication cost. Furthermore, the comparison results evidently indicate that our protocol is more secure than other schemes. Thus, our protocol is more feasible for practical applications.

Table 3: Security Comparison

| | Reddy et al.[20] | Lu et al.[14] | Mishra et al.[15] | Wang et al.[24] | Our scheme |
|---|---|---|---|---|---|
| C1: *No password verifier table* | Yes | Yes | Yes | Yes | Yes |
| C2: *Password Friendly* | Yes | Yes | Yes | Yes | Yes |
| C3: *No password exposure* | Yes | Yes | Yes | Yes | Yes |
| C4: *No smart card loss attack* | Yes | Yes | Yes | No | Yes |
| C5: *Resistance to known attack* | No | No | No | No | Yes |
| C6: *Sound repairability* | Yes | Yes | Yes | Yes | Yes |
| C7: *Provide key agreement* | No | Yes | Yes | No | Yes |
| C8: *No clock synchronization* | Yes | Yes | Yes | No | Yes |
| C9: *Timely typo detection* | Yes | Yes | Yes | Yes | Yes |
| C10: *Mutual authentication* | Yes | No | No | No | Yes |
| C11: *User anonymity* | Yes | No | Yes | No | Yes |
| C12: *Forward secercy* | Yes | Yes | No | No | Yes |
| C13: *Resistance to insider attack* | No | Yes | Yes | No | Yes |
| C14: *Resistance to verifier attack* | Yes | Yes | No | Yes | Yes |
| C15: *Provide re-registration phase* | No | No | No | Yes | Yes |

## Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 61962022), Key Research and Development Plan of Jiangxi Province (No. 20192BBE50077), and Excellent Scientific and Technological Innovation Teams of Jiangxi Province (No. 20181BCB24009).

## References

[1] P. Chandrakar and H. Om, "A secure and robust anonymous three-factor remote user authentication scheme for multi-server environment using ECC," *Computer Communications*, vol. 110, 2017.

[2] H. M. Chen, J. W. Lo, C. K. Yeh, "An efficient and secure dynamic ID-based authentication scheme for telecare medical information systems," *Journal of Medical Systems*, vol. 36, no. 6, pp. 3907–3915, 2012.

[3] T. Y. Chen, M. S. Hwang, C. C. Lee, J. K. Jan, "Cryptanalysis of a secure dynamic ID based remote user authentication scheme for multi-server environment," in *Fourth International Conference on Innovative Computing, Information and Control (ICICIC'09)*, pp. 725–728, IEEE, 2009.

[4] T. Y. Chen, C. C. Lee, M. S. Hwang, J. K. Jan, "Towards secure and efficient user authentication scheme using smart card for multi-server environments," *Journal of Supercomputing*, vol. 66, no. 2, pp. 1008–1032, Nov. 2013.

[5] M. C. Chuang and M. C. Chen, "An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics," *International Journal of Network Security*, vol. 18, no. 5, pp. 997–1000, 2014.

[6] T. H. Feng, C. H. Ling, and M. S. Hwang, "Cryptanalysis of Tan's improvement on a password authentication scheme for multi-server environments," *International Journal of Network Security*, vol. 16, no. 4, pp. 318–321, 2014.

[7] H. Guo, P. Wang, X. Zhang, Y. Huang, and F. Ma, "A robust anonymous biometric-based authenticated key agreement scheme for multi-server environments," *Plos One*, vol. 12, no. 11, pp. e0187403, 2017.

[8] D. He and S. Wu, "Security flaws in a smart card based authentication scheme for multi-server environment," *Wireless Personal Communications*, vol. 70, no. 1, pp. 323–329, 2013.

[9] M. S. Hwang, Li-Hua Li, "A New Remote User Authentication Scheme Using Smart Cards", *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28–30, Feb. 2000.

[10] Q. Jiang, Z. Ma, and G. Li, "A privacy enhanced authentication scheme for telecare medical information systems," *Journal of Medical Systems*, vol. 37, no. 1, pp. 9897, 2013.

[11] M. K. Khan, S. K. Kim, and K. Alghathbar, "Cryptanalysis and security enhancement of a 'more efficient secure dynamic ID-based remote user authentication scheme," *Computer Communications*, vol. 34, no. 3, pp. 305-309, 2011.

[12] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 2, no. 24, pp. 770–772, 1981.

[13] C. H. Ling and M. S. Hwang, "A secure and efficient one-time password authentication scheme for WSN," *International Journal of Network Security*, vol. 19, no. 2, pp. 177–181, 2017.

[14] Y. Lu, L. Li, H. Peng, and Y. Yang, "A biometrics and smart cards-based authentication scheme for multi-server environments," *Security & Communication Networks*, vol. 8, no. 17, pp. 3219–3228, 2015.

[15] D. Mishra, "Design and analysis of a provably secure multi-server authentication scheme," *Wireless Personal Communications*, vol. 86, no. 3, pp. 1095–1119, 2016.

[16] D. Mishra, A. K. Das, and S. Mukhopadhyay, "A secure user anonymity-preserving biometric-based multi-server authenticated key agreement scheme using smart cards," *Expert Systems with Applications*, vol. 41, no. 18, pp. 8129–8143, 2014.

[17] J. Moon, D. Lee, Y. Lee, and D. Won, "Improving biometric-based authentication schemes with smart card revocation/reissue for wireless sensor networks," *Sensors*, vol. 17, no. 5, pp. 1–24, 2017.

[18] S. Qiu, G. Xu, H. Ahmad, and Y. Guo, "An enhanced password authentication scheme for session initiation protocol with perfect forward secrecy," *Plos One*, vol. 13, no. 3, pp. e0194072, 2018.

[19] C. Quan, J. Jung, J. Kim, Q. Sun, D. Lee, and D. Won, "Cryptanalysis and improvement of a biometric and smart card based remote user authentication scheme," in *International Conference on Ubiquitous Information Management and Communication*, pp. 50, 2017.

[20] A. G. Reddy, A. K. Das, V. Odelu, and K. Y. Yoo, "An enhanced biometric based authentication with key-agreement protocol for multi-server architecture based on elliptic curve cryptography," *Plos One*, vol. 11, no. 5, pp. e0154308, 2016.

[21] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "A self-verifiable password based authentication scheme for multi-server architecture using smart card," *Wireless Personal Communications*, vol. 96, no. 18, pp. 1–25, 2017.

[22] W. Tao, J. Nan, and M. A. Jianfeng, "Cryptanalysis of two dynamic identity based authentication schemes for multi-server architecture," *China Communications*, vol. 11, no. 11, pp. 125–134, 2014.

[23] W. Tao, J. Nan, and M. A. Jianfeng, "Cryptanalysis of a biometric-based multi-server authentication scheme," *International Journal of Security and its Application*, vol. 10, no. 2, pp. 163–170, 2016.

[24] C. Wang, X. Zhang, and Z. Zheng, "Cryptanalysis and improvement of a biometric-based multi-server authentication and key agreement scheme," *Plos One*, vol. 11, no. 2, pp. e0149173, 2016.

[25] D. Wang, D. He, W. Ping, and C. H. Chu, "Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment," *IEEE Transactions on Dependable Secure Computing*, vol. 12, no. 4, pp. 428–442, 2015.

[26] D. Wang, W. Li, and W. Ping, "Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 1–1, 2018.

[27] D. Wang, L. I. W. Ting, P. Wang, "Crytanalysis of three anonymous authentication schemes for multi-server environment," *Journal of Software*, vol. 29, no. 7, pp. 1937–1952, 2018.

[28] Y. Y. Wang, J. Y. Liu, F. X. Xiao, and J. Dan, "A more efficient and secure dynamic ID-based remote user authentication scheme," *Computer Communications*, vol. 32, no. 4, pp. 583–585, 2009.

[29] H. Wijayanto and M. S. Hwang, "Improvement on timestamp-based user authentication scheme with smart card lost attack resistance," *International Journal of Network Security*, vol. 17, no. 2, pp. 160–164, 2015.

[30] F. Wu and L. Xu, "Security analysis and improvement of a privacy authentication scheme for telecare medical information systems," *Journal of Medical Systems*, vol. 37, no. 4, pp. 9958, 2013.

[31] T. Yanik and H. H. Kilinc, "A survey of sip authentication and key agreement schemes," *IEEE Communications Surveys Tutorials*, vol. 16, no. 2, pp. 1005–1023, 2014.

# Biography

**Tao Wan** received her B.S. degree in Mathematics from Hunan University, Changsha, China, and received her M.S. and Ph.D. degree in Computer Science from Xidian University, Xi'an, China. She is now an associate professor at East China Jiaotong University. Her research interests include cryptography, network and information security, e-commerce security technology.

**Xiaochang Liu** received her B.S. degree in Software Engineering from North University of China, Taiyuan, China. She is currently a M.S. candidate at East China Jiaotong University, Nanchang, China. Her research interests include network and information security, e-commerce security technology.

**Weichuan Liao** received his B.S. and M.S. degree in Mathematics from Hunan University, Changsha, China. He is now an associate professor at East China Jiaotong University. His research interests include cryptography, network and information security.

**Nan Jiang** received his Ph.D. degree in Computer Application Technology from Nanjing University of Aeronautics and Astronautics, Nanjing, China. Now he is an associate professor at East China Jiaotong University. From 2013 to 2014 he is a research scholar in Complex Networks and Security Research Lab at Virginia Tech. His research interests include wireless sensor networks, wireless protocol and architecture, distributed computing and complex network theory.