Cryptanalysis and Improvement of a Smart Card Based Authentication Scheme for Multi-server Architecture Using ECC

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]
Huangjiahu E Rd, Qingshanhu Qu, Nanchang Shi, Jiangxi Sheng 330029, China
School of Science, East China Jiaotong University, China[2]
(Email: wantao217@163.com)
(Received Apr. 4, 2018; Revised and Accepted Oct. 18, 2018; First Online June 16, 2019)

Abstract

Authentication and key agreement protocol becomes an important security issue for multi-server architecture. Wei *et al.* demonstrated that Pippal *et al.*'s protocol has several drawbacks and proposed an improved authentication scheme for multi-server 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 user impersonation attack, cannot protect user's anonymity, unable to check user password in time and is also vulnerable to Denial of Service attack. To enhance the security of Wei *et al.*'s protocol, we propose a secure biometric-based authentication scheme for multi-server environment based on elliptic curve cryptography using smart card. Compared with other related schemes, the security analysis and performance evaluation show that our proposed scheme can provide stronger security.

Keywords: Authentication; Biometric-based; Key Agreement; Multi-Server; Smart Card

1 Introduction

With the rapid development of Internet applications, an increasing number of remote user authentication schemes are usually used to provide services to users. In the early, most authentication schemes are based on password. Unfortunately, as widely used in real-life settings, there were vulnerable to some attacks, such as dictionary attack and compromised stolen-verifier attack. To overcome these attacks, smart card based password authentication schemes [2, 3, 6, 10, 13, 15, 23, 31] have been proposed, which become one of the most general authentication scheme. However, most of these schemes based on the single-server, when 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 identity and password. Obviously, it is very difficult and unsafe for users to remember and manage multiple information. In order to solve this problem, authentication schemes for the multi-server environment [5, 7, 11, 12, 18–21, 24–27, 29] have been proposed in recent year.

Recently, Lee *et al.* [16] analyzed Hsiang and Shih's scheme [9] and pointed out that their scheme is vulnerable to masquerade attack and server spoofing attack, and it cannot provide mutual authentication since the clerical error. To overcome the security flaws of Hsiang and Shih's scheme, Lee *et al.* proposed a secure dynamic ID based authentication scheme. But Li *et al.* [17] found that Lee *et al.*'s scheme is still vulnerable to forgery attack and server spoofing attack. Nevertheless, Chang *et al.* [4] indicated that their scheme is sensitive to the forgery attack. In 2013, He and Wu [8] demonstrated that Wang and Ma's scheme [28] is vulnerable to stolen smart card and leak of verifier attack and introduced the improvement scheme. Unfortunately, Pippal *et al.* [22] revealed that their scheme is still susceptible to impersonation attack, privileged insider attack and off-line password guessing attack. To solve above-mentioned security flaws, in 2014, Wei and Liu [30] proposed improvement of a robust smart card authentication scheme for multi-server architecture. But, we identify that Wei *et al.*'s scheme not only is vulnerable to DoS attack, user impersonation attack, but also lacks timely password check and users are easily tracked.

The remainder of this manuscript is organized as follows. We review the robust smart card authentication scheme for multi-server architecture proposed by Wei *et al.* in Section 2. We analyze the security flaws of Wei *et al.*'s scheme in Section 3. We present a proposed protocol in Section 4. We compare the performance of our proposed scheme with the previous schemes in Section 5. We

conclude this paper in Section 6.

2 Review of Wei *et al.*'s Scheme

Here we will review Wei *et al.*'s smart card based authentication scheme for multi-server architecture. The notations used throughout this paper are summarized as Table 1.

Their scheme involves three participants, the login user (U_i), the remote server (S_j) and the registration center (RC). Their scheme can be divided into four phase: initialization phase, registration phase, login and authentication phase and password change phase. We show the login and authentication phases in Figure 1. More details are provided in the following.

Table 1: Notations used in the paper

Symbols	Their meaning
RC	*the registration center*
U_i	*the i_{th} user*
UID_i	*the i_{th} user's identity*
S_j	*the j_{th} application server*
SID_j	*the j_{th} application server's identity*
PW_i	*the user U_i's password*
p and q	*two large prime numbers*
$h(\cdot)$	*a secure one-way hash function*
$\|$	*the concatenation operation*
$\oplus$	*exclusive-OR operation*
x	*random nonce generated by U_i*
y	*random nonce generated by S_j*
SK_{ij}	*section key shared between U_i and S_j*

2.1 Initialization Phase

Step I1: The registration center RC selects two large prime numbers p and q and computes $p = 2q + 1$.

Step I2: The registration center RC chooses a random nonce $g \in Z_p^*$, picks a random number $r_j \in Z_p^*$ as the private key of the remote server $S_j (1 \leq j \leq k)$, and sets $t = g^{\prod_{j=1}^{k} r_j} \bmod p$.

Step I3: The registration center RC selects a secure one-way hash function $h(\cdot) : \{0,1\}^* \rightarrow Z_p^*$.

2.2 Registration Phase

In Wei *et al.*'s scheme, the registration phase consists of two sub-phases, the server registration phase and the user registration. In this phase, the server and the user should register themselves to the registration center RC and obtains secret information to initial the system.

2.2.1 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 application server S_j sends a registration request along with its identity SID_j to the registration center RC, if he/she wishes to become a registered server.

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

Step S3: And then sends $\{r_j, t, p, q, h(\cdot)\}$ to the remote server S_j through a secure channel.

2.2.2 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 freely chooses an identity UID_i, a private password PW_i and a random number b, then transmits the registration request information $\{UID_i, h(PW_i \| b)\}$ to the registration center RC via a secure channel.

Step U2: Upon getting the registration information from U_i, the registration center RC continues to compute $V_{ij} = h(t \| r_j \| UID_i)$, $S_{ij} = V_{ij} \oplus h(UID_i \| h(PW_i \| b))$ when UID_i is valid, otherwise rejects the user registration request.

Step U3: The registration center RC securely issues the smart card containing $\{(S_{i1}, S_{i2}, \ldots, S_{ik}), p, q, h(\cdot)\}$ to the user U_i.

Step U4: After receiving the issued smart card, the user U_i stores the random nonce b into the smart card.

2.3 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 UID_i and password PW_i. Then, as illustrated in Figure 1, the login and authentication mechanism is performed as follows:

Step V1: The smart card first computes

$$V_{ij} = S_{ij} \oplus h(UID_i \| h(PW_i \| b)),$$

then generates a random nonce x and computes

$$\begin{aligned} W_{ij} &= h(UID_i || SID_j)^x \bmod p, \\ W_{ij}^* &= W_{ij} \oplus V_{ij}, \\ R_1 &= h(UID_i || W_{ij}^* || T_i). \end{aligned}$$

The smart card sends the login request message $M_1 = \{UID_i, W_{ij}^*, R_1, T_i\}$ to the remote server S_j.

Step V2: Upon receiving the message from the user U_i, the remote server S_j checks whether UID_i is valid and $T_i' - T_i$ is less than $\triangle T$. Moreover, S_j verifies whether $R_1' = h(UID_i||W_{ij}^*||T_i)$ is equal to R_1. If not, the communication is simply terminated.

Step V3: The remote server S_j chooses a random number y, and first computes

$$\begin{aligned} B_{ij} &= h(UID_i||SID_j)^y \bmod p, \\ V_{ij}' &= h(t||r_j||UID_i), \\ W_{ij}' &= W_{ij}^* \oplus V_{ij}', \\ Z_{ij} &= (W_{ij}')^y \bmod p, \\ R_2 &= h(UID_i||W_{ij}'||B_{ij}||Z_{ij}||T_j). \end{aligned}$$

Furthermore, the remote server S_j sends the response message $M_2 = \{B_{ij}, R_2, T_j\}$ to user U_i.

Step V4: After getting the message M_2, the smart card checks whether $T_i' - T_i \leq \triangle T$, if T_j is valid, the smart card computes $Z_{ij}' = B_{ij}^x \, mod \, p$, and checks whether $R_2' = h(UID_i\|W_{ij}\|B_{ij}\|Z_{ij}'\|T_j)$ is equal to R_2. If not, the smart card terminates the communication.

Step V5: The smart card computes

$$\begin{aligned} SK_{ij} &= h(UID_i\|W_{ij}\|B_{ij}\|Z_{ij}'), \\ R_3 &= h(UID_i||W_{ij}^*||B_{ij}\|Z_{ij}'\|T_k). \end{aligned}$$

Then, smart card transmits the message $M_3 = \{UID_i, R_3, T_k\}$ to the remote server S_j.

Step V6: Upon getting the message M_3, the S_j checks UID_i and T_k. If they are both valid, S_j checks $R_3'? = h(UID_i\|W_{ij}^*\|B_{ij}\|Z_{ij}\|T_k)$. If not, the server S_j terminates the communication. Otherwise, S_j generates the session key

$$SK_{ij}' = h(UID_i\|W_{ij}'\|B_{ij}\|Z_{ij}).$$

2.4 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: U_i inserts his smart card and inputs his identity UID_i and password PW_i.

Step P2: For each $(1 \leq j \leq k)$, the smart card computes $S_{ij}^{new} = S_{ij} \oplus h(UID_i||h(PW_i||b)) \oplus h(UID_i||h(PW_i^{new}||b))$.

Step P3: The smart card replaces $(S_{i1}, S_{i2}, \ldots, S_{ik})$ with $(S_{i1}^{new}, S_{i2}^{new}, \ldots, S_{ik}^{new})$.

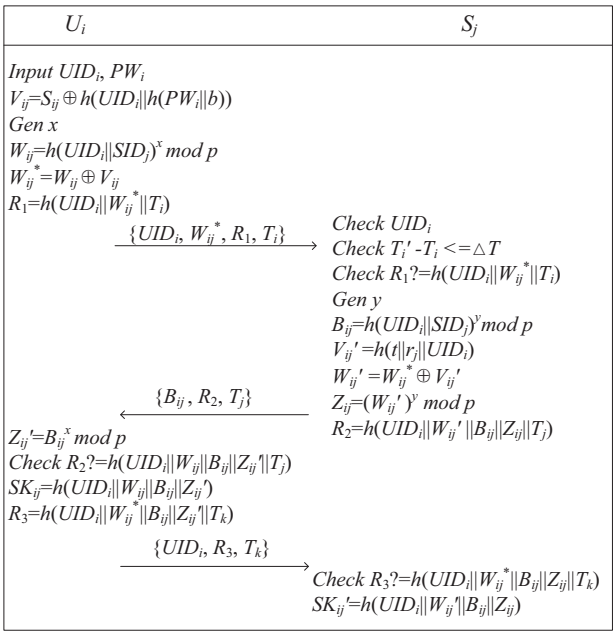

Figure 1: Login and authentication phase of Wei *et al.*'s scheme

3 Security Analysis of Wei *et al.*'s Scheme

In Wei *et al.*'s scheme, they proposed an improved smart card authentication scheme for multi-server architecture that can resist various well-known attacks, such as off-line password guessing attacks, impersonation attacks and privileged insider attacks. Unfortunately, we find that their scheme still has many vulnerabilities. an attacker can launch denial of service attack, because the user transmits data to remote server through the public channel. Secondly, an adversary can initiate impersonation attack once the stolen. Besides, there is no password checking after the user inputs his/her password, the wrong password cannot be found in time. Moreover, the user's behavior is easily to be traced. The detailed description is as follows.

3.1 Denial of Service Attack

From the login and authentication phase of Wei *et al.*'s scheme, we find that any attacker Z can easily forge a login request message that can pass S_j's authentication by eavesdropping a valid login request message and then launch DoS attack on the server.

An malicious attacker Z may eavesdrop the valid login request message $\{UID_i, W_{ij}^*, R_1, T_i\}$ that the user U_i transmitted to the server S_j and compute $R_1' = h(UID_i||W_{ij}^*||T_i')$, where T_i' is the current time. Then Z can forge the request message $\{UID_i, W_{ij}^*, R_1', T_i'\}$ that can pass S_j's verification.

After that, the server S_j select a random y, and com-

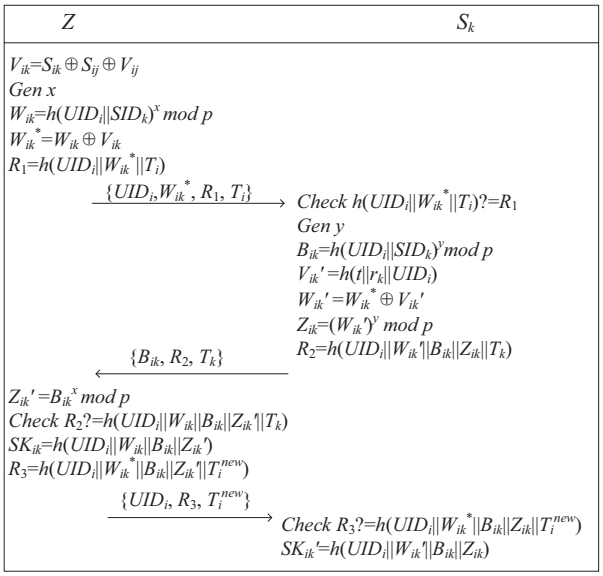

Figure 2: User impersonation attack on Wei *et al.*'s Scheme

putes

$$\begin{aligned} B_{ij} &= h(UID_i||SID_j)^y \bmod p, \\ V_{ij}' &= h(t||r_j||UID_i), \\ W_{ij}' &= W_{ij}^* \oplus V_{ij}', \\ Z_{ij} &= (W_{ij}')^y \bmod p, \\ R_2 &= h(UID_i||W_{ij}'||B_{ij}||Z_{ij}||T_j), \end{aligned}$$

where T_j is the current timestamp.

Then, S_j transmits message $M_2 = \{B_{ij}, R_2, T_j\}$ to the user U_i. The attacker Z will intercept the message to terminate the communication.

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

3.2 User Impersonation Attack

As shown in Wei *et al.*'s scheme, any registered server S_j can compute $V_{ij} = h(t||r_k||UID_i)$. Under the condition that the server S_j was captured by an attacker Z, Z can impersonate as U_i to log in to any registered server (*e.g.*, S_k) by stealing U_i's smart card without knowing UID_i and PW_i as show in Figure 2. The procedure is as follow:

- The attacker Z retrieves S_{ij} and S_{ik} from U_i's smart card, then computes $V_{ik} = S_{ik} \oplus S_{ij} \oplus V_{ij}$;
- Z generates a random number x, and computes

$$\begin{aligned} W_{ik} &= h(UID_i||SID_k)^x \bmod p, \\ W_{ik}^* &= W_{ik} \oplus V_{ik}, \\ R_1 &= h(UID_i||W_{ik}^*||T_i). \end{aligned}$$

  Then, Z forwards $M_1 = \{UID_i, W_{ik}^*, R_1, T_i\}$ to S_k;
- Upon receiving the login request message M_1, the remote server S_k checks the validity of T_i and compares $h(UID_i||W_{ik}^*||T_i)$ with R_1. Because they are equivalent, S_k will accept the login request;
- The server S_k generates a random number y to compute

$$\begin{aligned} B_{ik} &= h(UID_i||SID_k)^y \bmod p, \\ V_{ik}' &= h(t||r_k||UID_i), \\ W_{ik}' &= W_{ik}^* \oplus V_{ik}', \\ Z_{ik} &= (W_{ik}')^y \bmod p, \\ R_2 &= h(UID_i||W_{ik}'||B_{ik}||Z_{ik}||T_k). \end{aligned}$$

  Then, S_k transmits $M_2 = \{B_{ik}, R_2, T_k\}$ to U_i;
- Z intercepts M_2, and computes $Z_{ik}' = B_{ik}^x \bmod p$, checks whether $h(UID_i||W_{ik}||\ B_{ik}||Z_{ik}'||T_k)$ is equal to R_2. If it is holds, Z computes

$$\begin{aligned} SK_{ik} &= h(UID_i||W_{ik}||B_{ik}||Z_{ik}'), \\ R_3 &= h(UID_i||W_{ik}^*||B_{ik}||Z_{ik}'||T_i^{new}). \end{aligned}$$

  Finally, Z sends $M_3 = \{UID_i, R_3, T_i^{new}\}$ to S_k;
- After receiving M_3, S_k checks the validity of T_i^{new} and verifies whether $h(UID_i||W_{ik}^*\ ||B_{ik}||Z_{ik}||T_i^{new})$ is equal to R_3. If it holds, S_k generates the session key $SK_{ik}' = h(UID_i||W_{ik}'||B_{ik}||Z_{ik})$. Obviously, $SK_{ik}' = SK_{ik}$, a shared session key is established between the attacker Z and the remote server S_k.

At last, the attacker Z logs in to the server S_k by masquerading as U_i. Therefore, Wei *et al.*'s scheme cannot withstand user impersonation attack.

3.3 Unable to Check Password in Time

In the login and authentication phase of Wei *et al.*'s scheme, the device reader cannot check the identity UID_i and password PW_i of U_i in time, which may consume the computational and communication cost of remote server and smart card. The detailed description is as follows.

Once a legal user U_i attaches his/her smart card to a device reader, inputs his/her identity UID_i and an error password PW_i'. The smart card computes $V_{ij} = S_{ij} \oplus h(UID_i||h(PW_i'||b))$, and selects a random number x to computes

$$\begin{aligned} W_{ij} &= h(UID_i||SID_j)^x \bmod p, \\ W_{ij}^* &= W_{ij} \oplus V_{ij}, \\ R_1 &= h(UID_i||W_{ij}^*||T_i). \end{aligned}$$

Afterwards, the smart card transmits the message $M_1 = \{UID_i, W_{ij}^*, R_1, T_i\}$ to a remote server S_j.

After receiving the message M_1, S_j checks the validity of T_i and whether $h(UID_i||W_{ij}^*||T_i)$ is equal to R_1. Obviously, it holds. Then S_j chooses a random number y,

and computes

$$\begin{aligned} B_{ij} &= h(UID_i||SID_j)^y \, mod \, p, \\ V'_{ij} &= h(t||r_j||UID_i), \\ W'_{ij} &= W^*_{ij} \oplus V'_{ij}, \\ Z_{ij} &= (W'_{ij})^y \, mod \, p, \\ R_2 &= h(UID_i||W'_{ij}||B_{ij}||Z_{ij}||T_j). \end{aligned}$$

Eventually, the remote server S_j transfer the response message $M_2 = \{B_{ij}, R_2, T_j\}$ to U_i.

Upon getting the message M_2, the smart card computes $Z'_{ij} = B^x_{ij} \, mod \, p$, and check whether $R'_2 = h(UID_i||W_{ij}||B_{ij}||Z'_{ij}||T_j)$ is equal to R_2. Since U_i input an error password PW'_i, $V_{ij} = S_{ij} \oplus h(UID_i||h(PW'_i||b))$ will not equal to $V'_{ij} = h(t||r_j||\ UID_i)$, the smart card terminates the communication.

From the above discussion, we know that the error password was not be found in time, smart card and remote server have waste a large number of computational and communication resource.

3.4 No Provision of User Anonymity

With the wide application of network technology, the protection of user's privacy have received more and more attentions, user anonymity is a desirable property for remote user authentication. In Wei *et al.*'s protocol, the identity UID_i of user U_i is static, which will cause the user's login request be traced.

4 The Proposed Protocol

Based on the cryptanalysis of Wei *et al.*'s scheme, we present an enhanced biometric based authentication with key-agreement protocol for multi-server architecture based on elliptic curve cryptography. The proposed protocol consists of four phases: initialization phase, registration phase, login and authentication phase, and password change phase. There are also three participants: the user U_i, remote server S_j and registration center RC.

4.1 Initialization Phase

Registration server RC generates following parameters in order to initialize the system.

Step I1: The registration center RC chooses an elliptic curve equation E with an order n.

Step I2: The registration center RC selects a base point Q over E and chooses a one-way cryptographic hash function $h(\cdot)$.

Step I3: The registration center RC publishes the information $\{E, Q, h(\cdot)\}$.

4.2 Registration Phase

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

4.2.1 Server Registration Phase

In this phase, the remote server S_j sends a registration request to the registration center RC in order to become an authorized server. The registration process according to the following steps:

Step S1: The remote server S_j computes public key $P_b = P_r \cdot Q$ and sends registration request $\{P_b, SID_j\}$to RC.

Step S2: The registration center RC sends PSK to the remote server S_j, which can be used in further phases of authentication.

4.2.2 User Registration Phase

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

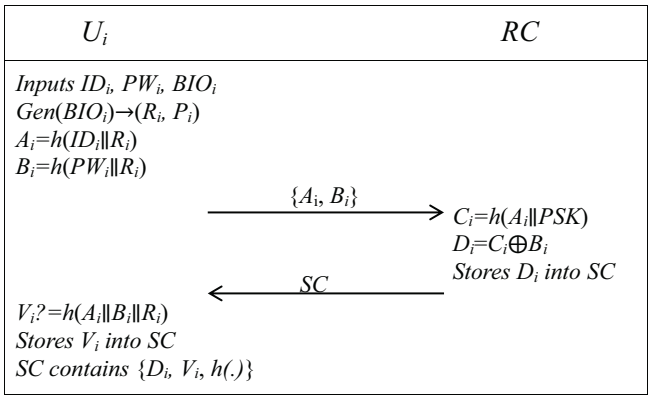

Figure 3: User registration phase of the proposed protocol

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. Next the user U_i computes $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$. Finally, the user U_i sends a request message $\{A_i, B_i\}$ to RC via a secure channel.

Step U2: Upon receiving the request message, RC computes $C_i = h(A_i||PSK), D_i = B_i \oplus C_i$.

Step U3: RC stores the parameters $\{D_i,\ h(\cdot)\}$ into a new smart card and delivers it to the user U_i via a secure channel.

Step U4: Upon getting the message, the user U_i computes $V_i = h(A_i||B_i||R_i)$ and stores $\{V_i\}$ into smart card. Thus the smart card finally contains the parameters $\{D_i, V_i, h(.)\}$.

4.3 Login and Authentication Phase

When a legal user U_i wants to login into some remote server S_j, he/she first attaches the smart card to a device reader, and inputs 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 string R_i from $Rep(BIO_i, P_i) \rightarrow R_i$. Then, the concrete login and authentication procedure, as shown in Figure 4, the login and authentication mechanism is performed as follows:

Step V1: The smart card SC computes $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$, and then verifies whether V_i is equal to $h(A_i||B_i||R_i)$. If V_i is invalid, SC terminates the communication; otherwise, the smart card SC generates a random number x and calculates $K = x \cdot Q$, $K' = x \cdot P_b$, $AID_i = A_i \oplus K'$, $C_i = D_i \oplus B_i$ and $M_1 = h(AID_i||C_i||K||K'||T_i)$. Then the smart card SC sends the login request message $\{M_1, K, AID_i, T_i\}$ to the remote server S_j.

Step V2: Upon receiving the message from the user U_i, the remote server S_j checks whether T_i' - T_i is less than $\triangle T$. The remote server computes $K' = P_r \cdot K$, $A_i = AID_i \oplus K'$, $C_i = h(A_i||PSK)$ and verifies whether M_1 is equal to $h(AID_i||C_i||K||K'||T_i)$. If the condition holds, the remote server S_j authenticates the user U_i, otherwise the process can be terminated.

Step V3: The remote server S_j further generates a random number N_1 and computes $M_2 = A_i \oplus N_1, M_3 = h(A_i||K'||SID_j||N_1)$ and $SK_{ij} = h(A_i||K'||C_i||SID_j||N_1)$. Furthermore, the remote server S_j sends the response message $\{M_2, M_3\}$to the user U_i.

Step V4: After getting the message M_2 and M_3, the user U_i computes $N_1 = M_2 \oplus A_i$ and verifies whether M_3 is equal to $h(A_i||K'||SID_j||N_1)$. If the condition holds, the user U_i authenticates the remote server S_j, otherwise the process can be terminated. Then, the user computes $SK_{ij} = h(A_i||K'||C_i||SID_j||N_1)$, $M_4 = h(SK_{ij}||K'||N_1)$ and sends the message $\{M_4\}$ to the remote server S_j.

Step V5: Upon receiving the message, the remote server S_j verifies whether M_4 is equal to $h(SK_{ij}||K'||N_1)$ and reconfirms the authenticity of U_i. Now, the user U_i and the server S_j can start communication with the computed session key SK_{ij}.

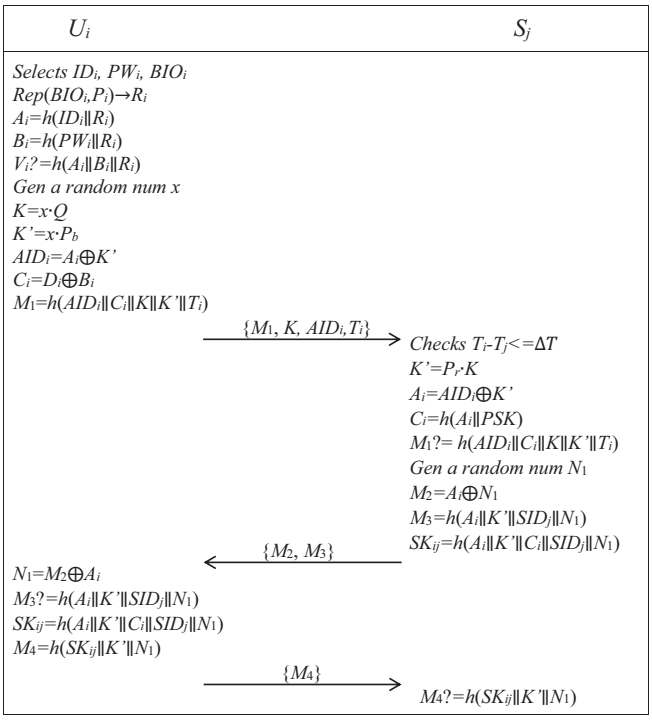

Figure 4: User registration phase of the proposed protocol

4.4 Password Changing Phase

This procedure invokes when a user (U_i) wish to update his/her existing password with new one. In this procedure, the user U_i can change his/her password as follows:

Step P1: The user U_i inserts smart card SC and inputs ID_i, PW_i and BIO_i.

Step P2: The smart card SC computes $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$, and then verifies the condition whether V_i is equal to $h(A_i||B_i||R_i)$. If this verification is valid, the smart card SC asks the user U_i for a new password. Otherwise, password change phase is terminated immediately by the smart card SC.

Step P3: The user U_i chooses a new password PW_i^{new} and then computes $A_i^{new} = h(ID_i||R_i)$, $B_i^{new} = h(PW_i||R_i)$, $C_i^{new} = h(A_i||PSK)$, $D_i^{new} = B_i \oplus C_i$, $V_i^{new} = h(A_i||B_i||R_i)$.

Step P4: In the memory, smart card SC respectively replaces D_i with D_i^{new} and V_i with V_i^{new}.

5 Analysis of the Proposed Protocol

In a multi-server architecture, there are three requirements for an authentication and key agreement protocol, namely, security, functionality and efficiency. In this section, we first present security analysis of our proposal, and then examine its performance in terms of functionality and efficiency by comparing it with previous related works.

5.1 User Anonymity

In our protocol, the real identity of user is not revealed throughout all the phases of communication. In the user registration phase, U_i submits $A_i = h(ID_i||R_i)$ and the real identity is protected with a one-way hash function. During the login phase, the parameter A_i is converted as anonymous in the form of $A_i = AID_i \oplus K'$. The identity is dynamic for every login session, due to its association with a random number x, where $K = x \cdot Q$ and $K' = x \cdot P_b$. An adversary cannot retrieve the x in anyway. Moreover, it is believed to be impossible to compute K' from K and P_b because of *ECDLP*. In the other hand, our protocol achieves the user untraceability. In the user login phase, the user U_i sends the message $\{M_1, K, AID_i, T_i\}$ to the remote server S_j. All the parameters are dynamic and dose not disclose the identity of U_i. Hence, our protocol achieves user anonymity and untraceability.

5.2 Resistance to Denial-of-Service Attack

The Denial-of-Service attack diminishes or eliminates the server's expected capability to make the server unavailable. With the help of timestamp T_i, the remote server S_j checks the freshness and legality of $M_1 = h(AID_i||C_i||K||K'||T_i)$ in the login request message. The current timestamp does not match the previous M_1 which is sent by adversary. Moreover, our scheme applies the fuzzy extractor to satisfy the usage requirements of biometrics. As a result, our scheme is secure against the Denial-of-Service attack.

5.3 Resistance to User Impersonation Attack

Under the user impersonation attack, an adversary who is an outsider hackerS tries to impersonate user U_i without the password PW_i or biometric information BIO_i. If an adversary wants to masquerade a legitimate user U_i, he/she requires to build a login message $\{M_1, K, AID_i, T_i\}$, where $M_1 = h(AID_i||C_i||K||K'||T_i)$, $K = x \cdot Q$, $AID_i = Ai \oplus K'$. Conversely, the adversary can barely compute $K' = x' \cdot Q$ and $K'' = x' \cdot P_b$ by choosing his/her own random number x'. But the adversary can't compute rest of the two parameters, due to the unavailability of valid ID_i, PW_i and R_i. Hence, our protocol is secure against the user impersonation.

5.4 Resistance to Server Impersonation Attack

Our protocol protects the server impersonation attack and it's description is given below:

- In order to act as a legitimate server, the adversary eavesdrops the valid login request message $\{M_1, K, AID_i, T_i\}$ that the user U_i transmitted to the server S_j, and generates a random number P'_r and N'_1. Then S_j computes $K'' = P'_r \cdot K$, $A'_i = AID_i \oplus K''$, $C'_i = h(A'_i||PSK)$, $M'_2 = A'_i \oplus N'_1$, $M'_3 = h(A'_i||K''||SID_j||N'_1)$, $SK'_{ij} = h(A'_i||K''||C'_i||SID_j||N'_1)$. The adversary sends $\{M'_2, M'_3\}$ to U_i.
- Upon receiving the $\{M'_2, M'_3\}$, the user U_i computes $N'_1 = A_i \oplus M_2$ and $M_3 = h(A_i||K'||SID_j||N'_1)$. Here, U_i identities it as a fake response from the malicious server because of M_3 is not equal to M'_3 and terminates the session. Hence, our protocol can resist the server impersonation attack.

5.5 Resistance to Smart Card Stolen Attack

The adversary can extract the information $\{D_i, V_i, h(\cdot)\}$ stored in the smart card by means of power analysis. Assume a legal user's smart card is stolen by an adversary and extracted the information $\{D_i, V_i, h(\cdot)\}$. Then, the adversary may try to get ID_i, PWi and R_i from the extracted information. However, adversary cannot obtain any valuable information from these values, where $D_i = B_i \oplus C_i$, $C_i = h(A_i||PSK)$, $B_i = h(PW_i||R_i)$, $A_i = h(ID_i||R_i)$ and $V_i = h(A_i||B_i||R_i)$ since all the important parameters such as ID_i, PW_i and R_i are protected by a one-way hash function. The adversary cannot obtain any login information using the smart card stored parameters D_i and V_i. At the same time, guessing the real identity ID_i, password PW_i and biometric R_i is impractical. Therefore, our protocol is secure against smart card stolen attack.

5.6 Resistance to Replay Attack

If an adversary intercepts the communication message $\{M_1, K, AID_i, T_i\}$ between U_i and S_j, he tries to replay them to S_j to masquerade as a legal user. However, once the message is replayed, the server S_j can immediately detect the attack and reject the request due to the apply of timestamp T_i. Hence, our protocol is secure against replay attack.

5.7 Resistance to Privileged Insider Attack

During our protocol, U_i does not send his ID_i, password PW_i or his biometrics BIO_i in user registration phase. U_i submits only $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$ to RC instead of original credentials. Hence, an insider cannot obtain the original sensitive information of any user. On the other hand, the $M_1 = h(AID_i||C_i||K||K'||T_i)$ is invalid in which P_r is unobtainable. Therefore, our protocol resists to privileged insider attack.

5.8 Resistance to Password Guessing Attack

An adversary may try to guess the password PW_i from the extracted smart card stored parameters $\{D_i, V_i, h(\cdot)\}$.

The stored parameter contains the password PW_i in the form $B_i = h(PW_i||R_i)$ where $Gen(BIO_i) \rightarrow (R_i, P_i)$. An adversary attempts to verify the condition $V_i? = h(A_i||B_i||R_i)$ while constantly guessing PW_i. Adversary 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 an adversary cannot get the value of ID_i. As a result, the adversary cannot guess the correct password PW_i. Therefore, our protocol resist to password guessing attack.

5.9 Forward Secrecy

Perfect forward secrecy protects the session keys even if long-term key is retrieved. Specifically, the session key in the proposed scheme is generated as $SK_{ij} = h(A_i||K'||C_i||SID_j||N_1)$ and the long term private key of the server PSK in $C_i = h(A_i||PSK)$ is shielded with a hash function and is not possible to derive due to its one-way property. Although the long term key is compromised with an adversary, he/she still cannot compute a valid session key, the parameter $K' = P_r \cdot K$ and $K = x \cdot Q$ is dynamic due to its association with random generated number x, which is not possible to extract due to the reason of *ECDLP*. Therefore, our protocol provides perfect forward secrecy.

5.10 Performance and Functionality Comparisons

In this section, we compare our proposed protocol with several related schemes [8, 22, 28, 30]. In Table 2, we provide the comparison based on the key security of these schemes, while we compare their efficiency in terms of computation and communication cost . The computation cost of the protocol is the times of executing operations. The following notations are used in Table 2.

- T_e: modular exponentiation operation;
- T_m: modular multiplication/inverse operation;
- T_h: hash operation;
- T_{epm}: the time for executing a scalar multiplication operation of elliptic curve.

We also define i as the length of one parameter in the transmitted messages, such as the length of R is i and the length of $\{R, CID_i\}$ is $2i$. As Amin and Islam [1] executed various cryptographic operations using MIRACL C/C++ Library, the computation cost for T_h is approximately 0.0004ms, T_e is approximately 1.8269ms and T_m is approximately 0.0147ms. As per Kilinc and Yanik [14] experiment on a personal computer involving a processor with Dual CPU E2200 2.20 GHz along with RAM size of 2048MB, the computation cost for T_{epm} is approximately 2.229ms.

In Table 2, we summarize the efficiency comparison according to the computation cost and communication cost between our protocol and other schemes [8, 22, 28, 30] in case that the login and authentication phase is done. From Table 4, it is easy to see that our scheme is more efficient than Wei *et al.*'s scheme [30], He *et al.*'s scheme [8] and Wang *et al.*'s scheme [28]. Moreover, our proposed protocol is lower computation cost than those of Wei *et al.*'s scheme [30].

From Table 3, it can be observed that the proposed protocol is more secure than the other four schemes. Our new protocol satisfies all the security requirements listed in Table 3. Wei *et al.*'s scheme [30] only satisfy five of the nine requirements, respectively. Pippal *et al.*'s scheme [22], He *et al.*'s scheme [8] and Wang *et al.*'s scheme [28] only satisfies three of the nine requirements. Hence, our scheme achieves stronger security than their solutions.

6 Conclusions

In this paper, we analyzed Wei *et al.*'s smart card based multi-server authentication scheme. Our analysis reveals its inherent security vulnerabilities, *i.e.*, denial of service attack, impersonation attack, unable to check password in time and no provision of user anonymity. In addition, this paper proposed an enhanced biometric based authentication with key agreement protocol for multi-server architecture based on elliptic curve cryptography. The mutual authentication of the proposed protocol achieved significant features such as biometric authentication, elliptic curve cryptography, 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.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (No.61962022), Key Research and Development Plan of Jiangxi Province (No.20192BBE50077), the project of Education Department of Jiangxi Province (No. GJJ160510).

References

[1] R. Amin, S. H. Islam, and M. K. Khan, "A two-factor rsa-based robust authentication system for multiserver environments," *Security & Communication Networks*, vol. 2017, no. 13, pp. 1–15, 2017.

[2] R. Amin, S. K. Islam, and G. P. Biswas, "An efficient and practical smart card based anonymity preserving user authentication scheme for tmis using elliptic curve cryptography," *Journal of Medical Systems*, vol. 39, no. 11, p. 180, 2015.

[3] T. Cao and S. Huang, "Cryptanalysis of a sensor smart card based password authentication scheme

Table 2: Efficiency comparison

	Wang *et al.*[19]	He *et al.*[5]	Pippal *et al.*[11]	Wei *et al.*[21]	Ours
User	$6T_h+3T_e$	$7T_h+2T_{epm}$	$4T_h+3T_e+T_m$	$7T_h+2T_e$	$7T_h+2T_{epm}$
Server	$6T_h+2T_e$	$6T_h+T_{epm}$	$3T_h+4T_e+T_m$	$6T_h+2T_e$	$4T_h+T_{epm}$
RC	$3T_h$	$3T_h$	$3T_h$	$3T_h$	T_h
Total	9.1405ms	6.6934ms	3.6872ms	7.3140ms	6.6922ms
Communication cost	6i	6i	5i	10i	7i

Table 3: Security comparison

	Wang *et al.*[19]	He *et al.*[5]	Pippal *et al.*[11]	Wei *et al.*[21]	Ours
User anonymity	No	No	No	No	Yes
Denial-of-Service attack	No	No	No	No	Yes
User impersonation attack	No	Yes	Yes	No	Yes
Server impersonation attack	Yes	No	Yes	Yes	Yes
Smart card stolen attack	No	No	No	No	Yes
Replay attack	Yes	Yes	Yes	Yes	Yes
Privileged insider attack	No	No	No	Yes	Yes
Password guessing attack	Yes	No	No	Yes	Yes
Forward secrcy	Yes	Yes	No	Yes	Yes

with user anonymity," *Sensor Letters*, vol. 11, no. 11, pp. 2149–2151(3), 2013.

[4] C. C. Chang, T. F. Cheng, and W. Y. Hsueh, "A robust and efficient dynamic identity-based multi-server authentication scheme using smart cards," *International Journal of Communication Systems*, vol. 29, no. 2, pp. 290–306, 2016.

[5] 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.

[6] 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.

[7] 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.

[8] D. B. He and S. H. 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] H. C. Hsiang and W. K. Shih, "Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment," *Computer Standards & Interfaces*, vol. 31, no. 6, pp. 1118–1123, 2009.

[10] 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.

[11] M. S. Hwang, J. W. Lo, C. Y. Liu, S. C. Lin, "Cryptanalysis of a user friendly remote authentication scheme with smart card," *Pakistan Journal of Applied Sciences*, vol. 5, no. 1, pp. 99–100, 2005.

[12] M. S. Hwang, T. H. Sun, "Using smart card to achieve a single sign-on for multiple cloud services," *IETE Technical Review*, vol. 30, no. 5, pp. 410–416, 2013.

[13] Q. Jiang, J. F. Ma, G. S. Li, and X. H. Li, "Improvement of robust smart-card-based password authentication scheme," *International Journal of Communication Systems*, vol. 28, no. 2, pp. 383–393, 2015.

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

[15] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks,"

International Journal of Network Security, vol. 13, no. 3, pp. 167–177, 2011.

[16] C. C. Lee, T. H. Lin, and R. X. Chang, "A secure dynamic id based remote user authentication scheme for multi-server environment using smart cards," *Expert Systems with Applicationsn*, vol. 38, no. 11, pp. 13863–13870, 2011.

[17] X. Li, J. Ma, W. D. Wang, and J. S. Zhang, "A novel smart card and dynamic id based remote user authentication scheme for multi-server environments," *Mathematical & Computer Modelling*, vol. 58, no. 1-2, pp. 85–95, 2013.

[18] C. T. Li, M. S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards", *Journal of Network and Computer Applications*, vol. 33, no. 1, pp. 1-5, 2010.

[19] C. T. Li, M. S. Hwang, "An online biometrics-based secret sharing scheme for multiparty cryptosystem using smart cards", *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 5, pp. 2181–2188, 2010.

[20] C. H. Ling, W. Y. Chao, S. M. Chen, M. S. Hwang, "Cryptanalysis of dynamic identity based on a remote user authentication scheme for a multi-server environment," in *International Conference on Advances in Mechanical Engineering and Industrial Informatics (AMEII'15)* , pp. 981–986, 2015.

[21] H. T. Pan, C. S. Pan, S. C. Tsaur, M. S. Hwang, "Cryptanalysis of efficient dynamic ID based remote user authentication scheme in multi-server environment using smart card," in *12th International Conference on Computational Intelligence and Security*, pp. 590–593, 2017.

[22] R. S. Pippal, C. D. Jaidhar, and S. Tapaswi, "Robust smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, vol. 72, no. 1, pp. 729–745, 2013.

[23] R. Ramasamy and A. P. Muniyandi, "An efficient password authentication scheme for smart card," *International Journal of Network Security*, vol. 14, no. 3, pp. 180–186, 2012.

[24] T. Maitra, S. H. Islam, and R. Amin, "An enhanced multi-server authentication protocol using password and smart-card: cryptanalysis and design," *Security & Communication Networks*, vol. 3, no. 17, pp. 4615–4638, 2016.

[25] J. L. Tsai, "Efficient multi-server authentication scheme based on one-way hash function without verification table," *Computers & Security*, vol. 27, no. 3-4, pp. 115–121, 2008.

[26] T. Wan, N. Jiang, and J. F. Ma, "Cryptanalysis of two dynamic identity based authentication schemes for multi-server architecturet," *China Communications*, vol. 11, no. 11, pp. 125–134, 2014.

[27] T. Wan, N. Jiang, and J. F. Ma, "Cryptanalysis of a biometric-based multi-server authentication scheme," *International Journal of Security and its Application*, vol. 10, no. 2, pp. 163–170, 2016.

[28] B. Wang and M. D. Ma, "A smart card based efficient and secured multi-server authentication scheme," *Wireless Personal Communications*, vol. 68, no. 2, pp. 361–378, 2013.

[29] R. C. Wang, W. S. Juang, and C. L. Lei, "User authentication scheme with privacy-preservation for multi-server environment," *IEEE Communications Letter*, vol. 13, no. 2, pp. 157–159, 2009.

[30] J. H. Wei, W. F. Liu, and X. X. Hu, "Cryptanalysis and improvement of a robust smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, vol. 77, no. 3, pp. 2255–2269, 2014.

[31] 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.

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.