Improved ECC-Based Three-Factor Multiserver Authentication Scheme

1. Introduction

With the development of information technologies [1–8] and the widespread application of the Internet of Things [9–12], mobile communication has emerged in many network communication environments. The multiserver environments in mobile communication improve the efficiency of user communications; therefore, it is more popular than single-server environments for users. The multiserver environment overcomes the limited storage and computing of the single-server environment and can provide more remote services. A typical multiserver environment is shown in Figure 1.

Owing to the convenience of multiserver environments, authentication problems in the communication process cannot be disregarded. To date, three methods can be used to achieve user authentication in the environment. The first is password-based authentication [13–17]. This is the simplest method to perform authentication; however, an attacker can easily guess or steal a password from a party and impersonate as a valid user. The second is two-factor authentication, which is based on a password and a smart card [18–24]. Compared with password-based authentication, two-factor authentication improves security. However, if the smart card is stolen, then the information stored in the smart card may be recovered. This will result in well-known attacks, such as offline guessing attacks. In the past few years, Wang et al. have proposed some two-factor authentication schemes in different application scenarios. In 2014, they proposed an anonymous two-factor authentication scheme in a distributed system [19]. In the same year, they proposed an anonymous two-factor authentication scheme in a wireless sensor network [20]. In 2016, Wang et al. [25] compared and evaluated some representative two-factor authentication schemes and proposed a new evaluation standard for two-factor authentication schemes. In 2018, Wang et al. [26] proposed an evaluation framework for a two-factor authentication scheme for real-time data access in industrial wireless sensor networks and evaluated the relevant schemes. The third is three-factor authentication,
which is based on passwords, smart cards, and biometrics [27–39]. In a public channel, an attacker may eavesdrop, modify, or replay transmitted messages. This poses a significant threat to the security of users. Because only the password- or smart card-based authentication scheme exhibits low security, applying biometrics to authentication schemes can overcome the insecurity of password- or smart card-based schemes. Therefore, a secure and efficient authentication scheme based on biometrics must be designed.

Compared with Rivest–Shamir-Adleman (RSA) or ElGaml cryptosystems, elliptic curve cryptography (ECC) provides a small key size and computation efficiency under the same security level. In recent years, several biometric-based authentication schemes based on ECC have been proposed. In 2013, Pippal et al. [27] proposed a three-factor authentication scheme in a multiserver environment and claimed that their scheme can overcome all types of network attacks. In 2014, He and Wang [28] proposed a multiserver environment authentication scheme based on robust biometrics, claiming that their scheme was the first three-factor authentication scheme applicable to multiserver environments. In 2015, Odelu et al. [30] reported that the scheme proposed in [28] was vulnerable to a known session-specific temporary information attack and an impersonation attack and hence did not provide strong user anonymity; therefore, they proposed a secure multiserver authentication protocol based on biometric technology using smart cards. In the same year, Li et al. [31] discovered that Pippal et al.’s [27] scheme can provide incorrect authentication but could not overcome impersonation, stolen smart card, and internal attacks. Therefore, Li et al. [31] proposed an improved scheme to overcome the problems above. In 2017, Kumari et al. [32] proposed a provable secure multicloud server authentication scheme based on biometrics. However, in 2018, Feng et al. [33] discovered that the scheme presented in [32] could not guarantee user anonymity, three-factor security, perfect forward security, etc.; hence, they proposed a multiserver environment authentication scheme based on anonymous biometrics. In the same year, Ali and Pal [34] analyzed Li et al.’s [31] scheme and discovered that it could not overcome password-guessing, user impersonation, insider, and smart card theft attacks nor could they guarantee user anonymity.

Ali and Pal [34] proposed a three-factor multiserver authentication scheme based on an elliptic curve cryptosystem to solve the abovementioned issues. Unfortunately, Wang et al. [36] discovered that the scheme presented in [34] was vulnerable to user impersonation, server impersonation, privileged insider, and denial-of-service attacks, among others, and could not provide both forward and three-factor confidentiality. Therefore, Wang et al. proposed an improved multiserver authentication scheme based on biometrics and claimed that their scheme can overcome offline password-guessing, user impersonation, server impersonation, known specific session temporary information, three-factor security, user anonymity, and privileged internal attacks. Some important related works are summarized in Table 1.

In this study, we investigated Wang et al.’s scheme subject to known session-specific temporary information, user impersonation, and server impersonation attacks. To overcome the abovementioned attacks, we refer to Wang et al.’s scheme and propose an improved authentication scheme. Finally, we demonstrate that our scheme is semantically secure in the ROR model and overcome known attacks using the ProVerif tool and the BAN logic.

The remainder of this paper is organized as follows. A simple review and cryptanalysis of the scheme proposed by Wang et al. is discussed in Sections 2 and 3, respectively. Section 4 elaborates the proposed scheme in detail. Section 5 demonstrates the security analysis of the proposed scheme. Section 6 presents a comparison of performance and security. Section 7 summarizes the paper.

2. Review of Wang et al.’s Scheme

Wang et al.’s scheme includes initialization, server and user registration, and login authentication phases. Their scheme involves three types of entities: users, servers, and a registration center. The notations used in the scheme and their descriptions are shown in Table 2.

2.1. Initialization. In this phase, the registration center (RC) selects an elliptic curve $E_q$, and the basic point $P$ of $E_q$ defines two hash functions $H(\cdot)$ and $h(\cdot)$. Subsequently, the
Secure channel. The RC receives this message, computes
Subsequently, User selects a random number \( r_i \) and computes
\( P_i = H(\text{SID}_i \| h(b_i) || r_i) \), and sends \( \{\text{ID}_i, P_i\} \) to the RC. The RC receives this message and calculates the following:

\[
A_i = H(x \| \text{ID}_i),
B_i = A_i \bigoplus P_i, \tag{1}
V_i = H((P_i \bigoplus H(\text{ID}_i)) \mod n),
\]

where \( 2^4 \leq n \leq 2^8 \). Note that \( H(P_i \bigoplus H(\text{ID}_i)) \) is the technique of fuzzy-verifier [40]. The RC stores \( \{B_i, V_i, E_{\text{key}}(\cdot), P_i, P_{\text{pub}}, n\} \) in the smart card (SC) and then sends the SC to User, in a secure channel. Subsequently, User, stores \( r_i \) in the SC.

2.3. Login and Authentication. In this phase, User and Server complete a mutual authentication and establish a session key (SK) with the aid of the RC.

Step 1 User enters \( \text{ID}_i \) and \( \text{PW}_i \), imprints \( b_i \), and logs in the SC. Subsequently, the SC computes
\[
P_i' = H(\text{PW}_i \| h(b_i) \| r_i),
V_i' = H((P_i' \bigoplus H(\text{ID}_i)) \mod n), \tag{2}
\]

and verifies if \( V_i = V_i' \). If they are equal, then User generates a random number \( N_1 \) and computes
\[
A_i' = B_i \bigoplus P_i',
R_i = N_1 P_i,
C_i = H(N_1 P_{\text{pub}}), \tag{3}
L_i = E_{C_i}(\text{ID}_i \| A_i' \| \text{SID}_i).
\]

Next, User sends \( \{R_i, L_i\} \) to the RC in the public channel.

### Table 1: The summary of authentication schemes.

| Scheme | Cryptographic techniques | Limitations |
|--------|--------------------------|-------------|
| Pippal et al. [27] | (1) Utilized one-way hash function | (1) Does not resist impersonation attacks |
| | (2) Based on Diffie–Hellman problem | (2) Does not resist internal attacks |
| | (3) Based on smart card | |
| Li et al. [31] | (1) Utilized one-way hash function | (1) Does not resist password-guessing attacks |
| | (2) Based on Diffie–Hellman problem | (2) Does not resist impersonation attacks |
| | (3) Based on smart card | (3) Does not resist internal attacks |
| | (4) Does not resist smart card theft attacks | (4) Does not provide forward secrecy |
| | (5) Does not support user anonymity | |
| Kumari et al. [32] | (1) Based on biometrics | (1) Does not support user anonymity |
| | (2) Utilized one-way hash function | (2) Does not resist man-in-the-middle attacks |
| | (3) Based on anonymous authentication | |
| Feng et al. [33] | (1) Utilized ECC | (1) Does not provide three-factor secrecy |
| | (2) Based on smart card | (2) Does not resist known session-specific temporary information attack |
| | (3) Based on biometrics | |
| | (4) Does not resist three-factor secrecy | |
| | (5) Does not support user anonymity | |
| Ali and Pal [34] | (1) Utilized ECC | (1) Does not resist impersonation attacks |
| | (2) Three-factor security | (2) Does not resist internal attacks |
| | (3) Based on data encryption scheme | (3) Does not provide forward secrecy |
| | (4) Does not provide three-factor secrecy | (4) Does not resist known session-specific temporary information attack |
| | (5) Does not resist known session-specific temporary information attack | |
| Wang et al. [36] | (1) Utilized ECC | (1) Does not resist impersonation attacks |
| | (2) Based on biometrics | (2) Does not resist known session-specific temporary information attack |
| | (3) Based on data encryption scheme | |

### Table 2: Notations and descriptions.

| Notations | Descriptions |
|-----------|--------------|
| RC | The registration center |
| Server | The j-th server |
| User | The i-th user |
| A | The attacker |
| \( P_{\text{pub}} \) | The public key of RC |
| \( H(\cdot) \) | Hash function |
| \( h(\cdot) \) | Biohash function |
| \( \text{ID}_i \) | User identity |
| \( \text{PW}_i \) | User’s password |
| \( b_i \) | User’s biometrics |
| SC | Smart card |
| SK | Session key |
| \( \oplus \) | Concatenation |
| \( \oplus \) | Bitwise XOR |
| \( E_{\text{key}}(\cdot)/D_{\text{key}}(\cdot) \) | Symmetric encryption/decryption algorithm with key |

RC selects a random number \( x \) and computes the public key \( P_{\text{pub}} = xP \), where \( x \) is the RC’s secret key and publishes \( \{E_{\text{pub}}, P, P_{\text{pub}}, H(\cdot), h(\cdot)\} \).
Step 2 After the RC receives \( \{R_i, L_i\} \), it computes
\[
C_i = H(xR_i),  \\
\left( ID_i \| A_i' \| SID_j \right) = D_{C_i}(L_i),  \\
A_i = H(x \| ID_i),
\]
and verifies if \( A_i' = A_i \). If they are equal, then the RC computes
\[
SM_j = H(SID_j \| x),  \\
Y_i = H(SID_j \| SM_j),  \\
M_i = E_{SM_j}(ID_i \| R_i \| Y_i \| H(A_i \| C_i')).
\]

Next, the RC sends \( \{M_i\} \) to Server, in the public channel.

Step 3 After Server, receives \( \{M_i\} \), it computes
\[
\left( ID_i \| R_i \| Y_i \| H(A_i \| C_i') \right) = D_{SM_j}(M_i),  \\
Y_i' = H(SID_j \| SM_j),
\]
and verifies if \( Y_i' = Y_i \). If they are equal, then Server, generates a random number \( N_2 \) and computes
\[
R_s = N_2P,  \\
E_i = N_2R_i,  \\
SK_j = H(E_i \| Ht(A_i \| C_i')),  \\
F_i = H(ID_i \| SK_j \| tR_s \| nSID_j).
\]

Subsequently, Server, sends \( \{R_s, F_i\} \) to User, in the public channel.

Step 4 After User, receives \( \{R_s, F_i\} \), he computes
\[
E_i' = N_1R_s,  \\
SK_i = H(E_i' \| Ht(A_i \| C_i')),  \\
F_i' = H(ID_i \| SK_i \| tR_s \| nSID_j),
\]
and verifies if \( F_i' = F_i \). If they are equal, then User, computes
\[
Q_i = H(SK_i \| R_s).
\]

Next, User, sends \( \{Q_i\} \) to Server, in the public channel.

Step 5 After Server, receives \( \{Q_i\} \), it computes
\[
Q_i = H(SK_i \| R_s),
\]
and verifies if \( Q_i' = Q_i \). If they are equal, then \( SK_i = SK_j \) is the session key for User, and Server,.

3. Cryptanalysis of Wang et al.’s Scheme

In this section, we demonstrate Wang et al.’s scheme subject to three security attacks. In our proposed attacks, we assumed that the attacker \( A \) is a legitimate user and has already registered with the RC.

3.1. Known Session-Specific Temporary Information Attack.

A known session-specific temporary information attack refers to a security attack in which an attacker attempts to obtain the current SK when temporary secret values such as random numbers are disclosed [41].

In this attack, we assume that the attacker \( A \) obtains temporary information \( N_i \) and captures \( \{R_i, L_i\} \) and \( \{R_s, F_i\} \), which are transmitted over the public channel. Based on the above, \( A \) can compute
\[
C_i = H(N_1P_{pub}),  \\
\left( ID_i \| A_i' \| SID_j \right) = D_{C_i}(L_i),  \\
E_i' = N_1R_s.
\]

Subsequently, \( A \) obtains \( C_i, A_i' \), and \( E_i' \); hence, it can determine SK = \( H(E_i' \| Ht(A_i' \| C_i')) \). Furthermore, based on the formulas above, \( A \) can obtain the user’s ID; in other words, the user’s anonymity is not protected.

3.2. User Impersonation Attack

Step 1 Based on Section 3.1, \( A \) can obtain \( ID_i, A_i', \) and \( SID_j \). Subsequently, \( A \) generates a random number \( N_A \) and computes
\[
R_A = N_AP,  \\
C_A = H(N_AP_{pub}),  \\
L_A = E_{C_A}(ID_i \| A_i' \| SID_j).
\]

\( A \) fakes User, to send \( \{R_A, L_A\} \) to the RC.

Step 2 Upon receiving \( \{R_A, L_A\} \), the RC computes
\[
C_A' = H(xR_A),  \\
\left( ID_i \| A_i' \| SID_j \right) = D_{C_A'}(L_A),  \\
A_i = H(x \| ID_i).
\]

It is clear that \( A_i' = A_i \). Next, the RC computes
\[
SM_j = H(SID_j \| x),  \\
Y_i = H(SID_j \| SM_j),  \\
M_A = E_{SM_j}(ID_i \| R_A \| Y_i \| H(A_i \| C_A')),  \\
\]
and sends \( \{M_A\} \) to Server,.

Step 3 After receiving \( \{M_A\} \), Server, computes
\[
\left( ID_i \| R_A \| Y_i \| H(A_i \| C_A') \right) = D_{SM_j}(M_A),  \\
Y_i' = H(SID_j \| SM_j).
\]
Security and Communication Networks

3.3. Server Impersonation Attack. This attack is also based on $C_j$, $ID_j$, $A_i'$, and $SID_j$ in Section 3.1. When User$_i$ sends $\{R_i, L_i\}$ to the RC, $A$ eavesdrops the message. Subsequently, when RC sends $\{M_j\}$ to Server$_j$, A intercepts the message. A generates a random number $N_A$ and computes

$$R_A = N_A P,$$
$$E_A = N_A R_A,$$
$$SK_{A_i} = H(E_A \| Ht(A_i' \| C_i')),$$
$$F_A = H(ID_j \| SK_{A_i} \| tR_A \| nSID_j),$$

and sends $\{R_A, F_A\}$ to User$_i$.

Upon receiving $\{R_A, F_A\}$, User$_i$ computes

$$E_A' = N_i R_A,$$
$$SK_i = H(E_A' \| Ht(A_i' \| C_i')),$$
$$F_A' = H(ID_i \| SK_i \| tR_A \| qSID_j),$$

and sends $\{R_A, F_A\}$ to Server$_j$. Upon receiving $\{R_A, F_A\}$, Server$_j$ computes

$$Q_A = H(SK_i \| R_A),$$

and sends $\{Q_A\}$ to Server$_j$. At this point, A intercepts the message and computes

$$Q_A' = H(SK_{A_i} \| R_A).$$

It is clear that $Q_A' = Q_A$. During the entire process, the user regards $A$ as Server$_j$.

4. Improved Scheme

To overcome the attacks, we proposed an improved scheme based on Wang et al.’s scheme in this section. Our scheme still operates in a multiserver environment, including the initialization, modified server and user registration, and modified login and authentication phases. It is noteworthy that the initialization phase in our scheme is the same as that in Wang et al.’s scheme, and we used a rectangle to denote our modifications.

4.1. Modified Server and User Registration. The server Server$_j$ selects its identity $SID_j$, and sends its identity to the RC through a secure channel. The RC receives this message and selects a random number $e_j$. Subsequently, the RC computes $SM_j = H(SID_j \| x | e_j)$, stores $\{SID_j, e_j\}$, and sends $SM_j$ to Server$_j$. When Server$_j$ receives $SM_j$, it stores it in the database.

The user User$_i$ selects his ID$_i$ and PW$_i$, and imprints $b_i$. Subsequently, User$_i$ selects a random number $r_i$ and computes

$$P_i = H(PW_i \| h(b_i)) r_i,$$
$$HID_i = H(ID_i \| r_i),$$

and sends $\{HID_i, ID_i, P_i\}$ to the RC. The RC receives this message, selects a random number $d_i$, and computes

$$A_i = H(x \| HID_i \| tID_i \| nd_i),$$
$$B_i = A_i \oplus P_i,$$
$$V_i = H(P_i \| H(HID_i)) \mod n,$$

where $2^{4} \leq n \leq 2^{8}$. The RC stores $\{HID_i, ID_i, d_i\}$ in the database, stores $\{B_i, V_i, E_{key}(\cdot), P, P_{pub} \| n\}$ in the SC, and sends the SC to User$_i$ in a secure channel. Next, User$_i$ stores $r_i$ in the SC. The complete registration process is shown in Figure 2.

4.2. Modified Login and Authentication. In this phase, User$_i$ and Server$_j$ complete a mutual authentication and use the RC as an information center to establish an SK. The complete login and authentication processes are shown in Figure 3.

Step 1 User$_i$ enters ID$_i$ and PW$_i$, imprints $b_i$, and logs in the SC. Next, the SC computes

$$P_i = H(PW_i \| h(b_i)) r_i,$$
$$HID_i = H(ID_i \| r_i),$$
$$V_i = H(P_i \| H(HID_i)) \mod n,$$

and verifies if $V_i = V_i$. If they are equal, User$_i$ generates a random number $N_i$ and computes
User \( i \) \( \rightarrow \) RC \( \rightarrow \) Server \( j \)

**Registration phase:**

Choose SID \( j \)

Choose ej

Compute SM\( j = H(SID_j \| x \| ej) \)

Store \{SID\( j, ej\} \) in database

Choose \( ID_i, PW_i, b_i \)

Choose \( r_i \)

Compute \( PI = H(PW_i \| h(b_i) \| r_i) \)

HID\( i = H(ID_i \| r_i) \)

\{HID\( i, ID_i, PI\} \) in database

Choose \( di \)

Compute \( AI = H(x \| HID_i \| ID_i \| di) \)

Bi = AI \( \cdot \) PI, Vi = H(Pi \( \cdot \) HID\( i \)) mod n

Store \{HID\( i, ID_i, di\} \) in database

Store \{Bi, Vi, Ekey(·), P, Ppub, n\} in SC

Store \{ri\} in SC

**Login and authentication phase:**

Input \{ID\( i, PW_i, bi\)\}

Compute \( P'_i = H(PW_i \| h(b_i) \| r_i) \)

HID\( i = H(ID_i \| r_i) \)

\{HID\( i, ID_i, P'_i\} \) \( \rightarrow \)

Choose \( d_i \)

Compute \( A'_i = H(x \| HID_i \| ID_i \| d_i) \)

B\( i = A'_i \cdot P'_i, V'_i = H(P'_i \cdot HID_i) \) mod n

Store \{HID\( i, ID_i, d_i\} \) in database

Store \{B\( i, V'_i, Ekey(·), P, Ppub, n\} \) in SC

Choose \( N_1 \)

Compute \( RS = N_1 \cdot P, E_i = N_1 \cdot R'_i \)

\( SK_i = H(E_i \| [H(A'_i \| C'_i)]) \)

Check \( F'_i = F_i \)

\( Y'_i = H(SID_j \| SM_j) \)

Check \( Y'_i = Y_i \)

Choose \( N_2 \)

Compute \( R'_i = N_2 \cdot P, E'_i = N_2 \cdot R'_i \)

\( SK'_i = H(E'_i \| [H(A'_i \| C'_i)]) \)

Check \( F'_i = F_i \)

**Figure 2:** Registration phase.

**Figure 3:** Login and authentication phase.
\[ A'_i = B_i \oplus P'_i, \]
\[ D_i = H(A'_i \oplus \text{HID}') \oplus N_1, \]
\[ G_i = H(N'_i \oplus A'_i \oplus \text{HID}'), \]
\[ R_i = G_i \oplus P, \]
\[ C_i = H(G_i P_{pub}), \]
\[ L_i = E_{C_i}(\text{HID'} || A'_i || \text{SID}). \] (26)

Subsequently, User, sends \( M_1 = \{D_i, \text{HID'}, L_i\} \) to the RC in the public channel.

Step 2 After the RC receives \( M \), it veriﬁes if \( \text{HID} \) in the public channel.

User generates a random number \( y \) in the database and computes
\[ A'_i = H(x || \text{HID} || \text{ID} || n d), \]
\[ N'_i = D_i \oplus H(A'_i \oplus \text{HID}'), \]
\[ G'_i = H(N'_i \oplus A'_i \oplus \text{HID}'), \]
\[ R'_i = G'_i \oplus P, \]
\[ C'_i = H(x \oplus R_i), \]
\[ \text{(HID'} || A'_i || \text{SID}) = D_{C'_i}(L_i), \]

and veriﬁes if \( A'_i = A'_j \). If they are equal, the RC computes
\[ SM_j = H(\text{SID} || x || t e_i), \]
\[ Y_i = H(\text{SID} || SM_j), \]
\[ M_i = E_{SM_j}(\text{HID'} || R'_i || Y_i || H(A'_i || C'_i)). \] (27)

Next, the RC sends \( M_2 = \{M_i\} \) to Server, in the public channel.

Step 3 After Server, receives \( M_2 \), it computes
\[ (\text{HID'} || R'_i || Y_i || H(A'_i || C'_i)) = D_{SM_j}(M_i), \]
\[ Y'_i = H(\text{SID} || SM_j), \]

and veriﬁes if \( Y'_i = Y'_j \). If they are equal, Server generates a random number \( y \) and computes
\[ R_s = N_2 P, \]
\[ E_s = N_2 R_s, \]
\[ K_s = H(E_s \oplus H(A'_i || C'_i), \]
\[ F'_s = H(\text{HID} || \text{SK} || t R_s || n \text{SID}). \] (30)

Subsequently, Server, sends \( M_3 = \{R_s, F'_s\} \) to User, in the public channel.

Step 4 After User, receives \( M_3 \), it computes
\[ E'_i = N_3 R_s, \]
\[ \text{SK}_i = H(E'_i \oplus H(A'_i || C'_i), \]
\[ F'_i = H(\text{HID} || \text{SK} || t R_s || n \text{SID}). \] (31)

and veriﬁes if \( F'_i = F_i \). If they are equal, \( \text{SK} \) is the session key for User, and Server.

5. Security Analysis

5.1. Formal Security Analysis. In this section, we show the security analysis of our improved scheme in the random oracle model [42]. First, we deﬁne the adversarial model [25, 26, 43–47] and simulate the adversary capabilities in a real attack. In the proposed scheme, three participants, User, Server, and RC, are involved. We use \( \prod^{x}_{y} \), \( \prod^{y}_{z} \), and \( \prod^{z}_{w} \) to represent the \( x \)th communication of User, the \( y \)th communication of Server, and the \( z \)th communication of RC, respectively. To perform a formal security analysis, we deﬁned the following query model for the attacker \( A \).

\[ \text{Execute}(\prod^{K}_{y}, M): A \text{ executes this query with message } M \text{ and then receives the response message from the entity } E \]
\[ \text{Reveal}(\prod^{K}_{y}): A \text{ executes this query to obtain the return result of current session key SK generated by } E \]
\[ \text{Corrupt}(\prod^{K}_{y}): A \text{ executes this query to obtain information } \{B_i, V, r_s, E_{key}(.), P, P_{pub}, n\} \text{ in the smart card } \]
\[ \text{Test}(\prod^{K}_{y}): \text{based on this query, an unbiased coin } c \text{ begins to be ﬂipped. If } c = 0, A \text{ returns SK to a random string, and if } c = 1, A \text{ returns SK to a session key } \]

In the ROR model, the following theorem describes the security of our proposed scheme \( P \).

**Theorem 1.** If \( A \) runs \( P \) in an ROR model against a scheme in polynomial time, \( I \) represents the total number of bits of the biometric. The \( \text{Adv}^{A}_{\text{AKE}} \) that \( A \)’s advantage breaks the security of \( SK \) in \( AKE \) scheme, and then \( \text{Adv}^{A}_{\text{AKE}} \leq (q_{id}^{A}/|\text{Hash}|) + 2 \max|C' \cdot q_{rand}^{A} + (q_{\text{rand}}/2^{n})| + 2 \text{ Adv}^{A}_{\Omega} (k), \) where \( q_{id} \) and \( q_{\text{rand}} \) are the number of Send(\( \prod^{K}_{y}, M \)) and Hash(\( \prod^{x}_{y} || E, M; |\text{Hash}|) \) is the range space of \( h(\cdot); C' \) and \( s \) are parameters of Zipf’s law [48]; \( \text{Adv}^{A}_{\Omega} (k) \) is the advantage of \( A \) breaking the symmetric cipher \( \Omega \).

**Proof.** We define a sequence of five games, namely, \( GM_i \) \( (i = 0, 1, 2, 3, 4) \). Let \( \text{Succ}^{\text{GM}}_{i} \) represent the event that \( U_A \) wins \( GM_i \). The \( \text{Adv}^{A}_{\text{GM}_i} = \text{Pr}[\text{Succ}^{\text{GM}}_{i}] - \text{Pr}[\text{Succ}^{\text{GM}}_{i}] \) represents the advantage of \( A \) winning \( GM_i \), where \( \text{Pr}[E] \) is the probability of event \( E \). The \( \text{Adv}^{A}_{\text{GM}_i} \) represents the advantage of \( U_A \) that breaks the security of \( SK \) in the proposed scheme. The detailed description of \( GM_i \) is as follows.

**Game \( GM_0 \):** \( GM_0 \) is the ﬁrst game that represents a real attack on the ROR model. At this point, select coin \( c \) to
start GMₜ. From semantic security, we can get \( \text{Adv}^{P} = [2 \cdot \text{Adv}^{P}_{GMₜ} - 1] \).

**Game GMₜ:** GMₜ means that A can perform the Execute query and get the message \( \{D_i, \text{HID}_i, L_i, \{M_i\} \} \) and \( \{R_i, F_i\} \) transmitted in the scheme. At the end of the game, A will perform Reveal and Test queries to determine whether SK = H(\( E_i || H(A_i || C_i) \)) can be obtained. But A cannot derive \( E_i, A_i, \) and \( C_i \), so the probability of GMₜ is the same as that of GMₜ₀, that is, \( \text{Adv}^{P}_{GMₜ} = \text{Adv}^{P}_{GMₜ₀} \).

**Game GMₜ₂:** GMₜ₂ has added Hash and Send queries, \( E_i, A_i, \) and \( C_i \), which are all protected by \( h(\bullet) \). But \( E_i, A_i, \) and \( C_i \) are not directly obtained in the transmission channel, and according to the birthday paradox, we can get \( \text{Adv}^{P}_{GMₜ₀} - \text{Adv}^{P}_{GMₜ₁} \leq (q_i^{2}/2)[\text{Hash}] \).

**Game GMₜ₃:** Corrupt query is added in GMₜ₃ and A can get the information \( \{B_i, V_i, r_i, E_{key}(\bullet), P, P_{pub,n}\} \) in the smart card. The User, uses the password and biometric information to register, and A wants to guess \( P_i = H(PW_i || h(b_i) || r_i) \), but the probability of guessing the biometrics is 1/2 [49], which is almost negligible. Using Zipf’s law [48], we can get \( \text{Adv}^{P}_{GMₜ₀} - \text{Adv}^{P}_{GMₜ₃} \leq \max(C' \cdot q_{send} \cdot q_{send}(2^i) \cdot \text{Hash}) \).

**Game GMₜ₄:** GMₜ₄ is the last part of the game. At this time, A attempts to decrypt the information \( \{L_i, M_i\} \) and uses the obtained information \( \{B_i, V_i, r_i, E_{key}(\bullet), P, P_{pub,n}\} \) to infer SK. Without the master key of RC, A cannot compute \( A_i = H(x \cdot \text{HID}_i || \text{ID}_i || \text{sid}_i) \) and \( C_i = H(G_i || P_{pub} = H(H(N_i \oplus A_i \oplus \text{HID}_i || P_{pub})). According to the security of \( \Omega \) symmetric encryption algorithm, we can obtain \( \text{Adv}^{P}_{GMₜ₀} - \text{Adv}^{P}_{GMₜ₄} \leq \text{Adv}^{P}_{GMₜ₄}(k) \).

All queries are performed by A. After querying the test query, only the coin \( c \) of GMₜ₄ is left. Thus, the probability of guessing coin \( c \) is \( \text{Adv}^{P}_{GMₜ₄} = 1/2 \). In summary, we can deduce

\[
\frac{1}{2} \text{Adv}_{A}^{\min} = \text{Adv}^{P}_{GMₜ₀} - \frac{1}{2} \leq \text{Adv}^{P}_{GMₜ₄} - \frac{1}{2} = \text{Adv}^{P}_{GMₜ₄} = \text{Adv}^{P}_{GMₜ₄} \leq \text{Adv}^{P}_{GMₜ₄} - \text{Adv}^{P}_{GMₜ₄} + \text{Adv}^{P}_{GMₜ₄} \leq \frac{q_i^{2}}{2}[\text{Hash}] + \max(C' \cdot q_{send} \cdot q_{send}(2^i) \cdot \text{Hash}) + \text{Adv}^{P}_{\Omega}(k).
\]

Therefore, the advantage of A breaking the scheme is \( \text{Adv}^{P}_{A} \leq (q_i^{2}/2)[\text{Hash}] + 2 \max(C' \cdot q_{send} \cdot q_{send}(2^i)) + 2 \text{Adv}^{P}_{\Omega}(k) \).

5.2. Formal Security Analysis by BAN Logic. In this subsection, we demonstrate through the BAN logic that after our scheme verifies the authenticity of each other’s identity and that the determined SK will not be obtained by others. In fact, the BAN logic is a rule used to define and analyze the communication process between two parties. Specifically, the conclusions obtained by the BAN logic are through rigorous logic analysis, which further explains the confidentiality and credibility of the communication information. The notations and rules of the BAN logic used in the BAN logic calculation performed in this study are cited in [24, 27, 28, 30, 31, 36, 50, 51]. The proof of our scheme is as follows:

5.2.1. Rules

- rule (1) Nonce verification rule: (\( P \equiv \#(X), P \equiv Q \equiv X \) \( \Rightarrow (P \equiv Q \equiv X) \))
- rule (2) Message meaning rule: (\( P \equiv P_{\rightarrow}^{Q}, P \equiv X_{\rightarrow} \) \( \Rightarrow (P \equiv Q \equiv X) \))
- rule (3) Jurisdiction rule: (\( P \equiv \#(X) \equiv X, P \equiv Q \equiv X \equiv X \Rightarrow (P \equiv X) \))
- rule (4) Jurisdiction rule: (\( P \equiv \#(X) \equiv (P \equiv \#(X,Y)) \))

5.2.2. Goals

- **Goal 1.** \( U \equiv S \equiv \text{SK} \)
- **Goal 2.** \( U \equiv \text{SK} \)
- **Goal 3.** \( S \equiv U \equiv \text{SK} \)
- **Goal 4.** \( S \equiv U \equiv \text{SK} \)

5.2.3. Idealize the Communication Messages

\[
M_1: U \rightarrow RC: \{D_i, \text{HID}_i, \{\text{HID}_i, U_{\rightarrow}^{A} \rightarrow \text{RC, SID}\} \}_{C_i} \]
\[
M_2: RC \rightarrow S: \{\text{HID}_i, R_i, Y_i, U_{\rightarrow}^{H(A_i)C_i} \rightarrow S\}_{SM_i} \]
\[
M_3: S \rightarrow U: \{R_i, \{\text{HID}_i, U_{\rightarrow}^{S, R_i, SID} \rightarrow H(A_i)C_i\} \}_{SM_i} \]

5.2.4. Initial Assumptions. \( A_1: S \equiv \text{SK} \rightarrow RC \)

\[
A_2: S \equiv \#(N_i) \]
\[
A_3: S \equiv RC \equiv U_{\rightarrow}^{H(A_i)C_i} \rightarrow S \]
\[
A_4: S \equiv U_{\rightarrow}^{H(A_i)C_i} \rightarrow S \]
\[
A_5: S \equiv \#(N_i) \]
\[
A_6: S \equiv \#(N_i) \rightarrow U_{\rightarrow}^{S, RC} \rightarrow S \]
\[
A_7: S \equiv \#(N_i) \rightarrow U_{\rightarrow}^{S, RC} \rightarrow S \]

5.2.5. The Proof of Our Proposed Scheme. For **Goal 1** By \( M_2 \), we have \( S \equiv \{\text{HID}_i, R_i, Y_i, U_{\rightarrow}^{H(A_i)C_i} \rightarrow S\} \). Based on \( A_1, S_1, \) and rule (2), we have \( S_2: S \equiv RC \sim \{\text{HID}_i, R_i, Y_i, U_{\rightarrow}^{H(A_i)C_i} \rightarrow S\} \). Based on \( A_2 \) and rule (4), we obtain \( S_3: S \equiv \# \{\text{HID}_i, R_i, Y_i, U_{\rightarrow}^{H(A_i)C_i} \rightarrow S\} \). Using \( S_3, S_2, \) and rule (1), \( S_4: S \equiv RC \equiv \{\text{HID}_i, R_i, Y_i, U_{\rightarrow}^{H(A_i)C_i} \rightarrow S\} \). Subsequently, we have \( S_5: S \equiv RC \equiv U_{\rightarrow}^{H(A_i)C_i} \rightarrow S \). Based on \( A_3, \)
Security and Communication Networks

5.3. Security Verification by ProVerif. We used the verification tool ProVerif to test the security of our proposed scheme. ProVerif is an important verification tool for verifying security fundamentals such as authentication, confidentiality, anonymity, and privacy [11, 24, 51, 52]. Furthermore, ProVerif can automatically verify the security of a scheme. It handles basic elements such as public key cryptography and the Diffie–Hellman mechanism.

The definition of the ProVerif code is shown in Figure 4. Our scheme comprised three entities: User, RC, and Server. Figures 5–7 show the user, RC, and server processes in our code, respectively. Five events were involved: UserAuthed, UserStarted, RACACUser, ServerACRC, and UserACServer. Event UserAuthed means that User, has been successfully authenticated. Event UserStarted means that User has started authentication. Event RACACUser means that the RC has successfully authenticated the User. Event ServerACRC means that Server has successfully authenticated the RC. Event UserACServer means that User, has successfully authenticated Server.

Next, we used ProVerif to query whether the attacker can obtain the identities of User, and Server as well as the SK and whether the events above were executed in sequence. Figure 8 shows the events and queries in the code.

Finally, we executed the code to perform authentication, and the results are shown in Figure 9. The result shows that ProVerif confirmed the security of our scheme. Therefore, the attacker cannot obtain parameters \(\{SK_i, SK_j, ID_i, SID_j\}\),

\[ S_5, \text{ and rule (3), we have } S_6: S_i \equiv U^{H(A_i|C_j)} S. \text{ By } M_3, \text{ we have } S_7: U \vdash \left\{ \begin{array}{l} \text{SK}_i, \text{SK}_j \vdash S_i, R_i, SID_i \end{array} \right\}. \text{ Based on } A_4, S_7, \text{ and rule (2), we have } S_8: U \equiv S_i \sim \left\{ \begin{array}{l} \text{SK}_i, \text{SK}_j \vdash U \leftrightarrow S_i, R_i, SID_i \end{array} \right\}. \]

Based on \(A_5, S_9, \) rule (4), and rule (1), we obtain \(S_9: U \equiv S_i \equiv \left\{ \begin{array}{l} \text{SK}_i, \text{SK}_j \vdash U \leftrightarrow S_i, R_i, SID_i \end{array} \right\} \) and the following:

\[ S_{10}: U \equiv S_i \equiv U \leftrightarrow S. \text{ (Goal 1).} \]

For Goal 2, based on \(A_6, S_{10}, \) and rule (3), we have \(S_{11}: U \equiv U \leftrightarrow S. \text{ (Goal 2).} \)

For Goal 3, based on \(M_3, \) we have \(S_{12}: S_i \equiv \left\{ \begin{array}{l} \text{SK}_i, \text{SK}_j \vdash U \leftrightarrow S_i, R_i \end{array} \right\}. \text{ Subsequently, based on } S_6, S_{12}, \text{ and rule (2), we have } S_{13}: S_i \equiv U \sim \left\{ \begin{array}{l} \text{SK}_i, \text{SK}_j \vdash U \leftrightarrow S_i, R_i \end{array} \right\}. \text{ Based on } A_7, S_{13}, \text{ rule (4), and rule (1), we obtain } S_{14}: S_i \equiv U \equiv \left\{ \begin{array}{l} \text{SK}_i, \text{SK}_j \vdash U \leftrightarrow S_i, R_i \end{array} \right\}. \text{ Therefore, we have } S_{15}: S_i \equiv U \equiv U \leftrightarrow S. \text{ (Goal 3).} \]

For Goal 4, based on \(A_8, S_{15}, \) and rule (3), we have \(S_{16}: S_i \equiv U \leftrightarrow S. \text{ (Goal 4).} \)

\[ \left( \begin{array}{l} \text{channel} \end{array} \right) \]

```
let ProcessUser =
  new IDi : bitstring (** the user’s ID **) 
  new PWi : bitstring (** the user’s password **) 
  new bi : bitstring; 
  new ri : bitstring; 
  let Pi = H1(con(PWi,H2(bi),ri)) in 
  let HIDi = H1(xor(IDi,ri)) in 
  out(sch,(HIDi,DI,pi)); (** registration **)
  in(sch,(bi,bitstring, xekey-bitstring,xEkey-bitstring));
  |
  \{event UserAuthed();
  let P' = H1(con(PW,H2(bi))); in
  let HID' = H1(xor(ID,ri)) in
  let V' = mod(H1(xor(P',H1(HID')))) in
  if V' = xi then
  new N1:bitstring;
  new SIDj:bitstring;
  let A'i = xor(xBi,P') in
  let Di = xor(H1(xor(Ai',HID')),N1) in
  let Gi = H1(xor(N1,xor(A'HID'))) in
  let Ri = mult(GL,i) in
  let Ci = H1(mult(GL,Pub)) in
  let Li = senc(con(HID,Di',SIDj,CI)) in
  out(ch,(Di,HID',Li)); (** authentication **)
  in(ch,(xRi:bitstring,xFBi));
  let E'i = mult(N1,xRi) in
  let SKi = H1(con(E'i,H1(con(A'i,CI)))) in
  let F' = H1(con(con(HID SKi),xRs),SIDj)) in
  if F' = xi then event UserAcServer();
  event UserAuthed();
  0
  ).
```

Figure 4: The terms definition in the ProVerif tool.

Figure 5: Process of User, in ProVerif tool.
let RCAuth =  
let UserReg =  
let ProcessRC = ServerReg | UserReg | RCAuth. 

let DR = sdec(yL in
let Ci ′ = H1(mult(x,Ri ′)) in
let HIDi = getmess(DR) in
let SID = getmess(DR) in
let AI ′ = getmess(DR) in
if AI ′ = AI then event RCAcUser();
let SMj = H1(con(SIDj,SMj)) in
let Yi ′ = H1(con(SIDj,yHIDi)) in
let AC = H1(con(Ai ′,Yi ′)) in
let Mi = senc(con(con(HIDi,Ri ′),Yi),AC),SMj)) in
out(ch,(Mi));
).

let ProcessRC = ServerReg | UserReg | RCAuth. 

Figure 6: Process of RC in ProVerif tool.

and all events are executed normally. Note that Figures 4–9 are shown in Appendix.

5.4 Informal Security Analysis

5.4.1. Known Session-Specific Temporary Information Attacks. Upon completing the login and authentication phase, if N1 or N2 is compromised, then A intercepts information {Ri,Fi} and computes Ei = N1,Ri, but it cannot compute A1 = H(x∥HIDi∥IDi∥ydi) and C1 = H(G,Ppub) = H(H(N1 ∥A1∥HIDi∥Ppub)). Therefore, A cannot compute the SK, and the scheme successfully overcomes known session-specific temporary information attacks.

5.4.2. User Impersonation Attacks. Assume that the A pretends to be a user and forges a message M1 = {Di,HIDi,Li}. Even if A forges a random number N1, it cannot compute A1 to forge Di and L1. A cannot obtain A1 for two reasons. First, upon completing the login and entering the authentication phase, A is encrypted by C1, and A cannot compute C1 to decrypt A1; therefore, A cannot be obtained. Second, in the registration phase, if the SC is stolen by a malicious user, then A can obtain Bi. However, because A = B∥P1, A requires {PW1,b1,r1} to compute P1, which is impossible. Therefore, the scheme successfully overcomes user impersonation attacks.

5.4.3. Server Impersonation Attacks. Assume that A pretends to be the server and forges a message M3 = {Ri,Fi}. Therefore, A must generate a random number N_A and compute {R3 = N_A∥P1, E_A = N_A∥Ri, SK_A = H(E_A∥Ht(A1∥C1))}. However, A cannot obtain {R1,A1,C1}. Even if A can obtain temporary information N1, it cannot compute {R1,A1,C1}, nor can it obtain sensitive information by
deciphering $L_i$. Therefore, the scheme can overcome server impersonation attacks.

5.4.4. Man-in-the-Middle Attacks. Upon completing the login and authentication phase, the $A$ intercepts the messages transmitted between User and Server, to impersonate the user or server. The $A$ may intercept $M_3$ to impersonate Server. However, $A$ cannot compute $F_i = H(HID∥SK∥ID∥SID)$; therefore, the session is terminated. In another case, $A$ may intercept $M_1, M_2$ to impersonate User. However, $A$ cannot compute $A_i$; therefore, it cannot pass the RC verification. Therefore, the scheme can overcome man-in-the-middle attacks.

5.4.5. Replay Attacks. Suppose that message $M_1, M_2$, or $M_3$ is replayed by $A$. However, our scheme overcomes this attack by refreshing random numbers $\{N_1, N_2\}$. By replaying one of the messages $M_1, M_2, M_3$, the mutual authentication values $F_i$ for the user will not pass, and the session will be terminated. Therefore, this scheme can overcome replay attacks.

5.4.6. Stolen SC Attacks. Suppose that the SC is stolen by a malicious user $A$ who will obtain $\{B_i, V_i, E_{key}(\cdot), P, P_{pub}, n, r_i\}$. However, based on those values, $A$ cannot compute $A_i = H(x∥HID∥ID∥SID)$; in addition, $A$ cannot obtain $N_1$ to compute $C_i = H(P_{pub}) = H(H(N_1∥A\oplus HID)∥P_{pub})$ and $E_i = N_1R_5$. Therefore, $A$ cannot compute $SK = H(E_i∥H(A_i∥C_i))$. Hence, it is clear that the scheme can successfully overcome stolen SC attacks.

5.4.7. Offline Password-Guessing Attacks. According to, $A$ obtains $\{B_i, V_i, E_{key}(\cdot), P, P_{pub}, n, r_i\}$. Moreover, $A$ can be biometric $b_i$ by shoulder surfing. $A$ launches an offline password-guessing attack by comparing $P_i = H(PW∥h(b_i))∥r_i)$. In addition, $P_i = A_i\oplus B_i = H(x∥HID∥ID∥SID‖b_i). However, $A$ cannot obtain HID, $x$, and $d_i$; therefore, the attacker cannot compute $P_i$. Hence, the scheme can overcome offline password-guessing attacks.

5.4.8. Privileged Insider Attacks. Assume that the privileged insider $A$ is $\{HID_i, ID_i, d_i\}$ stored in the RC database. However, $A$ cannot obtain $x$ and the user’s $ID_i$; therefore, it cannot compute $A_i = H(x∥HID_i‖ID_i‖d_i)$. Because $E_i = N_iR_i = N_iH(N_i∥A_i\oplus HID\{P_{pub} = H(H(N_i∥A_i\oplus HID)∥P_{pub})\}$. In other words, $A$ cannot compute $SK = H(E_i∥H(A_i∥C_i))$. Therefore, this scheme provides perfect forward secrecy.

5.4.9. Perfect Forward Secrecy. Suppose that $A$ obtains the RC’s long-term key $x$ and attempts to obtain the SK. If $A$ obtains $N_1$ and intercepts $\{R_i, E_i\}$, then it computes $E_i = N_iR_i$. However, $A$ cannot compute $A_i = H(x∥HID_i∥ID_i∥d_i)$ and $C_i = H(G_iP_{pub}) = H(H(N_i∥A_i\oplus HID)∥P_{pub})$. In other words, $A$ cannot compute $SK = H(E_i∥H(A_i∥C_i))$. Therefore, this scheme provides perfect forward secrecy.

5.4.10. User Anonymity. In the registration phase of the improved scheme, User, computes $HID_i = H(ID_i∥r_i)$ to protect the real identity of the user. In the authentication phase, the user transmits the virtual identity $HID_i$ and the attacker cannot obtain the real identity of the user. Therefore, our scheme provides user anonymity.

5.4.11. Three-Factor Secrecy. The three factors refer to the password, SC, and biometrics. Based on a previous analysis, $A_i$ and $C_i$ are the key parameters for launching an attack to compute the SK. $A$ obtains two of the three factors, i.e., the password and SC. Even if $A$ obtains the password and extracts the parameters from the SC, it cannot compute $A_i$ and $C_i$ to perform any attack. Passwords and biometrics: if $A$ obtains the password and biometrics to calculate $A_i$, it must obtain $B_i$ and $P_i$. However, $B_i$ is stored in an SC, whereas $P_i$ is protected by a random number. Biometrics and smart cards: if $A$ obtains the biometrics and SC to calculate $P_i$, it must obtain the PW. Therefore, $A$ cannot compute $A_i = B_i∥P_i$.

After analyzing the security of our improved scheme, we can conclude that our proposed scheme is “provably secure” against several well-known attacks with a higher probability. However, it does not mean that our scheme is a “perfectly secure” authentication scheme because many special attack approaches or tricks exist [19].
improved scheme overcame all known attacks. Hence, it is clear that only our proposed protocol is secure against well-known attacks. Although Wang et al.’s scheme [36] guaranteed perfect forward secrecy, it could not overcome known session-specific temporary information attack, user impersonation, and server impersonation attacks, nor could it provide three-factor secrecy. Although Ali et al.’s scheme [34] could not overcome known session-specific temporary information, user impersonation, and server impersonation attacks, nor could it provide three-factor secrecy. Although Wang et al.’s scheme [36] guaranteed perfect forward secrecy, it could not overcome known session-specific temporary information, user impersonation, and server impersonation attacks. Hence, it is clear that only our proposed protocol successfully overcame all known attacks and achieved a certain degree of security.

A comparison of the computational costs is shown in Table 4. We used JPBC-2.0.0 (Pairing-Based Cryptography Library) [53], IntelliJ IDEA 2020.2.1 community edition, and a Windows 10 computer with a 2.3 GHz Intel (R) Core i$5$ processor and 16 GB of memory to simulate the computational costs. It is noteworthy that a widely accepted Type A pairing was constructed on the curve $y^2 = x^3 + x$ over $F_{q^3}$, where $q$ is a prime satisfying $q = 3 \ mod \ 4$. In our experimental results, $T_m$ was 13.5 ms, $T_p$ was 0.48 ms, and $T_s$ was 0.12 ms. As shown in Table 4, the computational cost of our scheme was lower than that of the scheme in [34], whereas it was 13.5 ms higher than that of the scheme in [36]. However, when our scheme was utilized in a practical application, the 13.5 ms difference was almost negligible. Meanwhile, the scheme in [36] was subject to known session-specific temporary information, user impersonation, and server impersonation attacks. However, our improved scheme overcame all known attacks.

### 6. Performance Comparison

In this section, we compare our improved scheme with those of Ali and Pal [34] and Wang et al. [36] in terms of security and efficiency. Table 3 presents a comparison of security among the abovementioned schemes. It is evident that our scheme is secure against well-known attacks. Ali and Pal’s scheme [34] could not overcome known session-specific temporary information, user impersonation, and server impersonation attacks, nor could it provide three-factor and perfect forward secrecy. Although Wang et al.’s scheme [36] guaranteed perfect forward secrecy, it could not overcome known session-specific temporary information, user impersonation, and server impersonation attacks. Hence, it is clear that only our proposed protocol successfully overcame all known attacks and achieved a certain degree of security.

| Attack methods                                      | Ali and Pal’s scheme [34] | Wang et al.’s scheme [36] | Our scheme |
|-----------------------------------------------------|---------------------------|---------------------------|------------|
| User anonymity                                      | √                         | ×                         | √          |
| Offline password attacks                            | √                         | √                         | √          |
| Stolen smart card attacks                           | √                         | √                         | √          |
| Known session-specific temporary information attack | × [36]                    | ×                         | √          |
| User impersonation attack                           | × [36]                    | ×                         | √          |
| Server impersonation attack                         | × [36]                    | ×                         | √          |
| Replay attacks                                      | √                         | √                         | √          |
| Perfect forward secrecy                             | √ [36]                    | √                         | √          |
| Three-factor secrecy                                | √ [36]                    | √                         | √          |

Table 3: Security comparison.

Note. √, able to overcome the attack, and ×, unable to overcome the attack.

### Table 4: Computation cost comparison.

| Scheme                          | User computations | Server computations | RC | Total |
|---------------------------------|-------------------|---------------------|----|-------|
| Ali et al.’s scheme [34]        | $3T_m + 4T_p \approx 42.42$ ms | $4T_m + 3T_p + 2T_s \approx 55.56$ ms | $3T_m + 3T_p + 2T_s \approx 42.18$ ms | $10T_m + 10T_p + 3T_s \approx 140.16$ ms |
| Wang et al.’s scheme [36]       | $3T_m + 1T_s \approx 40.62$ ms | $2T_m + 1T_s \approx 27.12$ ms | $1T_m + 2T_s \approx 13.74$ ms | $6T_m + 4T_s \approx 71.01$ ms |
| Our scheme                      | $3T_m + 1T_s \approx 40.62$ ms | $2T_m + 1T_s \approx 27.12$ ms | $2T_m + 2T_s \approx 27.24$ ms | $7T_m + 4T_s \approx 94.98$ ms |

$T_m$, time for executing elliptic curve scalar point multiplication. $T_p$, time for performing elliptic curve point addition operation.

Table 5 shows a comparison of the communication costs. We assumed that the ECC points accounted for 320 bits because two 160-bit parameters form an ECC point. The hash operation was considered to be 256 bits, and the identity was 64 bits. The length of the ciphertext for a symmetric encryption was 256 bits. In Ali et al.’s scheme, the messages in the login and authentication phase were $\{DID, E_i, C_i, D_i\}, \{DID^{new}, K_i, L_i, F_i\}, \{DID^{new}, Q_i, M_i, K_i\}$, and $\{Z_i\}$, where $\{E_i, C_i, K_i, Q_i, Z_i\}$ belong to ECC, $\{D_i, L_i, M_i\}$ are hash values, and $\{DID, DID^{new}, F_i\}$ are ciphertexts. The total communication cost of Ali et al.’s scheme was 3712 bits. In Wang et al.’s scheme, the messages in the login and authentication phase were $\{R_i, L_i\}, \{M_i\}, \{R_0, F_i\}$, and $\{Q_i\}$, where $\{R_i, R_0\}$ belong to ECC, $\{F_i, Q_i\}$ are hash values, and $\{L_i, M_i\}$ are ciphertexts. The total communication cost of Wang et al.’s scheme was 1664 bits. In our scheme, the messages in the login and authentication phases were $\{D_i, HID, L_i\}, \{M_i\}$, and $\{R_0, F_i\}$, where $\{R_0\}$ belongs to ECC, $\{D_i, HID, F_i\}$ are hash values, and $\{L_i, M_i\}$ are ciphertexts. The total communication cost of our scheme was 1600 bits.

Table 5: Comparison of communication and massage rounds.

| Scheme                          | Communication cost (bits) | Massage rounds |
|---------------------------------|---------------------------|---------------|
| Ali and Pal’s scheme [34]       | 3712                      | 4             |
| Wang et al.’s scheme [36]       | 1664                      | 4             |
| Our scheme                      | 1600                      | 3             |
analysis mentioned in Table 3, our scheme also has strong security. Hence, our scheme is worthy of being adopted in secure three-factor authentication.

7. Conclusion
In this study, we performed a security analysis of Wang et al.’s scheme and discovered that their scheme could not overcome known session-specific temporary information, user impersonation, and server impersonation attacks. Additionally, we have proven the security of our proposed scheme through formal and informal security analysis. Subsequently, the communication security of our scheme was validated by the ProVerif tool, and the BAN logic indicated that mutual authentication can be completed safely. Finally, through a comparison of performance and security, the security and efficiency of our proposed scheme was proven. However, the computational cost of our scheme is still high. It will lead us to design lightweight authentication schemes in the future.

Data Availability
The data used to support the findings of this study are included within the article.

Conflicts of Interest
The authors declare no conflicts of interest.

References
[1] J.-S. Pan, P. Hu, and S.-C. Chu, "Novel parallel heterogeneous meta-heuristic and its communication strategies for the prediction of wind power," *Processes*, vol. 7, no. 11, p. 845, 2019.
[2] T.-Y. Wu, C.-M. Chen, K.-H. Wang, C. Meng, and E. K. Wang, "A provably secure certificateless public key encryption with keyword search," *Journal of the Chinese Institute of Engineers*, vol. 42, no. 1, pp. 20–28, 2019.
[3] J. M.-T. Wu, J. C.-W. Lin, and A. Tamrakar, "High-utility itemset mining with effective pruning strategies," *ACM Transactions on Knowledge Discovery from Data*, vol. 13, no. 6, pp. 1–22, 2019.
[4] Z. Meng, J.-S. Pan, and K.-K. Tseng, "PaDE: an enhanced Differential Evolution algorithm with novel control parameter adaptation schemes for numerical optimization," *Knowledge-Based Systems*, vol. 168, pp. 80–99, 2019.
[5] A.-Q. Tian, S.-C. Chu, J.-S. Pan, H. Cui, and W.-M. Zheng, "A compact pigeon-inspired optimization for maximum short-term generation mode in cascade hydroelectric power station," *Sustainability*, vol. 12, no. 3, p. 767, 2020.
[6] S. C. Chu, X. S. Xue, J. S. Pan, and X. J. Wu, "Optimizing ontology alignment in vector space," *Journal of Internet Technology*, vol. 21, no. 1, pp. 15–22, 2020.
[7] Z. G. Du, J. S. Pan, S. C. Chu, H. J. Luo, and P. Hu, "Quasi-affine transformation evolutionary algorithm with communication schemes for application of RSSI in wireless sensor networks," *IEEE Access*, vol. 8, pp. 858–8594, 2020.
[8] J. Zhang, H. Liu, and L. Ni, "A secure energy-saving communication and encrypted storage model based on RC4 for EHR,” *IEEE Access*, vol. 8, pp. 38995–39012, 2020.
[9] J. Wang, X. Gu, W. Liu, A. K. Sangaiah, and H. J. Kim, "An empower Hamilton loop based data collection algorithm with mobile agent for WSNs," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, pp. 1–14, 2019.
[10] J. Wang, Y. Gao, K. Wang, A. K. Sangaiah, and S.-J. Lim, "An affinity propagation-based self-adaptive clustering method for wireless sensor networks," *Sensors*, vol. 19, no. 11, p. 2579, 2019.
[11] H. Xiong, Y. Wu, C. Jin, and S. Kumari, “Efficient and privacy-preserving authentication protocol for heterogeneous systems in IIoT,” *IEEE Internet of Things Journal*, vol. 1, 2020.
[12] H. Xiong, Y. Zhao, Y. Hou et al., "Heterogeneous signcryption with equality test for IIoT environment,” *IEEE Internet of Things Journal*, vol. 1, 2020.
[13] J. H. Yang and C. C. Chang, "An id-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Computers & Security*, vol. 28, no. 3–4, pp. 138–143, 2009.
[14] S. H. Islam and G. Biswas, "A more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Journal of Systems and Software*, vol. 84, no. 11, pp. 1892–1898, 2013.
[15] T. T. Truong, M. T. Tran, and A. D. Duong, "Improvement of the more efficient and secure id-based remote mutual authentication with key agreement scheme for mobile devices on ecc,” in *Proceedings of the 2012 26th international Conference on Advanced information Networking and Applications Workshops*, pp. 698–703, IEEE, Fukuoka, Japan, March 2012.
[16] E. Erdem and M. T. Sandıkkaya, "OTPaaS—one time password as a service," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 3, pp. 743–756, 2018.
[17] S. A. Chaudhry, "Correcting "PALK: password-based anonymous lightweight key agreement framework for smart grid,” *International Journal of Electrical Power & Energy Systems*, vol. 125, p. 106529.
[18] M. Karuppiah and R. Saravanan, "A secure remote user mutual authentication scheme using smart cards," *Journal of Information Security and Applications*, vol. 19, no. 4, pp. 282–294, 2014.
[19] D. Wang, D. B. He, P. Wang, and C. H. Chu, "Anonymous two-factor authentication in distributed systems: certain goals are beyond attainment,” *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 4, pp. 428–442, 2014.
[20] D. Wang and P. Wang, "On the anonymity of two-factor authentication schemes for wireless sensor networks: attacks, principle and solutions,” *Computer Networks*, vol. 73, pp. 41–57, 2014.
[21] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan, and N. Kumar, "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, pp. 1–18, 2015.
[22] Q. Xie, D. S. Wong, G. Wang, X. Tan, K. Chen, and L. Fang, "Provably secure dynamic ID-based anonymous two-factor authenticated key exchange protocol with extended security model,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 6, pp. 1382–1392, 2017.
[23] C.-M. Chen, B. Xiang, Y. Liu, and K.-H. Wang, "A secure authentication protocol for Internet of vehicles,” *IEEE Access*, vol. 7, no. 1, pp. 12047–12057, 2019.
multi-server architecture in 5G networks,” *IEEE Access*, vol. 8, pp. 28096–28108, 2020.

[25] D. Wang, Q. Gu, H. Cheng, and P. Wang, “The request for better measurement: a comparative evaluation of two-factor authentication schemes,” in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pp. 475–486, Xi an, China, June 2016.

[26] D. Wang, W. Li, and P. Wang, “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. 4081–4092, 2018.

[27] 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.

[28] D. He and D. Wang, “Robust biometrics-based authentication scheme for multiserver environment,” *IEEE Systems Journal*, vol. 9, no. 3, pp. 816–823, 2014.

[29] X. Huang, Y. Xiang, E. Bertino, J. Zhou, and L. Xu, “Robust multi-factor authentication for fragile communications,” *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 6, pp. 568–581, 2014.

[30] V. Odelu, A. K. Das, and A. Goswami, “A secure biometrics-based multi-server authentication protocol using smart cards,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1953–1966, 2015.

[31] X. Li, J. Niu, S. Kumari, J. Liao, and W. Liang, “An enhancement of a smart card authentication scheme for multiserver architecture,” *Wireless Personal Communications*, vol. 80, no. 1, pp. 175–192, 2015.

[32] S. Kumari, X. Li, F. Wu, A. K. Das, K-K. R. Choo, and J. Shen, “Design of a provably secure biometrics-based multi-cloud-server authentication scheme,” *Future Generation Computer Systems*, vol. 68, pp. 320–330, 2017.

[33] Q. Feng, D. He, S. Zeadally, and H. Wang, “Anonymous biometrics-based authentication scheme with key distribution for mobile multi-server environment,” *Future Generation Computer Systems*, vol. 84, pp. 239–251, 2018.

[34] R. Ali and A. K. Pal, “An efficient three factor–based authentication scheme in multiserver environment using ECC,” *International Journal of Communication Systems*, vol. 31, no. 4, Article ID e3484, 2018.

[35] S. Hussain and S. A. Chaudhry, “Comments on “Biometrics-Based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment”,” *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10936–10940, 2019.

[36] F. Wang, G. Xu, C. Wang, and J. Peng, “A provably secure biometrics-based authentication scheme for multiserver environment,” *Security and Communication Networks*, vol. 2019, pp. 1–15, Article ID 2838615, 2019.

[37] C. , M. Chen, Y. Y. Huang, K. H. Wang, S. Kumari, and M. E. Wu, “A secure authenticated and key exchange scheme for fog computing,” *Enterprise Information Systems*, vol. 2020, pp. 1–16, 2020.

[38] H. Luo, G. Wen, and J. Su, “Lightweight three factor scheme for real-time data access in wireless sensor networks,” *Wireless Networks*, vol. 26, no. 2, pp. 955–970, 2020.

[39] C. Shehzad Ashraf, S. Taeshik, A. T. Fadi, and H. A. Mohammed, “Correcting design flaws: an improved and cloud assisted key agreement scheme in cyber physical systems,” *Computer Communications*, vol. 153, pp. 527–537, 2020.

[40] C.-G. Ma, D. Wang, and S.-D. Zhao, “Security flaws in two improved remote user authentication schemes using smart cards,” *International Journal of Communication Systems*, vol. 27, no. 10, pp. 2215–2227, 2014.

[41] R. Canetti and H. Krawczyk, “Analysis of key-exchange protocols and their use for building secure channels,” in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 453–474, Springer, Austria, May 2001.

[42] O. Goldreich and S. Halevi, “The random oracle methodology, revisited,” in *Proceedings of the 30th ACM Symposium. Theory of Computing (STOC)*, pp. 209–218, Chicago, IL, USA, June 1998.

[43] X. Huang, X. Chen, J. Li, Y. Xiang, and L. Xu, “Further observations on smart-card-based password-authenticated key agreement in distributed systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1767–1775, 2013.

[44] D. Wang, N. Wang, P. Wang, and S. Qing, “Preserving privacy for free: efficient and provably secure two-factor authentication scheme with user anonymity,” *Information Sciences*, vol. 321, pp. 162–178, 2015.

[45] D. Wang and P. Wang, “Two birds with one stone: two-factor authentication with security beyond conventional bound,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 708–722, 2016.

[46] P. Gope, A. K. Das, N. Kumar, and Y. Cheng, “Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4957–4968, 2019.

[47] S. Qiu, D. Wang, G. Xu, and S. Kumari, “Practical and provably secure three-factor Authentication protocol based on extended chaotic-maps for mobile lightweight devices,” *IEEE Transactions on Dependable and Secure Computing*, vol. 1, 2020.

[48] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, “Zipf’s law in passwords,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2776–2791, 2017.

[49] V. Odelu, A. K. Das, and A. Goswami, “A secure biometrics-based multi-server authentication protocol using smart cards,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1953–1966, 2015.

[50] C.-M. Chen, C.-T. Li, S. Liu, T.-Y. Wu, and J.-S. Pan, “A provable secure private data delegation scheme for cloud-assisted key agreement in distributed systems in emergency system,” *IEEE Access*, vol. 5, pp. 3410–3422, 2017.

[51] C.-T. Li, T.-Y. Wu, and C.-M. Chen, “A provably secure group key agreement scheme with privacy preservation for online social networks using extended chaotic maps,” *IEEE Access*, vol. 6, pp. 66742–66753, 2018.

[52] C.-M. Chen, K.-H. Wang, K.-H. Yeh, B. Xiang, and T.-Y. Wu, “Attacks and solutions on a three-party password-based authenticated key exchange protocol for wireless communications,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 8, pp. 3133–3142, 2019.

[53] A. De Caro, V. Iovino, and J. P. B. C., “Java pairing based cryptography,” in *Proceedings of the 2011 IEEE Symposium on Computers and Communications (ISCC)*, pp. 850–855, Kerkyra, Greece, July 2011.