IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 31, 2020, accepted February 16, 2020, date of publication February 21, 2020, date of current version March 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2975615

A Secure Biometrics and PUFs-Based Authentication Scheme With Key Agreement For Multi-Server Environments

JUN ZHAO[1], WEIXIN BIAN[1,2], DEQIN XU[1], BIAO JIE[1], XINTAO DING[1], WEN ZHOU[1], AND HUI ZHANG[1]

[1]School of Computer and Information, Anhui Normal University, Wuhu 241002, China
[2]Anhui Province Key Laboratory of Network and Information Security, Wuhu 241002, China

Corresponding author: Weixin Bian (bwx2353@ahnu.edu.cn)

This work was supported in part by the Anhui Provincial Natural Science Foundation under Grant 1708085MF145 and Grant 1808085MF171, and in part by the National Natural Science Foundations of China under Grant 61972438 and Grant 61976006.

ABSTRACT The emergence of multi-server authentication key protocol schemes provides a viable environment for users to easily access the services of multiple legitimate servers through a single registration. Biometric identification technology has the characteristics of forgery difficulty, duplication difficulty and guess difficulty, etc. Therefore, it is an indispensable authentication technology in smart card-based user authentication protocol. There are many shortcomings in the existing schemes based on biometrics, including leakages of biometrics information, smart card theft attack, lack of user anonymity, user impersonation attack, server impersonation, and so on. To overcome these shortcomings, we propose a new user authentication and key agreement scheme in the multi-server environment. To some extent, we not only are able to guarantee the communication security between the user and the servers, but also ensure the physical security of the smart card and biometrics information. In this respect, we use lightweight cryptographic primitives, such as Physically Unclonable Functions (PUFs), Fuzzy extractor and One-way hash functions, and so on. The proposed scheme can effectively protect user's anonymity without the use of password and provide mutual authentication and key agreement in the multi-server environment. Subsequently, we used informal analysis, Burrows-Abadi-Needham Logic (BAN-Logic) proof, and a widely accepted Real-Or-Random model to prove the security and robustness of proposed scheme. Finally, our authentication protocol can protect the security of communication.

INDEX TERMS Multi-server authentication, mutual authentication, physical unclonable function, biometric security and privacy, fuzzy extractor.

I. INTRODUCTION

With the continuous development of Internet and communication technologies and the growing demand for shared data resources, people need to access several different servers anytime, anywhere to meet their needs. In lots of areas, such as e-commerce, telemedicine information systems, and distributed cloud storage systems, secure and efficient communication between participants are becoming increasingly important. Clearly, privacy protection has become an important issue for secure and trusted communications. In this context, remote authentication is required to establish secure communication between the user (client) and the remote server. For example, only authorized private users can access resources stored in the cloud server [1], [2], [5], [6]. In order to deal with security, confidentiality and access rights, many documents have user authentication schemes for single-server environments [3], [4], [6].

In recent years, distributed environments have emerged and are rapidly evolving. In this environment, various servers cooperate to provide services and resources for user services. In this case, single-server authentication scheme is more difficult, above all, for these users who need to register with each server separately. Besides, in order to overcome the multi-registration problem of numerous different servers, a multi-server user authentication scheme [1], [2], [5] is proposed.

The associate editor coordinating the review of this manuscript and approving it for publication was Cristina Rottondi.

In a single registration mode, the multi-server authentication scheme allows users to access services from multiple servers over the Internet. Typically, a multi-server authentication scheme consists of a user, a group of servers, and a trusted registration center (RC), which is responsible for registering users and servers. The registration center RC maybe participate in the user's login and authentication stage. Once the j-th user U_i is registered in the RC, U_i can access any server that has registered in the RC. Actually, in reality, multi-server environment often occurs in various situations. For example, in a hospital, every doctor almost needs to access different servers to complete job. There exist dozens of different general-purpose servers, such as accounting server, drug server, patient data server, and Web services server. Therefore, in recent two decades, the multi-server authentication scheme has been increasing becoming a research hotspot [1], [2], [5]–[7].

A. RELATED WORK

In 1981, Lamport [8] first proposed an insecure password-based authentication scheme. In the Lamport's scenario, the server needs to maintain a password table; therefore, an important piece of information can be cracked by a hacker. Later, many researchers published many improved password-based authentication schemes based on this problem [9]–[13]. Nonetheless, one obvious insufficient of these single-server authentication schemes is the registration issue. If a new user wants to use a large number of network services, they must register on those servers. It is very cumbersome for a user to register with the server, which not only wastes user time but also wastes server resources. Many researchers have proposed various multi-server authentication schemes based on the shortcomings of the single-server authentication scheme [1], [2], [5]–[7].

In 2001, Li *et al.* [14] first proposed a multi-server authentication scheme based on neural network. In Li's scenario, the server does not need to store any authentication tables, and any legitimate remote user can get services from multiple servers without having to register with each server separately. However, there is a deficiency in the scheme of Li, because it takes a long time to train the neural network based on the neural network, then it will require extremely high communication and computational costs. In 2003, Lin *et al.* [15] proposed an improved scheme based on the discrete logarithm problem. In 2006, Cao *et al.* [16] pointed out that Lin *et al.*'s program could not resist counterfeiting attacks.

In 2008, Tsai *et al.* [17] considered that the registration center and all servers are trusted. Tsai *et al.* proposed a smart card-based multi-server identity authentication scheme. In Tsai's scenario, the authentication scheme is based on a one-way hash function and does not require any validation tables to be stored in the registry and server. In 2012, Tsaur *et al.* [18] found that most of these previously proposed schemes used timestamps to defend against replay attacks, while replay attacks required the cost of clock synchronization. To overcome this problem, they proposed a self-validating timestamp method to avoid the difficulty of clock synchronization in a multi-server environment.

In 2013, Yoon *et al.* [19] proposed the first biometric-based multi-server environment authentication scheme. Their scheme uses elliptic curve cryptography (ECC) to ensure security. However, He *et al.* [20] pointed out that Yoon's scheme is weaker against impersonation attacks and privileged internal attacks, because once an adversary gets a password and a smart card, it can easily impersonate a valid user. He *et al.* designed a new robust solution to this weakness, a three-factor authentication solution in a multi-server environment. However, user anonymity in the He program is relatively weak and cannot withstand instant messaging attacks. In 2014, Chuang *et al.* [21] proposed a biometric-based authentication scheme based on smart cards and biometrics to provide user anonymity.

In 2016, Chatterjee *et al.* [22] used Chebyshev chaotic map to design a new biometric-based authentication protocol. Comparing Chatterjee's solution with the existing one, Chatterjee's solution has the advantages of small key, fast calculation and high efficiency. In addition, Barman *et al.* [23] proposed a multi-server environment authentication scheme based on biometrics. Their approach uses fuzzy extraction methods to provide an appropriate match of biometric patterns.

Password-based multi-server authentication schemes use passwords and cryptographic keys in remote user authentication. However, there are some problems with password-based methods, such as long, random passwords that cannot be used in this scenario because it is difficult for users to remember such long, random passwords; otherwise, passwords need to be stored somewhere. In addition, passwords may be forgotten, lost, or shared with others, and it is not possible to identify who the actual user is. In conclusion, a multi-server authentication scheme without passwords has been put forward by us.

Today, most existing biometric-based authentication schemes perform mutual authentication, whereas session key protocols do not consider the security of diverse biological templates in a multi-server environment. In addition, the above existing work does not consider the physical security of the smart card, which is very important for the protection of the smart card. Some existing literatures have discussed that physical unclonable functions (PUF functions) have been successful in some other areas [24], [25], such as some basic settings for safety meters, street lamps, medical systems, and so on. In 2012, Esbach *et al.* [26] proposed to install the PUF function security architecture on the smart card, which proved the feasibility of the smart card in our scheme.

In this paper, our goal is to design a new multi-server authentication protocol, using fuzzy commitment methods for biometric verification, and using PUF functions to ensure the uniqueness of smart cards. In proposed scheme, once the user U_i is registered in the RC, U_i can access any server that has registered with the RC, and the RC doesn't have to

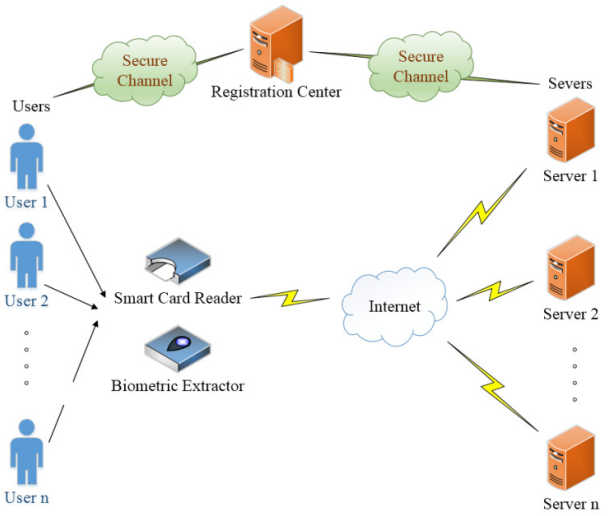

FIGURE 1. System model.

participate in the user's login and authentication phase. The Figure 1 shows the proposed system model in multi-server environment.

B. OUR CONTRIBUTIONS

A new biometrics and PUFs-based is designed for remote user authentication and session key protocol in multi-server environment. We summarize the main contributions of our scheme as follows:

- The biometrics and PUFs are used to ensure the uniqueness of the user and smart card respectively, which can ensure the physical security of proposed scheme.
- The biometrics key and auxiliary data are generated from user's biometrics template by using Fuzzy Extractor and stored in smart-card. Biometrics information are not stored in anywhere in the system, and avoid the risk of biometrics information loss. Discarded traditional password, in this case, it provides convenience for the user to use.
- Each server S_j and user U_i need to register with the trusted registration center RC. Users only need to register once in the RC to access all the servers registered in the RC. The RC doesn't have to participate in the user's login and authentication phase.

The remainder of this paper is organized as follows. Section II we first provide a brief introduction to one-way hash function, PUF and fuzzy extractor. In Section III, we present our scheme for multi-server authentication. Security of the proposed scheme is analyzed in Section IV. Finally, conclude our article with concluding remarks in Section V.

II. PRELIMINARIES

A. FUZZY EXTRACTOR

A hash function $h : A \rightarrow B$ is a deterministic mapping from a variable-length set $A = \{0, 1\}^*$ of documents (strings) to another set of fixed-length strings $B = \{0, 1\}^l$, called l-bits (called hash outputs or message digests). A one-way cryptographic hash function is a special hash function with the following properties:

1) For any input x ∈ A, it can be calculated in polynomial time or less time complexity and the output length is fixed. Furthermore, the hash function $h(.)$ is deterministic in nature, and the same input message outputs the same hash value under the action of the hash function.
2) Any change to the input $x \in A$ will cause the hash to be completely uncorrelated with $h(x)$, which seems to be random.
3) Preimage resistance: It is computationally difficult to implement information x from a hash value $h(x)$.
4) Weakly collision resistance: For any input $x \in A$, it is difficult to find an x' such that $h(x) = h(x')$.
5) Strong collision resistance: In a one-way hash function, collisions are defined as $h(x) = h(x')$ for any $x, x' \in A$ and $x \neq x'$. Strong collision resistance is difficult to find two x, $x' \in A$ such that $x \neq x'$ with $h(x) = h(x')$.

Definition: if $\mathrm{Adv}_A^{\mathrm{HASH}}(t)$ denotes the advantage of an adversary A in finding a hash collision in polynomial time t, then

$$Adv_A^{HASH}(t) = \Pr[ins_1, ins_2 \in_R .A : ins_1 \neq ins_2, h(ins_1) = h(ins_2)] \quad (1)$$

where, $\Pr[X]$ denotes the probability of a random event X, and $(\mathrm{ins}_1, \mathrm{ins}_2) \in_R$. A indicates that the input strings ins_1 and ins_2 and ins_1. An (ψ, t)-anversary A attacking the collision resistance of $h(.)$ means that the runtime of A is at most t, while it is like to satisfy the formula (2).

$$Adv_p^{\mathrm{HASH}}(t) \leq \psi. \quad (2)$$

B. PHYSICAL UNCLONABLE FUNCTION (PUF)

The PUF is characterized by a challenge-response pair (CRP). It is an integrated circuit (IC) that takes a string of bits as an input challenge and generates a series of bits called a response. The response R of the PUF to the challenge C can be expressed as: $R = PUF(C)$. PUF utilizes the uniqueness of the physical physics of the IC created during the manufacturing process to ensure that no two PUFs are identical. Since the PUF output depends on the physical characteristics of the IC, any attempt to tamper with the PUF will change its behavior and invalidate the PUF. Due to this unique feature, PUF has gained popularity as an important example of the physical security of resource-constrained devices. However, noise in the PUF output due to environmental conditions (eg., temperature) is still a limiting factor in PUF design and probably result in one or more output bits of the PUF being incorrect for approximate any input challenge. To solve this problem, the concept of a fuzzy extractor was introduced. A $(d, n, l, \varepsilon,)$-PUF needs to meet the following requirements to be called security:

1) For any two physical unclonable function $PUF_1(\cdot)$ and $PUF_2(\cdot)$, and $C_1 \in \{0, 1\}^K$ should satisfy the following

formula:

$$\Pr[HD(PUF_1(C_1), PUF_2(C_2)) > d] \geq 1 - \varepsilon. \quad (3)$$

Here, the term HD represents the Hamming distance.

2) For any physical unclonable function and any input $PUF_i(\cdot)$ and for any input $C_1, \ldots, C_n \in \{0,1\}^K$,

$$\Pr[HD(PUF_i(C_1), PUF_i(C_2)) > d] \geq 1 - \varepsilon. \quad (4)$$

3) For any two physical unclonable functions PUF_i (.) and $PUF_j(.)$, and for any input $C_1, \ldots, C_n \in \{0,1\}^K$, then

$$\Pr[H_\infty(PUF_i(C_k), PUF_j(C_l))_{1\leq j,k\leq n, i\neq j, k\neq l} > \lambda] \geq 1 - \lambda. \quad (5)$$

This condition indicates that different PUFs are evaluated using multiple inputs. While the internal distance i.e., the distance between two PUF responses from the same PUF instance and using the same challenge is smaller than d, the minimum entropy of the PUF is likely to be greater than λ [27]. The mutual distance i.e., the distance between two PUF responses with different PUF instances based on the same input challenge, is greater than d.

C. ENCRYPTED ONE-WAY HASH FUNCTION

As known to all, fuzzy extractor A (d, λ, ϵ) is consisted of two parts, one is FE,Gen [28], [24], it is a probabilistic key generation approach. Specially, a bit character R as an input, a key K and auxiliary data hd as two outputs, i.e., $(K, hd) =$ Fe.Gen(R. Furthermore, the other is FE.Rec method, in fact, it is a deterministic reconstruction strategy, the key K from the noisy input variable R' and the auxiliary data hd, are effectively recovered, $K = FE.Rec(R', hd)$.What is more, sometimes, while the Hamming distance between R' and R is at most d. A fuzzy extractor (FE) ensures security in the extraction of a strong cryptographic key if the min-entropy of input R is at least, λ and K is statistically ϵ-close to an uniformly distributed random variable in $\{0,1\}^K$. In practice, fuzzy extractor $A(d, \lambda, \epsilon)$ is said to be secure if the following condition holds:

1) $$\Pr[K = FE.\text{Rec}(R', hd) \leftarrow FE.Gen(R), HD(R, R') \leq d] = 1 \quad (6)$$

where, the term HD is the Hamming distance.

2) If the min-entropy $H_\infty(R) \geq \lambda$, then $(K, hd) \leftarrow FE.Gen(R)$ is statistically ϵ-close to $\left(K', hd\right)$. Where, $K' \leftarrow \{0,1\}^{|K|}$.

III. PROPOSED SCHEME

In this section, we will present our proposed remote multi-server authentication and key agreement scheme using biometrics and PUFs. In particular, the scheme mainly includes: server registration, user registration, login, mutual authentication and key agreement.

- In the registration phase, $\forall S_j$ needs to be registered in RC; then, $\forall U_i$ registers in RC.
- During the login phase, any registered user u only needs to enter the identity ID_u and the biometric information BIO_u, so that the protocol is initiated to authenticate the smart card SC_i.
- In the authentication and key exchange phase, mutual authentication is performed between the authorized registered user U_i and the registration server S_j, and a session key SK_{ij} is established between U_i and S_j.

Especially, the symbols used in the protocol are given in Table 1.

TABLE 1. Notations used in this paper.

Symbol	Description
S_j	j^{th} server
U_i	i^{th} user
ID_u	Identity of user U_i
BIO_u	Biometric data of user U_i
SID_j	Identity of server S_j
RC	Trusted registration center
K,b	RC's master key, RC's random key
SK_{ij}	The common session key between user U_i and server S_j
$h(.)$	A one-way hash function
N_1	Random nonce generated by U_i
N_2	Random nonce generated by S_j
$\oplus$	Bitwise exclusive-or (XOR) operator
$\|\|$	String concatenation operator

A. SERVER REGISTRATION PHASE

In the proposed solution, $\forall S_j$, $1 \leq$ j $\leq$ m, (m is the total number of servers available in the original network), needs to be registered in the trusted registry RC. Therefore, if S_j is willing to become an authorization server and provide services to registered users, it generally sends a registration request, including a unique identity SID_j. The RC sends two secret keys K_1 and K_2 to each S_j via the Internet Key Exchange Protocol (IKEv2) [23]. Note that K_2 is unique to each server S_j and it is used in the mutual authentication process of user U_i and server S_j. In figure 2, the process of server registration is concretely depicted. Additionally, the specific steps are listed as follows:

1) During initialization process, a master secret key K, a random secret b are selected by RC.
2) S_j submits its identity SID_j towards RC.
3) The validity of SID_j is checked. If invalid, the server SID_j returns existing information, and then submits a new SID_j. Subsequently, the two keys are RC computed as $K_1 = h(K||\, b)$ and $K_2 = h(SID_j||h(b))$. Moreover, both keys (K_1 and K_2) are sent to S_j employing a confidential channel. In this manner, S_j is successfully registered through RC.

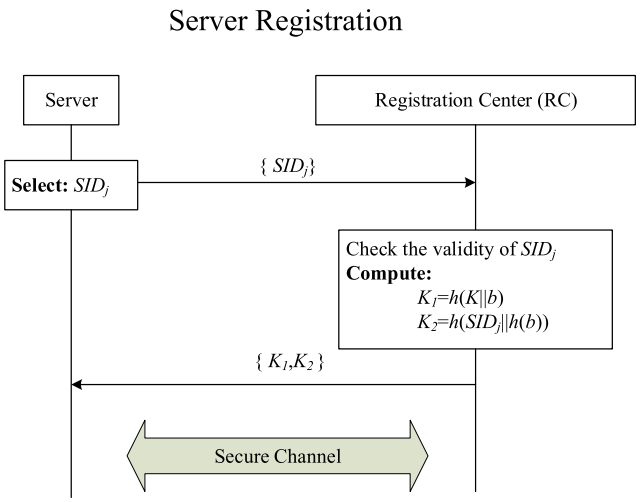

FIGURE 2. Server registration process.

B. USER REGISTRATION PHASE

At the beginning, $\forall U_i$ needs to register in the RC through a secure channel. At this stage, U_i needs to select a user identity ID_u and a random number C_u. Besides, U_i also provides his/her biometric data to the biosensor, which captures the biometric data BIO_u. In fact, U_i provides unique biological keys by using fuzzy-extracted FE.Gen algorithm, at some time, equally unique R_u is gained by using physical non-cloning function (PUF). After the RC accepted the user registration information, the private key of the RC will be stored in the smart card in an encrypted manner, and then the smart card is sent to the user. More specially, figure 3 summarizes the user registration process, the specific steps are shown as follows:

1) U_i gives biometric key K_u and auxiliary information hd using FE.Gen algorithm according to its biometric data BIO_u, that is, $(K_u,hd) = Fe.Gen(BIO_u)$. Next, the U_i achieves R_u under the action of PUF. Then U_i sends the registration information$\{ID_u, <C_u,R_u>,K_u\}$ to the RC.
2) After receiving the registration information sent by the i-th user U_i, the RC checks the validity of the user ID_u, if the u-th user's ID_u is invalid, RC returns that the user information ID_u has been registered and the new ID_u is selected for registration. Subsequently, the below operations are conduced: $V_i = h(b)\oplus h(ID_u||K_u||C_u)$, $Z_i = K_u\oplus h(ID_u||R_u)\oplus h(b)\oplus h(K||b), X_i = h(ID_u||h(K||b)||K_u)$, $Y_i = X_i \oplus h(K_u||h(b)||R_u)$. Clearly, the RC stores$\{ID_u, <C_u,R_u>\}$ and a smart card SC_i, i.e., the information$\{V_i,Z_i,Y_i\}$ saved into the card. Finally, RC sends SC_i to user.
3) After receiving the information sent by the RC, U_i computes $UC_u = C_u \oplus K_u$ and $A_u = h(ID_u||R_u||K_u)$. Finally, U_i put information $\{UC_u, hd, A_u\}$ into the SC_i, and embed the integrated circuit of PUF into the SC_i.

C. USER LOGIN PHASE

At this stage, the registered user U_i inserts the smart card SC_i into the card reader of the specific terminal and provides its identity ID_u. additionally, U_i also scans the biometrics at the biosensor for authentication. Specific steps are shown as follows:

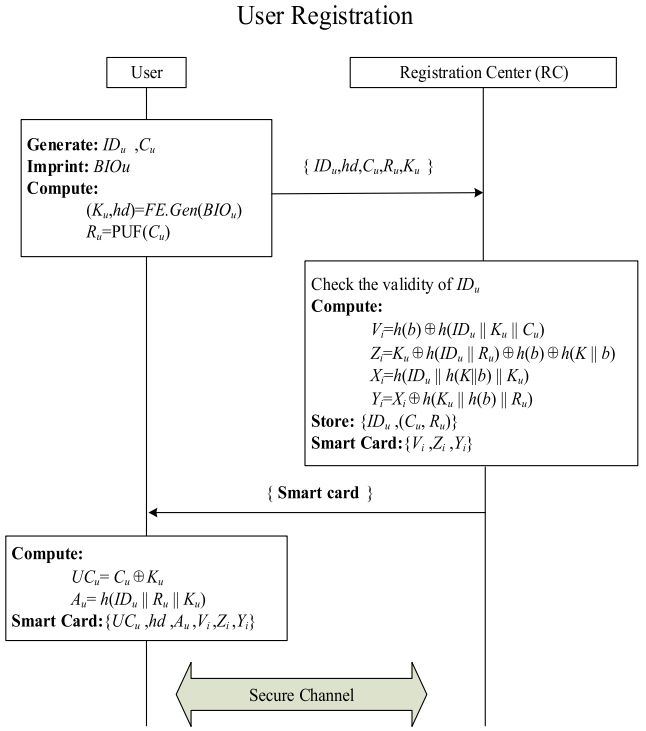

FIGURE 3. User registration process.

1) U_i scans his/her biometrics, and extracts feature BIO_u from the captured fingerprint image.
2) U_i inserts the smart card SC_i into the carder reader and enters the credential ID_u.
3) U_i generates K_u' as $K_u' = FE.Rec(BIO_u,hd)$, and extracts Cu', R_u' according to the forms $C_u' = UC_u \oplus K_u'$ and $R_u' = PUF(C_u')$. Besides, SC_i then compares the computed $h(ID_u||R_u'||K_u')$ with the stored A_u. If they are not equal, the session is terminated
4) After the completion of check the U_i, U_i obtain K, SI according to the forms $PK = V_i\oplus h(ID_u||K_u'||C_u')$ and $SI = h(SID_j||PK)$. U_i selects a random nonce N_1 and uses N_1 to encrypt to get encrypted information $A_{ij} = Z_i\oplus PK\oplus h(SI||N_1)\oplus h(ID_u||R_u')$, $M_1 = SI\oplus N_1$. Subsequently, U_i encrypts messages $SPK = PK\oplus h(N_1||ID_u)$, $SID_u = ID_u \oplus h(N_1)$, $SC_u = C_u'\oplus h(ID_u||PK)$, $SR = R_u'\oplus h(N_1||K_u')$, $X_i' = Y_i\oplus h(K_u'||PK||R_u')$. Finally, U_i gets an authentication message $A_1 = h(X_i'||ID_u||K_u'||N_1||SI)$ and sends login message$\{M_1, SID_u, SC_u, A_{ij},SPK, A_1\}$ to server S_j.

D. MUTUAL AUTHENTICATION PHASE

After the successfu1 login of a registered user U_i, the authentication of a server S_j is verified. After successful mutual authentication, the session key is established between U_i and S_j. The login and mutual authentication phases are briefly described in figure 4. The detailed steps are given below.

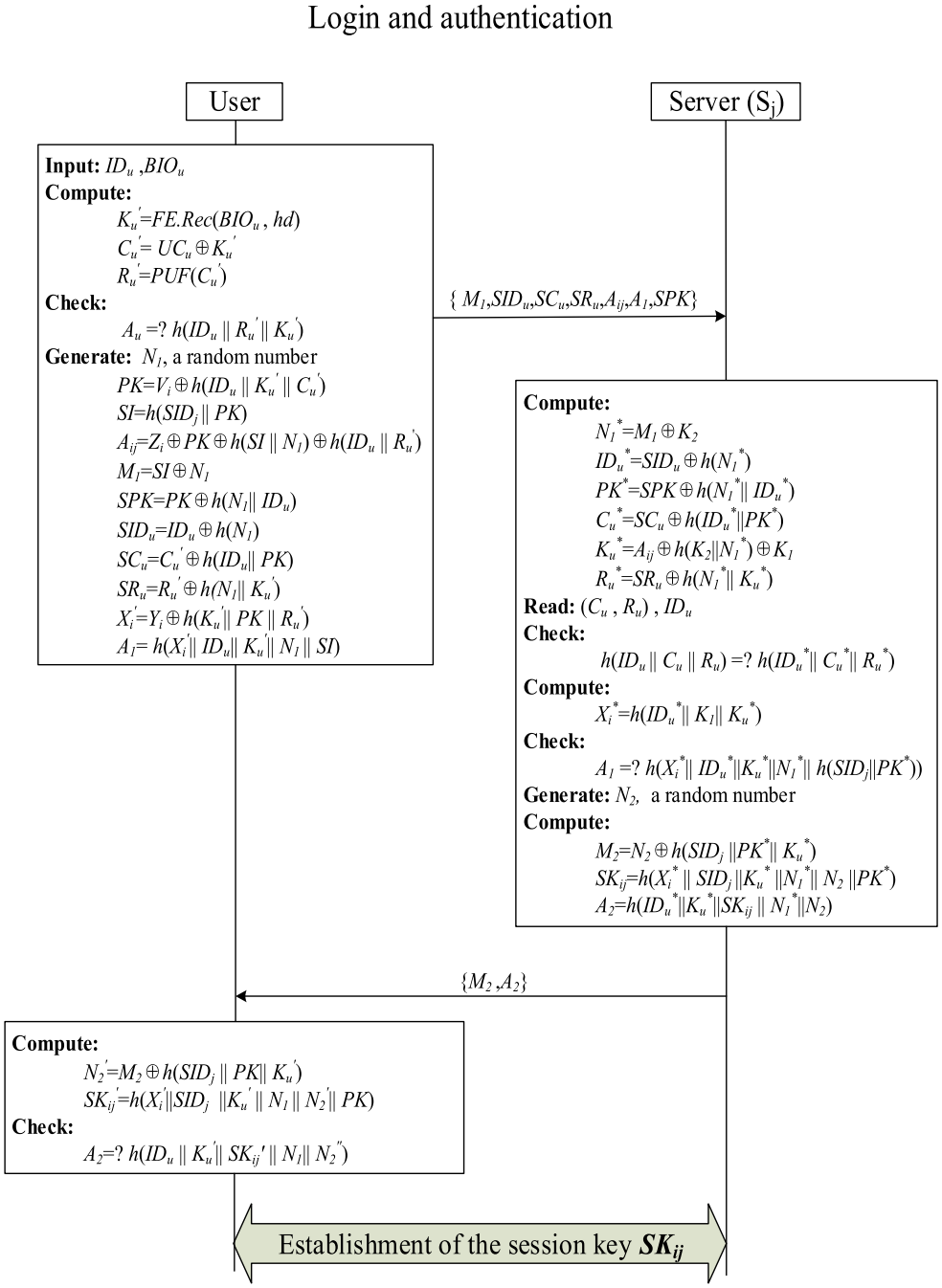

FIGURE 4. Login and mutual authentication process.

1) S_j receives the login message and decrypt messages $N_1^* = M_1 \oplus K_2$, $ID_u^* = SID \oplus h(N_1^*)$, $PK^* = SPK \oplus h(N_1^* || ID_u^*)$, $C_u^* = SC_u \oplus h(ID_u^* || PK^*)$, $K_u^* = A_{ij} \oplus h(K_u || N_1^*) \oplus K_1$. S_j reads the record$\{ID_u, <C_u, R_u>\}$ from the database and checks $h(ID_u||C_u||R_u) = ?h(ID_u^*||C_u^*||R_u^*)$, if they are not equal, the session is terminated.
2) In order to complete user verification, S_j have to obtain X_i^* as $X_i^* = h(ID_u^*||K_1||K_u^*)$. Then S_j computes $h(X_i^*||ID_u^*||K_u^*||N_1^*||h(SID_j||PK^*))$, and compares it with the login message A_1. If they are not equal, the session is terminated.
3) Then, S_j generates a nonce N_2. Next, S_j achieve M_2 as $M_2 = N_2 \oplus h(SID_j||PK^*||K_u^*)$ and generates a session key $SK_{ij} = h(X_i^*|| SID_j||K_u^*||N_1^*||N_2||PK^*)$. Finally, S_j generates an authentication message $A_2 = h(ID_u^*||K_u^*||SK_{ij}||N_1^*||N_2)$ and sends authentication request message $\{M_2, A_2\}$ to U_i.
4) The U_i receives the authentication request message $\{M_2, A_2\}$ and computes $N_2' = M_2 \oplus h(SID_j||PK\ ||K_u')$, $SK_{ij}' = h(X_i'||SID_j||K_u'||N_1||N_2'||PK)$. Following, SC_i compares the computed $h(ID_u||K_u'||SK_{ij}'||N_1||N_2')$ with the authe- entication message A_2. If they are not equal,

the session is terminated. Otherwise, the session key SK_{ij} is established for secure message communication between U_i and S_j.

IV. SECURITY ANALYSIS

A. FORMAL SECURITY USING THE ROR MODEL

We use the Real-Or-Random (ROR) model proposed by Abdalla *et al.* [29] to demonstrate the safety of the protocol. In the case of passive/active attacks, the ROR model can still provide session key SK security. Recently, formal security analysis based on the ROR model has been popularized, and the analysis method is applied to various authentication key exchange protocols [22], [30], [31].

1) ROR MODEL

In our proposed solution, there are three participants, one user U_i, one server S_j and one registry RC.

Participants: $\pi_{U_i}^u$, $\pi_{S_j}^t$ and π_{RC}^v are denoted as the instance u, t and v of U_i, S_j and RC, respectively.

Partnering: The instance $\pi_{U_i}^u$ of U_i has instance $\pi_{S_j}^t$ of S_j as its partner and conversely. $\pi_{S_j}^t$ is called the partner ID $pid_{U_i}^u$ of $\pi_{U_i}^u$. The partial transcript of the messages exchanged between U_i & S_j is unique, and is known as session ID $sid_{U_i}^u$ for the ongoing session in which $\pi_{U_i}^u$ takes part.

Freshness: If the session key SK_{ij} established between U_i and S_j is not leaked via the reveal oracle *Reveal* defined below, we call $\pi_{U_i}^u$ or $\pi_{S_j}^t$ fresh.

Adversary: Under the ROR model, attacker A uses the widely accepted Dolev-Yao (DY) threat model to intercept, modify, delete, and even inject some or all of the exchange information between U_i and S_j. Some operations of A are given as follows:

- Execute(π^t, π^u): This query is executed by A to obtain exchanged message between U_i and S_j. This query implement an active attack.
- Reveal(π^t): Using this query, A can know the session key SK_{ij} which is generated by π^t and its partner in the current session.
- Send(π^t, m): This query implements an active attack wherein A can send a message m to a participate instance π^t, and in reply, it receives a response from π^t.
- CorruptSmartCard($\pi_{U_i}^u$):This query is about SC_i modeling loss/stolen attack. A can extract all the sensitive secret information stored in its memory via power analysis attack.
- Test(π^t): Based on the indistinguishability of the model, the semantic security model of SK_{ij} is established between U_i and S_j. In this query, an unbiased coin c is flipped in the beginning of the game, and its output is used as a decider. The outcome is kept secret to A to check the output from the *Test* query. Let A execute this query. If the session key SK_{ij} shared between U_i and S_j is fresh, π^t returns SK_{ij} when $c = 1$or a random number when c = 0. Otherwise, it returns null.

a: SEMANTIC SECURITY OF THE SESSION KEY

In the ROR model, attacker A was tested in the experiment to distinguish between the real session key SK_{ij} and the instance's random key. Therefore, A is allowed to query a large number of *Test* operations to the sensor node instance or user instance. The output of the *Test* operation should match the random bit c. Ultimately, attacker A will output a guess bit c', if $c = c'$, then attacker A successfully obtains the correct information in the experiment. Suppose *Succ* indicates that A succeeded in the experiment. At a polynomial time t, the advantage of attacker A is to break the security of the proposed session key (SK), called P, defined as $Adv_p^A(t) = |2^*PR[Succ]-1| = |2^*Pr[c = c']-1|$, where $Pr[X]$ represents the likelihood of event X.

b: RANDOM ORACLE

Both attacker A and each participant are provided with a one-way hash function h(·), which is modeled as a random oracle, say *Hash* [31]. The *Hash* oracle is simulated by a two-tuple *(a, b)* table of binary strings. In this case, if a hash query $h(a)$ is made, the *Hash* oracle returns b when a is present in the table; otherwise, it returns a uniform random string b and the pair (a, b) is kept safe in the corresponding table [32].

2) SECURITY PROOF

Under the ROR model, the formal proof of the session key security of the system is as follows:

Theorem: Let $Adv_v^A(t)$ be polynomial-time t-adversary A's advantage function in breaking the SK security of the proposed scheme P:

$$Adv_p^A(t) \leq \frac{q_h^2}{|Hash|} + \frac{q_s}{2^{l-1} \cdot |D|} \tag{7}$$

where q_h, q_s, l, | *Hash* | and |D| are the he number of H queries, the number of *Send* queries, the number of bits in the biometric key, the range space of the hash function h(·) and the size of a uniformly distributed random dictionary D, respectively.

Proof: Proof of the formal security key is as follows, very similar to what has appeared in the literature [33], [31].

We need the next four game stages $Gm_j(j = 1, 2, 3, 4)$. We use $Succ_{Gm_j}^A$ indication that the attacker can win Gm_j.

- ***Game* Gm_0:** In the initial game Gm_0, the bit c is chosen by a polynomial-time t adversary A. Since the Gm_0, and the actual protocol in the ROR are basically identical, it follows that

$$Adv_p^A(t) = \left|2 \cdot Adv_{Gm_0}^A - 1\right| \tag{8}$$

- ***Game* Gm_1:** A invokes the *Execute* query in the game to implement the eavesdropping function. Then, A calls the *Test* query after the game is completed. The output of the *Test* operation is used as a deciding factor for distinguishing the actual session key SK_{ij} between U_i and S_j with the random number in the session. The session key formation is as follows. S_j computes the session key

$SK_{ij} = h(X_i^*||SID_j||K_u^*||N_1^*||N_2||PK^*)$ shared with U_i, and the same session key computed by U_i, is shared with S_j as $SK'_{ij} = h(X_i'||SID_j||K'_u||N_1||N'_2||PK)$. Suppose A is able to use some manipulation to get intercept message $Msg1 = \{M_1, SID_u, SC_u, SR_u, A_{ij}, SPK, A_1\}$ and $Msg2 = \{M_2, A_2\}$. The session key computation by A needs the long-term secrets ID_u, RC's master key K and b. A also the short-term secrets N_1 and N_2. Without these secret credentials, the chance of winning game Gm_1 by intercepting messages $Msg1$ and $Msg2$ is not increased. Since both games Gm_0 and Gm_1 are essentially indistinguishable, we have the following:

$$Adv^A_{Gm1} = Adv^A_{Gm_0} \tag{9}$$

- ***Game Gm₂*:** *Send* operations and *Hash* queries are used in this partial game. The simulation of this part of the game is similar to the active attack, by intercepting $Msg1 = \{M_1, SID_u, SC_u, SR_u, A_{ij}, SPK, A_1\}$ and $Msg2 = \{M_2, A_2\}$, then A tries to crack the session key between U_i and S_j. $Msg1$ and $Msg2$ relate to random numbers N_1 and N_2. Hence, there is no collision in hash outputs when A makes *Hash* queries on these intercepted messages (see Definition). Therefore, due to the collision resistance of the one-way cryptographic hash function h, the calculation of ID_u, RC's master key K, b, Biological key K_u, and short-term keys N_1 and N_2 is computationally infeasible. Since game Gm_2 is identical to game Gm_1 when the simulation of *Send* and *Hash* queries is not involved, the results from the birthday paradox give the following result:

$$\left|Adv^A_{Gm_2} - Adv^A_{Gm_1}\right| \le \frac{q_h^2}{2 \cdot |Hash|} \tag{10}$$

- ***Game Gm₃*:** In the game Gm_3, the *CorruptSmartCard* operation is used. Therefore, A has the secret credentials $\{UC_u, hd, A_u, V_i, Z_i, Y_i\}$ from U_i 's smart card SC_i's memory, where $UC_u = C_u \oplus K_u$, $A_u = h(ID_u||R_u||K_u)$. Without the secret credentials C_u, R_u,and biometric secret key K_u, it is computationally infeasible to derive the UC_u and A_u. Assuming UC_u is l bits, the guessing probability of $UC_u \in \{0,1\}^l$ by A is approximately $1/2^l$ [34]. Note that games Gm_2 and Gm_3 are identical when password and biometrics guessing attacks are not involved. Hence, we have the following result:

$$\left|Adv^A_{Gm_3} - Adv^A_{Gm_2}\right| \le \frac{q_s}{2^{l-1} \cdot |D|} \tag{11}$$

Since all games are executed, attacker A can only guess the correct bit c. Then come to the following conclusion:

$$Adv^A_{Gm_3} = \frac{1}{2} \tag{12}$$

According to formula (8), formula (9) and formula (12), we can get the following conclusions:

$$\begin{aligned} \frac{1}{2} \cdot Adv^A_p(t) &= \left|Adv^A_{Gm_0} - \frac{1}{2}\right| \\ &= \left|Adv^A_{Gm_1} - \frac{1}{2}\right| \\ &= \left|Adv^A_{Gm_1} - Adv^A_{Gm_3}\right| \end{aligned} \tag{13}$$

The following results are obtained by triangular inequality:

$$\begin{aligned} &\left|Adv^A_{Gm_1} - Adv^A_{Gm_3}\right| \\ &\le \left|Adv^A_{Gm_1} - Adv^A_{Gm_2}\right| + \left|Adv^A_{Gm_2} - Adv^A_{Gm_3}\right| \\ &\le \frac{q_h^2}{2 \cdot |Hash|} + \frac{q_s}{2^l \cdot |D|} \end{aligned} \tag{14}$$

The formula (13) and the formula (14) are combined to obtain:

$$\frac{1}{2} Adv^A_p(t) \le \frac{q_h^2}{2 \cdot |Hash|} + \frac{q_s}{2^l \cdot |D|} \tag{15}$$

Finally, multiply both sides of equation (15) by 2 and simplify to get the desired result:

$$Adv^A_p(t) \le \frac{q_h^2}{|Hash|} + \frac{q_s}{2^{l-1} \cdot |D|} \tag{16}$$

B. MUTUAL AUTHENTICATION USING BAN LOGIC

We use a formal analysis of Burrows-Abadi-Needham (BAN) logic [35] to demonstrate that in our proposed protocol, the interaction verification between user U_i and server S_j is safe. BAN logic has been widely used in interactive authentication, mainly to provide interactive authentication for authentication and session key protocols [2], [23].

The basic building blocks of BAN logic:

$A| \equiv X$: A believes in a statement X.

$\#X$: denotes freshness of X

$A \triangleleft X$: A sees X.

$A| \sim X$: A once said statement X.

$A \Longrightarrow X$: A has jurisdiction over X

$A \overset{K}{\longleftrightarrow} B$: K is used by A and B to communicate with each other.

$\{X, Y\}_K$: X and Y are encrypted with key K.

$(X, Y)_K$: X and Y are hashed with key K.

$< X >_K$: X is combined with key K.

The main rules of BAN logic are given below:

1) Message-meaning rule(R1):

$$\frac{A| \equiv A \overset{K}{\longleftrightarrow} B, A \triangleleft \{X\}_K}{A| \equiv B| \sim X}$$

2) Nonce-verification rule(R2):

$$\frac{A| \equiv \#(X), A| \equiv B| \sim (X)}{A| \equiv B| \equiv X}$$

3) Jurisdiction rule(R3):

$$\frac{A| \equiv B \Rightarrow X, A| \equiv B| \equiv X}{A| \equiv X}$$

4) Fresh rule(R4):

$$\frac{A| \equiv \#(X)}{A| \equiv \#(X, Y)}$$

5) Belief rule(R5):

$$\frac{A| \equiv (X), A| \equiv (Y)}{A| \equiv (X, Y)}$$

6) Session key rules(R6):

$$\frac{A| \equiv \#(X), A| \equiv B| \equiv X}{A| \equiv A \overset{K}{\longleftrightarrow} B}$$

According to the analysis process of BAN logic, our proposed protocol needs to meet the following two objectives:

G1: $U_i| \equiv U_i \overset{SK}{\longleftrightarrow} S_j$; G2: $S_j| \equiv U_i \overset{SK}{\longleftrightarrow} S_j$.

We first list the assumptions related to the proposed scheme:

A1: $U_i| \equiv \#(N_1)$ A2: $S_j| \equiv \#(N_2)$
A3: $U_i| \equiv S_j \Longrightarrow N_2$ A4: $S_j| \equiv U_i \Longrightarrow N_1$
A5: $U_i| \equiv U_i \overset{h(K||b)}{\longleftrightarrow} S_j$ A6: $S_j| \equiv U_i \overset{h(K||b)}{\longleftrightarrow} S_j$
A7: $U_i| \equiv U_i \overset{SK}{\longleftrightarrow} S_j$ A8: $S_j| \equiv U_i \overset{SK}{\longleftrightarrow} S_j$

Idealized forms of messages: In the proposed scheme, messages $Msg1 = \{M_1, SID_u, SC_u, SR_u, A_{ij}, SPK, A_1\}$ and $Msg2 = \{M_2, A_2\}$ can be written in their respective idealized forms as follows:

- $Msg1 : S_j \triangleleft < M_1, SID_u, SC_u, SR_u, A_{ij}, SPK, A_1 >$, that is $Msg1 : S_j \triangleleft < SI \oplus N_1, ID_u \oplus h(N_1), C'_u \oplus h(ID_u||PK), R'_u \oplus h(N_1||K'_u), K_u \oplus h(K||b) \oplus h(SI||N_1), h\left(X'_i||ID_u||K'_u||N_1||SI\right), PK \oplus h(N_1||ID_u) >_{h(K||b)}$
- $Msg2 : S_j \rightarrow U_i < M_2, A_2 >$, that is $Msg2 :< N_2 \oplus h\left(SID_j||PK^*||K^*_u\right), h\left(ID^*_u||K^*_u||SK_{ij}||N^*_1||N_2\right) >_{SK_{ij}}$.

The main security proof consists of the following steps:

1) Consider the message $Msg1$, Under the premise of assuming A6, we can use the message meaning rule R1 to obtain:

$$S1 : S_j| \equiv U_i| \sim N_1$$

2) At the conclusion of S1, the assuming A1 and nonce-verification rule R2 can be obtained:

$$S2 : S_j| \equiv U_i| \equiv N_1$$

3) Under the conclusion of S2, using hypothesis A4 and jurisdictional rule R3, we can get:

$$S3 : S_j| \equiv N_1$$

4) Server S_j believes that N_2 is fresh (available from assuming A2). N_1, N_2 are the two necessary parameters that make up the key $SK_{ij} = h(X^*_i||SID_j||K^*_u||N^*_1||N_2||PK^*)$. So using the session key rule R6 we can get:

$$S4 : S_j| \equiv U_i \overset{SK_{ij}}{\longleftrightarrow} S_j$$

5) Next, Consider the message $Msg2$, we can get:

$$S5 : U_i \triangleleft \langle N_2 \rangle US_j$$

6) Under the premise of S5, using assuming A7 and message meaning rule R1, we can infer:

$$S6 : U_i| \equiv S_j| \sim N_2$$

7) On the basis of S6, using the nonce-verification rule R2 and the hypothesis A2, we can obtain:

$$S7 : U_i| \equiv S_j| \equiv N_2$$

8) Then at S7, assume A3 and the governing rule R3 can be launched:

$$S8 : U_i| \equiv N_2$$

9) U_i believes that N_1 is fresh (as can be seen from hypothesis A1), so the key with the combination of N_1 and N_2 also has this property. Therefore, based on the session key rule R6, the assumptions A1 and S8, we can get:

$$S9 : U_i| \equiv U_i \overset{SK_{ij}}{\longleftrightarrow} S_j$$

It can be seen from the above proof that the defined targets G1 and G2 are implemented in the proposed scheme. Therefore, the scheme maintains a secure interactive authentication between U_i and S_j.

C. INFORM SECURITY ANALYSIS

1) *Protection Against Replay Attack*: In the proposed scenario, we use a random number that is more reliable than the timestamp to prevent replay attacks. The attacker cannot replay the message in the proposed scheme because each transmitted message contains a random number and the system will end directly if the random number is found to be inconsistent. In addition, the attacker cannot construct a new message because a valid message contains the biometric key K_u information, and since the user's biometric key K_u is secure, the replay attack will not work.
2) *Ensures Session Key Freshness Property*: In our proposed scheme, each session key contains a random number, and each random number is unique for each session. The unique key structure of each session ensures the freshness of the key.
3) *Protection User Anonymity*: In our scheme, user's *ID* anonymity is preserved at each login request. We compute an anonymous identity $SID_u = ID_u \oplus h(N_1)$ for U_i and this *ID* will be different at each login attempt because it is calculated with the random number N_1. Therefore, if you want to get ID_u, you have to get a random number N_1. But it is always very difficult, for the random number, it is usually hard to guess [39]. Moreover, it is extremely difficult to get the user ID_u in the next pass. In particular, the information including several random numbers and the Biological key K_u, is always wrapped in a hash function. Typically, the random number of each session is obviously different, it clearly leads to decipher the user ID_u more difficult. Therefore, our scheme protects the user's anonymity.
4) *Mutual Authentication*: In our proposed strategy, only the biometric BIO_u of the legitimate user can obtain an correct and unique bio-key $K_u^{'}$, i.e., $K'_u = FE.Rec(BIO_u, hd)$. Obvi- ously, $K_u^{'}$ is obtained based on the fuzzy extraction function. After obtaining the bio-key, you also need to get C_u (the random number selected during registration) and the same unique R_u

through the non-clonal function to verify the user's smart card. In the next step, the server obtains C_u and R_u by decryption, and then the server reads the information for user authentication. While the pipeline that server verifies relevant users, has accomplished, the following process is that the user's verification phase to the server. During validation, the user needs to verify the private key of RC to determine whether or not the server is correctly registered. Ideally, the user and server generate the session key after authentication. Therefore, the proposed scheme can provide mutual verification.

5) *Resist Stolen Smart Card Attack*: An attacker can obtain information$\{UC_u, hd, A_u, V_i, Z_i, Y_i\}$ stored on smart card. An attacker needs a valid user ID_u, key-value pair (C_u ,K_u) and corresponding biological key K_u to generate a valid login information. User ID_u and key-value pair (C_u ,K_u) are not stored directly on the smart card, user ID_u and key-value pair (C_u ,K_u) are hard to guess, so the login information is secure. The calculation of valid biological key K_u needs fuzzy extraction. Without correct biological information, it is impossible to generate valid biological key K_u, and the biological key is unique. Therefore, it is never possible that the biological key K_u has been effectively guessed. Since the biological key K cannot be guessed and the server's private key is not public, the login information is hardly computed. Hence, the proposed scheme can resist the attack of stolen smart card.
6) *Man-In-The-Middle Attack*: Attacker A attempts to modify related intercepted communication messages $Msg1 = \{M_1, SID_u, SC_u, SR_u, A_{ij}, SPK, A_1\}$ and $Msg2 = \{M_2, A_2\}$. Suppose A tries to modify $Msg1$, using a new random number N_1^a to make it a new valid information $Msg1` = \{M_1', SID_u', SC_u', SR_u', A_{ij}', SPK', A_1'\}$. Attacker A begins to calculate Msg_1' content according to user login phase. The operations are conduced: $M_1' = SI \oplus N_1^a$, $SID_u' = ID \oplus h(N_1^a)$, $SC_u' = C_u' \oplus h(ID_u||PK)$, $SR_u' = R_u' \oplus h(N_1^a||K_u')$, $A_{ij}' = Z_i \oplus PK \oplus h(SI||N_1^a) \oplus h(ID_u||R_u')$, $SPK' = PK \oplus h(N_1^a||ID_u)$, $A_1' = h(X_i'||ID_u||K_u'||N_1^a||SI)$. Clearly, it is able to see that A needs some secret information ID_u, key-value pair (C_u,K_u) and biological key K_u. Without this information, it is difficult to get a new valid one. Similarly, it is also difficult for attacker A to modify the intercepted communication message $Msg2$ and make it become a new effective message. Obviously, the proposed scheme can resist the man-in-the-middle attack.
7) *Impersonation Attacks*:

- User Impersonation Attack: To convince server S_j with the information came from a legitimate user U_i, an attacker A have to generate a new random nonce N_1^*. In the next moment, A attempt to calculate login request message $\{M_1, SID_u, SC_u, A_{ij}, SPK, A_1\}$ based on user login phase. The information calculated from user login phase is as follows: $M_1 = SI \oplus N_1^*$, $SID_u = ID_u \oplus h(N_1^*)$, $SC_u = C_u' \oplus h(ID_u||PK)$, $SR = R_u' \oplus h(N_1^*||K_u')$, $A_{ij} = Z_i \oplus PK \oplus h(SI||N_1^*) \oplus h(ID_u||R_u')$, $SPK = PK \oplus h(N_1^*||ID_u)$, $A_1 = h(X_i'||ID_u||K_u'||N_1^*||SI)$. Whereas, such attempt by A often is failure, while the secret credentials ID_u, key-value pair (C_u,K_u) and biological key K_u are unknown to A. In this case, the proposed scheme can resist user simulation attack.
- Server Impersonation Attack: In this attack, attacker A needs to convince the user U_i that the information is coming from a valid server S_j., initially, A generates a random number N_2^*, and then computes the verification information$\{M_2, A_2\}$. However, without short-term key N_1, user ID_u and server key K_1, A is difficult to form an effective verification information. To some extent, the proposed scheme can also resist server simulation attack.

V. PERFORMANCE ANALYSIS AND COMPARISON

To show the advantage of our proposed scheme, now we first compare the proposed scheme with four recently proposed multi-server authentication key protocol schemes. From Table 2, we can see that, the proposed scheme is secure against all the imperative security threats and accomplishes diverse features. We focus on the security against replay attack and anonymity, stolen smart card attack and Man-in-the-middle attack, user impersonation attack, cloud server impersonation attack, mutual authentication and session key freshness and protection smart card physical security. We note that none of these past schemes including Kumari *et al.* [5], Feng *et al.* [36], Sood *et al.* [37] and Shen *et al.* [38], fulfill all the essential security properties in contrast to our scheme which achieves all the security properties simultaneously.

The scheme presented by Barman *et al.* cannot ensure mutual authentication and session key freshness and protection smart card physical security. Feng *et al.*'s schemes suffer from stolen smart card attack and Man-in-the-middle attack. The scheme proposed by Sood *et al.* cannot prevent impersonation attack. Shen *et al.*'s scheme cannot satisfy both user and cloud server identity protection (anonymity) and mutual authentication. Shen *et al.*'s schemes require the support of RC to achieve the mutual authentication and does not provides the owner confirmation method in smart card. It is worth noting that none of the existing schemes are completely protection smart card physical security. However, our proposed protocol is able to protect smart card physical security.

Next, we compare our scheme with the existing multi-server schemes with respect to the computation cost of login and authentication phases. We evaluated the performance of our improved scheme and compared it with four recently proposed schemes in the literature, i.e., Barman *et al* [5], Feng *et al* [36], Sood *et al* [37], and Shen *et al* [38]. We apply hash function, PUF, fuzzy extractor and elliptic scalar point multiplication to determine the computational overhead for each authentication schemes. The comparison results are shown in Table 3. The following notation is used to represent the computation cost:

TABLE 2. A comparative summary: Security features.

Security properties	Our	Barman et al.	Feng et al.	Sood et al.	Shen et al.
Protection against replay attack	Yes	Yes	Yes	No	Yes
Protection user anonymity	Yes	Yes	Yes	Yes	No
Mutual authentication and session key freshness	Yes	No	Yes	No	No
Resist stolen smart card attack	Yes	Yes	No	Yes	No
Man-in-the-middle attack	Yes	Yes	No	Yes	Yes
Impersonation attack	Yes	Yes	Yes	No	Yes
Protection smart card physical security	Yes	No	No	No	No

TABLE 3. Comparison of computational cost(milliseconds).

Scheme	Login Phase	Verification Phase	Total Cost	Rough Estimation(in millisecond)
Barman et al.	$6C_h+C_{fcs}$	$11C_h$	$17C_h+C_{fcs}$	2.2651
Feng et al.	$3C_{ecm}+7C_h$	$5C_{ecm}+17C_h$	$24\ C_h+8C_{ecm}$	17.8632
Sood et al.	$7C_h$	$24C_h$	$31C_h$	0.0713
Shen et al.	$5C_h+3C_{ecm}$	$12C_h+3C_{ecm}$	$17C_h+6C_{ecm}$	13.3951
Our	$10C_h+C_{puf}+C_{fcs}$	$9C_h$	$19C_h+C_{puf}+C_{fcs}$	2.3897

- C_h: Computational complexity to execute a one-way cryptographic hash function
- C_{puf}: Computational complexity to execute a PUF function
- C_{ecm}: Computational complexity to execute an elliptic curve scalar point multiplication
- C_{fcs}: Computational complexity to execute a fuzzy extraction operation

Based on the experimental results reported in [5], we have $C_h \approx 0.0023$ms, $C_{fcs} \approx C_{ecm} \approx 2.226$ms and $C_{puf} \approx 0.12$ms. Based on these results, we calculate the rough computation time (in milliseconds) and present the results in Table 3. It is worth noting that our scheme has low computation cost compared to Feng *et al.* 's scheme, and its cost is also comparable with the schemes of Shen *et al.* Although our scheme has high computation cost compared to that for the schemes of Barman *et al.*, Sood *et al.*, our scheme offers superior security and more functionality features (see Table 3). Hence, it can be argued that the proposed scheme is secure and more efficient for multi-server authentication.

VI. CONCLUSION

In this paper, we presented a secure biometrics and PUFs-based authentication scheme with key agreement for multi-server environments, which allows users to login servers without password. Our scheme allows user to anonymously communicate with the server and users only need to register with the registry once to access multiple servers in the registry. The proposed protocol provides the desired security characteristics efficiently for smart card by exploiting the inherent security features of PUFs. Hence, we argue that the proposed scheme is be a viable and promising solution for the security of multi-server environment authentication.

REFERENCES

[1] G. Xu, S. Qiu, H. Ahmad, G. Xu, Y. Guo, M. Zhang, and H. Xu, "A multi-server two-factor authentication scheme with un-traceability using elliptic curve cryptography," *Sensors*, vol. 18, no. 7, p. 2394, Jul. 2018.
[2] N. M. R. Lwamo, L. Zhu, and C. Xu, "SUAA: A secure user authentication scheme with anonymity for the single & multi-server environments," *Inf. Sci.*, vol. 477, pp. 369–385, Mar. 2019.
[3] S. Kumari, M. K. Khan, and X. Li, "An improved remote user authentication scheme with key agreement," *Comput. Electr. Eng.*, vol. 40, no. 6, pp. 1997–2012, Aug. 2014.
[4] S. Ibjaoun, A. A. E. Kalam, V. Poirriez, A. A. Ouahman, and M. de Montfort, "Analysis and enhancements of an efficient biometric-based remote user authentication scheme using smart cards," in *Proc. IEEE/ACS 13th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2016, pp. 1–8.
[5] S. Kumari, A. K. Das, X. Li, F. Wu, M. K. Khan, Q. Jiang, and S. K. H. Islam, "A provably secure biometrics-based authenticated key agreement scheme for multi-server environments," *Multimedia Tools Appl.*, vol. 77, no. 2, pp. 2359–2389, 2018.
[6] H. Yao, C. Wang, X. Fu, C. Liu, B. Wu, and F. Li, "A privacy-preserving RLWE-based remote biometric authentication scheme for single and multi-server environments," *IEEE Access*, vol. 7, pp. 109597–109611, 2019.
[7] L. WB and C. C. Chang, "User identification and key distribution maintaining anonymity for distributed computer network," *Comput. Syst. Sci. Eng.*, vol. 15, no. 4, pp. 211–214, 2000.
[8] L. Lamport, "Password authentication with insecure communication," *Commun. ACM*, vol. 24, no. 11, pp. 770–772, Nov. 1981.
[9] T. Hwang, Y. Chen, and L. CS, "Non-interactive password authentications without password tables," in *Proc. IEEE region Conf. Comput. Commun. Syst.*, Sep. 1990, pp. 429–431.
[10] N.-Y. Lee and Y.-C. Chiu, "Improved remote authentication scheme with smart card," *Comput. Standards Inter.*, vol. 27, no. 2, pp. 177–180, 2005.
[11] W.-S. Juang, S.-T. Chen, and H.-T. Liaw, "Robust and efficient password-authenticated key agreement using smart cards," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2551–2556, Jun. 2008.
[12] M. K. Khan, J. Zhang, and X. Wang, "Chaotic hash-based fingerprint biometric remote user authentication scheme on mobile devices," *Chaos, Solitons Fractals*, vol. 35, no. 3, pp. 519–524, Feb. 2008.
[13] D.-Z. Sun, J.-P. Huai, J.-Z. Sun, J.-X. Li, J.-W. Zhang, and Z.-Y. Feng, "Improvements of Juang's password-authenticated key agreement scheme using smart cards," *Comput, Standards Inter.*, vol. 56, no. 6, pp. 2284–2291, 2009.
[14] L.-H. Li, L.-C. Lin, and M.-S. Hwang, "A remote password authentication scheme for multiserver architecture using neural networks," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1498–1504, Nov. 2001.
[15] I.-C. Lin, M.-S. Hwang, and L.-H. Li, "A new remote user authentication scheme for multi-server architecture," *Future Gener. Comput. Syst.*, vol. 19, no. 1, pp. 13–22, Jan. 2003.
[16] X. Cao and S. Zhong, "Breaking a remote user authentication scheme for multi-server architecture," *IEEE Commun. Lett.*, vol. 10, no. 8, pp. 580–581, Aug. 2006.
[17] J.-L. Tsai, "Efficient multi-server authentication scheme based on one-way hash function without verification table," *Comput. Secur.*, vol. 27, nos. 3–4, pp. 115–121, May 2008.
[18] W.-J. Tsaur, J.-H. Li, and W.-B. Lee, "An efficient and secure multi-server authentication scheme with key agreement," *J. Syst. Softw.*, vol. 85, no. 4, pp. 876–882, Apr. 2012.
[19] E.-J. Yoon and K.-Y. Yoo, "Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem," *J. Supercomput.*, vol. 63, no. 1, pp. 235–255, 2013.
[20] D. He and D. Wang, "Robust biometrics-based authentication scheme for multiserver environment," *IEEE Syst. J.*, vol. 9, no. 3, pp. 816–823, Sep. 2015.

[21] M.-C. Chuang and M. C. Chen, "An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1411–1418, Mar. 2014.

[22] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, "Secure biometric-based authentication scheme using chebyshev chaotic map for multi-server environment," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 824–839, Sep. 2018.

[23] S. Barman, A. K. Das, D. Samanta, S. Chattopadhyay, J. J. P. C. Rodrigues, and Y. Park, "Provably secure multi-server authentication protocol using fuzzy commitment," *IEEE Access*, vol. 6, pp. 38578–38594, 2018.

[24] P. Gope and B. Sikdar, "Privacy-aware authenticated key agreement scheme for secure smart grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3953–3962, Jul. 2019.

[25] S. Ryu, "PUF based smart meter security with SX chain," *Int. J. Control Autom.*, vol. 9, no. 9, pp. 407–414, Sep. 2016.

[26] T. Esbach, W. Fumy, and O. Kulikovska, "A new security architecture for smartcards utilizing PUFs," in *Proc. Securing Electron. Bus. Processes. (ISSE)*. Wiesbaden, Germany: Springer-Vieweg, 2012, pp. 180–194.

[27] F. Armknecht, D. Moriyama, and A. R. Sadeghi, "Towards a unified security model for physically unclonable functions," in *Proc. Cryptographers' Track at RSA Conf.* Cham, Switzerland: Springer, 2016, pp. 271–287.

[28] Y. Dodis, J. Katz, L. Reyzin, and A. Smith, "Robust fuzzy extractors and authenticated key agreement from close secrets," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 4117. 2006, pp. 232–250.

[29] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. 8th Int. Workshop Theory Pract. Public Key Cryptogr. (PKC)*, in Lecture Notes in Computer Science, vol. 3386. Berlin, Germany: Springer, 2005, pp. 65–84.

[30] A. Dua, N. Kumar, A. K. Das, and W. Susilo, "Secure message communication protocol among vehicles in smart city," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4359–4373, May 2018.

[31] D. Chattaraj, M. Sarma, and A. K. Das, "A new two-server authentication and key agreement protocol for accessing secure cloud services," *Comput. Netw.*, vol. 131, pp. 144–164, Feb. 2018.

[32] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 357–366, Jan. 2016.

[33] J. Srinivas, A. K. Das, N. Kumar, and J. Rodrigues, "Cloud centric authentication for wearable healthcare monitoring system," *IEEE Trans. Dependable Secure Comput.*, to be published, doi: 10.1109/TDSC.2018.2828306.

[34] V. Odelu, A. K. Das, and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 9, pp. 1953–1966, Sep. 2015.

[35] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990.

[36] Q. Feng, D. He, S. Zeadally, and H. Wang, "Anonymous biometrics-based authentication scheme with key distribution for mobile multi-server environment," *Future Gener. Comput. Syst.*, vol. 84, pp. 239–251, Jul. 2018.

[37] S. K. Sood, A. K. Sarje, and K. Singh, "A secure dynamic identity based authentication protocol for multi-server architecture," *J. Netw. Comput. Appl.*, vol. 34, no. 2, pp. 609–618, Mar. 2011.

[38] H. Shen, C. Gao, D. He, and L. Wu, "New biometrics-based authentication scheme for multi-server environment in critical systems," *J. Ambient Intell. Humanized Comput.*, vol. 6, no. 6, pp. 825–834, Jul. 2015, doi: 10.1007/s12652-015-0305-8.

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

JUN ZHAO is currently pursuing the M.S. degree with Anhui Normal University, China. His research interests include lightweight authentication, authenticated encryption, security in mobile communication, and biometrics privacy protect.

WEIXIN BIAN received the M.S. degree in computer science from the Guizhou University, China, in 2005, and the Ph.D. degree in computer science from the China University of Mining and Technology, China, in 2018. In 2006, he joined the School of Computer and Information, Anhui Normal University, where he is currently an Associate Professor. His research interests include information security, image processing, machine learning, and pattern recognition.

DEQIN XU received the M.S. degree in automation from the Guizhou University, China, in 2006. Her research interests include pattern recognition, image processing, and machine learning.

BIAO JIE received the M.S. degree in computer science from Yunnan Normal University, Yunnan, China, in 2006, and the Ph.D. degree in computer science from the Nanjing University of Aeronautics and Astronautics, China, in 2015. In 2006, he joined the School of Computer and Information, Anhui Normal University, where he is currently a Professor. His research interests include machine learning and medical image analysis.

XINTAO DING received the M.S. degree in computational mathematics from East China Normal University, in 2005, and the Ph.D. degree in geography and tourism from Anhui Normal University, China, in 2015. He is currently an Associate Professor with Anhui Normal University. His research interests include machine learning and pattern recognition.

WEN ZHOU received the Ph.D. degree from the School of Software Engineering, Tongji University, in 2018. In 2018, he joined the School of Computer and Information, Anhui Normal University, where he is currently a Lecturer. His research interests include computer vision and machine learning.

HUI ZHANG is currently pursuing the M.S. degree with Anhui Normal University, China. Her research interests include image processing, machine learning, and biometrics privacy protect.