IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 28, 2019, accepted December 16, 2019, date of publication December 30, 2019,
date of current version January 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962912

Secure Remote Multi-Factor Authentication Scheme Based on Chaotic Map Zero-Knowledge Proof for Crowdsourcing Internet of Things

WENZHENG LIU, XIAOFENG WANG, AND WEI PENG
College of Computer, National University of Defense Technology, Changsha 410073, China

Corresponding authors: Xiaofeng Wang (xf_wang@nudt.edu.cn) and Wei Peng (wpeng@nudt.edu.cn)

This work was supported in part by the Project of the National Key Research and Development Program of China under Grant 2017YFB0802300.

ABSTRACT Recently, application scenario of crowdsourcing IoT has covered to e-healthcare service, smart home, smart city, internet of vehicles due to the proliferation of smart devices such as smart mobile devices, smart wearable device, smart medical devices and smart furniture, etc. Patient's data collected by the smart devices send to the various remote medical servers. A group of medical professionals remote access patient data stored at the medical server database. Smart home users want to remote real-time access information of smart devices at home. All these operations need via wireless remote communication, which is suffering from various kinds of threat and attacks. Hence, there are a large number of multi-factor remote authentication and key agreement schemes designed for the application of crowdsourcing IoT. However, in most existing related multi-factor schemes, all factors for identity authentication only act as a parameter for encrypting the local secret key. In this paper, we propose a new secure remote multi-factor authentication scheme that includes three factors: 1) user identity; 2) password; and 3) user biometrics, which are authenticated by the remote server, act as a part of the secret key and participate in the key agreement process. We choose the chaotic map since it has a smaller key size and lower computational overhead, and then achieve remote multi-factor authentication and key agreement by artfully employ it to zero-knowledge technology and the fuzzy extractor technology. Our scheme is more secure and robust since the user revealing nothing sensitive information, and the adversary cannot impersonate any user even if he gets the server's master key. We have done security proof for our proposed scheme using the Random-Or-Real(ROR) model, Burrows-Abadi-Needham (BAN) logic, and ProVerif 2.00 to show that the presented scheme is secure. Also, we give an additional security analysis for other various attacks. Finally, according to the test and simulation result, the proposed scheme is very suitable for the power-constrained smart devices, and in the next generation 5G communication environment, its applicability and usability will be greatly enhanced.

INDEX TERMS Chaotic map, zero-knowledge proof, remote multi-factor authentication, Internet of Thing (IoT), crowdsourcing, random-or-real (ROR) model, BAN logic.

I. INTRODUCTION

The Internet of Things is rapidly becoming one of the fastest-growing areas due to the extensive range of equipment in both the research community and domestic markets. There are several open research issues within the field of IoT, such as device detection, schema alignment, access control, and data management [20]. Recently, crowdsourcing research has gradually become a research direction to improve the current research challenges of the Internet of Things.

According to the traditional definition, crowdsourcing is a company or organization outsources the tasks performed by employees in the past to a non-specific (and often large) mass network in a free and voluntary manner. With the rise of the Internet of Things, smart IoT devices are gradually increasing, and a large number of IoT devices crowdsourcing complete difficult and complicated work, which brings up the concept of crowdsourcing IoT. It has been used in fields such

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

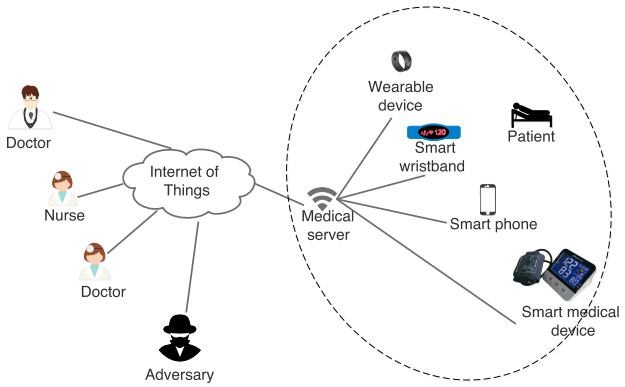

FIGURE 1. Crowdsourcing Internet of Things (IoT) in e-healthcare services.

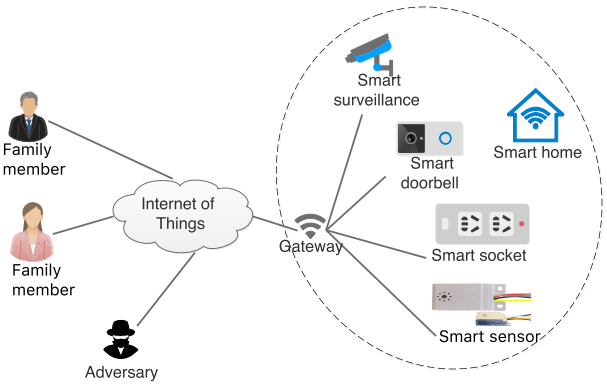

FIGURE 2. Crowdsourcing Internet of Things (IoT) in smart home.

as the e-healthcare systems, smart home, smart city, internet of vehicles, etc.

For e-healthcare system, it is necessary to collect patient sensitive information through smart devices and share with a group of medical professionals in a protected online environment, and for these types of treatments, where multiple professionals are involved, crowdsourcing Internet of Things (IoT) in e-healthcare services(Figure 1) is required. However, the growing use of the Internet provides opportunities for malicious users and attackers to gain unauthorized access to medical data through the use of various network and information attacks. In order to protect critical and private medical information, researchers need to pay more attention to designing appropriate security protocols for crowdsourcing in e-health services. This requires remote user authentication and key agreement schemes to provide access to the service to authorize only users.

The smart home is another application scenario for crowdsourcing IoT(Figure 2). Its network can be implemented with the help of smart device(such as smart doorbell, smart power control, smart sensors, surveillance cameras and so on), wherein all of these devices can communicate through a wireless channel by a home gateway node which acts as a bridge between smart device and the home user. To secure remote access information of smart devices, the home gateway node need remote authenticates the user's identity and establish a session key.

In addition, in other crowdsourcing IoT applications, remote authentication schemes for user access are also the focus of research.

A. RELATED WORK

In recent research, considering the power-constrained of most IoT smart devices, high access rate, and privacy protection for participants at wireless remote access communication, there are a large number of related scheme have been proposed.

Xu *et al.* [10] proposed a two-factor mutual authentication and key agreement scheme to reduce the computational cost based on the elliptic curve cryptography(ECC), which enables to provide anonymity by employing the dynamic identity. Yan *et al.* [12] proposed a biometric based user authentication scheme. But his scheme is vulnerable to the replay attack and can not ensure user anonymity. Mishra also pointed out that Yan's scheme [12] does not protect against the off-line password guessing attack. Therefore, Mishra *et al.* [13] further proposed an enhanced biometric-based authentication scheme using random numbers. In 2015, Tan and Zuowen [14] extended the security requirements of two-factor authentication schemes to three-factor authentication schemes, which are mutual authentication, server not knowing password and biometric, and three-factor security.

Compared to the traditional cryptographic schemes(such as RSA or ECC), schemes based on chaotic maps have shown better performance at low-power computing and have smaller security key size, which is suited for IoT smart devices. Guo *et al.* [31] first proposed a chaotic map based password authentication scheme for the e-healthcare information system, which avoids modular exponential computing or scalar multiplication on elliptic curve used in traditional authentication schemes. While Hao et al. pointed out Guo's scheme does not preserve user anonymity and inefficiency of double secret keys. Then Hao et al. proposed their improved scheme [7], which overcome Guo's weakness. In the same year, Lee and Fu [21] and Jiang *et al.* [32] modified Hao's scheme with higher security. Li *et al.* [22] finds both Lee's [21] and Jiang's [32] schemes are vulnerable to the service misuse attack and give a secure authentication scheme to cope with the security weaknesses. Lu et al. pointed out that Chun's improved scheme still has some weaknesses, such as a vulnerability to the user impersonation attack, a lack of local verification, and a violation of the session key security. They subsequently proposed a robust and efficient three-factor authentication scheme [33]. Moon *et al.* [6] found that Lu et al.'s scheme is not secure against the replay attack, the impersonation attack, and the outsider attack. To solve these security vulnerabilities, they propose a modified authentication scheme. In 2018, Roy *et al.* [1] found that the existing related scheme suffered from denial of server attack and did not provide a mechanism for revocation.

Then Roy proposed a lightweight three factors remote authentication and can resist various know attacks.

B. MOTIVATION

The existing related schemes do not fully exploit the unique characteristics of multi-factor authentication(Fig 3). Most of proposed related schemes use multi-factor to encrypt the secret key issued by the registration service. During the authentication process, the user completes the multi-factor verification locally and use them to decrypt the secret key, and then using the secret key for server side authentication and key agreement. All the authentication factors neither authenticated by the server nor participate in key agreement. Therefore, In the case of secret key leaks, the adversary can impersonate as a user completes the authentication process, and do not need to complete either of authentication factor verification. Compared with the traditional PKI and IBE schemes, these schemes have no essential difference.

In this paper, we aim to design a new secure lightweight remote multi-factor authentication scheme for crowdsourcing IoT application, which all authentication factors are authenticated by the remote server, act as a part of the secret key and participate in the key agreement process. In this scheme, the server no longer authenticates the secret key stored at the user's smart device client, but directly authenticates the user's authentication factor. To confirm the real user who operates on the client side, the server can remote authenticates that whether the user can actually input a plurality of factors provided at the time of registration.

To achieve this target, we introduce technologies including chaotic map, zero-knowledge proof, and fuzzy extractor. But we are not just giving a simple combination of these technologies. Chaotic map has better performance at low-power computing and smaller security key size compared to traditional cryptographic schemes(such as RSA or ECC). A zero-knowledge proof enables the prover to make sure the verifier is certain that some statements are correct, but the verifier does not learn anything except the validity of the statement. Fuzzy extractor technology can symbolize user biometrics. We design a scheme based on chaotic map cryptography, and then artfully employ it to privacy-preserving remote multi-factor authentication through fuzzy extractor technology and zero-knowledge technology by exploiting the mathematical properties of Chebyshev Polynomial.

C. OUR CONTRIBUTION

In this paper, we proposed a secure remote biometric-based authentication scheme based on chaotic map zero-knowledge for application of crowdsourcing Internet of Things. The main contributions are discussed as follows.

1) We first achieve a remote multi-factor authentication scheme based on chaotic map zero-knowledge proof. In our scheme, all authentication factors can be remotely authenticated by the server or gateway node and participate in the process of the key agreement(Fig 4). The server can

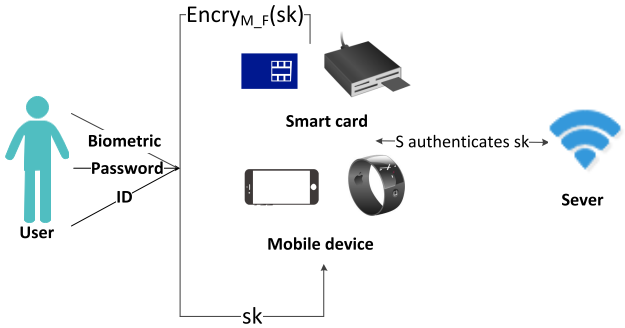

FIGURE 3. Multi-factor authentication process of existing related schemes.

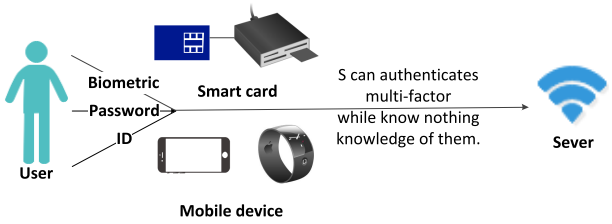

FIGURE 4. Multi-factor authentication process of our scheme.

authenticate all authentication factors at once or authenticate them one by one after a slight improvement for the scheme.

2) To protect the user's privacy, our scheme does not transmit or store any sensitive information from the user. The server and user complete the mutual authentication and key agreement phase by revealing nothing sensitive information. Because we use the chaotic map zero-knowledge proof to verify the user's sensitive information, the user can prove that he knows or owns a secret without revealing what it is.

3) Compare to the existing related schemes, our scheme has low computation and communication overheads and very useful for resource-constrained and battery-powered devices.

4) The proposed scheme can resist various know attacks and provides more security properties. An adversary cannot impersonate any user even if he gets the server's secret key. We give the formal security proof through the Real-Or-Random(RoR) model, BAN logic, and ProVerif 2.00 as well as give the additional security analysis for other various attacks.

D. THREAT MODEL

The threat model used in the proposed scheme is the well-know DolevYao [35] threat model (DY model), which accepts the following basic assumptions:

- The user U_i and S are communicated over a public insecure channel.
- The adversary $\mathcal{A}$ can execute eavesdropping, deletion, or modification of messages on public channels.
- Smart devices can be physically captured by $\mathcal{A}$, and all the credentials stored in those smart devices can be extracted by $\mathcal{A}$ using the power analysis attacks.

E. PAPER ORGANIZATION

Section II introduces the preliminary of zero-knowledge proof, fuzzy extractor, and Chebyshev polynomial chaotic maps briefly. Section III presents the procedure of our scheme in detail. In Section IV, the security of the proposed scheme is discussed. We compare the performance among our scheme and other related schemes in Section VI. Finally, Section VI concludes the paper and proposes the direction of future research.

II. MATHEMATICAL PRELIMINARIES

We apply zero-knowledge proof, fuzzy extractor, and Chebyshev polynomial chaotic maps for the proposed authentication scheme. For this purpose, we describe the fundamental concepts on zero-knowledge [37], fuzzy extractor on biometrics input [23], and Chebyshev polynomial chaotic maps [8], [9].

A. CHEBYSHEV POLYNOMIAL AND CHAOTIC MAP

The first class Chebyshev polynomial $T_n(x) : [-1, 1] \rightarrow [-1, 1]$ is defined as :

$$T_n = cos(narcos(x))(-1 \leq x \leq 1)$$

, or

$$T_n(x) = \begin{cases} 1, & if\ n = 0 \\ x, & if\ n = 1 \\ 2xT_{n-1}(x) - T_{n-2}(x) & if\ n \geq 2. \end{cases}$$

Theorem 1: The Chebyshev polynomial satisfies the semi-group propert:

$$T_r(T_s(x)) = T_s(T_r(x)),$$

for $r, s \in N$ and $x \in [-1, 1]$.

Definition 1 [9]: The enhanced Chebyshev polynomial holds on the interval $(-\infty, +\infty)$ and is defined as follows:

$$T_n = 2xT_{n-1}(x) - T_{n-2}(x)mod(p), \quad n \geqslant 2$$

where $n \geqslant 2, x \in (-\infty, +\infty)$, and p is a large prime number.

Theorem 2 [9]: The enhanced Chebyshev polynomial satisfies the semi-group propert:

$$T_r(T_s(x)) \equiv T_s(T_r(x)) \equiv T_{rs}(x)(modp),$$

where p is a large prime number and $x \in (-\infty, +\infty)$.

Theorem 3 [4]: Assume $a = b + c$, where $b, c \in N$ and $b, c \geq 2$, we have the following formula:

$$(2T_a(M)T_b(M)T_c(M)) + 1) \equiv (T_a^2(M) + T_b^2(M) + T_c^2(M))(modp). \quad (1)$$

Definition 2: Chaotic map-based discrete logarithm problem (CMDLP): For any given x and y, it is computationally infeasible to find integer r such that $T_r(x) = y$. The advantage probability of $\mathcal{A}$ to solve CMDLP is :

$$Adv_{\mathcal{A}}^{CMDLP}(t) = Pr[\mathcal{A}(x, y) = r : r \in Z_p^*, y = T_r(x)(modp)]. \quad (2)$$

Definition 3: Chaotic map-based computational Diffie-Hellman problem (CMCDHP): For any given x, T_s, and T_m, it is computationally infeasible to find integer $r = ms$ such that $T_r(x) = T_{ms}(x) = y$. The advantage probability of $\mathcal{A}$ to solve CMCDHP is :

$$Adv_{\mathcal{A}}^{CMCDHP}(t) = Pr[\mathcal{A}(x, y) = r : r \in Z_p^*, y = T_{ms}(x)(modp)]. \quad (3)$$

B. ZERO-KNOWLEDGE PROOF

A zero-knowledge proof enables the prover (P) to make sure the verifier (V) is certain that some statements are correct, but the verifier (V) does not learn anything except the validity of the statement. In our scheme, we refer to the zero-knowledge proof proposed by Schnorr [37]. For a large prime number p and the generate element g of Z_p^*, this zero-knowledge proof allows prover P to prove the knowledge of $s \in Z_p^*$ such that $y = g^s$ for some $y \in Z_p^*$ to verifier V.

Commitment: Prover P selects a random number $q \in Z_p^*$, and computes $T = g^q$ and then sends T to verifier V.

Challenge: Verifier V generates a random $c \in \{0, 1\}^n$ and sends it back to P.

Response: Prover P computes $z = q - cs(modp)$ and returns it to verifier V.

Verify: Verifier V accepts the Prover's proof if and only if $T = y^c g^z$.

C. BIOMETRICS AND FUZZY EXTRACTOR

Given biometric input B, such as fingerprint or face from the user, a fuzzy extractor could extract the random string θ and the auxiliary string σ. Once input a new biometric B^*, which differs from the original input biometric B up to the threshold value, and the auxiliary string σ, the fuzzy extractor will recover θ [36].

Definition 4 [23]: An $(\mathcal{M}, m, l, t, \epsilon) - fuzzy\ extractor$ is a pair of randomized procedures, Gen() and Rep(), with the following properties :

- Gen: $(\mathcal{M}, \sigma) = Gen(B)$. It takes $B \in \mathcal{M}$ as input and outputs a pair (θ, σ), where $\theta \in \{0, 1\}^l$ act as the biometric key and $\sigma \in \{0, 1\}^*$ is a help string.
- Rep: $\theta = Rep(B^*, \sigma)$. It takes a new biometrics B^* and the helper string σ as inputs. The correctness property of fuzzy extractors guarantees that if $dis(B, B^*) < t$, Rep can recovers the original θ.
- The security property guarantees that for any distribution W on $\mathcal{M}$ of m, the string θ is nearly uniform even for those who observe σ.

$\mathcal{M} = \{0, 1\}^n$ is a metric space.

m is the min-entropy of any distribution W on metric space $\mathcal{M}$;

l is the length of θ;

t is the error tolerance threshold;

ϵ is the statistical distance between two given probability distributions.

TABLE 1. Notations used in this paper.

Notation	Description
U_i	i^{th} User
ID_i	U_i's identity
PW_i	U_i's password
B_i	Biometric of U_i
θ_i	Biometric secret key of U_i
σ_i	Public reproduction parameter of θ_i
Gen	Fuzzy extractor probabilistic generation procedure
Rep	Fuzzy extractor deterministic reproduction procedure
S	server or gateway node
$H(\cdot)$	Collision-resistant cryptographic hash function
$\|\|$	concatenation
$\oplus$	Bitwise XOR operations
T	A period of time
T_s	current timestamp of U_i
T_s^*	current timestamp of S

III. PROPOSED SCHEME

In this section, we present the proposed scheme in detail. The proposed scheme has four phases, namely: 1) System setup; 2) registration; 3) login, authentication and key agreement; 4) Password, biometric change and smart card or device revocation phase. For describing and analyzing the proposed scheme, we use the notations listed in table 1.

A. SYSTEM SETUP

In this phase, Server S generates some parameters of the system.

S selects a large prime number p, and the entended Chebyshev polynomial $T_n(x)$, where $x \in (-\infty, +\infty)$. $H_1(\cdot) : \{0,1\}^* \rightarrow Z_p^*$ and $H_2(\cdot) : \{0,1\}^* \rightarrow \{0,1\}^n$ are hash function. The common reference string is $< p, H_1(\cdot), H_2(\cdot), T_n(x) >$.

B. REGISTRATION PHASE

Through the registration phase, the user U_i registers with the server and gets a certificate via a secure channel. The following steps need to be executed.

Step 1: U_i first chooses his own identity ID_i, personal password PW_i and imprints his biometric B_i to the registered device (It can be a smart device that installs related applications);

Step 2: The registered device produces $(\theta_i, \sigma_i) = Generation(B_i)$ for U_i by fuzzy extractor and generates a random number $r_i \in Z_p^*$. Then it computes $H_1(ID_i||PW_i||\theta_i||r_i||T) = x_i \in Z_p^*$, where T is a period of time(such as one week, one month and one year) and $H_2(ID_i||PW_i||\theta_i||r_i||T) = X_i \in \{0,1\}^n$;

Step 3: The registered device generates $T_{x_i}(X_i)$ and submits $< T_{x_i}(X_i), T, X_i, ID_i >$ to S via a secure channel;

Step 4: S receives the registration request and compute $M = T_{x_s}(X_i)$. Then S sends $< M >$ back to the registered device and stores $< ID_i, T_{x_i}(X_i), T, X_i >$ at the database.

Step 5: The registered device receives the M from the S and stores $< M, r_i, \sigma_i >$ at the smart card or the user's mobile device.

Table 2 shows the registration phase involved in the proposed.

C. LOGIN, AUTHENTICATION AND KEY AGREEMENT

To access the services from S, U_i must complete the login, authentication and key agreement phase. This phase are involved following steps.

Step 1: U_i first inserts smart card to the authentication device or opens the application installed in the smart device (we called all these devices SC) and inputs his identity ID_i, password PW_i and biometrics B_i^* at the sensor. The device computes $\theta_i = Rep(B_i^*, \sigma_i)$, $x_i = H_1(ID_i||PW_i||\theta_i||r||T)$ and $X_i = H_2(ID_i||PW_i||\theta_i||r||T)$.

Step 2: The SC selects two random numbers $p_a, e_a \in Z_p^*$ and computes $TID = ID_i \bigoplus H_2(T_{p_a}(M)||T_{s_1})$. Then it computes $PA = T_{pa}(X_i)$, $N_i = X_i \oplus e_a \oplus T_{s_1}$ and sends the message $M_1 =< TID, T_{s_1}, PA, N_i >$ to the server S at time T_{s_1}.

Step 3: S receives the user's message at time $T_{s_1}^*$ and then it verifies whether $|T_{s_1}^* - T_{s_1}| \leq \bigtriangleup T$ where $\bigtriangleup T$ is the maximum transmission delay. S computes $ID_i = H_2(T_{x_s}(PA)||T_{s_1}) \oplus TID$, searches $< ID_i, T_{x_i}(X_i), T, X_i >$ in the database and verifies whether T is out of date? S selects two random numbers p_s, e_s and computes $e_a = N_i \oplus T_{s_1} \oplus X_i$, $PS = T_{p_s}(X_i)$ and $w_s = p_s + x_s e_a$. S computes $K_{is} = H_2(T_{s_1}||T_{s_2}^*||e_a||e_s||T_{x_s}(PA))$, $N_j = X_i \oplus e_s \oplus T_{s_2}^*$ and sends message $M_2 =< K_{is} \oplus < PS, w_s >, N_j, T_{s_2}^* >$ back to the device SC at time $T_{s_2}^*$.

Step 4: The SC receives the message M_2 at time T_{s_2} and verifies whether $|T_{s_2}^* - T_{s_2}| \leq \bigtriangleup T$. Then SC computes $e_s = N_i \oplus T_{s_2}^* \oplus X_i$ and $K_{is}^* = H_2(T_{s_1}||T_{s_2}^*||e_a||e_s||T_{pa}(X_i))$ to get w and T_{p_s}. Then, the SC verifies if $2T_{w_s}(X_i)T_{p_s}(X_i)T_{e_a}(M) + 1 \equiv T_{w_s}^2(X_i) + T_{p_s}^2(X_i) + T_{e_a}^2(M)(modp)$? If not, the device terminates the phase. else, U_i completes the authentication of the S's identity. Then it computes $w_a = p_a + x_i e_s$ and $SK = H_2(T_{s_1}||T_{s_2}^*||T_{s_3}||w_a||w_s||T_{p_a}(M)))$. Then SC sends the message $M_3 =< w_a \oplus K_{is}^* \oplus T_{s_3}, T_{s_3} >$ to the S at time T_{s_3}.

Step 5: S receives the message M_3 at time $T_{s_3}^*$ and gets w_a. Then S verifies if $|T_{s_3}^* - T_{s_3}| \leq \bigtriangleup T$ and $2T_{w_a}(X_i)T_{p_a}(X_i)T_{e_s}(T_{x_i}(X_i)) + 1 \equiv T_{w_a}^2(X_i) + T_{p_a}^2(X_i) + T_{e_s}^2(T_{x_i}(X_i))(modp)$? If not, the device terminates the phase. else, S completes the authentication of the user's identity and computes $SK = H_2(T_{s_1}||T_{s_2}^*||T_{s_3}||w_a||w_s||T_{x_s}(PA)))$ as the session key.

Table 3 shows the login, mutual authentication and key agreement phase involved in the proposed.

D. PASSWORD, BIOMETRIC CHANGE AND SMART CARD OR DEVICE REVOCATION PHASE

A valid user U can changes his old password PW_i and old biometric B_i to new password PW_i' and another biometric B_i' by using the following steps.

Step 1: U_i sends the revocation quest to the Server.

Step 2: U_i completes the Login, mutual authentication and key agreement phase.

TABLE 2. Registration phase of users.

User U_i	Server S
Input ID_i, PW_i, B_i.	
Select a random number $r_i \in Z_p^*$	
Compute $(\theta_i, \sigma_i) = Gen(B_i)$,	
$x_i = H_1(ID_i \| PW_i \| \theta_i \| r_i \| T) \in Z_p^*$,	
$X_i = H_2(ID_i \| PW_i \| \theta_i \| r_i \| T) \in \{0,1\}^n$.	
$\xrightarrow[secure\ channel]{<T_{x_i}(X_i), T, X_i, ID_i>}$	
	compute $M = T_{x_s}(X_i)$,
	store $< ID_i, T_{x_i}(X_i), T, X_i >$ at the database.
	$\xleftarrow[secure\ channel]{<M>}$
store $< M, r_i, \sigma_i >$into smart card or mobile device.	

TABLE 3. Login, mutual authentication and key agreement phase.

User U_i	Server S
Input ID_i, PW_i, B_i^*.	
Select two random numbers $p_a, e_a \in Z_p^*$.	
computes	
$(\theta_i, \sigma_i) = Gen(B_i^*), x_i = H_1(ID_i \| PW_i \| \theta_i \| r_i \| T)$,	
$X_i = H_2(ID_i \| PW_i \| \theta_i \| r_i \| T)$,	
$TID = ID_i \oplus H_2(T_{p_a}(M) \| T_{s_1}), PA = T_{p_a}(X_i)$,	
$N_i = X_i \oplus e_a \oplus T_{s_1}$.	
$\xrightarrow[public\ channel]{<TID, T_{s_1}, PA, N_i>}$	
	verifies if $\|T_{s_1} - T_{s_1}^*\| < \triangle T$?
	computes $ID_i = H_2((T_{x_s}(PA)) \| T_{s_1}) \oplus TID$.
	searches $< ID_i, T_{x_i}(X_i), T, X_i >$ in the database.
	verifies whether T is out of date?
	select two random numbers $p_s, e_s \in Z_p^*$ and computes
	$e_a = N_i \oplus T_{s_1} \oplus X_i, PS = T_{p_s}(X_i), w_s = p_s + x_s e_a$,
	$K_{is} = H_2(T_{s_1} \| T_{s_2}^* \| e_a \| e_s \| T_{x_s}(PA))), N_j = X_i \oplus e_s \oplus T_{s_2}^*$
	$\xleftarrow[public\ channel]{<K_{is} \oplus <PS, w_s>, N_j, T_{s_2}^*>}$.
verifies if $\|T_{s_2} - T_{s_2}^*\| < \triangle T$?	
computes $e_s = N_i \oplus T_{s_2}^* \oplus X_i$,	
$K_{is}^* = H_2(T_{s_1} \| T_{s_2}^* \| e_a \| e_s \| T_{p_a}(M)))$.	
verifies if $2T_{w_s}(X_i)T_{p_s}(X_i)T_{e_a}(M) + 1$	
$\equiv T_{w_s}^2(X_i) + T_{p_s}^2(X_i) + T_{e_a}^2(M) mod p$?	
if verification holds, computes $w_a = p_a + x_i e_s$.	
and $SK = H_2(T_{s_1} \| T_{s_2}^* \| T_{s_3} \| w_a \| w_s \| T_{p_a}(M)))$.	
$\xrightarrow[public\ channel]{<K_{is}^* \oplus <w_a> \oplus T_{s_3}, T_{s_3}>}$	
	computes $w_a = K_{is}^* \oplus K_{is} \oplus < w_a > \oplus T_{s_3}$.
	verifies if $\|T_{s_3}^* - T_{s_3}\| \leq \triangle T$?
	$2T_{w_a}(X_i)T_{p_a}(X_i)T_{e_s}(T_{x_i}(X_i)) + 1$
	$\equiv T_{w_a}^2(X_i) + T_{p_a}^2(X_i) + T_{e_s}^2(T_{x_i}(X_i)) mod p$?
	If verification holds, authentication is successful.
	compute $SK = H_2(T_{s_1} \| T_{s_2}^* \| T_{s_3} \| w_a \| w_s \| (T_{x_s}(PA)))$
	as the session key share with U_i.

Step 3: U_i inputs his new password PW_i', another biometrics B_i' at the sensor and chooses a new period of time T'.

Step 4: SC selects a random number r_i' and produces $(\theta_i', \sigma_i') = Generation(B_i')$. SC computes $x_i' = H_1(ID_i \| PW_i' \| \theta_i' \| r_i' \| T')$, $X_i' = H_2(ID_i \| PW_i' \| \theta_i' \| r_i' \| T')$ and sends $SK \oplus < T_{x_i'}(X_i'), X_i' >$ to the S.

Step 5: S gets the message and stores $< T_{x_i'}(X_i'), T', X_i' >$ instead of $< T_{x_i}(X_i), T, X_i >$. Then S computes $M' = T_{x_s}(X_i')$ and sends $K_{is} \oplus M'$ back to the SC.

Step 6: SC stores $< M', r_i', \sigma_i' >$ at the smart card or the user's mobile device instead of $< M, r_i, \sigma_i >$.

Finally, the user's authenticates credential will not be available and automatic revocation after the time T expires.

If a legal user U_i's smart card or device is stolen or lost, it is required to revoke the lost SC and allow U_i to login using new SC. The proposed scheme perform the following steps.

Step 1: the U_i initiates revocation phase and chooses his own identity ID_i, new password PW_i^*, and imprints his biometric B_i^* to the SC;

Step 2: The SC produces $(\theta_i^*, \sigma_i^*) = Generation(B_i^*)$ for U_i by fuzzy extractor and generates a random number $r_i^* \in Z_p^*$.

Then it computes $H_1(ID_i||PW_i^*||\theta_i^*||r_i^*||T^*) = x_i^* \in Z_p^*$ and $H_2(ID_i||PW_i^*||\theta_i^*||r_i^*||T^*) = X_i^* \in \{0,1\}^n$;

Step 3: The SC generates $T_{x_i^*}(X_i^*)$ and submits revocation quest $< T_{x_i^*}(X_i^*), T^*, X_i^*, ID_i >$ to S via a secure channel;

Step 4: S receives the revocation request and verifies authenticity of U by checking other credentials, such as date of registration and registered id number. Then it computes $M^* = T_{x_s}(X_i^*)$, sends $< M^* >$ back to the SC, and stores $< ID_i, T_{x_i^*}(X_i^*), T^*, X_i^* >$ at the database.

Step 5: The SC receives the M^* from the S and stores $< M^*, r_i^*, \sigma_i^* >$.

IV. SECURITY ANALYSIS

In this section, we prove the semantic security of the proposed scheme by using the random-or-real model. And then, with the help of BAN logic [19], we provide the mutual authentication proof between the user and the server in our scheme. In the end, we also have given additional security analysis for other known attacks.

A. FORMAL SECURITY ANALYSIS USING RANDOM-OR-REAL MODEL

In this section, we give the formal analysis for our proposed scheme through the random-or-real(ROR) model [1], [2], [16]. To remove ambiguity, we mention a common notation C for both participants U_i and S. In order to break the security of scheme, we assure that an adversary $\mathcal{A}$ executes different attacks, which using various oracle queries as follows:

Execute(C, S): This query models passive attacks in which $\mathcal{A}$ can eavesdrops or outputs a message m exchanged between U_i and S in an actual execution of the scheme.

Send(C, m): An active attack that $\mathcal{A}$ sends a request message m to C, and C replies to $\mathcal{A}$ according to the rules of the scheme.

Revel(C): In this query, if the session key has been generated, C return it back to $\mathcal{A}$. Otherwise, return a *null* value.

Corrupt(U_i, a): This query simulate the capability of $\mathcal{A}$ to obtain sensitive information of the user U_i:

if a = 1, query returns U_i's password;

if a = 2, query returns U_i's biometric secret string θ_i

if a = 3, query returns U_i's smart device stored parameters.

Test(C): This query can be invoked only once. If there is no session key, a *null* value will be returned to $\mathcal{A}$. Otherwise C takes decision based on the output of the coin b:

if b = 1, C returns current session key SK;

if b = 0, C returns a random string.

Definition 5: If upon receiving the last expected protocol message, an instance C is said to be in accepted, it goes into an accept state. The session identification(s_id) is formed by the ordered concatenation of all communicated message M_1, M_2, M_3.

Definition 6: Two instances $U_i^{t_1}$ and $S_j^{t_2}$ are said to be partnered if they fulfilled following three conditions simultaneously:

1) both are in accept state;

2) both mutually authenticate each other and share the same s_id

3) they are mutual partners of each other.

Definition 7 (Freshness): C is said to be fresh, when the following conditions are met simultaneously:

1) C is in the accept state;

2) C has never been received *Reveal*(C) query;

3) C has been received less than two *Reveal*(C) query.

Definition 8 (Semantic Security): The advantage function of $\mathcal{A}$ in breaking the semantic security of the proposed authentication and key agreement (AKA) scheme by guessing the correct bit b': $Adv_C^{AKA} = [2Pr[Succ(\mathcal{A})] - 1] = [2Pr[b = b'] - 1]$.

Definition 9: The advantage probability of CMDLP is negligible for adversary $\mathcal{A}$ with execution time $t_{\mathcal{A}}$, that is $Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}) \leq \epsilon$, for any sufficiently small $\epsilon > 0$.

Theorem 4: Let $\mathcal{A}$ be a polynomial time bounded attacker running in time $\mathcal{A}$. To break the semantic security security of the proposed scheme, adversary $\mathcal{A}$ makes H_1 and H_2 hash oracle queries, *Send* queries and *Execute* queries at most q_{H_1}, q_{H_2}, q_s, and q_e times, respectively. Then

$$\begin{aligned} Adv_C^{AKA} \leqslant & \frac{3q_{H_1}}{2^{l_{H_1}}} + \frac{q_{H_2}^2 + 6q_{H_2}}{2^{l_{H_2}}} \\ & + \left(\frac{(q_s + q_e)^4 + 4q_s^2}{2^{l_r+1}}\right) \\ & + 2max\{q_s\left(\frac{1}{|D|}, \frac{1}{2^{l_b}}, \frac{2}{2^{lr}}, \varepsilon_{bm}\right)\} \\ & + 4q_{H_2}(1 + (q_s + q_e)^2) Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}) \end{aligned} \quad (4)$$

where l_{H_1} and l_{H_2} are the string length of hash results, respectively, l_r is the string length of random number, ε_{bm} is the probability of false positive [17], $\mathcal{D}$ is a finite dictionary with size $|\mathcal{D}|$, Adv_C^{AKA} is defined in Definition 8 and $Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}})$ is defined in Definition 9.

Proof: Let $Succ_i$ refer to an event of successful guessing bit b in *Test* query by an adversary $\mathcal{A}$ in the game G_i, $i = 0, 1, 2, 3, 4, 5$.

G_0 : The real scheme in random oracles and the initial game are assumed to be identical, we obtain

$$Adv_C^{AKA} = |2Pr|Succ_0| - 1|. \quad (5)$$

G_1 : Oracle queries such as *Reveal*, *Execute*, *Corrupt*, *Test*, H_1,H_2 and *Send* queries are simulated in G_1 and working procedures of these queries are described in Table 4. G_1 create three lists:

1) L_{H_1} and L_{H_2} answer hash oracles of H_1 and H_2, respectively;

2) L_A stores outputs of random oracle queries;

3) list L_T records transcripts between U_i and S.

Due to indistinguishability of games G_0 and G_1, we have

$$Pr[Succ_1] = Pr[Succ_0], \quad (6)$$

G_2 : In this game, we consider the collision situation with hash results and random numbers in the transcripts of M_1,M_2.M_3. The collision probability of H_1 query and H_2

TABLE 4. Simulation of hash, reaveal, test, corrupt, execute and send oracle queries.

$Hash\ H_1$ simulation query performs as follows: If the record $(m_i, H_1$ is found in list L_{H_1} corresponding to hash query $H_1(m_i)$, return H. Otherwise, select a number $H \in Z_p^*$ and add (m_i, H) into L_{H_1}. If the query is initiated by $\mathcal{A}$, (m_i, H) is stored in $L_{\mathcal{A}}$.
$Hash\ H_2$ simulation query performs as follows: If the record (m_i, H_2) is found in table list L_{H_2} corresponding to hash query $H_2(m_i)$, return H_2. Otherwise, select a string $H_2 \in \{0,1\}^{l_{H_2}}$ and add (m_i, H_2) into L_{H_2}. If the query is initiated by $\mathcal{A}$, (m_i, H_2) is stored in $L_{\mathcal{A}}$.
$Reveal(C)$ simulation query perform as follows: If C is in accept state, the current session key SK form by C is returned.
$Test(C)$ simulation query perform as follows: From $Revael(C)$ query, $\mathcal{A}$ obtain current session SK though flip a unbiased coin b. If b=1, return SK. Otherwise, return a random string from $\{0,1\}^*$.
$Corrupt(U_i, a)$ simulation query perform as follows: If $a = 1$, the query returns password PW_i. If $a = 2$, the query returns biometric key θ_i of the user. If $a = 3$, the query returns U_i's smart device stored parameters.
Simulation of $Execute(U_i, S)$ query occurs in succession as shown below. According to table 3, $< TID, T_{s_1}, PA, e_a > \leftarrow Send(U_i, start)$, $< K_{is} \oplus < PS, w_s >, N_j, T^*_{s_2} > \leftarrow Send(S, < TID, T, T_{s_1}, PA, N_i >)$, and $< K^*_{is} \oplus w_a > \leftarrow Send(U_i, < K_{is} \oplus < PS, w_s >, N_j, T^*_{s_2} >)$ Finally, $M_1 =< TID, T_{s_1}, PA, N_i >$,$M_2 =< K_{is} \oplus < PS, w_s >, N_j, T^*_{s_2} >$ and $M_3 =< w_a \oplus K^*_{is} \oplus T_{s_3}, T_3 >$ are returned.
$Send(C, m)$ simulation query performs as follows: Let U_i be the target state. For a $Send(U_i, start)$ query, U_i gives the following response. Computes x_i, X_i, PA as Table 3 and outputs $M_1 =< TID, T_{s_1}, PA, N_i >$ Let S be the target state. For a $Send(U_i, < TID, T_{s_1}, PA, N_i >)$ query, S gives the following response. Verifies if $\lvert T_{s_1} - T^*_{s_1} \rvert < \triangle T$ and executes zero-knowledge proof for S. Computes K_{is}, PS, w_s as Table 3 and outputs $M_2 =< K_{is} \oplus < PS, w_s >, N_j, T^*_{s_2} >$. Let U_i be the target state. For a $Send(U_i, < K_{is} \oplus < PS, w_s >, N_j, T^*_{s_2} >)$ query, S gives the following response. verifies if $\lvert T_{s_2} - T^*_{s_2} \rvert < \triangle T$ and computes SK, w_a as Table 3 and outputs $M_3 =< w_a \oplus K^*_{is} \oplus T_{s_3}, T_3 >$. S answer $Send(K^*_{is} \oplus w_a)$ query as follows. computes SK and authenticates zero-knowledge proof for U_{is}. If it is incorrect, terminates the session. Otherwise establishes SK as the session key. Finally, both U_i and S accept the successful termination of session.

query are at most $\frac{q_{H_1}}{2^{l_{H_1}+1}}$ and $\frac{q^2_{H_2}}{2^{l_{H_2}+1}}$, respectively. Messages M_1,M_2.M_3 contain random number p_a,e_a,p_s and e_s, and the most probability of collision for these numbers collision is at most $(\frac{(q_s+q_e)^4}{2^{l_r}+1})$. So, we have

$$Pr[Succ_2] - Pr[Succ_1]| \leq \frac{q_{H_1}}{2^{l_{H_1}+1}} + \frac{q^2_{H_2}}{2^{l_{H_2}+1}} + (\frac{(q_s+q_e)^4}{2^{l_r}+1}). \tag{7}$$

G_3 : In this game, $\mathcal{A}$ obtains the correct message without active participation of hash oracles. Hence, we consider the following three cases.

C_1 : First, Considering $Send(S, M_1)$ query. The maximum calculated probability of hash value.

$H_1(ID_i||PW_i||\theta_i||r_i||T)$ is $\frac{q_{H_2}}{2^{l_{H_2}}}$. For the random number e_a and p_a, we have the maximum probability for this as $\frac{q_s^2}{2^{l_r}}$.

C_2 : Then, we consider $Send(U_i, M_2)$. The maximum calculated probability of hash value $H(T_s||T_{s_2}||e_a||e_s||T_{x_s}(PA))$ is $\frac{q_{H_2}}{2^{l_{H_2}}}$. For the random number e_s and p_s, we get it maximum probability $\frac{q_s^2}{2^{l_r}}$.

C_3 : Finally, we consider $Send(S, M_3)$. The hash value $K^*_{is} = H(T_s||T^*_{s_2}||e_a||e_s||T_{pa}(X_i))$ and $x_i = H_1(ID_i||PW_i||\theta_i||r||T)$ hold with probability $\frac{q_{H_2}}{2^{l_{H_2}}}$ and $\frac{q_{H_1}}{2^{l_{H_1}}}$, respectively.

Considering three cases, we have

$$|Pr[Succ_3] - Pr[Succ_2]| \leq \frac{q_{H_1}}{2^{l_{H_1}}} + 3\frac{q_{H_2}}{2^{l_{H_2}}} + 2\frac{q_s^2}{2^{l_r}}. \quad (8)$$

G_4 : In this game, we consider mainly guessing attacks executed by $\mathcal{A}$.

C_1 : $\mathcal{A}$ executes $Corrupt(U_i, 1)$ to guess PW. The probability of this case is $\frac{q_s}{|D|}$.

C_2 : $\mathcal{A}$ executes $Corrupt(U_i, 2)$ to simulate the intentional or accidental guessing of user biometrics key θ_i. The probability of this case is at most $\{q_s(\frac{1}{2^{l_b}}, \varepsilon_{bm})\}$

C_3 : We consider that $\mathcal{A}$ guesses the session key without active involvement of oracle H_1 and H_2. Due to the SK is computed with hash of two chaotic map $T_{p_a}(M)$ and $T_{x_s}(PA)$. Hence, the probability for this case is at most $2q_{H_2}Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}})$.

C_4 : $\mathcal{A}$ guesses the zero-knowledge proof parameters w_s and w_a in this case. From the perspective of $\mathcal{A}$, w_s and w_a are like random number. So, for this case, the probability is at most $2\frac{q_s}{2^{l_r}}$.

We can conclude that the games G_3 and G_4 are indistinguishable. So, we obtain

$$|Pr[Succ_4] - Pr[Succ_3]| \leqslant max\{q_s(\frac{1}{|D|}, \frac{1}{2^{l_b}}, \frac{2}{2^{l_r}}, \varepsilon_{bm})\} + 2q_{H_2}Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}). \quad (9)$$

G_5 : This game consider strong forward security. $\mathcal{A}$ executes *Execute*, *Send*, *Hash* oracle queries on old transcripts only to break forward security. To avoid termination of the game, the *Test* query should returns the real session key for instance of U_i and S. Following the analysis of G_4, we have

$$|Pr[Succ_5] - Pr[Succ_4]| \leqslant 2q_{H_2}(q_s + q_e)^2 Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}) \quad (10)$$

Considering all above games G_i, $i = 0, 1, 2, 3, 4, 5$, $\mathcal{A}$ gains no advantage to guess the correct bit b, we get,

$$Pr[Succ_5] = \frac{1}{2}.$$

Using the triangular inequality, we have the following:

$$\begin{aligned} |Pr[Succ_0] - \frac{1}{2}| &= |Pr[Seccess_1] - Pr[Seccess_5]| \\ &\leqslant |Pr[Seccess_1] - Pr[Seccess_2]| \\ &\quad + |Pr[Seccess_2] - Pr[Seccess_5]| \\ &\leqslant |Pr[Seccess_1] - Pr[Seccess_2]| \\ &\quad + |Pr[Seccess_2] - Pr[Seccess_3]| \\ &\quad + |Pr[Seccess_3] - Pr[Seccess_4]| \\ &\quad + |Pr[Seccess_4] - Pr[Seccess_5]|. \end{aligned} \quad (11)$$

According to the results of each game, we have:

$$\begin{aligned} \frac{1}{2}Adv_C^{AKA} &= |Pr[Succ_0] - \frac{1}{2}| \\ &\leqslant \frac{q_{H_1}}{2^{l_{H_1}+1}} + \frac{q_{H_2}^2}{2^{l_{H_2}+1}} + (\frac{(q_s+q_e)^4}{2^{l_r+1}}) \\ &\quad + \frac{q_{H_1}}{2^{l_{H_1}}} + 3\frac{q_{H_2}}{2^{l_{H_2}}} + 2\frac{q_s^2}{2^{l_r}} \\ &\quad + max\{q_s(\frac{1}{|D|}, \frac{1}{2^{l_b}}, \frac{2}{2^{l_r}}, \varepsilon_{bm})\} \\ &\quad + 2q_{H_2}Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}) \\ &\quad + 2q_{H_2}(q_s + q_e)^2 Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}) \end{aligned} \quad (12)$$

Here, we have the required result:

$$\begin{aligned} Adv_C^{AKA} &\leqslant \frac{3q_{H_1}}{2^{l_{H_1}}} + \frac{q_{H_2}^2 + 6q_{H_2}}{2^{l_{H_2}}} \\ &\quad + (\frac{(q_s+q_e)^4 + 4q_s^2}{2^{l_r+1}}) \\ &\quad + 2max\{q_s(\frac{1}{|D|}, \frac{1}{2^{l_b}}, \frac{2}{2^{l_r}}, \varepsilon_{bm})\} \\ &\quad + 4q_{H_2}(1 + (q_s+q_e)^2) Adv_{\mathcal{A}}^{CMDLP}(t_{\mathcal{A}}) \end{aligned} \quad (13)$$

Hence, the theorem is proved.

B. AUTHENTICATION PROOF USING BAN LOGIC

The BAN logic is widely used for mutual authentication analyzing between the user and server [19]. In this section, we use BAN logic to demonstrate how the proposed scheme achieves the authentication goals. Basic BAN logic notations are defined as follows:

- $P \mid\equiv X$: P believes X;
- $P \triangleleft X$: P sees X;
- $\#(X)$: X is fresh;
- $P \Rightarrow X$: P has jurisdiction over X;
- $P \mid\sim X$: P once said X;
- X_K : X is encrypted with the key K;
- $\langle X \rangle_Y$: X combined with Y;
- $P \leftrightarrow []KQ$: P and Q know the key K and use it to communicate.
- $P \overset{X}{\rightleftharpoons} Q$: P and Q use X to prove their identities to on another.

SK: The session key used in the current session.

The main rules of the BAN logic are given below

Rule 1 (Message-meaning rule):

$$\frac{P \mid\equiv P \overset{K}{\leftrightarrow} Q, P \triangleleft \{X\}_K}{P \mid\equiv Q \mid\sim X} \quad \text{and} \quad \frac{P \mid\equiv P \overset{Y}{\rightleftharpoons} Q, P \triangleleft \{X\}_Y}{P \mid\equiv Q \mid\sim X}.$$

Rule 2 (Nonce-verification rule):

$$\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}.$$

Rule 3 (Jurisdiction rule):

$$\frac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}.$$

Rule 4 (Freshness-conjuncatenation rule):

$$\frac{P \mid\equiv \#(X)}{P \mid\equiv \#(X, Y)}.$$

Rule 5 (Additional rule):

$$\frac{P \mid\equiv (X, Y)}{P \mid\equiv X}, \quad \frac{P \triangleleft (X, Y)}{P \triangleleft X}, \quad \frac{P \mid\equiv Q \sim (X, Y)}{P \mid\equiv Q \sim X}.$$

According to the analytic procedure requirement of BAN logic, the proposed scheme must satisfy the following test goals:

$$G1 : U_i \mid \equiv (U_i \overset{SK}{\leftrightarrow} S).$$
$$G2 : S \mid \equiv (U_i \overset{SK}{\leftrightarrow} S).$$

The generic form of all the messages are given below:
Message 1 M_1. $U_i \rightarrow S$:

$$\{TID, T, T_{s_1}, PA, N_i\}.$$

Message 2 M_2. $S \rightarrow U_i$:

$$\{K_{is} \oplus < PS, w_s >, N_j, T^*_{s_2}\}.$$

Message 3 M_3. $U_i \rightarrow S$:

$$\{w_a \oplus K^*_{is} \oplus T_{s_3}, T_3\}.$$

The idealized forms are as follows:
Message 1 M_1. $U_i \rightarrow S$:

$$\{TID, T, T_{s_1}, PA_{X_i}, \langle e_a, T_{s_1}\rangle_{X_i}\}.$$

Message 2 M_2. $S \rightarrow U_i$:

$$\{\langle PS, w_s\rangle_{K_{is}}, \langle e_s, T^*_{s_2}\rangle_{X_i}, T^*_{s_2}\}.$$

Message 3 M_3. $U_i \rightarrow S$:

$$\{\langle w_a, T_{s_3}\rangle_{K^*_{is}}, T_{s_3}\}.$$

The basic assumptions are as follows:
1: $U_i \mid \equiv \#(T^*_{s_2})$
2: $S \mid \equiv \#(T_{s_1})$
3: $U_i \mid \equiv (U_i \overset{X_i}{\leftrightarrow} S)$
4: $S \mid \equiv (U_i \overset{X_i}{\leftrightarrow} S)$
5: $S \mid \equiv \#(T_{s_3})$
6: $S \mid \equiv U_i \Rightarrow (T_{s_1}, PA, N_i)$
7: $S \mid \equiv U_i \Rightarrow (w_a, T_{s_3})$
8: $U_i \mid \equiv S \Rightarrow (e_s, T^*_{s_2})$
9: $U_i \mid \equiv S \Rightarrow (PS, w_s)$
10: $S \mid \equiv x_s$
11: $S \mid \equiv e_s$
12: $S \mid \equiv T^*_{s_2}$
13: $S \mid \equiv w_s$
14: $U_i \mid \equiv x_i$
15: $U_i \mid \equiv e_a$
16: $U_i \mid \equiv T_{s_1}$
17: $U_i \mid \equiv T_{s_3}$
18: $U_i \mid \equiv w_a$
19: $U_i \mid \equiv M$

To achieve the goals G_1 and G_2, the main procedures of our proof are stated as follows:

From message 1, we have,

(G_2)

S_1 : $S \triangleleft \{TID, T, T_{s_1}, PA_{X_i}, \langle e_a, T_{s_1}\rangle_{X_i}\}$.

S_2 : According to AL, we obtain, $S \triangleleft \{\langle e_a, T_{s_1}\rangle_{X_i}, PA_{X_i}\}$.

S_3 : According to 4 and MML, we obtain, $S \mid \equiv U_i \mid\sim (T_{s_1}, PA, N_i)$.

S_4 : According to 2 and FCL, we obtain, $S \mid\equiv \#(T_{s_1}, PA, N_i)$.

S_5 : According to NVL, we have, $S \mid\equiv U_i \mid\equiv (T_{s_1}, PA, N_i)$.

S_6 : Using 6 and JL, we get, $S \mid\equiv (T_{s_1}, PA, N_i)$

S_7 : From S_6 and AL, we obtain, $S \mid\equiv T_{s_1}$, $S \mid\equiv PA$, $S \mid\equiv N_i$.

S_8 : According to 10,11, 12, we get, $S \mid\equiv x_s$, $S \mid\equiv e_s$, $S \mid\equiv T^*_{s_2}$.

S_9 : Since $K_{is} = H(T_s||T^*_{s_2}||e_a||e_s||T_{x_s}(PA))$ and the results in Steps S_7 and S_8 give $U_i \mid\equiv (U_i \overset{K_{is}}{\longleftrightarrow} S)$.

From message 3, we obtain

S_{10} : $S \triangleleft \{\langle w_a, T_{s_3}\rangle_{K^*_{is}}, T_{s_3}\}$.

S_{11} : From S_9, $K^*_{is} = K_{is}$ and MML, we get, $S \mid\equiv U_i \mid\sim (w_a, T_{s_3})$.

S_{12} : According to 5 and FCL, we obtain, $S \mid\equiv \#(w_a, T_{s_3})$.

S_{13} : According to NVL, we have, $S \mid\equiv U_i \mid\equiv (w_a, T_{s_3})$.

S_{14} : Using 7 and JL, we get, $S \mid\equiv (w_a, T_{s_3})$.

S_{15} : From S_{14} and AL, we obtain, $S \mid\equiv w_a$, $S \mid\equiv T_{s_3}$.

S_{16} : According S_7, S_8, S_{15} and

$SK = H(T_{s_1}||T^*_{s_2}||T_{s_3}||w_a||w_s||T_{x_s}(PA)))$,

we obtain, $S \mid\equiv (U_i \overset{SK}{\longleftrightarrow} S)$.

(G_1)

S_{17} : $U_i \triangleleft \{\langle PS, w_s\rangle_{K_{is}}, \langle e_s, T^*_{s_2}\rangle_{X_i}, T^*_{s_2}\}$.

S_{18} : According to AL, we obtain, $U_i \triangleleft \{\langle e_s, T^*_{s_2}\rangle_{X_i}\}$.

S_{19} : According to 4 and MML, we obtain, $U_i \mid S \equiv\mid\sim (e_s, T^*_{s_2})$.

S_{20} : According to 1 and FCL, we obtain, $u_i \mid\equiv \#(e_s, T^*_{s_2})$.

S_{21} : According to NVL, we have, $U_i \mid\equiv S \mid\equiv (e_s, T^*_{s_2})$.

S_{22} : Using 8 and JL, we get, $U_i \mid\equiv (e_s, T^*_{s_2})$

S_{23} : From S_{22} and AL, we obtain, $U_i \mid\equiv e_s$, $U_i \mid\equiv T^*_{s_2}$.

S_{24} : According to 15, 16, 19, and since $K^*_{is} = H(T_{s_1}||T^*_{s_2}||e_a||e_s||T_{pa}(X_i)) = K_{is}$ and the results in Steps S_{22} and S_{23}, we obtain $U_i \mid\equiv (U_i \overset{K_{is}}{\longleftrightarrow} S)$.

S_{25} : $U_i \triangleleft \{\langle PS, w_s\rangle_{K_{is}}\}$.

S_{26} : From S_{24}, $K^*_{is} = K_{is}$ and MML, we get, $U_i \mid\equiv S \mid\sim (PS, w_s\rangle_{K_{is}})$.

S_{27} : According to S_{24}, we obtain, $S \mid\equiv \#(PS, w_s)$.

S_{28} : According to NVL, we have, $U_i \mid\equiv S \mid\equiv (PS, w_s)$.

S_{29} : Using 9 and JL, we get, $S \mid\equiv (PS, w_s)$.

S_{30} : From S_{29} and AL, we obtain, $S \mid\equiv PS$, $S \mid\equiv w_s$.

S_{31} : According S_{30}, S_{23}, 16, 17, 18, 19 and $SK = H(T_{s_1}||T^*_{s_2}||T_{s_3}||w_a||w_s||T_{pa}(M})))$, we obtain $U_i \mid\equiv (U_i \overset{SK}{\longleftrightarrow} S)$.

As a result, (G_1) and (G_2) ensure that both U_i and S mutually authenticate each other.

C. SECURITY VERIFICATION BASED ON SIMULATION TOOL

We use a popular security verification simulation tool, ProVerif, to show several security properties. ProVerif [18] is an automatic cryptographic protocol verifier, in the formal model (so called Dolev-Yao model). This protocol verifier is based on a representation of the protocol by Horn clauses.

```
-- Query inj-event(Server_AuthEnd(sid)) ==> inj-eve
nt(Server_AuthStart(sid))
Completing...
Starting query inj-event(Server_AuthEnd(sid)) ==> i
nj-event(Server_AuthStart(sid))
RESULT inj-event(Server_AuthEnd(sid)) ==> inj-event
(Server_AuthStart(sid)) is true.
-- Query inj-event(User_AuthEnd(uid)) ==> inj-event
(User_AuthStart(uid))
Completing...
Starting query inj-event(User_AuthEnd(uid)) ==> inj
-event(User_AuthStart(uid))
RESULT inj-event(User_AuthEnd(uid)) ==> inj-event(U
ser_AuthStart(uid)) is true.
-- Query not attacker(SKuser[]); not attacker(SKser
ver[])
Completing...
Starting query not attacker(SKuser[])
RESULT not attacker(SKuser[]) is true.
Starting query not attacker(SKserver[])
RESULT not attacker(SKserver[]) is true.
liuwenzhengdeMacBook-Pro:chao_zerok_scheme liuwenzh
eng$ Proverif 2.00 result
```

FIGURE 5. ProVerif 2.00 simulation result of our scheme.

By using Proverif 2.00 to simulate the login, authentication and key agreement phase for user U_i and server S, we get the following results of mutual authentication and session key secrecy(Figure 5):

- RESULT inj-event(Server_AuthEnd(sid)) ==> inj-event(Server_AuthStart(sid)) is true.
- RESULT inj-event(User_AuthEnd(uid)) ==> inj-event(User_AuthStart(uid)) is true.
- RESULT not attacker(SKuser[]) is true.
- RESULT not attacker(SKserver[]) is true.

Hence, our scheme passed the ProVerif 2.00 security verification.

D. SECURITY ANALYSIS FOR OTHER VARIOUS ATTACKS

In this section, we give additional security analysis to show that our scheme can withstand the following various attacks.

1) REPLAY ATTACK

In the proposed scheme, S ignores the message if $|T_s - T_s^*| > \triangle T$ and stores the pair $(ID_i, T_{pa}(X_i))$ to protect the scheme from strong replay attack.

2) PASSWORD GUESSING ATTACK

To get user U_i's identity factor, ID_i, PW_i, or biometric B_i, an adversary needs to guess them all simultaneously. The property of the hash function makes it hard to execute a password guessing attack.

3) STOLEN VERIFIER ATTACK

By executing this attack, the adversary can access the user's verification information stored at the server database. In our scheme, the server only stores $< ID_i, T, T_{x_i}(X_i), X_i >$ for each user U_i. It does not store any sensitive information for authentication. Moreover, adversaries cannot pass the verification of zero-knowledge proof since they don't have the password PW_i and biometric B_i of user U_i.

4) STOLEN SMART CARD OR MOBILE DEVICE ATTACK

If adversaries steal the smart card or the mobile device of users and extract the information stored in it, they still cannot pass the authentication. Because there has some important verified information need imprint from the user when the authentication begins, such as password PW_i and biometric B_i.

5) PRIVILEGED INSIDER ATTACK

In this attack, we assume that the registration information $< T_{x_i}(X_i), T, X_i, ID_i >$ is known to an adversary. It is also assumed that $\mathcal{A}$ obtains the information stored in the smart device. It is also computationally difficult task for $\mathcal{A}$ to get PW and biometric key θ_i from stored information $< M, r_i, \sigma_i >$. Hence, our scheme can resist privileged insider attack.

6) KNOWN SESSION KEY SECRECY

According to the login, authentication and key agreement phase, the session key is computed as $SK = H_2(T_{s_1}||T_{s_2}^*||T_{s_3}||w_a||w_s||T_{p_a}(M)))$. Due to the use of $T_{s_1}, T_{s_2}^*, T_{s_3}, w_a, w_s, SK$ is generated in random. Hence, the adversary cannot obtains crucial information from the previous session key.

7) USER IMPERSONATION ATTACK

An adversary needs to input ID_i, PW_i, and B_i to impersonate a legal user. It is computationally difficult task for $\mathcal{A}$ to guess these identity factors.

8) SERVER IMPERSONATION ATTACK

An adversary cannot impersonate a server unless he provide $w_s = p_s + x_s e_a$ at a session, which need obtain the server master secret key x_s and two random numbers p_s and e_s. As a consequence, our scheme free from server impersonation attack.

9) SERVER-INSIDER ATTACK

In this attack, the adversary is the server internal staff and he can obtains x_s and user's verification information stored in the server. The adversary still cannot do whatever he wants in our scheme. Because the authentication process of our scheme needs to verify the zero-knowledge proof of user, while this secret is only can be obtained by the user himself. The adversary cannot impersonate any user even if he gets the server's master key.

10) MAN-IN-THE MIDDLE ATTACK

The adversary may try to modify message M_1, M_2, M_3 or establish independent connection with U_i and S. However,

TABLE 5. Comparison with the previous related proposed schemes.

security attributes	Xu et al. [10]	Moon [6]	Chain [4]	Roy [1]	Our scheme
Stolen smart card or mobile device attack	√	√	×	√	√
Replay attack	×	×	√	√	√
Password guessing attack	√	√	√	√	√
Privileged insider attack	√	√	√	√	√
Known session key secrecy	√	√	√	√	√
Session key security	√	√	√	√	√
User impersonation attack	√	√	√	√	√
Server impersonation attack	√	√	√	√	√
Server-insider attack	×	×	×	×	√
Revocation of smart device	×	×	×	√	√
Secure mutual authentication	√	√	√	√	√
Biometric remote authenticate	×	×	×	×	√
Password remote authenticate	×	×	×	×	√
Formal security analysis	×	√	×	√	√
Strong secure secret key	×	×	√	×	√

TABLE 6. Comparison of computation and communication costs.

	Xu et al. [10]	Moon [6]	Chain [4]	Roy [1]	Our scheme
U_i computation cost	$5T_h + 3T_m$ $\approx 10ms$	$4T_H + 2T_{ch}$ $\approx 4ms$	$10T_{ch}$ $\approx 17ms$	$9T_H + 1T_{Fe} + 2T_{ch}$ $\approx 5ms$	$T_{Fe} + 4T_{ch} + 4T_H$ $\approx 7ms$
S computation cost	$5T_h + 3T_m$ $\approx 6ms$	$10T_H + 2T_{ch}$ $\approx 2ms$	$10T_{ch}$ $\approx 9ms$	$5T_h + 1T_{ch}$ $\approx 1ms$	$4T_{ch} + 2T_H$ $\approx 3ms$
Communication rounds	3	4	4	2	3

an adversary cannot modify or regenerate any of the sent parameters as the message contains the hash value. Hence, our scheme can resist this attack.

11) STRONG SECURE SECRET KEY

In our scheme, authentication factors such as ID, PIN code, and biometric are part of the secret, and the server directly authenticates user's identity factors. In the login, authentication and key agreement phase, all the identity factors authenticated by the server and participate in key agreement. Hence, the proposed scheme has strong secure secret key.

V. PERFORMANCE COMPARISON

In this section, we discuss the efficiency of our proposed scheme and compare it with four proposed related existing schemes Xu [10] Moon [6] Chain [4] Roy [1] .

A. COMPARISON ON FUNCTIONALITY AND SECURITY

We make a table (Table 5) to show the detailed comparison of various security attacks and functions. Most of related schemes failed to provide biometric and password remote authenticate and suffer from server-insider attack. It is observed that our scheme not only gives the support of much more functionality but also overcomes more security weaknesses..

B. COMPARISON ON COMPUTATION AND COMMUNICATION COST

In this paper, we choose mobile phone Xiaomi 6 as a smart device for the user side and macbook pro 2014 15.4 with Intel i7 4770hq processor for the server side, respectively.

TABLE 7. Execution timings of various cryptographic operations.

Term	Description
T_{sym_ed}	Symmetric key encryption
T_h	One way hash function
T_m	Elliptic curve point multiplication
T_{Fe}	Fuzzy extractor operation
T_{ch}	Chebyshev polynomial computation

Xiaomi 6 has maximum clock speed of 2.45 GHz, 64 GB flash memory and 6 GB RAM equipped, and Android 9.0 installed. The macbook pro 2014 15.4 has maximum clock speed of 3.4GHz, with MAC OS and 16 GB RAM. We use C language under specific IDE and C/C++ MIRACL Library to implement all the cryptographic operations.

We have not considered the costs of the registration and password, biometric change and smart card or device revocation process since it only runs a limited number of times. Therefore, we consider the communication, computation cost of the login, authentication, and key agreement phase.

Table 6 compares the computational costs and communication rounds in login, authentication and key agreement phase of our proposed scheme and Xu [10] Moon [6] Chain [4] Roy [1]. Table 7) shows different notations. We study that the total user side computation overhead required for a user in our scheme is $T_{Fe}+4T_{ch}+4T_h$. According to the experiment, the average executing time is approximately 7 ms. While the server S need $4T_{ch} + 2T_H$, and the average executing time is approximately 3ms. Then we simulated a large number of crowdsourcing IoT users accessing server and recorded the time spent from 200 to 1000 users but without communication

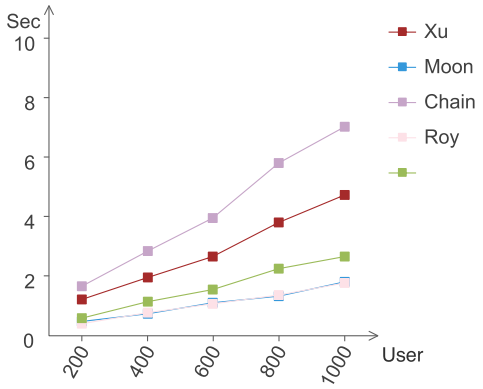

FIGURE 6. Time consumption without communication delay.

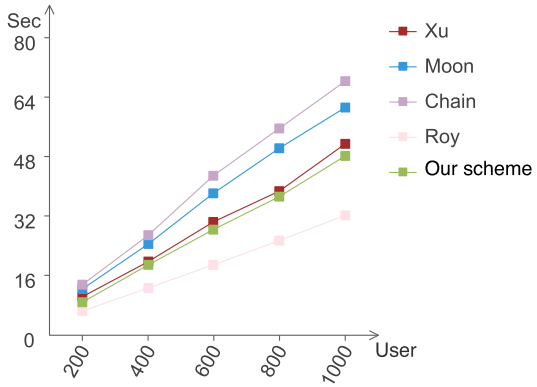

FIGURE 7. Time consumption at 4G wireless communication.

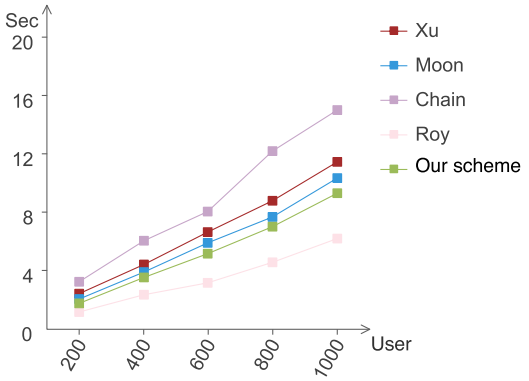

FIGURE 8. Time consumption at simulated ideal 5G wireless communication.

delay, which result shows in Figure 6. According to the experiment result, our scheme's executing time is nearly half of [10] and Chain [4] scheme, and for lightweight scheme Moon [6] and Roy [1], it also does not add much executing time.

For communication overhead, we did another experiment with the same experimental conditions, but this time we consider the communication delay, which result shows in Figure 7. Compare with the experiment without communication delay, we find that the time delay caused by communication delay is much higher than the time loss caused by the cryptographic calculation. In the next generation 5G communication environment, communication delay will be greatly improved. Therefore, we give a test of time consumption at simulated ideal 5G communication delay. From the Figure 8, we can see that the efficiency of our scheme has greatly increased and exceeded that of lightweight scheme [6].

VI. CONCLUSION

We have designed a secure, lightweight, and remote multi-factor authentication based on chaotic map zero-knowledge proof for application of crowdsourcing IoT. In the proposed scheme, In this scheme, the server no longer authenticates the secret key stored at the user's smart device client, but directly authenticates the user's authentication factor. All authentication factors act as a part of the secret key and participate in the procedure of authentication and key agreement. By using the RoR mod and BAN logic for formal security analysis and give an additional security analysis for other various attacks, we show that our scheme is secure from various attacks. Finally, according to the test and simulation, we show that our scheme has low computational and communication overhead, which is suited for the users with power-constrained smart devices and will be greatly enhanced in the next-generation 5G communication environment.

Future works: We are working on promoting our authentication scheme in the multi-server environment.

REFERENCES

[1] S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, S. Kumari, and M. Jo, "Chaotic map-based anonymous user authentication scheme with user biometrics and fuzzy extractor for crowdsourcing Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2884–2895, Aug. 2018.

[2] 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. Depend. Sec. Comput.*, vol. 15, no. 5, pp. 824–839, Sep. 2018.

[3] X. Jia, D. He, L. Li, and K.-K.-R. Choo, "Signature-based three-factor authenticated key exchange for Internet of Things applications," *Multimed Tools Appl.*, vol. 77, no. 14, pp. 18355–18382, Jul. 2018.

[4] K. Chain, K.-H. Chang, W.-C. Kuo, and J.-F. Yang, "Enhancement authentication protocol using zero-knowledge proofs and chaotic maps: Enhancement authentication protocol," *Int. J. Commun. Syst.*, vol. 30, no. 1, Jan. 2017, Art. no. e2945.

[5] A. Ghezzi, D. Gabelloni, A. Martini, and A. Natalicchio, "Crowdsourcing: A review and suggestions for future research," *Int. J. Manage. Rev.*, vol. 20, no. 2, pp. 343–363, Apr. 2018.

[6] J. Moon, Y. Choi, J. Kim, and D. Won, "An improvement of robust and efficient biometrics based password authentication scheme for telecare medicine information systems using extended chaotic maps," *J. Med. Syst.*, vol. 40, no. 3, p. 70, 2016.

[7] X. Hao, J. Wang, Q. Yang, X. Yan, and P. Li, "A chaotic map-based authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 37, no. 2, p. 9919, Apr. 2013.

[8] P. Bergamo, P. D'Arco, A. De Santis, and L. Kocarev, "Security of public-key cryptosystems based on Chebyshev polynomials," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 7, pp. 1382–1393, Jul. 2005.

[9] L. Zhang, "Cryptanalysis of the public key encryption based on multiple chaotic systems," *Chaos, Solitons Fractals*, vol. 37, no. 3, pp. 669–674, Aug. 2008.

[10] X. Xu, P. Zhu, Q. Wen, Z. Jin, H. Zhang, and L. He, "A secure and efficient authentication and key agreement scheme based on ECC for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 1, p. 9994, 2013.
[11] Q. Xie, W. Liu, S. Wang, L. Han, B. Hu, and T. Wu, "Improvement of a uniqueness-and-anonymity-preserving user authentication scheme for connected health care," *J. Med. Syst.*, vol. 38, no. 9, pp. 91, 2014.
[12] X. Yan, W. Li, P. Li, J. Wang, X. Hao, and P. Gong, "A secure biometrics-based authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 37, no. 5, p. 9972, Oct. 2013.
[13] D. Mishra, S. Mukhopadhyay, S. Kumari, M. K. Khan, and A. Chaturvedi, "Security enhancement of a biometric based authentication scheme for telecare medicine information systems with nonce," *J. Med. Syst.*, vol. 38, no. 5, p. 41, 2014.
[14] Z. Tan, "A user anonymity preserving three-factor authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 3, p. 16, Mar. 2014.
[15] L. Kocarev and S. Lian, *Chaos-Based Cryptography: Theory, Algorithms and Applications*. Berlin, Germany: Springer, 2011.
[16] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. Int. Conf. Theory Pract. Public Key Cryptogr.* New York, NY, USA: Springer-Verlag, 2005.
[17] D. Stebila, P. Udupi, and S. C. Shantz, "Multi-factor password-authenticated key exchange (full version)," Tech. Rep., 2008, p. 29.
[18] M. Abadi, B. Blanchet, and H. Comon-Lundh, *Models and Proofs of Protocol Security: A Progress Report*, in *Computer Aided Verification*. Berlin, Germany: Springer, 2009.
[19] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *SIGOPS Oper. Syst. Rev.*, vol. 23, no. 5, pp. 1–13, Nov. 1989.
[20] L. Da Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.
[21] T.-F. Lee, "An efficient chaotic maps-based authentication and key agreement scheme using smartcards for telecare medicine information systems," *J. Med. Syst.*, vol. 37, no. 6, p. 9985, 2013.
[22] C. T. Li, C. C. Lee, and C. Y. Weng, "A secure chaotic maps and smart cards based password authentication and key agreement scheme with user anonymity for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 9, p. 77, 2014.
[23] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.*, Interlaken, Switzerland, May 2004, pp. 523–540.
[24] A. K. Das and B. Bruhadeshwar, "An improved and effective secure password-based authentication and key agreement scheme using smart cards for the telecare medicine information system," *J. Med. Syst.* vol. 37, no. 5, p. 9969, 2013.
[25] C.-C. Lee, T.-H. Lin, and R.-X. Chang, "A secure dynamic ID based remote user authentication scheme for multi-server environment using smart cards," *Expert Syst. Appl.*, vol. 38, no. 11, pp. 13863–13870, 2011.
[26] C.-T. Li and C.-C. Lee, "A novel user authentication and privacy preserving scheme with smart cards for wireless communications," *Math. Comput. Model.*, vol. 55, nos. 1–2, pp. 35–44, Jan. 2012.
[27] V. Odelu, A. K. Das, S. Kumari, X. Huang, and M. Wazid, "Provably secure authenticated key agreement scheme for distributed mobile cloud computing services," *Future Gener. Comput. Syst.*, vol. 68, pp. 74–88, Mar. 2017.
[28] S. H. Islam, "A provably secure ID-based mutual authentication and key agreement scheme for mobile multi-server environment without ESL attack," *Wireless Pers. Commun.*, vol. 79, no. 3, pp. 1975–1991, Dec. 2014.
[29] B. Zhao, P. Liu, X. Wang, and I. You, "Toward efficient authentication for space-air-ground integrated Internet of things," *Int. J. Distrib. Sensor Netw.*, vol. 15, Jul. 2019, Art. no. 1550147719860390, doi: 10.1177/1550147719860390.
[30] J.-L. Tsai and N.-W. Lo, "A privacy-aware authentication scheme for distributed mobile cloud computing services," *IEEE Syst. J.*, vol. 9, no. 3, pp. 805–815, Sep. 2015.
[31] C. Guo and C.-C. Chang, "Chaotic maps-based password-authenticated key agreement using smart cards," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 6, pp. 1433–1440, Jun. 2013.
[32] Q. Jiang, J. Ma, X. Lu, and Y. Tian, "Robust chaotic map-based authentication and key agreement scheme with strong anonymity for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 2, p. 12, 2014.
[33] Y. Lu, L. Li, H. Peng, D. Xie, and Y. Yang, "Robust and efficient biometrics based password authentication scheme for telecare medicine information systems using extended chaotic maps," *J. Med. Syst.*, vol. 39, no. 65, pp. 1–10, 2015.
[34] J. Shu, "A biometric-based agreement key protocol using extended chaotic maps," *J. Chin. Comput. Syst.*, vol. 280, no. 1750, pp. 867–907, 2015.
[35] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.
[36] V. Odelu, A. K. Das, and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 9, pp. 1953–1966, Sep. 2015, doi: 10.1109/tifs.2015.2439964.
[37] C. P. Schnorr, "Efficient identification and signatures for smart cards," in *Proc. Conf. Theory Appl. Cryptol.*, 1989, pp. 239–252.

WENZHENG LIU received the M.S. degree in applied math from Zhejiang University, in 2016. He is currently pursuing the Ph.D. degree with the College of Computer, National University of Defense Technology. His research interests include applied of identity-based cryptography, the Internet of Thing, stream cipher, financial cryptography, data security, and mobile cloud computing.

XIAOFENG WANG received the Ph.D. degree from the National University of Defense Technology. His current research interests include trusted networks, network security, and distributed intelligent data processing.

WEI PENG was born in Dayi, Sichuan, China, in 1973. He received the M.S. and Ph.D. degrees in computer science from the National University of Defense Technology (NUDT), China, in 1997 and 2000, respectively. From 2001 to 2002, he was a Research Assistant with the School of Computer, NUDT, China, where he has been a Research Fellow, since 2003. His major research interests are the Internet routing, network security, and mobile wireless networks.