IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 30, 2021, accepted January 19, 2022, date of publication January 26, 2022, date of current version February 3, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3146393

A Provably Secure Three-Factor Authentication Protocol Based on Chebyshev Chaotic Mapping for Wireless Sensor Network

JIAQING MO, ZHONGWANG HU, AND WEI SHEN
School of Computer Science and Software, Zhaoqing University, Zhaoqing 526061, China

Corresponding author: Jiaqing Mo (mojiaqing@126.com)

This work was supported in part by the Special Innovation Project for Colleges and Universities in Education Department of Guangdong Province of China 2019 under Grant 2019KTSCX198.

ABSTRACT Wireless sensor network has been widely used and plays a vital role in the Internet of Things, smart cities, military, and other fields, and its security has also attracted the attention of many researchers. In view of the security defects in Shin and Kwon's scheme such as failure to provide three-factor security, lack of anonymity and untraceability, user impersonation attack, desynchronization attack and privileged insider attack, we suggest an improved provably secure three-factor user authentication scheme based on Chebyshev chaotic mapping for wireless sensor network, which employs fuzzy verifier technique to prevent attacker from offline guessing attack on user identity and password when the stolen/lost smartcard is acquired by the attacker. During the authentication phase, a dynamic identity mechanism is used to ensure the anonymity of the user and sensor to prevent desynchronization attack, and the Chebyshev chaotic mapping is introduced to improve security and reduce computation overhead. The rigorous security proof under the random oracle model and the formal verification via ProVerif show that our protocol overcomes the weaknesses in Shin and Kwon's scheme. In addition, by comparing the performance of our proposed scheme with that of others, we demonstrate that our proposal not only solves the security risks of Shin and Kwon.'s protocol, but also achieves a better tradeoff between security and efficiency, therefore, it is more suitable for user authentication in wireless sensor network environments.

INDEX TERMS Authentication protocol, three-factor security, forward secrecy, Chebyshev chaotic mapping, wireless sensor network.

I. INTRODUCTION

With the rapid development of network communication technology, computer intelligence and embedded technology, wireless sensor network (WSN) have been widely used in military surveillance, smart homes, industrial automation, medical care and other important fields, which make information processing in related fields more efficient and intelligent [1]. Unlike traditional network, WSN contains a large number of sensor nodes with environmental awareness and communication capabilities, which are often deployed in unattended environments [2]. The sensor sends the collected data to the remote user for further processing through the gateway via wireless channel. Due to their limited energy and computing capability, deployment in unattended or harsh environments, and data transmission through open wireless channels, wireless sensors face the potential risks of eavesdropping, interception and tampering of the sensitive transmitted information by attackers, resulting in serious consequences such as privacy disclosure. The key technology to solve this problem is to use a user authentication mechanism to make the user and sensor authenticate mutually and negotiate a session key encrypting the sensitive data transmitted between them to prevent unauthorized access from third parties. Because of the scanty resources of wireless sensors, the security technology in traditional networks is difficult to be directly applied to WSN. Devising a secure WSN remote authentication protocol under the condition of scanty resources has become one of the hotspots in the field of WSN. To address these issues, many researchers have presented a considerable number of WSN authentication protocols [3]–[7] to verify the identity of users and negotiate a

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

session key to encrypt the communication data between users and sensors to ensure the security of WSN.

Although the multifactor based authentication protocol for WSN has become a research hotspot and has attracted the attention of industry and academia, it still faces many challenges, such as richer application scenarios, higher security requirements, cryptographic primitives and communication processes that are limited by computing capability and bandwidth, respectively. Thus, it is not easy to analyze the existing authenticated protocols thoroughly to find out their security weaknesses and design a new protocol that can not only overcome these defects but also maintain appropriate efficiency. Recently, Shin and Kwon [42] put forward a three-factor authentication protocol for WSN to overcome the security vulnerabilities in a previous scheme. Although Shin and Kwon claimed that their new protocol eliminated the security flaws in the previous scheme and provided security proof that it is capable of resisting active and passive attacks. Unfortunately, when cryptanalyzing their new three-factor authorization scheme, we still find some serious security flaws, which makes it unsuitable for practical applications.

To thwart the security risks in Shin and Kwon's protocol, we present an improved three-factor WSN authentication protocol. Meanwhile, because chaotic mapping based cryptography requires less computation time and provides higher security than other public key cryptography [43], the improved scheme proposed in this paper utilizes Chebyshev chaotic mapping to solve the problems in Shin and Kwon's scheme and improve its security and efficiency. Our contributions are as follows:

First, we analyze Shin and Kwon's authentication protocol and find that it is unable to provide three-factor security, remains unprotected from user impersonation attack, desynchronization attack, sensor node capture attack, privileged insider attack, and lacks anonymity and untraceability. Second, we propose a secure three-factor authenticated protocol for WSN based on Chebyshev chaotic mapping to surmount the security defects of Shin and Kwon's scheme. Particularly, we make use of the fuzzy verifier technique [32] to thwart password guessing attack effectively. Third, we present the security proof under the random oracle model for our scheme and provide an informal security analysis to demonstrate that our improved protocol can defend against known attacks. In addition, we use the simulation tool ProVerif to rectify the proposal. All of these distinctions indicate that our proposal achieves mutual authentication and session key security. Finally, we assess the improved scheme by comparing with related schemes to show that our scheme acquires a better tradeoff between performance and security requirements.

The rest of our paper is organized as follows: Section II provides an overview of the recent related work. Section III introduces the preliminaries, and Shin and Kwon's protocol is reviewed and cryptanalyzed in Section IV. In Section V, we put forward our improved scheme, and the security analysis is presented in Section VI. Section VII summarizes the security and performance. Finally, the paper is concluded in Section VIII.

II. RELATED WORK

In 2009, Das [8] proposed an authentication scheme based on two factors (password and smartcard) for WSN. Their scheme does not need to maintain the data table of user information in the gateway node, nor does it need to save the information of a specific user on the sensor node. The scheme is mainly implemented by hash function, and claims to have the advantages of low computation cost and can prevent all kinds of network attacks. However, after analyzing Das's scheme, some scholars [9]–[11] find that it has some security vulnerabilities, such as offline password guessing attack, node capture attack, lack of user anonymity, insider attack, and user impersonation attack and so on, and they put forward their improved schemes. It is worth noting that although different authentication protocols have been proposed in the WSN environment as shown in the literatures [12]–[15], [63], Wang *et al.* [16] point out that these schemes and similar protocols based on password and smartcard basically cannot meet the security requirements of WSN and they suffer from smartcard loss attack, which may lead to unexpected serious consequences. In order to better assess the security of the authentication scheme in the WSN environment in the industrial field, Wang *et al.*, also put forward some evaluation criteria [17].

With the development of biotechnology and pattern recognition technology, biometric characteristics (such as fingerprints, iris, face form, etc.) have attracted the attention of an increasing number of researchers because they are not easy to forge, lose or forget, difficult to guess, and so on. In recent years, to further enhance the security of authentication protocol and expand the high security level application of WSN, researchers [18]–[20] introduce biometrics as an additional security factor on the basis of two-factor authentication protocols, and propose three-factor-based authentication protocols in WSN environments. From 2014 to 2015, Das proposed three different WSN authentication schemes [19], [21], [22] based on three factors. However, after analyzing these three schemes, Wu *et al.* [20] find that they can neither resist offline password guessing attack or user impersonation attack, nor ensure forward secrecy. To fix these defects, they presented an enhanced scheme. Unfortunately, the improved scheme of Wu *et al.* does not verify the password's correctness, and the user is declined access to the system until the gateway finds that the user's password is wrong. This will undoubtedly lead to unnecessary consumption of computing resources on the gateway. Lu *et al.* [23] also studied Das *et al.*'s protocol and found that it fails to ensure three-factor security and fails to resist user impersonation attack, and devised an enhanced protocol employing elliptic curve cryptography (ECC). After analyzing Lu *et al.*'s improved protocol, Mo and Chen [24] found that the scheme is still unable to ensure three-factor security, lacks strong key security, and is prone to information disclosure attack; thus, it is not suitable for application in WSN, and they propose an improved protocol to overcome these defects. Unfortunately, Yu and Park [57] showed that Mo and Chen's scheme is susceptible to weaknesses like masquerade attack, session key exposure attack and does not provide anonymity and untraceability. They suggested a new scheme for WSN. Later, Amin *et al.* [25] presented a three-factor WSN authentication protocol based on hash function for temporary information disclosure attack, user counterfeiting attack, and other security threats in Farash *et al.*'s scheme [26]. Jiang *et al.* [27] criticize that Amin *et al.*'s protocol [25] is incapable of withstanding some security risks, such as temporary information disclosure attack, user counterfeiting attack, tracking attack, and further come up with an improved protocol. Because Jiang *et al.*'s improved scheme adopts Rabin public key cryptography, which needs heavy computation, their protocol requires more computation time on the whole.

Recently, Amin *et al.* [28] present an authentication scheme in medical WSN environment, using synchronous update technique to preserve user anonymity and untraceability. The work of their paper has attracted the attention of many scholars. Jiang *et al.* [29] believe that there are some potential security risks of mobile device loss attack, sensor key leakage and desynchronization attack in Amin *et al.*'s protocol [28], and put forward an enhanced scheme to eliminate these shortcomings. Although Jiang et al[29] improved the security of Amin *et al.*'s protocol, Mo *et al.*'s analysis[30] find that their scheme cannot resist some security vulnerabilities, like temporary information disclosure attack, privileged insider attack, and denial of service attack. As a remedy, Mo *et al.* propose a corresponding enhanced version to eliminate these shortcomings. Li et al[31] also prove that Amin *et al.*'s scheme [28] suffers from security limitations such as denial of service attack and lack of forward secrecy. To solve these problems, they put forward an enhanced biometrics-based authentication scheme employing fuzzy verifier [32]. Unfortunately, their scheme cannot prevent replay attack because it does not use timestamps in the transmitted messages.

Independently, Wu *et al.* [61] and Kumar [62] suggested their hash-based authenticated protocol for WSN to secure the communication between user and sensors. However, we analyze Wu *et al.* [61]'s scheme and found that it is susceptible to failure of providing three-factor security, lack of anonymity and information leakage attack. Although Kumar's protocol [62] can effectively resist offline password guessing attack owing to fuzzy verifier technique, it still cannot prevent information leakage attack because there is a dependency between the three random numbers used to calculate session key.

Because public key cryptography techniques like RSA, bilinear pairings, ECC, and chaotic mapping are able to provide higher security, some new authentication protocols try to employ public key cryptography to guarantee the secrecy of session key and resist various attacks. Moghadam *et al.* [33] presented an authentication protocol based on the elliptic curve Diffie-Hellman problem in WSN environment to cope with weaknesses of stolen verifier attack and lack of forward secrecy. Nevertheless, Kwon *et al.* [34] found that Moghadam *et al.*'s protocol is prone to insider attack, session-specific random number leakage attack, and fails to provide forward secrecy, and therefore eliminated these deficiencies with an improved one to eliminate these deficiencies. However, we observe that Kwon *et al.*'s scheme can neither withstand sensor node capture attack and nor ensure three-factor security. Rangwani *et al.* [58] also design an authentication protocol based on ECC for WSN in the industrial Internet of Things circumstances and claimed that their scheme is robust to withstand diverse attacks and surpasses others. In 2021, Xie *et al.* [64] and Jabbari *et al.* [65] suggest an improved lightweight three-factor authentication scheme for WSN using ECC to thwart the security vulnerability in previous scheme, respectively. Nevertheless, we observe that these schemes [58], [64], [65] cannot provide three-factor security, scheme [58] fails to overcome counterfeit attack and replay attack on both the user and the gateway, and the protocol in [65] is insecure against information leakage attack. Moreover, the network model of [64] is unreasonable, because the user communicates directly with the sensor remotely without going through the gateway, which means that the energy of the sensor will be exhausted quickly, as explained in [24]. In 2019, Wang *et al.* [35] proposed a three-factor authentication scheme by using chaotic mapping theory to address the defects in the previous scheme. In the following year, Xu *et al.* [36] put forward an authenticated scheme in medical WSN based on Chebyshev chaotic mapping to improve the efficacy and security and claim that their scheme is more applicable to WSN circumstances. However, our cryptanalysis demonstrates that both of these two protocols are skeptical to be unprotected from some security defects, for instance, neither of them can withstand GWN impersonation attack. Additionally, Xu *et al.*'s scheme cannot withstand user impersonation attack and insider attack.

Although many WSN authentication schemes have made continuous improvements in the previous scheme, they are still found to have some security problems themselves. In 2014, Kim *et al.* [37] designed a user authenticated protocol that claims to be able to defend against user counterfeit attack and gateway node bypass attack. Analysis by Chang *et al.* [38] shows that Kim's scheme cannot resist counterfeiting attack, smartcard lost attack, man-in-the-middle attack, and cannot preserve user privacy. Subsequently, they proposed an improved protocol with dynamic identity technology to overcome these shortcomings. However, their protocol is analyzed separately by Park *et al.* [39] and Jung *et al.* [40] and is found to have weaknesses including offline password guessing attacks, user impersonation attack, and lack of forward secrecy. To enhance the security of the original scheme, Park *et al.* and Jung *et al.* designed an enhanced protocol based on three factors, respectively.

Unfortunately, in 2017, after analyzing these two new protocols, Wang *et al.* [41] found that they are unable to withstand offline dictionary attack, user counterfeit attack, and lack user anonymity and forward secrecy. Afterwards, an enhanced version using ECC algorithm was presented by Wang *et al.* [41], in which its security was proved by BAN logic. Later, Shin and Kwon [42] studied the authenticated protocol presented by Jung *et al.* [40] and argued that it has the following security defects: tracking attack, insecurity of gateway node key, information leakage attack, and user impersonation attack. They correspondingly propose an improved authenticated scheme and remark that their proposal is sufficient to prevent various kinds of attacks, active or passive. However, Shin and Kwon's protocol suffers from some serious weaknesses as cryptanalyzed in Section IV.

According to the previous analysis, designing an authentication scheme with high security is a process that requires continuous in-depth research and analysis of the existing authentication protocols, with proposed reasonable solutions after discovering its security risks.

III. PRELIMINARIES

A. CHEBYSHEV CHAOTIC MAPPING

According to [44], [45], the n degree Chebyshev polynomial is defined as $T_n(x)$: $[-1, 1] \rightarrow [-1, 1]$ and $T_n(x) = cos(narccos(x))$, where $n \in Z^+$ and $x \in [-1, 1]$. By definition, the iterative relation of $T_n(x)$ can be written as $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$ $(n \geq 2)$, where $T_1(x) = x$ and $T_0(x) = 1$. The Chebyshev polynomial $T_n(x)$: $[-1, 1] \rightarrow [-1, 1]$ is called chaotic mapping for $n > 1$. To enhance the characteristics of Chebyshev chaotic mapping, Zhang [45] proposed an extended Chebyshev polynomial in 2008, which is defined as $T_n(x) = (2xT_{n-1}(x) - T_{n-2}(x))\ mod\ P$, where P refers to a large prime number and $x \in (-\infty, +\infty)$. Furthermore, the extended Chebyshev polynomial still fulfills the semigroup property, i.e., $T_m(T_n(x)) = T_{mn}(x) = T_{nm}(x) = T_n(T_m(x))$, where m and $n \in Z^+$.

The security of extended Chebyshev chaotic mapping depends on the following computational problems:

Chebyshev Polynomial Discrete Logarithm Problem (CPDLP): Given y, x, $T_r(x)$, whence $x \in (-\infty, +\infty)$, it is not feasible to find the integer r such that $y = T_r(x)$.

Chebyshev Polynomial Computational Diffie-Hellman Problem (CPCDHP): Given x, $T_r(x)$, $T_s(x)$, whence $x \in (-\infty, +\infty)$ & r, $s \geq 2$, it is not feasible to compute $T_{rs}(x)$ such that $T_{rs}(x) = T_r(T_s(x))$ or $T_{rs}(x) = T_s(T_r(x))$.

Compared with other chaotic mapping, due to the difficulty of the CPDLP problem, CPCDHP problem, and the distinctive semigroup property, it is feasible to employ Chebyshev chaotic mapping to generate a secure session key in authentication protocols. More importantly, the computation cost of Chebyshev chaotic mapping is only 1/3 of the elliptic curve scalar multiplication [35], which makes it possible to greatly reduce the computational cost of resource-constrained devices such as wireless sensors. Because of these advantages, we use Chebyshev chaotic mapping to design an improved scheme over Shin and Kwon's protocol.

B. ADVERSARY MODEL

It is very important to understand the ability of an attacker in cryptanalyzing the security flaws of cryptographic protocols and designing new protocols. Therefore, we depict the adversary model in the WSN environment based on the Dolev-Yao model [46] as follows:

(1) The attacker can eavesdrop, intercept, tamper and forward the messages transmitted on the wireless channel.

(2) If the attacker acquires the user's lost/stolen smart-card, he can extract the secret information on the card [47], [48].

(3) For the sake of memory, the ID and password chosen by the user are often low entropy. The attacker can enumerate the Cartesian product set on the space of identity and password of the user to make a successful offline guess attack [55], [56].

(4) When examining whether the protocol satisfies certain security properties, such as forward secrecy, an attacker can obtain the system's master key.

(5) The random nonces in the protocol must be large enough to prevent the attacker from guessing them successfully within polynomial time [24].

The high-level view of adversary model in WSN architecture is shown in Figure 1.

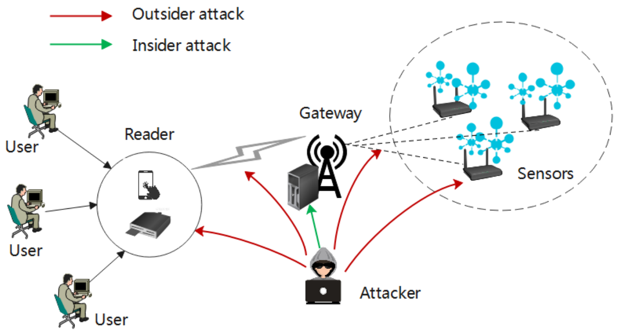

FIGURE 1. The high-level view of adversary model in WSN architecture.

C. SYMBOLS AND MEANING

The symbols and their meanings used in our cryptanalysis and the proposed protocol are listed in Table 1.

IV. REVIEW AND CRYPTANALYSIS OF SHIN AND KWON'S SCHEME

A. REVIEW OF SHIN AND KWON'S SCHEME

This section briefs Shin and Kwon's protocol. Their scheme is composed of four phases: system setup, user registration, login and authentication, and password change. Since the last phase is not related to our analysis, we ignore it.

TABLE 1. Symbols and meanings.

Symbol	Meaning
U_i	*i*th user
GWN	Gateway node
S_j	the *j*th sensor node
ID_i	U_i's identity
PW_i	U_i's password
BIO_i	U_i's biometric
SID_j	S_j's identity
X_g	GWN's secret key
T_i	Timestamp, i = 1, 2, ….
$h()$	One-way hash function
$H()$	Biometrics hash function[25, 49]
$Gen()/Rep()$	Fuzzy extractor generation/reproduction function
$\|\|$	Concatenation
$\oplus$	Bitwise XOR operation
$\Rightarrow$	The secure channel
$\rightarrow$	The public channel

1) SYSTEM SETUP

(1) GWN randomly selects two secret keys K_U and K_S.

(2) GWN selects a unique SID_j for each sensor S_j and computes its private key $X_j = h(SID_j||K_S)$.

(3) S_j stores SID_j and X_j in memory and is deployed to the target area.

2) USER REGISTRATION

(1) $U_i \Rightarrow GWN$: $\{TID_i, RPW_i\}$. The user U_i computes $Gen(BIO_i) = (\theta_i, \tau_i)$, $RPW_i = h(PW_i||\theta_i)$, $TID_i = h(ID_i|| r_i)$, where r_i is a random nonce.

(2) $GWN \Rightarrow U_i$: Smartcard $= \{A_i, B_i, C_i', h()\}$. GWN chooses a one-time pseudonym PID_i^1 for U_i, computes $HID_i = h(TID_i||K_U)$, $A_i = h(RPW_i||TID_i) \oplus HID_i$, $B_i = h(RPW_i||HID_i)$, $C_i' = h(TID_i||HID_i) \oplus PID_i^1$, then stores $\{A_i, B_i, C_i', h()\}$into the card and inserts$\{PID_i^1, TID_i\}$ into the database.

(3) Upon receiving the smartcard, U_i computes $D_i = r_i \oplus h(ID_i||\theta_i)$, and stores $\{D_i, \tau_i, Gen(), Rep()\}$ into the smartcard.

3) LOGIN AND AUTHENTICATION

(1) $U_i \rightarrow GWN$: $\{PID_i', M_i, M_{ug}, T_1\}$. U_i inputs ID_i, PW_i and imprints BIO_i, computes $\theta_i = Rep(BIO_i, \tau_i)$, $r_i = D_i \oplus h(ID_i||\theta_i)$, $TID_i = h(ID_i||r_i)$, $RPW_i = h(PW_i||\theta_i)$, $HID_i^* = A_i \oplus h(RPW_i||TID_i)$, $B_i^* = h(RPW_i||HID_i)$, and checks whether $B_i^* = B_i$. If the condition is not valid, U_i conceals the session; otherwise, it selects a random number $\mathrm{w_i}$ and calculates $PID_i^1 = C_i' \oplus h(TID_i||HID_i^*)$, $R_i = h(TID_i||PID_i^1||w_i)$, $M_i = w_i \oplus h(TID_i||HID_i^*||T_1)$, $M_{ug} = h(TID_{i||}HID_i^*||PID_i^1||R_{i||}T_1)$.

(2) $GWN \rightarrow S_j$: $\{PID_i^1, M_G, M_{gs}, T_2\}$. After receiving the message from U_i, GWN first checks the freshness of the time stamp T_1. If it is not fresh, GWN terminates the session; otherwise, it searches for TID_i in the database according to PID_i^1, and then calculates $HID_i = h(TID_i||K_U)$, $w_i^* = M_i \oplus h(TID_i||HID_i^*||T_1)$, $R_i^* = h(TID_i||PID_i^1||w_i^*)$, $M_{ug}^* = h(TID_{i||}HID_i^*||PID_i^1||R_i^*||T_1)$, and verifies whether $M_{ug}^* = M_{ug}$. If not, GWN terminates the session; otherwise, GWN calculates the $X_j = h(SID_j||K_S)$, $M_G = R_i^* \oplus h(X_j||T_2)$, $M_{gs} = h(PID_i^1||SID_j||X_j||R_i^*||T_1)$ and sends $\{PID_i^1, M_G, M_{gs}, T_2\}$ to S_j.

(3) $S_j \rightarrow GWN$: $\{M_j, M_{sg}, T_3\}$. S_j first checks the freshness of T_3, and terminates the session if T_3 is not fresh; otherwise, it calculates $R_i^* = M_G \oplus h(X_j||T_2)$, $M_{gs}^* = h(PID_i^1||SID_j||X_j||R_i^*||T_2)$, and verifies whether $M_{gs}^* = M_{gs}$. If it is not true, S_j terminates the session; otherwise, it selects a random number w_j, calculates $R_j = h(SID_j||w_j)$, $M_j^* = w_j \oplus h(X_j||T_3)$, $SK_{ij} = h(R_i^*||R_j)$, $M_{sg} = h(PID_i^1||SID_j||X_j||R_j||SK_{ij}||T_3)$ and finally sends the message $\{M_j, M_{sg}, T_3\}$ to GWN.

(4) $GWN \rightarrow U_i$: $\{P_i^2, M_G', M_{gu}, T_4\}$. GWN checks the freshness of T_3 first, it terminates the session if T_3 is not fresh, otherwise calculates $w_j = M_j \oplus h(X_j||T_3)$, $R_j^* = h(SID_j||w_j)$, $SK_{ij} = h(R_i^*||R_j^*)$, $M_{sg} = h(PID_i^1||SID_j||X_j||R_j^*||SK_{ij}^*||T_3)$, and checks whether $M_{sg}^* = M_{sg}$ is valid. If it is not valid, the session is terminated; otherwise, GWN randomly selects PID_i^2 to calculate $C_i^2 = \mathrm{h}(TID_i||HID_i^*) \oplus PID_i^2$, $P_i^2 = C_i^2 \oplus h(HID_i^*||T_4)$, $M_G' = R_j^* \oplus h(PID_i^1||HID_i^*)$, $M_{gu} = h(PID_i^1||HID_i^*||C_i^2||R_j^*||SK_{ij}^*||T_4)$ and replaces PID_i^1 with PID_i^2 in the database. Finally, GWN sends a message $\{P_i^2, M_G', M_{gu}, T_4\}$ to U_i.

(5) U_i checks the freshness of T_4, if T_4 is not fresh, terminates this session; otherwise, calculates $R_j^* = M_G' \oplus h(PID_i^1||HID_i^*)$, $SK_{ij}^* = h(R_i^*||R_j^*)$, $C_i^2 = P_i^2 \oplus h(HID_i^*||T_4)$, $M_{gu}^* = h(PID_i^1||HID_i^*||C_i^2||R_j^*||SK_{ij}^*||T_4)$, and checks whether $M_{gu}^* = M_{gu}$ holds. If not, U_i aborts the session; otherwise, it replaces C_i^1 with C_i^2 on the smartcard.

B. CRYPTANALYSIS ON SHIN AND KWON'S SCHEME

Although Shin and Kwon's scheme provides a formal security proof and claims their proposal can defend against a variety of passive and active attacks and meet various security requirements. However, our detailed cryptanalysis points out that their protocol is not as secure as they claim, and suffers from several serious security risks as follows.

1) FAILURE OF PROVIDING THREE-FACTOR SECURITY

Shin and Kwon's protocol is a three-factor scheme, which means that even if the attacker acquires two of the three factors, he is still not allowed to log on to the system. However, we find that when an attacker maliciously collects the biometric information of the user BIO_i and acquires the stolen/lost smartcard of the user, he can launch an offline guessing attack on the protocol and obtain the user's identity and password, which means their protocol cannot provide real three-factor security. The attack procedure is as follows:

(1) The attacker retrieves the secret data $\{A_i, B_i, C_i', D_i, h(), \tau_i, Gen(), Rep()\}$ from the smartcard via power analysis attack [47] as depicted in second item of the adversary model.

(2) The attacker calculates $Gen(BIO_i) = (\theta_i, \tau_i)$.

(3) The attacker selects a candidate (ID_i^*, PW_i^*) from the space of identity dictionary S_{ID} and space S_{PW} of password dictionary, and calculates $r_i = D_i \oplus$

$h(ID^*_i||\theta_i)$, $TID_i = h(ID^*_i||r_i)$, $RPW^*_i = h(PW_i||\theta_i)$, $HID^*_i = A_i \oplus h(RPW^*_i||TID^*_i)$.

(4) The attacker compares B_i with $h(RPW^*_i|| HID^*_i)$.

(5) If it matches, the correct ID_i and PW_i are successfully found by the attacker. Otherwise, he repeats steps (3) $\sim$ (5) until the correct ID_i and PW_i are found.

We herein use $|S_{ID}|$ and $|S_{PW}|$ to represent the size of S_{ID} and S_{PW}, respectively, and T_h to represent the execution time of the hash function. The time complexity of the above attack procedure is $O(|S_{ID}| * |S_{PW}| * 5 * T_h)$. In practice, since $|S_{ID}|$ and $|S_{PW}|$ are relatively small and $|S_{ID}| \leq |S_{PW}| \leq 10^6$ [50], [51], and T_h is negligible, the attacker can find the ID_i and PW_i of U_i in polynomial time.

Thus, Shin and Kwon's scheme cannot ensure three-factor security.

2) USER IMPERSONATION ATTACK

According to the above analysis, an attacker can guess the user's identity and password based on the obtained biometrics and smartcard. With this information, an attacker can imitate the user to log on to the system as follows:

(1) The attacker chooses a random number w'_i and calculates $PID^1_i = C^1_i \oplus h(TID_i||HID_i)$, $R'_i = h(TID_i||PID^1_i||w'_i)$, $M'_i = w'_i \oplus h(TID_{i||}HID_i||T'_1)$, $M'_{ug} = h(TID_{i||}HID^*_i||PID^1_i||R'_i||T'_1)$, and sends the login request message $\{PID^1_i, M'_i, M'_{ug}, T'_1\}$ to GWN.

(2) GWN first checks whether T'_1 is fresh. If T'_1 is fresh, it searches TID_i in its database by means of PID^1_i, calculates $HID_i = h(TID_i||K_U)$, $w^*_i = M_i \oplus h(TID_i||HID'_i||T'_1)$, $R^*_i = h(TID_i||PID^1_i||w^*_i)$, and verifies whether M'_{ug} matches $h(TID_i||HID^*_i||PID^1_i||R^*_i||T_1)$.

Obviously, the result is true. In this way, the attacker passes the authentication of the GWN. Thus, the attacker can successfully perform user impersonation attack on Shin and Kwon's scheme.

3) LACK OF ANONYMITY AND UNTRACEABILITY

Referring to Section IV-B-1), an attacker can obtain the relevant secret information $\{TID_i, HID_i\}$ to track the user as follows:

(1) The attacker calculates $Gen(BIO_i) = (\theta_i, \tau_i)$.

(2) The attacker calculates $r_i = D_i \oplus h(ID_i||\theta_i)$, $TID_i = h(ID_i||r_i)$, $RPW_i = h(PW_i||\theta_i)$, $HID^*_i = A_i \oplus h(RPW_i||TID_i)$.

(3) The attacker eavesdrops on the message $\{P^2_i, M'_G, M_{gu}, T_4\}$.

(4) The attacker calculates $C^2_i = P^2_i \oplus h(HID^*_i||T_4)$, $PID^2_i = h(TID_i||HID^*_i) \oplus C^2_i$.

By repeatedly eavesdropping on the message $\{P^2_i, M'_G, M_{gu}, T_4\}$ and revealing the secret value PID^2_i, the attacker can determine whether the same user is logged in by checking that PID^1_i matches PID^2_i when the user delivers the login message $\{PID^1_i, M_i, M_{ug}, T_1\}$ to GWN. Because the secret data TID_i and HID_i are fixed values, an attacker can continuously track the user through the above steps. Thus, Shin and Kwon's scheme fails to provide anonymity and untraceability.

4) DESYNCHRONIZATION ATTACK

In Shin and Kwon's scheme, to maintain user anonymity and prevent the attacker from tracking the user, GWN changes PID^1_i to PID^2_i in the database according to the login message $\{PID^1_i, M_i, M_{ug}, T_1\}$ received from the user, and masks PID^2_i in P^2_i and sends it to U_i. After verifying the authenticity of GWN, U_i restores C^2_i from P^2_i and updates C^1_i with C^2_i in the smartcard. When U_i logs in to the GWN again, C^1_i is used to restore PID^1_i and the user constructs a login message in which PID^1_i indicates the user's new pseudonym and sends it to the GWN. This synchronization mechanism forces users to change their pseudonym every time when they log in, preventing attackers from tracking users based on fixed parameters in the login request message to preserve user privacy. However, this mechanism is based on the ideal mode in which all the messages sent by all participants can be received correctly by the recipient during the authentication process. If an attacker intercepts the message $\{P^2_i, M'_G, M_{gu}, T_4\}$ sent by GWN to U_i, this synchronization mechanism will be broken, making it impossible for the user to log on to GWN again. This is because the message $\{P^2_i, M'_G, M_{gu}, T_4\}$ is intercepted, and the entry $\{PID^{1'}_i, TID_i\}$ in the database of GWN has been updated to $\{PID^2_i, TID_i\}$, but C^1_i on the smartcard cannot be updated to C^2_i without receiving the newest P^2_i. Therefore, when the user logs in to the GWN again, GWN will reject the login of U_i because the query with the keyword PID^1_i in the database returns nothing. An attacker can also break this synchronization mechanism by tampering with P^2_i. In this way, the attacker first intercepts message $\{P^2_i, M'_G, M_{gu}, T_4\}$, modifies P^2_i to $P^{2'}_i$ ($P^2_i \neq P^{2'}_i$), and then retransmits message $\{P^{2'}_i, M'_G, M_{gu}, T_4\}$ to U_i. Afterward, U_i will compute $R^*_j = M'_G \oplus h(PID^1_i||HID^*_i)$, $SK^*_{ij} = h(R^*_i||R^*_j)$, $C^{2'}_i = P^{2'}_i \oplus h(HID^*_i||T_4)$, $M^*_{gu} = h(PID^1_i||HID^*_i||C^{2'}_i||R^*_j||SK^*_{ij}||T_4)$, and verify whether $M^*_{gu} = M_{gu}$ holds or not. Obviously, this condition does not hold because $C^{2'}_i \neq C^2_i$. In this way, the consequence is that the user rejects this session and gives up updating C^1_i in the smartcard, which will eventually cause the user to be rejected when they log in to the GWN the next time.

Therefore, Shin and Kwon's scheme is vulnerable to desynchronization attack.

5) SENSOR NODE CAPTURE ATTACK

Suppose the attacker has hijacked sensor node S_j, he can extract $\{SID_j, X_j\}$ from the memory of S_j. With the eavesdropped messages $\{PID^1_i, M_G, M_{gs}, T_2\}$ and $\{M_j, M_{sg}, T_3\}$, the attacker can reveal the session key shared between U_i and S_j as follows.

(1) The attacker computes $R^*_i = M_G \oplus h(X_j||T_2)$.

(2) The attacker computes $w_j = M_j \oplus h(X_j||T_3)$.

(3) The attacker computes $R^*_j = h(SID_j||w_j)$.

(4) The attacker computes $SK_{ij} = h(R^*_i||R^*_j)$.

That is, the attacker can disclose the session key if the sensor node is captured. Thus, Shin and Kwon's protocol is unprotected from sensor node capture attack.

6) PRIVILEGED INSIDER ATTACK

Privileged insider attack is a security threat that has been ignored for a long time, and even protocol designers are not aware of the serious consequences of such attack [52], which has been mentioned in [52], [53]. When scrutinizing scheme [42], we find that neither Jung *et al.*'s scheme can resist the security risk of privileged insider attack, nor can Shin and Kwon's scheme, which is an improved version of Jung *et al.*'s scheme, defend against privileged insider attack. Assuming that the privileged insider has gained the user's registration request $\{TID_i, RPW_i\}$, as well as has obtained the user's smartcard for a short time, and extracts the secret parameters $\{A_i, B_i, C_i', D_i, h(), \tau_i, Gen(), Rep()\}$ on the smartcard via side-channel analysis [48], he can reveal the secret data about U_i stored in the memory of GWN.

(1) The attacker computes $HID_i = A_i \oplus h(RPW_i||TID_i)$.
(2) The attacker computes $PID_i^1 = C_i^1 \oplus h(TID_i||HID_i)$.
(3) The attacker records the triple $\{PID_i^1, TID_i, HID_i\}$.

Furthermore, the attacker can reveal the session key when U_i logs into GWN to access S_j as follows:

(1) The attacker eavesdrops messages $\{PID_i^1, M_i, M_{ug}, T_1\}$ and $\{P_i^2, M_G', M_{gu}, T_4\}$.
(2) The attacker computes $w_i = M_i \oplus h(TID_i||HID_i^*||T_1)$, $R_i = h(TID_i||PID_i^1||w_i)$, and $R_j = M_G' \oplus h(PID_i^1||HID_i)$.
(3) The attacker computes $SK_{ij} = h(R_i||R_j)$.

Using the triple $\{PID_i^1, TID_i, HID_i\}$ in his hand, the attacker can continuously disclose U_i's the newest dynamic pseudonym PID_i^1 by constantly eavesdropping messages $\{P_i^2, M_G', M_{gu}, T_4\}$ to continuously track user U_i as described in Section IV-B-3) at any time.

From the above discussion, it is evident that Shin and Kwon's scheme is vulnerable to privileged insider attack.

V. THE PROPOSED SCHEME

In this section, we propose an improved three-factor user authentication and key agreement protocol using Chebyshev chaotic mapping for WSN to overcome the security threats found in Shin and Kwon's protocol. Concretely, we employ three countermeasures to enhance Shin and Kwon's protocol as follows: (1) We use the fuzzy verifier technique to thwart the failure of providing three-factor security and user impersonation attack. (2) The Chebyshev chaotic mapping is employed to avoid lack of anonymity and untraceability, desynchronization attack, and sensor node capture attack. (3) We prevent the privileged insider attack by replacing the important parameters on the smartcard received by the user during the user registration phase. Similar to their scheme, our improved protocol consists of four phases as follows.

A. SYSTEM SETUP

GWN chooses a master key X_g, a large prime nonce q, a random nonce θ, a number $y \in [+\infty, -\infty]$, and computes $G = T_\theta(y)$, keeps $\{X_g, \theta\}$ secretly and publishes $\{y, q\}$. Then, GWN computes $K_j = h(SID_j||X_g)$ for each sensor S_j and stores it as the secret key in the memory of S_j.

B. USER REGISTRATION

Step 1: U_i inputs his ID_i, PW_i, BIO_i, chooses a random nonce a_i, computes $RPW_i = h(PW_i||a_i)$ and sends $\{ID_i, RPW_i\}$ to GWN via a secure channel.

Step 2: GWN chooses a random nonce b_i, and an integer number $m \in [2^4, 2^8]$, computes $PID_i = h(h(ID_i)||b_i)$, $K_i = h(PID_i||X_g)$, $A_1 = h(RPW_i||ID_i) \oplus K_i$, $A_2 = h(K_i) \oplus RPW_i$, and stores $(h(ID_i), b_i)$ in the database. Finally, GWN stores $\{A_1, A_2, m, G, h(), H()\}$ in a smartcard and delivers the card to U_i via a secure channel.

Step 3: Upon receipt of the smartcard, U_i calculates $C_i = a_i \oplus h(ID_i||PW_i||H(BIO_i))\ mod\ m$, $A_1^* = A_1 \oplus h(ID_i||a_i)$, $A_2^* = A_2 \oplus h(a_i||PW_i)$, updates (A_1, A_2) with (A_1^*, A_2^*) and saves C_i in the smartcard.

C. AUTHENTICATION

U_i logs into GWN to build a session key with S_j using his ID_i, PW_i, smartcard, and BIO_i as follows.

Step 1: U_i inserts his card to the card reader, keys his ID_i, PW_i, and inputs his BIO_i. Then, the smartcard computes $a_i = C_i \oplus h(ID_i||PW_i||H(BIO_i))\ mod\ m$, $RPW_i = h(PW_i||a_i)$, $A_1 = A_1^* \oplus h(ID_i||a_i)$, $A_2 = A_2^* \oplus h(a_i||PW_i)$, $K_i = h(RPW_i||ID_i) \oplus A_1$, and checks whether $A_2 = h(K_i) \oplus RPW_i$ holds. If not, the smartcard rejects the session. Otherwise, the card selects a random nonce u, computes $D_1 = T_u(y)$, $D_2 = (h(ID_i)||SID_j) \oplus T_u(G)$, $M_{UG} = h(h(ID_i)||SID_j||K_i||D_1||T_1)$, and sends a login request $Msg_{ug} = \{D_1, D_2, M_{UG}, T_1\}$ to GWN via a public channel.

Step 2: Upon receiving Msg_{ug}, GWN checks the freshness of T_1 and computes $D_2 \oplus T_\theta(D_1)$ to obtain $h(ID_i)||SID_j$, and retrieves b_i in the database using $h(ID_i)$, computes $PID_i' = h((ID_i)||b_i)$, $K_i' = h(PID_i'||X_g)$, and checks whether the received M_{UG} is equal to $h(h(ID_i)||SID_j||K_i||D_1||T_1)$. If yes, GWN selects a random nonce r_g, and computes $K_j = h(SID_j||X_g)$, $D_3 = r_g \oplus h(K_j||SID_j||T_2)$, $M_{GS} = h(D_1||SID_j||D_3||r_g||T_2)$. Finally, GWN sends $Msg_{gs} = \{D_1, D_3, M_{GS}, T_2\}$ to S_j.

Step 3: On receipt of the message, as T_2 is fresh, S_j computes $r_g' = D_3 \oplus h(K_j||\ SID_j||T_2)$, $M_{GS} = h(D_1||SID_j||D_3||r_g'||T_2)$, and checks whether M_{GS} is equal to $h(D_1||SID_j||D_3||r_g'||T_2)$. If the condition is false, S_j aborts the session; otherwise, S_j chooses a random nonce v, calculates $D_4 = T_v(y)$, $SK_{ji} = h(T_v(D_1)||r_g')$, $M_{SG} = h(D_4||r_g'||K_j||T_3)$, and sends the message $Msg_{sg} = \{D_4, M_{SG}, T_3\}$ to GWN.

Step 4: On receipt of the message, as T_3 is fresh, GWN computes $M_{SG}' = h(D_4||r_g||K_j||T_3)$ and checks whether $M_{SG}' = M_{SG}$ holds. If not, GWN conceals this session; otherwise, GWN computes $D_5 = r_g \oplus h(K_i'||T_4)$, $M_{GU} = h(D_5||K_i'||D_4||r_g||T_4)$, and sends $Msg_{gu} = \{D_4, D_5, M_{GU}, T_4\}$ to U_i.

Step 5: Upon receiving the message, as T_4 is fresh, U_i computes $r_g' = D_3 \oplus h(K_i||T_4)$, $SK_{ij} = h(T_u(D_1)||r_g')M_{GU}' = h(D_5||K_i||D_4||r_g'||T_4)$, and checks whether $M_{GU}' = M_{GU}$. If not, U_i aborts the session, else U_i accepts SK_{ij} as the shared key with S_j.

TABLE 2. The authentication phase.

U_i	GWN	S_j
Inputs ID_i, PW_i, BIO_i $a_i = C_i \oplus h(ID_i \parallel PW_i \parallel H(BIO_i))\ mod\ m$ $RPW_i = h(PW_i \parallel a_i)$ $A_2 = A_2^* \oplus h(a_i \parallel PW_i)$ $K_i = h(RPW_i \parallel ID_i) \oplus A_1$ Checks $A_2 ? = h(K_i) \oplus RPW_i$ Selects u $D_1 = T_u(y)$ $D_2 = (h(ID_i) \parallel SID_j) \oplus T_u(G)$ $M_{UG} = h(h(ID_i) \parallel SID_j \parallel K_i \parallel D_1 \parallel T_1)$ $Msg_{ug} = \{D_1, D_2, M_{UG}, T_1\}$ $\longrightarrow$		
	Checks T_1 $h(ID_i) \parallel SID_j = D_2 \oplus T_0(D_1)$ Retrieves b_i $PID_i' = h(h(ID_i) \parallel b_i)$ $K_i' = h(PID_i' \parallel X_g)$ Checks $M_{UG} ? = h(h(ID_i) \parallel SID_j \parallel K_i \parallel D_1 \parallel T_1)$ Selects r_g $K_j = h(SID_j \parallel X_g)$ $D_3 = r_g \oplus h(K_j \parallel SID_j \parallel T_2)$ $M_{GS} = h(D_1 \parallel SID_j \parallel D_3 \parallel r_g \parallel T_2)$ $Msg_{gs} = \{D_1, D_3, M_{GS}, T_2\}$ $\longrightarrow$	
		Checks T_2 $r_g' = D_3 \oplus h(K_j \parallel SID_j \parallel T_2)$ $M_{GS} = h(D_1 \parallel SID_j \parallel D_3 \parallel r_g' \parallel T_2)$ Checks $M_{GS}? = h(D_1 \parallel SID_j \parallel D_3 \parallel r_g' \parallel T_2)$ Selects v $D_4 = T_v(y)$ $SK_{ji} = h(T_v(D_1) \parallel r_g')$ $M_{SG} = h(D_4 \parallel r_g' \parallel K_j \parallel T_3)$ $\longleftarrow$ $Msg_{sg} = \{D_4, M_{SG}, T_3\}$
	Checks T_3 $M_{SG}' = h(D_4 \parallel r_g \parallel K_j \parallel T_3)$ Checks $M_{SG}'? = M_{SG}$ $D_5 = r_g \oplus h(K_i' \parallel T_4)$ $M_{GU} = h(D_5 \parallel K_i' \parallel D_4 \parallel r_g \parallel T_4)$ $\longleftarrow$ $Msg_{gu} = \{D_4, D_5, M_{GU}, T_4\}$	
Checks T_4 $r_g' = D_5 \oplus h(K_i \parallel T_4)$ $SK_{ij} = h(T_u(D_1) \parallel r_g')$ $M_{GU}' = h(D_5 \parallel K_i \parallel D_4 \parallel r_g' \parallel T_4)$ Checks $M_{GU}'? = M_{GU}$ Accepts SK_{ij}		

For ease of understanding, this phase is illustrated in Table 2.

D. PASSWORD UPDATE PHASE

This phase is executed if U_i intends to change his password to a new one.

Step 1: U_i inputs ID_i, PW_i, and imprints BIO_i.

Step 2: The card computes $a_i = C_i \oplus h(ID_i||PW_i||H(BIO_i))$ $mod\ m$, $RPW_i = h(PW_i||a_i)$, $A_1 = A_1^* \oplus h(ID_i||a_i)$, $A_2 = A_2^* \oplus h(a_i||PW_i)$, $K_i' = h(RPW_i||ID_i) \oplus A_1$, and checks condition $A_2? = h(K_i') \oplus RPW_i$. If not, the smartcard aborts the session.

Step 3: U_i inputs his new password PW_i^{new}, calculates $C_i^{new} = a_i \oplus h(ID_i||PW_i^{new}||H(BIO_i))$ $mod\ m$, $RPW_i^{new} = h(PW_i^{new}||a_i)$, $A_1^{*new} = A_1 \oplus h(RPW_i||ID_i) \oplus h(RPW_i^{new}||ID_i) \oplus h(ID_i||a_i)$, $A_2^{*new} = A_2 \oplus RPW_i \oplus RPW_i^{new} \oplus h(a_i||PW_i^{new})$.

Step 4: Finally, U_i replaces (A_1^*, A_2^*, C_i) with $(A_1^{*new}, A_2^{*new}, C_i^{new})$ in the card.

VI. SECURITY ANALYSIS

In this section, a formal security proof using random oracle model, a simulation verification via ProVerif, and a heuristic security analysis are provided to demonstrate the security of our scheme.

A. FORMAL SECURITY PROOF

This section identifies the security of our improved scheme under random oracle model. For simplicity, the security model used in our security proof follows the work of [24], [30], [54].

Theorem 1: Assume that P denotes our improved three-factor authentication and key agreement protocol, $\mathcal{A}$ denotes an attacker, and Adv_P^{AKE} (A) represents the advantage of $\mathcal{A}$ in breaking the semantic security of P, Adv_A^{CDH} (t) represents the advantage for $\mathcal{A}$ to break the *CPCDHP* problem within polynomial time t. Suppose $\mathcal{A}$ asks Send queries no more than q_h times. Then, we see that

$$\begin{aligned} Adv_P^{AKE}(A) \leq & \frac{2q_s + q_h^2 + (q_s+q_e)^2}{2^{l_s}} + \frac{(q_s+q_e)^2}{n-1} \\ & + 2q_h \max\left\{\frac{q_s}{|S_{PW}|}, \frac{q_s}{2^{l_b}}, \varepsilon\right\} \\ & + 2q_h Adv_A^{CDH}(t + (2q_s + 6q_e)\,T_c) \\ & + 4q_h(q_s+q_e)^2 Adv_A^{CDH}(t + (q_s+q_e)\,T_c) \end{aligned}$$

where $l_s, l_b, n, \varepsilon, T_c$ denotes the bit length of hash value, BIO_i, Chebyshev polynomial, the probability of "false positive", the execution time of Chebyshev polynomial.

Proof: We define a series of games from G_0 to G_5 to complete the proof that our improved three-factor protocol is secure against the attacker, and the probabilities of the events that $\mathcal{A}$ successfully surmises the coin z in game G_i are denoted by $Pr[S_i]$, respectively.

G_0: This game is performed in real scenario based on random oracles. Thus, we get

$$Adv_P^{AKE}(A) = 2Pr[S_0] - 1 \tag{1}$$

G_1: A series of oracles are simulated according to the definition of our improved authentication protocol. There are four *Send* queries with respect to Section V: $Send(U_i^a, GWN^a$, $\mathrm{Msg_{ug}})$, $Send(GWN^a, Saj$, $\mathrm{Msg_{gs}})$, $Send(Saj, GWN^a$, $\mathrm{Msg_{sg}})$, $Send(GWN^a$, U_i^a, $\mathrm{Msg_{gu}})$. In addition, we use lists L_h, L_A, L_T to store the result of hash queries, A's answer of queries, the transcript of the protocol. Thus, the simulation of G_1 is indispensable from G_0 and we obtain

$$Pr[S_1] = Pr[S_0] \tag{2}$$

G_2: Some collisions in protocol P are avoided in the game. If a collision happens on hash queries and transcripts Msg_{ug}, Msg_{gs}, Msg_{sg}, Msg_{gu}, we terminate the simulation and let $\mathcal{A}$ win according to the following three cases: (1) Collision on hash oracle, the probability is $\frac{q_h^2}{2^{l_s+1}}$; (2) Collision on random nonces u and v, the probability is no more than $\frac{(q_s+q_e)^2}{2(n-1)}$; (3) Collision on r_g, the probability is no more than $\frac{(q_s+q_e)^2}{2^{l_s+1}}$. Thus, G_1 is indistinguishable from G_2 and we obtain

$$|Pr[S_2] - Pr[S_1]| \leq \frac{q_h^2 + (q_s+q_e)^2}{2^{l_s+1}} + \frac{(q_s+q_e)^2}{2(n-1)} \tag{3}$$

G_3: In this game, we take into account the simulation in which $\mathcal{A}$ impersonates $\{Msg_{ug}, Msg_{gs}, Msg_{sg}, Msg_{gu}\}$ without simulating the hash query. Thus, G_3 is indistinguishable from G_2 and we obtain

$$|Pr[S_3] - Pr[S_2] \leq \frac{q_s}{2^{l_s}} \tag{4}$$

G_4: This game considers that $\mathcal{A}$ encounters the *CPCDHP* problem in breaking the security of the session key. If $\mathcal{A}$ can read the session key negotiated by U_i and S_j, it suggests that $\mathcal{A}$ has asked *Corrupt* query and solves the *CPCDHP* problem and $(T_v(T_u(y))\,||r_g)$ is stored in list L_A. Since $\mathcal{A}$ is not allowed to obtain three factors at the same time, the attacker only can obtain at most two of three factors to break the third factor. This game includes four cases as follows:

Case 1: $\mathcal{A}$ asks $Corrupt(U_i^a$, 1) and $Corrupt(U_i^a$, 2) to guess the password with no more than q_s *Send* queries using the password dictionary S_{PW}. The probability is $\frac{q_s}{|S_{PW}|}$.

Case 2: $\mathcal{A}$ asks $Corrupt(U_i^a$, 0) and $Corrupt(U_i^a$, 1) to break the biometric with q_s chances. The probability to obtain $H(BIO_i)$ is $\frac{q_s}{2^{l_b}}$

Case 3: $\mathcal{A}$ asks $Corrupt(U_i^a$, 0) and $Corrupt(U_i^a$, 2). The probability that $\mathcal{A}$ guesses correct C_i is ε at most.

Case 4: To break the session key $h(T_v(T_u(y))\ ||\ r_g)$, it is necessary for $\mathcal{A}$ to compute $T_v(T_u(y))$ with D_1 and D_4, where $D_1 = T_u(y)$ and $D_4 = T_v(y)$, $T_v(T_u(y))$ can be stored in L_A and $\mathcal{A}$ can ask q_h *Hash* query at most, then $\mathcal{A}$ has to ask *Send* queries to simulate *Execute* queries. The probability is $q_h Adv_A^{CDH}(t + (q_s+q_e)\,T_c)$

Thus, we obtain

$$\begin{aligned} |Pr[S_4] - Pr[S_3]| \leq & \max q_h \left\{\frac{q_s}{|S_{PW}|}, \frac{q_s}{2^{l_b}}, \varepsilon\right\} \\ & + q_h Adv_A^{CDH}(t + (2q_s + 6q_e)\,T_c) \end{aligned} \tag{5}$$

G_5: This game considers that $\mathcal{A}$ will try to break forward secrecy by simulating *Hash*, *Send*, *Execute* queries on transcripts $\{Msg_{ug}, Msg_{gs}, Msg_{sg}, Msg_{gu}\}$. $\mathcal{A}$ simulates *Test* queries and asks $Corrupt(U_i^a \backslash\ GWN^a \backslash\ Saj)$ by choosing two indices from $\{1, 2, \ldots, q_s{+}q_e\}$. Suppose $(T_v(T_u(y))\,||r_g) \in L_A$, and the game will abort if the session key $h(T_v(T_u(y))\,||r_g)$ cannot be returned. Thus, we obtain

$$\begin{aligned} |Pr[S_5] - Pr[S_4]| \leq & 2q_h(q_s+q_e)^2 Adv_A^{CDH} \\ & \times (t + (q_s+q_e)\,T_c) \end{aligned} \tag{6}$$

Therefore, all games considered, $\mathcal{A}$ has no superiority in surmising the coin z and we obtain

$$Pr[S_5] = \frac{1}{2} \tag{7}$$

From (1)~(7), we obtain:

$$\begin{aligned} Adv_P^{AKE}(A) \leq & \frac{2q_s + q_h^2 + (q_s+q_e)^2}{2^{l_s}} + \frac{(q_s+q_e)^2}{n-1} \\ & + 2q_\mathrm{h} \max\left\{\frac{q_s}{|S_{PW}|}, \frac{q_s}{2^{l_b}}, \varepsilon\right\} \\ & + 2q_h Adv_A^{CDH}(t + (2q_s + 6q_e)\,T_c) \\ & + 4q_h(q_s+q_e)^2 Adv_A^{CDH}(t + (q_s+q_e)\,T_c) \end{aligned}$$

B. SECURITY VERIFICATION USING PROVERIF

ProVerif [60] is a widely accepted simulation tool in verifying the security of cryptography protocols. In this section, we convert the communication entities in our protocol into three processes under pi calculus and run them concurrently in ProVerif to prove that our proposal is capable of guaranteeing the secrecy and achieving mutual authentication as follows.

```
  (∗Channels∗)
  free sc1: channel [private].
  free sc2: channel [private].
  free c1: channel [private].
  free c2: channel [private].

  (∗Type∗)
  type N.


  (∗Session key∗)
  free SKi_j, SKj_i:bitstring [private].


  (∗constants and variables∗)
  const m: bitstring.
  const q: N.
  const Xg: bitstring [private].
  const Theta: bitstring [private].
     const Rg: bitstring [private].
  const BIOi: bitstring.
  const BIOi’: bitstring.
  free G: bitstring [private].
  free y: bitstring.
  free Kj: bitstring [private].
  free IDi: bitstring [private].
  free SIDj: bitstring [private].


  (∗Constructor∗)
  fun con(bitstring, bitstring): bitstring.
  fun con3(bitstring, bitstring, bitstring): bitstring.
  fun con4(bitstring, bitstring, bitstring, bitstring): bitstring.
  fun con5(bitstring, bitstring, bitstring, bitstring, bitstring):
bitstring.
  fun mod(bitstring, bitstring): bitstring.
  fun xor(bitstring, bitstring): bitstring.
  fun h(bitstring): bitstring.
     fun H(bitstring): bitstring.
     fun CM(bitstring, bitstring): bitstring.


  (∗events∗)
  event evBeginSj(bitstring).
  event evEndSj(bitstring).
  event evBeginUi(bitstring).
  event evEndUi(bitstring).


  let processUi =
     new PWi: bitstring;
     new ai: bitstring;
     let RPWi = h(con(PWi, ai)) in
     out(sc1, (IDi, RPWi));
     in(sc1, (A1’: bitstring, A2’: bitstring, m′: bitstring, G′:
bitstring));
```

```
     let Ci = mod(xor(ai, h(con(con(IDi, PWi), H(BIOi)))),
m) in
     let A1” = xor(A1’, h(con(IDi, ai))) in
     let A2” = xor(A1’, h(con(IDi, ai))) in
     !(
     event evBeginUi(IDi);
     let ai’ = mod(xor(Ci, h(con(con(IDi, PWi), H(BIOi)))),
m) in
     let A2 = xor(A2”, h(xor(ai’, PWi))) in
     let Ki = xor(h(con(RPWi, IDi)), A1”) in
     if A2 = xor(h(Ki), RPWi) then
     new u: bitstring;
     new T1: bitstring;
     let D1 = CM(u, y) in
     let D2 = xor(con(h(IDi), SIDj), CM(u, G)) in
     let MUG = h(con5(h(IDi), SIDj, Ki, D1, T1)) in
     event evBeginUi(IDi);
     out(c1, (D1, D2, MUG, T1));
     in(c1, (D4’: bitstring, D5’: bitstring, MGU’: bitstring,
T4’: bitstring));
     let rg’ = xor(D5’, h(con(Ki, T4’))) in
     let SKi_j = h(con(CM(u, D1), rg’)) in
     let MGU = h(con5(D5’, Ki, D4’, rg’, T4’)) in
     event evEndUi(IDi);
     0
  ).


  let processGWN =
     let G = CM(Theta, y) in
  let Kj = h(con(SIDj, Xg)) in
     in(sc1, (IDi’: bitstring, RPWi’: bitstring));
     new bi: bitstring;
     let PIDi = h(con(h(IDi), bi)) in
     let Ki = h(con(PIDi, Xg)) in
     let A1 = h(con(RPWi’, IDi)) in
     let A2 = xor(h(Ki), RPWi’) in
     out(sc1, (A1, A2, m, G));
     !(
     in(c1, (D1’: bitstring, D2’: bitstring, MUG’: bitstring,
T1’: bitstring));
     let (hIDi’: bitstring, SIDj’: bitstring) = xor(D2’, CM(
Theta, D1’)) in
     let PIDi’ = h(xor(hIDi’, bi)) in
     let Ki’ = h(xor(PIDi, Xg)) in
     if MUG’ = h(con5(hIDi’, SIDj’, Ki’, D1’, T1’)) then
        new rg: bitstring;
        new T2: bitstring;
        let Kj = h(con(SIDj’, Xg)) in
        let D3 = xor(rg, h(con3(Kj, SIDj’, T2))) in
        let MGS = h(con5(D1’, SIDj’, D3, rg, T2)) in
           out(c2, (D1’, D3, MGS, T2));
     in(c2, (D4’: bitstring, MSG’: bitstring, T3’: bitstring));
  if MSG’ = h(con4(D4’, rg, Kj, T3’)) then
     new T4: bitstring;
  let D5 = xor(rg, h(con(Ki, T4))) in
let MGU = h(con5(D5, Ki’, D4’, rg, T4)) in
out(c1, (D4’, D5, MGU, T4));
  0
).
let processSensor =
  ! (
    event evBeginSj(SIDj);
  in(c2, (D1’: bitstring, D3’: bitstring, MGS’: bitstring,
T2’: bitstring));
  let rg’ = xor(D3’, h(con3(Kj, SIDj, T2’))) in
  if MGS’ = h(con5(D1’, SIDj, D3’, rg’, T2’)) then
  new v: bitstring;
  new T3: bitstring;
  let D4 = CM(v, y) in
  let SKj_i = h(con(CM(v, D1’), rg’)) in
  let MSG = h(con4(D4, rg’, Kj, T3)) in
  out(c2, (D4, MSG, T3));
  event evEndSj(SIDj);
  0
).
query attacker(SKi_j).
query attacker(SKj_i).
query id:bitstring;inj-event(evEndSj(id))==>inj-event(
evBeginSj(id)).
query id:bitstring;inj-event(evEndUi(id))==>inj-event(
evBeginUi(id)).
process !processUi | !processGWN | !processSensor
```

The above ProVerif verification produces the output as follows:

(1) Query not attacker(*SKi_j*[]) is true.
(2) Query not attacker(*SKj_i*[]) is true.
(3) Query inj-event(*evEndSj*(*id*)) ==>
inj-event(*evBeginSj*(*id*)) is true.
(4) Query inj-event(*evEndUi*(*id*)) ==>
inj-event(*evBeginUi*(*id*)) is true.

From the first two lines of the output, because the attacker fails to query *SKi_j* and *SKi_j*, it can be deduced that he is incapable of breaching the secrecy of the session key generated by U_i and S_j, respectively. Furthermore, the last two lines indicate that mutual authentication has been successfully achieved between U_i and S_j. Thus, our proposal holds the desired security properties of secrecy of the session key and mutual authentication.

C. INFORMAL SECURITY ANALYSIS

In this section, we show that our improved scheme thwarts security defects in Shin and Kwon's scheme as well as achieves some desired security properties.

1) THREE-FACTOR SECURITY

Three-factor security denotes that although the attacker compromises two of the three factors, he is incapable of breaking the security of the authenticated scheme. We demonstrate that our proposal fulfills this security property in three cases.

Case 1: Assume that the user's password and smartcard are compromised by the attacker.

There is no doubt that the attacker can extract the secret data $\{A_1^*, A_2^*, C_i, m, G, h(), H()\}$ from the smartcard according to item (2) of the adversary model. However, the attacker cannot pass the verification because he does not have the user's identity and biometrics in computing $a_i = C_i \oplus h(ID_i|| PW_i||H(BIO_i))\ mod\ m$.

Case 2: Assume that the user's biometrics and smartcard are compromised by the attacker.

The attacker could also retrieve secrets $\{A_1^*, A_2^*, C_i, m, G, h(), H()\}$ from the smartcard. To breach our proposal, the attacker would select a candidate pair (ID_i^*, PW_i^*) from the Cartesian product $S_{ID}*S_{PW}$ to launch an offline guessing attack on the identity and password via computing $a_i = C_i \oplus h(ID_i|| PW_i||H(BIO_i))\ mod\ m$, $A_1 = A_1^* \oplus h(ID_i||a_i)$, $A_2 = A_2^* \oplus h(a_i|| PW_i)$, $K_i' = h(RPW_i|| ID_i) \oplus A_1$, and checking whether $A_2? = h(K_i') \oplus RPW_i$.

Obviously, there will be $|S_{ID}*S_{PW}|/m$ candidates to prevent his guessing attack from being successful. As an example, we assume that the user's identity and password are numbers, thus we have $|S_{ID}| = |S_{PW}| = 10^8$. Suppose $m = 2^8$, correspondingly, there will be $|S_{ID}| * |S_{PW}|/m = 10^8 * 10^8/2^8 \approx 2^{45}$ candidates [55], [56]. Someone may suspect that the attacker can accidentally find a pair (ID_i^*, PW_i^*) that is not equal to (ID$_i$, PW$_i$) but $a_i = C_i \oplus h(ID_i^*|| PW_i^*||H(BIO_i))$ $mod\ m$ holds. The possibility is $1/2^8$. According to [55], if the system requires both the new password and the old password when the user logs in, the probability will be reduced to $(1/2^8)^2$. Thus, the fuzzy verifier makes it difficult for the attacker to succeed in an identity and password guessing attack in the proposal.

Case 3: Assume that the user's password and biometrics are compromised by the attacker.

Despite his knowledge of PW_i and $H(BIO_i)$ in computing $a_i = C_i \oplus h(ID_i||PW_i||H(BIO_i))\ mod\ m$, the attackers will still fail when he tries to pass *GWN*'s authentication because he knows nothing about the necessary parameters C_i and ID_i.

2) USER IMPERSONATION ATTACK

If the attacker intends to launch user impersonation attack successfully, he must first generate a valid login request message $Msg_{ug} = \{D_1, D_2, M_{UG}, T_1\}$. However, because the attacker does not know ID_i, SID_j and K_i, he cannot produce such an effective message Msg_{ug}. Thus, our improved scheme is immune to user impersonation attack.

3) PRIVILEGED INSIDER ATTACK

If the attacker acquires the user's registration message $\{ID_i, RPW_i\}$ and the smartcard that stores secret parameters $\{A_1^*, A_2^*, C_i, m, G, h(), H()\}$, where $C_i = a_i \oplus h(ID_i|| PW_i||H(BIO_i))\ mod\ m$, $A_1^* = A_1 \oplus h(ID_i||a_i)$, $A_2^* = A_2 \oplus h(a_i|| PW_i)$, and selects an item PW_i^* from S_{PW} to carry out offline password guessing attack via $a_i = C_i \oplus h(ID_i|| PW_i||H(BIO_i))\ mod\ m$. Unfortunately, because the attacker knows nothing about the user's biometrics, there is no doubt that his attack will fail. Therefore, the improved scheme is able to withstand privileged insider attack.

4) DESYNCHRONIZATION ATTACK

As we analyzed in Section IV, Shin and Kwon's scheme suffers from desynchronization attack due to the need to update some secret information between the *GWN* side and the user side to maintain user anonymity. In the improved scheme, the user's smartcard does not need to update authentication data simultaneously after each authentication with *GWN*. Therefore, our improved protocol is immunized from desynchronization attack.

5) GWN IMPERSONATION ATTACK

To impersonate *GWN*, the attacker has to produce two effective messages $\{D_1, D_3, M_{GS}, T_2\}$ and $\{D_4, D_5, M_{GU}, T_4\}$, and delivers them to the sensor S_j and the user U_i, respectively. In these two messages, $D_3 = r_g \oplus h(K_j|| SID_j|| T_2)$, $M_{GS} = h(D_1|| SID_j|| D_3||r_g||T_2)$, $D_5 = r_g \oplus h(K_i||T_4)$, $M_{GU} = h(D_5||K_i||D_4|| r_g||T_4)$. It is impossible for an attacker to generate these two valid messages to fool S_j and U_i because of the lack of knowledge about K_j, SID_j, and K_i. Thus, our improved scheme can withstand *GWN* impersonation attack.

6) SENSOR IMPERSONATION ATTACK

To impersonate the sensor S_j, the attacker must be able to forge a valid message $\{D_4, M_{SG}, T_3\}$ for deception. Nevertheless, without knowing *GWN*'s secret key X_g and S_j's identity SID_j, the attacker cannot produce such a valid message at all.

7) SENSOR NODE CAPTURE ATTACK

Provided that the attacker has captured the sensor S_j, he can extract the secret key K_j from S_j's memory. With the message $\{D_1, D_3, M_{GS}, T_2\}$ intercepted on the public channel, the attacker can acquire r_g by computing $D_3 \oplus h(K_j|| SID_j||T_2)$. Although the attacker obtained D_1 and r_g, which are required to negotiate a session key between S_j and U_i, he still cannot reveal the session key SK_{ij} because of the *CPDLP* problem in solving the random nonce v according to D_4. In other words, in our scheme, if one or more sensor nodes are captured, the attacker can neither obtain the session key generated by the sensor nodes and the user, nor influence the remaining sensor nodes. Therefore, our improved scheme is secure to resist sensor node capture attack.

8) FORWARD SECRECY

During the authentication process, with the assistance of *GWN*, U_i and S_j generate a shared session key $SK_{ij} = h(T_v(D_1)||r_g) = h(T_u(D_4)||r_g)$, where $D_1 = T_u(y)$, $D_4 = T_v(y)$, and $r_g = D_3 \oplus h(K_j|| SID_j||T_2)$ are produced by U_i, S_j, and *GWN*, respectively. If the secret key $\{X_g, \theta\}$ is disclosed to the attacker, he can reveal SID_j by computing $h(ID_i)|| SID_j = D_2 \oplus T_\theta(D_1)$, and reveals r_g by computing $K_j = h(SID_j||X_g)$ and $r_g = D_3 \oplus h(K_j|| SID_j||T_2)$, where D_1, D_2, D_3 are transmitted over the public channel. However, even if the parameter y is public, it is not feasible for the attacker to determine u or v from $D_1 = T_u(y)$ and $D_4 = T_v(y)$ because he cannot breach the *CPDLP* and *CPCDHP*.

9) KNOWN SESSION-SPECIAL TEMPORARY INFORMATION (KSSTI) ATTACK

The random nonces u, v, and r_g are needed to generate the session key. Suppose that the random nonces u and v are compromised. In the message Msg_{gs}, a hash function using K_j and SID_j as parameters is used to mask r_g in D_3. Without knowing K_j and SID_j, the attacker still cannot calculate the session key because he cannot recover r_g via D_3. Thus, our scheme is able to resist *KSSTI* attack.

10) MANY LOGGED-IN USERS WITH THE SAME LOGIN-ID ATTACK

In our scheme, if there are two different users U_a and U_b with the same identity and password, they may both plan to log in to the system. Accordingly, they must enter their identity and password, and imprint biometrics. Because their biometrics are not the same, $a_{ia} = C_{ia} \oplus h(ID_i|| PW_i||H(BIO_{ia}))$ mod m generated by U_a and $a_{ib} = C_{ib} \oplus h(ID_i|| PW_i||H(BIO_{ib}))$ mod m generated by U_b are not equal due to the fuzzy verifier. Therefore, our scheme is secure from many logged-in users with the same login-ID attack.

11) MAN-IN-THE-MIDDLE ATTACK

The attacker can intercept the messages Msg_{ug}, Msg_{gs}, Msg_{sg}, and Msg_{gu} from the public channel. However, without the knowledge of θ, b_i, K_j, K_i, it is impossible for him to forge the valid messages Msg_{ug}, Msg_{gs}, Msg_{sg}, Msg_{gu} to deceive any communication participant. Thus, our scheme can defend against man-in-the-middle attack.

12) DENIAL OF SERVICE ATTACK

In our scheme, for the sake of preventing duplicate registration and improving the security of the protocol, the user's secret information is stored in the gateway. However, the user's identity in the entries we save about users is not saved in plain text such as in [31], but in the form of h (ID_i), so that what the insiders see is a 160-bit binary string. It is impossible to determine whether it belongs to a specific user U_i. Therefore, in our scheme, it is difficult for an attacker to launch a denial of service attack on a specific user U_i.

13) COMPARISON OF SECURITY PROPERTIES

To better understand the security of our improved protocol, we compare it with related schemes [31], [33], [35], [36], [42], [57], [58] in terms of security properties, and the results are summarized in Table 3. From Table 3, it can be seen that our improved scheme has thwarted the security defects in [42], while other schemes suffer from some serious security risks to some extent. They can neither resist certain attacks nor provide some functionality features, e.g., protocols in [33], [42] cannot provide forward secrecy, which

TABLE 3. Comparison of security properties and algorithm.

	[35]	[31]	[42]	[36]	[33]	[57]	[58]	Proposed Scheme
S1	√	√	×	√	N/A	×	×	√
S2	√	√	×	×	×	×	×	√
S3	×	√	×	√	√	√	×	√
S4	√	√	√	√	√	√	√	√
S5	√	√	×	√	√	√	√	√
S6	√	√	×	√	×	√	√	√
S7	√	×	√	√	×	√	×	√
S8	√	×	×	×	√	√	√	√
S9	√	√	×	√	√	√	√	√
S10	√	√	√	√	√	√	√	√
S11	√	×	√	×	√	×	√	√
S12	√	√	√	√	√	√	√	√
S13	√	√	√	√	×	√	√	√
S14	√	√	√	×	√	√	√	√
S15	√	×	√	√	√	√	√	√
Al	*CCM*	*ECC*	*Hash*	*CCM*	*ECC*	*Hash*	*ECC*	*CCM*

S1: Provide three-factor security. *S2*: Resist user impersonation attack. *S3*: Resist *GWN* impersonation attack. *S4*: Resist sensor node impersonation attack. *S5*: Resist desynchronization attack. *S6*: Provide forward secrecy. *S7*: Resist replay attack. *S8*: Resist Privileged insider attack. *S9*: Resist sensor node capture attack. *S10*: User anonymity and untraceability. *S11*: Resist *KSSTI* attack. *S12*: Mutual authentication and key agreement. *S13*: Resist many logged-in users with the same login-ID attack. *S14*: Resist man-in-the-middle attack. *S15*: Resist denial of service attack. *Al*: Algorithm. *CCM*: Chebyshev chaotic mapping

means that the loss cannot be minimized when the system is broken, and schemes [31], [36], [57] cannot defend against KSTTI attack, indicating that some temporary information leakage will provide disclosure of the session key between U_i and S_j. In particular, although schemes [42], [57], [58] employ three factors (password, smartcard, biometrics) to ensure the security of the authentication process, they actually fail to ensure three-factor security as revealed by the analysis method mentioned in Section IV.

VII. PERFORMANCE ANALYSIS

This section compares the performance of the improved scheme in light of computation cost, communication cost and the traffic of sensor with the competitive protocols [31], [33], [35], [36], [42], [57], [58].

A. COMPUTATION COST

We analyze the computation cost of our improved scheme and the related schemes [31], [33], [35], [36], [42], [57], [58] during the authentication phase. For the sake of analysis, we follow the execution time of various operations in [35], [36], [59] as the benchmark which is summarized in Table 4 to evaluate the computation cost. It is worth noting that we ignore the *XOR* operation since its execution time is negligible. We compare the computation cost in Table 5. Meanwhile, for ease of understanding, we illustrate Table 5 in Figure 2. Consequently, although our scheme's computation time is higher than that of protocols [42], [57] which are hash-based approaches, it is still more efficient than the other four schemes [31], [33], [35], [58].

B. COMMUNICATION COST

Referring to [35], [59], [60], we set the length of the Chebyshev polynomial, the points on the elliptic curve, hash value, random nonce, identity of user and sensor node, timestamp, and block of symmetric encryption/decryption are 128 bits, 320 bits, 160 bits, 128 bits, 32 bits, 32 bits, 128 bits, respectively. During the authentication process, the improved scheme transmits four messages $\{D_1, D_2, M_{UG}, T_1\}$, $\{D_1, D_3, M_{GS}, T_2\}$, $\{D_4, M_{SG}, T_3\}$, and $\{D_4, D_5, M_{GU}, T_4\}$, which require ($128 + 128 + 160 + 32 = 448$ bits), ($128 + 160 + 160 + 32 = 480$ bits), ($128 + 160 + 32 = 320$ bits), and ($128 + 128 + 160 + 32 = 480$ bits), respectively. The comparison results of communication cost for the protocols are presented in Table 6 and Figure 3. It can be observed that our scheme is obviously the most efficient one in communication cost among these schemes.

C. TRAFFIC OF SENSOR NODES

Because the wireless sensors are energy scant, and their throughput can measure the energy consumption to some extent [25], we compare the traffic of the sensor node of these

TABLE 4. Computing time of various operations.

Symbols	Meanings	Time (ms)
T_h	time of a hash function	0.5
T_F	time of Rep function in fuzzy extractor	63.08
T_P	time of an elliptic curve point multiplication	63.08
T_s	time of symmetric encryption/decryption	0.1303
T_C	time of computing a Chebyshev polynomial	21.02
T_A	time of an elliptic point addition	7.01
T_{BH}	time of a biometric hash function	21.02
T_M	time of a modular square operation	1.89
T_Q	time of a quadratic residue operation	3.84

TABLE 5. Comparison of computation cost.

	[35]	[31]	[42]	[36]	[33]	[57]	[58]	Proposed Scheme
U_i	T_F+6T_h+3T_C	10T_h +3T_P	T_F+12T_h	5T_h+2T_C+T_{BH}+T_M	16T_h+4T_P+2T_S	T_F+11T_h	5T_h +3T_P +3T_A	11T_h +3T_C
GWN	7T_h +T_C	8T_h +T_P	15T_h	6T_h +T_Q	5T_h +3T_P +2T_S	11T_h	8T_h +3T_P +4T_A	9T_h + T_C
S_j	4T_h +2T_C	4T_h+2T_P	6T_h	4T_h+2T_C	3T_h +2T_P	6T_h	4T_h +3T_P +2T_A	4T_h+2T_C
Total cost	T_F+17T_h +6T_C	22T_h +6T_P	T_F+33T_h	15T_h+4T_C+T_{BH}+T_M+T_Q	24T_h+9T_P+4T_S	T_F+28T_h	17T_h +7T_P +7T_A	19T_h +6T_C
Computation cost (ms)	197.7	389.48	79.58	118.33	299.12	77.08	499.13	135.62

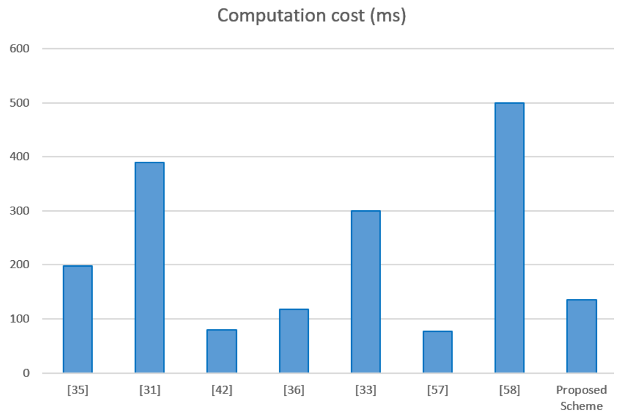

FIGURE 2. Comparison of computation cost.

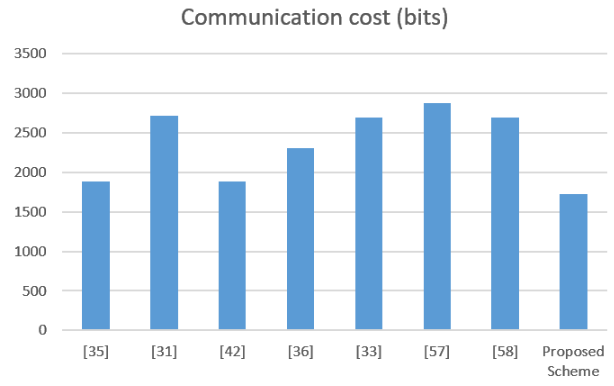

FIGURE 3. Comparison of communication cost.

TABLE 6. Comparison of communication cost.

Schemes	Communication cost (bits)
[35]	1888
[31]	2720
[42]	1888
[36]	2304
[33]	2688
[57]	2880
[58]	2688
Proposed Scheme	1728

schemes to understand the energy consumption of the sensor node. In our improved scheme, the received message $\{D_1, D_3, M_{GS}, T_2\}$ and the sent message $\{D_4, M_{SG}, T_3\}$ of the sensor require 480 bits and 320 bits, respectively. Table 7 and Figure 4 present comparison results about the traffic of sensor nodes of schemes [31], [33], [35], [36], [42], [57], [58]. It is evident that the sensor of the improved scheme consumes the least traffic among the related protocols which indicates that our scheme can prolong the lifetime of the sensor node more than other protocols.

It is a remarkable fact that in user authentication protocols, security is one of the most valued factors among all aspects to be considered in the design process. Although our scheme is not the most efficient protocol in terms of computation cost and the communication overhead, the sensor traffic of our protocol is lower than those of others, and moreover, our proposal can thwart the security flaws of other protocols and fulfill more security properties. Therefore, our scheme outperforms other protocols in overall performance.

TABLE 7. Comparison of the traffic of sensor nodes.

Schemes	Traffic of sensor node (bits)
[35]	1344
[31]	1280
[42]	864
[36]	800
[33]	1344
[57]	1120
[58]	1344
Proposed Scheme	800

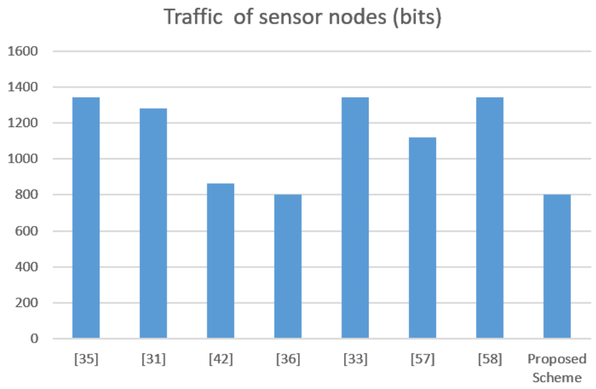

FIGURE 4. Comparison of sensor nodes traffic.

VIII. CONCLUSION

In this work, we cryptanalyze Shin and Kwon's three-factor authentication scheme and point out its security defects such as failure to provide three-factor security, user impersonation attack, desynchronization attack, sensor node capture attack, privileged insider attack. To eliminate these defects, we suggest an improved secure three-factor anonymous authentication protocol for WSN using Chebyshev chaotic mapping. Furthermore, we demonstrate that the improved scheme is secure against various known attacks among the communication participants during the authentication process by presenting a security proof under random oracle model and a security simulation verification via ProVerif. Finally, the comprehensive comparison with the competitive schemes in terms of security and performance shows that our scheme has advantages over them.

REFERENCES

[1] I. Khemapech, I. Duncan, and A. Miller, "A survey of wireless sensor networks technology," in *Proc. 6th Annu. Postgraduate Symp. Converg. Telecommun., Netw. Broadcast.*, vol. 13, 2005, pp. 1–6.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, 2002.

[3] Z. Cui, F. Xue, S. Zhang, X. Cai, Y. Cao, W. Zhang, and J. Chen, "A hybrid BlockChain-based identity authentication scheme for multi-WSN," *IEEE Trans. Services Comput.*, vol. 13, no. 2, pp. 241–251, Mar./Apr. 2020.

[4] K. Sowjanya, M. Dasgupta, and S. Ray, "An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems," *Int. J. Inf. Secur.*, vol. 19, no. 1, pp. 129–146, Feb. 2020.

[5] F. G. Darbandeh and M. Safkhani, "A new lightweight user authentication and key agreement scheme for WSN," *Wireless Pers. Commun.*, vol. 114, no. 4, pp. 3247–3269, Oct. 2020.

[6] C. Wang, D. Wang, Y. Tu, G. Xu, and H. Wang, "Understanding node capture attacks in user authentication schemes for wireless sensor networks," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 507–523, Jan./Feb. 2020.

[7] J. Lee, S. Yu, M. Kim, Y. Park, and A. K. Das, "On the design of secure and efficient three-factor authentication protocol using honey list for wireless sensor networks," *IEEE Access*, vol. 8, pp. 107046–107062, 2020.

[8] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1086–1090, Mar. 2009.

[9] D. Nyang and M.-K. Lee, "Improvement of Das's two-factor authentication protocol in wireless sensor networks," IACR Cryptol. ePrint Arch., vol. 2009, p. 631, 2009. [Online]. Available: https://eprint.iacr.org/2009/631.pdf

[10] H.-F. Huang, Y.-F. Chang, and C.-H. Liu, "Enhancement of two-factor user authentication in wireless sensor networks," in *Proc. 6th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Oct. 2010, pp. 27–30.

[11] D. He, Y. Gao, S. Chan, C. Chen, and J. Bu, "An enhanced two-factor user authentication scheme in wireless sensor networks," *Ad Hoc Sensor Wireless Netw.*, vol. 10, no. 4, pp. 361–371, Jan. 2010.

[12] B. Vaidya, D. Makrakis, and H. Mouftah, "Two-factor mutual authentication with key agreement in wireless sensor networks," *Secur. Commun. Netw.*, vol. 9, no. 2, pp. 171–183, Jan. 2016.

[13] F. Wu, L. Xu, S. Kumari, and X. Li, "A new and secure authentication scheme for wireless sensor networks with formal proof," *Peer-Peer Netw. Appl.*, vol. 10, no. 1, pp. 16–30, Jan. 2017.

[14] A. Singh, A. K. Awasthi, and K. Singh, "Cryptanalysis and improvement in user authentication and key agreement scheme for wireless sensor network," *Wireless Pers. Commun.*, vol. 94, no. 3, pp. 1881–1898, Jun. 2017.

[15] P. Chandrakar, "A secure remote user authentication protocol for healthcare monitoring using wireless medical sensor networks," *Int. J. Ambient Comput. Intell.*, vol. 10, no. 1, pp. 96–116, Jan. 2019.

[16] D. Wang and P. Wang, "Understanding security failures of two-factor authentication schemes for real-time applications in hierarchical wireless sensor networks," *Ad Hoc Netw.*, vol. 20, pp. 1–15, Sep. 2014.

[17] D. Wang, W. Li, and P. Wang, "Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 9, pp. 4081–4092, May 2018.

[18] X. Li, J. Peng, J. Niu, F. Wu, J. Liao, and K. R. Choo, "A robust and energy efficient authentication protocol for industrial Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1606–1615, Jun. 2018.

[19] A. K. Das, "A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks," *Peer-Peer Netw. Appl.*, vol. 9, no. 1, pp. 1–22, 2014.

[20] F. Wu, L. Xu, S. Kumari, and X. Li, "An improved and provably secure three-factor user authentication scheme for wireless sensor networks," *Peer-Peer Netw. Appl.*, vol. 11, no. 1, pp. 1–20, Jan. 2018.

[21] A. K. Das, "A secure and efficient user anonymity-preserving three-factor authentication protocol for large-scale distributed wireless sensor networks," *Wireless Pers. Commun.*, vol. 82, no. 3, pp. 1377–1404, Jan. 2015.

[22] A. K. Das, "A secure and effective biometric-based user authentication scheme for wireless sensor networks using smart card and fuzzy extractor," *Int. J. Commun. Syst.*, vol. 30, no. 1, p. e2933, Jan. 2017.

[23] Y. Lu, G. Xu, L. Li, and Y. Yang, "Anonymous three-factor authenticated key agreement for wireless sensor networks," *Wireless Netw.*, vol. 25, no. 4, pp. 1461–1475, May 2019.

[24] J. Mo and H. Chen, "A lightweight secure user authentication and key agreement protocol for wireless sensor networks," *Secur. Commun. Netw.*, vol. 2019, pp. 1–17, Dec. 2019.

[25] R. Amin, S. K. H. Islam, G. P. Biswas, M. K. Khan, L. Leng, and N. Kumar, "Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks," *Comput. Netw.*, vol. 101, pp. 42–62, Jun. 2016.

[26] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment," *Ad Hoc Netw.*, vol. 36, pp. 152–176, Jan. 2016.

[27] Q. Jiang, S. Zeadally, J. Ma, and D. He, "Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks," *IEEE Access*, vol. 5, pp. 3376–3392, 2017.

[28] R. Amin, S. K. H. Islam, G. P. Biswas, M. K. Khan, and N. Kumar, "A robust and anonymous patient monitoring system using wireless medical sensor networks," *Future Gener. Comput. Syst.*, vol. 80, pp. 483–495, Mar. 2018.

[29] Q. Jiang, J. Ma, C. Yang, X. Ma, J. Shen, and A. C. Shehzad, "Efficient end-to-end authentication protocol for wearable health monitoring systems," *Comput. Elect. Eng.*, vol. 63, pp. 182–195, Oct. 2017.

[30] J. Mo, W. Shen, and W. Pan, "An improved anonymous authentication protocol for wearable health monitoring systems," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–13, Apr. 2020.

[31] X. Li, J. Peng, M. S. Obaidat, F. Wu, M. K. Khan, and C. Chen, "A secure three-factor user authentication protocol with forward secrecy for wireless medical sensor network systems," *IEEE Syst. J.*, vol. 14, no. 1, pp. 39–50, Mar. 2020.

[32] C.-G. Ma, D. Wang, and S.-D. Zhao, "Security flaws in two improved remote user authentication schemes using smart cards," *Int. J. Commun. Syst.*, vol. 27, no. 10, pp. 2215–2227, Oct. 2014.

[33] M. F. Moghadam, M. Nikooghadam, M. A. B. A. Jabban, M. Alishahi, L. Mortazavi, and A. Mohajerzadeh, "An efficient authentication and key agreement scheme based on ECDH for wireless sensor network," *IEEE Access*, vol. 8, pp. 73182–73192, 2020.

[34] D. K. Kwon, S. J. Yu, J. Y. Lee, S. H. Son, and Y. H. Park, "WSN-SLAP: Secure and lightweight mutual authentication protocol for wireless sensor networks," *Sensors*, vol. 21, no. 3, p. 936, Jan. 2021.

[35] F. Wang, G. Xu, and G. Xu, "A provably secure anonymous biometrics-based authentication scheme for wireless sensor networks using chaotic map," *IEEE Access*, vol. 7, pp. 101596–101608, 2019.

[36] G. Xu, F. Wang, M. Zhang, and J. Peng, "Efficient and provably secure anonymous user authentication scheme for patient monitoring using wireless medical sensor networks," *IEEE Access*, vol. 8, pp. 47282–47294, 2020.

[37] J. Kim, D. Lee, W. Jeon, Y. Lee, and D. Won, "Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks," *Sensors*, vol. 14, no. 4, pp. 6443–6462, Apr. 2014.

[38] I.-P. Chang, T.-F. Lee, T.-H. Lin, and C.-M. Liu, "Enhanced two-factor authentication and key agreement using dynamic identities in wireless sensor networks," *Sensors*, vol. 15, no. 12, pp. 29841–29854, 2015.

[39] Y. Park and Y. Park, "Three-factor user authentication and key agreement using elliptic curve cryptosystem in wireless sensor networks," *Sensors*, vol. 16, no. 12, p. 2123, 2016.

[40] J. Jung, J. Moon, D. Lee, and D. Won, "Efficient and security enhanced anonymous authentication with key agreement scheme in wireless sensor networks," *Sensors*, vol. 17, no. 3, p. 644, Mar. 2017.

[41] C. Wang, G. Xu, and J. Sun, "An enhanced three-factor user authentication scheme using elliptic curve cryptosystem for wireless sensor networks," *Sensors*, vol. 17, no. 1, p. 2946, 2017.

[42] S. Shin and T. Kwon, "A lightweight three-factor authentication and key agreement scheme in wireless sensor networks for smart homes," *Sensors*, vol. 19, no. 9, p. 2012, Apr. 2019.

[43] L. Kocarev and Z. Tasev, "Public-key encryption based on Chebyshev maps," in *Proc. Int. Symp. Circuits Syst. (ISCAS)*, vol. 3, 2003, p. 3.

[44] J. C. Mason and D. C. Handscomb, *Chebyshev Polynomials*. Boca Raton, FL, USA: CRC Press, 2002.

[45] 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.

[46] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[47] S. Mangard, E. Oswald, and T. Popp, *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Berlin, Germany: Springer, 2008.

[48] T. H. Kim, C. Kim, and I. Park, "Side channel analysis attacks using AM demodulation on commercial smart cards with SEED," *J. Syst. Softw.*, vol. 85, no. 12, pp. 2899–2908, Dec. 2012.

[49] 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.

[50] J. Bonneau, "The science of guessing: Analyzing an anonymized corpus of 70 million passwords," in *Proc. IEEE Symp. Secur. Privacy*, May 2012, pp. 538–552.

[51] M. Dell'Amico, P. Michiardi, and Y. Roudier, "Password strength: An empirical analysis," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[52] A. K. Das, A. K. Sutrala, V. Odelu, and A. Goswami, "A secure smartcard-based anonymous user authentication scheme for healthcare applications using wireless medical sensor networks," *Wireless Pers. Commun.*, vol. 94, no. 3, pp. 1899–1933, 2017.

[53] W. Li, B. Li, Y. Zhao, P. Wang, and F. Wei, "Cryptanalysis and security enhancement of three authentication schemes in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–11, Jul. 2018.

[54] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proc. 1st ACM Conf. Comput. Commun. Secur. (CCS)*, 1993, pp. 62–73.

[55] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 708–722, Jul./Aug. 2018.

[56] D. Wang, D. He, P. Wang, and C.-H. Chu, "Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 4, pp. 428–442, Jul./Aug. 2015.

[57] S. Yu and Y. Park, "SLUA-WSN: Secure and lightweight three-factor-based user authentication protocol for wireless sensor networks," *Sensors*, vol. 20, no. 15, p. 4143, Jul. 2020.

[58] D. Rangwani, D. Sadhukhan, S. Ray, M. K. Khan, and M. Dasgupta, "A robust provable-secure privacy-preserving authentication protocol for industrial Internet of Things," *Peer-Peer Netw. Appl.*, vol. 14, no. 3, pp. 1548–1571, May 2021.

[59] J. Mo, Z. Hu, and Y. Lin, "Remote user authentication and key agreement for mobile client–server environments on elliptic curve cryptography," *J. Supercomput.*, vol. 74, no. 11, pp. 5927–5943, Nov. 2018.

[60] B. Blanchet, "An efficient cryptographic protocol verifier based on prolog rules," in *Proc. IEEE Comput. Soc. Found. (CSFW)*, Jun. 2001, pp. 82–96.

[61] T.-Y. Wu, L. Yang, Z. Lee, S.-C. Chu, S. Kumari, and S. Kumar, "A provably secure three-factor authentication protocol for wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–15, Apr. 2021, doi: 10.1155/2021/5537018.

[62] D. Kumar, "A secure and efficient user authentication protocol for wireless sensor network," *Multimedia Tools Appl.*, vol. 80, no. 1, pp. 1–24, 2021.

[63] X. Liu, Z. Guo, J. Ma, and Y. Song, "A secure authentication scheme for wireless sensor networks based on DAC and Intel SGX," *IEEE Internet Things J.*, early access, Jul. 19, 2021, doi: 10.1109/JIOT.2021.3097996.

[64] Q. Xie, Z. Ding, and B. Hu, "A secure and privacy-preserving three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things," *Secur. Commun. Netw.*, vol. 2021, pp. 1–12, Sep. 2021, doi: 10.1155/2021/4799223.

[65] A. Jabbari and J. B. Mohasefi, "Improvement of a user authentication scheme for wireless sensor networks based on Internet of Things security," *Wireless Pers. Commun.*, vol. 116, no. 3, pp. 2565–2591, Feb. 2021.