A Secure and Privacy-Preserving Three-Factor Anonymous Authentication Scheme for Wireless Sensor Networks in Internet of Things

Qi Xie, Zixuan Ding, and Bin Hu

Key Laboratory of Cryptography of Zhejiang Province, Hangzhou Normal University, Hangzhou 311121, China

Correspondence should be addressed to Qi Xie; qixie68@126.com

Received 23 July 2021; Revised 16 August 2021; Accepted 24 August 2021; Published 29 September 2021

1. Introduction

In recent years, with the rapid development of Internet of things (IoT) technology, wireless sensor networks (WSN) are widely used in medical, military, agriculture, and other fields [1]. A large number of wireless sensor nodes are deployed in the target fields to collect the data in WSN, but sensor nodes have low computing power and low storage capacity and are easy to be captured; on the contrary, compared with the traditional wired network, messages are transmitted through wireless channels, and it may be easily attacked by means of eavesdropping, capture, replay, forgery, and so on. In order to protect the data security and users’ privacy, it is very important to design secure and privacy-preserving authentication and key agreement protocol for WSN in IoT.

Many authentication protocols have been proposed in the past ten years; however, these protocols exist one or more security flaws [2]. In 2013, Li et al. proposed a communication scheme in IoT [3], which provides authentication, integrity, nonrepudiation, and confidentiality. However, this scheme is based on bilinear pairing, so it is hard to be deployed in WSN [4]. In 2014, Turkanović et al. [5] proposed a hash function-based authentication scheme for WSN. Farash et al. [6] pointed out that it suffers from impersonation attack, smart card loss attack, and session key disclosure attack; then, Farash et al. designed a new two-factor authentication (2FA) protocol. Amin and Biswas [7] also showed that Turkanović et al.’s scheme [5] suffers from offline password-guessing attacks and impersonation attacks, and Amin et al. proposed a 2FA protocol for multi-gateway WSN. Meanwhile, Amin et al. found that, in Farash et al.’s [6] scheme, there exists some security flaws, such as impersonation attack, smart card loss attack, and offline password-guessing attack.

In order to improve the security of authentication protocol, Diffie–Hellman key agreement algorithm, Chebyshev chaotic map [8], and elliptic curve cryptography (ECC) are used to design secure user authentication and key agreement protocol [9, 10]. In 2009, Das [11] proposed an

Copyright © 2021 Qi Xie et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Internet of things is playing more and more important role in smart healthcare, smart grids, and smart transportation, and using wireless sensor network (WSN), we can easily obtain and transmit information. However, the data security and users’ privacy are the biggest challenges for WSN because sensor nodes have low computing power and low storage capacity and are easy to be captured, and wireless networks are vulnerable. In 2021, Shuai et al. proposed a lightweight three-factor anonymous authentication scheme for WSN. However, we found that their protocol is vulnerable to stolen-verifier attack, modification of messages’ attack, and no perfect forward secrecy. Then, a new three-factor anonymous authentication scheme using elliptic curve cryptography (ECC) is proposed. Through informal and formal security analyses, our scheme can resist various known attacks and maintains low computational complexity.
authentication protocol based on ECC for WSN, but their scheme suffers from privilege insider attacks and gateway bypass attacks [12]. Later, Kumar et al. [13] proposed an efficient authentication protocol for WSN. He et al. [14] showed that their scheme suffers from offline password-guessing attack and privilege insider attacks. To overcome these security flaws, they proposed an improved authentication scheme for WSN. Unfortunately, Li et al. [15], Wu et al. [16], and Mir et al. [17] pointed out that He et al.’s scheme is still insecure, and it may suffer from offline password-guessing attack and impersonation attack. Therefore, Li et al. [15] proposed a three-factor authentication (3FA) scheme to overcome these flaws because two-factor authentication (2FA) schemes usually suffer from offline password-guessing attacks [18]. Compared with 2FA schemes, 3FA schemes can improve the security because 3FA schemes use biometrics to avoid password-guessing attacks. Yeh et al. [19] and Chang and Hai [20] proposed 3FA schemes for WSN to resist various known attacks, but these schemes suffer from smart card loss attacks, impersonation attacks, and so on. So, Challa et al. [21] proposed the signature-based authentication scheme to achieve security, but the computation cost is high. In 2021, Tanveer et al. [22] proposed a lightweight user authentication and key exchange scheme for smart home, and Xie et al. [23] designed an ECC-based secure and privacy-protected authentication protocol for smart city. Shuai et al. [24] proposed a 3FA scheme for WSN, which uses a bio-hash function to enhance security.

1.1. Motivations and Contributions. In 2021, Shuai et al. [24] proposed a lightweight 3FA anonymous authentication scheme; however, we pointed out that Shuai et al.’s scheme is vulnerable to stolen-verifier attack, modification of messages attack, and no perfect forward security. To solve these problems, we propose a new 3FA scheme based on ECC and Fuzzy Extractor algorithm. We summarize our contributions as follows:

1. We pointed out that Shuai et al.’s scheme suffers from the stolen-verifier attack, modification of messages attack, and no perfect forward security
2. A new three-factor authentication scheme based on ECC and fuzzy extractor algorithm used for WSNs is proposed
3. We use formal verification tool ProVerif [25] which is based on applied pi calculus to prove the security of the proposed scheme
4. The informal security analysis shows that the proposed scheme can resist various known attacks
5. We evaluate the computational cost of the proposed scheme with some related schemes; the result shows that the proposed scheme has better performance

1.2. Attack Model. Referring to the Dolev-Yao threat model [26], we present the abilities of an adversary as follows:

1. $U_A$ has the ability to eavesdrop on all the messages which are transmitted via an open channel
2. $U_A$ can modify, insert, replay, modify, and reroute the eavesdropped messages
3. If $U_A$ obtains the smart card of the user $U_i$, he/she can get all the data kept in the smart card
4. $U_A$ can obtain all data stored in sensor node if $U_A$ captures a sensor node
5. $U_A$ maybe an insider attacker

The rest of the paper is as follows. We review the scheme of Shuai et al. in Section 2. Section 3 shows the security analysis of Shuai et al.’s scheme. We propose the new scheme in Section 4. Sections 5 and 6 present the informal and formal security analyses of the proposed scheme. In Section 7, we exhibit the performance analysis between the proposed scheme and some related schemes. Finally, the paper concludes in Section 8.

2. Review the Shuai et al.’s Scheme

Shuai et al.’s scheme [24] consists of three phases: registration phase, login and authentication phase, and password change phase.

2.1. Registration Phase. The registration phase includes user (may be health professional) registration and medical sensor node registration. The user registration phase is as follows:

Step UR1: the user $U_i$ chooses identity ID, and inputs password PW$_i$ and fingerprint $f_{gi}$ via the sensor device; the device generates a random number $m_i$. After that, the device computes $MB_i = BH(m_i, f_{gi})$ and $MPW_i = h(ID_i||PW_i||MB_i||m_i)$ and then sends $\{ID_i, MPW_i\}$ and the personal credential to GWN via a private channel.

Step UR2: once the message is received, GWN generates random numbers $n_i$, $r_i$, and $K_1$ and computes $HID_i = ID_i@r_i$, $X_i = h(ID_i,K_1||r_i)$, $Y_i = X_i@MPW_i$, and $V_i = h(X_i||MPW_i)$. GWN stores $\{ID_i, HID_i, n_i, K_1\}$ and user’s credential in its memory and stores $\{HID_i, Y_i, V_i, K_1, h(\cdot), BH(\cdot)\}$ into a smart card; GWN issues the smart card to $U_i$ via a private channel.

Step UR3: once the smart card is received, $U_i$ writes $m_i$ into the smart card. At the end of the user registration phase, the smart card contains $\{HID_i, Y_i, V_i, K_1, h(\cdot), BH(\cdot), m_i\}$.

The registration phase of sensor node is as follows.

Step SR1: the medical sensor node $SN_j$ chooses identity SID$_j$ and sends it to GWN via a private channel.

Step SR2: on receiving SID$_j$, GWN first checks the uniqueness of the SID$_j$; if the SID$_j$ is not unique, it refuses the registration request. Otherwise, GWN generates a random number $K_2$ and stores $\{SID_j, K_2\}$ in its memory. Then, GWN transmits $K_2$ to $SN_j$ via a private channel.

Step SR3: on receiving $K_2$, $SN_j$ stores $K_2$. 

Security and Communication Networks
2.2. Login and Authentication Phase

Step LA1: the user $U_i$ inserts the smart card and enters identity ID$_i$, password PW$_i$, and fingerprint $f_{g_i}$. The smart card computes $MB^*_i = BH(m_i\|f_{g_i})$, MPW$_i^* = h(ID_i\|PW_i\|MB^*_i\|m_i)$, $X^*_i = Y_i\oplus MPW^*_i$, and $V^*_i = h(X^*_i\|MPW^*_i)$ and checks if $V^*_i$ and $V_i$ are equal. If not, it terminates the session. Otherwise, proceed to the next step.

Step LA2: if the user $U_i$ is legal, the smart card generates a random number $R$ and current timestamp $T_i$; $U_i$ selects an identity SID$_i$ of sensor node SN$_j$; the smart card computes $UG = h(HID_i\|X_i\|K^*_i)$, $M_1 = E_{UG}(R|SID_i)$, and $CK^*_i = h(ID_i\|R\|X_i\|SID_i\|K^*_i\|T_i)$. Then, $U_i$ sends message $\{HID_i, M_1, CK^*_i, T_i\}$ to GWN via a public channel.

Step LA3: on receiving the message from $U_i$, GWN checks the time stamp $T_i$ first. GWN gets the current time $T^*_i$ and compares with $T_i$ if $|T^*_i - T_i| > \Delta T$, where $\Delta T$ is the predefined threshold value, and GWN terminates the session. Otherwise, according to HID$_i$, GWN extracts identity ID$_i$, random number $r_i$, and $K^*_i$ of user $U_i$ from the storage table. Then, GWN computes $X^*_i = h(ID_i\|K^*_i\|m_i)$, $UG = h(HID_i\|X^*_i\|K^*_i)$, $R^* = SID_i = D_{UG}(M_1)$, and $CK^*_i = h(ID_i\|R^*\|X^*_i\|SID_i\|K^*_i\|T_i)$ and compares $CK^*_i$ with $CK^*_i$. If they are not equal, terminate the session. Otherwise, the user $U_i$ is legal. In addition, GWN generates a timestamp $T^*_2$ and session key SK and computes $M_2 = (SK||ID_i)\oplus h(K^*_i||SID_i)$ and $CK^*_2 = h(ID_i\|SID_i\|SK\|K^*_i\|T^*_2)$. Finally, GWN sends the message $\{M_2, CK^*_2, T^*_2\}$ to sensor node SN$_j$ via an open channel.

Step LA4: on receiving the message $\{M_2, CK^*_2, T^*_2\}$, SN$_j$ gets the current time $T^*_2$ and compares with $T^*_2$ if $|T^*_2 - T^*_2| > \Delta T$, terminate the session. Otherwise, SN$_j$ computes $SK||ID_i)\oplus h(K^*_i||SID_i)$ and $CK^*_2 = h(ID_i\|SID_i\|SK\|K^*_i\|T^*_3)$. Then SN$_j$ compares $CK^*_2$ with $CK^*_2$. If they are not equal, terminate the session. Otherwise, SN$_j$ generates a timestamp $T^*_3$ and computes $CK^*_3 = h(SID_i\|ID_j\|SK\|T^*_3)$. Finally, SN$_j$ updates $K^*_2 = h(K^*_2)$ and sends the message $\{CK^*_3, T^*_3\}$ to GWN via an open channel.

Step LA5: on receiving the message $\{CK^*_3, T^*_3\}$, GWN gets the current time $T^*_3$; if $|T^*_3 - T^*_3| < \Delta T$, compute $CK^*_3 = h(SID_i\|ID_j\|SK\|T^*_3)$. Then, GWN compares $CK^*_3$ with $CK^*_3$. If they are not equal, terminate the session. Otherwise, GWN generates a random number $r^*_3$ and $T^*_4$ and computes $HID^*_i = ID_j\oplus r^*_3$, $GU = h(R\|HID^*_i\|X_i\|K^*_i)$, $M_4 = E_{GU}(SK\|HID^*_i\|SID_i)$, and $CK^*_4 = h(SK\|HID^*_i\|SID_i\|HID_i\|T^*_5)$. Then, GWN updates $K^*_i$, SID$_i$, and HID$_i$ with $K_3 = h(K_3)$, $K_2 = h(K_2)$, and $HID_i = HID^*_i$. Finally, GWN sends the message $\{M_3, CK^*_4, T^*_4\}$ to $U_i$ via an open channel.

Step LA6: on receiving the message $\{M_3, CK^*_4, T^*_4\}$, $U_i$ gets the current time $T^*_4$; if $|T^*_4 - T^*_4| < \Delta T$, compute $GU = h(R\|HID^*_i\|X_i\|K^*_i)$, $(SK\|HID^*_i\|SID_i) = D_{GU}(M_4)$, and $CK^*_4 = h(ID_i\|SK\|HID^*_i\|T^*_4)$. Then, $U_i$ compares $CK^*_4$ with $CK^*_4$. If they are equal, $U_i$ updates $K_i$ and HID$_i$ with $K_1 = h(K_1)$ and HID$_i = HID^*_i$ and completes the authentication.

2.3. Password Change Phase

Step PC1: the user $U_i$ inserts the smart card and enters identity ID$_i$, password PW$_i$, and fingerprint $f_{g_i}$. The smart card computes $MB^*_i = BH(m_i\|f_{g_i})$, MPW$_i^* = h(ID_i\|PW_i\|MB^*_i\|m_i)$, $X^*_i = Y_i\oplus MPW^*_i$, and $V^*_i = h(X^*_i\|MPW^*_i)$ and compares $V^*_i$ with $V_i$, which is stored in the smart card. If the values are equal, the smart card allows $U_i$ to enter a new password PW$_i$. Otherwise, it rejects the request for password change.

Step PC2: the smart card computes $MPW^*_i = h(ID_i\|PW^*_i\|MB^*_i\|m_i)$, $Y^*_i = X_i\oplus MPW^*_i = Y_i\oplus MPW^*_i\oplus MPW^*_i$, and $V^*_i = h(X^*_i||MPW^*_i)$. Step PC3: finally, the smart card deletes $Y_i$ and $V_i$ and stores $Y^*_i$ and $V^*_i$.

3. Analysis of the Shuai et al.’s Scheme

In this section, we will show that Shuai et al.’s protocol has some security flaws.

3.1. Modification of Messages/Desynchronization Attack. In Shuai et al.’s scheme, SN$_j$ updates $K_2 = h(K_2)$ and sends the message $\{CK^*_3, T^*_3\}$ to GWN via an open channel in Step LA5. On receiving the message $\{CK^*_3, T^*_3\}$, GWN gets the current time $T^*_3$; if $|T^*_3 - T^*_3| < \Delta T$, compute $CK^*_3 = h(SID_i\|ID_j\|SID_i\|T^*_3)$. If $CK^*_3 = CK^*_3$, GWN updates $K_2 = h(K_2)$ otherwise. Suppose an attacker $U_A$ intercepts or changes information $\{CK^*_3, T^*_3\}$, GWN will not update $K_2 = h(K_2)$ before the session terminated. Therefore, SN$_j$ and GWN store different $K_2$. The sensor node SN$_j$ is paralyzed.

The same attack method can be used between GWN and the user $U_i$. If an attacker $U_A$ intercepts or changes information $\{M_3, CK^*_4, T^*_4\}$ between Step LA5 and Step LA6, $U_i$ will not update the value of $K_1$. However, GWN has updated $K_1$ already. Later on, $U_i$ cannot pass the authentication of GWN.

3.2. Stolen-Verifier Attack. In their scheme, GWN stores $\{SID_j, K_2\}$: SID$_j$ is the identity of sensor node SN$_j$; the random number $r_3$ is generated by GWN for the sensor node SN$_j$.

Assuming that SID$_j$ and $K_2$ of each node is known by the attacker $U_A$, $U_A$ can eavesdrop on $\{M_2, CK^*_2, T^*_2\}$ via an open channel. By computing $h(SK\|ID_j) = M_2\oplus h(K_2||SID_j)$, the attacker $U_A$ gets session key SK and user’s identity ID$_j$.

If attacker $U_A$ knows $\{SID_j, K_2\}$, he/she can intercept all messages and impersonate any sensor node. After knowing $\{M_3, CK^*_4, T^*_4\}$, $U_A$ can forge $\{M_3, CK^*_4, T^*_4\}$ and send the message to the sensor node SN$_j$, where $M_3 = (SK\|ID_j)\oplus h(K_2||SID_j)$ and $CK^*_4 = h(ID_j\|SID_j\|SK\|K^*_4\|T^*_4)$. SN$_j$, ID$_j$, and $T^*_2$ can be randomly generated by the attacker $U_A$. The sensor node verifies the message by computing $(SK\|ID_j) = M_2\oplus h(K_2||SID_j)$ and $CK^*_2 = h(ID_j\|SID_j||SK\|K^*_4\|T^*_4)$.
on the curve. GWN chooses a secret parameter defined over a large prime number and can be eavesdropped on by the attacker. The next long-term key $K^*_2$ is updated by $K^*_2 = h(K_3)$. It is easy for the attacker to eavesdrop next $M^*_2$ via an open channel; then, the next session key $K^*_2$ can be computed by $(SK^* || ID_j) = M^*_2 \oplus h(K^*_2 || ID_j)$. Therefore, the scheme of Shuai et al. cannot provide perfect forward/backward security.

4. Our Proposed Scheme

In this section, we propose a new three-factor anonymous authentication scheme using ECC and fuzzy extractor algorithm. Table 1 shows the notations and intuitive abbreviations mentioned in the proposed scheme.

4.1. System Setup Phase. GWN chooses an elliptic curve $E(GF(q))$ defined over $GF(q)$, where $GF(q)$ is a finite field defined over a large prime number $q$. $P$ is a generator point on the curve. GWN chooses a secret parameter $K_{GWN}$. GWN computes public key as $PK = K_{GWN} \cdot P$ and publishes $Rep(\cdot)$, $Gen(\cdot)$, $h(\cdot)$, and $PK_{GWN}$, where $Rep(\cdot)$ and $Gen(\cdot)$ are reproduction and generation algorithm of fuzzy extractor algorithm, respectively. $h(\cdot)$ is a hash function.

4.2. User Registration Phase

Step UR1: $U_i$ chooses its ID and sends ID to GWN via a private channel.

Step UR2: GWN verifies the effectiveness and legitimacy of ID; if not, GWN requests $U_i$ to choose a new ID. Otherwise, GWN computes $a_i = h(ID || K_{GWN})$. GWN stores the information $\{a_i, PK_{GWN}, P\}$ into a smart card (SC) and transmits it to $U_i$.

Step UR3: $U_i$ inserts the SC into a card reader and enters its ID, $PW_i$, and fingerprint $fng_i$; the device computes

\[ (\sigma_i, \tau_i) = Gen(fng_i), \]
\[ MPW_i = h(ID_i || PW_i || \sigma_i), \]
\[ F_i = a_i \oplus h(ID_i || \sigma_i || PW_i), \]

where $PK_{GWN}$ is the public key of GWN, $T_1$ is the current timestamp, and $U_i$ sends the message $\{MPW_i, \tau_i, F_i, PK_{GWN}, P\}$ are stored in SC.

4.3. Sensor Node Registration Phase

Step SR1: GWN chooses a unique identity SID, and computes $b_j = h(SID || K_{GWN})$. Then, GWN sends $\{b_j, SID, P\}$ to SN via a private channel.

Step SR2: upon receiving $\{b_j, SID, P\}$, SN stores them into its memory.

4.4. Login and Authentication Phases

Step LA1: $U_i$ inserts the smart card into the device and inputs the identity ID and the password PW and enters the fingerprint $fng$. Then, the device calculates

\[ a'_i = Rep(fng, \tau_i), \]
\[ MPW'_i = h(ID'_i || PW'_i || \sigma'_i), \]

If $MPW'_i \neq MPW_i$, SC refuses the login request of $U_i$. Otherwise, go on.

Step LA2: $U_i$ creates a random number $m_i$ and computes

\[ a''_i = F_i \oplus h(ID_i || \sigma || PW_i), \]
\[ m_1 = m_i \cdot P, \]
\[ M_1 = (ID_i || SID) \oplus h(m_1 \cdot PK_{GWN} || T_1), \]
\[ M_3 = h(a''_i \cdot M_1 || M_2 || T_1), \]

where $PK_{GWN}$ is the public key of GWN, $T_1$ is the current timestamp, And $U_i$ sends the message...
MES\(_i = \{M_1, M_2, M_3, T_1\}\) to GWN via a public channel.

Step LA3: on receiving MES\(_i\), GWN first checks the timestamp. GWN creates the current time \(T^*_1\); if \([T^*_1 - T_1] > \Delta T\), terminate the session. Otherwise, GWN computes

\[
(Id'_i, Sid'_i) = M_i \oplus h(M_i \cdot k_i \| T'_i),
\]

\[
a'_i = h(Id'_i \| K_{GW}),
\]

\[
M'_i = h(a'_i \| M_i \| M_2 \| T_1).
\]

If \(M'_i \neq M_3\), GWN declines the request. Otherwise, GWN generates the current time \(T_2\) and computes

\[
S_i = h(d_i \| Sid \| T_2),
\]

\[
b'_i = h(Sid'_i \| K_{GW}),
\]

\[
N_1 = (S_i \| Sid) \oplus h(b'_i \| Sid \| T_2),
\]

\[
N_2 = h(M_i \| Sid_i \| Sid \| T_2).
\]

GWN transmits the message MES\(_2 = \{M_1, N_1, N_2, T_2\}\). to SN\(_j\) via an open channel.

Step LA4: after obtaining the message MES\(_2 = \{M_1, N_1, N_2, T_2\}\), SN\(_j\) checks whether \([T^*_2 - T_2] \leq \Delta T\), where \(T^*_2\) is the current timestamp. If not, SN\(_j\) rejects the session. Otherwise, SN\(_j\) computes

\[
(S'_i \| Sid'_i) = N_1 \oplus h(b'_i \| Sid \| T_2),
\]

\[
N'_2 = h(M_i \| Id'_i \| Sid_i \| T_2).
\]

If \(N'_2 \neq N_2\), terminate the session. Otherwise, SN\(_j\) generates a random number \(c_j\) and the current time \(T_3\) and computes

\[
N_3 = c_j \cdot P,
\]

\[
SK = h(c_j \cdot M_i \| Sid \| Id'_i \| S'_i),
\]

\[
N_4 = h(S'_i \| SK \| N_3 \| Id'_i \| T_3).
\]

SN\(_j\) sends the message MES\(_3 = \{N_3, N_4, T_2, T_3\}\) to U\(_j\) via an open channel.

Step LA5: upon receiving the message MES\(_3 = \{N_3, N_4, T_2, T_3\}\), U\(_j\) generates the current timestamp \(T^*_3\) and ensures that \([T^*_3 - T_3] \leq \Delta T\); if it is not, reject the session; otherwise, U\(_j\) computes

\[
S'_i = h(a'_i \| Sid \| T_3),
\]

\[
SK_n = h(m_i \cdot N_4 \| Sid \| Id'_i \| S'_i),
\]

\[
N'_4 = h(S'_i \| SK_n \| N_3 \| Id'_i \| T_3).
\]

If \(N'_4 \neq N_4\), terminate the session. Otherwise, the authentication is completed. Figure 1 demonstrates the steps of the mutual authentication and the key agreement phase.

5. Informal Security Analysis

In this section, we discuss the possible attacks on the proposed scheme.

5.1. Stolen and Hyphen: Verifier Attack. In our proposed scheme, GWN does not store information related to the verification table. Therefore, there is no stolen-verifier attack against our proposed scheme.

5.2. Offline Password Guessing Attack. In our proposed scheme, MES\(_1\), MES\(_2\), MES\(_3\), and MES\(_4\) are transmitted via an open channel; even if an attacker eavesdrops on the communication and obtains these messages, he/she cannot guess the password. Because the password and fingerprint are used in login verification and not transmitted openly. Though an attacker obtains the message \{MPW\(_i\), \(\tau_i\), \(F_i\), \(PK_G\), \(P\)\} stored in smart card, where MPW\(_i\) = \(h(ID_2 \| PW \| \sigma_i)\) and \(F_i = a_i \oplus h(ID_2 \| \sigma_i \| PW)\), he/she cannot verify whether the guessed password is correct without knowing the biometric key \(\sigma_i\).

5.3. Replay Attack. Suppose that an adversary \(U_A\) impersonates user \(U_i\) and intercepts and replays MES\(_i = \{M_1, M_2, M_3, T_1\}\). The replayed MES\(_i\) cannot pass the GWN’s verification process if the timestamp is invalid. Even if a replay of MES\(_1\) worked, and \(U_A\) gets MES\(_3\), however, the session key \(SK_u = h(m_i \cdot N_3 \| Sid \| Id'_i \| S'_i)\), where \(S'_i = h(a'_i \| Sid \| T_2)\) and \(m_i\) is a random number created by \(U_i\); \(U_A\) cannot obtain \(a'_i\) or \(m_i\). Therefore, it is useless to replay MES\(_1\).

Suppose that \(U_A\) replays GWN’s messages or sensor nodes’ messages. First, the replayed messages cannot pass the validity verification of the timestamp. In addition, \(U_i\), GWN, and SN\(_j\) generate new random numbers in a new session, which are used in the verification and generation of the session key. Therefore, our scheme is resistant to replay attacks.

5.4. Forger Attack and Impersonation Attack. Suppose an attacker impersonates the user \(U_i\) and sends MES\(_1 = \{M_1, M_2, M_3, T_1\}\) to GWN, where \(M_1 = h(a'_i \| M_2 \| M_3 \| T_1)\) and \(a'_i = F_i \oplus h(ID_2 \| \sigma_i \| PW)\); if the attacker does not have \(ID_2\), \(PW\), and \(fng_i\), he/she cannot forge \(M_1\). In other words, the attacker cannot impersonate a user.

If the attacker tries to impersonate GWN and forge MES\(_2 = \{M_1, N_1, N_2, T_2\}\), \(N_2 = h(M_1 \| Sid \| Id'_i \| S'_i)\), \(b'_i = h(Sid'_i \| K_{GW})\), the attacker does not know \(K_{GW}\), so the forged \(N_2\) cannot pass the verification of SN\(_j\).

If the attacker impersonates the sensor node, he/she cannot forge valid \(N_4 = h(S'_i \| SK_n \| N_3 \| Id'_i \| T_3)\) without knowing \(SID_j\) and \(b'_j\).

5.5. Smart Card Loss Attack. Suppose the smart card stolen by an attacker \(U_A; U_A\) can get \(\langle MPW_i, \tau_i, F_i, PK_G, P \rangle\), where \(MPW_i = h(ID, PW \| \sigma_i)\), \(\tau_i\) is the reproduction parameter of the fuzzy extractor algorithm, \(F_i = a_i \oplus h(ID \| \sigma_i \| PW)\), \(PK_G\) is the public key of GWN, and \(P\) is the base point of the elliptic curve. MPW\(_i\) and \(F_i\) are protected by the user’s biometric information and password. Therefore, an attacker
In the proposed scheme, each sensor node $SN_j$ stores $\{b_j, SID_j, P\}$, where $b_j = h(SID_j || K_{GWN})$. $SID_j$ is the identity of the sensor, and $P$ is the base point on the curve. An attacker cannot get $K_{GWN}$ even if he/she captures the sensor. In other words, capturing a sensor node cannot influence other sensor nodes.

Therefore, the proposed scheme resists sensor capture attacks.

5.7. Known-Key Attack. The session key $SK_j = h(c_j \cdot M_j || SID_j || ID_j^* || S_j^*) = SK_u = h(m_j \cdot N_j || SID_j || ID_j^* || S_j^*)$, where $c_j$ and $m_j$ are random numbers generated in every session, and the CDH problem is intractable. Therefore, even if an attacker gets session keys, he/she cannot solve the CDH problem.
5.8. Anonymity and Unlinkability. In the authentication phase of the proposed scheme, the user’s identity is contained in the message $MES_1 = \{M_1, M_2, M_3, T_1\}$, where $M_1 = m_1 \cdot P$, $M_2 = (ID_i \| \text{SID}_j \| T_1)$, and $M_3 = h(a_i^* \| M_1 \| M_2 \| T_1)$. The user’s identity $ID_i^*$ is protected by $h(m_i \cdot \text{PK}_G || T_1)$; only the gateway can obtain the user’s real identity. So, our scheme meets the requirement of anonymity. At the same time, because the random number
and the timestamp $T_1$ are contained in $M_2$, which is changed in each session, therefore, our scheme is also unlinkability.

5.9. Perfect Forward Secrecy. In the proposed scheme, the session key $SK = h(c_j \cdot m_j \cdot P)[SID, ID, S]$. Even if an adversary can know the user’s all secret information and the secret key of GWN, $c_jP$, and $m_jP$, but he/she still cannot compute $c_j \cdot m_j \cdot P$ because of the intractability of the computational Diffie–Hellman (CDH) problem. So, the proposed scheme can achieve perfect forward secrecy.

6. Formal Security Analysis Using ProVerif

ProVerif is a formalized cryptographic protocol verification tool based on the Dolev–Yao model, which can describe various cryptographic primitives. When using the ProVerif tool to validate a cryptographic protocol, the tool will present a corresponding sequence of attacks if the protocol is vulnerable.

As shown in Figure 2, we defined channels, basic types, and functions. The proposed scheme involves 5 events, namely, ULoginPhase(), UAuthenticationPhase(), UserSessionKey(), SNSessionKey(), and GWNAuthentication().
Figure 5: Process of GWN.

```
(*--process of GWN--*)
let GWNProcess(PKg:bitstring, P:bitstring, Kgwn:bitstring)=
in (chn1,(M1:bitstring,M2:bitstring,M3:bitstring,T1:bitstring));
if checktimestampfresh(T1, true) then
  let (IDi:bitstring, SIDj:bitstring)=Split(XOR(M2,Hash(Con(EccMul(M1,Kgwn),T1)))) in
  let nai=Hash(Con(IDi,Kgwn)) in
  let nM3=Hash(Con(nai,Con(M1,Con(M2,T1)))) in
  if nM3=M3 then
    event GWNAuthentication(server);
  new Time2: timestamp;
let T2=bit_timestamp(Time2) in
let Si=Hash(Con(nai,Con(SIDj,T2))) in
let bj=Hash(Con(SIDj,Kgwn)) in
let N1=XOR(Con(Si,IDi),Hash(Con(bj,Con(SIDj,T2)))) in
let N2=Hash(Con(M1,Con(IDi,Con(Si,Con(SIDj,T2)))))) in
out (chn2,(M1,N1,N2,T2)).
```

Figure 6: Process of sensor node.

```
(*--process of Sensor Node--*)
let SensorProcess(bj:bitstring, SIDj:bitstring, P:bitstring)=
in (chn2,(M1:bitstring,N1:bitstring,N2:bitstring,T2:bitstring));
if checktimestampfresh(T2, true) then
  let (Si:bitstring,IDi:bitstring)=Split(XOR(N1,Hash(Con(bj,Con(SIDj,T2)))))) in
  let nN2=Hash(Con(M1,Con(IDi,Con(Si,Con(SIDj,T2)))))) in
  if nN2=N2 then
    new cmj:nonce;
    new Time3:timestamp;
    let cj=bit_nonce(cmj) in
let T3=bit_timestamp(Time3) in
let N3=EccMul(cj,P) in
let SKj=Hash(Con(EccMul(cj,M1),Con(SIDj,Con(IDi,Si)))) in
let N4=Hash(Con(Si,Con(SKj,Con(N3,Con(IDi,T3)))))) in
event SNSessionKey(sensor);
out (chn1,(N3,N4,T2,T3)).
```

Figure 7: Main process.

```
(*--Main process--*)

process
  let ai=Hash(Con(IDi,Kgwn)) in
  let PKg=EccMul(Kgwn,P) in
  let bj=Hash(Con(SIDj,Kgwn)) in
  let (sigma:bitstring,taoi:bitstring)=Gen(fng) in
  let MPWi=Hash(Con(IDi,Con(PWi,sigma))) in
  let Fi=XOR(ai,Hash(Con(IDi,Con(sigma,MPWi)))) in
  ( (!UserProcess(IDi,MPWi,taoi,Fi,PKg,P,SIDj)) | (!GWNProcess(PKg,P,Kgwn)) | (!SensorProcess(bj,P,Kgwn)) )
```
ULoginPhase() indicates the login phase of the user, UAuthenticationPhase() indicates the user sends authentication request, GWNAuthentication() indicates the gateway pass the authentication of the user, SNSessionKey() indicates sensor node agrees on the session key, and UserSessionKey() indicates the user agrees the session key. Figure 3 shows the above events and queries.

Figure 7 exhibits the main process. According to the result in Figure 8, the proposed scheme can provide security of the session key, the password of the user, and the secret parameter of GWN. Meanwhile, the process of mutual authentication is executed in sequence.

7. Performance Comparison

In this section, we analyze the security and performance comparison between our schemes with some related schemes. Table 2 shows the comparison of attacks/properties of the schemes. Compared with Shuai et al.'s scheme, our scheme is more secure to various known attacks and has some good properties. As shown in Table 3, we can see the comparison of computational cost between the proposed scheme and the related schemes [19–21, 23, 24], where $T_H$ represents hash operation time, $T_{ECC}$ is the time of the symmetric encryption/decryption operation, and $T_{SE}$ denotes the time cost of ECC operation. In the environment [18] of Windows 10 64 bit laptop, Intel (R) Core (TM)
In this paper, we first pointed out Shuai et al.’s scheme is vulnerable to desynchronization attack, stolen-verifier attack, and no perfect forward security. In addition, we propose a new three-factor authentication using ECC and fuzzy extractor algorithm, which not only defends against the above attacks but also defends other attacks as shown in informal security analysis. We also simulate the proposed scheme for its formal security verification using the ProVerif tool to prove the security. Its performance analysis shows that it has less communication cost than the related schemes, and it can be applied to WSN in IoT. In the future, we will design block chain-based anonymous authentication scheme for WSN in IoT.

8. Conclusion

Data Availability

The data used to support the findings of the study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Key R&D Program of China (Grant no. 2017YFB0802000, URL: http://www.most.gov.cn/) and the National Natural Science Foundation of China (Grant nos. 61702152 and 61702153, URL: http://www.nsfc.gov.cn/).

References

[1] M. T. Lazarescu, “Design of a WSN platform for long-term environmental monitoring for IoT applications,” IEEE Journal on emerging and selected topics in circuits and systems, vol. 3, no. 1, pp. 45–54, 2013.

[2] 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,” Journal of Medical Systems, vol. 38, no. 9, 2014.

[3] F. Li and P. Xiong, “Practical secure communication for integrating wireless sensor networks into the internet of things,” IEEE Sensors Journal, vol. 13, no. 10, pp. 3677–3684, 2013.

[4] W. K. Seah, Z. A. Eu, and H. P. Tan, “Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges,” in Proceedings of the 2009 1st International Conference on Wireless Communication, Vehicular Technology, pp. 1–5, IEEE, Aalborg, Denmark, May 2009.

[5] M. Turkanović, B. Brumen, and M. Höflibl, “A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion,” Ad Hoc Networks, vol. 20, pp. 96–112, 2014.

[6] M. S. Farash, M. Turkanović, S. Kumari, and M. Höflibl, “An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment,” Ad Hoc Networks, vol. 36, pp. 152–176, 2016.

[7] R. Amin and G. P. Biswas, “A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks,” Ad Hoc Networks, vol. 36, pp. 58–80, 2016.

[8] Q. Xie, J. Zhao, and X. Yu, “Chaotic maps-based three-party password-authenticated key agreement scheme,” Nonlinear Dynamics, vol. 74, no. 4, pp. 1021–1027, 2013.

[9] S. A. Chaudhry, “Designing an efficient and secure message exchange protocol for internet of vehicles,” Security and Communication Networks, vol. 2021, Article ID 5554318, 9 pages, 2021.

[10] M. Rana, A. Shaﬁq, I. Altaf et al., “A secure and lightweight authentication scheme for next generation IoT infrastructure,” Computer Communications, vol. 165, pp. 85–96, 2021.

[11] M. L. Das, “Two-factor user authentication in wireless sensor networks,” IEEE Transactions on Wireless Communications, vol. 8, no. 3, 2009.

[12] M. K. Khan and K. Alghathbar, “Cryptanalysis and security improvements of two-factor user authentication in wireless sensor networks,” Sensors, vol. 10, no. 3, pp. 2450–2459, 2010.

[13] P. Kumar, S. G. Lee, and H. J. Lee, “E-SAP: efficient-strong authentication protocol for healthcare applications using wireless medical sensor networks,” Sensors, vol. 12, no. 2, pp. 1625–1647, 2012.

[14] D. He, N. Kumar, J. Chen, C. C. Lee, N. Chilamkurti, and S. S. Yeo, “Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks,” Sensors, vol. 12, no. 2, pp. 1625–1647, 2012.

[15] X. Li, J. Niu, S. Kumari, J. Liao, W. Liang, and M. K. Khan, “A new authentication protocol for healthcare applications using wireless medical sensor networks with user anonymity,” Security and Communication Networks, vol. 9, no. 15, pp. 2643–2655, 2016.

[16] F. Wu, L. Xu, S. Kumari, and X. Li, “An improved and anonymous two-factor authentication protocol for healthcare applications with wireless medical sensor networks,” Multimedia Systems, vol. 23, no. 2, pp. 195–205, 2017.

[17] O. Mir, J. Munilla, and S. Kumari, “Efficient anonymous authentication with key agreement protocol for wireless medical sensor networks,” Peer-to-Peer Networking and Applications, vol. 10, no. 1, pp. 79–91, 2017.

[18] Q. Xie, D. S. Wong, G. Wang, X. Tan, K. Chen, and L. Fang, “Provably secure dynamic ID-based anonymous two-factor authenticated key exchange protocol with extended security model,” IEEE Transactions on Information Forensics and Security, vol. 12, no. 6, pp. 1382–1392, 2017.

[19] H. L. Yeh, T. H. Chen, P. C. Liu, T. H. Kim, and H. W. Wei, “A secured authentication protocol for wireless sensor networks using elliptic curves cryptography,” Sensors, vol. 11, no. 5, pp. 4767–4779, 2011.

[20] C. C. Chang and L. D. Hai, “A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks,” IEEE Transactions on Wireless Communications, vol. 15, no. 3, pp. 357–366, 2015.
[21] S. Challa, M. Wazid, A. K. Das et al., “Secure signature-based authenticated key establishment scheme for future IoT applications,” IEEE Access, vol. 5, pp. 3028–3043, 2017.

[22] M. Tanveer, G. Abbas, Z. H. Abbas, M. Bilal, A. Mukherjee, and K. S. Kwak, “LAKE-6SH: lightweight user authenticated key exchange for 6LoWPAN-based smart homes,” IEEE Internet of Things Journal, vol. 14, no. 8, pp. 1–14, 2021.

[23] Q. Xie, K. Li, X. Tan, L. Han, W. Tang, and B. Hu, “A secure and privacy-preserving authentication protocol for wireless sensor networks in smart city,” EURASIP Journal on Wireless Communications and Networking, vol. 119, no. 1, pp. 1–17, 2021.

[24] M. Shuai, N. Yu, H. Wang, L. Xiong, and Y. Li, “A lightweight three-factor Anonymous authentication scheme with privacy protection for personalized healthcare applications,” Journal of Organizational and End User Computing, vol. 33, no. 3, pp. 1–18, 2021.

[25] B. Blanchet, “An efficient cryptographic protocol verifier based on prolog rules,” in Proceedings of the 14th IEEE Computer Security Foundations Workshop (CSFW-14), pp. 82–96, Cape Breton, NS, Canada, June 2014.

[26] D. Dolev and A. Yao, “On the security of public key protocols,” IEEE Transactions on Information Theory, vol. 29, no. 2, pp. 198–208, 1983.