Hindawi
Security and Communication Networks
Volume 2021, Article ID 3116593, 12 pages
https://doi.org/10.1155/2021/3116593

Review Article

Attacks and Solutions for a Two-Factor Authentication Protocol for Wireless Body Area Networks

Chien-Ming Chen,[1] **Zhen Li**,[1] **Shehzad Ashraf Chaudhry**,[2] **and Long Li**[3]

[1]*College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, China*
[2]*Department of Computer Engineering, Istanbul Gelisim University, Istanbul, Turkey*
[3]*Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Gullin, China*

Correspondence should be addressed to Long Li; lilong@guet.edu.cn

Received 1 September 2021; Accepted 4 October 2021; Published 21 October 2021

Academic Editor: Azees M

As an extension of the 4G system, 5G is a new generation of broadband mobile communication with high speed, low latency, and large connection characteristics. It solves the problem of human-to-thing and thing-to-thing communication to meet the needs of intelligent medical devices, automotive networking, smart homes, industrial control, environmental monitoring, and other IoT application needs. This has resulted in new research topics related to wireless body area networks. However, such networks are still subject to significant security and privacy threats. Recently, Fotouhi et al. proposed a lightweight and secure two-factor authentication protocol for wireless body area networks in medical IoT. However, in this study, we demonstrate that their proposed protocol is still vulnerable to sensor-capture attacks and the lack of authentication between users and mobile devices. In addition, we propose a new protocol to overcome the limitations mentioned above. A detailed comparison shows that our proposed protocol is better than the previous protocols in terms of security and performance.

1. Introduction

Since the beginning of human civilization, the efficient and fast transmission of information has always been an unswerving pursuit for mankind. From writing to printing, from cell towers to radio, from telephones to mobile Internet, the speed of modern technology development has always depended on the speed of information dissemination, and new ways of information dissemination often bring about radical changes in society. 5G (fifth-generation mobile communication technology) is the current stage of progress in the latest wave of mobile communication [1]. 5G is a new generation of broadband mobile communication with high speed, low latency, and large connection characteristics. It is a network infrastructure that enables the interconnection of people, machines, and things. 5G has three major application scenarios: enhanced mobile broadband, ultra-high reliability and low-latency communications, and massive machine-like communications. Enhanced mobile broadband mainly responds to the explosive growth of Internet traffic, and it results in improved user experience for mobile Internet users. Low-latency communication is mainly for applications with high requirements for latency and reliability, such as telemedicine, autonomous driving, and virtual reality. Massive machine-like communication is mainly for applications that involve the sensing and collection of data, such as Internet of Things (IoT) [2–4], smart cities [5–7], smart homes, and environmental monitoring [8–10].

In the long run, consumer demand for health will continue to rise, and the development potential of the medical and health fields is huge. Currently, 5G is particularly useful for the healthcare sector, especially for the Internet of Things in the medical field [11–13]. 5G will empower the existing smart healthcare service system, and it will improve the service capability and management efficiency of wireless body area networks, telemedicine, and emergency rescue. It will also give rise to the development and prosperity of smart healthcare.

Owing to rapid advancements in life informatization, people's requirements for medical monitoring are constantly improving. There is also a high demand for more convenient and effective telemedicine and health-sign monitoring. A wireless body area network (WBAN) [14, 15] is a network composed of different intelligent components, such as sensors, nodes, and actuators. The network is designed for collecting and monitoring data from the human body and its surrounding environment. Its typical architecture is shown in Figure 1. For the elderly, sensors/wearable devices on the elderly send the information collected to a gateway node. For the patient, the sensor acquires the patient's body monitoring data, connects it to a bedside monitor or other receiver, and transmits it wirelessly to a doctor for monitoring or diagnosis. The gateway acts as a local server which analyzes, stores, and manages the data sent by the sensor or monitor. Users, who can be doctors, nurses, or other medical professionals, can communicate with the gateway and access the data they want to know via mobile devices or computer-based devices on a LAN with the gateway. For example, a nurse can specifically track and check a patient's body data, so that if an abnormality is detected, the patient's condition can be checked and dealt with in a timely manner.

Because data transmission over a WBAN takes place over a public channel, attackers can access highly sensitive health information of patients. To ensure the security of a WBAN, a secure authentication and key agreement (AKA) protocol should be implemented before communication. Numerous AKA protocols have been proposed [16–21]. However, many of these AKA protocols have proven to be insecure against many types of attacks. Recently, Fotouhi et al. [22] proposed a lightweight and secure two-factor AKA protocol for WBANs in the healthcare-based IoT. They claimed that their proposed protocol is secure against many attacks, such as key disclosure simulation attacks, special session temporary information attacks, and offline password guess attacks.

In this study, we first demonstrate that Fotouhi et al.'s proposed protocol [22] is still vulnerable to sensor-capture attacks. Additionally, their proposed protocol fails to provide authentication between users and mobile devices. To overcome these security pitfalls, we propose a secure and efficient AKA protocol for WBANs. The security analysis shows that our proposed protocol is secure. We also provide a detailed comparison to demonstrate that our proposed protocol achieves improved efficiency and security.

The remainder of this paper is organized as follows. In Section 2, we briefly review the authentication protocol proposed by Fotouhi et al. In Section 3, we provide a reasonable cryptanalysis of Fotouhi et al.'s proposed protocol. In Section 4, we propose a new protocol for improving the flaws in the old protocol. In Section 5, we perform a security analysis, which includes both formal and informal analyses, to demonstrate the security and stability of our proposed protocol. In Section 6, we analyze the security and performance of our proposed protocol in terms of security, performance, and communication cost. Finally, we provide the conclusions to this study.

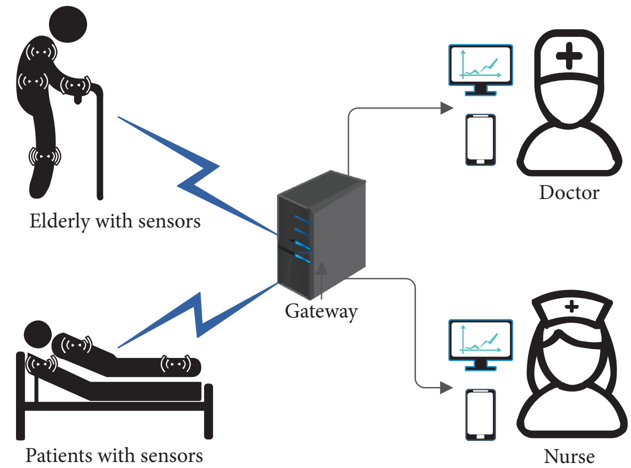

Figure 1: The typical architecture of a WBAN.

2. Review of Fotouhi et al.'s Protocol

In this section, we briefly review Fotouhi et al.'s authentication protocol. Their proposed protocol includes four phases: initialization, registration, authentication, and password modification. Here we describe only the first two phases. The detailed steps of their proposed protocol can be found in [22]. The notations used in this study are listed in Table 1.

2.1. Sensor Node Registration. In this phase, the corresponding gateway injects the necessary information into each sensor node. We assume that a gateway GW_j is the corresponding gateway of SN_k. GW_j generates two random numbers, R_y and R_z, after which it injects $\{SID_k, SG_k, QID_k, GID_j, R_y, R_z\}$ into the memory of SN_k, where $SG_k = h(SID_k \| G_j \| N_l)$. GW_j also stores $\{SID_k, N_l, QID_k, R_y, h(R_z)\}$ in its database.

2.2. User Registration. Assuming that a user, U_i, desires to register to GW_j, the following steps are performed:

Step 1: U_i sends ID_i and HPW_i to GW_j through a secure channel, where $HPW_i = h(PW_i \| R_0)$.

Step 2: if U_i is an unregistered user, GW_j generates a pseudoidentity CID_i and a random number R_x, and it stores $\{ID_i, HPW_i, CID_i, R_x\}$ in GW_j's database. GW_j then calculates $A_1 = h(CID_i \| R_x \| GID_j \| GID_j) \oplus HPW_i$ and $A_2 = h(ID_i \| G_j) \oplus h(ID_i \| HPW_i)$, after which it sends $\{CID_j, GID_j, A_1, A_2\}$ to U_i through a secure channel.

Step 3: U_i calculates $A_3 = h(ID_i \| PW_i) \oplus R_0$, after which it stores $\{CID_i, GID_j, A_1, A_2, A_3\}$ in the mobile device.

2.3. Authentication Phase. Assuming that U_i desires to communicate with SN_k, the following steps are performed:

Table 1: Notations table.

Symbol	Description
$U_i, \mathrm{ID}_i, \mathrm{PW}_i$	i-th user, his/her identity, his/her password
$\mathrm{GW}_j, \mathrm{GID}_j, G_j$	j-th gateway, its identity, its secret key
$\mathrm{SN}_k, \mathrm{SID}_k$	k-th sensor, its identity
N_l	Network identifier of the sensor set
SG_k	Shared key between sensor and gateway
SK_u	Session key generated by user
SK_g	Session key generated by gateway
SK_s	Session key generated by user
M_i	i-th message
$\mathrm{CID}_i, \mathrm{QID}_k$	Temporary pseudoidentity of U_i and SN_k
$R_s, R_0, R_u, R_g, R_x, R_y, R_z$	Temporary random number
$\mathrm{Gen}(\cdot)$, $\mathrm{Rep}(\cdot)$	Biometric extraction function, decryption function
BIO_i	Biometric information of the i-th user
$h(\cdot)$	Hash function
$\oplus$	Bitwise XOR operation
$\|$	Concatenate operation

Step 1: U_i generates a random number, R_u, after which it calculates $R_0 = A_3 \oplus h(\mathrm{ID}_i \| \mathrm{PW})$, $\mathrm{HPW}_i = h(\mathrm{PW}_i \| R_0)$, $B_1 = A_1 \oplus \mathrm{HPW}_i$, $B_2 = B_1 \oplus \mathrm{HPW}_i \oplus R_u$, $B_3 = \mathrm{SID}_k \oplus H(\mathrm{ID}_i \| R_u)$, and $B_4 = h(\mathrm{CID}_i \oplus \mathrm{GID}_j \oplus \mathrm{SID}_k \oplus B_1 \oplus \mathrm{ID}_i \oplus R_u)$. Afterwards, U_i transmits M_1 to GW_j, where $M_1 = \{\mathrm{CID}_i, \mathrm{GID}_j, B_2, B_3, B_4\}$.

Step 2: GW_j obtains the corresponding ID_i, R_x, and HPW_i from its database. GW_j then calculates $B_1 = h(\mathrm{CID}_i \| R_x \| \mathrm{GID}_j \| G_j)$ and $R_u = B_2 \oplus B_1 \oplus \mathrm{HPW}_i$, after which it verifies the correctness of B_4. GW_j then generates two random numbers, R_g and R_z', obtains SID_k with B_3, obtains R_y from its database, and generates a new pseudonym QID_k'. GW_j then calculates $\mathrm{SG}_k = h(\mathrm{SID}_k \| G_j \| N_l)$, $S = h(\mathrm{SG}_k \| \mathrm{GID}_j)$, $B_5 = (R_u \oplus \mathrm{HPW}_i) \oplus S \oplus R_y$, $B_6 = R_g \oplus S \oplus \mathrm{SID}_i \oplus R_y$, $B_7 = \mathrm{QID}_k' \oplus R_g \oplus R_y$, $B_8 = h(R_g \| R_y \| S) \oplus R_z'$, and $B_9 = h(\mathrm{QID}_k \| B_7 \| B_8 \| \mathrm{SG}_k \| R_u \oplus \mathrm{HPW}_i \| R_g)$. Afterwards, GW_j transmits $\{\mathrm{QID}_k, B_5, B_6, B_7, B_8, B_9\}$ to SN_k.

Step 3: SN_k verifies the correctness of QID_k. If it is correct, SN_k calculates $S = h(\mathrm{SG}_k \| \mathrm{GID}_j)$, $(R_u \oplus \mathrm{HPW}_i) B_5 \oplus S \oplus R_y$, and $R_g = B_6 \oplus S \oplus \mathrm{SID}_k \oplus R_y$. If B_9 is correct, SN_k generates a random number, R_s, and it calculates $R_z' = h(R_g \| R_y \| S) \oplus B_8$, $\mathrm{QID}_k' = B_7 \oplus R_g \oplus R_y$, and $B_{10} = R_g \oplus S \oplus R_z$. SN_k then stores QID_k', R_z', and $R_y' = h(R_y)$, and it calculates $\mathrm{SK}_s = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$. It then calculates $B_{11} = h(\mathrm{SG}_k \| R_g) \oplus h(R_y) \oplus R_s$ and $B_{12} = h(B_{10} \| B_{11} \| \mathrm{SK}_s \| \mathrm{SID}_k \| \mathrm{GID}_j \| R_s)$, after which it transmits $\{B_{10}, B_{11}, B_{12}\}$ to GW_j.

Step 4: GW_j calculates $R_y' = h(R_y)$ and $R_z' = R_g \oplus S \oplus B_{10}$. It then verifies whether $h(R_z)$ is equal to $h(R_z')$. If the verification is passed, it calculates $R_s = B_{11} \oplus h(\mathrm{SG}_k \| R_g) \oplus R_y'$ and obtains the session key $\mathrm{SK}_g = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$. It further verifies the correctness of B_{12}, generates a new CID_i' for U_i, stores QID_k' and R_z', and replaces R_y' and $h(R_x)$ with R_y and R_x, respectively. It then calculates $B_{13} = h(\mathrm{CID}_i' \| h(R_x) \| \mathrm{GID}_j \| G_j) \oplus h(R_u \| \mathrm{HPW}_i)$, $B_{14} = h(R_u \| \mathrm{ID}_i) \oplus R_g$, $B_{15} = h(R_u \| R_g \| \mathrm{HPW}_i) \oplus R_s$, $B_{16} = h(h(\mathrm{ID}_i \| G_j) \| R_s) \oplus \mathrm{CID}_i'$, and $B_{17} = h(\mathrm{SK}_g \| \mathrm{ID}_i \| B_{13} \| \mathrm{CID}_i')$. GW_j then generates $\{B_{13}, B_{14}, B_{15}, B_{16}, B_{17}\}$ and transmits it to U_i.

Step 5: U_i calculates $R_g = B_{14} \oplus h(R_u \| \mathrm{ID}_i)$, $R_s = B_{15} \oplus h(R_u \| R_g \| \mathrm{HPW}_i)$, and $\mathrm{CID}_i' = B_{16} \oplus h((A_2 \oplus h(ID_i \| HPW_i)) \| R_s)$. U_i then calculates the session key $\mathrm{SK}_u = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$ and verifies B_{17}. When the verification is passed, U_i calculates $A_1' = B_{13} \oplus h(R_u \| \mathrm{HPW}_i)$ and stores CID_i' and A_1'.

3. Cryptanalysis of Fotouhi et al.'s Protocol

This section shows that Fotouhi et al.'s protocol [22] is vulnerable to sensor-capture attacks and a lack of authentication between users and mobile devices.

3.1. Threat Model

3.1. Threat Model. The attacker model briefly describes the capabilities of an attacker. In this study, we use the $D-Y$ model [23–25] and assume that the attacker is A. The detailed capabilities are as follows:

(1) A can eavesdrop and intercept information transmitted by public channels and can forge, delete, replay, and tamper with such information

(2) A can extract the information from the captured sensor nodes

(3) A can access the information stored in the gateway

3.2. Sensor-Capture Attack. Assuming that A captures SN_k and obtains $\{\mathrm{SID}_k, \mathrm{SG}_k, \mathrm{GID}_j, R_y, R_z, \mathrm{QID}_k\}$ in the memory of sensor SN_k, A can calculate the session key SK through the following steps:

Step 1: calculate $S = h(\mathrm{SG}_k \| \mathrm{GID}_j)$, and then obtain $(R_u \oplus \mathrm{HPW}_i)$ by calculating $B_5 \oplus S \oplus R_y$

Step 2: obtain R_g by calculating $B_6 \oplus S \oplus \mathrm{SID}_k \oplus R_y$

Step 3: obtain R_s by calculating $h(\mathrm{SG}_k \| R_g) \oplus h(R_y) \oplus B_{11}$

Therefore, A can calculate the correct session key $\mathrm{SK} = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$ shared among U_i, GW_j, and SN_k.

3.3. Lack of Authentication between Users and Mobile Devices. Assuming that an attacker A captures U_i's mobile device, A performs the following steps:

Step 1: because A does not know PW_i, A randomly generates PW_i' and then inputs ID_i and PW_i' to the captured mobile device. The mobile device calculates and transmits M_1 with the fake password PW_i' to GW_j.

Step 2: GW_j verifies GID_j and CID_i, after which it calculates B_1 and R_u. Afterwards, GW_j attempts to verify the correctness of B_4, and GW_j realizes that M_1 sent from U_i is not legal.

Essentially, A does not need to capture a mobile device because the attacker can eavesdrop the M_1 between any user and GW_j and then send M_1 to GW_j.

The scenario mentioned above illustrates two weaknesses in Fotouhi et al.'s proposed protocol. First, the mobile device does not verify the password that a user inputs. Regardless of whether the password or account number entered by U_i is correct, the mobile device sends all the necessary messages to GW_j. Second, GW_j calculates B_1 and R_u before verifying B_4. Owing to the limited computing power of a gateway, if an attacker has been sending a large number of error messages to a gateway through multiple mobile devices, the gateway may be paralyzed and unable to respond to the requests of other users, which will result in immeasurable losses in medical Internet environments.

4. The Improved Protocol

In this section, we present an enhanced lightweight and secure two-factor authentication protocol (AELSA) for medical IoT and WBANs to address and enhance the outstanding vulnerabilities and fragile shortcomings of Fotouhi et al.'s protocol. AELSA also applies to the WBAN architecture and includes three main participants: (a) the physician or nurse as the user, (b) the gateway node as the server, and (c) as the sensor. The sensors can include the dynamic collection of patient data for real-time data. On the other hand, the gateway represents a server, which acts as an authentication and data-delivery center for ensuring mutual authentication between the physician and the sensor. The physician or nurse, as the user, can access the information from the sensor, which is delivered using the gateway through a device, such as a mobile device or a computer that can log into the system. AELSA comprises four main phases: (a) initialization, (b) registration, (c) login, and (d) mutual authentication and key exchange phases. The registration phase includes the user registration and sensor registration phases. The symbols used are also listed in Table 1.

4.1. Initialization Phase. We assume that all the gateways are considered trusted parts, the gateways are identified through GID_j when transmitting messages, and the gateways generate G_j as their private key during initialization. In this phase, important parameters and functions of the system are generated and published, such as initializing the stored information within the gateway.

4.2. Registration Phase. This phase comprises a sensor node enrollment phase and a user enrollment phase with the following steps.

4.2.1. Sensor Node Enrollment. In the sensor registration phase of AELSA, if a new sensor SN_k wants to join the WBAN, it must interact with the data and submit registration information to the gateway GW_j. First, SN_k sends its SID_k and N_l to GW_j over a secure channel. After GW_j receives the message, it determines whether SID_k is a new identity and generates a new pseudoidentity QID_k for SN_k if it is a new identity. Next, it computes SG_k as a shared key for SN_k and GW_j, where $\mathrm{SG}_k = h(\mathrm{SID}_k \| G_j \oplus N_l)$, and it stores $\{\mathrm{QID}_k, N_l\}$ into the memory. Afterwards, GW_j securely sends $\{\mathrm{SG}_k, \mathrm{QID}_k\}$ to SN_k. Once SN_k receives the message, it encrypts SG_k using its SID_k, $\mathrm{RSG}_k = \mathrm{SG}_k \oplus \mathrm{SID}_k$, and it stores $\{\mathrm{RSG}_k, \mathrm{QID}_k\}$.

4.2.2. User Enrollment. In this stage, the user completes the registration in GW_j based on the generation function of the bioinformation embedded in the mobile device as well as other information. The user enters their identity ID_i, password PW_i, and bioinformation BIO_i on the mobile device. The mobile device then generates σ_i and τ_i using the generation function Gen. It uses σ_i to mask and protect PW_i, calculates $\mathrm{HPW}_i = h(\mathrm{PW}_i \| \sigma_i)$, and sends $\{\mathrm{ID}_i, \mathrm{HPW}_i\}$ to GW_j on the anti-interference channel. Upon receiving $\{\mathrm{ID}_i, \mathrm{GW}_j\}$ determines whether the identity is new. A new identity represents an unregistered identity. If it is new, it then calculates $\mathrm{CID}_i = h(\mathrm{ID}_i)$ and stores $\mathrm{CID}_i, \mathrm{HPW}_i$. It then selects a secret random number R_0 and computes $A_1 = h(\mathrm{CID}_i \| \mathrm{GID}_j \| R_0 \oplus G_j) \oplus \mathrm{Hpw}_i$ and $A_2 = h(\mathrm{GID}_j \| \mathrm{HPW}_i) \oplus (R_0 \oplus G_j)$, which, in turn, store A_1 into memory. It then transmits the secure message $\{A_2, \mathrm{GID}_j\}$ to U_i over the private channel. After U_i receives the secure message, it computes $A_3 = h(ID_i \| \mathrm{HPW}_i)$ and stores $\{A_2, A_3, \mathrm{GID}_j, \mathrm{Gen}(.), \mathrm{Rep}(.), an\ d\ \tau_i\}$, where Rep can decrypt σ_i using the biological information BIO_i and τ_i.

4.3. Login Phase. Compared to the protocol proposed by Fotouhi et al., AELSA adds a login phase in which the mobile device verifies the legitimacy of U_i's identity and effectively prevents the consumption of redundant functions resulting from the nonuse of authentication. It is assumed that when U_i logs into the mobile device, U_i enters ID_i^* and PW_i^* and enters biological information BIO_i^*, such as the fingerprint and iris. The mobile device calculates $\mathrm{Rep}(\mathrm{BIO}_i^*, \tau_i)\sigma_i^*$, $\mathrm{HPW}_i^* = h(\mathrm{PW}_i^* \| \sigma_i^*)$, and $A_3^* = h(\mathrm{ID}_i^* \| \mathrm{HPW}_i)$. It then verifies A_3 by comparison. If $A_3 = A_3^*$, the mobile device allows U_i to log in. Otherwise, it denies U_i to log into the system and sends an alert. Figure 2 shows the detailed process of the user login phase.

4.4. Mutual Authentication and Key Exchange Phase. In the key exchange phase, the user, gateway, and sensor negotiate to create a three-way trusted key for ensuring the correctness and security of future messages. This phase comprises five steps, as described below. Among other things, Figure 3 shows the stages of mutual authentication and key exchange.

Step 1: user U_i selects the SID_k of the sensor to be accessed, generates a random number R_u, and creates a timestamp T_1. U_i computes $(R_0 \oplus G_j) = A_2 \oplus h(\mathrm{GID}_j \| \mathrm{HPW}_i)$, $B_1 = \mathrm{SID}_k \oplus h(\mathrm{GID}_j \| \mathrm{HPW}_i)$, $B_2 = R_u \oplus h(\mathrm{GID}_j \| \mathrm{HPW}_i \oplus SID_k)$, and $B_3 = (R_0 \oplus G_j) h(\mathrm{GID}_j \| R_u)$, after which U_i transmits the message M_1 $\{\mathrm{CID}_i, \mathrm{GID}_j, B_1, B_2, B_3, T_1\}$ to the gateway GW_j.

Step 2: after receiving the message M_1, GW_j verifies the legitimacy of T_1 by determining whether it matches $|T_1 - T_C|\Delta T$. GW_j searches and obtains the corresponding HPW_i and QID_k in the memory based on CID_i in M_1. Afterwards, GW_j computes $\mathrm{SID}_k = B_1 \oplus h(\mathrm{GID}_j \| \mathrm{HPW}_i)$, $R_u = B_2 \oplus h(\mathrm{GID}_j \| \mathrm{HPW}_i \oplus \mathrm{SID}_k)$, $(R_0 \oplus G_j) = B_3 \oplus h(\mathrm{GID}_j \| R_u)$, and $A_1^* = h(\mathrm{CID}_i \| \mathrm{GID}_j \| R_0 \oplus G_j) \oplus \mathrm{HPW}_i$, and it verifies $A_1 \stackrel{?}{=} A_1^*$. If the verification fails, GW_j aborts the conversation. Otherwise, GW_j confirms the legitimacy of the identity of U_i, after which it generates a random number R_g and a new timestamp T_2, and it computes $\mathrm{SG}_k = h(\mathrm{SID}_k \| G_j \oplus N_l)$, $B_4 = R_u \oplus \mathrm{HPW}_i \oplus \mathrm{SG}_k$, $B_5 = R_g \oplus h(\mathrm{SG}_k \| \mathrm{SID}_k)$, and $B_6 = h(\mathrm{QID}_k \| B_4 \| B_5 \| SG_k \| R_u \oplus \mathrm{HPW}_i \| R_g)$. Finally, GW_j sends $M_2\{\mathrm{QID}_k, B_4, B_5, B_6, T_2\}$ to the sensor node SN_k.

Step 3: once M_2 is received, SN_k verifies that $|T_2 - T_C| \leqq \Delta T$, and if this is true, then the message M_2 is fresh. Afterwards, SN_k obtains the corresponding RSG_k in storage based on QID_k. It computes $\mathrm{SG}_k = \mathrm{RSG}_k \oplus \mathrm{SID}_k$, $(R_u \oplus \mathrm{HPW}_i) = B_4 \oplus \mathrm{SG}_k$, $R_g = B_5 \oplus h(\mathrm{SG}_k \| \mathrm{SI}D_k)$, and $B_6^* = h(\mathrm{QID}_k \| B_4 \| B_5 \| \mathrm{SG}_k \| R_u \oplus \mathrm{HPW}_i \| R_g)$, and it verifies whether $B_6^* \stackrel{?}{=} B_6$. If the verification is successful, SN_k creates a random number R_s and a timestamp T_3, after which it computes the keys $\mathrm{SK}_s = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$, $B_7 = h(\mathrm{SG}_k \| R_g) \oplus R_s$, and $B_8 = h(R_g \| R_s \| \mathrm{SG}_k \| T_3)$. SN_k then sends $M_3\{B_7, B_8, T_3\}$ to GW_j over the public channel.

Step 4: after receiving message M_3, GW_j verifies the freshness of timestamp T_3 using $|T_3 - T_C| \leqq \Delta T$. After verifying that it passes, GW_j generates timestamp T_4 and computes $R_s = h(\mathrm{SG}_k \| R_g) \oplus B_7$ and $B_8^* = h(R_g \| R_s \| \mathrm{SG}_k \| T_3)$, after which it verifies the legitimacy of B_8. If B_8 qualifies, the key $\mathrm{SK}_g = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$, $B_9 = h(R_u \oplus \mathrm{GI}D_j \| \mathrm{HPW}_i) \oplus (R_g \| R_s)$, and $B_{10} = h(R_0 \oplus G_j \| \mathrm{SK}_g \| R_u)$. Finally, GW_j generates M_4 $\{B_9, B_{10}, T_4\}$ and passes M_4 back to U_i.

Step 5: in the final step, after receiving the message M_4, U_i verifies whether $|T_4 - T_C| \leqq \Delta T$, and if this is correct, it computes $(R_g \| R_s) = B_9 \oplus h(R_u \oplus \mathrm{GID}_j \| \mathrm{HPW}_i)$, $\mathrm{SK}_u = h(R_u \oplus \mathrm{HPW}_i \| R_g \| R_s)$, and $B_{10}^* = h(R_0 \oplus G_j \| \mathrm{SK}_u \| R_u)$. Finally, U_i verifies whether $B_{10}^* \stackrel{?}{=} B_{10}$, and if this is true, the verification and key exchange phase is complete.

5. Security Analysis

In this section, we use the random oracle model (ROR) to conduct a rigorous formal security analysis of the improved protocol. In addition, an informal security analysis is carried out to logically analyze the protocol. Through the following security analysis, it is easy to prove the security and robustness of the improved protocol.

5.1. Formal Security Analysis. In this section, the ROR model is mainly used to prove the security and feasibility of our proposed protocol, and we successfully demonstrated that users and sensor nodes can securely establish session keys through the gateway. In the proof process, U represents a user, G represents a gateway, and S represents a sensor node. The detailed proof of the procedure is presented as follows.

5.1.1. ROR Model. In this section, we will use the ROR model to prove the security and reliability of our proposed new scheme, where $\mathscr{A}$ represents the attacker. There are three participants which are user U, gateway G, and sensor S. Suppose Π_U^x represents the x-th communication of the user, Π_{U*}^i represents the i-th instance of the user, Π_G^j represents the j-th instance of the gateway, and Π_S^k represents the k-th instance of the sensor. The attacker has special capabilities and can initiate the following queries:

Execute $(\Pi_{U*}^x, \Pi_G^j, \Pi_S^k)$: by executing this query, $\mathscr{A}$ can intercept and obtain the messages transmitted between the various participant instances on the public channel. Passive attacks can be executed by this query

Send (Π_U^x, M): in this query, $\mathscr{A}$ can get the corresponding response by sending message M to Π_U^x. $\mathscr{A}$ can perform man-in-the-middle attacks and impersonation attacks.

U_i	$Mobile\ device$
U_i enters ID_i^* and PW_i^* imprints BIO_i'	
$\xrightarrow{Input\ to\ mobile\ device}$	
	$Rep(BIO_i^*, \tau_i) = \sigma_i^*$ $HPW_i^* = h(PW_i^* \parallel \sigma_i^*)$ $A_3^* = h(ID_i^* \parallel HPW_i)$ Verify $A_3 \stackrel{?}{=} A_3^*$ If true, user authentication passed

Figure 2: Login phase.

U_i	GW_j	SN_k
Selects SID_k, R_u, T_1 Computes $(R_0 \oplus G_j) = A_2 \oplus h(GID_j \parallel HPW_i)$ $B_1 = SID_k \oplus h(GID_j \parallel HPW_i)$ $B_2 = R_u \oplus h(GID_j \parallel HPW_i \oplus SID_k)$ $B_3 = (R_0 \oplus G_j) \oplus h(GID_j \parallel R_u)$ $\xrightarrow{M_1=\{CID_i, GID_j, B_1, B_2, B_3, T_1\}}$		
	Verify $\lvert T_1 - T_C \rvert \leqq \Delta T$ Gets HPW_i, QID_k Computes $SID_k = B_1 \oplus h(GID_j \parallel HPW_i)$ $R_u = B_2 \oplus h(GID_j \parallel HPW_i \oplus SID_k)$ $(R_0 \oplus G_j) = B_3 \oplus h(GID_j \parallel R_u)$ $A_1^* = h(CID_i \parallel GID_j \parallel R_0 \oplus G_j) \oplus HPW_i$ Check $A_1 \stackrel{?}{=} A_1^*$ Selects R_g, T_2 $SG_k = h(SID_k \parallel G_j \oplus N_l)$ $B_4 = R_u \oplus HPW_i \oplus SG_k$ $B_5 = R_g \oplus h(SG_k \parallel SID_k)$ $B_6 = h(QID_k \parallel B_4 \parallel B_5 \parallel SG_k \parallel R_u \oplus HPW_i \parallel R_g)$ $\xrightarrow{M_2=\{QID_k, B_4, B_5, B_6, T_2\}}$	
		Verify $\lvert T_2 - T_C \rvert \leqq \Delta T$ Gets RSG_k based on QID_k $SG_k = RSG_k \oplus SID_k$ $(R_u \oplus HPW_i) = B_4 \oplus SG_k$ $R_g = B_5 \oplus h(SG_k \parallel SID_k)$ $B_6^* = h(QID_k \parallel B_4 \parallel B_5 \parallel SG_k \parallel R_u \oplus HPW_i \parallel R_g)$ Verify $B_6^* \stackrel{?}{=} B_6$ Selects R_s,T_3 Computes $SK_s = h(R_u \oplus HPW_i \parallel R_g \parallel R_s)$ $B_7 = h(SG_k \parallel R_g) \oplus R_s$ $B_8 = h(R_g \parallel R_s \parallel SG_k \parallel T_3)$ $\xleftarrow{M_3=\{B_7, B_8, T_3\}}$
	Verify $\lvert T_3 - T_C \rvert \leqq \Delta T$ Computes $R_s = h(SG_k \parallel R_g) \oplus B_7$ $B_8^* = h(R_g \parallel R_s \parallel SG_k \parallel T_3)$ Check $B_8^* \stackrel{?}{=} B_8$ Selects T_4 $SK_g = h(R_u \oplus HPW_i \parallel R_g \parallel R_s)$ $B_9 = h(R_u \oplus GID_j \parallel HPW_i) \oplus (R_g \parallel R_s)$ $B_{10} = h(R_0 \oplus G_j \parallel SK_g \parallel R_u)$ $\xleftarrow{M_4=\{B_9, B_{10}, T_4\}}$	
$\lvert T_4 - T_C \rvert \leqq DeltaT$ Compute $(R_g \parallel R_s) = B_9 \oplus h(R_u \oplus GID_j \parallel HPW_i)$ $SK_u = h(R_u \oplus HPW_i \parallel R_g \parallel R_s)$ $B_{10}^* = h(R_0 \oplus G_j \parallel SK_u \parallel R_u)$ Checks $B_{10}^* \stackrel{?}{=} B_{10}$ If ture, communication is possible		

Figure 3: Mutual authentication and key agreement phase.

Hash (Π_U^x, string): in this query, the hash value of the input string can be obtained by $\mathscr{A}$.

Corrupt (Π_U^x): through this query, $\mathscr{A}$ can send this query to the instance Π_U^x and Π_U^x returns the secret value of U: long-term private key, password, and secret parameters stored in the smart card (based on the smart card). $\mathscr{A}$ can simulate the execution of forward secrecy, privilege insider (internal) attacks, and stolen smart card attacks.

Reveal (Π_U^x): $\mathscr{A}$ can send this query to the instance Π_U^x and Π_U^x returns the current session key SK generated by its partner to $\mathscr{A}$. $\mathscr{A}$ can simulate the execution of known session key attacks.

Test (Π_U^x): $\mathscr{A}$ can perform this query by flipping a coin C. If C results in 1, the attacker will get the correct session key; otherwise, the attacker will receive a random string.

Theorem 1. *In the above* ROR *model, we redefine the $\mathscr{A}$'s capabilities and allow the attacker to execute the above query, so the probability P of our proposed new protocol being broken is expressed as* $\mathrm{Adv}_{\mathscr{A}}^{\mathrm{v}}(\xi) \le q_{\mathrm{send}}/2^{l-2} + 3q_{\mathrm{hash}}^2/2^{l-1} + 2\max\{C', q_{\mathrm{send}}^{s'}, q_{\mathrm{send}}/2^l\}$, *where q_{hash} represents the number of* hash *queries performed and q_{send} represents the* *number of queries performed. The number of bits of biological information is expressed by l, C' and s' are* Zipf$'s$ *law [26].*

Proof. We define GM_0 to GM_5 to mimic and verify the behavior that may be performed by $\mathscr{A}$. $\mathrm{Succ}_{\mathscr{A}}^{GM_i}(\xi)$ is used to denote the probability of success of $\mathscr{A}$'s attack on the protocol in GM_i. The specific process is as follows:

GM_0: in GM_0, $\mathscr{A}$ does not initiate any queries. Therefore, in GM_0, the probability P that the protocol is broken in this query round is

$$\mathrm{Adv}_{\mathscr{A}}^{\mathrm{v}}(\xi) = \left|2\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_0}(\xi)\right] - 1\right|. \quad (1)$$

GM_1: GM_1 adds Execute query, and the others have no difference with GM_0. We can obtain

$$\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_1}(\xi)\right] = \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_0}(\xi)\right]. \quad (2)$$

GM_2: GM_2 adds Send query, and there is no difference with GM_1. Therefore, we can get

$$\left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_2}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_1}(\xi)\right]\right| \leq \frac{q_{\mathrm{send}}}{2^l}. \quad (3)$$

GM_3: GM_3 and GM_2 are indistinguishable except that it adds the Hash query and deletes the Send query. We can obtain

$$\left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_3}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_2}(\xi)\right]\right| \leq \frac{q_{\mathrm{hash}}^2}{2^{l+1}}. \quad (4)$$

GM_4: in GM_4, whether a session key is secure or not can be seen in the following two cases. The first case is whether the protocol can ensure perfect forward secrecy security when $\mathscr{A}$ obtains the long-term private key. The second is whether the protocol can resist the temporary information leakage attack when the temporary information is compromised.

(1) Perfect forward secrecy: using Π_G^j, $\mathscr{A}$ tries to obtain the long-term key SG_k between the gateway and the sensor, or $\mathscr{A}$ uses Π_{U*}^x or Π_S^k to try to get a certain secret value in the registration phase
(2) Known session-specific temporary information attacks: $\mathscr{A}$ uses one of Π_G^j or Π_{U*}^i or Π_S^k to try to obtain temporary information from one entity

In both cases, $\mathscr{A}$ only needs to use Send and Hash queries to compute $SK_u = h(R_u \oplus HPW_i \| R_g \| R_s)$. For the first case, assuming that $\mathscr{A}$ obtains the long-term key SG_k, although $R_u \oplus HPW_i$ can be computed by intercepting B_4, $\mathscr{A}$ has no access to SID_k and thus cannot compute R_g and R_s and thus even less likely to compute SK. For the second case, assuming that $\mathscr{A}$ obtains the temporary information R_u, $\mathscr{A}$ has no access to the other random numbers R_g and R_s and thus cannot crack this protocol. Therefore, we get

$$\left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_4}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_3}(\xi)\right]\right| \leq \frac{q_{\mathrm{send}}}{2^l} + \frac{q_{\mathrm{hash}}^2}{2^{l+1}}. \quad (5)$$

GM_5: in GM_5, $\mathscr{A}$ can execute smart card stolen attacks. $\mathscr{A}$ uses $\mathrm{Corrupt}(\Pi_U^x)$ to get the information stored in $SC\{A_2, A_3, GID_j, \mathrm{Gen}(.), \mathrm{Rep}(.), \tau_i\}$. The mobile user uses password PW_i and biological information BIO_i to register. If $\mathscr{A}$ tries to guess $A_3^* = h(ID_i^* \| HPW_i)$, since HPWi is encrypted with biological information, the probability of $\mathscr{A}$ guessing the biometric σ_i is $1/2^l$ [27]. $\mathscr{A}$ can also guess low-entropy passwords; using Zipf$'s$ law [26], we can get

$$\left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_5}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_4}(\xi)\right]\right| \leq \max\left\{C', q_{\mathrm{send}}^{s'}, \frac{q_{\mathrm{send}}}{2^l}\right\}. \quad (6)$$

GM_6: GM_6 is used to verify whether the proposed protocol is resistant to impersonation attacks. In GM_6, if $\mathscr{A}$ issues a $h(R_u \oplus HPW_i \| R_g \| R_s)$ query, the game is terminated. So we can obtain

$$\left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_6}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_5}(\xi)\right]\right| \leq \frac{q_{\mathrm{hash}}^2}{2^{l+1}}. \quad (7)$$

Since GM_6 has half the probability of success and failure,

$$\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_6}(\xi)\right] = \frac{1}{2}. \quad (8)$$

To sum up, we can obtain the following conclusions:

$$\begin{aligned}\frac{1}{2}\mathrm{Adv}_{\mathscr{A}}^{\mathrm{V}}(\xi) &= \left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_0}(\xi)\right] - \frac{1}{2}\right| \\ &= \left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_0}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_6}(\xi)\right]\right| \\ &= \left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_1}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_6}(\xi)\right]\right| \\ &\leq \sum_{i=0}^{5}\left|\Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_{i+1}}(\xi)\right] - \Pr\left[\mathrm{Succ}_{\mathscr{A}}^{GM_i}(\xi)\right]\right| \\ &= \frac{q_{\mathrm{send}}}{2^{l-1}} + \frac{3q_{\mathrm{hash}}^2}{2^{l-1}} + \max\left\{C', q_{\mathrm{send}}^{s'}, \frac{q_{\mathrm{send}}}{2^l}\right\}.\end{aligned} \quad (9)$$

Finally, we can get

$$\mathrm{Adv}_{\mathscr{A}}^{\mathrm{v}}(\xi) \leq \ = \frac{q_{\mathrm{send}}}{2^{l-1}} + \frac{3q_{\mathrm{hash}}^2}{2^{l-1}} + 2\max\left\{C', q_{\mathrm{send}}^{s'}, \frac{q_{\mathrm{send}}}{2^l}\right\}. \quad (10)$$

Therefore, we can use the ROR model to demonstrate that our proposed new protocol can provide perfect forward security against common attacks such as smart card theft attacks, man-in-the-middle attacks, and other more common attacks. □

5.2. Informal Security Analysis. In this section, we prove that our proposed protocol is secure against common attacks. The security of our proposed protocol and the reasons it can withstand attacks are analyzed.

5.2.1. Resisting Sensor Node Capture Attacks. If an attacker captures a sensor node and obtains its memory information, although the attacker already knows the parameters RSG_k and QID_k, to obtain SK, the attacker must also know SID_k and the long-term key SG_k between the gateway and the sensor node, which is obtained from RSG_k and SID_k through heterodyning. However, SID_k is not stored in the memory of the sensor node. Therefore, our proposed protocol is improved to effectively prevent sensor node capture attacks.

5.2.2. Ensuring Authentication between Users and Mobile Devices. An attacker can replay eavesdropped messages and obtain valuable information through replay and feedback. For example, an attacker can replay message M_1 by imitating the user. However, our improved protocol does not provide this opportunity to the attacker. This is because we add a timestamp T to verify the freshness of the message, and we set a reasonable timestamp threshold. Moreover, we add biometric authentication to ensure accurate authentication between users and mobile devices, thereby preventing attackers from attacking the gateway using large amounts of useless information resulting from the lack of authentication between users and devices.

5.2.3. Perfect Forward Secrecy. If an attacker cannot obtain the previous session key when the private long-term key is destroyed, the authentication protocol has perfect forward confidentiality [28, 29]. Assuming that an attacker has obtained the long-term key SG_k between the gateway and the sensor, although it can be obtained through the message B_4 of the common channel ($R_u \oplus \mathrm{HPW}_i$), R_g and R_s are protected by the long-term key SG_k in addition to SID_k. Therefore, an attacker cannot obtain SID_k while obtaining the long-term key. As such, it can be inferred that the attacker cannot crack the long-term key in the case of obtaining the past session key. Thus, our proposed protocol demonstrates perfect forward security.

5.2.4. Resisting Session-Specific Temporary Information Attacks. If short-term secret information, such as random numbers, is cracked and obtained by an attacker, the attacker cannot calculate the key SK. Because the improved protocol uses a three-way random number and the encrypted value of the user's password information composition, an attacker cannot obtain the user's password information through the knowledge of the random number. Therefore, our proposed protocol can resist temporary information leakage attacks.

5.2.5. Resisting Offline Password-Guessing Attacks. In the authentication stage, we use the pseudo-password HPW_i as a substitute for the user password to ensure the security and privacy of the password. Because the user password is obtained through the user's biological information and password encryption, assuming that the attacker obtains HPW_i, the user password cannot be calculated. In the login phase, assuming that the attacker obtains A_3 and ID_i, the attacker cannot calculate PW_i from these data. Therefore, our proposed protocol can resist offline password-guessing attacks.

5.2.6. Resisting Privileged Insider Attacks. Assuming that an attacker is an insider of the gateway and has access to the gateway's memory information [30], the attacker can obtain CID_i, HPW_i, and QID_i. After obtaining this internal information, the attacker cannot compute any valuable information, and thus, the exact protocol is completely resistant to privileged insider attacks.

5.2.7. Resisting Relay Attacks. In the general three-party authentication protocol, the general steps involve authenticating communications between the user and the server. The server then communicates with the sensor or other devices for authentication, after which the sensor and other devices pass the information to the user through the server, and the information finally reaches the user, server, sensors, and other devices involved in the three-party authentication process. However, the transmission process is prone to relay attacks [30, 31], where information can easily be intercepted by the attacker using disguised devices to obtain the correct information sent by the official server or the user, so that they can disguise themselves as legitimate servers and send instructions to the user or disguise themselves as legitimate users to obtain valuable information. However, in our proposed protocol, the server GW_j properly verifies the legitimacy of user U_i and sensor SN_k by comparing A_1 and B_8. Additionally, the sensors and users verify the legitimacy of the server, and they employ a timestamp to verify the freshness of the message. Thus, our proposed protocol is resistant to relay attacks.

5.2.8. Resisting Stolen-Verifier Attacks. In a stolen authentication attack, we assume that the user authentication value stored on the server side is stolen by an attacker, and the attacker can directly use the authentication value to disguise themselves as a user and log into the system. Further, we assume that the secret information stored on the server side is also stolen, and the attacker can use this information to obtain the public key. Assuming that an attacker obtains the stored information inside the gateway GW_j, which is $\{\mathrm{CID}_i, \mathrm{HPW}_i, A_1, \mathrm{QID}_k, N_l\}$, the key to determining SK involves obtaining SG_k and obtaining Ru using SG_k. However, SG_k cannot be obtained using the information in the memory of GW_j. Therefore, our proposed protocol can resist stolen authentication attacks.

Table 2: Comparisons of security.

Security properties	Fotouhi et al. [22]	Kumari et al. [32]	Srinivas et al. [33]	Gope and Hwang [34]	Ours
Perfect forward secrecy	✓	✓	✓	✓	✓
Resists impersonation attacks	✓	✓	✓	✓	✓
Resists offline password-guessing attacks	✓	×	×	×	✓
User anonymity security	✓	✓	✓	✓	✓
Mutual authentication	✓	✓	✓	✓	✓
Resists replay attacks	×	✓	✓	✓	✓
Resists sensor-capture attacks	×	×	✓	✓	✓
Resists known session temporary information attacks	✓	×	✓	✓	✓
Resists relay attacks	✓	✓	✓	✓	✓
Resists man-in-the-middle attacks	✓	×	✓	✓	✓
Provable security	×	×	×	×	✓

Table 3: The computational cost of complex operations.

Operations	Host node(s)
Hash function	0.00032
Fuzzy function	0.0171
Chaotic map function	0.0171
Encryption and decryption	0.0056

Table 4: Calculation cost comparison.

Protocol	User	Gateway	Sensor	Total (ms)
Fotouhi et al.'s [22]	$10T_h$	$17T_h$	$7T_h$	$37T_h = 10.88$
Kumari et al.'s [32]	$8T_h + 2T_s$	$4T_h + T_s$	$4T_h + 2T_s$	$16T_h + 5T_s = 33.12$
Srinivas et al.'s [33]	$4T_h + 2T_c + 2T_s$	$6T_h + 2T_s$	$3T_h + 2T_c$	$4T_c + 4T_s + 13T_h = 94.96$
Gope et al.'s [34]	$7T_h$	$9T_h$	$3T_h$	$19T_h = 6.08$
Ours	$9T_h + 1T_{fe}$	$10T_h$	$4T_h$	$23T_h + 1T_{fe} = 24.46$

6. Security and Performance Comparisons

In this section, we discuss the typical costs of the authentication protocols from three aspects: protocol security, computing cost, and storage consumption [22, 32–34].

6.1. Security Comparisons. As shown in Table 2, we compared the security analysis of the mentioned protocols and used ✓ and × to signify whether the protocol meets the security requirements involved. The security of the protocol proposed by Kumari et al. [32] was disproved by Li et al. [35] in that it cannot resist sensor node capture attacks, session-specific temporary information attacks, sensor node impersonation attacks, and man-in-the-middle attacks. Therefore, Li et al. designed a mutual authentication and key agreement protocol for wireless sensor networks. However, it was later proved to be unsafe. The protocol proposed by Srinivas et al. [33] cannot resist offline password-guessing attacks. The security of the protocol proposed by Gope and Hwang [34] was disproved by Adavoudi-Jolfaei et al. [36] in that the adversary can obtain the session key between the user and the sensor using the dy model. Compared to the protocols mentioned above, our proposed protocol can resist such attacks and meet the security requirements.

6.2. Performance Comparisons. We performed a performance comparison between the new authentication protocol and the other four authentication protocols listed in Table 4. Additionally, we made the following calculations in terms of the time consumption of cryptographic operations, as shown in Table 3, including hash functions, symmetric key encryption/decryption, chaotic mapping functions, and fuzzy extraction functions, as the most important operations [22]. The meanings of symbols in Table 4 are as follows: T_h denotes the time of the regular hash operation, T_{fe} denotes the operation time of the fuzzy function, T_s denotes the operation time of symmetric encryption and decryption, and T_c denotes the operation time of the chaotic map function.

In the login and mutual authentication phase, we compared the computation times of the user, gateway, and sensor node sides along with other protocols to design our proposed protocol. As shown in Table 4, the newly designed protocols guarantee security and time appropriateness. Although our new protocol takes slightly more time than the protocols proposed in Fotouhi et al.'s [22] and Gope and Hwang's [34], it ensures improved security. This is because the extra time spent is mainly in the user login phase, where the user biometric information needs to be compared, a very important and indispensable step that amounts to a partial performance sacrifice to improve the security of the

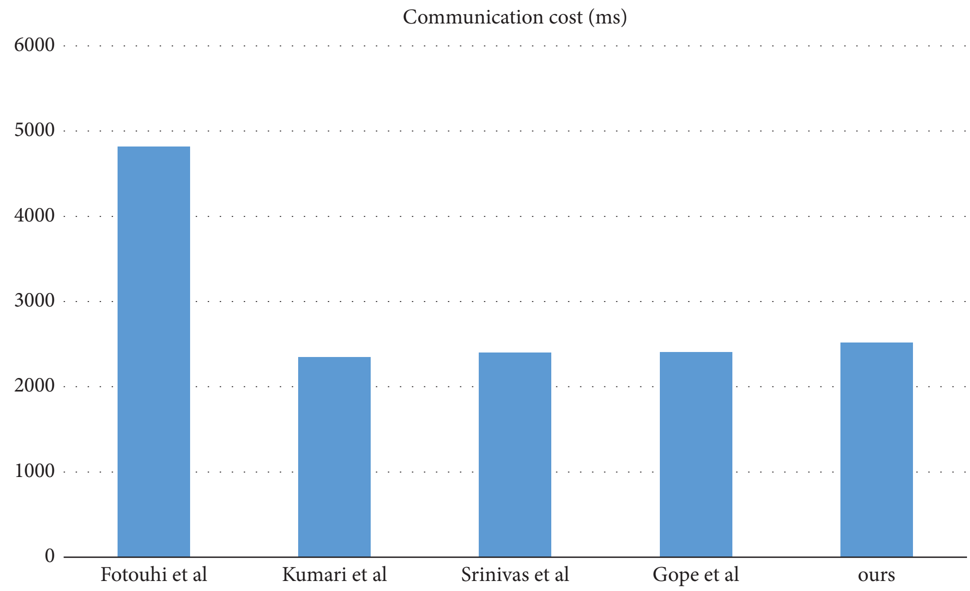

Figure 4: Communication cost.

protocol. As a result, the new protocol is more secure than the two protocols and ensures that the user's legitimacy is verified. Compared to Kumari et al.'s [32] and Srinivas et al.'s [33] proposed protocols, it is evident that our proposed protocol significantly reduces the computational cost. In addition, we compared the communication costs, as shown in Figure 4. Considering the computational cost and communication in terms of cost and security for the new protocol, it is evident that our proposed protocol can be better adapted to the wireless human medical environment regional network, thereby providing improved service experience for hospital staff and individual patients.

7. Conclusion

In this study, we improve on the WBAN-based authentication protocol proposed by Fotouhi et al. in medical IoT. The improved protocol compensates for the defects in the original protocol, and it can resist attacks that cannot be resisted by the original protocol. It also improves the authentication speed of the protocol, thereby reducing computational expenditure. Moreover, it is advantageous in that it is lightweight compared to the original protocol. The improved protocol adds biometric authentication and login authentication to significantly increase the security of the user login process, and it also makes extensive use of single hash, heterogeneous, and joint operations to reduce computational cost. Our proposed protocol is highly secure against a range of attacks, such as sensor node capture attacks, replay attacks, and internal privilege attacks. It demonstrates excellent performance in terms of security and efficiency. Therefore, it can be considered more suitable for the WBAN-based medical IoT. For every new technology development there are bound to be technical implementation and realization challenges, and the Internet of Healthcare is facing some problems in terms of adoption for the time being. Most of the problems exist because there is no all-in-one healthcare IoT solution; all solutions are tailored to specific challenges and therefore can be too expensive for any organization. The second is the lack of a set of standards for the healthcare industry to protect extremely sensitive healthcare data from security risks and threats. It is hoped that this paper will provide a reference for addressing the security aspects of healthcare data.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work of Long Li was supported by Guangxi Key Laboratory of Trusted Software (no. KX202033).

References

[1] M. Shafi, A. F. Molisch, P. J. Smith et al., "5G: a tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1201–1221, 2017.

[2] H. Xiong, X. Huang, M. Yang, L. Wang, and S. Yu, "Unbounded and efficient revocable attribute-based encryption with adaptive security for cloud-assisted internet of things," *IEEE Internet of Things Journal*, vol. 2021, Article ID 3094323, 2021.

[3] H. Xiong, Y. Wu, C. Jin, and S. Kumari, "Efficient and privacy-preserving authentication protocol for heterogeneous systems in IIoT," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11713–11724, 2020.

[4] X. Chen, M. Li, H. Zhong, Y. Ma, and C. H. Hsu, "DNNOff: offloading DNN-based intelligent IoT applications in mobile

edge computing," *IEEE Transactions on Industrial Informatics*, vol. 2021, Article ID 3075464, 2021.

[5] J. W. Jiao Wang, J.-S. P. Jiao Wang, S.-C. C. Jeng-Shyang Pan, Z.-Y. M. Shu-Chuan Chu, and H. L. Zhen-Yu Meng, "Improved black hole algorithm for intelligent traffic navigation," *Journal of Internet Technology*, vol. 22, no. 4, pp. 725–734, 2021.

[6] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 2021.

[7] S. Lv and Y. Liu, "PLVA: privacy-preserving and lightweight V2I authentication protocol," *IEEE Transactions on Intelligent Transportation Systems*, vol. 2021, Article ID 3059638, 2021.

[8] P. Wang, C. M. Chen, S. Kumari, M. Shojafar, R. Tafazolli, and Y. N. Liu, "HDMA: hybrid D2D message authentication scheme for 5G-enabled VANETs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 2020, Article ID 3013928, 2020.

[9] J. Song, Q. Zhong, W. Wang, C. Su, Z. Tan, and Y. Liu, "FPDP: flexible privacy-preserving data publishing scheme for smart agriculture," *IEEE Sensors Journal*, vol. 2020, Article ID 3017695, 2020.

[10] E. K. Wang, X. Liu, C. M. Chen, S. Kumari, M. Shojafar, and M. S. Hossain, "Voice-transfer attacking on industrial voice control systems in 5G-aided IIoT domain," *IEEE Transactions on Industrial Informatics*, vol. 2020, Article ID 3023677, 2020.

[11] W. Zhang, Y. Wu, H. Xiong, and Z. Qin, "Accountable attribute-based encryption with public auditing and user revocation in the personal health record system," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 15, no. 1, pp. 302–322, 2021.

[12] J. Sun, H. Xiong, X. Liu, Y. Zhang, X. Nie, and R. H. Deng, "Lightweight and privacy-aware fine-grained access control for IoT-oriented smart health," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6566–6575, 2020.

[13] C.-M. Chen, C.-T. Li, S. Liu, T.-Y. Wu, and J.-S. Pan, "A provable secure private data delegation scheme for mountaineering events in emergency system," *Ieee Access*, vol. 5, pp. 3410–3422, 2017.

[14] E. Jovanov, A. Milenkovic, C. Otto et al., "A WBAN system for ambulatory monitoring of physical activity and health status: applications and challenges," *IEEE*, vol. 2005, Article ID 1615290, 3813 pages, 2005.

[15] M. R. Yuce, "Implementation of wireless body area networks for healthcare systems," *Sensors and Actuators A: Physical*, vol. 162, no. 1, pp. 116–129, 2010.

[16] J. Liu, Z. Zhang, X. Chen, and K. S. Kwak, "Certificateless remote anonymous authentication schemes for wirelessbody area networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 2, pp. 332–342, 2013.

[17] Z. Zhao, "An efficient anonymous authentication scheme for wireless body area networks using elliptic curve cryptosystem," *Journal of Medical Systems*, vol. 38, no. 2, pp. 13–17, 2014.

[18] S. Chatterjee, A. K. Das, and J. K. Sing, "A novel and efficient user access control scheme for wireless body area sensor networks," *Journal of King Saud University - Computer and Information Sciences*, vol. 26, no. 2, pp. 181–201, 2014.

[19] C. Wang and Y. Zhang, "New authentication scheme for wireless body area networks using the bilinear pairing," *Journal of Medical Systems*, vol. 39, no. 11, pp. 1–8, 2015.

[20] P. K. Dhillon and S. Kalra, "Multi-factor user authentication scheme for IoT-based healthcare services," *Journal of Reliable Intelligent Environments*, vol. 4, no. 3, pp. 141–160, 2018.

[21] F. Wu, X. Li, A. K. Sangaiah et al., "A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor networks," *Future Generation Computer Systems*, vol. 82, pp. 727–737, 2018.

[22] M. Fotouhi, M. Bayat, A. K. Das, F. Han, S. M. Pournaghi, and M. A. Doostari, "A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care IoT," *Computer Networks*, vol. 177, Article ID 107333, 2020.

[23] 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.

[24] D. Wang, D. He, P. Wang, and C. H. Chu, "Anonymous two-factor authentication in distributed systems: certain goals are beyond attainment," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 4, pp. 428–442, 2014.

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

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

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

[28] P. Li, J. Su, and X. Wang, "iTLS: lightweight transport-layer security protocol for iot with minimal latency and perfect forward secrecy," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6828–6841, 2020.

[29] L. Xiong, D. Peng, T. Peng, H. Liang, and Z. Liu, "A lightweight anonymous authentication protocol with perfect forward secrecy for wireless sensor networks," *Sensors*, vol. 17, no. 11, p. 2681, 2017.

[30] 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 Communications and Mobile Computing*, vol. 2021, Article ID 5537018, 2021.

[31] M. Safkhani, C. Camara, P. Peris-Lopez, and N. Bagheri, "RSEAP2: an enhanced version of RSEAP, an RFID based authentication protocol for vehicular cloud computing," *Vehicular Communications*, vol. 28, Article ID 100311, 2021.

[32] S. Kumari, X. Li, F. Wu, A. K. Das, H. Arshad, and M. K. Khan, "A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps," *Future Generation Computer Systems*, vol. 63, pp. 56–75, 2016.

[33] J. Srinivas, D. Mishra, and S. Mukhopadhyay, "A mutual authentication framework for wireless medical sensor networks," *Journal of Medical Systems*, vol. 41, no. 5, p. 80, 2017.

[34] P. Gope and T. Hwang, "A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7124–7132, 2016.

[35] J. Li, W. Zhang, S. Kumari, K. K. R. Choo, and D. Hogrefe, "Security analysis and improvement of a mutual authentication and key agreement solution for wireless sensor networks using chaotic maps," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 6, Article ID e3295, 2018.

[36] A. Adavoudi-Jolfaei, M. Ashouri-Talouki, and S. F. Aghili, "Lightweight and anonymous three-factor authentication and access control scheme for real-time applications in wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 12, no. 1, pp. 43–59, 2019.