Computers, Materials & Continua
DOI:10.32604/cmc.2021.014966
Article

Tech Science Press

Cogent and Energy Efficient Authentication Protocol for WSN in IoT

Tariq Mahmood Butt[1], Rabia Riaz[1], Chinmay Chakraborty[2], Sanam Shahla Rizvi[3] and Anand Paul[4,*]

[1]Department of CS & IT, University of Azad Jammu and Kashmir, Muzaffarabad, 13100, Pakistan
[2]Birla Institute of Technology, Mesra, Jharkhand, 814142, India
[3]Raptor Interactive (Pty) Ltd., Eco Boulevard, Witch Hazel Ave, Centurion, 0157, South Africa
[4]The School of Computer Science and Engineering, Kyungpook National University, Daegu, 41566, Korea
[*]Corresponding Author: Anand Paul. Email: paul.editor@gmail.com
Received: 29 October 2020; Accepted: 02 February 2021

Abstract: Given the accelerating development of Internet of things (IoT), a secure and robust authentication mechanism is urgently required as a critical architectural component. The IoT has improved the quality of everyday life for numerous people in many ways. Owing to the predominantly wireless nature of the IoT, connected devices are more vulnerable to security threats compared to wired networks. User authentication is thus of utmost importance in terms of security on the IoT. Several authentication protocols have been proposed in recent years, but most prior schemes do not provide sufficient security for these wireless networks. To overcome the limitations of previous schemes, we propose an efficient and lightweight authentication scheme called the Cogent Biometric-Based Authentication Scheme (COBBAS). The proposed scheme is based on biometric data, and uses lightweight operations to enhance the efficiency of the network in terms of time, storage, and battery consumption. A formal security analysis of COBBAS using Burrows–Abadi–Needham logic proves that the proposed protocol provides secure mutual authentication. Formal security verification using the Automated Validation of Internet Security Protocols and Applications tool shows that the proposed protocol is safe against man-in-the-middle and replay attacks. Informal security analysis further shows that COBBAS protects wireless sensor networks against several security attacks such as password guessing, impersonation, stolen verifier attacks, denial-of-service attacks, and errors in biometric recognition. This protocol also provides user anonymity, confidentiality, integrity, and biometric recovery in acceptable time with reasonable computational cost.

Keywords: Internet of things; wireless sensor networks; authentication; Burrows–Abadi–Needham logic; fuzzy extractor; elliptic curve cryptography

1 Introduction

The core purpose of the Internet of things (IoT) is a convergence of the physical and digital worlds. On the IoT, a set of sensors is attached to a thing (object or device) from which sensors collect various data and transmit it to a central system via a public network. The central system organizes data and extracts results before sending it to an authorized recipient. Consequently, an authorized user can remotely connect with that object or thing. Statistics show that the IoT market is continuously growing. By 2019, the IoT market growth was 212 billion US dollars, and in 2020, it is expected to reach up to 248 billion US dollars [1]. Moreover, the number of connected devices is expected to reach 100 billion by 2030. There are many examples of IoT, such as smart cities, smart homes, industry and building automation systems, and health care systems. A wireless sensor network (WSN) embedded in a building/home provides services such as heat control, air conditions, refrigerator, and lighting control, security and surveillance. Conventional cryptographic algorithms may not be practicable for WSN or IoT due to insufficient computational and storage resources of remote sensor systems. Furthermore, traditional password-based protocols can be vulnerable because they are easily breakable, especially by social engineering. In the IoT, a user registers on a network themselves to acquire data from the sensors. This registration is performed by a gateway node; after successful registration, a user may be able to access secret information. During the registration or login phase, an intruder can easily obtain data and secret information because this information is transmitted through public networks. In such situations, an efficient, lightweight, and intelligent scheme is required to ensure the security of wireless sensors. To ensure that communication between a user and sensor nodes remains secure, various authentication schemes have been proposed over the last decades; however, most of these schemes fail to provide sufficient security for practical applications and future development. In this study, we propose an authentication scheme using light operations, providing a higher level of security than previously proposed related schemes.

For the analysis of the proposed scheme, we use the Dolev–Yao model [2], which is based on the assumption that an adversary can attack at any time and at any level. In the login and authentication phase, an adversary can steal the password or impersonate a legal user or node. Similarly, an adversary can repudiate and change the content of a message. An attacker can also send fake messages to the gateway and sensor nodes, and involve the nodes in useless tasks. All the above threats are considered in the proposed protocol. We have relied on hashing and encryption algorithms for security. This protocol also uses ECC and RC5 to protect networks from attackers. The contributions of this research are briefly summarized below.

- A detailed analysis of recent biometric-based authentication schemes, highlighting their limitations, is presented, particularly a cryptanalysis of the scheme proposed by Riaz et al. [3].
- A new scheme named Cogent Biometric-Based Authentication Scheme (COBBAS) is proposed that provides sufficient security and lightweight operations, enhancing the network efficiency in terms of communication and computational overload.
- Time stamps have been used in the majority of existing schemes to ensure data freshness. Because a time stamp requires clock synchronization between the user's mobile device or PC and a WSN, it is an unreasonable way to ensure data freshness. COBBAS uses a nonce value instead of a time stamp to ensure data freshness.
- The authenticity of the proposed scheme is formally analyzed using Burrows–Abadi–Needham (BAN) logic. Moreover, the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool is used to demonstrate that the proposed scheme is secure.
- Informal security analysis of COBBAS is performed to check its protection against various cybersecurity attacks.

- The efficiency of the proposed scheme in terms of time and computational cost is also compared with recent existing schemes.

In the remaining paper, Section 2 provides a comprehensive literature review of prior schemes. The detailed work of the Secure User Biometric Based Authentication Scheme (SUBBASe) and its limitations are presented in Section 3. The proposed scheme is explained in Section 4. The results, including formal and informal analyses, are presented in Section 5. Section 6 concludes.

2 Literature Review

To ensure the security of WSNs, many password-based authentication schemes have been proposed over the last few decades [4–13]. More recently, Jian Jun et al. [14] proposed a biometric authentication scheme consisting of four phases: registration, login, authentication, and password change. As transmitted messages are not encrypted in this scheme, if an unauthorized user obtains control over a sensor node, he/she can easily capture all the information stored on that sensor node. In this scheme, secure channels are not provided; thus, it has major problems with data confidentiality and integrity.

Khan and Alghathbar's (K-A) scheme [10] is a password-based authentication scheme; however, it is defenseless against non-repudiation and mutual authentication between the user and gateway node. Yuan modified the K-A scheme to remove these weaknesses in their protocol [15]. In addition, Das proposed a scheme that consists of a registration phase, login, authentication, and a key agreement phase. His scheme resists insider attacks, online password-guessing attacks, and biometric key-guessing attacks. The author passwords, smart cards, and user biometrics for authentication. Hence, this authentication scheme depends on three factors [16].

To address the weaknesses of Yuan's scheme, Wei et al. [17] proposed a scheme that suffers from many vulnerabilities such as the misuse of biometrics, stolen smart card attacks, gateway node impersonation attacks, and a lack of session key establishment. Wei et al. proposed a three-phase scheme that removed some of the weaknesses of the Yuan scheme. Similarly, Wu et al. [18] identified the weaknesses in Das' scheme; i.e., it is defenseless against offline password-guessing and de-synchronization attacks. To improve Das' scheme, Wu et al. proposed a two-step registration phase of user registration and sensor node registration. In this scheme, they provide a secure mechanism against insider, offline password-guessing, user forgery, and gateway forgery attacks.

Park et al. [19] proposed a three-factor authentication scheme. This scheme draws its efficiency from the use of elliptic curve cryptography and a fuzzy extractor. Maurya et al. [20] proposed another fuzzy extractor-and ECC-based scheme consisting of four phases. However, Maurya's scheme was inefficient with regard to computation time.

Kang et al. [21] proposed a scheme to address the problems of a lack of user anonymity and offline password-guessing attacks in previous schemes. Their analysis shows that their scheme provides a high level of security without the need for time synchronization. The Bi-Phase Authentication Scheme (BAS) [22] was proposed to improve Wong et al. [6] scheme. BAS consists of initial and final authentication phases, and uses special hardware called Full Function Devices (FFDs) and Reduced Function Devices (RFDs). BAS has several weaknesses; for instance, the protocol requires extra hardware, message confidentiality is not considered, and a session key is not established after user authentication.

To address the weaknesses of the BAS scheme, the SUBBASe was proposed in [3]. SUBBASe provides mutual authentication and network defense against several common security attacks such as stolen verifier attacks, message confidentiality, replay attacks, guessing attacks, and network traffic attacks. However, it suffers from security vulnerabilities such as biometric recognition errors, user anonymity issues, perfect forward secrecy, and gateway node impersonation attacks. To overcome the vulnerabilities of SUBBASe, Riaz et al. [23] proposed a scheme with two phases. The first phase was "Registration" and the second was "Login and Authentication." This scheme overcomes the weaknesses of SUBBASe, removes biometric recognition errors, and ensures user anonymity. However, it still suffers from gateway node impersonation attacks and Sybil attacks. In addition, these schemes use a time stamp to ensure data freshness. Because the time stamp requires clock synchronization between a user's mobile device or PC and the WSN, this timestamp is not a practical way to ensure data freshness.

Authentication in WSNs has attracted considerable research attention in the domain of IoT and smart homes. In 2019, Shin et al. [24] proposed a smart card-based authentication scheme for smart homes. Their scheme consists of five phases, but takes considerable time to complete its run. Lightweight three-factor authentication schemes have been proposed for IoT and 5G in [25,26] respectively.

All these schemes require smart cards to store user biometric information and provide authentication. These schemes provides desirable attributes for IoT environments, and authors' shows that the computation and communication costs of their proposed scheme are suitable for extremely low-cost IoT devices. However, the need for smart cards was removed using the scheme proposed in [27]. Biometric and smartcard-based authentication schemes have been proposed for health care in [28,29]. A stream-based authentication mechanism, using key authorization infrastructure, specifically addressing security concerns of multi-homing sub-aqueous big data networks was presented in [30]. A novel Fractal-Based Authentication Technique was proposed by implementing a Sierpinski triangle in [31]. Their scheme reduces the probability of password guessing, and provides security against attacks such as shoulder surfing. To reduce energy consumption in IoT, a game-based mechanism was suggested in [32], and an electoral system was proposed in [33]. These schemes select the most appropriate cluster heads or community heads to enhance the efficiency of a network. Human activity recognition in home automation systems is an emerging topic addressed in [34]. Their scheme provides an efficient technique to observe human behavior within a smart home. The study of human behaviors can also be helpful in designing authentication mechanisms.

All these schemes are valuable additions to IoT systems; however, they also suffer from several security threats. To overcome the flaws in these schemes, we propose a scheme called "COBBAS." It not only provides sufficient security, but also uses nonce values instead of time stamps to ensure data freshness. Furthermore, this scheme removes biometric recognition errors using fuzzy extraction. The authenticity of the proposed scheme is formally analyzed using BAN logic. Moreover, the AVISPA tool is used to prove that the proposed scheme is secure.

3 Review of SUBBASe

In this section, we first provide a brief description of the details of SUBBASe [3], and then, conduct a security analysis to explain its vulnerabilities. For convenience, the notations used in this paper are given in Tab. 1.

This scheme has two phases. In the first phase, the user enrolls themselves with the network, which is called the enrollment phase. The second is an authentication phase where the user is authenticated by a trusted node, and the required information is provided to the user.

Table 1: Notations used in SUBBASe and proposed scheme

Abbreviations	Description
U_i	Ith user
ID_i	Identity of ith user
TN	Trusted node
SN	ID of sensor node
Sk	Session key
ΔT	Time interval required
$\|\|$	Concatenation operator
x_o	Secrete known to trusted node
E_{sk}	Encryption with session key
D_{sk}	Decryption with session key
B_i	Biometric of user
RI	Requested information
PW	Password
HPW	Hash of password
N_a	Nonce by users
N_s	Nonce by gateway node
P_{fe}	Helping string in fuzzy extractor
R_i	Random string generated by fuzzy extractor for ith user
A_i	Hash of biometric of ith user

Before deployment, each node in the network is preloaded with the following information: ID of sensor node and secret value x_o. We describe both phases and security weaknesses below.

3.1 Enrollment Phase

(a) In the enrollment phase, the user registers with the network. The user imprints their biometric data, and calculates its hash value. Then, the user inputs ID_i and sends it to a trusted node, i.e., ID_i and A_i.

(b) The trusted node receives the value from the user and calculates s, which is a hash value of the ID_i of a user and a secret x_o, i.e., $s = h(ID_i||x_o)$. Then, the trusted node sends this value to the user.

3.2 Authentication Phase

(a) In this phase, the user imprints finger B_i and calculates the hash value of biometric $A'_i = h(B_i)$. Then, the user inputs his/her ID_i, takes a time stamp T_o, requests information RI, and sends the following message to the sensor node.

$$M_1 = ID_i, RI, A'_i, T_o \quad where\ A'_i = h(B_i), \tag{1}$$

(b) After receiving a message from the user, the sensor node calculates the time interval if $T_1 - T_0 > \Delta T$; if this condition is true, the request will be rejected. Otherwise, the sensor node computes y, and sends the following message to the trusted node.

$$M_2 = ID_i, y, T_2 \quad where\ y = h(ID_i \| A'_i \| SN) \tag{2}$$

(c) The trusted node receives the message and checks the time stamp if $T_3 - T_2 > \Delta T$ then request will be rejected; otherwise, the trusted node checks A'_i and compares A_i with the previously saved A_i. If $A_i = A'_i$ is not satisfied, then the trusted node sends a reject message $\{reject\ldots\}$ to the sensor node; otherwise, it sends M_5 = *[In-Progress...]* The message in-progress means that all parameters have been verified, and data will be provided presently after some calculations. Then, the trusted node calculates *s* and sends the following message to *SN*.

$$M_3 = Accpt,\ s,\ T_4, \quad where\ s = h(ID_i||x_o) \tag{3}$$

(d) The sensor node first verifies the timestamp; if condition $T_5 - T_4 > \Delta T$ becomes true, then the request is rejected. In this case, the sensor node computes $s = h\,(ID_i||x_o)$, calculates the values $d = (RI)$ and $sk = h(IDi||T_6||s)$, and encrypts the data with the help of this session key. Subsequently, this sensor node and user will use this session key to access a session. $E_{SK}(d)$ represents the encryption of the required user information.

$$M_4 = Accpt,\ e,\ T_6, \quad where\ e = E_{SK}(d) \tag{4}$$

(e) The user verifies the time stamp if $T_7 - T_6 \geq \Delta T$ is true; if they have *s*, then they can obtain their required information. The user first calculates their session key, and then, decrypts data *e* with the help of a session key. The user calculates the session key with ID_i T_6 and *s*, $sk = h(ID_i||T_6||s)$. With *sk*, users can decrypt the required information, $D_{SK}(e)$.

3.3 Weaknesses of SUBBASE

3.3.1 Insecure User ID

In SUBBASe, the ID_i of the user is sent on public network without encryption. In the enrollment phase, the user inputs his/her ID_i, imprints his/her biometric B_i, calculates A_i, which is the hash of B_i, and sends it to the trusted node (*TN*). Similarly, in the authentication phase, the user imprints finger B_i and calculates the hash value of biometric $A'_i = h(B_i)$. The user inputs his/her ID_i, takes a time stamp T_o and requested information *RI*, and sends the following message to the sensor node. $M_1 = ID_i$, *RI*, A'_i, T_o. In both phases, ID_i is sent openly on a public network, making it insecure.

3.3.2 Biometric Recognition Error

A hash function returns a different value even if a single bit changes in input. Conversely, biometric input contains various noise and cannot reproduce 100% identical output over multiple access attempts. In the enrollment phase, the users imprint their biometric B_i and calculate its hash value A_i. Moreover, in the authentication phase, the user again imprints finger B_i and calculates the hash value of biometric $A'_i = h(B_i)$. Based on the above discussion, it is possible that a user's device will produce a different hash value A_i. Consequently, errors in biometric recognition will occur, causing termination of the authentication process.

3.3.3 Vulnerable Session Key

The session keys perform a crucial role in security. Moreover, sensor nodes have limited computational resources. The SUBBASe session key (*sk*) is created by the sensor node through the following operations; the sensor node computes $s = h(ID_i||x_o)$ to calculate $sk = h(ID_i||T_6||s)$. Then, the sensor node and user will use this key as a session key for the ongoing session. E_{SK}(d) represents the encryption of the required user information. Session key *sk* is calculated using parameters ID_i, T_6, and *s*. As we know that ID_i is not encrypted, the adversary can easily obtain it. Similarly, $s = h(ID_i||x_o)$ also depends on ID_i and x_o. In other words, the session key only depends on ID_i, x_o, and T_6, which is not efficient. Moreover, the calculation of hash values by a sensor node is not efficient in terms of the computational load. This task can instead be performed by a trusted node that has high computational capacity.

4 Proposed Scheme: COBBAS

In proposed scheme, the following information is preloaded onto the network nodes.

- The node *ID*
- A secret value x_o

The secret value x_o is shared among the user, sensor node (SN), and gateway node (GN). The login and authentication phases are performed by both the sensor and gateway nodes. Two types of devices are used in WSNs, namely FFDs and RFDs. In this scheme, an FFD acts as an authenticator, whereas an RFD continuously manages communication among the devices. In this scheme, the fingerprint of a user is collected, and a random string is generated using a fuzzy extractor. The collection of fingerprints does not require special hardware, because a user can easily imprint his/her biometric on his/her personal tablet or PC to login into the network. COBBAS uses *SHA-256* to perform a one-way hash function. It uses RC5 with a key size of 20 bytes, as this size is the most suitable for resource-constrained devices. The phases of the proposed scheme are described below.

4.1 Registration Phase

In the registration phase, the user is registered with the network. The steps are briefly discussed below.

(a) The user inputs his/her ID_i, password PW, and biometric B_i using a tablet or PC. The user computes ID_i^* and a hash of his/her password HPW, and generates random strings R_i and P_{fe} from input B_i using the fuzzy extractor Gen algorithm: $Gen(B_i) = R_i, P_{fe}$. Moreover, the user calculates A_i and HPW from R_i and PW, respectively. Then, the user sends message M_1, comprising the following values to the gateway node through a secure channel.

$$M_1 = \left\langle ID_i^*, A_i, HPW \right\rangle, \quad where\ ID_i = ID_i^* \oplus x_o,\ and\ A_i = h(R_i) \tag{5}$$

(b) The gateway node calculates the authentication measures using the received values. The gateway node calculates N_i, which will be used by the gateway node to authenticate the user during the login and authentication phase. S_i is used for user anonymity, and M_i is used by the sensor node to authenticate the gateway node in the login and authentication phase. The gateway node broadcasts M_2, which comprises the following values (N_i, S_i, M_i), to the user and sensor node.

$$M_2 = \langle N_i, S_i, M_i \rangle, \quad where\ N_i = HPW \oplus x_o,\ S_i = ID_i \oplus x_o,\ and\ M_i = S_i \oplus h(A_i \oplus HPW) \tag{6}$$

(c) The gateway node broadcasts the above parameters to the user and all nodes in the network. After this step, the registration phase is complete. Fig. 1 describes the process of registration.

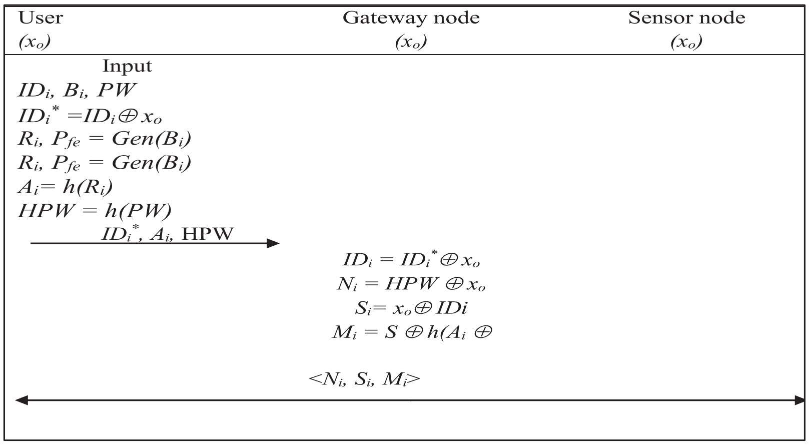


Figure 1: Registration phase of proposed scheme

4.2 *Login and Authentication Phase*

The steps included in login and authentication phase are given below.

(a) The user again enters his/her ID_i and password PW^*, and imprints his/her biometric B_i^*. Using B_i^* and a helping string P_{fe}, the user generates R_i^* with the help of the fuzzy extractor *Rep* algorithm as $Rep(B_i^*,\ P_{fe}) = R_i$. Then, the user calculates the hash of R_i^*, $A_i^* = h(R_i^*)$ and the hash of the password, $HPW^* = h(PW^*)$. Moreover, the user calculates variable Y_i as $Y_i = ID_i \oplus N_i$, where N_i is $N_i = HPW \oplus x_o$. Subsequently, this user calculates ID_i^* as $ID_i^* = Y_i \oplus HPW^*$. The user selects a random number r_u and nonce value N_a, and calculates $X_i = (P_{ec} \times r_u) \oplus x_o$, which is the product of a point on an elliptic curve and a generated random number r_u that is then XOR with secret value x_o. Finally, the user sends the following message to the gateway node.

$$M_3 = \left\langle ID_i^*,\ A_i^*,\ X_i HPW^*,\ RI,\ N_a \right\rangle, \quad where\ X_i = (Pec \times ru) \oplus x_o\ and\ ID_i^* = Y_i \oplus HPW^* \tag{7}$$

(b) The gateway node first verifies the user as $ID_i = ID_i^* \oplus x_o$. If the condition holds true, then the gateway node proceeds further; otherwise, the request is rejected. Moreover, the gateway node compares A_i^* and HPW^*; with the previously saved values A_i and HPW. If all these conditions are satisfied, then the gateway node calculates $S_i^* = x_o \oplus ID_i^*$ and $M_i^* = S_i^* \oplus h(A_i^* \oplus HPW^*)$. Moreover, GN generates a random number r_s and calculates the public key $D_i = r_s \times p_{ec}$ and $C_i = X_i^* \times r_s$. The gateway node then sends the following message along with the required information RI to the sensor node.

$$M_4 = \left\langle ID_i^*,\ X_i,\ A_i^* C_i,\ ID_i,\ HPW^*,\ RI,\ M_i^*,\ N_a \right\rangle, \quad where\ M_i^* = S_i^* \oplus h(A_i^* \oplus HPW^*) \tag{8}$$

(c) The sensor node first verifies the gateway node and user by comparing $M_i = M_i^*$ and calculates $X_i^* = X_i \oplus x_o$. The sensor node calculates the session key using ID_i, C_i, N_a, and x_o as $SK_s = h(C_i || ID_i || N_a || x_o)$. It encrypts the required information RI using the sensor session key SK_s as $e = ESK_s(RI)$. Moreover, SN generates a nonce N_s and sends message M_5 to the user.

$$M_5 = \langle D_i,\ e,\ N_s,\ J_i \rangle, \quad where\ e = ESK_s(RI)\ and\ Ji = ESK_s(N_a) \tag{9}$$

(d) The user calculates C_i with the help of D_i, $C_i = D_i \times r_u$. The user then calculates the session key $SK_u = h(C_i||ID_i||N_a||x_o)$ and decrypts data e with the help of their session key SK_u as $DSK_u\ (e) = RI$.

(e) The user sends an acknowledgment to the sensor node to ensure mutual authentication as $(N_s \oplus x_o)SK_u$. The authentication phase ends with this step.

(f) The sequence of message exchanges in the login and authentication phases are shown in Fig. 2.

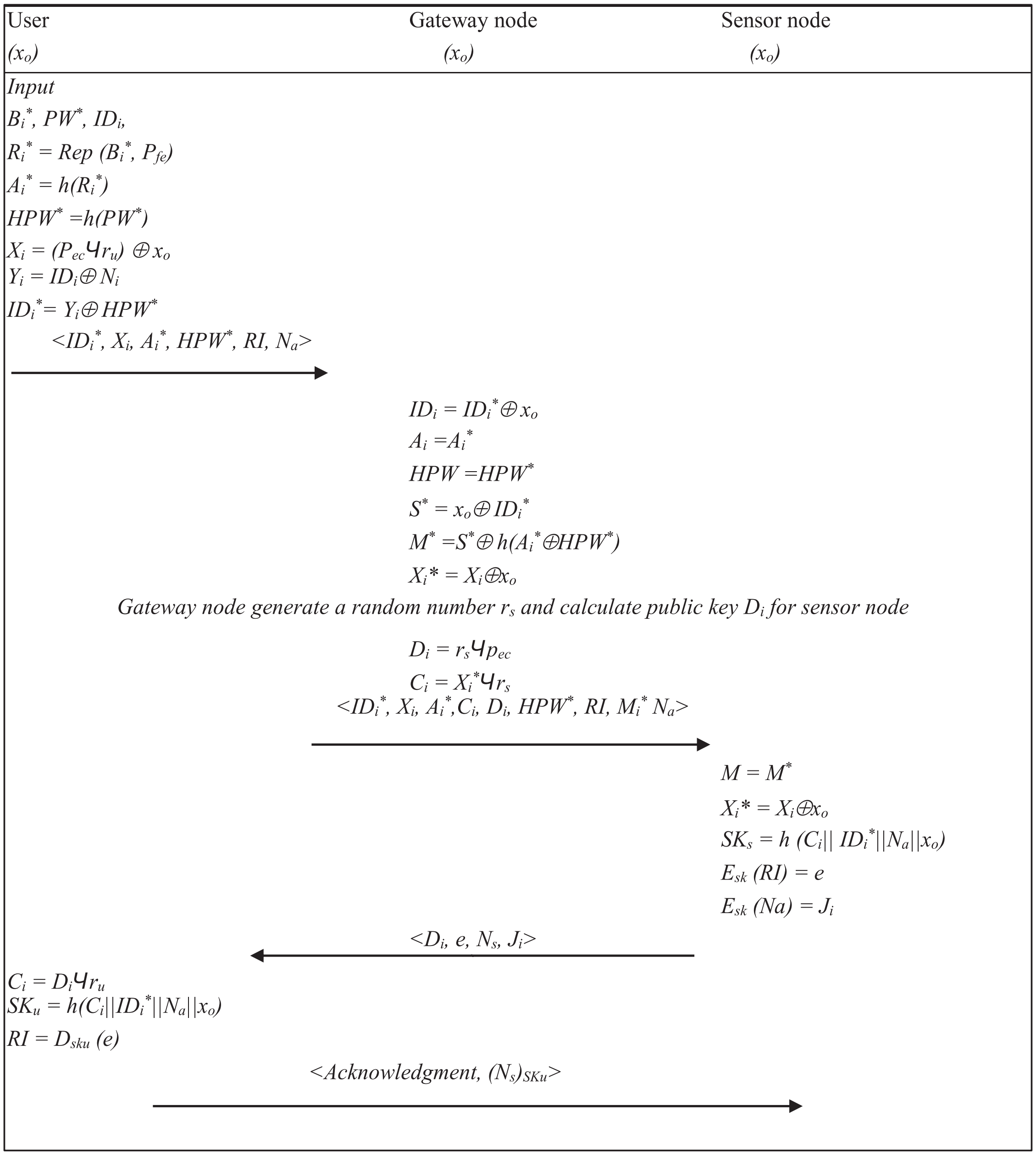

User
(x_o)
Gateway node
(x_o)
Sensor node
(x_o)
Input
B_i^*, PW^*, ID_i,
$R_i^* = Rep\ (B_i^*, P_{fe})$
$A_i^* = h(R_i^*)$
$HPW^* = h(PW^*)$
$X_i = (P_{ec}Чr_u) \oplus x_o$
$Y_i = ID_i \oplus N_i$
$ID_i^* = Y_i \oplus HPW^*$
$<ID_i^*, X_i, A_i^*, HPW^*, RI, N_a>$
$ID_i = ID_i^* \oplus x_o$
$A_i = A_i^*$
$HPW = HPW^*$
$S^* = x_o \oplus ID_i^*$
$M^* = S^* \oplus h(A_i^* \oplus HPW^*)$
$X_i* = X_i \oplus x_o$
Gateway node generate a random number r_s and calculate public key D_i for sensor node
$D_i = r_sЧp_{ec}$
$C_i = X_i^*Чr_s$
$<ID_i^*, X_i, A_i^*, C_i, D_i, HPW^*, RI, M_i^*\ N_a>$
$M = M^*$
$X_i* = X_i \oplus x_o$
$SK_s = h\ (C_i||\ ID_i^*||N_a||x_o)$
$E_{sk}\ (RI) = e$
$E_{sk}\ (Na) = J_i$
$<D_i, e, N_s, J_i>$
$C_i = D_iЧr_u$
$SK_u = h(C_i||ID_i^*||N_a||x_o)$
$RI = D_{sku}\ (e)$
$<Acknowledgment, (N_s)_{SKu}>$

Figure 2: Login and authentication phases of proposed scheme

5 Results

5.1 *Formal Security Analysis Using BAN Logic*

COBBAS provides mutual authentication between the sensor node and user. In this section, we prove this using BAN logic [35]. The postulates of BAN logic are described, and the formal proof of the proposed scheme, which comprises “Assumptions,” “Messages,” “Goals,” and “Analysis” parts, is given below. The basic symbols used for BAN logic are described in Tab. 2.

Table 2: Notations and symbols used in BAN logic

Notation	Meaning
$\#X$	Statement X is fresh
$U|\sim X$	U once said X
$\{X\}_K$	Formula X is encrypted by key K
$\langle X\rangle_K$	Formula X is combined by key K
$U \Rightarrow X$	U control over the Statement X
$U \lhd X$	U see Statement X
$U\overset{K}{\rightarrow}S$	U and S share key K for communication
SK	Session key
$U \mid\equiv X$	U believes X
U_i	ith user

5.1.1 *Inference Rules*

Rule 1: Message meaning rule: $\frac{U|\equiv U\overset{K}{\leftrightarrow}S, U\lhd\{X\}_K}{U|\equiv S|\sim X}$. If U believes that he/she shares key K with S and U sees message X encrypted with key K, U believes that S once said X.

Rule 2: Nonce verification rule: $\frac{U|\equiv\#(X), U|\equiv S|\sim X}{U|\equiv S|\equiv X}$ If U believes X is fresh and S once said X, U believes S believes X.

Rule 3: Belief rule $\frac{U|\equiv S|\sim(x,y)}{U|\equiv S|\sim(x)}$. If U believes that X once said (X, Y), then U believes that S once said (X).

Rule 4: Freshness rule: $\frac{U|\equiv\#(X)}{U|\equiv\#(X,Y)}$. If part of a message is fresh, then the entire message is fresh.

Rule 5: Jurisdiction rule: $\frac{U|\equiv S|\Rightarrow X, \wedge U|\equiv S|\equiv X}{U|\equiv X}$. If U believes that S has jurisdiction over X and believes S believes X, U believes X.

Rule 6: Seeing rule: $\frac{\mathrm{U} \lhd (\mathrm{X},\mathrm{Y})}{\mathrm{U} \lhd (\mathrm{X})}$ If U sees (X, Y), then U sees X as well. In addition, the second seeing rule $\frac{\mathrm{U}| \equiv \mathrm{U} \overset{K}{\leftrightarrow} \mathrm{S} \lhd \{X\}_K}{\mathrm{U}| \equiv \mathrm{x}}$ means that U can see message X only if he/she knows the shared secret key K.

5.1.2 Idealized Form

The message exchange of COBBAS in idealized form is given below.

Message 1: U→GN: $\langle$ ID_i^*,$(r_u \times P_{ec})x_o$, N_a, (B_i)xo, (HPW)$x_o\rangle$

Message 2: GN→SN: $\langle N_a$, $(B_i$, ID_i, HPW$)x_o$, $(r_u \times P_{ec})x_o$, $(r_s \times P_{ec})x_o\rangle$

Message 3: SN→U: $\langle$ $(r_s \times P_{ec})x_o$, N_S, $(\mathrm{U}\overset{sk}{\leftrightarrow}\mathrm{SN})x_o\rangle$

Message 4: U→SN: $\langle N_S$, $(\mathrm{U}\overset{sk}{\leftrightarrow}\mathrm{SN})x_o\rangle$

5.1.3 Goals

To ensure secure operation, the proposed protocol should meet the following security goals.

Goal 1: U|≡ U $\overset{sk}{\longleftrightarrow}$ SN (U believes that U shares a secret session key with SN)

Goal 2: SN |≡ U $\overset{sk}{\longleftrightarrow}$ SN (SN believes that U shares a secret session key with SN)

Goal 3: U|≡ SN |≡U $\overset{sk}{\longleftrightarrow}$ SN (U believes that SN believes that U shares a secret session key with SN)

Goal 4 **:** SN|≡U |≡ U $\overset{sk}{\longleftrightarrow}$ SN (SN believes that U believes that U shares a secret session key with SN)

5.1.4 Assumptions

To proceed with the proof, following assumptions are made.

Assumption 1: U |≡ GN $\overset{xo}{\longleftrightarrow}$ U

Assumption 2: GN |≡GN $\overset{xo}{\longleftrightarrow}$ U

Assumption 3: GN |≡ GN $\overset{xo}{\longleftrightarrow}$ SN

Assumption 4: SN |≡ GN $\overset{xo}{\longleftrightarrow}$ SN

Assumption 5: SN |≡ U $\overset{xo}{\longleftrightarrow}$ SN

Assumption 6: U |≡ U $\overset{xo}{\longleftrightarrow}$ SN

Assumption 7: U |≡ #(N_a)

Assumption 8: SN |≡ #(N_s)

5.1.5 Analysis

Step 1: From Message 1,

$GN \lhd \langle\ (ID_i)x_o,\ (r_u\ \times P_{ec})x_o,\ N_a,\ (B_i)x_o,\ (HPW)x_o\rangle.$

Step 2: From the message meaning rule and Assumption 2

$GN\,|\!\equiv U\,|\!\sim ID_i,\ r_u\ \times P_{ec},\ N_a,\ B_i,\ HPW$

Step 3: From Message 2,

$SN \lhd \langle\ (r_u\ \times P_{ec})x_o,\ (B_i,\ HPW,\ B_i)x_o,\ N_a,\ (r_s\ \times P_{ec})x_o\rangle$

Step 4: From the message meaning rule and Assumption 4

$SN\,|\!\equiv GN\,|\!\sim r_u\ \times P_{ec},\ B_i,\ HPW,\ ID_i,\ N_a,\ r_s\ \times P_{ec}$

Step 5: From Message 3,

$U \lhd\ \ \langle\ N_a,\ N_s,\ (U\overset{sk}{\longleftrightarrow}SN)x_o,\ (r_s\ \times P_{ec})x_o\rangle.$

Step 6: From the message meaning rule and Assumption 6

$U\,|\!\equiv SN\,|\!\sim B_i,\ HPW,\ ID_i,\ N_a,\ r_s\ \times P_{ec},\ (U\ \overset{sk}{\leftrightarrow}\ SN)x_o$

Step 7: From Message 4,

$SN \lhd \langle\ (N_s)x_o\ (U\ \overset{sk}{\longleftrightarrow}\ SN)x_o\rangle.$

Step 8: From the message meaning rule and Assumption 5

$SN\,|\!\equiv U\,|\!\sim N_s\ ,\ (U\ \overset{sk}{\longleftrightarrow}\ SN)x_o.$

Step 9: From Step 6, the freshness rule, and Assumption 7,

$$U|\equiv\#(U\overset{sk}{\longleftrightarrow}SN) \tag{10}$$

Step 10: From Step 6 and the second seeing rule,

$$U|\equiv\#U\overset{sk}{\longleftrightarrow}SN\ (Goal\ 1) \tag{11}$$

Step 11: From Step 8, the freshness rule, and Assumption 8,

$$SN|\equiv\#(U\overset{sk}{\longleftrightarrow}SN) \tag{12}$$

Step 12: From Step 8 and the second seeing rule,

$$SN|\equiv\#U\overset{sk}{\longleftrightarrow}SN\ (Goal\ 2) \tag{13}$$

Step 13: From Steps 6 and 9, the nonce verification rule, and Assumption 7,

$$U|\equiv\#U\overset{sk}{\longleftrightarrow}SN\ (Goal\ 3) \tag{14}$$

Step 14: From the nonce verification rule, Assumption 8, and Steps 8 and 11,

$$SN|\equiv\#U\overset{sk}{\longleftrightarrow}SN\ (Goal\ 4) \tag{15}$$

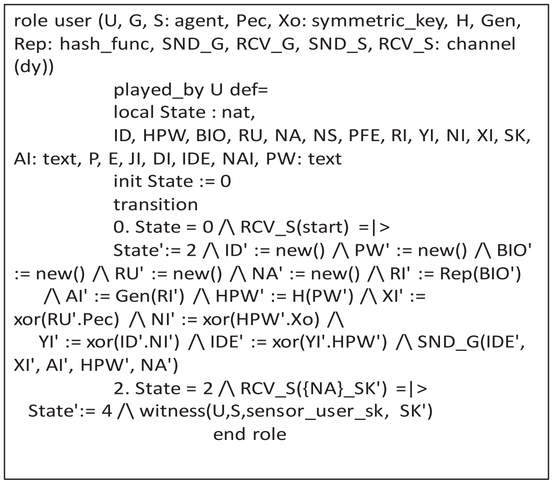
Key freshness is vital to security protocols. The results of Step 9, i.e., Eq. (10), prove that the user trusts the freshness of the key shared between the user and sensor node. Similarly, from

Step 11 Eq. (12), it is clear that the sensor node also believes that the key shared between itself and the user is fresh. Moreover, Step 10 shows that the user believes that he/she and the sensor node share the same secret key (Goal 1). Step 12 verifies that the sensor node also believes that it shares the same secret key with the user (Goal 2). Steps 13 and 14 verify that the user believes that the sensor node believes that the user and sensor node share the same secret key and vice versa (Goals 3 and 4).

5.2 *Formal Security Analysis with AVISPA*

AVISPA is an automated protocol validation tool. This tool uses high-level protocol specification language (HLPSL) [36]. AVISPA provides a suite of applications for building and analyzing formal models of security protocols written in HLPSL.

In this section, it is proven that the proposed scheme is safe against intruder attacks. The session key generated by the sensor node is safely received by the user. HLPSL is a role-based language. In the proposed scheme, three entities are involved: user (U), gateway node (GN), and sensor node (SN). The roles of these entities are described in the HLPSL code in Figs. 3–5, respectively.



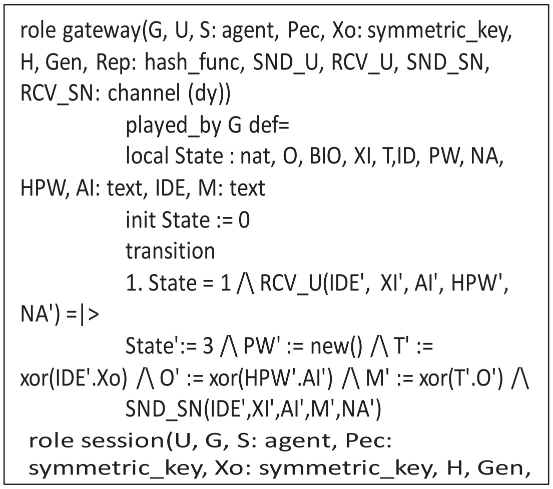
```
role user (U, G, S: agent, Pec, Xo: symmetric_key, H, Gen,
Rep: hash_func, SND_G, RCV_G, SND_S, RCV_S: channel
(dy))
              played_by U def=
              local State : nat,
              ID, HPW, BIO, RU, NA, NS, PFE, RI, YI, NI, XI, SK,
AI: text, P, E, JI, DI, IDE, NAI, PW: text
              init State := 0
              transition
              0. State = 0 /\ RCV_S(start)  =|>
              State':= 2 /\ ID' := new() /\ PW' := new() /\ BIO'
:= new() /\ RU' := new() /\ NA' := new() /\ RI' := Rep(BIO')
    /\ AI' := Gen(RI') /\ HPW' := H(PW') /\ XI' :=
xor(RU'.Pec) /\ NI' := xor(HPW'.Xo) /\
    YI' := xor(ID'.NI') /\ IDE' := xor(YI'.HPW') /\ SND_G(IDE',
XI', AI', HPW', NA')
              2. State = 2 /\ RCV_S({NA}_SK') =|>
  State':= 4 /\ witness(U,S,sensor_user_sk,  SK')
                            end role
```

Figure 3: Role specification of user in HLPSL

Once the basic roles have been defined, we need to define a composed role and session role (Fig. 6) to integrate them so that several roles can be executed together. Lastly, the environment role is defined in Fig. 7, which contains "intruder knowledge" and "goal section."

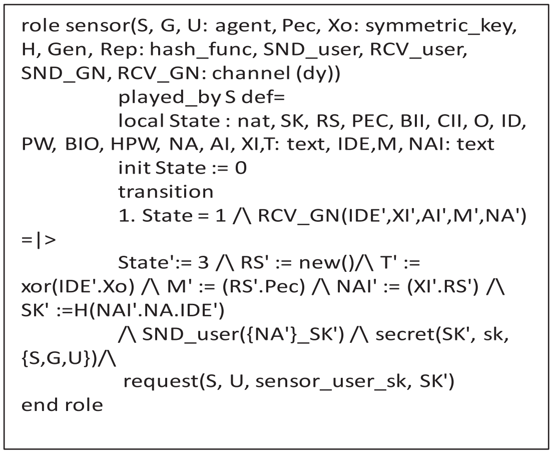
The results of the AVISPA analysis, using on-the-fly model-checker (OFMC) and attack search (AtSeE) backends to ensure the security of the proposed protocol, are shown in Figs. 8 and 9. To estimate its security against a replay attack, the OFMC checks whether a legitimate entity can execute the protocol by searching for a passive adversary. Moreover, the OFMC checks whether the proposed protocol is secure against the man-in-the-middle attack using the Dolev–Yao model.

The OFMC backend takes 0.04 s to visit eight nodes. The replay attack and Dolev–Yao model checks were performed successfully, showing that the proposed protocol is safe against replay and man-in-the-middle attacks. Figs. 7–9 show the goals section and simulation results.



```
role gateway(G, U, S: agent, Pec, Xo: symmetric_key,
H, Gen, Rep: hash_func, SND_U, RCV_U, SND_SN,
RCV_SN: channel (dy))
            played_by G def=
            local State : nat, O, BIO, XI, T,ID, PW, NA,
HPW, AI: text, IDE, M: text
            init State := 0
            transition
            1. State = 1 /\ RCV_U(IDE', XI', AI', HPW',
NA') =|>
            State':= 3 /\ PW' := new() /\ T' :=
xor(IDE'.Xo) /\ O' := xor(HPW'.AI') /\ M' := xor(T'.O') /\
            SND_SN(IDE',XI',AI',M',NA')
 role session(U, G, S: agent, Pec:
 symmetric_key, Xo: symmetric_key, H, Gen,
```

Figure 4: Role specification of gateway in HLPSL



```
role sensor(S, G, U: agent, Pec, Xo: symmetric_key,
H, Gen, Rep: hash_func, SND_user, RCV_user,
SND_GN, RCV_GN: channel (dy))
            played_by S def=
            local State : nat, SK, RS, PEC, BII, CII, O, ID,
PW, BIO, HPW, NA, AI, XI,T: text, IDE,M, NAI: text
            init State := 0
            transition
            1. State = 1 /\ RCV_GN(IDE',XI',AI',M',NA')
=|>
            State':= 3 /\ RS' := new()/\ T' :=
xor(IDE'.Xo) /\ M' := (RS'.Pec) /\ NAI' := (XI'.RS') /\
SK' :=H(NAI'.NA.IDE')
            /\ SND_user({NA'}_SK') /\ secret(SK', sk,
{S,G,U})/\
             request(S, U, sensor_user_sk, SK')
end role
```

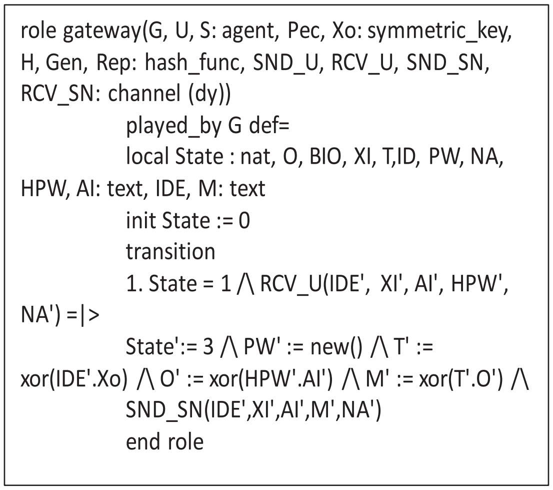
Figure 5: Role specification of sensor node

5.3 Informal Analysis

This section presents the security analysis of COBBAS with a focus on the shortcomings of previous authentication mechanisms, i.e., user anonymity, integrity, and biometric recognition error. It also provides an in-depth analysis of how the proposed scheme is resilient against various security attacks.

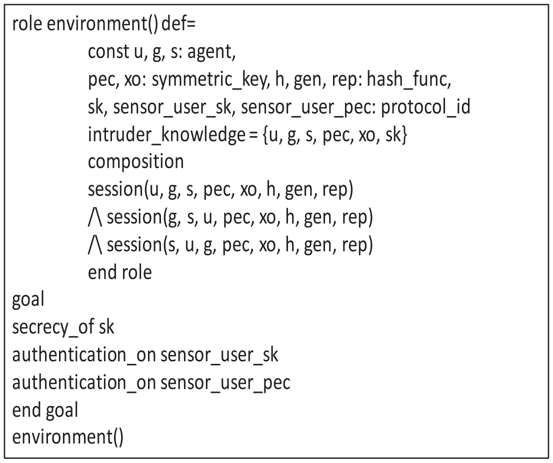
5.3.1 User Anonymity

The proposed scheme ensures the user's anonymity because of the shared secret x_o. The user calculates a variable Y_i, which is XOR with ID_i and N_i. Here, N_i was shared by the gateway node during the registration phase. Then, the user calculates $ID_i^{*}= Y_i HPW*$ and sends this result to the gateway node. The use of previously shared secrets and values confirms that the user is anonymous.



```
role gateway(G, U, S: agent, Pec, Xo: symmetric_key,
H, Gen, Rep: hash_func, SND_U, RCV_U, SND_SN,
RCV_SN: channel (dy))
            played_by G def=
            local State : nat, O, BIO, XI, T,ID, PW, NA,
HPW, AI: text, IDE, M: text
            init State := 0
            transition
            1. State = 1 /\ RCV_U(IDE',  XI', AI', HPW',
NA') =|>
            State':= 3 /\ PW' := new() /\ T' :=
xor(IDE'.Xo) /\ O' := xor(HPW'.AI') /\ M' := xor(T'.O') /\
            SND_SN(IDE',XI',AI',M',NA')
            end role
```

Figure 6: Role session



```
role environment() def=
            const u, g, s: agent,
            pec, xo: symmetric_key, h, gen, rep: hash_func,
            sk, sensor_user_sk, sensor_user_pec: protocol_id
            intruder_knowledge = {u, g, s, pec, xo, sk}
            composition
            session(u, g, s, pec, xo, h, gen, rep)
            /\ session(g, s, u, pec, xo, h, gen, rep)
            /\ session(s, u, g, pec, xo, h, gen, rep)
            end role
goal
secrecy_of sk
authentication_on sensor_user_sk
authentication_on sensor_user_pec
end goal
environment()
```

Figure 7: Environment and goals in HLPSL

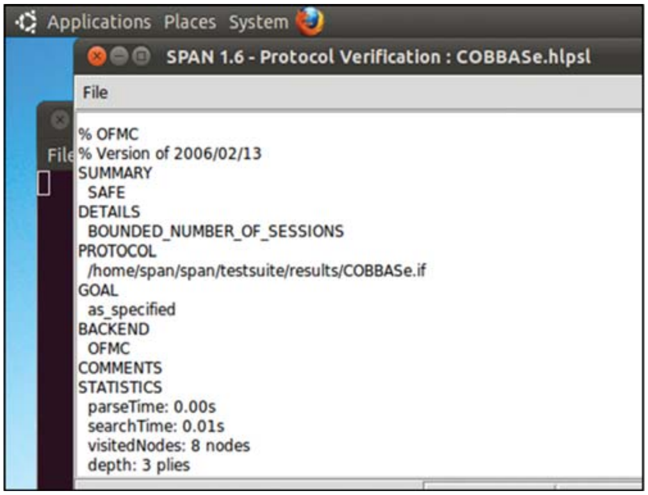
5.3.2 Replay Attack

In this scheme, the user sends N_a, a nonce value, to the gateway node. A nonce is the number generated by a node or user for one session only. These variables cannot be used in the next section. The nonce N_a sent from the user will be received by the sensor node. The sensor node encrypts N_a and sends it to the user. The user receives the values and other data with his/her generated nonce. This confirms that the message has not been replayed.

5.3.3 Biometric Recognition

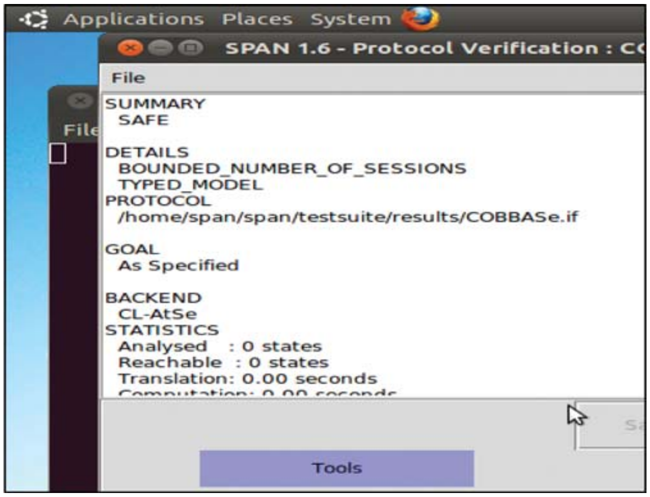
The proposed scheme avoids biometric recognition errors using a fuzzy extractor. When the user inputs his/her biometric B_i using a tablet or computer, the protocol first calculates R_i and P_{fe} using the fuzzy extractor, where R_i is a random string that represents B_i and P_{fe} is a helping string.

$\mathrm{Gen}(B_i) = (R_i, P_{fe})$


Applications Places System
SPAN 1.6 - Protocol Verification : COBBASe.hlpsl
File

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/COBBASe.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.01s
  visitedNodes: 8 nodes
  depth: 3 plies
```

Figure 8: Simulation results with OFMC


Applications Places System
SPAN 1.6 - Protocol Verification : CO
File
Tools

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/span/testsuite/results/COBBASe.if

GOAL
  As Specified

BACKEND
  CL-AtSe
STATISTICS
  Analysed   : 0 states
  Reachable  : 0 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

Figure 9: Simulation results with ATSE

When the user wants to login to the network again, the fuzzy extractor calculates R_i^* using the *Rep* algorithm, which takes B_i^* and P_{fe} as input and calculates R_i^* accordingly.

$Rep(B_i^*,\ P_{fe}) = R_i^*$

The advantage of a fuzzy extractor is that, if there is a small difference between B_i and B_i^* for the same user, the fuzzy extractor can calculate R_i via helping string P_{fe}.

5.3.4 Integrity

The integrity of a scheme is established if an adversary cannot alter the contents of a transmitted message. In this scheme, integrity is ensured using hash functions. The ID_i of the user is sent as $ID_i^*= Y_iHPW*$, where Y_i is calculated with the help of N_i and x_o. Similarly, the password of the user is shared by hashing the value of the password PW, which is calculated as

$HPW = h(PW)$. Moreover, the biometric imprint of the user is secured through hashing as $R_i = Gen(B_i)$ and $A_i = h(R_i)$.

5.3.5 Complexity of Equipment

Previously proposed schemes using smart cards or biometrics required special hardware. To use smart cards, a card reader is necessary. In this scheme, the user imprints his/her biometrics on his/her tablet or PC to login to the network. Hence, special hardware is not required in this scheme.

5.3.6 Insider Attack

An insider attack is launched by an adversary through an authorized system. It is difficult to identify and protect against insider attacks.

In this scheme, this type of attack is not beneficial for the attacker because of one-way hashing. All the information that is sent is calculated using a one-way hash function.

5.3.7 Password-Guessing Attack

The proposed scheme resists password-guessing attacks because the user imprints his/her personal biometrics for logging in. The password is encrypted with a one-way hash function. Even the gateway node does not know the original password. Hence, it is difficult for an adversary to obtain the original password or biometric imprint of the user.

Tab. 3 compares the proposed scheme with previous related schemes based on various security features. It clearly shows that the proposed scheme provides mutual authentication and session key establishment, and is robust to biometric recognition error.

Table 3: Comparison of security features with related schemes

Security feature	SUBBASe [3]	Wei et al. [17]	Park et al. [19]	Maurya et al. [20]	Riaz et al. [23]	Shin et al. [24]	COBBAS
User anonymity	×	√	√	√	√	√	√
Insider attack	×	√	√	√	√	√	√
Replay attack	√	×	×	√	√	√	√
Password guessing	√	√	√	√	√	√	√
Impersonation attack	√	√	×	√	√	√	√
Sybil attack	×	√	√	√	×	√	√
Mutual authentication	√	√	√	√	√	√	√
Session key establishment	√	√	√	√	√	√	√
Biometric recognition error	×	×	√	√	√	√	√
Extra hardware needed	√	×	×	×	√	√	√

5.4 Performance Analysis of COBBAS

MICAz motes were used to evaluate the time utilization and energy consumption of the COBBAS protocol on sensor nodes.

The results are then compared with the related schemes proposed by Wei et al. [17], Maurya et al. [20], Shin et al. [24], Park et al. [19], SUBBASe [3], and Riaz et al. [23]. The time and energy consumed by the related schemes were also calculated for the MICAz motes. The MICAz mote is constructed using second-and third-generation sensor node technology by Crossbow Technology

USA [37]. MICAz motes can measure biometric pressure and seismic waves, and are equipped with humidity, light, and temperature sensors [38].

The current I on the MICAz mote is 8 mA, and its voltage is 3 V [20]. The total energy required for elliptic-curve Diffie-Hellman (ECDH) key exchange is 57 mJ [3]. Therefore, the time required for key exchange can be calculated by:

$$E = V \times I \times t \tag{16}$$

$$57 = 3 \times 8 \times \mathrm{t}$$

$$\mathrm{t} = 2.375 \text{ ms (i.e., } T_{ECDH}\text{)}$$

where E is the energy consumed, V is the voltage of the node, I is the current in mA, and t is the time required for key exchange. Therefore, the time required for one *ECDH* key exchange (T_{ECDH}) on the MICAz mote is 2.375 ms. In addition, the computational cost for the fuzzy extractor is lesser than the cost of hashing [17]. Therefore, for simplicity, we assumed the same values for both operations. The time for the one-way hash function (T_H) was 3.636 [24]. Here, T_{RC5} denotes the time required to perform one *RC5* encryption or decryption on the MICAz mote. On the MICAz mote, execution time for one RC5 encryption or decryption was 0.26 ms [3]. Moreover, the time for symmetric key encryption/decryption cost is one hash function [3]. Therefore, we assumed that the time for symmetric key encryption decryption (T_{sym}) on the MICAz mote was 3.636 ms. The computational times on the MICAz mote for different cryptographic operations is given in Tab. 4

Table 4: Execution time and energy consumption on sensor node

Quantity	Description	Values
I (Current)	Current I on MICAz mote	8 mA
EC_{DH} Energy	Amount of energy required to perform ECDH key exchange	57 mj
T_{RC5}	Time required to encrypt decrypt a message using RC5	0.26 ms
T_H	Time on one-way hash function	3.636 ms
V (Voltage)	Voltage of MICAz mote	3.0 V
T_p	Time for ECC point multiplication	114 ms
T_{sym}	Time for symmetric encryption decryption	3.636 ms
T_{ECDH}	Time for ECDH key exchange	2.375 ms

According to [24], the execution time of a one-way hash function (T'_H) on a PC is 2.58 μs. The time required for ECC point multiplication (T'_P) on a PC was 1.226 ms, while the time for ECC point multiplication (T_P) on the MICAz sensor was 114 ms. The time required for symmetric key encryption decryption (T'_{sym}) on a PC was 8.7 ms [20]. The computational time on PC for different cryptographic operations, as considered in, [20,24] is given in Tab. 5.

5.4.1 Time Analysis

In this section, the proposed scheme and existing schemes are compared in terms of time consumption. We have compared only the login and authentication phases of COBBAS with previous schemes because registration and password updates are not used frequently. Tab. 6 summarizes the time analysis.

Table 5: Execution time on PC/device

Terms	Description	Time on PC/device
T'_H	Time for one-way hash function on PC/device	2.58 μs
T'_p	Time for ECC point multiplication	1.226 ms
T'_{sym}	Time for symmetric key encryption decryption	8.7 ms

Table 6: Comparison of time consumption with related schemes

Entity	SUBBASe [3]	Wei et al. [17]	Park et al. [19]	Maurya et al. [20]	Riaz et al. [23]	Shin et al. [24]	COBBAS
User	$T'_H + T'_{sym}$	$4T'_H + T'_{sym}$	$2T'_P + 10T'_H$	$4T'_H + 2T'_p + 2T'_{sym}$	$4T'_H + 2T'_p + T_{RC5}$	$14T'_H$	$4T'_H + 2T'_P + T_{ECDH}$
Gateway	$2T'_H$	$5T'_H + 2T'_{sym}$	$11\ T'_H$	$T'_H + 2T'_p + 2T'_{sym}$	T'_H	$15\ T'_H$	$T'_H + 2T'_P$
Sensor	$3T_H + T_{sym}$	$2T_H + T_{sym}$	$2T_P + 4T_H$	T_{sym}	$T_H + T_{ECDH} + 2T_p$	$6T_H$	$T_H + T_{ECDH}$
Total cost	$3T'_H + 3T + T_{sym} + T'_{sym}$	$9T'_H + 2T_H + 3T'_{sym} + T_{sym}$	$21T'_H + 4T_H + 2T_P + 2T'_P$	$5T'_H + 4T'_p + 4T'_{Sym} + T_{Sym}$	$5T'_H + T_{ECDH} + T_{RC5} + 2T_p + 2T'_p + T_H$	$29T'_H + 6T_H$	$5T'_H + 4T'_P + 2T_{ECDH} + T_H$
Running time (ms)	22.4	37.0	245.1	43.4	236.7	21.9	13.3

Table 7: Comparison of energy consumption with related schemes

Protocol	Total energy consumption (mJ)	Total energy consumption (J)
SUBBASe [3]	E = 3000 mV × 8 mA ×22.4 = 537600 mJ	537.6
Wei et al. [17]	E = 3000 mV × 8 mA × 37.0 = 888000 mJ	888
Park et al. [19]	E = 3000 mV × 8 mA × 245.1 = 5882400 mJ	5882.4
Maurya et al. [20]	E = 3000 mV × 8 mA × 43.4 = 1041600 mJ	1041.6
Riaz et al. [23]	E = 3000 mV × 8 mA × 236.7 = 5680800 mJ	5680.8
Shin et al. [24]	E = 3000 mV × 8 mA × 21.9 = 525600 mJ	525.6
COBBAS	E = 3000 mV × 8 mA × 13.3 = 319200 mJ	319.2

Tab. 6 shows that the times required for Maurya et al. [20] scheme, Wei et al. [17] scheme, Park et al. [19] scheme, Shin et al. [24] scheme, Riaz et al. [23] scheme, and SUBBASe [3] were 43.4, 37.0, 245.1, 21.9, 236.7, and 22.4 ms, respectively. The time required for the COBBAS scheme was 13.3 ms. The authentication time of the proposed scheme is much faster than that of the current schemes. Moreover, this comparison shows that the proposed scheme outperforms and provides a higher level of security, even with light computation.

5.4.2 Energy Analysis

The authentication process is completed by exchanging several messages among the entities involved in the network. During this process, energy is consumed by the sensor node. This section compares the energy consumption of the proposed scheme with related schemes. We measured the energy consumed by the proposed protocol and related schemes on the MICAz mote. The energy is calculated with the help of Eq. 16. The energy consumed by each protocol for one node's authentication is given in Tab. 7. This table shows the energy consumed by SUBBASe [3], Wei et al. [17], Park et al. [19] Maurya et al. [20], Riaz et al. [23] and Shin et al. [24] schemes are 537.6, 888, 5882.4, 1041.6, 5680.8, and 525.6 J, respectively. The energy consumed by the proposed method was only 319.2 J, which shows that the proposed scheme is efficient compared to related schemes in terms of energy input.

6 Conclusions

In this study, we analyzed various studies related to authentication mechanisms in recent years. To overcome the flaws in previous schemes, we proposed an efficient authentication scheme, comprising only two phases using simple and lightweight computations. The COBBAS scheme protects WSNs from different types of attacks, and provides user anonymity along with biometric error recovery. The mutual authentication of the proposed scheme was proved using BAN logic. In addition, AVISPA analysis proved that the proposed scheme is safe from intruder-based interventions. Furthermore, an informal security analysis showed that COBBAS provides better security than previous schemes with reasonable resource utilization. Additionally, its computational cost and energy consumption are believed to be suitable for resource-constrained networks. Moreover, the proposed scheme is energy efficient, and provides a higher level of security than related proposed schemes.

Funding Statement: This project was funded by the National Research Foundation of Korea. Grant Number: 2020R1A2C1012196.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Statista Research Department, "Global IoT end-user spending worldwide 2017–2025," January, 2021. [Online]. Available: https://www.statista. com/statistics/976313/global-iot-market-size/.

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

[3] R. Riaz, S. S. Rizvi, S. Shokat, S. J. Kwon and N. A. Gillani, "SUBBASE: An authentication scheme for wireless sensor network based on user biometric," *Wireless Communication and Mobile Computing*, vol. 2019, no. 11, pp. 1–11, 2019.

[4] R. Watro, D. Kong, S. F. Cuti, C. Gardiner, C. Lynn *et al.,* "TinyPK: Securing sensor networks with public key technology," in *Proc. of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks*, New York, USA, 2004.

[5] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Transaction on Wireless Communication*, vol. 3, no. 3, pp. 1086–1090, 2009.

[6] K. H. Wong, Y. Zheng, J. Cao and S. Wang, "A dynamic user authentication scheme for wireless sensor networks," in *IEEE Int. Conf. on Sensor Networks, Ubiquitous, and Trustworthy Computing*, Taichung, 2006.

[7] H. R. Tseng, R. H. Jan and W. Yang, "A robust password-based authentication scheme for heterogeneous sensor networks," *Communication of IICM Taiwan*, vol. 11, no. 3, pp. 1–13, 2008.

[8] T. Chen and W. Shih, "A robust mutual authentication protocol for wireless sensor networks," *Electronics and Telecommunications Research Institute*, vol. 32, no. 5, pp. 704–712, 2010.

[9] T. H. Lee, "Simple dynamic user authentication protocols for wireless sensor network," in *Second Int. Conf. on Sensor Technologies and Applications*, Cap Esterel, 2008.

[10] M. K. Khan and K. Alghathbar, "Cryptanalysis and security improvements of two factor user authentication in wireless sensor network," *Sensors*, vol. 10, no. 3, pp. 2450–2459, 2010.

[11] S. G. Yoo, K. Y. Park and J. Kim, "A security-performance-balanced user authentication scheme for wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 8, no. 3, pp. 382810, 2012.

[12] P. Kumar, A. Gurtov, M. Ylianttila, S. Lee and H. Lee, "A strong authentication with user privacy for wireless sensor networks," *Electronics and Telecommunications Research Institute*, vol. 35, no. 5, pp. 889–899, 2013.

[13] S. Yu, J. Lee, K. Lee, K. Park and Y. Park, "Secure authentication protocol for wireless sensor networks in vehicular communication," *Sensors*, vol. 18, no. 10, pp. 3191, 2018.

[14] J. Yuan, C. Jiang and J. Zuowen, "A biometric based user authentication for wireless sensor networks," *Wuhan University Journal of Natural Sciences*, vol. 15, no. 3, pp. 272–276, 2010.

[15] J. J. Yuan, "An enhanced two-factor user authentication in wireless sensor network," *Telecommunication Systems*, vol. 55, no. 1, pp. 105–113, 2014.

[16] A. K. Das, "A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 9, no. 1, pp. 223–244, 2016.

[17] F. Wei, J. Ma, Q. Jiang, J. Shen and C. Ma, "Cryptanalysis and improvement of an enhanced two-factor user authentication scheme in wireless sensor network," *Information Technology and Control*, vol. 45, no. 1, pp. 62–70, 2016.

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

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

[20] A. K. Maurya and V. Sastry, "Fuzzy extractor and elliptic curve based efficient user authentication protocol for wireless sensor networks and internet of things," *Information-An International Interdisciplinary Journal*, vol. 8, no. 4, pp. 136, 2017.

[21] D. Kang, J. Jung, H. Kim, Y. Lee and D. Won, "Efficient and secure biometric based user authenticated agreement scheme with anonymity," *Security and Communication Network*, vol. 2018, no. 12, pp. 1–14, 2018.

[22] R. Riaz, T. S. Chung, S. S. Rizvi and N. Yaqub, "BAS: The biphase authentication scheme for wireless sensor network," *Security and Communication Networks*, vol. 2017, no. 3, pp. 1–10, 2017.

[23] R. Riaz, S. S. Rizvi, M. Shaheen and S. J. Kwon, "Enhanced biometric based user authentication scheme for wireless sensor networks using fuzzy extractor and elliptic curve cryptography," *Journal of Information Communication and Technologies*, vol. 2016, pp. 1–14, 2020.

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

[25] H. Lee, D. Kanga, J. Ryu, D. Wo, H. Kim *et al.,* "A three-factor anonymous user authentication scheme for internet of things environments," *Journal of Information Security and Applications*, vol. 52, pp. 102494, 2020.

[26] S. Shin and T. Kwon, "A privacy-preserving authentication, authorization, and key agreement scheme for wireless sensor networks in 5g-integrated internet of things," *IEEE Access*, vol. 8, pp. 67555–67571, 2020.

[27] M. Wazid, A. K. Das, S. Shetty, J. J. Rodrigues and Y. Park, "LDAKM-EIoT: Lightweight device authentication and key management mechanism for edge-based iot deployment," *Sensors*, vol. 19, no. 24, pp. 5539, 2019.

[28] B. D. Deebak, F. Al-Turjman, M. Aloqaily and O. Alfandi, "An authentic-based privacy preservation protocol for smart e-healthcare systems in IoT," *IEEE Access*, vol. 7, pp. 99, 2019.

[29] Z. Ali, A. Ghani, I. Khan, S. A. Chaudhry, S. H. Islam *et al.,* "A robust authentication and access control protocol for securing wireless healthcare sensor networks," *Journal of Information Security and Applications*, vol. 52, no. 4, pp. 102502, 2020.

[30] N. M. F. Qureshi, I. F. Siddiqui, A. Abbas, A. K. Bashir, C. S. Nam *et al.,* “Streambased authentication strategy using iot sensor data,” *Wireless Personal Communications*, vol. 116, pp. 1217–1229, 2020.

[31] A. Ali, H. Rafique, T. Arshad, M. A. Alqarni, S. H. Chauhdary *et al.,* “A fractal-based authentication technique using sierpinski triangles in smart devices,” *Sensors*, vol. 19, no. 3, pp. 678, 2019.

[32] M. Sohail, S. Khan, R. Ahmad, D. Singh and J. Lloret, “Game theoretic solution for power management in iot-based wireless sensor networks,” *Sensors*, vol. 19, no. 18, pp. 3835, 2020.

[33] D. Singh, G. U. Rehman, A. Ghani, M. Zubair and M. I. Saeed, “SOS: Socially omitting selfishness in IoT for smart and connected communities,” *International Journal of Communication System*, pp. e4455, 2020.

[34] D. Singh, M. Kaur, G. Kaur, P. K. Sharma and A. Jolfaei, “Binary cuckoo search metaheuristic-based supercomputing framework for human behavior analysis in smart home,” *Journal of Supercomputing*, vol. 76, no. 4, pp. 2479–2502, 2020.

[35] Y. Zhou, L. Yu, S. Pan and Z. Wang, “BAN logic analysis method based on yahalom protocol,” *Revista de la Facultad de Ingenieria*, vol. 31, pp. 134–142, 2016.

[36] The team AVISPA, “A Beginner’s Guide to Modelling and Analysing Internet Security Protocols,” Retrieved May 15, 2019, [Online]. Available: http://www.avispaproject.org/package/tutorial.pdf.

[37] M. Johnson, M. Healy, P. v. Ven, M. J. Hayes, J. Nelson *et al.,* “A comparative review of wireless sensor network mote technologies,” in *IEEE SENSORS, 2009 Conf.*, Canterbury 8013, New Zealand, 2009.

[38] R. P. Narayanan, T. V. Sarath and V. V. Vineeth, “Survey on motes used in wireless sensor networks: Performance & parametric analysis,” *Wireless Sensor Network*, vol. 8, no. 4, pp. 51–60, 2016.