Computer Systems Science & Engineering
DOI:10.32604/csse.2022.022962
Article

Tech Science Press

Secure and Anonymous Three-Factor Authentication Scheme for Remote Healthcare Systems

Munayfah Alanazi* **and Shadi Nashwan**

Department of Computer Science, College of Computer and Information Sciences, Jouf University, Sakaka, 42421, Saudi Arabia
*Corresponding Author: Munayfah Alanazi. Email: 401205995@ju.edu.sa
Received: 24 August 2021; Accepted: 25 September 2021

Abstract: Wireless medical sensor networks (WMSNs) play a significant role in increasing the availability of remote healthcare systems. The vital and physiological data of the patient can be collected using the WMSN via sensor nodes that are placed on his/her body and then transmitted remotely to a healthcare professional for proper diagnosis. The protection of the patient's privacy and their data from unauthorized access is a major concern in such systems. Therefore, an authentication scheme with a high level of security is one of the most effective mechanisms by which to address these security concerns. Many authentication schemes for remote patient monitoring have been proposed recently. However, the majority of these schemes are extremely vulnerable to attacks and are unsuitable for practical use. This paper proposes a secure three-factor authentication scheme for a patient-monitoring healthcare system that operates remotely using a WMSN. The proposed authentication scheme is formally verified using the Burrows, Abadi and Needham's (BAN) logic model and an automatic cryptographic protocol verifier (ProVerif) tool. We show that our authentication scheme can prevent relevant types of security breaches in a practical context according to the discussed possible attack scenarios. Comparisons of the security and performance are carried out with recently proposed authentication schemes. The results of the analysis show that the proposed authentication scheme is secure and practical for use, with reasonable storage space, computation, and communication efficiency.

Keywords: Mutual authentication; biometric feature; perfect forward secrecy; user anonymity; proVerif tool; BAN logic model

1 Introduction

Wireless medical sensor networks (WMSNs) represent an important trend that has emerged recently to enhance the quality of healthcare services. The vital signs (e.g., blood pressure, blood sugar, etc.) can be obtained via sensor nodes placed on the patient's body, and they are transmitted via the WMSN to the monitoring device of a healthcare professional, enabling them to keep track of the patient's health [1]. In general, remote healthcare systems using WMSNs can not only monitor the health of patients in real time but also save time and money. In the same context, such healthcare systems increase the productivity of medical professionals, enable a reduction in healthcare locations, compensate for the lack of healthcare in

remote locations, and provide immediate and continuous health advice to communities, particularly in an emergency—their benefits have been demonstrated during the current COVID-19 pandemic [2,3].

The main elements of the healthcare system, as shown in Fig. 1, are healthcare professionals, medical sensors, and a gateway node (GWN). The medical sensors are placed on the patient's body to collect the patient's physiological data and relay them to the GWN over the WMSN with minimal computational resources. The GWN is a trusted node which represents the provider of the healthcare service and has adequate computational resources to serve as a link between sensors and healthcare professionals [4–6].

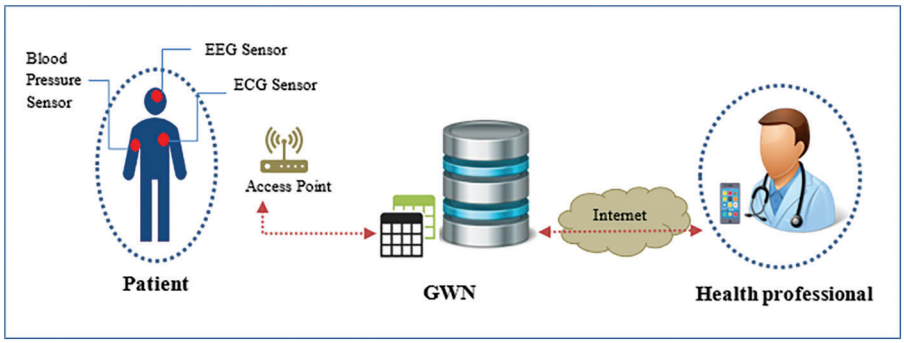

Figure 1: The healthcare monitoring architecture using WMSN

The essential challenges in the implementation and use of a WMSN are associated with the patient's privacy and the credibility of the received medical instructions [3]. Due to the open nature of wireless networks, unauthorized parties can access, modify, and forward the transmitted messages to deliver incorrect instructions or advice to patients [7,8]. It is particularly dangerous if the unauthorized party is able to instruct the patient to disable the wearable sensor devices, such as heart pumps [9]. Moreover, unauthorized access to the sensitive data that have been collected by the sensor nodes can lead to a loss of employment or government health benefits for the patient, as well as inaccurate or fabricated medical records [3]. Furthermore, other types of attacks can be carried out due to the limited capabilities of the sensor nodes, such as smartcard loss, removing the anonymity of the healthcare professionals or patients, and man-in-the-middle, impersonation, insider, desynchronization, and replay attacks [3–4,10–15]. Therefore, the primary concern when implementing a healthcare system is ensuring the confidentiality, availability, and integrity of the services in order to protect the patients' privacy and the data that are transmitted between the different elements of the system [16,17]. Thus, an authentication scheme is considered the most effective method to achieve a high level of security in such systems.

Several authentication schemes have been proposed to provide a high level of security for healthcare systems using WMSNs. In 2015, He et al. [18] proposed a new two-factor authentication scheme for healthcare systems using WMSNs. They claimed that their scheme was secure against well-known attacks. However, Wu et al. [19] found that this scheme was vulnerable to different types of attacks, such as off-line estimation, user impersonation, and sensor node capture attacks. In 2017, an improved anonymous two-factor authentication protocol for healthcare applications with WMSNs was presented by Wu et al. [19], and they claimed that their improved authentication scheme was secure. Later, Srinivas et al. [20] indicated that the scheme proposed in [19] was vulnerable to smartcard theft and insider and user impersonation attacks. In 2018, a new two-factor authentication scheme for WMSNs was proposed by Amin et al. [21]. They claimed that their protocol could protect against existing well-known attacks. In 2019, Shuai et al. [9] noted that the authentication schemes proposed by Wu et al. [19] and Ali et al. [22] could not protect against a desynchronization attack or achieve a perfect forward secrecy feature. Therefore, they suggested a three-factor authentication scheme for remote patient observation using

sensor wireless networks. They claimed that their suggested scheme was lightweight and secure and could resolve the above-mentioned security concerns. In 2020, Fotouhi et al. [23] demonstrated that the authentication scheme that was proposed by Srinivas et al. [20] was unable to prevent an offline estimation attack, unable to achieve sensor anonymity with untraceability, and failed to provide forward secrecy services. Moreover, they also reported that the authentication schemes that were proposed in [19] and [21] were unable to ensure sensor anonymity, untraceability, or provide perfect forward secrecy services. Thus, they proposed a lightweight, secure two-factor authentication scheme for healthcare monitoring systems in order to prevent the mentioned attacks. In 2021, Nashwan [3] noted that the authentication schemes that were proposed by Fotouhi et al. [23] and Shuai et al. [9] could not support full mutual authentication or sensor node anonymity services, nor could it protect against a sensor node impersonation attack. Nashwan [3] proposed an authentication scheme for healthcare IoT systems using WMSNs to resolve the mentioned security concerns and to support a high level of security in such systems.

As mentioned previously, the authentication scheme is an essential strategy in preventing the current well-known attacks in remote healthcare systems. In this paper, we have designed a secure three-factor authentication scheme for healthcare systems using a WMSN to ensure a high level of security with reasonable computational and communication efficiency. The mutual authentication between the elements of the system has been verified using Burrows, Abadi and Needham's (BAN) logic mode. In addition, we have proven that the proposed authentication scheme is safe against various popular attacks using an automatic cryptographic protocol verifier (ProVerif) tool. The success of the proposed authentication scheme has been discussed in the context of different attack scenarios based on a comparison with other recently proposed authentication schemes. The results of the comparison illustrate that our authentication scheme is practical to use, with credible computation and communication efficiency.

The rest of this paper is presented as follows: our authentication scheme is presented in Section 2. The first part of section 3 discusses the formal verification of the proposed authentication scheme using BAN logic and the ProVerif tool. An informal security analysis of the proposed authentication scheme is performed in the second part of section 3. Section 4 presents the performance evaluation in terms of the computation, communication, and storage costs. Finally, we present our conclusions in Section 5.

2 Proposed Authentication Scheme

This section presents our proposed authentication scheme, which is a secure three-factor authentication scheme. The proposed authentication scheme includes four stages, namely healthcare professional registration, medical sensor node registration, login authentication and key agreement, and the password update stages. Moreover, there are three types of elements in our authentication scheme, namely the healthcare professional (Ui), GWN, and medical sensor node (SNj). In addition, the proposed authentication scheme is based on a symmetric cryptographic technique and a collection of one-way hash functions to achieve the desired security services. Furthermore, the fuzzy extractor function is used to randomly convert the biometric data of the healthcare professional into string values. The definition of the abbreviations that have been used in relation to the proposed authentication scheme throughout the next sections is listed in Tab. 1.

2.1 Healthcare Professional Registration Stage

The healthcare professional registration stage is depicted in Fig. 2. During this stage, the healthcare professional (Ui) becomes a legal user by completing the following steps with the service provider (GWN).

Step 1: The Ui selects his/her own identity (IDi) and password (PWi) and imprints his/her personal biometrics (BIOi) using an extraction generation function as < Fi, and Pi > = Gen (BIOi). After this,

Ui calculates the BPWi = h1 (Fi), Vi = h3 (IDi || PWi || BPWi) and sends the M1: {IDi and Vi} to GWN as a registration request message using a reliable communication channel.

Step 2: After receiving M1: {IDi, and Vi} from the Ui, GWN checks whether the (IDi) has already been registered. If true, the GWN sends a denial notification message and requests that the Ui select another IDi. Otherwise, the GWN initiates sequence numbers as SSi0 = SSi1 = 0, computes SNi = h1 (SSi0), generates a pseudo-identity TIDi = h2 (IDi || SNi), and initiates temporally identity TIDi* = ϕ. Moreover, it computes the KGWN−U = h2 (IDi || XGWN), Di = KGWN−U ⊕ Vi, and Ci = h3 (IDi || Vi || KGWN−U), wherein the XGWN represents the GWN's secret key. After this, the GWN stores the Di, h1 (Ci), and SSi1 within a new smartcard (SC), transmits the SC to the Ui in a safe manner, and stores the values of the IDi, SSi0, TIDi, and TIDi* in the database of the healthcare service.

Step 3: Upon receiving the SC from GWN, the Ui completes the registration process by storing the Rep (.) and Pi.

Table 1: Definition of abbreviations

Abbreviation	Definition
GWN	A trusted gateway.
Ui	Healthcare professional.
IDi	The identity of Ui.
PWi	The password of Ui.
BIOi	Biometric information of Ui.
TIDi*	Temporally identity of Ui.
Fi, Pi	Random string values of the biometric of Ui.
XGWN	Secret key of GWN.
SNj	Sensor node.
IDSNj	Identity of sensor node.
SNj, SNi	Session identities of sensor node and healthcare professional.
SC	Smartcard of Ui.
SSj0, SSj1	Initial sequential number for GWN and sensor node.
SSi0, SSi1	Initial sequential number for GWN and healthcare professional.
r1, r2, r3	Random numbers.
SJj	Session key.
Gen (.)	Fuzzy extractor generation function.
Rep (.)	Fuzzy extractor reproduction function.
Ekey (.)	Encryption function using shared key (key).
Dkey (.)	Decryption function using shared key (key).
h1, h2, h3	Hash functions.
||	Concatenation function.
⊕	XOR function.
ϕ	Null value.

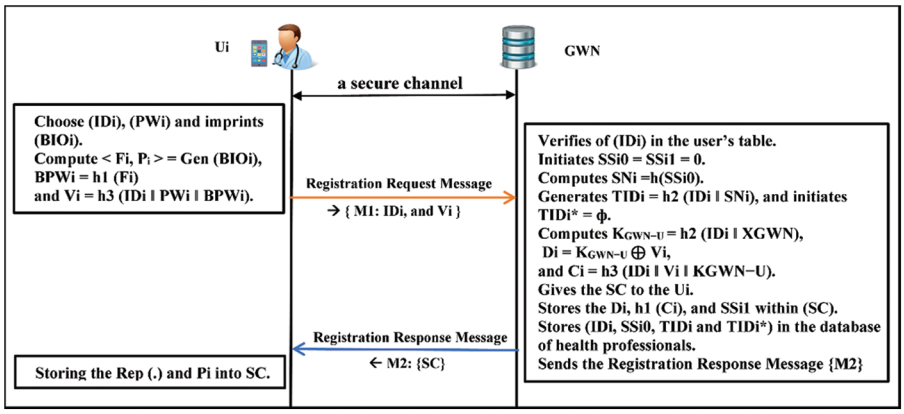

Figure 2: Healthcare professional registration stage

2.2 Sensor Node Registration Stage

Fig. 3 shows the sensor node registration stage. When a new sensor node (SNj) is activated to read the patient's physiological data and receive medical instructions from the Ui, the identification data of SNj should be registered in the GWN according to the following steps:

Step 1: SNj sends a registration request message to GWN as M1: {IDSNj} over a reliable communication channel; the identity of SNj (IDSNj) was assigned to the sensor when it was developed.

Step 2: After receiving the registration request from SNj, the GWN generates an authentication session number SNj0 = (r1) randomly, sets the sensor sequence numbers as SSj0 = SSj1 = 0, inserts the SNj node's data into the sensor node database as [IDSNj, SSj0, and SNj0], and sends a response registration message M2: {SSj1 and SNj0} to SNj securely.

Step 3: Upon receiving M2 from GWN, the SNj stores the SSj1 and SNj0 parameters in its memory.

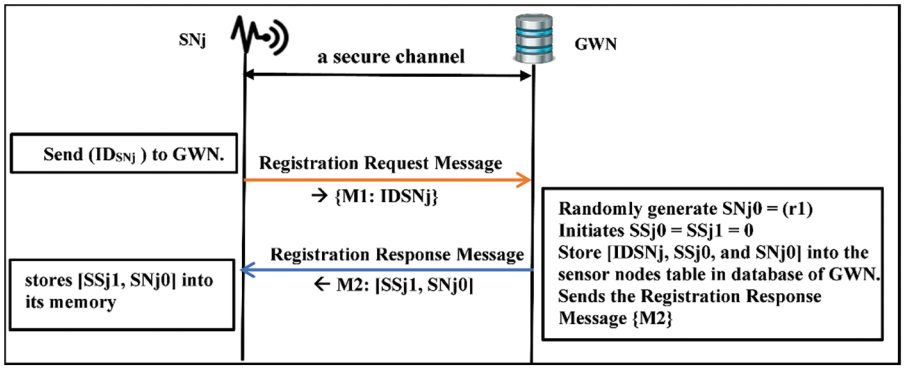

Figure 3: Sensor node registration stage

2.3 Login Authentication and Key Agreement Stage

Figs. 4a–4b shows the login and authentication and key agreement stage. During this stage, the Ui, GWN, and SNj can achieve mutual authentication and exchange the shared key between them. Therefore, after completing this stage, the SNj will enable the Ui to obtain the patient's vital signs through the GWN. The execution steps can be summarized as follows:

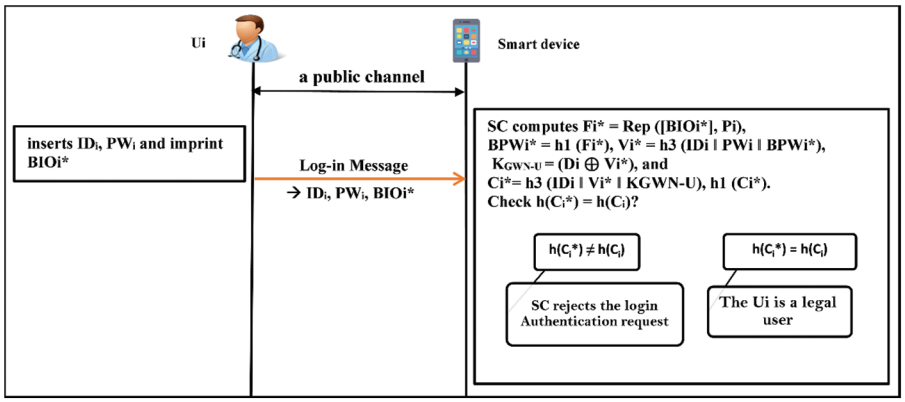

Figure 4: continued

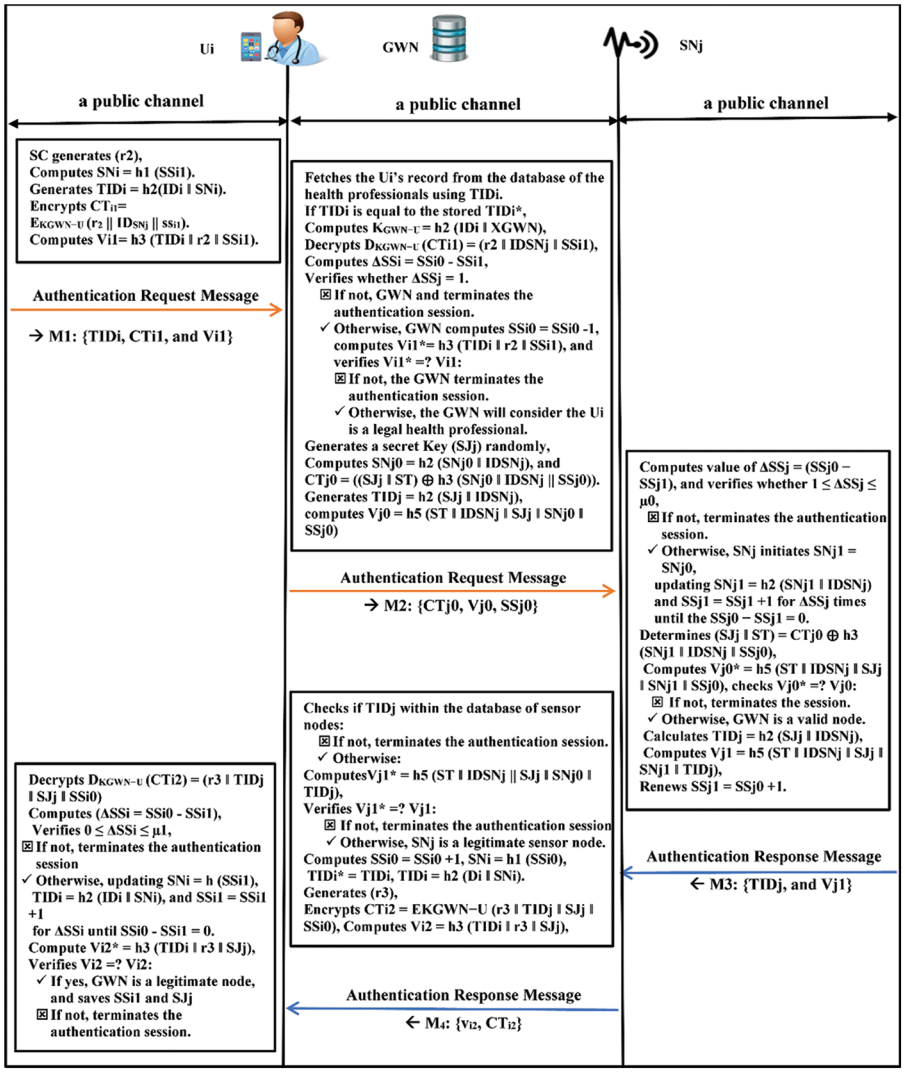

Figure 4: (a) Login process in the login authentication and key agreement stage (b) Authentication and key agreement process

Step 1: When the Ui installs the SC in his/her smart device, the Ui enters the IDi and PWi and imprints the BIOi*. Then, the SC computes Fi* = Rep ([BIOi*], Pi), BPWi* = h1 (Fi*), Vi* = h3 (IDi || PWi || BPWi*), K_{GWN-U} = (Di ⊕ Vi*), and Ci*= h3 (IDi || Vi* || KGWN-U), wherein the Rep (.) is a fuzzy extraction function.

Then, the SC checks if the value of computed h1 (Ci*) matches with h1 (Ci) that was embedded in the smartcard by GWN. If not, then it will reject the login authentication request. Otherwise, the SC will consider the IDi, PWi, and BIOi* as valid values and the Ui as a legal user.

Next, the SC generates a random value (r2), computes SNi = h1 (SSi1), generates the pseudonym identity TIDi = h2 (IDi || SNi), encrypts CTi1= E_{KGWN-U} (r2 || IDSNj || SSi1), computes Vi1= h3 (TIDi || r2 || SSi1), and sends the login authentication message M1: {TIDi, CTi1, and Vi1} to GWN via an insecure communication channel, wherein the IDSNj represents the identity of the sensor node that the Ui intended to access.

Step 2: After arriving (M1) from the Ui, the GWN fetches the Ui's record from the database of the healthcare service using the received value of the TIDi. Then, we have one of the following cases:

Case 1: If TIDi is equal to the stored TIDi, then GWN computes KGWN−U = h2 (IDi || XGWN), decrypts D_{KGWN-U} (CTi1) = (r2 || IDSNj || SSi1), computes Vi1*= h3 (TIDi || r2 || SSi1), and verifies whether Vi1* matches Vi1. If not, the authentication session will be rejected by the GWN. Otherwise, the GWN will consider the Ui as a legal healthcare professional.

Case 2: If TIDi is equal to the stored TIDi*, then GWN computes K_{GWN-U} = h2 (IDi || XGWN), decrypts D_{KGWN-U} (CTi1) = (r2 || IDSNj || SSi1), computes ΔSSi = SSi0 - SSi1, verifies whether ΔSSj = 1. If not, then both the M1 and the authentication session will be rejected by GWN. Otherwise, GWN computes SSi0 = SSi0–1, computes Vi1*= h3 (TIDi || r2 || SSi1), and verifies whether Vi1* matches Vi1. If not, the authentication session will be rejected by GWN. Otherwise, the GWN will consider the Ui as a legal healthcare professional.

Case 3: If TIDi does not exist, the GWN will consider the Ui as a legal healthcare professional and terminate the authentication session.

To achieve mutual authentication with the intended SNj, the GWN generates a secret key (SJj) randomly and computes SNj0 = h2 (SNj0 || IDSNj) and CTj0 = ((SJj || ST) ⊕ h3 (SNj0 || IDSNj || SSj0)), wherein the (ST) value determines whether the GWN needs to obtain vital signs from the SNj or forward medical instructions for SNj. Next, GWN generates the pseudonym identity for the SNj as TIDj = h2 (SJj || IDSNj), computes Vj0 = h5 (ST || IDSNj || SJj || SNj0 || SSj0), renews SSj0 = SSj0 + 1, and sends the authentication request message M2: {CTj0, Vj0, and SSj0} via a public communication channel to SNj.

Step 3: Upon arriving (M2) from GWN, the SNj computes ΔSSj = (SSj0 − SSj1) value and checks whether $1 \leq \Delta SSj \leq \mu 0$, wherein μ0 is determined according to the requirements of the system. If not, then both the M2 and the authentication session will be rejected by SNj. Otherwise, the SNj initiates SNj1 = SNj0, and it repeats the updating of the values of SNj1 = h2 (SNj1 || IDSNj) and SSj1 = SSj1 + 1 for ΔSSj times until the SSj0 − SSj1 = 0.

After this, SNj determines (SJj || ST) = CTj0 ⊕ h3 (SNj1 || IDSNj || SSj0), computes Vj0* = h5 (ST || IDSNj || SJj || SNj1 || SSj0), and checks if Vj0* equals Vj0. If not, the SNj aborts the session. Otherwise, the SNj will consider GWN as a valid node. Next, SNj calculates TIDj = h2 (SJj || IDSNj), computes Vj1 = h5 (ST || IDSNj || SJj || SNj1 || TIDj), renews SSj1 = SSj0 + 1, and sends the login authentication response message M3: {TIDj, and Vj1} to GWN via a public communication channel.

Step 4: Upon arriving (M3) from SNj, GWN checks if TIDj is within the database of sensor nodes. If not, then GWN refuses M3 and aborts the authentication session. Otherwise, GWN computes Vj1* = h5

(ST || IDSNj || SJj || SNj0 || TIDj) and then verifies whether Vj1* matches Vj1. If not, then GWN refuses the M3 and aborts the authentication session. Otherwise, the GWN will consider the SNj as a legitimate sensor node.

Next, GWN computes SSi0 = SSi0 + 1, SNi = h1 (SSi0), TIDi* = TIDi, TIDi = h2 (Di || SNi), generates (r3), encrypts CTi2 = E_{KGWN-U} (r3 || TIDj || SJj || SSi0), computes Vi2 = h3 (TIDi || r3 || SJj), and sends a login authentication response message M4: {Vi2, and CTi2} to Ui GWN via an unsafe communication channel.

Step 5: After arriving (M4) from GWN, the Ui decrypts (r3 || TIDj || SJj || SSi0) = D_{KGWN-U} (CTi2), computes (ΔSSi = SSi0−SSi1), and verifies whether $0 \leq \Delta SSi \leq \mu 1$, where μ1 is determined according to the specifications of the system. If not, then Ui refuses M4 and aborts the authentication session. Otherwise, the Ui repeats the updating of the values of SNi = h (SSi1), TIDi = h2 (IDi || SNi), and SSi1 = SSi1 + 1 for ΔSSi until SSi0−SSi1 = 0. Next, the Ui computes Vi2* = h3 (TIDi || r3 || SJj) and verifies that Vi2 matches Vi2. If so, the Ui will consider GWN as a legitimate node and save the values of SSi1 and SJj. Otherwise, Ui aborts the authentication session.

2.4 Password Change Stage

The password change during the healthcare professional stage can be accomplished between Ui and SC and is not subject to GWN's consent. Fig. 5 shows the main processes of this stage, which can be summarized as follows:

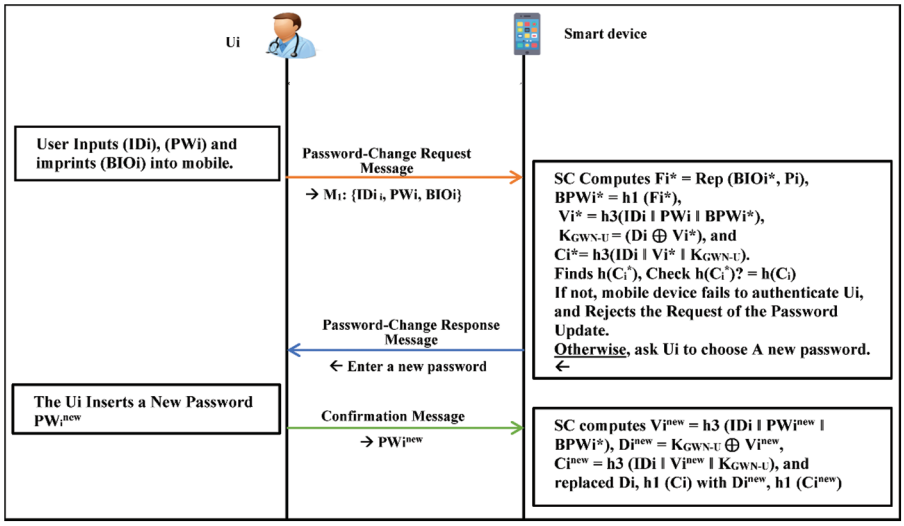

Figure 5: Password change stage

Step 1: Ui enters the IDi and old PWi and imprints the BIOi.

Step 2: SC Computes Fi* = Rep (BIOi*, Pi), BPWi* = h1 (Fi*), Vi* = h3 (IDi || PWi || BPWi*), K_{GWN-U} = (Di ⊕ Vi*), and Ci*= h3(IDi || Vi* || K_{GWN-U}). After this, the SC checks if the value of h1 (Ci*) matches with the h1 (Ci) that have been embedded within it by the GWN. If not, the password change request will be refused by the SC. Otherwise, the SC requests that the Ui insert a new password.

Step 3: Ui inserts a new password PWi^{new}.

Step 4: SC computes Vi^{new} = h3 (IDi || PWi^{new} || BPWi*), Di^{new} = K_{GWN-U} ⊕ Vi^{new}, Ci^{new} = h3 (IDi || Vi^{new} || K_{GWN-U}), and the h1 (Ci) and Di values are replaced with h1 (Ci^{new}) and Di^{new} values, respectively.

3 Security Analysis

This section verifies the security features of the proposed authentication scheme. First, a formal security analysis validates that our authentication scheme can support mutual authentication and secure authentication session features using the BAN logic model and Proverif tool. Second, an informal security analysis demonstrates that our authentication scheme provides suitable security features and can protect against related types of attacks, taking into account all possible attack scenarios. Finally, the last part of the analysis compares the security features of our authentication scheme with recently proposed, related authentication schemes.

3.1 Formal Security Analysis

The registration and password change stages are either not used frequently or are performed through a secure communication channel. Therefore, this part focuses on the soundness of the login authentication and key agreement stage.

3.1.1 Validation Using BAN Logic Model

The BAN logic model will be used to ensure that the authentication messages exchanged during the authentication and key agreement stage between the healthcare professional node (Ui), medical sensor node (SNj), and GWN are reliable, original, and up-to-date [9,22,24]. The notation, rules of the model, lists of our authentication goals, idealization of the exchange messages, and assumptions that are used in the verification process are illustrated in Tabs. 2–6, respectively.

Table 2: Basic notation

Notation	Description
X, Y	Statements
F, Q	Communication principles
K	Shared key
F|≡ X	F can consider X is true.
F◁ X	F sees X.
F|∼ X	F says X, then F can send a message containing X.
F ⇒ X	F jurisdiction over X.
#(X)	X is fresh.
(X, Y)	X or Y is a part of the formula (X, Y).
⟨X⟩Y	X is combined with Y.
{X} K	X is encrypted by K.
$F\overset{K}{\leftrightarrow}Q$	F and Q communicate with each other using K.
$F\overset{X}{\Leftrightarrow}Q$	A secret X is known only for F and Q.
SK	A session key.

The authentication and key agreement stage uses freshness authentication parameters to achieve mutual authentication. The KGWN-U is a cipher key that is used to cipher authentication messages between the Ui and GWN symmetrically. The SJj is an agreed secret key between all communication nodes. It comprises a

set of sequential numbers, pseudonym identity, and random numbers such as (SSi0, SSi1, SSj0, and SSj1), (TIDi, and TIDj), (r2, and r3), respectively.

Table 3: Believing Rules

Rule	Name	Formula
(R1)	Message meaning rule	$\frac{F\mid\equiv F \overset{K}{\leftrightarrow} Q,\ F\triangleleft(X)K}{F\mid\equiv Q\mid\sim X}$
(R2)	Freshness conjuncatenation rule	$\frac{F\mid\equiv\#(X)}{F\mid\equiv\#(X,\ Y)}$
(R3)	Belief rule	$\frac{F\mid\equiv X,\ F\mid\equiv Y}{F\mid\equiv(X,\ Y)}$
(R4)	Nonce verification rule	$\frac{F\mid\equiv\#(X),\ F\mid\equiv Q\mid\sim X}{F\mid\equiv Q\mid\equiv X}$
(R5)	Jurisdiction rule	$\frac{F\mid\equiv Q\Rightarrow X,\ F\mid\equiv Q\mid\equiv X}{F\mid\equiv X}$
(R6)	Session key rule	$\frac{F\mid\equiv\#(X),\ F\mid\equiv Q\mid\equiv X}{F\mid\equiv F \overset{K}{\leftrightarrow} Q}$

Table 4: Authentication goals

Goals	Description
(G1)	GWN $\mid\equiv$ GWN $\overset{SK}{\leftrightarrow}$ Ui .
(G2)	GWN $\mid\equiv$ Ui $\mid\equiv$ GWN $\overset{SK}{\leftrightarrow}$ Ui.
(G3)	SNj $\mid\equiv$ SNj $\overset{SK}{\leftrightarrow}$ GWN.
(G4)	SNj $\mid\equiv$ GWN $\mid\equiv$ SNj $\overset{SK}{\leftrightarrow}$ GWN.
(G5)	GWN $\mid\equiv$ GWN $\overset{SK}{\leftrightarrow}$ SNj.
(G6)	GWN $\mid\equiv$ SNj $\mid\equiv$ GWN $\overset{SK}{\leftrightarrow}$ SNj.
(G7)	Ui $\mid\equiv$ Ui $\overset{SK}{\leftrightarrow}$ GWN.
(G8)	Ui $\mid\equiv$ GWN $\mid\equiv$ Ui $\overset{SK}{\leftrightarrow}$ GWN.

Table 5: Idealized form

Message	Idealized form
(M1)	TIDi, Vi1, CTi1: ⟨(r2, SSi1)⟩ KGWN-U.
(M2)	CTj, SSj0, Vj: h5 (SJj, SNj0, SSj0).
(M3)	TIDj, Vj2: (SJj, SSj0, TIDj, SNj1) .
(M4)	CTi2, Vi2: ⟨(r3, TIDj, SJj, SSi0)⟩ KGWN-U.

Table 6: Initial assumptions

Assumption	Description
(A1)	Ui $\models$ # (r2, SSi1)
(A2)	Ui $\models$ GWN $\Rightarrow$ (r2, SSi1)
(A3)	GWN $\models$ # (SJj, SNj0, SSj0, r3, SSi0)
(A4)	GWN $\models$ Ui $\Rightarrow$ (SJj, r3, SSi0)
(A5)	GWN $\models$ SNj $\Rightarrow$ (SJj, SNj0, SSj0)
(A6)	SNj $\models$ # (SNj1, SSj1)
(A7)	SNj $\models$ GWN $\Rightarrow$ (SNj1, SSj1)
(A8)	Ui $\models$ Ui $\overset{\text{KGWN-U}}{\leftrightarrow}$ GWN.
(A9)	GWN $\models$ GWN $\overset{\text{KGWN-U}}{\leftrightarrow}$ Ui
(A10)	GWN $\models$ GWN $\overset{\text{SNj0}}{\Leftrightarrow}$ SNj
(A11)	SNj $\models$ SNj $\overset{\text{SNj0}}{\Leftrightarrow}$ GWN

In order to validate the authentication process of the authentication and key agreement stage, we need to prove that our goals are fulfilled according to the following points:

(1) Using (M1), Q1: (GWN $\triangleleft$ TIDi, Vi1, CTi1: $\langle$(r2, SSi1)$\rangle$ KGWN-U) can be seen. From (Q1), (A9), (R3), and (R1), Q2: (GWN $\models$ Ui $\mid\sim$ $\langle$ (r2, SSi1) $\rangle$ KGWN-U) can be obtained. Using (A3) and the (R2), Q3: (GWN $\models$ # $\langle$ (r2, SSi1) $\rangle$ KGWN-U) can be obtained. Using (Q2), (Q3), and (R4), Q4: (GWN $\models$ Ui $\models$ $\langle$ (r2, SSi1) $\rangle$ KGWN-U) can be obtained. Therefore, from (Q3), (Q4), and (R6), Q5 : (GWN $\models$ GWN $\overset{\text{SK}}{\leftrightarrow}$ Ui) can be deduced, which represents (**G1**). Considering (A4), (Q5), and (R4), Q6: (GWN $\models$Ui $\models$ GWN $\overset{\text{SK}}{\leftrightarrow}$ Ui) can be deduced, which leads to (**G2**) as well.

(2) In the same manner, **c**onsider (M2), F1: (SNj $\triangleleft$ CTj, SSj0, Vj: (SJj, SNj0, SSj0)) can be seen. Therefore, from (F1), (A11), (R3), and (R1), F2: (SNj $\models$ GWN $\mid\sim$ (SJj, SNj0, SSj0)) can be obtained. Next, using (A6) and (R2), F3: (SNj $\models$ # (SJj, SNj0, SSj0)) can be obtained. Then, using (F2), (F3), and (R4), F4: (SNj $\models$ GWN $\models$ (SJj, SNj0, SSj0)) can be obtained. Therefore, from (F3), (F4), and (R6), F5: (SNj $\models$ SNj $\overset{\text{SK}}{\leftrightarrow}$ GWN) can be deduced, which represents (**G3**). Using (A6), (F5), and (R4), F6 : (SNj $\models$ GWN $\models$SNj $\overset{\text{SK}}{\leftrightarrow}$ GWN) can be deduced, which leads to (**G4**) as well.

(3) Similarly, based on (M3), W1: (GWN $\triangleleft$ TIDj, Vj2: (SJj, SSj0, TIDj, SNj1)) can be seen. Thus, from (W1), (A10), (R3), and (R1), W2: (GWN $\models$ SNj $\mid\sim$ (SJj, SSj0, TIDj, SNj1)) can be obtained. Next, using (A3) and (R2), W3: (GWN $\models$ # (SJj, SSj0, TIDj, SNj1)) can be obtained. Then, using (W2), (W3), and (R4), W4: (GWN $\models$ SNj (SJj, SSj0, TIDj, SNj1)) can be obtained. Therefore, from (W3), (W4), and (R6), W5: (GWN $\models$ GWN $\overset{\text{SK}}{\leftrightarrow}$ SNj) can be deduced, which leads to (**G5**). Using (A7), (W5), and (R4), W6: (GWN $\models$SNj $\models$ GWN $\overset{\text{SK}}{\leftrightarrow}$SNj) can be deduced, which leads to (**G6**) as well.

(4) Finally, using (M4), E1: (Ui $\triangleleft$ CTi2, Vi2: $\langle$(r3, TIDj, SJj, SSi0)$\rangle$ KGWN-U) can be seen. Thus, from (E1), (A8), (R3), and (R1), E2: (Ui $\models$ GWN $\mid\sim$ (r3, TIDj, SJj, SSi0) KGWN-U) can be obtained. Next, using (A1) and (R2), E3 (Ui $\models$ # (r3, TIDj, SJj, SSi0) KGWN-U) can be obtained. Then, using (E2), (E3), and (R4), E4: (Ui $\models$ GWN $\models$ (r3, TIDj, SJj, SSi0) KGWN-U) can be deduced. Thus, from (E3), (E4), and (R6), E5: (Ui $\models$ Ui $\overset{\text{SK}}{\leftrightarrow}$ GWN) can be deduced, which leads to (**G7**).

Furthermore, according to (A2), (E5), and the (R4), E6: (Ui $|\equiv$ GWN $|\equiv$ Ui $\overset{SK}{\leftrightarrow}$ GWN) can be deduced, which leads to (**G8**).

According to (1), (2), (3), and (4), our goals are proven using the BAN logic model. Thus, the proposed authentication scheme can support mutual authentication among the Ui, SNj, and GWN elements during the authentication and key agreement stage.

3.1.2 Validation Using ProVerif Tool

This section validates the proposed authentication scheme using one of the most commonly used verification tools that has been developed for the automated verification of the security features of authentication schemes, called the ProVerif tool [19,25]. We have verified our proposed scheme in terms of the security of the established session key and mutual authentication, wherein this tool supposes that an adversary can block, delete, modify, and forward the exchanged messages between communication nodes. Therefore, if the results of the verification procedures are true, then the authentication scheme can resist all well-known attacks and the authentication parameters are exchanged securely. If not, the traces of existing attacks are presented.

In order to execute the verification procedures, we have provided a group of premises that are used in our verification program code, as illustrated in Fig. 6. The pubchHPGWN and pubchGWNHP are public communication channels used by the healthcare professional and the GWN to exchange the challenge and response messages between them. Moreover, the pubchGWNSN and pubchSNGWN are public communication channels used by the GWN and the sensor node to exchange the challenge and response messages between them (lines 1–2). Furthermore, we prototyped three sets of data: the type key for the secret keys, type coins to set the generated random numbers, and type host to define the healthcare professional, sensor node, and GWN as the participants in our scheme (line 7). Next, tables including the registration data of the participants were generated (lines 14–15). Then, we declared four free names, secret1, secret2, secret3, and secret4, to verify the secrecy of the session key (SJj) that will be established (line 16). Next, we defined eight authentication events that determine the start and end of the authentication processes to check the effectivity of mutual authentication between participants (lines 17–24). Finally, we declared eight queries to verify whether our authentication scheme could satisfy the session key secrecy and mutual authentication (lines 25–32).

Fig. 7 shows the code of the basic functions that are used to execute the main steps of the authentication stages. The h, xor, concate2, concate3, concat4, and concat5 represent the hash function, exclusive-or operation, and different levels of concatenation functions, respectively (lines 33–39). Besides this, the encrypt and decrypt symbols for encryption and decryption functions were used (lines 40–41). Finally, we defined a group of data type converter functions (lines 42–45).

The steps of the authentication and key agreement stage are performed as the simultaneous execution of three different processes in order to execute the role of each participant. Fig. 8 illustrates the code statements to simulate role of the healthcare professional, called the processHP process. The first section of the code statements represents **Step 1** in the healthcare professional side (lines 50–60). **Step 5** is represented in the second section of code statements (lines 61–64). The (StartGWNHPparam) event of GWN is set at line 48 and the (endHPGWNparam) event of the healthcare professional is set at line 65. Finally, the verification query code to check the secrecy of the session key (SJj) through the pubchHPGWN public channel is set at line 66.

```
free pubchHPGWN, pubchGWNHP: channel.
free pubchGWNSN, pubchSNGWN: channel.
free pubchHP: channel.
type key.
type host.
type coins.
free HPnode, SNnode, GWN: host.
free SNj0: coins[private].
free SNi: bitstring[private].
free SSj0: bitstring.
free SSi0: bitstring.
free XGWN: key[private].
free Vi: bitstring [private] .
table GWNTabPH(bitstring, bitstring, bitstring).
table GWNTabSN(bitstring, bitstring, bitstring).
free secret1, secret2, secret3, secret4: bitstring [private].
event StartHPGWNparam(host)
event endHPGWNparam(host).
event StartGWNHPparam(host).
event endGWNHPparam(host).
event StartGWNSNparam(host).
event endGWNSNparam(host).
event StartSNGWNparam(host).
event endSNGWNparam(host).
query z: host; inj-event(endHPGWNparam(z))==> inj-event(StartHPGWNparam(z)).
query z: host; inj-event(endGWNHPparam(z))==> inj-event(StartGWNHPparam(z)).
query z: host; inj-event(endGWNSNparam(z))==> inj-event(StartGWNSNparam(z)).
query z: host; inj-event(endSNGWNparam(z))==> inj-event(StartSNGWNparam(z)).
query attacker(secret1).
query attacker(secret2).
query attacker(secret3).
query attacker(secret4).
```

Figure 6: The code premises

```
fun h(bitstring): bitstring.
fun xor(bitstring, bitstring): bitstring.
equation forall x: bitstring, y: bitstring; xor(xor(x, y), y) = x.
fun concate2(bitstring, bitstring): bitstring.
fun concate3(bitstring, bitstring, bitstring): bitstring.
fun concate4(bitstring, bitstring, bitstring, bitstring): bitstring.
fun concate5(bitstring, bitstring, bitstring, bitstring, bitstring): bitstring.
fun encrypt (bitstring, key): bitstring.
reduc forall m:bitstring, k:key decrypt(encrypt(m, k), k)= m.
fun keytostring(key): bitstring [data, typeConverter].
fun coinstostring(coins): bitstring [data, typeConverter].
fun stringtokey(bitstring): key [data, typeConverter].
fun stringtocoins(bitstring): coins [data,typeConverter].
```

Figure 7: Code of the basic functions

```
let processHP (IDSNj: bitstring, Di: bitstring, Vi: bitstring)=
in (pubchHP, IDi :bitstring);
event StartGWNHPparam(GWN);
new KGWN-U: bitstring;
let KGWN-U = stringtokey((xor (Di, Vi));
let Ci'= h(concate3(IDi, Vi, keytostring (KGWN-U))in.
if h(Ci') = h(Ci) then
new SSi1: bitstring;
let SSi1 = (SSi0)in
let SNi = h(SSi1)in
new r2: coins;
let TIDi = h(concate2(ID, SNi))in
let CTi1 = encrypt (concate3(coinstostring(r2), IDSNj, SSi1), stringtokey (KGWN-U)) in
let Vi1 = h(concate3 (TIDi, coinstostring(r2), SSi1)) in
out(pubchHPGWN, (TIDi, CTi1, Vi1));
in(pubchGWNHP, (CTi2:bitstring , Vi2: bitstring));
let (r3: coins, TIDj: bitstring, SJj: Key, SSi0: bitstring) = decrypt(CTi1, stringtokey(KGWN-U))in
let Vi2' = h(concate3 (TIDi, coinstostring(r3), keytostring(SJj)))in
if Vi2' = Vi2 then
event endHPGWNparam(HPnode);
out(pubchGWNHP, encrypt(secret1, SJj)).
```

Figure 8: Healthcare professional process

Fig. 9 illustrates the role of the GWN, called the processGWN process. The first section of the code statements represents **Step 2** in the authentication and key agreement stage from the GWN side (lines 69–84), while **Step 4** is represented in the second section of the code statements (lines 85–98). The (StartHPGWNparam) event of the healthcare professional is set at line 74, and the (StartSNGWNparam) event of the sensor node is set at line 80. The (endGWNHPparam) and (endGWNSNparam) events of the GWN are set at lines 97 and 90, respectively. The verification query code to check the secrecy of the session key (SJj) through the pubchGWNHP public channel is set at line 99.

```
67  let processGWN(IDSNj: bitstring) =
68  in(pubchHPGWN, (XIDi: bitstring, CTi1: bitstring, Vi1: bitstring));
69  if XIDi = TIDi then
70  get GWNTabPH (= XIDI, IDi: bitstring, SSi0:bitstring)in
71  let KGWN-U = h2 (concate2(IDi, XGWN))in
72  let (r2:coins, IDSNj: bitstring, SSi1: bitstring)= decrypt(CTi1, stringtokey(KGWN-U))in
73  let Vi1'= h3(TIDi, coinstostring (r2), SSi01)in
74  event StartHPGWNparam(HPnode);
75  if Vi1' = Vi1 then
76  new SJj: key;
77  new ST: bitstring;
78  new TIDj: bitstring;
79  let SNj0 = h(cocate2(stringtocoins(SNj0), IDSNj))in
80  event StartSNGWNparam(SNnode);
81  let CTj0 = XOR(Concat2(keytostring(SJj), ST), h(concate3(SNj0, IDSNj, SSj0))) in
82  let TIDj = h(concate2(stringtokey(SJj), IDSNj)in
83  let Vj0 = h(concate5(ST, IDSNj, stringtokey(SJj), SNj0, SSj0))
84  out(pubchGWNSN, (CTj0, Vj0, SSj0));
85  in (pubchSNGWN, (TIDj: bitstring, Vj1: bitstring);
86  If xIDj = TIDj then
87  get GWNTabSN (= xIDj, IDSNj: bitstring, SSj0:bitstring)in
88  let Vj1' = h (concate5(ST, IDSNj, SJj, SNj0, TIDj)in
89  if Vj1'= Vj1 then
90  event endGWNSNparam(GWN);
91  out(pubchSNGWN, encrypt(secret3, SJj));
92  let SNi = h1 (SSi0)in
93  let TIDi = h2 (concate2(IDi, SNi)in
94  new r3: coins;
95  let CTi2 = encrypt (concate5(coinstostring (r3), TIDj, keytostring(SJj), SSi0)), KGWN-U)in
96  Let Vi2 = h (concate3 (TIDi, coinstostring(r3),keytostring(SJj)))in
97  event endGWNHPparam(GWN);
98  out (pubchGWNHP, (Vi2, CTi2));
99  out (pubchGWNHP, encrypt (secret2, SJj)).
```

Figure 9: GWN process

Fig. 10 illustrates the role of the sensor node, called the processSN process. The code statements represent **Step 3** in the attended sensor node (lines 101–111). The (StartGWNSNparam) event of the GWN is set at line 85, and the (endSNGWNparam) event of the sensor node is set at line 102. The verification query code to check the secrecy of the session key (SJj) through the pubchGWNSN public channel is set at line 112.

```
100 let processSN(IDSNj: bitstring) =
101 in (pubchGWNSN, (CTj0: bitstring, Vj0: bitstring));
102 event StartGWNSNparam(GWN);
103 let (SJj: key, ST: bitstring) = XOR (CTj0, h3(coinstostring (SNj0), IDSNj, SSj0)) in
104 let vj0'= h(concate5(ST, IDSNj, keytostring(SJj), coinstostring (SNj0), SSj0)) in
105 if Vj0'= Vj0 then
106 new SNj1: coins;
107 let TIDj = h(concate2(keytostring(SJj), IDSNj)in
108 let SNj1 =(SNj0)in
109 let Vj1 = h(cocate5(ST, IDSNj, keytostring(SJj),coinstostring(SNj1), TIDj)in
110 out (pubchSNGWN, TIDj, Vj1);
111 event endSNGWNparam (SNnode);
112 out (pubchGWNSN, encrypt (secret4, SJj)).
```

Figure 10: Sensor node process

Fig. 11 illustrates the code statement of the main process that executes the processes of the participants simultaneously. The code statements (lines 114 – 122) represent the registration stages of the healthcare professional and sensor node, wherein the authentication data are initiated. In addition, the code statements to launch an unbounded number of authentication sessions between the processes are represented (lines 123 – 127).

```
113 process
114 new IDi, PW, SNi, SSi0, Di, Vi, Ci, IDSNj: bitstring;
115 new XGWN, KGWN-U: key;
116 let SNi = h1(SSi0)in
117 let TIDi = h2(concate2(IDi, SNi))in
118 let KGWN-U = stringtokey (h2 (concate2(IDi, keytostring (XGWN)))in
119 let Di = xor (keytostring (KGWN-U), Vi) in
120 let Ci = h(concate3(IDi, Vi, KGWN-U)) in
121 insert GWNTabPH(TIDi, IDi, SSi0);
122 insert GWNTabSN(TIDj, IDSNj, SSj0);
123 (
124 (!processHP(IDSNj, Di, Ci, Vi))|
125 (!processGWN(IDSNj))|
126 (!processSN(IDSNj))
127 )
```

Figure 11: Main process

Fig. 12. shows the results of the verification queries. The first four results demonstrate that the authentication events are executed in a stable order. Thus, our proposed scheme can satisfy mutual authentication among the heath professional (HPnode), GWN (GWN), and sensor node (SNnode). The second four results illustrate that the attacker cannot trace secret1, secret2, secret3, and secret4 (free names). Thus, our proposed scheme can preserve the secrecy of the session key (SJj).

```
RESULT inj-event(endHPGWNparam(HPnode))==> inj-event(StartHPGWNparam(HPnode)) is true.
RESULT inj-event(endGWNHPparam(GWN))==> inj-event(StartGWNHPparam(GWN)) is true.
RESULT inj-event(endGWNSNparam(GWN))==> inj-event(StartGWNSNparam(GWN)) is true.
RESULT inj-event(endSNGWNparam(SNnode))==> inj-event(StartSNGWNparam(SNnode)) is true.
RESULT not attacker (secret1[])is true.
RESULT not attacker (secret2[])is true.
RESULT not attacker (secret3[])is true.
RESULT not attacker (secret4[])is true.
```

Figure 12: Verification results

3.2 Informal Security Analysis

3.2.1 Security Services Achievement

This section presents an informal discussion of the ability of the proposed authentication scheme to achieve a suitable set of security services, which comprise authentication key agreement, mutual authentication, anonymity and untraceability, and perfect forward secrecy.

The Proposed Authentication Scheme Supports the Authentication Key Agreement.

Proof. During the execution of the authentication and key agreement stage, the GWN randomly generates (SJj) as a shared secret key to accomplish mutual authentication with SNj and Ui, wherein the SJj key is updated for each authentication session between them. Thus, our authentication scheme can generate a session shared key between the authentication elements.

The Proposed Authentication Scheme Supports Mutual Authentication Service

Proof. We have a set of verification points in the login authentication and key agreement processes that are executed to satisfy the mutual authentication services. The GWN verifies the (M1) message via the received parameters (TIDi, SSi1, and V1i) to check the legitimacy of Ui. Meanwhile, the SNj confirms the legitimacy of the GWN by verifying the (M2) message via the SSj0 and Vj0 values. The GWN checks the SNj by verifying the received values of the TIDj and Vj1 through the (M3) message. Finally, the Ui verifies the GWN's authenticity by checking the received values of the SSi0 and Vi2 by the (M4) message.

Therefore, the proposed authentication scheme is able to support mutual authentication services among the Ui, SNj, and GWN.

The Proposed Authentication Scheme Supports Anonymity and Untraceability Service.

Proof. To maintain Ui and SNj's anonymity and untraceability in our authentication scheme, the authentication messages exchanged during the authentication and key agreement stage do not contain the real identities of the Ui (IDi) and the SNj (IDSNj). Instead, our authentication scheme uses pseudonym identities (TIDi) and (TIDj) that are generated by one-way hash functions after completing each authentication session. Thus, it is almost impossible for an unauthorized party to obtain the real identity of either the Ui or the SNj from the messages exchanged between the authentication nodes. Thus, our authentication scheme can support the anonymity and untraceability of the service.

The Proposed Authentication Scheme Supports Perfect Forward Secrecy Service.

Proof. In our proposed authentication scheme, if an unauthorized party acquires the long-term keys of the authentication nodes, which are SNj0 and KGWN-U, it still cannot obtain the session key (SJj) that is generated by the GWN randomly. The reason for this is that, after executing the authentication and key agreement stage successfully, the keys, SNj0 and KGWN-U, will be changed by one-way hash functions. Thus, our authentication scheme is able to provide a perfect forward secrecy service.

3.2.2 Attacks Resistance Analysis

An attacker can collect, decrypt, replace, track, imitate, and resend the authentication messages as they are transmitted over unsecured communication channels. In this section, we demonstrate that our authentication scheme can resist different types of known attacks in such an environment.

The Proposed Scheme Resists Desynchronization Attack.

Proof. The proposed authentication scheme uses different authentication parameters that can retain the synchronization between the authentication nodes, such as the pseudonym identities (TIDi and TIDj), sequential numbers (SSi0, SSi1, SSj0, and SSj1), and hash values (SNi, SNj0, and SNj1). Hence, the proposed scheme employs additional methods to preserve the consistency and synchronization of such values and prevent a desynchronization attack. To demonstrate how our authentication scheme achieves this, we take into account the following possible attack scenarios:

Scenario 1: Assume that an attacker has interrupted the (M1) message. In this case, the attacker cannot to disrupt the synchronization among the GWN and Ui permanently. This attack suspends the authentication process temporarily, before the Ui and GWN have updated the values of the SSi1 and SSi0. Thus, this scenario will have no effect on the synchronization during the subsequent authentication session.

Scenario 2: Assume that an attacker has interrupted the (M2) message. In such a case, the attacker cannot disrupt the synchronization between the SNj and GWN permanently. During the subsequent authentication session, the values of SNj1 and SSj1 will be updated by the SNj ΔSSj times as $SNj1 = h2(SNj1 \parallel IDSNj)$ and $SSj1 = SSj1 + 1$, respectively. As a result, the SNj will compute the TIDj value, which can synchronize the value of TIDj that is stored in the GWN. Thus, this case cannot cause an

asynchronous state among the GWN and SNj permanently, and it will have no effect on the subsequent authentication session.

Scenario 3: Assume that an attacker has interrupted the (M3) message. In such a case, the attacker cannot disrupt the synchronization between the SNj and GWN permanently. The result of this scenario is equivalent to scenario 2. Thus, this scenario will not be taken into account.

Scenario 4: Assume that an attacker has interrupted the (M4) message. In such a case, the attacker cannot disrupt the synchronization between the Ui and GWN permanently. In the upcoming authentication session, the TIDi value in the GWN will be updated, while the TIDi value in the Ui will not update. Fortunately, the previous value of TIDi is stored through the TIDi* value in the GWN, i.e., TIDi = TIDi*. Thus, when the next session is initiated by the Ui using the unchanged TIDi, the GWN is able to recognize the Ui and complete the subsequent authentication. Thus, this scenario cannot cause an asynchronous state between the GWN and Ui permanently, and it will have no effect on the subsequent authentication session.

Therefore, according to the above-discussed scenarios, our authentication scheme can protect against a desynchronization attack.

The Proposed Scheme Resists Stolen Password Table Attack.

Proof. In the proposed authentication scheme, the service provider (GWN) does not contain any details about the Ui's password or biometrics data. Thus, our authentication scheme is already able to resist a stolen verified table attack.

The Proposed Scheme Resists Incorrect Password Login Attack.

Proof. A detection mechanism is maintained in our authentication scheme to prevent an incorrect password login attack during the first steps of the authentication and key agreement stage without excessive computation when the SC obtains any incorrect login authentication data. The value of the h1 (Ci) stored in the smartcard is used to check the user's legitimacy. If the user inputs an incorrect password and biometric, then the computed h1 (Ci*) value is not equal to the stored value of h1 (Ci). Therefore, the SC will reject the login request. As a result, the proposed authentication scheme resists an incorrect password login attack.

The Proposed Scheme Resists Smartcard Attack.

Proof. The proposed authentication scheme uses three authentication factors (i.e., identity, password, and biometric). Even if an attacker is able to steal hidden information from a smartcard, he or she will be unable to log in. The explanation for this is that the attacker also needs to know the authorized user's identity IDi and biometric information Bi in order to create a login message.

The Proposed Scheme Resists Man-in-the-Middle Attack.

Proof. In our authentication scheme, the challenge and response messages that are exchanged among the elements of the system are protected by the SNi, SNj0, SNj, and K_{GWN-U}. Thus, an unauthorized party cannot create valid authentication messages without these values. Thus, our authentication scheme can resist a man-in-the-middle attack.

The Proposed Scheme Resists Insider Privileged Attack.

Proof. Our authentication scheme does not allow inside workers to carry out privileged insider attacks. When the healthcare professional registration stage is executed, the PWi and BIOi values of the Ui are transmitted as hidden values through the hash value that is represented as Vi = h3 (IDi || PWi || BPWi). The one-way property of the hash function prevents the insider from disclosing the real value. As a result, the proposed authentication scheme can resist a privileged insider attack.

The Proposed Scheme Resists Impersonation Attack.

Proof. To ensure that our authentication scheme can protect against an impersonation attack, we consider the following possible attack scenarios:

Scenario 1: To impersonate the Ui entity during authentication, assume that an attacker has intercepted the login request message M1: {TIDi, CTi1, and Vi1} that was sent to the GWN node, where TIDi = h2 (IDi || SNi), SNi = h1 (SSi1), CTi1= E_{KGWN-U} (r2 || IDSNj || SSi1), and vi1= h (TIDi || r2 || SSi1). The encrypted value (CTi1) is not available, since the attack cannot know the secret key (KGWN−U) or the actual (SNi) value. As a result, the attacker would be unable to impersonate Ui by computing (Vi1) with completely separate (r2) and (SSi1) values.

Scenario 2: To impersonate the GWN node during authentication, assume that an attacker has intercepted the authentication request message M2: {CTj0, Vj0, and SSj0} that has been sent to SNj. Since the attacker cannot know the hidden keys or the value of (CTj0), the encrypted value of (CTj0) is infeasible. As a consequence, the attacker cannot impersonate the GWN by computing (Vj0) using separate (SJj), (SNj0), and (SSj0).

Scenario 3: To impersonate the SNj node during authentication, assume that an attacker has intercepted the authentication response message M3: {TIDj, and Vj1} that has been sent to the GWN. Since the attacker does not know the SJj, SNj0, and SSj0, they are unable to compute Vj1 and TIDj. As a consequence, the attacker cannot impersonate the SNj by computing (Vj1) using separate (SJj), (SNj0), and (SSj0).

Therefore, according to the above-discussed scenarios, our authentication scheme can protect against an impersonation attack.

The Proposed Scheme Resists Replay Attack.

Proof. To ensure that our authentication scheme can resist a replay attack, we consider the following possible attack scenarios:

Scenario 1: Consider that an attacker resends the previous intercepted M1: {TIDi, CTi0, and Vi1} to the service provider (GWN) without any alterations, wherein TIDi = h2 (IDi || SNi), SNi = h1 (SSi1), CTi0 = E_{KGWN-U} (r2 || IDSNj || SSi1), and Vi1= h1 (TIDi || r2 || SSi1). As a result, the GWN will decrypt the CTi0 and then verify SSi1, which represents the serial number of the present authentication session, which is modified as (SSi1 = SSi1 + 1) during each successful authentication session. Since the SSi0 would have been checked in the previous authentication session, the GWN would refuse the login authentication request.

Scenario 2: Consider that an attacker resends the previously intercepted M2: {CTj0, Vj0, SSj0}, without any alterations, wherein CTj0 = ((SJj || ST) ⊕ h3 (SNj0 || IDSNj || SSj0)), Vj0 = h5 (ST || IDSNj || SJj || SNj0 || SSj0), and SSj0 represents the previous authentication session's serial number, which is modified as SSj0 =SSj0 + 1. Since the SSj0 would have been checked in the previous authentication session, the SNj entity would refuse the login authentication request.

Both authentication messages (i.e., M1 and M2) use the serial numbers, which are changed after each subsequent authentication session. Thus, our authentication scheme can prevent a replay attack during authentication in all the mentioned attack scenarios.

3.3 Security Comparisons

In this section, we compare our authentication scheme with other recently proposed authentication schemes [9,21,22,23].

The comparison results in Tab. 7 show that our authentication scheme can satisfy all the security features, while the other schemes presented in [9,21,22,23] did not provide security features such as fully

mutual authentication among the elements of the system or medical sensor node anonymity. Moreover, the perfect forward secrecy service was not satisfied in [21] and [22]. Furthermore, our authentication scheme can resist all well-known attacks, while the authentication schemes presented in [21] and [22] cannot resist a desynchronization attack. The authentication scheme in [21] cannot resist healthcare professional impersonation, insider, and stolen password verifier table attacks. Moreover, our authentication scheme is the only one that can resist a man-in-the-middle attack. Therefore, our authentication scheme can achieve a high level of security compared to other recently proposed authentication schemes.

Table 7: Security services and attack resistance comparisons

Security features	[9]	[21]	[22]	[23]	Our scheme
Full mutual authentication.	✗	✗	✗	✗	√
Healthcare professional anonymity.	√	√	√	√	√
Sensor node anonymity.	✗	✗	✗	✗	√
Three authentication factors (3F).	√	✗	√	✗	√
Perfect forward secrecy.	√	✗	✗	√	√
Desynchronization attack resistance.	√	✗	✗	√	√
Smartcard loss attack resistance.	√	√	√	√	√
Replay attack resistance.	√	√	√	√	√
GWN impersonate attack resistance.	√	√	√	√	√
Healthcare professional impersonate attack resistance.	√	✗	√	√	√
Sensor node impersonate attack resistance.	√	√	√	√	√
Man-in-the-middle attack resistance.	✗	✗	✗	✗	√
Wrong login attack resistance.	√	√	√	√	√
Insider attack resistance.	√	✗	√	√	√
Stolen password verifier table attack resistance.	√	✗	√	√	√

4 Performance Analysis

This section assesses the efficiency of our authentication scheme and compares its costs in terms of the storage space used, communication size, and run time of computation with the authentication schemes recently proposed in [9,21,22,23]. The computation and communication costs are calculated for the login authentication and key agreement stage, whereas the costs of the storage space used are calculated for the healthcare professional registration and sensor node registration stages, whether for healthcare professionals or sensor nodes.

In order to perform fairly accurate comparisons, we assume the following: the size of sequential numbers, security codes, random numbers, passwords, and identities are set to be 128 bits; the output of the used hash functions is equal to 160 bits, and the input/output of the encryption/decryption functions are multiples of 128 bits. Moreover, we assume that the running times of the fuzzy extractor generating function, SHA-1 hash function, and AES cryptographic function are ($T_{fe} = 0.0171$s), ($T_h = 0.00032$s), and ($T_{E/D} = 0.0056$s), respectively, as in [3,10,26–29].

4.1 Storage Space Cost Analysis

The cost optimization of the used storage space in the healthcare professional/smartcards and the medical sensor nodes is one of the major issues in such systems. The size of the hash functions that are embedded in the smartcards is not taken into account in order to simplify the analysis. The storage space costs of smartcards and sensor nodes in our authentication scheme and the authentication schemes proposed in [9,21–23] are shown in Tab. 8.

Table 8: Storage space cost analysis

Authentication schemes	Healthcare professional/Smartcard (bits)	Sensor node (bits)
[9]	864	256
[21]	608	288
[22]	768	320
[23]	736	800
Our scheme	640	288

In our authentication scheme, the storage space cost of the healthcare professional's smartcard to store the (Rep (.), Pi, Di, h1(Ci), and SSi1) is (64 + 128 + 160 + 160 + 128) = 640 bits, while that cost of storing the (SSj1 and SNj0) in the sensor node is (128 + 160) = 288 bits. Tab. 8 shows that our authentication scheme requires the least storage space for the healthcare professional's smartcard. Furthermore, the storage space that is needed for the sensor node in our authentication scheme is greater than that of the authentication scheme proposed in [9] but less than in other authentication schemes.

4.2 Communication Cost Analysis

The communication costs can be calculated according to the total size of the transmitted authentication messages among elements of the system during the login authentication and key agreement stage. The total communication costs of our authentication scheme and the authentication schemes proposed in [9,21,22,23] are shown in Tab. 9.

Table 9: Communication cost analysis

Authentication scheme	M1	M2	M3	M4	M5	Total (bits)
[9]	576	448	320	320	160	1,824
[21]	864	480	320	640	-	2,304
[22]	1088	1376	448	1056	-	3,968
[23]	736	896	480	800	-	2,912
Our scheme	704	448	320	800	-	2,272

In our authentication scheme, the size of M1 = (160 + 3 [128] + 160) = 704 bits, M2 = (160 + 160 + 128) = 448 bits, M3 = (160 + 160) = 320 bits, and M4 = (160 + 5 × (128)) = 800. The overall results of the communications costs for our authentication scheme and the other authentication schemes proposed in [9,21–23] indicate that our authentication scheme carries the lowest communication costs.

4.3 Computation Cost Analysis

In this section, the computation costs are compared among our authentication scheme and the authentication schemes proposed in [9,21–23]. The overall time required to execute the cryptographic functions in each element of the system is computed. The total computation costs for our authentication scheme and other authentication schemes proposed in [9,21–23] are shown in Tab. 10.

Table 10: Computation cost analysis

Authentication schemes	Healthcare professional	GWN	Sensor node	Total	Cost (s)
[9]	$1T_{fe} + 14T_h$	$21T_h$	$7T_h$	$1T_{fe} + 42T_h$	0.03054
[21]	$12T_h$	$18T_h$	$6T_h$	$36T_h$	0.01152
[22]	$10T_{E/D} + 11T_h$	$17T_{E/D} + 16T_h$	$5T_{E/D} + 7T_h$	$32T_{E/D} + 34T_h$	0.19008
[23]	$10T_h$	$18T_h$	$6T_h$	$34T_h$	0.01088
Our scheme	$1T_{fe} + 8T_{E/D} + 10T_h$	$8T_{E/D} + 6T_h$	$5T_h$	$1T_{fe} + 16T_{E/D} + 21T_h$	0,11342

The results show that our authentication scheme carried lower costs of computation than the authentication scheme proposed in [22]; in both of them, hash and encryption/decryption functions are used simultaneously. Meanwhile, the computation costs of our authentication scheme are higher than those of other authentication schemes that only use one-way hash functions during the authentication process.

5 Conclusion

A secure and anonymous three-factor authentication scheme for healthcare systems is proposed in this paper based on a WMSN to solve the present security issues in such systems. The proposed authentication scheme offers promising security services, such as fully mutual authentication, perfect forward service, anonymity, and untraceability. To verify the security level of our authentication scheme, the BAN logic model and ProVerif tool were used, and its resistance to attacks is discussed considering all possible attack scenarios. Thus, the proposed authentication scheme can protect against desynchronization, impersonation, smartcard loss, replay, man-in-the-middle, insider, and password table attacks. Furthermore, the performance cost analysis shows that our authentication scheme is practical to use, with reasonable costs in terms of the storage space, computation, and communication. Finally, our authentication scheme can be used by healthcare professionals in healthcare systems to track and diagnose the medical status of patients safely and remotely.

Acknowledgement: The authors express their thanks to colleagues in the Computer Sciences Dept. at Jouf University for their collaboration and support.

Funding Statement: The authors would like to thank the Deanship of Graduate Studies at Jouf University for funding and supporting this research through the initiative of DGS, Graduate Students Research Support (GSR) at Jouf University, Saudi Arabia.

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

References

[1] S. R. Patil, D. R. Gawade and S. N. Divekar, "Remote wireless patient monitoring system," *International Journal of Electronics & Communication Technology*, vol. 6, no. 1, pp. 9–13, 2015.

[2] A. Ibrahim and W. Zhuopeng, "IOT patient health monitoring system," *Journal of Engineering Research and Application*, vol. 8, no. 1, pp. 77–80, 2018.

[3] S. Nashwan, "An end-to-end authentication scheme for healthcare IoT systems using WMSN," *Computers, Materials and Continua*, vol. 68, no. 1, pp. 607–642, 2021.

[4] J. Mo, Z. Hu and Y. Lin, "Cryptanalysis and security improvement of two authentication schemes for healthcare systems using wireless medical sensor networks," *Security and Communication Networks*, vol. 2020, no. 1, pp. 1–10, 2020.

[5] L. V. Morales, D. D. Ruiz and S. J. Rueda, "Comprehensive security for body area networks: A survey," *International Journal of Network Security*, vol. 21, no. 2, pp. 342–354, 2019.

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

[7] K. Dhakal, A. Alsadoon, P. W. Prasad, R. S. Ali, L. Pham *et al.*, "A novel solution for a wireless body sensor network: Telehealth elderly people monitoring," *Egyptian Informatics Journal*, vol. 21, no. 2, pp. 91–103, 2020.

[8] A. Al-Qerem, F. Kharbat, S. Nashwan, S. Ashraf and K. Blaou, "General model for best feature extraction of EEG using discrete wavelet transform wavelet family and differential evolution," *International Journal of Distributed Sensor Networks*, vol. 16, no. 3, pp. 91–103, 2020.

[9] M. Shuai, B. Liu, N. Yu and X. Xiong, "Lightweight and secure three-factor authentication scheme for remote patient monitoring using on-body wireless networks," *Security and Communication Networks*, vol. 2019, no. 12, pp. 1–14, 2019.

[10] S. Nashwan, "AAA-Wsn: Anonymous access authentication scheme for wireless sensor networks in big data environment," *Egyptian Informatics Journal*, vol. 22, no. 1, pp. 15–26, 2021.

[11] S. Nashwan, "SAK-Aka: A secure anonymity key of authentication and key agreement protocol for LTE network," *International Arab Journal of Information Technology*, vol. 14, no. 5, pp. 790–801, 2017.

[12] S. Nashwan, "Secure authentication protocol for NFC mobile payment systems," *International Journal of Computer Science and Network Security*, vol. 17, no. 8, pp. 256–263, 2017.

[13] S. Nashwan, "Synchronous authentication key management scheme for inter-eNB handover over LTE networks," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 8, pp. 100–107, 2017.

[14] L. Xiong, T. Peng, H. Liang and Z. Liu, "A lightweight anonymous authentication protocol with perfect forward secrecy for wireless sensor networks," *Sensors*, vol. 17, no. 24, pp. 1–28, 2017.

[15] J. Jung, J. Kim, Y. Choi and D. Won, "An anonymous user authentication and key agreement scheme based on a symmetric cryptosystem in wireless sensor networks," *Sensors*, vol. 16, no. 8, pp. 1–30, 2016.

[16] P. Kumar and H. J. Lee, "Security issues in healthcare applications using wireless medical sensor networks: A survey," *Sensors*, vol. 12, no. 1, pp. 55–91, 2012.

[17] P. H. Waghmare and A. N. Bhute, "Healthcare monitoring system using smartphone," *International Journal of Innovative Research in Science*, vol. 6, no. 6, pp. 12407–12413, 2017.

[18] D. He, K. Kumar, J. Chen, C. Lee, N. Chilamkurti *et al.*, "Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks," *Multimedia Systems*, vol. 21, no. 1, pp. 49–60, 2015.

[19] F. Wu, X. Li, A. K. Sangaiah, L. Xu, S. Kumari *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, no. 1, pp. 727–737, 2017.

[20] J. Srinivas, D. Mishra and S. Mukhopadhyay, "A mutual authentication framework for wireless medical sensor networks," *Journal of Medical Systems*, vol. 41, no. 5, pp. 80–99, 2017.

[21] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan and N. Kumar, "A robust and anonymous patient monitoring system using wireless medical sensor networks," *Future Generation Computer Systems*, vol. 80, no. 4, pp. 483–495, 2018.

[22] R. Ali, A. K. Pal, S. Kumari, A. K. Sangaiah, X. Li *et al.*, "An enhanced three factor-based authentication protocol using wireless medical sensor networks for healthcare monitoring," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 1, pp. 74, 2018.

[23] M. Fotouhi, M. Bayat, A. K. Das, H. A. Far, S. M. Pournaghi *et al.*, "A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care IoT," *Computer Networks*, vol. 177, no. 1, pp. 107333, 2020.

[24] S. Nashwan and B. Alshammari, "Formal analysis of MCAP protocol against replay attack," *British Journal of Mathematics & Computer Science*, vol. 22, no. 1, pp. 1–14, 2017.

[25] B. Blanchet, "Modeling and verifying security protocols with the applied pi calculus and proVerif," *Foundations and Trends in Privacy and Security*, vol. 1, no. 1, pp. 1–135, 2016.

[26] L. Xiong, T. Peng, H. Liang and Z. Liu, "A lightweight anonymous authentication protocol with perfect forward secrecy for wireless sensor networks," *Sensors*, vol. 17, no. 24, pp. 1–28, 2017.

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

[28] S. Nashwan and I. I. H. Nashwan, "Reducing the overhead messages cost of the SAK-aKA authentication scheme for 4G/5G mobile networks," *IEEE Access*, vol. 9, pp. 97539–97545, 2021.

[29] S. Nashwan and I. I. H. Nashwan, "An analytic model for reducing authentication signaling traffic in an and-to-end authentication scheme," *Sensors*, vol. 21, no. 15, pp. 1–15, 2021.