Arabian Journal for Science and Engineering (2021) 46:8457–8473
https://doi.org/10.1007/s13369-021-05532-8

RESEARCH ARTICLE-COMPUTER ENGINEERING AND COMPUTER SCIENCE

An Efficient Mutual Authentication and Symmetric Key Agreement Scheme for Wireless Body Area Network

Chukhu Chunka[1] · **Subhasish Banerjee**[2]

Received: 1 August 2020 / Accepted: 4 March 2021 / Published online: 20 March 2021
© King Fahd University of Petroleum & Minerals 2021

Abstract
A Wireless Body Area Networks (WBANs) is a wireless network in which sensors are embedded inside the body of a human, to monitor the health of patient continuously without any constraint in his normal daily life activities. As the information from the embed sensor is transmitted through wireless network and device has a limited battery power, therefore, the assurance of security in such tiny devices related to medical patients is highly recommended. Thus, the shared information must be maintained in terms of integrity, confidentiality, non-repudiation, untraceable key establishment, and mutual authentication in WBAN. In this context, to achieve high security and efficiency in WBAN, an efficient mutual authentication and secret key agreement scheme have been proposed in this paper and also listed out some drawbacks of an existing mutual authentication and key agreement of Li et al.'s scheme. To confirm the efficiency and security, the proposed scheme has been verified using formal security analysis tool namely, ProVerif and BAN logic. The low communication and computation costs indicate that our scheme is more suitable for practical application in healthcare as compared to other existing schemes.

Keywords Authentication · WBAN · Master Key · Hub Node

1 Introduction

With the advancement of wireless technologies many mini nature devices are correspondingly developed, one of them is sensors or wearable sensor devices which are connected and implanted to the body to sense the physiological signals of the human body by frequently monitoring and examining. In WBAN, the sensor screens the health of humans by monitoring the parameters like body temperature, heart rate, the sugar level of blood, blood pressure level, and respiratory rate, etc. In order to prevent the old-age problem and to reduce the chronic conditions, the cost-effective healthcare infrastructure is recommended. Nowadays, medical professionals reduce stress because of advanced technologies, like implanting the sensor devices in the human body. As a result, they get high-quality medical facilities and treatments at home without any intervention of medical professionals. The application of WBAN is Emerging Medical Response System (ENRS), Ubiquitous Health Monitoring (UHM), Computer-Assisted Rehabilitation, etc. Figure 1 illustrates the communication segments of WBAN. In WBAN, personal sensitive information must be protected from unauthorized admission. Hence, maintaining security and privacy is the prime concern in healthcare.

In the body area network, there are three kinds of nodes i.e., hub node, first-level node and second-level node. The wearable node which is attached to the body is called second-level nodes (S_n), whereas intermediate node is a first-level node (I_n) that collects the information from S_n and forward to hub node (*HNode*) for further processing. The I_n node generally has more loading capacity, high-processing speed, superior computing capabilities, and high-communication competence than S_n. The *HNode* is said to be a local server which is at the center [4] of WBAN and may assume as a trusted server [34]. The *HNode* collects all the complex data from sensor nodes and forwards to healthcare or diagnostic center. The Tier -1 shows the connection between S_n and I_n known as Intra-BAN communication. Similarly, connection between I_n and *HNode*, known as

✉ Chukhu Chunka
chukhuchunka20@gmail.com

Subhasish Banerjee
subhasishism@gmail.com

[1] Department of Computer Science & Engineering, ITER SOA Deemed To Be University, Bhubaneswar, Odisha, India

[2] Department of Computer Science & Engineering, National Institute of Technology, Yupia, Arunachal Pradesh, India

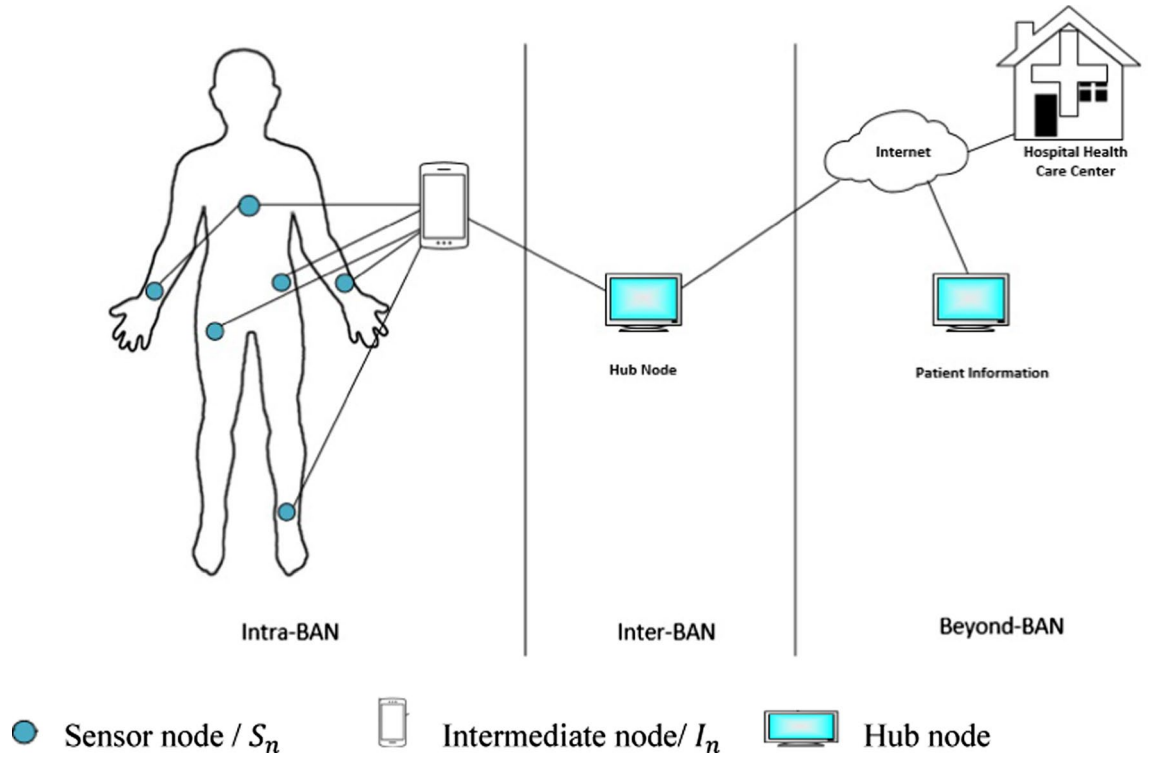

Fig.1 Communication segments of WBAN

Inter-BAN communication, is Tier-2. The connections between the *HNode* and healthcare center are considered under Tier 3 which is outside the WBAN. Tier 3 medical center provides the services to the users or patients. The connection between the two sections comes with various difficulties and challenges. Therefore, in this paper, we have concentrated on the establishment of safe communications within these two segments.

The major contributions in this paper are mentioned bellow:

i. To analyze the Li et al.'s [1] scheme and find out the possible security breaches like, linkable to the session, sensor node capture and eavesdropping; using informal security analysis.
ii. To design a new efficient authentication and key agreement scheme by using only the cryptographic hash function and XOR operation to overcome the drawback of Li et al.'s scheme.
iii. To verify the secrecy and authenticity between *HNode*, and S_n, the proposed scheme has rigorously analyzed through informal security analysis, as well as with formal security analysis like, BAN logic to prove the correctness of our scheme and ProVerif Tools to verify secrecy of the scheme.
iv. Finally, the proposed scheme has been compared with other related schemes in the context of computational cost, memory overhead, communication message exchange and security functionalities.

The remaining part of the paper is arranged as follows: Sect. 2 provides the related work. Brief review on Li et al.'s scheme [1] of WBAN has defined in Sect. 3. In Sect. 4, to defeat Li et al.'s scheme, a new efficient scheme of authentication and key agreement has been proposed. Section 5 includes the discussion of informal and formal security analysis of the proposed scheme using formal verification called BAN Logic and ProVerif simulation tool to show the credibility of the scheme. The comparison of our scheme with other existing related schemes is defined in Sect. 6 and finally, Sect. 7 defines conclusion and future scope of this research.

2 Related Work

Many researchers have developed authentication and key agreement schemes for different environments like for single sever [3–5], multi-server environment [6, 7], and wireless sensor network [8, 9]. In WBAN, while considering the

secure transmission of medical patients' information over wireless networks, the entity must mutually authenticate each other. The Non-cryptographic physiological signal-based [10, 11] schemes have been proposed to make secure communication in WBAN. Recently, many investigators have undertaken the further research in WBAN network to overcome the various security challenges and to increase the efficiency, overall. In 2010, Venkata Subramanian et al. [10] proposed a scheme in WBAN, where, it has been observed that the identical physiological signals are difficult to measure in the different parts of the body. Therefore, to improve the security in the WBAN, researchers integrated biometric characteristics too. Since then many of the researcher's work on biometrics key distribution through physiological signal for WBAN [12, 13]. In 2006, Poon et al. [12] proposed an authenticated and secure communication link in WBAN system by using identifier (biometrics) physiological signal. However, the static biometrics have some restrictions, i.e., biometrics cannot be replaced in the event even if it is lost or stolen during the recording of the physiological signals. Moreover, the physiological signals change significantly and are inaccessible. Therefore, dynamic biometrics is more secure with low lastingness. In ProxiMate [14] experimental prototype build using an open-source software platform that allows wireless devices to securely pair with one another autonomously by generating a common cryptographic key directly from amplitude and phase components. On the contrary, cryptography-based schemes [15–22] have some specialized restriction depending on hardware functionality and software or programming requirement for wearable sensors in WBAN.

In 1985, Miller and Kobiltz proposed Elliptic Curve Cryptography (ECC) mechanism in public key infrastructure, which has been used further as a prevalent tool to maintain the secrecy in WBANs [12, 23–28]. In 2016, Shen et al. [20] proposed a multilayer authentication protocol based on ECC which maintains the integrity, privacy, and valid information in WBAN. Where, the authentication has been established between personal digital assistant (PDA) and sensor and also between PDA and Application provider (AP).

In 2016, Zhao et al. [29] surveyed on Physiological valued based key agreement among the biosensor nodes. In the same year, Ibrahim et al. [30] also proposed a scheme called secure mutual authentication between the sensor node and the hub node. The author claims that it satisfies all the security requirements by performing the XORed operation and cryptographic hash function, only. However, later, it has been observed that the Ibrahim et al.'s scheme may suffer from key escrow problem, impersonation on the hub, sensor and blocking or congestion attack [1, 2].

Further, to overcome the weaknesses of Ibrahim et al.'s scheme, Li et al. [1] also proposed an enhanced scheme and built up a session key in an unknown and un-linkable session with more security functions. Besides, the authors exhibited that their scheme is energy efficient and has low power computational expense than other related existing schemes. But later, in 2018, Koya et al. [2] discovered that the Li et al.'s scheme suffers from sensor node impersonation attack. To overcome the shortcoming, the Koya et al. further proposed a hybrid authentication and key agreement scheme of the original scheme of Li et al.'s where drawback has been settled by using the physiological signals.

In 2018, [31] Kompara et al. surveyed on intra-body area network communication security in which they have classified the key agreement schemes into four types: old model, physical valued, hybrid key, and secret key agreement schemes. Consecutively, in 2019, [32] Kompara et al. proposed a scheme that evacuates the drawback like linkable to session and sensor node capture attacks of Koya et al.'s scheme. However, Kompara et al.'s scheme may also suffer from time synchronization issues. In the same year 2019, Konan et al. [33] also proved that Kompara et al.'s scheme has memory storage problems. In 2019, Xu et al. [34] proposed a scheme where he made guarantee to maintain the forward secrecy without asymmetric encryption. Recently 2020, Gupta et al. [35] proposed to enhance scheme of Koya et al.'s but it has been found that Gupta et al.'s suffers from higher computation costs and communication overhead. Abdullah et al. [36] proposed a secure anonymity guarantee preserving protocol for WBAN and defined two techniques namely, P-I for authentication, and P-II for re-authentication to increase efficiency.

However, regrettably, during our research, we found that Li. et al.'s [1] scheme still exist few flaws like Hub node impersonation attack, linkable session, sensor node capture, impersonation, and eavesdropping attacks. Hence, to overcome such issues, we have designed an efficient mutual authentication and symmetric key agreement scheme and succeeded to reduce the overall complexity. The main key features of our scheme are to avoid the use of timestamp, verification of sensor node identity at the Hub node end, and resolves the security functionality of Li. et al. [1].

3 Brief Evaluation of Li et al.'s Scheme

The Li et al.'s scheme in WBAN [1] comprises of three phases: namely, the initialization phase, sensor node registration, and the authentication and key agreement phase. There are three types of nodes: first-level node (I_n), second-level node (S_n), and local server ($HNode$). The first-level node is the intermediary node (e.g., smartphone, smartwatch) which gathers the information from second-level nodes. However, they have a higher processing power, storage capacities, and higher capabilities of battery power, whereas second-level nodes are resource-constrained. The hub node or local server

is a powerful node that connects to healthcare service providers. The network type is illustrated in Fig. 1. In Li et al.'s scheme, system administrator (*SAdmin*) performs initialization and registration phase in a protected communication network, while the authentication phase is carried out in the unprotected network. In the initialization phase, *SAdmin* sets up *HNode*, register S_n and I_n. Mutual authentication and key agreement are performed between the S_n and *HNode*, through intermediate node I_n. we have utilized the notations in the scheme as summarized in Table 1.

The detailed description of the Li et al.'s scheme is defined below:

3.1 Registration Phase

Initially,*SAdmin* creates the master key MK_{hn} for *HNode* and *SAdmin* configures the S_n by assigning identity Sid_n, secret key of S_n as NK_n for each nodes and computes $P_n = Sid_n \oplus h(MK_{hn}||NK_n)$ and $Q_n = MK_{hn} \oplus P_n \oplus NK_n$. Whereas intermediate node I_n chooses his single identity $Iid_{in}^{'}$ by himself. The *SAdmin* stores the tuple $< Sid_n, P_n, Q_n >$ onto S_n and also *HNode* stores all $Iid_{in}^{'}$ for each of I_n node. The secret key NK_n of each node is not kept on any of the devices except used for computing P_n and Q_n parameters.

Table 1 Notation and Meaning

Notation	Meaning of the Notation
SAdmin	System Administrator
S_n	Sensor node
HNode	Hub or Center Node
I_n	Intermediator of *HNode* and S_n
Sid_n	S_n secret identity
$Iid_{in}^{'}$	I_n identity
tid_n	S_n identity used for temporary
MK_{hn}	*HNode* secret master key
NK_n, F_n	*HNode* create a secret parameter for S_n which is temporary
R_n	S_n create secret temporary parameters
IN_{in}	Secret random parameters for I_n
P_n, Q_n, X_n	Parameters used for authentication
Y_n	Auxiliary parameters required for authentication
β	Integrity parameters
α, η, μ	Parameters used by *HNode* to authenticate S_n
γ	Parameter constructed from temporary secrets
t_n	Timestamp used by both *HNode* and S_n
K_s	Shared session key of *HNode* and S_n
$h(.)$	Cryptography Hash
||	Concatenation operation
⊕	Cryptography XOR operation
$Varible^*$	Variable has been computed without checking the integrity
$Variable^+$	The + parameters used for next authentication phase

3.2 Authentication and Key Agreement

Figure 2 demonstrates the authentication and key agreement scheme of Li et al.'s. The S_n anonymously authenticates *HNode* through the help of I_n as follows:

Step 1: Node S_n picks R_n and produces a timestamp t_n. After that, S_n computes $X_n = P_n \oplus Sid_n$, $Y_n = X_n \oplus R_n$, the temporary identity $tid_n = h(Sid_n \oplus t_n||R_n)$ and sends the parameters $< tid_n, Y_n, P_n, Q_n, t_n >$ to I_n.

Step 2: I_n forward the parameters without any modification to *HNode* by putting I_n's identity,$Iid_{in}^{'}$.

Step 3: On the receiver side, *HNode* receives the parameters $< tid_n, Y_n, P_n, Q_n, t_n, Iid_{in}^{'} >$ and performs the operations as follows.

- *HNode* verifies the $Iid_{in}^{'}$ in its database to find whether it is present or not. If not, the authentication procedure stops or aborts. Apart from $Iid_{in}^{'}$ verification, *HNode* also finds the strength of timestamp t_n by checking the strength of the predicate $(t^* - t_n < \Delta t)$. Where t^* is the time when the message is received and Δt is the maximum transmission delay. Otherwise, terminate the entire process for authentication, if time is not within the given Δt.
- Further *HNode* computes, $NK_n^* = MK_{hn} \oplus P_n \oplus Q_n$, $X_n^* = h(MK_{hn}||NK_n^*)$, $Iid_n^* = X_n^* \oplus P_n$ and $R_n^* = X_n^* \oplus Y_n$, $tid_n^* = h(Iid_n^* \oplus t_n||R_n^*)$.
- Verifies, $tid_n = ?tid_n^*$. Terminates if the computed value is not matched or fails.
- Picks F_n and computes $\alpha = X_n \oplus F_n$ and $\gamma = R_n \oplus F_n$
- *HNode* picks a new secret key NK_n^+ and perform new $P_n^+ = Sid_n \oplus h\left(MK_{hn}||NK_n^+\right)$, $Q_n^+ = MK_{hn} \oplus P_n^+ \oplus NK_n^+$,$\eta = \gamma \oplus P_n^+, \mu = \boldsymbol{\gamma} \oplus Q_n^+, \beta = h\left(X_n\middle|\middle|R_n\middle|\middle|F_n||\eta|\,|\mu\right)$, and computes session key $K_s = h\left(Sid_n||R_n||F_n||X_n\right)$ and forwards $< \alpha, \beta, \eta, \mu, Iid_{in}^{'} >$ to I_n.

Step 4. I_n drops her identity $Iid_{in}^{'}$ and just forwards the rest of parameters $< \alpha, \beta, \eta, \mu >$ to S_n

Step 5. S_n on receiving the parameters $< tid_n, \alpha, \beta, \eta, \mu >$ performs as follows.

- Computes $F_n^* = X_n \oplus \alpha$, $\beta^* = h(X_n||\left|R_n\right|\,|F_n^*||\eta|\,|\mu)$ and checks $\beta = ?\beta^*$. Terminate if it fails.
- Computes $\gamma = R_n \oplus F_n$, $P_n^+ = \gamma \oplus \eta$, $Q_n^+ = \gamma \oplus \mu$ and the session key $NK_n^*\left(= K_s\right) = h(Sid_n, X_n, R_n, F_n)$ is stored for further secret communication. Change the parameters P_n, Q_n with the parameters P_n^+, Q_n^+ in its memory.

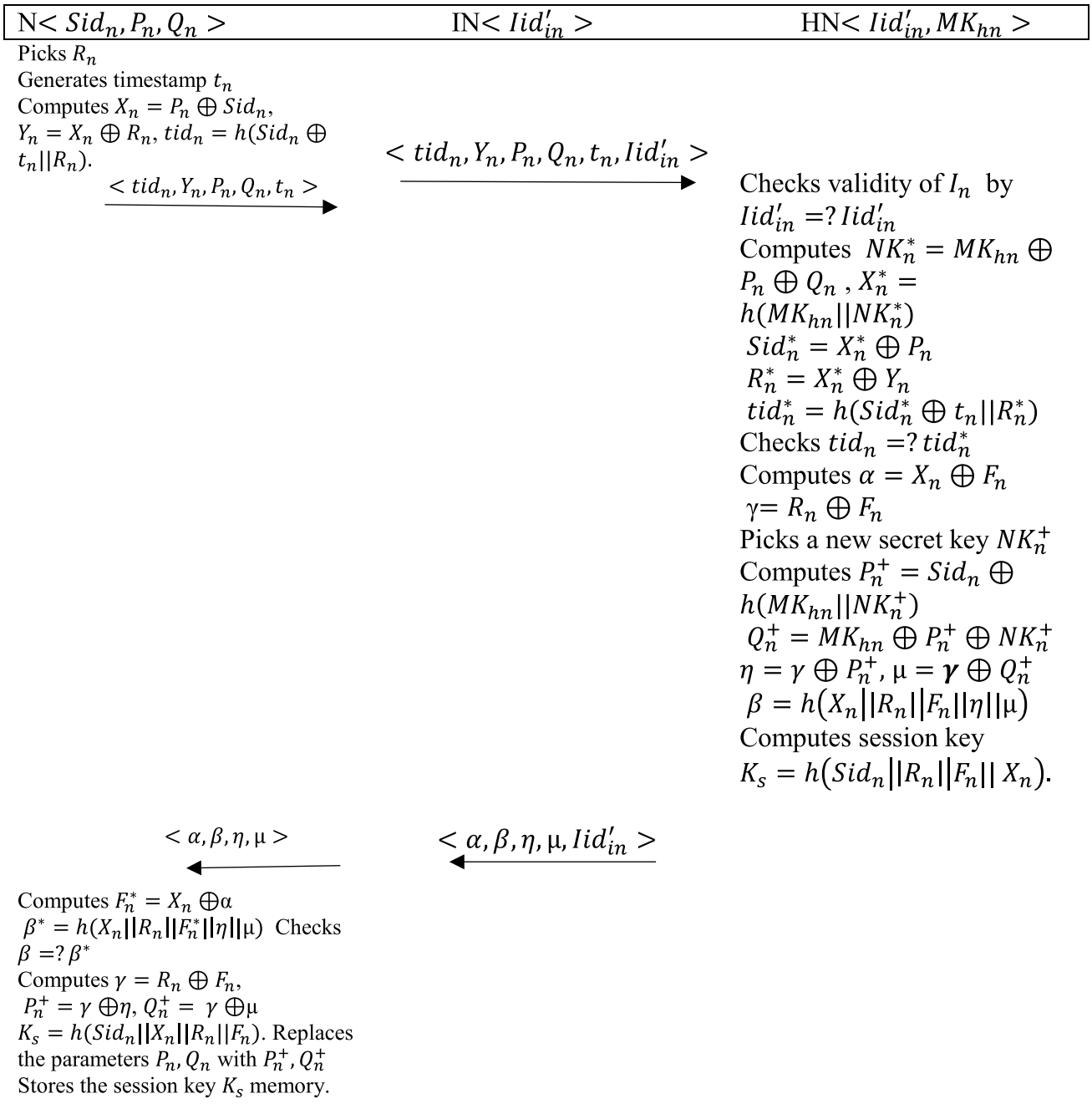

Fig. 2 Two-hop centralized WBAN authentication and key agreement protocol [1]

3.3 Cryptanalysis of the Li et al.'s Scheme

Li et al. claims that their scheme achieves an anonymous mutual authentication and key agreement. In contradict, we oppose that information sent between the S_n and *HNode* are not secured against the sensor node capture attack, S_n and *HNode* impersonation attack, linkable to session communication between S_n and *HNode*, and eavesdropping attack.

3.3.1 Linkable to Session

The scheme sends the secret parameters (R_n, NK_n, F_n) in the form of Y_n, Q_n and α respectively, over the public channel for authentication purpose by S_n and *HNode*. Therefore, the attacker may intercept the communication between S_n and *HNode*, and get messages $< tid_n, Y_n, P_n, Q_n, t_n, Iid_{in}^{'} >$ and $< \alpha, \beta, \eta, \mu, Iid_{in}^{'} >$. Hence, the attacker gets access the value of γ easily just by performing $Y_n \oplus \alpha$, while underlying

secret values remain unknown to him. However, it is sufficient to extract the parameters P_n^+ and Q_n^+, by computing $P_n^+ = \eta \oplus \gamma$ and $Q_n^+ = \mu \oplus \gamma$. Whereas the main purpose of these values is for using in next authentication and key agreement process. Therefore, an attacker captures successive authentication messages can consequently connect this session to a single sensor node. Hence, the attacker can effortlessly link the session between the S_n and *HNode*.

3.3.2 Sensor Node Capture Attack

In this kind of attack, the attacker may compromise any of the sensor node S_n in the WBAN and after extracting the stored parameters, can perform the various operation on network and finally can compromise the entire network easily. In Li, et al.'s scheme, the main reason for not sustaining against node capture attack is because of sensor's identity Sid_n which is not stored in *HNode* to check the legitimacy. Once the identity of S_n is impersonate, adversary removes the original Sid_n by performing XORed of P_n and Q_n and embed a new identity Sid_n^{new}, P_n^{new} and Q_n^{new}. As we know that *HNode* does not verify authenticity of the sensor node by the validating identities with the received identitySid_n, therefore, the adversary can change the Sid_n for unlimited times. Hence, the scheme does not resist against sensor node capture attack.

3.3.3 Use of Timestamps

Timestamp-based protocol experiences time synchronization issues and are expensive too [37, 38]. The estimation of these timestamps' starting with a one-time zone then onto the next time zone, for example, S_n to *HNode*. The message arrived at the receiver side must be within valid timestamp or trusted nodes for authentication. Even if the slight change in time, the whole scheme will break down. Here, no confirmation or validation is possible when the timestamp is lost while transferring through a dubious channel. Hence, to overcome the synchronization problem, use of fresh random number is always recommended.

3.3.4 Eavesdropping Attack

The adversary can take an advantage by sniffing or eavesdropping the messages sent over the public channel like $\eta, \alpha,$ *and* μ. The attacker stores η *and* μ values by performing XORed operation. For every new authentication, it is required to update P_n and Q_n with newR_n,F_n andSid_n. During the update phase, the adversary may eavesdrop to perform a reply attack. Similarly, the Koya et al. [2] suffers the similar problem$\eta \oplus \mu = P_n \oplus Q_n$.

4 Proposed Scheme

In this section, we have proposed an enhanced authentication and key agreement scheme which removes the securities pitfalls of Li et al.'s [1] schemes. The scheme increases the efficiency of sensor nodes in terms of, communication overhead and computational complexity. In proposed scheme, we have considered that the Hub node can never be captured or negotiated by an adversary because compromising Hub node means the entire network will break down [34]. Thus, we consider that the database (DB) is protected from database security threats, and the administrator gives the privilege to legitimate sensors only to access DB because it is required to be updated periodically [46]. Hence, Hub node is trusted and it will not maltreat the encryption keys of the authorized users or the keys of sensor nodes shared among them. The proposed scheme is alike to Li et al.'s scheme; it includes *HNode* be trustworthy and protected. The notations used for our scheme is the same as the original scheme. The scheme consists of four phases i.e., initialization phase, registration phase, authentication and key agreement phase, and sensor node addition phase. The *SAdmin* performs the initialization and registration before the authentication. The phases are as follows:

4.1 Initialization Phase

In this phase, *SAdmin* initializes the S_n, I_n and *HNode* in offline mode. The following are the steps involved:

Step 1: Generates a master key MK_{hn} for the *HNode*.
Step 2: Secret key of S_n, NK_n, is stored in *HNode* for further authentication.
Step 3: Generates unique identity Iid'_{in} of Intermediate node (I_n) and stores in I_n memory.

4.2 Registration Phase of S_n

SAdmin performs the following tasks to register S_n as follows.

Step 1: SAdmin chooses a secret identity Sid_n for each sensor node S_n and saves in *HNode* memory.
Step 2: SAdmin computes $P_n = h(MK_{hn}||NK_n) \oplus h(Sid_n)$ and $Q_n = MK_{hn} \oplus NK_n \oplus Sid_n$
Stores the $< Sid_n, P_n, Q_n >$ in S_n memory.

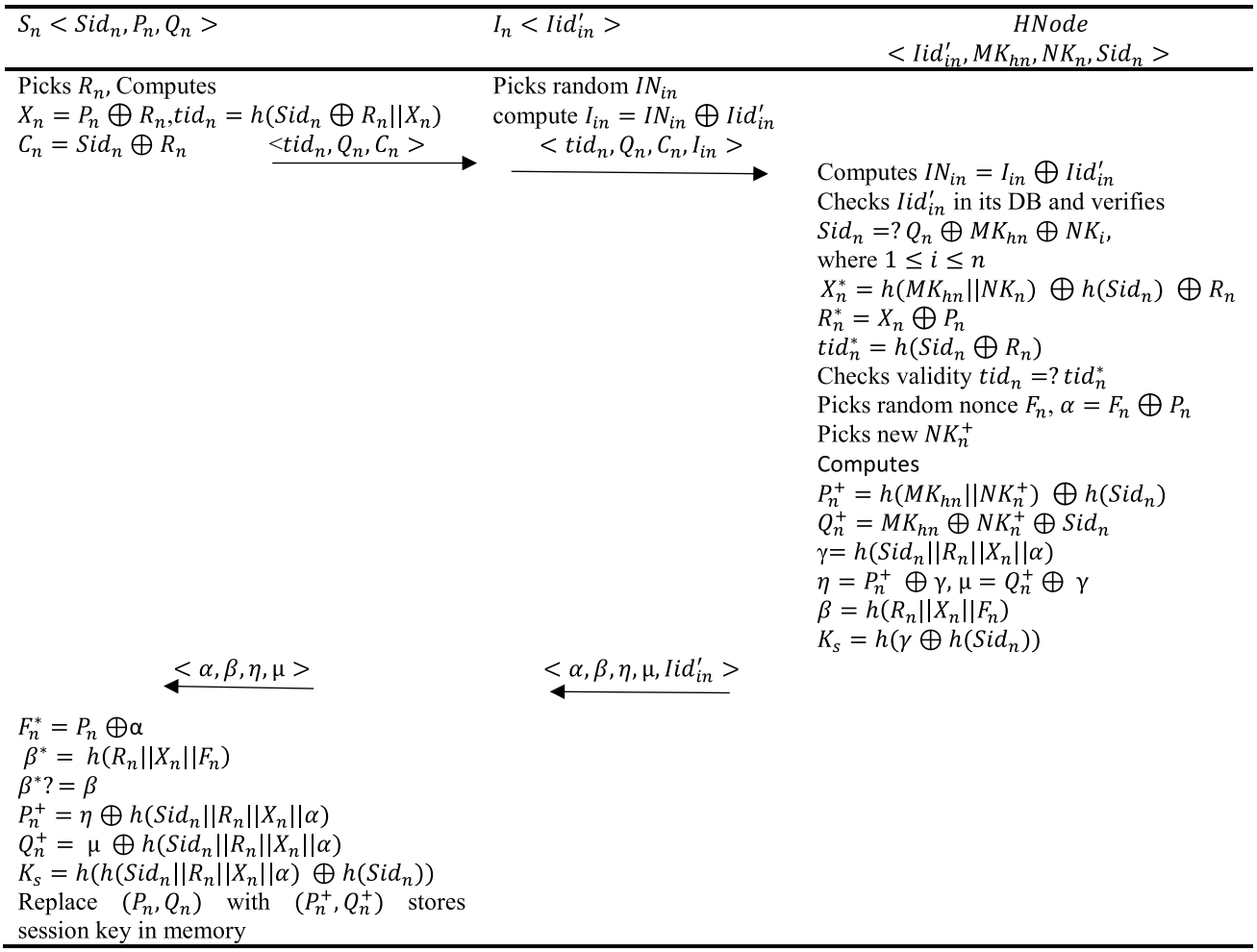

Fig.3 Authentication and Key agreement phase

4.3 Authentication and Key Agreement Phase

Figure 3 illustrates the authentication and key agreement phase of sensor node, intermediate node and hob node and detailed descriptions are given below:

Step 1: $S_n \rightarrow I_n$: S_n computes the following values.

Picks R_n, Computes $X_n = P_n \oplus R_n$, $tid_n = h(Sid_n \oplus R_n||X_n)$, *and* $C_n = Sid_n \oplus R_n$. Later, forward the $<tid_n, Q_n, C_n >$ to I_n.

Step 2: $I_n \rightarrow S_n$: I_n computes the following values.

Picks random IN_{in}, computes $I_{in} = IN_{in} \oplus Iid'_{in}$ and forward the parameters $< tid_n, Q_n, C_n, I_{in} >$ to *HNode* without any modification in received parameters apart from appending I_{in} which is computed by I_n.

Step 3: *HNode* computes the following parameters to validate the legitimate user and session key generation.

Computes $IN_{in} = I_{in} \oplus Iid'_{in}$, checks the identity of intermediate node Iid'_{in} in its DB. Similarly, the legitimate sensor identity is also checked with all the stored values of NK_n and comparing $Q_n \oplus MK_{hn} \oplus NK_n$ with Sid_n in its DB to validate the sensor (S_n). If, it will be not matched in *HNode* DB, then the entire process will be aborted. Otherwise, it further computes $NK_n \oplus Sid_n = Q_n \oplus MK_{hn}$, $X_n^* = h(MK_{hn}||NK_n) \oplus h(Sid_n) \oplus R_n$, $R_n^* = X_n \oplus P_n$, $tid_n^* = h(Sid_n \oplus R_n)$. Again check the $tid_n = ? tid_n^*$, if the values are not same, it aborts the process. If it is valid then picks random nonce F_n and new NK_n^+. Further, Computes $= F_n \oplus P_n$ $P_n^+ = h(MK_{hn}||NK_n^+) \oplus h(Sid_n)$, $Q_n^+ = MK_{hn}^+ \oplus NK_n^+ \oplus Sid_n$ $\gamma = h(Sid_n||R_n||X_n||\alpha)$,

$\eta = P_n^+ \oplus \gamma, \mu = Q_n^+ \oplus \gamma, \beta = h(R_n||X_n||F_n)$. Finally, the new session key $K_s = h(\gamma \oplus h(Sid_n))$ is computed. Later, *HNode* forward the parameters $< \alpha, \beta, \eta, \mu, Iid'_{in} >$ to I_n for further processing and key agreement.

Step 4: Once I_n receives the parameters $< \alpha, \beta, \eta, \mu, Iid'_{in} >$ from *HNode* then it forwards $< \alpha, \beta, \eta, \mu >$ to sensor node S_n.

Step 5: After receiving, S_n performs the following:

Computes, $F_n^* = P_n \oplus \alpha$ and $\beta^* = h(R_n||X_n||F_n)$, and verifies $\beta^* ? = \beta$. If verified successfully, it computes $P_n^+ = \eta \oplus h(Sid_n||R_n||X_n||\alpha)$ and the session key

$K_s = h(h(Sid_n||R_n||X_n||\alpha) \oplus h(Sid_n)$, otherwise, terminate session. And finally, replace (P_n, Q_n).

4.4 Sensor Node Addition

In this phase, new node can be added in targeted region of WBAN when the sensor node is depleted because of intensity utilization issue or physically trapped by an adversary from the patient body or required new sensor to sense some data. Therefore, it is needed to add new sensors dynamically into WBAN. When new wearable sensor S_n^{new} enters to the current network, the system administrator deploys the new node by performing the system set up phases in offline mode. The steps to perform the addition of new sensor S_n^{new} are given below:

Step 1: *SAdmin* assigns a unique identity Sid_n^{new} and secret key NK_n^{new} for new sensor node and stores these in *HNode*.

Step 2: *SAdmin* further computes $P_n^{new} = h(MK_{hn}||NK_n^{new}) \oplus h(Sid_n^{new})$ and $Q_n^{new} = MK_{hn} \oplus NK_n^{new} \oplus Sid_n^{new}$.

Step 3: At the end, sensor node stores the, $\langle Sid_n^{new}, P_n^{new}, Q_n \rangle$ in S_n^{new}'s memory.

Hence, the addition of new S_n^{new} can be done as similar to initialization or setup phase of our proposed scheme in WBAN.

5 Security Study of Our Proposed Scheme

In this section, the security analysis of our proposed scheme has been evaluated. The security study brought out certain flaws in Li et al.'s scheme in which we have defeated in this proposed work. The complete analysis of our scheme is given below:

5.1 Informal Security Analysis

In this subsection, we have analyzed the scheme in an informal method to prove that the proposed scheme resists against modern attacks.

5.1.1 Resistance Against Eavesdropping Attack

According to Dolev-Yao threat model [39], an attacker can impersonate all the messages sent over an insecure channel. If the attacker collects all the parameters $tid_n, Q_n, C_n, \alpha, \beta, \eta, and\ \mu$ even then it would be infeasible to construct any of the secret parameters. The secret value $tid_n = h(Sid_n \oplus R_n||X_n)$, $P_n = h(MK_{hn}||NK_n) \oplus h(Sid_n)$, *and* $\beta = h(R_n||X_n||F_n)$ is secured by non-reversible one-way hash function $h(.)$ and unlike Li et al.'s scheme the secret parameter P_n is never been shared in our proposed scheme. Moreover, for attacker, it would be difficult to know the identity of the sensor node as identity is protected with the hash function, secret value, and XORed with the random nonce. Hence, it would be difficult for an attacker to get the session secret key $K_s = h(\gamma \oplus h(Sid_n))$.

5.1.2 Resistance Against Anonymous and Unlikabilities

The main objective of the attacker is to get the services by generating fraud authentication request and/or intercepting the communication link. While communicating between S_n and *HNode*, the messages $tid_n, Q_n, C_n, \alpha, \beta, \eta, and\ \mu$, are shared through the public network. Where, the temporary identity of the sensor node, $tid_n = h(Sid_n \oplus R_n||X_n)$, contains fresh random values for each session and Sid_n is protected from the hash. Moreover, unlike Li et al.'s scheme, the Sid_n is also checked in *HNode* to verify legitimate sensor node or intruder node for further computation. Therefore, an attacker cannot trace the valid Sid_n for linking purpose. For every session, there is a different random value F_n and $\alpha = F_n \oplus P_n$ are performed at *HNode*. During the authentication and key agreement, two links cannot be together because the sent parameters contain the fresh, secret, and random values every time. As we know that randomly selected parameters cannot figure out by an attacker to accomplish a fixed parameter. Therefore, the communication parameters are fresh, secret, and random that conducted for an alternate session. So, an attacker cannot establish a two-link or more sessions to the same node S_n.

5.1.3 Resistance Against SENSOR Node Impersonation Attack

In this attack, the attacker is able to create legitimate tuple $< tid_n, P_n, Q_n >$ to prove himself as a legitimate sensor on behalf of original one. Therefore, in our proposed scheme, the attacker can listen to the message shared between two entities, but unable to create valid Sid_n as the temporary identity of the sensor is shielded by the one-way hash function. If the attacker compromises any sensor S_n parameters, still the attacker cannot disclose the master key MK_{hn} and NK_n as it is ensured by hash. Hence, we can conclude that the scheme is protected against S_n impersonation attack. Where, Koya et al.'s [2] scheme used the Bio-key to prevents sensor node impersonation attack.

5.1.4 Resistance Against Hub Node Impersonation Attack

In our scheme, we have assumed that Hub node can never be captured or negotiated by an adversary because compromising Hub node means the entire network will break down. Hence, Hub node impersonation attack is possible only if the attacker able to retrieve the valid tuples $\langle \beta, \eta, \mu \rangle$ of *HNode*. The *HNode*'s master key MK_{hn} and temporary secret key of S_n, i.e., NK_n are known to only the *SAdmin*. Even if the attacker

captures the communicated parameters, still it is infeasible for an attacker to get the master key MK_{hn} and secret key NK_n, as both the keys are secured from one-way hash.

5.1.5 Resistance Against Replay Attack

In a replay attack, the attacker tries to fool both S_n and *HNode* by using previous transmitted messages or get the valid authentication request message and resends it into the network. To avoid replay attack in proposed protocol, during authentication, *HNode* sends parameters $<\alpha, \beta, \eta, \mu>$ to S_n which shows that for each new session, protocol uses freshness and random values to create a new session $K_s = h\big(\gamma \oplus h(Sid_n)\big)$ every time. Hence, we can claim that our scheme resists against a replay attack**.**

5.1.6 Resistance to Forward / Backward security

In this feature, by knowing the session key K_s of any session, the privacy of any past or future session must not to be revealed to the adversary or not influence by the adversary. In our scheme, the session is figured out using the values $Sid_n, R_n, X_n, \alpha, P_n$ and Q_n. In session key $K_s = h\big(\gamma \oplus h(Sid_n\big))$, all the parameters are secure by one-way hash function and also γ is constructed using random nonce and freshness values for each new session (P_n^+, Q_n^+). Therefore, knowing K_s does not reveal any qualities or values for generating the other session keys.

5.1.7 Resistance Against Man-in-the-Middle Attack

In this attack, the attacker alters the communication between two-parties and make both the party believe that they are exchanging the message with each other without any modification, actually the attacker is in middle. In our scheme, if the attacker captures the message parameters $<tid_n, Q_n, C_n>$ sent by S_n to I_{in} and impersonate the intermediate identity $< Iid'_{in} >$ even then the attacker cannot perform the Man-in-the-middle attack because the *HNode* database stores all the registered intermediate identity. Therefore, new proposed scheme resists against man-in-the-middle attack.

5.1.8 Resistance Against Denial-of-Service Attack (DOS) or Jamming Attack

In this scheme, we do not use the timestamps instead we use the random numbers only. The XORed, concatenation, and hash function $h(.)$ are used in every computation of parameters. The adversary has the power to capture the $\langle \alpha, \beta, \eta, \mu \rangle$ parameters but never be able to extract the master key K_{hn}, secret key NK_n and identity of a legitimate user Sid_n. Because to process further, the sensor's Sid_n is checked in *HNode*. *HNode* is trustworthy that any unauthorized user cannot be compromised. Hence, our scheme resists the DOS attacks and also adversary cannot perform as a legitimate sensor.

5.1.9 Resistance Against Capture Sensor Node Attack

To play the sensor node capture attack, the adversary must reveal the real user values $<tid_n, P_n, Q_n>$. To know those values the adversary needs to find sensor personal identity Sid_n and X_n which is impossible for the adversary because the identity of the sensor node is checked in the *HNode* of DB. If the sensor node is captured by the adversary, even then for the attacker it is hard to compute the master key MK_{hn} because the MK_{hn} is shielded by randomness and one-way hash. Here, even if the adversary captures n th numbers of S_n the attacker cannot get any advantages. Therefore, our proposed scheme resists against the node capture attack.

5.1.9.1 Resistance Against Fault Node Addition or SCALABILITY The scalability of our proposed protocol is guaranteed when the network remains non-degrade and maintains the security of the system during the joining of a new node or removing a node from the system. The proposed scheme removes the unauthorized nodes or illegitimate nodes from the system as the hub node checks the sensor identity in the hub node database whether it is registered in-network or not. If register, then only register sensors node allows for the session and discard illegitimate nodes. The proposed scheme even achieves scalability by reducing the communication cost, memory overhead, and most of the security functionalities. Hence, proposed protocol achieves the efficiency and better scalability than other related schemes [1, 2].

5.1.9.2 Resistance Against Ephemeral Secret Key Leakage In this attack, attacker compromises the private keys of sensors and the session key from eavesdropped messages. In our proposed scheme, to achieve authentication between sensor node and hub node, The following parameters are needed to compute, $X_n^* = h(MK_{hn}||NK_n) \oplus h\big(Sid_n\big) \oplus R_n$, and $R_n^* = X_n \oplus P_n$, $tid_n^* = h\big(Sid_n \oplus R_n\big)$. Again check the $tid_n = ?tid_n^*$, if their value is not same it aborts the process. if it is valid, picks random nonce F_n, and new NK_n^+. Further, $\alpha = F_n \oplus P_n$, $P_n^+ = h(MK_{hn}||NK_n^+) \oplus h(Sid_n)$, $Q_n^+ = MK_{hn}^+ \oplus NK_n^+ \oplus Sid_n$, $\gamma = h(Sid_n||R_n||X_n||\alpha), \eta = P_n^+ \oplus \gamma$, $\mu = Q_n^+ \oplus \gamma$, $\beta = h\big(R_n||X_n||F_n\big)$ and key $K_s = h\big(\gamma \oplus h(Sid_n)\big)$ are computed. Since it would be also difficult to construct secret parameters of sensor node secret key NK_n and F_n as it is temporary which is stored in Hub node. Once the mutual authentication is done, secret key and random nonce are discarded. Therefore, attacker cannot perform Ephemeral secret key leakage even if the message is eavesdropped.

5.1.9.3 Resistance Against Hub Node Stolen Database Attack In proposed scheme, we have considered the Hub node can never be captured or negotiated by a foe since compromising the Hub node means the whole system will pause down [34]. Even, in our scheme considered database is protected from database security threats, and only the administrator gives the privilege to genuine sensors to contact database as database is updated periodically [46]. Hence, Hub node is trusted and it will not harm the encryption keys of the authorized users or the keys of sensor nodes shared between them.

5.2 Security Analysis Using BAN-Logic

BAN logic [1, 37, 38] is used to verify the mutual authentication and key agreement between the S_n and $HNode$. To prove our scheme and to demonstrate a secure mechanism, the following are the four goals we need to prove using BAN-logic.

5.2.1 Basic Notation

The basic notation used in BAN logic [1, 40, 41] is listed below.

- $C|\equiv D$: C believes if D is true.
- $C \lhd D$: C sees D, D may be the data or messages which can be read by C and repeats D.
- $C|\sim D$: C said D, C sent a data including D, in this logic C does not know the current data send or past data, the logic concludes C believes D.
- $C|=\rangle D$: C control or jurisdiction over D, in this logic, C has the authority and trusted the quality of message or data.
- $\#(D)$: D is fresh, the logic said D is fresh, not used before for any data or authentication.
- $<D>_{Key}$: D is combined with E
- $C \overset{E}{\leftrightarrow} F$: the secret key shared between C and F which is only known to both.

5.2.2 Inference Rules

There are five rules of BAN logic [40, 41] need to prove to show the efficiency of our proposed scheme.

R_1: [Message meaning rule]: $\frac{C|\equiv C \overset{E}{\leftrightarrow} F, C \lhd \langle D \rangle_E}{C|\equiv F|\sim D}$.

R_2: [Nonce-verification rule]: $\frac{C|\equiv \#(D),\ C|\equiv F|\sim D}{C|\equiv F|\equiv D}$.

R_3: [Jurisdiction rule]: $\frac{C|\equiv F|\Rightarrow D,\ C|\equiv F|\equiv D}{C|\equiv D}$.

R_4: [Freshness-conjuncatenation rule]: $\frac{C|\equiv \#(D)}{C|\equiv \#(D,E)}$.

R_5: [Belief rule]: $\frac{C|\equiv (C,E)}{C|\equiv \#(C)}$.

5.2.3 Assumption

A1: $HNode|\equiv \left(S_n \overset{Sid_n}{\leftrightarrow} HNode \right)$.

A2: $HNode|\equiv \#(F_n)$.

A3: $HNode|\equiv S_n|=\rangle \left(S_n \overset{X_N}{\leftrightarrow} HNode \right)$.

A4: $S_n|\equiv S_n| \equiv \left(S_n \overset{Sid_n}{\leftrightarrow} HNode \right)$.

A5: $S_n|\equiv \#(R_n)$.

A6: $S_n|\equiv HNode|=\rangle \left(S_n \overset{K_s}{\leftrightarrow} HNode \right)$.

5.2.4 Goal

Goal_1: $HNode|\equiv S_n| \equiv \left(S_n \overset{X_N}{\leftrightarrow} HNode \right)$.

Goal_2: $HNode|\equiv \left(S_n \overset{X_N}{\leftrightarrow} HNode \right)$.

Goal_3: $S_n|\equiv HNode| \equiv \left(S_n \overset{K_s}{\leftrightarrow} HNode \right)$.

Goal_4: $S_n|\equiv \left(S_n \overset{K_s}{\leftrightarrow} HNode \right)$.

5.2.5 Message

Message 1: $S_n \rightarrow HNode : < S_n \overset{X_n}{\leftrightarrow} HNode, R_n >_{S_n \overset{Sid_n}{\leftrightarrow} HNode}$.

Message 2: $HNode \rightarrow S_n : < S_n \overset{X_n}{\leftrightarrow} HNode, R_n, NK_n^+, F_n, S_n \overset{K_s}{\leftrightarrow} HNode >_{S_n \overset{Sid_n}{\leftrightarrow} HNode}$.

5.2.6 Formal Verification of Proposed Scheme Using BAN Logic Rules

To achieve the **Goal_1 to Goal_4**, we need to prove following steps.

Step_1: Applying message meaning rule, from 5.2.4 message 1, and assumption A1, we assume

$$\frac{HNode| \equiv (S_n \overset{Sid_n}{\leftrightarrow} HNode, HNode \lhd \left\langle S_n \overset{X_n}{\leftrightarrow} HNode, F_n \right\rangle_{S_n \overset{Sid_n}{\leftrightarrow} HNode}}{HNode|\equiv S_n| \sim \left(S_n \overset{X_n}{\leftrightarrow} HNode, F_n \right)}$$

Step_2: Applying the freshness rule, and assumption A2, we assume

$$\frac{HNode| \equiv \#(F_n)}{HNode| \equiv \#\left(S_n \overset{X_n}{\leftrightarrow} HNode, F_n \right)}$$

Step_3: Putting the Nonce-verification rule, Step_1 and Step_2, we can assume

$$\frac{S_n| \equiv \#\left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+, \end{matrix} \begin{matrix} Sid_n \\ S_n \leftrightarrow HNode \end{matrix}\right), S_n| \equiv HNode| \sim \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}{S_n| \equiv HNode| \equiv \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}$$

$$\frac{HNode| \equiv \#\left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode, F_n \end{matrix}\right), HNode| \equiv S_n| \sim \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode, F_n \end{matrix}\right)}{HNode| \equiv S_n| \equiv \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode, F_n \end{matrix}\right)}$$

Step_4: Applying belief rule and step_3, we can assume.

$$\frac{HNode| \equiv S_n| \equiv \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode, F_n \end{matrix}\right)}{HNode| \equiv S_n| \equiv \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode \end{matrix}\right)} \textbf{Goal\_1}$$

Step_5: Applying jurisdiction rule, A3 and Step_4, we assume.

$$\frac{HNode| \equiv S_n| \Rightarrow \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode \end{matrix}\right), HNode| \equiv S_n| \equiv \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode \end{matrix}\right)}{HNode| \equiv \left(\begin{matrix} X_n \\ S_n \leftrightarrow HNode \end{matrix}\right)} \textbf{Goal\_2}$$

Step_6: Applying message meaning rule, message 2 and assumption A4, we assume

$$\frac{S_n| \equiv (S_n \overset{Sid_n}{\leftrightarrow} HNode, S_n \lhd \left\langle \begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix} \right\rangle_{S_n \overset{Sid_n}{\leftrightarrow} HNode}}{S_n| \equiv HNode| \sim \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}$$

Step_7: Applying freshness rule and A5, we assume

$$\frac{S_n| \equiv \#(R_n)}{S_n| \equiv \#\left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}$$

Step_8: Applying nonce verification rule, Step_6 and Step_7, we can assume

$$\frac{S_n| \equiv \#\left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right), S_n| \equiv HNode| \sim \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}{S_n| \equiv HNode| \equiv \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}$$

Step_9: Applying belief rule and step_8, we can assume.

$$\frac{S_n| \equiv HNode| \equiv \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}{S_n| \equiv HNode| \equiv \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode \end{matrix}\right)} \textbf{Goal\_3}$$

Step_10: Applying jurisdiction rule, A6 and Step_9, we assume.

$$\frac{S_n| \equiv HNode| \Rightarrow \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode \end{matrix}\right), S_n| \equiv HNode| \equiv \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode, F_n, R_n, X_n, P_n^+ \end{matrix}\right)}{S_n| \equiv \left(\begin{matrix} K_s \\ S_n \leftrightarrow HNode \end{matrix}\right)}$$

Goal_4.

The above steps prove that our scheme reach all the goal. Hence, the mutual authentication and key agreement between the *HNode* and S_n is proved.

5.3 Security Analysis Using ProVerif Simulation Tool

In this subsection, we have proved the authentication and session secrecy of our proposed scheme using the simulation tool called ProVerif [42, 43]. The authenticity of all nodes S_n, *HNode* and I_n is verified. The detailed description of our proposed scheme is defined in Tables 2, 3, 4, 5, 6, 7, 8 . Here, we use two channels Ch1 for S_n and Ch2 for *HNode* through which communication is done.

5.3.1 Output of the Proposed Scheme

ok, secrecy assumption verified: fact unreachable attacker(Idin'[])

ok, secrecy assumption verified: fact unreachable attacker(Rn[])

ok, secrecy assumption verified: fact unreachable attacker(khn[])

ok, secrecy assumption verified: fact unreachable attacker(kn[])

Table 2 Types of variables, XOR, Equation and channels declaration

```
(*types of variable declaration*)
type key[private].
type nonce.
type sensor.
type intermediate_node.
type Hub.
(*channel*)
free ch1: channel. free ch2: channel.
(*cryptography function*)
fun XOR(bitstring,bitstring):bitstring.
fun XORS(bitstring,nonce):bitstring.
fun XORT(bitstring,key):bitstring.
fun XORU(key,key):bitstring.
fun XORV(nonce,bitstring):bitstring.
fun XORW(key,bitstring):bitstring.
reduc forall x:bitstring,y:bitstring;
XORagain(XOR(x,y),y)=x.
```

ok, secrecy assumption verified: fact unreachable attacker(SIdn[])

ok, secrecy assumption verified: fact unreachable attacker(KEY[])

RESULT not attacker(khn[]) is true.

RESULT not attacker(kn[]) is true.

RESULT not attacker(SIdn[]) is true.

RESULT not attacker(KEY[]) is true.

RESULT not attacker(Idin'[]) is true.

RESULT not attacker(Rn[]) is true.

Starting query inj-event(HNAccept(IN_46,N)) = = > inj-event(INAccept(HN))

RESULT inj-event(HNAccept(IN_46,N)) = = > inj-event(INAccept(HN)) is true.

Starting query inj-event(HNAccept(IN_48,N_47)) = = > inj-event(sensorAcceptS(HN_49,IN_48))

RESULT inj-event(HNAccept(IN_48,N_47)) = = > inj-event(sensorAcceptS(HN_49,IN_48)) is true.

Output: To test identity, random, secret key, and privacy, we test the "Query not attacker" the results is true which indicates the all secret parameters not derivable by the adversary. The "Injective Correspondence" shows the one-to-one relationship of authentication. The event "event" shows that Intermediate node, Hub Node, and sensor node accept to run the protocol.

6 Performance evaluation and comparisons

In this section, we compared our scheme with other related schemes i.e., Li et al. [1], Koya et al. [2], Ibrahim et al. [30], Kompara et al. [32], Xu et al. [34], Gupta et al. [35], Almuhaideb et al. [36] and Sowjanya et al. [44]. The performance analysis is done based on security functionality, communication cost, computation cost, and memory overhead. Table 9 illustrates the types of security aspect that prevents various related schemes.

6.1 Security Functionality of Various Existing Schemes and Our Scheme

Table 9 shows the security comparison of various related schemes of authentication and key agreement. In Li et al. [1] scheme, the sensor node capture attack, DoS attack, eavesdropping, anonymity, unlinkable, and dynamic addition of node cannot be prevented. On other hand Koya

Table 3 Concatenation, hash function and secret key

```
(*concatenation operation*)
fun
concat(bitstring,bitstring):bitstring.
fun concats(bitstring,nonce):bitstring.
fun concatx(key,nonce):bitstring.
fun concatk(key,bitstring):bitstring.
fun concatj(key,key):bitstring.
reduc forall x:bitstring,y:bitstring;
split(concat(x,y))=(x,y).
```

```
(*hash function*)
fun
hash(bitstring):bitstring.
fun hashk(key):bitstring.
(*secret key*)
not attacker(new Idin').
not attacker(new Rn).
not attacker(new khn).
not attacker(new kn).
not attacker(new SIdn).
not attacker(new KEY).
```

Table 4 Events definition and Queries of proposed scheme

```
(*event definition*)
event INAcceptN(sensor).
event HNAccept(intermediate_node,sensor).
event INAccept(Hub).
event sensorAcceptS(Hub,intermediate_node).
(*------event query-------*)
query N:sensor,IN:intermediate_node,HN:Hub;
inj-event(HNAccept(IN,N))==>inj-event(INAccept(HN)).
query N:sensor,IN:intermediate_node,HN:Hub;
inj-event(HNAccept(IN,N))==>inj-event(sensorAcceptS(HN,IN)).
```

Table 5 Process of the sensor

```
(*--------Process Sensor---------------------*)
let
processsensor(SIdn:bitstring,khn:key,kn:key,Pn:bitstring,Rn:nonce,
Fn:nonce,gamma:bitstring,knpulse:key,n:sensor,IN:intermediate_node
,hn:Hub,Qn:bitstring)=
let Xn=XORS(Pn,Rn)in
let Cn=XORS(SIdn,Rn)in
let tidn=hash(concat(Cn,Xn))in
out (ch1,(tidn,Qn,Cn));
in(ch1,(alpha:bitstring,beta:bitstring,eta:bitstring,mu:bitstring)
);
let Fn'=XOR(Pn,alpha)in
let beta'=hash(concats(concats(Xn,Rn),Fn))in
if beta'=beta then
event sensorAcceptS(hn,IN);
let ele=hash(concat(concat(concats(SIdn,Rn),Xn),alpha))in
let pnpulse=XOR(ele,eta)in
let qnpulse=XOR(ele,mu)in
let m=hash(SIdn)in
let Key=hash(XOR(ele,m)).
```

Table 6 Process of the Hub Node

```
(*-------------------HUB NODE-------------------*)
let
processHub(Idin':bitstring,khn:key,kn:key,SIdn:bitstring,INin:nonc
e,Xn:bitstring,Rn:nonce,Fn:nonce,knpulse:key,Pn:bitstring,n:sensor
,IN:intermediate_node,hn:Hub,SIdn':bitstring,Idin'':bitstring)=
in(ch1,(tidn:bitstring,Qn:bitstring,Cn:bitstring,Iin:bitstring));
let Iin'=XORS(Idin',INin)in
let SIdn=XORT(XORT(Qn,khn),kn)in
if SIdn'=SIdn then
if Idin''=Idin' then
event HNAccept(IN,n);
let m3=hash(SIdn)in
let X'=XOR(m3,XORV(Rn,hash(concatj(khn,kn))))in
let Rn'=XOR(Xn,Pn)in
let tidn'=hash(XORS(SIdn,Rn))in
if tidn=tidn' then
let alpha=XORS(Pn,Fn)in
let hPpulse=XOR(m3,XORV(Rn,hash(concatj(khn,knpulse))))in
let hQpulse=XOR(XORU(khn,kn),SIdn)in
let gamma=hash(concat(concat(concats(SIdn,Rn),Xn),alpha))in
let eta=XOR(hQpulse,gamma)in
let mu=XOR(hPpulse,gamma)in
let beta=hash(concats(concats(Xn,Rn),Fn))in
let m4=hash(SIdn)in
let Key'=hash(XOR(gamma,m4))in
out(ch2,(alpha,beta,eta,Idin')).
```

Table 7 Process of the Intermediate Node

```
(*------------------IN-----------*)
let
processsintermediate_node(INin:nonce,Idin':bitstring,n:sensor,IN:i
ntermediate_node,hn:Hub,Cn:bitstring)=
in(ch1,(tidn:bitstring,Xn:bitstring,Qn:bitstring));
let Iin=XORS(Idin',INin)in
event INAcceptN(n);
out (ch1,(tidn,Qn,Cn,Iin));
in(ch2,(beta:bitstring,eta:bitstring,mu:bitstring,Idin'':bitstring
));
out(ch1,(beta,eta,mu));
0.
```

Table 8 Constant computed by process sensor, Hub node and intermediate node

```
(*constant computed*)
process
new Knpulse:key;
new Rn:nonce;
new Fn:nonce;
new IN:intermediate_node;
new hn:Hub;
new n:sensor;
new Idin':bitstring;
new INin:nonce;
new Pn:bitstring;
new Xn:bitstring;
new SIdn:bitstring;
new kn:key;
new khn:key;
new gamma:bitstring;
new alpha:bitstring;
new Pnpulse:bitstring;
new Qn:bitstring;
new knpulse:key;
new SIdn':bitstring;
new Idin'':bitstring;
new Cn:bitstring;
new KEY:key;
(*constant computation*)
let m5=hash(SIdn)in
let Pn'=XOR(m5,hash(concatj(khn,kn)))in
((processsensor(SIdn,khn,kn,Pn,Rn,Fn,gamma,knpulse,n,IN,hn,Qn))|(!
processHub(Idin',khn,kn,SIdn,INin,Xn,Rn,Fn,Knpulse,Pn,n,IN,hn,SIdn
',Idin''))|(!processsintermediate_node(INin,Idin',n,IN,hn,Cn)))
```

et al. [2] scheme can prevent anonymity, unlinkable, replay attack, and addition of dynamic node. In addition, [30, 32, 34–36, 44] have satisfied less security functionality than the proposed scheme. Table 9 shows the attacks can be prevented in our scheme that satisfies the security functionalities.

6.2 Communication cost of message exchanges

Table 10 illustrates the communication cost of message exchange overhead among the S_n, I_n, and *HNode*. Considering the other related schemes time stamp size is 32-bits, $Iid'_{in} = 16$ bits and other parameters as 160-bits each. While comparing to other related schemes our scheme has the less communication cost.

6.3 Computational Cost in Terms of Hash Functions and XORed Functions

While talking about the cryptography security, the cryptography function like hash function and XOR operation is considered. Here, we consider t_{Xor} for XOR operation and t_H for hash function. Compared to other schemes, our proposed scheme in the authentication and key agreement phase, sensor node performs the $7t_{Xor}$ operation and $7t_H$ of hash operation. On the other hand, *HNode* uses the total of $14t_{Xor} + 7t_H$ which is at par lesser than Koya et al.'s [2] scheme. Table 11 shows the computation cost in terms of hash and XORed function.

6.4 Memory Overhead of Sensor Node and Hub Node

In our proposed scheme, sensor node stores its identity Sid_n, parameters P_n, Q_n and its session key K_s in its memory.

Table 9 Illustration of security functionality comparison

Security Attacks	Proposed scheme	[1]	[2]	[30]	[32]	[34]	[35]	[36]	[44]
Sensor node capture	Yes	No	No	Yes	Yes	Yes	Yes	-	No
DoS Attack	Yes	No	No	Yes	Yes	Yes	-	-	Yes
Eavesdropping	Yes	No	No	Yes	Yes	Yes	Yes	Yes	No
Mutual Authentication	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Dynamic node addition	Yes	No	Yes	Yes	Yes	No	Yes	-	No
Anonymity and unlinkable	Yes	No	No	Yes	Yes	No	Yes	Yes	-
Replay Attack	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
Impersonation Attack	Yes	No	Yes	Yes	No	Yes	Yes	Yes	Yes
Man-in-the-Middle Attack	Yes	Yes	Yes	Yes	No	-	-	-	Yes
Scalability	Yes	No	Yes	Yes	No	No	Yes	No	No
Ephemeral secret key leakage	Yes	Yes	Yes	No	No	No	No	No	Yes
Hub node stolen database Attack	Yes	Yes	Yes	No	No	No	No	No	No

Table 10 Nos. of message exchange and Communication cost among S_n, I_n and HNode

Authentication Techniques	Proposed Scheme	[1]	[2]	[30]	[34]	[35]	[36]	[44]
Nos. of message exchange	4	4	6	3	4	4	4	2
S_n to I_n	480	640 + 32	640 + 32	480	480 + 32	800 + 32	160 + 32 + 16	320
I_n to *HNode*	480	640 + 32 + 16	640 + 32	640	480 + 32 + 16	640 + 32 + 16 + 16	160 + 32 + 16	–
HNode to I_n	800 + 16	640 + 16	480	640	640 + 32 + 16	800 + 32	160 + 32	–
I_n to S_n	480	640	480	480	640 + 32 + 16	800 + 32	160 + 32	320

Table 11 Computational Cost in terms of hash functions and XORed functions

Performance Properties	Scheme	S_n	I_n	*HNode*
Proposed scheme	2-Tier	$7t_{Xor} + 7t_H$	$1t_{Xor} + 0t_H$	$14t_{Xor} + 7t_H$
Li et al.'s [1]	2-Tier	$7t_{Xor} + 3t_H$	$0t_{Xor} + 0t_H$	$12t_{Xor} + 5t_H$
Koya et al.'s [2]	2-Tier	$5t_{Xor} + 3t_H$	$5t_{Xor}$+$3t_H$	$20t_{Xor}$+$10t_H$
Ibrahim et al.'s [30]	2-Tier	$2t_{Xor} + 5t_H$	$0t_{Xor} + 0t_H$	$4t_{Xor} + 8t_H$
Xu et al.'s [34]	2-Tier	$5t_{Xor} + 5t_H$	–	$9t_{Xor} + 7t_H$
Gupta et al.'s [35]	2-Tier	$6t_{Xor} + 7t_H$	$0t_{Xor} + 4t_H$	$11t_{Xor} + 10t_H$
Almuhaideb et al.'s [36]	2-Tier	$4t_{Xor} + 1t_H$	–	$4t_{Xor} + 2t_H$
Sowjanya et al.'s [44]	WHMS	$0t_{Xor} + 2t_H$	–	$0t_{Xor} + 3t_H$

Let considered each parameter to be 128 bits each. So, the required memory in the sensor node is 640-bits similar to other related schemes Li et al.'s [1], Koya et al.'s [2], and Ibrahim et al.'s [30]. Similarly, hub node stores $Iid'_{in}, MK_{hn}, NK_n, Sid_n$ and its corresponding session key K_s. The Iid'_{in} is assumed to be 16 bits and n numbers of sensor nodes in a network of *HNode*. Here we use SHA-1 to hash the values, and to produce of SHA-1 is 160 bits. The parameters $MK_{hn} = NK_n = Sid_n = K_s = 160$ bits. Therefore, the total storage required in *HNode* is 160 + 16 bits. The comparison of memory overhead is illustrated in Table 12.

Table 12 Memory overhead in proposed scheme

Storage cost	S_n	I_n	*HNode*
Proposed scheme	4(160)-bits	16-bits	160 + 16 bits
Li et al.'s [1]	4(160)-bits	16-bits	160 + 16 bits
Koya et al.'s [2]	4(160)-bits	16-bits	320(n + 160) bits
Ibrahim et al.'s [30]	4(160)-bits	16-bits	320(n + 160) bits

7 Conclusion

In this paper, we have proposed an improved secure lightweight authentication scheme for sensor node and hub node in WBAN. The scheme includes cryptographic functions like XOR operations, concatenations, nonce, and hash functions. One of the problems in Li et al.'s scheme is the session traceable for a different session. Therefore, firstly we have removed the session traceable problem. Secondly, as an adversary may create a new Sid_n to perform authentication and key agreement, thus, to resolve, in our proposed scheme, *SAdmin* stores all the legitimate Sid_n in *HNode*'s DB. As a result, an adversary cannot perform any impersonation attack. Thirdly, using the timestamp, may result of the time synchronization issue. Therefore, we have designed our proposed scheme without use of any timestamp variables. In addition, as the sensor node is resource-limited, and less battery power supply, thus, our proposed scheme has been constructed such a way which will consume less computation cost, however, memory storage is almost the equal of all previous related schemes. The BAN logic has also been used to determine the correctness of exchange information of our scheme and also helped to prove our scheme is protected from eavesdropping attacks. Lastly, the key secrecy evolution has been performed using formal verification, i.e., ProVerif simulation tool and we have proved that our proposed scheme is secure as per our claims. However, WBANs have many challenges like sensor device energy supply, mobility, health information privacy, etc.. Hence, needs to develop some more appropriate intelligence sensor devices and protocols that work under low battery, require less computation and highly secure that resolve the specific problems and helps doctors to diagnose the patients.

References

1. Li, X.; Ibrahim, M.H.; Kumari, S.; Sangaiah, A.K.; Gupta, V.; Choo, K.K.R.: Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks. Comput. Netw. **129**, 429–443 (2017)
2. Koya, A.M.; Deepthi, P.P.: Anonymous hybrid mutual authentication and key agreement scheme for wireless body area network. Comput. Netw. **140**, 138–151 (2018)

3. IEEE Standards Association, 2012. IEEE standard for local and metropolitan area networks-part 15.6: wireless body area networks. *IEEE std*, *802*(6): 2012.
4. Li, X.; Niu, J.; Khan, M.K.; Liao, J.: An enhanced smart card based remote user password authentication scheme. J. Netw. Comput Appl. **36**(5), 1365–1371 (2013)
5. Jiang, Q.; Khan, M.K.; Lu, X.; Ma, J.; He, D.: A privacy preserving three-factor authentication protocol for e-Health clouds. J. Supercomput. **72**(10), 3826–3849 (2016)
6. Li, X.; Xiong, Y.; Ma, J.; Wang, W.: An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards. J. Netw. Comput. Appl. **35**(2), 763–769 (2012)
7. Li, X.; Ma, J.; Wang, W.; Xiong, Y.; Zhang, J.: A novel smart card and dynamic ID based remote user authentication scheme for multi-server environments. Math. Comput. Model. **58**(1–2), 85–95 (2013)
8. Wang, D.; Wang, P.: On the anonymity of two-factor authentication schemes for wireless sensor networks: Attacks, principle and solutions. Comput. Netw. **73**, 41–57 (2014)
9. Wang, D.; Wang, P.: Understanding security failures of two-factor authentication schemes for real-time applications in hierarchical wireless sensor networks. Ad Hoc Netw. **20**, 1–15 (2014)
10. Venkatasubramanian, K.K.; Banerjee, A.; Gupta, S.K.S.: PSKA: Usable and secure key agreement scheme for body area networks. IEEE Trans. Inf Technol. Biomed. **14**(1), 60–68 (2009)
11. Zhang, Z.; Wang, H.; Vasilakos, A.V.; Fang, H.: ECG-cryptography and authentication in body area networks. IEEE Trans. Inf Technol. Biomed. **16**(6), 1070–1078 (2012)
12. Poon, C.C.; Zhang, Y.T.; Bao, S.D.: A novel biometrics method to secure wireless body area sensor networks for telemedicine and m-health. IEEE Commun. Mag. **44**(4), 73–81 (2006)
13. Miao, F.; Bao, S.D.; Li, Y.: Biometric key distribution solution with energy distribution information of physiological signals for body sensor network security. IET Inf. Secur. **7**(2), 87–96 (2013)
14. Mathur, S.; Miller, R.; Varshavsky, A.; Trappe, W. and Mandayam, N.; 2011, June. Proximate: proximity-based secure pairing using ambient wireless signals. In *Proceedings of the 9th international conference on Mobile systems, applications, and services* (pp. 211–224). ACM.
15. Li, M.; Yu, S.; Guttman, J.D.; Lou, W.; Ren, K.: Secure ad hoc trust initialization and key management in wireless body area networks. ACM Trans. sensor Netw. (TOSN) **9**(2), 18 (2013)
16. Liu, J.; Zhang, Z.; Chen, X.; Kwak, K.S.: Certificateless remote anonymous authentication schemes for wirelessbody area networks. IEEE Trans. Parallel Distrib. Syst. **25**(2), 332–342 (2013)
17. Zhao, Z.: An efficient anonymous authentication scheme for wireless body area networks using elliptic curve cryptosystem. J. Med. Syst. **38**(2), 13 (2014)
18. Xiong, H.: Cost-effective scalable and anonymous certificateless remote authentication protocol. IEEE Trans. Inf. Forensics Secur. **9**(12), 2327–2339 (2014)
19. He, D.; Zeadally, S.: Authentication protocol for an ambient assisted living system. IEEE Commun. Mag. **53**(1), 71–77 (2015)
20. Shen, J.; Tan, H.; Moh, S.; Chung, I.; Liu, Q.; Sun, X.: Enhanced secure sensor association and key management in wireless body area networks. J. Commun. Netw. **17**(5), 453–462 (2016)
21. He, D.; Zeadally, S.; Kumar, N.; Lee, J.H.: Anonymous authentication for wireless body area networks with provable security. IEEE Syst. J. **11**(4), 2590–2601 (2016)
22. Liu, J.; Zhang, L.; Sun, R.: 1-RAAP: An efficient 1-round anonymous authentication protocol for wireless body area networks. Sensors **16**(5), 728 (2016)
23. Tan, C.C.; Wang, H.; Zhong, S.; Li, Q.: IBE-lite: a lightweight identity-based cryptography for body sensor networks. IEEE Trans. Inf Technol. Biomed. **13**(6), 926–932 (2009)
24. Huang, C.; Lee, H.; Lee, D.H.: A privacy-strengthened scheme for E-healthcare monitoring system. J. Med. Syst. **36**(5), 2959–2971 (2012)
25. Challa, S.; Das, A.K.; Odelu, V.; Kumar, N.; Kumari, S.; Khan, M.K.; Vasilakos, A.V.: An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks. Comput. Electr. Eng. **69**, 534–554 (2018)
26. Yeh, K.H.: A secure IoT-based healthcare system with body sensor networks. IEEE Access **4**, 10288–10299 (2016)
27. Shen, J.; Chang, S.; Shen, J.; Liu, Q.; Sun, X.: A lightweight multi-layer authentication protocol for wireless body area networks. Futur. Gener. Comput. Syst. **78**, 956–963 (2018)
28. Zebboudj, S.; Cherifi, F.; Mohammedi, M.; Omar, M.: Secure and efficient ECG-based authentication scheme for medical body area sensor networks. Smart Health **3**, 75–84 (2017)
29. Zhao, H.; Xu, R.; Shu, M.; Hu, J.: Physiological-signal-based key negotiation protocols for body sensor networks: a survey. Simul. Model. Pract. Theory **65**, 32–44 (2016)
30. Ibrahim, M.H.; Kumari, S.; Das, A.K.; Wazid, M.; Odelu, V.: Secure anonymous mutual authentication for star two-tier wireless body area networks. Comput. Methods Programs Biomed. **135**, 37–50 (2016)
31. Kompara, M.; Hölbl, M.: Survey on security in intra-body area network communication. Ad Hoc Netw. **70**, 23–43 (2018)
32. Kompara, M.; Islam, S.H.; Hölbl, M.: A robust and efficient mutual authentication and key agreement scheme with untraceability for WBANs. Comput. Netw. **148**, 196–213 (2019)
33. Konan, M.; Wang, W.: A secure mutual batch authentication scheme for patient data privacy preserving in wban. Sensors **19**(7), 1608 (2019)
34. Xu, Z.; Xu, C.; Liang, W.; Xu, J.; Chen, H.: A lightweight mutual authentication and key agreement scheme for medical Internet of Things. IEEE Access **7**, 53922–53931 (2019)
35. Gupta, A.; Tripathi, M.; Sharma, A.: A provably secure and efficient anonymous mutual authentication and key agreement protocol for wearable devices in WBAN. Comput. Commun. **160**, 311–325 (2020)
36. Almuhaideb, A.M.; Alqudaihi, K.S.: A Lightweight and Secure Anonymity Preserving Protocol for WBAN. IEEE Access. **8**, 178183–178194 (2020)
37. Madhusudhan, R.; Shashidhara.: A secure and lightweight authentication scheme for roaming service in global mobile networks. J. Inf. Secur. Appl. **38**, 96–110 (2018)
38. Clifford Neuman, B.; Stuart, G.S.: A Note on the Use of Timestamps as Nonces. Oper. Syst. Rev. **27**(2), 10–14 (1993)
39. Dolev, D.; Yao, A.: On the security of public key protocols. IEEE Trans. Inf. Theory **29**(2), 198–208 (1983)
40. Burrows, M.; Abadi, M. and Needham, R.M.: A logic of authentication. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, *426*(1871): 233–271 (1989).
41. Islam, S.H.; Biswas, G.P.: A pairing-free identity-based two-party authenticated key agreement protocol for secure and efficient communication. J. King Saud Univ.-Comput. Inf. Sci. **29**(1), 63–73 (2017)
42. Blanchet, B.: Modeling and verifying security protocols with the applied pi calculus and ProVerif. Found. Trends® Priv. Secur. **1**(1–2), 1–135 (2016)
43. Blanchet, B.: June. An efficient cryptographic protocol verifier based on prolog rules. csfw **1**, 82–96 (2001)

44. Sowjanya, K.; Dasgupta, M.; Ray, S.: An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems. Int. J. Inf. Secur. **19**(1), 129–146 (2020)
45. Kilinc, H.H.; Yanik, T.: A survey of SIP authentication and key agreement schemes. IEEE Commun. Surv. Tutor. **16**(2), 1005–1023 (2013)
46. Yamano, K., Sharp Corp, 2006. Node structure information management method and radio network system. U.S. Patent 7,103,354.