I.J. Wireless and Microwave Technologies, **2024, 6, 26-47**
Published Online on December 8, 2024 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2024.06.03

Secure Communication in Wireless Body Area Networks with ECC-Based Key Management and Renewal Protocols

Yusuf Taofeek*
Department of Cyber Security Science, Federal University of Technology, Minna, Nigeria
E-mail: yusuf.pg824211@st.futminna.edu.ng
ORCID iD: https://orcid.org/0009-0008-7589-6354
*Corresponding Author

Waziri Onomza Victor
Department of Cyber Security Science, Federal University of Technology, Minna, Nigeria
E-mail: victor.waziri@futminna.edu.ng

Olalere Morufu
Department of Cyber Security Science, Federal University of Technology, Minna, Nigeria
E-mail: lerejide@futminna.edu.ng
ORCID iD: https://orcid.org/0000-0001-6055-2198

Noel Moses Dogonyaro
Department of Cyber Security Science, Federal University of Technology, Minna, Nigeria
E-mail: moses.noel@futminna.edu.ng
ORCID iD: https://orcid.org/0000-0002-6774-6144

Received: 02 June, 2024; Revised: 25 July, 2024; Accepted: 15 September, 2024; Published: 08 December, 2024

Abstract: Wireless body area networks (WBANs) are employed to monitor and collect physiological data using wearable and implantable sensor nodes for remote medical applications. Due to the sensitive nature of the data transmitted over open wireless channels, robust security measures are critical. WBANs are particularly susceptible to threats such as eavesdropping, man-in-the-middle attacks, node capture, and replay attacks. Moreover, the constrained nature of sensor nodes and duplicate occurrence of keys makes it challenging for cryptographic key management protocols to function efficiently without increasing the computational and storage overhead of the security protocol. This work proposes a secure and efficient key management protocol designed for key establishment, revocation, and renewal of compromised keys in WBANs. The protocol leverages elliptic curve cryptography (ECC) and integrates a keychain hash function mechanism to eliminate the reuse of keys and execute key computations efficiently. Furthermore, it utilizes an authentication method based on authentication tables, which reduces the number of messages exchanged and minimizes computational overhead. To guarantee the protocol's robustness, a formal validation using BAN logic was conducted, validating that the protocol meets key security requirements, which include, confidentiality, integrity, and authentication. An informal security analysis further demonstrates the protocol's resilience against impersonation, eavesdropping, man-in-the-middle, replay, and injection attacks. A performance evaluation of the protocol’s computational and storage costs was also carried out. The results show that while our protocol incurs lower computational costs compared to some related works, it exhibits slightly higher costs compared to others. In terms of storage overhead, our protocol outperforms most existing solutions. Future work will focus on optimizing the protocol and exploring further efficiency improvements in its implementation within a WBAN testbed.

Index Terms: Key management, Cryptography, Authentication, WBAN, Key revocation and renewal.

1. Introduction

The rise in the utilization of wireless networks and the miniaturization of electronic devices has inspired the advancement of wireless body area networks (WBAN) [1]. The technology comprises miniaturized intelligent sensor nodes, usually implanted or worn around the body for continuous monitoring of variations in users' health conditions. And real-time feedback from caregivers and medical personnel [2]. The use of the internet and various accessible technologies like Bluetooth, Wi-Fi, and mobile networks has shown that WBAN can go beyond personal health and be extended to many other areas of life [3]. However, communication over an unsecure channel, made WBAN vulnerable to a series of cyber-attacks, like eavesdropping on transmissions, man-in-the-middle attacks, replay attacks, disclosure of message content, and modification and injection attacks. Additionally, due to their constrained resources and peculiar domain applications, this technology faces challenges such as high computation time, memory usage, and communication overhead [4]. To protect the privacy and security of WBAN systems, such as data confidentiality, integrity and authentication. A secure and efficient cryptographic mechanism is required. Cryptographic key plays a very crucial role in securing networks, and managing keys which includes the establishment of keys, distribution of keys and key renewal or revocation in case of attacks is very challenging in WBAN. As a result, key management protocol is crucial in the design and deployment of secure WBAN. Therefore, key management protocol must efficiently perform the establishment of keys, distribution of keys and renewal or revocation of keys [5,6,7]. Although key management protocol significantly affects the level of security and performance of the network.

Several research has been introduced to address problems of secure and efficient key management protocols. Existing key management schemes require large key sizes and high-cost cryptographic resources to achieve high-security levels, which significantly increases the computational, communication, energy and storage costs for key establishment, distribution and renewal or revocation [8,9], A key management scheme proposed in [10] uses low-cost functions such as the one-way hash function, exclusive-OR and concatenation for data transmission in WBAN. While the scheme is resistant to several cyber-attacks such as impersonation, replay, man-in-the-middle, and session key disclosure, it does not address key renewal and revocation. In addition, the implementation efficiency of the scheme is considerably high with respect to of execution and computation time, because of the duplicate occurrence of keys and direct or full-scale implementation of the ECC cryptographic algorithm. Similarly, an efficient and scalable protocol for key management and distribution was developed for large-scale wireless sensor networks (WSN) in [11]. For secure and efficient key distribution, node revocation, node addition and key updates, using lightweight symmetric cryptography, hash function and pseudorandom functions. The scheme utilizes tier-based architecture for its shared key establishment. The approach minimized the number of exchange messages and limited the impact of compromised nodes to the affected tier. Nevertheless, using a centralized key management server makes the system susceptible to a single point of failure. Also, additional measures have been incorporated into the key exchange mechanism to prevent replay attacks and data modification. While these enhancements improve security, they also increase the scheme's computational and energy overhead. The author in [12] proposed a lightweight patient monitoring protocol based on ECC to meet the security requirement of WBAN solutions. The protocol’s resilience against common attackers was demonstrated using profVerif verification tool. A comparison with related works indicated that the protocol achieves high computation efficiency. However, the protocol’s key agreement steps did not support the revocation and renewal of compromised keys. Additionally, its reliance on the property of a Physical Unclonable Function (PUF) to prevent attackers from computing the tuple is not entirely reliable. In the event of the master secret key leakage and the attacker successfully guess the sensor node's identity ID, they could potentially compute a valid identity of sensor nodes. Thus, compromising the protocol's defense against impersonation and node capture attacks. Moreover, due to the participating devices' requirements, the protocol imposes a significant storage overhead.

Among all the cryptographic protocols, Elliptic Curve Cryptography (ECC) is considered the most appropriate for constrained devices because of its important benefits on overhead and performance [13]. The primary attraction of ECC is in the ability to modify its key derivation functions with less computationally intensive hash function primitives. Additionally, it offers robust security with significantly reduced key size. The reduced key size allows more efficient cryptographic operations on constrained nodes without compromising security levels. Hence, the choice of ECC is to improve the efficiency of the key management protocols. Considering the security requirements in WBAN, research can be split into three distinct tiers namely: Intra-WBAN communication security (WBAN tier 1 architecture); Inter-WBAN communication security (WBAN tier 2 architecture); Beyond-WBAN communication security (WBAN tier 3 architecture) [14]. As shown in Fig. 1, the tier 1 architecture confers all transmission with the on-body nodes and personal server. The collected data on the personal server is transmitted to the next connecting device or PDA in tier 2. Similarly, the gathered physiological data is transmitted to store in a remote web server or electronic health repository in tier 3. As stated earlier, the entire data exchange and storage process is expected to be protected. Therefore, in light of this background, this study focuses on secure and efficient key establishment, distribution, and renewal techniques required in the intra-WBAN architecture, which is the primary aspect of security in tier 1.

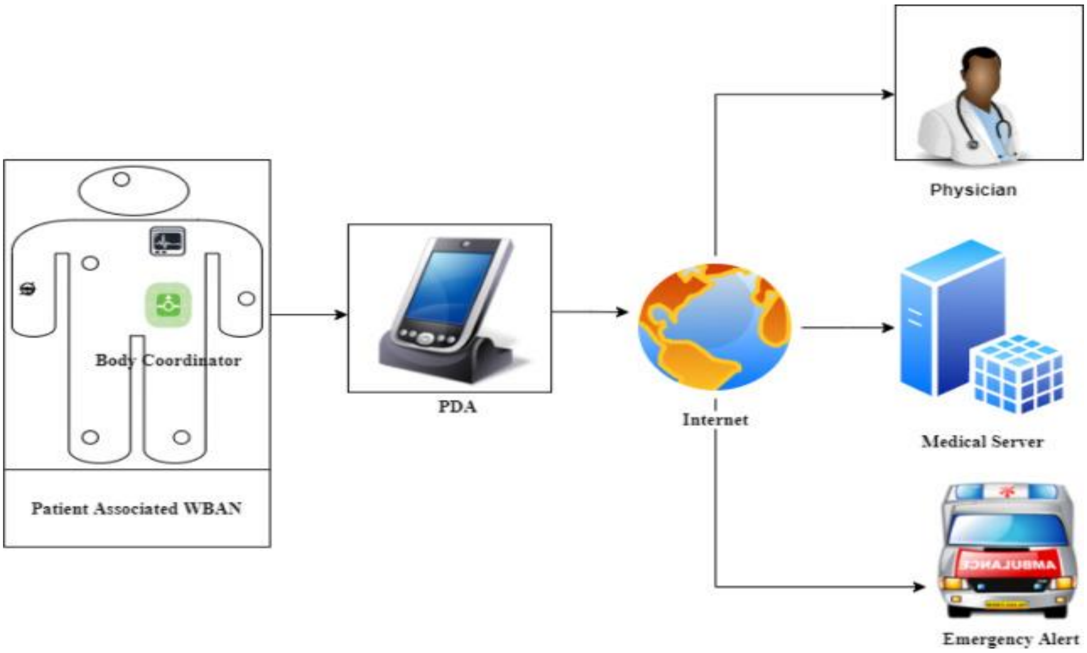


Fig. 1. General wireless body area network architecture

Based on the above-discussed challenges, this research has proposed a secure and authenticated key management and renewal protocol that utilizes lightweight functions such as key chain hash function, exclusive-OR, and concatenation. To eliminate multiple key usages and prevent key-related attacks like replay, perfect forward secrecy, and man-in-the-middle attacks. It achieves secure key establishment, renewal and revocation while optimizing performance overhead for secure information exchange within WBAN.

The main contributions of this research are:

- Propose an authenticated key establishment, distribution and revocation scheme using an optimized ECC cryptographic algorithm
- Present a solution that utilizes a keychain hash function to eliminate the need for multiple keys, while also being resilient against key-related attacks in WBAN
- Perform an informal analysis of the protocols using mathematical propositions and proofs to demonstrate their resilience against prevalent attacks in WBAN
- Carry out a formal analysis of the designed protocols using the BAN logic model
- Presents the heuristic security and performance analysis of the protocols

The remainder of this paper is structured in the following sections: A comprehensive literature review is presented in section 2. Section 3 outlines the system and attacker models, including the mathematical preliminary. The proposed key management protocols for WBAN are presented in section 4. Section 5 offers both formal proof and informal security analysis of these protocols. Lastly, section 6 summarizes the major findings and concludes the study.

2. Review of Related Literature

This section of the study reviewed various key management protocols, the review discusses the strengths and weaknesses of the existing key management protocols for WBAN. The work in [15] proposed a key establishment scheme to reduce resource usage and ensure security requirements concurrently. The scheme achieves its goals by leveraging the transmission pattern and adoption of heterogeneous WSNs. The process minimized the number of keys established and the overhead cost of establishing key decreases. Performance evaluations confirm the scheme’s efficiency and robustness against different attacks. However, an adversary can eavesdrop on the broadcast messages in the cluster, because a compromised node shares the intra-cluster broadcast encryption key with all nodes in the cluster. Furthermore, the protocol witnessed slit computation overhead because of the use of PKC for the H-node. The resilience in [15] was improved in the key distribution protocol proposed in [16] to provide authenticated nodes with cryptographic keys that are based on ECC encryption primitives.

The scheme is designed to be robust against adversaries who possess either working knowledge of the protocol or pre-programmed keys. However, the scheme introduces significant overhead due to its timing constraints, and its resilience against replay and denial-of-service attacks is not fully guaranteed. The authors in [17] present a novel key management scheme that utilizes Hyper Elliptic Curve Cryptography (HECC) for pre-distribution and a simple hash and permutation function-based pairwise key generation. A unique seed key is pre-distributed into sensor nodes from the polynomial coefficient, and the key ring is computed through point addition and doubling operation over a unique key and pre-shared into the network nodes. The scheme achieved strong resilience and faultless connectivity. Furthermore, security analysis shows that the probability of a node capture attack in the scheme is zero. The work in [18] proposed energy-efficient secure communication that used symmetric key encryption between pairs of nodes. An ECC algorithm for unique identity of each node, and initial symmetric keys establishment between pairs of nodes. Security analysis shows that the scheme guarantees the secrecy of keys and CIA of messages. However, the process of informing the recipient of decryption and MAC computation key retrieval slightly increases the message size.

In [19] a novel ECC-based key establishment protocol that offers efficient computing and transmission overheads was presented. The proposed protocol is divided into three phases: system setup, node registration and secure key exchange. Security analysis reveals that the scheme is resilient to the following attacks: impersonation, man-in-the-middle, modification, replay and forward secrecy. [20] proposed a mutual authentication key establishment technique based on the ECC approach that guaranteed system security from known malicious attacks. The scheme can reduce the dependency on central certification authority to establish secret keys among nodes for each session in the network. Security and performance analysis shows that the solution consumes little memory, utilizes low energy in its processes and at the same time guarantees security against man-in-the-middle attacks, replay attacks, impersonation attacks and denial of service attacks among known cyber-attacks. The authors in [21] proposed an ECC-based authenticated lightweight key establishment protocol that relies on data aggregation to generate shared keys between smart meters and aggregators, then between aggregator and service provider. The solution is capable of providing mutual authentication, traceability, anonymity and forward secrecy. Performance and security analysis shows that the scheme is computationally efficient and secure against impersonation and replay attacks. Reference [22] proposed an improved protocol for key agreement and authentication leveraging hash functions and XOR operation. The scheme is resilient to node compromise, impersonation and base station attacks. Simulation results and resilience analysis indicate that the scheme has addressed various identified weaknesses in terms of storage, computation and communication overheads. [23] proposed a blockchain-based authentication technique and modified the user identifier and shared secret keys generation process to improve the efficiency of the authentication activity, however, the scheme does not consider node capture or impersonation attacks that may necessitate revocation and renewal of keys, furthermore, the evaluation of the scheme only covers the association request phase, therefore, the study does not provide the overall time consumption of the protocol. [24] proposed a novel, centralized session key mechanism for LPWAN, utilizing BYKA, for improved replay attack resilience and reduced transmission overheads. However, the work does not present a comprehensive analysis of the scalability and synchronization aspects of the presented session key approach in large-scale LPWAN deployments. [25] presents a server-less mutual authentication scheme for edge networks. The protocol utilizes a public-key algorithm, challenge-response mechanism, identifier, time stamps, and session keys. The approach eliminates the need for secret keys and reduces infrastructure requirements. The results evaluation shows the scheme's effectiveness when compared with other existing works.

Reference [26] introduces a secure and fault-tolerant data communication framework for Wireless Sensor Networks (WSNs), that combines Elliptic Curve Diffie-Hellman (ECDH) and Elliptic Curve Cryptography (ECC) with Message Passing Interface (MPI) parallel computation. However, the technique demonstrates improved execution time and memory usage. However, the performance analysis only considers a limited number of sensors, and it is uncertain how the framework would work in larger-scale distributions. [27] presents a secure data aggregation and authentication scheme (SDAA) for underwater vehicular wireless sensor networks (UVWSNs) that is effective in preventing and detecting malicious node behaviour. The study was implemented and validated, the result shows enhanced performance in energy efficiency and delay reduction, relative to existing secure MAC approaches. However, environmental factors like water pressure and temperature which can impact the performance and reliability of the solution in real-world underwater conditions are not considered. [28] proposed a lightweight group key management that utilizes fuzy logic and crossover-boosted particle swarm optimization (CPSO) secure key revocation technique for compromised nodes. The proposed scheme presents a dependable, efficient, and secure transmission approach for IoT decentralized systems. Furthermore, the computation and communication overhead of the scheme was minimized, thereby achieving greater efficiency when compared with other existing approaches. Reference [29] presents an advanced mutual authentication and key agreement protocol featuring batch authentication capacities, resistant to replay, impersonation, and man-in-the-middle attacks. Performance analysis shows that the approach outperforms other state-of-the-art solutions.

[30] proposed an anonymous authenticated key exchange scheme that is efficient for wearables and sensors. The scheme, utilizes authenticated encryption and associated data such as bitwise XOR and hash function, for secret session key establishment, used for secure transmission between network parties. Security analysis of the scheme shows that the study achieved wearable device's security and performance requirements. However, the computational overhead on the cloud server side is slightly high. Additionally, the five messages transmitted during the login process are high compared to some related work. [31] Present a low-cost mutual authentication key agreement scheme that leverages exclusive XOR and cryptographic hash function operations. The session key is derived for each round by refreshing the seed parameter, and the derived session key is used to validate a pair of legitimate nodes in the network. Formal security verification of the scheme was conducted with BAN logic, results show that the proposed scheme outperforms

some related work with respect to computation, storage and transmission cost. However, the study did not use standard cryptographic algorithms, making the performance and security claims unsubstantiated. A robust three-factor authentication protocol was proposed in [32] for secure data sharing in WBAN. The study used RSA encryption and decryption algorithms. The key agreement protocol only utilizes the RSA algorithm to secure the secret values of user and body sensors but does not involve session key computation to preserve the efficiency of the protocol. A thorough evaluation of the protocol's security was conducted, and the result indicates the protocol's resilience to several known attacks. The performance analysis indicates that the storage and computation costs are better than those of comparable schemes. However, the protocol's resilience against forward secrecy, which relies on the non-retrievability of a secret parameter, can be compromised by a powerful adversary capable of breaking integer factorization. Additionally, the protocol's communication overhead is higher than that of related schemes due to the number of transmitted messages needed to meet the scheme design attributes. The work in [33] proposed a blockchain-based secure key management in a fog computing domain. The scheme employs a one-way hash chain for the computation of public and private key pairs and then uses ECC for secure key sharing. The session key computed for both edge nodes is computed using the key pair provided by the fog server and securely saved on the blockchain. Both formal and informal security analysis was carried out, and it was observed that the scheme is resilient against most known attacks. The performance analysis demonstrated that the scheme performed effectively in comparison with related work in terms of computation communication and storage overheads.

3. Preliminaries

This section presents an overview of the system model and the attacker model, also, we introduce the mathematical backgrounds of the cryptographic and key chain hash function algorithm.

3.1 System model

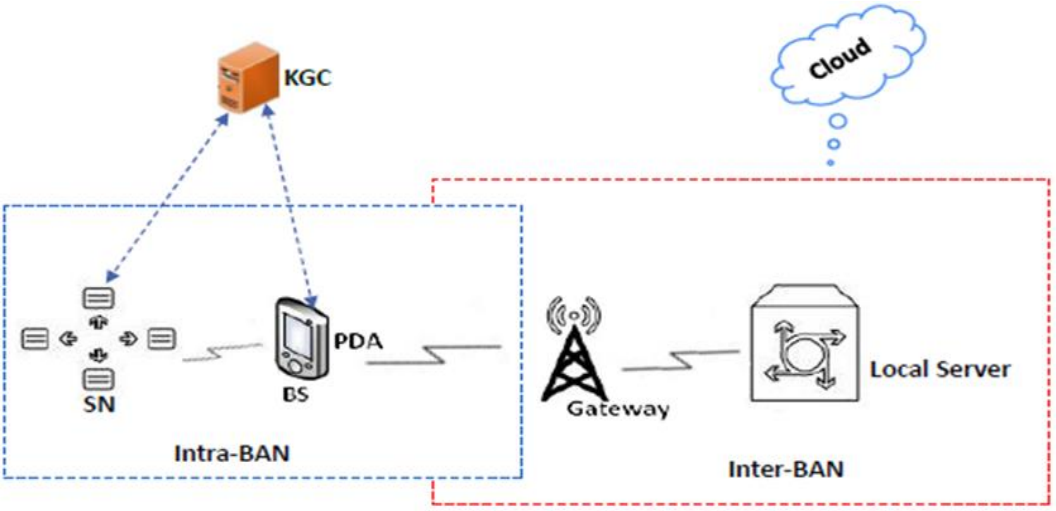


Fig. 2. System model of the proposed scheme.

This study considered a system model that comprises n strategically deployed sensor nodes in a centralized multi-hop WBAN architecture. The proposed model consists of three types of nodes as represented in Figure 2. They include the tier 1 sensor nodes (SNs) in the intra-BAN region, that are deployed in/and or on the user's body to monitor and collect physiological data [34]. We assumed that all the sensor nodes have uniform but constrained resources in terms of memory, power and computation capabilities. Because of the constrained storage space and limited transmission range from the local server. The collected physiological data must be forwarded to the base station that operates as the central controller for storage and computation processes for onward transmission to the local server.

The base station mostly personal digital assistance (PDA), is assumed to have more resources than the SNs in terms of processing, computing, memory and power, which makes them suitable to act as a go-between the SNs and the local server (Ls). Therefore, the base station serves as an intermediary node to collect and collate the sensed data and send them to the tier 2 or off-body devices in the inter-BAN region. Furthermore, because the WBAN entities are deployed on the same body, it is assumed that they are within one another's transmission range, hence they can communicate directly in a single and multi-hop way using wireless mediums. There is a dedicated key generation center (KGC) to initialize the system and register the sink and the network sensors. The local server represents the last class of nodes in WBAN. The Ls is responsible for all connections to all the healthcare service providers along the WBAN network.

32 Attacker model

This section presents the attacker model with some applicable assumptions regarding the proposed key management protocol. These assumptions are broadly admitted in [35]. Therefore, this work considers an attacker with the capacity to conduct both passive and active attacks from static or mobile positions. The attacker aims to access sensed physiological data in the WBAN through eavesdropping and impersonation. In WBAN, it is pertinent to prevent malicious entities from eavesdropping and impersonating authentic nodes during the collection and transmission of the patient's life-critical physiological data. In the event an attacker successfully hurled any of the mentioned attacks, we assumed that it could initiate further attacks like data replay, forward and backward secrecy, and man-in-the-middle threats. The following valid assumptions are made about the attacker.

- It can intercept and access messages broadcast through the communication channel to obtain the messages, extract the secret and give commands. For instance, an automated insulin pump controlled by an attacker can send a wrong or compromised instruction, to inject an insulin overdose into the patient's bloodstream
- The attacker has the capacity to alter, modify, rebroadcast and reroute the messages it earlier eavesdropped. For instance, a false electrocardiogram sensor reading may be provided to the doctor, which may lead to improper attention to the patient
- The attacker could decrypt and modify encrypted messages with the secret key in its possession and later replay the modified message. This action of the attacker can lead to the loss of the patient's sensitive data since the original message has been modified and altered by the attacker.

When carrying out an impersonation attack, the attacker pretends to be an authentic device of the network, which enables the attacker to fabricate a physical address and forge authentication credentials of legitimate nodes, to launch impersonation attacks. Similarly, we considered the attacker's ability to eavesdrop and falsely inject traffic. The attacker can achieve its goal of false traffic injection by intercepting exchanged encrypted messages from the transmission channel, and decrypting the message to view their contents, inject false data or replace some of the previously sent messages. In our bid to limit the scope of the key management protocol, this study will not be concerned with details of the intrusion detection system or any indicator of compromise in the network.

3.3 Mathematical Overview of the Cryptographic Algorithm and Authentication Mechanism

a. Elliptic Curve Cryptography

The elliptic curve cryptography (ECC) was introduced in the 80's by Miller & Koblitz, ECC is an approach to public key cryptography, which has its foundation in the technique of algebraic structure of elliptic curve defined over a finite field [36]. The elliptic curve is determined by the general Weiestrass equation presented in equation (1)

$$y^2 \equiv x^3 + \alpha x + \beta (mod\ \rho) \tag{1}$$

The expression defines the mathematical operations, and $mod\ \rho \neq 0$ such that P represents a large prime number. α, β are elements of the curve, while the variables x, y are set of all points. For each given value of $\alpha\ and\ \beta$, the graph contains positive and negative values of y for each value of x. The expression in equation (2) is a condition that must also be satisfied, together with the point at infinity represented by ∞ on the curve [37].

$$4\alpha^3 + 27\beta^2 \neq 0 (mod \rho) \tag{2}$$

b. Elliptic curve group operations

The elliptic curve group operations involve computations of the points on the curve. Essentially, the operation performed on elliptic curve points includes point addition, point doubling, scalar multiplication and inverse operation.

- **Point addition**: Given two points on a curve, that is, $P = (x_{1,} x_2)$ and $Q = (x_1, x_2)$. the sum of the two points P and Q are expressed by $R = (x_3, y_3)$ such that, the expression on equation 3 lies on the curve.

$$P+Q=R \tag{3}$$

- **Point doubling:** the process involves taking a point p on an elliptic curve, and then find another point $2P$, such that:

$$p + p = 2p \tag{4}$$

$$\text{Let } P = (x, y) \text{ and, if } P \neq +P\text{. Then } 2P = (x_3, y_3) \tag{5}$$

Where x, y are the gradient of the tangent at point P on the curve, whose coordinate is denoted by x_3, y_3.

- **Scalar multiplication:** in this operation, given an integer $\mathcal{K}$ and a point P, scalar multiplication computation involves the addition of point p to itself $\mathcal{K}$ times, as expressed in equation (6),

$$\mathbb{R} = \{P + P + \ldots + P\}\mathcal{K} \text{ times} \tag{6}$$

- **Inverse operation:** Consider a point P on the curve with coordinates $p = (x_p, y_p)$, the inverse of point P is define as $Q = -p\,(x_p - y_p)$ and presented in equation (7) such that:

$$P + Q = 0 \tag{7}$$

For more detailed information on elliptic curve cryptography and elliptic curve group operations, please refer to [30].

Consider a point P on the curve with coordinates

c. Hash-function mechanism

A hash function can be described as a one-way mathematical function that takes an arbitrary-size input into a fixed-size output that can be applied for efficient authentication. A key chain hash function comprises of chain of multiple one-way hash functions computed using a secure hash algorithm. A lightweight hash chaining method is utilized as discussed in [38]. The KGC takes as inputs the key chain that contains P non-colliding hash in place of the elliptic curve domain parameters. In each single chain, every value is considered as a potential key, thus, in a key chain pool, the next key is computed by taking the hash of the previous key. The keys can undergo a hashing process at any given number of times $1(0_i = 1_i = L)$. Figure 3 presents the illustrative expression of the key chain pool operation.

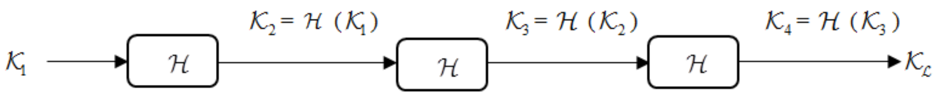

$\mathcal{K}_1$
$\mathcal{H}$
$\mathcal{K}_2 = \mathcal{H}(\mathcal{K}_1)$
$\mathcal{H}$
$\mathcal{K}_3 = \mathcal{H}(\mathcal{K}_2)$
$\mathcal{H}$
$\mathcal{K}_4 = \mathcal{H}(\mathcal{K}_3)$
$\mathcal{K}_{\mathcal{L}}$

Fig. 3. key chain hash function

4. Proposed Scheme

This study presents a four-phase secure key management protocol for WBAN, leveraging ECC cryptographic primitives enhanced by lightweight cryptographic techniques such as hash-function-based keychain mechanisms and exclusive-OR operations. In the keychain algorithm, each seed value within a single chain is considered a potential key, with the next key derived by hashing the previous key any number of times $1\,(0_i = 1_i = L)$. . This approach reduces the computational demands of our protocol, as keychain computation is less complex compared to the multiplicative group operations required for ECC key derivation functions [39]. To further minimize computational overhead across all phases of our protocol, the adopted authentication mechanism bypasses encrypting messages when public-key

Table 1. List of symbols, notations and their definitions

Symbols	Descriptions
$\mathcal{BS}s$	Base station
$\mathbb{N}$	Decentralized node
$\mathcal{A}$	Adversary or attacker
$\mathcal{ID}$	Node identity
Pk	Public key
$\mathcal{S}$k	Secret key
G	Basepoint generator
$\mathcal{K}_{\mathcal{DH}}$	Key Diffie-Helman
m	Bit index mask
$r\mathrm{T}\psi$	Registration timestamp
$c\mathrm{T}\psi$	Receiving timestamp
$\oplus$	Current timestamp
	Bitwise exclusive-OR

algorithms are used. Instead, it focuses on ensuring that the received public key is authentic, as specified in the authentication table, and that message integrity is maintained [40,41]. These optimizations enhance scalability, allowing the protocol to operate efficiently without imposing additional overhead on memory, computation, or communication resources as the problem space expands, thereby conserving energy.

The mathematical expressions here present the procedures for the derivation of unique keys based on the ECC algorithm for the phases of the proposed protocol.

The key generation operation involves the following:

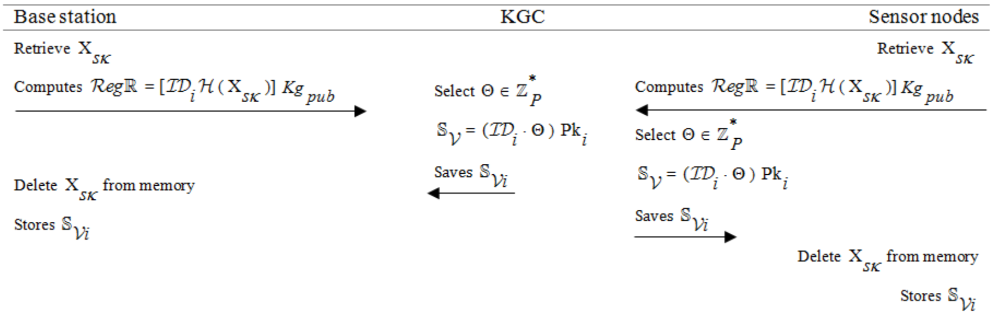
1. **Setup**: The KGC received as input a set of ECC key derivation functions (KDF) $\eta = (\rho,\alpha,\beta,G,n,h)$ which is traded with the key chain hash function, expressed in section 3.3, and then computes the algorithms as follows. Select an integer $\mathcal{S}\mathrm{k} \in \mathbb{Z}_P\ [1,\, n-1]$ and compute KGC's public key $Kg_{pub} = \mathrm{Sk.g}$. Where $\mathcal{S}\mathrm{k}$ is a randomly selected integer at the interval $[1, n-1]$.
2. **Compute hash functions:** $\mathcal{H} : \{1,\, 0\}^* \rightarrow \{1,\, 0\}^m$, the bit indexes by mask $(\mathcal{m})$ of order n, $\{m_1, m_2, m_3, \ldots, m_n\}$.
3. **Generate partial secret key** $\mathrm{X}_{s\kappa}$: For each participant i, the KGC computes its partial secret key, thus, $\mathrm{X}_{s\kappa} = Kg_{pub} . \eta i$ where the KDF $\eta_i = \mathcal{H}\ (G_\iota)$ the hash of the basepoint generator and Kg_{pub} is the public key.
4. **Compute a secret key:** Choose a random integer $\mathcal{S}\mathrm{k} \in \mathbb{Z}_P^*$ where $\mathcal{S}\mathrm{k} \in \mathbb{Z}_P^*\ [1,\, n-1]$ is the randomly selected integer at intervals to output the secret key.
5. **Compute a public key** $\mathrm{P\,k}$: Given that $\mathcal{S}\mathrm{k} \in [1,\, n-1]$, then compute $\mathrm{P\,k} = \mathcal{S}\mathrm{k}\ \eta \cdot$ where η is the seed key of the KDF

4.1. *Initialization and registration phase*

The initialization and registration phase comprise two sub-phases, the first phase includes the $\mathcal{BS}s$ and KGC, while the second phase involves node $\mathbb{N}$ and the KGC. The registration phase is outlined in the following steps, using the mathematical concepts described in Section 4, and illustrated in Figure 4 and Algorithm 1

4.1.1 *Registration between base station and* KGC

- The base stations $\mathcal{BS}s$ retrieves the partial secret keys $\mathrm{X}_{s\kappa}$ from its memory then computes a registration request $\mathcal{R}eg\mathbb{R} = [\mathcal{ID}_i \mathcal{H}(\mathrm{X}_{s\kappa})]\ Kg_{pub}$ and send it to KGC
- KGC receive the request, select a random value $\Theta \in \mathbb{Z}_P^*$ and compute a secret $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i$ and send it to the base station
- Base stations $\mathcal{BS}s$ receive the secret value $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i$ deletes $\mathrm{X}_{s\kappa}$ then save $\mathbb{S}_{\mathcal{V}i}$
- *4.1.2 Registration between sensor nodes and* KGC
- The sensor nodes $\mathbb{N}$ retrieves their partial secret keys $\mathrm{X}_{s\kappa}$, computes a registration request $\mathcal{R}eg\mathbb{R} = [\mathcal{ID}_i \mathcal{H}(\mathrm{X}_{s\kappa})]\ Kg_{pub}$ and send it to KGC
- KGC receive the request, selects a random value $\Theta \in \mathbb{Z}_P^*$ and computes a secret value $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i$ and send it to the base station
- Base stations $\mathcal{BS}s$ receive $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i$, deletes $\mathrm{X}_{s\kappa}$ then save the secret value $\mathbb{S}_{\mathcal{V}i}$.



Base station
KGC
Sensor nodes
Retrieve $\mathrm{X}_{s\kappa}$
Retrieve $\mathrm{X}_{s\kappa}$
Computes $\mathcal{R}eg\mathbb{R} = [\mathcal{ID}_i \mathcal{H}(\mathrm{X}_{s\kappa})]\ Kg_{pub}$
Select $\Theta \in \mathbb{Z}_P^*$
Computes $\mathcal{R}eg\mathbb{R} = [\mathcal{ID}_i \mathcal{H}(\mathrm{X}_{s\kappa})]\ Kg_{pub}$
$\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i$
Select $\Theta \in \mathbb{Z}_P^*$
Saves $\mathbb{S}_{\mathcal{V}i}$
$\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i$
Delete $\mathrm{X}_{s\kappa}$ from memory
Stores $\mathbb{S}_{\mathcal{V}i}$
Saves $\mathbb{S}_{\mathcal{V}i}$
Delete $\mathrm{X}_{s\kappa}$ from memory
Stores $\mathbb{S}_{\mathcal{V}i}$

Fig. 4. Registration phases of base station and sensor nodes with the KGC

Algorithm 1. Algorithm for the registration phase

```
Input: ${\mathrm{P}, \mathcal{ID}\mathrm{X}_{s\kappa}, \eta_i}$
Output: $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)\ \mathrm{Pk}_i)$
    1: $\mathcal{BS}s / \mathbb{N}$ retrieves $\mathrm{X}_{s\kappa}$
    2: Computes $\mathcal{R}eg\mathbb{R} = [\mathcal{ID}_i \mathcal{H}\ (\mathrm{X}_{s\kappa})]\ \ Kg_{pub}$ send to KGC
    3: KGC randomly select $\Theta \in \mathbb{Z}_P^*$ and
    4: Compute $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)$
    5: KGC send $\mathbb{S}_{\mathcal{V}} = (\mathcal{ID}_i \cdot \Theta)$ encrypted with $\mathrm{Pk}_i$
    6: If $\mathbb{S}_{\mathcal{V}i} = (\mathcal{ID}_{\mathcal{BS}s} \cdot \Theta) = (\mathcal{ID}_{\mathbb{N}} \cdot \Theta)$ then
    7: $\mathcal{BS}s$ and $\mathbb{N}$ deletes $\mathrm{X}_{s\kappa}$ and Stores $\mathbb{S}_{\mathcal{V}i}$
    8: End if
```

4.2. *Key establishment phase*

The proposed key establishment phase supports the establishment of a shared key and pairwise key for communication in the WBAN network.

i. *Shared key establishment:* the shared key establishment process adopts the non-interactive variation of the Diffie-Hellman key exchange presented in [18]. The base stations and sensor nodes participate in the shared key computation as follows.

Step 1. The base station $\mathcal{BS}s$ randomly select $\mathcal{S}\mathrm{k} \in \mathbb{Z}_P^*\ [1, n-1]$ and computes $\mathcal{K}_{\mathcal{DH}}(\mathcal{BS}s, \mathbb{N}) = \mathcal{S}\mathrm{k}_{\mathcal{BS}s} \cdot \mathrm{Pk}_{\mathbb{N}}$. Since $\mathcal{BS}s$ knows its secret key $\mathcal{S}\mathrm{k}$ and the public $\mathrm{Pk}_{\mathbb{N}}$ of node $\mathbb{N}$.

Step 2. Node $\mathbb{N}$ similarly, compute $\mathcal{K}_{\mathcal{DH}}(\mathbb{N}, \mathcal{BS}s) = \mathcal{S}\mathrm{k}_{\mathbb{N}} \cdot \mathrm{Pk}_{\mathcal{BS}s}$. Therefore, both computations yield the same shared key since the multiplication is commutative.

Step 3. Hence, both computations output the same shared key for the $\mathcal{BS}s$ and the network node $\mathbb{N}$. $\mathcal{K}_{\mathcal{DH}}(\mathbb{N}, \mathcal{BS}s)$ $= \mathcal{S}\mathrm{k}_{\mathbb{N}} \cdot \mathrm{Pk}_{\mathcal{BS}s} = \mathcal{S}\mathrm{k}_{\mathbb{N}} \cdot [\mathcal{S}\mathrm{k}_{\mathcal{BS}s} .G] = [\mathcal{S}\mathrm{k}_{\mathbb{N}} . G] \cdot \mathcal{S}\mathrm{k}_{\mathcal{BS}s} = \mathrm{Pk}_{\mathbb{N}} . \mathcal{S}\mathrm{k}_{\mathcal{BS}s}$.

ii. Authentication and pairwise key establishment: the pairwise key establishment process involves mutual authentication of all registered nodes in the network. Once mutual authentication is completed, the pairwise key is established between network entities. The authentication and key establishment process comprises the following steps, depicted in Figure 5, is described in the following steps and illustrated in Figure 5, with the corresponding algorithm provided in Algorithm 2.

Step 1. Node $\mathbb{N}$ generate a current timestamp $cT\psi$ and compute a request $AuR = (ID_i, cT\psi, Pk_i)$ then send the request to $\mathcal{BS}s$.

Step 2. The base station $\mathcal{BS}s$ receives the request and verifies the timeliness of the request by verifying the transmission delay associated with the request $c\mathrm{T}\psi - \mathrm{T}\psi^* \leq \Delta\mathrm{T}$. If the transmission delay $\Delta\mathrm{T}$ is valid, the $\mathcal{BS}s$ executes the authentication check, otherwise the request is rejected.

Step 3. The base station $\mathcal{BS}s$ executes authentication check $h_{\mathrm{N}} = \mathcal{H}\left(\mathrm{Pk}_{\mathbb{N}} / Mask_{\mathcal{BS}s}\right)$, if valid i.e. $M_{\mathbb{N}} = \mathrm{Pk}_{\mathbb{N}}\ \mathrm{I}\ Mask_{\mathcal{BS}s}$ Then $\mathcal{BS}s$ executes [Table ($\mathbb{N}$) $= M_{\mathbb{N}} / h_{\mathbb{N}}$] if the resulting hash corresponds to the saved hash in the table, then $\mathbb{N}$ is authentic. Otherwise, reject the request.

Step 4. The base station sends an acknowledgement message to $\mathbb{N}$, such that $\text{Å}c\mathcal{K} = (\mathcal{ID}_i, c\mathrm{T}\psi, \mathrm{Pk}_{\mathcal{BS}s})$

Step 5. Node receives the $\text{Å}c\mathcal{K}$ message, verify the validity of the message as described in step 2. If passed, $\mathbb{N}$ proceed to step 3. Hence, both nodes have been successfully authenticated. Then

Step 6. Node $\mathbb{N}$ retrieve $\mathbb{S}_{\mathcal{V}}$ and computes the pre-pairwise key $\mathbb{P}\kappa_{\mathbb{N}} = \mathcal{H}\ (\Theta \,\|\, \mathcal{ID}_{\mathbb{N}})$

Step 7. Similarly, the base station retrieves $\mathbb{S}_{\mathcal{V}}$ and computes $\mathbb{P}\kappa_{\mathcal{BS}s} = \mathcal{H}\ (\Theta \,\|\, \mathcal{ID}_{\mathcal{BS}s})$

Step 8. The base station $\mathcal{BS}s$ and $\mathbb{N}$ exchange $\mathcal{M}$ = $[\,\mathcal{ID}_{\mathcal{BS}s},\mathbb{P}\kappa_{\mathcal{BS}s}\|\mathrm{Pk}_{\mathcal{BS}s}] \leftrightarrow [\mathcal{ID}_{\mathbb{N}},\mathbb{P}\kappa_{\mathbb{N}}\|\mathrm{Pk}_{\mathbb{N}}]$ to derive a common key.

Step 9. The base station $\mathcal{BS}s$ computes $\mathrm{K}_{\mathcal{P}}\mathcal{B} = (\mathcal{K}_{\mathcal{DH}}\mathcal{BS}s, \mathbb{N} \oplus \mathbb{P}\kappa_{\mathbb{N}} \oplus \mathrm{Pk}_{\mathbb{N}})$.

Step 10. Node $\mathbb{N}$ also computes $\mathrm{K}_{\mathcal{P}}\mathbb{N} = (\mathcal{K}_{\mathcal{DH}}\mathbb{N}\,,\ \mathcal{BS}s \oplus \mathbb{P}\kappa_{\mathcal{BS}s} \oplus \mathrm{Pk}_{\mathcal{BS}s})$.

Thus, $\mathcal{BS}s$ and $\mathbb{N}$ share the same outputted pairwise keys. $\mathrm{K}_{\mathcal{P}}\mathcal{B} = \mathrm{K}_{\mathcal{P}}\mathbb{N} = (.\mathcal{K}_{\mathcal{DH}}\mathbb{N}\,.,\ \mathcal{BS}s \oplus \mathbb{P}\kappa_{\mathbb{N}} \oplus \mathrm{Pk}_{\mathbb{N}})$.

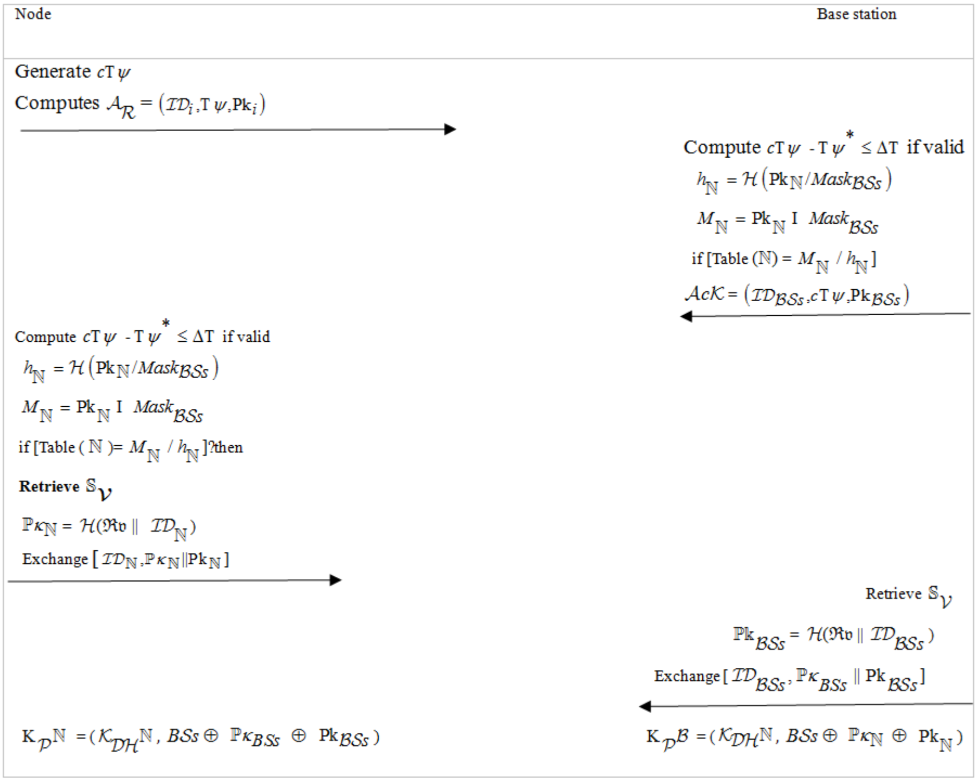

Node
Base station
Generate $c\mathrm{T}\psi$
Computes $\mathcal{A}_{\mathcal{R}} = (\mathcal{ID}_i,\mathrm{T}\psi,\mathrm{Pk}_i)$
Compute $c\mathrm{T}\psi - \mathrm{T}\psi^{*} \le \Delta\mathrm{T}$ if valid
$h_{\mathbb{N}} = \mathcal{H}(\mathrm{Pk}_{\mathbb{N}}/Mask_{\mathcal{BS}s})$
$M_{\mathbb{N}} = \mathrm{Pk}_{\mathbb{N}}\ \mathrm{I}\ Mask_{\mathcal{BS}s}$
if [Table ($\mathbb{N}$) = $M_{\mathbb{N}} / h_{\mathbb{N}}$]
$\mathcal{A}c\mathcal{K} = (\mathcal{ID}_{\mathcal{BS}s},c\mathrm{T}\psi,\mathrm{Pk}_{\mathcal{BS}s})$
Compute $c\mathrm{T}\psi - \mathrm{T}\psi^{*} \le \Delta\mathrm{T}$ if valid
$h_{\mathbb{N}} = \mathcal{H}(\mathrm{Pk}_{\mathbb{N}}/Mask_{\mathcal{BS}s})$
$M_{\mathbb{N}} = \mathrm{Pk}_{\mathbb{N}}\ \mathrm{I}\ Mask_{\mathcal{BS}s}$
if [Table ($\mathbb{N}$)= $M_{\mathbb{N}} / h_{\mathbb{N}}$]?then
Retrieve $\mathbb{S}_{\mathcal{V}}$
$\mathbb{P}\kappa_{\mathbb{N}} = \mathcal{H}(\mathfrak{R}\mathfrak{v} \| \mathcal{ID}_{\mathbb{N}})$
Exchange [$\mathcal{ID}_{\mathbb{N}},\mathbb{P}\kappa_{\mathbb{N}}\|\mathrm{Pk}_{\mathbb{N}}$]
Retrieve $\mathbb{S}_{\mathcal{V}}$
$\mathrm{Pk}_{\mathcal{BS}s} = \mathcal{H}(\mathfrak{R}\mathfrak{v} \| \mathcal{ID}_{\mathcal{BS}s})$
Exchange [$\mathcal{ID}_{\mathcal{BS}s}, \mathbb{P}\kappa_{\mathcal{BS}s} \| \mathrm{Pk}_{\mathcal{BS}s}$]
$\mathrm{K}_{\mathcal{P}}\mathbb{N} = (\mathcal{K}_{\mathcal{DH}}\mathbb{N}, \mathcal{BS}s \oplus \mathbb{P}\kappa_{\mathcal{BS}s} \oplus \mathrm{Pk}_{\mathcal{BS}s})$
$\mathrm{K}_{\mathcal{P}}\mathcal{B} = (\mathcal{K}_{\mathcal{DH}}\mathbb{N}, \mathcal{BS}s \oplus \mathbb{P}\kappa_{\mathbb{N}} \oplus \mathrm{Pk}_{\mathbb{N}})$

Fig. 5. Authentication and pairwise key establishment

Algorithm 2. Algorithm for authentication and pairwise key establishment protocol

```
Input: { $\mathrm{P}, \mathcal{ID}_i, \mathcal{H}, \mathrm{T}\psi, bit\_mask$ }
Output: [Table ( $\mathcal{BS}s$ ) = $M_{\mathcal{BS}s}$ / $h_{\mathcal{BS}s}$]
    1: $\mathbb{N}$ generate $c\mathrm{T}\psi$
    2: Compute $\mathcal{A}_{\mathcal{R}} = (\mathcal{ID}_i,\mathrm{T}\psi,\mathrm{Pk}_i)$
    3: $\mathbb{N}$ send $\mathcal{A}_{\mathcal{R}}$ to $\mathcal{BS}s$
    4: Check the freshness of the $\mathcal{A}_{\mathcal{R}}$
    5: Compute $c\mathrm{T}\psi - \mathrm{T}\psi^{*} \le \Delta\mathrm{T}$
    6: If $\Delta\mathrm{T}$ is valid, do
    7: Execute $h_{\mathbb{N}} = \mathcal{H}(\mathrm{Pk}_{\mathbb{N}}/Mask_{\mathcal{BS}s})$
    8: While $M_{\mathbb{N}} = \mathrm{Pk}_{\mathbb{N}}\ \mathrm{I}\ Mask_{\mathcal{BS}s}$ is valid, do
    9: Execute [Table ( $\mathbb{N}$ )= $M_{\mathbb{N}} / h_{\mathbb{N}}$] else stop, do
    10: Send $\mathcal{A}c\mathcal{K} = (\mathcal{ID}_{\mathcal{BS}s},c\mathrm{T}\psi,\mathrm{Pk}_{\mathcal{BS}s})$
    11: $\mathbb{N}$ receive $\mathcal{A}c\mathcal{K}$. perform steps 4-9, and
    12: $\mathbb{N}$ and $\mathcal{BS}s$ Authenticate each other
    13: While authentication is established do
```

```
14: $\mathbb{N}$ retrieve $\mathbb{S}_{\mathcal{V}}$
15: Compute $\mathbb{P}\kappa_{\mathbb{N}} = \mathcal{H}(\mathfrak{Rv} \| \mathcal{ID}_{\mathbb{N}})$
16: $\mathcal{BSs}$ retrieve $\mathbb{S}_{\mathcal{V}}$ then
17: Compute $\mathbb{P}\kappa_{\mathcal{BSs}} = \mathcal{H}(\mathfrak{Rv} \| \mathcal{ID}_{\mathcal{BSs}})$
18: $\mathbb{N}$ and $\mathcal{BSs}$ exchange $(\mathcal{ID}_{\mathbb{N}} \oplus \mathbb{P}\kappa_{\mathbb{N}} \oplus \mathrm{Pk}_{\mathbb{N}}) \leftrightarrow (\mathcal{ID}_{\mathcal{BSs}} \oplus \mathbb{P}\kappa_{\mathcal{BSs}} \oplus \mathrm{Pk}_{\mathcal{BSs}})$ to derive common keys
19: $\mathbb{N}$ compute $\mathrm{K}_{\mathcal{P}}\mathbb{N}. = (\mathcal{K}_{\mathcal{DH}}\mathbb{N}, \mathcal{BSs} \oplus \mathbb{P}\kappa_{\mathcal{BSs}} \oplus \mathrm{Pk}_{\mathcal{BSs}})$
20: $\mathcal{BSs}$ compute $\mathrm{K}_{\mathcal{P}}\mathcal{B} = (\mathcal{K}_{\mathcal{DH}}\mathbb{N}, \mathcal{BSs} \oplus \mathbb{P}\kappa_{\mathbb{N}} \oplus \mathrm{Pk}_{\mathbb{N}})$
21: If $\mathrm{K}_{\mathcal{P}}\mathbb{N} = \mathrm{K}_{\mathcal{P}}\mathcal{B} = \mathrm{K}_{\mathcal{P}}$ then
22: $\mathbb{N}$ and $\mathcal{BSs}$ share the same pairwise keys
23: End if
```

4.3 Key revocation phase

We assume the presence of an Intrusion Detection System (IDS) in the network to detect malicious nodes. The base station works as a server, managing and receiving reports of malicious activities sent by the IDS. However, this study does not focus on the details of the IDS or indicators of compromise, as these are beyond the scope of this work. The revocation process begins when the IDS reports a compromised node to the base station.

This process consists of the following steps, shown in Figure 5, with the corresponding algorithm presented in Algorithm 3.

Step 1**.** The base station $\mathcal{BSs}$ received $\mathcal{L}_{c\mathcal{N}} = (c\mathcal{N}_{1,} \, c\mathcal{N}_{2,} \, \ldots\ldots.c\mathcal{N}_{n,})$ from the IDS.

Step 2. The base station deletes $\mathrm{K}_{\mathcal{P}}$ shared with all the compromised nodes in the list from its memory.

Step 3. The base station updates its memory by deleting the authentication table and $\mathcal{ID}s$ that correspond to the compromised nodes in the list.

Step 4. The base station retrieves $\mathbb{S}_{\mathcal{V}}$ and compute the new pairwise key $\mathrm{K}^{*}_{\mathcal{PB}}$ by executing steps 8 – 9 in section 4.2.

Step 5. The base station broadcast $\mathcal{ML}_{c\mathcal{N}}$ that contains $(\mathcal{ID}_{\mathcal{BSs}}, \mathcal{L}_{c\mathcal{N}}, \mathrm{K}^{*}_{\mathcal{PB}}, c\mathrm{T}\psi)$ list of compromised nodes in the vicinity of the node $\mathbb{N}$.

Step 6. Node $\mathbb{N}$ receives $\mathcal{ML}_{c\mathcal{N}}$, compute $c\mathrm{T}\psi - \mathrm{T}\psi^{*} \leq \Delta\mathrm{T}$, If valid, decrypt $\mathcal{ML}_{c\mathcal{N}}$ and replace $\mathrm{K}_{\mathcal{P}}$ with $\mathrm{K}^{*}_{\mathcal{PB}}$. Then compares the $\mathcal{ID}$ in $\mathcal{L}_{c\mathcal{N}}$ with the $\mathcal{ID}s$ in its memory. If not valid, $\mathbb{N}$ ignored $\mathcal{ML}_{c\mathcal{N}}$, otherwise, execute steps 1-3.

Step 7. Node $\mathbb{N}$ send $\mathcal{AcK}$ that contains $(\mathcal{ID}_{\mathbb{N}}, c\mathrm{T}\psi)$ to $\mathcal{BSs}$.

Step 8. The base station $\mathcal{BSs}$ receive $\mathcal{AcK}$ decrypt and send updates its memory.

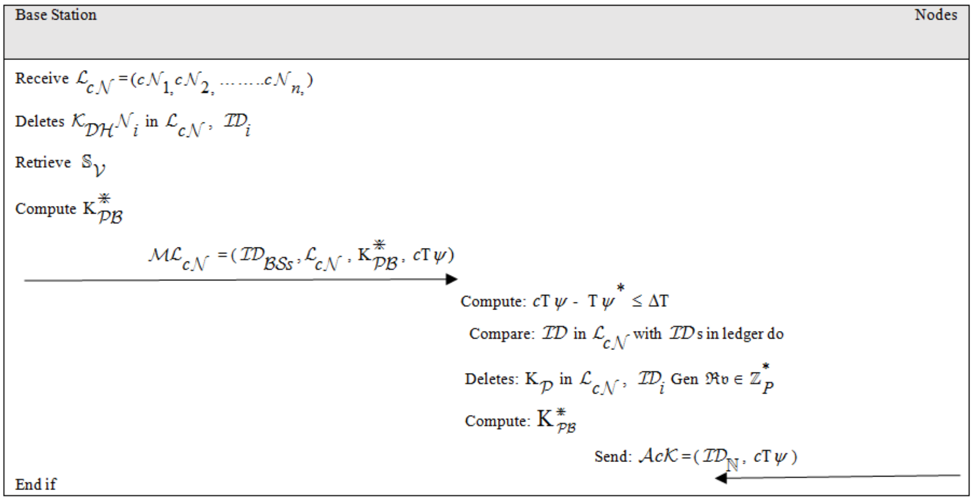

Base Station
Nodes
Receive $\mathcal{L}_{c\mathcal{N}} = (c\mathcal{N}_{1,} c\mathcal{N}_{2,} \ldots\ldots.c\mathcal{N}_{n,})$
Deletes $\mathcal{K}_{\mathcal{DH}}\mathcal{N}_{i}$ in $\mathcal{L}_{c\mathcal{N}}$, $\mathcal{ID}_{i}$
Retrieve $\mathbb{S}_{\mathcal{V}}$
Compute $\mathrm{K}^{*}_{\mathcal{PB}}$
$\mathcal{ML}_{c\mathcal{N}} = (\mathcal{ID}_{\mathcal{BSs}}, \mathcal{L}_{c\mathcal{N}}, \mathrm{K}^{*}_{\mathcal{PB}}, c\mathrm{T}\psi)$
Compute: $c\mathrm{T}\psi - \mathrm{T}\psi^{*} \leq \Delta\mathrm{T}$
Compare: $\mathcal{ID}$ in $\mathcal{L}_{c\mathcal{N}}$ with $\mathcal{ID}$s in ledger do
Deletes: $\mathrm{K}_{\mathcal{P}}$ in $\mathcal{L}_{c\mathcal{N}}$, $\mathcal{ID}_{i}$ Gen $\mathfrak{Rv} \in \mathbb{Z}^{*}_{P}$
Compute: $\mathrm{K}^{*}_{\mathcal{PB}}$
Send: $\mathcal{AcK} = (\mathcal{ID}_{\mathbb{N}}, c\mathrm{T}\psi)$
End if

Fig. 6. Key Revocation Phase

Algorithm 3. Algorithm for key revocation protocol

```
Input: $\{ \mathrm{P}, \mathcal{L}_{c\mathcal{N}}, \mathcal{ID}_i \mathrm{K}_{\mathcal{P}}, \mathrm{T}\psi \}$
Output: $\mathcal{A}c\mathcal{K} = (\mathcal{ID}_i \cdot c\mathrm{T}\psi)\ \mathbb{S}\mathbb{k}$
    1: $\mathcal{BS}s$ receive $\mathcal{L}_{c\mathcal{N}} = ( c\mathcal{N}_{1,} c\mathcal{N}_{2,} \ldots\ldots.. c\mathcal{N}_{n,} )$
    2: Deletes $\mathrm{K}_{\mathcal{P}}$ in $\mathcal{L}_{c\mathcal{N}}$
    3: Then delete $\mathcal{ID}_i$ and authentication table $\mathcal{L}_{c\mathcal{N}}$
    4: Retrieve $\mathbb{S}_{\mathcal{V}}$ then compute $\mathrm{K}^{\circledast}_{\mathcal{PB}}$
    5: Compute $\mathbb{S}\mathbb{k} = \mathcal{H}(\Delta\mathrm{T} \,||\, \mathcal{ID}_{\mathcal{BS}})$
    6: Broadcast $\mathcal{ML}_{c\mathcal{N}} = (\mathcal{ID}_{\mathcal{BS}}, \mathcal{L}_{c\mathcal{N}}, K^{\circledast}_{\mathcal{PB}}, c\mathrm{T}\psi)$
    7: $\mathbb{N}$ Receive $\mathcal{ML}_{c\mathcal{N}}$ then Compute $c\mathrm{T}\psi - \mathrm{T}\psi^{*} \leq \Delta\mathrm{T}$ to validate $\mathcal{ML}_{c\mathcal{N}}$ do
    8: Decrypt $\mathcal{ML}_{c\mathcal{N}}$ then compares $\mathcal{ID}$ in $\mathcal{L}_{c\mathcal{N}}$ with $\mathcal{ID}$ in its memory do
    9: Execute step 2-3 if
    10: Else ignore $\mathcal{ML}_{c\mathcal{N}}$ then
    11: Send $\mathcal{A}c\mathcal{K} = (\mathcal{ID}_{\mathbb{N}}, c\mathrm{T}\psi)$
    12: End if
```

4.4 Key renewal phase

Regardless of how a cryptographic key is established, utilized, or stored, it will eventually lose its validity and must be renewed. Extended use of the same key through multiple sessions increases its vulnerability to various attacks [38]. In this study, we consider the revocation of compromised nodes as the basis for renewing cryptographic keys.

4.4.1 Pairwise key renewal subphase.

The process is designed to generate new pairwise keys alongside the key revocation process. The process consists of three steps.

Step 1. The base station retrieves $\mathbb{S}_{\mathcal{V}}$ and compute $\mathbb{P}\mathbb{k}_{\mathbb{N}}$.

Step 2. The base station $\mathcal{BS}s$ and node $\mathbb{N}$ exchange $\mathcal{M} = [\mathcal{ID}_{\mathcal{BS}s}, \mathbb{P}\kappa_{\mathcal{BS}s} || \mathrm{Pk}_{\mathcal{BS}s}] \leftrightarrow [\mathcal{ID}_{\mathbb{N}}, \mathbb{P}\kappa_{\mathbb{N}} || \mathrm{Pk}_{\mathbb{N}}]$

Step 3. The base station $\mathcal{BS}s$ and node $\mathbb{N}$ compute $(\mathcal{K}_{\mathcal{DH}}\mathcal{BS}s, \mathbb{N} \oplus \mathbb{P}\kappa_{\mathbb{N}} \oplus \mathrm{Pk}_{\mathbb{N}}) = (\mathcal{K}_{\mathcal{DH}}\mathbb{N}, \mathcal{BS}s \oplus \mathbb{P}\kappa_{BSs} \oplus \mathrm{Pk}_{\mathcal{BS}s})$.

Finally, we repeat step 4 from Section 4.3 to renew the pairwise key

5. Formal and Informal Security Evaluation of the Proposed Protocols

5.1 Formal validation of the protocol with BAN logic

This section describes the formal validation analysis of the protocol design using Burrows Abadi and Needham's (BAN) logic. BAN logic is a popular formal model used to establish logical structure set bases to verify the security of protocols and prove the correctness of authentication and key management schemes [6].

5.2 The notations and basic rules of BAN logic.

To validate a protocol using BAN logic, the participants and their beliefs must be established at the beginning. Additionally, specific BAN logic notation must be used to express these beliefs. The symbols Ƥ and Q represent the principals, while the variable X depends on the specific statement. BAN logic notations and rules are presented as follows.

1. Ƥ Ƀ X: Ƥ believes X, that is, the principal Ƥ consider X as true.
2. Ƥ ĉ X: Ƥ sees X, for instance, once the principal Ƥ obtained a message that includes X, therefore Ƥ sees X.
3. Ƥ š X: Ƥ said X, the principal Ƥ will always believe X when it said the message that contains the statement X was sent by Ƥ.
4. Ƥ © X: Ƥ controls X, that is, Ƥ has authority on X, in other words, Ƥ has jurisdiction over X.

5. # (X): it represents that the formula X is fresh, that is X is not contained in any message sent at any time preceding the current protocol iteration.
6. $(\mathrm{X})_k$: this implies that formula X is encrypted by a key $\mathcal{K}$.
7. $\mathrm{P} \xleftrightarrow{K} \mathrm{Q} : \mathcal{K}$ is a shared key that is only known to Ƥ and Q for their communications.
8. $\rightarrow \mathrm{Pk}$ Ƥ: the key Pk is Ƥ's public key.
9. $\mathrm{P} \overset{\mathrm{X}}{\rightleftharpoons} \mathrm{Q}$: : the formula X is a known secret to only Ƥ and Q and other participants that they reveal it to.
10. $(\mathcal{K})_{\mathcal{H}}$ this denotes that the key $\mathcal{K}$ is hashed using the hashing process.
11. $\mathrm{Pk} \cap$ Mask: this denotes that the received public key bit corresponds with *m* indexes in the mask.

5.3 BAN logic rules

R1: Message meaning rule:
R2: Key exchange rule:
R3: Timestamp verification rule:
R4: Jurisdiction rule:
R5: Credential possession rule:
R6: Authentication check rule:
R7: Believe rule:
R8: Common key rule:

5.4 Proof of key revocation scheme using BAN logic

This section outlines the phased evaluation and proof of the proposed scheme, using BAN logic postulates from subsection 5.2

1. Registration phase. In this phase, assume that $\mathcal{BS}s$ represents the base station, $\mathbb{N}$ represent the network node, and KGC denotes the key generation center. Let $\mathrm{X}_{s\kappa}$ be the partial secret key, $\mathcal{R}eg\mathbb{R}$ the registration request, $\mathbb{S}_{\mathcal{V}}$ and the secret value. M_1 denotes the first message transmitted by the base station/network nodes and KGC in this phase. M_2 represents the second message transmitted from the base station and network nodes.

i. The idealized message formats

As shown in Figure 4 and based on the assumptions made for the registration phase, the idealized message format for this phase is as follows. Note that the analysis comprises only the messages that contribute directly to the process. The idealized message format for the registration phase is presented in equations 8 and 9.

$$\mathrm{M}_1 : \mathcal{BS}s \ /\mathbb{N} \ \longrightarrow \ \mathrm{KGC} : [\mathcal{ID}_i \mathcal{H} \ (\mathrm{X}_{s\kappa})] \ \ Kg_{pub} \tag{8}$$

$$\mathrm{M}_2 : \mathrm{KGC} \ \longrightarrow \ \mathcal{BS}s \ /\mathbb{N} : [\mathbb{S}_{\mathcal{V}}] \ \ \mathrm{Pk}_i \tag{9}$$

ii. Assumptions made for the analysis of the registration phase

A1: $\mathcal{BS}s$ Ƀ (KGC, Kg_{pub}) implies that the base station $\mathcal{BS}s$ believes that Kg_{pub} is the public key of the key generation center KGC.

A2: $\mathbb{N}$ Ƀ (KGC, Kg_{pub}) implies that the network node $\mathbb{N}$ believes that Kg_{pub} is the public key of the key generation center KGC.

A3: $\mathcal{BS}s$ Ƀ $\mathrm{Pk}\,(\mathcal{BS}s \ \mathrm{X}_{s\kappa})$ implies that the base station $\mathcal{BS}s$ believes that $\mathrm{X}_{s\kappa}$ is his retrieve secret value that was sent to the KGC.

A4: $\mathcal{BS}s$ Ƀ # ($\mathcal{BS}s$) implies that the base station $\mathcal{BS}s$ believes in the presence of a private key that matches its public key.

A5: $\mathcal{BS}s$ Ƀ # ($\mathbb{N}$) implies that the base station believes that the decentralized node $\mathbb{N}$ possesses a private key that matches its public key.

A6: KGC Ƀ $\mathcal{BS}s / \mathbb{N} \Longrightarrow (\mathrm{M}_1, \ \mathcal{R}eg\mathbb{R})$ implies that the key generation center KGC believe that the base station and the decentralized node control the generation of messages M_1 and the registration request $\mathcal{R}eg\mathbb{R}$.

A7: KGC Ƀ # (M_1 implies that the key generation center believes that a segment of the message M_1 is fresh and has not been sent before.

A8: $\mathcal{BS}s$ / $\mathbb{N}$ Ƀ # (M_2) implies that the base station $\mathcal{BS}s$ and the decentralized node believes that a segment of the message M_2 is fresh and has not been previously transmitted.

A9: $\mathcal{BS}s$ / $\mathbb{N}$ Ƀ KGC ($\mathbb{S}_{\mathcal{V}}$) implies that the base station and the network node believe that $\mathbb{S}_{\mathcal{V}}$ is the secret value computed and sent to the key generation center.

iii. Goals Definition

This section presents the goals to be achieved in this phase of the protocol. The goal is to ensure secure registration of the base station and the decentralised node with the KGC. Furthermore, the variables KGC, $\mathcal{BS}s$ and $\mathbb{N}$ represent the principals. Thus, the $\mathcal{BS}s$ / $\mathbb{N}$ trust the computed secret value by the KGC and believe that it is true.

Goal1**:** KGC Ƀ $\mathcal{BS}s$ / $\mathbb{N}$ š M_1, means that the KGC should believe that the base station and decentralized node have said message M_1, which is the registration request.

Goal2: KGC Ƀ M_1, implies that the key generation center should believe that the message M_1, which comprises the registration request is true.

Goal3: $\mathcal{BS}s$ / $\mathbb{N}$ Ƀ KGC š $\mathbb{S}_{\mathcal{V}}$ (M_2), implies that the base station and the decentralised node should believe that the secret value and the message are said by the key generation center.

Goal4: KGC Ƀ # M_1, implies that the key generation center KGC should believe that the message M_1 is fresh and has not been transmitted previously.

Goal5: $\mathcal{BS}s$ / $\mathbb{N}$ Ƀ # M_2, implies that the base station and the decentralized node should believe that the message M_2 is fresh and has not been transmitted before.

iv. Analysis

By utilizing the messaging rule in section 5.3 on A_1, A_2, A_6, A_7 and M_1. G_1 is satisfactorily achieved. In other words, if the KGC receive a message M_1 encrypted with (KGC, Kg_{pub}) from $\mathcal{BS}s$ and $\mathbb{N}$, and also $\mathcal{BS}s$ and $\mathbb{N}$ believes that Kg_{pub} is the public key of the KGC. Then KGC trusts that $\mathcal{BS}s$ and $\mathbb{N}$ controls the generation of M_1 and the $\mathcal{R}eg\mathbb{R}$, thus, the KGC should believe that $\mathcal{BS}s$ and $\mathbb{N}$ said M_1 which includes $\mathcal{R}eg\mathbb{R}$. This implies that KGC Ƀ $\mathcal{BS}s$ and $\mathbb{N}$ š $\mathcal{R}eg\mathbb{R}$, therefore G_1 is achieved. The analysis is presented as a logical expression in equation 10.

$$\frac{\mathcal{BS}s\mathrm{B}\left(\mathrm{KGC},Kg_{pub}\right),\mathbb{N}\mathrm{B}\left(\mathrm{KGC},Kg_{pub}\right),\mathrm{KGCB}\mathcal{BS}s/\mathbb{N}\mathrm{B}\left(\mathrm{M}_1,\mathcal{R}eg\mathbb{R}\right)}{\mathrm{KGCB}\mathcal{BS}s/\mathbb{N}\mathrm{sM}_1} \tag{10}$$

By utilizing the key exchange rule from section 5.3 on A_3, A_{10} and M_2. G_3 is achieved. That is, $\mathcal{BS}s$ believes that $X_{s\kappa}$ is the retrieved secret value that was sent to the KGC. Similarly, $\mathcal{BS}s$ and $\mathbb{N}$ believe that $\mathbb{S}_{\mathcal{V}}$ is the secret value computed and sent to the KGC. Therefore, the $\mathcal{BS}s$ and $\mathbb{N}$ believe that the $\mathbb{S}_{\mathcal{V}}$ and M_2 were said by the KGC, which implied that G_3 is achieved. The analysis is presented as a logical expression in equation 11.

$$\frac{\mathcal{BS}s\mathrm{B}(\mathrm{KGC},\mathrm{X}_{s\kappa})\ \mathcal{BS}s/\mathbb{N}\mathrm{B}(\mathrm{KGC},\mathbb{S}_{\mathcal{V}})}{\mathcal{BS}s/\mathbb{N}\mathrm{BKGC}\mathbb{S}_{\mathcal{V}}\mathrm{M}_2}\ \text{therefore, } G_3 \text{ is achieved} \tag{11}$$

By applying the belief rule from section 5.3**,** G_2 is achieved. Given that, from G_1 and G_3 it can be confirm that KGC believe the message M_1. Then applying the belief rule on A_7, which implies that KGC received the message M_1 from $\mathcal{BS}s$ and $\mathbb{N}$. Therefore, KGC should believe that the message M_1 which comprises the registration request $\mathcal{R}eg\mathbb{R}$ is true. The analysis is presented as a logical expression in equation 12.

$$\frac{\mathrm{KGCB}\mathcal{BS}s\mathrm{and}\mathbb{N}\left(\mathrm{M}_1,\mathcal{R}eg\mathbb{R}\right)}{\mathrm{KGCBM}_1} \text{ therefore, } G_2 \text{ is achieved} \tag{12}$$

By utilizing the timestamp verification rule from section 5.3 on A_8. G_4 is achieved. Precisely, if the KGC believes that the message M_1 is fresh and has not been previously transmitted, then G_4 achieved. Therefore it implies that, KGC believes that M_1 is fresh and has not been transmitted previously. The analysis is presented as a logical expression in equation 13.

$$\frac{\mathrm{KGCB}\#\left(\mathrm{M}_1\right)}{\mathrm{KGCB}\#\mathrm{M}_1} \text{therefore } G_4 \text{ is achieved} \tag{13}$$

By utilizing the timestamp verification rule on A_9, which implies $\mathcal{BS}s$ and $\mathbb{N}$ believes the message M_2 is fresh and has not been transmitted before. Hence, G_5 is achieved. The analysis is presented as a logical expression in equation 14.

$$\frac{\mathcal{BS}s/\mathbb{N}\mathrm{B}\#\left(\mathrm{M}_2\right)}{\mathcal{BS}s/\mathbb{N}\mathrm{B}\#\mathrm{M}_2} \text{ therefore } G_5 \text{ is achieved} \tag{14}$$

2. *Key establishment phase.* This section assume that $\mathcal{BS}s$ is the base station, $\mathbb{N}$ is the network node, KGC is the key generation center, $\mathrm{Pk}_{\mathbb{N}}$ is the public key of $\mathbb{N}$, $\mathrm{Pk}_{\mathcal{BS}s}$ is the public key of the base station, $\mathcal{K}_{\mathcal{DH}}\mathbb{N}$, $\mathcal{BS}s$ is the shared key between $\mathbb{N}$ and $\mathcal{BS}s$, $\mathcal{A}uR$ is the authentication request in M_1, $c\mathrm{T}\psi$ is the current timestamp, $h_{\mathbb{N}} / M_{\mathbb{N}}$ are the authentication checks, $\mathcal{A}c\mathcal{K}$ acknowledgement message in M_2, $\mathbb{S}_{\mathcal{V}}$ is the secret value, $\mathbb{Pk}_{\mathbb{N}}$ is the pre-pairwise key of $\mathbb{N}$, $\mathbb{Pk}_{\mathcal{BS}s}$ is the pre-pairwise key of $\mathcal{BS}s$, $\mathrm{K}_{\mathcal{P}}\mathbb{N}$ is the pairwise key of $\mathbb{N}$, M_3 is the exchange message sent to $\mathcal{BS}s$ by $\mathbb{N}$, and M_4 is the exchange message sent to N by $\mathcal{BS}s$.

i. The idealized message formats

As shown in Figure 5 and using the assumptions made for the key establishment phase, the idealized message format for this phase is as follows. Specifically, the analysis involves only the messages that contribute directly to the protocol. The idealized message format for the key establishment phase is presented in equations 15,16,17 and 18.

$$\mathrm{M}_1 : \mathbb{N} \rightarrow \mathcal{BS}s : \mathcal{A}_{\mathcal{R}} \ \left(\mathcal{ID}_i, \mathrm{T}\psi, \mathrm{Pk}_i\right) \tag{15}$$

$$\mathrm{M}_2 : \mathcal{BS}s \rightarrow \mathbb{N} : \mathcal{A}c\mathcal{K} \ \left(\mathcal{ID}_{\mathcal{BS}s}, c\mathrm{T}\psi, \mathrm{Pk}_{\mathcal{BS}s}\right) \tag{16}$$

$$\mathrm{M}_3 \text{: } \mathbb{N} \ \longrightarrow \ \mathcal{BS}s \ : [\mathcal{ID}_{\mathbb{N}}, \mathbb{Pk}_{\mathbb{N}} \parallel \mathrm{Pk}_{\mathbb{N}}] \tag{17}$$

$$\mathrm{M}_4 : \mathcal{BS}s \rightarrow \mathbb{N} : [\mathcal{ID}_{\mathcal{BS}s}, \mathbb{Pk}_{\mathcal{BS}s} \parallel \mathbb{Pk}_{\mathcal{BS}s}] \tag{18}$$

ii. Assumptions made for the analysis of the key establishment phase

A_1: $\mathcal{BS}s \text{ B } \mathbb{N} \xleftrightarrow{KpN} \mathcal{BS}s$ implies that the pairwise key is only known to $\mathbb{N}$ and $\mathcal{BS}s$ for their communication.

A_2: $\mathcal{BS}s \text{ B } \mathbb{N} \xleftrightarrow{Kp\mathcal{BS}s} \mathcal{BS}s$ implies that the pairwise key is only known to $\mathcal{BS}s$ and $\mathbb{N}$ for their communication.

iii. Goal definition:

The goal of this phase is to achieve mutual authentication among the decentralized node and the base station. Furthermore, the base station and the decentralized node must verify the validity of M_1 and M_2. The following goals are expected to be achieved in this phase.

Goal 1: $\mathcal{BS}s \text{ B } \mathbb{N}$ š M_1, means that the base station $\mathcal{BS}s$ should believe that the decentralized node $\mathbb{N}$ has said message M_1 which includes $\mathcal{A}c\mathcal{R}$. Thus, $\mathcal{BS}s$ authenticates the network node $\mathbb{N}$.

Goal 2 $\mathbb{N}$ B $\mathcal{BS}s$š M_2, implies that the network node $\mathbb{N}$ should believe that the base station $\mathcal{BS}s$ has said M_2, which comprises $\mathcal{A}c\mathcal{K}$. Thus, $\mathbb{N}$ authenticate the base station $\mathcal{BS}s$.

iv. Analysis

By utilizing the authentication check rule from section 5.3 on A_2 and M_3. G_1 is achieved, specifically, if the $\mathcal{BS}s$ receive the message M_3 that comprises of $K_{\mathcal{P}}\mathbb{N}$ from the network node $\mathbb{N}$, and $\mathcal{BS}s$ believes that $K_{\mathcal{P}}\mathbb{N}$ is the pairwise key known to them for communication. Then it implies that $\mathcal{BS}s$ should believe that $\mathbb{N}$ said the message M_3. Therefore $\mathbb{N}$ is authenticated, and the analysis is presented as a logical expression in equation 19.

$$\frac{\mathrm{BSs}\mathcal{C}(M_3, K_{\mathcal{P}}\mathbb{N})\mathcal{BS}sB\mathcal{BS}s \xleftrightarrow{K_{\mathcal{P}}B} \mathbb{N}}{\mathcal{BS}sB\mathbb{N}sM_3} \text{ therefore } G_1 \text{ is achieved} \tag{19}$$

Similarly, by utilizing the following authentication rule on A_1 and M_4, G_2 is achieved. In other words, if $\mathbb{N}$ received the message M_4 which is a common key from the base station, and $\mathbb{N}$ believes that $K_{\mathcal{P}}\mathbb{N}$ is the pairwise key known to them for communication. Then, it implies that $\mathbb{N}$ should believe that $\mathcal{BS}s$ said the message M_4. Therefore, $\mathcal{BS}s$ is authentic and G_2 is achieved. The analysis is presented as a logical expression in equation 20.

$$\frac{\mathbb{N}\, C(M_4, K_{\mathcal{P}}\mathcal{B})\mathbb{N}B\mathcal{BS}s \xleftrightarrow{K_{\mathcal{P}}\mathbb{N}} \mathbb{N}}{\mathbb{N}B\mathcal{BS}ssM_4} \text{ thus, } G_2 \text{ is achieved} \tag{20}$$

3. Key revocation phase. In this section, assume that $\mathcal{BS}s$ is the base station, $\mathbb{N}$ is the network node, $\mathcal{L}_{c\mathcal{N}}$, is the list of compromised nodes, $K_{\mathcal{P}}$ is the pairwise key shared between $\mathcal{BS}s$ and $\mathbb{N}$, $\mathcal{ML}_{c\mathcal{N}}$ is the broadcast message containing the list of compromised nodes, $\mathcal{A}c\mathcal{K}$ is the acknowledgement message sent by $\mathbb{N}$ to $\mathcal{BS}s$, M_1 is the first message exchange between the base station $\mathcal{BS}s$ and the network nodes $\mathbb{N}$, M_2 is the second message in the protocol exchanged between the network node and base station.

i. The idealized message formats

As presented in Figure 6 and using the assumptions made for the key revocation phase, the idealized message format for this phase is as follows. The idealized message format for the key revocation phase is presented in equations 21 and 22.

$$M_1 : \mathcal{BS}s \to \mathbb{N} : \mathcal{ML}_{c\mathcal{N}}\ (\mathcal{ID}_{\mathcal{BS}s}, \mathcal{L}_{c\mathcal{N}},\ K^{\ast}_{\mathcal{PB}},\ c\mathrm{T}\psi) \tag{21}$$

$$M_2 : \mathbb{N} \to \mathcal{BS}s : \mathcal{A}c\mathcal{K}\ (\mathcal{ID}_{\mathbb{N}},\ c\mathrm{T}\psi) \tag{22}$$

ii. Assumptions made for the analysis of the key revocation phase

A_1: $\mathbb{N}$ Ƀ ($\mathcal{BS}s$, $K^{\ast}_{\mathcal{PB}}$) implies that the network node $\mathbb{N}$ believes that $K^{\ast}_{\mathcal{PB}}$ is the new pairwise key of the base station $\mathcal{BS}s$

A_2: $\mathbb{N}$ Ƀ $\mathcal{BS}s \Rightarrow (M_1,\ \mathcal{ML}_{c\mathcal{N}})$ implies that the network node $\mathbb{N}$ believe that the base station $\mathcal{BS}s$ control the generation of M_1 and the revocation broadcast.

A_3: $\mathbb{N}$ Ƀ $\mathcal{BS}s$ š (M_1, $\mathcal{ML}_{c\mathcal{N}}$) implies that the network node $\mathbb{N}$ believes that the base station said M_1 which is the revocation broadcast

A_4: $\mathcal{BS}s$ Ƀ #($\mathbb{N}$) implies that the base station believes that the network node $\mathbb{N}$ possesses a private key that matches its public key.

A_5: $\mathcal{BS}s$ Ƀ (Pk I Mask) implies that the received public key bit corresponds with *m* indexes in the mask.

A_6: $\mathcal{BS}s$ Ƀ # (M_2) implies that the base station $\mathcal{BS}s$ believes that the message M_2 is fresh and has not been transmitted before.

iii. Goal definition

The goal of this phase is to effectively revoke the detected compromised nodes within the WBAN network. Moreover, the base station and network nodes must validate M_1 and M_2 to be true. The following goals are expected to be achieved in this phase.

G_1: $\mathbb{N}$ Ƀ $\mathcal{BS}ss$ M_1, implies that the network node $\mathbb{N}$ should believe that the base station has said M_1, which comprises of $\mathcal{L}_{c\mathcal{N}}$ Thus, $\mathbb{N}$ authenticates $\mathcal{BS}s$

G_2 **:** $\mathcal{BS}s$ Ƀ M_2 , implies that the base station $\mathcal{BS}s$ should believe that the message M_2 which includes the $\mathcal{ID}_{\mathbb{N}}$ is true.

iv. Analysis

By utilizing the credential possession rule from section 5.3 on $\mathrm{A}_1, \mathrm{A}_4$ and M_1. G_1 is achieved**.** That is, if $\mathbb{N}$ receives the message M_1 and believe the $\mathrm{K}^{⁎}_{\mathcal{PB}}$ is the new pairwise key computed by $\mathcal{BS}s$, and that $\mathbb{N}$ possesses a private key that matches its public key, then, it implies that $\mathbb{N}$ should believe that $\mathcal{BS}s$ said the message M_1, therefore $\mathcal{BS}s$ is authenticated and the analysis is presented as a logical expression in equation 23.

$$\frac{\mathbb{N}\mathrm{B}\mathcal{BS}s(\mathrm{K}^{⁎}_{\mathcal{PB}})\mathbb{N}(Sk,\mathrm{Pk})}{\mathbb{N}\mathrm{B}\mathcal{BS}s\mathrm{s}\mathrm{M}_1} \text{ hence } G_1 \text{ is achieved} \tag{23}$$

By applying the believe rule from section 5.3 on A_5 and M_2. G_2 is achieved, that is, if $\mathcal{BS}s$ receives M_2 , an acknowledgement message from $\mathbb{N}$, which implies that the received public key bit corresponds with *m* indexes in the mask. Therefore, the $\mathcal{BS}s$ should believe that the message M_2 which includes the $\mathcal{ID}_{\mathbb{N}}$ is true. And so $\mathcal{BS}s$ is authenticated, hence G_2 is achieved, the analysis is presented as a logical expression in equation 24

$$\frac{\mathbb{N}\rightarrow\mathcal{BS}s{:}\mathcal{A}c\mathcal{K},\mathrm{M}_2\left(\mathcal{ID}_{\mathbb{N}},c\mathrm{T}\psi\right)\mathrm{Pk}\cap\mathrm{Mask}}{\mathcal{BS}s\mathrm{B}\mathrm{M}_2} \tag{24}$$

Given the above analysis and validation process. it is clear that the KGC, $\mathcal{BS}s$ and network nodes seamlessly achieved an efficient and secure authenticated key agreement, revocation and renewal protocol.

5.5 Security analysis

An informal evaluation of the protocol's security is presented in this section. The protocol adopts several measures to maintain a balance between security and efficiency, including the use of lightweight algorithms, the elimination of duplicate keys to forestall key escrow problems, and minimizing the number and size of message exchanges.

i. Proposition 1. The proposed scheme protects against impersonation attempts

Proof: in this case, an attacker $\mathcal{A}$ can successfully create a shared key among two authenticated nodes $\mathbb{N}$ and $\mathbb{S}$ by pretending and impersonating the $\mathcal{ID}$ of $\mathbb{N}$ and $\mathbb{S}$. To make such happen, both $\mathcal{ID}s$ of $\mathbb{N}$ and $\mathbb{S}$, and the derive common keys between $\mathbb{N}$ and $\mathbb{S}$ must know by $\mathcal{A}$. However, such an attempt will fail in the proposed protocol since $\mathcal{A}$ can only select its $\mathcal{ID}$ randomly to compute $\mathcal{ID}_{\mathcal{A},}$ but deriving the common key is not feasible since the process contains partial secret values which are not available to $\mathcal{A}$. In addition, $\mathcal{A}$ cannot pass the authentication check provided in the protocol, because the validity of the network participants is defined by $h_{\mathbb{N}}{=}\mathcal{H}\left(\mathrm{Pk}_{\mathbb{N}}/Mask_{\mathcal{BS}s}\right)$ and $M_{\mathbb{N}}{=}\mathrm{Pk}_{\mathbb{N}}\mathrm{I}\ Mask_{\mathcal{BS}s}$.

ii. Preposition 2. The proposed solution can withstand an eavesdropping attack.

Proof: the attacker $\mathcal{A}$ can eavesdrop on public communication channels and listen to all the messages sent over the channel. Thus, in this instance $\mathcal{A}$ is expected to possess the following setup values *m*, $\mathrm{Pk}, \mathcal{S}\mathrm{k}$, $\mathcal{ID}$. However, acquiring these values does not enable $\mathcal{A}$ to compute any of the secret parameters. For instance, the 16 bits of the Pk and *m* indexed in the mask value are secure through the non-reversibility of the hash function executed on the remaining bits. Seemingly, if $\mathcal{A}$ can generate a couple of public and private values, the bits of the public key that are pointed out by the *m* indexed mask of the corresponding node in the network cannot be equal. Hence, the proposed scheme can defend against eavesdropping attacks.

iii. Preposition 3. The proposed protocol resists man-in-the-middle attacks.

Proof: an attacker $\mathcal{A}$ can leverage vulnerabilities in the open communication channel to intercept, modify then transmit the message between nodes in the network without them knowing. To abate such an attack, the proposed scheme utilizes two verifiers $\mathrm{T}\psi$ and *m*. Accordingly, since $\mathcal{A}$ does not own the secret key credentials of the node $\mathbb{N}$ and $\mathbb{S}$, therefore, the shared key cannot derive a common key nor pass the $\mathrm{T}\psi$ and *m* validation procedure. Hence, the inability of $\mathcal{A}$ to successfully transmit the intercepted message. So, the proposed protocol is secure against man-in-the-middle attacks.

iv. Preposition 4. The proposed solution is secure against replay and injection attacks.

Proof: in the attack scenario, the attacker $\mathcal{A}$ intercept and capture a valid message, then resend it into the network to gain unauthorized access. Accordingly, communicating entities can only compute the shared keys, that is, $\mathcal{K}_{\mathcal{DH}}(\mathcal{BSs},\mathbb{N}) = \mathcal{S}\mathrm{k}(\mathcal{BSs})\cdot\mathrm{Pk}(\mathbb{N})$ and $\mathcal{K}_{\mathcal{DH}}(\mathbb{N},\mathcal{BSs})=\mathcal{S}\mathrm{k}(\mathbb{N})\cdot\mathrm{Pk}(\mathcal{BSs})$ for node $\mathbb{S}$ and $\mathbb{N}$ only if they know each other's $\mathcal{ID}$s and other secret credentials in the proposed protocol. Nevertheless, it will be difficult for $\mathcal{A}$ to capture and replay transmitted messages because it does not possess the secret credentials. Furthermore, the use of timestamp $\mathrm{T}\psi$ in the pairwise key establishment, validation of the timestamps receiver system defends against replay attack, because the projected timeline for the transmission lag is short.

v. Preposition 5. The proposed scheme guarantees perfect forward secrecy.

Proof: a perfect forward secrecy resilient scheme is such that an attacker $\mathcal{A}$ with recorded messages previously transmitted cannot gain access to the used session keys of compromised nodes. In the proposed protocol, assumed $\mathcal{A}$ can access $\mathcal{S}\mathrm{k}(\mathbb{N})$ and $\mathcal{S}\mathrm{k}(\mathcal{BSs})$, together with the $\mathcal{ID}_n$ and $\mathcal{ID}_{\mathcal{BS}}$ including other secret credentials. It is still not possible for A to compute $\mathrm{K}_{\mathcal{PW}}$ without the authentication validation. Because, $\mathcal{A}'$s Pk bits that were pointed out by m will not be equal to that of the authenticated recipient node. Hence, the proposed protocol is resilient against perfect forward secrecy.

vi. Preposition 6. The proposed scheme can achieve known key security.

Proof: as part of our protocol the idea of known key security requires that the exposure or compromise of one session key should not affect the security of another session keys. For example, assume an attacker $\mathcal{A}$ obtain session keys $\mathrm{K}_{\mathcal{P}}\mathcal{B}$ and $\mathrm{K}_{\mathcal{P}}\mathbb{N}$ in a given session. Based on the session derivation method in this study, each session key is generated from the following keying credentials $\mathbb{S}_{\mathcal{V}}$, $\mathrm{X}_{\mathcal{SK}}$, $c\mathrm{T}\psi$ and η_i for each session. Hence, the session keys for each session are unique, an assurance that the proposed protocol is capable of achieving known key security.

vii. Preposition 7. The proposed protocol is resilient against desynchronization attack

Proof: in general, once the session key is established the KGC, $\mathcal{BSs}$ and $\mathbb{N}$ do not require further parameter updates. Therefore, the synchronization attack is not feasible. However, both $\mathcal{BSs}$ and $\mathbb{N}$ need to compute a common key using the $\mathbb{S}_{\mathcal{V}}$ keying parameter. Fortunately, the two computations yield the same common key, since the multiplication is commutative and guarantees the synchronization update of $\mathcal{BSs}$ and $\mathbb{N}$.

viii. Preposition 8. The proposed protocol is resilient against node capture attack

proof: this type of attack implies that the attacker has the node's secret value $\mathbb{S}_{\mathcal{V}}$ and then recover the $\mathrm{X}_{\mathcal{SK}}$. However, the attacker cannot recompute the session key, except the attacker can efficiently solve the large number's factorization problem.

ix. Preposition 9. The proposed protocol guarantees mutual authentication

proof: based on the principle of the authentication table [40], it is recognized that only authentic N and BSs can execute an authentication check. Thus, N and BSs can prove the validity of each other only when the resulting hash corresponds to the saved hash in the authentication table. Therefore, the proposed protocol guarantees mutual authentication.

x. Preposition 10. The proposed protocol supports scalability

proof: the protocol is designed to ensure seamless scalability in the network by allowing the addition or removal of nodes as the situation demands. Through the node addition and revocation features of the protocol, without compromising the performance or security of the system.

5.6 Performance analysis

Due to WBANs' limited computational and memory capacity, minimizing computing and storage overhead becomes a crucial performance factor when designing cryptographic protocols. Consequently, this work prioritizes computational and storage efficiency as key objectives in protocol design. In this subsection, we present a detailed analysis of the scheme's performance in terms of the stated performance parameters. Our approach considers adopted execution time values based on the widely utilized standard implementation framework in the ECC TinyPairing library in sensor network research [41] as a basis for our evaluations and then compares our results with some related works.

i. Computation cost

To evaluate the computational efficiency of the proposed protocol, all cryptographic operations, such as public and private keys generation, encryption/decryption processes, and the SHA-256 hashing algorithm, are implemented based on the ECC primitives. The execution times of these operations measured in milliseconds (ms), are detailed in table 2, and the notations used are explained. To minimized the computation overhead, the registration phase and the shared key computation take place before the network becomes active. Therefore, this analysis focuses on the computation cost for pairwise key establishment, revocation and renewal computation operations. Between the $\mathcal{BS}s$ and $\mathbb{N}$, a total of 9 hash function operations, 12 exclusive-OR operations, 9 key generations operations, and 2 random number generation operations. Therefore, the total computation cost for the designed protocol is 9 x 0.385ms + 12 x 2ms + 9 x 2ms + 2 x 2.78 = 51.03ms. Table 4 displays a comparison of the computational costs between our work and related protocols. The results indicate that while our protocol incurs lower computational costs than [41], it has slightly higher costs compared to [12, 29]. This increased overhead is primarily due to the inclusion of key revocation and renewal processes in our key management system, which are absent in the other protocols.

Table 2. Computation time for a cryptographic operation

Operations and their notations	Running time (ms)
Time for public key encryption (T_{pke})	3.85
Time for public key decryption (T_{pkd})	3.85
Time for hash function operation (T_h)	0.385
Time for EC point multiplication (T_{pm})	63.08
Time for point addition (T_{pa})	0.000010875
Time for exclusive-OR (T_{xor})	2
Time for key generation (T_{keyGen})	2
Time for random number generation (T_{rnd})	2
Time for modular exponentiation (T_{me})	2.78

ii. Storage cost

Considering the constrained nature of the sensor node's resources. Storage cost is one of the critical parameters for evaluating the protocol's performance in WBAN. For convenience, the parameters analyzed are defined as follows: l_{ID} : length of the identifier, l_{At} : length of authentication table, l_{pk} : length of public key, l_{prk} : length of the private key, l_{sv} : length secret value, l_{ts} : length of timestamp, $l_{\mathcal{H}}$: length of hash, and their lengths in bits presented in table 3. In the proposed protocol, each sensor node stores the required keying credentials, for instance, each node stores a unique identity, public and private keys, secret value and authentication table that corresponds to each of the nodes in the network. Therefore, a device storage overhead is 8 + 256 + 256 + 32 + 4 = 556bits. We compared the storage cost of our scheme with some related works in table 4. Based on the results displayed in table 4, the proposed protocol has less storage overhead than the related works in [12, 29], except for the protocol in [41] which has considerably lower storage overhead than ours. This is because, in addition to fixed memory space needed to store public and private keys, bits mask and secret value, an amount of memory proportionate to the number of nodes is required in the networks.

Table 3. Storage parameters notations and their sizes

Storage parameters	l_{ID}	l_{At}	l_{pk}	l_{prk}	l_{sv}	l_{ts}	$l_{\mathcal{H}}$
Length (bits)	8	4	256	256	32	24	20

Table 4. Computation and storage cost comparison with related works

Protocols	Computation cost (ms)	Storage cost (bits)
[12]	16.230	928
[29]	1.57	1,120
[41]	2088.1	384
Proposed protocol	51.03	556

6. Conclusions

Several key management protocols have been proposed recently for domain-specific applications, such as WBAN. However, a thorough examination of these schemes reveals that many suffer from various security weaknesses, and even the purported secure protocols leave room for improvement in security and performance efficiency. In this paper, we propose a key management protocol based on ECC cryptography that leverages a keychain hash function, lightweight operations, and an authentication table. With the keychain algorithm, each seed value in the chain is considered a potential key, where the next key is derived by hashing the previous key multiple times $1\,(0_i = 1_i = \mathrm{L})$. This approach eliminates the reuse of keys and prevents key compromised attacks such as replay attacks, known key security, impersonation, violations of perfect forward secrecy, and man-in-the-middle attacks. Furthermore, the authentication mechanism sidesteps message encryption when using public key algorithms, instead ensuring the authenticity of the received public key as specified in the authentication table. These optimizations enable the protocol to operate efficiently without adding extra overhead to computation and memory resources, while also enhancing scalability as the problem space expands, thereby conserving energy. We evaluate the security attributes of the proposed protocol using BAN logic formal validation. Analysis of the protocol in phases satisfactorily proves that the protocol achieves the target security goals of confidentiality, integrity and authentication requirements. In addition, informal security analysis of the protocol indicates that the scheme is resilient against key compromised attacks such as replay, known key security, perfect forward secrecy and man-in-the-middle attacks. Performance analysis on the computation and storage cost of the protocol was carried out, and the results show that while our protocol incurs lower computational costs than [41], it has slightly higher costs compared to [12, 29]. Also, the proposed protocol has less storage overhead than the related works in [12, 29], except for the protocol in [41] which has considerably lower storage overhead than ours. In the future, we plan to explore more efficiency and optimization techniques for the test bed implementation of the protocol in WBAN

References

[1] Hajar, M. S., Al-Kadri, M. O., & Kalutarage, H. K. (2021). A survey on wireless body area networks: Architecture, security challenges and research opportunities. *Computers & Security*, *104*, 102211.

[2] A. Joshi, A. Mohapatra, “Authentication protocols for wireless body area network with key management approach” *Journal of Discrete Mathematical Sciences and Cryptography*, *22*(2), pp. 219-240, 2019

[3] S. J. Hussain, M. Irfan, N. Z. Jhanjhi, K. Hussain, and M. Humayun, “Performance enhancement in wireless body area networks with secure communication” *Wireless Personal Communications*, *116*(1), pp. 1-22, 2020.

[4] F. Noor, T. Kordy, A. Alkhodre, O. Benrhouma, A Nadeem., A. Alzahrani, “Securing wireless body area network with efficient secure channel free and anonymous certificateless signcryption” *Wireless Communications and Mobile Computing*, *2021*.

[5] Kumari, S., & Om, H. (2016). Authentication protocol for wireless sensor networks applications like safety monitoring in coal mines. *Computer Networks*, *104*, 137-154.

[6] AbuAlghanam, O., Qatawneh, M., Almobaideen, W., & Saadeh, M. (2022). A new hierarchical architecture and protocol for key distribution in the context of IoT-based smart cities. *Journal of Information Security and Applications*, *67*, 103173.

[7] Shah, S., Munir, A., Waheed, A., Alabrah, A., Mukred, M., Amin, F., & Salam, A. (2023). Enhancing Security and Efficiency in Underwater Wireless Sensor Networks: A Lightweight Key Management Framework. *Symmetry*, *15*(8), 1484.

[8] Sowjanya, K., Dasgupta, M., & Ray, S. (2021). A lightweight key management scheme for key-escrow-free ECC-based CP-ABE for IoT healthcare systems. *Journal of Systems Architecture*, *117*, 102108

[9] Khan, A. F., & Anandharaj, G. (2021). Ahkm: an improved class of hash based key management mechanism with combined solution for single hop and multi hop nodes in iot. *Egyptian Informatics Journal*, *22*(2), 119-124.

[10] Soni, M., & Singh, D. K. (2022). LAKA: lightweight authentication and key agreement protocol for internet of things based wireless body area network. *Wireless Personal Communications*, *127*(2), 1067-1084.

[11] Guermazi, A., Belghith, A., Abid, M., & Gannouni, S. (2017). KMMR: An efficient and scalable key management protocol to secure multi-hop communications in large scale wireless sensor networks. *KSII Transactions on Internet and Information Systems (TIIS)*, *11*(2), 901-923.

[12] Xie, Q., Liu, D., Ding, Z., Tan, X., & Han, L. (2023). Provably secure and lightweight patient monitoring protocol for wireless body area network in IoHT. *Journal of Healthcare Engineering*, *2023*(1), 4845850.

[13] Alese, B. K., Philemon, E. D., & Falaki, S. O. (2012). Comparative analysis of public-key encryption schemes. *International Journal of Engineering and Technology*, *2*(9), 1552-1568.

[14] Hussain, S. Z., & Kumar, M. (2021). Secured key agreement schemes in wireless body area network-a review. *Indian Journal of Science and Technology*, *14*(24), 2005-2033.

[15] Kim, D., Kang, S., & An, S. (2016). Secure and Efficient Time Synchronization for Border Surveillance Wireless Sensor Networks. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, *99*(1), 385-401.

[16] Louw, J., Niezen, G., Ramotsoela, T. D., & Abu-Mahfouz, A. M. (2016). A key distribution scheme using elliptic curve cryptography in wireless sensor networks. In *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)* (pp. 1166-1170). IEEE.

[17] Arunkumar, S., & Puranik, V. (2018). Key Management Scheme for Wireless Sensor Networks using HECC. *International Journal of Advanced Studies of Scientific Research*, *3*(9).

[18] Gudivada, R. B., & Hansdah, R. C. (2018, May). Energy efficient secure communication in wireless sensor networks. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)* (pp. 311-319). IEEE.

[19] Abbasinezhad-Mood, D., Nikooghadam, M., Mazinani, S. M., Babamohammadi, A., & Ostad-Sharif, A. (2019). More efficient key establishment protocol for smart grid communications: design and experimental evaluation on ARM-based hardware. *Ad Hoc Networks*, *89*, 119-131.

[20] Khan, S., Alzahrani, A. I., Alfarraj, O., Alalwan, N., & Al-Bayatti, A. H. (2019). Resource efficient authentication and session key establishment procedure for low-resource IoT devices. *IEEE Access*, *7*, 170615-170628.

[21] Salem, F. M., Ibrahim, E., & Elghandour, O. (2020). A lightweight authenticated key establishment scheme for secure smart grid communications. *Int. J. Safety Security Eng.*, *10*(4), 549-558.

[22] Rehman, Z. U., Altaf, S., & Iqbal, S. (2020). An efficient lightweight key agreement and authentication scheme for WBAN. *IEEE Access*, *8*, 175385-175397.

[23] Zagrouba, R., AlAbdullatif, A., AlAjaji, K., Al-Serhani, N., & Alhaidari, F, (2020). Authenblue: A New Authentication Protocol for the Industrial Internet of Things.

[24] Pathak, G., Gutierrez, J., Ghobakhlou, A., & Rehman, S. U. (2022). LPWAN Key Exchange: A Centralised Lightweight Approach. *Sensors*, *22*(13), 5065.

[25] Sheu, R. K., Pardeshi, M. S., & Chen, L. C. (2022). Autonomous Mutual Authentication Protocol in the Edge Networks. *Sensors*, *22*(19), 7632.

[26] Hegde, M., and Andrew, J. (2023). A Lightweight Authentication Framework for Fault-tolerant Distributed WSN. *IEEE Access*.

[27] Erskine, S. K., Chi, H., & Elleithy, A. (2023). SDAA: Secure Data Aggregation and Authentication Using Multiple Sinks in Cluster-Based Underwater Vehicular Wireless Sensor Network. *Sensors*, *23*(11), 5270.

[28] Vellingiri, J., Vedhavathy, T. R., Senthil Pandi, S., & Bala Subramanian, C. (2024). Fuzzy logic and CPSO-optimized key management for secure communication in decentralized IoT networks: A lightweight solution. *Peer-to-Peer Networking and Applications*, 1-19.

[29] Karati, A., & Chang, L. C. (2024). AnonMAKE: Toward Secure and Anonymous Mutually Authenticated Key Exchange Protocol for Vehicular Communications. *IEEE Transactions on Intelligent Transportation Systems*.

[30] Tu, S., Badshah, A., Alasmary, H., & Waqas, M. (2023). EAKE-WC: Efficient and anonymous authenticated key exchange scheme for wearable computing. *IEEE Transactions on Mobile Computing*.

[31] Kumar, M., & Hussain, S. Z. (2023). An efficient and secure mutual authentication protocol in wireless body area network. *EAI Endorsed Transactions on Pervasive Health and Technology*,

[32] Liu, K., Xu, G., Cao, Q., Wang, C., Jia, J., Gao, Y., & Xu, G. (2023). A Rivest–Shamir–Adleman-Based Robust and Effective Three-Factor User Authentication Protocol for Healthcare Use in Wireless Body Area Networks. *Sensors*, *23*(21), 8992.

[33] Gowda, N. C., Manvi, S. S., Malakreddy, B., & Lorenz, P. (2023). BSKM-FC: Blockchain-based secured key management in a fog computing environment. *Future generation computer systems*, *142*, 276-291.

[34] M. Omar, I. Belalouache, S. Amrane, and B. Abbache, "Efficient and energy-aware key management framework for dynamic sensor networks". Computers & Electrical Engineering, *72*, pp. 990-1005, 2018.

[35] Y. Ding, H. Xu, M. Zhao, H. Liang, and Y. Wang, "Group authentication and key distribution for sensors in wireless body area network". International Journal of Distributed Sensor Networks, 17(9), 15501477211044338, 2021.

[36] Choi, Y., Lee, D., Kim, J., Jung, J., Nam, J., & Won, D. (2014). Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography. *Sensors*, *14*(6), 10081-10106.

[37] Tripathy, A., Pradhan, S. K., Tripathy, A. R., & Nayak, A. K. (2019). A New Hybrid Cryptography Technique in Wireless Sensor Network. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, *8*(10), 121-131.

[38] Kumar, V., Malik, N., Dhiman, G., & Lohani, T. K. (2021). Scalable and storage efficient dynamic key management scheme for wireless sensor network. *Wireless Communications and Mobile Computing*, *2021*, 1-11.

[39] C. A. Lara-Nino, A. Diaz-Perez, and M. Morales-Sandoval "Key-Establishment Protocols for Constrained Cyber-Physical Systems". In Security in Cyber-Physical Systems pp. 39-65. Springer, Cham. 2021

[40] Gandino, F., Celozzi, C., & Rebaudengo, M. (2017). A key management scheme for mobile wireless sensor networks. *Applied Sciences*, *7*(5), 490.

[41] Umar, M., Wu, Z., & Liao, X. (2020). Mutual authentication in body area networks using signal propagation characteristics. *IEEE Access*, *8*, 66411-66422.

[42] Zhao, G., Di, B., & He, H. (2022). A novel decentralized cross‐domain identity authentication protocol based on blockchain. *Transactions on Emerging Telecommunications Technologies*, *33*(1), e4377.

Authors' Profiles

Yusuf Taofeek received the B.Sc Ed. Degree from Usmanu Danfodiyo University, Sokoto, in 2000. He obtained a Postgraduate Diploma in computer science and M.Tech. degree in cyber security science from Federal University of Technology, Minna, Nigeria, in 2005 and 2016 respectively. He is currently pursuing the PhD degree in cyber security science with Federal University of Technology, Minna. He is a member of cyber security expert association of Nigeria. His research interests include Cryptography, Network security, IoT security, Blockchain security and sensor network security.

Waziri Onomza Victor is a professor of cyber security science at the Federal University of Technology Minna. He obtained his PhD and M.Tech degrees from the Federal University of Technology Minna respectively. His research interests include Cryptography, Steganography, Number theory, Cloud computing, and Network Security.

Olalere Morufu is an Associate Professor at the National Open University of Nigeria. He obtained his PhD in Security in Computing specializing in Access Control from the University Putra Malaysia in 2016, and his M.Sc. degree in Computer Science from the University of Ilorin Nigeria in 2014. His research interests include access control, biometric security technology, malware detection, and information security.

Noel Moses Dogonyaro obtained his B.Tech in Physics/Computer Science, M.Tech in Cyber Security Science, and PhD in Cyber Security Science from the Federal University of Technology, Minna, Nigeria. Noel is a Lecturer in the Department of Cyber Security Sciences, Federal University of Technology, Minna, Nigeria. His major areas of research include Cryptography, Network Security, Information Security and Blockchain Technology.

How to cite this paper: Yusuf Taofeek, Waziri Onomza Victor, Olalere Morufu, Noel Moses Dogonyaro, "Secure Communication in Wireless Body Area Networks with ECC-Based Key Management and Renewal Protocols", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.14, No.6, pp. 26-47, 2024. DOI:10.5815/ijwmt.2024.06.03