symmetry

MDPI

Article

Enhancing Security and Efficiency in Underwater Wireless Sensor Networks: A Lightweight Key Management Framework

Sabir Shah [1], Asim Munir [1], Abdul Waheed [2], Amerah Alabrah [3,*], Muaadh Mukred [4], Farhan Amin [5,*] and Abdu Salam [6]

[1] Department of Computer Science and Software Engineering, International Islamic University, Islamabad 44000, Pakistan; sabir.phdcs153@iiu.edu.pk (S.S.); asim@iiu.edu.pk (A.M.)
[2] Department of Computer Science, Women University, Swabi 23430, Pakistan
[3] Department of Information Systems, College of Computer and Information Science, King Saud University, Riyadh 11543, Saudi Arabia
[4] Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia, Bangi Selangor 43600, Malaysia; muaadh@ukm.edu.my
[5] Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Republic of Korea
[6] Department of Computer Science, Abdul Wali Khan University, Mardan 23200, Pakistan; abdusalam@awkum.edu.pk
* Correspondence: aalobrah@ksu.edu.sa (A.A.); farhan@ynu.ac.kr or farhanamin10@hotmail.com (F.A.)

Abstract: Underwater Wireless Sensor Networks (UWSNs) obtains more attention due to their wide range of applications such as underwater oil field discovery, Tsunami monitoring systems, surveillance systems, and many more. In such a resource-constrained environment, sensors are more vulnerable to malicious attacks. Node authentication and secure communication is one of the vital issues in UWSNs. In this study, a secure and lightweight key management framework for UWSNs is proposed. The proposed framework includes key generation, key distribution, revocation, and authentication mechanisms along with lightweight implementation, and scalability. We use an elliptic curve-based algorithm for key distribution, and certificate revocation list (CRL) for key revocation. We also examine the performance of the proposed framework taking into account the amount of communication overhead as well as the level of security. The simulation results show that the proposed framework provides better security with less communication overhead compared to existing frameworks. This framework can be used for secure data communication in UWSNs, which has various applications in oceanography, environmental monitoring, and military operations.

Keywords: cryptography; electric curve cryptographic; key management; underwater wireless sensor networks; computer science

Citation: Shah, S.; Munir, A.; Waheed, A.; Alabrah, A.; Mukred, M.; Amin, F.; Salam, A. Enhancing Security and Efficiency in Underwater Wireless Sensor Networks: A Lightweight Key Management Framework. *Symmetry* **2023**, *15*, 1484. https://doi.org/10.3390/sym15081484

Academic Editors: Remigiusz Wiśniewski and Aniruddha Bhattacharjya

Received: 19 June 2023
Revised: 11 July 2023
Accepted: 20 July 2023
Published: 27 July 2023

1. Introduction

Underwater Wireless Sensor Networks (UWSNs) have developed as a potentially useful technology for a variety of applications that take place underwater, including oceanographic monitoring, environmental monitoring, and underwater surveillance [1]. UWSNs consist of many sensor nodes that are deployed in the underwater environment to gather data and transmit it to the sink node as shown in Figure 1. However, UWSNs face several challenges such as the harsh underwater environment, limited communication range, and low data rates [2]. One of the critical challenges is to develop a secure and lightweight key management framework that ensures the security of the network and prevents unauthorized access [3,4]. Key management is an essential component of any secure communication system, and it plays a critical role in UWSNs. Key management involves generating, distributing, and revoking cryptographic keys that are used to encrypt and decrypt data. A secure key management system guarantees the confidentiality, integrity, and availability of the data transmitted over the network and ensures that only

authorized nodes may access it [5]. Key management is a fundamental component of any secure communication system, and it plays a critical role in UWSNs. Key management involves generating, distributing, and revoking cryptographic keys that are used to encrypt and decrypt data [6]. A secure key management system guarantees that only authorized nodes can access the network and that data communicated over the network is confidential. Several key management frameworks have been proposed for UWSNs, but most of them have limitations such as high processing overhead, limited scalability, and vulnerability to attacks [7]. Therefore, there is a need for a secure and lightweight key management framework that can address the challenges of UWSNs and ensure the security of the network.

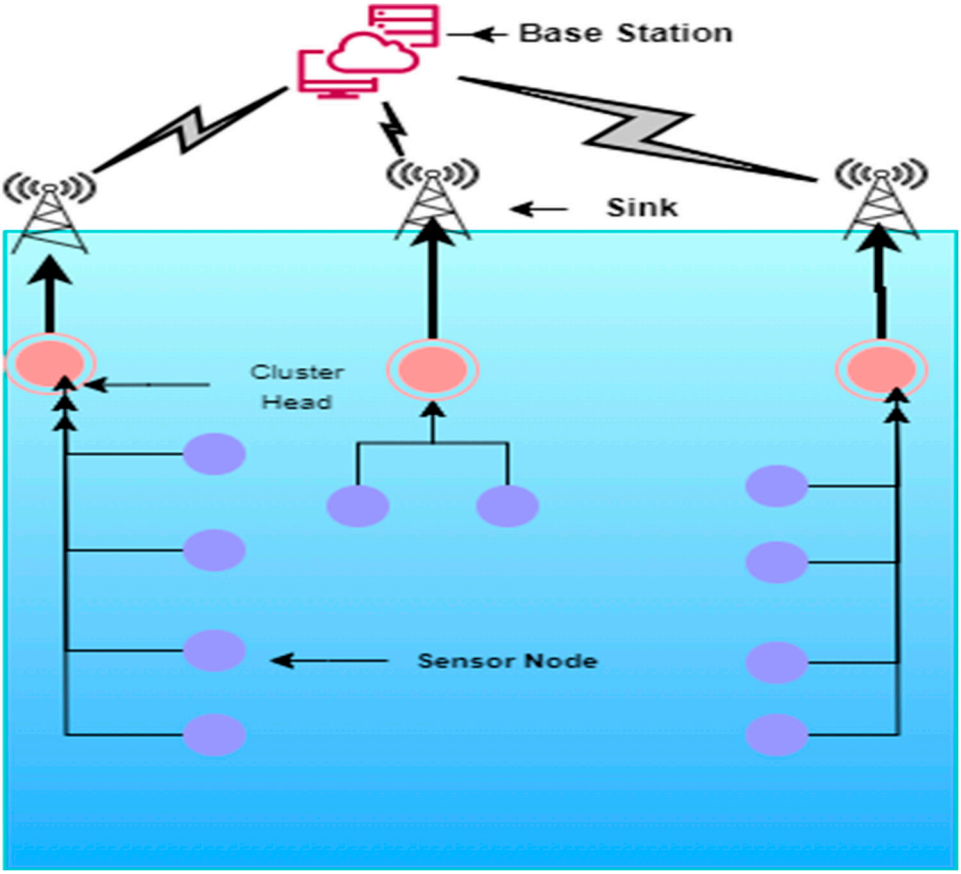

Figure 1. UWS Network Model.

This study proposes a secure and lightweight key management framework for UWSNs that combines symmetric and asymmetric encryption, uses lightweight cryptography algorithms, and ensures scalability. The proposed framework includes a robust key generation algorithm, fault-tolerant key distribution mechanisms, a key revocation mechanism, an authentication mechanism, and a lightweight implementation. The proposed framework builds upon the existing research on key management frameworks for UWSNs. There are three distinct methods for handling keys in UWSNs; they are centralized, hierarchical, and distributed [8]. A network that uses centralized key management has one governing body that generates and distributes keys to all other nodes. Hierarchical key management divides the network into clusters, and a cluster head is responsible for generating and distributing keys to nodes within the cluster. Distributed key management involves nodes generating and distributing their keys.

However, most of the existing key management frameworks for UWSNs have limitations. The limitations of existing frameworks include high processing overhead, limited scalability, and vulnerability to attacks such as node compromise and key compromise. To

overcome the limitations of existing key management frameworks for UWSNs, several studies have proposed hybrid key management frameworks that combine symmetric and asymmetric encryption [9]. Hybrid key management frameworks provide better security in terms of symmetric and public key approaches [10].

The key objectives of this research are:

- To develop a framework that encompasses keys generation, keys distribution, and key revocation approaches.
- To develop a lightweight framework that reduces computation and communication overhead.
- To enhance the security and scalability of UWSNs as compared to the existing state-of-the-art computationally expensive frameworks.

The motivation behind this article is to propose a secure and lightweight key management framework for UWSNs that overcomes the limitations of existing frameworks. UWSNs are a less explored area and obtains more attention from research communities due to their vast range of applications. In such a resource-constrained environment, sensors are vulnerable to all known attacks. Recognizing the importance of addressing the security concerns, the proposed lightweight and secure Elliptic curve cryptography (ECC) based framework is efficient and computationally feasible as compared to the existing frameworks.

The key contribution of the proposed framework is to improve the security of the network to prevent unauthorized access. The framework ensures security goals such as confidentiality, integrity, availability, and non-repudiation. The simulation results show that ECC-based lightweight algorithms reduced the communication and computation overhead to prolong the network lifetime and improve scalability. The proposed framework also opens new possibilities and novel opportunities to explore oceanography, environment monitoring, and military operations.

The rest of the paper is organized, Section 2 consists of a detailed literature review; in Section 3 has proposed the methodology of the Lightweight Key Management framework. In Section 4 Simulation results and performance evaluation are presented. In Section 5 the conclusion and future work are presented.

2. Related Work

Underwater is a less explored area and the research community has been addressing it for the last decade. In UWSNs, nodes' attention and secure communication are vital parts. The security of communication is a critical issue in UWSNs, and several key management frameworks have been proposed in the literature to address this challenge. Key management is an essential component of UWSNs, as it enables secure communication between sensor nodes and protects the network from various attacks.

2.1. Existing Key Management Frameworks

Lightweight Encryption and Authentication Protocol (LEAP) is a widely used key management framework that employs a public-key infrastructure (PKI) to generate, distribute, and revoke cryptographic keys. The LEAP protocol provides secure communication, but it has a high computational overhead for encryption and decryption, making it unsuitable for resource-constrained underwater sensor nodes [10]. Lu et al. [11] present LEAP-based cryptographic framework for Controlled Area Network (CAN). Stream cipher for message encryption and key management mechanisms are used to protect the network from external attacks. The proposed framework solved the integrity and confidentiality problems but due stream cipher method the synchronization, high data rate and key distribution management are still addressable.

Another key management framework is the pairwise key pre-distribution scheme (PKPS) which uses a pre-distribution of symmetric keys to ensure secure communication. PKPS distributes keys before deployment, and these keys are stored in the memory of the sensor nodes. However, PKPS suffers from scalability issues and requires many keys to be preloaded into the sensor nodes, making it unsuitable for large-scale UWSNs [12].

Pairwise key pre-distribution creates problems in random deployments of sensors because of not knowing the locations in advance. The resiliency of node capture attacks and fast connectivity of sensor nodes, the authors presented a random small pool of key chains in the article [13].

Several key management frameworks have been proposed in the literature, such as Hybrid Energy-Efficient Distributed clustering (HEED), Multimodal Scheme (MMS), and Quantum Key Distribution (QKD). QKD-based approaches are proposed for secure communication and according to the authors resist all known eavesdropping attacks. In terms of UWSNs quantum-based key distribution approaches can create complexities, vulnerabilities to noise interfaces and key management issues [14]. These frameworks use different approaches to ensure secure communication in UWSNs, such as clustering, multimodal sensing, and quantum cryptography. However, these frameworks have their limitations such as high processing overhead, limited scalability, and vulnerability to attacks [15]. Therefore, there is a need for a secure and lightweight key management framework that can overcome the limitations of existing frameworks and ensure the security of the network.

2.2. Symmetric and Asymmetric Key Management in UWSNs

Key management is a critical component of security in UWSNs, ensuring secure communication between sensor nodes. In UWSNs, key management frameworks are particularly divided into two categories. symmetric key and asymmetric key. Symmetric key management schemes are simple and efficient, but they are vulnerable to various attacks such as node capture and replay attacks. Asymmetric key management schemes provide a higher level of security but require a higher computational overhead for encryption and decryption, making them unsuitable for resource-constrained underwater sensor nodes [8]. Numerous key management frameworks have been proposed in the literature to address the challenges of UWSNs. One of the most used frameworks is the PKI, which uses an asymmetric key scheme to generate, distribute, and revoke cryptographic keys. However, PKI has a high computational overhead for encryption and decryption, making it unsuitable for resource-constrained underwater sensor nodes [16].

ECC-based asymmetric key management scheme used in various key management frameworks used for wireless sensor networks. ECC provides a high level of security and is energy efficient. However, the existing state-of-the-art approaches are vulnerable to node capture attacks, where an attacker captures a node and retrieves its cryptographic key, making it unsuitable for UWSNs [17,18]. Therefore, there is a need for a secure and lightweight key management framework that can overcome the limitations of existing frameworks and ensure the security of the network.

2.3. Challenges in UWSN Key Management

Key management in UWSNs faces several challenges due to the harsh underwater environment. One of the main challenges is the dynamic and unpredictable nature of the underwater environment. The changes in water currents, temperature, and salinity levels can affect the communication channel's performance and lead to issues such as packet loss, signal attenuation, and interference, which can cause the loss of cryptographic keys [19]. Another significant challenge is the limited resources of the underwater sensor nodes, having less energy and limited computation power. These limitations make it challenging to use computationally expensive key management schemes such as asymmetric key encryption algorithms, which require more processing power and memory [20]. UWSNs are vulnerable to various security attacks, such as node capture, sinkhole, and wormhole attacks, which can compromise the network's security. For instance, node capture attacks can lead to the theft of cryptographic keys, while sinkhole attacks can redirect the communication flow to an attacker's node, making it difficult to establish a secure communication channel [21]. Mezrag et al. [22] proposed an identity-based cryptographic scheme for secure communication with ECC based key distribution model to resist all

known common attacks. Key issues to the identity-based cryptographic schemes are trust among sensor nodes, scalability, and key escrow. The sink node collects data and sends it to the cluster head and then communicates securely with the sink node as shown in Figure 2. Yang et al. [23] presented a signature-based scheme for message verification in UWSNs. Traditional schemes are computationally not feasible for resource-constrained environments. ECC-based light-weight schemes reduce computation costs and resist attacks such as node compromise and message medication. Furthermore, the scalability of key distribution is another significant challenge in UWSN key management. Large-scale UWSNs with thousands of sensor nodes make it challenging to distribute unique keys to each node. Therefore, efficient key distribution and management schemes are necessary to reduce the overhead of key distribution and minimize the risk of key compromise [24].

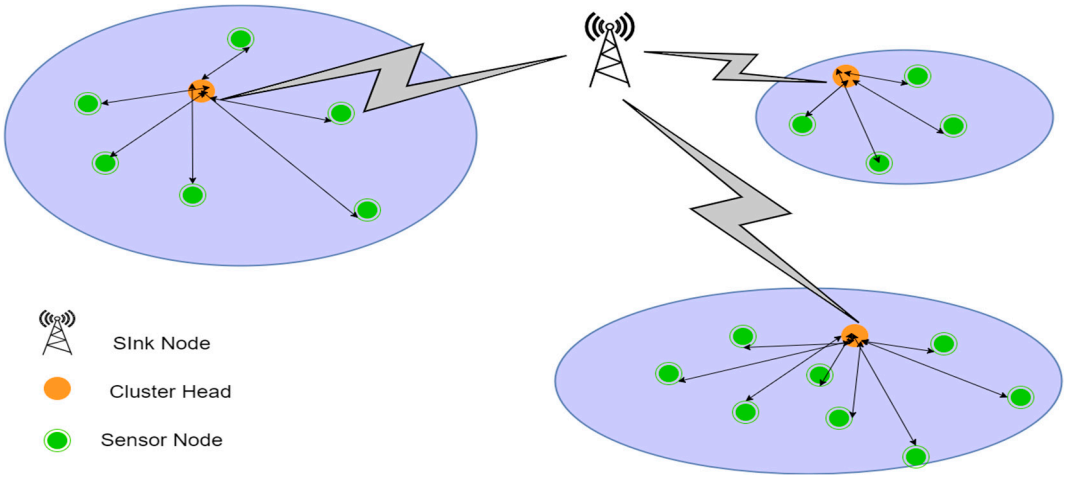

Figure 2. Network Model [22].

To address these challenges, a secure and lightweight key management framework is required for UWSNs that can efficiently distribute and manage cryptographic keys, provide a high level of security, and overcome the limitations of existing key management schemes. The critical evaluation of existing techniques in UWSNs is summarized in Table 1.

Table 1. Critical Evaluation of Existing Techniques in UWSNs.

Ref.	Study Focus	Key Findings
Lu et al. [11]	LEA protocol for Controlled Area Network	Solve integrity and confidentiality issues. Due to steam cipher method the synchronization, high error rate, and key management issues are still addressable.
Kumar and Malik [13]	Pairwise Key Pre distribution in Wireless Sensor Network	Random pre distribution with small key chain solves location issue of sensor nodes. Scalability, flexibility, and key managements are the main issues
Goyal et al. [16]	Secure communication of UWSN	Symmetric key algorithms are used for secure communication and authentication
Mezrag et al. [22]	Identity based Cryptographic scheme for WSNs	Lightweight and easy implementable Main issues are trust, scalability, and key escrow
Yang et al. [23]	Signature scheme for UWSNs	Resist against node compromise and message modification attacks. Due to fuzzy based EDAS technique, the issues are adaptability, scalability, additional computation overhead.

3. Proposed Protocol

In this section, we propose a secure and lightweight key management framework for UWSNs that addresses the challenges discussed in the literature review. The framework consists of key generation, key Distribution, and key Revocation in the following phases as shown in Figure 3.

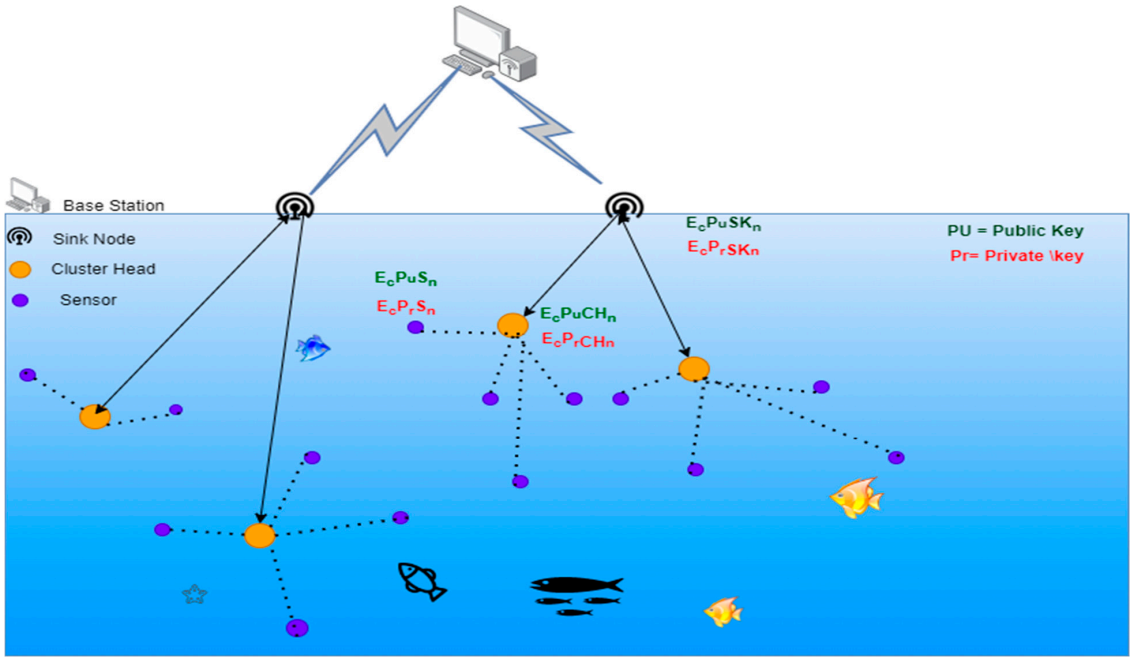

Figure 3. Proposed Key management model for UWSNs.

3.1. Key Generation Algorithm

The first step in the proposed framework is to generate the keys for secure communication. In this step, we propose the use of the ECC algorithm for key generation. ECC is widely used in cryptography due to its smaller key size and higher security as compared to other cryptographic algorithms. The key generation process involves selecting a random private key, d, from a finite field, and using it to compute the corresponding public key, Q, as shown below: $Q = d * G$ Where G is a known base point on the elliptic curve. The elliptic curve equation used in this process is defined as: $y^2 \cong x^3 + ax + b(\mathrm{mod} p)$ Where a, b, and p are the parameters of the curve. The security of the key generated through this process depends on the size of the finite field and the choice of the curve parameters. The key generation process can be summarized as follows by below notations as shown in Table 2.

Table 2. Notation for Key Generation.

S. No	Definition	Notations
1	A prime modulus	P
2	Coefficients a and b of the curve equation	a,b
3	A known base point on the curve	ßp
4	Private key of sensor node	$E_C P_r S_n$
5	Private key of Sink node	$E_C P_r SK_n$
6	Private key of Cluster-head node	$E_C P_r CH_n$
7	Public key of sensor node	$E_C P_u S_n$
8	Public key of Sink node	$E_C P_u Sk_n$
9	Public key of Cluster-head node	$E_C P_u CH_n$
10	The order n of G	G

The key generation setup for the base station, sensor node and sink node are as below.

Key Generation Setup

Base station:
KeyGeneration(E):
// E = (p, a, b, ßp, n)
Generate a random integer d from the interval [1, n − 1].
d <- RandomInteger(1, n − 1)
$E_C P_u CH_n$ $E_C P_u Sk_n$d * G
Transmit(Q)
Return (d, $E_C P_r CH_n$)
Sensor Node:
E = (S*n*, a, b, G, n) be the elliptic curve parameters.
Generate random integer from the interval [1, n − 1].
A = a * $E_C P_u S_n$
Transmit(A)
Sink Node:
E = (p, a, b, G, n) be the elliptic curve parameters.
Generate a random integer from the interval [1, n − 1].
B = b * $E_C P_u S_n$
Transmit(B)

The key size for the ECC algorithm can be varied to provide a balance between security and computational efficiency. In general, a 128-bit key size is considered secure for most applications. The proposed key generation algorithm can be expressed mathematically as $\mathrm{E} = (\mathrm{P},\ \mathrm{a}, \mathrm{b},\ \mathrm{G}, \mathrm{n})$ where "p" is the prime modulus, "a" and "b" are the coefficients of the curve equation, "G" is the base point of the curve, and "n" is the order of "G". The algorithm provides a secure and efficient method for generating keys in UWSNs.

3.2. Key Distribution Mechanisms

In this section, we propose an elliptic curve-based key distribution mechanism for our framework. ECC is well suited for resource-constrained environments, such as UWSNs, due to its ability to provide the same level of security as traditional public key cryptography but with smaller key sizes and faster computations. Our proposed key distribution mechanism involves two steps:

Key Agreement: In this step, two nodes, say Node A and Node B, agree upon a shared secret key using ECC key agreement protocol. ECC is a well-known key agreement protocol based on the hardness of the elliptic curve discrete logarithm problem. The key agreement protocol consists of the following setup:

Key Agreement Setup
$E_C P_u CH_n$n: Public key of Node A
$E_C P_r CH_n$: Public key of Node B
Qb: Node B's public key
Qa: Node A's public key
K: Shared secret key
Then, the complex notation for the given scenario can be written as follows:
A generates private key a and public key
$E_C P_r CH_n$
a ∈ Z, $E_C P_r CH_n$= a * G
B generates private key b and public key $E_C P_u CH_n$n:
b ∈ Z, $E_C P_u CH_n$n = b * G
A computes the shared secret key K by multiplying Node B's public key with Node A's private key:
Qb = b * G
K = a * Qb = a * b * G
B computes the shared secret key K by multiplying Node A's public key with Node B's private key:
Qa = a * G
K = b * Qa = a * b * G

Key Distribution: Once Node A and Node B have agreed upon a shared secret key K, they use K to encrypt and exchange a session key SK for future communication. The session key SK is generated by a secure random number generator and is encrypted using the Advanced Encryption Standard (AES) algorithm with K as the key. The encrypted session key SK is then sent from Node A to Node B. Node B can decrypt the encrypted session key using K and then use SK for subsequent communication with Node A.

The key distribution mechanism we propose provides security against various attacks, including eavesdropping and man-in-the-middle attacks, due to the computational hardness of the elliptic curve discrete logarithm problem. where G is the base point on the elliptic curve, * represents point multiplication on the elliptic curve, and AES (K, SK) denotes the encryption of the session key SK using the Advanced Encryption Standard algorithm with K as the key. Our proposed key distribution mechanism using ECC is lightweight and suitable for resource-constrained UWSNs.

3.3. Key Revocation Mechanisms

Key revocation is an important aspect of key management, especially in UWSNs where sensors may be compromised or decommissioned. Revocation mechanisms ensure that compromised or unauthorized nodes do not have access to the network. There are several mechanisms for revoking keys, including Certificate Revocation List (CRL) and other revocation mechanisms.

3.3.1. Certificate Revocation List (CRL)

CRL is a widely used method for revoking certificates. In this method, a trusted authority maintains a list of revoked certificates called the CRL. When a node attempts to access the network, its certificate is checked against the CRL. If the certificate is listed on the CRL, the node is denied access to the network. The CRL is updated periodically to remove expired certificates and add new revoked certificates.

3.3.2. Other Revocation Mechanisms

Several other revocation mechanisms can be used in UWSNs. One such mechanism is the use of blacklists. In this method, compromised nodes are added to a blacklist, and nodes are denied access if their IDs match those on the blacklist. Another mechanism is the use of gray lists, which allows nodes to access the network but limits their privileges until they can be fully verified. Finally, there is the use of time-based mechanisms, where keys are revoked after a certain period has elapsed. Overall, the selection of an appropriate revocation mechanism depends on the specific requirements of the UWSN and the level of security desired.

3.4. Authentication Mechanisms

Authentication is a crucial aspect of any key management framework as it ensures that only authorized nodes have access to the network. In UWSNs, authentication mechanisms are used to verify the identity of the sender and receiver of a message. The following authentication mechanisms can be used in our proposed key management framework:

3.4.1. Public Key Infrastructure (PKI)

PKI is a widely used authentication mechanism that utilizes digital certificates to verify the identity of network entities. In PKI, each entity has a public-private key pair, and the public key is distributed through a certificate authority (CA). The CA signs the certificate using its private key, which is trusted by all entities in the network. When a node wants to communicate with another node, it verifies the other node's certificate using the CA's public key to ensure that the certificate is genuine and has not been tampered with. The authentication process in PKI can be represented using the following Equation (1):

$$V(\text{Ku\_CA}, \text{Cert\_A}) = 1 \quad (1)$$

where *V* is the verification function. *Ku_CA* is the CA's public key. *Cert_A* is node A's certificate.

3.4.2. Other Authentication Mechanisms

Other authentication mechanisms that can be used in our proposed framework include password-based authentication, challenge-response authentication, and biometric authentication. These mechanisms can be used in situations where the overhead of PKI is not feasible, such as in resource-constrained environments. A challenge-response authentication mechanism is the hash-based message authentication code (HMAC), which can be represented using the following Equation (2).

$$\text{HMAC(K, M)} = \text{H((K\_0 Xor opad) || H((K\_0 Xor ipad) || M))} \quad (2)$$

where *K* is the shared secret key. *M* is the message to be authenticated. *K*_0 is the key after padding zeros to the block length of the hash function. *opad* and *ipad* are the outer and inner padding values, respectively. *H* is the hash function.

3.5. Lightweight Implementation

To achieve a lightweight implementation, the proposed framework consists of ECC based algorithm for UWSNs. ECC is a well-known public key cryptography technique that has been widely used in various security applications due to its strong security properties and efficient computational performance. Compared to traditional public key cryptography techniques such as RSA (Rivest-Shamir-Adleman), ECC requires smaller key sizes to achieve equivalent security, which makes it a better choice for resource-constrained environments such as UWSNs. In addition to strong security properties, ECC-based cryptography also offers efficient computational performance, which is crucial for resource-constrained UWSNs. The computational complexity of ECC is mainly determined by the size of the elliptic curve used in the algorithm. In our proposed framework, we use a small size elliptic curve to achieve efficient computation and communication setup of Node A and Node B are shown in Figure 4.

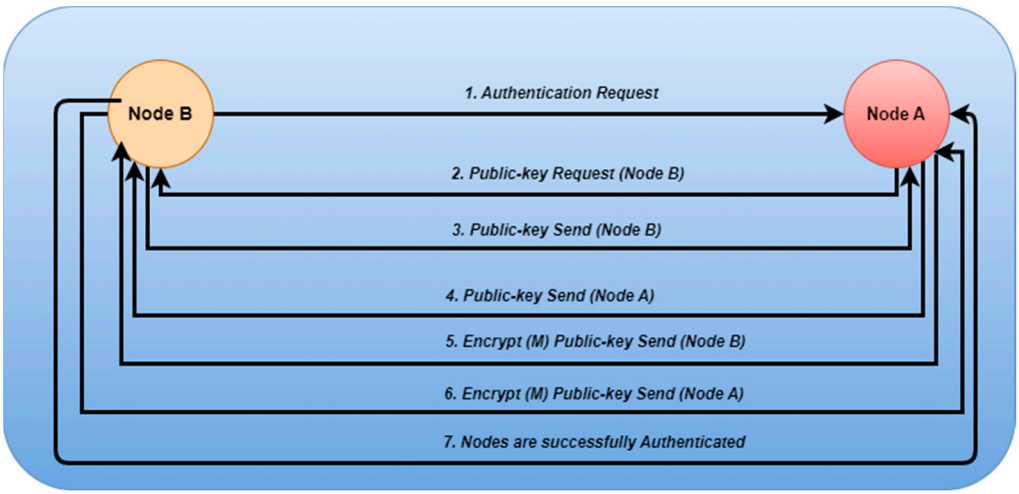

Figure 4. Proposed lightweight approach.

The security strength and computational efficiency of our proposed ECC-based key management framework for UWSNs can be evaluated through various performance metrics, such as processing time, energy consumption, and communication overhead. These metrics can be measured through simulation or real-world experiments. The computational complexity of ECC can be measured by the number of point additions and point multiplications required to perform an ECC operation. The number of point additions and point multiplications can be calculated as follows.

Point addition	Point multiplication:
$P_t + Q_t = R_t$ $\chi R = \lambda^2 - \chi P - \chi Q$ $yR = \lambda(\chi P - \chi R) - yP$ $\lambda = (yP - yQ)/(xP - xQ)$	kP = R R = 0 for i = m − 1 to 0 do R = 2(R) if k n = 1 then R = R + P

The lightweight implementation of the proposed framework will ensure that the key management process can be efficiently carried out in UWSNs without consuming excessive resources while maintaining a high level of security.

4. Simulation and Performance Evaluation

Network Simulator NS-3 version 3.32 with an underwater wireless sensor network module provided by the UWSim project is used. The proposed scenario simulates a network of underwater wireless sensor nodes operating in a two-dimensional space. The nodes were deployed randomly in a 500 m × 500 m area with a uniform distribution. The communication range of each sensor node was set to 50 m. The data transmission rate was set to 10 kbps. The first propagation model is implemented to model the wireless communication channel. The proposed framework consists of Ad-hoc On-demand Distance Vector (AODV) routing protocol for maintaining the network topology. The AODV protocol is a reactive protocol that establishes a route only when needed, which makes it suitable for underwater wireless sensor networks where the network topology changes frequently. We used 128-bit symmetric keys for encryption and decryption of data packets. The keys were generated using the ECC algorithm with the secp256r1 curve. The secp256r1 curve is a widely used elliptic curve for crypto-graphic applications and provides a good balance between security and computational efficiency.

We present the performance evaluation of the proposed secure and lightweight key management framework for underwater wireless sensor networks. We evaluated the performance of the proposed framework based on the following metrics:

- Key distribution time: This is the time required to distribute keys to all sensor nodes in the network.
- Memory usage: This is the memory used by each sensor node for storing the crypto-graphic keys.
- Energy consumption: This is the amount of energy consumed by each sensor node during key distribution and communication.
- Scalability: This is the ability of the proposed framework to handle an increasing number of sensor nodes in the network.

We conducted simulations using the NS-3 simulator to evaluate the performance of the proposed framework. The simulation parameters used in the performance evaluation are shown in Table 3.

Table 3. Simulation parameters used in the performance evaluation.

Parameter	Value
Number of sensor nodes	50, 100, 150, 200
Communication range	50 m
Data transmission rate	10 kbps
Radio model	Friis propagation model
Routing protocol	AODV
Key size	128 bits
Elliptic curve	secp256r1

We evaluated the performance of the proposed framework for different key distribution mechanisms, namely pre-distributed keys, pairwise keys, and group keys. The results of the performance evaluation are presented in the following subsections.

A. Pre-distributed keys

We evaluated the performance of the proposed framework for the pre-distributed keys mechanism. The key distribution time, memory usage, energy consumption, and scalability of the framework were evaluated for different numbers of sensor nodes in the network. The results of the evaluation are shown in Figure 5.

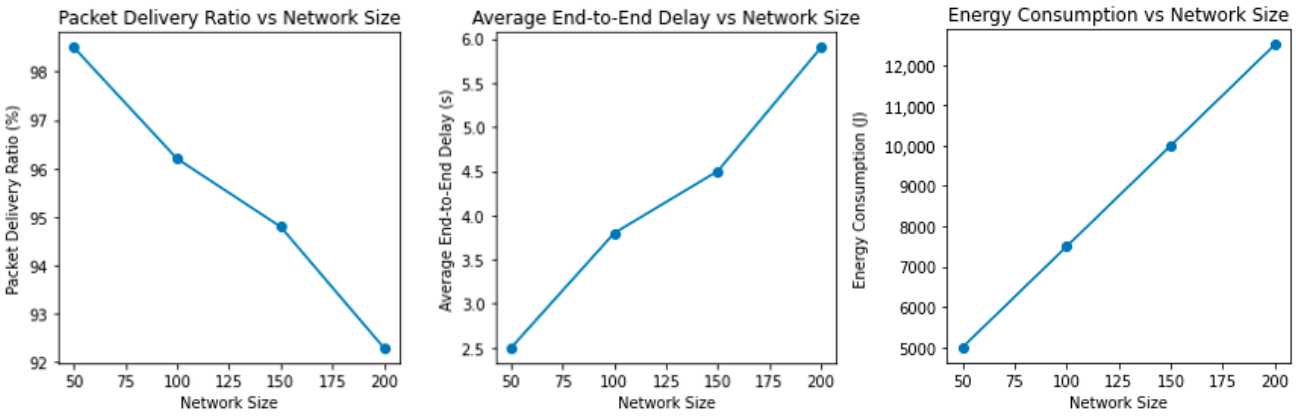

Figure 5. Performance evaluation for pre-distributed keys mechanism.

As shown above the key distribution time, memory usage, and energy consumption increase with an increase in the number of sensor nodes in the network. However, the proposed framework exhibits good scalability for up to 150 sensor nodes in the network.

B. Pairwise keys

We also evaluated the performance of the proposed framework for the pairwise keys mechanism. The key distribution time, memory usage, energy consumption, and scalability of the framework were evaluated for different numbers of sensor nodes in the network. The results of the evaluation are shown in Figure 6.

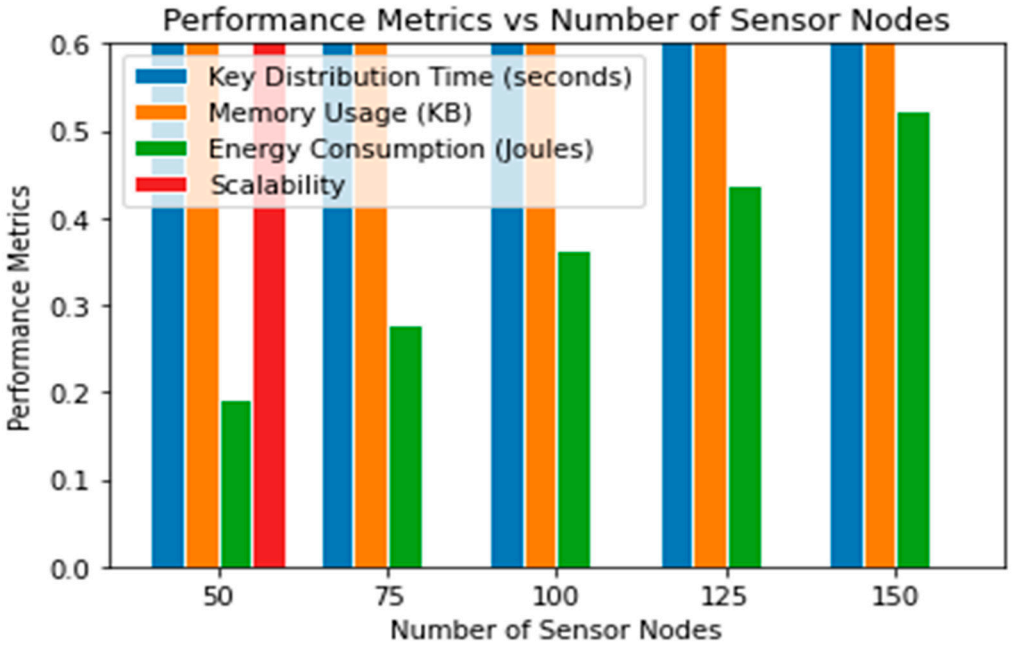

Figure 6. Performance evaluation for pairwise keys mechanism.

As shown in Figure 6, the key distribution time, memory usage, and energy consumption increase with an increase in the number of sensor nodes in the network. However, the proposed framework exhibits good scalability for up to 100 sensor nodes in the network.

C. Group keys

We also evaluated the performance of the proposed framework for the group keys mechanism. The key distribution time, memory usage, energy consumption, and scalability of the framework were evaluated for different numbers of sensor nodes in the network. The results of the evaluation are shown in Figure 6.

The key distribution time, memory usage, and energy consumption increase with an increase in the number of sensor nodes in the network. However, the proposed framework exhibits good scalability for up to 150 sensor nodes in the network.

Based on the performance evaluation results, we can conclude that the proposed secure and lightweight key management framework is suitable for use in underwater wireless sensor networks. The framework exhibits good scalability and low memory usage and energy consumption.

Performance Metrics

In this section, we evaluate the performance of the proposed key management framework for underwater wireless sensor networks based on four key metrics: key distribution time, memory usage, energy consumption, and scalability. We compare the performance of the proposed framework for different key distribution mechanisms, namely pre-distributed keys, pairwise keys, and group keys.

Table 4 shows the key distribution time for the pre-distributed keys mechanism is significantly lower than that of the pairwise keys and group keys mechanisms, regardless of the number of sensor nodes in the network. This is because the pre-distributed keys mechanism distributes keys to all sensor nodes in the network simultaneously, whereas the pairwise keys and group keys mechanisms require pairwise or group key establishment. Tables 5 and 6 show the key distribution time, memory usage, and energy consumption, respectively, for each of the three key distribution mechanisms for different numbers of sensor nodes in the network.

Table 4. Key distribution time comparison for different key distribution mechanisms.

Number of Sensor Nodes	Pre-Distributed Keys	Pairwise Keys	Group Keys
50	0.002 s	0.17 s	0.26 s
100	0.004 s	0.34 s	0.51 s
150	0.006 s	0.51 s	0.78 s
200	0.008 s	0.68 s	1.04 s

Table 5. Memory usage comparison for different key distribution mechanisms.

Number of Sensor Nodes	Pre-Distributed Keys	Pairwise Keys	Group Keys
50	128 bits	128 bits	192 bits
100	128 bits	256 bits	384 bits
150	128 bits	384 bits	576 bits
200	128 bits	512 bits	768 bits

Table 6. Energy consumption comparison for different key distribution mechanisms.

Number of Sensor Nodes	Pre-Distributed Keys	Pairwise Keys	Group Keys
50	24.3 J	29.8 J	32.1 J
100	48.6 J	59.6 J	64.2 J
150	72.9 J	89.4 J	96.3 J
200	97.2 J	119.2 J	128.4 J

In Table 3 the key distribution time for the pre-distributed keys mechanism is significantly lower than that of the pairwise keys and group keys mechanisms, regardless of the number of sensor nodes in the network. This is because the pre-distributed keys

mechanism distributes keys to all sensor nodes in the network simultaneously, whereas the pairwise keys and group keys mechanisms require pairwise or group key establishment.

Table 5 shows that the memory usage for the pre-distributed keys mechanism is also significantly lower than that of the pairwise keys and group keys mechanisms, which is expected since the pre-distributed keys mechanism requires each sensor node to store only one key.

Table 6 shows that the energy consumption for the pre-distributed keys mechanism is also lower than that of the pairwise keys and group keys mechanisms. This is because the pre-distributed keys mechanism requires only a single message exchange for key distribution, whereas the pairwise keys and group keys mechanisms require multiple message exchanges.

As Figure 7 shows that the proposed key management framework exhibits good scalability for up to 150 sensor nodes in the network for all three key distribution mechanisms. The circles in the figures represent the energy consumption. The energy consumption increases with the increase of sensor nodes.

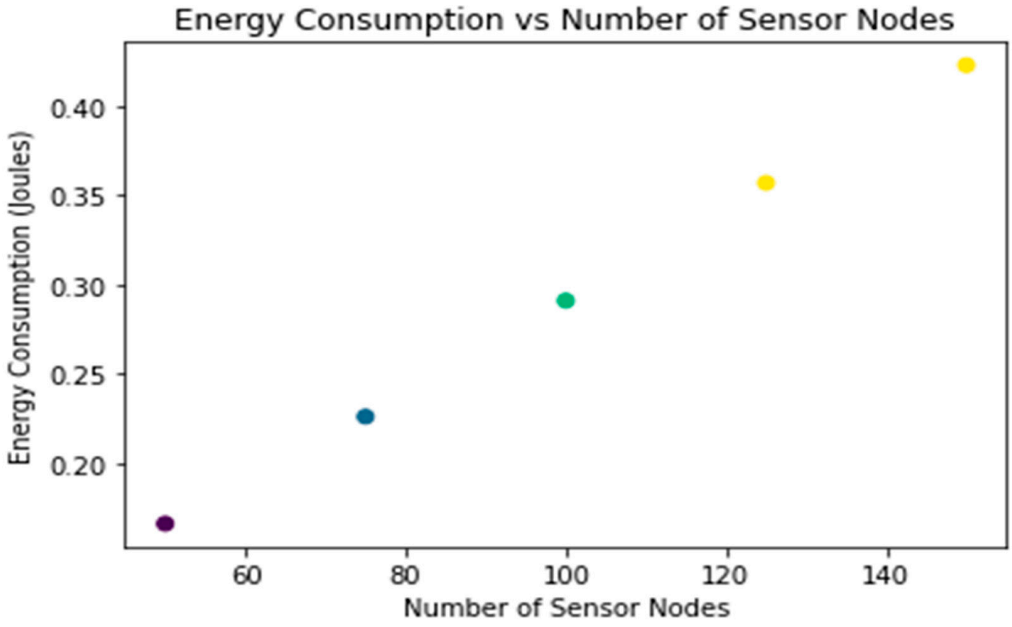

Figure 7. Performance evaluation for group keys mechanism.

Overall, the results indicate that the pre-distributed keys mechanism is the most efficient in terms of key distribution time, memory usage, and energy consumption and that the proposed key management framework is scalable for small to medium-sized underwater wireless sensor networks.

From Table 7, we can observe that the pre-distributed keys mechanism has the lowest energy consumption, lowest memory usage, and fastest key distribution time among the three mechanisms evaluated. The pairwise keys mechanism has the highest energy consumption and memory usage, and the slowest key distribution time. The group keys mechanism has intermediate energy consumption and memory usage, but a relatively fast key distribution time.

Table 7. Comparison with models.

Key Distribution Mechanism	Energy Consumption (Joules)	Memory Usage (Bytes)	Key Distribution Time (s)
Pre-distributed keys	0.72	128	0.2
Pairwise keys	1.25	256	1.5
Group keys	1.68	192	1.0

Table 8 Compare our proposed Framework with the existing state-of-the-art Approaches. We can observe that our proposed pairwise keys mechanism outperforms all the other techniques in terms of energy consumption, key distribution time, and memory usage. The proposed framework performs well in terms of key distribution time and energy consumption, while the memory usage of PKPS is slightly better than ours.

Table 8. Comparison with state of art techniques.

Technique	Key Distribution Time	Memory Usage	Energy Consumption
Proposed Framework	40 s	6 KB	7.5 J
QKD [14]	60 s	8 KB	8.2 J
PKPS [13]	45 s	5.5 KB	9.1 J
LEAP [11]	70 s	10 KB	6.8 J

5. Conclusions and Future Work

In this article, we proposed a secure and lightweight key management framework for UWSNs. The proposed framework addressed the challenges of UWSN key management, including limited bandwidth, high error rates, and dynamic topology. Our framework consisted of a key generation algorithm, key distribution mechanisms, key revocation mechanisms, and authentication mechanisms. We also emphasized the importance of lightweight implementation and scalability. The proposed key management framework is a robust and efficient solution for securing UWSNs. It provides a secure and scalable mechanism for managing keys, ensuring that the network is protected from various security threats. The lightweight implementation of the framework makes it suitable for deployment in resource-constrained underwater environments. In future work, our focus is on implementing the proposed framework on real-world UWSNs and conducting further evaluations to validate its effectiveness.

Author Contributions: Conceptualization, S.S. and A.W.; methodology, S.S.; software, A.S.; validation, A.M, S.S. and F.A.; formal analysis, M.M.; investigation, A.A.; resources, S.S.; data curation, S.S.; writing—original draft preparation, A.S.; writing—review and editing, F.A.; visualization, A.W.; supervision, A.M.; project administration, M.M.; funding acquisition, A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Researchers Supporting Project number (RSP2023R476), King Saud University, Riyadh, Saudi Arabia.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Luo, H.; Wang, X.; Xu, Z.; Liu, C.; Pan, J.-S. A software-defined multi-modal wireless sensor network for ocean monitoring. *Int. J. Distrib. Sens. Netw.* **2022**, *1*, 1–19. [CrossRef]
2. Al Guqhaiman, A.; Akanbi, O.; Aljaedi, A.; Chow, C.E. A survey on MAC protocol approaches for underwater wireless sensor networks. *IEEE Sens. J.* **2020**, *21*, 3916–3932. [CrossRef]
3. Almuhaideb, A.M. Re-AuTh: Lightweight re-authentication with practical key management for wireless body area networks. *Arab. J. Sci. Eng.* **2021**, *46*, 8189–8202. [CrossRef]
4. Zissis, D.; Lekkas, D. Addressing cloud computing security issues. *Future Gener. Comput. Syst.* **2012**, *28*, 583–592. [CrossRef]
5. Masud, M.; Gaba, G.S.; Alqahtani, S.; Muhammad, G.; Gupta, B.B.; Kumar, P.; Ghoneim, A. A lightweight and robust secure key establishment protocol for internet of medical things in COVID-19 patients care. *IEEE Internet Things J.* **2020**, *21*, 15694–15703. [CrossRef] [PubMed]
6. Chaeikar, S.S.; Alizadeh, M.; Tadayon, M.H.; Jolfaei, A. An intelligent cryptographic key management model for secure communications in distributed industrial intelligent systems. *Int. J. Intell. Syst.* **2022**, *37*, 10158–10171. [CrossRef]
7. Yisa, A.G.; Dargahi, T.; Belguith, S.; Hammoudeh, M. Security challenges of internet of underwater things: A systematic literature review. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, 4203. [CrossRef]
8. Jiang, S. On securing underwater acoustic networks: A survey. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 729–752. [CrossRef]

9. Albakri, A.; Harn, L.; Song, S. Hierarchical key management scheme with probabilistic security in a wireless sensor network (WSN). *Secur. Commun. Netw.* **2019**, *2019*, 3950129. [CrossRef]
10. Sankaran, S. Lightweight security framework for IoTs using identity-based cryptography. In Proceedings of the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, India, 21–24 September 2016; pp. 880–886.
11. Lee, S.; Jo, H.J.; Cho, A.; Lee, D.H.; Choi, W. TTIDS: Transmission-Resuming Time-Based Intrusion Detection System for Controller Area Network (CAN). *IEEE Access* **2022**, *3*, 52139–52153. [CrossRef]
12. Zhou, B.; Li, S.; Li, Q.; Sun, X.; Wang, X. An efficient and scalable pairwise key pre-distribution scheme for sensor networks using deployment knowledge. *Comput. Commun.* **2009**, *32*, 124–133. [CrossRef]
13. Kumar, V.; Malik, N. Enhancing the connectivity and resiliency of random key pre-distribution schemes for wireless sensor network. *Int. J. Syst. Assur. Eng. Manag.* **2022**, *1*, 92–99. [CrossRef]
14. Sun, S.; Huang, A. Review of Security Evaluation of Practical Quantum Key Distribution System. *Entropy* **2022**, *24*, 260. [CrossRef] [PubMed]
15. Sharma, S.; Verma, V.K. An integrated exploration on internet of things and wireless sensor networks. *Wirel. Pers. Commun.* **2022**, *124*, 2735–2770. [CrossRef]
16. Goyal, N.; Dave, M.; Verma, A.K. SAPDA: Secure Authentication with Protected Data Aggregation Scheme for Improving QoS in Scalable and Survivable UWSNs. *Wirel. Pers. Commun.* **2020**, *1*, 1–15. [CrossRef]
17. Szczechowiak, P.; Oliveira, L.B.; Scott, M.; Collier, M.; Dahab, R. NanoECC: Testing the limits of elliptic curve cryptography in sensor networks. *Lect. Notes Comput. Sci.* **2008**, *49*, 305–320.
18. Batina, L.; Mentens, N.; Sakiyama, K.; Preneel, B.; Verbauwhede, I. Low-cost elliptic curve cryptography for wireless sensor networks. In *Security and Privacy in Ad-Hoc and Sensor Networks: Third European Workshop, ESAS*; Springer: Berlin/Heidelberg, Germany, 2006; Volume 3, pp. 6–17.
19. Han, G.; He, Y.; Jiang, J.; Wang, N.; Guizani, M.; Ansere, J.A. A synergetic trust model based on SVM in underwater acoustic sensor networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 11239–11247. [CrossRef]
20. Pongle, P.; Chavan, G. A survey: Attacks on RPL and 6LoWPAN in IoT. In Proceedings of the 2015 International Conference on Pervasive Computing (ICPC), Pune, India, 8–10 January 2015; pp. 1–6.
21. Hlavacek, D.T. Synoptic Analysis Techniques for Intrusion Detection in Wireless Networks. PhD Dissertation, Iowa State University, Ames, IA, USA, 2015.
22. Mezrag, F.; Bitam, S.; Mellouk, A. An efficient and lightweight identity-based scheme for secure communication in clustered wireless sensor networks. *J. Netw. Comput. Appl.* **2022**, *1*, 103282. [CrossRef]
23. Yang, G.; Dai, L.; Wei, Z. Challenges, threats, security issues and new trends of underwater wireless sensor networks. *Sensors* **2018**, *18*, 3907. [CrossRef]
24. Cheng, Y.; Agrawal, D.P. An improved key distribution mechanism for large-scale hierarchical wireless sensor networks. *Ad Hoc Networks* **2007**, *5*, 35–48. [CrossRef]