sensors

MDPI

Article

A Computationally Efficient Online/Offline Signature Scheme for Underwater Wireless Sensor Networks

Syed Sajid Ullah [1,2], Saddam Hussain [3,*], Mueen Uddin [4], Roobaea Alroobaea [5], Jawaid Iqbal [6], Abdullah M. Baqasah [7], Maha Abdelhaq [8] and Raed Alsaqour [9]

1 Department of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway; syed.s.ullah@uia.no
2 Department of Electrical and Computer Engineering, Villanova University, Villanova, PA 19085, USA
3 School of Digital Science, Universiti Brunei Darussalam, Jalan Tungku Link, Gadong BE1410, Brunei
4 College of Computing and IT, University of Doha for Science and Technology, Doha 24449, Qatar; mueenmalik9516@gmail.com
5 Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; r.robai@tu.edu.sa
6 Department of Computer Science, Capital University of Science and Technology, Islamabad 44000, Pakistan; jawaid5825@gmail.com
7 Department of Information Technology, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; a.baqasah@tu.edu.sa
8 Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; msabdelhaq@pnu.edu.sa
9 Department of Information Technology, College of Computing and Informatics, Saudi Electronic University, Riyadh 93499, Saudi Arabia; r.alsaqor@seu.edu.sa
* Correspondence: saddamicup1993@gmail.com

Citation: Ullah, S.S.; Hussain, S.; Uddin, M.; Alroobaea, R.; Iqbal, J.; Baqasah, A.M.; Abdelhaq, M.; Alsaqour, R. A Computationally Efficient Online/Offline Signature Scheme for Underwater Wireless Sensor Networks. *Sensors* **2022**, *22*, 5150. https://doi.org/10.3390/s22145150

Academic Editors: Iván García-Magariño and Shah Nazir

Received: 2 March 2022
Accepted: 29 June 2022
Published: 8 July 2022

Abstract: Underwater wireless sensor networks (UWSNs) have emerged as the most widely used wireless network infrastructure in many applications. Sensing nodes are frequently deployed in hostile aquatic environments in order to collect data on resources that are severely limited in terms of transmission time and bandwidth. Since underwater information is very sensitive and unique, the authentication of users is very important to access the data and information. UWSNs have unique communication and computation needs that are not met by the existing digital signature techniques. As a result, a lightweight signature scheme is required to meet the communication and computation requirements. In this research, we present a Certificateless Online/Offline Signature (COOS) mechanism for UWSNs. The proposed scheme is based on the concept of a hyperelliptic curves cryptosystem, which offers the same degree of security as RSA, bilinear pairing, and elliptic curve cryptosystems (ECC) but with a smaller key size. In addition, the proposed scheme was proven secure in the random oracle model under the hyperelliptic curve discrete logarithm problem. A security analysis was also carried out, as well as comparisons with appropriate current online/offline signature schemes. The comparison demonstrated that the proposed scheme is superior to the existing schemes in terms of both security and efficiency. Additionally, we also employed the fuzzy-based Evaluation-based Distance from Average Solutions (EDAS) technique to demonstrate the effectiveness of the proposed scheme.

Keywords: underwater wireless sensor networks; certificateless online/offline signature; authentication scheme

1. Introduction

Currently, there has been a growing interest in monitoring marine ecosystems for scientific research, military applications, and commercial exploitation [1]. The UWSN is the most effective method of monitoring the marine environment. In principle, the UWSN is a wireless communication network comprised of tens or hundreds of battery-powered

sensor nodes [2]. Unlike wireless connections between ground sensors, the underwater channel has a high latency and low bandwidth, which uses a lot of power. In addition, changing or recharging a battery in UWSNs is far more complex than in ground WSNs. That is why the current security algorithms struggle with power usage [3]. Due to the constrained resources, the sensor nodes suffer from an energy consumption problem [4]. Therefore, almost all of the existing research and technology on UWSNs is focused on power savings at the expense of security and capability.

Security is one of the key elements in the design of the UWSNs' protocol and mechanism. As a result of their low cost and proximity to the events they monitor, sensor nodes are prime targets for malicious attacks of many kinds. In addition, the public communication channel makes it possible for any device to participate in the flow of information. Therefore, an attacker might easily control the sensors and unsecured UWSN communication lines. The research available on UWSNs focuses on self-organization, communication, flexibility, low power consumption, and adaptability. Unfortunately, the current studies have a lot of limitations when it comes to how well UWSNs can resist security threats, because resources are very limited, and the security situation is usually server-based because of certain data and communication sites [5].

In the context of security, authentication is necessary. Global WSN authentication solutions, such as public-based RSA [6] and Blom's symmetric matrix multiplication algorithm [7], have been presented, but they do not work for UWSNs because of their increased computational and communicational complexity. As a result, UWSNs require the development of an authentication system based on signatures [8].

A digital signature is a common solution for ensuring data authenticity in UWSNs. However, traditional digital signature schemes are based on expensive scaler point multiplication of the ECC, hyperelliptic curve devisor multiplication, and bilinear pairing operations, limiting their transmission to resource-limited devices such as sensors and IoT devices. An alternate solution to the problem is to utilize an offline/online signature, where the signature process is divided into online and offline phases. The offline phase performs computationally intensive tasks, while the online phase produces the signature on the message in real time. When installed on UWSNs, the gateway can simplify the online signature to generate authentic messages. Reducing the communication bandwidth and computation time is the key to the actual use of an online/offline signature technique. However, ensuring both the security and effectiveness of an online/offline approach in the real world remains a challenge. This is the main focus of the current paper.

1.1. Motivation and Contributions

The computation time and communication overhead are inversely related to the hardness of the underlying security concerns that must be spent on signature formation. Traditional signature techniques such as RSA and bilinear pairing, both of which are based on sub-exponential issues, need a significant amount of computation time and communication overhead and are not suitable for devices that have limited resources. Elliptic curve cryptography (ECC) is utilized instead. Their fundamental issue is a fully exponential one, and it is possible to generate their signatures in a significantly shorter amount of time.

However, it is still challenging to find a cryptographic solution that is appropriate for UWSNs. There are hardly any articles that concentrate on the cryptographic security and privacy for UWSNs [9–14]. However, bilinear pairing with elliptic curves is used to apply authenticity in various environments [15]. Since HEC has a higher efficiency and a shorter key length than ECC, bilinear pairing, and RSA, it is often regarded as the most compact and effective form of cryptographic mechanisms. In the proposed work, we focused on proposing a new security solution for UWSNs devices by dividing our algorithm into online and offline phases to further reduce the computational time and communication bandwidth during the device operation. The contributions to this paper are as follows:

- Firstly, we propose a new certificateless online/offline signature scheme based on a hyperelliptic curve cryptosystem for underwater wireless sensor networks.

- Secondly, we present the generic syntax of the proposed certificateless online/offline signature scheme for underwater wireless sensor networks.
- Thirdly, we provide the mathematical construction for the proposed certificateless online/offline signature scheme for underwater wireless sensor networks. The construction is actually an extension of the syntax. The designed approach offers the security necessity of unforgeability against both type one and type two adversaries, an antireplay attack.
- Finally, we compared the computational and communicational overhead of our proposed method with earlier certificateless online and offline signature solutions. According to the findings, the proposed strategy uses significantly fewer computing and communication resources than earlier solutions.

1.2. Paper Organization

In the upcoming section (i.e., Section 2), we will review the existing literature. Section 3 presents our proposed network and the construction of an online/offline signature for UWSNs. Section 4, presents the deployment of the proposed scheme on UWSNs. Section 5 presents the formal security analysis and Section 6 added the performance analysis. Section 7 is a review of our contributions while Section 8 concludes the research.

2. Related Works

Related studies have been presented to secure the UWSNs in recent years [9–14]. Unfortunately, the present key management and cryptographic solutions have some common problems, including computational and communicational complexity and the expansion of ciphertext [4]. Therefore, in the proposed approach, we considered an online/offline signature with a lightweight hyperelliptic curve cryptosystem to reduce the computational and communicational complexities for UWSN communications. Table 1 summarizes the related works.

Evan, Goldreich, and Micali [16] proposed the online/offline signature concept in 1990. The authors divided the signing algorithm into two phases: online and offline. In the absence of a message, heavier computations are transferred to the offline phase, while lighter computations are performed online. During the production process or whenever the device's power is connected, offline action can be conducted on the background computation device. Shamir and Thuman [17] refined the Trapdoor hash function-based online/offline signature technique in 2001. This improves the online efficiency. However, the technique increases the signature costs and has a trapdoor leak issue. In 2007, Chen [18] created an online/offline signature system employing the dual trapdoor hash function. However, in normal situations, neither method works.

Recently, Liu et al. [19] proposed an identity-based online/offline signature using the elliptic curve discrete logarithm problem (ECDLP). Addobea et al. [20] proposed COOS for mobile health devices in 2020. This study aims to reduce the computational and communication resources required by mobile health devices. According to Xu and Zeng [21], the propose scheme of Addobea et al. [20] is unable to accomplish correctness, a key security property that should be provided by a signature scheme. In the same year, Khan et al. [22] provided a new COOS solution for IoHT employing hyperelliptic curve discrete logarithm problem hardness (HCDLP). According to Hussain et al. [23], the given approach of Khan et al. [22] is insecure when subject to adaptive chosen message attacks. It has been proven that an adversary can fake a valid signature on a message by substituting their own public key in place of the one that is supposed to be used. An attribute-based online/offline signature system for mobile crowdsourcing was presented in 2021 by Hong et al. [24]. Sadly, the authors did not present a mathematical or network model. The solution is theoretical.

Table 1. Summary of the literature.

Authors Name & Reference No.	Advantages	Limitations
Liu et al. [19]	• Propose an identity-based online/offline signature. • The authors utilized ECC to minimize the cost consumptions.	• Suffers from key escrow problem • The cost consumptions can be reduced further
Addobea et al. [20]	• Propose COOS for mobile health devices in 2020. • Aims to reduce the computational and communication resources required by mobile health devices.	• Suffers from high computational and communicational resource due to heavy bilinear pairing operations. • Unable to accomplish correctness [21]
Khan et al. [22]	• Propose a new COOS solution for IoHT. • Reduced the computational and communicational resources utilizing HCDLP.	• Insecure when subject to adaptive chosen message attacks [23]
Hong et al. [24]	• Present an online/offline signature system for mobile crowdsourcing.	• The authors did not present a mathematical or network model.

The above schemes are based on sophisticated cryptographic methods, i.e., bilinear pairing and ECC, and thus combined with the high cost of computation and communication. These approaches are therefore not compatible with UWSNs equipped with minimal computation and communication resources. To construct an effective cryptographic solution for UWSNs that requires minimal computational resources, there is a critical need for a more concrete and efficient online/offline signature scheme. Our design scheme is based on the HCC, which is a generalized form of an elliptic curve.

3. Construction of the Proposed Scheme

3.1. Security Threats

In certificateless public key cryptography, two types of adversaries are considered i.e., type-1 (T_1) and type-2 (T_2).

The certificateless signature scheme has a unique security concept in comparison to those used by traditional signature schemes. According to the definitions found in [25], a certificateless signature scheme ought to take into account two distinct kinds of adversaries: a Type-I (T_1) adversary and a Type-II (T_2) adversary. The adversary T_1 is meant to stand in for a typical threat posed by a third party against the certificateless signature scheme. This means that T_1 does not have access to the master key, but it is able to request public keys and replace existing public keys with values of its choosing. The adversarial T_2 is a representation of a malicious Key Generation Center (KGC) that is responsible for generating users' partial private keys. It is permissible for the adversary T_2 to have access to the master key, but they are not authorized to replace the target user's public key.

3.2. Hyperelliptic Curve Cryptosystem (HEC)

Koblitz [26] is the one who first introduced the hyperelliptic curve cryptosystem (HEC), which belongs to a class of algebraic curves. It is also possible to think of it as a more generalized version of the elliptic curves cryptosystem (ECC) [27]. The HEC points, as opposed to ECC points, cannot be obtained from a group in any way [28]. The additive Abelian group that can be generated from a devisor is the subject of computation by the HEC. In comparison to RSA, bilinear pairing, and ECC, the HEC's parameter size is significantly smaller while maintaining the same level of security. This makes the HEC appealing to resource-constrained devices.

The curve whose genus value is 1 is typically referred to as the ECC curve. Figure 1 [29] illustrates a HEC that has a genus that is higher than 1. In a similar manner, the group order of the finite field ($\mathbb{F}_{\mathbb{q}}$.) for the (genus = 1) needed operands that were 160 bits long, which necessitated the need for at least $\mathbb{g}$.$\mathbb{l}\mathbb{o}\mathbb{g}_{\mathbb{2}}(\mathbb{q})$ $\approx$ $2^{\mathbb{160}}$, where g is the genus of the curve over, $\mathbb{F}_{\mathbb{q}}$., which is the set of a finite field of order q. In a similar manner, the curve with a genus equal to two needed operands that were 80 bits long. In addition, the curves with a genus equal to three required operands were 54 bits in length [30].

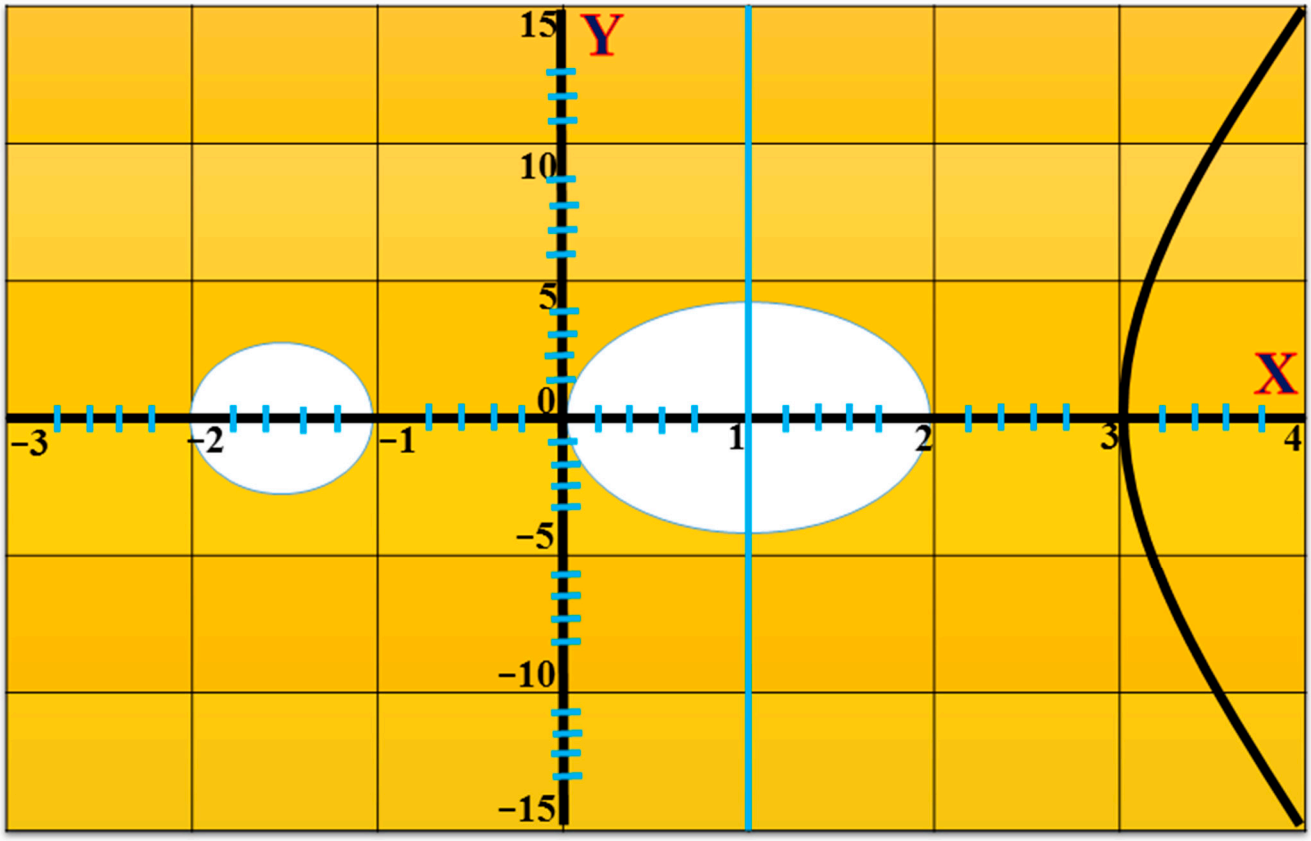

Figure 1. Hyperelliptic curve (genus = 2).

Let us assume that $\mathbb{F}$ is a finite field and that $\overline{\mathbb{F}}$ is the algebraic closure of $\mathbb{F}$. An HEC of a genus ($\mathbb{g} > 1$) over $\mathbb{F}$ is a set of solutions to the following equation of the curve in the form ($\mathbb{x}, \mathbb{y}$) ϵ $\overline{\mathbb{F}}$ x $\overline{\mathbb{F}}$.

$$\text{HEC}: \; y^2 + h(x)y = f(x)$$

If there are no pairs of ($\mathbb{x}$, $\mathbb{y}$) ϵ $\overline{\mathbb{F}}$ x $\overline{\mathbb{F}}$ that satisfy the condition, then the curve in question is regarded to be nonsingular. In addition, the curve in question must be able to satisfy both the previously mentioned curve equation, as well as the subsequent given partial differential equation.

$$2y + h(x) = 0 \text{ and } h'(x)y - f'(x) = 0$$

The polynomial h(x) ϵ $\mathbb{F}$[u] is a degree of $\mathbb{g}$, and h(x) ϵ $\mathbb{F}$[u] is the monic polynomial of degree $2\mathbb{g} + 1$.

3.3. Complexity Assumptions

During the course of the investigation, we found it necessary to presume the following assumptions:

$\mathbb{F}_{\mathbb{q}}$ is a finite field with order $\mathbb{q}$, where $\mathbb{q} \approx 2^{\mathbb{80}}$;
D is a divisor of a HEC, which is a finite sum of points;
D = $\sum p_i \epsilon$ HEC m_i p_i, where m_i ϵ $\mathbb{F}_{\mathbb{q}}$.

3.3.1. Definition 1. Hyperelliptic Curve Discrete Logarithm Problem (HCDLP)

We made the following supposition for HECDLP.

Let $\eta \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ and $\mathcal{W} = \eta.\mathcal{D}$; then, finding η from $\mathcal{W}$ is called HCDLP.

3.3.2. Definition 2. Hyperelliptic Curve Computational Diffie-Hellman Problem (HCCDH)

For HCCDHP, we make the following suppositions.

Let $\eta, Y \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ and $\mathcal{W} = \eta.\mathcal{D}$, $\mathcal{T} = Y.\eta.\mathcal{D}$; then, finding η from $\mathcal{W}$ and Y from $\mathcal{T}$ is called HCCDH.

3.4. Network Model

In Figure 2, we present the proposed network model for the online/offline sig-nature scheme for the underwater wireless sensors network. The proposed network model consists of a Network Manager (NM), an Intermediate Getaway, Underwater Sensors, and Surface Users.

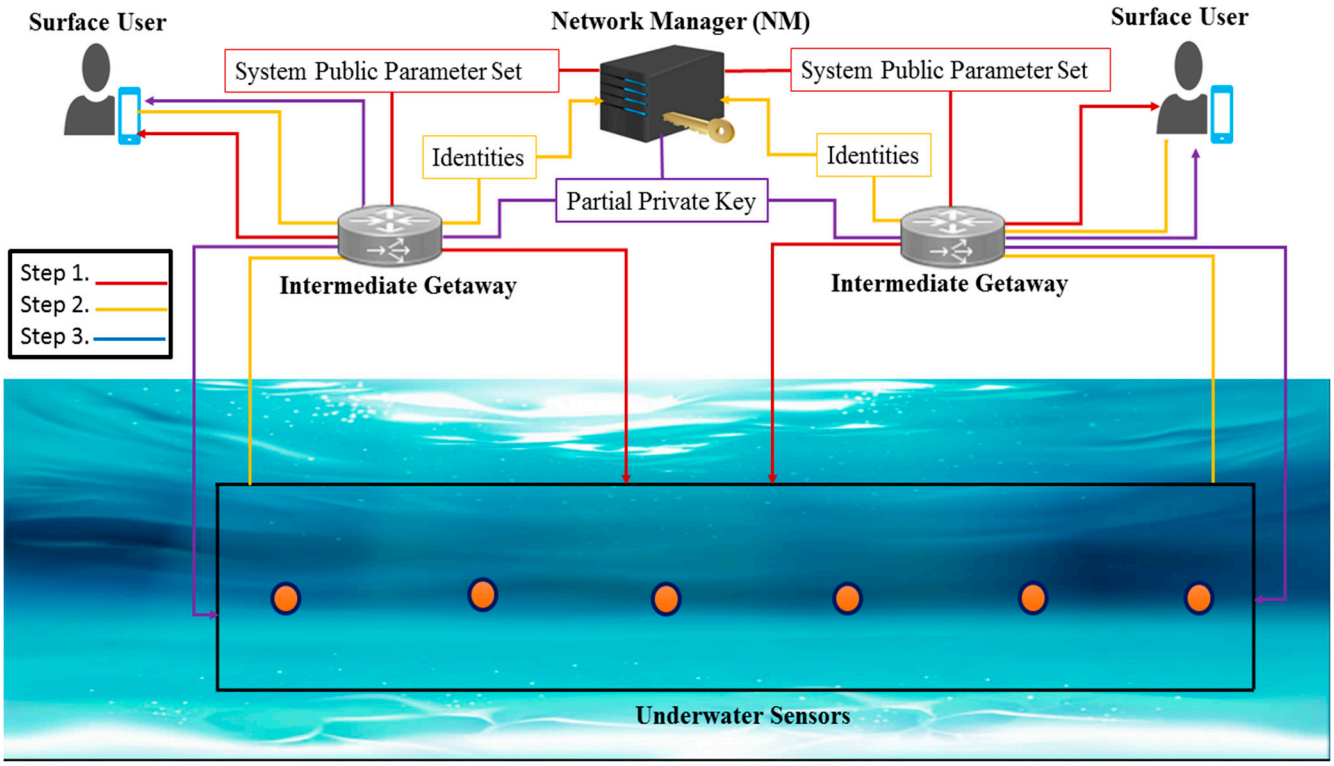

Figure 2. Proposed network model.

- Network Manager (NM): It is the responsibility of the NM to establish a secure connection between all of the entities within the networks, and it is a third party that can be trusted.
- Underwater Sensors: These are the sensors that sense the underwater environment and transmit data to the surface of the water.
- A surface user is a device or a client that is interested in underwater sensors, such as an Internet of Things device or a client.
- Intermediary Getaway: The intermediate getaway is a collection of nodes that act as a conduit for data and requests between different entities.

The NM is in charge of the registration process that takes place prior to the creation of communication links. The NM first registers the communication parties in order to facilitate secure communication. A great amount of processing power, memory, and computational capability are available on the intermediate gateway device. Sensors with limited

resources collect data and pass it to the intermediary gateway, which then processes it. In the presence of a message, the intermediate gateway then goes through the process of signature generation on the message.

3.5. Proposed Online/Offline Signature Algorithm for UWSNs

The symbols that were used in the construction of the proposed online/offline signature algorithm are listed in Table 2 of the following section. Additionally, Figure 3 presents the flowchart of the proposed algorithm.

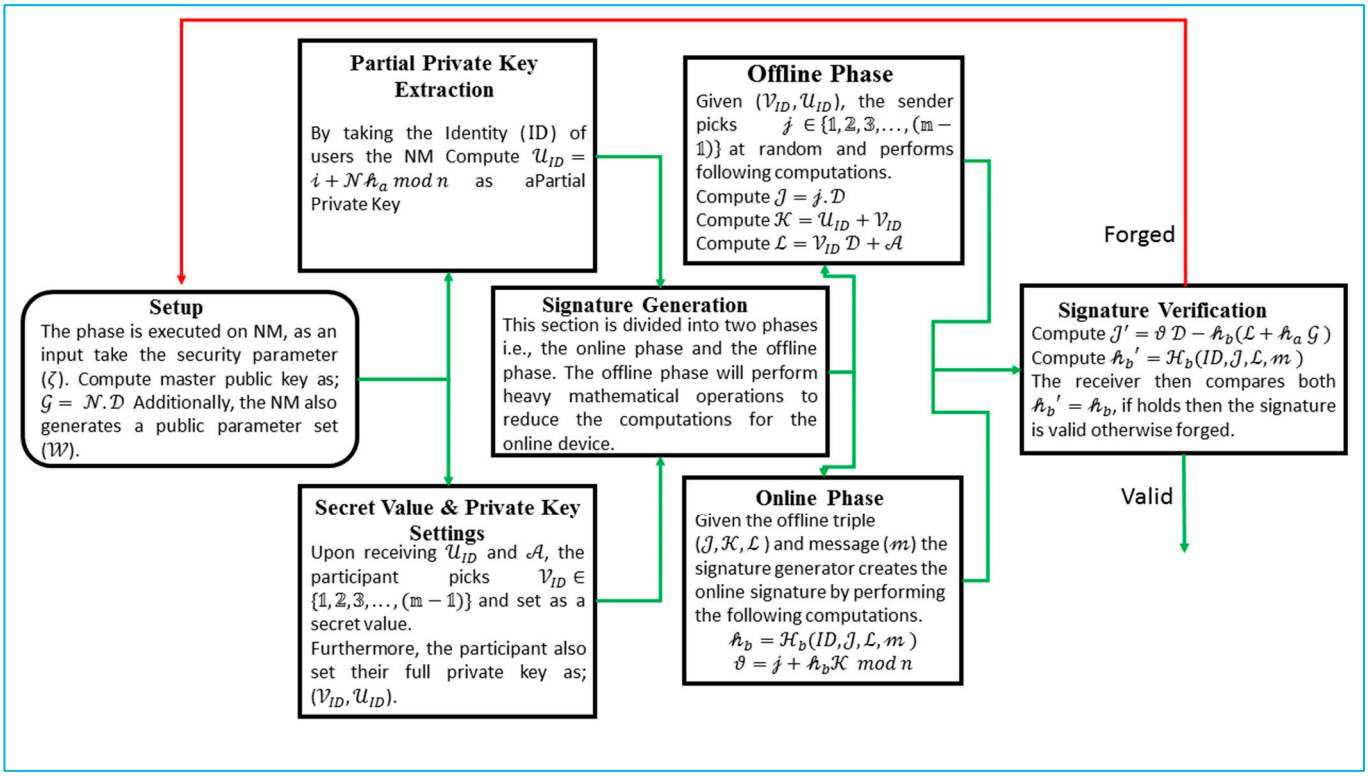

Figure 3. Flowchart of the proposed algorithm.

Table 2. Notation table.

S/N	Definition	Notations
1	Security Parameter	ζ
2	Public Parameter Set	$\mathcal{W}$
3	NM Master Key	$\mathcal{G}$
4	Identity of Users	ID
5	Partial Private Key	$\mathcal{U}_i$
6	Secret Value	$\mathcal{V}_i$
7	Full Private Key	$(\mathcal{V}_i, \mathcal{U}_i)$
8	Signature	$(\mathcal{L}, \mathcal{H}_b, \vartheta)$
9	assessment scores	μ
10	Average Value	ϑ

Table 2. *Cont.*

S/N	Definition	Notations
11	Positive Distance from Average	$\mathcal{PD}_{avg}$
12	Negative Distance from Average	$\mathcal{ND}_{avg}$
13	Weighted Sum of the Positive Distance	$\mathcal{WSPD}_{avg}$
14	Negative Distance	$\mathcal{ND}$
15	Weighted Sum of the Negative Distance	$\mathcal{WSND}_{avg}$
16	Positive Distance	$\mathcal{PD}$

Setup: The phase is carried out on NM, it take the security parameter (ζ) as an input. In addition, the NM will carry out the following procedures in order to produce a public parameter set designated as "($\mathcal{W}$)".

- Select the genus ($g = 2$) of HCC with the key size of 80 bits;
- Select $\mathcal{N} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ to compute the master public key as $\mathcal{G} = \mathcal{N}.\mathcal{D}$, where $\mathcal{D}$ is a devisor of the hyperelliptic curve cryptosystem (HCC);
- Choose two one-way hash functions $\mathcal{H}_a, \mathcal{H}_b$;
- Finally, the NM advertise $\mathcal{W} = \{\text{HCC}, \mathcal{H}_a, \mathcal{H}_b, \mathcal{G}, n, \mathcal{D}\}$ in the entire network while keeping the $\mathcal{N}$ with itself.

Partial Private Key Extraction: By taking the identity (ID) of users, the NM perform the following computations:

- First pick $\dot{\iota} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$;
- Compute $\mathcal{A} = \dot{\iota}.\mathcal{D}$;
- $\mathscr{f} = \mathcal{H}_a(ID, \mathcal{A})$;
- Compute $\mathcal{U}_{ID} = \dot{\iota} + \mathcal{N}\mathscr{f} \ mod \ n$.

The NM then send $\mathcal{U}_{ID}$ and $\boldsymbol{\mathcal{A}}$ to the participants. Upon receiving them, the participants can check the validity of the equation as

$$\mathcal{U}_{ID}.\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{A}} + \mathcal{G}\mathscr{h}_a$$

The partial private key is legitimate if the aforementioned equation is true; else, it is invalid.

Secret Value and Private Key Settings: Upon receiving $\mathcal{U}_{ID}$ and $\mathcal{A}$, the participants pick $\mathcal{V}_{ID} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ and set it as a secret value.

Furthermore, the participants also set their full private key as ($\mathcal{V}_{ID}$,$\mathcal{U}_{ID}$).

Signature Generation: This section is divided into two phases, i.e., the online phase and the offline phase. The offline phase will perform heavy mathematical operations to reduce the computation for the online phase.

Offline Phase: Given ($\mathcal{V}_{ID}$,$\mathcal{U}_{ID}$), the sender picks $\dot{\mathscr{l}} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and performs the following computations.

- Compute $\mathcal{J} = \dot{\mathscr{l}}.\mathcal{D}$;
- Compute $\mathcal{K} = \mathcal{U}_{ID} + \mathcal{V}_{ID}$;
- Compute $\mathcal{L} = \mathcal{V}_{ID}\ \mathcal{D} + \mathcal{A}$.

The triple ($\mathcal{J}, \mathcal{K}, \mathcal{L}$) is then assigned to the online phase.

Online Phase: Given the offline triple ($\mathcal{J}, \mathcal{K}, \mathcal{L}$), fresh nonce ($\tau$) and message ($m$), the signature generator creates an online signature by performing the following computations.

$$\mathscr{h}_b = \mathcal{H}_b(ID, \mathcal{J}, \mathcal{L}, \tau, \ m)$$

$$\vartheta = \lambda + \hbar_b \mathcal{K} \; mod \; n$$

Finally, the sender computes the triple of $(\mathcal{L}, \hbar_b, \vartheta)$ as a full signature.

Signature Verification: For an identity (ID) and message (m) with the computed signature triple $(\mathcal{L}, \hbar_b, \vartheta)$ on m, the receiver verifies the signature by performing the following operations:

- Compute $\ell = \mathcal{H}_a(ID, \mathcal{A})$;
- Compute $\mathcal{J}' = \vartheta \, \mathcal{D} - \hbar_b(\mathcal{L} + \hbar_a \, \mathcal{G} \,)$;
- Compute $\hbar_b{}' = \mathcal{H}_b(ID, \mathcal{J}', \mathcal{L}, \tau, m \,)$.

The receiver then compares both $\hbar_b{}' = \hbar_b$; if it holds, then the signature is valid; otherwise, it is forged.

The consistency can be proved from the following equation.

$$=> \; \mathcal{J}' = \vartheta \, \mathcal{D} - \hbar_b(\mathcal{L} + \hbar_a \, \mathcal{G})$$

4. Deployment of the Proposed Scheme

For deployment, we consider underwater sensors, and surface users want communication to share data. In this communication, there will be other entities like NM and the intermediate getaway. To make a connection and authentic sources of data, each entity will follow the following steps of the suggested online/offline signature. Figure 4 shows the deployment of the proposed scheme.

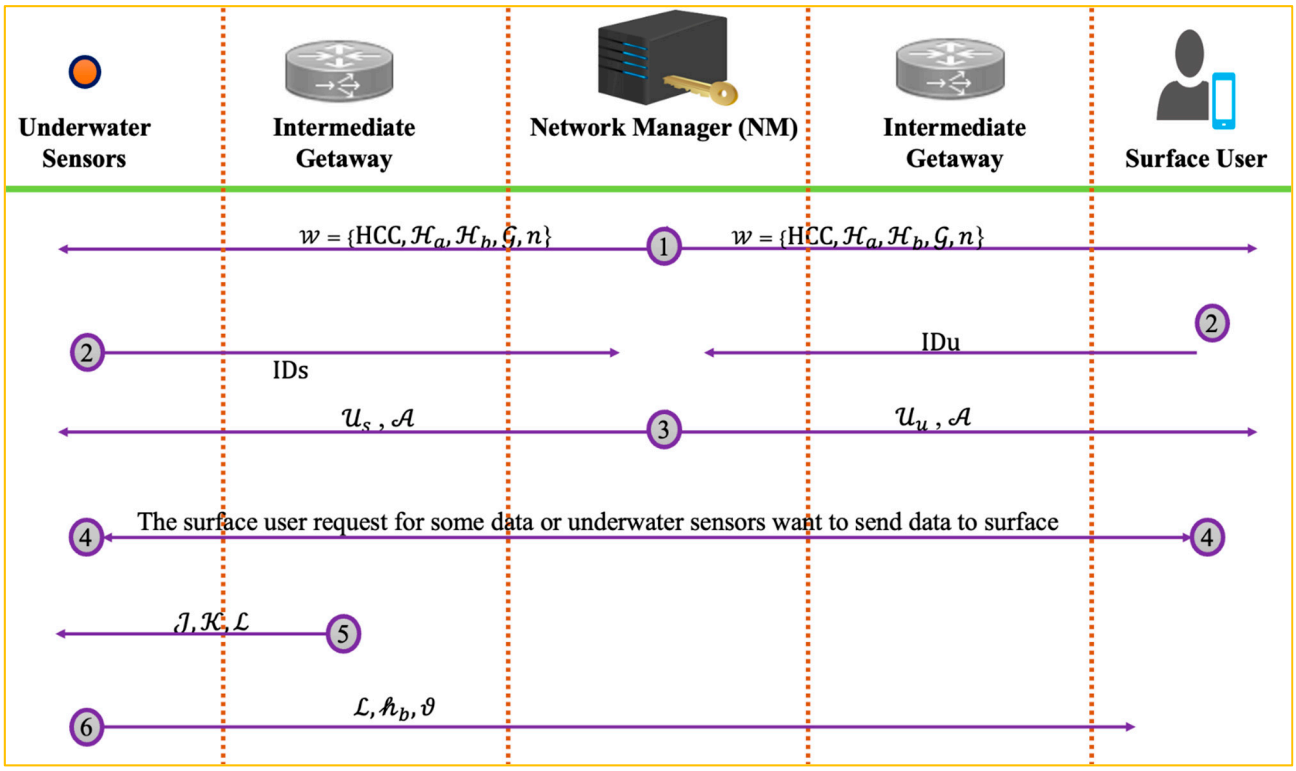

Figure 4. Deployment of the proposed scheme.

4.1. Setup, Connectivity, and Keys Extraction

To connect devices, the NM as an input takes the security parameter (ζ), and the KGC generates a public parameter set $(\mathcal{W})$. For this, the NM select a genus $(g = 2)$ of HCC with a key size of 80 bits, select $\mathcal{N} \; \epsilon \; \{1, 2, 3, 4, 5, \ldots, (n-1)\}$, compute the master public key as $\mathcal{G} = \mathcal{N}.\mathcal{D}$, where $\mathcal{D}$ is a devisor of the hyperelliptic curve cryptosystem (HCC), and choose two one-way hash functions $\mathcal{H}_0, \mathcal{H}_1$. Finally, the NM advertise $\mathcal{W} = \{HCC, \mathcal{H}_a, \mathcal{H}_b, \mathcal{G}, n\}$ in the entire network while keeping the $\mathcal{N}$ with itself.

To contact the network, the underwater sensors and surface user send their identities (IDs, IDu) to NM. By taking the IDs, IDu, the NM first pick $\dot{\iota} \in \{1, 2, 3, 4, 5, \ldots, (n-1)\}$, compute $\mathcal{A} = \dot{\iota}.\mathcal{D}$, $\hbar_a = \mathcal{H}_a(ID, \mathcal{A})$, and compute $\mathcal{U}_i = \dot{\iota} + \mathcal{N}\hbar_a \ mod\ n$. The NM then send $\mathcal{U}_i$ and $\mathcal{A}$ to the underwater sensors and surface user as a partial private key. Upon receiving it, the users can check the validity $\mathcal{U}_i$ of the equation as $\mathcal{U}_i.\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{A}} + \mathcal{G}\hbar_a$. If this equation holds, then the partial private key is valid; otherwise, it is invalid. Upon receiving $\mathcal{U}_i$ and $\mathcal{A}$, the participant picks $\mathcal{V}_i \in \{1, 2, 3, 4, 5, \ldots, (n-1)\}$ and set it as a secret value. Furthermore, the underwater sensors and surface user also set their full private key as $(\mathcal{V}_i, \mathcal{U}_i)$.

4.2. Signature Generation

In this step, the underwater sensors generate the signature on data. As we know, the underwater sensors have limited energy. This section is divided into two phases, i.e., the online phase and the offline phase of the signature. The offline phase will perform heavy mathematical operations to reduce the computations for the online device. The heir of the intermediate gateway performs the offline phase and underwater sensors online phase. The intermediate gateway picks $\mathcal{J} \in \{1, 2, 3, 4, 5, \ldots, (n-1)\}$ at random, computes $\mathcal{J} = \mathcal{J}.\mathcal{D}$, computes $\mathcal{K} = \mathcal{U}_i + \mathcal{V}_i$, and computes $\mathcal{L} = \mathcal{V}_i\ \mathcal{D} + \mathcal{A}$. The intermediate gateway then assigns the triple of $(\mathcal{J}, \mathcal{K}, \mathcal{L})$ to underwater sensors.

The underwater sensors take the triplet $(\mathcal{J}, \mathcal{K}, \mathcal{L})$ and data (m) and generate an online signature. For this, it calculates $\hbar_b = \mathcal{H}_b(ID, \mathcal{J}, \mathcal{L}, \boldsymbol{\tau}, m)$ and $\vartheta = \mathcal{J} + \hbar_b\mathcal{K}\ mod\ n$. Finally, the underwater sensors compute the triple of $(\mathcal{L}, \hbar_b, \vartheta)$ as a full signature and send it to the surface user.

4.3. Signature Verification

The surface user can verify the signature triple $(\mathcal{L}, \hbar_b, \vartheta)$ on m by computing $\hbar_a = \mathcal{H}_a(ID, \mathcal{A})$, computing $\mathcal{J}' = \vartheta\,\mathcal{D} - \hbar_b(\mathcal{L} + \hbar_a\,\mathcal{G})$, and computing $\hbar_b{}' = \mathcal{H}_b(ID, \mathcal{J}, \mathcal{L}, \boldsymbol{\tau}, m)$. The surface user then compares both $\hbar_b{}' = \hbar_b$; If it holds, the signature is considered legitimate; if not, it is considered to be forged.

5. Security Analysis

5.1. Theorem 1

Definition 3. *"Under the security assumptions of the random oracle model (ROM), an adversary (T_1) is unforgeable against the adaptive chosen message and identity attacks without knowledge of the partial private key and secret value."*

Proof. Assume $(\mathcal{D},\ \mathcal{o}\mathcal{D})$ as a random HCDLP stance that outputs $\boldsymbol{o}$. An algorithm $(\mathcal{A}\ell)$ will perform the subsequent simulations for interacting with T_1. □

Setup: In this phase, $\mathcal{A}\ell$ performs the following steps.

1. The $\mathcal{A}\ell$ sets the public key as $\mathcal{G} = \mathcal{o}.\mathcal{D}$ and advertises $\mathcal{W} = \{\text{HCC}, \mathcal{H}_a, \mathcal{H}_b, \mathcal{G}, n, \mathcal{D}\}$ in the entire network.
2. For $1 \leq \mathcal{p} \leq Q_{\mathcal{H}_a}$, the $\mathcal{A}\ell$ chooses ID_p at random as a challenging ID for this particular game, while $Q_{\mathcal{H}_a}$ represents the utmost number of the $\mathcal{H}_a$ querying oracle.
3. The $\mathcal{A}\ell$ picks $\mathcal{f}_p \in \{\mathbb{1}, 2, \mathbb{3}, \ldots, (\mathbb{n}-\mathbb{1})\}$ at random and sets $\mathcal{A}_p = -\mathcal{f}_p(\mathcal{o}.\mathcal{D})$, defines $\mathcal{C}_p = \mathcal{H}_a(ID, \mathcal{A})$, and adds the triple of $(\text{ID}_\text{p}, \mathcal{A}_p, \mathcal{f}_p)$ to the ${\mathcal{H}_a}^{list}$.
4. Finally, the $\mathcal{A}\ell$ gives T_1 the global parameters set as $\mathcal{W} = \{\text{HCC}, \mathcal{H}_a, \mathcal{H}_b, \mathcal{G}, n, \mathcal{D}\}$.
5. After that, the $\mathcal{A}\ell$ starts answering the queries from T_1 as

$\boldsymbol{\mathcal{H}_a\ Queries}$**:** The T_1 inputs $(\text{ID}_\text{i}, \mathcal{A}_i)$, and with that, the $\mathcal{A}\ell$ calls the ${\mathcal{H}_a}^{list}$. If the ${\mathcal{H}_a}^{list}$ has the $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$, $\mathcal{A}\ell$ provides it to the T_1. If not, the $\mathcal{A}\ell$ picks $\mathcal{f}_i \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n}-\mathbb{1})\}$ at random and adds $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$ to the ${\mathcal{H}_a}^{list}$ and response $\mathcal{C}_i$ to the T_1.

$\boldsymbol{\mathcal{H}_b\ Queries}$ **:** The T_1 inputs $(ID_i, \mathcal{J}_i, \mathcal{L}_i, m_i)$, and with that, the $\mathcal{A}\ell$ calls the ${\mathcal{H}_b}^{list}$. If the ${\mathcal{H}_b}^{list}$ already has the requested query, it simply returns back to the T_1. If not, the $\mathcal{A}\ell$

picks $\hbar_i \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and adds $(ID_i, \mathcal{J}_i, \mathcal{L}_i, \boldsymbol{\tau}, m_i, \hbar_i)$ to the $\mathcal{H}_b{}^{list}$ and response $\hbar_i$ to the T_1.

Partial Private Key Extraction Queries: Upon requesting the private key associated with $\mathrm{ID_i}$, the $\mathcal{A}\ell$ first verifies if $\mathrm{ID_i} = \mathrm{ID_p}$ stays or not. The $\mathcal{A}\ell$ also maintains the Ext^{list}.

1. If $\mathrm{ID_i} = \mathrm{ID_p}$, the $\mathcal{A}\ell$ terminates the simulation and outputs an error.
2. If $\mathrm{ID_i} \neq \mathrm{ID_p}$, the $\mathcal{A}\ell$ choose $\mathcal{V}_{IDi} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random as of the secret value allied with $\mathrm{ID_i}$. The $\mathcal{A}\ell$ picks $\mathcal{U}_{IDi} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ and computes $\mathcal{L}_\mathrm{i} = \mathcal{U}_{IDi}.\mathcal{D} + \mathcal{V}_{IDi}.\mathcal{D} - \mathscr{f}_i\ \mathscr{o}.\mathcal{D}$. If the $\mathcal{H}_a(\mathrm{ID_i}, \mathcal{A}_i, \mathscr{f}_i)$ already exists, then the $\mathcal{A}\ell$ terminates the simulation and outputs an error. The process is termed the Event by $\mathrm{EVE_1}$. The $\mathcal{A}\ell$ then adds $(\mathrm{ID_i}, \mathcal{A}_i, \mathscr{f}_i)$ and $(\mathrm{ID_i}, \mathcal{U}_{IDi}, \mathcal{V}_{IDi})$ to the Ext^{list}. To end with, the $\mathcal{A}\ell$ outputs $\mathcal{L}_i$ and $\mathcal{U}_{IDi}$.

The probability of $\mathrm{EVE_1}$ is the utmost $\frac{(Q_{\mathcal{H}_a} + Q_E)}{2^{\wedge}+1}$, where Q_E represent the querying of the key extraction oracle.

Secret Value Extraction Queries:

1. If $\mathrm{ID_i} = \mathrm{ID_p}$, the $\mathcal{A}\ell$ terminates the simulation and outputs an error.
2. If $\mathrm{ID_i} \neq \mathrm{ID_p}$, the $\mathcal{A}\ell$ searches $(\mathrm{ID_i}, \mathcal{U}_{IDi}, \mathcal{V}_{IDi})$ from the Ext^{list} and responds to the allied secret value $(\mathcal{V}_{ID})$.

Signature Generation Queries: Suppose a query for a signature with an identity (ID) and message (m).

1. If $\mathrm{ID_i} = \mathrm{ID_p}$, the $\mathcal{A}\ell$ picks ϑ_p,$\hbar_p \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and sets $\mathcal{L}_\mathrm{p} = \mathscr{o}.\mathcal{D} - \mathcal{C}_p(\mathscr{o}.\mathcal{D})$ and computes $\mathcal{J}_p = \vartheta_p.\mathcal{D} - \hbar_p\big(\mathcal{L}_\mathrm{p} + \mathcal{C}_p\mathcal{G}\big)$, where $\mathcal{H}_b(ID_p, \mathcal{J}_p, \mathcal{L}_p, \boldsymbol{\tau}, m_i)$. If $\mathcal{H}_b(ID_p, \mathcal{J}_p, \mathcal{L}_p, m_i)$ already exists, $\mathcal{A}\ell$ terminates the simulation and outputs an error. The process is the Event $\mathrm{EVE_2}$.
2. Finally, the $\mathcal{A}\ell$ outputs the triple $(\mathcal{L}_\mathrm{p}, \hbar_p, \vartheta_p)$ as the signature. The probability of $\mathrm{EVE_2}$ is utmost $\frac{(Q_{\mathcal{H}_a} + Q_{Sig})}{2^{\wedge}}$, where Q_{Sig} represents the querying of the signature generation oracle.
3. If $\mathrm{ID_i} \neq \mathrm{ID_p}$, the signature is normal, as the $\mathcal{A}\ell$ has the partial private key and secret value. Thus, the $\mathcal{A}\ell$ can ordinarily perform the online signature generation.

Forgery: Let the T_1 generate a forgeable digital signature $(\mathcal{L}^*, \hbar^*, \vartheta^*)$ on the message (m^*) for a given identity (ID^*), though ID^* is not submitted to the secret value extraction oracle and partial private key extraction oracle, and (m^*, ID^*) is not a query to the signature generation oracle.

1. If $\mathrm{ID}^* \neq {\mathrm{ID_p}}^*$ and $\mathcal{L}^* \neq {\mathcal{L}_p}^*$, then the $\mathcal{A}\ell$ terminates the simulation and outputs an error. The process is termed the Event $\mathrm{EVE_3}$. The probability of $\mathrm{EVE_2}$ is utmost $\frac{1}{Q_{\mathcal{H}_a}}$, where $Q_{\mathcal{H}_a}$ represent the utmost number of $\mathcal{H}_a$ querying the oracle.
2. If not, then according to the forking lemma [19], another algorithm $(\mathfrak{M})$ exists that is able to produce two valid digital signatures $(ID_p, \mathcal{J}_p, \mathcal{L}_p, m^*, \hbar_1, \vartheta_1)$ and $(ID_p, \mathcal{J}_p, \mathcal{L}_p, \boldsymbol{\tau}, m^*, \hbar_2, \vartheta_2)$ in a probabilistic polynomial time, where $\hbar_1 \neq \hbar_2$ while $\mathcal{C}_p$ remains the same due to $(ID_p, \mathcal{A}_p) = \mathscr{f}_p$. Thus, the subsequent equations hold as

$$\mathcal{J} = \vartheta_1.\mathcal{D} - \hbar_1\big(\mathcal{L}_\mathrm{p} + \mathscr{f}_p\mathcal{G}\big)$$

$$\mathcal{J} = \vartheta_2.\mathcal{D} - \hbar_2\big(\mathcal{L}_\mathrm{p} + \mathscr{f}_p\mathcal{G}\big)$$

After the calculations, we obtain $(\vartheta_1 - \vartheta_2)\mathcal{D} = (\hbar_1 - \hbar_2)\mathscr{o}.\mathcal{D}$, then get $\mathscr{o} = (\vartheta_1 - \vartheta_2)/(\hbar_1 - \hbar_2)$ and output $\mathscr{o}$ as a solution for the HCDLP instance, respectively.

5.2. Theorem 2

Definition 4. *There is an adversary (T_2) who is existentially unforgeable against the adaptive chosen message and identity attacks and has the knowledge of the partial private key/master secret key but does not have the participant's secret value in the ROM under the security HCDLP assumptions.*

Proof. Assume $(\mathcal{D},\ o\mathcal{D})$ as a random HCDLP stance that outputs o. An algorithm $(\mathcal{A}\ell)$ will perform the subsequent simulations for interacting with T_2. □

Setup: In this phase, $\mathcal{A}\ell$ performs the following steps.

1. The $\mathcal{A}\ell$ sets the public key as $\mathcal{G} = o.\mathcal{D}$ and advertises $\mathcal{W} = \{\text{HCC}, \mathcal{H}_a, \mathcal{H}_b, \mathcal{G}, n, \mathcal{D}\}$ in the entire network.
2. For $1 \leq p \leq Q_{\mathcal{H}_a}$, the $\mathcal{A}\ell$ chooses ID_p at random as a challenging ID for this particular game, while $Q_{\mathcal{H}_a}$ represents the utmost number of $\mathcal{H}_a$ querying oracles.
3. Finally, the $\mathcal{A}\ell$ gives T_2 the global parameters set $\mathcal{W} = \{\text{HCC}, \mathcal{H}_a, \mathcal{H}_b, \mathcal{G}, n, \mathcal{D}\}$ and master secret key $(\mathcal{N})$.

After that, the $\mathcal{A}\ell$ starts answering the queries from T_2 as:

$\mathcal{H}_a$ *Queries***:** The T_2 inputs $(\text{ID}_\text{i}, \mathcal{A}_i)$, and with that, the $\mathcal{A}\ell$ calls the ${\mathcal{H}_a}^{list}$. If the ${\mathcal{H}_a}^{list}$ has the $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$, $\mathcal{A}\ell$ provides it to the T_2. If not, the $\mathcal{A}\ell$ picks $\mathcal{f}_i \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and adds $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$ to the ${\mathcal{H}_a}^{list}$ and response $\mathcal{f}_i$ to the T_2.

$\mathcal{H}_b$ *Queries* : The T_2 inputs $(ID_i, \mathcal{J}_i, \mathcal{L}_i, \tau, m_i)$, and with that, the $\mathcal{A}\ell$ calls the ${\mathcal{H}_b}^{list}$. If the ${\mathcal{H}_b}^{list}$ already has the requested query, it simply returns back to the T_2. If not, the $\mathcal{A}\ell$ picks $\mathcal{h}_i \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and adds $(ID_i, \mathcal{J}_i, \mathcal{L}_i, \boldsymbol{\tau}, m_i, \mathcal{h}_i)$ to the ${\mathcal{H}_b}^{list}$ and response $\mathcal{h}_i$ to the T_2.

Partial Private Key Extraction Queries: Upon requesting the private key associated with ID_i, the $\mathcal{A}\ell$ first verifies if $\text{ID}_\text{i} = \text{ID}_\text{p}$ stays or not. The $\mathcal{A}\ell$ also maintains the Ext^{list}.

1. If $\text{ID}_\text{i} = \text{ID}_\text{p}$, the $\mathcal{A}\ell$ sets $\mathcal{A}_i = o.\mathcal{D}$ and obtains $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$ from ${\mathcal{H}_a}^{list}$. The $\mathcal{A}\ell$ then picks i_i at random and computes $\mathcal{U}_{IDi} = i_\text{i} + \mathcal{N}\mathcal{h}_i$ and adds $(\text{ID}_\text{i}, \mathcal{U}_{IDi}, \perp)$ to the list $(\text{ID}_\text{i}, \mathcal{U}_{IDi}, i_\text{i})$, where $\perp$ represents the unknown secret value for the identity ID_i. To end with, the $\mathcal{A}\ell$ returns $\mathcal{U}_{IDi}$.
2. If $\text{ID}_\text{i} \neq \text{ID}_\text{p}$, the $\mathcal{A}\ell$ finds $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$ from ${\mathcal{H}_a}^{list}$. The $\mathcal{A}\ell$ then chooses $i_{\text{i}1}, i_{\text{i}2} \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and computes $\mathcal{U}_{IDi} = i_{\text{i}2} + \mathcal{N}\mathcal{f}_i$ and adds $(\text{ID}_\text{i}, \mathcal{U}_{IDi},\ i_{\text{i}1})$ to the list. To end with, the $\mathcal{A}\ell$ returns $\mathcal{U}_{IDi}$

Signature Generation Queries: Suppose a T_2 query for a signature with an identity (ID) and message (m).

1. If $\text{ID}_\text{i} = \text{ID}_\text{p}$, the $\mathcal{A}\ell$ picks ϑ_i,$\mathcal{h}_i \in \{\mathbb{1}, \mathbb{2}, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and sets $\mathcal{A}_i = o.\mathcal{D}$ and finds $(\text{ID}_\text{i}, \mathcal{A}_i, \mathcal{f}_i)$ from ${\mathcal{H}_a}^{list}$, and additionally, the $\mathcal{A}\ell$ also sets $\mathcal{L}_i = \mathcal{A}_i = o.\mathcal{D}$ and computes $\mathcal{J}_i = \vartheta_i.\mathcal{D} - \mathcal{h}_i(\mathcal{L}_i + \mathcal{f}_i\mathcal{G})$, where $\mathcal{h}_i = \mathcal{H}_b(ID_i, \mathcal{J}_i, \mathcal{L}_i, \tau, m_i)$. If $\mathcal{H}_b(ID_i, \mathcal{J}_i, \mathcal{L}_i, m_i)$ already exists, $\mathcal{A}\ell$ terminates the simulation and outputs an error. The process is termed the Event EVE_2.
   - Computes $\mathcal{J}_p = \vartheta_p.\mathcal{D} - \mathcal{h}_p\big(\mathcal{L}_\text{p} + \mathcal{f}_p\mathcal{G}\big)$, where $\mathcal{H}_a(ID_p, \mathcal{J}_p, \mathcal{L}_p, \tau, m_i)$. If $\mathcal{H}_a(ID_p, \mathcal{J}_p, \mathcal{L}_p, \tau, m_i)$ already exists, $\mathcal{A}\ell$ terminates the simulation and outputs an error. The process is termed the Event EVE_2. Finally, the $\mathcal{A}\ell$ outputs the triple $(\mathcal{L}_i, \mathcal{h}_i,\ \vartheta_i)$ as the signature. The probability of EVE_2 is the utmost $\frac{(Q_{\mathcal{H}_b} + Q_{Sig})}{2^\wedge}$, where Q_{Sig} represents the querying of the signature generation oracle.
2. If $\text{ID}_\text{i} \neq \text{ID}_\text{p}$, the signature is normal, as the $\mathcal{A}\ell$ has the partial private key and secret value. Thus, the $\mathcal{A}\ell$ can ordinarily perform the online signature generation.

Forgery: Let the T_2 generate a forgeable digital signature $(\mathcal{L}^*, \mathcal{h}^*, \vartheta^*)$ on the message (m^*) for a given identity (ID^*), though ID^* is not submitted to the secret value extraction oracle, and (m^*,ID^*) is not query to the signature generation oracle.

1. If $\text{ID}^* \neq {\text{ID}_\text{p}}^*$ and $\mathcal{L}^* \neq {\mathcal{L}_p}^*$, then the $\mathcal{A}\ell$ terminates the simulation and outputs an error. The process is termed as the Event EVE_3. The probability of EVE_2 is not less than $\frac{1}{Q_{\mathcal{H}_a}}$, where $Q_{\mathcal{H}_a}$ represent the utmost number of $\mathcal{H}_a$ querying oracles.
2. If not, then according to the forking lemma [19], another algorithm $(\mathfrak{M})$ exists that is able to produce two valid digital signatures $(ID_p, \mathcal{J}, \mathcal{L}_p, m^*, \mathcal{h}_1,\ \vartheta_1)$ and

$(ID_p, \mathcal{J}, \mathcal{L}_p, m^*, \hbar_2, \vartheta_2)$ in a probabilistic polynomial time, where $\hbar_1 \neq \hbar_2$ and $\mathcal{A}' = \mathcal{L}'\mathcal{D}\,\ell_p$ remain the same. Thus, the subsequent equations hold as:

$$\mathcal{J} = \vartheta_1.\mathcal{D} - \hbar_1(\mathcal{L}_p + \ell_p\mathcal{G})$$

$$\mathcal{J} = \vartheta_2.\mathcal{D} - \hbar_2(\mathcal{L}_p + \ell_p\mathcal{G})$$

After the calculations, we obtain $(\vartheta_1 - \vartheta_2)\mathcal{D} = (\hbar_1 - \hbar_2)(o + \mathcal{N}\ell_p)\mathcal{D}$, then get $o = \frac{(\vartheta_1 - \vartheta_2)}{(\hbar_1 - \hbar_2)} - \mathcal{N}\ell_p$ and output o as a solution for the HCDLP instance, respectively.

5.3. Theorem 3

Definition 5. *If the NM impersonates an authentic participant in order to forge the signature and has knowledge of the participant's partial private key and secret value (an alternate secret value that is not real), we can demonstrate to the mediator that the NM is dishonest.*

Proof. According to the above two theorems, the proposed scheme is unforgeable against both malicious type-1 and type-2 adversaries. The process is split into two steps, i.e., forging the private key and signing the message. □

Forging the Private Key: Let ID be the identity of the participant, and ($\mathcal{V}_{ID}$,$\mathcal{U}_{ID}$) is the respective private key. The NM simulates the participant to generate a signature in two possible ways:

1. By knowing the participant's secret value $\mathcal{V}_{ID}$.
2. By replacing the participant's secret value $\mathcal{V}_{ID}$. As we know that the $\mathcal{V}_{ID}$ is picked at random from the $\{\mathbb{1}, 2, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$, it is infeasible of the NM to obtain the $\mathcal{V}_{ID}$.

Thus, the NM has to pick a secret value $\mathcal{V}_{ID}$ for the participants to produce another private key using the identity ID. The procedure is mentioned below.

1. The NM picks $\mathcal{V}_{ID}$ for the replacement of the participant's secret value.
2. The NM picks $i' \in \{\mathbb{1}, 2, \mathbb{3}, \ldots, (\mathbb{n} - \mathbb{1})\}$ at random and computes $\mathcal{A}' = i'.\mathcal{D}$ and $\mathcal{U}_{ID}{}' = i' + \mathcal{N}\ell_p{}' mod\ n$. Let $\mathcal{A}', \mathcal{U}_{ID}{}'$ satisfy and produce a private key ($\mathcal{V}_{ID}{}'$, $\mathcal{U}_{ID}{}'$).

Signing message: After forging the participant private key ($\mathcal{V}_{ID}{}'$, $\mathcal{U}_{ID}{}'$), the NM executes the signature generation algorithm. The triple $(i', \hbar', \vartheta')$ on the message m is for a given identity (ID) of the participant. The participant can run the signature generation algorithm twice to make sure that $(i', \hbar', \vartheta')$ is forged by the NM or an adversary conspired with the NM. Let the participant produce two signatures, $(\mathcal{L}, \hbar_1, \vartheta_1)$ and $(\mathcal{L}, \hbar_2, \vartheta_2)$, and submit the $(\mathcal{L}, \hbar_1, \vartheta_1)$ and $(\mathcal{L}, \hbar_2, \vartheta_2)$ to the intermediary trusted authority.

Note: Here, $\mathcal{L}' \neq \mathcal{L}$,. If the NM aims to make $\mathcal{L}' = \mathcal{L}$, then the NM needs to satisfy $(i' + \mathcal{V}_{ID}{}')\mathcal{D} = (i + \mathcal{V}_{ID})\mathcal{D}$. Furthermore, the NM also needs to know the value $\mathcal{A}' = (i + \mathcal{V}_{ID} - \mathcal{V}_{ID}{}')\mathcal{D} = i'\mathcal{D}$, but the NM does not know about $\mathcal{V}_{ID}$. Thus, according to the HCDLP, it is infeasible for the NM to obtain i, ℓ_p and $\mathcal{U}_{ID}$. Hence, $\mathcal{L}' \neq \mathcal{L}$.

Now, if the above three signatures are valid, then the $\mathcal{L}$ in the triple $(\mathcal{L}, \hbar_1, \vartheta_1)$ and $(\mathcal{L}, \hbar_2, \vartheta_2)$ are the same. We obtain $\mathcal{L}' \neq \mathcal{L}$ in $(\mathcal{L}', \hbar', \vartheta')$. Hence, $(\mathcal{L}', \hbar', \vartheta')$ definitely is forged by the NM or an adversary conspired with the NM.

6. Cost Efficiency

Here, we compared the proposed certificateless online/offline signature scheme with previously suggested online/offline signature schemes based on the communication bandwidth and computation time.

6.1. Computation Time

The proposed scheme is compared with some of the most recent online/offline signature schemes, i.e., Addobea et al. [19], Dan et al. [20], Khan et al. [22], and Hong et al. [24], in order to evaluate how well it performs in terms of the amount of computation that is required. A MIRACLE "C" Library [31] used to evaluate the effectiveness of the proposed

strategies in light of the costly mathematical operations. For testing the simulation results, a device with the features used is stated in Table 3 [27]. The key operation of our comparative analysis is explained in Tables 4–6, respectively. For our comparative analysis, we consider the costly mathematical operations pairing operations ($\mathcal{PO}$), bilinear pairing scalar multiplication ($\mathcal{PBSM}$), ECC-based scalar multiplication ($\mathcal{EBSM}$), and hyperelliptic curve devisor multiplication ($\mathcal{HCDM}$). Previous observations show that the running processing time of a single point multiplication varies significantly: $\mathcal{EBSM}$ takes 0.83 ms, $\mathcal{PO}$ consumes 20.01 ms, and $\mathcal{PBSM}$ consumes 6.38 ms [32]. Owing to the 80-bit key size, $\mathcal{HCDM}$ is estimated to be half of ECC, so it will consume 0.415 ms [22].

Table 3. Hardware and software specifications.

System	Specification
Library	Multi-Precision Integer and Rational Arithmetic C Library
Hardware Processor	PIV 3 GHZ
RAM	512 MB
OS	Windows XP

Table 4. Computation of the costs of both online and offline signature generation.

Operations/Ref. No	Addobea et al. [20]	Liu et al. [19]	Khan et al. [22]	Hong et al. [24]	Proposed
Pairing Operations ($\mathcal{PO}$)					
Bilinear Pairing Scalar Multiplication ($\mathcal{PBSM}$)	3 $\mathcal{PBSM}$				
ECC Based Scalar Multiplication ($\mathcal{EBSM}$)		**2 $\mathcal{EBSM}$**		3 $\mathcal{PBSM}$	
Hyperelliptic Curve Devisor Multiplication ($\mathcal{HCDM}$)			**4 $\mathcal{HCDM}$**		2 $\mathcal{HCDM}$
Total cost of Signature Generation	19.14 ms	1.66 ms	1.66 ms	2.49 ms	0.83 ms

Table 5. Computation of the costs of both online and offline signature verification.

Operation/Ref. No	Addobea et al. [20]	Liu et al. [19]	Khan et al. [22]	Hong et al. [24]	Proposed
Pairing Operations ($\mathcal{PO}$)	3 $\mathcal{PO}$				
Bilinear Pairing Scalar Multiplication ($\mathcal{PBSM}$)	4 $\mathcal{PBSM}$				
ECC Based Scalar Multiplication ($\mathcal{EBSM}$)		2 $\mathcal{EBSM}$		4 $\mathcal{EBSM}$	
Hyperelliptic Curve Devisor Multiplication ($\mathcal{HCDM}$)			3 $\mathcal{HCDM}$		2 $\mathcal{HCDM}$
Total Signature Verification Time	85.55 ms	1.66 ms	1.245 ms	3.32 ms	0.83 ms

Table 6. Total computation costs of both the online and offline phases.

Operation/Ref. No	Addobea et al. [20]	Liu et al. [19]	Khan et al. [22]	Hong et al. [24]	Proposed
Pairing Operations ($\mathcal{PO}$)	3 $\mathcal{PO}$				
Bilinear Pairing Scalar Multiplication ($\mathcal{PBSM}$)	7 $\mathcal{PBSM}$				
ECC Based Scalar Multiplication ($\mathcal{EBSM}$)		4 $\mathcal{EBSM}$		7 $\mathcal{EBSM}$	
Hyperelliptic Curve Devisor Multiplication ($\mathcal{HCDM}$)			7 $\mathcal{HCDM}$		4 $\mathcal{HCDM}$
Total Computation Time	3 $\mathcal{PO}$ + 7 $\mathcal{PBSM}$ = 104.69 ms	4 $\mathcal{EBSM}$ = 3.32 ms	7 $\mathcal{HCDM}$ = 2.905 ms	7 $\mathcal{EBSM}$ = 5.81 ms	4 $\mathcal{HCDM}$ = 1.66 ms

The sender of the message executes the certificateless online/offline signature generation algorithm of the proposed scheme, which involves two $\mathcal{HCDM}$ to produce the certificateless online/offline signature. Additionally, the certificateless online/offline signature verifier requires two $\mathcal{HCDM}$ to authenticate the online/offline signature. Table 4 shows the computation time required by the suggested online/offline cryptographic schemes in terms of costly operations. Moreover, Table 5 demonstrates the efficiency evaluation comparison between the proposed scheme and the previous design schemes in milliseconds. According to Table 6, the essential time-designed scheme is almost 98.41% of Addobea et al. [20], 50% of Liu et al. [19], 42.85% of Khan et al. [22], and 71.42% of Hong et al. [24]. Additionally, Figure 5 demonstrates the computational time evaluation analysis of certificateless online/offline signature generation and verification. The vertical axis indicates the computation time in milliseconds for a clear representation of the computation timeframe. It is obvious that the new strategy is more effective than the previous.

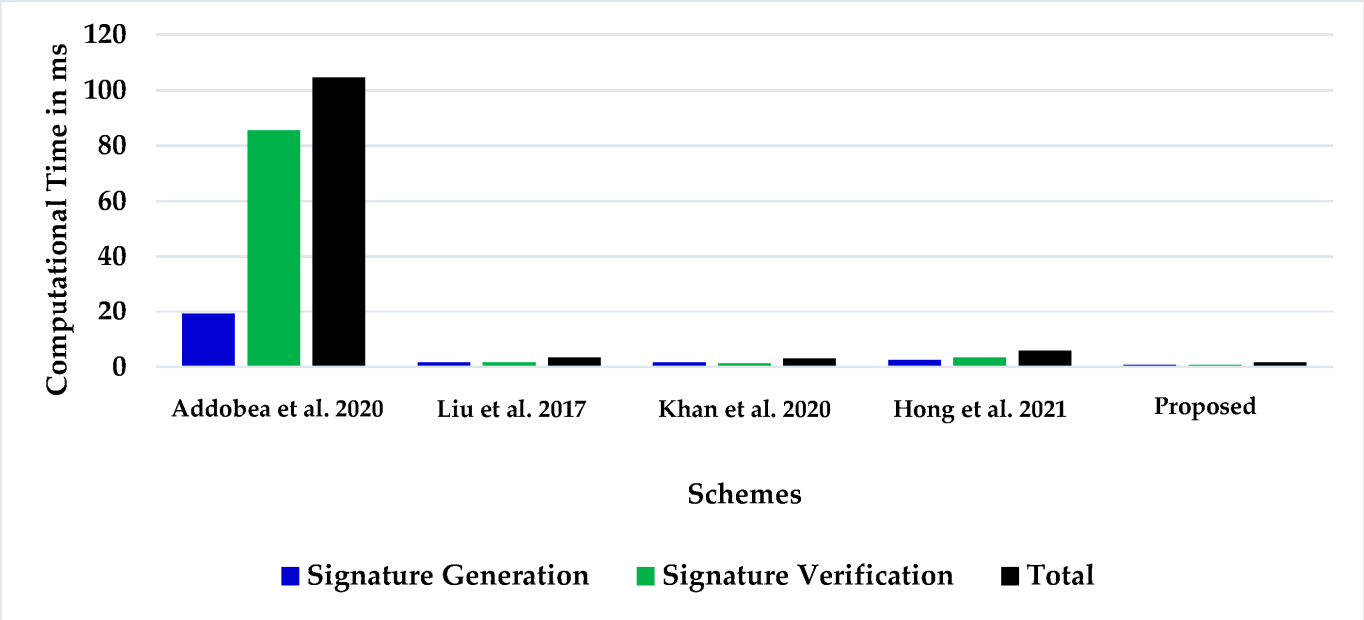

Figure 5. Computation time evaluation [19,20,22,24].

Percentage Improvement in terms of the Computation Time.
The computation time improvement is shown in Table 7 below.

Table 7. Computation overhead improvement.

Ref. No.	Computation Cost of Previous Scheme in MS	Computation Cost of Proposed	Percentage Improvement
Addobea et al. [20]	104.69	1.66	98.41
Liu et al. [19]	3.32	1.66	50
Khan et al. [22]	2.905	1.66	42.85
Hong et al. [24]	5.81	1.66	71.42

6.2. Communication Overhead

Specifically, we compare the proposed scheme with a few recent online/offline signature schemes, including those presented by Addobea et al. [20], Liu et al. [19], Khan et al. [22], and Hong et al. [24], in order to illustrate how the designed approach is more efficient in terms of the communication overhead. In order to do so, we assume that the length of elements in |G1| = |G2| = |G| = 1024 bits for bilinear pairing, |q| = 160 bits for the elliptic curve cryptosystem, |n| = 80 bits for the hyperelliptic curve cryptosystem, |m| = 100 bits, and |H| = 256 for the hash function [33]. Furthermore, Tables 8 and 9 depict the percentage improvement in the communication overhead that may be achieved by using the designed technique. Additionally, Figure 6 shows the results of an examination of the communication overhead of the certificateless online/offline signature systems. The vertical axis depicts the communication overhead in bits, which allows for a clear visual representation of the communication overhead. It demonstrates unequivocally that the designed strategy is more efficient than the previously designed approaches.

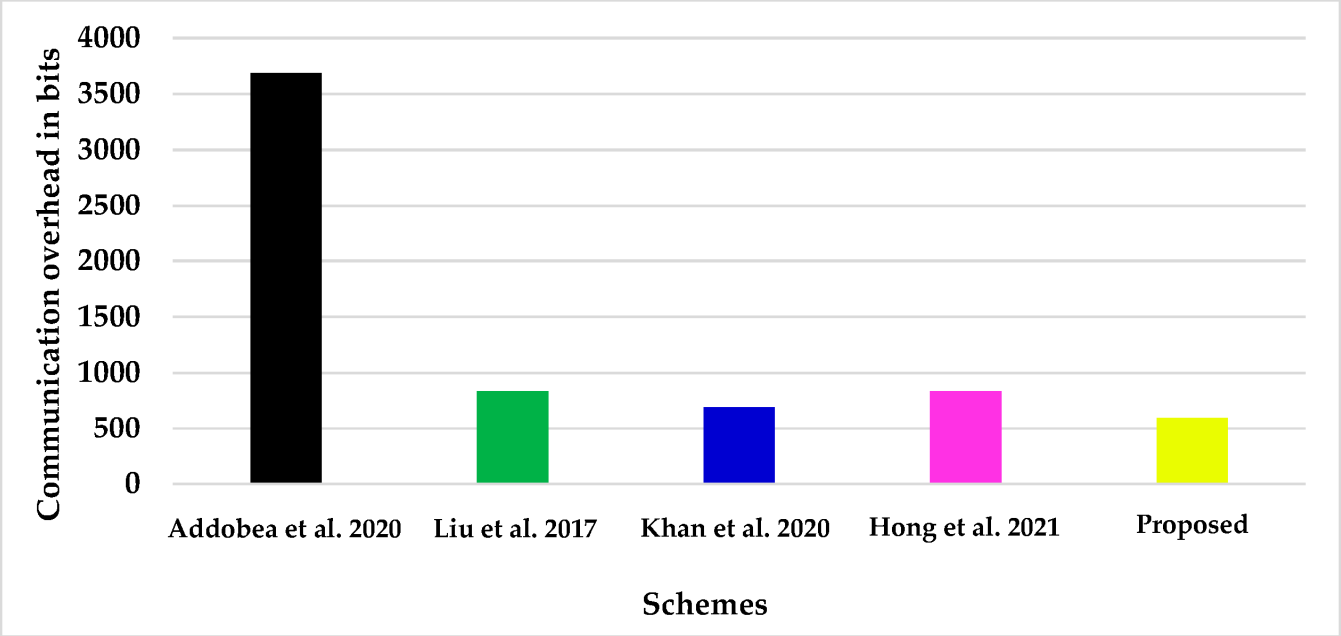

Figure 6. Communication overhead evaluation [19,20,22,24].

Table 8. Efficiency analysis of the communication overhead.

Operation/Ref. No	Addobea et al. [20]	Liu et al. [19]	Khan et al. [22]	Hong et al. [24]	Proposed
Ciphertext Size	$3\|G\| + \|m\| + 2\|\mathcal{H}\|$	$3\|n\| + \|m\| + 1\|\mathcal{H}\|$	$2\|q\| + \|m\| + 2\|\mathcal{H}\|$	$3\|n\| + \|m\| + 1\|\mathcal{H}\|$	$3\|q\| + \|m\| + 1\|\mathcal{H}\|$
Total communication overhead in bits	3684 bits	836 bits	692 bits	836 bits	596 bits

Table 9. Communication overhead improvement.

Ref. No.	CO of Previous Scheme in MS	CO of Proposed	Percentage Improvement
Addobea et al. [20]	3684	596	83.82
Liu et al. [19]	836	596	28.70
Khan et al. [22]	692	596	13.87
Hong et al. [24]	836	596	28.70

Percentage Improvement in terms of the communication overhead
The communication overhead improvement is shown in Table 9 below.

6.3. Performance Evaluation Using EDAS

EDAS is a standard approach that is utilized for testing and evaluating a variety of alternative options. Gorhabaee et al. [34] were the first people to apply the approach. The Positive Distance from Average and Negative Distance from Average solutions are the two functions that are used in EDAS to measure how far a solution is from the average [35]. EDAS is a multi-criteria decision-making (MCDM) approach that calculates the distance of all other solutions from the average solution and uses that specific information to select the best among the alternatives [36].

The EDAS is generally selected for a comparative analysis in a situation to solve the conflicting criteria [30]. Table 10 shows a comparative analysis of the selected performance metrics. In addition, the EDAS technique is used to select the most effective values for the four different methods, depending on the selected parameters.

Table 10. Performance metrics of the suggested schemes.

Weightage	**0.25**	**0.25**	**0.25**	**0.25**
Ref. NO.	**Computation Overhead (ms)**	**Communication Overhead (bits)**	**Security (Yes/NO)**	**Computational and Communicational Efficiency (Yes/NO)**
Addobea et al. [20]	104.69	3684	1	0
Liu et al. [19]	3.32	836	1	0.5
Khan et al. [22]	2.905	692	0	1
Hong et al. [24]	5.81	836	1	0.5
Proposed	1.66	596	1	1

Furthermore, the assessment scores (μ) were used to calculate the ranking based on the chosen parameters among the existing schemes. Table 10 evaluates the performance matrices of the previously proposed schemes, including ours.

Step One (Average Solution):
In this step, the average of the selected matrices is calculated.

$$(\phi) = [\vartheta_b]_{1\times\beta} \tag{1}$$

while

$$= \frac{\sum_{i=1}^{y} X_{ab}}{y} \tag{2}$$

In the stage before this one, one of the criteria for determining which solution to recommend is the performance of the matrices that were chosen. Precisely, in this step, the average of the selected matrices is calculated. As can be seen in Table 11, each calculated value on a chosen matrix can be derived as a solution to Equations (1) and (2), respectively.

Table 11. Average of the selected matrices.

Ref. NO.	Computation Overhead (ms)	Communication Overhead (bits)	Security (Yes/NO)	Computational and Communicational Efficiency (0,0.5,1)
Addobea et al. [20]	104.69	3684	1	0
Liu et al. [19]	3.32	836	1	0.5
Khan et al. [22]	2.905	692	0	1
Hong et al. [24]	5.81	836	1	0.5
Proposed	1.66	596	1	1
Average	23.677	1328.8	0.8	0.6

Step Two: Positive Distance from Average ($\mathcal{PD}_{avg}$)

In this step, the P_{dav} is calculated using the following equations:

$$\mathcal{PD}_{avg} = \left[\left(\mathcal{PD}_{avg}\right)_{ab}\right]_{\beta\times\beta} \tag{3}$$

If the state b^{th} is favorable, then

$$\left(\mathcal{PD}_{avg}\right)_{ab} = \frac{\mathcal{MAX}(0,(Ave_b - X_{ab}))}{Ave_b} \tag{4}$$

For the less favorable, it becomes

$$\left(\mathcal{PD}_{avg}\right)_{ab} = \frac{\mathcal{MAX}(0,(X_{ab} - Ave_b))}{Ave_b} \tag{5}$$

where $\mathcal{PD}_{avg}$ represents the Positive Distance from Average from the given average value on the a^{th} rating performance matrices.

Step Three: Negative Distance from Average ($\mathcal{ND}_{avg}$)

The $\mathcal{ND}_{avg}$ is calculated in this step using the following equations:

$$\left(\mathcal{ND}_{avg}\right) = \left[\left(\mathcal{ND}_{avg}\right)_{ab}\right]_{\beta\times\beta} \tag{6}$$

If the b^{th} criterion is more favorable than

$$\left(\mathcal{ND}_{avg}\right)_{ab} = \frac{\mathcal{MAX}(0,(Ave_b - X_{ab}))}{Ave_b} \tag{7}$$

and less desirable, then the given above equations become

$$\left(\mathcal{ND}_{avg}\right)_{ab} = \frac{\mathcal{MAX}(0,(X_{ab} - Ave_b))}{Ave_b} \tag{8}$$

where $\left(\mathcal{ND}_{avg}\right)_{ab}$ represents the Negative Distance from Average solution.

Step Four: Weighted Sum of the Positive Distance ($\mathcal{WSPD}_{avg}$)

The $\mathcal{WSPD}_{avg}$ for the given schemes are considered at this stage, as shown in Table 12.

$$\mathcal{WSPD}_{avg} = \sum_{b=1}^{y} \lambda_b(\mathcal{PD})_{ab} \tag{9}$$

Table 12. Weighted sum of the positive distance.

Ref. NO.	Computation Overhead (ms)	Communication Overhead (bits)	Security (Yes/NO)	Computational and Communicational Efficiency (Yes/NO)	$\mathcal{WSPD}_{avg}$
Addobea et al. [20]	0	0	0.0625	0	0.0625
Liu et al. [19]	0.214944883	0.092715232	0.0625	0	0.37016012
Khan et al. [22]	0.219326773	0.119807345	0	0.166666667	0.50580078
Hong et al. [24]	0.188653546	0.092715232	0.0625	0	0.34386878
Proposed	0.232472442	0.137868754	0.0625	0.166666667	0.59950786

Step Five: The Weighted Sum of the Negative Distance ($\mathcal{WSND}_{avg}$)

For the $\mathcal{WSND}_{avg}$ for the selected scheme obtained in this phase employing the following formula, the results are shown in Table 13.

$$\mathcal{WSND}_{avg} = \sum_{b=1}^{y} \lambda_b (\mathcal{ND})_{ab} \tag{10}$$

Table 13. Weighted sum of the negative distance.

Ref. NO.	Computation Overhead (ms)	Communication Overhead (bits)	Security (Yes/NO)	Computational and Communicational Efficiency (Yes/NO)	$\mathcal{WSND}_{avg}$
Addobea et al. [20]	0.855397643	0.443106562	0	0.25	1.54850421
Liu et al. [19]	0	0	0	0.041666667	0.04166667
Khan et al. [22]	0	0	0.25	0	0.25
Hong et al. [24]	0	0	0	0.041666667	0.04166667
Proposed	0	0	0	0	0

Step Six (Ranking)

The scores that were generated based on the $\mathcal{WSPD}_{avg}$ and $\mathcal{WSND}_{avg}$, are presented accordingly in the following Equations (11) and (12).

$$\mathcal{N}(\mathcal{WSPD}_{avg}) = \frac{\mathcal{WSPD}_{avg}}{\mathcal{MAX}_a(\mathcal{WSPD}_{avg})} \tag{11}$$

$$N(\mathcal{WSND}_{avg}) = 1 - \frac{\mathcal{WSND}_{avg}}{\mathcal{MAX}_a(\mathcal{WSND}_{avg})}. \tag{12}$$

The score values based on $\mathcal{N}(\mathcal{WSND}_{avg})$ and $\mathcal{N}(\mathcal{WSND}_{avg})$ are based on the evaluation scores (μ) for the rated schemes, as stated in Equation (13).

$$\mu = \frac{1}{2}(\mathcal{NWSPD}_{avg} - \mathcal{NWSND}_{avg}), \text{ where } 0 \leq \mu \geq 1 \tag{13}$$

We obtained the final result by utilizing both the $\mathcal{WSPD}_{avg}$ and $\mathcal{WSND}_{avg}$ average.

Following the steps outlined above establishes the extent of μ and provides the final ranking based on the parameters selected for the adopted schemes. According to the evaluation results, the best online/offline signature scheme obtains the highest scores. As may can be seen in Table 14, the proposed scheme has received very good evaluation scores (μ).

Table 14. Ranking under the selected parameters.

Ref. NO.	$\mathcal{WSPD}_{avg}$	$\mathcal{WSND}_{avg}$	$\mathcal{N}(\mathcal{WSPD}_{avg})$	$\mathcal{N}(\mathcal{WSND}_{avg})$	μ	Ranking
Addobea et al. [20]	0.0625	1.548504206	0.104252177	0.932675561	0.51846387	5
Liu et al. [19]	0.370160115	0.041666667	0.617439968	0.601266845	0.60935341	3
Khan et al. [22]	0.505800784	0.25	0.84369333	0.455155933	0.64942463	2
Hong et al. [24]	0.343868777	0.041666667	0.573585101	0.629587638	0.60158637	4
Proposed	0.599507862	0	1	0.354215509	0.67710775	1

According to the conclusive findings of the EDAS technique, the overall performance of our scheme is superior than that of the earlier online/offline signature schemes. On the basis of a comparison study using fuzzy logic-based EDAS, the new scheme is superior to that of Khan et al. [22] and Liu et al. [19], which come in second and third, respectively. The Hong et al. [24] approach, on the other hand, comes in fourth place in the chosen matrix.

7. Summary of the Findings

To the best of our knowledge,we designed the first ever online/offline signature scheme for UWSNs. The proposed scheme makes the least possible use of computational and communicational resources by employing lightweight HEC. In addition to that, the proposed scheme uses the idea of online/offline signatures in order to lessen the load on the sensors nodes. A fuzzy-based EDAS technique was applied in the proposed system in order to illustrate both the practicability and effectiveness of the given approach. According to the results of the findings, the proposed scheme is superior in terms of the chosen parameters. Finally, an application shown where the proposed scheme is deployed.

8. Conclusions

The paper presents a lightweight certificateless online/offline signature scheme for underwater wireless sensor networks (UWSNs). The signature is completed in two stages, according to the proposed scheme, the first of which takes place online and the second of which takes place offline. In the absence of a message, the offline phase is responsible for carrying out computationally complex operations, whereas the online phase is responsible for carrying out computations that are more straightforward and less intensive. In addition to this, the proposed scheme utilized a lightweight hyperelliptic curve cryptosystem that has an 80-bit key size in order to bring down the overall cost of the UWSNs even further. Additionally, the newly proposed scheme is compared with the previously suggested online and offline signature schemes with regards to the amount of computation time and communication overhead. In comparison to the previous schemes, the proposed schemes minimize the amount of time needed for computation from 50% to 98.41% and reduces the amount of communication overhead from 13.87% to 83.82%. In addition, the proposed scheme is proven secure in the random oracle model under the hyperelliptic curve discrete logarithm problem. The feasibility of a proposed scheme is demonstrated by a security analysis and comparisons with the relevant current schemes. A decision-making strategy known EDAS was also used to demonstrate the design effectiveness in multiple criteria. Finally, we presented a scenario in which the proposed approach can be practically applied on underwater wireless sensor networks.

Author Contributions: Conceptualization, S.S.U., S.H., M.U., R.A. (Roobaea Alroobaea), J.I., A.M.B., M.A. and R.A. (Raed Alsaqour); data curation, S.S.U. and S.H., Formal analysis, S.S.U., S.H., J.I., R.A. (Roobaea Alroobaea), and M.A.; funding acquisition, R.A. (Roobaea Alroobaea), J.I., A.M.B., M.A. and R.A. (Raed Alsaqour); investigation, S.S.U., S.H. and R.A. (Raed Alsaqour); methodology, S.S.U., S.H., M.U., R.A. (Roobaea Alroobaea), M.A., and R.A. (Raed Alsaqour); visualization, S.S.U., S.H., J.I., A.M.B. and M.A.; writing—original draft, S.S.U., S.H., M.U., R.A. (Roobaea Alroobaea), J.I., A.M.B., M.A. and R.A. (Raed Alsaqour); writing—review & editing, S.H., M.U., M.A., R.A (Raed Alsaqour). All authors have read and agreed to the published version of the manuscript.

Funding: The authors are grateful to the Taif University Researchers Supporting Project, number TURSP-2020/36, Taif University, Taif, Saudi Arabia. In addition, this research was funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R97), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are grateful to the Taif University Researchers Supporting Project number (TURSP-2020/36), Taif University, Taif, Saudi Arabia. In addition, this research was funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R97), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

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

References

1. Alfouzan, F.A. Energy-efficient collision avoidance MAC protocols for underwater sensor networks: Survey and challenges. *J. Mar. Sci. Eng.* **2021**, *9*, 741. [CrossRef]
2. Sandhiyaa, S.; Gomathy, C.A. Survey on underwater wireless sensor networks: Challenges, requirements, and opportunities. In Proceedings of the 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 11–13 November 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1417–1427.
3. Gul, H.; Ullah, G.; Khan, M.; Khan, Y. EERBCR: Energy-efficient regional based cooperative routing protocol for underwater sensor networks with sink mobility. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *2021*, 1–13. [CrossRef]
4. Yang, G.; Dai, L.; Wei, Z. Challenges, threats, security issues and new trends of underwater wireless sensor networks. *Sensors* **2018**, *18*, 3907. [CrossRef] [PubMed]
5. Heidemann, J.; Ye, W.; Wills, J.; Syed, A.; Li, Y. Research challenges and applications for underwater sensor networking. In Proceedings of the IEEE Wireless Communications and Networking Conference, Las Vegas, NV, USA, 3–6 April 2006; IEEE: Piscataway, NJ, USA, 2006; Volume 1, pp. 228–235.
6. Carman, D.W.; Kruus, P.S.; Matt, B.J. *Constraints and Approaches for Distributed Sensor Network Security (Final)*; DARPA Project Report; Cryptographic Technologies Group, Trusted Information System, NAI Labs: Los Angeles, CA, USA, 2000; Volume 1, pp. 1–39.
7. Blom, R. An optimal class of symmetric key generation systems. In *Workshop on the Theory and Application of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 1984; pp. 335–338.
8. Khan, M.A.; Alzahrani, B.A.; Barnawi, A.; Al-Barakati, A.; Irshad, A.; Chaudhry, S.A. A resource friendly authentication scheme for space–air–ground–sea integrated Maritime Communication Network. *Ocean. Eng.* **2022**, *250*, 110894. [CrossRef]
9. Luo, Y.; Pu, L.; Peng, Z.; Shi, Z. RSS-based secret key generation in underwater acoustic networks: Advantages, challenges, and performance improvements. *IEEE Commun. Mag.* **2016**, *54*, 32–38. [CrossRef]
10. Peng, C.; Du, X.; Li, K.; Li, M. An ultra-lightweight encryption scheme in underwater acoustic networks. *J. Sens.* **2016**, *2016*, 8763528. [CrossRef]
11. Hamid, M.A.; Abdullah-Al-Wadud, M.; Hassan, M.M.; Almogren, A.; Alamri, A.; Kamal, A.R.; Mamun-Or-Rashid, M. A key distribution scheme for secure communication in acoustic sensor networks. *Future Gener. Comput. Syst.* **2018**, *86*, 1209–1217. [CrossRef]
12. Ateniese, G.; Capossele, A.; Gjanci, P.; Petrioli, C.; Spaccini, D. SecFUN: Security framework for underwater acoustic sensor networks. In Proceedings of the OCEANS 2015—Genova, Genova, Italy, 18–21 May 2015; pp. 1–9.
13. Capossele, A.; Petrioli, C.; Saturni, G.; Spaccini, D.; Venturi, D. Securing underwater communications: Key agreement based on fully hashed MQV. In Proceedings of the International Conference on Underwater Networks & Systems 2017, Halifax, NS, Canada, 6–17 November 2017; pp. 1–5.
14. Dini, G.; Duca, A.L. A secure communication suite for underwater acoustic sensor networks. *Sensors* **2012**, *12*, 15133–15158. [CrossRef]
15. Karati, A.; Islam, S.H.; Karuppiah, M. Provably secure and lightweight certificateless signature scheme for IIoT environments. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3701–3711. [CrossRef]
16. Even, S.; Goldreich, O.; Micali, S. On-Line/Off-Line Digital Signature Schemes. In *Advances in Cryptology-CRYPTO'89 Proceedings*; Springer: New York, NY, USA, 1990.
17. Shamir, A.; Tauman, Y. Improved online/offline signature schemes. In Proceedings of the Annual International Cryptology Conference, Santa Barbara, CA, USA, 19–23 August 2001; Springer: Berlin/Heidelberg, Germany, 2001; pp. 355–367.
18. Chen, X.; Zhang, F.; Susilo, W.; Mu, Y. Efficient generic on-line/off-line signatures without key exposure. In Proceedings of the International Conference on Applied Cryptography and Network Security, Kamakura, Japan, 21–24 June 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 18–30.

19. Liu, D.; Zhang, S.; Zhong, H.; Shi, R.; Wang, Y. An efficient identity-based online/offline signature scheme without key escrow. *Int. J. Netw. Secur.* **2017**, *19*, 127–137.
20. Addobea, A.A.; Hou, J.; Li, Q. MHCOOS: An offline-online certificateless signature scheme for m-health devices. *Secur. Commun. Netw.* **2020**, *2020*, 7085623. [CrossRef]
21. Xu, F.; Zeng, H. Cryptanalysis of Two Signature Schemes for IoT and Mobile Health Systems. *Wirel. Pers. Commun.* **2021**, *19*, 1–9. [CrossRef]
22. Khan, M.A.; Rehman, S.U.; Uddin, M.I.; Nisar, S.; Noor, F.; Alzahrani, A.; Ullah, I. An online-offline certificateless signature scheme for Internet of health things. *J. Healthc. Eng.* **2020**, *2020*, 6654063. [CrossRef]
23. Hussain, S.; Sajid Ullah, S.; Shorfuzzaman, M.; Uddin, M.; Kaosar, M. Cryptanalysis of an online/offline certificateless signature scheme for Internet of Health Things. *Intell. Autom. Soft Comput.* **2021**, *30*, 983–993. [CrossRef]
24. Hong, H.; Hu, B.; Sun, Z. An Efficient and Secure Attribute-Based Online/Offline Signature Scheme for Mobile Crowdsensing. *Hum.-Cent. Comput. Inf. Sci.* **2021**, *11*, 26.
25. Choi, K.Y.; Park, J.H.; Lee, D.H. A new provably secure certificateless short signature scheme. *Comput. Math. Appl.* **2011**, *61*, 1760–1768. [CrossRef]
26. Wollinger, T.; Pelzl, J.; Paar, C. Cantor versus Harley: Optimization and analysis of explicit formulae for hyperelliptic curve cryptosystems. *IEEE Trans. Comput.* **2005**, *54*, 861–872. [CrossRef]
27. Wollinger, T.; Pelzl, J.; Wittelsberger, V.; Paar, C.; Saldamli, G.; Koç, Ç.K. Elliptic and hyperelliptic curves on embedded µP. *ACM Trans. Embed. Comput. Syst. (TECS)* **2004**, *3*, 509–533. [CrossRef]
28. Hussain, S.; Ullah, I.; Khattak, H.; Adnan, M.; Kumari, S.; Ullah, S.S.; Khan, M.A.; Khattak, S.J. A lightweight and formally secure certificate based signcryption with proxy re-encryption (CBSRE) for Internet of Things enabled smart grid. *IEEE Access* **2020**, *8*, 93230–93248. [CrossRef]
29. Hussain, S.; Ullah, S.S.; Gumaei, A.; Al-Rakhami, M.; Ahmad, I.; Arif, S.M. A novel efficient certificateless signature scheme for the prevention of content poisoning attack in named data networking-based internet of things. *IEEE Access* **2021**, *9*, 40198–40215. [CrossRef]
30. Ullah, S.S.; Ullah, I.; Khattak, H.; Khan, M.A.; Adnan, M.; Hussain, S.; Amin, N.U.; Khattak, M.A. A lightweight identity-based signature scheme for mitigation of content poisoning attack in named data networking with internet of things. *IEEE Access* **2020**, *8*, 98910–98928. [CrossRef]
31. Rehman, M.; Khattak, H.; Alzahrani, A.S.; Ullah, I.; Adnan, M.; Ullah, S.S.; Amin, N.U.; Hussain, S.; Khattak, S.J. A lightweight nature heterogeneous generalized signcryption (HGSC) scheme for named data networking-enabled Internet of Things. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 8857272. [CrossRef]
32. Tourani, R.; Misra, S.; Mick, T.; Panwar, G. Security, privacy, and access control in information-centric networking: A survey. *IEEE Commun. Surv. Tutor.* **2017**, *20*, 566–600. [CrossRef]
33. Ullah, S.S.; Hussain, S.; Gumaei, A.; AlSalman, H. A secure NDN framework for Internet of Things enabled healthcare. *Comput. Mater. Contin.* **2021**, *67*, 223–240.
34. Keshavarz Ghorabaee, M.; Zavadskas, E.K.; Olfat, L.; Turskis, Z. Multi-criteria inventory classification using a new method of Evaluation Based on Distance from Average Solution (EDAS). *Informatica* **2015**, *26*, 435–451. [CrossRef]
35. Zadeh, L.A. Fuzzy logic. *Computer* **1988**, *21*, 83–93. [CrossRef]
36. Mehmood, G.; Khan, M.Z.; Waheed, A.; Zareei, M.; Mohamed, E.M. A trust-based energy-efficient and reliable communication scheme (trust-based ERCS) for remote patient monitoring in wireless body area networks. *IEEE Access* **2020**, *8*, 131397–131413. [CrossRef]