mathematics

MDPI

Article

Provably Secure Linearly Homomorphic Aggregate Signature Scheme for Electronic Healthcare System

Yanyan Gu [1], Limin Shen [1,*], Futai Zhang [2] and Jinbo Xiong [2]

[1] School of Computer and Electronic Information, Nanjing Normal University, Nanjing 210023, China; yanygu@njnu.edu.cn
[2] College of Computer and Cyber Security, Fujian Normal University, Fuzhou 350117, China; futai@fjnu.edu.cn (F.Z.); jbxiong@fjnu.edu.cn (J.X.)
* Correspondence: shenlimin@njnu.edu.cn

Abstract: In recent years, deploying Internet of Things (IoT) in electronic healthcare systems (EHS) has made great progress in healthcare detection. It is extremely important to reduce the cost of communication and ensure the authenticity and integrity of data. A linearly homomorphic signature scheme can solve the above problems. However, when the scale of EHS is too large, the transmission, storage and verification of signatures need a high cost. An aggregate signature can combine many signatures generated by many different users into a short one. Therefore, only one aggregate signature needs to be processed during verification, transmission and storage. Combining the advantages of aggregate signature and linearly homomorphic signature, this paper proposes an aggregate signature scheme based on a linearly homomorphic signature for EHS, which has both linear homomorphism and aggregation, and realizes double data compression. Moreover, our scheme can resist a potential real attack, named a coalition attack. The security of this scheme is rigorously demonstrated based on the computational Diffie–Hellman assumption in the random oracle model.

Keywords: homomorphic signature; aggregate signature; linearly homomorphic aggregate signature; electronic healthcare system

MSC: 94A62

Citation: Gu, Y.; Shen, L.; Zhang, F.; Xiong, J. Provably Secure Linearly Homomorphic Aggregate Signature Scheme for Electronic Healthcare System. *Mathematics* **2022**, *10*, 2588. https://doi.org/10.3390/math10152588

Academic Editor: Daniel-Ioan Curiac

Received: 16 June 2022
Accepted: 21 July 2022
Published: 25 July 2022

1. Introduction

Recently, the vigorous development of IoT has brought more and more convenience to our daily life. The typical IoT technology connects intelligent devices such as sensors, mobile terminals and intelligent facilities to the network through wireless or wired means. The application of IoT technology in medical treatment makes the intelligent electronic healthcare system possible. The EHS using IoT technology breaks through the limitation of time and place. Patients can get medical advice through IoT network at anytime and anywhere. The sensors are deployed on the patient, collect various medical data of the patient, such as blood pressure, temperature, pulse, PH value, etc., and transmit these data to Medical-Server through the IoT network instantly. Medical Staff judge the patient's status according to the received data and provide the best medical advice. The Medical Staff can provide the best treatment for patients as soon as possible without a face-to-face with patients.

In order to obtain the information about patients as soon as possible so that Medical Staff can provide the best treatment suggestions to patients soon enough, it is extremely important to reduce the cost of communication. The network coding method proposed by Ahlswede et al. in 2000 is an effective technology for reducing the cost of communication and improving the robustness and fault tolerance [1]. Although network coding has many advantages compared with traditional routing, it is also vulnerable to the pollution attack. If the data are polluted, the paths of the data packet will be polluted, and the destination

node cannot correctly parse the original data. To a certain extent, the transmission of the polluted data packets to the Medical-Server will result in the waste of medical resources.

At present, in order to solve the problem of pollution attacks, a large number of researchers have put forward many solutions. According to different research methods, it can be divided into three categories: network-based error correction code, information theory and cryptography. Cryptographic methods are employed most extensively. Most of the cryptography methods use a linearly homomorphic signature, so that both the intermediate node and the destination node can detect whether the malicious information exists. The intermediate node can also discard the polluted data packets, which greatly reduces the waste of resources in the communication process. Previously, many linearly homomorphic signature schemes were proposed in [2–8]. However, most of the existing linearly homomorphic signature schemes, when the scale of the medical system is too large, not only need to pay high transmission and storage costs, but also need to verify each signature from the users one by one. In the real world, if a Medical-Server verifies the signature sent by each patient one by one, it will inevitably increase the burden of the Medical-Server and reduce the efficiency of medical services to a certain extent.

In order to tackle the above issues, we propose an aggregate signature scheme based on a linearly homomorphic signature proposed by Boneh et al. [3]. The homomorphic signature can be used for data compression. An aggregate signature can compress multiple signatures generated by different users on different messages into a short one, which has great advantages in reducing bandwidth and storage cost. Combining the advantages of the linearly homomorphic signature scheme and the aggregate signature scheme, our scheme has good properties and can realize double data compression. The Medical-Server can verify the validity of the multiple patient-generated signature by verifying only one aggregate signature, so as to improve the efficiency of medical services. However, there is a potentially powerful attack in the aggregate signature scheme, the coalition attack, which may destroy the authenticity and integrity of the aggregate signature. A coalition attack means that some signers use a set of single signatures, including at least one invalid single signature, which can generate a valid aggregate signature. Therefore, considering the influence of the above situations, we propose an aggregate signature scheme based on a linear homomorphic signature scheme which can resist a coalition attack for EHS.

1.1. Related Work

With the development of IoT technology, more and more electronic healthcare systems apply this technology, which makes wireless smart healthcare treatment possible. At present, many various electronic healthcare systems have been proposed [9–18]. A typical EHS model is shown in Figure 1. No matter where the patient is or who has deployed the sensors, the patient can forward collected medical data to the Medical-Server with the aim of obtaining medical suggestions in time. In order to obtain the best treatment time, it is very important to reduce the cost of communication and ensure the security of medical data.

The concept of the homomorphic signature was proposed by Johnson et al. in CT-RSA 2002 [2]. With the further study of the linearly homomorphic signature [4,5,19,20], it is found that the linearly homomorphic signature can be used to address the above problems. In Boneh et al.'s scheme [3], the signature on any vector of a linear combination of the basis vectors for a linear subspace $\boldsymbol{V}$ can be publicly derived by the signatures on those basis vectors. That is to say, once receiving the signatures on a set of basis vectors for $\boldsymbol{V}$ and the corresponding linear coefficients, a signature on any vector of $\boldsymbol{V}$ can be computed. Attrapadung et al. presented a homomorphic network coding signature scheme in the standard model [6]. A homomorphic subspace signature for network coding was introduced by Cheng et al. [7]. Then, Chang et al. first described the definition and security model of a certificateless homomorphic signature (CLHS) and proposed a CLHS scheme for network coding [8]. Subsequently, many researchers applied a linearly homomorphic signature to the electronic healthcare system which was deployed with the IoT. Li et al.

proposed a lightweight certificateless linearly homomorphic network coding signature scheme for EHS [9] which required a relatively low computational cost to sign and verify data packets. However, when the scale of the EHS is too large, the Medical-Server needs to verify the message signature from each patient one by one, which will inevitably increase the storage cost and bandwidth burden of the EHS.

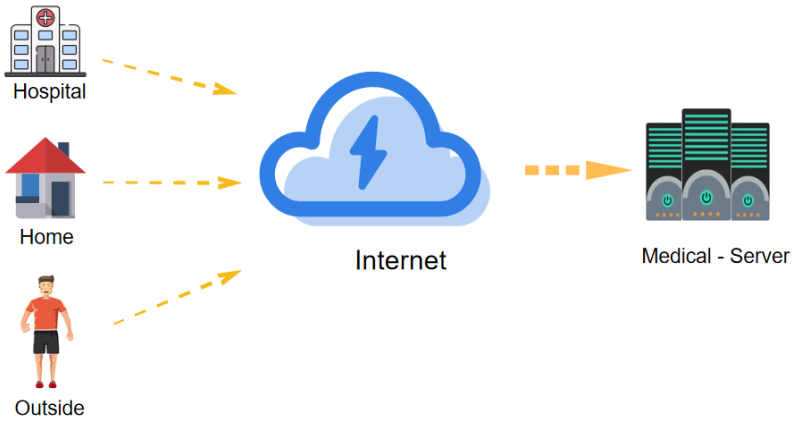

Figure 1. A typical EHS model.

In 2003, the conception of the aggregate signature was introduced by Boneh et al. [21]. The aggregate signature can compress multiple signatures generated by different users on different messages into a single short aggregate signature. Hence, it is useful in reducing storage cost and bandwidth, and can be an important primitive to improve the efficiency of EHS. Many researchers are interested in designing efficient and secure aggregate signature schemes. Many aggregate signature schemes have been put forward, such as aggregate signature schemes in public key infrastructures [22,23], identity-based aggregate signature schemes [24–26], certificateless aggregate signature schemes [27–29] and certificate-based aggregate signature schemes [30,31]. Many aggregate signature schemes have been used in the IoT, such as the anonymous traceable aggregate signature scheme proposed by Li et al. [32]. Unfortunately, many of the existing aggregate signature schemes can not resist coalition attacks, which can destroy the aggregate messages' validity and integrity. In recent years, some aggregate signature schemes have been able to resist coalition attacks [13,33–37]. In 2020, a potential and realistic attack called fully chosen-key attacks was introduced by Wu et al. [37]. Meanwhile, they gave a certificateless aggregate signature scheme secure against these coalition attacks. Combining the advantages of linearly homomorphic signatures and aggregate signatures, many researchers proposed linearly homomorphic aggregate signature (LHAS) schemes in various cryptographic systems to realize double data compression. The diagram of the LHAS scheme is shown in Figure 2. The patients sign data to generate data packets, which are transmitted to the Aggregator-Server through the IoT network. Then the data packets are submitted to the verifier for verification.

To guarantee the security of multi-source network coding and secure sensor data aggregation, Zhang et al. proposed a homomorphic aggregate signature (HAS) scheme [38] which is secure under the lattice-based in homogeneous smallest integer solution assumption, and then Jing gives another lattice-based HAS scheme [39] which has a shorter signature length and high efficiency. This paper gives an LHAS scheme for EHS based on the linearly homomorphic signature described in [3]. This LHAS scheme holds both linearly homomorphic property and aggregate property, can achieve double data compression. One is that the linearly homomorphic signature can be used in data compression, and the other is the aggregation of homomorphic signatures can realize the batch validation for data integrity. So, the LHAS scheme is very helpful in batch verification for data integrity in IoT.

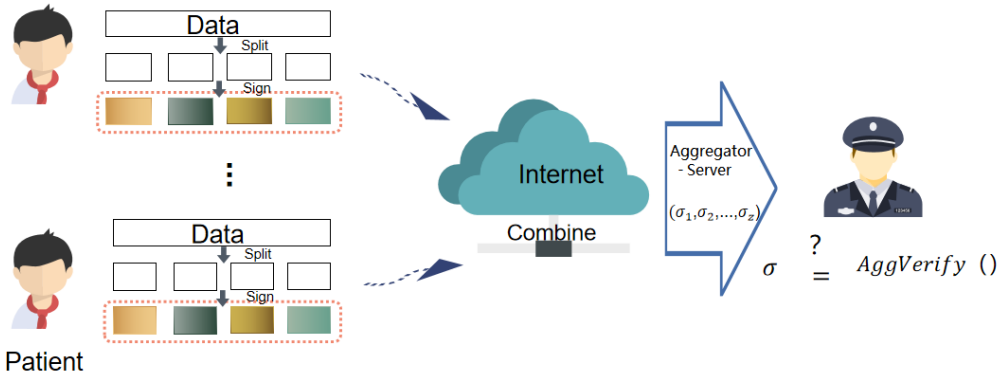

Figure 2. The diagram of the LHAS scheme.

1.2. Our Contributions

The new deployment of IoT in electronic healthcare systems reduces the need for medical professionals and improves the efficiency of the diagnosis. In this system, it is very crucial to ensure the authenticity, integrity of medical data and transmit it to Medical Staff in real time. In the limited network throughput and computing efficiency, how to improve the transmission rate of medical data while ensuring the authenticity and integrity of data is an urgent problem to be solved. Therefore, we construct an LHAS scheme that can resist a coalition attack. The main contributions are as follows:

- A security model of the LHAS scheme is proposed. The security model guarantees an aggregate signature is valid if and only if all the individual signatures generating the aggregate signature are valid;
- An aggregate signature scheme based on a linearly homomorphic signature is proposed, which makes the scheme have both a linearly homomorphic property and an aggregate property, and realizes double data compression. The transmission efficiency is further improved and the storage cost of medical data in EHS is reduced.
- The security of our LHAS scheme is rigorously presented under the proposed security model. Moreover, through the analysis of comparative performance, we show that the scheme is effective in reducing transmission and storage overhead.

1.3. Organization

In the following section, we introduce the preliminaries demanded in this paper, including some basic notions such as bilinear pairing and complexity assumptions. Section 3 presents the definition and security model of the LHAS scheme. In Section 4, we give our LHAS scheme with a designated verifier, then in Section 5 we provide the detailed security analysis. We present the performance analysis in Section 6. Finally, Section 7 is the conclusion.

2. Preliminaries

This section revisits some basic notions required in this paper.

2.1. Bilinear Pairing

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of the same prime order p, and let h be a generator in $\mathbb{G}$.

Definition 1. *If a map $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ satisfies the following properties, it is called a bilinear pairing:*

- *Bilinear: for all $\zeta, \eta \in \mathbb{G}$ and $\iota, \tau \in \mathbb{Z}_p^*$, $\hat{e}(\zeta^\iota, \eta^\tau) = \hat{e}(\zeta, \eta)^{\iota\tau}$.*
- *Non-degenerate: $\hat{e}(h, h) \neq 1_T$, where 1_T is the identity element of $\mathbb{G}_T$.*
- *Computable: for all $\zeta, \eta \in \mathbb{G}$, $\hat{e}(\zeta, \eta)$ is efficiently computable.*

2.2. Complexity Assumptions

Definition 2. *Computational Diffie–Hellman problem (CDH problem): Given the elements* $g, g^{\iota}, g^{\tau} \in \mathbb{G}$, *to compute* $g^{\iota\tau} \in \mathbb{G}$.

The CDH assumption states that the CDH problem is hard, i.e., there is no probabilistic polynomial-time (PPT) algorithm that can solve the CDH problem.

3. Outline of Linearly Homomorphic Aggregate Signature Schemes

3.1. Definition of LHAS Schemes

A linearly homomorphic aggregate signature scheme consists of seven PPT algorithms: **Setup**, **Key Extract**, **Sign**, **Combine**, **Verify**, **Agg** and **AggVerify** which satisfy the following functionality:

- **Setup:** Given a security parameter k and output the system parameters SP;
- **KeyExtract:** Given the SP, output a public key PK and a secret key SK. (SP is the input to the following algorithms);
- **Sign:** Given a secret key SK, a file identifier $name \in \{0,1\}^k$, a vector $\boldsymbol{v}_j \in \mathbb{Z}_p^N$, outputs a signature σ_j;
- **Combine:** Given a public key PK, a file identifier $name \in \{0,1\}^k$ and a set of tuples $\{(\beta_j, \sigma_j), \beta_j \in \mathbb{Z}_p\}$, where σ_j is the signature on vector $\boldsymbol{v}_j$, outputs a signature σ on the linear combination $\sum \beta_j \boldsymbol{v}_j$ without knowledge of SK;
- **Verify:** Given a public key PK, a file identifier $name \in \{0,1\}^k$, a signature σ and a vector $\boldsymbol{y} \in \mathbb{Z}_p^N$ ($\boldsymbol{y} = \sum \beta_j \boldsymbol{v}_j$), outputs either 1 or 0 (*accept* or *reject*);
- **Agg:** Given an aggregating subset of users $U \in \mathfrak{U}(|U| = l)$, and the signatures σ_i on the linear combination $\boldsymbol{y}_i = \sum \beta_{ij} \boldsymbol{v}_{ij}$ ($i = 1, \ldots, l$), outputs the aggregate signature;
- **AggVerify:** Given an aggregating subset U of l aggregating signers with their public keys, an aggregate signature on the vectors $\boldsymbol{y}_i = \sum \beta_{ij} \boldsymbol{v}_{ij}$ ($i = 1, \ldots, l$) with file identifiers, where vectors $\boldsymbol{v}_{i1}, \boldsymbol{v}_{i2}, \ldots, \boldsymbol{v}_{im_i}$ is a set of basis of subspaces $\boldsymbol{V}_i$ ($i = 1, \ldots, l$), outputs *accept* if the aggregate signature is valid, or *reject* otherwise.

3.2. System Model of LHAS-EHS

The LHAS-EHS model we proposed mainly includes five components, i.e., Health-Center, User-Sensor, Combine-Node, Aggregator-Server and Medical-Server. Health-Center initializes the system and generates the various public parameters required. The sensor node, User-Sensor, deployed on the patient, is used as the source node of the EHS. The source node collects various data from patients and regards the collected medical data packet as a subspace $\boldsymbol{V}$ with the file identifier $name \in \{0,1\}^*$. The subspace is split into m vectors. The source node should sign the vector $\boldsymbol{v}_j$ before sending the data packets to the next node. Let $\boldsymbol{V}=\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m\} \subset \mathbb{F}_p^N$ ($0 < m < N$) denote the subspace with the file identifier $name \in \{0,1\}^*$, where $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m$ is any basis of $\boldsymbol{V}$. Then, the User-Sensor loads the vector and its associated signature information into a data packet. The node sends data packets to the Combine-Node (such as router and repeater). The Combine-Node encodes the received data packets and combines them. Then, the Combine-Node forwards the merged data packets to the next node. When the Aggregator-Server receives a certain number of data packets, it generates an aggregate signature and randomly forwards the aggregate information to a destination Medical-Server, which may come from a different medical domain. (In order to ensure the sharing of medical data between Medical-Servers, blockchain technology can be considered to establish a medical alliance chain to ensure the information exchange between Medical-Servers and improve the efficiency.) Then, the Medical-Server checks the received aggregate signature's validity and decodes the data packets. If the verification is passed, the medical staff will judge the patient's physical state according to the received medical data and provide corresponding medical suggestions. Meanwhile, ensuring the security of the feedback path of medical suggestions is also a matter of concern. If necessary, we can use a classical signature scheme to solve this

problem, such as the BLS signature scheme [40]. The LHAS-EHS system model is shown in Figure 3.

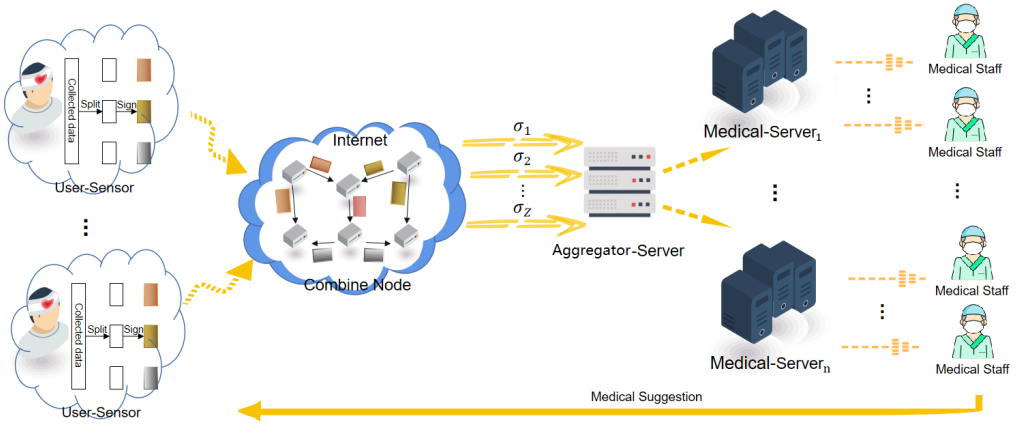

Figure 3. System model of the LHAS-EHS.

- Health-Center: The Health-Center generates the system public parameters by running the algorithm Setup. In the initialization phase, it generates the required public-private key pairs for User-Sensor and Medical-Server;
- User-Sensor: The User-Sensor as the source node connects the patient's body to obtain various medical information. It is a resource-limited device, which belongs to a care district. The node divides the collected data packet into m vectors and signs them. Then, the node loads vectors and related signature information into a data packet. After processing the collected data, the node sends data packets to the neighbor nodes, the Combine-Node;
- Combine-Node: The nodes combine all data packets and the corresponding signatures with the coding coefficient. Then, the nodes forward merged data packet to the Aggregator-Server;
- Aggregator-Server: A device that is honest but curious and has certain calculation and communication capabilities. The device can randomly obtain a Medical-Server's public key, collect and aggregate a set of the merged data packets. Finally, send the generated aggregate signature to the destination Medical-Server;
- Medical-Server: A device that has strong computing power and storage space can process all medical data collected by the User-Sensors. The device works as a designated verifier who can verify the aggregate signature using its secret key. If the aggregate signature can pass the verification, the Medical-Server provides the patients' data information to the medical staff. Then, the medical staff will provide corresponding medical suggestions to patients according to the data received.

3.3. Security Model

Obviously, the goal of an adversary is the existential forgery of an aggregate signature. A linearly homomorphic aggregate signature scheme is secure if the following conditions are satisfied:

- The basic signature scheme involved is existentially unforgeable against adaptive chosen message attacks (EUF-CMA secure);
- The security of aggregate algorithm should stand up against all kinds of coalition attacks.

A detailed instruction for the basic linearly homomorphic signature scheme's security model was provided in [3], so we mainly focus on the security of the aggregation algorithm. An adversary's purpose is to use a set of individual single signatures with at least one invalid single signature to forge a valid aggregate signature. The security model is similar

to the one in [13], the adversary can access relevant oracles to obtain all signer's secret keys. Now we revisit the security model in [13] through the following game between an adversary $\mathcal{A}$ and a challenger $\mathcal{B}$. The game consists of three steps: **Setup**, **Queries** and **Forge**.

- **Setup:** When inputting the security parameter k, the challenger $\mathcal{B}$ generates the system parameters SP. Furthermore, $\mathcal{B}$ randomly generates the public-secret key pair (PK_{ver}, SK_{ver}) for a designated verifier, then $\mathcal{B}$ gives $\mathcal{A}$ the SP and PK_{ver};
- **Queries:** $\mathcal{A}$ can access the following queries:
  - **Secret key request query** $\mathcal{O}_{SK}(u_i)$: $\mathcal{A}$ requests such a query, $\mathcal{B}$ generates the key pairs (PK_{u_i}, SK_{u_i}) by running the algorithm **Key Extract**, then returns SK_{u_i} to $\mathcal{A}$;
  - **AggVerify request query** $\mathcal{O}_{AggV}(\{SP, u_i, PK_{u_i}, \boldsymbol{V}_i, \beta_j^i, i = 1, \ldots, l\}, \sigma)$: On receiving such a query, $\mathcal{B}$ responds whether σ is valid by running algorithm **AggVerify**. Where subspace $\boldsymbol{V}_i$ is depicted as a set of basis vectors $\boldsymbol{v}_1^i, \boldsymbol{v}_2^i, \ldots, \boldsymbol{v}_{m_i}^i, i = 1, \ldots, l$;
- **Forge:** Finally, $\mathcal{A}$ outputs its forgery $(\{u_i, \boldsymbol{V}_i, \beta_j^i, \sigma_j^i, i = 1, \ldots, l\}, \sigma^*)$.

If the following two conditions are satisfied, $\mathcal{A}$ will win the game:

- The aggregate signature σ^* is valid;
- At least one single signature is invalid.

4. The Linearly Homomorphic Aggregate Signature Scheme

In this section, we adopt the homomorphic signature scheme [3] as the basis for constructing our LHAS scheme. Our scheme can guarantee that if the aggregate signature is valid then all signers involved in the Aggregate algorithm signed the corresponding message correctly. The LHAS scheme consists of seven algorithms: **Sign**, **KeyExtract**, **Combine**, **Verify**, **Agg** and **AggVerify**.

- **Setup:** Given a security parameter k, Health-Center runs this algorithm:
  - Generate a bilinear group $(\mathbb{G}, \mathbb{G}_T, p, \hat{e})$, where $p > 2^k$. $\mathbb{G}, \mathbb{G}_T$ are two multiplicative cyclic groups with the same prime order p;
  - Generate a generator h in $\mathbb{G}$;
  - Let $H : \{0,1\}^* \to \mathbb{G}$ and $H_0 : \{0,1\}^* \to \mathbb{Z}_p^*$ be two collision resistant hash functions.

  The system parameters $SP = \{\mathbb{G}, \mathbb{G}_T, p, \hat{e}, h, H, H_0, N\}$, where N is the maximum dimension of a subspace.
- **KeyExtract:** A specific User-Sensor picks $\alpha \in \mathbb{Z}_p^*$ randomly, then computes $u = h^\alpha \in \mathbb{G}$. The sensor's public-secret key pair is $(PK, SK) = (u, \alpha)$;
- **Sign:** For a specific User-Sensor with the secret key α, the node first divides the collected medical data into m packets, where each packet can be regarded as a m-dimensional vector $\boldsymbol{v}_j$=$(v_{j1}, v_{j2}, \ldots, v_{jN}) \in \mathbb{Z}_p^N, j = 1, 2, \ldots, m$. Let $\boldsymbol{V}$=$\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m\}$ $\subset \mathbb{F}_p^N$ $(0 < m < N)$ denote the subspace with the file identifier $name \in \{0,1\}^*$, where $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m$ is any basis of $\boldsymbol{V}$. Then the node can compute the signature σ_j as the following:
$$\sigma_j = (\Pi_{i=1}^N H(name, i)^{v_{ji}})^\alpha;$$
- **Combine:** Given a specific User-Sensor with the public key PK, a file identifier $name$ and $\{(\beta_j, \sigma_j), \beta_j \in \mathbb{Z}_p, j = 1, \ldots, m\}$, where β_j is the coefficient and σ_j is the signature on vector $\mathbf{v}_j$, then the Combine-Node computes:
$$\sigma = \Pi_{j=1}^m \sigma_j^{\beta_j};$$

- **Verify:** Given a signature σ, vectors $\boldsymbol{v}_j$=$(v_{j1}, v_{j2}, \ldots, v_{jN}) \in \mathbb{Z}_p^N$, coefficients $\beta_j, j = 1, \ldots, m$ and a specific User-Sensor's public key u, compute $\boldsymbol{y} = \sum_{j=1}^{m} \beta_j \boldsymbol{v}_j$, and denote $\boldsymbol{y} = (y_1, \ldots, y_N)$, then check if the equation,

$$\hat{e}(\sigma, h) = \hat{e}(\Pi_{i=1}^{N} H(name, i)^{y_i}, u),$$

holds or not. If it holds, then accept; otherwise, reject.
- **Agg:** Let $U \in \mathfrak{U}$ be the aggregating subset of User-Sensors, and let $|U| = l$. Every User-Sensor with public key u_z generates the signature σ_z on the linear combination $\boldsymbol{y}_z = \sum \beta_{zj} \boldsymbol{v}_{zj}$ $(z = 1, \ldots, l)$, where vectors $\boldsymbol{v}_{z1}, \boldsymbol{v}_{z2}, \ldots, \boldsymbol{v}_{zm_z}$ are a set of the basis of an m_z-dimensional subspace $\boldsymbol{V}_z$ $(z = 1, \ldots, l)$. Moreover, denote X as the public key of a Medical-Server $(X = h^x)$, the designated verifier. The Aggregator-Server computes:

$$\begin{aligned} t &= H_0(\hat{e}(\sigma_1, X), \ldots, \hat{e}(\sigma_l, X)), \\ \sigma &= (\Pi_{z=1}^{l} \sigma_z)^t. \end{aligned}$$

The aggregate signature is σ.
- **AggVerify:** Given an aggregating subset U of l aggregating User-Sensors, the signers, with public key $u_z, z = 1, \ldots, l$, an aggregate signature on the vectors $\boldsymbol{y}_z = \sum \beta_{zj} \boldsymbol{v}_{zj}$ $(z = 1, \ldots, l)$ with file identifiers $name_1, name_2, \ldots, name_l$, where vectors $\boldsymbol{v}_{z1}, \boldsymbol{v}_{z2}, \ldots, \boldsymbol{v}_{zm_z}$ is a set of basis of the m_z-dimensional subspace $\boldsymbol{V}_z$ and $\boldsymbol{v}_{zj} = (v_{zj_1}, v_{zj_2}, \ldots, v_{zj_N})$. The Medical-Server with private key x accepts if the following equation holds:

$$\hat{e}(\sigma, h) = \Pi_{z=1}^{l} \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^{t'}),$$

where

$$\begin{aligned} t' &= H_0(\hat{e}(\Pi_{i=1}^{N} H(name_1, i)^{y_{1i}}, u_1^x), \ldots, \hat{e}(\Pi_{i=1}^{N} H(name_l, i)^{y_{li}}, u_l^x)), \\ \boldsymbol{y}_z &= \sum_{j=1}^{m_z} \beta_{zj} \boldsymbol{v}_{zj}, \ \ \boldsymbol{y}_z = (y_{z1}, y_{z2}, \ldots, y_{zN}), \ \ z = 1, \ldots, l. \end{aligned}$$

Correctness: If the combine signature σ_z is generated on the specific User-Sensor with public key u_z, then the following equations hold for $z = 1, \ldots, l$:

$$\hat{e}(\sigma_z, X) = \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^x).$$

So,

$$\begin{aligned} t &= H_0(\hat{e}(\sigma_1, X), \ldots, \hat{e}(\sigma_l, X)) \\ &= H_0(\hat{e}(\Pi_{i=1}^{N} H(name_1, i)^{y_{1i}}, u_1^x), \ldots, \hat{e}(\Pi_{i=1}^{N} H(name_l, i)^{y_{li}}, u_l^x)) \\ &= t'. \end{aligned}$$

Then,

$$\begin{aligned} \hat{e}(\sigma, h) &= \hat{e}((\Pi_{z=1}^{l} \sigma_z)^t, h) \\ &= \Pi_{z=1}^{l} \hat{e}(\sigma_z, h^t) \\ &= \Pi_{z=1}^{l} \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^{t'}). \end{aligned}$$

5. Security Analysis

5.1. The Security of Basic Signature Scheme

Theorem 1. *Suppose an EUF-CMA adversary $\mathcal{A}$ can break the basic signature scheme with advantage ϵ, and suppose $\mathcal{A}$ can run in time t, make at most q_H times H random oracle queries (on at most q_{id} different identifiers, $q_{id} \leq q_H$) and q_σ times sign queries. Then, there exists a challenger $\mathcal{B}$ to solve the CDH problem with advantage $\epsilon' \geq \frac{1}{q_{id}} \epsilon$ and in time $t' \leq t + (q_{id} \cdot N + q_s \cdot m) \cdot Exp$, where Exp marks the time cost of an exponentiation operation in $\mathbb{G}$.*

Proof. Suppose $\mathcal{A}$ can break the basic scheme's EUF-CMA security, then with inputting a random instance (g, g^a, g^b), the challenger $\mathcal{B}$ can use $\mathcal{A}$ to compute g^{ab}, and solve the CDH problem.

At first, $\mathcal{B}$ generates $(\mathbb{G}, \mathbb{G}_T, p, \hat{e})$, and randomly chooses $\iota \in \mathbb{Z}_p^*$, computes $g_0 = g^\iota, g_1 = g^{a\iota}, g_2 = g^{b\iota}$, then provides $\mathcal{A}$ with $\{\mathbb{G}, \mathbb{G}_T, p, \hat{e}, g_0, PK = g_1\}$. Here, hash function H is considered a random oracle controlled by $\mathcal{B}$.

Then the adversary $\mathcal{A}$ generates an integer N; note that $\mathcal{B}$ generates all $H(name, i), i = 1, \ldots, N$ when it receives an H query on a *name*. $\mathcal{B}$ chooses a random integer index $\lambda \in [1, q_{id}]$, assuming H's λ-th query for a different name is on $name^*$.

$\mathcal{B}$ respectively responds to H queries and sign queries as follows:

H **queries:** $\mathcal{B}$ maintains a list H^{list}: $(name, i, \zeta_i, h_i)$. $\mathcal{A}$ requests the value of $H(name, i)$, $\mathcal{B}$ follows the steps below:

- if the corresponding tuple $(name, i, \zeta_i, h_i)$ already exists in the list, output h_i;
- otherwise,
  - if $name = name^*$, randomly choose $\zeta_i \in \mathbb{Z}_p^*$, and set $h_i = g_2^{\zeta_i}, i = 1, \ldots, N$;
  - else, choose $\zeta_i \in \mathbb{Z}_p^*$ at random, and set $h_i = g_0^{\zeta_i}, i = 1, \ldots, N$;
  - add $(name, i, \zeta_i, h_i)$ to the list, $i = 1, \ldots, N$;
- output h_i.

We suppose $\mathcal{A}$ invariably makes the suitable H queries before others.

Sign queries: $\mathcal{A}$ requests such a query on a subspace $\boldsymbol{V} \subset \mathbb{F}_p^N$, described by basis vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m \in \mathbb{F}_p^N$, where vector $\boldsymbol{v}_j$=$(v_{j1}, v_{j2}, \ldots, v_{jN})$, and $\mathcal{B}$ follows the steps bellow:

- Generate a file identifier $name \in \{0,1\}^*$ randomly. Seek H^{list} or make H queries get $H(name, i)$ and $\zeta_i, i = 1, \ldots, N$.
  - if $name = name^*$, then abort;
  - else, set $\mathbf{s} = (\zeta_1, \ldots, \zeta_N)$, compute $\sigma_j = g_1^{\mathbf{s}\cdot\mathbf{v}_j}$;
- output *name* and $\sigma = (\sigma_1, \ldots, \sigma_m)$.

Forge: Finally, $\mathcal{A}$ outputs a forgery $\sigma^* = (\sigma_1, \ldots, \sigma_m)$ on a vector space $\boldsymbol{V}^* \subset \mathbb{F}_p^N$ described by basis vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m \in \mathbb{F}_p^N$, an identifier *name* and coefficients $\beta_j \in \mathbb{Z}_p, j = 1, \ldots, m$.

- if $name \neq name^*$, then abort;
- else set $\boldsymbol{s} = (\zeta_1, \ldots, \zeta_N)$, compute $\boldsymbol{y} = \sum_{j=1}^m \beta_j \cdot \boldsymbol{v}_j$, if $\boldsymbol{s} \cdot \boldsymbol{y} = 0$, then abort;
- else compute $\theta = (\Pi_{j=1}^m \sigma_j^{\beta_j})^{\frac{1}{\iota} \cdot \frac{1}{\boldsymbol{s}\cdot\boldsymbol{y}}}$, and output θ.

We observe that if the forgery $\sigma^* = (\sigma_1, \ldots, \sigma_m)$ described above is valid, then:

$$\begin{aligned} \hat{e}(\Pi_{j=1}^m \sigma_j^{\beta_j}, g_0) &= \hat{e}(\Pi_{i=1}^N H(name, i)^{y_i}, g_1) \\ &= \hat{e}(\Pi_{i=1}^N h_i^{y_i}, g_1) \\ &= \hat{e}(g_2^{\boldsymbol{s}\cdot\boldsymbol{y}}, g_1) \\ &= \hat{e}(g_2^{\boldsymbol{s}\cdot\boldsymbol{y}}, g_0^a); \end{aligned}$$

furthermore,

$$\Pi_{j=1}^m \sigma_j^{\beta_j} = g_2^{a\cdot\boldsymbol{s}\cdot\boldsymbol{y}},$$

so,

$$\theta = (\Pi_{j=1}^m \sigma_j^{\beta_j})^{\frac{1}{\iota} \cdot \frac{1}{\boldsymbol{s}\cdot\boldsymbol{y}}} = g^{ab}.$$

Analysis. It is easy to get that $\mathcal{B}$ can solve the CDH problem with advantage $\epsilon' \geq \frac{1}{q_{id}}\epsilon$ and in time $t' \leq t + (q_{id} \cdot N + q_s \cdot m) \cdot Exp$. □

5.2. The Security of Aggregate Signature Algorithm

Theorem 2. *Suppose H_0 is a collision-resistant Hash function, then the aggregate signature in the above LHAS scheme is valid, if and only if every individual signature used in the Aggregate algorithm is valid.*

Proof. Suppose each individual signature σ_z involved in the aggregation is valid, where σ_z is the combine signature on the linear combination $\boldsymbol{y}_z = \sum \beta_{zj} \boldsymbol{v}_{zj}$ $(z = 1, \ldots, l)$, where vectors $\boldsymbol{v}_{z1}, \boldsymbol{v}_{z2}, \ldots, \boldsymbol{v}_{zm_z}$ is a set of basis of an m_z-dimensional subspace $\boldsymbol{V}_z$ $(z = 1, \ldots, l)$. Recall that the verifier's public key is X, and x is the corresponding secret key, i.e., $X = h^x$, then

$$\hat{e}(\sigma_z, X) = \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^x), z = 1, \ldots, l.$$

So we have:

$$\begin{aligned} t &= H_0(\hat{e}(\sigma_1, X), \ldots, \hat{e}(\sigma_l, X)) \\ &= H_0(\hat{e}(\Pi_{i=1}^{N} H(name_1, i)^{y_{1i}}, u_1^x), \ldots, \hat{e}(\Pi_{i=1}^{N} H(name_l, i)^{y_{li}}, u_l^x)) \\ &= t', \end{aligned}$$

and

$$\begin{aligned} \hat{e}(\sigma, h) &= \hat{e}((\Pi_{z=1}^{l} \sigma_z)^t, h) \\ &= \Pi_{z=1}^{l} \hat{e}(\sigma_z, h^t) \\ &= \Pi_{z=1}^{l} \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^{t'}). \end{aligned}$$

This means that the aggregate signature is valid.

On the other hand, if the aggregate signature is valid, then:

$$\hat{e}(\sigma, h) = \Pi_{z=1}^{l} \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^{t'}),$$

and

$$\begin{aligned} \hat{e}(\sigma, h) &= \hat{e}((\Pi_{z=1}^{l} \sigma_z)^t, h) \\ &= \Pi_{z=1}^{l} \hat{e}(\sigma_z, h^t) \\ &= \Pi_{z=1}^{l} \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^{t}) \\ &= \Pi_{z=1}^{l} \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^{t'}). \end{aligned}$$

This means that $t = t'$, so we have

$$\begin{aligned} & H_0(\hat{e}(\sigma_1, X), \ldots, \hat{e}(\sigma_l, X)) \\ = \; & H_0(\hat{e}(\Pi_{i=1}^{N} H(name_1, i)^{y_{1i}}, u_1^x), \ldots, \hat{e}(\Pi_{i=1}^{N} H(name_l, i)^{y_{li}}, u_l^x)). \end{aligned}$$

The hash function's collision resistance shows that:

$$\hat{e}(\sigma_z, X) = \hat{e}(\Pi_{i=1}^{N} H(name_z, i)^{y_{zi}}, u_z^x), z = 1, \ldots, l.$$

This guarantees the validity of each single signature σ_z is valid.

The above analysis implies that the aggregate signature in our LHAS scheme is valid, if and only if every single signature used in the algorithm **Agg** is valid. That is to say, even if an adversary can get all signers' *SK* by accessing the relevant oracles, the adversary cannot forge a valid aggregate signature using a set of single signatures which are not all valid. □

6. Performance Analysis

In this section, we mainly conducted various performance analyses of our LHAS scheme with respect to aggregate and un-aggregate. We used the simulator built in Python, performed on the Virtual Machines (VM) with 64-bit Ubuntu 20.04. The hardware and software specifications are shown in Table 1. In order to provide a numerical result, we choose the MNT curve in PBC library. Various notations used in this section are listed in Table 2. We omit the $\mathbb{Z}_p$ hash operation because the cost of this operation is negligible.

Table 1. The hardware and software specification.

CPU	Intel(R) Core(TM) it-4200H CPU @ 2.8 GHz
Operation System	64-bit Ubuntu 20.04
Library	PyCharm, PBC Library, Charm-Crypto Library

Table 2. Definition of notations.

Notation	Definition
Aggregate	aggregate scheme
Un-Aggregate	un-aggregate scheme
$\|m\|$	the overall length of $\{ m_1, m_2, \ldots, m_l \}$
T_M	the computation cost of scalar multiplication calculation in $\mathbb{G}$ or $\mathbb{G}_T$
T_E	the computation cost of exponentiation calculation in $\mathbb{G}$ or $\mathbb{G}_T$
T_P	the computation cost of paring operation in $\mathbb{G}_T$
T_H	the computation cost of map-to-point

We assume that l users participate in the communication of LHAS-EHS. The communication cost comparison of the un-aggregate scheme and aggregate scheme is shown in Table 3. It is clear that our LHAS scheme reduces the $(l-1)|\mathbb{G}|$ communication cost of medical data in the process of data transmission from the Combine-Node to the Medical-Server. Concurrently, it can reduce the $(l-1)|\mathbb{G}|$ storage cost of the Medical-Server. This is because the Aggregator-Server is used to aggregate signatures generated by l users into a short one, so that the Medical-Server only needs to verify one aggregated signature. Therefore, our LHAS scheme can reduce communication cost compared with the un-aggregate signature. This means that our LHAS scheme is efficient for EHS.

Table 3. Performance comparison of communication cost.

	Un-Aggregate	Aggregate
Combine-Node → Aggregator-Server	$l\|\mathbb{G}\| + \|m\|$	$l\|\mathbb{G}\| + \|m\|$
Aggregator-Server→ Medical-Server	$l\|\mathbb{G}\| + \|m\|$	$\|\mathbb{G}\| + \|m\|$

For comparison, we assume that a medical data packet m can be divided into 100 vectors; the dimensional of the vector is 5 and the number of combined vectors is 100. Table 4 and Figure 4 from different directions give the verification efficiency comparison of the Medical-Server, i.e., the un-aggregate scheme and aggregate scheme. In Figure 4, the y-axis represents the time required to verify the signature, and the x-axis represents the number of signatures verified. In the aggregation experiment, we take one Medical-Server as an example. The number of signatures that the Aggregator-Server waits for before aggregation is set to 10, 20, 30, 40, 50, 60, 70, 80, 90, 100. In order to resist coalition attacks, more pairing operations are required in the process of aggregation verification, which leads to more verification overhead. As can be seen from Figure 4, the gap is negligible, and the computing power of the Medical-Server is very strong, which can complete this overhead.

Table 4. Performance comparison of verification efficiency.

	Un-Aggregate	Aggregate
Medical-Server.Verify	$2lT_P+Nl(T_H+T_E)+l(N-1)T_M$	$(2l+1)T_P+((N+1)l+1)T_E+(Nl+1)T_H+l(N-1)T_M$

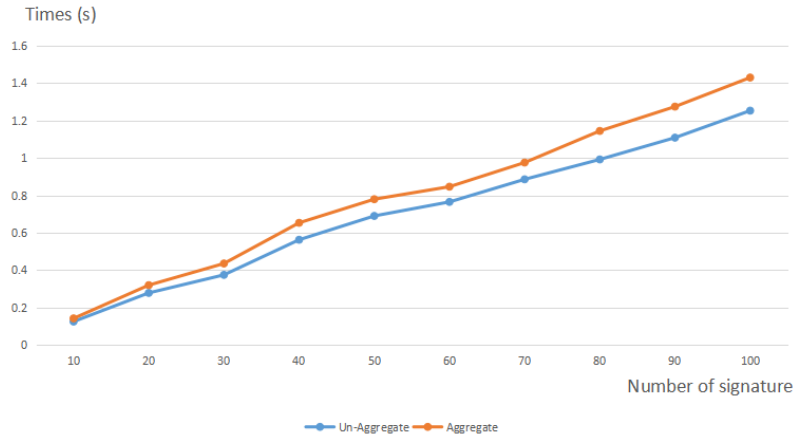

Figure 4. Performance comparison of verification efficiency.

In order to fit the practical application, we chose a larger number of users for the experiment. The number of users l ranges from 100 to 1000 with an interval of 100. We make the y-axis represent the time required to verify the signature, and the x-axis represent the number of users. For the security of the aggregate signature, we have to pay for a certain verification efficiency. As shown in Figure 5, the verification efficiency of un-aggregate scheme is higher than that of aggregate scheme. It can be seen that this overhead gap is not large. Moreover, it also saves storage cost. The experiment results indicate that our proposed LHAS scheme is practical for EHS.

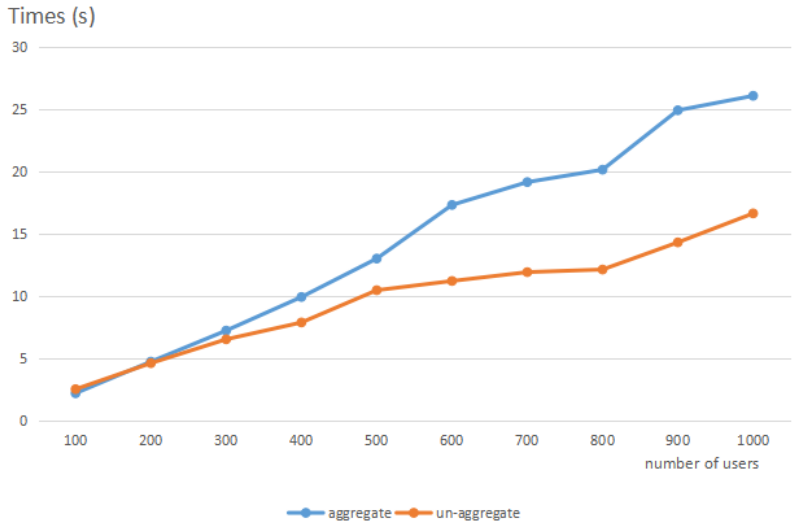

Figure 5. Performance comparison of verification efficiency.

If the coalition attack is not considered, the verification efficiency of the aggregate signature is greatly improved. For the security of the aggregate signature, many pairing operations are required in aggregate verification. This will increase the verification overhead. Since there is no guarantee that the Aggregator-Server is fully trusted in the IoT network, this problem is worth considering. Therefore, how to improve the efficiency of signature verification while resisting coalition attacks is the next problem we need to solve.

7. Conclusions

Firstly, this paper proposes an LHAS-EHS scheme for the electronic healthcare system deployed with IoT technology. This scheme has the characteristics of linear homomorphism and aggregation, and can realize double data compression. It is mainly used for the batch verification of data integrity and data security aggregation. Then, the security of the LHAS scheme is rigorously presented based on the CDH assumption in Theorem 1, and Theorem 2 indicates that our LHAS scheme can resist coalition attacks based on the collision resistance of hash functions, i.e., even if an adversary can get all signers' secret keys, he (she) cannot forge a valid aggregate signature using single signatures that contain invalid ones. Finally, the transmission and verification efficiency of aggregate and un-aggregate schemes are evaluated through performance analysis. The results show that our LHAS-EHS scheme not only reduces the cost of transmission and storage, but also further improves the security. The results show that the proposed LHAS scheme is very suitable for EHS.

Author Contributions: Conceptualization, Y.G. and L.S.; Formal analysis, Y.G. and L.S.; Funding acquisition, L.S., J.X. and F.Z.; Writing—original draft, Y.G.; Writing—review & editing, Y.G., L.S., F.Z. and J.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partially supported by National Natural Science Foundation of China (Grant No. 62172096, 61802195, 61872090) and Natural Science Foundation of Jiangsu Province (Grant No. BK20190696).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Ahlswede, R.; Cai, N.; Li, S.Y.; Yeung, R.W. Network information flow. *IEEE Trans. Inf. Theory* **2000**, *46*, 1204–1216. [CrossRef]
2. Johnson, R.; Molnar, D.; Song, D.; Wagner, D. Homomorphic signature schemes. In Proceedings of the Cryptographers' Track at the RSA Conference, San Jose, CA, USA, 18–22 February 2002; Springer: Berlin/Heidelberg, Germany, 2002; pp. 244–262.
3. Boneh, D.; Freeman, D.; Katz, J.; Waters, B. Signing a linear subspace: Signature schemes for network coding. In Proceedings of the International Workshop on Public Key Cryptography, Irvine, CA, USA, 18–20 March 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 68–87.
4. Zhang, Y.; Jiang, Y.; Li, B.; Zhang, M. An efficient identity-based homomorphic signature scheme for network coding. In Proceedings of the International Conference on Emerging Internetworking, Data & Web Technologies, Wuhan, China, 10–11 June 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 524–531.
5. Wu, B.; Wang, C.; Yao, H. A certificateless linearly homomorphic signature scheme for network coding and its application in the IoT. *Peer-to-Peer Netw. Appl.* **2021**, *14*, 852–872. [CrossRef]
6. Attrapadung, N.; Libert, B. Homomorphic network coding signatures in the standard model. In Proceedings of the International Workshop on Public Key Cryptography, Taormina, Italy, 6–9 March 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 17–34.
7. Cheng, C.; Lee, J.; Jiang, T.; Takagi, T. Security analysis and improvements on two homomorphic authentication schemes for network coding. *IEEE Trans. Inf. Forensics Secur.* **2016**, *11*, 993–1002. [CrossRef]
8. Chang, J.; Ji, Y.; Shao, B.; Xu, M.; Xue, R. Certificateless homomorphic signature scheme for network coding. *IEEE/ACM Trans. Netw.* **2020**, *28*, 2615–2628. [CrossRef]
9. Li, Y.; Zhang, F.; Sun, Y. Lightweight certificateless linearly homomorphic network coding signature scheme for electronic health system. *IET Inf. Secur.* **2021**, *15*, 131–146. [CrossRef]
10. Al-Zubaidie, M.; Zhang, Z.; Zhang, J. REISCH: Incorporating lightweight and reliable algorithms into healthcare applications of WSNs. *Appl. Sci.* **2020**, *10*, 2007. [CrossRef]
11. Du, H.; Wen, Q.; Zhang, S. An efficient certificateless aggregate signature scheme without pairings for healthcare wireless sensor network. *IEEE Access* **2019**, *7*, 42683–42693. [CrossRef]
12. Arshad, K.; Imran, M.A. Wireless Sensing for Human Activity Recognition Using USRP. In Proceedings of the Body Area Networks: Smart IoT and Big Data for Intelligent Health Management: 16th EAI International Conference, BODYNETS 2021, Virtual Event, 25–26 October 2021; Springer Nature: Berlin, Germany, 2022; Volume 420, p. 52.
13. Shen, L.; Ma, J.; Liu, X.; Miao, M. A provably secure aggregate signature scheme for healthcare wireless sensor networks. *J. Med. Syst.* **2016**, *40*, 1–10. [CrossRef]

14. Verma, G.K.; Singh, B.; Singh, H. Bandwidth efficient designated verifier proxy signature scheme for healthcare wireless sensor networks. *Ad Hoc Netw.* **2018**, *81*, 100–108. [CrossRef]
15. Deng, L.; Yang, Y.; Gao, R. Certificateless designated verifier anonymous aggregate signature scheme for healthcare wireless sensor networks. *IEEE Internet Things J.* **2021**, *8*, 8897–8909. [CrossRef]
16. Li, Q.; Zhu, H.; Xiong, J.; Mo, R.; Ying, Z.; Wang, H. Fine-grained multi-authority access control in IoT-enabled mHealth. *Ann. Telecommun.* **2019**, *74*, 389–400. [CrossRef]
17. Li, Q.; Xia, B.; Huang, H.; Zhang, Y.; Zhang, T. TRAC: Traceable and revocable access control scheme for mHealth in 5G-enabled IIoT. *IEEE Trans. Ind. Inform.* **2021**. [CrossRef]
18. Dong, J.; Curtmola, R.; Sethi, R.; Nita-Rotaru, C. Toward secure network coding in wireless networks: Threats and challenges. In Proceedings of the 2008 4th Workshop on Secure Network Protocols, Orlando, FL, USA, 19 October 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 33–38.
19. Wang, F.; Hu, Y.; Wang, B. Lattice-based linearly homomorphic signature scheme over binary field. *Sci. China Inf. Sci.* **2013**, *56*, 1–9. [CrossRef]
20. Boneh, D.; Freeman, D.M. Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures. In Proceedings of the International Workshop on Public Key Cryptography, Taormina, Italy, 6–9 March 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 1–16.
21. Boneh, D.; Gentry, C.; Lynn, B.; Shacham, H. Aggregate and verifiably encrypted signatures from bilinear maps. In Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, 4–8 May 2003; Springer: Berlin/Heidelberg, Germany, 2003; pp. 416–432.
22. Bellare, M.; Namprempre, C.; Neven, G. Unrestricted aggregate signatures. In Proceedings of the International Colloquium on Automata, Languages, and Programming, Wroclaw, Poland, 9–13 July 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 411–422.
23. Wen, Y.; Ma, J.; Huang, H. An aggregate signature scheme with specified verifier. *Chin. J. Electron.* **2011**, *20*, 333–336.
24. Gentry, C.; Ramzan, Z. Identity-based aggregate signatures. In Proceedings of the International Workshop on Public Key Cryptography, New York, NY, USA, 24–26 April 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 257–273.
25. Herranz, J. Deterministic identity-based signatures for partial aggregation. *T Comput. J.* **2006**, *49*, 322–330. [CrossRef]
26. Shim, K.A. An ID-based aggregate signature scheme with constant pairing computations. *J. Syst. Softw.* **2010**, *83*, 1873–1880. [CrossRef]
27. Zhang, L.; Zhang, F. A new certificateless aggregate signature scheme. *Comput. Commun.* **2009**, *32*, 1079–1085. [CrossRef]
28. Liu, J.; Wang, L.; Yu, Y. Improved security of a pairing-free certificateless aggregate signature in healthcare wireless medical sensor networks. *IEEE Internet Things J.* **2020**, *7*, 5256–5266. [CrossRef]
29. Zhao, Y.; Hou, Y.; Wang, L.; Kumari, S.; Khan, M.K.; Xiong, H. An efficient certificateless aggregate signature scheme for the Internet of Vehicles. *Trans. Emerg. Telecommun. Technol.* **2020**, *31*, e3708. [CrossRef]
30. Zuo, W.; Liu, Y. A Provably Secure Certificate-Based Aggregate Signature Scheme. In Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 4 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 2099–2103.
31. Verma, G.K.; Singh, B.; Kumar, N.; Kaiwartya, O.; Obaidat, M.S. PFCBAS: Pairing free and provable certificate-based aggregate signature scheme for the e-healthcare monitoring system. *IEEE Syst. J.* **2019**, *14*, 1704–1715. [CrossRef]
32. Li, T.; Wang, H.; He, D.; Yu, J. Permissioned Blockchain-Based Anonymous and Traceable Aggregate Signature Scheme for Industrial Internet of Things. *IEEE Internet Things J.* **2021**, *8*, 8387–8398. [CrossRef]
33. Zhang, F.; Shen, L.; Wu, G. Notes on the security of certificateless aggregate signature schemes. *Inf. Sci.* **2014**, *287*, 32–37. [CrossRef]
34. Shen, L.; Ma, J.; Liu, X.; Wei, F.; Miao, M. A secure and efficient ID-based aggregate signature scheme for wireless sensor networks. *IEEE Internet Things J.* **2016**, *4*, 546–554. [CrossRef]
35. Shen, L.; Ma, J.; Miao, Y.; Liu, H. Provably secure certificateless aggregate signature scheme with designated verifier in an improved security model. *IET Inf. Secur.* **2019**, *13*, 167–173. [CrossRef]
36. Xie, Y.; Xu, F.; Li, X.; Zhang, S.; Zhang, X.; Israr, M. EIAS: An efficient identity-based aggregate signature scheme for WSNS against coalition attack. *CMC-Comput. Mater. Contin.* **2019**, *59*, 903–924. [CrossRef]
37. Wu, G.; Zhang, F.; Shen, L.; Guo, F.; Susilo, W. Certificateless aggregate signature scheme secure against fully chosen-key attacks. *Inf. Sci.* **2020**, *514*, 288–301. [CrossRef]
38. Zhang, P.; Yu, J.; Wang, T. A homomorphic aggregate signature scheme based on lattice. *Chin. J. Electron.* **2012**, *21*, 701–704.
39. Jing, Z. An efficient homomorphic aggregate signature scheme based on lattice. *Math. Probl. Eng.* **2014**, *2014*. [CrossRef]
40. Boneh, D.; Lynn, B.; Shacham, H. Short signatures from the Weil pairing. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Singapore, 6–10 December 2001; Springer: Berlin/Heidelberg, Germany, 2001; pp. 514–532.