Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8402138, 13 pages
https://doi.org/10.1155/2021/8402138

Research Article

Lightweight Privacy-Preserving Data Sharing Scheme for Internet of Medical Things

Zhuo Zhao,[1] Chingfang Hsu,[1] Lein Harn,[2] Qing Yang,[1] and Lulu Ke[1]

[1]*Computer School, Central China Normal University, Wuhan 430079, China*
[2]*Department of Computer Science Electrical Engineering, University of Missouri-Kansas City, Kansas City, 64110 MO, USA*

Correspondence should be addressed to Chingfang Hsu; cherryjingfang@gmail.com

Received 16 June 2021; Accepted 30 August 2021; Published 13 September 2021

Academic Editor: Pengfei Wang

Internet of Medical Things (IoMT) is a kind of Internet of Things (IoT) that includes patients and medical sensors. Patients can share real-time medical data collected in IoMT with medical professionals. This enables medical professionals to provide patients with efficient medical services. Due to the high efficiency of cloud computing, patients prefer to share gathering medical information using cloud servers. However, sharing medical data on the cloud server will cause security issues, because these data involve the privacy of patients. Although recently many researchers have designed data sharing schemes in medical domain for security purpose, most of them cannot guarantee the anonymity of patients and provide access control for shared health data, and further, they are not lightweight enough for IoMT. Due to these security and efficiency issues, a novel lightweight privacy-preserving data sharing scheme is constructed in this paper for IoMT. This scheme can achieve the anonymity of patients and access control of shared medical data. At the same time, it satisfies all described security features. In addition, this scheme can achieve lightweight computations by using elliptic curve cryptography (ECC), XOR operations, and hash function. Furthermore, performance evaluation demonstrates that the proposed scheme takes less computation cost through comparison with similar solutions. Therefore, it is fairly an attractive solution for efficient and secure data sharing in IoMT.

1. Introduction

Internet of Things (IoT) is a system, which connects different sorts of sensors and computing devices using network to gather and share medical data. IoT lets devices become smarter, processing becomes intelligent, and communication becomes informative [1]. IoT has bred kinds of new technology solutions used in many disparate domains due to its convenience. Certainly, IoT has also penetrated into the healthcare system and has brought great changes. Internet of Medical Things (IoMT) is substantially IoT devices applied to medical industry [2]. The application of IoMT brings lots of conveniences to patients and medical professionals. For example, in IoMT, medical professionals can receive the data and information they need and provide telemedicine for patients anywhere [3].

IoMT provides continuous health monitoring. It relies on different sorts of physiological sensors that are placed on the patients without reducing the user's comfort to collect live health data and information, such as oxygen saturation rate, heart rate, pulse, temperature, blood pressure, and respiration [4–8]. Due to the sensibility of personal health data and information and the limited resources of sensors, it is crucial that security and lightweight computation are included as a fundamental element in IoMT [9]. Cloud computing is a kind of outsourcing platform that has large storage memory and computing resources. Due to its advantages, it can be combined with IoMT to eliminate the issues of storing large data. With the help of cloud computing servers, patients can efficiently store, manage, and share great amount of medical information. By storing data in the cloud, it can be providing easy access for users and improve storage utilization of the health information system [10]. However, the information of the patients (such as the identity of the patients, electronic medical records, and personal condition related to health) is highly private and vulnerable. Data breaches are harmful to patients as

the sensitive information will reveal patients' identity privacy and data security. Hence, the security of health data is the major concern for sharing schemes. Besides, the completeness of shared patients' health data is extremely important [11]. For example, if an adversary tampers patient's conditions related to health, it will mislead medical professionals into making faulty analyses and affect the patient's health. Therefore, integrity verification can prevent tampering by malicious attackers. Moreover, the scheme must provide authentication for users to verify users' legitimacy. This is due to the fact that unauthorized users may tamper with medical records; falsified data will lead to misdiagnosis by medical professionals [12]. Meanwhile, the physiological sensors, used in medical systems, have limited storage memory and power and low computation speed and bandwidth. Accordingly, this motivates us to design a low-cost and lightweight data sharing scheme applied to the IoMT, which consumes less power and meets higher security requirements.

Many researchers have devoted to designing effective data sharing schemes in cloud computing over the past few years. However, some [13–15] are not suitable to be deployed in IoMT system because of the use of bilinear pairings which lack efficiency. These heavy calculations with the high resource constraints are not lightweight enough. Analysis in [16] demonstrates that a bilinear pairing operation has very high computation cost. On the contrary, the computation complexity of elliptic curve cryptography (ECC) is several times smaller than that of pairing operation. This is because in the ECC algorithm, the arithmetic requirements are low, the key size is small, and the operand length is shorter. As a result, based on the previous discussion, ECC is regarded as a better encryption technology for resource-constrained devices.

Hence, for the purpose of ensuring the anonymity of patients, preserving shared data privacy, and improving the computation efficiency of physiological sensors in IoMT, this paper constructs a lightweight privacy-preserving data sharing scheme applied to the IoMT using ECC. In this scheme, after collecting the health data, patients with physiological sensors must encrypt collected health data to prevent personal privacy from leaking. Then, the patient generates a fake identity to protect his identity and achieve anonymity. With the help of a cloud server, health data can be shared with authorized users after uploading by patients. Furthermore, to realize the authorized access, patients should designate the identity set of users. Before accessing the health data, users must authenticate to the cloud server. Users are eligible to access encrypted health data only if their identities and access time are valid. Finally, the main contributions of this paper are summarized below.

(1) A lightweight privacy-preserving data sharing scheme for IoMT using ECC is proposed, which anonymizes the identity of patients and designs authorized access to shared health data

(2) The proposed scheme realizes lightweight computations by ECC, hash, and XOR operations, which does not require heavy computations such as bilinear pairings

(3) The proposed protocol can resist possible attacks and achieve all desired security features, including replay attack, eavesdropping attack, correctness, freshness of encryption key, authentication, anonymity of patient, integrity certification, and forward secrecy of encryption key

(4) Compared with the similar solutions, the proposed scheme satisfies all desired security features and achieves more lightweight computations on patients

The remaining of this paper is adjusted as follows. Previous studies are conducted in Section 2. The basic knowledge of mathematical preliminaries is introduced in Section 3. Then, Section 4 illustrates the model of the proposed scheme including the network model, types of attack, security properties, design goals, and syntax of the proposed scheme. This data sharing scheme including three phases, system initialization, data encryption and upload, and data sharing, is given in Section 5. The security verification of this scheme is provided in Section 6. The performance evaluation and the comparisons with similar schemes in terms of computation cost and security are presented in Section 7. Finally, we culminate conclusions of this paper in Section 8.

2. Related Work

Cloud computing has emerged as a convenient platform of sharing data that enables multiple users from different domains to obtain their needed information simultaneously. It is highly necessary to authenticate users who want to access the health data. However, it worth noting that existing solutions may suffer from a series of issues such as data owner privacy, completeness of the data, data access control, and computation cost in encryption/decryption. These issues have been of widespread concerns.

In 2010, Itani et al. [17] presented a lightweight protocol such that mobile clients can verify the completeness of storage information in mobile cloud computing. In 2013, Wang et al. [18] constructed a cloud storage system that can realize privacy protection, where users can use third-party auditor to verify the completeness of outsourced data. Later, in 2014, Wang et al. [19] presented a novel data integrity verification mechanism using ring signature that is able to ensure identity privacy. Yang et al. [20] designed a data sharing solution in cloud. This solution provided integrity verification while guarantying users' identity privacy. In order to achieve sensitive data concealing in data integrity certificating, Shen et al. [21] presented an efficient data sharing protocol in 2019.

Due to the limited storage of small devices, the large data needs to be outsourced. Outsourced data may contain private information, so ensuring data security has become a challenge. Some works focused on designing valid schemes for this issue. For example, Wang et al. [22] provided a processing mechanism to achieve a flexible user access control. However, this solution takes no account of the energy consumption due to data owner needs to share the pairwise keys with users, which consumes plenty of storage memory. Later, a novel certificateless proxy reencryption (CL-PRE)

scheme was presented by Xu et al. [23], which is used to share information in cloud server securely. This paper showed that the certificateless scheme can cut down the cost of computation and communication for data owners. Nevertheless, this scheme can consume a large amount of computation because of the use of bilinear pairing operation. Khan et al. [24] designed a proxy reencryption scheme for reducing the energy consumption and memory consumption, in which the computational complexity of bilinear pairing still remains. A cloud computing technology-based electronic health record system supporting data privacy preserving was presented in [25]. Ramesh et al. [26] proposed a secure model using e-stream cipher ChaCha20. This model provides integrity verification of sensitive data and guarantees the authenticity of the data. Wang et al. [27] constructed a system framework based on cloud for the electronic medical field. They had utilized identity-based encryption and proxy reencryption in this study for security purpose. This study also provided users authorized by the data owner with the right to access health information. He et al. [28] designed an encryption technology for wireless body area networks to check the completeness of the stored medical data that provides better performance.

A scheme for sharing personal health data and access control was designed by Jiang et al. [29]. This scheme is applied to mobile healthcare social networks, and it adopts attribute-based encryption as the main encryption method. Ding et al. [13] presented a health storage system to resolve data integrity verification, which provides convenience for the patient and physician safety communications. Sowjanya et al. [30] introduced an end-to-end authentication protocol. The protocol reduces the overall complexity due to the use of elliptic curve cryptography (ECC). Zhang et al. [31] presented a practical scheme for cloud-assisted electronic health information systems using identity-based encryption to enable the sensitive data sharing efficiently.

Most of the available schemes are not secure enough. In addition, some of the schemes use complex operations such as bilinear pairing, which make the calculation cost more and are not lightweight enough for IoMT. What is more, the anonymity of patients is often ignored by some schemes. As a result, to guarantee the anonymity of patients and provide access control for shared health data, we design a lightweight privacy-preserving data sharing scheme for IoMT that is based on ECC, hash, and XOR operations.

3. Preliminaries

The work of elliptic curve cryptosystem (ECC) was firstly put forth by Koblitz [32] and Miller [33] individually. ECC is a public key encryption technique. Elliptic curve is a kind of cubic curve over finite fields, which is based on the algebraic structure. ECC with the benefit of lightweight and high security has aroused widespread concerns in modern cryptography. 160-bit ECC key and 1024-bit RSA key can provide equivalent security, which leads to the fact that the encryption key generated by ECC is smaller and more efficient. An elliptic curve E is simply described by the equation $y^2 (\mathrm{mod}\, p) = x^3 + ax + b (\mathrm{mod}\, p)$, where p is a large prime number. In addition, $(4a^3 + 27b^2) \neq 0 (\mathrm{mod}\, p)$ needs to be satisfied in order to exclude singular elliptic curves. Z_p indicates a prime finite field and $a, b, x, y \in Z_p$. Then, we omit $(\mathrm{mod}\, p)$ for the sake of simplicity. The three operations of ECC over G_E are defined below.

(1) Point addition: given two random points, P and Q, on the elliptic curve E, the point R on E represents the addition of these two points. The formula is as follows: $P + Q + R = 0$. Here, R refers to the third point where the line connecting P and Q intersects the elliptic curve. And the point $-R$ is the reflection of point R on the x-axis

(2) Point doubling: it refers to the addition of a point on E with itself. The point Q represents the addition of a point P on the same curve E. The formula is as follows: $2P + Q = 0$. Here, the point $-Q$ is the reflection of point Q (point of intersection of tangent line at P with E) on the x-axis

(3) Scalar point multiplication: it means a point that repeatedly performs point doubling and point addition operations. Let $n \in Z_q^*$ be a positive integer and then $n \bullet P$ is given by $P + P + \cdots + P$ (n times)

There are two hard problems in the elliptic curve domain, which are widely used in designing encryption schemes because there is no probabilistic polynomial time algorithm that can effectively run on computer. The following computational hard problems over ECC [34] have been widely utilized for secure schemes.

Elliptic Curve Discrete Logarithm Problem (*ECDLP*): let $k \in Z_q^*$ be a positive integer, and let $P, Q \in G_1$ be two elliptic curve random points. The ECDLP is to determine k given P and Q, where $P = k \bullet Q$. It is obvious that knowing k and Q is easy to calculate P, but conversely, it is not feasible to calculate k by knowing P and Q, if the prime number q is large.

Elliptic Curve Computational Diffie-Hellman Problem (*ECCDHP*): the ECCDHP is stated as it is difficult for any random instance (B, $c \bullet B$, $d \bullet B$) to compute the value $c \bullet d \bullet B$, where B is the base point of the elliptic curve and $c, d \in Z_q^*$ are two positive integers.

4. Model of the Proposed Scheme

We first design a network model suitable for IoMT and a security model for the data sharing scheme in this section. And then, the types of attack and security properties and illustration of the design goals and the syntax of the proposed scheme are provided.

4.1. Network Model. A network model for IoMT is presented. It consists four types of entities, i.e., a trusted authority (TA), patients, cloud severs (CS), and users. Their relationship in the network model is shown in Figure 1.

(1) Trusted authority (TA): TA acts as a public and private secret generation system and is a fully trusted

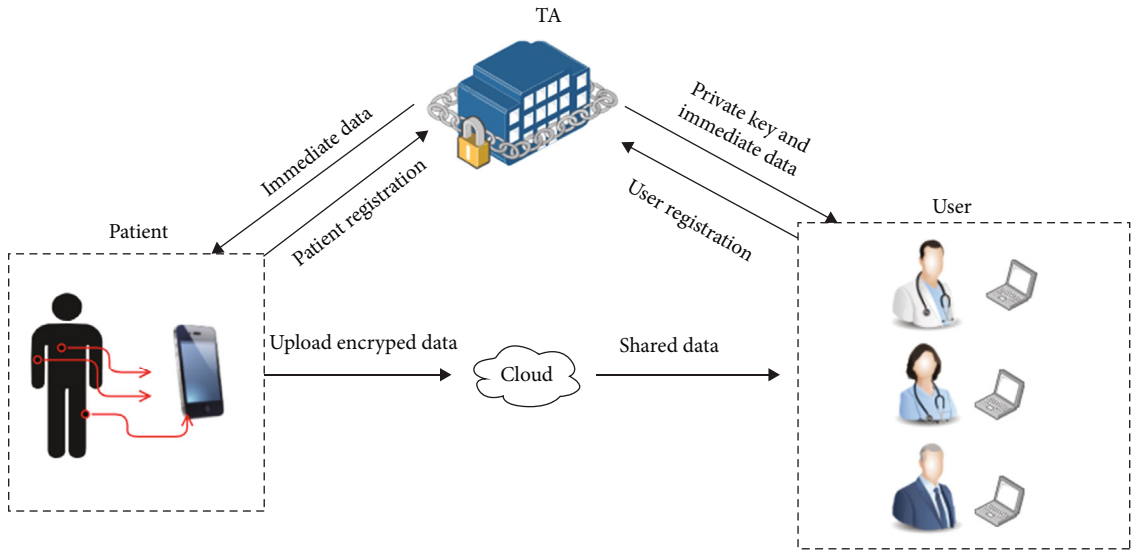

Figure 1: Proposed architecture for IoMT.

authority. In this scheme, system initialization is performed by TA. Patients and users must register with TA before receiving system services. In addition, TA could communicate with different entities via a secure channel. The fact that a secure channel exists does not mean that the data can be shared through the secure channel, due to shared data can be in a large amount

(2) Patient: it refers to data owners with physiological sensors. Patients gather personal health data through these physiological sensors. Patients must register with TA before accepting the service of system. And then, they can upload data to cloud server for storing and sharing health data with authenticated legitimate users due to their own limited memory. Since all shared data is uploaded to cloud server through a public channel, patients should encrypt the gathering information and hide identity to preserve personal privacy and health information security. Besides, his real identity is only known by TA and authorized users

(3) Cloud server (CS): CS is responsible for storing the encrypted information of patients and authenticating users who want to access data because it has a large storage memory and strong computing power. Besides, CS is considered as semitrusted. In other words, if the stored data is lost, it may fake the missing data to hide it from users for economic reasons

(4) User: this entity appertains to medical professional, who can communicate with CS to obtain patients' health information for medical analysis and diagnosis. Before accessing the health data, legitimate users should register with TA. In this scheme, it is important to note that only identified and authorized users can obtain the required health information from CS and decrypt the patients' encrypted data

Now, we will give the description of our proposed scheme. There are three main phases in the proposed sharing scheme, namely, (1) system initialization phase, (2) data encryption and upload phase, and (3) data sharing phase. The subphases of these phases are detailed below.

(1) Setup: trusted authority (TA) executes this phase for defining the system public parameters, choosing a unique nonce $S_{TA} \in Z_q^*$ as its own private key, and computing the public key PK_{TA}, separately

(2) User registration: this phase is processed by the TA. After TA receives the identity Uid sent by user, it generates the warrant of the user warr and private key sk_{Uid}. Further, TA sends (warr, sk_{Uid}) to the user via secret channel

(3) Patient registration: it is performed by the patient and the TA. Firstly, it is run by the patient for generating the temporary identity Ptid and choosing user identity set S and then sends them to TA. Secondly, it is run by the TA for checking the patient's Ptid and computes the intermediate result a_n for data encryption and then sends a_n to the patient via secret channel and S to CS

(4) Encryption: this phase is performed by patients and it encrypts sensitive data M to M'

(5) Upload: it is performed by the patient, by sending the ciphertext M' and related parameters to the CS

(6) User request: it is executed at the user side, by sending request to the CS

(7) Verify integrity: this phase is performed by the user and the CS, for verifying the integrity of the ciphertext M'

(8) Decryption: it is performed by the user, and the cipher text M is decrypted by taking input the ciphertext M' and related parameters

4.2. Security Model. To analyze the security of the proposed data sharing scheme more accurately, we briefly introduce the two types of attacks. Then, we define the required security features and design goals. The detailed security analysis about these security requirements will be described in Section 6.

We consider the following two types of attack.

(1) Replay attack: this attack may repeat the message or delay the message. This can be done by adversary who intercept the message of an old conversation and retransmit it

(2) Eavesdropping attack: it refers to the attacker passively monitoring the communication between users to obtain the transmitted data when the network communications are unsecure

For secure data sharing, the proposed scheme must meet the following security properties.

(1) Correctness: the proposed scheme allows legitimate users to correctly detect whether the information stored in CS is complete. Besides, only authorized users can obtain encrypted data within a valid time and restore the data correctly

(2) Freshness of encryption key: the encryption key generated by the patient in the data encryption and upload phase is only used once. Freshness of encryption key ensures that attackers cannot reuse one encryption key to recover other encrypted sensitive data

(3) Authentication: the purpose of authenticating user is to ensure that, for a given user U, any user N other than U, executing the agreement and impersonating U, CS or TA will not accept the identity of U. The proposed scheme should be required to guarantee that only authorized users designated by the patient himself could access the encrypted health data through CS. And unauthorized users cannot obtain the shared health information. What is more, the authorized users could only access the data for a limited time. The authentication process can prevent user impersonation attack in which attackers act like a legitimate user

(4) Anonymity of patient: since the patient's identity will reveal privacy-sensitive information, it is essential to keep the user's identity confidential. Anonymity means hiding the patient's identity to prevent others from knowing it. In this scheme, the anonymity of patient is ensured if any attackers cannot obtain the real identity Pid of any patient

(5) Integrity certification: the messages transmitted on the public channel can be certificated by the receiver. Besides, any incomplete shared data will be detected by users before decrypting the data. This feature is very important to verify that health data has not been tampered with during transmission and storage process

(6) Forward secrecy of encryption key: the forward secrecy could ensure that past users cannot access the sensitive data uploaded in the future

Furthermore, it is important to propose a solution for security and privacy in IoMT, which should reduce the computational cost and consume few resources. Hence, the security design goals of our data sharing scheme for IoMT should meet the following points.

Privacy preserving: data privacy includes the privacy of the patient's identity and the privacy of shared medical data. The medical data contains electronic medical records and personal condition related to health. If the health information is leaked or accessed by unauthorized adversaries, there is no doubt that it will have a great impact on patients. Hence, it is necessary to guarantee that shared health data is kept confidential from CS and any unauthorized users. Then, this article needs to provide access control for shared data. All users who want to access data need to verify their identity. Any unauthorized users that are not defined by the patient and CS cannot access the encrypted health data. In addition, the proposed scheme needs to anonymize the identity of patients to protect the identity information from being leaked. Consequently, the proposed scheme should provide the anonymity of patient and data access control to ensure the privacy of patient identity and the security of personal health information.

Lightweight operations: the physiological sensors deployed on patients are resource-constrained devices; therefore, the proposed scheme needs to reduce the amount of calculation of patients to improve efficiency of data sharing. To address this issue, we aim to design a lightweight data sharing scheme using ECC. This is because ECC can implement higher security with a small key. Besides, it can also insulate privacy with lower computational complexity as compared to bilinear pairing. Accordingly, this scheme realizes lightweight computations by ECC, hash, and XOR operations.

Effectiveness: in the proposed scheme, it is important to ensure that patients can efficiently share health data with users. Firstly, patients should securely upload health data to CS for sharing with authorized users. Secondly, authorized users should be able to decrypt the required health data for effective medical analysis.

5. Proposed Scheme

For the purpose of privacy protection, we design a secure data sharing scheme for IoMT. This scheme contains the following three phases: (1) system initialization phase, (2) data encryption and upload phase, and (3) data sharing phase. In addition, Table 1 provides the main notations used throughout this paper.

Table 1: Notation table.

No.	Notation	Explanation
1	p	A large prime number
2	E	An elliptic curve of prime order p
3	G_E	An additive elliptic curve group of order q
4	B	Base point of G_E
5	q	Order of G_E
6	O	Point at infinity
7	Z_q	A set with q elements
8	Z_q^*	$Z_q^* = Z_q - \{0\}$
9	h	One-way hash function, $h : \{0,1\} \times G \longrightarrow Z_q^*$
10	S_{TA}	Secret key of trusted authority (TA)
11	$\mathrm{PK}_{\mathrm{TA}}$	Public key of TA
12	warr	Warrant of user
13	Uid, Pid	Identity of user and patient
14	$\mathrm{sk}_{\mathrm{Uid}}$, $\mathrm{sk}_{\mathrm{CS}}$	Secret key of the user and cloud server (CS)
15	Ptid	Temporary identity of patient
16	M	Health data
17	M'	Encrypted data
18	$X \| Y$	Concatenate operation
19	$\oplus$	Bitwise XOR operation
20	$A \longrightarrow B$	Entity A sends the message towards entity B through a public channel

5.1. System Initialization Phase. Firstly, TA generates public parameters and its own secret key. Then, any user in the scheme who wants to access health data should first register with TA. Next, he can obtain his secret key and warrant generated by TA. Like users, patients also need to register with TA before receiving system services. During registration, the patient transfers his temporary identity instead of his real identity via open channel. Hence, the patient's identity information is protected. In addition, the patient needs to define a user identity set. This phase is described in detail below and its process is described in Figure 2.

(1) *Setup*: firstly, TA selects a hash function $h : \{0,1\} \times G \longrightarrow Z_q^*$. Then, TA selects its secret key $S_{\mathrm{TA}} \in Z_q^*$ and CS's secret key $\mathrm{sk}_{\mathrm{CS}} \in Z_q^*$ and calculates its public key according to $\mathrm{PK}_{\mathrm{TA}} = S_{\mathrm{TA}} \bullet B$. TA keeps secret key S_{TA} secretly and publishes public system parameters $\{E, B, h, \mathrm{PK}_{\mathrm{TA}}, G_E\}$. Besides, TA sends $\mathrm{sk}_{\mathrm{CS}}$ to CS via a secure channel

(2) *User registration*: after receiving identity $\mathrm{Uid}_j \in \{0,1\}^*$ from user, TA selects random $r \in Z_q^*$ and computes the private key $\mathrm{sk}_{\mathrm{Uid}} = \mathrm{Uid}_j \cdot r$ for him. Then, TA chooses random $a_1, a_2 \in Z_q^*$ and computes $b_1 = a_1 \cdot B$, $b_2 = a_2 \cdot B$. Then, the warrant of the user is $\mathrm{warr} = a_1 + a_2 \cdot h(\mathrm{Uid}_j \| t_1)$, where t_1 means that authorized users can effectively access shared health information within this time. Next, TA transfers $\mathrm{sk}_{\mathrm{Uid}}$ and warr towards user through a secure channel. Finally, TA computes $E_1 = \mathrm{sk}_{\mathrm{CS}} \bullet h(\mathrm{Uid}_j \| b_1 \| b_2 \| t_1)$ and sends $\{\mathrm{Uid}_j, b_1, b_2, t_1, E_1\}$ to CS. After receiving $\{\mathrm{Uid}_j, b_1, b_2, t_1, E_1\}$, CS computes $E_1' = \mathrm{sk}_{\mathrm{CS}} \bullet h(\mathrm{Uid}_j' \| b_1' \| b_2' \| t_1')$ and checks whether the equation $E_1' = E_1$ holds. If not established, CS terminates this session. On the contrary, CS keeps $\{\mathrm{Uid}_j, b_1, b_2, t_1\}$ locally for the later computation

(3) *Patient registration*: patient Pid first chooses $k \in Z_q^*$ and computes $P_1 = k \cdot B$, $P_2 = k \cdot \mathrm{PK}_{\mathrm{TA}}$, $y_n = h(P_2) \bigoplus \mathrm{Pid}$. Next, the patient defines a set, $S = \langle \mathrm{Uid}_j \rangle_{j=1}^{t}$, which represents a collection of the identities of users who can access his health information. If the identity of user meets $\mathrm{Uid}_j \subseteq S$ and the access time is valid, he can access shared data M. Then, the patient generates a timestamp t_2 and computes his temporary identity $\mathrm{Ptid} = h(\mathrm{Pid} \| P_2 \| S \| t_2)$. After receiving register information $\langle S, P_1, \mathrm{Ptid}, y_n, t_2 \rangle$ from the patient, TA checks the validity of the predicate $(t^* - t_2 (?/<) \Delta t)$, where t^* is the message receiving time and the maximum transmission delay is described by Δt, and aborts if the predicate is not justified. Otherwise, TA calculates $P_2^* = P_1 \cdot S_{\mathrm{TA}}$, $\mathrm{Pid}^* = y_n \oplus h(P_2^*)$, $\mathrm{Ptid}^* = h(\mathrm{Pid}^* \| P_2^* \| S \| t_2)$. After that, TA checks whether the equation $\mathrm{Ptid}^* = \mathrm{Ptid}$ holds. If not, CS drops the

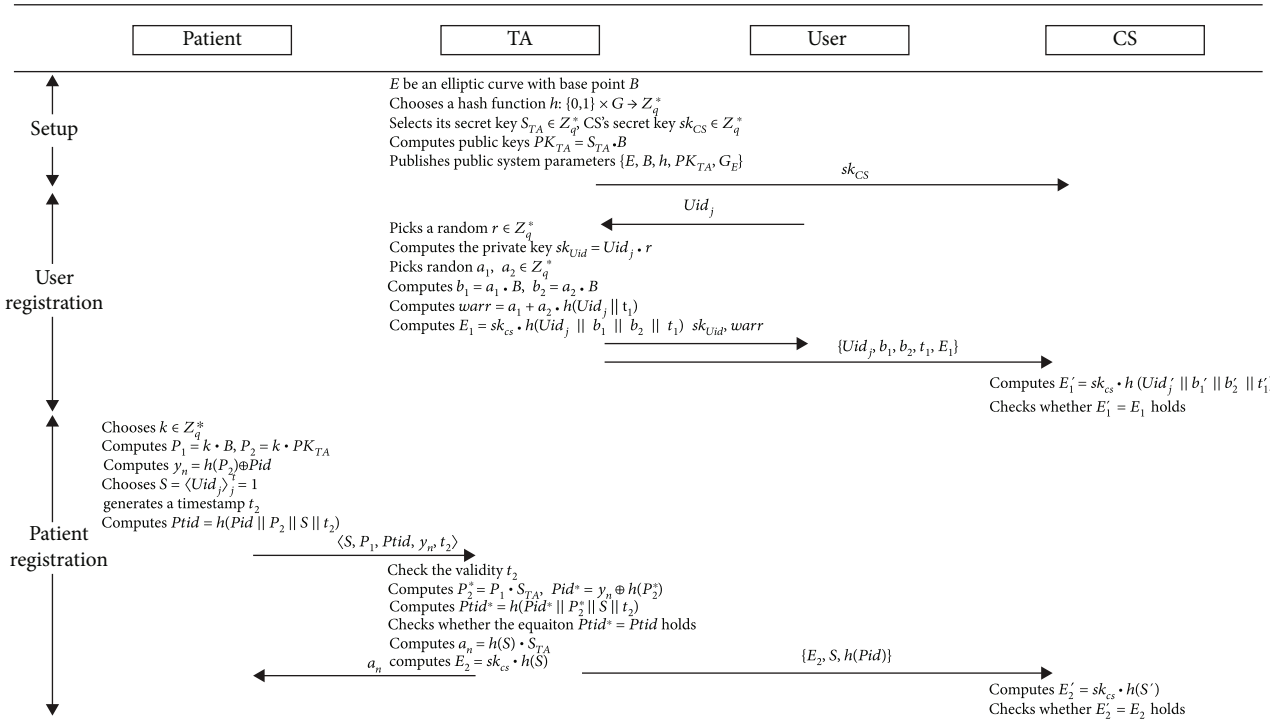

FIGURE 2: System initialization phase.

received message and terminates this session. Otherwise, TA computes $a_n = h(S) \cdot S_{\mathrm{TA}}$ and transfers a_n to the patient via a secure channel. Then, TA computes $E_2 = \mathrm{sk}_{\mathrm{CS}} \bullet h(S)$ and sends $\{E_2, S, h(\mathrm{Pid})\}$ to CS. After receiving $\{E_2, S, h(\mathrm{Pid})\}$, CS computes $E_2' = \mathrm{sk}_{\mathrm{CS}} \bullet h(S')$ and checks whether the equation $E_2' = E_2$ holds. If the equation does not hold, CS terminates the session. On the contrary, CS keeps S locally for the later verification

5.2. Data Encryption and Upload Phase. In this proposed scheme, we are given that the maximum length of shared health data is l. Patient should encrypt data $M \in \{0,1\}^l$ to M' to ensure the privacy of M and then upload M' to CS. This phase is described in detail below and its process is described in Figure 3.

(1) *Encryption*: patient Pid needs to encrypt the gathering data M with a fresh encryption key K. Firstly, the patient randomly chooses random $x, y \in Z_q^*$, and computes $d_1 = a_n \bigoplus x \bigoplus \mathrm{Pid}$, $Y = y \cdot B$, $Z = x \cdot Y$, $\alpha = h(\mathrm{Pid} \| d_1 \| Y)$, $K = h(x \| a_n \| Z \| \mathrm{Pid})$. And then, the patient uses the formula $M' = K \oplus M$ to encrypt M and get ciphertext M'

(2) *Upload*: patient Pid generates a timestamp t_3 and computes $\beta = h(M' \| \mathrm{Pid} \| t_3)$. Then, the patient sends $\langle h(\mathrm{Pid}), Y, d_1, \alpha, \beta, M', t_3 \rangle$ to CS. On receiving this message, CS firstly examines the freshness of the timestamp t_3. If examination is successful, CS stores the information. On the contrary, CS drops this message and terminates this session

5.3. Data Sharing Phase. In order to obtain shared health data, user should verify his identity with TA and CS. He first generates timestamp and forwards related parameters towards CS through public channel. Then, CS will send encrypted data and intermediate parameters to the user if his warrant is valid and his visit time is within the valid time. Next, user verifies that the encrypted data is complete. If verification is successfully done, the user needs to verify himself with TA and obtain the intermediate parameter. If verified successfully, he can download and decrypt M'. This phase is described in detail below and its process is described in Figure 4.

(1) *User request*: user Uid_j first sends his request to CS when he wants to access the shared data M. Then, he generates a timestamp t_4 and transfers $\langle \mathrm{Uid}_j, h(\mathrm{Pid}), t_4, \mathrm{warr} \rangle$ to CS

(2) *Verify integrity*: firstly, CS checks whether Uid_j is in the corresponding set S. If not, CS drops user's requested message and terminates this session. Next, CS checks the validity of the timestamp t_4. Then, CS checks user's warrant with the equation $\mathrm{warr} \cdot B = b_1 + b_2 \cdot h(\mathrm{Uid}_j \| t_1)$. If they are equal, CS sends $\langle \beta, M', t_3 \rangle$ towards the user. After receiving $\langle \beta, M', t_3 \rangle$, the user examines that the data M is complete by computing the equation $\beta = h(M' \| \mathrm{Pid} \| t_3)$. If the

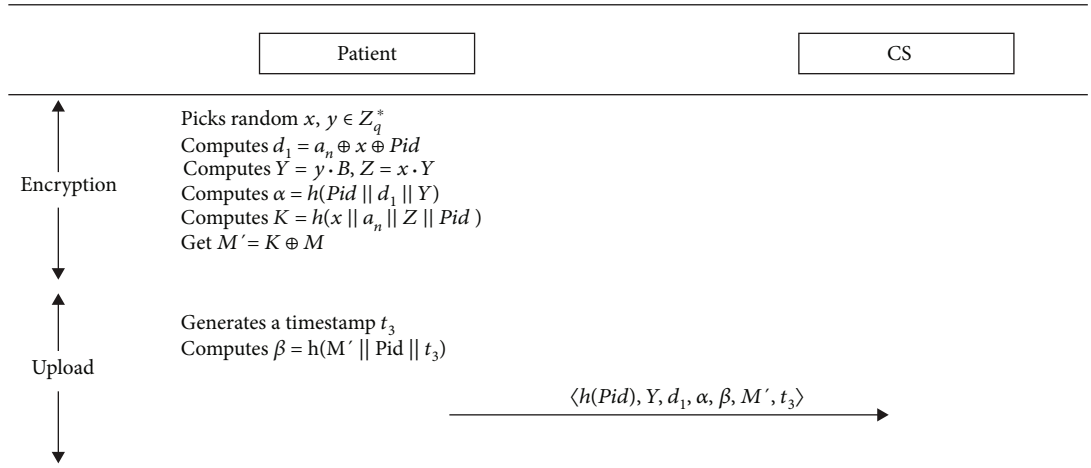

Figure 3: Data encryption and upload phase.

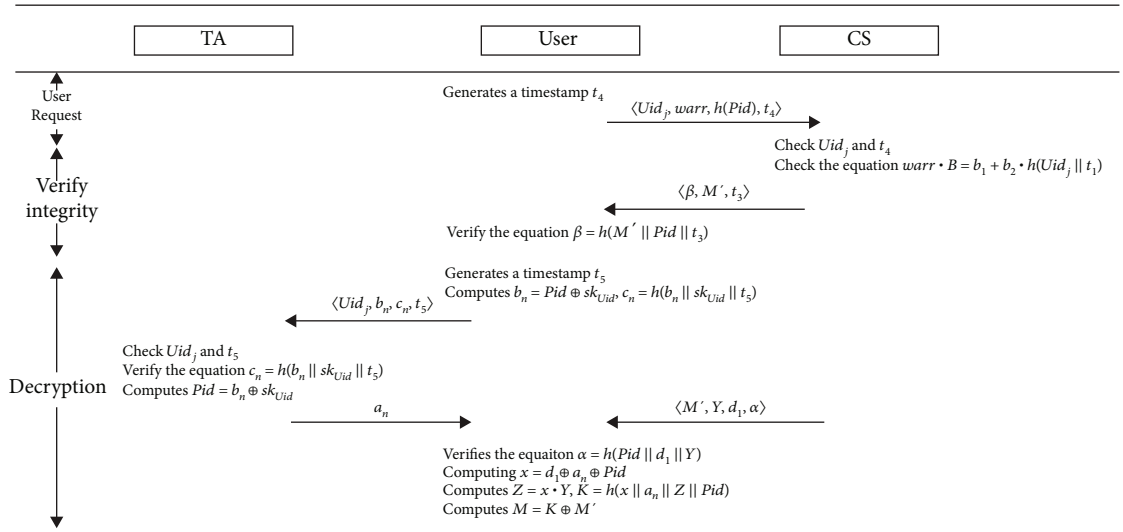

Figure 4: Data sharing phase.

equation is true, the user proceeds to the next step. Otherwise, the user terminates this session

(3) *Decryption*: the user first generates a timestamp t_5, computes $b_n = \text{Pid} \oplus \text{sk}_{\text{Uid}}$, $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$, and sends $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$ to TA in order to obtain intermediate parameters for decrypting. After receiving this message, TA verifies the freshness of the timestamp t_5 and the legitimacy of identity Uid_j. If not, TA drops this message and terminates the session. Otherwise, TA verifies the equation $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$ and computes $\text{Pid} = b_n \oplus \text{sk}_{\text{Uid}}$ and then transfers the a_n of patient Pid to user via a secure channel for decrypting data. After getting $\langle M', Y, d_1, \alpha \rangle$ from CS, the user first verifies the equation $\alpha = h(\text{Pid} \| d_1 \| Y)$. If the equation holds, he retrieves symmetric key K by computing $x = d_1 \oplus a_n \oplus \text{Pid}$, $Z = x \cdot Y$, $K = h(x \| a_n \| Z \| \text{Pid})$. Finally, the user gets the plain text of encrypted data by computing $M = K \oplus M'$

6. Security Analysis

This section analyzes how the proposed scheme can effectively meet the security properties and two types of attack of the proposed scheme presented in Section 4.2.

6.1. Security Properties

(1) Correctness: in the *data sharing phase*, legitimate user verified by CS can correctly examine that the encrypted data is complete, which is stored in CS. After receiving $\langle \beta, M', t_3 \rangle$ from CS, the user first examines the completeness of data M' by computing $h(M' \| \text{Pid} \| t_3)$. The user compares the calculated result with the received value $\beta = h(M' \| \text{Pid} \| t_3)$. Other illegal users cannot fake this authentication response since the secret identity of patient, Pid, is unknown to them. In *data encryption and upload phase*, the correctness of this property is guaranteed

by the one-way nature of hash function. Besides, only authorized users can obtain encrypted data within a valid time and restore the data correctly. Legitimate user can obtain decryption key K by computing the following equations, $x = d_1 \oplus a_n \oplus \text{Pid}$, $Z = x \cdot Y$, and $K = h(x \| a_n \| Z \| \text{Pid})$. Finally, the user computes $M = K \oplus M'$ to recover the plaintext of shared data

(2) Freshness of encryption key: in the *data encryption*, the encryption key, $K = h(x \| a_n \| Z \| \text{Pid})$, is a hash output, where x is a random integer selected by the patient. This key is different in every encryption

(3) Authentication: since the data transmission is carried out on a public channel, it is important to authenticate users who want to access shared information. The authenticity of the user identity is confirmed by TA and CS. In the *data sharing phase*, the user first sends his request $\langle \text{Uid}_j, h(\text{Pid}), t_4, \text{warr} \rangle$ to CS. CS checks whether the Uid_j is in the corresponding set S. If not, TA drops user's requested message and terminates the session. Next, CS checks the validity of the timestamp t_4. Then, CS checks user warrant with the equation $\text{warr} \cdot B = b_1 + b_2 \cdot h(\text{Uid}_j \| t_1)$. If the equation does not hold, CS terminates the session. After the user passes the CS verification, he must also verify with the TA to obtain the intermediate parameter required for decryption. Hence, the user sends $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$ to TA to get intermediate parameter of decryption, where $b_n = \text{Pid} \oplus \text{sk}_{\text{Uid}}$, $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$. On receiving $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$, TA verifies the validity of the timestamp t_5 and Uid_j; if verification is successful, then TA verifies the equation $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$ and computes $\text{Pid} = b_n \oplus \text{sk}_{\text{Uid}}$. Next, TA transfers a_n of patient Pid to user secretly for data decryption. Since the user's private key sk_{Uid} and Pid are secret and are not known by others, no adversary can pretend to be him to authenticate to the TA. Therefore, authentication of the user's identity before obtaining sensitive data ensures more secure communication

(4) Anonymity of patient: the patient transmits messages through a public channel. Because the user's identity Pid is hidden in Ptid or $h(\text{Pid})$, the proposed scheme can guarantee the anonymity of user, as identity of the patient Pid is masked as Ptid or $h(\text{Pid})$. In the *system initialization phase*, the patient transfers his temporary identity, $\text{Ptid} = h(\text{Pid} \| P_2 \| S \| t_2)$ to TA, where $P_2 = k \cdot \text{PK}_{\text{TA}} = P_1 \cdot S_{\text{TA}}$. Besides, in other phases, the patient's identity information is transmitted in the form of hash values, $h(\text{Pid})$. Hence, adversary cannot obtain the real identity Pid of any patient

(5) Integrity certification: to satisfy integrity service, all transmitted messages of the proposed scheme are attached with a verifiable value. In the *system initialization phase*, TA receives the message $\langle S, P_1, \text{Ptid}, y_n, t_2 \rangle$ and checks the integrity of Ptid and S by verifying the timestamp condition $t^* - t_2 < \Delta t$ and verifies $\text{Ptid}^* = \text{Ptid}$ by computing $P_2^* = P_1 \cdot S_{\text{TA}}$, $\text{Pid}^* = y_n \oplus h(P_2^*)$, and $\text{Ptid}^* = h(\text{Pid}^* \| P_2^* \| Sv \| t_2)$. CS receives $\{\text{Uid}_j, b_1, b_2, t_1, E_1\}$ or $\{E_2, S, h(\text{Pid})\}$ and checks the integrity of $\{\text{Uid}_j, b_1, b_2, t_1\}$ or S by computing $E_1' = \text{sk}_{\text{CS}} \bullet h(\text{Uid}_j' \| b_1' \| b_2' \| t_1')$ and checking whether the equation $E_1' = E_1$ holds or by computing $E_2' = \text{sk}_{\text{CS}} \bullet h(S')$ and checking whether the equation $E_2' = E_2$ holds. In the *data sharing phase*, after receiving $\langle \beta, M', t_3 \rangle$, the user verifies that the data M is complete by computing the equation $\beta = h(M' \| \text{Pid} \| t_3)$. During *decryption*, TA receives the message $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$ and checks the integrity of b_n by verifying the timestamp condition $t^* - t_2 < \Delta t$ and verifies the equation $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$. User receives the message $\langle M', Y, d_1, \alpha \rangle$ and checks the integrity of d_1 and Y by verifying the equation $\alpha = h(\text{Pid} \| d_1 \| Y)$. As a result of using Pid and sk_{Uid} over the transmitted messages (which are not known by any adversary), any modification on the data by adversaries is detectable. The proposed scheme takes advantage of the one-way nature of the hash function to ensure that the attacker cannot tamper with the transmitted data

(6) Forward secrecy of encryption key: the disclosure of encryption key K does not influence the security of any past encrypted data. The freshness of the encryption key $K = h(x \| a_n \| Z \| \text{Pid})$ ensures that the proposed scheme meets this feature. The one-way nature of the hash function h prevents all secret parameters from being obtained by attackers. In addition, x, a_n, and Z are all dynamic change with the sessions, where $a_n = h(S) \cdot S_{\text{TA}}$, $Z = x \cdot Y$

6.2. Possible Attacks

Theorem 1 (replay attack). *The proposed scheme can resist the replay attack.*

Proof. The use of timestamp can protect the information transmitted in the proposed scheme from replay attack launched by the adversary. CS and TA can distinguish a replay attack by the examination of the freshness of the timestamp t_i as $t^* - t_i < \Delta t$, where t^* is the current time that the CS or TA gets the message and Δt is the maximum transmission delays. Besides, the use of timestamp t_i ensures the transmitted message cannot be tampered with by an adversary. For example, in the *system initialization phase*, there is an adversary $\mathcal{A}$ and he intercepted a message $\langle S, P_1, \text{Ptid}, y_n, t_2 \rangle$. $\mathcal{A}$ replays message $\langle S', P_1', \text{Ptid}', y_n', t_2' \rangle$. But process will terminate since on receiving $\langle S', P_1', \text{Ptid}', y_n', t_2' \rangle$, TA verifies the freshness of the timestamp

t_2' by computing $t^* - t_2'$ and found that the message $\langle S', P_1', \text{Ptid}', y_n', t_2' \rangle$ is not fresh, as shown in the following equation $t^* - t_2' > \Delta t$. In the *data encryption and upload phase*, $\mathscr{A}$ records message $\langle h(\text{Pid}), Y, d_1, \alpha, \beta, M', t_3 \rangle$. $\mathscr{A}$ initiates a session by transmitting message $\langle h(\text{Pid})', Y', d_1', \alpha', \beta', M'', t_3' \rangle$. But process will terminate since after obtaining the message, CS checks the freshness of the timestamp t_3'. And similarly, in the *data sharing phase*, $\mathscr{A}$ records message $\langle \text{Uid}_j, h(\text{Pid}), t_4, \text{warr} \rangle$ or $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$. $\mathscr{A}$ initiates a session by transmitting message $\langle \text{Uid}_j', h(\text{Pid})', t_4', \text{warr}' \rangle$ or $\langle \text{Uid}_j', b_n', c_n', t_5' \rangle$. But process will terminate since after obtaining the message, CS or TA can detect this message is illegal by verifying the freshness of the timestamp t_4' or the timestamp t_5'. Hence, the proposed scheme stands with the replay attack.□ □

Theorem 2 (eavesdropping attack). *From the intercepted communication parameters, an adversary cannot obtain any secret information.*

Proof. In the data sharing phase of the proposed scheme, an adversary $\mathscr{A}$ can capture the transmitted data by monitoring public channels. He collects the tuple $\langle M', Y, d_1, \alpha \rangle$ from CS to user and the tuple $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$ from user to TA. It is noted that the encryption key $K = h(x \| a_n \| Z \| \text{Pid})$. $\mathscr{A}$ cannot reach x, a_n, Z, and Pid, depending on the intercepted messages. This is due to the parameter x, selected at random by patient, is unknown to $\mathscr{A}$. And since a_n is secretly transmitted by TA to user and patient, no one else knows the value of a_n. The parameter $d_1 = a_n \bigoplus x \bigoplus \text{Pid}$ guarantees that even if $\mathscr{A}$ obtains d_1, he cannot calculate x, Pid, and a_n. The hash function h guarantees that even if $\mathscr{A}$ obtains parameter $\alpha = h(\text{Pid} \| d_1 \| Y)$, he cannot guess the input parameter of h. Besides, $\mathscr{A}$ cannot calculate $Z = x \cdot Y$ because $\mathscr{A}$ does not know x. Finally, the parameter $b_n = \text{Pid} \oplus \text{sk}_{\text{Uid}}$ guarantees that even if $\mathscr{A}$ obtains b_n, he cannot calculate sk_{Uid} and Pid. Therefore, the proposed scheme can protect the encryption key K from being learned by the adversary $\mathscr{A}$, and $\mathscr{A}$ cannot obtain sensitive data from the ciphertext M'. In conclusion, the proposed scheme stands with eavesdropping attack.□ □

7. Performance Analysis

We concretely analyze the performance of the proposed scheme, including computational and communication overheads. Besides, there is a comparison regarding the execution time and security of the proposed scheme and other schemes in [6, 13, 30].

7.1. Computation Cost. The computation cost is analyzed by calculating the operations used in each phase of the scheme. It is noting that the proposed scheme uses t_h, t_{xor}, t_{ecm}, and t_{add} to denote the calculating time needed for the hash function, XOR operation, ECC scalar multiplication, and addition operation, respectively.

7.1.1. System Initialization Phase. In *setup*, TA selects its secret key $S_{TA} \in Z_q^*$ and CS's secret key $\text{sk}_{CS} \in Z_q^*$ and computes $\text{PK}_{TA} = S_{TA} \bullet B$, and the computation overhead is t_{ecm}. In *user registration*, TA first picks a random $r \in Z_q^*$ and computes the private key $\text{sk}_{\text{Uid}} = \text{Uid}_j \cdot r$ for user. Next, TA picks random $a_1, a_2 \in Z_q^*$ and computes $b_1 = a_1 \cdot B$, $b_2 = a_2 \cdot B$. The warrant of user represents as $\text{warr} = a_1 + a_2 \cdot h(\text{Uid}_j \| t_1)$. Then, TA computes $E_1 = \text{sk}_{CS} \bullet h(\text{Uid}_j \| b_1 \| b_2 \| t_1)$. After receiving $\{\text{Uid}_j, b_1, b_2, t_1, E_1\}$, CS computes $E_1' = \text{sk}_{CS} \bullet h(\text{Uid}_j' \| b_1' \| b_2' \| t_1')$.Hence, the calculation cost is $6t_{ecm} + 3t_h + t_{add}$. In *patient registration*, patient Pid first chooses $k \in Z_q^*$ and computes $P_1 = k \cdot B$, $P_2 = k \cdot \text{PK}_{TA}$, $y_n = h(P_2) \bigoplus \text{Pid}$. Next, the patient chooses $S = \langle \text{Uid}_j \rangle_{j=1}^{t}$, generates a timestamp t_2, and computes his temporary identity $\text{Ptid} = h(\text{Pid} \| P_2 \| S \| t_2)$. Then, TA computes $P_2^* = P_1 \cdot S_{TA}$, $\text{Pid}^* = y_n \oplus h(P_2^*)$, $\text{Ptid}^* = h(\text{Pid}^* \| P_2^* \| S \| t_2)$, $a_n = h(S) \cdot S_{TA}$, and $E_2 = \text{sk}_{CS} \bullet h(S)$. After receiving $\{E_2, S, h(\text{Pid})\}$, CS computes $E_2' = \text{sk}_{CS} \bullet h(S')$. Hence, the computation overhead of the algorithm is $6t_{ecm} + 6t_h + 2t_{xor}$.

7.1.2. Data Encryption and Upload Phase. In *encryption*, patient Pid picks random $x, y \in Z_q^*$ and computes $d_1 = a_n \bigoplus x \bigoplus \text{Pid}$, $Y = y \cdot B$, $Z = x \cdot Y$, $\alpha = h(\text{Pid} \| d_1 \| Y)$, and $K = h(x \| a_n \| Z \| \text{Pid})$. Then, the patient uses the formula $M' = K \oplus M$ to encrypt M. Hence, the calculation cost is $2t_{ecm} + 2t_h + 3t_{xor}$. In *upload*, the patient generates a timestamp t_2 and computes $\beta = h(M' \| \text{Pid} \| t_3)$ and the computation overhead is t_h.

7.1.3. Data Sharing Phase. In *user request*, the user generates timestamp t_4 and transfers $\langle \text{Uid}_j, h(\text{Pid}), t_4, \text{warr} \rangle$ to CS. Hence, the computation cost of the algorithm is 0. In *verify integrity*, CS examines user's warrant by computing the formula $\text{warr} \cdot B = b_1 + b_2 \cdot h(\text{Uid}_j \| t_1)$. Next, the user examines the completeness of data M by computing the formula $\beta = h(M' \| \text{Pid} \| t_3)$, so the computation cost of the algorithm is $t_{ecm} + 2t_h + t_{add}$. In *decryption*, the user generates a timestamp t_5, computes $b_n = \text{Pid} \oplus \text{sk}_{\text{Uid}}$, $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$, and sends $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$ to TA. Then, TA verifies the equation $c_n = h(b_n \| \text{sk}_{\text{Uid}} \| t_5)$ and computes $\text{Pid} = b_n \oplus \text{sk}_{\text{Uid}}$. Finally, the user downloads $\langle M', Y, d_1, \alpha \rangle$ from CS and verifies the equation $\alpha = h(\text{Pid} \| d_1 \| Y)$, computing $x = d_1 \oplus a_n \oplus \text{Pid}$, $Z = x \cdot Y$, $K = h(x \| a_n \| Z \| \text{Pid})$, and $M = K \oplus M'$. Hence, the computation overhead of the algorithm is $t_{ecm} + 4t_h + 5t_{xor}$.

The calculation cost of the XOR operation is so small that it can be ignored. Table 2 illustrates the calculated cost of each stage in the proposed scheme.

7.2. Communication Cost. Table 3 lists the communication cost consumed by each transmission. The proposed scheme chooses SHA-1 as hash function, and the SHA-1 outputs a hash digest with length of 160 bits. In addition, we presume the length of elliptic curves $|q| = 160$ bits, the shared data $|M| = 320$ bits, the timestamp $|t_i| = 32$ bits, and the identity

Table 2: Computation cost of the proposed scheme.

Phase	Algorithm	Explanation
System initialization phase	Setup	t_{ecm}
	User registration	$6t_{ecm} + 3t_h + t_{add}$
	Patient registration	$6t_{ecm} + 6t_h + 2t_{xor} \approx 6t_{ecm} + 6t_h$
Data encryption and upload phase	Encryption	$2t_{ecm} + 2t_h + 3t_{xor} \approx 2t_{ecm} + 2t_h$
	Upload	t_h
Data sharing phase	User request	0
	Verify integrity	$t_{ecm} + 2t_h + t_{add}$
	Decryption	$t_{ecm} + 4t_h + 5t_{xor} \approx t_{ecm} + 4t_h$

Table 3: Communication cost of the proposed scheme.

Communication between entities	Communication cost
(User ⟶ TA)	416 bits
(User ⟶ CS)	384 bits
(Patient ⟶ TA)	$512 + 32t$
(Patient ⟶ CS)	1152 bits

$|\text{id}| = 32$ bits. In the transmission (user ⟶ TA), user sends Uid_j during the *system initialization phase* and $\langle \text{Uid}_j, b_n, c_n, t_5 \rangle$ during the *data sharing phase*. The size of these messages is $32 \times 2 + 160 \times 2 + 32 = 416$ bits. In the transmission (user ⟶ CS), user sends the tuple, $\langle \text{Uid}_j, h(\text{Pid}), t_4, \text{warr} \rangle$ of size 384 bits. In the transmission (patient ⟶ TA), the patient sends the tuple, $\langle S, P_1, \text{Ptid}, y_n, t_2 \rangle$ of size $512 + 32t$ bits, where t is the number of user identity to access his health data. In the transmission (patient ⟶ CS), the patient sends the tuple $\langle h(\text{Pid}), Y, d_1, \alpha, \beta, M', t_3 \rangle$ of size 1152 bits.

7.3. Comparisons with Related Schemes. In order to compare several schemes more intuitively, we construct Table 4 according to [7]. Table 4 illustrates the calculation cost of different operations. And we demonstrate the calculation overheads of the proposed scheme and other schemes in [6, 13, 30] according to Table 4. Table 5 summarizes the calculation overheads by patient in the proposed data sharing scheme and other recently proposed schemes. From the comparison in Table 5, the proposed scheme is extremely more lightweight than schemes in [6, 13, 30], because of the executing of ECC, hash, and XOR operations.

According to the data in Table 5, the proposed scheme reduced the computational cost from Ding et al. [13] which is $(1803.8t_h - 295t_h)/(1873.5t_h) = 83.6\%$. Computation cost reduction from Chen and Peng [6] is $(817.5t_h - 295t_h)/(817.5t_h) = 63.91\%$. Computation cost reduction from Sowjanya et al. [30] is $(584t_h - 295t_h)/(584t_h) = 49.49\%$.

The analysis of security features for the proposed scheme in comparison with the scheme of Ding et al. [13], Chen and Peng [6], and Sowjanya et al. [30] is in Table 6. From this table, the schemes in [6, 13] do not meet the anonymity of patients. Besides, Ding et al. [13] do not give the protection against replay attack. Chen and Peng [6] and Sowjanya et al. [30] may suffer from eavesdropping attack. It is clear from the result of the comparison that the proposed scheme is more secure than these similar schemes because it can resist

Table 4: Calculation overheads of different operations with t_h as the time unit.

Symbol	Description	Cost
t_h	SHA-1 hash function	t_h
t_{ecm}	ECC scalar multiplication	$72.5t_h$
t_{exp}	Modular exponentiation	$600t_h$
t_{sym}	Symmetric encryption	t_h
t_{mm}	Modular multiplication	$2.5t_h$
t_{ma}	Modular addition	$0.3t_h$

Table 5: Comparisons of the computation cost by patient.

Schemes	Computation cost by patient
Ding et al. [13]	$3t_{ecp} + t_h + t_{mm} + t_{ma} = 1803.8t_h$
Chen and Peng [6]	$3t_{ecm} + t_{exp} = 817.5t_h$
Sowjanya et al. [30]	$8t_{ecm} + 3t_h + t_{sym} = 584t_h$
Ours	$4t_{ecm} + 5t_h = 295t_h$

Table 6: Comparisons of security features.

Security features	Ding et al. [13]	Chen and Peng [6]	Sowjanya et al. [30]	Ours
F_1	No	Yes	Yes	Yes
F_2	Yes	No	No	Yes
F_3	Yes	Yes	Yes	Yes
F_4	No	No	Yes	Yes
F_5	Yes	Yes	Yes	Yes
F_6	Yes	Yes	Yes	Yes

F_1: resist replay attack; F_2: resist eavesdropping attack; F_3: provide authentication; F_4: provide anonymity of patient; F_5: provide integrity certification; F_6: provide forward security.

the above two kinds of attacks and can meet all desired security features.

In summary, compared with the three similar schemes, it is seen that the proposed scheme can perform less computations and meet more security features. Besides, our scheme provides the anonymity of patient's identity and the authentication of access to shared health data. Thus, the proposed scheme is more lightweight and secure for IoMT.

8. Conclusions

We propose a novel design of lightweight privacy-preserving data sharing scheme for IoMT. The presented scheme can not only provide anonymous feature for patient while achieving the data sharing between patients and users but also ensure that only authorized users designated by the patient himself could access the encrypted health data. Furthermore, this scheme realizes lightweight computations by ECC, hash, and XOR operations. Compared with similar solutions, the proposed scheme can satisfy all desired security features as well as achieve more lightweight computations on both patients and users. It is absolutely attractive for data sharing in IoMT.

Data Availability

The data used to support the findings of this study are included within the article.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Consent

Informed consent was obtained from all individual participants included in the study.

Conflicts of Interest

The authors declare that they have no conflict of interest.

Acknowledgments

This work was partially supported by the National Nature Science Foundation of China (Grant Nos. 61772224, 62172181, and 62072133), the Fundamental Research Funds for the Central Universities (No. CCNU19TS019), the Research Planning Project of National Language Committee (No. YB135-40), and the key projects of Guangxi Natural Science Foundation (no. 2018GXNSFDA281040).

References

[1] P. P. Ray, "A survey on Internet of Things architectures," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 3, pp. 291–319, 2018.

[2] F. Al-Turjman, M. H. Nawaz, and U. D. Ulusar, "Intelligence in the Internet of Medical Things era: a systematic review of current and future trends," *Computer Communications*, vol. 150, pp. 644–660, 2020.

[3] M. M. Islam, A. Rahaman, and M. R. Islam, "Development of smart healthcare monitoring system in IoT environment," *SN Computer Science*, vol. 1, no. 3, pp. 1–11, 2020.

[4] B. Latré, B. Braem, I. Moerman, C. Blondia, and P. Demeester, "A survey on wireless body area networks," *Wireless Networks*, vol. 17, no. 1, pp. 1–18, 2011.

[5] M. Shuai, B. Liu, N. Yu, L. Xiong, and C. Wang, "Efficient and privacy-preserving authentication scheme for wireless body area networks," *Journal of Information Security and Applications*, vol. 52, article 102499, 2020.

[6] R. Chen and D. Peng, "Analysis and improvement of a mutual authentication scheme for wireless body area networks," *Journal of Medical Systems*, vol. 43, no. 2, pp. 1–10, 2019.

[7] X. Li, M. H. Ibrahim, S. Kumari, A. K. Sangaiah, V. Gupta, and K. K. R. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," *Computer Networks*, vol. 129, pp. 429–443, 2017.

[8] Z. Guan, Z. Lv, X. Du, L. Wu, and M. Guizani, "Achieving data utility-privacy tradeoff in Internet of Medical Things: a machine learning approach," *Future Generation Computer Systems*, vol. 98, pp. 60–68, 2019.

[9] M. Armbrust, A. Fox, R. Griffith et al., "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[10] J. Wu, L. Ping, X. Ge, W. Ya, and J. Fu, "Cloud storage as the infrastructure of cloud computing," in *2010 International Conference on Intelligent Computing and Cognitive Informatics*, pp. 380–383, Kuala Lumpur, Malaysia, 2010.

[11] Y. Ming and T. Zhang, "Efficient privacy-preserving access control scheme in electronic health records system," *Sensors*, vol. 18, no. 10, article 3520, 2018.

[12] D. Schröder and H. Schröder, "Verifiable data streaming," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, pp. 953–964, Raleigh North Carolina USA, 2012.

[13] R. Ding, H. Zhong, J. Ma, X. Liu, and J. Ning, "Lightweight privacy-preserving identity-based verifiable IoT-based health storage system," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8393–8405, 2019.

[14] X. Chen, W. Susilo, J. Li et al., "Efficient algorithms for secure outsourcing of bilinear pairings," *Theoretical Computer Science*, vol. 562, pp. 112–121, 2015.

[15] V. Odelu and A. K. Das, "Design of a new CP-ABE with constant-size secret keys for lightweight devices using elliptic curve cryptography," *Security and Communication Networks*, vol. 9, no. 17, 4059 pages, 2016.

[16] S. Ding, C. Li, and H. Li, "A novel efficient pairing-free CP-ABE based on elliptic curve cryptography for IoT," *IEEE Access*, vol. 6, pp. 27336–27345, 2018.

[17] W. Itani, A. Kayssi, and A. Chehab, "Energy-efficient incremental integrity for securing storage in mobile cloud computing," in *2010 International Conference on Energy Aware Computing*, pp. 1-2, Cairo, Egypt, 2010.

[18] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[19] Boyang Wang, Baochun Li, and Hui Li, "Oruta: privacy-preserving public auditing for shared data in the cloud," *IEEE*

Transactions on Cloud Computing, vol. 2, no. 1, pp. 43–56, 2014.

[20] G. Yang, J. Yu, W. Shen, Q. Su, Z. Fu, and R. Hao, "Enabling public auditing for shared data in cloud storage supporting identity privacy and traceability," *Journal of Systems and Software*, vol. 113, pp. 130–139, 2016.

[21] W. Shen, J. Qin, J. Yu, R. Hao, and J. Hu, "Enabling identity-based integrity auditing and data sharing with sensitive information hiding for secure cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 2, pp. 331–346, 2019.

[22] W. Wang, Z. Li, R. Owens, and B. Bhargava, "Secure and efficient access to outsourced data," in *Proceedings of the 2009 ACM workshop on Cloud Computing Security*, pp. 55–66, Chicago Illinois USA, 2009.

[23] L. Xu, X. Wu, and X. Zhang, "CL-PRE: a certificateless proxy re-encryption scheme for secure data sharing with public cloud," in *Proceedings of the 7th ACM symposium on Information, Computer and Communications Security*, pp. 87-88, Seoul Korea, 2012.

[24] A. N. Khan, M. L. M. Kiah, S. A. Madani, M. Ali, A. U. R. Khan, and S. Shamshirband, "Incremental proxy re-encryption scheme for mobile cloud computing environment," *Journal of Supercomputing*, vol. 68, no. 2, pp. 624–651, 2014.

[25] S. K. Nayak and S. Tripathy, "Privacy preserving provable data possession for cloud based electronic health record system," in *2016 IEEE Trustcom/BigDataSE/ISPA*, pp. 860–867, Tianjin, China, 2016.

[26] D. Ramesh, R. Mishra, and D. R. Edla, "Secure data storage in cloud: an e-stream cipher-based secure and dynamic updation policy," *Arabian Journal for Science and Engineering*, vol. 42, no. 2, pp. 873–883, 2017.

[27] X. A. Wang, J. Ma, F. Xhafa, M. Zhang, and X. Luo, "Cost-effective secure E-health cloud system using identity based cryptographic techniques," *Future Generation Computer Systems*, vol. 67, pp. 242–254, 2017.

[28] D. He, S. Zeadally, and L. Wu, "Certificateless public auditing scheme for cloud-assisted wireless body area networks," *IEEE Systems Journal*, vol. 12, no. 1, pp. 64–73, 2018.

[29] S. Jiang, X. Zhu, and L. Wang, "EPPS: efficient and privacy-preserving personal health information sharing in mobile healthcare social networks," *Sensors*, vol. 15, no. 9, pp. 22419–22438, 2015.

[30] K. Sowjanya, M. Dasgupta, and S. Ray, "An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems," *International Journal of Information Security*, vol. 19, no. 1, pp. 129–146, 2020.

[31] X. Zhang, Y. Tang, S. Cao, C. Huang, and S. Zheng, "Enabling identity-based authorized encrypted diagnostic data sharing for cloud-assisted E-health information systems," *Journal of Information Security and Applications*, vol. 54, article 102568, 2020.

[32] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[33] V. S. Miller, "Use of elliptic curves in cryptography," *Advances in Cryptology — CRYPTO '85 Proceedings. CRYPTO 1985*, H. C. Williams, Ed., , pp. 417–426, Springer, Berlin, Heidelberg, 1985.

[34] S. Ray, G. P. Biswas, and M. Dasgupta, "Secure multi-purpose mobile-banking using elliptic curve cryptography," *Wireless Personal Communications*, vol. 90, no. 3, pp. 1331–1354, 2016.