Frontiers of Optoelectronics (2022) 15:29
https://doi.org/10.1007/s12200-022-00028-1

Frontiers of Optoelectronics

RESEARCH ARTICLE

Public auditing for real-time medical sensor data in cloud-assisted HealthIIoT system

Weiping Ye[1,2,3,4] · **Jia Wang**[1,2,3,4] · **Hui Tian**[1,2,3,4] · **Hanyu Quan**[1,3,4]

Received: 24 October 2021 / Accepted: 20 December 2021 / Published online: 29 June 2022

Abstract
With the advancement of industrial internet of things (IIoT), wireless medical sensor networks (WMSNs) have been widely introduced in modern healthcare systems to collect real-time medical data from patients, which is known as HealthIIoT. Considering the limited computing and storage capabilities of lightweight HealthIIoT devices, it is necessary to upload these data to remote cloud servers for storage and maintenance. However, there are still some serious security issues within outsourcing medical sensor data to the cloud. One of the most significant challenges is how to ensure the integrity of these data, which is a prerequisite for providing precise medical diagnosis and treatment. To meet this challenge, we propose a novel and efficient public auditing scheme, which is suitable for cloud-assisted HealthIIoT system. Specifically, to address the contradiction between the high real-time requirement of medical sensor data and the limited computing power of HealthIIoT devices, a new online/offline tag generation algorithm is designed to improve preprocessing efficiency; to protect medical data privacy, a secure hash function is employed to blind the data proof. We formally prove the security of the presented scheme, and evaluate the performance through detailed experimental comparisons with the state-of-the-art ones. The results show that the presented scheme can greatly improve the efficiency of tag generation, while achieving better auditing performance than previous schemes.

Keywords Healthcare industrial internet of things (HealthIIoT) · Medical sensor data · Online/offline signature · Public auditing

1 Introduction

As a fast-growing application of internet of things (IoT) in the industrial sector, industrial IoT (IIoT), which can collect, monitor and deliver valuable information through embedded sensors, has shown enormous potential for improving quality of service in many industries [1, 2]. This is especially true in the field of healthcare. The promising IIoT has played a vital role in promoting the informatization and intelligence of healthcare systems, which is known as healthcare IIoT (HealthIIoT) [2, 3]. The HealthIIoT has been recognized as an important tool for providing real-time and high-quality healthcare services, where wireless medical sensors implanted inside or worn on patients are utilized for collecting health data, such as blood pressure, breathing pattern, heart rate, and so on [4–6]. These crucial health data can be remotely and flexibly accessed by doctors to diagnose the patients' condition in time and conduct further treatment, thereby significantly improving the healthcare services [1–4]. As a key component of HealthIIoT, a wireless medical sensor network (WMSN) has been implemented to assist in containing the spread of COVID-19. Since early diagnosis and isolation are effective and imperative strategies for epidemic prevention and control, WMSN can effectively identity those who exhibit symptoms and are most likely to be infected with the virus, and help the diagnosis system to automatically collect COVID-19 data [7, 8]. HealthIIoT is thus profoundly changing the healthcare industry.

✉ Hui Tian
htian@hqu.edu.cn

1 College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China

2 Wuhan National Laboratory for Optoelectronics, Wuhan 430074, China

3 Xiamen Key Laboratory of Data Security and Blockchain Technology, Xiamen 361021, China

4 Fujian Key Laboratory of Big Data Intelligence and Security, Xiamen 361021, China

In HealthIIoT, it is important to support real-time operation and processing on a large amount of medical data collected by WMSN. Considering the limited computing and storage capabilities of the sensor devices, medical sensor data gleaned in WMSN are generally sent to the cloud for storage and maintenance [9–12]. With the almost infinite computing power and storage resources, cloud computing technology has been widely adopted in modern healthcare systems to provide infrastructures and services to make up for the technical limitations of HealthIIoT in communication, processing, and storage [10–12].

Under this circumstance, the cloud-assisted HealthIIoT system came into being [2, 4, 11, 12], in which WMSN collects vital health data from patients, such as physiologic parameters and motion data; and these medical sensor data are transmitted to patients' mobile terminal for integration and preprocessing; then these preprocessed data will be immediately outsourced to remote cloud servers [2, 4, 11, 12]. Cloud-based medical data provides access anywhere/anytime, through which relevant doctors can deliver efficient, convenient and real-time healthcare services including health monitoring, disease prevention, diagnosis and treatment [12]. In brief, the cloud-assisted HealthIIoT paradigm enhances the ability of collection of vital health data in real-time, enables both patients and doctors to access information and interact in a cost-effective manner, and provides high-quality and efficient healthcare services [2, 4, 11, 12].

The cloud-assisted HealthIIoT system has many advantages, but it still faces some serious security and privacy challenges in practical application [2–4, 13]. One of the essentials is how to ensure the integrity of medical data outsourced to the cloud [14], which may be challenged by the following. First, after uploading medical data to remote cloud servers, the user loses substantial control over these data, which makes the traditional data security solution invalid in the cloud environment [15]. Second, for its own benefit, the cloud service provider (CSP) may conceal the data corruption fact caused by intentional and unintentional hardware and software failures [16]. In addition, frequent data loss accidents in recent years have seriously affected the trust relationship between the CSP and users [17]. Medical data are closely related to patients' privacy and health maintenance. Once they are tampered with or partly lost, it will have a significant impact on medical diagnosis and treatment. Therefore, ensuring the integrity of medical sensor data in the cloud-assisted HealthIIoT system is an essential task.

To ensure the correctness and completeness of outsourced data, cloud auditing, also known as remote data integrity checking (RDIC), has emerged [18–20]. Generally speaking, there are two implementation models for cloud auditing, i.e., private auditing and public auditing. In the former, the verification operation is only performed between the user and CSP, which imposes a heavy computation and communication burden on user and may cause a dispute over the auditing result. To solve these problems, the public auditing model introduces a neutral third-party auditor (TPA) to perform the auditing process on behalf of the user, which can significantly reduce the user's communication and computation costs, as well as provide a credible auditing result. Therefore, the public auditing model is believed to be the right direction of cloud auditing's development [14, 16, 24–34].

With the continuous development of cloud computing, its application scenarios, security and efficiency requirements are becoming more and more diversified. To meet these challenges, a great number of auditing schemes have been proposed, such as dynamic data auditing [23–26], privacy-preserving auditing [27, 28], public auditing for the shared data [29–31], certificateless public auditing [32–34] and online/offline auditing [36, 37].

So far, cloud auditing has made great achievements. However, in the cloud-assisted HealthIIoT system, there are still some vital problems that have not been properly resolved.

The first is the contradiction between the high real-time requirement of medical sensor data and the limited computing power of HealthIIoT devices. Unlike general application scenarios, the HealthIIoT system has high real-time requirement for medical sensor data to provide continuous medical monitoring, timely health examination, prompt diagnosis and adequate treatment. However, most of existing cloud auditing schemes require users to perform expensive computations to preprocess the data before outsourcing them to the cloud, which undoubtedly places a heavy burden on lightweight devices, thereby making traditional auditing schemes inapplicable in the cloud-assisted HealthIIoT environment. Therefore, it is crucial to design an efficient tag generation algorithm for resource-constrained devices to preprocess medical sensor data in real time.

The second is the importance and sensitivity of medical sensor data. Public auditing introduces TPA to perform auditing process on behalf of the user, which can significantly reduce the user's overheads, and provide a credible auditing result. However, there is a risk of data leakage, because TPA may obtain the sampled data content from the proof generated by the CSP during auditing process. Therefore, it is vital for public auditing scheme to protect data privacy from TPA.

To solve the problems mentioned above, this paper presents a novel public auditing scheme for cloud-assisted HealthIIoT system. Specifically, our major contributions in this work can be summarized as follows:

1. We propose an efficient public auditing scheme for cloud-based real-time medical sensor data, which is suitable for cloud-assisted HealthIIoT system that consist of a large number of lightweight devices.
2. To address the contradiction between the high real-time requirement of medical sensor data and the limited computing power of HealthIIoT devices, we design an efficient online/offline tag generation algorithm. Specifically, most of heavy computations are conducted offline while online tag generation only performs lightweight computations, thereby enabling resource-constrained devices to efficiently preprocess medical sensor data in real time before outsourcing them to the cloud.
3. To protect the privacy of medical sensor data, we employ a secure hash function to blind the data proof. According to the preimage resistance of the secure hash function, TPA cannot derive any actual data information from the data proof during the verification phase. Therefore, the presented scheme can take advantage of TPA to perform auditing process, while ensuring that TPA cannot directly or indirectly obtain any actual medical data information during the auditing process.
4. We formally prove the security of the presented scheme, and evaluate the performance by theoretical analyses and experimental comparisons with the state-of-the-art schemes. The results show that the presented scheme can efficiently achieve secure auditing for medical sensor data, and outperform previous schemes in terms of computation and communication costs.

The remainder of this paper is organized as follows. In Sect. 2, we review the related work. Section 3 introduces background and preliminaries. Then, we explain the presented scheme in detail in Sect. 4 and provide the security analysis in Sect. 5. Section 6 gives the performance evaluation of the presented scheme through theoretical analyses and experimental comparisons. Finally, we draw the conclusion of this work in Sect. 7.

2 Related work

The cloud-assisted HealthIIoT system collects patients' health data in real time through WMSN, and employs the cloud computing to store and manage these data, which provide patients and doctors with an open, flexible, and cost-effective platform. Despite these advantages, it still faces some serious security challenges. One of the biggest concerns is how to ensure the correctness and completeness of these cloud-based medical sensor data, because the intact and untampered medical data are a key prerequisite for providing accurate medical diagnosis and treatment.

Cloud auditing, which can effectively and securely verify whether the CSP is honestly and correctly storing the outsourced data, has received extensive attention from both academia and industry. There are two kinds of implementation models for cloud auditing, namely, the private auditing and the public auditing. As one of the earliest private auditing schemes, Juels et al. [21] presented proof of retrievability (PoR) to ensure the data possession in the cloud. In the private auditing, the auditing task is directly performed between the CSP and user, which definitely increases the computation and communication burden on user and makes the auditing result controversial. To address these issues, Ateniese et al. [22] first proposed a public auditing scheme, i.e., provable data possession (PDP), which allows a TPA to implement the auditing on behalf of the user. The PDP scheme achieves public auditing, which greatly reduces the burden on user while providing a more reliable and dependable verification result. Subsequently, a great number of successful cloud auditing schemes have been proposed to meet various novel and distinctive requirements for cloud storage services.

For supporting dynamic data, some auditing schemes introduced different kinds of authenticated data structures to ensure data freshness in addition to verifying data integrity. Erway et al. [23] first introduced the rank-based authenticated skip list to present the dynamic PDP, which sets a general auditing framework for dynamic data. Wang et al. [24] employed the merkle hash tree (MHT) to achieve public auditing for dynamic data, in which the root value of MHT is generated as the verification proof to ensure the latest version. Zhu et al. [25] designed the index-hash table (IHT), which is stored in the TPA instead of CSP, to reduce the computation and communication costs. However, the sequential structure of the IHT is not applicable for update operation such as inserts and deletes. Subsequently, Tian et al. [26] presented a 2-dimension authenticated data structures called dynamic hash table (DHT) to achieve efficient updating performance.

To prevent the TPA from extracting the data content through linear proof combinations, a number of auditing schemes [27, 28] adopted random masking to blind data proof to protect data privacy, which is mainly divided into two kinds of strategies. In the first one [27], the CSP generates a mask number $R = yr$ to blind the data proof M by computing $M' = M + rH(R)$, in which y is a global parameter, r is a number chosen randomly, and H is a hash function. In the other one [28], the TPA first computes a mask number $R = y^r$ with a random number r and a global parameter y, then transmits R to CSP together with the challenge message; the CSP generates the masked data proof of M as $M' = e(u, R)^M$ to respond to the challenge, where e is a bilinear map and u is a global parameter.

With the increasing popularity of cloud collaboration, some auditing schemes for shared data, which can be accessed and processed by various users in a group, have been proposed. In addition to checking data integrity, shared data auditing should further support privacy preservation [29], identity traceability [30], and group dynamics [31]. For example, Wang et al. [29] proposed a public auditing scheme for shared data called Oruta, which uses the ring signature to generate verifiable tags to protect the user's identity privacy. Yang et al. [30] designed an identity-block list (IBL) to record modification information of all data blocks, which can achieve the traceability of data modification. Tian et al. [31] designed a novel lazy-revocation mechanism to ingeniously achieve dynamic management of user groups.

To address certificate management in the traditional public key cryptography and key escrow in identity-based cryptography, Wang et al. [32] first introduced the certificateless signature into a cloud auditing scheme, in which the user's private key included two independent parts that are generated by semi-trusted key generation center and user respectively. He et al. [33] proposed a certificateless public auditing scheme for cloud-assisted wireless body area networks. For the group sharing data under multiple users, Li et al. [34] designed a corresponding certificateless solution to address the user revocation issue.

To achieve efficient cloud auditing for resource-constrained devices, the online/offline signature, which was first proposed by Even et al. [35], has been introduced into PDP schemes, where the verifiable tags are generated in two phases: the offline phase and the online phase. That is, the heavy computations for generating verifiable tags can be performed offline in advance, thereby achieving an efficient online tag generation in real time. Li et al. [36] proposed two privacy-preserving public auditing protocols for lightweight devices using online/offline signatures: the basic one and the improved one. Specifically, the basic protocol is only practical for short data. The improved protocol utilizes the MHT to eliminate this restriction and support the auditing of dynamic data, but the authenticated data structure would incur heavy computation and communication costs. Wang et al. [37] presented a semi-generic online/offline PDP trans-formation framework which is applicable to PDP-related schemes with metadata aggregate ability and public metadata expansibility. However, such a general auditing model was not optimized for specific application scenarios.

Although a great number of successful auditing schemes have been proposed to meet various requirements, they cannot be directly applied in cloud-assisted HealthIIoT system, because of the particularities of medical sensor data, such as the high real-time requirement and the sensitivity of medical data. Thus, in this paper, we are motivated to present a tailored efficient public auditing scheme for medical sensor data in the cloud.

3 Background and preliminaries

3.1 System model

The system model of the presented scheme is shown in Fig. 1, which includes four types of entities, i.e., CSP, Patient (including wireless medical sensors and mobile terminal), Doctor, and TPA.

3.1.1 Cloud service provider (CSP)

An entity with powerful computing resources and storage spaces, providing scalable and on-demand data storage and maintenance services.

3.1.2 Patient

The data owner, includes wireless medical sensors and mobile terminal. *Wireless medical sensors* continuously collect the patient's health data in real time, and periodically transmit medical sensor data to *mobile terminal* that integrates and preprocesses these data. Finally, the patient uploads these preprocessed data to the CSP through the mobile terminal. The patient authorizes TPA to check the correctness and completeness of outsourced data in the cloud.

3.1.3 Doctor

The data user, can access the real-time medical sensor data stored in the cloud. Before utilizing medical cloud data to perform diagnosis and treatment, the doctor authorizes TPA to verify data integrity.

3.1.4 Third party auditor (TPA)

A neutral entity, is authorized to verify the integrity of medical sensor data stored in the cloud on behalf of the patient and doctor.

In the cloud-assisted HealthIIoT system, wireless medical sensor network collects patient's health data and transmits them to a mobile terminal (such as smart phone and smart watch). This lightweight mobile terminal performs the preprocessing of these data, which includes dividing the data into multiple blocks and generating a corresponding tag for each data block. Finally, these preprocessed data are outsourced to remote cloud servers.

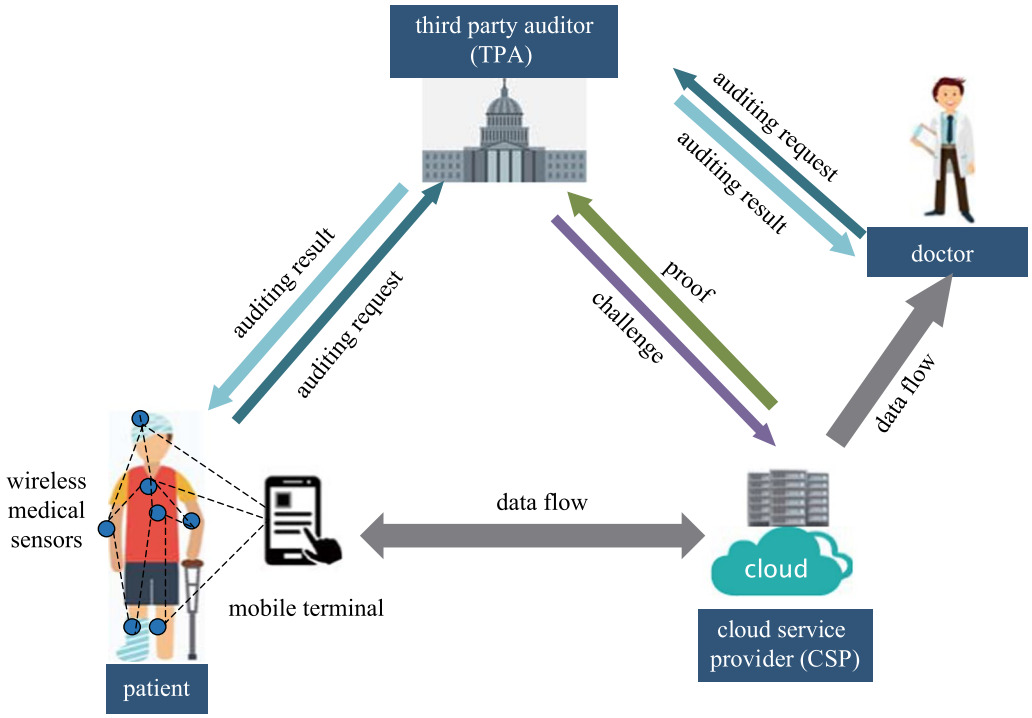

Fig. 1 Public auditing model for medical sensor data in the cloud-assisted HealthIIoT system

Patients enjoy storage and maintenance services by outsourcing medical data to the CSP. However, since patients have lost the substantial control over these data, they may be keen to check the correctness and integrity of their data periodically by authorizing the TPA to perform the auditing process. Meanwhile, before further analyzing cloud-based medical data, doctors need to ensure the data integrity to prevent the data from being tampered with or partially deleted, which will affect the diagnosis and even lead to misdiagnosis.

3.2 Threat model

Usually, the CSP would provide a dependable storage service as requested, but the CSP may cover up the data corruption fact to maintain its own credibility and business interest. In addition, the TPA is assumed to be credible but curious. Specifically, the TPA can perform auditing tasks according to users' requirements, but it may be curious about the content of medical data. The CSP may launch the following attacks to pass the verification performed by the TPA:

- Forging attack: the CSP tries to forge the data blocks and corresponding tags to pass the verification.
- Replacing attack: the CSP attempts to use other data blocks and corresponding tags that are stored well in the cloud as a replacement for damaged data blocks and tags.
- Replaying attack: the CSP tries to use the proof information generated in the previous auditing process to deceive the TPA.

3.3 Design goals

In the presented scheme, we try to achieve the following objectives to effectively support public auditing for real-time medical sensor data in cloud-assisted HealthIIoT system under the above threats:

1. Public auditing: Any TPA authorized by users can verify the integrity of medical sensor data in the cloud.
2. Blockless verification: The TPA does not need to retrieve the whole file to check data integrity.
3. Storage correctness: The CSP that does not correctly store patients' data as required cannot pass the verification.
4. Lightweight: The auditing process should be performed with the minimum communication and computation costs.
5. Data privacy preservation: The TPA is assumed to be credible but curious, so it is necessary to ensure that the

TPA cannot directly or indirectly obtain any actual data information during the auditing process.

6. Batch auditing: the TPA can perform multiple auditing tasks from different patients and doctors simultaneously in a cost-effective manner.

3.4 Preliminaries

3.4.1 Bilinear map

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups with the large prime order p, and g be the generator of $\mathbb{G}_1$. A bilinear map e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ has the following properties:

1. Computability: The bilinear map e should be efficiently computable.
2. Bilinearity: For $\forall\ x, y, l \in \mathbb{G}_1$ and $\forall\ a, b \in \mathbb{Z}_p$, $e(x^a, y^b) = e(x, y)^{ab}$, and $e(x, y \cdot l) = e(x \cdot l, y) = e(x, y) \cdot e(x, l)$.
3. Non-degeneracy: $e(g, g) \neq 1$.

3.4.2 Zhang–Safavi–Susilo signature (ZSS signature)

Based on the bilinear pairings, Zhang et al. [38] proposed an efficient short signature scheme, which requires fewer pairing operations and is more efficient than a Boneh–Lynn–Shacham (BLS) signature [39]. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be the multiplicative cyclic groups of a large prime order q, where P is a generator of $\mathbb{G}_1$. H_1 is a secure hash function with H_1: $\{0, 1\}^* \rightarrow \mathbb{Z}_p^*$, and e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is a bilinear map. ZSS signature consists of the following four algorithms: a parameter generation algorithm *ParamGen*, a key generation algorithm *KeyGen*, a signature generation algorithm *Sign* and a signature verification algorithm *Ver*.

1. *ParamGen*. The system parameters are $\{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H\}$.
2. *KeyGen*. Randomly selects $x \in \mathbb{Z}_p^*$ as the secret key, and computes the public key as $P_{\text{pub}} = xP$.
3. *Sign*. Given a secret key x, and a message m, computes the signature S as follow: $S = \frac{1}{H(m)+x} P$.
4. *Ver*. Given a public key P_{pub}, a message m, and a signature S, verify if $e(H(m)P + P_{\text{pub}}, S) = e(P, P)$.

3.4.3 Computational Diffie–Hellman (CDH) assumption

Let $\mathbb{G}$ be a multiplicative cyclic group with the large prime order q, and P be the generator of $\mathbb{G}$. For unknown $a, b \in \mathbb{Z}_p^*$, given P, aP and bP, it is computationally infeasible to compute abP.

3.4.4 Discrete logarithm (DL) assumption

Let $\mathbb{G}$ be a multiplicative cyclic group of a large prime order q. Given two group elements P and Q, it is computationally infeasible to find an integer $n \in \mathbb{Z}_p^*$ where $Q = nP$.

4 Presented scheme

In the cloud-assisted HealthIIoT system, wireless medical sensors implanted inside or on the patient continuously collect important health data, and periodically transmit medical sensor data to the patient's lightweight mobile terminal such as smart phone and smart watch. After integrating and preprocessing these data, the patient uploads the real-time data and corresponding verifiable tags to remote cloud servers for storage and maintenance. Upon receiving these data, the CSP will analyze and process the medical data in real time. In the meantime, relevant doctors can remotely access the processed medical sensor data to monitor the patient's health status, conduct detailed analysis and perform a quick and efficient diagnosis and treatment.

The correct and complete medical cloud data are a key prerequisite for providing precise medical diagnosis, treatment, and further analysis. Therefore, patients and relevant doctors desire to ensure the correctness and completeness of medical data in the cloud. However, most existing public auditing schemes require users to conduct expensive computations, which is not suitable for lightweight devices. To address the contradiction between the real-time requirement of medical sensor data and the resource-constrained wireless sensor devices and mobile terminal in HealthIIoT, we introduce the online/offline signature mechanism into the public auditing for real-time medical sensor data in the cloud. Specifically, the data tag generation is divided into two phases, that is, phase 1: offline tag generation and phase 2: online tag generation. The offline tag generation is performed by the mobile terminal before it receives medical sensor data and in which most heavy computations are executed. The offline tag generation can be conducted when the mobile terminal is idle or in the middle of a transmission gap from medical sensors. Upon receiving medical sensor data, the mobile terminal only needs to perform lightweight computation with the offline pre-computed results, where the online tag generation can be executed very efficiently. Therefore, the online/offline tag generation mechanism is very applicable for the HealthIIoT system, which greatly reduces the computation burden on the lightweight mobile terminal and meets the high real-time requirement of medical sensor data.

The presented scheme consists of four polynomial-time procedures, i.e., System initialization, Key generation, Tag generation (offline and online phases), and Auditing. Table 1 lists some notations to be used in the presented scheme.

4.1 System initialization

Let l be a prime power, and $E(F_l)$ be an elliptic curve on the finite field F_l. P is a point on $E(F_l)$ of a large prime order q. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be the multiplicative cyclic groups, where P is a generator of $\mathbb{G}_1$. H_1 and H_2 are two secure hash functions with H_1: $\{0, 1\}^* \rightarrow \mathbb{Z}_p^*$, H_2: $\mathbb{G}_1 \rightarrow \mathbb{Z}_p^*$, and e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is a bilinear map. Finally, the system parameters are set as $SP = \{E, l, q, P, \mathbb{G}_1, \mathbb{G}_2, H_1, H_2\}$.

4.2 Key generation

The user first selects a random number $x \in \mathbb{Z}_p^*$ to generate the trapdoor hash function key pair $(Y = xP, x)$. Then, $a \in \mathbb{Z}_p^*$ is randomly chosen as the user's private key sk, i.e., $sk = a$, and the public key is $pk = aP$.

4.3 Tag generation

The tag generation procedure is mainly composed of the following two phases: offline tag generation and online tag generation.

4.3.1 Phase 1 (Offline tag generation)

Without receiving actual medical sensor data from WMSN, the patient randomly generates a series of random numbers w_i and auxiliary parameters r_i, where $(w_i, r_i) \in \mathbb{Z}_p^*$ and $1 \le i \le k$. Meanwhile, (w_i, r_i) are mapped to an element h_i of the group $\mathbb{G}_1$ through a chameleon hash function with the trapdoor key as follow:

$$h_i = w_i P + r_i Y. \tag{1}$$

A random number $fid \in \mathbb{Z}_p^*$ is chosen as the file identifier. Furthermore, the patient generates the offline block tag σ_i using his/her private key sk as follow:

$$\sigma_i = \frac{1}{H_1(fid||i) + H_2(h_i)a} P. \tag{2}$$

Finally, the patient gets the offline block tag pool $\{\sigma_i\}_{1 \le i \le k}$ that will be used in the online phase, where k is the maximum number of data blocks.

4.3.2 Phase 2 (Online tag generation)

Upon receiving the medical data file F, mobile terminal of the patient first divides the file $F \in \{0, 1\}^*$ into n data blocks, namely $F = \{m_1, m_2, \ldots, m_n\}$, where m_i is the data block, $i \in [1,n]$ and $n \le k$. For each data block m_i $(1 \le i \le n)$, the online block tag r_i' is calculated as follow:

$$r_i' = x^{-1}(w_i - m_i) + r_i \bmod q. \tag{3}$$

So far, the patient generates the complete data block tag $\{\sigma_i, r_i'\}_{1 \le i \le n}$. Finally, the preprocessed file $\{fid, F, \{\sigma_i, r_i'\}_{1 \le i \le n}\}$ is uploaded to CSP. It is worth noting that x^{-1} can be calculated in the offline phase to further reduce the online computation burden.

Upon the receipt of these medical sensor data and their corresponding block tags, the CSP first calculates the chameleon hash as follow:

$$h_i' = m_i P + r_i' \cdot Y. \tag{4}$$

It then checks the validity of block tags as follow:

$$e(H_1(fid||i) \cdot P + H_2(h_i') \cdot pk, \sigma_i) = e(P, P). \tag{5}$$

Table 1 Notations

Notation	Description	Notation	Description
$E(F_l)$	An elliptic curve on the finite field F_l	q	A large prime order
$\mathbb{G}_1, \mathbb{G}_2$	Two multiplicative cyclic groups	P	A generator of $\mathbb{G}_1$
H_1	A hash function: $\{0, 1\}^* \rightarrow \mathbb{Z}_p^*$	e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$	A bilinear map
H_2	A hash function: $\mathbb{G}_1 \rightarrow \mathbb{Z}_p^*$	$(Y = xP, x)$	The trapdoor hash function key pair
$sk = a$	The user' private key	$pk = aP$	The user's public key
(w_i, r_i)	A random number w_i and auxiliary parameter r_i	fid	The file identifier
σ_i	The offline block tag	$F = \{m_1, m_2, \ldots, m_n\}$	m_i is the data block of the file F
r_i'	The online block tag	$chal = \{i, v_i \mid i \in L\}$	The challenge information
Ω	The data proof	T	The tag proof

If the equation holds, the CSP accepts, and the patient can delete them locally; otherwise, the CSP rejects their storage and requests the patient to re-upload the correct medical data and tags.

4.4 Auditing

When patients want to check whether their medical data in the cloud are completely and correctly stored, or relevant doctors need to utilize medical sensor data to analyze patients' health condition, they need to ensure the correctness and completeness of medical data in the cloud. The auditing process in the presented scheme includes the following three steps:

Step 1 (Challenge): To check the integrity of cloud-based medical sensor data, patients or the appropriate doctors make an auditing request to the TPA. Upon receiving the auditing request, the TPA randomly selects c data blocks from all data blocks to form a challenge set $L=\{l_1, l_2, \ldots, l_c \mid 1 \le l_i \le n\}$ and chooses a random number $v_i \in \mathbb{Z}_p^*$ for each $i \in L$. Finally, the TPA sends the challenge $chal=\{i, v_i \mid i \in L\}$ to CSP.

Step 2 (Proof generation): Upon receiving the challenge information, the CSP first calculates the chameleon hash value as follow:

$$h_i' = m_i P + r_i' \cdot Y. \tag{6}$$

Then, the data proof Ω and tag proof T are calculated as follows:

$$\Omega = \sum_{i \in L} v_i H_2(h_i'), \tag{7}$$

$$\mathrm{T} = P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i}. \tag{8}$$

In the generation of data proof, we take advantage of the preimage resistance of the secure hash function to protect data privacy in the following verification. Finally, the CSP returns the proof $pf=\{\Omega, \mathrm{T}\}$ to the TPA.

Step 3 (verification): Upon the receipt of the proof $pf=\{\Omega, \mathrm{T}\}$ from the CSP, the TPA checks the data integrity with the following equation:

$$e(\mathrm{T}, P) \cdot e\left(\Omega \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) = e(P, P). \tag{9}$$

If the equation holds, it outputs TRUE; otherwise, it outputs FALSE.

5 Security analysis

In this section, some security analyses concerning the proposed scheme will be presented, including correctness, secure signature, collision resistance of the chameleon hash function, unforgeability of the proof and data privacy preservation.

Theorem 1 (Correctness) *If the CSP is honestly and correctly storing the outsourced data, then the response proof generated by CSP can pass the verification challenged by TPA.*

Proof According to the characteristics of the bilinear map, Eq. (9) in the verification phase of the auditing process can be proven correct as follow:

$$\begin{aligned}
&e(\mathrm{T}, P) \cdot e\left(\Omega \cdot pk + P \sum_{i \in L} H_1(fid||i) v_i, P\right) \\
&= e\left(P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i}, P\right) \cdot e\left(\sum_{i \in L} (v_i H_2(h_i)) \cdot aP + P \sum_{i \in L} H_1(fid||i) v_i, P\right) \\
&= e\left(P - P^2 \sum_{i \in L} v_i (H_1(fid||i) + H_2(h_i)a) \frac{1}{P}, P\right) \cdot e\left(P \sum_{i \in L} v_i (H_2(h_i)a + H_1(fid||i)), P\right) \\
&= e\left(P\left(1 - \sum_{i \in L} v_i (H_1(fid||i) + H_2(h_i)a)\right), P\right) \cdot e\left(P \sum_{i \in L} v_i (H_2(h_i)a + H_1(fid||i)), P\right) \\
&= e(P, P).
\end{aligned}$$

Theorem 2 (Secure signature) *The signature scheme S = <System initialization, Key generation, Tag generation, Auditing> in this work is designed based on ZSS signature* [38]*, which is infeasible for a forger who only knows the public key to produce a valid block-signature pair after obtaining polynomially many signatures on data blocks.*

Proof This theorem follows from ZSS signature [38], where it has been proven it is existentially unforgeable under an adaptive chosen message attack with the assumption that the CDH problem is hard in bilinear groups. The proof can be found in Ref. [38], and is omitted here.

Theorem 3 (Collision resistance) *The chameleon hash function in the presented scheme is collision resistant under the DL assumption.*

Proof Suppose there is a probabilistic polynomial-time algorithm ε which on input a public hash key Y, outputs two distinct pairs of data block and auxiliary parameter (m_i, r_i) and (m_i', r_i') such that $m_i \neq m_i'$ and $m_i \cdot P + r_i \cdot Y = m_i' \cdot P + r_i' \cdot Y$. However, it contradicts the DL assumption, where it is computationally infeasible to compute $Y = xP$. Therefore, the chameleon hash function in the presented scheme is collision resistant.

Theorem 4 (Unforgeability of the proof) *In the presented scheme, it is computationally infeasible for the CSP to forge a valid proof to pass the verification. That is, the presented scheme can effectively resist the forging attack from the CSP.*

Proof Upon receiving the challenge $chal = \{i, v_i | i \in L\}$ from the TPA, the CSP generates the corresponding proof $pf = \{\Omega, \mathrm{T}\}$ that responds to the challenge. We prove the unforgeability of Ω and T respectively as follows:

(1) Unforgeability of Ω

The following game is designed to prove the unforgeability of Ω: the CSP provides forged proof information $pf' = \{\Omega', \mathrm{T}\}$ to respond to the challenge from the TPA, where

$$\Omega = \sum_{i \in L} v_i H_2(h_i') \neq \Omega' = \sum_{i \in L} v_i H_2(h_i''). \tag{10}$$

As Eq. (10) suggests, $\exists i \in L, h_i' \neq h_i''$. If the CSP passes the verification with forged proof information $pf' = \{\Omega', \mathrm{T}\}$, the CSP wins the game; otherwise, the CSP fails.

Assume that the CSP wins the game, then

$$\begin{aligned} & e(\mathrm{T}, P) \cdot e\left(\Omega' \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) \\ & = e(\mathrm{T}, P) \cdot e\left(\sum_{i \in L} v_i H_2(h_i'') \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) \\ & = e(P, P). \end{aligned} \tag{11}$$

The correct proof information is $pf = \{\Omega, \mathrm{T}\}$, so we can get

$$\begin{aligned} & e(\mathrm{T}, P) \cdot e\left(\Omega' \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) \\ & = e(\mathrm{T}, P) \cdot e\left(\sum_{i \in L} v_i H_2(h_i') \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) \\ & = e(P, P). \end{aligned} \tag{12}$$

According to the bilinear mapping described in Sect. 3, we can deduce that $h_i' = h_i''$, $\forall i \in L$, which contradicts the assumption. Therefore, we can conclude that Ω cannot be forged.

(2) Unforgeability of T

We design the following game to prove the unforgeability of T: the CSP provides a forged proof $pf'' = \{\Omega, \mathrm{T}'\}$ to respond to the challenge, where

$$\mathrm{T} = P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i} \neq \mathrm{T}' = P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i'}. \tag{13}$$

As Eq. (13) suggests, $\exists i \in L, \sigma_i \neq \sigma_i'$. If the CSP passes the verification with the forged proof $pf'' = \{\Omega, \mathrm{T}'\}$, the CSP wins the game; otherwise, the CSP fails.

Assume that the CSP wins the game, then,

$$e\left(P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i'}, P\right) \cdot e\left(\Omega \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) = e(P, P). \tag{14}$$

Based on the correct proof $pf = \{\Omega, \mathrm{T}\}$, we have

$$e\left(P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i}, P\right) \cdot e\left(\Omega \cdot pk + \sum_{i \in L} H_1(fid||i) \cdot v_i \cdot P, P\right) = e(P, P). \tag{15}$$

According to the bilinear mapping, we can deduce that $\sigma_i = \sigma_i'$, $\forall i \in L$, which contradicts the assumption. Therefore, we can conclude that T cannot be forged.

In summary, the presented scheme can effectively resist the forging attack.

Theorem 5 (Data privacy preservation) *In the presented scheme, the TPA cannot obtain the specific content of any medical sensor data from the proofs received from the CSP. That is, the TPA is only authorized to verify data integrity and should not learn any actual data information during the auditing process.*

Proof In the proof generation phase of the auditing process, the CSP generates the following proof information $pf = \{\Omega, \mathrm{T}\}$, where Ω is the data proof and T is the tag proof.

$$\Omega = \sum_{i \in L} v_i H_2(h_i'), \tag{16}$$

$$\mathrm{T} = P - P^2 \sum_{i \in L} \frac{v_i}{\sigma_i}. \tag{17}$$

Then the CSP returns proof $pf = \{\Omega, \mathrm{T}\}$ to the TPA.

We can know that T is the aggregate value of the block tags σ_i, which does not contain any content of medical data. Although Ω is the data proof that aggregated by medical data m_i, in which $h'_i = m_i \cdot P + r'_i \cdot Y$. According to the preimage resistance of the secure hash function, the TPA cannot derive any actual data information m_i from Ω during the verification phase.

Therefore, the presented scheme can protect the privacy of medical sensor data from the TPA during auditing process.

6 Performance evaluation

In this section, we make theoretical analyses and evaluate the performance by detailed experiments and comparisons with the state-of-the-art schemes [33, 37].

Table 2 Comparison of communication costs

Schemes	Challenge	Proof generation
CLPA [33]	$c\cdot(\lvert\mathbb{Z}_p^*\rvert + \lvert N\rvert)$	$\lvert\mathbb{G}_1\rvert + \lvert\mathbb{Z}_p^*\rvert$
OOPDP [37]	$c\cdot(\lvert\mathbb{Z}_p^*\rvert + \lvert N\rvert)$	$\lvert\mathbb{G}_1\rvert + (s+1)\cdot\lvert\mathbb{Z}_p^*\rvert$
PAMSD	$c\cdot(\lvert\mathbb{Z}_p^*\rvert + \lvert N\rvert)$	$\lvert\mathbb{G}_1\rvert + \lvert\mathbb{Z}_p^*\rvert$

c is the number of challenged blocks; s is the number of segments in the data block; $|N|$ is the size of the elements in the set $[1, n]$; $|\mathbb{Z}_p^*|$ is the size of the elements in the group $\mathbb{Z}_p^*$; $|\mathbb{G}_1|$ is the size of the elements in the group $\mathbb{G}_1$

6.1 Theoretical analyses

6.1.1 Communication costs

We compare communication costs during the auditing process among the presented scheme (called PAMSD) and state-of-the-art ones (i.e., CLPA [33], OOPDP [37]), which are summarized in Table 2. In the challenge phase, the communication costs of three schemes are the same; all are $c\cdot(|\mathbb{Z}_p^*| + |N|)$. By contrast, in the proof generation phase, the communication costs of OOPDP are related to the number of segments in data block, which are $|\mathbb{G}_1| + (s+1)\cdot|\mathbb{Z}_p^*|$. Therefore, the communication costs of OOPDP are much higher than those of CLPA and PAMSD. Moreover, the communication costs of CLPA and PAMSD are both $|\mathbb{G}_1| + |\mathbb{Z}_p^*|$, but PAMSD employs a secure hash function to blind the data proof to support data privacy preservation in the proof generation, which is crucial to medical data.

In summary, PAMSD is superior to CLPA and OOPDP in terms of communication costs and data privacy preservation.

6.1.2 Computation costs

Table 3 respectively lists the computation costs of the presented scheme and the two comparison schemes in the offline tag generation, online tag generation, proof generation, and verification phases. CLPA does not support online/offline tag generation mechanism, so it can be considered that all tags generation in CLPA are performed online. Therefore, in the offline tag generation phase, we only compare the computation costs of PAMSD with those of OOPDP. As shown in Table 3, the computation costs of PAMSD in the offline tag generation are $n\cdot(Hash_{G1} + 3Mul_{G1} + Invert_{Zp} + Hash_{Zp})$, which are slightly higher than $2n\cdot Exp_{G1}$ of OOPDP. However, in the online tag generation phase, the computation costs of PAMSD are only $n\cdot Mul_{Zp}$, which are much lower than those of either CLPA or OOPDP. Considering the high real-time requirement of medical sensor data, which is critical for the cloud-assisted HealthIIoT system, it is appropriate to exchange slightly higher offline computation costs for online efficiency. It is worth noting that CLPA had the highest computation costs among three schemes, because it performed the entire tag generation online. Therefore, online/offline tag generation mechanism can greatly improve online efficiency.

In the proof generation phase, the computation costs of PAMSD are $(3c+1)\cdot Mul_{G1} + c\cdot(Invert_{G1} + Hash_{G1}) + c\cdot Add_{G1}$, which are slightly higher than those of either CLPA or OOPDP. However, the proof is generated by the CSP with significant

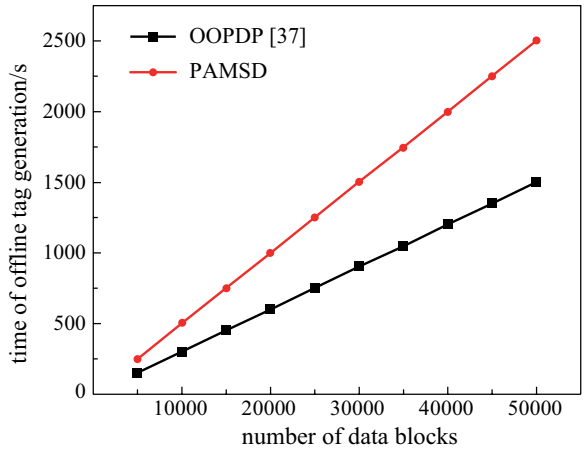

Fig. 2 Computation costs of offline tag generation for blocks in different numbers

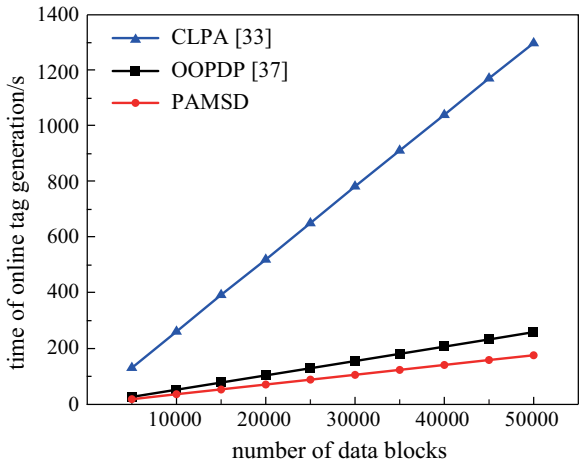

Fig. 3 Computation costs of online tag generation for blocks in different numbers

computing power, which will not affect the overall performance of the auditing process. In fact, cloud servers are supposed to do the heavy computations for users. In the verification phase, the computation costs of PAMSD are $2Pair+Add_{G1}+c{\cdot}Hash_{Zp}+2Mul_{G1}+Mul_{G2}$, which are much lower than those of CLPA and OOPDP.

In summary, in the cloud-assisted HealthIIoT system, it is crucial to perform an efficient real-time data preprocessing before outsourcing data to the cloud. PAMSD designs a new online/offline tag generation mechanism to significantly reduce the online computation costs, which is very suitable for cloud-based real-time medical sensor data.

6.2 Comparative experiments

Detailed comparative experiments are used to evaluate performance. All experiments are performed on a Dell workstation equipped with an Intel Xeon E3-1225 v5 CPU at 3.31 GHz, 8 GB RAM and 7200RPM SATA 2 TB in Linux system (Ubuntu 16.04.2 LTS ×64 with kernel version 4.8.0). All encryption algorithms are implemented in Python environment based on the Pairing Based Cryptography (PBC) library version 0.5.14, with the MNT d159 curve with a length of 160 bits. In addition, all experimental results are the averages of 20 trials.

6.2.1 Computation costs in the tag generation

We separately evaluate the computation costs of tag generation in online and offline phases. In the experiments, the block size is set as 4 KB, and the number of data blocks increases from 5000 to 50000 with intervals of 5000. Since the entire tag generation of CLPA is performed online, it is not included in the offline tag generation comparative experiments. Figures 2 and 3 respectively show the relationship between the offline and online tag generation time and data blocks in different numbers.

The experimental results of offline tag generation, as shown in Fig. 2, suggest that: (1) the computation costs

Table 3 Comparison of computation costs

Schemes	CLPA [33]	OOPDP [37]	PAMSD
Offline tag generation	–	$2n{\cdot}Exp_{G1}$	$n{\cdot}(Hash_{G1}+3Mul_{G1}+Invert_{Zp}+Hash_{Zp})$
Online tag generation	$(n+1){\cdot}Hash_{G1}+2n{\cdot}Mul_{G1}+n{\cdot}Add_{G1}$	$n{\cdot}(s+2){\cdot}Mul_{Zp}+n{\cdot}s{\cdot}Exp_{Zp}+n{\cdot}Hash_{Zp}$	$n{\cdot}Mul_{Zp}$
Proof generation	$c{\cdot}Mul_{G1}+(c-1){\cdot}Add_{G1}$	$(c+s-2){\cdot}Mul_{G1}+(c+1){\cdot}Exp_{G1}$	$(3c+1){\cdot}Mul_{G1}+c{\cdot}(Invert_{G1}+Hash_{G1})+c{\cdot}Add_{G1}$
Verification	$2Pair+(c+3){\cdot}Mul_{G1}+(c+2){\cdot}Add_{G1}+(c+2){\cdot}Hash_{G1}+2Hash_{Zp}$	$3Pair+Invert_{G1}+2Mul_{G1}+3Exp_{G1}+c{\cdot}Hash_{Zp}+Mul_{G2}$	$2Pair+Add_{G1}+c{\cdot}Hash_{Zp}+2Mul_{G1}+Mul_{G2}$

c is the number of challenged blocks; n is the number of data blocks; s is the number of segments in the data block; $Hash_{Zp}$ is the execution time of the hash function H_1: $\{0,1\}^* \rightarrow \mathbb{Z}_p^*$; $Hash_{G1}$ is the execution time of the hash function H_2: $\mathbb{G}_1 \rightarrow \mathbb{Z}_p^*$; Exp_{G1} and Exp_{Zp} are the average time of exponential operation of group $\mathbb{G}_1$ and $\mathbb{Z}_p^*$ respectively; Mul_{G1} and Mul_{G2} are the average time of multiplication operations on group $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively; $Pair$ is the average time to perform the pairing operation; $Invert_{G1}$ and $Invert_{Zp}$ are the time for the inversion operation on group $\mathbb{G}_1$ and $\mathbb{Z}_p^*$; Add_{G1} is the average time of the addition operation on the group $\mathbb{G}_1$

of PAMSD and OOPDP are proportional to the number of data blocks; and (2) to preprocess the same number of data blocks, PAMSD takes more time than OOPDP.

Figure 3 shows the computation costs in the online tag generation, which shows that: (1) The time of online tag generation on all three schemes increases with the number of data blocks. (2) Under the same block number, the computation costs of PAMSD are lower than those of CLPA and OOPDP. (3) CLPA takes much more time than OOPDP and PAMSD to perform the online tag generation.

Since CLPA do not support online/offline tag generation mechanism, its computation costs of online tag generation are much higher than those of either OOPDP or PAMSD. As described in Sect. 4.3, in the offline tag generation phase, PAMSD computes the chameleon hash values with random numbers and auxiliary parameters in advance, then uses these hash values to generate offline tag pool $\{\sigma_i\}_{1 \le i \le k}$. Therefore, the offline computation costs of PAMSD are higher than those of OOPDP.

Considering the high real-time requirement of medical sensor data, we pay more attention to the computing overhead in the online phase that is performed by lightweight mobile terminals. The online tag generation computation costs of the presented scheme are lower than those of either CLPA or OOPDP, which is highly suitable for the cloud-assisted HealthIIoT environment. In other words, the relatively large offline computing overhead will not affect the overall performance of the system.

6.2.2 Computation costs in the verification

To evaluate the computation costs of the verification, in the comparison experiments, the block size and block number are respectively set as 4 KB and 5000, and the number of challenged blocks is increased from 300 to 460 with intervals of 20.

Figure 4 shows the experimental result of the verification time in different numbers of challenge blocks, from which we can learn that: (1) In the three schemes, the verification time of TPA is proportional to the number of challenge blocks, but the growth rate of PAMSD and OOPDP, both of which has a much smaller initial verification time (the number of challenge blocks is equal to 300), is much lower than that of CLPA. (2) The computation costs of PAMSD in the verification phase are lower than those of CLPA and OOPDP.

Compared with CLPA, the online/offline tag generation mechanism greatly improves the verification efficiency. Meanwhile, in comparison with OOPDP, which is also based on the online/offline signature, PAMSD reduces the computation costs of the TPA in the verification phase while improving the efficiency of online tag generation. In a word, PAMSD is superior to both CLPA and OOPDP in verification performance.

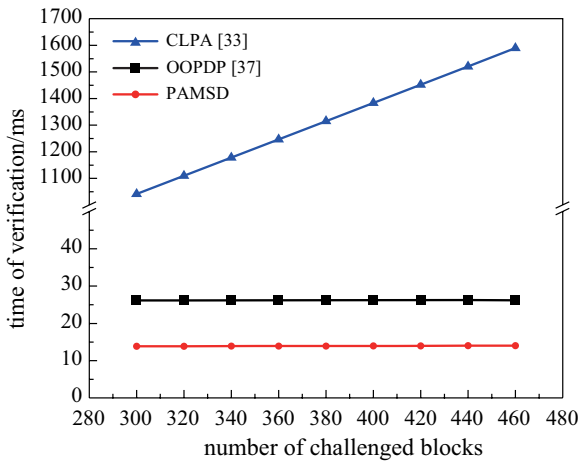

Fig. 4 Verification time in different numbers of challenged block

7 Conclusion

The cloud-assisted HealthIIoT system significantly improves healthcare services, where wireless medical sensors continuously collect real-time medical data concerning patients' vital health parameters, and the flexible access to these cloud-based medical sensor data enables doctors to perform timely medical monitoring and diagnosis. Aiming to address data integrity issues for real-time medical sensor data in the cloud, which is crucial to the cloud-assisted HealthIIoT system, this paper presents an efficient public auditing scheme based on online/offline signature. To address the contradiction between the high real-time requirement of medical sensor data and the limited computing power of HealthIIoT devices, we design a novel online/offline tag generation algorithm. Most of the heavy computations are conducted in the offline phase before receiving medical sensor data to be outsourced, therefore, the online tag generation requires only lightweight preprocessing. Moreover, we employ the secure hash function to blind auditing proof to protect data privacy. We formally prove the security of the presented scheme, and evaluate the performance through theoretical analyses and experimental comparisons with the state-of-the-art ones. The results show that the presented scheme can significantly improve the efficiency of tag generation, while achieving better auditing performance than previous schemes.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (Grant No. U1405254), the Natural Science Foundation of Fujian Province of China (No. 2018J01093), the Open Project Program of Wuhan National Laboratory for Optoelectronics (No. 2018 WNLOKF009), and the Scientific Research Funds of Huaqiao University (No. 605-50Y19028).

Author contributions WY and JW contributed equally to the paper. All authors read and approved the final manuscript.

Declarations

Competing interests The authors declare that they have no competing interests.

References

1. Al-Turjman, F., Alturjman, S.: Context-sensitive access in industrial internet of things (IIoT) healthcare applications. IEEE Trans. Ind. Inf. **14**(6), 2736–2744 (2018)
2. Miao, Y., Tong, Q., Choo, K.K.R., Liu, X., Deng, R.H., Li, H.: Secure online/offline data sharing framework for cloud-assisted industrial internet of things. IEEE Internet Things J. **6**(5), 8681–8691 (2019)
3. Huang, H., Gong, T., Ye, N., Wang, R., Dou, Y.: Private and secured medical data transmission and analysis for wireless sensing healthcare system. IEEE Trans. Ind. Inf. **13**(3), 1227–1237 (2017)
4. Sun, J., Chen, D., Zhang, N., Xu, G., Tang, M., Nie, X., Cao, M.: A privacy-aware and traceable fine-grained data delivery system in cloud-assisted healthcare IIoT. IEEE Internet Things J. **8**(12), 10034–10046 (2021)
5. Hao, J., Jayachandran, M., Kng, P.L., Foo, S.F., Aung Aung, P.W., Cai, Z.: FBG-based smart bed system for healthcare applications. Front. Optoelectron. **3**(1), 78–83 (2010)
6. Zahid, M.N., Jiang, J., Rizvi, S.: Reflectometric and interferometric fiber optic sensor's principles and applications. Front. Optoelectron. **12**(2), 215–226 (2019)
7. Uddin,T., Borhan Uddin, M., Muzahidul Islam, A.K.M., Islam, S., Shatabda, S.: Application of internet of things for early detection of COVID-19 using wearables. In: Proceeding of 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM). pp. 405–410 (2021)
8. Zhang, M., Chu, R., Dong, C., Wei, J., Lu, W., Xiong, N.: Residual learning diagnosis detection: an advanced residual learning diagnosis detection system for COVID-19 in industrial internet of things. IEEE Trans. Ind. Inf. **17**(9), 6510–6518 (2021)
9. Lee, C.C., Chen, S.D., Li, C.T., Cheng, C.L., Lai, Y.M.: Security enhancement on an RFID ownership transfer protocol based on cloud. Futur. Gener. Comput. Syst. **93**, 266–277 (2019)
10. Botta, A., Donato, W., Persico, V., Pescapé, A.: Integration of cloud computing and internet of things: a survey. Futur. Gener. Comput. Syst. **56**, 684–700 (2016)
11. Doukas, C., Maglogiannis, I.: Bringing IoT and cloud computing towards pervasive healthcare. In: Proceeding of 2012 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. pp. 922–926 (2012)
12. Lounis, A., Hadjidj, A., Bouabdallah, A., Challal, Y.: Healing on the cloud: secure cloud architecture for medical wireless sensor networks. Futur. Gener. Comput. Syst. **55**, 266–277 (2016)
13. Lee, C.C.: Security and privacy in wireless sensor networks: advances and challenges. Sensors (Basel) **20**(3), 744 (2020)
14. Zhang, X., Zhao, J., Xu, C., Li, H., Wang, H., Zhang, Y.: CIPPPA: conditional identity privacy-preserving public auditing for cloud-based WBANs against malicious auditors. IEEE Trans. Cloud Comput. **9**(4), 1362–1375 (2021)
15. Brandenburger, M., Cachin, C., Knežević, N.: Don't trust the cloud, verify: integrity and consistency for cloud object stores. ACM Trans. Priv. Secur. **20**(3), 1–30 (2017)
16. Hwang, M.S., Sun, T.H., Lee, C.C.: Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service. J. Circuits Syst. Comput. **26**(5), 1750072 (2017)
17. Yan, H., Li, J., Zhang, Y.: Remote data checking with a designated verifier in cloud storage. IEEE Syst. J. **14**(2), 1788–1797 (2020)
18. Thangavel, M., Varalakshmi, P., Preethi, T., Renganayaki, S., Subhapriya, G.R., Banu A.Z.: A review on public auditing in cloud environment. In: Proceeding of 2016 International Conference on Information Communication and Embedded Systems (ICICES). pp. 1–6 (2016)
19. Tian, H., Chen, Y., Jiang, H., Huang, Y., Nan, F., Chen, Y.: Public auditing for trusted cloud storage services. IEEE Secur. Priv. **17**(1), 10–22 (2019)
20. Jaya Rao, G., Pasupuleti, S.K., Kandukuri, R.: Review of remote data integrity auditing schemes in cloud computing: taxonomy, analysis, and open issues. Proc. Int. J. Cloud Comput. **8**(1), 20–49 (2019)
21. Juels, A., Kaliski, B.S.Jr.: PORs: proofs of retrievability for large files. In: Proceedings of the 14th ACM Conference on Computer and Communications Security. pp. 584–597 (2007)
22. Ateniese, G., Burns, R., Curtmola, R., Herring, J., Kissner, L., Peterson, Z., Song, D.: Provable data possession at untrusted stores. In: Proceedings of the 14th ACM Conference on Computer and Communications Security. pp. 598–609 (2007)
23. Erway, C.C., Küpçü, A., Papamanthou, C., Tamassia, R.: Dynamic provable data possession. In: Proceedings of the 16th ACM Conference on Computer and Communications Security. pp. 213–222 (2009)
24. Wang, Q., Wang, C., Li, J., Ren, K., Lou, W.: Enabling public verifiability and data dynamics for storage security in cloud computing. In: Proceedings of European Symposium on Research in Computer Security. pp. 355–370 (2009)
25. Zhu, Y., Ahn, G., Hu, H., Yau, S.S., An, H.G., Hu, C.J.: Dynamic audit services for outsourced storages in clouds. IEEE Trans. Serv. Comput. **6**(2), 227–238 (2013)
26. Tian, H., Chen, Y., Chang, C.C., Jiang, H., Huang, Y., Chen, Y., Liu, J.: Dynamic-hash-table based public auditing for secure cloud storage. IEEE Trans. Serv. Comput. **10**(5), 701–714 (2017)
27. Wang, C., Chow, S.S.M., Wang, Q., Ren, K., Lou, W.: Privacy-preserving public auditing for secure cloud storage. IEEE Trans. Comput. **62**(2), 362–375 (2013)
28. Yang, K., Jia, X.: An efficient and secure dynamic auditing protocol for data storage in cloud computing. IEEE Trans. Parallel Distrib. Syst. **24**(9), 1717–1726 (2013)
29. Wang, B., Li, B., Li, H.: Oruta: privacy-preserving public auditing for shared data in the cloud. IEEE Trans. Cloud Comput. **2**(1), 43–56 (2014)
30. Yang, G., Yu, J., Shen, W., Su, Q., Fu, Z., Hao, R.: Enabling public auditing for shared data in cloud storage supporting identity privacy and traceability. J. Syst. Softw. **113**, 130–139 (2016)
31. Tian, H., Nan, F., Jiang, H., Chang, C.C., Ning, J., Huang, Y.: Public auditing for shared cloud data with efficient and secure group management. Inf. Sci. **472**, 107–125 (2019)

32. Wang, B., Li, B., Li, H., Li, F.: Certificateless public auditing for data integrity in the cloud. In: Proceedings of 2013 IEEE Conference on Communications and Network Security (CNS). pp. 136–144 (2013)
33. He, D., Zeadally, S., Wu, L.: Certificateless public auditing scheme for cloud-assisted wireless body area networks. IEEE Syst. J. **12**(1), 64–73 (2018)
34. Li, J., Yan, H., Zhang, Y.: Certificateless public integrity checking of group shared data on cloud storage. IEEE Trans. Serv. Comput. **14**(1), 71–81 (2021)
35. Even, S., Goldreich, O., Micali, S.: On-line/off-line digital signatures. J. Cryptol. **9**(1), 35–67 (1996)
36. Li, J., Zhang, L., Liu, J.K., Qian, H., Dong, Z.: Privacy-preserving public auditing protocol for low-performance end devices in cloud. IEEE Trans. Inf. Forensics Secur. **11**(11), 2572–2583 (2016)
37. Wang, Y., Wu, Q., Qin, B., Tang, S., Susilo, W.: Online/offline provable data possession. IEEE Trans. Inf. Forensics Secur. **12**(5), 1182–1194 (2017)
38. Zhang, F., Safavi-Naini, R., Susilo, W.: An efficient signature scheme from bilinear pairings and its applications. In: International Workshop on Public Key Cryptography. pp. 277–290 (2004)
39. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the Weil pairing. J. Cryptol. **17**(4), 297–319 (2004)

Weiping Ye received the B.Sc. degree in Computer Science and Technology in 2013 from Huaqiao University, Xiamen, China. He is currently pursuing the M.Sc. degree in Computer Science from Huaqiao University, Xiamen, China. His interests are in the areas of cloud computing security and blockchain.

Jia Wang received the B.Sc. degree in 2016 from Nanjing University of Posts and Telecommunications, Nanjing, China. She received the M.Sc. degree in Computer Science from Huaqiao University in 2020, Xiamen, China. Her interests are in the areas of cloud computing security.

Hui Tian received the Ph.D. degree in 2010 in Computer Science from Huazhong University of Science and Technology, Wuhan, China. He is now a full professor in the College of Computer Science and Technology, Huaqiao University, Xiamen, China. His present research interests include network and information security, cloud computing security, and digital forensics. He has published more than 90 papers in refereed proceedings of conferences, journals and books, and got 21 patents. He is a senior member of IEEE, a member of ACM, a senior member of China Computer Federation (CCF), a member of the Technical Committee on Internet of CCF and a member of the Technical Committee on Information Storage of CCF.

Hanyu Quan received the Ph.D. degree and M.S. degree respectively in Information Security and Cryptography from Xidian University, Xi'an, China in 2019 and 2014. From 2015 to 2017, he was a visiting Ph.D. student at the Wireless Network and Cyber Security Research Lab in the Department of Electrical and Computer Engineering at the University of Arizona, USA. He is now a Lecturer of the College of Computer Science and Technology, Huaqiao University, Xiamen, China. His research interests include privacy-enhancing technologies and applied cryptography. He has published more than 10 papers in refereed proceedings of conferences and journals.