Hindawi
Security and Communication Networks
Volume 2022, Article ID 3368819, 13 pages
https://doi.org/10.1155/2022/3368819

Research Article

PPSEB: A Postquantum Public-Key Searchable Encryption Scheme on Blockchain for E-Healthcare Scenarios

Gang Xu,[1,2] Shiyuan Xu,[1] Yibo Cao,[1] Fan Yun,[1] Yu Cui,[1] Yiying Yu,[1] and Ke Xiao[1]

[1] *School of Information Science and Technology, North China University of Technology, Beijing 100144, China*
[2] *Advanced Cryptography and System Security Key Laboratory of Sichuan Province, Chengdu 610025, China*

Correspondence should be addressed to Ke Xiao; xiaoke@ncut.edu.cn

Received 17 November 2021; Revised 11 December 2021; Accepted 9 March 2022; Published 29 March 2022

Academic Editor: A. Peinado

In the current E-healthcare scenarios, medical institutions are used to encrypt the information and store it in an Electronic Health Record (EHR) system in order to ensure the privacy of medical information. To realize data sharing, a Public-key Encryption with Keyword Search (PEKS) scheme is indispensable, ensuring doctors search for medical information in the state of ciphertext. However, the traditional PEKS scheme cannot resist the keyword guessing quantum computing attacks, and its security depends on the confidentiality of the secret key. In addition, classical PEKS hand over the search process to a third party, affecting the search results' accuracy. Therefore, we proposed a postquantum Public-key Searchable Encryption scheme on Blockchain (PPSEB) for E-healthcare scenarios. Firstly, we utilized a lattice-based cryptographic primitive to ensure the security of the search process and achieve forward security to avoid key leakage of medical information. Secondly, we introduced blockchain technology to solve the problem of third-party untrustworthiness in the search process. Finally, through security analysis, we prove the correctness and forward security of the solution in the E-healthcare scenarios, and the comprehensive performance evaluation demonstrates the efficiency of our scheme compared with other existing schemes.

1. Introduction

In the current medical scenarios, medical institutions generate a large amount of patient medical data. These data are difficult to supervise, lack necessary technical support, and cost medical institutions many resources. To solve this problem, many medical institutions have adopted EHR systems to reduce the burden and cost of maintaining medical information [1]. The EHR system is a digital health file with medical information as the main body and information sharing as the core. It aims to realize that patients can manage their medical data, and doctors can also access the patient's medical data if they have permission. However, outsourcing management of the EHR system is not an ideal choice. Because the third-party organization responsible for storing the EHR system has too much power, once a malicious attacker buys it, it can launch a collision attack on the medical data in the system to threaten the privacy of medical data. To avoid this situation, medical institutions usually encrypt medical data through various encryption schemes [2] and store it in the EHR system. Therefore, how to realize the sharing of medical data between patients and doctors in the ciphertext state is a problem to be solved. Thus, Public Key Encryption with Keyword Search (PEKS) [3]is a marvelous candidate in cloud-assisted E-healthcare scenarios, realizing medical data retrieval without privacy leakage. As efficient encryption primitive, it ensures searchable encrypted medical data through keywords.

Although the existing proposed PEKS schemes [4–6] have brought significant benefits to the Internet of Things, there are four significant obstacles to the widespread PEKS in systems in recent decades. Initially, most PEKS schemes were established based on traditional hardness cryptography problems. Nevertheless, with the advent of quantum computers [7] and quantum information [8], the PEKS scheme will be threatened exponentially. Recent breakthrough articles [7] indicate that shortly, it is possible to adopt quantum computers in a realistic view, putting forward higher requirements for postquantum cryptographic searchable encryption schemes than before. Secondly, the most computational cost of cloud servers is to search target data from the third-party service agency since cloud servers need to execute a verification procedure for the corresponding keyword. Due to the exorbitant public-key encryption operations, the existing PEKS scheme introduces a significant calculation overhead. In the E-healthcare scenarios, the cloud server can work with medical data from mobile medical detection devices simultaneously to retrieve the data of multiple doctors. Therefore, it has a performance bottleneck on the medical cloud servers. Thirdly, with the explosive utilization of mobile medical detection equipment, most schemes have key exposure problems [9]. The existing PEKS scheme cannot guarantee the forward privacy of the key. The existing PEKS scheme cannot guarantee the forward privacy of the key. Once the doctor's secret key is compromised, the attacker can trace the trapdoor content previously submitted by the doctor, thereby further infringing on the confidentiality of the outsourced data [10]. In this regard, we optimize the lattice cryptography in our scheme to make the key have relations with period to ensure that the key exposure at the previous period will not affect the medical data confidentiality at the later period and achieve the forward security of the key [11]. Last but not least, the search function of the traditional PEKS scheme is generally delivered to the service party. However, the untrustworthiness of the service party will cause attackers to generate Keyword Guess Attacks (KGA) on medical information. Fortunately, blockchain can effectively solve this problem [12–17]. Blockchain is a new database technology that can realize decentralized distributed architecture design. Its core technical concept was proposed by Satoshi Nakamoto [18] in 2008. Blockchain, as a distributed public ledger, records all transactions packaged in the block without the need for third-party control and ensures the safety and traceability of each transaction record [19]. After a single block is generated, all nodes in the blockchain network use a consensus algorithm to determine whether the block is on the chain, and each block is connected by a hash function, thereby effectively ensuring the immutability of transaction information. Therefore, using blockchain technology to replace the service party in PEKS is an effective way to solve the problem of the untrustworthiness of the service party. For example, [20] replaces the traditional centralized server with a decentralized blockchain system, supports forward and backward privacy, and realizes privacy protection. [21] proposed a novel PEKS scheme, which eliminates the reliance on third-party institutions and makes the entire program completely decentralized. Therefore, to solve the above-mentioned hindrances, we propose a postquantum public-key searchable encryption on blockchain for cloud-assisted E-healthcare scenarios, called PPSEB, based on lattice cryptography [22, 23], one of the postquantum cryptographic primitives, ensuring a robust security level. In addition, we reduce the security of PPSEB to the Learning WithError (LWE) hardness assumption, which can oppose keyword guessing attacks based on quantum computing launched by malicious attackers effectively.

In our proposed scheme, the patient initially encrypts medical data and its keywords under the public key of the doctor and transmits the corresponding ciphertext to the cloud server for storage. Then, the medical doctor will utilize his/her secret key to compute a trapdoor corresponding to the keyword and then uploads it to the blockchain. Further, the smart contracts on blockchain search for the keyword ciphertext corresponding to the trapdoor and return its number to the cloud server. Finally, the cloud server sends the ciphertext of medical information matching the keyword to the doctor. In summary, we elaborate our main contributions as follows:

(1) We propose a postquantum Public-key Searchable Encryption on Blockchain (PPSEB) for the E-healthcare scenarios. PPSEB is constructed on lattice-based public-key searchable encryption based on the LWE hardness assumption.

(2) We then introduce blockchain technology into our proposed scheme in response to the untrustworthiness of third parties during the search process. Therefore, we achieve the decentralization architecture of the PPSEB oracle and enhance the security level.

(3) PPSEB achieves forward security in order to solve the key leakage of various existing public-key searchable encryption algorithms.

(4) We give the computational proof of the correctness and forward security of PPSEB. Furthermore, the comprehensive implementation performance evaluation represents that our scheme is efficient in terms of testing time and computational cost compared with existing outperforming E-healthcare schemes and is suitable for medical scenarios.

The structure of our paper is organized as follows. In Section 2, we propose the design goals and security models of our scheme, considering three existing challenges for the proposed PPSEB scheme and the solution to make PPSEB work better in the medical scenarios. In Section 3, we propose our preliminaries of lattice and trapdoor. In Section 4, we present our PPSEB scheme and the main steps of our scheme, including, *PPSEB*.Initialization, *PPSEB*.KeyExt, *PPSEB*.Encrypt, *PPSEB*.PEKS, *PPSEB*.Trapdoor, *PPSEB*.Verification, and *PPSEB*.Decrypt. In Section 5, we provide the security analysis of PPSEB based on correctness and provable security. In Section 6, a precise performance evaluation is proposed by our paper. Finally, we conclude this paper in Section 7.

2. Design Goals and Security Models

2.1. Design Goals. In this paper, we propose three existing challenges for the proposed PPSEB scheme:

(1) How to make PPSEB resistant to the untrustworthy problem of the service party. In the traditional

searchable encryption scheme, a third-party organization is generally responsible for searching medical information, which makes malicious attackers collude with third-party organizations to provide unreliable search results. Therefore, we use blockchain to replace traditional third-party agencies.

(2) How to achieve the forward security of PPSEB. Key exposure is a thorny problem faced by existing searchable encryption schemes. Once the private key of the doctor is lost, the attacker can forge the doctor to initiate an inquiry for medical information, and the privacy of medical information cannot be guaranteed. Therefore, how to use lattice-based cryptography to ensure that the leakage of the master key used at this time will not result in the leakage of the past session key is a problem to be solved.

(3) How to realize PPSEB to resist KGA under quantum computing. The existing searchable encryption scheme cannot guarantee the security of the search process under the attack of quantum computing, and there is a significant commonality between the keywords of medical information. Once the attacker is equipped with a quantum computer, it is possible to launch KGA on medical information through quantum computing, which severely threatens the blockchain system based on traditional cryptography and then exposes the private information contained in the medical information. Consequently, resisting KGA launched by quantum opponents is also a challenging problem. In order to make PPSEB work better in the medical scenarios, the solution in this article should have the following characteristics:

(1) Postquantum KGA: PPSEB can resist KGA attacks under quantum computing.

(2) Forward security: PPSEB achieves forward security to solve the problem of private key exposure.

(3) Efficiency: PPSEB has a higher computational efficiency by reducing the size of the trapdoor.

2.2. Security Model. In this section, we show the ciphertext indistinguishability of our scheme. We can describe several scenarios through games between challenger S and adversary A, in which S generates system security public parameters, initializes the public keys of patient and doctor. A will receive them from S and is permitted to access the oracles as below.

Hash Oracle(HO): A has been permitted to access all values of HO in time t, where $t = 1, 2, \ldots, \eta$ and is the total number in the period. Then, A will receive the corresponding hash value.

Break-in phase: After obtaining the query about $SK_{r\|t}$ of the doctor in time t by A, S will return the corresponding $SK_{r\|t}$ in t time to A. We note that t^* is the break-in period, which satisfies $t > t^*$.

Trapdoor Oracle(TO): A inputs a keyword w to ask S for a trapdoor T_w. Then, we make the restriction $t > t^*$ in order to make sure the forward security, where t^* is break-in period.

Challenge phase: A takes (w_0^*, w_1^*) in t^* and then submits them to S to be the challenge keywords. S then selects b at random and obtains $CT_{t^*}^*$. Consequently, S returns $CT_{t^*}^*$ to A.

Guess phase: At last, A will output $b' \in \{0, 1\}$. It wins the game iff $b' = b$.We define $A\ dv_A^S(k) = |\mathrm{Pr}ob[b' = b] - 1/2|$, which means the benefit of A to distinguish ciphertexts in t^* successfully.

3. Preliminary

Definition 1 (Lattice). Let $A = [a_1, a_2, \ldots, a_n] \in \mathbb{R}^m$ be n linearly independent vectors in m-dimensional space. A lattice L is composed of the linear combination of all integer coefficients of $a_1, a_2, \ldots, a_n$, and we can define: $L(A) = \{\sum_{i=1}^n x_i a_i\colon i = 1, 2, \ldots, n, x_i \in \mathbb{Z}\}$, $a_1, a_2, \ldots, a_n$ is known as a basis of L. Given a prime number q, a matrix $A \in \mathbb{Z}_q^{n\times m}$, we define $L_q(A) = \{y \in \mathbb{Z}^m\colon y = A^T x \bmod q, x \in \mathbb{Z}\}$, $L_q^{\perp}(A) = \{y \in \mathbb{Z}^m\colon Ay = 0 \bmod q\}$.

Definition 2 (LWE). Assume q be a prime number, given a random matrix $A \in \mathbb{Z}_q^{n\times m}$, vector $b \in \mathbb{Z}_q^m$ and the error distribution D on $\mathbb{Z}_q$, find that the vector $s \in \mathbb{Z}_q^n$ satisfies $b = A^T s + e \bmod q$, where $e \in D^m$.

Definition 3 (Statistical Distance). Given two variables X, Y over a domain D, we define the statistical distance of X and Y: $D(X, Y) = 1/2 \cdot \sum_{b\in D} |\Pr[X = a] - \Pr[Y = a]|$.

Definition 4 (Discrete Gaussian Distribution). Let $\rho_{c,\sigma}(x) = \exp - \pi\|x - c\|^2/\sigma^2$ be the standard

The Gaussian function c represents the center and σ represents the standard deviation. Then we define: $D_{L,c,\sigma}(x) = \rho_{c,\sigma}(x)/\rho_{c,\sigma}(L)$, which is a Gaussian Distribution over Lattice L.

Lemma 1 (TrapGen) [24]. *Let $q \geq 3$, $m \geq 2n \log q$. There is a polynomial-time algorithm TrapGen, which outputs a matrix $A \in \mathbb{Z}_q^{n\times m}$ statistically close to the uniform distribution and a trapdoor base $Tr_A \in \mathbb{Z}^{m\times m}$, such that $\|Tr_A\| \leq O(n \log q)$ and $\|\widetilde{Tr_A}\| \leq O(\sqrt{n \log q})$.*

Lemma 2 (SamplePre) [25]. *Given $L_q^{\perp}(A)$, a trapdoor base $Tr_A \in \mathbb{Z}^{m\times m}$, a parameter $s \geq \|\widetilde{Tr_A}\|\omega(\sqrt{\log m})$, and a vector $v \in \mathbb{Z}_q^n$. Then, the SamplePrealgorithm outputs a vector w statistically close to $D_{L_q^{\perp}(A),s}$, such that $Aw = v \bmod q$.*

Lemma 3 (SampleL) [26]. *Set a positive integer $m > n$, $q \geq 3$. Given $L_q^{\perp}(A)$ and its trapdoor base T_A, matrix $B \in \mathbb{Z}_q^{n\times m'}$, parameter $s \geq \|\widetilde{T_A}\|\omega(\sqrt{\log(m + m')})$, and vector $u \in \mathbb{Z}_q^n$. The Sample Lalgorithm computes $e \in \mathbb{Z}^{m+m'}$ statistically close to $D_{L_q^u(A|B),s}$ such that $(A|B)e = u \bmod q$.*

Lemma 4 (SampleR) [26]. *Set a positive integer $m > n$, $q \geq 3$. Given $L_{q,}^{\perp}(B)$ and its trapdoor base $T_B \in \mathbb{Z}^{m\times m}$, matrix $A \in \mathbb{Z}_q^{n\times m'}$, $R \in \mathbb{Z}_q^{m'\times m}$, $s \geq \|\widetilde{T_B}\| s' \omega(\sqrt{\log m})$ and vector $u \in \mathbb{Z}_q^n$. The SampleRalgorithm outputs a vector $e \in \mathbb{Z}^{m+m'}$ over $D_{L_q^u(A|AR+B),s}$ and satisfies $(A|AR+B)e = u \bmod q$, where $s' = \max_{\|x\|=1}\|Rx\|$.*

Lemma 5 (NewBasisDel) [27]. *Set a positive integer $m > 2n\log q$, $q \geq 3$. Given $L_q^{\perp}(A)$ and a trapdoor base $T_A \in \mathbb{Z}^{m\times m}$, an invertible matrix $R \in D_{m\times m}$, $D_{m\times m}$ is invertible on $\mathbb{Z}_q^{m\times m}$, $s \geq \|\widetilde{T_A}\| \cdot \sqrt{n\log q} \cdot \omega(\sqrt{\log m}) \cdot \sqrt{m} \cdot \omega$ $(\log^{1.5} m)$. The NewBasisDel algorithm outputs $L_q^{\perp}(B)$ and a trapdoor base $T_B \in \mathbb{Z}^{m\times m}$ responding to $L_q^{\perp}(B)$, where $B = AR^{-1}$.*

Lemma 6 (SampleRwithBasis) [27]. *Given a positive integer $m > 2n\log q$, $q \geq 3$, and a random matrix $A \in \mathbb{Z}_q^{n\times m}$, its column vector can generate $\mathbb{Z}_q^n$. The Sample R with Basis algorithm outputs an invertible matrix $R \in D_{m\times m}$, a lattice $L_q^{\perp}(B)$ and its trapdoor base $T_B \in \mathbb{Z}^{m\times m}$, where $B = AR^{-1} \bmod q$, T_B satisfies $\|\widetilde{T_B}\| \leq O(\sqrt{n\log q})$.*

Definition 5. (PEKS scheme): One general PEKS scheme includes five algorithms as *PEKS* = (Initialization, KeyExt, PEKS, Trapdoor, Verification), these algorithms are defined in the following sentences:

$(s, X) \longleftarrow$ Initialization$(\perp)$: In this step, it generally initializes some security parameters s, and parameters regard to the Gaussian Distribution X in one time period j. The output is just these parameters which will utilize in the next step.

$(pk, sk) \longleftarrow$ KeyExt(s): After inputting the parameter s, it will output the public key pk and secret key sk, which consist (pk, sk).

$s_\varepsilon \longleftarrow PEKS(pk, \varepsilon)$:The algorithm takes a public key pk and one keyword ε as input, and outputs a ciphertext s_ε of ε.

$t_\varepsilon \longleftarrow$ Trapdoor(sk, s_ε): Having input the secret key sk and one keyword ε, it outputs one trapdoor t_ε in this algorithm.

$(1 \text{ or } 0) \longleftarrow$ Verification$(t_\varepsilon, s_\varepsilon)$: With the input of a trapdoor $t_{\varepsilon'}$ and a searchable ciphertext s_ε, this algorithm designs to output the comparison decision bit 1 if $\varepsilon\prime = \varepsilon$, or 0 otherwise.

4. Our Proposed Scheme

4.1. Blockchain Architecture. Blockchain is essentially a decentralized database, which is a string of blocks that are associated using cryptography methods. Each transaction includes hash function, Merkle tree, and so on. In this paper, we replace the search party in searchable encryption with blockchain to ensure the credibility of the search process. As shown in Figure 1, our paper optimizes and adjusts the five-layer architecture of the original blockchain and adds a data retrieval function to the application layer to ensure that the blockchain network can base on the algorithm written in the smart contract realizing the retrieval of the keyword ciphertext.

4.2. System Model. In this section, we give an introduction to the system model of our PPSEB scheme in Figure 2, with four main entities, including patient, doctor, a cloud server, and blockchain network.

(1) Patient: The patient integrates Electronic Health Record (EHR), including various medical information such as drug-using records as a patient. Moreover, the patient encrypts the EHR and uploads it to the Cloud Server. Then the patient generates a set of keywords $\{\text{keywords, sequence number}\}$ related to the specified keyword and adds blocks to the blockchain.

(2) Doctor: The doctor needs to generate a trapdoor to search for information about patients. The doctor submits the corresponding trapdoor to the blockchain.

(3) Blockchain: After receiving the trapdoor from the doctor, the blockchain network will start chain code retrieval to search the corresponding sequence number and submit it to the CloudServer.

(4) Cloud Server: After receiving the query request, the Cloud Server can use trapdoor to search for all encrypted data and return the query results of the ciphertext corresponding to the keywords to the doctor. During the entire process, the server is unable to obtain any information about the data and keywords.

4.3. The Scheme of PPSEB. In this section, we present our proposed scheme in detail. There are mainly seven steps of our scheme, including *PPSEB*.Initialization, *PPSEB*.KeyExt, *PPSEB*.Encrypt, *PPSEB.PEKS*, *PPSEB*.Trapdoor, *PPSEB*.Verification, and *PPSEB*.Decrypt, which are elaborated in the following paragraphs and algorithms.

$(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s) \longleftarrow PPSEB$.Initialization (k, X, δ, σ): Firstly, we have to input one security parameter k, the discrete Gaussian Distribution X and its parameters $\delta = (\delta_1, \delta_2, \ldots, \delta_\eta)$, $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_\eta)$ in one period j, where $j = 1, 2, \ldots, \eta$. After that, the initialization step is shown as follows.

$(SK_{r\|j}, pk_{r\|j}) \longleftarrow PPSEB$.KeyExt $((X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s), j, sk_{r\|i}, i)$: After inputting the set Algorithm 1.

$(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$ obtained from the Initialization step, we also have to input the current period j together with the secret key $sk_{r\|i}$ in the previous period i. Then, the doctor will procedure the following operations, which shows in Algorithm 2.

$(N, W, I_M) \longleftarrow PPSEB$.Encrypt$(M, pk_{r\|j})$: Firstly, the patient divides the medical data M into groups, named $M = (M_1, M_2, \ldots, M_n)$, and generates an index $N = (1, 2, \ldots, n)$ for each group. After that, the patient extracts keywords from each group of medical data and records them as $W = (w_1, w_2, \ldots, w_n)$. Finally, the patient

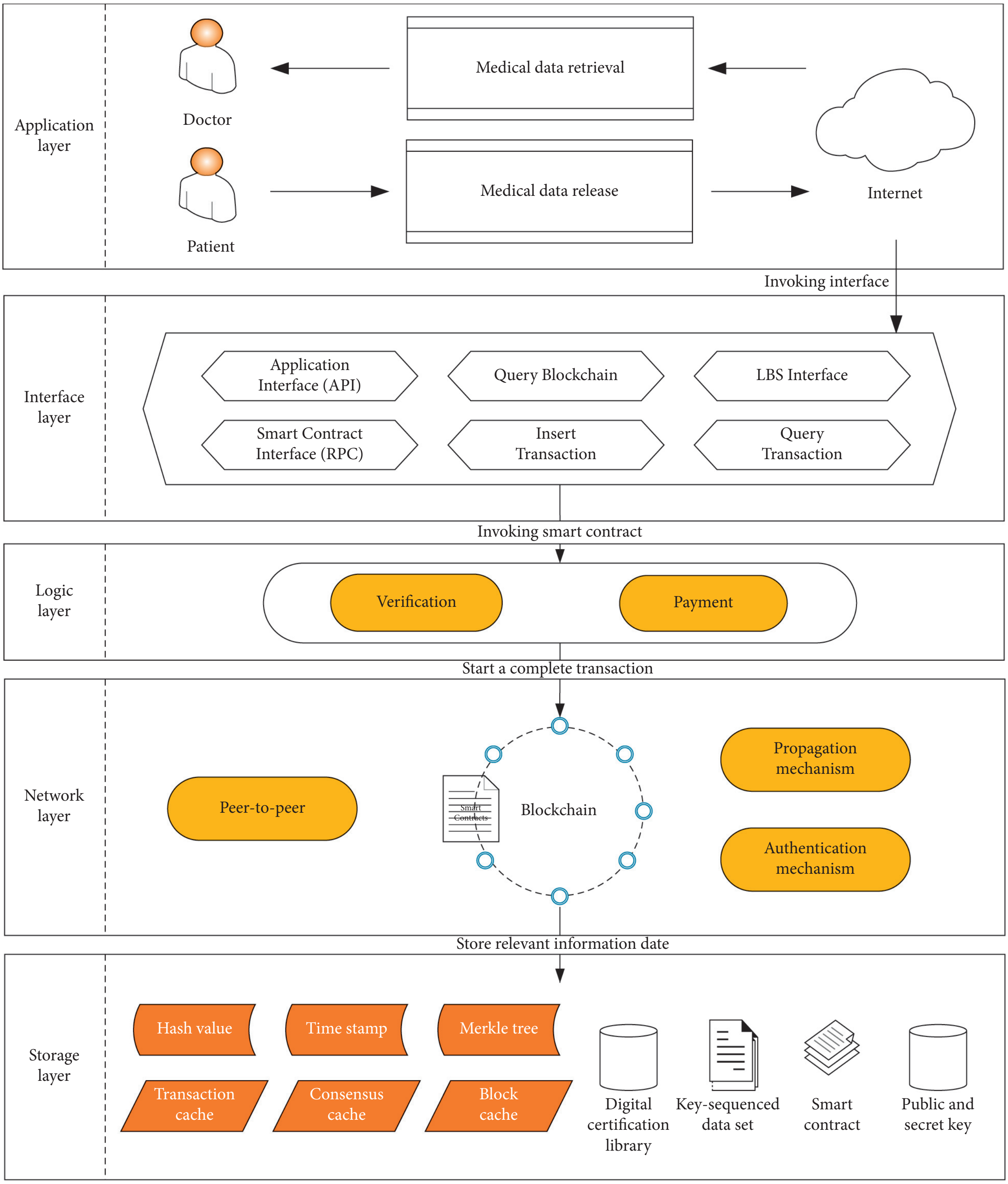

FIGURE 1: Blockchain architecture.

encrypts each group of medical data with the doctor's public key $pk_{r\|j}$ at time j, obtains a ciphertext set $CM = (CM_1, CM_2, \ldots, CM_n)$, and generates an index set of the medical data ciphertext $I_M = \{(1, CM_1), (2, CM_2), \ldots, (n, CM_n)\}$, and it will be stored in the cloud server.

$(CT_j) \longleftarrow PPSEB.PEKS((X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s), j, SK_{r\|j}, w)$: The patient will procedure *PPSEB.PEKS* algorithm and input the set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$, the public key $pk_{r\|j}$, the current time j, and keyword w. This Probabilistic Polynomial Time (PPT) algorithm shows in detail as below. For each keyword $w_i \in W$, the patient executes *PPSEB.PEKS* algorithm, obtains $CT_W = (CT_{j_1}, CT_{j_2}, \ldots, CT_{j_n})$, and pairs each keyword ciphertext with the number to generate keyword index set $I_W = \left\{(1, CT_{j_1}), (2, CT_{j_2}), \ldots, (n, CT_{j_n})\right\}$. When we get I_W, the patient calculates the hash value H_1 of I with his own

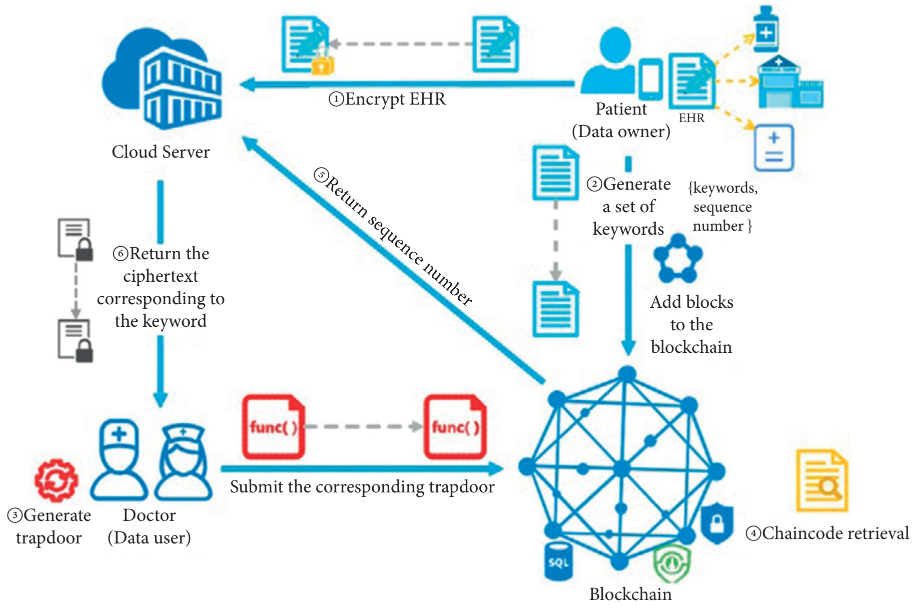

Figure 2: System architecture.

```
Input: security parameter k, discrete Gaussian Distribution $X$, security Gaussian Distribution $\delta$, $\sigma$.
Output: The set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$
(1) Select one uniform vector randomly $\mu \longleftarrow \mathbb{Z}_q^n$
(2) Assume that $N = \{0, 1, \ldots, \eta\}$ and compute $\mathbb{Z}_q^{n\times m} \times N \longrightarrow \mathbb{Z}_q^{m\times m}$ and $\{0,1\}^{l_1} \times N \longrightarrow \mathbb{Z}_q^{m\times m}$
(3) Set these two hash functions: $H_1$: $\mathbb{Z}_q^{n\times m} \times N$ and $H_2$: $\{0,1\}^{l_1} \times N$
(4) Call TrapGen(q,n) algorithm to generate $pk_s \in \mathbb{Z}_q^{n\times m}$ and $sk_s \in \mathbb{Z}_q^{m\times m}$, where $pk_s$ and $sk_s$ are public key and secret key of patient,
    respectively
(5) Call TrapGen(q,n) algorithm to generate $pk_r \in \mathbb{Z}_q^{n\times m}$ and $sk_r \in \mathbb{Z}_q^{m\times m}$, where $pk_r$ and $sk_r$ are public key and secret key of doctor,
    respectively
(6) Return the set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$
```

Algorithm 1: $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s) \longleftarrow PPSEB.$Initialization$(k, X, \delta, \sigma)$.

```
Input: set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$, current time period $j$, secret key $sk_{r\|i}$ in previous time period $i$
Output: $SK_{r\|j}$ and $pk_{r\|j}$, where is the secret key during this period $j$
 (1) Compute $H_1(pk_r\|i)H_1(pk_r\|i-1)\ldots H_1(pk_r\|1) \in \mathbb{Z}_q^{m\times m}$
 (2) Set $R_{r\|j} = H_1(pk_r\|i)H_1(pk_r\|i-1)\ldots H_1(pk_r\|1)$
 (3) Compute $pk_r(R_{r\|j})^{-1} = pk_r(H_1(pk_r\|i)H_1(pk_r\|i-1)\ldots H_1(pk_r\|1))^{-1} \in \mathbb{Z}_q^{n\times m}$
 (4) Set $pk_{r\|i} = pk_r(R_{r\|i})^{-1} = pk_r(H_1(pk_r\|i)H_1(pk_r\|i-1)\ldots H_1(pk_r\|1))^{-1}$
 (5) Compute $H_1(pk_r\|j)H_1(pk_r\|j-1)\ldots H_1(pk_r\|i+1) \in \mathbb{Z}_q^{m\times m}$
 (6) Set $R_{r\|i\longrightarrow j} = H_1(pk_r\|j)H_1(pk_r\|j-1)\ldots H_1(pk_r\|i+1)$
 (7) Call NewBasisDel $(pk_{r\|i}, R_{r\|i\longrightarrow j}, sk_{r\|j}, \delta_j)$ to compute $SK_{r\|j} \longleftarrow sk_{r\|j}$, where $SK_{r||j}$ is the secret key during this period $j$
 (8) Compute $pk_{r\|i}(R_{r\|i\longrightarrow j})^{-1} = pk_r(R_{r\|j})^{-1} \in \mathbb{Z}_q^{n\times m}$
 (9) Set $pk_{r\|j} = pk_r(R_{r\|j})^{-1}$
(10) Return $SK_{r\|j}$ and $pk_{r\|j}$
```

Algorithm 2: $(SK_{r\|j}, pk_{r\|j}) \longleftarrow PPSEB.KeyExt((X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s), j, sk_{r\|i}, i)$.

private key to generate a digital signature, writes down the transaction $I\ D$ and timestamp, generates the corresponding transaction, and submits it to the master node for verification. After that, all nodes of the blockchain network execute the consensus algorithm, and the master node jointly packs the transaction orders in a period of time to form a block and then sends it to the affiliate node. Then, the affiliate node receives the block sent by the master node and verifies the transaction slip contained in the block. Firstly, the affiliate node extracts the public key of the patient stored in the transaction sheet from the node and decrypts the digital signature and get the hash value H_2 of I_W. If $H_1 = H_2$, the affiliate node declares that the verification is successful. Otherwise, it means that the data may be tampered with and return this transaction to the patient. Assuming that the maximum number of malicious nodes that can exist in the consensus algorithm is f, if the number of verifications passes $Num = f + 1$, the block will be stored in each node of the blockchain network Algorithm 3.

$\text{Trap}_{w\|j} \longleftarrow$ *PPSEB*.Trapdoor $((X, \delta,\ \sigma, \mu, H_1, H_2,\ sk_r, sk_s), (pk_{r\|j}, sk_{r\|j}), j, w)$: The doctor will procedure this algorithm after inputting the set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$, the public key and secret key pair $(pk_{r\|j}, sk_{r\|j})$ of the medical doctor during this period j , and one keyword $w \in W$. The detailed description is shown in Algorithm 4.

Finally, the doctor will send $Trap_{w\|j}$ to the blockchain through an efficient and secure communication channel.

N_0 *or* False $\longleftarrow$ *PPSEB*.Verification $((X, \delta, \sigma,\ \mu, H_1, H_2, sk_r, sk_s), CT_j, t_{w\|j})$: This PPT algorithm produced by the blockchain inputs including the set $(X, \delta, \sigma,\ \mu,\ H_1,\ H_2, sk_r, sk_s)$, the ciphertext CT_j, one trapdoor $Trap_{w\|j}$ in this period j of the doctor. If it outputs true; it means that the trapdoor $Trap_{w\|j}$ and the ciphertext CT_j contain the uniform keyword w. Then, the blockchain returns the number N_0 of the ciphertext corresponding to the keyword to the cloud server. The cloud server finds the ciphertext of the keyword according to N_0 and returns it to the doctor Algorithm 5.

$M_0 \longleftarrow$ *PPSEB*.Decrypt $(CM_0, j, SK_{r\|j})$: After the doctor obtains the ciphertext CM_0 of the medical data returned by the cloud server, he/she decrypts it with his $SK_{r\|j}$ at time j to obtain the plaintext of medical data M_0.

5. Security Analysis

In this section, we will demonstrate our scheme's correctness and provable security to achieve the security of the keyword ciphertext in our scheme under random oracle.

5.1. Correctness

5.1. Correctness. In this section, we suppose that the key pair at time j of doctors and patients are $(pk_{r\|j}, sk_{r\|j})$, $(pk_{s\|j}, sk_{s\|j})$, respectively. Then, we set w as the keyword of the ciphertext CT_j and then w' is a keyword that matches the trapdoor $Trap_{w'\|j}$. It is well known that the cloud server can use $Trap_{w'\|j}$ at a time j to recover $(y_{j1}', y_{j2}', \ldots, y_{jl}') = CT_{j1} - Trap_{w'\|j}^T CT_{j2}$ in *PPSEB*.Verification. Since the relationship between w and w' is uncertain, we divide the discussion into the following two situations:

Case 1: If $w \neq w'$, then $CT_{j1} - Trap_{w'\|j}^T CT_{j2} \neq CT_{j1} - Trap_{w\|j}^T CT_{j2}$, so we can decrypt the ciphertext CT_j and obtain that: for $i = 1, 2, \ldots, l$, there must be $y_{ji} \neq 1$.

Case 2: If $w = w'$, then there is $CT_{j1} - Trap_{w'\|j}^T CT_{j2} = CT_{j1} - Trap_{w\|j}^T CT_{j2} = noi_j + (y_{j1}, y_{j2}, \ldots, y_{jl})\lfloor q/2 \rfloor - Trap_{w\|j}^T CT_{j2}$. Among them, $noi_j - Trap_{w\|j}^T CT_{j2}$ is a noise vector. According to [25], we need to ensure that the error vector is less than $q/5$, so that the decryption process does not make mistakes. Consequently, we can compute that: for $i = 1, 2, \ldots, l$, $y_{ji}' = 1$.

So, the cloud server can ensure that the keyword w can correspond to the ciphertext $CT_j = (CT_{j1}, CT_{j2})$ and the trapdoor $Trap_{w'\|j}^T$; that is, PPSEB can achieve correctness. Last but not least, the cloud server sends the encrypted medical data corresponding to the keyword w to the doctor, and the doctor obtains the corresponding plaintext data after decrypting it according to its key.

5.2. Provable Security

Theorem 1. *In the PPSEB, the difficulty of the attacker to crack the indistinguishability of the ciphertext can be reduced to the difficulty of the LWE problem.*

Proof. Suppose that there is an attacker A under the random oracle model, which can crack the indistinguishability of the ciphertext in polynomial time. On this basis, we have created a challenger C having the ability to solve the LWE problem. □

5.2.1. Setup. To begin with, challenger C sends $(u_k, v_{k1}, v_{k2}, \ldots, v_{kl})$, $k = 0, 1, \ldots, m$ from a random oracle machine. Then, C guesses $\tau = j^*$ as a point in time when A breaks the indistinguishability of the ciphertext. After that, C creates two lists, named L_1 and L_2. Finally, C interacts with attacker A. The steps are as follows:

(1) Challenger C runs the SampleR algorithm to obtain R, then C selects $\tau + 1$ vectors from $R^*, R_1^*, \ldots, R_\tau^*$ and assembles it into a matrix $F^* \in \mathbb{Z}_q^{n\times m}$, making u_k the k-th column of F^*.

(2) Challenger C obtains $pk_r = F^* R^* R_1^* \cdots R_\tau^*$. Because F^* is independent of $\mathbb{Z}_q^{n\times m}$ and $R_1^*, R_2^*, \cdots, R_\tau^*$ are irreversible matrices, pk_r is independent of $\mathbb{Z}_q^{n\times m}$. Then, C selects a matrix as $pk_s \in \mathbb{Z}_q^{n\times m}$ and sets $\mu = u_0 \in \mathbb{Z}_q^n$ to get a set $(pk_r, pk_s, \mu, H_1, H_2)$. Last but not least, C sends $(pk_r, pk_s, \mu, H_1, H_2)$ to attacker A.After receiving the set $(pk_r, pk_s, \mu, H_1, H_2)$, A executives H_1 query and H_2 query.

H_1 query: A initiates an inquiry to each $pk_r\|j$, where $j = 1, 2, \ldots, \tau$. C computes $R_j^* = H_1(pk_r\|j)$ and sends R_j^* to A.

Case 1: $j = \tau + 1$. Challenger C gets $pk_{r\|j-1} = pk_r \cdot (R^* R_1^* \cdots R_\tau^*)^{-1}$ and runs Sample R with Basis algorithm to get R_j and the basis $sk_{r\|j}$ of lattice $L_q^\perp(A_{r\|j})$, where $A_{r\|j} = R_j^{-1} \cdot A_{r\|j-1}$. Then, C appends $(pk_r\|j, pk_{r\|j}, R_j, sk_{r\|j})$ to the list L_1. Consequently, C transmits R_j to attacker A.

Case 2: $j > \tau + 1$. Challenger C finds $(pk_r\|j-1, pk_{r\|j-1}, R_{j-1}, sk_{r\|j-1})$ from the L_1. Then, C selects a

```
Input: set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$, current time period $j$, secret key $SK_{r\|j}$ in current period $j$
Output: $CT_j$
Set a binary string $B_j \longleftarrow \mathbb{Z}_q^{n\times l}$, where $l$ is the security level of test in medical data cloud storage
Select a unitive matrix $B_j \longleftarrow \mathbb{Z}_q^{n\times l}$ of $(n \times l)$ dimension
Select noise $noi_{j1}, noi_{j2}, \dots, noi_{jl} \longleftarrow \mathbb{Z}_q$ through $X$
Set $noi_j = (noi_{j1}, noi_{j2}, \dots, noi_{jl})$
Select each noise vector $noiv_{j1}, noiv_{j2}, \dots, noiv_{jl} \longleftarrow \mathbb{Z}_q$ on the basis of $X^m$
Set the noise vector matrix $noiv_j = (noiv_{j1}, noiv_{j2}, \dots, noiv_{jl}) \in \mathbb{Z}_q^{m\times l}$
Assume $\beta_j = H_2(w\|j)$ and then compute $CT_{j1} = \mu^T B_j + noi_j + y_j \lfloor q/2 \rfloor$ and $CT_{j2} = (pk_{r\|j}\beta_j^{-1})^T B_j + noiv_j$ as ciphertext
Set ciphertext $CT_j = (CT_{j1}, CT_{j2}) = (\mu^T B_j + noi_j + y_j \lfloor q/2 \rfloor, (pk_{r||j}\beta_j^{-1})^T B_j + noiv_j)$
Return $CT_j$ to doctor
```

Algorithm 3: $(CT_j) \longleftarrow PPSEB.PEKS((X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s), j, SK_{r||j}, w)$.

```
Input: set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$, current period $j$, public-secret key pair $(pk_{r\|j}, sk_{r\|j})$, one keyword $w$
Output: $sk_{w\|j}$ and $Trap_{w\|j}$
Compute $\beta_j = H_2(w\|j)$
Set $R_{r\|j} = H_1(pk_r\|i) H_1(pk_r\|i-1) \dots H_1(pk_r\|1)$
Call NewBasisDel $(pk_{r\|j}, \beta_j, sk_{r\|j}, \delta_j)$ to generate one short lattice basis $sk_{w\|j} \in \mathbb{Z}_q^{m\times m}$ in random
Call SamplePre $(pk_{r\|j}\beta_j^{-1}, sk_{w\|j}, \mu, \sigma_j)$ to generate the trapdoor $Trap_{w\|j} \in \mathbb{Z}_q^m$
Return $Trap_{w\|j}$
```

Algorithm 4: $Trap_{w\|j} \longleftarrow PPSEB.$Trapdoor$((X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s), (pk_{r\|j}, sk_{r\|j}), j, w)$.

```
Input: set $(X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s)$, ciphertext $CT_j$, current period $j$, trapdoor $Trap_{w\|j}$
Output: $N_0$ or False
Compute $(y_{j1}, y_{j2}, \dots, y_{jl}) = CT_{j1} - Trap_{w\|j}^T CT_{j2}$
Set $y_j = (y_{j1}, y_{j2}, \dots, y_{jl})$
Select integer $q$ satisfies $\{1, 2, \dots, q\} \subset \mathbb{Z}^+$
for $(i = 1, 2, \dots, l)$ do
    if $|y_{ji} - \lfloor q/2 \rfloor| \geq \lfloor q/4 \rfloor$ then
        The medical cloud sever will abort it and Return False.
    else
        Set $y_{ji} = 1$ up to $y_{jl} = 1$
end if
endfor
if $y_j = (1, 1, \dots, 1) \in \{1\}^l$ then
        Return $N_0$
else
        Return False
end if
```

Algorithm 5: *True or False* $\longleftarrow PPSEB.$Verification$((X, \delta, \sigma, \mu, H_1, H_2, sk_r, sk_s), CT_j, t_{w\|j})$.

matrix R_j, and carries out the New Basis Del algorithm to compute $sk_{r||j}$ as the basis of $L_q^{\perp}(pk_{r\|j})$, where $pk_{r\|j} = pk_{r\|j-1} \cdot R_j^{-1}$. Consequently, C appends $(pk_r\|j, pk_{r\|j}, R_j, sk_{r\|j})$ to L_1, and transmits R_j to attacker A.

H_2 query: The attacker A queries w, at the same time challenger C performs the following operations:

Case 1: $w = w^*$ and $j = j^*$. The challenger C calculates $R^* = H_2(w\|j)$ and sends R^* to A.

Case 2: $w \neq w^*$ or $j \neq j^*$. The challenger C looks for $(pk_r\|j, pk_{r\|j}, R_j, sk_{r\|j})$ in L_1, selects a matrix $R_{w\|j}$, and executes the NewBasisDel algorithm to generate a basis $sk_{w\|j}$ of $L_q^{\perp}(pk_{r\|j} \cdot R_{w\|j}^{-1})$. Finally, C saves $(w\|j, pk_{r\|j} \cdot R_{w\|j}^{-1}, R_{w\|j}, sk_{w\|j})$ in L_2, and sends $R_{w\|j}$ to A.

5.2.2. Trapdoor Query. When C receives a query for a keyword w from A, C first looks at L_2, and if there is no $(w\|j, pk_{r\|j} \cdot R_{w\|j}^{-1}, R_{w\|j}, sk_{w\|j})$ in L_2; then this process will be restarted.

Otherwise, C gets the private key $sk_{w\|j}$, runs the SamplePre algorithm to generate a trapdoor $Trap_{w\|j}$, and sends it to A.

5.2.3. Break-In Phase. In this process, attacker A can query the private key of the doctor in the $j > j^*$ period, and $j^* = \tau$ is set a break-in time. After A queries H_1 on $pk_r\|j$, C sends the private key $sk_{r\|j}$ to A.

In time i, which is the prior period, we can find $(pk_r\|j, pk_{r\|i}, R_i, sk_{r\|i})$ from L_1 because the attacker A will perform H_1 queries on $pk_r\|i$. Further, we calculate $pk_{r\|i} = pk_{r\|\tau+1} = pk_r \cdot (R_\tau^* \cdots R_2^* R_1^*)^{-1} \cdot H_1 (pk_r\|\tau + 1)^{-1}$, which $sk_{r\|i}$ is the basis of the lattice $L_q^\perp (pk_{r\|i})$. After that, challenger C calculates $R_{r\|i\longrightarrow j} = H_1 (pk_r\|j) \cdots H_1 (pk_r\|i \longrightarrow 1)$ and runs the NewBasisDel algorithm to obtain $pk_{r\|j} = pk_{r\|i} \cdot R_{r\|i \longrightarrow j}^{-1}$ and $sk_{r\|j}$ in time j. Consequently, C sends $sk_{r\|j}$ to attacker A.

5.2.4. Challenge Phase. Assuming that w_0^* and w_1^* are two keywords, challenger C randomly selects a quantity from $\{0, 1\}$ and assigns it to b. Then we need to divide into the following cases according to the value of b.

Case 1: $b = 0$. The challenger C sends ciphertext $(CT_{\tau 1}^*, CT_{\tau 2}^*)$ of w_0^* to A.

Case 2: $b = 1$. We create $v_0 = (v_{01}, v_{02}, \ldots, v_{0l})$, $v^* = (v_1, v_2, \ldots, v_m)^T$, and $y_j^* = (1, 1, \ldots, 1)$. Then, $CT_{\tau 1}^* = v_0 + \lfloor q/2 \rfloor \cdot y_j^*$ and $CT_{\tau 2}^* = v^*$ can be obtained. Consequently, C sends the ciphertext $(CT_{\tau 1}^*, CT_{\tau 2}^*)$ of w_1^* to A.

5.2.5. Guess Phase. In this process, attacker A outputs $b' = 0$ or $b' = 1$ as the response of theChallenge phase.

Analysis: To begin with, according to the basic probability knowledge, the probability of C outputting the ciphertext of the keyword w_1 is 1/2.

Suppose that A can break the indistinguishability of the ciphertext with the probability p. In addition, the probability that challenger C can correctly obtain the break time is 1/m. Consequently, C can solve the LWE hardness with the probability of $p/2m$. In a nutshell, the difficulty of the attacker to crack the indistinguishability of the ciphertext can be reduced to the difficulty of the LWEhardness.

6. Performance Evaluation

In this section, to guarantee the forward security, anti-quantum KGA, and suitability in the medical scenarios of our PPSEB scheme, we analyze the computational expense, security property, and network communication costs of our scheme and compare our scheme with existing PEKS schemes [3, 5, 28, 29]on the actual performance in the medical background through experiments and numerical simulation technique. The experiments evaluating and testing the actual performance of our scheme are operated on a MacOS with an Intel Core i7 CPU and 16 GB RAM. The implementation of schemes is based on the C++ language, and we use medical data extremely close to actual applications of daily life to complete the experiments. Meanwhile, in order to realize the security of the q-arylattices, the parameters satisfy $m > 2n \log q$, $q \geq 3$, since the algorithms counting on lattice-based cryptography are relied on q, m, n. The notations of the following specific descriptions in the experiments are provided in Table 1. The accurate experimental data of 200 trials on average are shown in the following figures, and the results accord with our design objective extremely.

Our PPSEB is highly efficient compared with other PEKS schemes. As is illustrated in Table 2, the theoretical communication costs of each scheme are listed accurately.

We prove the theoretical value, and the experimental result reflects in Figure 3, demonstrating that the trapdoor size of the PPSEB scheme is the least one among the whole schemes. Along with the stabilizing growth in communication costs, our algorithm is superior to the others, indicating a hidden potential to reduce network resource consumption.

As to the actual performance, Figure 3 indicates that the PPSEB scheme reveals a considerable efficiency advantage. The PEKS size of PPSEB is relatively close to the scheme [3, 5, 28]and much less than the scheme [29]. The trapdoor size in our scheme is a quarter of [29]. However, in terms of postquantum, our proposed PPSEB is more secure than the scheme [3, 5, 28] while being applied in medical data encryption protection. Thus, it is pretty sound and acceptable for PPSEB to increase the nominal communication costs corresponding to PEKS size.

In addition, we not only analyze the computational expense and security property of our scheme but also compare it with existing PEKS schemes [3, 5, 28] through experimental medical data. As shown in Figure 4, the testing time of our scheme is also much shorter than the other existing PEKS schemes. Significantly, the more the number of retrieving keywords increases, the more apparent the superiority becomes.

Besides, we test the testing time and computational expense of the PEKS schemes and record the results in Table 3.

Our scheme realizes nearly the same as a scheme [3] in saving the computational expense and searching efficiency according to the comparison in Figure 5. When the number of retrieving keywords is 180, the testing time of [5] is 7.2s, and ours is 0.477s, which is 15.09 times that of PPSEB. As a result, our scheme is not only advantageous in terms of postquantum property, but also relatively efficient than the other schemes. Consequently, although the introduction of blockchain technology has brought a certain amount of complexity and extra overhead to our system, it is certified that our PPSEB scheme can realize the property of postquantum, forward security on maintaining the confidentiality of medical data and superiority in the applications of medical scenarios. From a more practical view, it is both convenient and swift for doctors to master the patient's physical condition, obtain the patient's medical records, and make the correct diagnosis promptly in practical medical scenarios. In

Table 1: Notations of descriptions.

Notations	Descriptions
$Time_{me}$	The modular exponentiation time
$Time_{sm}$	The scalar multiplication time
$Time_{hp}$	The hash-to-point time
$Time_{pa}$	The point addition time
$Time_{bp}$	The bilinear pairing time
$Time_{hf}$	The hash function time
$Time_{m}$	The multiplication time
S_1	One element bit size in G_1
S_T	One element bit size in G_T
S_p	One element bit size in $\mathbb{Z}_p$
S_q	One element bit size in $\mathbb{Z}_q$
S_l	The security level with a value of 10

Table 2: Communication costs.

Schemes	Size of PEKS algorithm	Size of trapdoor algorithm
Our scheme	$(S_l + mS_l)S_q$	mS_q
Boneh et al. [3]	$S_p + S_1$	S_1
Ma et al. [5]	$5S_1 + 3S_T$	$3S_1$
Ma et al. [28]	$S_p + S_1$	S_1
Shao et al. [29]	$S_l + S_1$	S_1

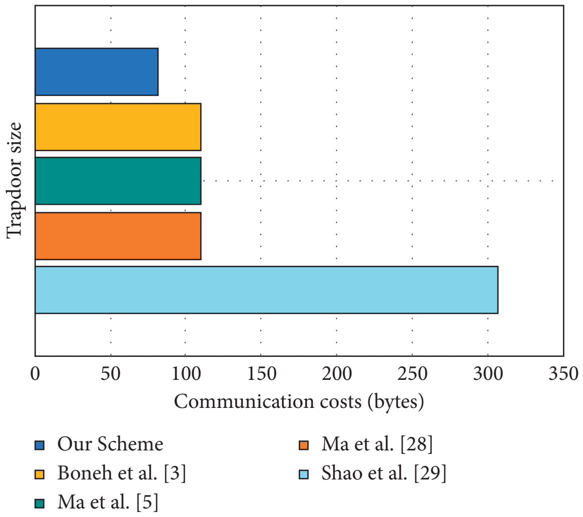

Figure 3: Communication costs comparison corresponding to trapdoor size.

addition, the more profound performance of PPSEB on managing medical data of Electronic Health Records systems, such as electronic medical record and electronic prescription, need to be tested experimentally and further study in development.

In Figure 6, we compared the PEKS computational expense of PPESB with [3, 5, 28, 29]. Among them, the PEKS computational expense of our scheme is much smaller than other schemes, which shows that our scheme has higher efficiency under the same number of retrieving keywords.

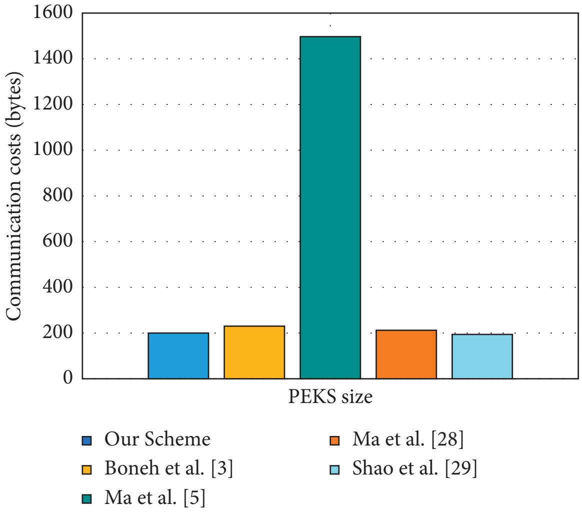

FIGURE 4: Communication costs comparison corresponding to PEKS size.

TABLE 3: Testing time and computational expense.

Schemes	Testing time	PEKS computational expense
Our scheme	$mS_l Time_m$	$(mnS_l + nS_l + m^2 n)Time_m + Time_{hf}$
Boneh et al. [3]	$Time_{hf} + Time_{bp}$	$2Time_{pa} + Time_{hp} + 4Time_{sm} + Time_{bp} + 3Time_{hf}$
Ma et al. [5]	$5Time_{me} + 4Time_{bp} + Time_{hf}$	$3Time_{hf} + 9Time_{me} + 3Time_{bp}$
Ma et al. [28]	$Time_{hf} + Time_{bp} + Time_{sm}$ $+2Time_{pa} + 2Time_{hp}$	$Time_{hf} + 3Time_{bp} + 2Time_m$ $+Time_{pa} + 4Time_{sm} + 3Time_{hp}$
Shao et al. [29]	$Time_{hf} + Time_{bp}$	$2Time_{hf} + 2Time_{me} + Time_{bp}$

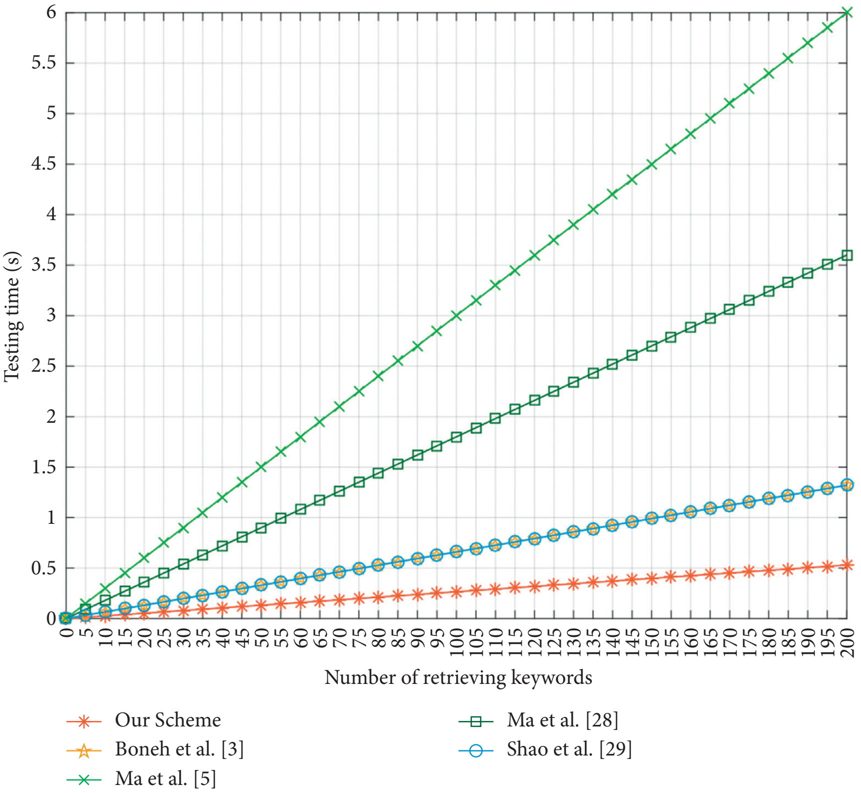

FIGURE 5: The testing time comparison.

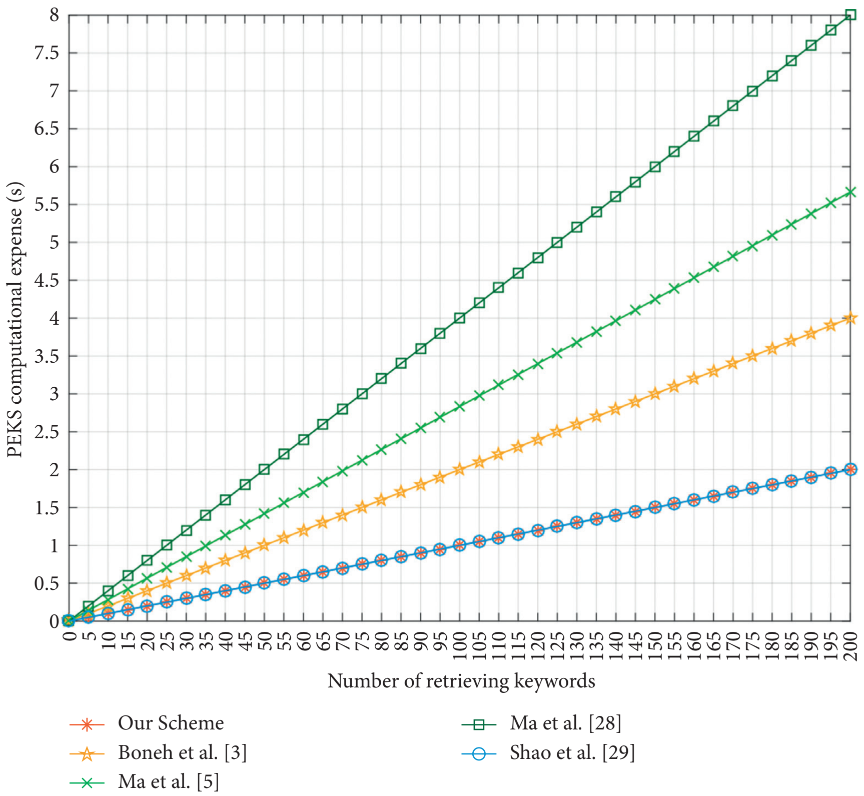

Figure 6: PEKS computational expense comparison.

7. Conclusion

In our paper, we proposed postquantum Public-key Searchable Encryption on Blockchain (PPSEB) for E-healthcare scenarios. PPSEB is capable of resisting keyword-guessing quantum computing attacks. Moreover, our proposed scheme combines public-key searchable encryption and blockchain, avoiding turning over the searching process to a third party and enhancing the security level. Furthermore, we assure forward security, maintaining the confidentiality of medical data. Both security analysis and comprehensive performance evaluation demonstrate that PPSEB can achieve the property of searching efficiency and lightweight of lower computational cost in retrieving keywords and generating trapdoor compared with other existing E-healthcare schemes.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Open Fund of Advanced Cryptography and System Security Key Laboratory of Sichuan Province (Grant No. SKLACSS-202101), NSFC (Grant nos. 62176273, 61962009, U1936216, and 62076042), the Foundation of Guizhou Provincial Key Laboratory of Public Big Data (Nos. 2019BDKFJJ010 and 2019BDKFJJ014), the Fundamental Research Funds for Beijing Municipal Commission of Education, Beijing Urban Governance Research Base of North China University of Technology, the Natural Science Foundation of Inner Mongolia (2021MS06006), Baotou Kundulun District Science and technology plan project (YF2020013), and Inner Mongolia discipline inspection and supervision big data laboratory open project fund (IMDBD2020020).

References

[1] I. M. Baytas, K. Lin, F. Wang, A. K. Jain, J. Zhou, and J. Zhou, "PhenoTree: interactive visual analytics for hierarchical phenotyping from large-scale electronic health records," *IEEE Transactions on Multimedia*, vol. 18, no. 11, pp. 2257–2270, 2016.

[2] Y. Chen, S. Dong, T. Li, Y. Wang, and H. Zhou, "Dynamic multi-key FHE in asymmetric key setting from LWE," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 5239–5249, 2021.

[3] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search,"vol. 3027, pp. 506–522, in *Proceedings of the 23th Annual International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT 2004)*, vol. 3027, pp. 506–522, Springer, Berlin, Heidelberg, May 2004.

[4] Q. Huang and H. Li, "An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks," *Information Sciences*, vol. 403-404, pp. 1–14, 2017.

[5] M. Ma, D. He, N. Kumar et al., "Certificateless searchable public key encryption scheme for industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 759–767, 2017.

[6] R. Chen, Y. Mu, G. Yang et al., "Dual-server public-key encryption with keyword search for secure cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 789–798, 2016.

[7] T. D. Ladd, F. Jelezko, R. Laflamme et al., "Quantum computers," *Nature*, vol. 464, no. 7285, pp. 45–53, 2012.

[8] A. Galindo and M. A Martín-Delgado, "Information and computation: classical and quantum aspects," *Reviews of Modern Physics*, vol. 74, no. 2, pp. 347–423, 2002.

[9] W. Li, X. Li, J. Gao, and H. Wang, "Design of secure authenticated key management protocol for cloud computing environments," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1276–1290, 2021.

[10] E. Uchiteleva, A. R. Hussein, A. Shami, and A. Shami, "Lightweight dynamic group rekeying for low-power wireless networks in IIoT," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4972–4986, 2020.

[11] Y. Cheng, S. Xu, M. Zang et al., "LPPA: a lightweight privacy-preserving authentication scheme for the Internet of drones," in *Proceedings of the 21st International Conference on Communication Technology(ICCT 2021)*, pp. 656–661, IEEE, Tianjin, China, October 2021.

[12] T. Li, Z. Wang, G. Yang, Y. Cui, Y. Chen, and X. Yu, "Semi-selfish mining based on hidden Markov decision process," *International Journal of Intelligent Systems*, vol. 36, no. 7, pp. 3596–3612, 2021.

[13] T. Li, Z. Wang, Y. Chen et al., "Is semi-selfish mining available without being detected?" *International Journal of Intelligent Systems, early access*, vol. 33, no. 1, 2021.

[14] Y. Chen, J. Sun, Y. Yang, T. Li, X. Niu, and H. Zhou, "PSSPR: a source location privacy protection scheme based on sector phantom routing in WSNs," *International Journal of Intelligent Systems*, vol. 37, no. 2, pp. 1204–1221, 2022.

[15] X. Liu, R. Zhang, G. Xu, X.-B. Chen, and N. N. Xiong, "Confidentially judging the relationship between an integer and an interval against malicious adversaries and its applications," *Computer Communications*, vol. 180, pp. 115–125, 2021.

[16] G. Xu, Y. Cao, S. Xu et al., "A novel post-quantum blind signature for log system in blockchain," *Computer Systems Science and Engineering*, vol. 41, no. 3, pp. 945–958, 2022.

[17] T. Li, Y. Chen, Y. Wang et al., "Rational protocols and attacks in blockchain system," *Security and Communication Networks*, vol. 2020, pp. 1–11, Article ID 8839047, 2020.

[18] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," 2008, https://bitcoin.org/bitcoin.pdf.

[19] S. Xu, X. Chen, and Y. He, "EVchain: an anonymous blockchain-based system for charging-connected electric vehicles," *Tsinghua Science and Technology*, vol. 26, no. 6, pp. 845–856, 2021.

[20] B. Chen, L. Wu, H. Wang, L. Zhou, and D. He, "A blockchain-based searchable public-key encryption with forward and backward privacy for cloud-assisted vehicular social networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5813–5825, 2020.

[21] C. Liu, Y. Xiao, V. Javangula, Q. Hu, S. Wang, and X. Cheng, "NormaChain: a blockchain-based normalized autonomous transaction settlement system for IoT-based E-commerce," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4680–4693, 2019.

[22] D. Micciancioand and O. Regev, "Lattice-based cryptography,"vol. 4117, pp. 131–141, in *Proceedings of the 26th Annual International Cryptology Conference (CRYPTO 2006)*, vol. 4117, pp. 131–141, Springer, Berlin, Heidelberg, May 2006.

[23] S. Xu, X. Chen, C. Wang et al., "A lattice-based ring signature scheme to secure automated valet parking,"vol. 12938, pp. 70–83, in *Proceedings of the International Conference on16th Wireless Algorithms, Systems, and Applications (WASA 2021)*, vol. 12938, pp. 70–83, Springer, Nanjing, China, September 2021.

[24] J. Alwen and C. Peikert, "Generating shorter bases for hard random lattices," *Theory of Computing Systems*, vol. 48, no. 3, pp. 535–553, 2011.

[25] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions,"vol. 14, pp. 197–206, in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC 2008)*, vol. 14, pp. 197–206, ACM, Victoria, British Columbia, May 2008.

[26] S. Agrawal, D. Boneh, and X. Boyen, "Efficient lattice (H)IBE in the standard model,"vol. 6110, pp. 553–572, in *Proceedings of the 29th Annual International Conference on the Theoryand Application of CryptographicTechniques (EUROCRYPT 2010*, vol. 6110, pp. 553–572, Springer, Riviera, France, June 2010.

[27] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE,"vol. 6223, pp. 98–115, in *Proceedings of the 30th Annual International Cryptology Conference (CRYPTO 2010)*, vol. 6223, pp. 98–115, Springer, Santa Barbara, CA, USA, May 2010.

[28] M. Ma, D. He, M. K. Khanand, and J. Chen, "Certificateless searchable public key encryption scheme for mobile healthcare system," *Computers & Electrical Engineering*, vol. 65, pp. 413–424, 2017.

[29] Z.-Y. Shao, B. Yang, and B. Yang, "On security against the server in designated tester public key encryption with keyword search," *Information Processing Letters*, vol. 115, no. 12, pp. 957–961, 2015.